

## RESEARCH ARTICLE

## OPEN ACCESS

## REFINAMENTO GLOBAL, LINEAR E FECHADO (CLOSED-FORM) DO REGISTRO DE NUVENS DE PONTOS 3D POR INTERPOLAÇÃO DE ROTAÇÕES

Rubens A. L. Benevides\*<sup>1</sup>, Daniel Rodrigues dos Santos<sup>2</sup> and Nadisson Pavan<sup>1</sup>

<sup>1</sup>Mestre pelo Programa de Pós Graduação em Ciências Geodésicas (PPGCG) da Universidade Federal do Paraná (UFPR), Curitiba-PR, Brasil; <sup>2</sup>Seção de Cartografia, Instituto de Militar de Engenharia (IME), Rio de Janeiro-RJ, Brasil

### ARTICLE INFO

#### Article History:

Received 21<sup>st</sup> September, 2022

Received in revised form

29<sup>th</sup> September, 2022

Accepted 06<sup>th</sup> October, 2022

Published online 30<sup>th</sup> October, 2022

#### Key Words:

Registro de nuvens de pontos, Otimização global, SLERP, Laser Scanner Terrestre.

#### \*Corresponding author:

Rubens A. L. Benevides,

### ABSTRACT

O refinamento global do registro de nuvens de pontos 3D é um problema constante para sensores que necessitam registrar múltiplos pares de nuvens de pontos 3D, uma vez que erros de deriva são gerados resultando em desalinhamentos angulares entre a posição real e a calculada pelo sensor. No tratamento destes erros, lugares revisitados são utilizados para correções globais na trajetória. Todavia, os modelos de correção disponíveis são iterativos e dependentes de parâmetros de transformação iniciais. Neste sentido, propomos um modelo totalmente linear capaz de refinar, simultaneamente, todas as poses 3D de um circuito fechado em uma única etapa. Para isto, as rotações das poses são mapeadas em quatérnios, que são interpolados linearmente. Os intervalos de interpolação são deduzidos pelo Método dos Mínimos Quadrados (MMQ). As translações também são globalmente ajustadas por MMQ de forma linear em uma etapa independente. A combinação de ambos os modelos foi testada em três *datasets* de Laser Scanner Terrestre (LST). Os resultados obtidos mostraram que o método proposto reduziu significativamente os erros de cada pose absoluta do LST.

Copyright © 2022, Rubens A. L. Benevides et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Citation: Rubens A. L. Benevides, Daniel dos Santos and Nadisson Pavan. 2022. "Refinamento global, linear e fechado (closed-form) do registro de nuvens de pontos 3d por interpolação de rotações", *International Journal of Development Research*, 12, (11), 59926-59934.

## INTRODUCTION

No campo da robótica e da visão computacional, o registro de nuvens de pontos 3D consiste em alinhar duas ou mais nuvens de pontos 3D. Isso normalmente é feito para combinar várias leituras do sensor em uma única leitura consistente e precisa. No formalismo matemático o problema do registro é definido como: dado um par de nuvens de pontos  $M$  (móvel) e  $S$  (fixa) em  $\mathbb{R}^3$ , encontrar a transformação  $T$  minimize a distância entre  $M$  e  $S$ :

$$T = \sum_{i=1}^n ||T(M_i) - S_i|| \quad (1)$$

Onde  $n$  é a quantidade de pontos correspondentes e  $T$  é uma matriz do grupo algébrico  $SE(3)$ , que representa em coordenadas homogêneas uma transformação de corpo rígido 3D. Na literatura,  $T$  é abreviada como pose e consiste de uma matriz organizada como segue:

$$SE(3) = \left\{ \begin{array}{l} T | T = \begin{bmatrix} R & t \\ 0_{1 \times 3} & 1 \end{bmatrix}, R \in \mathbb{R}^{3 \times 3}, \\ t \in \mathbb{R}^3, R^T R = R R^T = I, |R| = 1 \end{array} \right\} \quad (2)$$

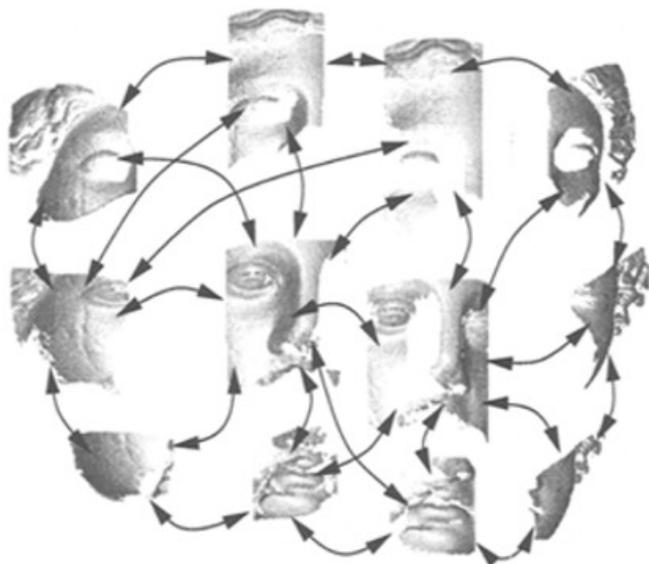
Devido ao grande número de correspondências possíveis que podem ser estabelecidas entre duas nuvens de pontos, o registro de pares de

nuvens de pontos é um desafio. Como resultado, muitas estratégias foram propostas para resolver esse problema, que em geral, primeiro é resolvido aos pares. Neste contexto, os métodos mais comuns são as variações do algoritmo ICP (*Iterative Closest Point*), originalmente proposto em (CHEN; MEDIONI, 1992) e em (BESL; MCKAY, 1992). O ICP é um método iterativo, mas é pouco robusto e bastante suscetível a mínimos locais da equação 1, resultando em sobreposições incorretas entre os pares de nuvens de pontos, caso o alinhamento inicial esteja distante da configuração correta. Inúmeras variações do algoritmo foram propostas para contornar essa limitação (RUSINKIEWICZ; LEVOY, 2001); (CHOI et al., 2020); (HUANG et al., 2022). Contudo, o ICP nunca deixou de ser um método local, por isso é majoritariamente utilizado como refinamento dos métodos baseados em descritores locais. Os métodos baseados em descritores locais buscam correspondências entre feições geométricas (pontos, retas, planos) contidas nas nuvens, e são mais confiáveis, embora mais lentos e menos precisos devido ao uso reduzido de pontos, sendo por isso chamados de métodos grosseiros (DIEZ et al., 2015). Uma combinação dessas diferentes estratégias geralmente permite bons resultados. Quando o ICP é utilizado como etapa de refinamento de um método grosseiro, o método de registro é chamado *coarse-to-fine*, e isto permite uma infinidade de métodos para se registrar pares de nuvem de pontos 3D. Todavia, nenhuma dessas combinações é solução para o acúmulo de erros que ocorre quando se tem múltiplas

nuvens de pontos a serem registradas. Isto porque as transformações calculadas entre os pares de nuvens de pontos sempre possuem erros, que cresce de acordo com o número de nuvens registradas em sequência, dando origem aos problemas de deriva (*drift*) que também é muito comum em plataformas que operam em modo de Mapeamento e Localização Simultâneos (*Simultaneous Localization and Mapping -SLAM*), dado que nestas o registro de centenas de nuvens de pontos 3D é algo comum (HUANG *et al.*, 2021). Neste trabalho apresentamos um método para a correção deste erro que se baseia em circuitos fechados. O método parte do pressuposto que um determinado sensor, ao visitar um determinado local, deve zerar seu erro de deriva ao fechar o circuito. O modelo é deduzido de forma totalmente linear e fechada, o que significa que nenhuma iteração precisa ser feita e nenhuma matriz deve ser invertida. O modelo é testado no registro de nuvens de pontos 3D obtidas por Laser Scanner Terrestre (LST).

## REVISÃO DE LITERATURA

A problemática da reconstrução 3D com múltiplas nuvens de pontos está no fato de que o espaço para otimização é muito amplo (Figura 1). Assim sendo, a literatura apresenta soluções divididas em três etapas: (1) constrói-se o grafo de estações seguindo a sobreposição entre nuvens; (2) estabelece-se um método para registro dos pares; (3) aplica-se um MRG no circuito de poses relativas obtido a partir do grafo. Neste trabalho, o método proposto consiste no refinamento simultâneo de todas as nuvens do *dataset*.



Fonte: (BENJEMAA; SCHMITT, 1998)

**Figura 1. Problemática da reconstrução 3D com múltiplas nuvens de pontos. Cada par de nuvens de pontos é um problema de otimização, isto define um enorme espaço de possibilidades na solução**

Segundo (HUBER; HEBERT, 2003) os trabalhos que investigam a otimização da reconstrução 3D são categorizados em quatro categorias, conforme a Tabela 1.

**Tabela 1. Classificação dos problemas envolvidos na reconstrução de um *dataset* de nuvens de pontos 3D**

A sobreposição entre pares é conhecida?	Sim	Não	
Nº de nuvens = 2	<i>Pair-wise registration</i>	<i>Pair-wise matching</i>	<i>surface</i>
Nº de nuvens > 2	<i>Multi-view registration</i>	<i>Multi-view matching</i>	<i>surface</i>

Fonte: autores (2022).

A seguir, trabalhos do tipo *Multi-view registration* e *Multi-view surface matching* são discutidos e contextualizados. Como dito

anteriormente, a composição multiplicativa de poses para referenciar todas as nuvens em uma única origem gera o erro de deriva (*drift*), que sempre cresce com a trajetória do sensor. Para distribuir este erro na rede global de nuvens existem os Modelos de Refinamento Global (MRG), normalmente presentes no *backend* de métodos SLAM (CADENA *et al.*, 2016). Uma das formas do MRG operar nas poses é representando-as por grafos  $g = (v, a)$  onde cada vértice  $v$  representa uma estação do sensor e cada aresta  $a$ , uma pose relativa entre duas estações. Estes grafos definem um problema de otimização extremamente não linear (GRISSETTI *et al.*, 2010), (KÜMMERLE *et al.*, 2011), (NEUGEBAUER, 1997), (WILLIAMS; BENNAMOUN, 1999), (WILLIAMS; BENNAMOUN, 2001). A solução varia de acordo com a topologia do grafo, em grafos acíclicos são explorados caminhos de menor erro, e.g. Árvores de Expansão Mínimas (*Minimum Spanning Tree*), como feito em (HUBER; HEBERT, 2003) que em seguida aplica o MRG de (NEUGEBAUER, 1997). Para grafos que formam ciclos, a detecção de lugares revisitados permite distribuir o *drift* propagando-o para poses anteriores ao impor uma restrição de fechamento do circuito (*loop-closure*). Em (THEILER *et al.*, 2015) estas duas abordagens são combinadas no grafo completo de estações do LST aplicando o MRG descrito em (BORRMANN *et al.*, 2008), que é uma generalização de (LU; MILIOS, 1997) conhecida como modelo LUM para casos tridimensionais. Para evitar o enorme espaço de busca dos grafos completos, (DONG *et al.*, 2018) adotaram vetores de descritores na construção do grafo hierárquico de poses e em seguida aplicaram o MRG de (WILLIAMS; BENNAMOUN, 1999). Em (CHOI *et al.*, 2015), trabalhando com dados RGB-D, a estatística robusta descrita em (BLACK; RANGARAJAN, 1996) é aplicada no grafo de pares em conjunto com o MRG *graph-2-optimization* ( $g^2o$ ) de (KÜMMERLE *et al.*, 2011).

O  $g^2o$  é um dos *frameworks* mais bem estabelecidos na literatura de otimização de poses e está presente em bibliotecas de rápido crescimento, como a Open3D (ZHOU *et al.*, 2018), além disso foi incorporado à toolbox de SLAM-LIDAR do Matlab® em 2020. Trabalhando com nuvens 3D do Kinect®, (WEBER *et al.*, 2015) refina globalmente o grafo de poses do sensor ao ponderar as arestas segundo correspondências encontradas com o descritor FPFH, de (RUSU *et al.*, 2009). No contexto de Mobile Laser Scanning (MLS), (VLAMINCK *et al.*, 2019) registram pares de nuvens 3D com o algoritmo 4PCS de (AIGER *et al.*, 2008) e definem um MRG que distribui o erro nas rotações com a técnica de interpolação linear esférica (*Spherical Linear Interpolation - SLERP*) (SHOEMAKE, 1985), para isto adotaram uma métrica entre poses. Em (PAVAN; DOS SANTOS, 2017) circuitos fechados do grafo de poses do LST são aproveitados para refinar as rotações globalmente por decomposição em valores singulares (*Singular Value Decomposition-SVD*). Em seguida é aplicado o MRG LUM de (LU; MILIOS, 1997) para ajustar os parâmetros de translação. Em muitos destes trabalhos, a abordagem é iterativa e dependente de valores iniciais dos parâmetros de transformação. Vale lembrar que nossa contribuição é propor um MRG linear. Nosso MRG opera fechando o circuitos de poses de forma similar a (VLAMINCK *et al.*, 2019), mas sem necessidade de definir uma métrica no espaço das poses ou de solucionar sistemas por SVD como feito em (PAVAN; DOS SANTOS, 2017).

## MATERIAIS E MÉTODOS

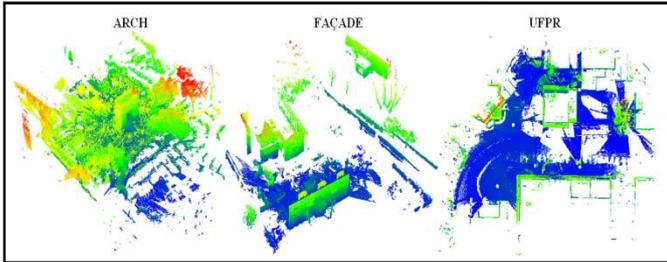
Três *datasets* foram utilizados para teste do MRG proposto, são eles: *dataset* (1) *Arch*, é composto por 5 nuvens de pontos em ambiente urbano ao redor do Arco de Tito, em Roma, nele há uma elevada presença de vegetação e artefatos móveis nas nuvens. As estações são esparsas e apenas cinco delas ao longo de 70 m de circuito, isto faz com que a sobreposição entre as nuvens seja baixa, em média, 50%; *dataset* (2): *Facade*, este *dataset* possui 7 nuvens de pontos e retrata o levantamento de uma fachada, há artefatos móveis, carros, pessoas e vegetação de forma pouco significativa. A sobreposição média entre os pares é de 60%, os pares se distribuem em um circuito irregular de 28.7 m. Estes dois primeiros *datasets* pertencem a (THEILER *et al.*,

2015). O terceiro *dataset* é: (3) UFPR, foi obtido pelos autores na Universidade Federal do Paraná, este *dataset* define um caso extremo onde a posição entre as estações são muito esparsas e praticamente não há sobreposição entre as nuvens de pontos. Este *dataset* conta com 8 estações distribuídas em 169 m de circuito. A Tabela 2 resume as características de cada *dataset* descrito.

**Tabela 2. Datasets utilizados para teste do MRG proposto**

Dataset	LST	nuvens	Pts./nuvem	Sobreposição Média (%)
Arch	Faro Focus 3D	5	$\cong 20 M$	40
Facade	Faro Focus 3D	7	$\cong 20 M$	60
UFPR	Leica BLK360	8	$\cong 20M$	35

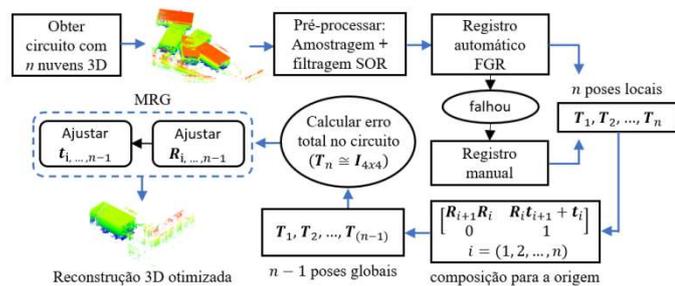
Fonte: autores (2022).



Fonte: (THEILET et al., 2015); autores (2022)

**Figura 2. Datasets de nuvens obtidas por LST para teste do MRG proposto**

Todas as nuvens de pontos foram manipuladas com scripts escritos em Python-3 utilizando a biblioteca Open 3D (ZHOU et al., 2018). A Figura 3 apresenta o fluxograma do MRG proposto.



Fonte: autores (2022).

**Figura 3. Fluxograma dos dados até aplicação do MRG proposto**

Na Figura 3, como os dados de LST são muito densos, primeiro é feito um pré-processamento dos dados, ele consiste em uma filtragem e amostragem das nuvens com o objetivo de facilitar o registro dos pares de nuvens de pontos. A amostragem se dá calculando o centroide com voxels de 10 cm e a filtragem foi feita com o algoritmo *Statistical Outlier Removal* (SOR) de (RUSU et al., 2009). Na tentativa de registro grosseiro dos pares de nuvens empregou-se o algoritmo *Fast Global Registration* (FGR) de (ZHOU et al., 2016). De posse das poses locais faz-se a composição multiplicativa para poses globais, para isto fixa-se uma nuvem como inicial definindo sua pose  $T$  como a identidade. Em seguida é aplicado o MRG proposto. O MRG proposto opera em um circuito fechado em duas etapas, primeiro é feito o refinamento global das rotações, depois, o refinamento global das translações. A seguir apresenta-se, de forma breve, os quatérnios e a técnica SLERP. Depois faz-se a demonstração do modelo com o Método dos Mínimos Quadrados aplicado nesta técnica. O modelo de ajuste das translações é apresentado logo em seguida. Adota-se a notação matemática padrão, valores escalares são representados por letras minúsculas, vetores são representados por letras minúsculas em negrito e matrizes em letras maiúsculas em negrito. Multiplicações entre elementos não comutativos são feitas da esquerda para a direita. Todo e qualquer sistema de coordenadas é assumido dextrógiro.

**Quatérnios & SLERP:** Quatérnios são definidos na matemática como números de quatro dimensões com 3 unidades imaginárias e uma real. A base canônica dos quatérnios é formada por  $i, j, k$  e  $1$ , sendo os três primeiros elementos definidos por  $i = j = k = \sqrt{-1}$ . As propriedades dos quatérnios são construídas a partir das multiplicações entre os elementos de sua base, a saber (NEVES, 2008):

$$\begin{aligned} ij &= k = -ji \\ jk &= i = -kj \\ ki &= j = -ik \end{aligned} \quad (3)$$

logo, o produto entre quatérnios não é comutativo. Um quatérnio  $q$  é escrito como  $q = q_0 + q_x i + q_y j + q_z k$  onde  $q_0, q_x, q_y, q_z \in \mathbb{R}$ . A multiplicação de dois quatérnios  $q$  e  $p$  pode ser feita com a propriedade distributiva. O conjugado  $q^*$  de um quatérnio qualquer é obtido negando-se a parte imaginária:  $q^* = q_0 - q_x i - q_y j - q_z k$ . Um quatérnio é dito unitário se a sua norma-2 é igual a 1:

$$\|q\| = \sqrt{q_0^2 + q_x^2 + q_y^2 + q_z^2} = 1 \quad (4)$$

Assim como as matrizes, quatérnios podem ser utilizados para parametrizar rotações. Uma rotação  $r_{(\theta, n)}$ , de ângulo  $\theta$  no sentido anti-horário em torno do eixo unitário  $n = (n_x, n_y, n_z)^T$ , pode ser mapeada para um quatérnio fazendo-se:

$$q = \cos \frac{\theta}{2} + \text{sen} \frac{\theta}{2} n \quad (5)$$

assim, quatérnios são mais eficientes para representar rotações, pois utilizam apenas 4 parâmetros, e são mais estáveis numericamente, pois obedecem apenas à restrição de ser unitário. Matrizes de rotação operam com 9 parâmetros e devem obedecer a 6 restrições para serem ortonormais (SHOEMAKE, 1985). Existe um homomorfismo entre o espaço algébrico dos quatérnios unitários  $S^3$  e o espaço das matrizes de rotação,  $SO(3)$ , que estabelece uma relação de 2 para 1 entre os seus elementos. Para cada matriz de rotação 3D codificada em uma matriz  $R$  existem dois quatérnios,  $q$  e  $-q$ , que a representam. Na linguagem da álgebra de grupos se diz que o espaço  $SO(3)$  é duplamente coberto por  $S^3$  (DAM et al., 1998). Este homomorfismo entre os espaços  $SO(3)$  e  $S^3$  permite mapear quatérnios para matrizes de rotação com a fórmula:

$$R = \begin{bmatrix} 1 - 2q_y^2 - 2q_z^2 & 2q_x q_y - 2q_z q_0 & 2q_x q_z + 2q_y q_0 \\ 2q_x q_y + 2q_z q_0 & 1 - 2q_x^2 - 2q_z^2 & 2q_y q_z - 2q_x q_0 \\ 2q_x q_z - 2q_y q_0 & 2q_y q_z + 2q_x q_0 & 1 - 2q_x^2 - 2q_y^2 \end{bmatrix} \quad (6)$$

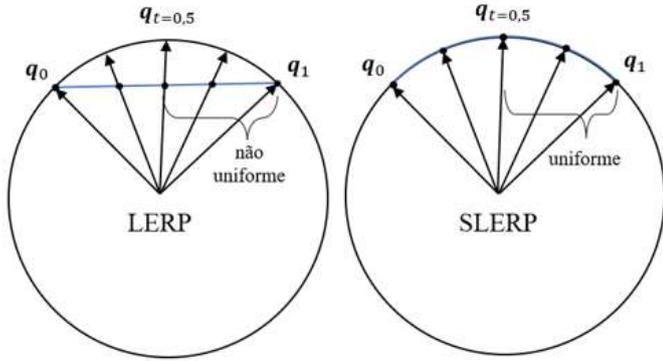
E para mapear de volta matrizes de rotação para quatérnios basta calcular o vetor unitário  $n$  e o ângulo  $\theta$  de rotação a partir dos elementos  $r_{ij}$  da matriz  $R$  como segue:

$$\begin{aligned} n_x &= \sqrt{r_{11} + r_{21} + r_{31}} \\ n_y &= \sqrt{r_{21} + r_{22} + r_{23}} \\ n_z &= \sqrt{r_{31} + r_{32} + r_{33}} \end{aligned} \quad (7)$$

$$|\theta| = \cos^{-1} \left( \frac{\text{tr}(R) - 1}{2} \right)$$

onde  $\text{tr}(R)$  é o traço da matriz, e então aplicar na eq. 6 para obter o quatérnio correspondente. A parametrização das rotações por quatérnios leva a fórmulas simples de interpolação entre duas rotações. Dadas duas matrizes de rotação distintas  $R_0$  e  $R_1$ , mapeadas em quatérnios  $q_0$  e  $q_1$ , para encontrar a rotação  $q_\tau$ , interpolada a uma distância  $\tau$  entre  $q_0$  e  $q_1$ , com  $\tau \in [0, 1]$ , pode-se fazer  $q_\tau = (1 - \tau)q_0 + \tau q_1$ . Todavia, essa interpolação linear chamada LERP (Linear InterPolation) não preserva a velocidade angular das rotações, pois

não considera a geometria intrínseca do grupo  $S^3$ . Como o conjunto dos quatérnios unitários define uma esfera unitária em  $\mathbb{R}^4$  (hiperesfera), a interpolação apropriada entre dois quatérnios tem de ser ao longo de um arco de circunferência máxima nesta esfera, o que garante intervalos regularmente espaçados entre os quatérnios, tal técnica se chama SLERP, e trás o benefício da velocidade angular constante nas rotações interpoladas. A Figura 4 mostra a interpretação geométrica das técnicas LERP e SLERP em um corte da hiperesfera unitária  $\mathbb{R}^4$ .



Fonte: autores (2022).

**Figura 4. Interpretação geométrica da interpolação LERP e SLERP entre dois quatérnios  $q_0$  e  $q_1$**

Realizar a SLERP entre dois quatérnios  $q_0$  e  $q_1$  é simples, primeiro encontra-se o ângulo  $\Omega$  entre eles com o produto interno:

$$\Omega = \arccos(q_0 \cdot q_1) \quad (8)$$

o quatérnio  $q_\tau$  a uma posição angular intermediária  $\alpha$  entre  $q_0$  e  $q_1$  é dado por:

$$q_\tau = q_0 \frac{\text{sen}(\Omega - \alpha)}{\text{sen}(\Omega)} + q_1 \frac{\text{sen}(\alpha)}{\text{sen}(\Omega)} \quad (9)$$

substituindo  $\alpha$  por  $\tau\Omega$ , onde  $\tau \in [0,1]$ , obtém-se (SHOEMAKE, 1985):

$$q_\tau = q_0 \frac{\text{sen}((1 - \tau)\Omega)}{\text{sen}(\Omega)} + q_1 \frac{\text{sen}(\tau\Omega)}{\text{sen}(\Omega)} \quad (10)$$

a equação acima é útil na implementação, mas a notação exponencial é mais concisa e será usada na demonstração do modelo, a saber:

$$q_\tau = q_0 (q_0^{-1} q_1)^\tau \quad (11)$$

**Ajuste Global das Rotações:** Definido um circuito fechado de estações do LST, as matrizes de rotação  $R_i$  das poses relativas são mapeadas para quatérnios unitários, que então são compostos para a origem global seguindo a sequência de multiplicações a seguir:

$$\begin{aligned} r_0 &= q_I \\ r_1 &= q_1 r_0 \\ &\vdots \\ r_n &= q_n r_{n-1} \\ r_{n+1} &= q_{n+1} r_n \end{aligned} \quad (12)$$

onde,  $q_I$  é a identidade dos quatérnios  $q_I = (1 + 0i + 0j + 0k)$ ;  $r_i$  é o quatérnio que representa a rotação global do vértice  $X_i$  referenciado ao vértice  $X_0$ , com  $i = 1, 2, \dots, n$ ;  $q_i$  é o quatérnio que representa a rotação relativa do vértice  $X_i$  referenciado ao vértice  $X_{i-1}$ , com  $i = 1, 2, \dots, n$ ;  $q_{n+1}$  é o quatérnio representa a rotação relativa do vértice  $X_0$  para  $X_n$  que fecha o circuito (*loop-closure*). Se as estimativas das rotações não apresentassem erros, o quatérnio  $r_{n+1}$

seria igual à identidade  $q_I$ . Na prática, esse quatérnio apresenta uma aproximação com a identidade inversamente proporcional aos erros no circuito. Essas sequências de rotações podem ser feitas no sentido contrário da trajetória do sensor, as rotações dos vértices  $X_1, X_2, \dots, X_n$ , referenciadas ao vértice  $X_0$  podem ser obtidas pelo inverso das seguintes composições:

$$\begin{aligned} p_1 &= (q_2, q_3, \dots, q_{n+1})^{-1} \\ p_2 &= (q_3, q_4, \dots, q_{n+1})^{-1} \\ &\vdots \\ p_n &= (q_{n+1})^{-1} \end{aligned} \quad (13)$$

ond e  $p_i$  é o quatérnio que representa a rotação global do vértice  $X_i$  referenciado ao vértice  $X_0$ , com  $i = 1, 2, \dots, n$ . Para otimizar as rotações os quatérnios são interpolados com a SLERP, como segue:

$$\begin{aligned} p_1 &= (q_2, q_3, \dots, q_{n+1})^{-1} \\ p_2 &= (q_3, q_4, \dots, q_{n+1})^{-1} \\ &\vdots \\ p_{n-1} &= (q_n, q_{n+1})^{-1} \\ p_n &= (q_{n+1})^{-1} \end{aligned} \quad (14)$$

onde,  $\hat{q}_i$  é o quatérnio interpolado entre  $r_i$  e  $p_i$  que representa a rotação global ótima do vértice  $X_i$  referenciado ao vértice  $X_0$ , com  $i = 1, 2, \dots, n$ ; e  $\tau_i \in [0,1]$  é o intervalo de interpolação com  $i = 1, 2, \dots, n$ . As propriedades de multiplicação dos quatérnios permitem escrever  $p_i$  como:

$$\begin{aligned} p_1 &= r_1 r_{n+1}^{-1} \\ p_2 &= r_2 r_{n+1}^{-1} \\ &\vdots \\ p_n &= r_n r_{n+1}^{-1} \end{aligned} \quad (15)$$

substituindo a eq. 15 na 14 se obtém:

$$\begin{aligned} \hat{q}_1 &= r_1 (r_1^{-1} r_1 r_{n+1}^{-1})^{\tau_1} = r_1 r_{n+1}^{-\tau_1} \\ \hat{q}_2 &= r_2 (r_2^{-1} r_2 r_{n+1}^{-1})^{\tau_2} = r_2 r_{n+1}^{-\tau_2} \\ &\vdots \\ \hat{q}_n &= r_n (r_n^{-1} r_n r_{n+1}^{-1})^{\tau_n} = r_n r_{n+1}^{-\tau_n} \end{aligned} \quad (16)$$

para otimizar os parâmetros  $\tau_i$ , toma-se vantagem da restrição de que o inverso de  $\hat{q}_i$ , ao ser multiplicado por  $q_i \hat{q}_{i-1}$ , deve se aproximar o máximo possível do quatérnio identidade, como segue:

$$\begin{aligned} \hat{q}_1^{-1} q_1 &= (r_1 r_{n+1}^{-\tau_1})^{-1} q_1 \approx q_I \\ \hat{q}_2^{-1} q_2 \hat{q}_1 &= (r_2 r_{n+1}^{-\tau_2})^{-1} q_2 r_1 r_{n+1}^{-\tau_1} \approx q_I \\ &\vdots \\ \hat{q}_n^{-1} q_n \hat{q}_{n-1} &= (r_n r_{n+1}^{-\tau_n})^{-1} q_n r_{n-1} r_{n+1}^{-\tau_{n-1}} \approx q_I \\ q_{n+1} \hat{q}_n &= q_{n+1} r_n r_{n+1}^{-\tau_n} \approx q_I \end{aligned} \quad (17)$$

realizando as devidas inversões de quatérnios e a substituição da eq. 12 na eq. 17, as equações acima são reescritas como:

$$\begin{aligned} \hat{q}_1^{-1} q_1 &= r_{n+1}^{\tau_1} r_1^{-1} r_1 = r_{n+1}^{\tau_1} \approx q_I \\ \hat{q}_2^{-1} q_2 \hat{q}_1 &= r_{n+1}^{\tau_2} r_2^{-1} r_2 r_{n+1}^{-\tau_1} = r_{n+1}^{\tau_2} r_{n+1}^{-\tau_1} \approx q_I \\ &\vdots \\ \hat{q}_n^{-1} q_n \hat{q}_{n-1} &= r_{n+1}^{\tau_n} r_n^{-1} r_n r_{n+1}^{-\tau_{n-1}} = r_{n+1}^{\tau_n} r_{n+1}^{-\tau_{n-1}} \approx q_I \\ q_{n+1} \hat{q}_n &= r_{n+1} r_{n+1}^{-\tau_n} \approx q_I \end{aligned} \quad (18)$$

utilizando a propriedade da multiplicação de potência de quatérnios  $q^{\tau_1} q^{\tau_2} = q^{\tau_1 + \tau_2}$ , válida para a multiplicação de dois quatérnios iguais, segue-se que:

$$\begin{aligned} \hat{q}_1^{-1} q_1 &= r_{n+1}^{\tau_1} r_1^{-1} r_1 = r_{n+1}^{\tau_1} \approx q_I \\ \hat{q}_2^{-1} q_2 \hat{q}_1 &= r_{n+1}^{\tau_2} r_2^{-1} r_2 r_{n+1}^{-\tau_1} = r_{n+1}^{\tau_2} r_{n+1}^{-\tau_1} \approx q_I \\ &\vdots \\ \hat{q}_n^{-1} q_n \hat{q}_{n-1} &= r_{n+1}^{\tau_n} r_n^{-1} r_n r_{n+1}^{-\tau_{n-1}} = r_{n+1}^{\tau_n} r_{n+1}^{-\tau_{n-1}} \approx q_I \\ q_{n+1} \hat{q}_n &= r_{n+1} r_{n+1}^{-\tau_n} \approx q_I \end{aligned} \quad (19)$$

assim como os números reais, todo quatérnio elevado a 0 deve ser igual à identidade. Com isto é conjecturado que para obter intervalos SLERP ótimos é necessário minimizar o sistema de equações formado nas potências do lado direito da eq. 19. Este é um sistema de equações lineares simples, como segue:

$$\begin{aligned} \tau_1 &= \theta_1 \\ \tau_2 - \tau_1 &= \theta_2 \\ &\vdots \\ \tau_n - \tau_{n-1} &= \theta_{n-1} \\ 1 - \tau_n &= \theta_n \end{aligned} \tag{20}$$

para essa tarefa é aplicado o modelo paramétrico linear do MMQ. A solução que minimiza esse sistema é dada por:

$$\hat{\tau} = (J^T J)^{-1} J^T \mathbf{b} \tag{21}$$

onde  $J$  é a jacobiana da eq. 10 em relação às variáveis  $\tau_1, \tau_2, \dots, \tau_n$ ; e  $\mathbf{b}$  é o vetor  $\theta_1, \theta_2, \dots, \theta_n$ . A solução fornece o vetor  $\hat{\tau} = (\tau_1, \tau_2, \dots, \tau_n)$  de intervalos da SLERP. O formato da eq. 20 é função apenas da quantidade de estações, logo, um circuito fechado de  $n$  vértices define  $n - 1$  equações, como  $J$  e  $\mathbf{b}$  seguem um padrão simples, o vetor  $\hat{\tau}$  é previamente conhecido:

$$\hat{\tau} = [\tau_1 \quad \tau_2 \quad \tau_3 \quad \tau_4] = \left[ \frac{1}{5} \quad \frac{2}{5} \quad \frac{3}{5} \quad \frac{4}{5} \right] \tag{22}$$

Os intervalos ótimos da SLERP entre  $\mathbf{r}_i$  e  $\mathbf{p}_i$  deduzidos por MMQ mostram uma subdivisão linear do intervalo  $[0,1]$  em  $n$  partes. Isto era esperado dado que  $\mathbf{r}_i$  e  $\mathbf{p}_i$  acumulam quantidades de multiplicações diferentes. Os intervalos de interpolação  $\tau_i$  são conhecidos previamente em função da quantidade de vértices no circuito porque o modelo é linear. Como dito na introdução, esta solução é obtida de forma fechada (*closed-form*) e não necessita de linearizações, iteratividade ou definição de parâmetros. Até o presente momento isto é algo que os autores desconhecem na literatura de otimização de poses.

**Ajuste Global das Translações:** O refinamento global das translações é similar ao que ocorre em (LU e MILIOS, 1997), mas adaptado para 3 graus de liberdade conforme (BORRMANN et al., 2008). Em um circuito fechado cada translação absoluta também pode ser obtida em duas direções. Portanto, para ajustar as translações, primeiro se escrevem as translações referenciadas na origem em função das rotações, como segue:

$$\begin{aligned} {}^1_0\mathbf{t} &= \mathbf{R}_1 {}^1_0\mathbf{t} + \mathbf{t}_1 \\ {}^2_0\mathbf{t} &= \mathbf{R}_1 {}^2_1\mathbf{t} + {}^1_0\mathbf{t} \\ &\vdots \\ {}^n_0\mathbf{t} &= \mathbf{R}_{(n)n-1} {}^n_{n-1}\mathbf{t} + {}^{n-1}_0\mathbf{t} \\ {}^0_0\mathbf{t} &= \mathbf{R}_{(n+1)} {}^{n+1}_n\mathbf{t} + {}^n_0\mathbf{t} \end{aligned} \tag{23}$$

Na equação acima,  ${}^i_0\mathbf{t}$  é a translação global do vértice  $X_i$  referenciado ao vértice  $X_0$ , com  $i = 1, 2, \dots, n$ ;  $\mathbf{R}_i$  é a identidade do grupo  $SO(3)$ ;  $\mathbf{t}_i$  é a translação identidade  $[0,0,0]$ ;  ${}^{i-1}_i\mathbf{t}$  é a translação relativa do vértice  $X_i$  referenciado ao vértice  $X_{i-1}$ , com  $i = 1, 2, \dots, n$ ; e  ${}^0_0\mathbf{t}$  é a translação que representa o erro de fechamento do circuito, caso não houvessem erros,  ${}^0_0\mathbf{t}$  seria igual a  $\mathbf{t}_1$ . Em seguida, cada equação acima é derivada em relação às translações relativas, o que em um circuito fechado da origem à hipermatriz jacobiana a seguir:

$$\mathbf{A} = \begin{bmatrix} \mathbf{I} & 0 & 0 & 0 \\ -\mathbf{I} & \mathbf{I} & 0 & 0 \\ 0 & -\mathbf{I} & \mathbf{I} & 0 \\ 0 & 0 & -\mathbf{I} & \mathbf{I} \\ 0 & 0 & 0 & -\mathbf{I} \end{bmatrix}; \mathbf{I} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{24}$$

e o vetor de observações  $\mathbf{L}$ , que integrará o modelo de otimização, é organizado concatenando-se verticalmente as translações absolutas (eq. 22). Como o objetivo deste trabalho é integrar o MRG das

translações com o MRG das rotações, as translações que compõem o vetor  $\mathbf{L}$  são obtidas utilizando as rotações ajustadas, para isto mapeiam-se os quatérnios  $\hat{\mathbf{q}}_i$  para matrizes  $\hat{\mathbf{R}}_i$  que, quando multiplicadas pelas translações relativas, resultam em translações absolutas otimizadas, como segue:

$$\mathbf{l} = \begin{bmatrix} \mathbf{R}_1 {}^1_0\mathbf{t} \\ \hat{\mathbf{R}}_1 {}^1_1\mathbf{t} \\ \hat{\mathbf{R}}_2 {}^2_2\mathbf{t} \\ \vdots \\ \hat{\mathbf{R}}_{(n)n+1} {}^n_n\mathbf{t} \end{bmatrix} \tag{25}$$

a solução por MMQ com base na restrição de fechamento do circuito, é dada por:

$$\hat{\mathbf{t}} = (\mathbf{A}^T \mathbf{A})^{-1} (\mathbf{A}^T \mathbf{L}) \tag{26}$$

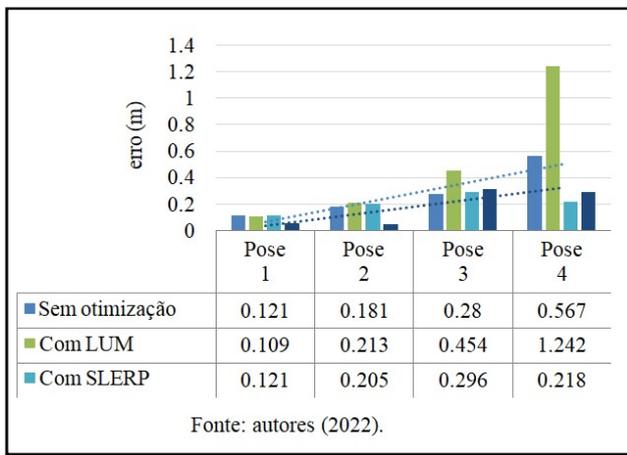
onde  $\hat{\mathbf{t}}$  é o vetor de translações absolutas duplamente refinadas  $\hat{\mathbf{t}} = (\hat{\mathbf{t}}_{10}, \hat{\mathbf{t}}_{20}, \hat{\mathbf{t}}_{30}, \hat{\mathbf{t}}_{40})$ . Concatenando-se estas com as matrizes de rotação ajustadas obtém-se poses totalmente otimizadas em uma única iteração, pois esse sistema, assim como o anterior, é linear.

## RESULTADOS

Nesta etapa, a diferença que o refinamento faz na pose de cada *dataset* é avaliada. O erro de uma pose é a diferença entre ela e a pose *groundtruth*. Diferenças entre poses tem uma componente de translação  $\epsilon \mathbf{t}_i = \|\mathbf{t}_i^g - \hat{\mathbf{t}}_i\|$  e uma de rotação  $\epsilon \mathbf{R}_i = \|\mathbf{R}_i^g - \hat{\mathbf{R}}_i\|$ . O superíndice  $g$  refere-se às poses *groundtruth* do respectivo *dataset*, e  $i$  varia de 1 até a  $n$ -ésima pose do *dataset*. O erro na translação é apenas uma distância euclidiana. O erro na rotação é a norma-2 da diferença entre as matrizes de rotação, isto coincide com a distância Frobenius no grupo  $SO(3)$ , definindo uma métrica no espaço das rotações com  $\epsilon \mathbf{R} \in [0, 2\sqrt{2}]$  (HUYNH, 2009). A seguir, cada *dataset* é analisado individualmente de forma quantitativa e qualitativa. O refinamento das rotações afeta as translações, mas o ajuste de translações não tem impacto nas rotações, então consideram-se quatro etapas na análise do erro de translação: (a) sem refinamento; (b) com refinamento das translações (LU e MILIOS, 1998); (c) com refinamento SLERP; e (d) com refinamento SLERP+LUM. Abrevia-se (LU e MILIOS, 1998) apenas como LUM. Rotações são afetadas apenas pela etapa SLERP, então são analisadas em duas etapas: (a) sem refinamento; e (b) com refinamento SLERP.

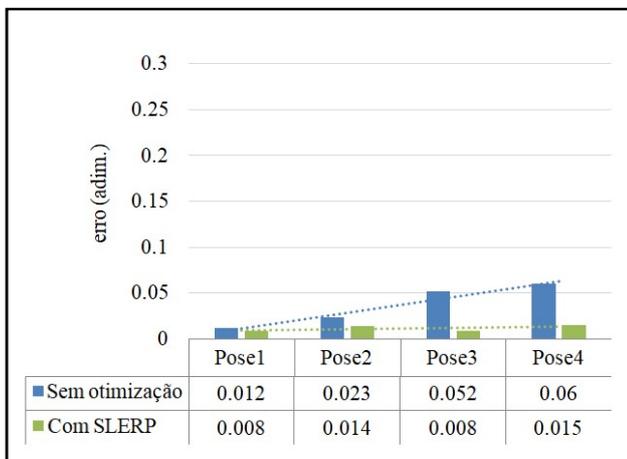
**Dataset Arch:** O registro dos pares do *dataset Arch* foi inteiramente manual. A baixa sobreposição entre as nuvens, a complexidade da cena urbana, com artefatos móveis e vegetação, e as grandes diferenças entre as orientações iniciais das nuvens, não permitiram o registro automático. Na Figura 5 está o gráfico dos erros de translação em cada pose após cada etapa de otimização. A etapa LUM aumentou o erro das poses, as etapas SLERP e SLERP+LUM diminuíram os erros. A SLERP, isoladamente, entregou a melhor qualidade. As retas ajustadas aos dados mostram o antes e o depois da etapa completa SLERP+LUM, sendo perceptível que o erro diminuiu em magnitude após a otimização completa. Na Figura 6 está o erro da rotação de cada pose após a etapa SLERP, se percebe claramente que o refinamento SLERP foi benéfico para todas as rotações.

A otimização SLERP mostra que o erro diminuiu em magnitude e também se torna mais uniforme ao longo do circuito, pois o erro de fechamento é distribuído ao longo das poses de forma crescente. Poses iniciais recebem menos correção e poses finais recebem mais correção. Nas Fig. 7 A) estão as análises qualitativas feitas visualmente antes do refinamento, em B) estão as análises após o refinamento com o modelo SLERP+LUM.



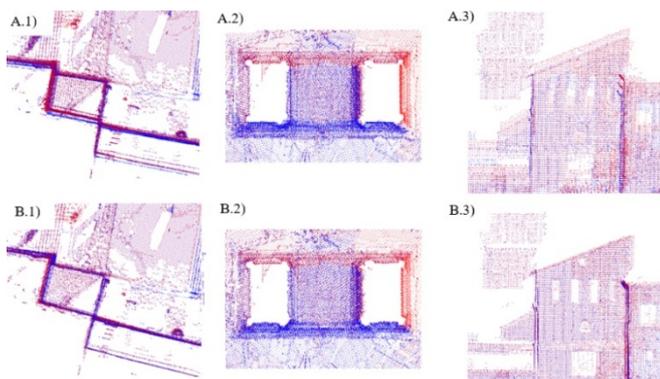
Fonte: autores (2022).

**Figura 5. Erro nas translações das poses do dataset Arch após cada etapa de otimização**



Fonte: autores (2022).

**Figura 6. Erro nas rotações das poses do dataset Arch após cada etapa de otimização**



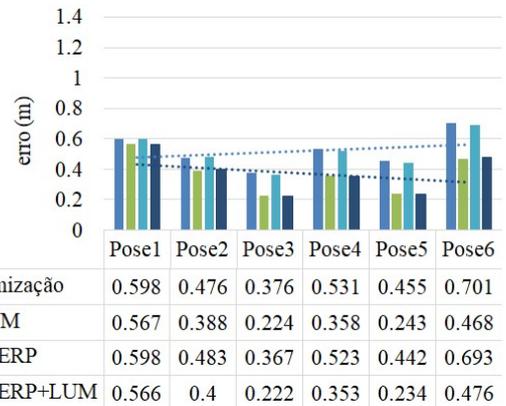
Fonte: autores (2022).

**Figura 7. Análise visual do dataset Arch antes (A) e depois (B) do refinamento das poses com modelo o SLERP+LUM**

Nas Fig. 7 A.1 e B.1 tem-se um recorte das nuvens que mostram paredes antes e após a correção SLERP+LUM. Percebe-se uma melhora geral no alinhamento das paredes. Na Nas Fig. 7 A.2 e B.2 tem-se a vista superior do arco escaneado, percebe-se uma melhora da organização dos cantos no interior do arco após o refinamento. Nas Fig. 7 A.3 e B.3 tem-se uma fachada com janelas, após a correção SLERP+LUM fica claro a melhor disposição das nuvens.

**Dataset Façade:** A reconstrução do dataset Façade se deu de forma automática pelo algoritmo FGR, neste dataset a sobreposição média entre nuvens é maior que em Arch. Há menos artefatos móveis,

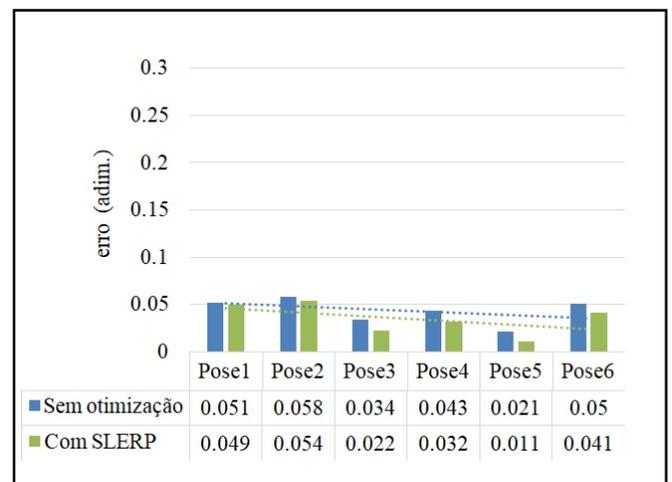
vegetações, e as orientações iniciais entre as nuvens são similares. Segue-se o mesmo modelo de apresentação dos resultados. Na Fig.8 está o gráfico dos erros de translação em cada pose após cada etapa de otimização. Todas as etapas entregaram qualidade superior às poses sem refinamento, mas desta vez a etapa LUM entregou qualidade superior a SLERP. De fato, as rotações não são a principal fonte de erro do registro neste dataset, as translações, por serem mais espaçadas que em Arch, foram as que mais se beneficiaram das correções. A etapa SLERP+LUM obteve desempenho praticamente igual ao modelo LUM isolado.



Fonte: autores (2022).

**Figura 8. Erro nas translações das poses do dataset Façade após cada etapa de otimização**

Nas rotações, Fig. 9, o modelo SLERP foi benéfico em todas as poses. As retas ajustadas mostram que o erro diminuiu em magnitude e se inverteu na sua direção de crescimento. Este comportamento era esperado, dado que o MRG distribui globalmente o erro defechamento em todas as anteriores.

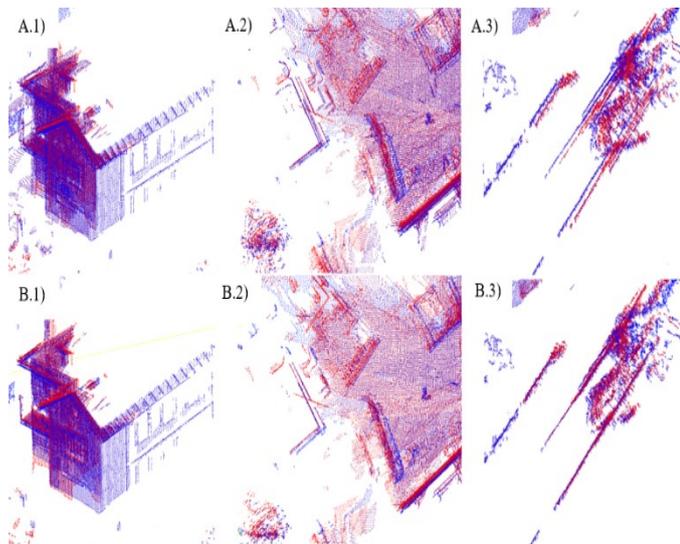


Fonte: autores (2022).

**Figura 9. Erro nas rotações das poses do dataset Façade após cada etapa de otimização**

Nas Fig. 10 A) estão as análises qualitativas feitas visualmente antes do refinamento, em B) estão as análises após o refinamento com o modelo SLERP+LUM. Nas Fig. 10 A.1 e B.1 tem-se um recorte de fachada, nela percebem-se melhoras sutis nos vértices dos telhados e na organização da janela.

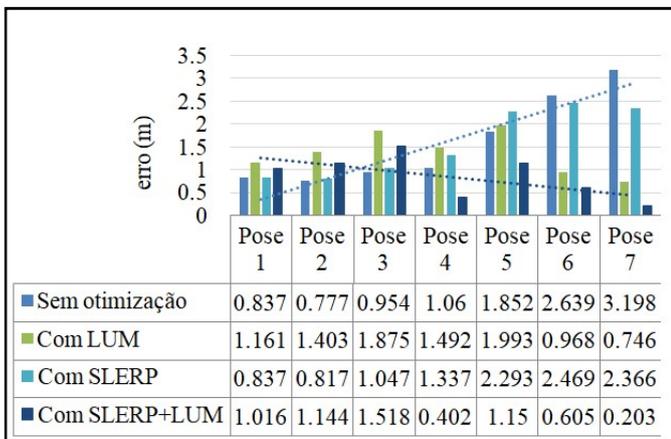
Nas Fig. 10 A.2 e B.2 tem-se a vista superior de paredes, nestas tem-se uma melhor sobreposição entre as paredes no canto inferior direito. Uma melhor sobreposição entre as paredes no canto inferior direito das Fig. 10 A.3 e B.3 também é notável.



Fonte: autores (2022).

**Figura 10. Análise visual do dataset Facade antes (A) e depois (B) do refinamento das poses com modelo o SLERP+LUM.**

**Dataset UFPR:** No dataset UFPR as estações do LST estão propositalmente muito afastadas. A reconstrução do dataset se deu de forma manual. Dificilmente algum algoritmo de registro funcionaria neste caso, pois a sobreposição média entre nuvens é ainda menor que em Arch. Há poucos artefatos móveis e vegetações. Segue-se o mesmo modelo de apresentação dos resultados anteriores. Na Fig. 11 está o gráfico dos erros de translação em cada pose após cada etapa de otimização.

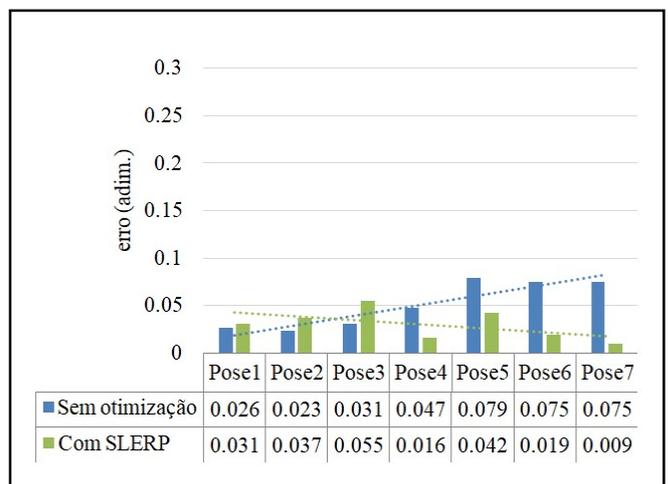


Fonte: autores (2022).

**Figura 11. Erro nas translações das poses do dataset UFPR após cada etapa de otimização**

Na Figura 11, todas as etapas do modelo entregaram qualidade superior às poses sem refinamento. Neste dataset em específico, o erro de registro entre as nuvens é propositalmente elevado. Isto permitiu observar o comportamento ideal de cada modelo, onde a combinação SLERP+LUM resultou no melhor resultado. O erro na Pose7 diminuiu praticamente 3 metros.

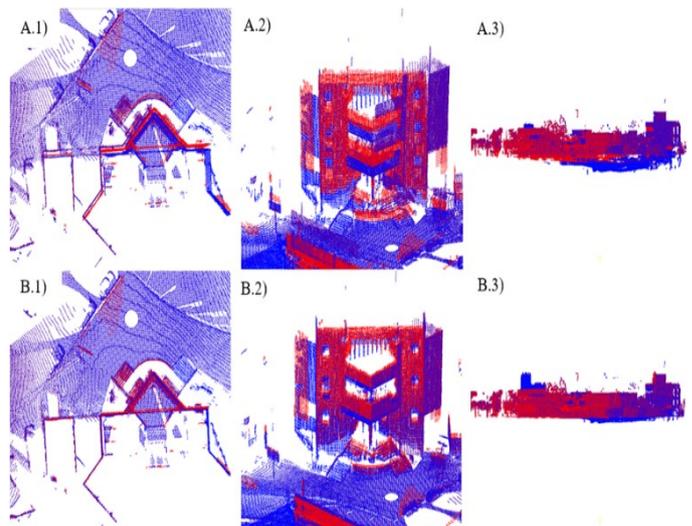
O modelo LUM aumenta o erro nas cinco primeiras poses, mas a redução que ele proporciona nas duas últimas é compensatória. Na última pose o modelo LUM decresce o erro de 3 metros para menos de 0,75 m. O modelo SLERP também aumenta o erro nas poses iniciais para diminuir nas finais. A seguir, o erro nas rotações. Na Fig. 12, o modelo SLERP novamente aumentou o erro nas poses iniciais para diminuir drasticamente nas finais, distribuindo o erro ao longo do circuito e diminuindo sua magnitude.



Fonte: autores (2022).

**Figura 12. Erro nas rotações das poses do dataset UFPR após cada etapa de otimização**

As retas ajustadas nas rotações antes e depois do modelo mostram que o erro se tornou decrescente ao longo do circuito. Se percebe que o modelo SLERP foi muito benéfico para as quatro últimas rotações e piorou um pouco as três primeiras. Nas Fig. 13 A) e B) estão as análises visuais de três recortes antes o refinamento com o modelo SLERP+LUM.



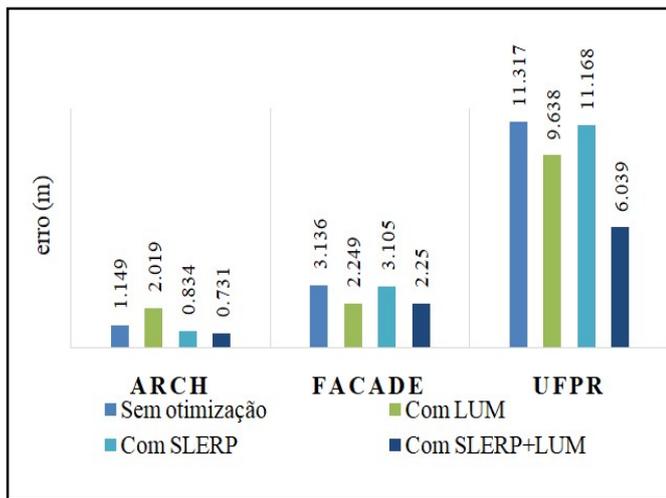
Fonte: autores (2022).

**Figura 13. Análise visual do dataset UFPR antes (A) e depois (B) do refinamento das poses com modelo o SLERP+LUM**

Nas Fig. 13 A.1 e B.1 tem-se o recorte da vista superior de um prédio. Após o refinamento, o alinhamento das paredes e da escadaria do prédio é evidentemente melhor, isto fica claro ao se observar as nuvens vermelha e azul. Nas Fig. 13 A.2 e B.2 tem-se o mesmo prédio em uma visualização frontal inclinada, onde se vê que a correção das nuvens não ocorreu apenas no plano XY, há também significativo deslocamento vertical entre elas, que diminuiu e resultou em uma melhor sobreposição após aplicação do modelo.

Por fim, nas Fig. 13 A.3 e B.3 tem-se a visualização lateral (plano XZ) de todo o dataset UFPR. Após o refinamento das poses ocorreu nivelamento global das nuvens, que se aproximaram mais do plano definido pelo solo e garantiram maior fidelidade geométrica do ambiente escaneado.

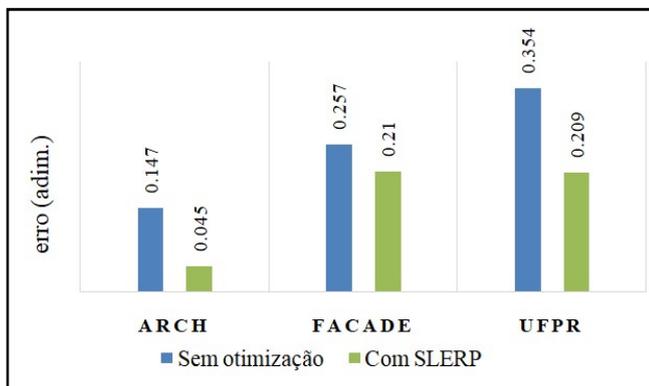
**Análise Geral dos resultados:** No gráfico da Fig. 14 apresenta-se a soma dos erros de todas as poses, por dataset, para a translação.



Fonte: autores (2022).

Figura 14. Redução do erro de translação por dataset

No dataset Arch o modelo LUM degradou a qualidade das poses, e o modelo SLERP+LUM foi o mais benéfico. No dataset Façade todos os modelos foram benéficos às poses. O modelo que mais atenuou os erros foi o LUM, com desempenho quase idêntico a ele está o modelo combinado SLERP+LUM. No dataset UFPR todos os modelos também foram benéficos, todavia a combinação SLERP+LUM maximizou o benefício de ambos. Na Fig. 15 está o resumo dos resultados nas rotações. A Fig. 15 mostra que todos os datasets se beneficiaram da etapa SLERP na diminuição dos erros. A análise total dos erros reforça o que foi visto de forma individual.



Fonte: autores (2022).

Figura 15. Redução do erro de rotação por dataset

## CONCLUSÕES

Este trabalho demonstrou os benefícios de um MRG de poses para nuvens de pontos 3D obtidas por LST que utiliza a técnica SLERP e se integra ao modelo de (LU; MILIOS, 1997) de forma totalmente linear. Cada rotação e translação, de cada pose, de cada circuito, em cada dataset, foi comparada com sua respectiva referência *groundtruth*. Esta comparação considerou a aplicação individual e conjunta dos modelos SLERP e LUM. Na configuração combinada SLERP+LUM é onde está a contribuição inédita deste trabalho. O modelo SLERP+LUM opera em uma única iteração sem a necessidade da definição de parâmetros e traz melhoras significativas no registro de todos os datasets testados. Muitas análises podem ser conduzidas futuramente com os modelos investigados neste trabalho. A mais direta delas consiste no estabelecimento de pesos para os modelos SLERP e LUM, onde em cada translação pode ser atribuído um valor de confiança proporcional à sobreposição entregue pelo registro local do par, ou ainda, inversamente proporcional ao RMSE entre as correspondências do par.

**Conflitos de interesses:** Os autores declaram não ter conflito de interesses.

**Agradecimentos:** O primeiro autor agradece à Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) pela concessão de Bolsa de Mestrado.

## REFERÊNCIAS

- AIGER, D.; MITRA, N. J.; COHEN-OR, D. 4-points congruent sets for robust pairwise surface registration. ACM SIGGRAPH 2008 papers. p.1–10, 2008.
- BESL, P. J.; MCKAY, N. D. Method for registration of 3-D shapes. Sensor fusion IV: control paradigms and data structures. Anais. v. 1611, p.586–606, 1992.
- BLACK, M. J.; RANGARAJAN, A. On the unification of line processes, outlier rejection, and robust statistics with applications in early vision. *International Journal of Computer Vision*, v. 19, n. 1, p. 57–91, 1996.
- BORRMANN, D.; ELSEBERG, J.; LINGEMANN, K.; NÜCHTER, A.; HERTZBERG, J. Globally consistent 3D mapping with scan matching. *Robotics and Autonomous Systems*, v. 56, n. 2, p. 130–142, 2008.
- CADENA, C.; CARLONE, L.; CARRILLO, H.; et al. Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age. *IEEE Transactions on robotics*, v. 32, n. 6, p. 1309–1332, 2016.
- CHEN, Y.; MEDIONI, G. G. Object modeling by registration of multiple range images. *Image Vision Comput.*, v. 10, n. 3, p. 145–155, 1992.
- CHOI, O.; PARK, M.-G.; HWANG, Y. Iterative K-closest point algorithms for colored point cloud registration. *Sensors (Switzerland)*, v. 20, n. 18, p. 1–24, 2020. Incheon National University, Incheon, 2022, South Korea.
- CHOI, S.; ZHOU, Q.-Y.; KOLTUN, V. Robust reconstruction of indoor scenes. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. Anais. p.5556–5565, 2015.
- DAM, E. B.; KOCH, M.; LILLHOLM, M. Quaternions, interpolation and animation. Citeseer, 1998.
- DIEZ, Y.; ROURE, F.; LLADÓ, X.; SALVI, J. A qualitative review on 3D coarse registration methods. *ACM Computing Surveys (CSUR)*, v. 47, n. 3, p. 1–36, 2015. ACM, NY, USA.
- DONG, Z.; YANG, B.; LIANG, F.; HUANG, R.; SCHERER, S. Hierarchical registration of unordered TLS point clouds based on binary shape context descriptor. *ISPRS Journal of Photogrammetry and Remote Sensing*, v. 144, p. 61–79, 2018.
- GRISSETTI, G.; KUMMERLE, R.; STACHNISS, C.; BURGARD, W. A tutorial on graph-based SLAM. *IEEE Intelligent Transportation Systems Magazine*, v. 2, n. 4, p. 31–43, 2010.
- HUANG, H.; SUN, Y.; WU, J.; et al. On Bundle Adjustment for Multiview Point Cloud Registration. *IEEE Robotics and Automation Letters*, v. 6, n. 4, p. 8269–8276, 2021. Robotics Institute, The Hong Kong University of Science and Technology, Hong Kong.
- HUANG, J.; TAO, B.; ZENG, F. Point cloud registration algorithm based on ICP algorithm and 3D-NDT algorithm. *International Journal of Wireless and Mobile Computing*, v. 22, n. 2, p. 125–130, 2022. Wuhan University of Science and Technology, Hubei, Wuhan, China.
- HUBER, D. F.; HEBERT, M. Fully automatic registration of multiple 3D data sets. *Image and Vision Computing*, v. 21, n. 7, p. 637–650, 2003.
- HUYNH, D. Q. Metrics for 3D rotations: Comparison and analysis. *Journal of Mathematical Imaging and Vision*, v. 35, n. 2, p. 155–164, 2009.
- KÜMMERLE, R.; GRISSETTI, G.; STRASDAT, H.; KONOLIGE, K.; BURGARD, W. g2o: A general framework for graph optimization. 2011 IEEE International Conference on Robotics and Automation. Anais. p.3607–3613, 2011.

- LU, F.; MILIOS, E. Globally consistent range scan alignment for environment mapping. *Autonomous Robots*, v. 4, n. 4, p. 333–349, 1997.
- NEUGEBAUER, P. J. Reconstruction of real-world objects via simultaneous registration and robust combination of multiple range images. *International Journal of Shape Modeling*, v. 3, n. 01n02, p. 71–90, 1997.
- PAVAN, N. L.; DOS SANTOS, D. R. A global closed-form refinement for consistent TLS data registration. *IEEE Geoscience and Remote Sensing Letters*, v. 14, n. 7, p. 1131–1135, 2017.
- RUSINKIEWICZ, S.; LEVOY, M. Efficient variants of the ICP algorithm. *Proceedings Third International Conference on 3-D digital imaging and modeling*. Anais. p.145–152, 2001. IEEE.
- RUSU, R. B.; BLODOW, N.; BEETZ, M. Fast point feature histograms (FPFH) for 3D registration. 2009 IEEE international conference on robotics and automation. Anais. p.3212–3217, 2009.
- SHOEMAKE, K. Animating rotation with quaternion curves. *Proceedings of the 12th annual conference on Computer graphics and interactive techniques*. Anais. p.245–254, 1985.
- THEILER, P. W.; WEGNER, J. D.; SCHINDLER, K. Globally consistent registration of terrestrial laser scans via graph optimization. *ISPRS Journal of Photogrammetry and Remote Sensing*, v. 109, p. 126–138, 2015.
- VLAMINCK, M.; LUONG, H.; PHILIPS, W. Have I seen this place before? A fast and robust loop detection and correction method for 3D lidar SLAM. *Sensors*, v. 19, n. 1, p. 23, 2019.
- WEBER, T.; HÄNSCH, R.; HELLWICH, O. Automatic registration of unordered point clouds acquired by Kinect sensors using an overlap heuristic. *ISPRS Journal of Photogrammetry and Remote Sensing*, v. 102, p. 96–109, 2015.
- WILLIAMS, J.; BENNAMOUN, M. Multiple view 3D registration using statistical error models. *Vision modeling and visualization*. Anais. v. 99, 1999.
- WILLIAMS, J.; BENNAMOUN, M. Simultaneous registration of multiple corresponding point sets. *Computer Vision and Image Understanding*, v. 81, n. 1, p. 117–142, 2001.
- ZHOU, Q.-Y.; PARK, J.; KOLTUN, V. Fast global registration. *European conference on computer vision*. Anais. p.766–782, 2016.
- ZHOU, Q.-Y.; PARK, J.; KOLTUN, V. Open3D: A modern library for 3D data processing. *arXiv preprint arXiv:1801.09847*, 2018.

\*\*\*\*\*