



UNIVERSIDADE FEDERAL DO CEARÁ
CENTRO DE TECNOLOGIA
DEPARTAMENTO DE TELEINFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE TELEINFORMÁTICA
MESTRADO ACADÊMICO EM ENGENHARIA DE TELEINFORMÁTICA

POLYCARPO SOUZA NETO

**GENERALIZED CLOUD PARTITIONING ITERATIVE CLOSEST POINT: UMA
AVALIAÇÃO QUANTITATIVA DO REGISTRO DE NUVENS DE PONTOS 3D**

FORTALEZA

2019

POLYCARPO SOUZA NETO

GENERALIZED CLOUD PARTITIONING ITERATIVE CLOSEST POINT: UMA
AVALIAÇÃO QUANTITATIVA DO REGISTRO DE NUVENS DE PONTOS 3D

Dissertação apresentada ao Curso de Mestrado Acadêmico em Engenharia de Teleinformática do Programa de Pós-Graduação em Engenharia de Teleinformática do Centro de Tecnologia da Universidade Federal do Ceará, como requisito parcial à obtenção do título de mestre em Engenharia de Teleinformática. Área de Concentração: Sinais e Sistemas

Orientador: Prof. Dr. George André Pereira Thé

Coorientador: Prof. Dr. José Marques Soares

FORTALEZA

2019

Dados Internacionais de Catalogação na Publicação
Universidade Federal do Ceará
Biblioteca Universitária
Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

S234g Souza Neto, Polycarpo.
Generalized cloud partitioning iterative closest point: uma avaliação quantitativa do registro de nuvens de pontos 3D / Polycarpo Souza Neto. – 2019.
87 f. : il. color.

Dissertação (mestrado) – Universidade Federal do Ceará, Centro de Tecnologia, Programa de Pós-Graduação em Engenharia de Teleinformática, Fortaleza, 2019.

Orientação: Prof. Dr. George André Pereira Thé.

Coorientação: Prof. Dr. José Marques Soares.

1. Visão computacional. 2. Iterative Closest Point. 3. Registro de nuvens de pontos. I. Título.

CDD 621.38

POLYCARPO SOUZA NETO

GENERALIZED CLOUD PARTITIONING ITERATIVE CLOSEST POINT: UMA
AVALIAÇÃO QUANTITATIVA DO REGISTRO DE NUVENS DE PONTOS 3D

Dissertação apresentada ao Curso de Mestrado Acadêmico em Engenharia de Teleinformática do Programa de Pós-Graduação em Engenharia de Teleinformática do Centro de Tecnologia da Universidade Federal do Ceará, como requisito parcial à obtenção do título de mestre em Engenharia de Teleinformática. Área de Concentração: Sinais e Sistemas

Aprovada em: 04 de Fevereiro de 2019

BANCA EXAMINADORA

Prof. Dr. George André Pereira Thé (Orientador)
Universidade Federal do Ceará (UFC)

Prof. Dr. José Marques Soares (Coorientador)
Universidade Federal do Ceará (UFC)

Prof(a). Dr(a). Fátima Nelsizeuma Sombra de
Medeiros
Universidade Federal do Ceará (UFC)

Prof. Dr. Flávio de Barros Vidal
Universidade de Brasília (UnB)

Dedico este trabalho à Deus, meus pais, meu orientador e meu filho Guilherme.

AGRADECIMENTOS

À Deus por me conceder a oportunidade de ingressar e concluir o mestrado mesmo em meio a dificuldades que vão além dos muros da universidade.

Aos meus pais, Adauto (*in memorian*) e Judimar, pelo suporte que representam ao decorrer dos meus vinte e cinco anos de vida.

Ao Prof. George Thé eu agradeço a orientação incansável, o empenho, a confiança desde o início e principalmente sua paciência que ajudaram a tornar possível este sonho tão especial. Muito obrigado por tudo, essa vitória é em grande parte sua também.

À Prof(a) Fátima Sombra por me proporcionar o conhecimento não apenas de disciplinas, mas a manifestação do caráter e compromisso com a ciência.

Ao Prof. José Marques que compartilhou seus conhecimentos durante as calorosas reuniões e acompanhou a minha jornada enquanto mestrando.

Aos meus colegas da Sala de Estudos 1 e do INTERFACES (Grupo de Pesquisa em Processamento de Imagens 3D) que contribuíram no meu aprendizado das mais variadas formas.

Ao colega de mestrado Izaias Junior por introduzir as primeiras noções sobre registro de nuvens de pontos.

À Fundação Cearense de Apoio ao Desenvolvimento Científico e Tecnológico (FUN-CAP) pelo apoio financeiro por meio da bolsa de estudos.

À Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001 pelo auxílio financeiro que compete ao Programa de Pós-graduação em Engenharia de Teleinformática (PPGETI -UFC).

*"Nunca saberemos o quão forte devemos ser até
que a resiliência seja nossa única opção."*

(O autor.)

RESUMO

Em aplicações de reconstrução 3D, uma questão importante é a correspondência de nuvens de pontos de diferentes perspectivas de um determinado objeto ou cena. Tradicionalmente, esse problema é resolvido pelo uso do algoritmo *Iterative Closest Point* (ICP). Para melhorar a eficiência desta técnica, foi proposta uma metodologia de redução dos conjuntos de dados em sub-nuvens nos três eixos ortogonais. Foi ainda proposto um critério de convergência automático baseado numa medida de micro-desalinhamento. Neste trabalho, a técnica proposta foi comparada com outras dez técnicas. A avaliação dos resultados foi feita usando a métrica RMSE, uma análise da representação eixo-ângulo equivalente de rotação, e o custo computacional. Os testes foram realizados em condições ideais e em condições que simulam adversidades, como a existência de ruído, rotações em um eixo genérico e a diferença de densidade entre os conjuntos de dados. Os experimentos foram realizados em diversos conjuntos de dados diferentes, adquiridos por vários sensores, e revelou que a abordagem autoral alcançou uma correspondência de nuvem mais precisa, em um tempo menor que as outras técnicas do estado da arte.

Palavras-chave: Registro de nuvens de pontos; *Iterative Closest Point*; Particionamento; Micro-desalinhamento.

ABSTRACT

In 3D reconstruction applications, an important issue is the matching of point clouds from different perspectives of a particular object or scene. Traditionally, this problem is solved by using the Iterative Closest Point (ICP) algorithm. To improve the efficiency of this technique, a methodology for reducing data sets in sub-clouds on the three orthogonal axes was proposed. An automatic convergence criterion was also proposed based on a micro-misalignment measure. In this work, the proposed technique was compared with ten other techniques. The results were evaluated using the RMSE metric, an analysis of the equivalent axis-angle representation of rotation, and the computational cost. The tests were carried out under ideal conditions and in conditions that simulate adversities, such as the existence of noise, rotations on a generic axis and the difference in density between the data sets. The experiments were carried out on several different data sets, acquired by several sensors, and revealed that the authorial approach achieved a more accurate cloud match, in a shorter time than other state-of-the-art techniques.

Keywords: Point cloud registration; Iterative Closest Point; Partitioning; Micro-misalignment

LISTA DE FIGURAS

Figura 1	– Produção científica na área de registro de nuvem de pontos no período 2014-2019 segundo o Web of Science (Pesquisa realizada em 10 de dez. de 2018).	21
Figura 2	– Divisão dos trabalhos quantificados na Figura 1 por área temática.	21
Figura 3	– Exemplo de duas point clouds. A primeiro é o modelo Bunny da stanford e o segundo é uma nuvem capturada especialmente para esse trabalho com o sensor da Razer.	29
Figura 4	– Representação do ponto A no $frame L$	32
Figura 5	– Translação.	34
Figura 6	– Registro de vários modelos de nuvens de pontos Na parte superior, indicado pelos números 1,2 e 3 o registro de pares de nuvens e na parte inferior o registro de múltiplas tomadas parciais de uma cena <i>indoor</i>	35
Figura 7	– Uma análise da taxa de convergência do <i>Sparse</i> para diferentes valores de p . Quando p diminui, o número de iterações do algoritmo ICP aumenta.	43
Figura 8	– À esquerda o BnB e o ICP atualizam em colaboração os limites superiores e na direita com a orientação vinda do BnB, o ICP explora apenas um conjunto promissor de limites inferiores.	44
Figura 9	– À esquerda é uma representação do PFH e à direita a representação do FPFH (onde cada ponto usa apenas seus vizinhos diretos para calcular seu próprio SPFH.	46
Figura 10	– Representação eixo-ângulo.	48
Figura 11	– Metodologia adotada para comparação das variadas abordagens de registro. Em (1) os dados que são realizados o registro. Em (2) os métodos de registro. Em (3) o resultado do registro que passa por uma avaliação qualitativa (inspeção visual). Em (4) os resultados são comparados as métricas de avaliação quantitativas do erro de registro.	49
Figura 12	– Conjuntos de dados usados nos 9 experimentos. Os conjuntos de dados estão divididos na ilustração de acordo com as aplicações: em verde os dados de objetos, em vermelho de cenas <i>indoor</i> e em azul de cenas <i>outdoor</i>	50
Figura 13	– Exemplo de particionamento usando o GCP-ICP. O particionamento neste exemplo foi feito na direção top-down.	55
Figura 14	– Representação de um pequeno desalinhamento em eixos sucessivos.	57

Figura 15 – Fluxo do algoritmo proposto.	59
Figura 16 – Comparação do tempo de registro para os modelo Horse e Hammer usando 100 iterações ou $k = 100$	62
Figura 17 – Valores de RMSE para $k = 100$ relativos ao alinhamento de tomadas parciais do modelo <i>Dragon</i>	62
Figura 18 – Comparação do a) custo computacional, b) erro de registro (RMSE) e c) erro de rotação no alinhamento de cinco primeiros objetos, pelo método GCP-ICP, versus outros métodos locais.	66
Figura 19 – Registro de pares de nuvens de objetos sob diferentes condições de rotação. Os modelos mostrados são: (a) <i>Bunny</i> , (b) <i>Dragon</i> , (c) <i>Buddha</i> , (d) <i>Horse</i> e (e) <i>Hammer</i>	67
Figura 20 – Registro de pares de nuvens de objetos sob rotações em um eixo genérico. Os modelos mostrados são: (a) <i>Eagle</i> e (b) <i>Hippo</i>	69
Figura 21 – Resultado do registro de pares do modelo <i>Bunny</i> usando o GCP-ICP. Três níveis diferentes de ruído são testados.	70
Figura 22 – Comparação dos resultados do registro para o modelo <i>Buddha</i> . Em a) a perspectiva inicial das duas nuvens, em b) o registro para a amostragem aleatória de 50% e em c) o resultado do registro para a amostragem aleatória de 70% e em d) o resultado de registro com o GCP-ICP.	72
Figura 23 – Registro de múltiplas rotações dos objetos <i>Dragon</i> e <i>Buddha</i> . Em (a) e (c) temos as poses iniciais (esquerda) e em (b) e (d) o resultado do alinhamento.	73
Figura 24 – Resultado do alinhamento entre pares de nuvens de cenários <i>indoor</i> . Na parte superior, o alinhamento das nuvens adquiridas especialmente para esse experimento usando o sensor <i>Razer Stargazer SR 300</i> e na parte inferior com os dados obtidos de Mellado <i>et al.</i> (2014).	75
Figura 25 – Alinhamento da cena <i>Lounge chair</i> sob diferentes perspectivas.	76
Figura 26 – Alinhamento da cena <i>Hannover 2</i>	78
Figura 27 – Alinhamento da cena <i>Bremen 1</i>	80

LISTA DE TABELAS

Tabela 1 – Informações sobre os bancos de dados e modelos usados nos experimentos.	50
Tabela 2 – Redução de densidade dos pontos.	51
Tabela 3 – Parâmetros do método proposto.	59
Tabela 4 – Experimentos para aferição da qualidade do método GCP-ICP.	60
Tabela 5 – Tempo consumido para alinhar os pares de nuvens de pontos (em segundos).	63
Tabela 6 – RMSE calculado para o registro com cada método.	64
Tabela 7 – Rotação medida para as variadas técnicas (em graus)	65
Tabela 8 – Tempo consumido para alinhar os pares de nuvens de pontos (em segundos).	68
Tabela 9 – RMSE calculado para o registro com cada método.	68
Tabela 10 – Tempo consumido para alinhar os pares de nuvens de pontos sob diferentes quantidades de ruído (em segundos).	70
Tabela 11 – RMSE calculado para o registro com cada método para os pares de nuvem do modelo <i>Bunny</i> sob diferentes quantidades de ruído (em metros).	70
Tabela 12 – Efeito da amostragem no registro de nuvens de pontos.	71
Tabela 13 – Comparação do alinhamento de múltiplos <i>scans</i> de objetos.	73
Tabela 14 – Custo computacional da etapa do alinhamento entre pares de cenas <i>indoor</i>	74
Tabela 15 – RMSE do alinhamento entre pares de cenas <i>indoor</i>	74
Tabela 16 – Custo computacional para alinhamento de múltiplas aquisições de cenários <i>indoor</i>	76
Tabela 17 – RMSE do alinhamento de múltiplas aquisições de cenários <i>indoor</i>	76
Tabela 18 – Resultado do alinhamento de cenas <i>outdoor</i>	77
Tabela 19 – Tempo de registro do alinhamento de cenas <i>outdoor</i> para nuvens com dife- rentes densidades (em segundos).	78
Tabela 20 – RMSE do registro de cenas <i>outdoor</i> para nuvens com diferentes densidades.	79

LISTA DE ABREVIATURAS E SIGLAS

1D	unidimensionais
2D	bimensionais
3D	tridimensionais
AUV	Veículo subaquático autônomo
BnB	<i>Branch in Bound</i>
CP-ICP	<i>Cloud Partitioning Iterative Closest Point</i>
D	profundidade
DoF	Graus de liberdade
FPFH	<i>Fast Point Feature Histogram</i>
GCP-ICP	<i>Generalized Cloud Partitioning Iterative Closest Point</i>
GICP	<i>Generalized Iterative Closest Point</i>
GoICP	<i>Global Optimally Iterative Closest Point</i>
ICP	<i>Iterative Closest Point</i>
IEEE	<i>Institute of Electrical and Electronics Engineers</i>
IoT	Internet das Coisas
LiDaR	<i>Light Detection and Ranging</i>
MLE	<i>Maximum Likelihood Estimation</i>
MSE	Erro Quadrático Médio
NDT	<i>Normal Distribution Transform</i>
NICP	<i>Normal Iterative Closest Point</i>
PCL	<i>Point Cloud Library</i>
PFH	<i>Point Feature Histogram</i>
RANSAC	<i>Random Sample Consensus</i>
RGB	imagem de cor
RMSE	Raiz do Erro Quadrático Médio
SLAM	Localização e mapeamento simultâneos
SPFH	<i>Simplified Point Feature Histogram</i>
ToF	<i>Time-of-flight</i>

LISTA DE SÍMBOLOS

X	Eixo X
Y	Eixo Y
Z	Eixo Z
x	Coordenada referente ao eixo X
y	Coordenada referente ao eixo Y
z	Coordenada referente ao eixo Z
P	Superfície genérica
p_i	Ponto em uma superfície genérica P
P^{k_n}	Vizinhança de cada ponto de uma superfície P
k_n	Vizinhos mais próximos
P_{final}	Superfície P após a remoção de <i>outliers</i>
v	<i>Voxel</i>
w	Centro do <i>voxel</i>
x_0	Coordenada inicial do centro do <i>voxel</i> referente ao eixo X
y_0	Coordenada inicial do centro do <i>voxel</i> referente ao eixo Y
z_0	Coordenada inicial do centro do <i>voxel</i> referente ao eixo Z
N_S	Número de pontos na nuvem <i>source</i>
N_T	Número de pontos na nuvem <i>target</i>
S	Nuvem de pontos <i>source</i>
T	Nuvem de pontos <i>target</i>
ψ	Transformação rígida que aproxima dois conjuntos de dados
\vec{s}_i	i -ésimo vetor da nuvem de pontos <i>source</i>
\vec{m}_i	i -ésimo vetor da nuvem de pontos <i>target</i>
R	Matriz de rotação
\vec{q}_R	Vetor <i>quaternion</i> de rotação
\vec{q}_T	Vetor <i>quaternion</i> de translação

\vec{q}	Vetor <i>quaternion</i> formado dos vetores de rotação e translação
$F(\vec{q})$	Função custo do algoritmo <i>Iterative Closest Point</i>
C_{ST}	Matriz de covariância cruzada dos conjuntos de dados
$Q(C_{ST})$	Matriz de obtenção do quaternion para o registro
θ	Ângulo de micro-desalinhamento resultante da multiplicação dos ângulos dos eixos principais
α	Ângulo de rotação em Z
β	Ângulo de rotação em Y
γ	Ângulo de rotação em X
R_Z	Matriz de rotação em torno de Z
R_Y	Matriz de rotação em torno de Y
R_X	Matriz de rotação em torno de x
c_z	Cosseno de γ em torno de Z
c_y	Cosseno de β em torno de Y
c_x	Cosseno de α em torno de X
c_w	Cosseno de θ em torno de um eixo genérico
s_z	Seno de γ em torno de Z
s_y	Seno de β em torno de Y
s_x	Seno de α em torno de X
s_w	Seno de θ em torno de um eixo genérico
k	Parâmetro da quantidade de sub-nuvens
E_i	Erro de distância entre dois pontos
φ	Critério de parada
ε	Erro calculado numa dada iteração
S_j	j-ésima sub-nuvem <i>source</i>
T_j	j-ésima sub-nuvem <i>target</i>
ξ	Eixo de corte
Γ_j	Sub-conjunto de pontos da sub-nuvem <i>source</i>

B_j	Sub-conjunto de pontos da sub-nuvvem <i>target</i>
F_{p2p}	FUnção custo com a métrica ponto-ao-plano
n_{t_i}	Normal da superfície
τ	Representação parametrizada de uma transformação rígida
C_i^S	Matriz de covariância de S
C_i^T	Matriz de covariância de T
P_i	Projeção ortogonal
z_i	Vetor residual
p	Valor da norma p no intervalo $[0, 1]$
\underline{E}_r	Limite inferior de rotação
\overline{E}_r	Limite superior de rotação
\underline{E}_t	Limite inferior de translação
\overline{E}_t	Limite superior de translação
E_{best}	Melhor erro até o momento
p_i	Ponto atual
n_j	Vetor normal
p_k	Ponto mais próximo
w_{k_n}	Distância entre p_i e p_k
\bar{I}	Conjunto de pontos aproximados pelo RANSAC
$ I $	Cardinalidade de I
Ψ_{RANSAC}	Matriz de transformação obtida com RANSAC
Ψ_{ICP}	Matriz de transformação do refinamento obtida com ICP

SUMÁRIO

1	INTRODUÇÃO	19
1.1	Relevância do tema	20
1.2	Motivação	21
1.3	Objetivo	25
1.4	Contribuições	26
1.5	Produção científica	26
1.6	Organização geral da dissertação	27
2	FUNDAMENTAÇÃO TEÓRICA	28
2.1	Nuvem de pontos	28
2.2	Pré-processamentos realizados em nuvens de pontos	29
2.2.1	<i>Amostragem dos pontos</i>	29
2.2.1.1	<i>Amostragem aleatória</i>	30
2.2.1.2	<i>Amostragem uniforme</i>	30
2.2.2	<i>Remoção de outliers</i>	30
2.2.3	<i>Correção de densidade</i>	31
2.3	Transformações no espaço tridimensional	31
2.3.1	<i>Orientação</i>	31
2.3.2	<i>Posição</i>	33
2.3.3	<i>Matriz de transformação</i>	34
2.4	Registro de nuvens de pontos	35
2.4.1	<i>Iterative Closest Point (ICP)</i>	36
2.4.2	<i>ICP ponto-ao-plano</i>	38
2.4.3	<i>Generalized ICP (GICP)</i>	39
2.4.4	<i>Normal Distribution Transform (3D-NDT)</i>	40
2.4.5	<i>Sparse ICP</i>	41
2.4.6	<i>Globally Optimally ICP (GoICP)</i>	43
2.4.7	<i>Cloud Partitioning ICP (CP-ICP)</i>	44
2.4.8	<i>Fast Point Feature Histogram (FPFH)</i>	45
2.5	Representação eixo-ângulo equivalente	47
3	METODOLOGIA	49

3.1	Conjuntos de dados disponíveis	49
3.2	Pré-processamento dos dados	51
3.2.1	<i>Remoção de outliers</i>	51
3.2.2	<i>Redução da densidade das nuvens</i>	51
3.3	Generalized Cloud Partitioning ICP(GCP-ICP)	52
3.3.1	<i>Visão geral das etapas do ICP no método proposto</i>	52
3.3.1.1	<i>Seleção</i>	52
3.3.1.2	<i>Correspondência</i>	53
3.3.1.3	<i>Métrica de erro e minimização</i>	53
3.3.2	<i>Formulação matemática</i>	53
3.3.2.1	<i>Particionamento</i>	54
3.3.2.2	<i>Verificação de correspondência</i>	55
3.3.2.3	<i>Critério de convergência</i>	57
3.3.3	<i>Detalhes de implementação</i>	59
3.4	Experimentos	60
4	RESULTADOS	61
4.1	Resultados prévios	61
4.1.1	<i>Custo computacional</i>	61
4.1.2	<i>Estabilidade do RMSE</i>	62
4.2	Experimento A: Registro de pares de objetos com conhecimento das rotações de cada objeto	63
4.3	Experimento B: Registro de pares de objetos submetidos a sucessivas rotações em um eixo genérico.	67
4.4	Experimento C: Registro de pares de nuvem na presença de ruído	69
4.5	Experimento D: Efeito de métodos de amostragem clássicos frente ao particionamento	71
4.6	Experimento E: Registro de múltiplas rotações de um mesmo objeto.	72
4.7	Experimento F: Alinhamento de pares de nuvens de pontos de cenas <i>indoor</i>	74
4.8	Experimento G: Alinhamento de múltiplos <i>scans</i> de cenas <i>indoor</i>	75
4.9	Experimento H: Alinhamento de tomadas parciais de cenas <i>outdoor</i>	77

4.10	Experimento I: Registro de pares de cenas <i>outdoor</i> com densidades de pontos diferentes	78
5	CONCLUSÕES E TRABALHOS FUTUROS	81
	REFERÊNCIAS	82

1 INTRODUÇÃO

A revolução digital expandiu rapidamente seu alcance para todos os aspectos da vida humana. O aumento de dispositivos contendo algum tipo de câmera e a conectividade deles com a *internet* promete grandes mudanças nas instituições de ensino, hospitais e na área de segurança. A revolução digital também se manifesta no aumento da automação e a crescente implantação de sistemas baseados em Internet das Coisas (IoT) e de sistemas robóticos inteligentes para soluções para a chamada indústria 4.0 (STORK, 2015) .

Com a crescente usabilidade de câmeras no nosso dia-a-dia, torna-se importante capturar imagens tridimensionais (3D), visto que o ambiente onde interagimos está repleto de situações que as imagens bimensionais (2D) não conseguem capturar muitas vezes por oclusões parciais. Além disso, sistemas robóticos inteligentes precisam de conhecimento espacial preciso e mapas 3D do ambiente para navegação segura. O mapeamento 3D é particularmente importante para cenários onde robôs e seres humanos compartilham o mesmo espaço ou mesmo prevenção de colisão (KALDESTAD *et al.*, 2014).

Dados 3D de objetos, de cenas em ambientes internos e ao ar livre podem ser capturados usando diferentes sensores e técnicas, tais como *Light Detection and Ranging* (LiDaR), detecção de luz estruturada, como por exemplo, o *Intel® RealSense™* e o *Microsoft® Kinect™* e câmeras *Time-of-flight* (ToF) (WEINMANN, 2016). A existência desse tipo de câmera já esta presente inclusive nos celulares modernos, caso da tecnologia de *Face ID unlock* presente no *Iphone X* da *Apple®*. O desenvolvimento de equipamentos de escaneamento 3D torna o registro de nuvens de pontos de um objeto possível. Devido à oclusão, no entanto, o objeto não pode ser inteiramente capturado de um único ponto de vista. Portanto, os *scanners* devem adquirir conjuntos de pontos de diferentes pontos de vista, de modo a cobrir toda a superfície do objeto. Esses conjuntos de pontos podem ser transformados em um único *frame* a partir do registro dessas múltiplas visões.

Fundamental em muitas áreas, o problema de alinhamento de nuvens de pontos atraiu imensa atenção da comunidade científica de visão computacional (TAM *et al.*, 2013) e robótica (POMERLEAU *et al.*, 2015). O objetivo do registro de nuvens de pontos é estimar a transformação rígida ideal entre diferentes conjuntos de pontos. De acordo com o número de nuvens de pontos envolvidas, esse problema pode ser dividido, de maneira geral, no registro de pares e no registro de múltiplas tomadas parciais, com o primeiro sendo amplamente estudado. Comumente, o problema do registro em pares de nuvens é resolvido pelo algoritmo *Iterative*

Closest Point (ICP) em Besl e McKay (1992) ou uma de suas variantes. Para conseguir o registro, essas abordagens, em sua maioria realizam alternadamente correspondências e estimam a transformação rígida.

Em robótica móvel, em experimentos de navegação, a localização pode ser obtida registrando-se os obstáculos e cenário ao redor do robô (POMERLEAU *et al.*, 2013). Em (HIMRI *et al.*, 2018) é mostrado um sistema de navegação e mapeamento para um Veículo subaquático autônomo (AUV) para operar próximo a um ambiente subaquático controlado, com o objetivo de reconhecer objetos (ou partes de objetos) e usá-los como pontos de referência para Localização e mapeamento simultâneos (SLAM). Aplicações de registro são usadas ainda em reconhecimento facial (FELIPE *et al.*, 2018; SIQUEIRA *et al.*, 2018) e de expressões (CHENG *et al.*, 2017), podendo servir como discriminador de características para a etapa de classificação (SAVRAN; SANKUR, 2017). O registro de nuvem de pontos está presente em tarefas de reconhecimento, onde a qualidade de um processo de classificação está intimamente ligada à obtenção de uma boa etapa de registro (GUO *et al.*, 2014).

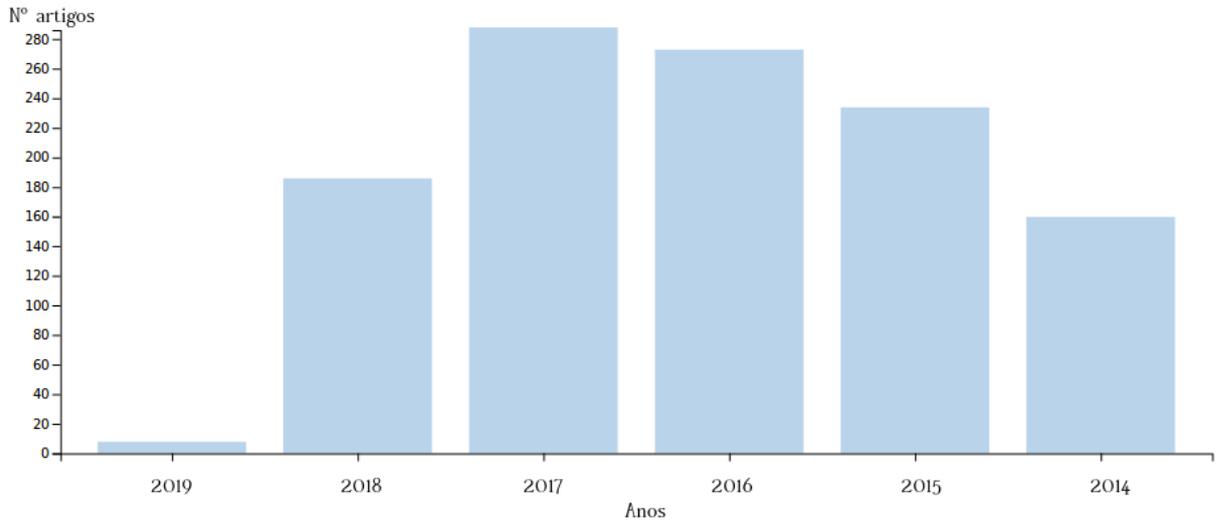
Além do crescimento e difusão de dispositivos de profundidade, bibliotecas de tratamento desses dados também apareceram nos últimos anos, caso da *Point Cloud Library* (PCL) (RUSU; COUSINS, 2011). A PCL é um *framework* largamente difundido, já consolidado e que é utilizado para executar os processamentos de nuvens de pontos.

Dado este contexto, o presente trabalho se encaixa na problemática da utilização dos recursos de visão computacional 3D, como ferramenta para melhor descrever objetos e cenas, sendo uma tomada de decisão promissora para correção de oclusão. Tal abordagem vem do registro de nuvens de pontos 3D de diferentes perspectivas, onde registro é o termo que faz referência à fusão entre os dados parciais tomados de diferentes perspectivas.

1.1 Relevância do tema

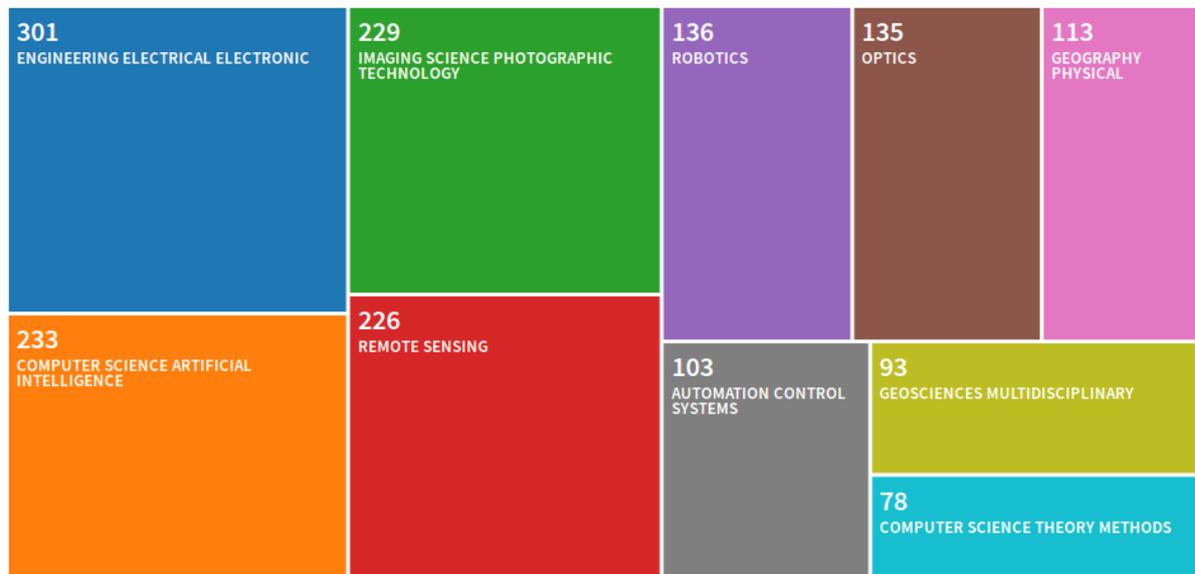
Nos últimos anos, a área de registro cresceu consideravelmente, com publicações de novos métodos, *surveys* e *reviews*. Uma constatação disso pode ser vista nas Figuras 1 e 2, onde a partir de uma busca no Web of Science com a *string* de busca "*point cloud registration*" foram retornados 1647 trabalhos para os últimos seis anos (2014-2019) e um mapa com a divisão desses trabalhos por área afim.

Figura 1 – Produção científica na área de registro de nuvem de pontos no período 2014-2019 segundo o Web of Science (Pesquisa realizada em 10 de dez. de 2018).



Fonte: Adaptado de Web of Science (2018).

Figura 2 – Divisão dos trabalhos quantificados na Figura 1 por área temática.



Fonte: Web of Science (2018).

1.2 Motivação

Como já foi mencionado pelos exemplos na seção de motivação, visão computacional tem grande importância na resolução de problemas de engenharia. São esboçados por meio do que a literatura reportou, alguns critérios motivadores do estudo que culminaram nessa dissertação.

Algoritmos de registro são usados para alinhar representações parciais de uma cena

ou objeto em um único *frame*. O problema vem sendo tratado extensivamente em vários estudos nos últimos 26 anos. Isso começou com abordagens como o ICP. O ICP é usado para calcular a transformação rígida ideal, superpondo regiões coincidentes das duas nuvens de pontos por um processo baseado na correspondência desses pontos e a minimização de uma métrica que descreve o erro do alinhamento (BESL; MCKAY, 1992). O problema do registro de nuvens de pontos é abordado na literatura a partir de diferentes escolas; em essência, a maioria das técnicas se baseia em métodos executados em algum espaço de características (métodos esparsos), métodos aplicados ao espaço de todos os pontos da superfície (métodos densos) ou baseado em informações geométricas como sólidos (3D), projeções de planos (2D) e segmentos de linha unidimensionais (1D) (métodos baseados em objetos) (TAZIR *et al.*, 2018).

Métodos esparsos são geralmente usados em ambientes *outdoor* e tarefas de reconhecimento (WEINMANN, 2016). De acordo com (TAZIR *et al.*, 2018) métodos esparsos são baseados no uso de características, que podem ser pontos facilmente identificados por terem uma alta representatividade local em relação aos outros pontos. Um bom algoritmo de detecção de características deve estimá-las em todas as nuvens. Eles devem ser robustos à presença de ruído e variações de orientação. Como método esparsos, algumas abordagens podem ser citadas, caso do *Point Feature Histogram* (PFH), introduzido por Rusu *et al.* (2008) para descrever a geometria local em torno de cada ponto, a fim de classificá-los, por meio de um histograma multidimensional, de acordo com sua natureza local (superfície plana ou um ponto de borda). Rusu *et al.* (2009) foi proposto o *Fast Point Feature Histogram* (FPFH) que modifica a matemática do modelo de PFH, a fim de reduzir sua complexidade computacional.

Os métodos que se enquadram nesta categoria não requerem conhecimento *a priori* (SERAFIN *et al.*, 2016). Alguns métodos identificam regiões nas imagens onde a estimação das características pode ser mais eficiente, mas a maneira como esses pontos são determinados é demorada e, portanto, muitas vezes não são adequados para as aplicações que exigem baixo custo computacional, como aplicações de tempo real. Um exemplo disso pode ser visto em Siqueira *et al.* (2018) ao entregar uma resposta de classificação de indivíduos mais rápida, pois tem uma dependência mínima de características para o processo de registro. Uma dependência de métodos baseados em características é a necessidade de nuvens densas (com uma quantidade vasta de pontos), o que inviabiliza para esse tipo de aplicação o uso de nuvens esparsas (nuvens com poucos pontos) adquiridas com sensores de baixa resolução, muitas vezes sendo adquiridas com quantidade considerável de ruído (TAZIR *et al.*, 2018).

Métodos baseados em objetos sofrem de segmentação imperfeita, tendendo a rejeitar muitos dados úteis, uma vez que exploram informações de modelos geométricos simples (DUBÉ *et al.*, 2016). Fernández-Moral *et al.* (2016) propõem um método de registro baseado na superfície plana. Este trabalho representa uma extensão de um trabalho já publicado em Fernández-Moral *et al.* (2013), que trata de reconhecimento em ambientes internos por extração de planos. A extensão é principalmente focada em adicionar uma estrutura probabilística para explicar o modelo de incerteza do sensor. Outras técnicas também podem ser usadas para essa finalidade, como o *Random Sample Consensus* (RANSAC) (FISCHLER; BOLLES, 1981), que essencialmente, tem um ajuste de distribuição de dados no espaço de um polinômio que representa uma equação de um plano. No entanto, para situações onde a densidade das nuvens não é a mesma, a dispersão dos pontos representa um problema para obtenção de uma segmentação precisa (TAZIR *et al.*, 2018).

Por último, a tradicional escola de registro usando métodos densos, que inclui o ICP e suas mais de 450 variantes (até 2013 e somente as publicadas no *Institute of Electrical and Electronics Engineers (IEEE) Xplore* ©) (POMERLEAU *et al.*, 2015), que ainda é popular devido à simplicidade e disponibilidade em bibliotecas de código aberto. Entre essas variantes, algumas implementações recentes desta abordagem local merecem alguma atenção.

Uma abordagem eficiente para registro de dados 3D foi apresentada por Segal *et al.* (2009). Esta versão probabilística do ICP chamada *Generalized Iterative Closest Point* (GICP) é baseada no princípio *Maximum Likelihood Estimation* (MLE). Esse algoritmo explora os caminhos planos locais em ambas as nuvens de pontos, o que leva ao conceito de plano-a-plano no registro. Nessa contribuição, os autores generalizam métricas originalmente introduzidas em Besl e McKay (1992) e Chen e Medioni (1992) e, embora eficientes, são afetadas pela não uniformidade da densidade de pontos sobre as superfícies que estão sendo correspondidas. Sua exposição dos resultados é feita por meio de gráficos com erro de distância e análises por inspeção visual. Em Holz *et al.* (2015), é mostrado que as densidades pontuais não uniformes causam estimativas imprecisas, o que degrada o desempenho do algoritmo. Além disso, em (AGAMENNONI *et al.*, 2016) os autores afirmam que o GICP não funciona bem em ambientes externos e ambientes não estruturados.

Bouaziz *et al.* (2013) apresenta o *Sparse ICP*, que visa abordar o problema do registro de nuvens de pontos em condições onde os dados possuem *outliers* ou sofrem com problema de ausência de dados nas nuvens. Mavridis *et al.* (2015) desenvolveram uma variante do *Sparse*

ICP, corrigindo por meio de uma pesquisa baseada em *simulated annealing* o problema de correspondências presentes no método anterior. Similarmente aos demais trabalhos, estes se baseiam na análise de RMSE, custo computacional e avaliação qualitativa dos resultados dos registros.

Serafin e Grisetti (2015) propuseram o *Normal Iterative Closest Point* (NICP), onde estenderam o algoritmo do GICP acrescentando informações na função de erro e na seleção de correspondências que, segundo os autores, aumentam a robustez do registro. O NICP trabalha na projeção das duas nuvens em imagens parciais. Para a nuvem de referência, essa imagem é iterativamente calculada, o que leva a um custo computacional considerável. Essas imagens de intervalo servem principalmente para a seleção de pontos correspondentes. Os pontos correspondentes são selecionados a partir da imagem, de modo que são pontos que compartilham o mesmo *pixel* e têm normais e curvaturas compatíveis.

Prakhya *et al.* (2015) realizaram uma extensão de formulação prévia da covariância do registro gerada pelo ICP, saindo do domínio 2D para o 3D. Tal trabalho, traz uma expectativa de explorar a covariância do ICP como guia de uma solução que leve o algoritmo a convergir para um mínimo global, evitando a suscetibilidade de queda em mínimos locais. No artigo de Yang *et al.* (2015) é apresentado o *Global Optimally Iterative Closest Point* (GoICP), uma versão do ICP que visa também fugir de problemas com mínimos locais. O método GoICP é baseado em um esquema *Branch in Bound* (BnB) que pesquisa todo o espaço de movimento 3D. Ao explorar a estrutura especial da geometria, são obtidos novos limites superior e inferior para a função de erro de registro. A forma como os autores mostram seus resultados é através do RMSE e avaliações qualitativas do registro.

O *Normal Distribution Transform* (NDT) desenvolvido por Magnusson *et al.* (2007) discretiza o ambiente em células, onde cada uma é modelada por uma matriz representando a probabilidade de ocupação de seus pontos (linear, planar e esférica). Então, uma otimização não linear é executada para calcular a transformação entre as duas nuvens. No entanto, de acordo com Das *et al.* (2014), o NDT não é adequado para sistemas com baixa capacidade computacional, visto o seu custo computacional para uso.

Fazendo uma breve análise sobre o ICP de Besl e McKay (1992), mas não menos importante, podem ser citados outros gargalos além do problema de cair em mínimos indesejáveis. O custo computacional para o cálculo do ponto mais próximo no ICP clássico é $O(N^2)$, onde N é o número de pontos em cada conjunto de dados. Nota-se que nesta análise, apenas o cálculo do

ponto mais próximo foi considerado, que é o passo mais demorado do ICP. Outro ponto que deve ser levado em consideração é que o ICP precisa de uma boa inicialização, como dados com boa sobreposição, uma rotação aplicada previamente sobre os dados e um número ideal de iterações.

Para resolver o problema de custo computacional por exemplo, são adotados métodos de discretização dos dados, tornando-os mais facilmente tratáveis pelos algoritmos de registro (RODOLÀ *et al.*, 2015). Esse passo funciona como um etapa prévia de seleção para vários algoritmos, inclusive o próprio ICP. Essa prática de seleção é bem comum e pode ser feita por meio de métodos amostragem. Na literatura podemos encontrar estratégias como a amostragem uniforme (VITTER, 1984), a amostragem aleatória (ZHANG, 1994) e a amostragem de acordo com a orientação das normais (RUSINKIEWICZ; LEVOY, 2001). No entanto, como o registro é realizado em conjuntos de dados de quaisquer natureza, estratégias de seleção podem trazer redução do custo computacional para nuvens densas. Para nuvens de pontos esparsas, estas estratégias de seleção podem trazer degradação de características e consequentemente de informações carregadas pelos pontos, o que gera um registro com resultados indesejáveis (TAZIR *et al.*, 2018).

Em 2015 foi proposto o *Cloud Partitioning ICP* (PEREIRA *et al.*, 2015), onde é proposto um pré-processamento iterativo de particionamento das nuvens de pontos e o resultado deste particionamento é submetido ao ICP clássico. Os autores propuseram uma variante em SOUZA NETO *et al.* (2018), onde o critério de parada é o RMSE da iteração onde o ICP converge (estudado *a priori*), o que trouxe uma melhoria na convergência, tanto em termos de precisão, como velocidade em relação ao ICP clássico, porém não automatizando totalmente o método.

A partir da revisão da literatura, vê-se que a comunidade científica tem dado importância ao tema aqui estudado e que este trabalho encontra-se inserido no contexto do estado da arte.

1.3 Objetivo

Este trabalho tem por objetivo geral o desenvolvimento de uma técnica de registro baseada no particionamento dos dados e ordenação dos pontos, de desempenho satisfatório, a ser utilizada previamente ao ICP no intuito do registro de nuvens de pontos 3D. Para que seja alcançado esse objetivo, são listados os objetivos específicos:

- Analisar as etapas envolvidas no processo de registro, em especial o algoritmo ICP;

- Melhorar a etapa de correspondência entre os pontos dos conjuntos de dados submetidos ao ICP;
- Avaliar a generalização do método e reportar os resultados por meio de outras métricas de qualidade, que não apenas o RMSE ou análises por inspeção visual;
- Dar condições ao leitor para que os experimentos e resultados desenvolvidos neste trabalho possam ser reproduzidos.

1.4 Contribuições

Neste trabalho, uma nova abordagem para registro de nuvens de pontos é introduzida. As contribuições obtidas são as seguintes:

- Uma estratégia de seleção de pontos baseada no particionamento das nuvens em qualquer um dos eixos principais (ortogonais);
- A elaboração de um critério de convergência automático e restritivo, chamado aqui de micro-desalinhamento.

1.5 Produção científica

Ao longo do desenvolvimento do trabalho, foram apresentados e publicados em anais de conferências os seguintes trabalhos:

- SOUZA NETO, Polycarpo; SILVA PEREIRA, Nicolás; PEREIRA THÉ, George André. Improved Cloud Partitioning Sampling for Iterative Closest Point: Qualitative and Quantitative Comparison Study. **In: 15th International Conference on Informatics in Control, Automation and Robotics**, 2018, Porto. Proceedings of ICINCO 2018... [S.l.]: SCITEPRESS Digital Library, 2018. p. 49-61. v. 2.
- SOUZA NETO, Polycarpo; SILVA PEREIRA, Nicolás; PEREIRA THÉ, George André; MARQUES SOARES, José. Avaliação Quantitativa de Desempenho do Registro de Imagens Tridimensionais de Faces por Meio do Algoritmo CP-ICP. **In: XXII Congresso Brasileiro de Automática**, João Pessoa. Anais do CBA 2018 ... [S.l.];, 2018. p. 1-8. v. 1.

1.6 Organização geral da dissertação

O restante deste trabalho está organizado em capítulos como mostrado posteriormente:

- No Capítulo 2 é feita a fundamentação teórica descrevendo os temas principais relacionados a este trabalho, como registro de nuvens de pontos e a descrição das técnicas de comparação;
- No Capítulo 3 é feita a descrição do algoritmo proposto (de agora em diante chamado Generalized Cloud Partitioning Iterative Closest Point) e a descrição dos conjuntos de dados, dos pré-processamentos, alguns detalhes de implementação e a organização dos experimentos realizados;
- No Capítulo 4 são apresentados os resultados e discussões sobre a utilização do método de registro proposto para vários cenários de experimentação;
- Por fim, no Capítulo 5, são feitas as considerações finais sobre o trabalho, citando as contribuições, as conclusões do trabalho e as perspectivas para trabalhos futuros.

2 FUNDAMENTAÇÃO TEÓRICA

Este capítulo destina-se ao embasamento teórico necessário para a compreensão das técnicas e algoritmos envolvidos na presente dissertação.

2.1 Nuvem de pontos

O termo nuvem de pontos é usado para descrever um conjunto de pontos (conjunto de dados) em um determinado espaço. Em Rusu e Cousins (2011), uma nuvem de pontos pode ser dita como uma estrutura de dados usada para representar um conjunto finito de pontos multidimensionais. Como o foco deste trabalho está centrado em operações com nuvens de pontos 3D representando partes de superfícies ou objetos em um ambiente, vamos nos concentrar em nuvens de pontos 3D como uma coleção de pontos 3D que, por sua vez, são caracterizados por coordenadas espaciais x, y, z e podem, opcionalmente, ter atributos extras (tais como informações de intensidade, informações como mapas térmicos, normal e escala).

As nuvens de pontos podem ser adquiridas por meio de sensores de profundidade. São várias as formas de aquisição, podendo acontecer por sensores *Time-of-flight*, onde o princípio por trás desta tecnologia baseia-se na medição da distância aos pontos das superfícies através do tempo que a radiação emitida demora a chegar aos objetos e voltar (WEINMANN, 2016). Sabendo esse tempo, a velocidade, e a direção da emissão da radiação, é possível saber a distância a que uma determinada superfície se encontra e quais as coordenadas 3D dos pontos. Como exemplo deste tipo de sensor pode ser citado o o *Riegl VZ-6000*¹, para medições a longa distância e para curto alcance (80 cm), o *CamCube* da *PMD Technologies*² e alguns modelos com a tecnologia *Intel RealSense*³.

Existem ainda outros sensores, baseado no princípio de luz estruturada. Esse princípio é a projeção de um padrão numa cena, onde o modo como o padrão se deforma quando atinge as superfícies possibilita o cálculo da profundidade e de informações das superfícies atingidas. Um exemplo desse tipo de sensor é o *Kinect* e o *Razer Stargazer SR 300*⁴. Ambos os sensores são câmeras RGB-D, ou seja, uma câmara capaz de capturar simultaneamente imagem de cor (RGB) e de profundidade (D).

¹ 1, <http://www.riegl.com/nc/products/terrestrial-scanning/produktdetail/product/scanner/33/>

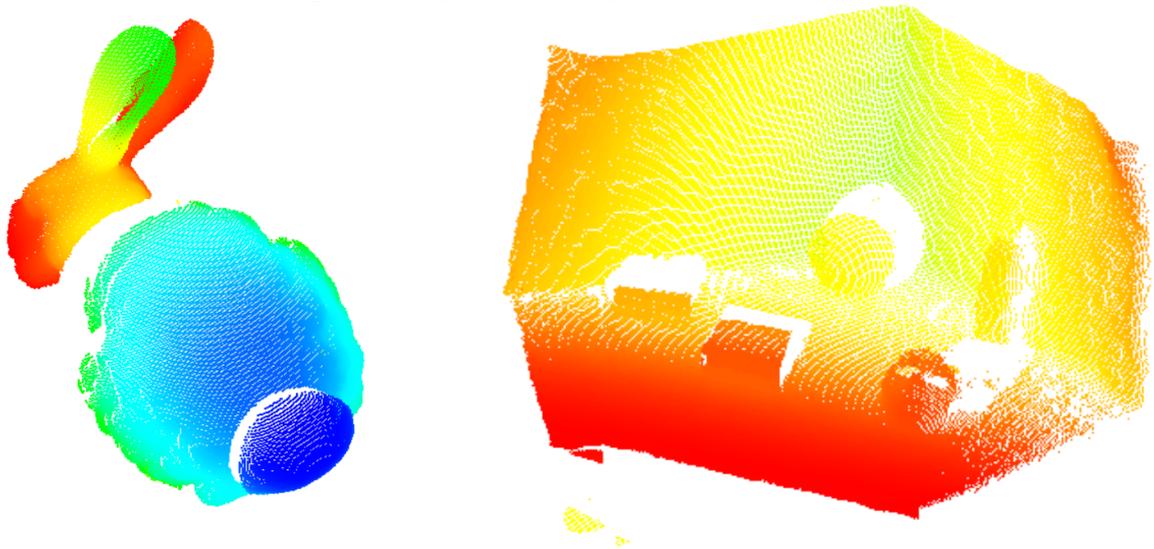
² 2, <https://www.pmdtec.com/>

³ 3, <https://click.intel.com/realsense.html>

⁴ 5, <https://support.razer.com/gaming-headsets-and-audio/razer-stargazer/>

Dois exemplos de *point clouds* podem ser vistos na Figura 3. A primeira é originada do banco de dados público da Stanford (UNIVERSITY, 2014) e a segunda foi adquirida especialmente para este trabalho com um sensor de baixo custo e baixa resolução, o Real Sense SR 300.

Figura 3 – Exemplo de duas point clouds. A primeira é o modelo Bunny da Stanford e o segundo é uma nuvem capturada especialmente para esse trabalho com o sensor da Razer.



Fonte: o autor.

2.2 Pré-processamentos realizados em nuvens de pontos

Nesta seção serão descritos sobre alguns pré-processamentos necessários antes de usar os dados de nuvens de pontos adquiridos por alguns sensores.

2.2.1 Amostragem dos pontos

Quando uma parte de um todo é utilizada para inferir algo sobre o conjunto completo, tem-se uma amostragem daquele conjunto. A ideia ao fazer a amostragem (em qualquer contexto) é concluir sobre alguma propriedade com base numa parcela menor daquele conjunto. No contexto do registro de nuvens de pontos, os métodos de amostragem são usados para redução da quantidade de dados expostos aos algoritmos de alinhamento, a fim de reduzir o custo computacional da correspondência entre conjuntos de dados.

2.2.1.1 Amostragem aleatória

A amostragem aleatória é uma boa técnica de amostragem, visando buscar apenas pontos internos (ZHANG, 1994). Aplica-se esta técnica para a primeira nuvem P e escolhemos um conjunto de N_P pontos de P aleatoriamente. A forma de avaliar a probabilidade da uma boa amostragem é considerando isso a partir de um ponto de vista probabilístico. Seja ε a rejeição de *outliers*, a probabilidade de escolher *inliers* é $1 - \varepsilon$. Além disso, a probabilidade de que a escolha seja de um subconjunto com N_P pontos que são todos os *inliers* é $(1 - \varepsilon)^{N_P}$. O valor p_i é uma probabilidade de pelo menos uma subamostra ser um *inlier*. A equação que sintetiza a lógica acima é vista abaixo:

$$p(\varepsilon, N_P, p_i) = 1 - (1 - (1 - \varepsilon)^{N_P})^{p_i}. \quad (2.1)$$

2.2.1.2 Amostragem uniforme

O método de amostragem uniforme monta uma grade 3D local sobre a nuvem de pontos, e reduz os pontos baseado na distância. Então em cada parte dessa grade todos os pontos presentes serão aproximados ao seu centróide das subdivisões na grade.

2.2.2 Remoção de outliers

Durante o processo de aquisição, alguns artefatos são incluídos como pontos, porém são coordenadas a quem do objeto que se deseja capturar, estando distante da distribuição real dos pontos que representa a superfície da nuvem.

A solução usada para remoção de *outliers* é baseada em uma análise estatística da vizinhança dos pontos, P^{k_n} , de uma superfície genérica P . Para cada ponto $p_i \in P$, a distância média \bar{d} é calculada para os vizinhos mais próximos. Então, uma distribuição sobre o espaço médio das distância para toda a nuvem de pontos P é montada e sua média μ_{k_n} e o desvio padrão σ_{k_n} são estimados (RUSU, 2013).

O objetivo é reter os pontos cuja a distância média \bar{d} para os k_n vizinhos, tenha comportamento semelhante a da maioria dos pontos da nuvem. Como isso descreve uma medida da densidade da nuvem de pontos ao redor de um ponto específico, a nuvem de pontos remanescente P_{final} é simplesmente estimada da seguinte forma:

$$P_{final} = [\{p_{final_i} \in P | (\mu_{k_n} - \alpha\sigma_{k_n}) \leq \bar{d} \leq (\mu_{k_n} + \alpha\sigma_{k_n})\}]. \quad (2.2)$$

onde α é um fator de restrição de densidade desejado.

2.2.3 Correção de densidade

A distribuição dos pontos que amostram a superfície da nuvem de pontos é chamada densidade de pontos. A densidade de amostragem dos pontos é importante no registro de superfície para definir uma vizinhança, que é o conjunto dos pontos próximos a um certo ponto que representam a geometria local daquela superfície. Uma vizinhança deve ser grande o suficiente para que os pontos nela contidos descrevam de forma satisfatória a geometria local, porém pequena o suficiente para que as características locais sejam mantidas (BERGER *et al.*, 2017).

Aquisições por sensores 3D normalmente produzem uma amostragem não uniforme na superfície, onde a densidade dos pontos varia espacialmente. Isso pode ser devido à distância do objeto em relação à posição do sensor, orientação do sensor ou pelas características geométricas da forma do objeto que se deseja adquirir a nuvem de pontos. Com isso, faz-se necessário um processo de equalização desses pontos, sem prejudicar a disposição de características e informações presentes na nuvem.

Como forma de equalizar a densidade dos pontos das nuvens pode ser usado o *voxel-grid*, onde cada *voxel* de volume arbitrário representa um conjunto de pontos. O centróide do *voxel* então passa a ser um ponto que representa a massa dentro daquele *voxel*, assim não prejudicando a disposição das características. A introdução do conceito de densidade é feita aqui visando um experimento futuro que mostra o quanto densidades diferentes afetam a qualidade do registro.

2.3 Transformações no espaço tridimensional

Nesse tópico será introduzida a base sobre as transformações no espaço tridimensional.

2.3.1 Orientação

Sendo um sistema de coordenadas com sua origem em O definido por segmentos de reta perpendiculares entre si, cada vetor \vec{OA} pode ser associado o seu ponto final A identificado por x, y, z . Assim, o ponto A pode ser representado como na Figura 4. Para denotar que um ponto

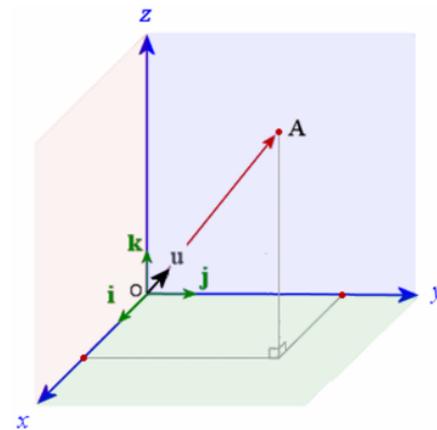
A pertence a um *frame* L , por exemplo, será usada a notação ${}^L A$, da mesma forma como foi usada por Craig (2014). Assim, a representação do ponto A pode se configurar da seguinte forma:

$${}^L A = \begin{bmatrix} A_x \\ A_y \\ A_z \end{bmatrix}. \quad (2.3)$$

Pode-se, ainda, representá-lo em termos dos vetores unitários de L :

$$A = \begin{bmatrix} A_x \\ A_y \\ A_z \end{bmatrix} \begin{bmatrix} \hat{X}_L & \hat{Y}_L & \hat{Z}_L \end{bmatrix} = A_x \hat{X}_L + A_y \hat{Y}_L + A_z \hat{Z}_L. \quad (2.4)$$

Figura 4 – Representação do ponto A no *frame* L .



Fonte: Bourne (2014).

Se tratando de corpos rígidos, a posição do centro de massa do corpo poderia denotar a sua posição em relação a um determinado *frame*.

Para descrever a orientação de um corpo, pode-se adicionar um sistema de coordenadas M ao corpo e então descrevê-lo em relação a L . Para isso, basta escrever M em termos dos vetores unitários de L . Desta forma, para qualquer vetor especificado no sistema M , é possível escrevê-lo, também, em termos dos componentes de L . Dessa forma, sendo \vec{P} um vetor em M ,

tem-se que sua representação em L é dada por:

$$\vec{P} = \begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix} \begin{bmatrix} {}^L\hat{X}_M & {}^L\hat{Y}_M & {}^L\hat{Z}_M \end{bmatrix} \quad (2.5)$$

$$= \begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix} \begin{bmatrix} \hat{X}_M\hat{X}_L & \hat{Y}_M\hat{X}_L & \hat{Z}_M\hat{X}_L \\ \hat{X}_M\hat{Y}_L & \hat{Y}_M\hat{Y}_L & \hat{Z}_M\hat{Y}_L \\ \hat{X}_M\hat{Z}_L & \hat{Y}_M\hat{Z}_L & \hat{Z}_M\hat{Z}_L \end{bmatrix}. \quad (2.6)$$

Na equação 2.6 a matriz 3x3 define a orientação de M em relação à L . Ela é denominada de matriz de rotação, ou seja, ${}^L_M R$. Uma das propriedades dessa matriz é que sua inversa é igual a transposta, ou seja, ${}^L_M R^{-1} = {}^L_M R^T = {}^M_L R$. Essa propriedade pode ser verificada analisando a variação dos índices dos vetores unitários da matriz da equação 2.6 e de sua transposta. De fato, sabe-se que a inversa de uma matriz com colunas ortonormais é igual a sua transposta (CRAIG, 2014).

2.3.2 Posição

Levando em consideração somente a posição do corpo no espaço (limitando-se ao seu centro de massa), podemos descrever a posição de \vec{P} (pertencente ao *frame* M , à priori) em L . Considerando que não há rotação entre os sistemas, os eixos de M são paralelos aos eixos de L , porém estão deslocados ao longo de um vetor entre as origens de seus sistemas, como na Figura 5a. Nesse caso, podemos escrever a posição de \vec{P} em relação a L em função do vetor entre as origens dos sistemas e o vetor \vec{P} em *frame* M (UICKER *et al.*, 2011). A Figura 5a mostra graficamente essa relação, assumindo o *frame* O como a referência (L) e Q sendo o deslocado (M). A Figura 5b, mostra a posição aparente do ponto \vec{P} antes da correção de orientação.

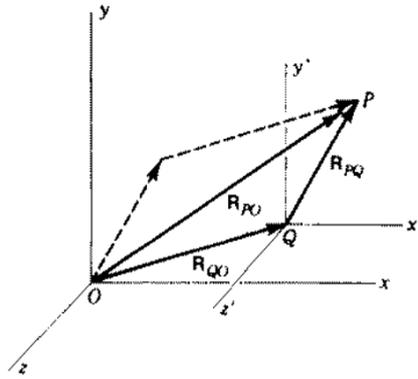
Assim, faz-se possível descrever a posição do ponto P no sistema de referência (L) como:

$${}^L P = {}^M P + \vec{V}_{ML}. \quad (2.7)$$

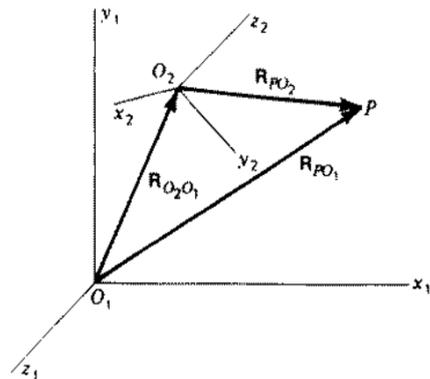
V_{ML} é o vetor formado pelas as origens dos sistemas M e L e pode ser denotado como o vetor de translação entre os *frames*. Um determinado sistema de coordenadas pode então ser descrito em relação a um sistema de referência por uma matriz de rotação R e um vetor de translação \vec{V} . Logo, para representar completamente um vetor \vec{P} de M em L , é necessário corrigir a orientação

Figura 5 – Translação.

(a) Diferença de posição



(b) Posição aparente

Fonte: Uicker *et al.* (2011).

e a posição:

$${}^L P = {}^L_M R \cdot ({}^M P) + \vec{V}_{ML}. \quad (2.8)$$

2.3.3 Matriz de transformação

Pode-se dizer que a descrição do *frame* M em relação ao L é dada por uma transformação ${}^L_M \Psi = \{{}^L_M R, \vec{V}_{ML}\}$. Ψ pode ser organizada na forma de uma matriz 4x4, de forma que em uma só matriz tem-se a matriz de orientação e o vetor de translação bem definidos da seguinte forma:

$${}^L_M \Psi = \begin{bmatrix} \hat{X}_M \hat{X}_L & \hat{Y}_M \hat{X}_L & \hat{Z}_M \hat{X}_L & (\vec{V}_{ML})_{\hat{X}_L} \\ \hat{X}_M \hat{Y}_L & \hat{Y}_M \hat{Y}_L & \hat{Z}_M \hat{Y}_L & (\vec{V}_{ML})_{\hat{Y}_L} \\ \hat{X}_M \hat{Z}_L & \hat{Y}_M \hat{Z}_L & \hat{Z}_M \hat{Z}_L & (\vec{V}_{ML})_{\hat{Z}_L} \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (2.9)$$

Craig (2014) mostra as possíveis configurações que a matriz de rotação pode ter se rotacionada de certos ângulos em torno de cada eixo. É definida uma matriz com todos os seus parâmetros em função dos três ângulos de rotação (α, β, γ) , um para cada eixo. Como exemplo de três rotações sequenciais sobre ângulos fixos, tem-se:

$${}^L_M R_{XYZ}(\gamma, \beta, \alpha) = R_Z(\alpha) R_Y(\beta) R_X(\gamma) \quad (2.10)$$

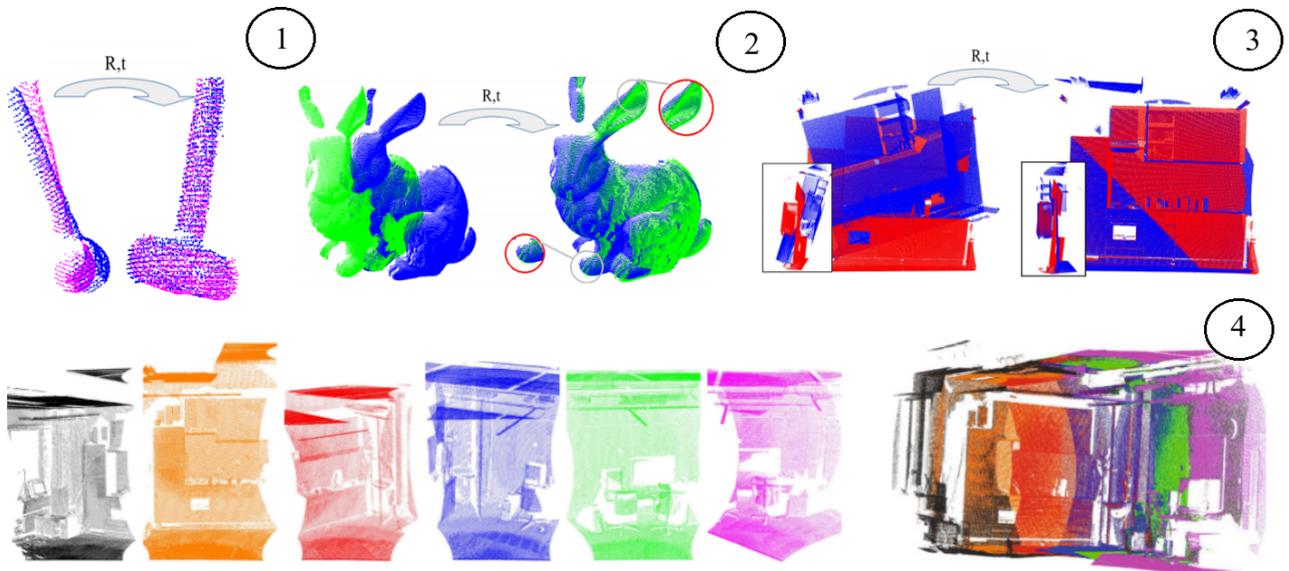
$$= \begin{bmatrix} \cos(\alpha) & -\sin(\alpha) & 0 \\ \sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos(\beta) & 0 & \sin(\beta) \\ 0 & 1 & 0 \\ -\sin(\beta) & 0 & \cos(\beta) \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\gamma) & -\sin(\gamma) \\ 0 & \sin(\gamma) & \cos(\gamma) \end{bmatrix}$$

$$= \begin{bmatrix} \cos(\alpha) \cos(\beta) & \cos(\alpha) \sin(\beta) \sin(\gamma) - \sin(\alpha) \cos(\gamma) & \cos(\alpha) \sin(\beta) \cos(\gamma) + \sin(\alpha) \sin(\gamma) \\ \sin(\alpha) \cos(\beta) & \sin(\alpha) \sin(\beta) \sin(\gamma) + \cos(\alpha) \cos(\gamma) & \sin(\alpha) \sin(\beta) \cos(\gamma) - \cos(\alpha) \sin(\gamma) \\ -\sin(\beta) & \cos(\beta) \sin(\gamma) & \cos(\beta) \cos(\gamma) \end{bmatrix}.$$

2.4 Registro de nuvens de pontos

O problema de alinhar consistentemente várias visualizações de dados de nuvem de pontos 3D em um modelo completo é conhecido como registro. O registro de figuras 3D consiste em, dado um par de nuvens de pontos, uma *source* S e uma *target* T , em posturas iniciais arbitrárias, o problema é encontrar a melhor transformação rígida ψ , tal que $\psi(S) \approx T$ represente o alinhamento sob uma métrica, geralmente a distância euclidiana. Alguns exemplos de registro de nuvens de pontos pode ser visto na Figura 6 para 4 modelos diferentes. Do lado esquerdo temos as duas nuvens de pontos antes do alinhamento sob modelos adquiridos com uma diferença de orientação e do lado direito o resultado depois do alinhamento.

Figura 6 – Registro de vários modelos de nuvens de pontos Na parte superior, indicado pelos números 1,2 e 3 o registro de pares de nuvens e na parte inferior o registro de múltiplas tomadas parciais de uma cena *indoor*.



Fonte: o autor.

Os métodos de registro são divididos em escolas baseadas no estudo de todos os pontos da nuvem, onde encontra-se técnicas como o ICP e suas variantes e a escola baseada no alinhamento inicial no espaço de características (*coarse alignment*). A seguir, algumas técnicas são comentadas e sua formulação matemática é brevemente explicada com a finalidade

de entender os métodos de comparação no Capítulo 4, já que tratam-se dos mesmos algoritmos.

2.4.1 Iterative Closest Point (ICP)

O algoritmo ICP foi proposto em Besl e McKay (1992) e visa encontrar uma transformação que otimize uma rotação e translação em dois conjuntos de dados (conjuntos de segmentos de linha, curvas implícitas, conjuntos de triângulos, superfícies implícitas, paramétricas, conjuntos de pontos, etc).

O algoritmo ICP usa um dos conjuntos de dados como referência, realizando rotações e translações em outro conjunto, a fim de minimizar a seguinte função de custo:

$$F(\vec{q}) = \frac{1}{N} \sum_{i=1}^N \| \vec{t}_i - (R(\vec{q}_R)\vec{s}_i + \vec{q}_T) \| \quad (2.11)$$

onde

- N=número de pontos;
- \vec{t}_i = i-ésimo vetor relacionado a nuvem de pontos *target*;
- \vec{s}_i =i-ésimo vetor relacionado a nuvem de pontos *source*;
- R= matriz de rotação obtida com o ICP;
- \vec{q}_T = vetor de translação obtido com o ICP.

O método ICP é baseado na análise do vetor quaternion \vec{q} , que é uma composição $\vec{q} = [\vec{q}_R | \vec{q}_T]^t$, onde $\vec{q}_T = [q_0, q_1, q_2, q_3]$ representa o vetor que mapeia a rotação e $\vec{q}_R = [q_4, q_5, q_6]^t$ é o vetor responsável pela translação. O vetor \vec{q}_R representa o autovetor correspondente ao máximo autovalor da matriz $Q(C_{ST})$ (abaixo), onde a parte C_{ST} representa a matriz de covariância cruzada dos conjuntos de dados de entrada.

$$Q(C_{ST}) = \begin{bmatrix} tr(C_{ST}) & \Delta^t \\ \Delta & C_{ST} - C_{ST}^t - Q_{11}I_{3 \times 3} \end{bmatrix} \quad (2.12)$$

Onde Δ é um vetor coluna das componentes cíclicas da matriz $A_{ij} = (C_{ST} - C_{ST}^t)$. Os valores de \vec{q}_R são obtidos a fim de aplicá-los a R e obter a rotação ideal em cada iteração seguindo a lógica de HORN (1987).

$$R = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & \cdots & 2(q_1q_3 + q_0q_2) \\ 2(q_1q_2 + q_0q_3) & \ddots & 2(q_2q_3 - q_0q_1) \\ 2(q_1q_3 - q_0q_2) & \cdots & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix} \quad (2.13)$$

Por fim, o vetor \vec{q}_T representa a translação óti:

$$\vec{q}_T = \vec{\mu}_T - R(\vec{q}_R\vec{\mu}_S), \quad (2.14)$$

onde $\vec{\mu}_S$ e $\vec{\mu}_T$ representam o centroide dos conjuntos de dados.

Após cada iteração, as melhores correspondências são encontradas produzindo progressivamente o registro. Isso é repetido até que o algoritmo chegue a sua convergência. Isso acontece se a distância entre dois conjuntos de dados é inferior a um determinado limiar calculado, chamado Erro Quadrático Médio (MSE). O resultado obtido a partir do algoritmo ICP é a rotação e a traslação ótimas entre os dois conjuntos de dados.

Analisando o ICP como um conjunto de etapas, podemos descrevê-lo como a seguir.

1. Seleção de pontos: É onde é feita a seleção de um conjunto de pontos de apenas um ou de ambos os conjuntos pontos de entrada. Esta primeira fase visa reduzir o número de pontos das nuvens aplicando métodos de sub-amostragem. A forma como esses pontos são selecionados tem um impacto direto na convergência do algoritmo que fará uso deles, e especialmente no custo computacional para a convergência, principalmente quando os conjuntos de dados são bastante densos. Alguns exemplos de formas de seleção são as variantes de amostragem em Rusinkiewicz e Levoy (2001).
2. Correspondência: Este passo representa a operação chave no algoritmo ICP. Consiste em sobrepor pontos que são correspondentes entre as duas nuvens. Essas correspondências são obtidas buscando para cada ponto da nuvem *source* o ponto mais próximo na nuvem *target*. Várias técnicas de busca pelo vizinho mais próximo são usadas para reduzir o custo computacional dessa etapa, que pelo menos no ICP o custo é da ordem quadrática do número de pontos. Dentre as várias opções de otimizar essa etapa, é afirmado pelos autores de Elseberg *et al.* (2012) que o uso de uma *kd-tree* é a melhor técnica para encontrar o vizinho mais próximo.
3. Ponderação dos pesos das correspondências: É aqui onde acontece a definição de um peso especial para alguns pares de pontos o que acaba mitigando ou, pelo menos, reduzindo combinações falsas (HOLZ *et al.*, 2015).
4. Rejeição de pontos: Rejeitar os pares de pontos que não contribuem positivamente para a convergência do algoritmo, como *outliers*, pontos oclusos (pontos que não são visíveis dependendo do *viewpoint*) ou não emparelhados (pontos de uma nuvem que não encontram correspondentes na segunda nuvem).
5. Métrica de erro: Define-se aqui a função objetivo que é minimizada em cada iteração do algoritmo. Há três opções, são as métricas ponto-a-ponto, plano-a-plano ou ponto-ao-plano (POMERLEAU *et al.*, 2013). Essas métricas é o que diferencia as variantes do ICP.

6. Minimização: Para se obter um registro satisfatório, deve-se minimizar uma métrica que represente um alinhamento adequado entre conjuntos de dados, sendo as métricas baseadas em distância (por exemplo o Raiz do Erro Quadrático Médio (RMSE)) as mais adotadas.

2.4.2 ICP ponto-ao-plano

O algoritmo ICP em Besl e McKay (1992) foi o pioneiro na arte de alinhar dados em perspectivas diferentes. O surgimento dessa técnica abriu margem para inúmeras abordagens para resolver a transformação rígida que minimiza o erro entre conjuntos de pontos. A matriz que representa a transformação rígida é composta por uma rotação R e uma translação t .

Chen e Medioni (1992) introduziram a métrica ponto-ao-plano presente na Equação 2.15 e provaram que esta métrica era estável. O cálculo é feito usando a distância entre o ponto de da nuvem *source*, s_i , o plano descrito pelo ponto t_i da nuvem *target* e a normal da sua superfície local n_{t_i} .

$$F_{p2p} = \sum_{i=1}^N \left\| \psi(\vec{s}_i - \vec{t}_i) \cdot n_{t_i} \right\|^2 \quad (2.15)$$

Ao contrário da métrica ponto-a-ponto, ela não tem uma solução de forma fechada⁵, então a minimização é feita com *solvers* não lineares (como Levenberg-Marquadt em Fitzgibbon (2003), ou por linearização como em Low (2004) (onde pequenos ângulos de rotação satisfazem a premissa que $\sin\theta \approx \theta$ e $\cos\theta \approx 1$). Essa linearização da matriz de rotação pode ser vista na Equação 2.16.

$$R(\alpha, \beta, \gamma) \approx \begin{bmatrix} 1 & -\gamma & \beta \\ \gamma & 1 & -\alpha \\ -\beta & \alpha & 1 \end{bmatrix}. \quad (2.16)$$

Dependendo da superfície e da distribuição de pontos, usando a métrica de erro ponto-a-plano pode-se obter um resultado consideravelmente mais robusto. Um procedimento padrão para minimizá-lo é usar a otimização em Fitzgibbon (2003). Nessa solução são acumuladas as restrições ponto-a-plano de todas as correspondências em uma matriz G e estima-se a transformação rígida resolvendo um sistema linear como na Equação 2.17.

$$G^T \cdot G \cdot \tau = G^T \cdot \rho \quad (2.17)$$

⁵ somente pode ser expressa em termos de um número delimitado de certas funções bem conhecidas.

onde τ é uma representação parametrizada dos 6Graus de liberdade (DoF) da transformação minimizada, vista na Equação .

$$\psi = \tau = \begin{bmatrix} \alpha \\ \beta \\ \gamma \\ t_x \\ t_y \\ t_z \end{bmatrix} \quad (2.18)$$

Os valores de G e b são vistos nas Equações 2.19 e 2.20 abaixo:

$$G = \begin{bmatrix} \vec{t}_1 \cdots n_I \end{bmatrix}. \quad (2.19)$$

que é uma matriz $6 \times i$ e

$$\rho = \begin{bmatrix} \vdots \\ (\vec{s}_i - \vec{t}_i) \cdot n_{t_i} \\ \vdots \end{bmatrix}. \quad (2.20)$$

é um vetor coluna de i elementos.

2.4.3 Generalized ICP (GICP)

O *Generalized ICP*, proposto por Segal *et al.* (2009), é uma variante do ICP que possui um embutido um modelo probabilístico para a etapa de minimização. A técnica mantém o resto do algoritmo original. Notavelmente, as correspondências são ainda computadas com a distância euclidiana padrão, em vez de uma medida probabilística. Aqui é assumido que os pontos tem um correspondente na outra nuvem e que *outliers* não estão presentes.

No modelo probabilístico, assumimos a existência de um conjunto de pontos subjacentes, \hat{S} e \hat{T} , que geram S e T de acordo com $s_i \approx N(\hat{s}_i, C_i^S)$ e $t_i \approx N(\hat{t}_i, C_i^T)$. Neste caso, C_i^S e C_i^T são as matrizes de covariância associadas aos pontos medidos.

Para uma transformação rígida arbitrária, ψ , definimos $d_i^\psi = t_i - \psi * s_i$, e considera-se a distribuição $d_i^{(\psi^*)}$. Como s_i e t_i são assumidos como sendo extraídos de gaussianas independentes,

$$\begin{aligned} &= d_i^{(\psi^*)} \approx N(t_i - (\psi^*)s_i, C_i^T + (\psi^*)C_i^S(\psi^*)^T) \\ &= N(0, C_i^T + (\psi^*)C_i^S(\psi^*)^T) \end{aligned} \quad (2.21)$$

Agora usamos o MLE⁶ para calcular iterativamente ψ pela Equação 2.22

$$\begin{aligned}
&= \psi = \max_{\psi} \prod_i p_d(d_i^{\psi}) \\
&= \max_{\psi} \sum_i \log(p_d(d_i^{\psi})) \\
&= \min_{\psi} \sum_i (d_i^{\psi T} (C_i^T + (\psi^*) C_i^S(\psi))^{-1} d_i^{\psi})
\end{aligned} \tag{2.22}$$

Se a matriz de covariância do conjunto que sofre a transformação for igual a identidade e do outro for uma matriz nula, toda essa formulação se reduz ao ICP em Besl e McKay (1992). Com a estrutura demonstrada acima, o conjunto de covariâncias é uma escolha flexível para C_i^S e C_i^T . Para o algoritmo ICP ponto-a-plano também pode ser feita uma adaptação para uma forma probabilística, como visto na Equação 2.23:

$$\psi = \min_{\psi} \left(\sum_i \| P_i d_i \|^2 \right) \tag{2.23}$$

, onde P_i é a projeção da normal em t_i . Isso minimiza a distância da transformação de s_i para por t_i e sua normal. P_i é uma matriz de projeção ortogonal, $P_i = P_i^2 = P_i^{\psi}$. Isso significa que $\| P_i \cdot d_i \|^2$ pode ser reformulado na forma da Equação 2.24

$$\begin{aligned}
&= \| P_i \cdot d_i \|^2 = (P_i \cdot d_i)^T \cdot (P_i \cdot d_i) \\
&= d_i^T P_i \cdot d_i
\end{aligned} \tag{2.24}$$

Assim como na observação feita do ICP clássico como um caso específico do GICP, pode ser feita a mesma afirmação para o ICP ponto-ao-plano. No caso onde $C_i^S = P_i^{-1}$ e $C_i^T = 0$.

2.4.4 Normal Distribution Transform (3D-NDT)

O NDT foi proposto em Biber e Straßer (2003) para o registro de figuras parciais para propósitos com dados em duas dimensões e posteriormente foi estendido para aplicações com nuvens de pontos por Magnusson *et al.* (2007). Diferente de outras abordagens que usam a nuvem de pontos inteira ou *keypoints* para realizar o registro, aqui pressupõe-se a distribuição gaussiana local de pontos e usa a .

Para registrar duas nuvens de pontos primeiro, o espaço é discretizado em *voxels* ou células (para mais detalhes ver Magnusson *et al.* (2007)). Seja V_i o conjunto de pontos v de t em

⁶ é um método de estimação dos parâmetros de um modelo estatístico. Dado um conjunto de dados para um modelo estatístico, a estimativa por máxima verossimilhança estima valores para os diferentes parâmetros do modelo (CASELLA; BERGER, 2002).

um *voxel* i . Uma distribuição gaussiana de pontos é assumida para cada *voxel*, resultando nos conjuntos de distribuições G_T e G_S .

O vetor médio $\vec{\mu}_i$ e a matriz de covariância C_i para cada distribuição são estimados de acordo com as Equações 2.25 e 2.26, onde desenvolvidas essas equações determinamos a PDF de cada *voxel* na Equação 2.27 dada por:

$$\mu_i = \frac{1}{|V_i|} \sum_{v \in V_i} v, \quad (2.25)$$

$$C_i = \frac{1}{|V_i| - 1} \sum_{v \in V_i} (v - \mu_i)(v - \mu_i)^T, \quad (2.26)$$

$$p_i(x) = \frac{1}{\sqrt{(2\pi)^3 |C_i|}} \exp\left(-\frac{(x - \mu_i)^T C_i^{-1} (x - \mu_i)}{2}\right). \quad (2.27)$$

Seja ψ a matriz de transformação de 6DoF de S para T , com R e t os componentes de rotação e translação respectivamente. A distância entre duas transformações i, j é definido como

$$d(i, j) = -d_1 \exp\left(-\frac{d_2}{2} \mu_{i,j} (R^T C_i R + C_j)^{-1} \mu_{ij}\right) \quad (2.28)$$

e a transformação que minimiza a função é dada por $F(\psi)$ na Equação 2.29

$$F(\psi) = \sum_{i=1, j=1}^{|G_S|, |G_T|} \text{dist}(i, j), \quad (2.29)$$

onde $\mu_{ij} = R\mu_i + t - \mu_j$ e d_1, d_2 são fatores de regularização.

2.4.5 Sparse ICP

Os algoritmos de registro visam minimizar uma métrica de distância entre duas superfícies S e T . Esse problema pode ser formulado como

$$\min_{R, t} \int_{\vec{s}_i \in S} \phi(R\vec{s}_i + t, T) + I_{SO(3)}(R), \quad (2.30)$$

onde $R \in R^{3 \times 3}$ denota a matriz de rotação e $t \in R^3$ é o vetor de translação da transformação de registro. A matriz R é restrita ao grupo ortogonal especial $SO(3)$ usando a função indicadora $I_{SO(3)}(S)$, que tende para zero quando $S \in SO(3)$ e para $+\infty$, quanto mais rigidez houver na

transformação. A função $\phi(\vec{s}_i, T)$ mede a distância de um ponto arbitrário $s \in R^3$ para a superfície T e é definido como

$$\phi(\vec{s}_i, T) = \min_{\vec{s}_i \in T} \phi(\vec{s}_i, \vec{t}_i), \quad (2.31)$$

onde a métrica $\phi(\vec{s}_i, \vec{t}_i)$ mede a distância entre dois pontos no espaço. Esta equação é frequentemente referida como transformação de distância da superfície T .

Na maioria das aplicações, geralmente lidamos com superfícies discretizadas, representadas por um conjunto de pontos $\vec{s}_i \in S$, onde $i = 1, \dots, n$. Neste caso, podemos definir o vetor de distâncias residuais $\vec{z} \in R^n$, cujos elementos são definidos como:

$$z_i(R, t) = \phi(Rs_i + t, T) + I_{SO(3)}(R), i = 1, \dots, n. \quad (2.32)$$

Usando esta definição, a Equação 2.33 pode ser reescrita como:

$$\min_{R, t} \sum_{i=1}^n z_i(R, t). \quad (2.33)$$

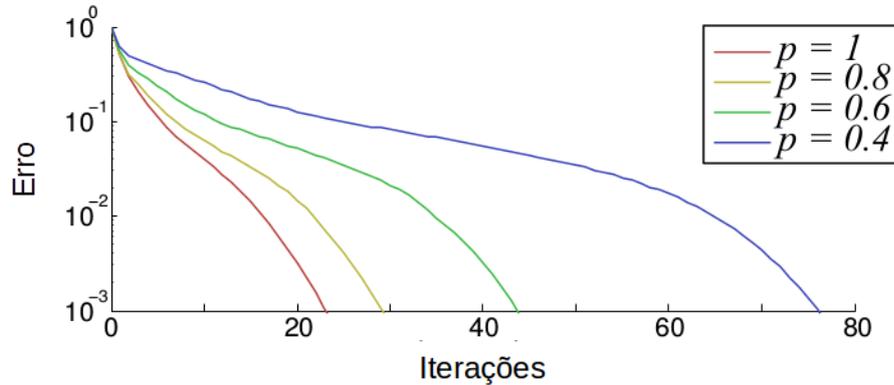
Muitos métodos usam a norma euclidiana como a métrica de distância. No entanto, na presença de *outliers*, um otimizador distorcerá a solução para reduzir o efeito associado a pontos distantes da distribuição. Para evitar esse problema, é definida a distância na Equação 2.34 como:

$$\phi(\vec{s}_i, \vec{t}_i) = \mu_p(\|\vec{s}_i - \vec{t}_i\|) \quad (2.34)$$

Esta formulação usando a norma p , para $p \in [0, 1]$, impõe uma penalidade menor para pontos a distâncias grandes em relação ao que acontece com outras normas, tornando a otimização resiliente a *outliers*.

Um método para resolver o problema de otimização que descrevemos aqui é usar o *Sparse* em Bouaziz *et al.* (2013). Em cada iteração do ICP, a distância residual diminui lentamente, assim o algoritmo fornece uma convergência garantida para um mínimo local. No entanto, a taxa de convergência real depende da norma usada durante a otimização. Nestes casos, como a penalidade induzida pelos *outliers* se torna menor, a inclinação da função objetivo é reduzida e o otimizador é forçado a executar etapas menores. Isso é claramente mostrado na Figura 7, onde a taxa de convergência do *Sparse* ICP é mostrada. No caso, observamos que a diminuição do valor de p reduz significativamente a taxa de convergência do método.

Figura 7 – Uma análise da taxa de convergência do *Sparse* para diferentes valores de p . Quando p diminui, o número de iterações do algoritmo ICP aumenta.



Fonte: Adaptado de Bouaziz *et al.* (2013).

Nos experimentos em Bouaziz *et al.* (2013), são mostrados os desempenhos do algoritmo para variações de p , tentando estabelecer um consenso entre custo computacional e qualidade de alinhamento. Os autores colocam que a menos que indicado de outra forma, selecionaram $p = 0,4$ como forma de oferecer uma solução com um bom *trade-off* entre quantidade de iterações e erro. Essa mesma tomada de decisão é feita neste trabalho, a fim reproduzir os resultados e gerar uma justa comparação.

2.4.6 Globally Optimally ICP (GoICP)

Em Yang *et al.* (2015) é proposto o que os autores chamam do primeiro algoritmo de registro realmente de solução puramente global. Em vez de procurar o espaço 6D com um único otimizador BnB⁷ que seria menos eficiente, os autores propuseram estrutura de pesquisa BnB conjuntos. Um BnB externo que pesquisa o espaço de rotação, enquanto resolve os limites de erro e a translação correspondente é encontrada por um BnB interno.

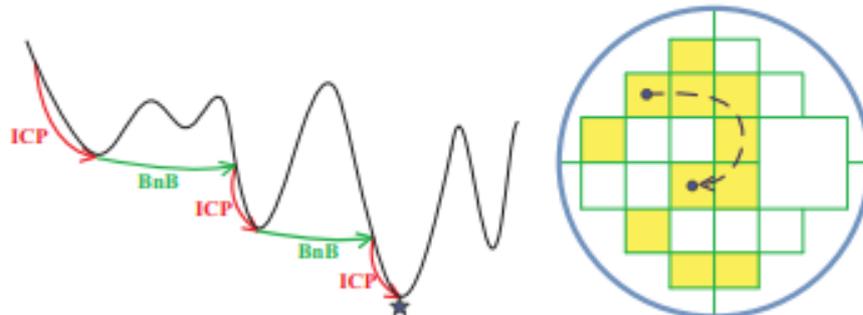
Na rotação, para um cubo C_r , os limites superior e inferior do erro de registro podem ser dados por $\overline{E}_r = \min_{\forall t \in C_t} \sum_i e_i(R_{r_o}, t)^2$ e $\underline{E}_r = \min_{\forall t \in C_t} \sum_i (e_i(R_{r_o}, t) - \gamma_{r_i}, 0)^2$ onde C_t é o cubo de translação. Para resolver \underline{E}_r com o BnB interno, os limites para um cubo de translação C_t podem ser escolhidos como $\underline{E}_t = \max \sum_i (e_i(R_{r_o}, t) - \gamma_{r_i}, 0)^2$ e $\overline{E}_t = \max \sum_i (e_i(R_{r_o}, t) - (\gamma_{r_i} + \gamma_t), 0)^2$ (para mais informações ver Yang *et al.* (2015), seção 4.2, corolário 3). Definidos todos os raios de incerteza de rotação γ_{r_i} para zero em \underline{E}_t e \overline{E}_t , obtemos a translação para resolver \overline{E}_r .

⁷ é um algoritmo que enumera os candidatos a solução de um problema de otimização, por meio do qual subconjuntos de candidatos ruins a solução são descartados por meio da aferição de limites superior e inferior da função otimizada (YANG *et al.*, 2015).

Especificamente, cada um dos BnBs têm sua fila de prioridade Q_r , Q_t , respectivamente. A prioridade dos cubos é o oposto do limite inferior, o que significa que os BnBs sempre exploram o cubo de menor limite inferior. Sempre que a diferença entre o melhor erro E_{best} e o limite \underline{E} for inferior a um limiar ε , o BnB para.

Para melhorar a velocidade do BnB interno, são fornecidos um conjunto iniciais E_{t-best} . Se $\overline{E}_r = E_{t-best} \geq E_{best}$, E_{best} não será atualizado. Se $\underline{E}_r = E_{t-best} \geq E_{best}$, significa que foi encontrado o C_r com a melhor solução, onde o erro E_{t-best} que foi fornecido pelo BnB interno é melhor ou pelo igual ao erro E_{best} no BnB de fora. O domínio do ICP integrado no método são todos os cubos onde o limite é menor que E_{best} . Aqui o ICP funciona como um complemento da etapa de registro a partir de um conjunto de soluções. Na Figura 8, temos uma ilustração que esboça a explicação sobre a integração do método.

Figura 8 – À esquerda o BnB e o ICP atualizam em colaboração os limites superiores e na direita com a orientação vinda do BnB, o ICP explora apenas um conjunto promissor de limites inferiores.



Fonte: Adaptado de Yang *et al.* (2015).

2.4.7 Cloud Partitioning ICP (CP-ICP)

Visando uma redução de custo computacional, em Pereira *et al.* (2015) foi proposto o método *Cloud Partitioning Iterative Closest Point* (CP-ICP), baseado no particionamento de nuvens de pontos. A idéia por trás do CP-ICP é separar a nuvem completa em grupos menores (k grupos no total) designados aqui sub-nuvens, e depois repetir o processo de registro para cada par de segmentos até que um critério de parada baixo seja encontrado. Ao fazer isso, reduz-se o custo da etapa de busca pelo vizinho mais próximo porque as nuvens de pontos são de tamanho reduzido.

Para obter uma melhor visão sobre a afirmação acima, considere N_S e N_T sendo o tamanho de duas nuvens sob o registro pelo ICP. O custo computacional para o cálculo do ponto

mais próximo no ICP clássico é $O(N_S N_T)$, enquanto com o particionamento em sub-nuvens é menor e pode variar entre $O(N_S N_T / k^2)$ e $kO(N_S N_T / k^2)$. Nesta análise, é considerado apenas o cálculo do ponto mais próximo, porque é o passo mais cara computacionalmente do algoritmo ICP.

Para aplicarmos o método do CP-ICP, são seguidos os seguintes passos:

1. Divide-se cada conjunto de dados em k sub-nuvens;
2. Para cada uma das k iterações, calcula-se o algoritmo ICP para um par de sub-nuvens (considere os pontos abaixo sub-itens do passo 2):
 - Aplica-se a transformação obtida para a nuvem de pontos *source* (nomeada a partir de agora como nuvem de entrada pós-ICP), corrigindo, assim, o alinhamento;
 - A nuvem de entrada pós-ICP é então comparada com a nuvem *target*, utilizando o RMSE a cada iteração;
 - Caso o valor de RMSE obtido numa dada iteração seja inferior ao critério de parada, o algoritmo encerra e o registro desejado é obtido.

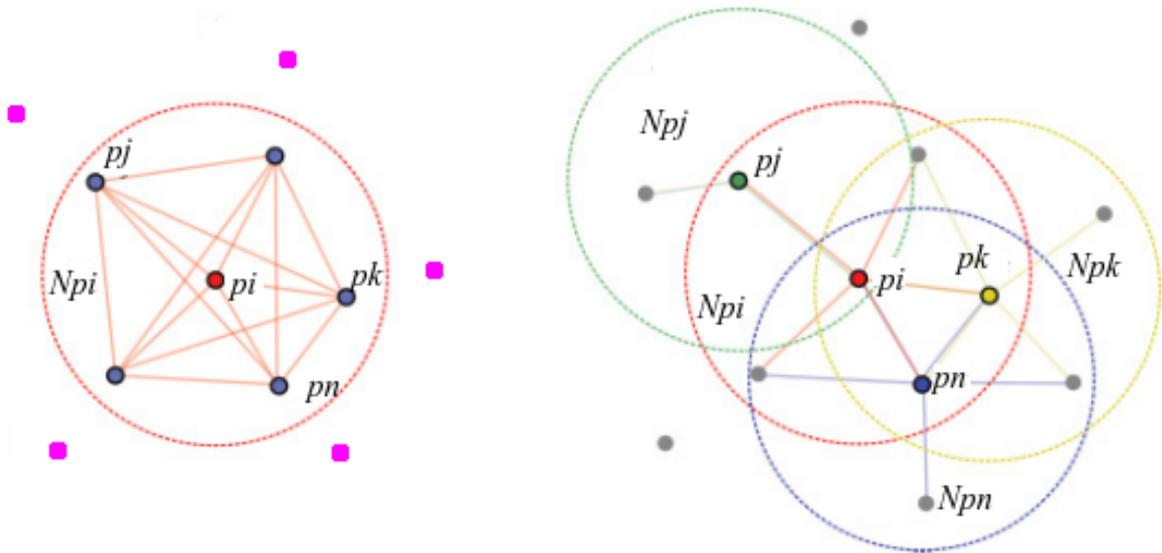
2.4.8 Fast Point Feature Histogram (FPFH)

O método PFH foi apresentado por Rusu *et al.* (2008). Este método consiste em extrair informações geométricas da vizinhança de um ponto. Dado um ponto de consulta p_i a partir de uma nuvem de pontos P , uma esfera de raio r envolve a vizinhança k_{p_i} . Todos os pontos que estão dentro da esfera estão conectados com os outros através de malha interconectada (ver Figura 9). Para cada ponto $p_j \in N_{p_i}$ com um vetor normal n_j , o algoritmo seleciona outro ponto $p_k \in N_{p_i}$ onde o ângulo entre n_j o vetor definido por $(p_k - p_j)$ é mínimo. Basicamente, isso significa que o algoritmo está focado em zonas côncavas da forma. Para cada par de pontos p_j e p_k , um referencial chamado *frame* de Darboux (u, v, w) é calculado ($\vec{u} = \vec{n}_j, \vec{v} = (p_k - p_j) \times \vec{u}, \vec{w} = \vec{u} \times \vec{v}$). Então a informação angular é calculada com estas funções:

$$\begin{aligned}
 f_1 &= \alpha = v \cdot n_k; \\
 f_2 &= d = \|p_k - p_j\|; \\
 f_3 &= \phi = u \cdot (p_k - p_j) / d; \\
 f_4 &= \theta = \text{atan}(w \cdot n_k, u \cdot n_k).
 \end{aligned} \tag{2.35}$$

Finalmente, o algoritmo constrói um histograma dividido em compartimentos, onde o espaço de caixa é organizado cobrindo todos os valores das características. Para cada ponto

Figura 9 – À esquerda é uma representação do PFH e à direita a representação do FPFH (onde cada ponto usa apenas seus vizinhos diretos para calcular seu próprio SPFH).



Fonte: Adaptado de Diez *et al.* (2015).

de consulta p_i , um histograma do descritor é calculado de acordo com o valor da informação angular de cada par (p_j, p_k) .

A principal desvantagem do PFH é portanto sua alta complexidade $O(Nk_n^2)$, onde N é o número de pontos da nuvem e k_n o número de vizinhos de cada ponto. Por esta razão, em Rusu *et al.* (2009), esse custo foi reduzido para $O(Nk_n)$. A primeira modificação é a redução do espaço de características de 16D para 8D, pois é computado somente f_1, f_3 e f_4 do conjunto de Equações em 2.35. Esse conjunto foi chamado de *Simplified Point Feature Histogram* (SPFH).

Para cada ponto é computado o SPFH somente entre o ponto em questão e seus k_n -vizinhos (não são todos os pares). Após isso, para cada ponto, os seus k_n -vizinhos são determinados novamente e o FPFH é computado levando em consideração o SPFH do ponto atual e o das regiões vizinhas:

$$FPFH(p) = SPFH(p) + \frac{1}{k} \sum_{i=1}^k \frac{1}{w_{k_n}} \cdot SPFH(p_k). \quad (2.36)$$

, onde w_k representa a distância entre o ponto atual (p_i) e o ponto mais próximo (p_k).

Em Costanzo *et al.* (2018) propõe uma aplicação de identificação de objetos em prateleiras de supermercados e para tal, existe dentro do seu *pipeline* uma etapa de registro, a qual foi feita baseada numa abordagem usando FPFH para extração de característica, um alinhamento inicial baseado no RANSAC e o ICP que herda essa transformação inicial e refina a pose dos objetos.

O alinhamento inicial é obtido seguindo a abordagem em Buch *et al.* (2013), ou seja, o *Prerejectiv* RANSAC. Este método usa os descritores FPFH locais como os parâmetros da função *sample consensus*, e encontra os pontos e fornece a transformação geométrica. Esse alinhamento é calculado sem ter conhecimento *a priori* de objeto ou sua orientação (um ponto alto dos métodos esparsos).

O uso dessa classe pelos autores é devido no processo de classificação, ao reconhecer um modelo, a taxa de correspondência variava muito entre cenas e mostrava-se semelhante para objetos diferentes. Fora ainda identificados parâmetros críticos para a aplicação, caso do número de características mais próximas, Υ . Para encontrar o valor ideal adotaram:

$$[I, \psi_{RANSAC}] = RANSAC(S, T, \Upsilon), \quad (2.37)$$

e consideraram

$$fitness(\Upsilon) = \frac{I}{N_S}. \quad (2.38)$$

onde $|I|$ é a cardinalidade do conjunto I de *inliers* entre as duas nuvens de pontos após o alinhamento do RANSAC calculado usando o valor atual de Υ e ψ_{RANSAC} é uma matriz de transformação homogênea de ordem 4. Finalmente, como refinamento, o algoritmo ICP é aplicado a *pose* estimada pelo alinhamento com o FPFH-OPRANSAC. A *pose* final é estimada pela transformação com o ICP (ψ_{ICP}) a partir da transformação inicial usando RANSAC.

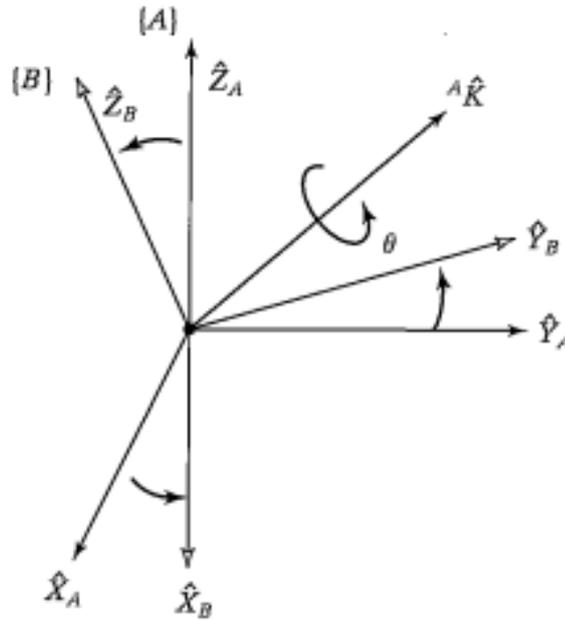
$$\Psi = \psi_{RANSAC} + \psi_{ICP}. \quad (2.39)$$

2.5 Representação eixo-ângulo equivalente

Este é um exemplo de uma representação eixo-ângulo equivalente. Se o eixo é uma direção geral (em vez de uma das direções Z,Y,X) qualquer orientação pode ser obtida através da seleção correta do eixo e do ângulo. Considerar a seguinte descrição de um *frame* B:

Na Figura 10 temos a representação eixo-ângulo:

Figura 10 – Representação eixo-ângulo.



Fonte: Craig (2014)

Se o eixo de rotação é um eixo geral, pode ser mostrado que a matriz de rotação equivalente é:

$$R_K(\theta) = \begin{bmatrix} k_x k_x v\theta + c\theta & k_x k_y v\theta - k_z s\theta & k_x k_z v\theta + k_y s\theta \\ k_x k_y v\theta - k_z s\theta & k_y k_y v\theta + c\theta & k_y k_z v\theta - k_x s\theta \\ k_x k_z v\theta - k_y s\theta & k_y k_z v\theta + k_x s\theta & k_z k_z v\theta + c\theta \end{bmatrix}, \quad (2.40)$$

onde: $c\theta = \cos(\theta)$, $s\theta = \sin(\theta)$, $v\theta = 1 - \cos(\theta)$ e ${}^A\hat{K} = [k_x, k_y, k_z]^T$. A equação 2.40 converte da representação eixo-ângulo para a representação da matriz de rotação. O resultado parcial da transformação é dado aqui pela Equação 2.41:

$${}^A_B R_K(\theta) = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}, \quad (2.41)$$

então

$$\theta = A \cos\left(\frac{\text{tr}({}^A_B R_K(\theta)) - 1}{2}\right), \quad (2.42)$$

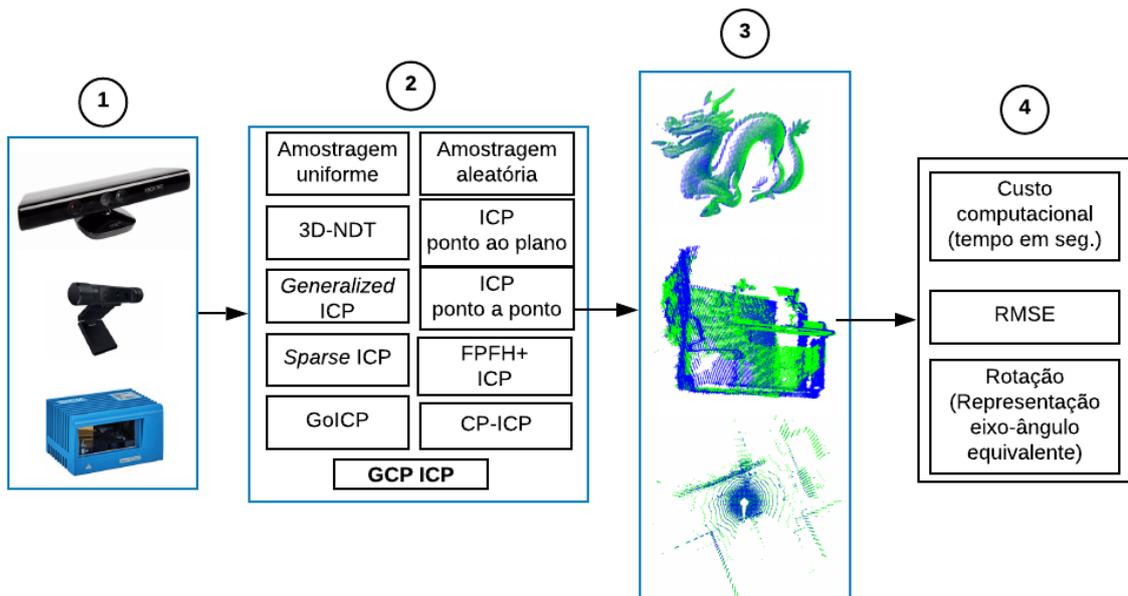
e

$$\hat{K} = \frac{1}{2\sin\theta} \begin{bmatrix} r_{32} - r_{23} \\ r_{13} - r_{31} \\ r_{21} - r_{12} \end{bmatrix}, \quad (2.43)$$

3 METODOLOGIA

Neste capítulo, é apresentada a metodologia, descrevendo os conjuntos de dados usados nos experimentos, bem como as eventuais modificações realizadas nos modelos de nuvens de pontos. Ainda neste capítulo são mencionadas brevemente os experimentos. Na Figura 11 é mostrada a lógica de experimentação, desde a aquisição dos dados até a avaliação quantitativa dos resultados.

Figura 11 – Metodologia adotada para comparação das variadas abordagens de registro. Em (1) os dados que são realizados o registro. Em (2) os métodos de registro. Em (3) o resultado do registro que passa por uma avaliação qualitativa (inspeção visual). Em (4) os resultados são comparados as métricas de avaliação quantitativas do erro de registro.

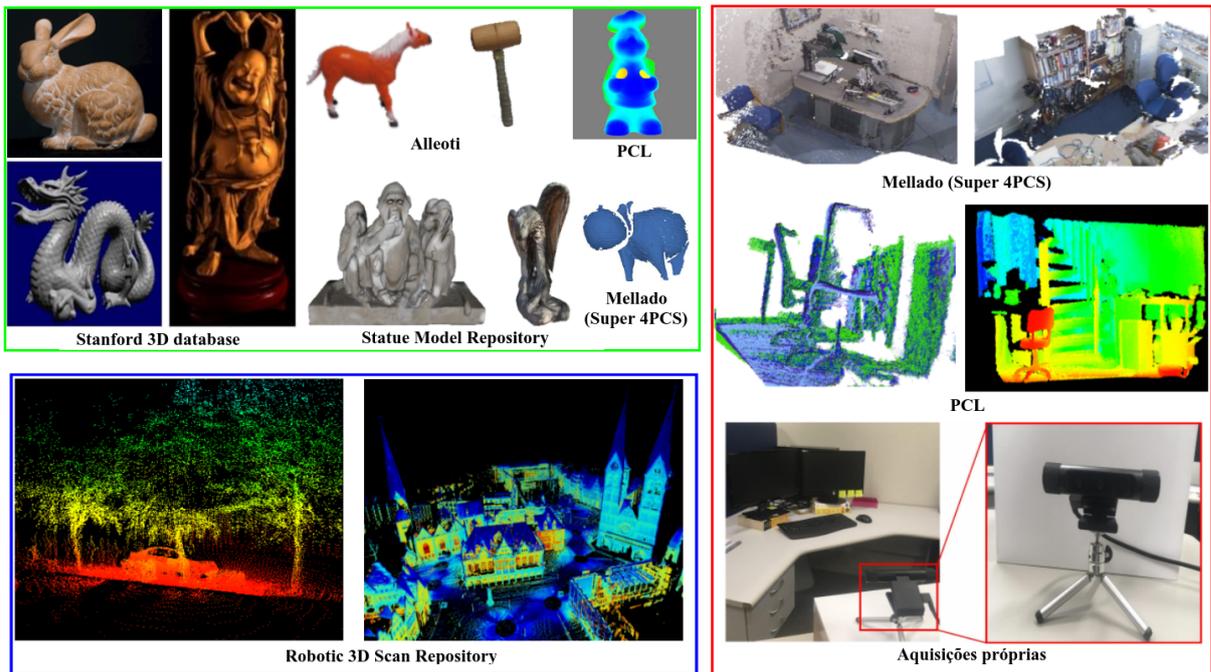


Fonte: o autor.

3.1 Conjuntos de dados disponíveis

A análise de desempenho do método proposto GCP-ICP foi realizada utilizando oito diferentes bases de dados. Todos os conjuntos de dados, juntamente com os modelos 3D obtidos de cada um deles, são mostrados na Figura 12.

Figura 12 – Conjuntos de dados usados nos 9 experimentos. Os conjuntos de dados estão divididos na ilustração de acordo com as aplicações: em verde os dados de objetos, em vermelho de cenas *indoor* e em azul de cenas *outdoor*.



Fonte: o autor.

Uma rápida sumarização dos conjuntos de dados e o conteúdo presente em cada um deles pode ser visto na Tabela 1.

Tabela 1 – Informações sobre os bancos de dados e modelos usados nos experimentos.

Tipo de nuvem	Nome	Rotação	Banco de dados
Objetos	<i>Bunny</i>	0/45	University (2014)
	<i>Dragon</i>	0/24/48	
	<i>Buddha</i>	0/24/48/72/96	
	<i>Horse</i>	0/180	Aleotti <i>et al.</i> (2014)
	<i>Hammer</i>	0/45	
Indoor	<i>Hippo</i>	rotações diversas sobre os três eixos principais (rotações sobre eixo genérico)	Mellado <i>et al.</i> (2014)
	<i>Chef</i>		PCL (2018)
	<i>Monkey</i>		Bouaziz <i>et al.</i> (2013)
	<i>Eagle</i>		
Indoor	<i>Lab.1</i>	-	O autor.
	<i>Lab.2</i>		
	<i>Office</i> <i>Stage</i>		Mellado <i>et al.</i> (2014)
Indoor	<i>House Office</i>	-	PCL (2018)
	<i>Lounge chair</i>		
Outdoor	<i>Hannover 1/4</i>	-	Wulf (2016)
	<i>Bremen1/2</i>		

Fonte: o autor.

3.2 Pré-processamento dos dados

Nesta seção são apresentados alguns pré-processamentos efetuados nas nuvens de pontos antes da realização dos experimentos.

3.2.1 Remoção de outliers

Pontos distantes da superfície da nuvem de pontos são classificados como *outliers*. Esses pontos têm coordenadas bem diferentes da distribuição dos demais pontos. Em alguns casos, são aleatoriamente distribuído no volume de dados, onde sua densidade é menor do que a dos pontos que realmente amostram a superfície.

A presença de *outliers* pode prejudicar a etapa de correspondência dos pontos, levando a um mau alinhamento. Para tal, nesta dissertação foi usado o método SOR descrito na Seção 2.2.2. Os conjuntos tratados foram o *Monkey* e os dados adquiridos com o sensor da Razer. Os parâmetros utilizados foram número $k_n = 50$ e $\sigma_{k_n} = 1$. A implementação do algoritmo é baseada naquela disponibilizada na biblioteca PCL (RUSU; COUSINS, 2011).

3.2.2 Redução da densidade das nuvens

Nesta seção foi feito uso do algoritmo *VoxelGrid*, seguindo a implementação presente na PCL (RUSU; COUSINS, 2011). O método implementado nesta classe cria uma grade de *voxels* 3D (limites formados por cubóides) sobre os dados da nuvem de pontos de entrada. Então, em cada *voxel*, os pontos presentes serão reduzidos e representados pelo seu centróide. Esse pré-processamento foi utilizado para gerar nuvens de pontos de densidades diferentes. Na Tabela 2 é feita a sumarização dos bancos de dados pré-processados e os tamanhos dos bancos de dados de dados e a diferença de densidade entre estes. O pré-processamento é realizado visando o último experimento desta dissertação.

Tabela 2 – Redução de densidade dos pontos.

	Nuvem		≈ %
	Menor	Maior	
<i>Bremen 1</i>	163.966	1.903.429	8,6
<i>Bremen 2</i>	697.209	1.903.429	37
<i>Hannover 3</i>	7.712	12.137	50
<i>Hannover 4</i>	6.087	12.186	63

Fonte: o autor.

3.3 Generalized Cloud Partitioning ICP(GCP-ICP)

Nesta seção é exposto o detalhamento do algoritmo proposto no âmbito desta dissertação, o *Generalized Cloud Partitioning Iterative Closest Point* (GCP-ICP). Este algoritmo é baseado no CP-ICP proposto por Pereira *et al.* (2015), onde aqui são feitas algumas contribuições a fim de automatizar e generalizar algumas etapas do algoritmo anteriormente proposto. Dentre as modificações estão a generalização do particionamento nos três eixos de coordenadas (X, Y e Z) e a aferição de um critério de parada automático variável para cada modelo a ser alinhado.

3.3.1 Visão geral das etapas do ICP no método proposto

Nesta seção é feita uma breve descrição das etapas de um algoritmo de alinhamento, tomando por base o modelo do ICP em Besl e McKay (1992), visando explicar como estas etapas acontecem dentro do método proposto.

3.3.1.1 Seleção

Como mencionado anteriormente, na correspondência ponto-a-ponto como nas variantes do ICP, os pontos passam por um processo para saber qual ponto é o mais representativo para uma certa vizinhança.

No entanto, a seleção desse ponto pode sofrer com a ambiguidade de se trabalhar com todo o conjunto de dados, além do limiar de distância ser muito pequeno e levar a falsas correspondências. Esse problema é parcialmente resolvido em Pereira *et al.* (2015) quando é feita uma seleção iterativa dos pontos a serem registrados por meio de uma etapa de particionamento. A acomodação dos pontos em porções menores leva a um custo $O(N_S N_T / k^2)$ da etapa de alinhamento para cada par de sub-nuvens. Porém, esse particionamento é feito apenas no eixo Z (PEREIRA, 2016), o que não garante que a maior distribuição dos dados ou que os cortes que beneficiem o alinhamento sejam feitos naquele eixo .

Visando a generalização desse particionamento, na variante atual do CP-ICP o eixo de corte fica a critério do usuário da aplicação, podendo ser feito em qualquer um dos eixos principais (ortogonais), que é portanto a primeira contribuição trazida nesta dissertação. Essa generalização é o que dá o prefixo *Generalized* ao nome do CP-ICP.

Aqui a etapa de seleção consiste em duas sub-tarefas: (1) particionamento em um eixo genérico ξ e (2) agrupamento dos pontos de cada partição. A primeira tarefa executa a

divisão espacial da nuvem de pontos em k sub-nuvens. A segunda agrupa todos os pontos de cada sub-nuvem mediante os seus índices, onde para um número k de partições, onde o índice de cada sub-nuvem é j , um ponto de índice \vec{s}_i ou \vec{t}_m , respectivamente um ponto nas nuvens *source* ou *target* deve no processo de subdivisão está presente apenas numa dada sub-nuvem j , não estando nos intervalos que representam outra sub-nuvem com índice diferente, caso existam.

3.3.1.2 Correspondência

Visando a redução de custo computacional, em especial da etapa de correspondências do ICP clássico, o algoritmo proposto faz uso do ICP da PCL (RUSU; COUSINS, 2011), que conta com uma etapa de busca baseada no *kd-tree* (ELSEBERG *et al.*, 2012) para reduzir o custo computacional já compactado pela etapa de particionamento.

3.3.1.3 Métrica de erro e minimização

Aqui a métrica utilizada é a ponto-a-ponto. O registro computado entre as sub-nuvens passa por uma verificação de qualidade baseada num valor restritivo de critério de parada, obtido pelo micro-desalinhamento ente a nuvem *target* (completa) e uma cópia levemente rotacionada sucessivamente em combinações dos eixos principais.

O critério de parada restritivo e baseado nas características de cada modelo leva o algoritmo a encontrar uma solução ótima, podendo inclusive acelerar a verificação entre as k soluções possíveis, não sendo necessário percorrer todas as sub-nuvens. Ao extrair um valor de um breve desalinhamento entre duas instâncias de um mesmo dado (nuvem *target*), o valor é restritivo porque além das distâncias serem pequenas, existe total sobreposição entre as duas instâncias. A forma como essa medida interfere na qualidade é que só haverá registro satisfatório quando o valor de erro para uma dada iteração for menor que a métrica, onde nos casos onde essa comparação for estabelecida nos primeiros índices de sub-nuvens leva a convergência e não necessidade de verificação da qualidade nos próximos índices.

3.3.2 Formulação matemática

Nesta seção é mostrado o desenvolvimento da formulação matemática do método GCP-ICP.

3.3.2.1 Particionamento

Dados dois modelos de entrada, a saber, os modelos de *source* (S) e *target* (T), o particionamento é a operação de agrupar os pontos compostos por cada um deles em conjuntos menores, até um máximo de k grupos, daqui em diante chamadas de sub-nuvens e indexadas pela letra j . O agrupamento ocorre por meio de $k - 1$ cortes transversais ao longo dos eixos principais X, Y, Z . Como os cortes podem ser feitos em qualquer um dos três eixos, consideramos que os cortes podem ser feitos em um eixo genérico, denotado por $\hat{\xi}$. Matematicamente, dado

$$S = \{\vec{s}_i\}, i \in N, 1 \leq i \leq N_S \quad (3.1)$$

, e

$$T = \{\vec{t}_m\}, m \in N, 1 \leq m \leq N_T \quad (3.2)$$

, nas Equações 3.1 e 3.2 representam os conjuntos de pontos contidos nas nuvens de entrada.

Tendo a representação das nuvens de pontos originais, escolhemos um eixo de corte (a critério do usuário da aplicação). Depois, é realizada uma ordenação dos valores das coordenadas na direção escolhida. A etapa de ordenação é necessária para a etapa de particionamento, pois a forma como os conjuntos de dados são estruturados não é a mesma que os pontos descrevem topologicamente um objeto ou cena (não é a mesma como aparece no visualizador). Sendo assim, os dados serão organizados em ordem crescente dos valores das coordenadas numa direção específica, por meio da função *sort*. Os conjuntos serão definidos pelo produto interno entre os vários vetores que representam os pontos (ordenados) da nuvem e o vetor unitário do eixo adotado, $\hat{\xi}$.

As sub-nuvens são então definidas como S_j e T_j para as partições dos modelos *source* e *target*, respectivamente. Para determinar quais pontos pertencerão exatamente a cada uma das j -ésima sub-nuvens, ou em outras palavras, para determinar matematicamente os subconjuntos de S_j e T_j , fazemos a divisão entre o total de pontos de cada nuvem, N_S e N_T e a quantidade de cortes a serem realizados, $k - 1$. Essencialmente, as sub-nuvens j -ésima de S e T , onde os tamanhos são N_{S_j} e N_{T_j} , são esses pontos \vec{s}_i e \vec{t}_m cuja coordenada ao longo do eixo $\hat{\xi}$ está dentro de uma faixa de pontos. Matematicamente esses subconjuntos são definidos como nas Equações 3.3 e 3.4:

$$S_j = \{\vec{s}_i \mid (j-2) \cdot \frac{N_S}{k} < \vec{s}_i \cdot \hat{\xi} < (j-1) \cdot \frac{N_S}{k}\} \quad (3.3)$$

$$T_j = \{\vec{t}_m \mid (j-2) \cdot \frac{N_T}{k} < \vec{t}_m \cdot \hat{\xi} < (j-1) \cdot \frac{N_T}{k}\} \quad (3.4)$$

Nessas equações, a coordenada ξ é obtida do produto interno entre o vetor que descreve um dado ponto no espaço (\vec{s}_i ou \vec{t}_m) e o vetor unitário ao longo do eixo $\hat{\xi}$. Para facilitar a compreensão da seção seguinte, também são definidos subconjuntos para os índices dos pontos nas sub-nuvens definidas anteriormente. Vejamos as Equações 3.5 e 3.6 abaixo:

$$\Gamma_j = \{i \mid (j-2) \cdot \frac{N_S}{k} < \vec{s}_i \cdot \hat{\xi} < (j-1) \cdot \frac{N_S}{k}\} \quad (3.5)$$

$$\Omega_j = \{m \mid (j-2) \cdot \frac{N_T}{k} < \vec{t}_m \cdot \hat{\xi} < (j-1) \cdot \frac{N_T}{k}\} \quad (3.6)$$

Uma representação fiel do particionamento realizado aqui pode ser visto na Figura 13. Na figura temos cortes ao longo de Y para o modelo *Dragon* da University (2014).

Figura 13 – Exemplo de particionamento usando o GCP-ICP. O particionamento neste exemplo foi feito na direção top-down.



Fonte: o autor.

3.3.2.2 Verificação de correspondência

No núcleo de alinhamento do algoritmo proposto, é usado o ICP da PCL, baseado na proposição de Besl e McKay (1992). No entanto, aqui verificação de correspondência acontece

no espaço reduzido de sub-nuvens. Em termos simples, em comparação com a função custo do ICP clássico, aparece uma dependência do índice da sub-nuvem, j , e também uma soma externa para levar em conta explicitamente o cálculo das distâncias entre os pares de pontos nessas sub-nuvens dos modelos *source* e *target*. A função custo desta variante do ICP proposta é vista abaixo:

$$F_j(\vec{q}) = \frac{1}{N_{S_j}} \sum_{m \in \Omega_j} \sum_{i \in \Gamma_j} \| \vec{t}_m - (R_j(\vec{q}_R)\vec{s}_i + \vec{q}_T) \|^2, \quad (3.7)$$

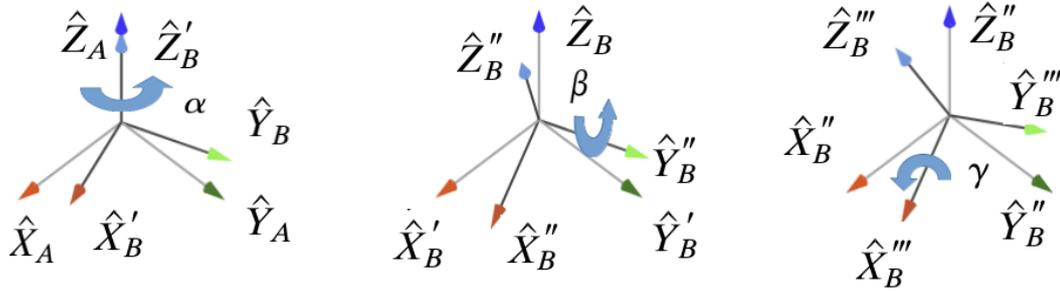
que é definido para $j = 1, \dots, k - 1$.

Nesta equação, a soma interna abrange todo o conjunto de pontos de uma determinada sub-nuvem j -ésima do modelo *source*, enquanto a externa, a do modelo *target*. Isto implica que o intervalo de índice dos pontos \vec{t}_m e \vec{s}_i inserindo a soma interna depende do índice de sub-nuvem em si, e é por isso que são colocadas as letras em grego Γ_j e Ω_j na equação. Durante a implementação, deve-se ser capaz de determinar o conteúdo dos conjuntos S_j e T_j dos quais os conjuntos Γ_j e Ω_j podem ser construídos.

Além disso, nessa função de custo, os parâmetros \vec{q}_R e \vec{q}_T são, respectivamente, o vetor de orientação representado por um *quaternion* unitário e o vetor de translação; o primeiro pode ser usado para gerar a matriz de rotação R_j (para detalhes, veja HORN (1987)). Esses vetores podem ser colocados juntos no formato compacto $\vec{q} = [\vec{q}_R | \vec{q}_T]^t$, permitindo assim recuperar o vetor de registro completo relacionado às sub-nuvens S_j e T_j , da mesma maneira que é descrito por Besl e McKay (1992), mas no espaço reduzido de sub-nuvens.

Deve-se finalmente enfatizar que o vetor de registro e, conseqüentemente, a matriz de rotação são obtidos toda vez que um par de sub-nuvens é submetido à correspondência; no GCP-ICP, essa correspondência parcial é estendida para o espaço total de nuvens de pontos através da transformação da nuvem *source* de entrada pela matriz R_j , que é a matriz de rotação relativa ao alinhamento de um dado par de sub-nuvens (S_j, T_j) e posterior verificação de qualidade do alinhamento. O desalinhamento entre as nuvens é calculado e comparado com φ , um valor de critério de parada que pode acelerar a tarefa de registro. Detalhes sobre a proposição autoral para esse valor são dados a seguir.

Figura 14 – Representação de um pequeno desalinhamento em eixos sucessivos.



Fonte: o autor.

3.3.2.3 Critério de convergência

A proposta para calcular o valor que serve como limiar para a convergência do GCP-ICP, anteriormente mencionado será devidamente explicado nesta subseção. Atribuímos a φ o critério de parada a partir do RMSE entre a nuvem de pontos *target* e uma cópia ligeiramente deslocada dela. Por ligeiro deslocamento entende-se uma mudança de orientação nos três graus de liberdade relativos à rotação. Esse erro proposital de rotação de agora em diante será chamado micro-desalinhamento.

Uma representação de como acontece micro-desalinhamento é ilustrada na Figura 14. Essencialmente, tendo um *frame T* representando a pose original do modelo *target*, um *frame* micro-desalinhado *M* pode ser obtido se rotações pequenas forem aplicadas sucessivamente ao redor dos eixos principais. No exemplo ilustrado, começamos a girar em torno de *Z* por um ângulo α , depois por *Y* por β e finalmente por *X* por γ .

Esta sucessão de fenômenos de rotação pode ser descrita matematicamente pela Equação 3.8:

$$\begin{aligned}
 {}^B_A R_{Z'Y'X'} &= R_Z(\alpha)R_Y(\beta)R_X(\gamma) \\
 &= \begin{bmatrix} c_z & -s_z & 0 \\ s_z & c_z & 0 \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} c_y & 0 & s_y \\ 0 & 1 & 0 \\ -s_y & 0 & c_y \end{bmatrix} \\
 &\times \begin{bmatrix} 1 & 0 & 0 \\ 0 & c_x & -s_x \\ 0 & s_x & c_x \end{bmatrix} \\
 &= \begin{bmatrix} c_y c_z & c_z s_z s_y - c_x s_z & s_x s_z + s_x s_y s_z \\ c_y s_z & c_x c_z + s_x s_y s_z & c_x s_y s_z c_z s_x \\ -s_y & c_y s_x & c_x c_y \end{bmatrix}, \tag{3.8}
 \end{aligned}$$

onde c_z , c_y e c_x são notações de formato abreviado para $\cos(\alpha)$, $\cos(\beta)$ e $\cos(\gamma)$. O mesmo vale para a função seno. Além da notação, ambos são medidos em radianos.

No caso em que os ângulos de rotação são iguais em torno dos três eixos, $\alpha = \beta = \gamma$, podemos substituir os três ângulos por um único θ e a matriz de rotação resultante se assemelhará a matriz vista a seguir:

$${}^M_T R_{Z'Y'X'}(\theta) = \begin{bmatrix} c_w^2 & c_w s_w^2 - s_w c_w & c_w^2 s_w + s_w^2 \\ c_w s_w & c_w^2 + s_w^3 & c_w s_w^2 c_w s_w \\ s_w & c_w s_w & c_w^2 \end{bmatrix}. \quad (3.9)$$

Indo mais além e limitando θ a ser um valor pequeno em radianos, a aproximação bem conhecida $\cos(\theta) = 1$ e $\sin(\theta) = \theta$ se aplica (LOW, 2004), levando a matriz a assumir a forma:

$${}^M_T R_{Z'Y'X'}(\theta) = \begin{bmatrix} 1 & -\theta & \theta \\ \theta & 1 & -\theta \\ -\theta & \theta & 1 \end{bmatrix}. \quad (3.10)$$

Vale observar que essa matriz reduz-se a identidade quando $\theta = 0$, o que significa que não há modificação de pose entre os *frames*.

Se a matriz de rotação resultante do micro-desalinhamento for multiplicada pela nuvem de pontos *target*, uma nuvem de pontos levemente desalinhada é obtida, em outras palavras, uma instância micro-desalinhada dela está disponível, a partir da qual podemos calcular uma medida de erro abaixo que será o critério de parada:

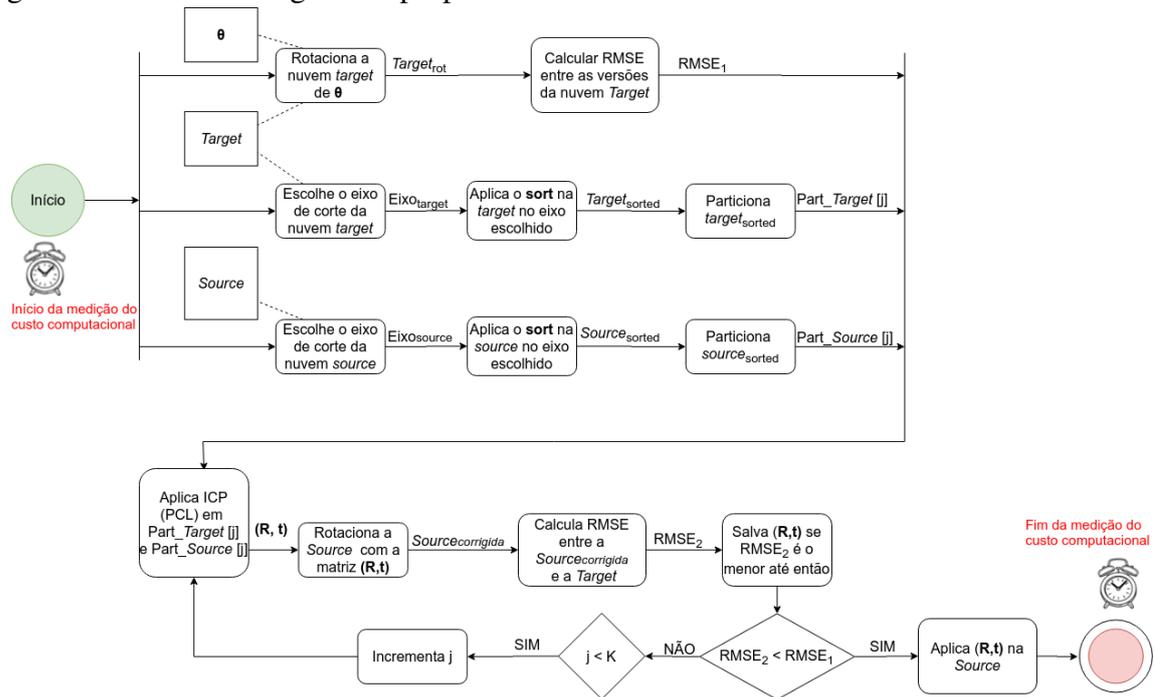
$$\varphi = \sqrt{\frac{1}{N_T} \sum_{i=1}^{N_T} \|E_i\|_2^2}, \quad (3.11)$$

onde a somatória se estende por todo o conjunto de pontos da nuvem *target*, E_i é a distância euclidiana entre seu ponto *i-ésimo* e seu vizinho mais próximo na cópia micro-desalinhada.

Conforme mencionado até agora, esse é um valor limite para interromper a tarefa de registro; especificamente, ele representa um limite inferior a ser comparado a cada iteração com o erro de alinhamento ε_j e, portanto, em cada iteração da abordagem proposta, almeja-se a situação onde $\varepsilon_j \leq \varphi$.

Conhecidas as etapas de particionamento, alinhamento e de convergência do método proposto, uma sintetização gráfica desses passos pode ser vista no fluxograma da Figura 15.

Figura 15 – Fluxo do algoritmo proposto.



Fonte: o autor.

3.3.3 Detalhes de implementação

Para que futuramente nossa técnica possa ser reproduzida, os eventuais leitores devem ter o mesmo ambiente controlado, como linguagem usada, bibliotecas e parâmetros. Abaixo, na Tabela 3, seguem os parâmetros utilizados em nossa técnica:

Tabela 3 – Parâmetros do método proposto.

Parâmetro de configuração	
Palpite inicial	-
Máximo de iterações (ICP)	30
Micro-desalinhamento	$-\pi/10 : -\pi/72$

Fonte: o autor.

Tanto o código da abordagem, quanto a remoção de *outliers*, bem como a maioria dos métodos de comparação e as métricas de avaliação quantitativa, foram feitos em C++, com auxílio da PCL (RUSU; COUSINS, 2011) e da FLANN (MUJA; LOWE, 2009). Os experimentos foram efetuados num computador Intel® Core™ i5-4200U CPU 1,60GHz 4, com placa gráfica Intel® Haswell Mobile e 8GB de memória.

3.4 Experimentos

Como apontado anteriormente, o objetivo deste trabalho é a proposta de um novo método de registro de nuvens de pontos. Para avaliar o método proposto, o GCP-ICP foi comparado com algumas técnicas do estado da arte. Dentre as técnicas usadas para comparação está o ICP e suas variantes, identificadas pelas letras (a,b,c), o NDT (d), o *Sparse ICP* (e), GoICP (f), a abordagem baseada no método FPFH (g), o CP-ICP (h) e as amostragens uniforme (i) e aleatória (j).

Para avaliar a qualidade dos registros em cada teste, foram utilizadas três diferentes métricas, sendo elas o custo computacional (1), o RMSE (2) e a representação eixo-ângulo de rotação de cada matriz de transformação (3). No entanto, a métrica (3) só é utilizada nos experimentos que a verdade terrestre é informada pelo autores dos respectivos bancos de dados. A sumarização das técnicas, das métricas e do objetivo por trás de cada experimento estão sumarizados na Tabela 4.

Tabela 4 – Experimentos para aferição da qualidade do método GCP-ICP.

Experimento	Algoritmos	Métricas	Objetivo
A	a, e, f, g, h	1, 2, 3	Avaliar o registro entre pares de nuvens de pontos
B	a, f, g	1, 2	Testar a qualidade do registro de pares de nuvens de pontos sob rotações arbitrárias
C	a, e, f, g, h	1, 2	Identificar o quanto o ruído afeta no registro de nuvens de pontos
D	i, j	1, 2	Alcançar registro satisfatório de múltiplas de visões parciais de objetos
E	a, f, g	1, 2	Entender como a subamostragem afeta a correspondência entre nuvens de pontos
F	a, b, c	1, 2	Avaliar o registro entre pares de nuvens de pontos de cenas indoor
G	a, b, c	1, 2	Alcançar registro satisfatório de múltiplas de visões parciais de cenas indoor
H	b, c, d	1, 2	Alinhar múltiplos scans de cenas outdoor
I	b, c, d	1, 2, 4	Avaliar o quanto a densidade dos pontos influi no registro

4 RESULTADOS

Neste capítulo serão apresentados e discutidos os resultados dos experimentos realizados nesta dissertação, fazendo uso dos algoritmos e das métricas de avaliação mencionados no Capítulo 3.

4.1 Resultados prévios

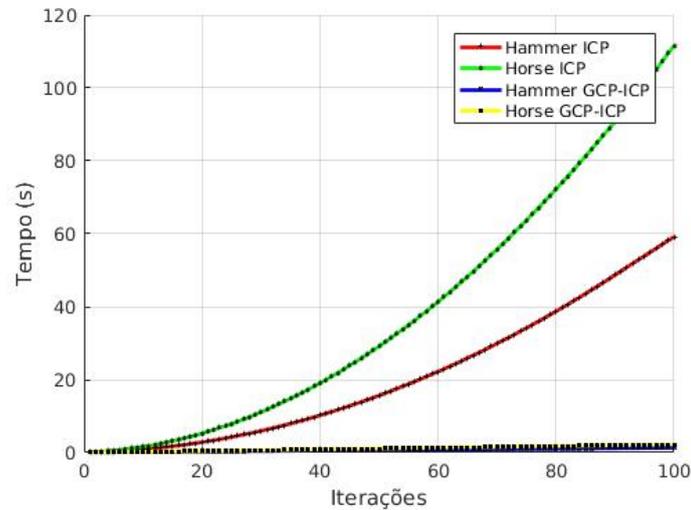
Nas seções seguintes, são feitas considerações sobre o comportamento da técnica proposta em situações variadas, dando ao leitor elementos que justificam a qualidade do método proposto.

4.1.1 *Custo computacional*

Esta abordagem de agrupamento de pontos em sub-nuvens é *per se* uma ideia interessante, pois leva a baixos esforços computacionais em comparação com o ICP clássico. A lógica é que quanto mais sub-nuvens, mais soluções para matrizes de transformação são obtidas e, portanto, há mais candidatos a serem testados ao procurar por um registro global aceitável. Também este conceito de aceitação é interessante, já que damos ao algoritmo a flexibilidade necessária para terminar a correspondência sempre que um bom alinhamento é encontrado, de acordo com a medida da qualidade do alinhamento discutida na seção anterior.

Como é mostrado na Figura 16, há aparentemente uma bifurcação entre as curvas que representam o ICP tradicional e o GCP-ICP no tempo decorrido para diferentes quantidades de iterações, tanto no modelo *Horse*, quanto no modelo *Hammer*. A bifurcação entre os tempos para as duas aplicações mostra que o fatiamento possibilita uma redução de custo computacional, visto que o tempo do ICP cresce bastante a medida que o número de iterações também aumenta.

Figura 16 – Comparação do tempo de registro para os modelo Horse e Hammer usando 100 iterações ou $k = 100$.

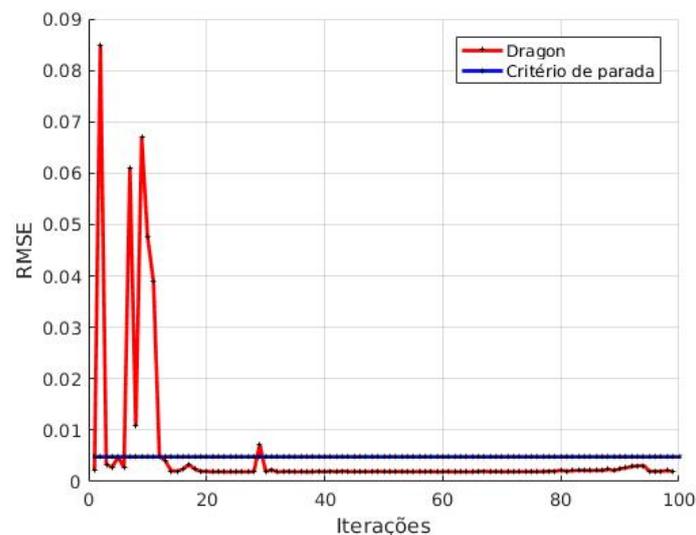


Fonte: O autor.

4.1.2 Estabilidade do RMSE

Na Figura 17, temos o RMSE obtido para 100 iterações do método GCP-ICP relativos ao alinhamento do objeto *Dragon*. Percebe-se que durante as 100 iterações, relativas ao alinhamento de k pares de sub-nuvens existe uma estabilidade do RMSE, havendo picos. Supõe-se que isso aconteça devido a uma menor sobreposição dos dados entre alguns pares de sub-nuvens, possivelmente fruto de auto-occlusão ou ausência de dados.

Figura 17 – Valores de RMSE para $k = 100$ relativos ao alinhamento de tomadas parciais do modelo *Dragon*.



Fonte: O autor.

Acredita-se então que o particionamento leva a uma redução da ambiguidade que é decorrente do fato de se registrar o conjunto de dados completo. Aqui são obtidas um número k de soluções, onde apenas uma delas é replicada globalmente mediante a avaliação por um critério de convergência bastante restritivo (linha em azul). Particionar e comparar o registro de cada par de partições com uma métrica rígida levam a um registro ótimo e resiliente a situações de oclusão.

4.2 Experimento A: Registro de pares de objetos com conhecimento das rotações de cada objeto

O objetivo do primeiro experimento é verificar o desempenho dos diferentes algoritmos no registro de pares de nuvens de pontos; em essência, duas tomadas parciais de um dado objeto estão sujeitadas aos algoritmos GoICP, uma abordagem baseada no FPFH, *Sparse ICP*, ICP clássico, CP-ICP, além da técnica proposta. A avaliação quantitativa do registro baseou-se no cálculo do RMSE e na orientação estimada a partir da representação eixo-ângulo equivalente (visto que a verdade terrestre está disponível para esses modelos).

A Tabela 5 traz o tempo de registro. De acordo com os números apresentados na tabela, a técnica autoral GCP-ICP foi capaz de finalizar o registro em um tempo menor que as outras abordagens do estado da arte para os modelos *Bunny*, *Dragon* e *Buddha*, todos do repositório University (2014). Para os modelos *Horse* e *Hammer* do banco de dados Aleotti *et al.* (2014), completou o registro em um tempo curto, embora não tenha sido o mais rápido dentre aqueles com que foi comparado.

Tabela 5 – Tempo consumido para alinhar os pares de nuvens de pontos (em segundos).

	<i>Bunny</i>	<i>Dragon</i>	<i>Buddha</i>	<i>Horse</i>	<i>Hammer</i>
CP-ICP	4,922	3,847	9,085	0,338	0,202
GoICP	36,537	35,847	36,288	42,348	36,198
FPFH+ICP _{p2pt}	125,438	91,969	413,457	5,526	3,086
<i>Sparse ICP</i>	22,543	23,260	53,506	2,628	1,454
ICP _{p2pt}	8,202	6,903	15,585	0,696	0,351
GCP-ICP	1,687	1,420	2,745	3,832	0,215

Fonte: O autor.

Este desempenho foi esperado já pelas razões mencionadas até agora, ou seja, o espaço reduzido pelo particionamento. Quanto ao custo computacional das demais técnicas,

algumas têm valores mais elevados por serem técnicas globais, como o GoICP, outras por além de computarem um alinhamento inicial baseado em características, tem a *pose* refinada usando o ICP, caso do FPFH+ICP. Deve-se destacar o alto custo computacional do *Sparse ICP*. Como o valor adotado para a norma foi $p = 0.4$, como sugerido no artigo original Bouaziz *et al.* (2013), acaba existindo um *trade-off* entre custo computacional e qualidade, sendo necessário um número maior de iterações.

A Tabela 6 traz, por sua vez, o desalinhamento entre as nuvens de entrada após o registro, conforme medido pela métrica RMSE. Os resultados revelaram que o GCP-ICP obteve melhores resultados (ou ainda na mesma ordem de grandeza) de alinhamento. O resultado obtido é fruto da qualidade trazida pela restrição do critério de parada.

Vê-se que as técnicas que tiveram resultados equivalentes em alguns cenários foram o o ICP, o *Sparse ICP* e o CP-ICP. Para o *Sparse ICP*, é possível inferir que o valor adotado de p por ser pequeno, torna o algoritmo robusto a possíveis perturbações mas diminui a qualidade do registro para outros casos. O GoICP, método global, deveria entregar uma métrica de RMSE que atestasse um registro suficiente, no entanto, o alinhamento é feito entre uma nuvem completa e outra parcial (aleatoriamente amostrada), o que influencia no cálculo da métrica.

Tabela 6 – RMSE calculado para o registro com cada método.

	<i>Bunny</i>	<i>Dragon</i>	<i>Buddha</i>	<i>Horse</i>	<i>Hammer</i>
CP-ICP	0,011	0,002	0,003	0,029	0,010
GoICP	0,089	0,055	0,032	0,523	0,207
FPFH+ICP _{p2pt}	0,004	0,003	0,004	0,004	0,005
<i>Sparse ICP</i>	0,057	0,002	0,003	0,027	0,011
ICP _{p2pt}	0,002	0,002	0,003	0,020	0,004
GCP-ICP	0,002	0,003	0,003	0,004	0,004

Fonte: O autor.

Como neste experimento são conhecidas as rotações entre as tomadas de cada objeto, chamadas também de verdades universais (*ground-truth*), pode ser verificado também de forma quantitativa o alinhamento por meio da representação eixo-ângulo equivalente de rotação; isso é trazido na Tabela 7. Na primeira linha da tabela, a verdade terrestre perseguida pelos diferentes métodos é mostrada; a tabela revela que o GCP-ICP obteve resultados mais próximos desses valores em quatro cenários, valendo ressaltar o resultado obtido com o modelo *Horse*, onde a rotação é de 180 graus. Esse resultado reforça os valores de RMSE encontrados na tabela anterior, haja vista que esta não é uma medida representativa (RMSE), embora seja a mais

utilizada, inclusive neste trabalho. Dentre os algoritmos utilizados para comparação, o CP-ICP de SOUZA NETO *et al.* (2018), embora tenha por base o princípio do particionamento como a técnica proposta, não possui valor de critério de parada bem definido, sendo necessário um estudo a priori para cada objeto. Ainda sobre o CP-ICP, os resultados onde este difere ainda mais da versão atual é nas condições onde as rotações são sobre eixos diferentes de Z e nas grandes rotações. Com isso pode ser afirmado que ter um critério de parada bem definido e a versatilidade de escolher eixos diferentes para o particionamento possibilita um alinhamento de maior qualidade.

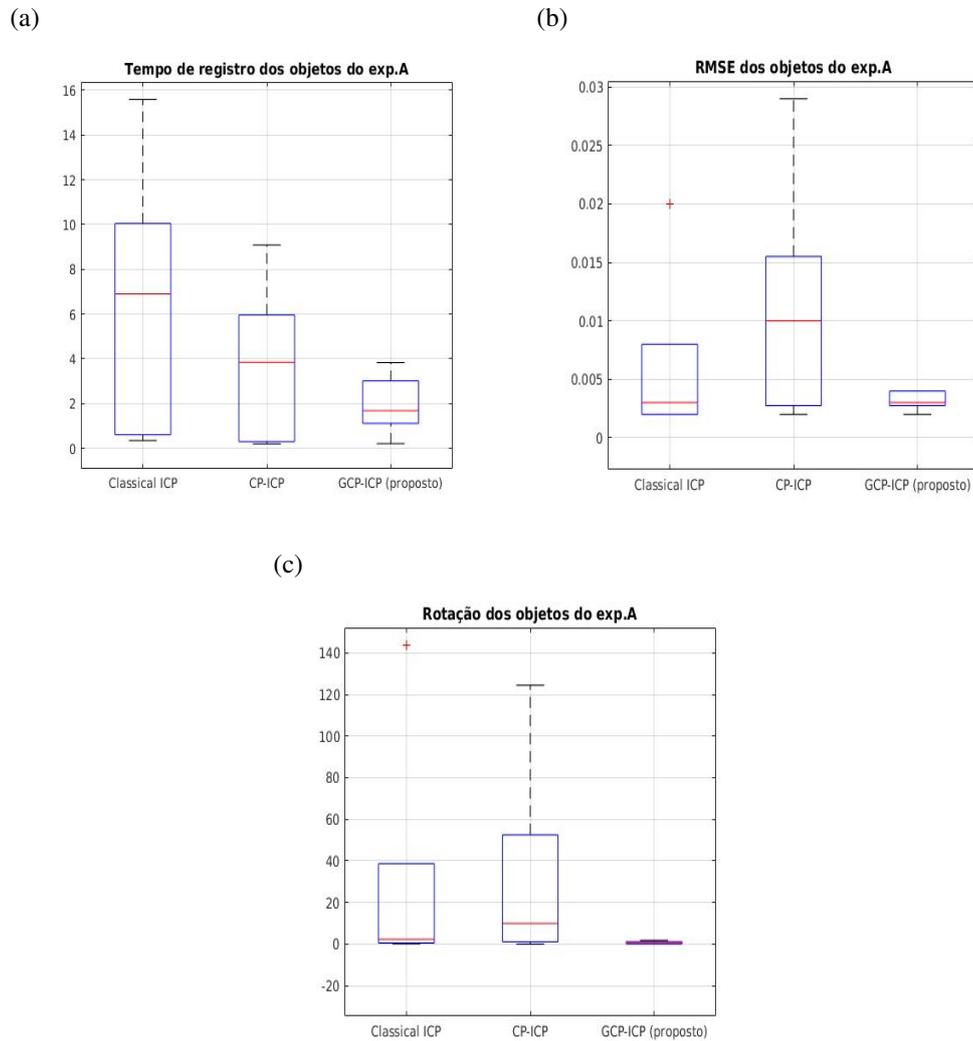
Tabela 7 – Rotação medida para as variadas técnicas (em graus)

	<i>Bunny</i>	<i>Dragon</i>	<i>Buddha</i>	<i>Horse</i>	<i>Hammer</i>
<i>Ground-truth</i>	45°	24°	24°	180°	45°
CP-ICP	16,498°	24,009°	22,543°	55,869°	35,075°
GoICP	34,480°	61,281°	15,612°	42,348°	36,198°
FPFH+ICP _{p2pt}	46,706°	49,605°	50,154°	187,515°	58,929°
ICP _{p2pt}	41,301°	23,863°	21,679°	36,342°	45,577°
GCP-ICP	43,246°	24,112°	24,039°	179,000°	44,591°

Fonte: O autor.

Os resultados apresentados nas Tabelas 5, 6 e 7 estão resumidas em gráficos de caixa na Figura 18. A partir do gráfico mais à esquerda, podemos observar que a técnica proposta reduz o tempo médio de registro em relação aos demais métodos e menor variabilidade dentro do conjunto de modelos estudados; isso sugere que a rapidez na correspondência de nuvens é, no nosso caso, independente do modelo. A mesma conclusão pode ser feita para o RMSE, onde os valores de convergência são muito baixos, o que é, por sua vez, relacionado à restrição do critério de parada adotado. Finalmente, no que diz respeito ao erro de rotação, o GCP-ICP foi mais eficiente em atingir rotações que aproximam-se da verdade terrestre (veja a menor abertura nos *boxplot*).

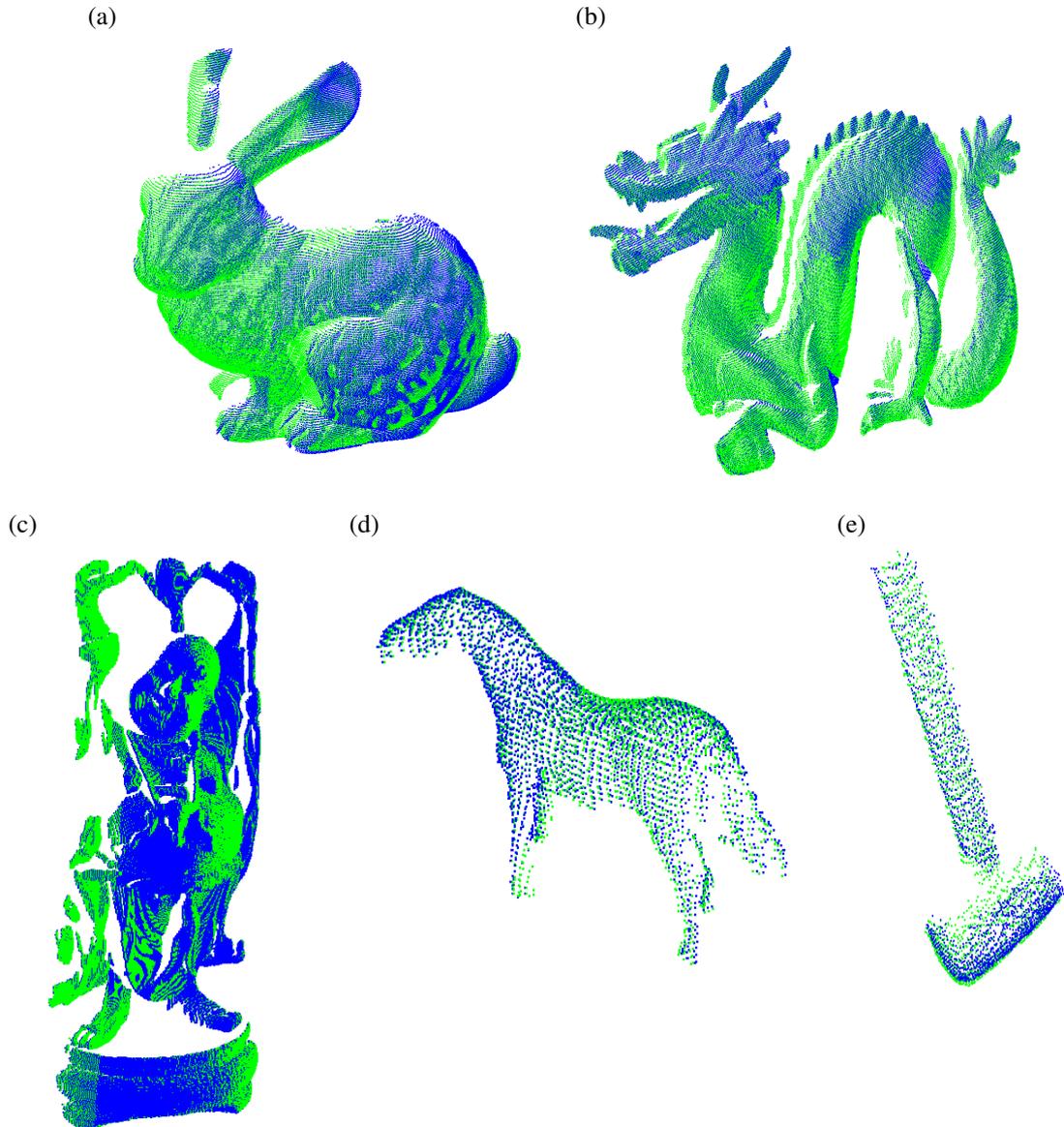
Figura 18 – Comparação do a) custo computacional, b) erro de registro (RMSE) e c) erro de rotação no alinhamento de cinco primeiros objetos, pelo método GCP-ICP, versus outros métodos locais.



Fonte: O autor.

Na Figura 19 podemos observar a qualidade do alinhamento por meio do resultado do registro dos cinco objetos estudados.

Figura 19 – Registro de pares de nuvens de objetos sob diferentes condições de rotação. Os modelos mostrados são: (a) *Bunny*, (b) *Dragon*, (c) *Buddha*, (d) *Horse* e (e) *Hammer*.



Fonte: O autor.

4.3 Experimento B: Registro de pares de objetos submetidos a sucessivas rotações em um eixo genérico.

O objetivo deste experimento é investigar o registro entre pares de nuvens de um determinado objeto na situação em que a pose de uma delas é uma orientação genérica obtida de 20 fenômenos de rotação sucessivos (e arbitrários) nos eixos ortogonais; nenhuma verdade terrestre estava disponível, portanto. Os algoritmos usados na comparação são os mesmos do experimento anterior, com exceção do CP-ICP que já demonstrou dificuldade no registro de rotações que não são sobre o eixo Z. Foram utilizados como objeto de estudo os modelos *Chef*

do repositório em PCL (2018), *Monkey* e *Eagle* do Bouaziz *et al.* (2013) e *Hippo* de Mellado *et al.* (2014). A avaliação conta com a comparação de custo computacional e a métrica de RMSE.

Pode ser afirmado que este é o experimento mais desafiador, porque o eixo de rotação é genérico e, ainda assim, como pode ser visto na Tabela 8, o algoritmo GCP-ICP precisou de um tempo inferior para resolver o alinhamento em alguns modelos, ou ainda esteve próximo daquele que mostrou ser mais rápido.

Tabela 8 – Tempo consumido para alinhar os pares de nuvens de pontos (em segundos).

	<i>Monkey</i>	<i>Chef</i>	<i>Eagle</i>	<i>Hippo</i>
GoICP	36,791	36,030	37,498	35,943
FPFH+ICP _{p2pt}	60,997	8,357	459,809	38,228
ICP _{p2pt}	3,576	1,007	16,892	4,991
GCP-ICP	3,414	1,116	7,873	2,876

Fonte: O autor.

Embora as rotações aconteçam sobre um eixo genérico, o particionamento acontece em relação ao sistema de coordenadas da nuvem original, ou seja, o algoritmo se beneficia das partições não variarem com a rotação. Em três dos modelos, como é visto na Tabela 9, o método GCP-ICP obteve valores de RMSE menores, sendo um deles o *Cheff*, onde seu custo computacional foi um pouco maior que do algoritmo ICP. Com esse exemplo, temos uma situação onde ser um pouco mais custoso resultou na entrega de uma qualidade maior

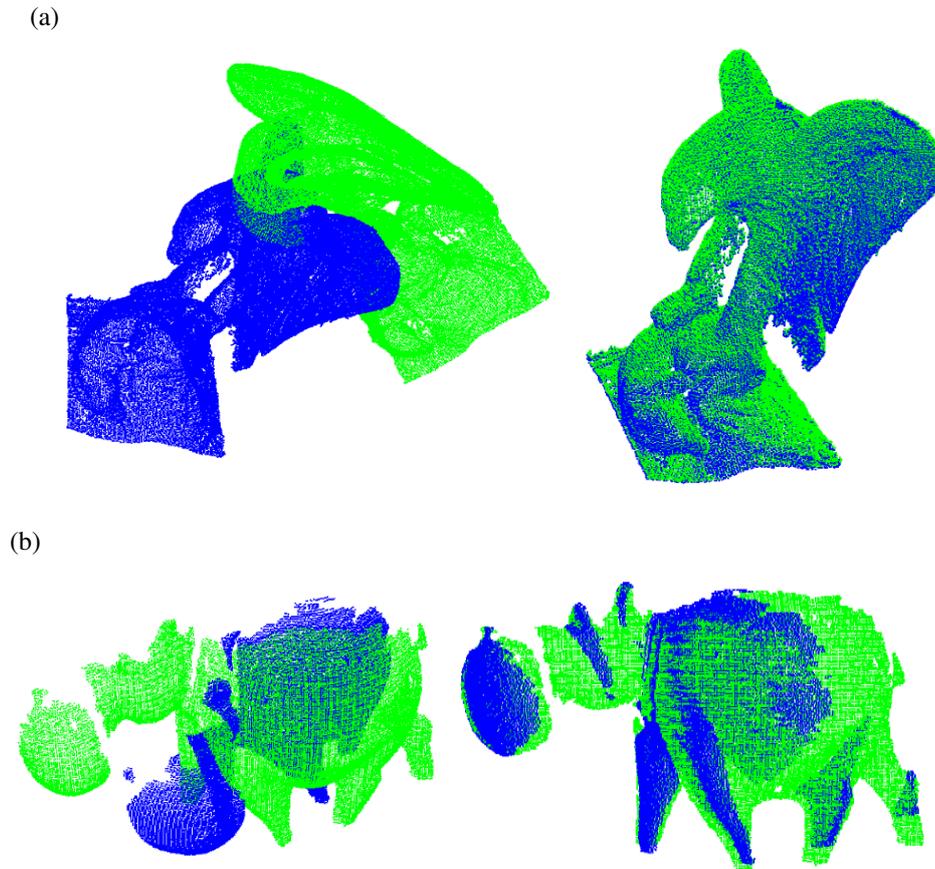
Tabela 9 – RMSE calculado para o registro com cada método.

	<i>Monkey</i>	<i>Chef</i>	<i>Eagle</i>	<i>Hippo</i>
GoICP	0,487	0,155	0,665	0,556
FPFH+ICP	0,245	0,020	0,470	0,040
ICP	0,017	0,033	0,018	0,033
GCP-ICP	0,061	0,003	0,021	0,017

Fonte: O autor.

Abaixo, na Figura 20 temos a representação dos modelos *Eagle* e *Hippo*. Do lado esquerdo temos a rotação antes do alinhamento e na direita o resultado do registro para cada modelo. Como pode ser observado, o alinhamento para estes modelos foi satisfatório.

Figura 20 – Registro de pares de nuvens de objetos sob rotações em um eixo genérico. Os modelos mostrados são: (a) *Eagle* e (b) *Hippo*.



Fonte: O autor.

4.4 Experimento C: Registro de pares de nuvem na presença de ruído

Neste experimento foi examinado como a adição de ruído gaussiano afetaria a qualidade e o tempo de registro. Ambas as nuvens de pontos foram corrompidas com ruído, com desvio padrão variando de 0,001 a 0,003. Ao avaliar o tempo de convergência de cada método, descobriu-se que, para algumas das técnicas, o tempo de registro diminuía a medida que o nível de ruído aumentava de 0,002 para 0,003. Em contraste, o GCP-ICP não mostra o mesmo comportamento com a flutuação de ruído. Os resultados da avaliação do custo computacional podem ser vistos na Tabela 10.

Tabela 10 – Tempo consumido para alinhar os pares de nuvens de pontos sob diferentes quantidades de ruído (em segundos).

	$\sigma = 0.001$	$\sigma = 0.002$	$\sigma = 0.003$
GoICP	36,104	36,768	36,618
FPFH+ICP _{p2pt}	125,667	129,518	127,861
<i>Sparse</i> ICP	32,693	25,234	27,280
ICP _{p2pt}	7,850	7,778	7,680
GCP-ICP	2,028	5,563	5,972

Fonte: O autor.

No que diz respeito ao RMSE, existe um alinhamento geral satisfatório. Como pode ser visto na Tabela 11, a técnica *Sparse* ICP que teve um resultado satisfatório no Experimento A, caiu de qualidade quando comparada ao GCP-ICP em condições adversas. Dentre os algoritmo de comparação, o ICP foi o que se mostrou mais estável e tem resultados equivalentes ao do algoritmo proposto.

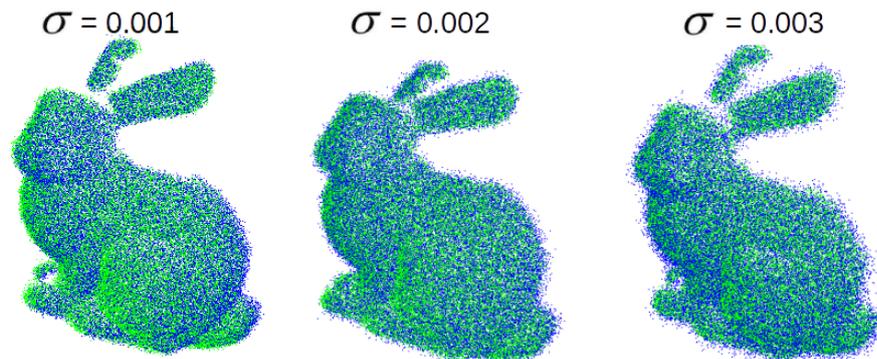
Tabela 11 – RMSE calculado para o registro com cada método para os pares de nuvem do modelo *Bunny* sob diferentes quantidades de ruído (em metros).

	$\sigma = 0.001$	$\sigma = 0.002$	$\sigma = 0.003$
GoICP	0,027	0,037	0,044
FPFH+ICP _{p2pt}	0,023	0,015	0,015
<i>Sparse</i> ICP	0,018	0,056	0,028
ICP _{p2pt}	0,002	0,002	0,002
GCP-ICP	0,002	0,002	0,002

Fonte: O autor.

Conforme ilustrado na Figura 21, temos o resultado do alinhamento do modelo *Bunny*.

Figura 21 – Resultado do registro de pares do modelo *Bunny* usando o GCP-ICP. Três níveis diferentes de ruído são testados.



Fonte: O autor.

4.5 Experimento D: Efeito de métodos de amostragem clássicos frente ao particionamento

No experimento D a intenção é mostrar os efeitos da aplicação de métodos de sub-amostragem tradicionais baseados na remoção de pontos *versus* a amostragem baseada em particionamento. A amostragem uniforme e duas variantes da amostragem aleatória foram usados como métodos de comparação. As variantes da amostragem aleatória foram de 50% e 70% dos pontos, inspirado nos experimentos feitos pelo autor em SOUZA NETO *et al.* (2018). Os modelos usados para avaliação foram o *Buddha* e o *Dragon*. Quanto a esses modelos temos os ângulos de aquisição originais, podendo assim como no experimento A avaliar o esforço de rotação em relação as verdades universais, ambas de 24 graus em torno do eixo Z.

A Tabela 12 sintetiza a avaliação de desempenho pelo tempo, RMSE e rotação. Baseado nos resultados, confirma-se a ineficácia de usar métodos de amostragem tradicionais. Embora para os dois objetos a ordem de grandeza dos valores de RMSE seja a mesma (variando apenas na terceira casa decimal), quando observada a rotação final obtida, percebemos que o ICP original tem dificuldade ao alinhar os pares de nuvens.

Tabela 12 – Efeito da amostragem no registro de nuvens de pontos.

	<i>Dragon</i>			<i>Buddha</i>		
	Tempo	RMSE	Rotação	Tempo	RMSE	Rotação
Uniforme	0,298	0,004	23,683	0,462	0,004	13,184
Aleatório 50%	5,058	0,002	23,873	8,428	0,003	21,765
Aleatório 70%	2,268	0,002	23,843	5,140	0,003	21,490
GCP-ICP	1,420	0,003	24,112	2,745	0,003	24,039

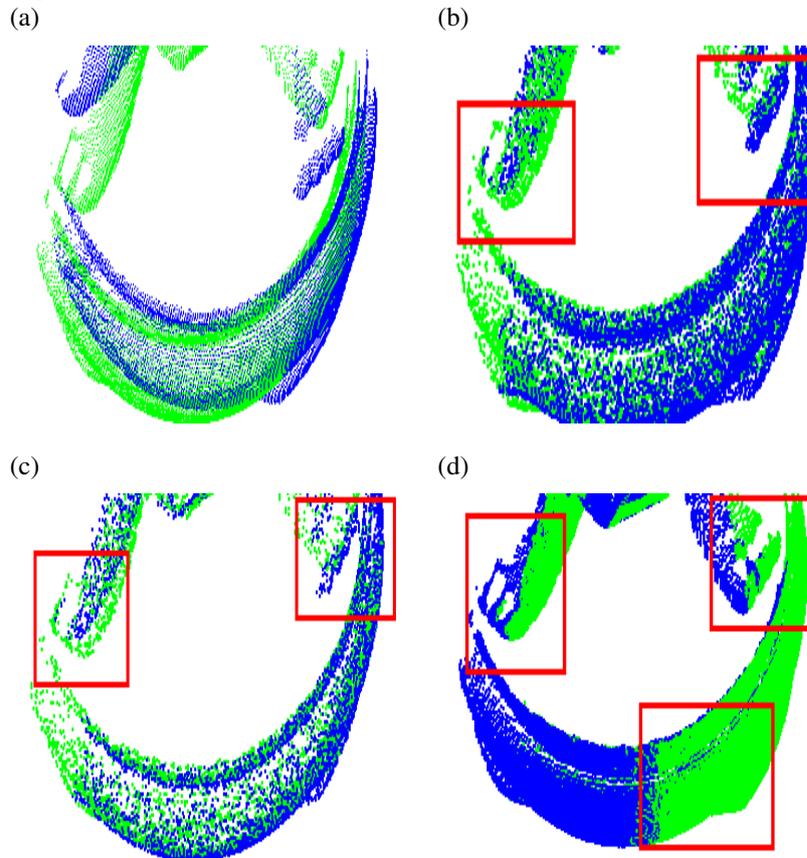
Fonte: O autor.

Quando removidos pontos da nuvem original, a distribuição desses e a representatividade de algumas regiões são alteradas, o que pode implicar em falsas correspondências entre as nuvens envolvidas no processo de registro. O método de particionamento presente no GCP-ICP mostra-se eficaz porque atua num caminho contrário, não removendo pontos originais das nuvens e sim os delimita em porções menores, que a depender do índice de cada partição, terá grandes chances de correspondência.

A prova dessa afirmação é a avaliação da parte inferior do modelo de *Buddha*, bem como a observação dos detalhes marcados pelos quadrados em vermelho na Figura 22. Os pés do *Buddha*, destacados na imagem, provam que a velocidade trazida por um método de amostragem

pode não ser satisfatória, dada a propagação de falsas correspondências, o que não acontece no GCP-ICP.

Figura 22 – Comparação dos resultados do registro para o modelo *Buddha*. Em a) a perspectiva inicial das duas nuvens, em b) o registro para a amostragem aleatória de 50% e em c) o resultado do registro para a amostragem aleatória de 70% e em d) o resultado de registro com o GCP-ICP.



Fonte: O autor.

4.6 Experimento E: Registro de múltiplas rotações de um mesmo objeto.

Neste experimento, várias aquisições de um determinado objeto estão disponíveis para registro, cada uma adquirida de uma pose diferente; os dados utilizados nesta análise são 5 varreduras do *Buddha* com mudança de orientação por quantidades de 24° em torno de Z entre tomadas sucessivas, e 3 varreduras para o modelo *Dragon* com o mesmo incremento de rotação. Dessa forma, a correspondência final representa o registro paritário sucessivo, com os modelos de entrada.

Como pode ser visto na Tabela 13, como nos demais experimentos até o momento, novamente o GCP-ICP mostra-se mais rápido. Em geral, os valores de RMSE que servem como

avaliação do registro foram satisfatórios, sendo os menores dentre os algoritmos comparados.

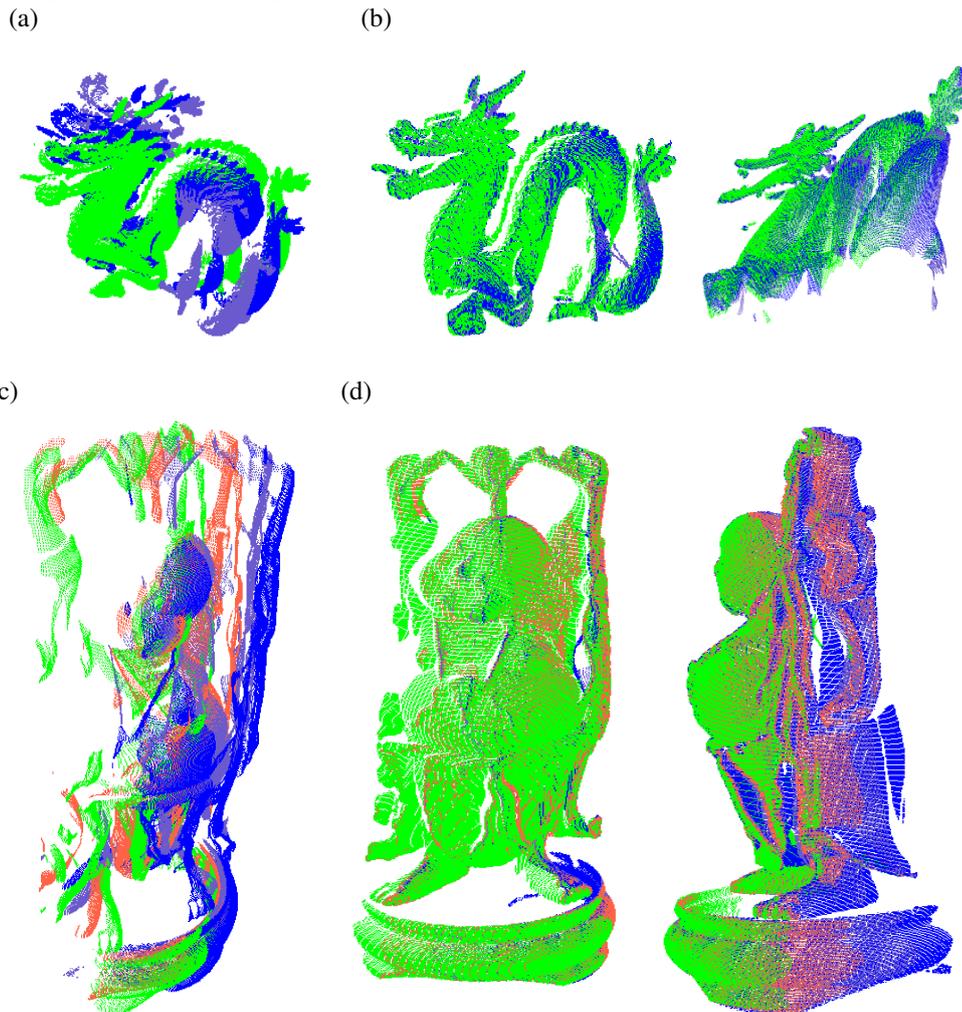
Tabela 13 – Comparação do alinhamento de múltiplos *scans* de objetos.

	<i>Dragon</i>		<i>Buddha</i>	
	Tempo	RMSE	Tempo	RMSE
GoICP	72,134	0,063	144,460	0,060
FPFH+ICP _{p2pt}	149,267	0,020	1430,884	0,016
ICP _{p2pt}	11,075	0,004	56,216	0,003
GCP-ICP	7,708	0,003	32,442	0,003

Fonte: O autor.

Complementar ao resultado das métricas, temos uma breve visualização da qualidade do alinhamento para este experimento pode ser vista na Figura 23.

Figura 23 – Registro de múltiplas rotações dos objetos *Dragon* e *Buddha*. Em (a) e (c) temos as poses iniciais (esquerda) e em (b) e (d) o resultado do alinhamento.



Fonte: O autor.

4.7 Experimento F: Alinhamento de pares de nuvens de pontos de cenas *indoor*

Neste experimento, duas poses de uma determinada cena são registradas. As cenas consideradas são aquisições próprias de dois modelos diferentes do laboratório de pesquisa, denominados Laboratório 1 (Lab.1) e Laboratório 2 (Lab.2), além dos modelos *Office* e *Stage* originalmente provenientes do banco de dados em Mellado *et al.* (2014).

Com uma breve observação da Tabela 14, que traz o tempo de registro de cada técnica para cada um dos ambientes estudados, vemos que, em geral, as técnicas demandaram mais esforços neste experimento em relação aos experimentos anteriores. A boa notícia é que, mais uma vez, a abordagem proposta supera a maioria das técnicas analisadas. Como já foi dito anteriormente, a redução do espaço de busca é o fator que proporciona essa tamanha redução de custo computacional.

Tabela 14 – Custo computacional da etapa do alinhamento entre pares de cenas *indoor*.

	Lab.1	Lab.2	<i>Office</i>	<i>Stage</i>
ICP _{p2pt}	11,670	12,872	14,104	15,124
ICP _{p2pln}	7,346	9,022	10,197	10,767
<i>Generalized</i> ICP	11,124	44,924	537,961	23,559
GCP-ICP	1,815	1,156	1,567	1,856

Fonte: O autor.

No que diz respeito à qualidade da correspondência, relatada na Tabela 15, a abordagem proposta obteve o RMSE mais baixo em dois dos quatro cenários, tendo resultados na mesma ordem de grandeza nos cenários *Office* e *Stage* se comparados ao algoritmo vencedor. Vale a pena mencionar o fato de que os modelos Lab.1 e 2 foram adquiridos em baixa resolução, o que sugere que algumas dessas técnicas apresentam alguma dependência de desempenho da tecnologia de aquisição (natureza do sensor), mas o GCP-ICP manteve-se insensível a isso para os testes feitos.

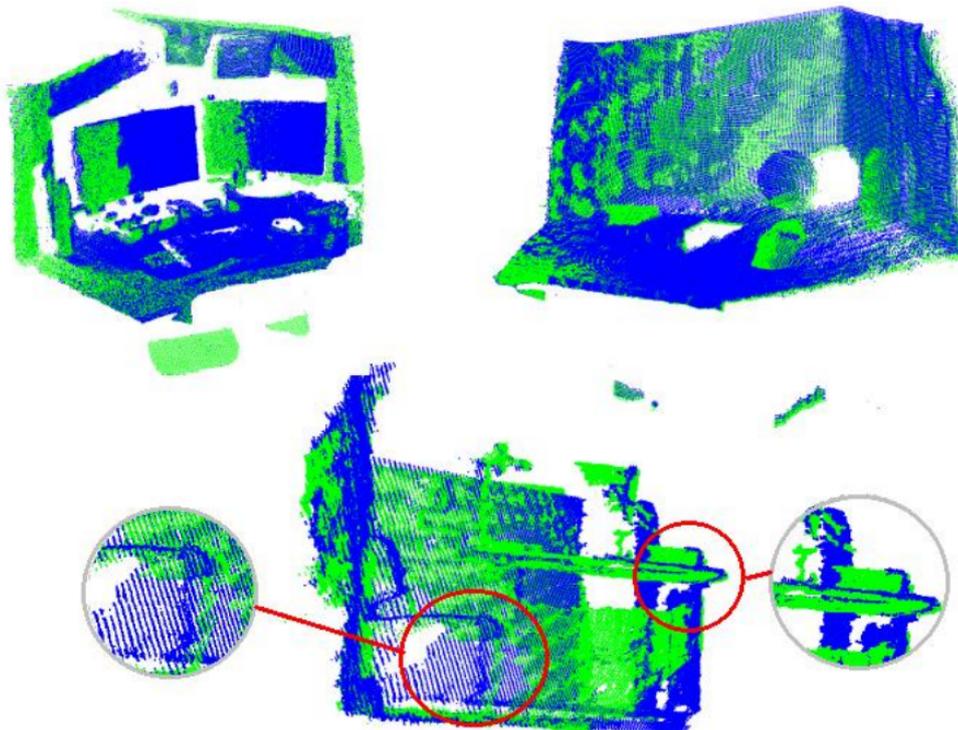
Tabela 15 – RMSE do alinhamento entre pares de cenas *indoor*.

	Lab.1	Lab.2	<i>Office</i>	<i>Stage</i>
ICP _{p2pt}	0,068	0,009	0,046	0,005
ICP _{p2pln}	0,069	0,009	0,059	0,005
<i>Generalized</i> ICP	0,086	0,018	0,340	0,109
GCP-ICP	0,007	0,009	0,048	0,006

Fonte: O autor.

O resultado do alinhamento entre os pares de cenas *indoor* dos modelos Lab.1 e 2 e *Stage* pelo método proposto pode ser visto na Figura 24. Para o modelo *Stage* algumas regiões são ressaltadas por círculos em vermelho e representadas com *zoom*. Vê-se que para os três modelos, adquiridos com sensores de baixa resolução, o resultado foi satisfatório novamente.

Figura 24 – Resultado do alinhamento entre pares de nuvens de cenários *indoor*. Na parte superior, o alinhamento das nuvens adquiridas especialmente para esse experimento usando o sensor *Razer Stargazer SR 300* e na parte inferior com os dados obtidos de Mellado *et al.* (2014).



Fonte: O autor.

4.8 Experimento G: Alinhamento de múltiplos *scans* de cenas *indoor*

Neste experimento, várias visões parciais estão disponíveis para registro, cada uma adquirida a partir de uma *pose* diferente, mas desta vez cenas *indoor* (em vez de objetos) são os modelos a serem registrados. Na Tabela 16 é relatado o tempo decorrido de cada uma das técnicas utilizadas para a tarefa. Aqui foram considerados três ambientes diferentes, nomeados *House office*, *Lounge chair* e *Stage*. Vê-se que a técnica proposta novamente obteve custo computacional inferior.

Tabela 16 – Custo computacional para alinhamento de múltiplas aquisições de cenários *indoor*.

	<i>House office</i>	<i>Lounge chair</i>	<i>Stage</i>
ICP _{p2pln}	21,179	21,762	23,290
ICP _{p2pt}	30,372	30,602	32,874
<i>Generalized ICP</i>	587,123	438,621	309,683
GCP-ICP	16,610	17,261	16,680

Fonte: O autor.

Avaliando agora a qualidade da correspondência pelos valores de RMSE relatados na Tabela 17, o leitor pode observar que o método proposto atinge os melhores resultados ou no mínimo equivalentes aos das técnicas clássicas. Embora cenas possam ter uma complexidade maior que objetos, o alinhamento pelo método GCP-ICP se beneficia aqui do fato de que existe boa sobreposição entre as varreduras.

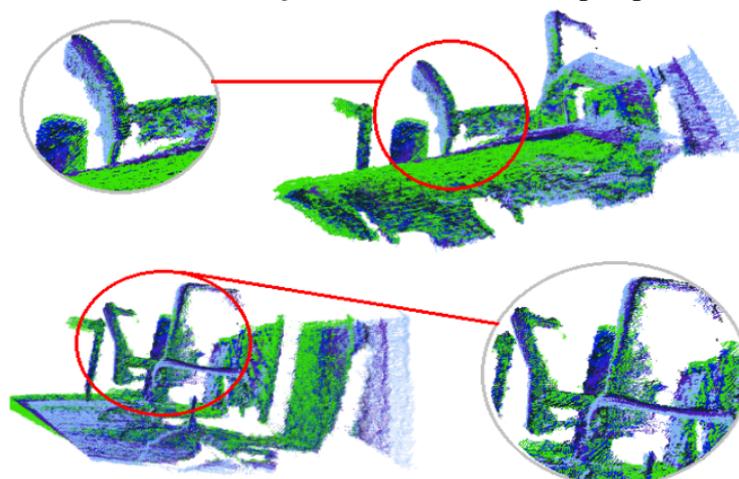
Tabela 17 – RMSE do alinhamento de múltiplas aquisições de cenários *indoor*.

	<i>House office</i>	<i>Lounge chair</i>	<i>Stage</i>
ICP _{p2pln}	0,030	0,018	0,032
ICP _{p2pt}	0,030	0,017	0,030
<i>Generalized ICP</i>	0,057	0,049	0,128
GCP-ICP	0,030	0,017	0,030

Fonte: O autor.

Para fins de avaliação qualitativa, podemos ver o alinhamento do cenário *Lounge chair* na Figura 25. Como pode ser observado, esse cenário conta com grande sobreposição entre as várias nuvens, o que facilitou o registro pelo método proposto.

Figura 25 – Alinhamento da cena *Lounge chair* sob diferentes perspectivas.



Fonte: O autor.

4.9 Experimento H: Alinhamento de tomadas parciais de cenas *outdoor*

Este experimento refere-se ao registro de cenas *outdoor*. Neste experimento, foi investigado o alinhamento de cenas *outdoor* da cidade de *Hannover*, em dois espaços distintos. Os métodos usados para comparação com o GCP-ICP são o NDT, o *Generalized ICP* e o ICP ponto-ao-plano. A avaliação quantitativa foi feita de forma semelhante aos testes anteriores e os modelos vêm do banco de dados em Wulf (2016).

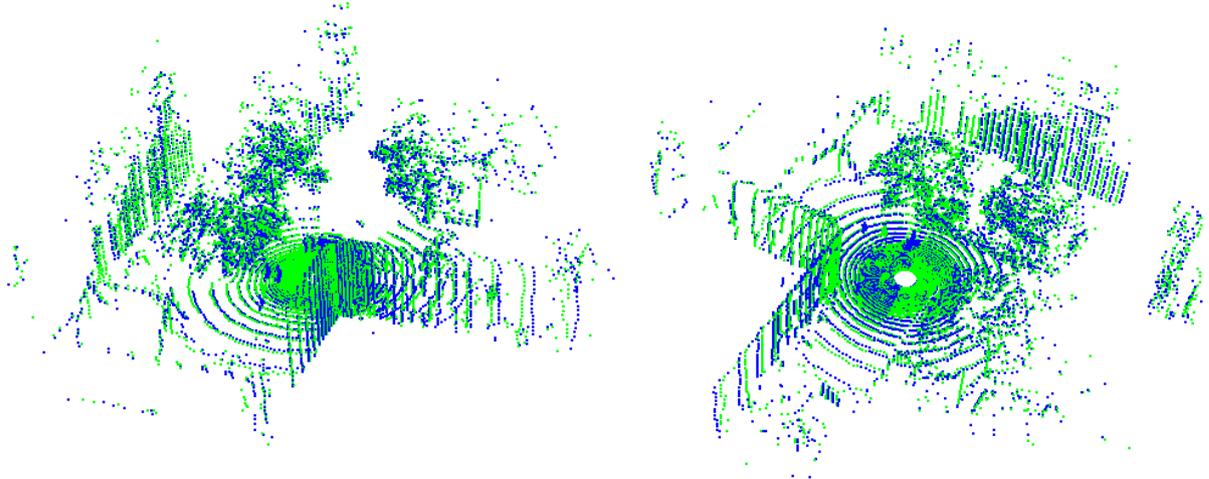
Na Tabela 18 o tempo de registro e as métricas do RMSE são relatados. Mais uma vez, neste problema de registro de nuvens, a abordagem de particionamento provou ser um meio eficiente para a solução do registro de nuvens de pontos, com valores de RMSE na menores que os dos demais algoritmos e com redução das demandas computacionais.

Tabela 18 – Resultado do alinhamento de cenas *outdoor*.

	<i>Hannover 1</i>		<i>Hannover 2</i>	
	Tempo	RMSE	Tempo	RMSE
ICP _{pt2pln}	1,512	0,617	1,517	0,335
Generalized ICP	46,998	0,710	21,651	0,342
3D NDT	15,047	0,710	20,177	0,377
GCP-ICP	0,485	0,405	0,489	0,325

Fonte: O autor.

Deve-se enfatizar que a abordagem GCP-ICP é 31 a 44 vezes mais rápida que o 3D-NDT e de 44 a 96 vezes mais rápido que o *Generalized ICP*. Vale argumentar também que, dada a lentidão do NDT e, especialmente, o *Generalized ICP*, eles não parecem adequados para sistemas com baixa disponibilidade de recursos computacionais. Além da avaliação quantitativa, a inspeção visual do alinhamento obtida pela abordagem proposta, é representada na Figura 26. A diferença de perspectiva é proposital para facilitar uma melhor visualização do alinhamento.

Figura 26 – Alinhamento da cena *Hannover 2*..

Fonte: O autor.

4.10 Experimento I: Registro de pares de cenas *outdoor* com densidades de pontos diferentes

O objetivo deste experimento é verificar o desempenho de técnicas clássicas frente ao algoritmo proposto quando os dados submetidos ao registro possuem densidades pontuais diferentes. Nas Tabelas 19 e 20 estão representadas as métricas de custo computacional e RMSE. Como pode ser observado na Tabela 19, novamente o método GCP-ICP obteve um registro em um tempo inferior, pelo menos da ordem de 30 a 200 vezes menor que o NDT, por exemplo.

A redundância, embora útil para tornar robusto o alinhamento, como no ICP tradicional, impacta no custo computacional. Portanto, o uso de subconjuntos representativos de pontos entregam o mesmo resultado, se não melhor precisão, com menos custo computacional.

Tabela 19 – Tempo de registro do alinhamento de cenas *outdoor* para nuvens com diferentes densidades (em segundos).

	<i>Bremen 1</i>	<i>Bremen 2</i>	<i>Hannover 3</i>	<i>Hannover 4</i>
ICP _{pt2pt}	32,811	229,220	1,503	1,510
Generalized ICP	561,096	336,119	2,911	3,065
NDT	1579,706	3365,678	9,987	17,776
GCP-ICP	22,808	25,067	0,297	0,082

Fonte: O autor.

Abaixo, na Tabela 20, avaliando a qualidade do registro, temos um resultado novamente satisfatório para o GCP-ICP, onde os valores de RMSE foram inferiores quando comparados com as demais técnicas.

Tabela 20 – RMSE do registro de cenas *outdoor* para nuvens com diferentes densidades.

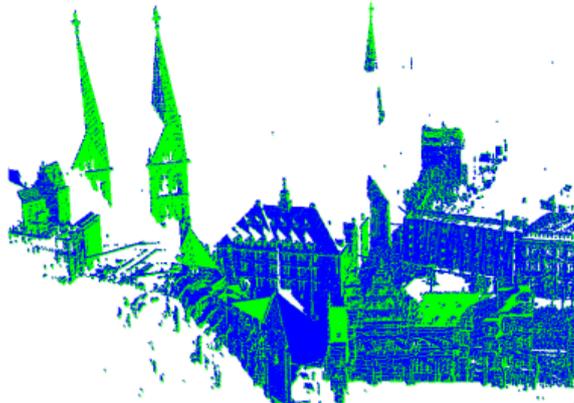
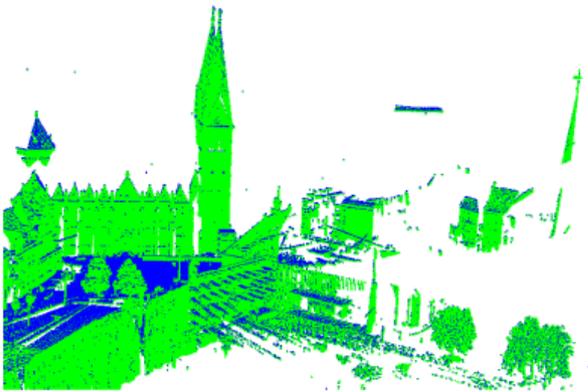
	<i>Bremen 1</i>	<i>Bremen 2</i>	<i>Hannover 3</i>	<i>Hannover 4</i>
ICP _{pt2ptn}	0,062	0,065	0,089	0,053
<i>GeneralizedICP</i>	0,063	0,065	0,089	0,053
NDT	0,074	0,066	0,093	0,057
GCP-ICP	0,609	0,053	0,041	0,024

Fonte: O autor.

Duas nuvens de pontos de um mesmo objeto ou cena passadas por um filtro de *voxel-grid*, são levadas a terem resoluções diferentes (número de pontos, densidade, etc). Métodos de registro baseados nas métricas de ponto-a-ponto, ponto-ao-plano e plano-a-plano clássicas e que utilizam todo o conjunto de dados falham por tentar estimar a orientação por causa das grandes discrepâncias na distribuição de pontos entre os conjuntos de dados. Essa dificuldade acontece pois não há correspondências diretas entre os pontos das duas nuvens, a níveis de coordenada. Além disso, métodos com base nas características geométricas como o NDT, estimam a correspondência entre os pontos tomando essas informações como base, sendo afetados pela mudança de resolução, aqui causada pelo pré-processamento com *voxel-grid*.

Mesmo na adversidade de contar com nuvens de diferentes densidades, o que poderia afetar o alinhamento de sub-nuvens bem diferentes (visto que o particionamento é baseado na quantidade de pontos), não prejudica a performance do GCP-ICP que consegue entregar um alinhamento satisfatório em custo computacional e RMSE. A afirmação anterior pode ainda ser confirmada pela avaliação qualitativa do alinhamento da cena *Bremen 1*, vista na Figura 27. Na imagem, temos o resultado do alinhamento sob diferentes perspectivas. Ainda que o experimento seja complicado, observamos que é possível reconhecer detalhes "finos" na estrutura, o que visualmente é um indicador de bom alinhamento.

Figura 27 – Alinhamento da cena *Bremen 1*.



Fonte: O autor.

5 CONCLUSÕES E TRABALHOS FUTUROS

Neste trabalho, foram apresentados melhoramentos de uma abordagem recentemente proposta, o CP-ICP, para o registro de nuvens de pontos. O *pipeline* desta variante do ICP foi modificado para acomodar uma maneira mais eficiente de registro, trazendo como contribuições uma etapa de particionamento em qualquer um dos eixos ortogonais e o alinhamento nesse espaço reduzido sob minimização da função custo em relação a um critério de parada restritivo, proposto no âmbito desta dissertação.

Algumas imposições interessantes foram realizadas nesse estudo. A primeira delas foi fornecer não apenas o registro de bases conhecidas, com nuvens pré-processadas, mas também adquirir alguns conjuntos de dados com um sensor de baixa resolução. Os testes feitos não simulam apenas o registro em boas condições, mas sob adição de ruído, rotações arbitrárias em um eixo genérico e alinhamento de nuvens de diferentes densidades, bem como o registro de objetos, cenas *indoor* e *outdoor*.

Ao longo dos mais variados experimentos, para objetos e cenários, sob diferentes condições, a técnica proposta foi capaz de resolver o problema de alinhamento em praticamente todos os testes e ainda em um tempo reduzido. Sobre os baixos valores de tempo encontrados, estes derivam da etapa de particionamento. Em muitos casos, a expressa redução do custo computacional nos faz inferir que o método elimina a necessidade de um processo de subamostragem anterior, que em casos particulares é prejudicial para a qualidade do alinhamento, como visto em um dos experimentos. No que diz respeito à qualidade do alinhamento, o critério de parada bastante restritivo leva a condições de alinhamento razoáveis, com baixos valores de RMSE e quando disponível para comparação, com baixo erro de rotação frente as verdades terrestres.

Como trabalhos futuros fica a implementação da escolha do eixo de corte automatizada, baseada em medidas estatísticas (desvio padrão das coordenadas, maior dispersão dos dados em uma direção, variância ou medidas que sejam oriundas da matriz de covariância, etc) e o estudo de outras métricas complementares ao critério de parada do algoritmo (que não apenas o RMSE).

Ainda que existam limitações, no que pode ser observado pela revisão bibliográfica e os resultados encontrados, a técnica apresentada aqui tem um alto grau de generalismo e serve a múltiplos propósitos.

REFERÊNCIAS

- AGAMENNONI, G.; FONTANA, S.; SIEGWART, R. Y.; SORRENTI, D. G. Point clouds registration with probabilistic data association. In: IEEE. **Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on**. [S.l.], 2016. p. 4092–4098.
- ALEOTTI, J.; RIZZINI, D. L.; CASELLI, S. Perception and grasping of object parts from active robot exploration. **Journal of Intelligent & Robotic Systems**, Springer, v. 76, n. 3-4, p. 401–425, 2014.
- BERGER, M.; TAGLIASACCHI, A.; SEVERSKY, L. M.; ALLIEZ, P.; GUENNEBAUD, G.; LEVINE, J. A.; SHARF, A.; SILVA, C. T. A survey of surface reconstruction from point clouds. **Computer Graphics Forum**, v. 36, n. 1, p. 301–329, 2017.
- BESL, P. J.; MCKAY, N. D. Method for registration of 3-d shapes. In: INTERNATIONAL SOCIETY FOR OPTICS AND PHOTONICS. **Sensor fusion IV: control paradigms and data structures**. [S.l.], 1992. v. 1611, p. 586–606.
- BIBER, P.; STRASSER, W. The normal distributions transform: A new approach to laser scan matching. In: IEEE. **Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems**. [S.l.], 2003. v. 3, p. 2743–2748.
- BOUAZIZ, S.; TAGLIASACCHI, A.; PAULY, M. Sparse iterative closest point. **Computer Graphics Forum (Symposium on Geometry Processing)**, v. 32, n. 5, p. 1–11, 2013.
- BOURNE Murray. **Vectors in 3-D Space**. 2014. Disponível em: <<http://www.intmath.com/vectors/7-vectors-in-3d-space.php>>. Acesso em: 7 abr. 2017.
- BUCH, A. G.; KRAFT, D.; KAMARAINEN, J.-K.; PETERSEN, H. G.; KRÜGER, N. Pose estimation using local structure-specific shape and appearance context. In: IEEE. **Robotics and Automation (ICRA), 2013 IEEE International Conference on**. [S.l.], 2013. v. 1, n. 1, p. 2080–2087.
- CASELLA, G.; BERGER, R. L. **Statistical inference**. 1a. ed. [S.l.]: Duxbury Pacific Grove, CA, 2002.
- CHEN, Y.; MEDIONI, G. Object modelling by registration of multiple range images. **Image and Vision Computing**, Elsevier, v. 10, n. 3, p. 145–155, 1992.
- CHENG, S.; MARRAS, I.; ZAFEIRIOU, S.; PANTIC, M. Statistical non-rigid icp algorithm and its application to 3d face alignment. **Image and Vision Computing**, Elsevier, v. 58, p. 3–12, 2017.
- COSTANZO, M.; MARIA, G. D.; LETTERA, G.; NATALE, C.; PIROZZI, S. Flexible Motion Planning for Object Manipulation in Cluttered Scenes. In: **Informatics in Control, Automation and Robotics: 15th International Conference, ICINCO 2018 Porto, Portugal, July 29-31, 2018**. [S.l.: s.n.], 2018. p. 978–989.
- CRAIG, J. J. **Introduction to robotics : mechanics and control**. 3rd international ed.. ed. [S.l.]: Pearson Higher Education, 2014. ISBN 9781292040042,1292040041.

- DAS, A.; DIU, M.; MATHEW, N.; SCHARFENBERGER, C.; SERVOS, J.; WONG, A.; ZELEK, J. S.; CLAUSI, D. A.; WASLANDER, S. L. Mapping, planning, and sample detection strategies for autonomous exploration. **Journal of Field Robotics**, Wiley Online Library, v. 31, n. 1, p. 75–106, 2014.
- DIEZ, Y.; ROURE, F.; LLADÓ, X.; SALVI, J. A qualitative review on 3d coarse registration methods. **ACM Computing Surveys (CSUR)**, ACM, v. 47, n. 3, p. 45, 2015.
- DUBÉ, R.; DUGAS, D.; STUMM, E.; NIETO, J.; SIEGWART, R.; CADENA, C. Segmatch: Segment based loop-closure for 3d point clouds. **arXiv preprint arXiv:1609.07720**, 2016.
- ELSEBERG, J.; MAGNENAT, S.; SIEGWART, R.; NÜCHTER, A. Comparison of nearest-neighbor-search strategies and implementations for efficient shape registration. **Journal of Software Engineering for Robotics**, v. 3, n. 1, p. 2–12, 2012.
- FELIPE, L.; NUNES, D. M.; ZAGHETTO, C.; VIDAL, F. D. B. 3D Face Recognition on Point Cloud Data. In: **15th International Conference on Informatics in Control, Automation and Robotics**. [S.l.: s.n.], 2018. v. 2, n. ICINCO. ISBN 9789897583216.
- FERNÁNDEZ-MORAL, E.; MAYOL-CUEVAS, W.; ARÉVALO, V.; GONZALEZ-JIMENEZ, J. Fast place recognition with plane-based maps. In: **IEEE Robotics and Automation (ICRA), 2013 IEEE International Conference on**. [S.l.], 2013. p. 2719–2724.
- FERNÁNDEZ-MORAL, E.; RIVES, P.; ARÉVALO, V.; GONZÁLEZ-JIMÉNEZ, J. Scene structure registration for localization and mapping. **Robotics and Autonomous Systems**, Elsevier, v. 75, p. 649–660, 2016.
- 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**, ACM, v. 24, n. 6, p. 381–395, 1981.
- FITZGIBBON, A. W. Robust registration of 2d and 3d point sets. **Image and vision computing**, Elsevier, v. 21, n. 13-14, p. 1145–1153, 2003.
- GUO, Y.; BENNAMOUN, M.; SOHEL, F.; LU, M.; WAN, J. 3d object recognition in cluttered scenes with local surface features: a survey. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, IEEE, v. 36, n. 11, p. 2270–2287, 2014.
- HIMRI, K.; RIDAO, P.; GRACIAS, N.; PALOMER, A.; PALOMERAS, N.; PI, R. Semantic slam for an auv using object recognition from point clouds. **IFAC-PapersOnLine**, Elsevier, v. 51, n. 29, p. 360–365, 2018.
- HOLZ, D.; ICHIM, A. E.; TOMBARI, F.; RUSU, R. B.; BEHNKE, S. Registration with the point cloud library: A modular framework for aligning in 3-d. **IEEE Robotics & Automation Magazine**, IEEE, v. 22, n. 4, p. 110–124, 2015.
- HORN, B. K. Closed-form solution of absolute orientation using unit quaternions. **Journal of the Optical Society of America A**, Optical Society of America, v. 4, n. 4, p. 629–642, 1987.
- KALDESTAD, K. B.; HADDADIN, S.; BELDER, R.; HOVLAND, G.; ANISI, D. A. Collision avoidance with potential fields based on parallel processing of 3d-point cloud data on the gpu. In: **IEEE Robotics and Automation (ICRA), 2014 IEEE International Conference on**. [S.l.], 2014. p. 3250–3257.

LOW, K.-L. Linear least-squares optimization for point-to-plane icp surface registration. **Chapel Hill, University of North Carolina**, Citeseer, v. 4, n. 10, 2004.

MAGNUSSON, M.; LILIENTHAL, A.; DUCKETT, T. Scan registration for autonomous mining vehicles using 3d-ndt. **Journal of Field Robotics**, Wiley Online Library, v. 24, n. 10, p. 803–827, 2007.

MAVRIDIS, P.; ANDREADIS, A.; PAPAIOANNOU, G. Efficient sparse icp. **Computer Aided Geometric Design**, Elsevier, v. 35, p. 16–26, 2015.

MELLADO, N.; AIGER, D.; MITRA, N. J. Super 4pcs fast global pointcloud registration via smart indexing. In: WILEY ONLINE LIBRARY. **Computer Graphics Forum**. [S.l.], 2014. v. 33, n. 5, p. 205–215.

MUJA, M.; LOWE, D. G. Fast approximate nearest neighbors with automatic algorithm configuration. **VISAPP (1)**, v. 2, n. 331-340, p. 2, 2009.

PCL. **Point cloud library**. 2018. Disponível em: <<http://pointclouds.org/>>.

PEREIRA, N. da S. **Cloud Partitioning Iterative Closest Point (CP-ICP): um estudo comparativo para registro de nuvens de pontos 3D**. Dissertação (Mestrado) — Universidade Federal do Ceará, jul 2016.

PEREIRA, N. S.; CARVALHO, C. R. *et al.* Point cloud partitioning approach for icp improvement. In: IEEE. **Automation and Computing (ICAC), 2015 21st International Conference on**. [S.l.], 2015. v. 1, n. 1, p. 1–5.

POMERLEAU, F.; COLAS, F.; SIEGWART, R.; MAGNENAT, S. Comparing icp variants on real-world data sets. **Autonomous Robots**, Springer, v. 34, n. 3, p. 133–148, 2013.

POMERLEAU, F.; COLAS, F.; SIEGWART, R. *et al.* A review of point cloud registration algorithms for mobile robotics. **Foundations and Trends® in Robotics**, Now Publishers, Inc., v. 4, n. 1, p. 1–104, 2015.

PRAKHYA, S. M.; BINGBING, L.; RUI, Y.; LIN, W. A closed-form estimate of 3d icp covariance. In: IEEE. **Machine Vision Applications (MVA), 2015 14th IAPR International Conference on**. [S.l.], 2015. p. 526–529.

RODOLÀ, E.; ALBARELLI, A.; CREMERS, D.; TORSSELLO, A. A simple and effective relevance-based point sampling for 3d shapes. **Pattern Recognition Letters**, Elsevier, v. 59, p. 41–47, 2015.

RUSINKIEWICZ, S.; LEVOY, M. Efficient variants of the icp algorithm. In: IEEE. **3-D Digital Imaging and Modeling, 2001. Proceedings. Third International Conference on**. [S.l.], 2001. v. 1, n. 1, p. 145–152.

RUSU, R. B. **Semantic 3D Object Maps for Everyday Robot Manipulation**. [S.l.]: Springer, 2013.

RUSU, R. B.; BLODOW, N.; BEETZ, M. Fast point feature histograms (fpfh) for 3d registration. In: CITESEER. **Robotics and Automation, 2009. ICRA'09. IEEE International Conference on**. [S.l.], 2009. p. 3212–3217.

- RUSU, R. B.; COUSINS, S. 3d is here: Point cloud library (pcl). In: IEEE. **Robotics and automation (ICRA), 2011 IEEE International Conference on**. [S.l.], 2011. p. 1–4.
- RUSU, R. B.; MARTON, Z. C.; BLODOW, N.; DOLHA, M.; BEETZ, M. Towards 3d point cloud based object maps for household environments. **Robotics and Autonomous Systems**, Elsevier, v. 56, n. 11, p. 927–941, 2008.
- SAVRAN, A.; SANKUR, B. Non-rigid registration based model-free 3d facial expression recognition. **Computer Vision and Image Understanding**, Elsevier, v. 162, p. 146–165, 2017.
- SCIENCE, C. A. W. of. **Pesquisas na área de registro de nuvens de pontos indexadas pelo Web of Science**. 2018. Disponível em: <http://wcs.webofknowledge.com/RA/analyze.do?product=WOS&SID=6AVC4X5RPL3IOUfXv18&field=TASCA_JCRCategories_JCRCategories_en&yearSort=false>.
- SEGAL, A.; HAEHNEL, D.; THRUN, S. Generalized-icp. In: **Robotics: science and systems**. [S.l.: s.n.], 2009. v. 2, n. 4, p. 435.
- SERAFIN, J.; GRISSETTI, G. Nicp: Dense normal based point cloud registration. In: IEEE. **Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on**. [S.l.], 2015. v. 1, n. 1, p. 742–749.
- SERAFIN, J.; OLSON, E.; GRISSETTI, G. Fast and robust 3d feature extraction from sparse point clouds. In: IEEE. **Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on**. [S.l.], 2016. p. 4105–4112.
- SIQUEIRA, R. S.; ALEXANDRE, G. R.; SOARES, J. M.; THE, G. A. P. Triaxial Slicing for 3-D Face Recognition From Adapted Rotational Invariants Spatial Moments and Minimal Keypoints Dependence. **IEEE Robotics and Automation Letters**, v. 3, n. 4, p. 3513–3520, oct 2018. ISSN 2377-3766. Disponível em: <<https://ieeexplore.ieee.org/document/8408720/>>.
- SOUZA NETO, P.; PEREIRA, N. S.; THÉ, G. A. P. Improved cloud partitioning sampling for iterative closest point : Qualitative and quantitative comparison study. In: IEEE. **15th International Conference on Informatics in Control, Automation and Robotics**. [S.l.], 2018. v. 1, n. 2, p. 978–989.
- STORK, A. Visual computing challenges of advanced manufacturing and industrie 4.0 [guest editors' introduction]. **IEEE computer graphics and applications**, IEEE, v. 35, n. 2, p. 21–25, 2015.
- TAM, G. K.; CHENG, Z.-Q.; LAI, Y.-K.; LANGBEIN, F. C.; LIU, Y.; MARSHALL, D.; MARTIN, R. R.; SUN, X.-F.; ROSIN, P. L. Registration of 3d point clouds and meshes: a survey from rigid to nonrigid. **IEEE Transactions on Visualization and Computer Graphics**, Institute of Electrical and Electronics Engineers, Inc., 3 Park Avenue, 17 th Fl New York NY 10016-5997 United States, v. 19, n. 7, p. 1199–1217, 2013.
- TAZIR, M. L.; GOKHOOL, T.; CHECCHIN, P.; MALATERRE, L.; TRASSOUDAIN, L. Cicp: Cluster iterative closest point for sparse–dense point cloud registration. **Robotics and Autonomous Systems**, Elsevier, v. 108, p. 66–86, 2018.
- UICKER, J. J.; PENNOCK, G. R.; SHIGLEY, J. E. *et al.* **Theory of machines and mechanisms**. [S.l.]: Oxford University Press New York, 2011. v. 1.

UNIVERSITY, S. **The Stanford 3D Scanning Repository**. 2014. Disponível em: <<http://graphics.stanford.edu/data/3Dscanrep/>>.

VITTER, J. S. Faster methods for random sampling. **Communications of the ACM**, ACM, v. 27, n. 7, p. 703–718, 1984.

WEINMANN, M. **Reconstruction and analysis of 3D scenes**. [S.l.]: Springer, 2016.

WULF, O. **Robotic 3D Scan Repository**. 2016. Disponível em: <<http://kos.informatik.uni-osnabrueck.de/3Dscans/>>.

YANG, J.; LI, H.; CAMPBELL, D.; JIA, Y. Go-icp: A globally optimal solution to 3d icp point-set registration. **IEEE transactions on pattern analysis and machine intelligence**, IEEE, v. 38, n. 11, p. 2241–2254, 2015.

ZHANG, Z. Iterative point matching for registration of free-form curves and surfaces. **International journal of computer vision**, Springer, v. 13, n. 2, p. 119–152, 1994.