Nonrigid Image Matching Benchmark

Tarcizio Augusto Santos Lafaiete Erickson Rangel do Nascimento
Universidade Federal de Minas Gerais
tarcizio-augusto@ufmg.br,erickson@dcc.ufmg.br

Resumo

Nas últimas décadas, a área de Visão Computacional tem se utilizado do aprendizado de máquina como forma de avançar no estado da arte de diversos problemas. No entanto, os estudos em casos de deformações não rígidas ainda carecem de datasets robustos para seu avanço. Este trabalho apresenta o Nonrigid Image Matching Benchmark, um conjunto de dados sintéticos projetado para superar essas limitações, utilizando um pipeline de simulação fotorealista e fisicamente plausível para gerar deformações temporalmente consistentes. O benchmark inclui uma diversidade de objetos, texturas, iluminações e níveis de deformação. Adicionalmente, são abordados conceitos fundamentais para o entendimento dos problemas relacionados ao tópico de deformação de objetos não rígidos e sobre a correspondência de imagens que é a tarefa foco do dataset. Além da criação do conjuto, foi introduzida uma métrica de registro de superfície baseada no ARAP [27] para avaliar métodos de correspondência de maneira geométrica e contínua em 2D e 3D, superando limitações de outros métricas tradicionais. Foi realizada uma avaliação de uma miríade de métodos dentro do conjuto de testes do dataset que constatou a degradação significativa de desempenho em ambientes desafiadores, ressaltando assim, a relevância do conjunto. Este trabalho fornece um recurso valioso e um framework de avaliação para pesquisas futuras na temática.

***Palavras-chave**: Visão Computacional, Deformações Não Rígidas, Correspondência Visual, Benchmark, Métricas de Avaliação.*

Abstract

In recent decades, the field of Computer Vision has increasingly relied on machine learning to advance the state of the art in various problems. However, research involving non-rigid deformations still lacks robust datasets to drive meaningful progress. This work introduces the Nonrigid Image Matching Benchmark, a synthetic dataset designed to overcome these limitations by using a photorealistic and physically plausible simulation pipeline to generate temporally consistent deformations. The benchmark includes a wide variety of objects, textures, lighting conditions, and deformation levels. Additionally, fundamental concepts related to non-rigid object deformation and image correspondence—the core task targeted by the dataset—are presented. Beyond dataset creation, an ARAP-based surface registration metric [27] was introduced to evaluate correspondence methods geometrically and continuously in both 2D and 3D, overcoming limitations of other traditional metrics. An evaluation of a wide range of methods was conducted using the dataset's test set, revealing significant performance degradation in challenging environments, thus underscoring the relevance of the benchmark. This work provides a valuable resource and an evaluation framework for future research in this area.

***Keywords**: Computer Vision, Non-Rigid Deformations, Visual Correspondence, Benchmark, Evaluation Metrics.*

1. Introdução

Nas últimas décadas, a área de Visão Computacional tem passado por evoluções significativas em tarefas como reconhecimento, segmentação e reconstrução. Esses avanços têm sido impulsionados por diversos fatores, entre eles o aumento da capacidade computacional, o desenvolvimento de métodos baseados em arquiteturas de rede neural complexas e, principalmente, a ampla disponibilidade de conjuntos de dados visuais catalogados. Este último fator tem-se mostrado cada vez mais central, visto que modelos modernos, especialmente redes neurais profundas, dependem fortemente de grandes volumes de dados anotados para alcançar alto desempenho. Entretanto, certas aplicações ainda carecem de datasets representativos e bem estruturados, como é o caso de tarefas que envolvem deformações não rígidas em objetos.

Em geral, a captura e anotação de deformações complexas têm limitado o avanço de métodos especializados e evidenciado a necessidade contínua de novos conjuntos que permitam tanto o treinamento quanto a avaliação rigorosa de modelos. Este problema se encontra principalmente na dificuldade de repetibilidade e controle fino das deformações físicas no mundo real, assim como na falta de

escalabilidade dos conjuntos de dados, tanto em número de objetos quanto na variedade de deformações. Um exemplo desses problemas é o dataset DeepDeform [3], que é um dos maiores datasets de dados reais com correspondências anotadas manualmente. Ele possui cerca de 20 anotações manuais para dez quadros em cada uma das 400 sequências e teve as dificuldades de expansão dos dados relatadas pelos autores no artigo.

Neste sentido, buscando o desenvolvimento de um conjunto de dados mais abrangente e que pudesse superar as problemáticas apresentadas acima, o Nonrigid Image Matching Benchmark propõe um pipeline de simulação fotorealista e fisicamente plausível. Ele foi cuidadosamente projetado para capturar deformações não rígidas temporalmente consistentes e com controle preciso.

A proposta de dataset apresenta avanços relevantes e significativos para a área de correspondência visual não rígida, que é uma tarefa pouco explorada. Contudo, este trabalho ainda possui algumas lacunas como: a não utilização de métodos recentes ou denso/semi-densos como RoMa [32] em suas avaliações experimentais, o uso de métricas que não normalizam as variações no número de keypoints entre os métodos, problemas de realismo das simulações com deformações exageradas e irreais, entre outras pontuações.

Esta monografia em Sistemas de Informação visa então trabalhar em cima das lacunas apresentadas, com o objetivo geral de expandir a robustez do trabalho, provar através de experimentos sua relevância para a academia e gerar uma publicação relevante dentro da área especificada de correspondência de objetos deformados, contribuindo assim de forma científica para os avanços do campo. Além disso, traz como objetivos específicos, a investigação e o desenvolvimento de métricas de avaliação deste tipo de tarefa, o aprofundamento e estudo de métodos no estado da arte dentro deste campo e o contato com problemas relacionados com transformações tridimensionais e a avaliação do *dataset* com diversos *baselines* para validação.

2. Referencial teórico

2.1. Correspondência Visual em Visão Computacional.

A Correspondência Visual é uma tarefa que consiste em identificar pontos ou regiões semelhantes entre duas ou mais imagens, possibilitando a associação entre diferentes visões de uma mesma cena ou objeto. Este conceito é fundamental para uma ampla gama de aplicações, como reconstrução tridimensional e localização e mapeamento simultâneo (SLAM).

Nesta tarefa, os algoritmos utilizados baseiam seu funcionamento em três etapas principais:

1. **Detecção de Pontos de Interesse**:
   - Algoritmos identificam *keypoints* em regiões distintivas da imagem (como cantos, bordas ou texturas), criando uma representação esparsa da cena.
2. **Extração de Descritores**:
   - Através dos pontos de interesse são criadas representações numéricas densas a partir da extração de características presentes na vizinhança do *keypoint*.
3. **Casamento de pares**:
   - A partir dos descritores, pode-se realizar uma busca entre os espaços de cada uma das imagens, através de cálculos de similaridade, a fim de encontrar pontos iguais ou semelhantes entre elas, assim formando pares de correspondência.

2.1.1. Métodos Clássicos

Entre os métodos clássicos mais conhecidos, ou seja, aqueles que não utilizam de aprendizado de maquina em seu funcionamento, destacam-se o SIFT [15], o ORB [26], o FREAK [1] e o Daisy [29]. Desses, o SIFT[15] e o ORB [26], funcionam tanto como detectores, quanto como descritores. Por outro lado, o FREAK [1] e o Daisy [29], cumprem apenas a função de descritores; neste trabalho, portanto, eles foram combinados aos detectores do FAST [25] e do SIFT [15], respectivamente. No processo de casamento de pares (*matching*), para os descritores contínuos SIFT [15] e Daisy [29] foi utilizada a técnica de busca pelo vizinho mais próximo, com base no calculo de similaridade dos cossenos entre os vetores das duas imagens. Já para os descritores binários, FREAK [1] e ORB [26], empregou-se o casamento por força bruta, no qual cada ponto da primeira imagem é comparado com o da segunda por meio do calculo de distância de Hamming.

2.1.2. Métodos de aprendizado de máquina

Com o advento do aprendizado de máquina no campo da visão computacional, estes métodos que estabeleceram as bases para a correspondência visual começaram a ser superados devido a suas limitações em cenários com variações extremas e com pouca textura. Assim, diversos métodos se utilizando de aprendizado profundo, transformers e redes convolucionais (CNN) vêm sendo o estado da arte desta tarefa.

Neste contexto, se destacam como descritores e detectores, que se utilizam de CNN convencional como o núcleo de seu processamento o SuperPoint [4], o xFeat [21], o DEDode [7] e o DISK [30]. Também se utilizando de uma arquitetura semelhante, mas apenas como descritores, tem-se o TFeat [2] e o SOSNet [28]. Para estes dois últimos foi-se utilizado, neste trabalho, a detecção de *keypoints* por meio do detector do SIFT [15].

Para além das CNNs convencionais, há também os métodos que adicionam outros procedimentos no núcleo de seu funcionamento, como por exemplo, o R2D2 [23] que produz um mapa de repetibilidade e outro de confiabilidade, além de operar em uma arquitetura FPN/U-Net [24] que

permite a operação em múltiplas escalas. Outro exemplo deste tipo de algoritmo é o Alike [31], que semelhante ao anterior, também utiliza a mesma técnica de operação em múltiplas escalas.

Dois outros métodos que se destacam por suas características únicas, são o DALF [20] e o DELF [18]. O primeiro, utiliza uma ResNet [11] em seu backbone e possui uma sub-rede capaz de aprender transformações afins completas. O segundo, também utiliza uma ResNet [11], contudo, seu diferencial está na utilização de um mecanismo de atenção para selecionar as regiões importantes da imagem.

Entre os métodos que utilizam transformer, vale a citação do LightGlue [14]. Ele segue a formula padrão desta arquitetura utilizando-se de *Cross-attention* iterativa entre os descritores. Diferentemente dos métodos citados acima ele é apenas um *matcher*, ou seja, sua função é apenas de realizar o casamento entrer os pares de imagens distintas. Assim sendo, no seu uso neste trabalho, combinou-o com a detecção e descrição do SuperPoint [4]. Todos os demais métodos supracitados nesta subseção se valem do procedimento de casamento pelo vizinho mais próximo ja descrito.

2.2. Deformações não rígidas.

As deformações não rígidas representam um grande desafio em tarefas de Visão Computacional, pois envolvem mudanças na geometria dos objetos que não podem ser descritas ou replicadas utilizando as transformações simples, como rotação, translação e escala. Não obstante, essas deformações são comuns em cenários reais em superfícies flexíveis como roupas, sapatos e outros objetos maleáveis. A principal dificuldade está na modelagem da correspondência entre imagens de um mesmo objeto em estados diferentes de deformação devido as movimentações não previsíveis dos pontos e oclusões parciais. Exemplos relevantes de métodos voltados para ambientes deformáveis são: o GeoBit [17] que introduz a ideia de um descritor binário baseado em geodésia para lidar com deformações em imagens RGB-D; o DEAL [19] que é um descritor sensível à deformação, ele utiliza-se de uma amostragem polar e um transformador de distorção espacial para fornecer invariância a rotação, escala e deformações e o DALF [20] uma rede que trabalha cooperativamente por meio de uma abordagem de características que reforça a distintividade e a invariância dos descritores.

2.3. Datasets e dados simulados

Um dos maiores datasets de objetos deformados é o DeepDeform [3], ele é formado por anotações de 20 objetos para 10 frames em cada uma das 400 sequências. Suas anotações foram feitas de maneira manual e com anotações esparsas, o que torna este propenso a erros em suas anotações. Outro dataset de dados reais é o Kinect 1/2 [17] que resolve o problema da esparsidade do DeepDeform [3] com anotações manuais densas, utilizando-se do algoritmo TPS [5], contudo este conjunto de dados anotados é pequeno.

Um aspecto comum entre as anotações por meio de dados reais é a dificuldade de escalabilidade dos datasets, com os autores do DeepDeform [3], por exemplo, relatando as possíveis dificuldades de expansão deste conjunto. Para isso a simulação de dados sintéticos surgiu como uma ferramenta valiosa para superar estes problemas. Entre esses conjuntos de dados, destaca-se o Drunkard's [22], que utiliza uma modelagem aproximada por Thin Plate Spline (TPS) para simular deformações e realizar o registro não rígido de nuvens de pontos. Esse conjunto inclui deformações não isométricas, ou seja, que não preservam as distâncias ou formas locais dos objetos, além de apresentar também deformações globais. Já o DeformThings4D [13] também é um dataset simulado porém se utiliza de modelos animados, o que garante precisão semântica e temporal. Não obstante, há uma dificuldade de expansão do dataset devido ao alto custo para a geração de novos modelos animados.

Neste sentido, o *Nonrigid Image Matching Benchmark* apresenta uma abordagem promissora com a utilização de malhas 3D escaneadas e o algoritmo ARAP (As-Rigid-As-Possible) [27] para gerar deformações temporalmente consistentes e fisicamente adequadas, além disso conta-se com um controle preciso de iluminação, viewpoint e escala. Essa abordagem permite a geração de pares de imagens com correspondências densas e anotações multimodais (profundidade, segmentação).

2.4. As-rigid-as-possible

O algoritmo "As-Rigid-As-Possible"(ARAP) [27] foi desenvolvido com o objetivo de realizar deformações em malhas tridimensionais de maneira a preservar a rigidez local ao máximo. Para isto o algoritmo alterna entre dois passos até convergir, sendo o Passo 1 uma estimação da rotação ótima R_i para cada vértice i de forma a melhor alinhar os vizinhos antes/depois da deformação e o Passo 2 com a atualização das posições de forma a ajustar as posições dos vértices para minimizar a fórmula de energia abaixo:

$$E(\mathbf{V}') = \sum_{i=1}^{n} \sum_{j \in \mathcal{N}(i)} w_{ij} \left\| (\mathbf{v}'_i - \mathbf{v}'_j) - \mathbf{R}_i(\mathbf{v}_i - \mathbf{v}_j) \right\|^2 \tag{1}$$

Nesta fórmula v'_i e v'_j são os vértices antes/depois da deformação e w_{ij} são os pesos que representam a força da conexão entre i e j durante a deformação. Geralmente, são calculados utilizando-se das cotangentes dos ângulos opostos à aresta (i, j).

As vantagens deste algoritmo são os resultados naturais para deformações de superfícies e seu baixo custo computacional. Contudo, ele apresenta limitações por requerer uma solução iterativa de sistemas lineares e não funcionar ade-

quadamente com topologias variáveis.

2.5. Métricas de Avaliação.

As métricas *Matching Score* (MS) [16] e a *Mean Matching Accuracy* (MMA) [19] são amplamente utilizadas para quantificar a assertividade das correspondências dentro de um limiar de pixels. Ambas as métricas levam em consideração a quantidade de *matches* corretos encontrados, contudo o *Matching Score* [16] utiliza como normalizador o número mínimo de *keypoints* encontrados em uma das imagens. Assim sendo, ele pode ser entendido como uma medida de cobertura, verificando a eficiência dos métodos em aproveitar os pontos detectados. Devido a sua formula, é evidente notar que ele penaliza detectores gulosos que acham muitos pontos, mas utilizam poucos em seus *matchings*.

Complementar a isto, o *Mean Matching Accuracy*, utiliza como normalizador o número total de matches encontrados. Desta forma, ele age como uma medida de precisão, verificando a assertividade dos métodos. Esta métrica então, beneficia algoritmos que encontram poucos pares, mas que são extremamente precisos, o que pode acabar mascarando o baixo nível de descoberta de pares de determinados *baselines*.

Outra métrica relevante é a *Keypoints Mean Repeatability Rate* (MMR) [8] que mensura a capacidade de detectores identificarem pontos estáveis entre imagens. Para isso, pega-se os pontos de uma imagem e os projeta para a outra com homografia. Após isso, verifica-se quantos pontos projetados ficaram a uma distância menor dos *keypoints* desta imagem em relação ao número total de pontos corretamente projetados.

3. Nonrigid Imagem Matching Benchmark

A proposta do *benchmark* é criar uma base de dados para treinamento e avaliação de métodos de correspondência com objetos deformados e garantir dentro deste dataset uma diversidade de deformações não rígidas temporalmente consistentes, objetos com texturas e formas variadas e iluminação realista.

3.1. Obtenção dos dados base

Para cumprir esta tarefa, o Nonrigid Image Matching Benchmark é formado de dados sintéticos, diferentemente do DeepDeform [3] e Kinect 1/2 [17], devido às dificuldades de anotação e expansão do *dataset* com a utilização de dados reais. Para a obtenção de uma grande variedade de objetos foi adotado o Scanned Objects Dataset [6] da Google que contém 1030 modelos escaneados que possuem em seus metadados as anotações para segmentação (categoria, marca, descrição) e para a variedade de cenários realísticos e com iluminação global foi utilizado o HDRI 360º do PolyHeaven [10].

3.2. Aplicação da deformação

Diferentemente do DeformThings4D [13], em que as deformações são geradas através da animação dos objetos, o que é caro para a expansão do conjunto. A abordagem adotada neste trabalho é de criar deformações fisicamente plausíveis, ou seja, que preservam as propriedades dos materiais e com foco em deformações isométricas, que preservam distâncias locais.

Para isto segue-se um *pipeline* de deformações para cada cena/objeto do conjunto base de dados. Primeiramente, os objetos, ou malhas, são representados por vértices $\mathbf{V}$ e arestas $\mathbf{E}$. Em seguida, seleciona-se aleatoriamente $\mathbf{N}$ vértices na malha original como pontos de controle, a cada iteração então um subconjunto $\mathbf{B}$ é movido em uma direção aleatória com intensidade α multiplicado por $\mathrm{S_{box}}$ (o tamanho da bounding box do objeto). A cada lote de vértices movido, o ARAP é executado para deformar o resto da malha, minimizando assim o erro de rigidez local seguindo a fórmula abaixo em que $\mathbf{R}$ é a matriz de rotação e $\mathbf{t}$ é o vetor de translação.

$$argmin_{\mathbf{R},\mathbf{t}} f(\mathbf{R},\mathbf{t},V(\cdot)) = \frac{1}{N}\sum_{i=1}^{N} \|\mathbf{V}_{og,i} - (\mathbf{R}^T\mathbf{V}_{def,i} + \mathbf{t})\|^2 \tag{2}$$

Estas deformações são aplicadas em 4 níveis temporais, sendo $\mathbf{T} = 0$ o estado original e $\mathbf{T} = 3$ o estado de deformação máxima do objeto.

3.3. Composição dos dados

Os dados estão divididos em dois grandes grupos sendo eles o conjunto de teste e um conjunto de treino.

3.3.1. Conjunto de testes

Este *benchmark* tem 6000 pares para a realização do *evaluation*, estes pares são divididos em três subcategorias sendo elas:

- **Single Object**: Composto por imagens com apenas um objeto disposto na cena. Este objeto é colocado em 4 cenários diferentes e passa por transformações de *viewpoint* e iluminação e para cada uma destas tem-se os 4 níveis de deformação. Ele possui 2800 pares.
- **Multi-Object**: Composto por uma grade 3x3x2 de objetos na cena, tendo o mesmo procedimento de variação das transformações o subconjuto *Single Object* e também possui 2800 pares.
- **Scale**: Composto por apenas um objeto em cada cena sem transformações de iluminação e *viewpoint*, mas com 5 cenários para cada objeto com uma redução gradual de 20% no tamanho destes.

3.3.2. Conjunto de treinamento

No conjunto de treino, os dados estão em dois *subsets* com objetivos complementares. O primeiro conjunto é com-

posto por 800 objetos isolados, cada um renderizado em 20 diferentes configurações de ponto de vista e iluminação. Para cada configuração, são aplicados os quatro níveis de deformação temporalmente consistentes. Com isso, cada objeto no dataset possui 2400 pares anotados com diferentes deformações, totalizando 1.920.000 pares de imagens. Enquanto isso, o segundo *dataset* é composto por 83 cenas com 20 objetos cada e cada uma destas com 10 configurações e 2 níveis de deformação. Com isto este conjunto possui 15.770 pares anotados.

3.3.3. Anotações e Metadados

Este projeto inclui como metadados e anotações: um arquivo .json contendo os pares de imagens anotados e para cada sequência e cenário, são disponibilizados: a máscara binária (com valores 0 ou 255), o mapa de profundidade, a segmentação dos objetos, um arquivo .json com as configurações do cenário e o respectivo UV Map.

4. Trabalho Desenvolvido

Neste *benchmark* para a avaliação do desempenho dos modelos já estava sendo utilizado as métricas: Matching Score (MS) [16], Mean Matching Accuracy (MMA) [19] e Keypoints Mean Repeatability (MMR) [8]. Contudo, a distribuição e o número de *keypoints* possuem variações consideráveis entre os diversos métodos, o que torna estas métricas de correspondência não diretamente comparáveis. Devido a isto, é importante a adoção de outras métricas para a avaliação dos modelos.

4.1. ARAP-based surface registration

A nova métrica implementada foi o ARAP-based surface registration que foi apresentada no artigo DALF [20], em que os autores elaboraram esta métrica com o intuito de avaliar seu modelo proposto em relação a outros métodos em uma tarefa desafiadora de registros não rígidos de superfícies. Esta métrica busca combinar uma filtragem robusta de *outliers* e a utilização do algoritmo ARAP [27] para encontrar o alinhamento ótimo entre as malhas formadas pelos dois conjuntos de *keypoints*, com isso conseguimos medir a distância entre os pontos alinhados tanto no 3D quanto no 2D através de um *ground truth*. Para isto, define-se *thresholds* para contar quantos vértices/pontos estão corretamente alinhados.

Esta métrica consegue apresentar uma solução para o problema de comparabilidade entre os métodos avaliados no projeto, uma vez que ela possui uma independência do número de *keypoints* ao utilizar-se do alinhamento da superfície 3D realizando assim uma avaliação contínua e geométrica.

4.2. Pipeline

O pipeline da métrica proposta recebe como entrada duas nuvens de pontos, comumente chamadas de referência e alvo, e consiste em seis etapas principais:

- **Remoção de outliers**: Este processo é feito através de uma função conhecida como nr_RANSAC em que utiliza-se o *PyTorch* para calcular um conjunto de escolhas aleatórias e uma hipótese, de forma a encontrar a hipótese que maximiza o número de *inliers* e considera esta como a melhor estimativa.
- **Pré-processamento da nuvem de pontos**: Em seguida, mais dois pré-processamento são feitos nos pontos, primeiramente reduzir-se a resolução das imagens com uma abordagem piramidal com filtros gaussianos. Em seguida, aplica-se um filtro para remoção de ruídos e blobs.
- **Geração das Malhas 3D**: Com os dados processados, projeta-se os pontos do 2D para o 3D utilizando-se da matriz k com os parâmetros intrínsecos da câmera. Com esta projeção é construído as malhas triangulares de referência e alvo e mapeia-se os *keypoints* 2D para índices na malha 3D.
- **Aplicação do ARAP**: Com as malhas triangulares e seus índices mapeados, aplica-se o algoritmo ARAP [27] com o objetivo de gerar um alinhamento da malha alvo na direção da malha de referência.
- **Acurácia em 3D**: Neste ponto, pega-se os pontos presentes na malha de referência e os da malha alinhada resultante e realiza o casamento entre eles de forma a sempre selecionar os pontos com as menores distâncias euclidianas entre si. Com isso, calcula-se a proporção de pares de pontos que obtiveram uma distância menor que o threshold estabelecido.
- **Acurácia em 2D**: Por fim, a malha resultante e a malha alvo são projetadas em coordenadas 2D. A partir disso, utilizam-se os *UV Maps* da imagem de referência e da imagem alvo para estabelecer a correspondência espacial entre os pontos da malha alvo e os da malha de referência, gerando assim o *ground truth* necessário para a avaliação em 2D. Em seguida, calcula-se a distância euclidiana entre esses pontos correspondentes e os pontos da projeção da malha resultante, a fim de determinar a proporção de pares cuja distância é inferior a um determinado limiar (threshold).

4.3. Experimentos

Com a métrica de ARAP-based surface registration implementada e integrada ao pipeline de avaliação, foi possível realizar uma nova série de experimentos visando comparar o desempenho de diversos métodos de correspondência visual. O objetivo desta análise é avaliar a qualidade desses *baselines* na tarefa de correspondência com objetos deformados e entender a relevância deste conjunto de dados para a área.

Neste trabalho, os experimentos executados, foram realizados apenas com o conjuto de dados de teste *Single Object*, descrito na secção 3 do artigo. Para os baselines, foram utilizados os já citados: SIFT [15], ORB [26], FREAK [1], Daisy [29], SuperPoint [4], xFeat [21], DEDode [7], DISK [30], TFeat [2], SOSNet [28], R2D2 [23], Alike [31], DALF [20], DELF [18] e LightGlue [14]. Todos estes, com os complementos de descritores, detectores e *matchers* devidamente descritos durante a secção 2 do trabalho. Além disso, todos foram testados sem nenhum tipo de *fine-tuning* nos dados de treinamento, justamante para medir a dificuldade da tarefa proposta.

Com estas definições, os experimentos foram executados para todos os *baselines* em todos os 2800 pares de imagens do conjunto *Single Object*. Este conjunto, por sua vez, é subdivido em diferentes configurações que avaliam os métodos em condições diversas sendo elas: três níveis de deformação, variações de ponto de vista e mudanças de iluminação, além de ambientes com dois ou mais destes fatores combinados. Para cada configuração diferente e para cada baseline, os resultados foram armazenados em arquivos no formato JSON, com a estrutura apresentada na Figura 1.

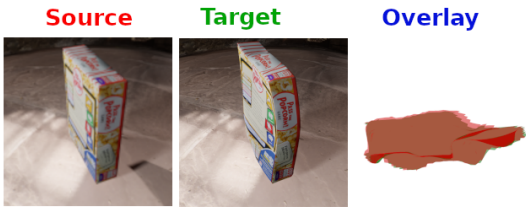

```
{
    "Matching Score": 0.28736397551830467,
    "Matching Accuracy": 0.48888883776384523,
    "Repeatability": 0.5085440863642502,
    "ARAP Registration 2D Accuracy": [
        0.2572368533035332,
        0.42416206508084386,
        0.610506145308964
    ],
    "ARAP Registration 3D Accuracy": [
        0.1550174610604281,
        0.35773489648570944,
        0.6070376932561975
    ],
    "ARAP Success": 383,
    "Skip Proportion": 0.9575
}
```


Figura 1. Exemplo da estrutura JSON contendo os resultados do método xFeat para uma deformação de nível médio.

Percebe-se que diferente das métricas *Matching Score* (MS) [16], *Mean Matching Accuracy* (MMA) [19] e *Keypoints Mean Repeatability Rate* (MMR) [8], para o ARAP-based, tanto no 3D, quanto no 2D, foram utilizados três níveis de tolerância diferentes, sendo 0.5, 1.0 e 1.5 centimétros e 2, 3 e 5 pixels, respectivamente. Além disso, pode-se notar a presença de dois outros dados no arquivo, sendo eles o *ARAP Success* e *ARAP proportion*. Eles servem como um controle para os experimentos, contabilizando a quantidade de vezes que a métrica, podê ser executada, visto que, em alguns casos em que se tem menos de 8 pontos de *matching* é impossível calcular a métrica, sendo assim ela retorna uma acurácia de 0 para esta instância, o que acaba afetando o desempenho médio dos métodos avaliados.

4.4. Resultados qualitativos

Como dito anteriormente, uma das vantagens da métrica baseada no *ARAP* é a possibilidade de realizar uma análise contínua das malhas dos objetos e seus equivalantes deformados. Afim de explicitar este processo de sobreposição das malhas alvo e referência. Na Figura 2, pode-se visualizar este resultado para dois níveis de deformação diferente para o *baseline* DALF.



(a) Sobreposição para o nível de deformação fácil

(b) Sobreposição para o nível de deformação difícil

Figura 2. Resultados da métrica ARAP-based surface registration para uma instância de teste utilizando o método DALF

4.5. Resultados quantitativos

Esta secção apresenta e discute os desempenhos observados durante as avaliações de métodos. As análises foram organizadas em diferentes perspectivas, da seguinte forma: variação dos nívies de deformação, variação da tolerância de erro da métrica e adição de *viewpoint* e variação de iluminação nos dados.

4.5.1. Níveis de deformações

Neste primeiro experimento, foram utilizados apenas os pares de objetos que sofriam transformações de deformação, assim excluindo as variações de ponto de vista e iluminação. Isto foi feito, com o objetivo de entender o impacto dos níveis de deformação nos algoritmos. Na Figura 3, se apresenta os resultados para cada métodos nos três níveis de deformação temporal e com a tolerância da métrica fixada no padrão mais restritivo.

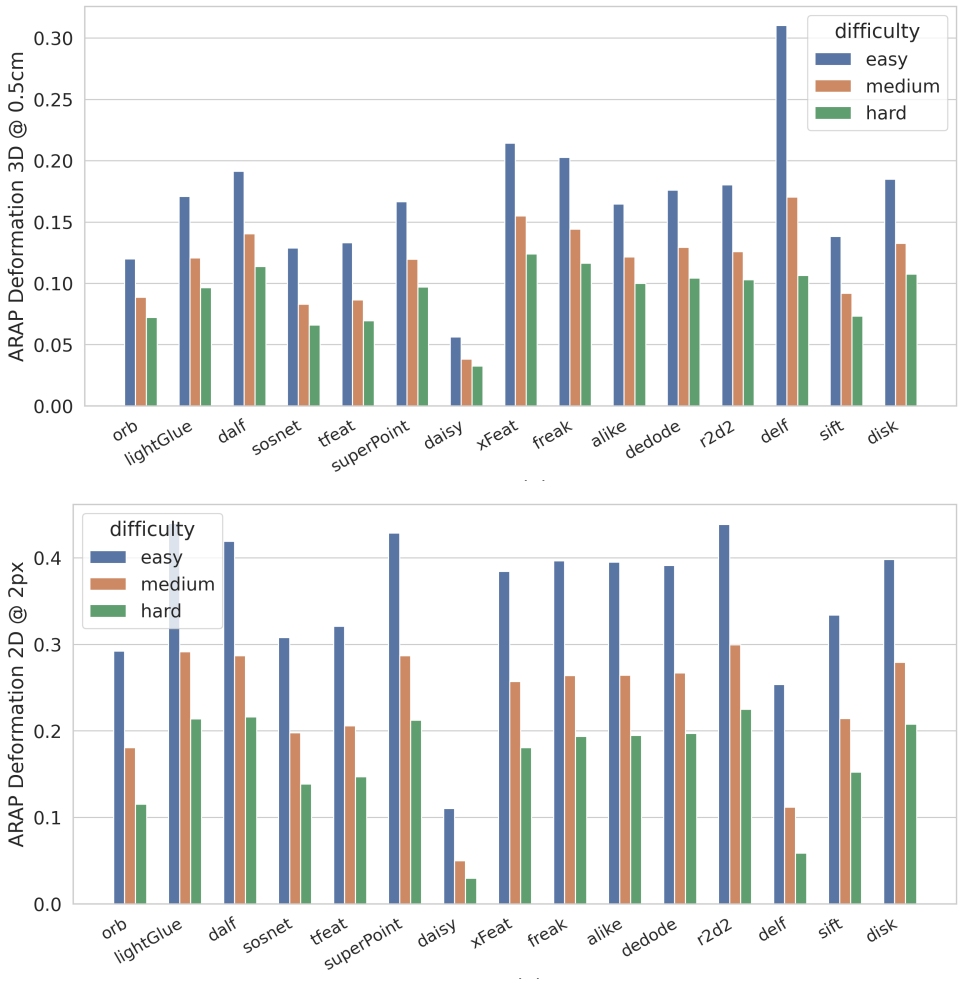


Figura 3. Gráfico com os resultados obtidos com diferentes métodos para a métrica ARAP-based surface registration em 3D e em 2D com as tolerâncias de 0.5 cm e 2 pixeis respectivamente.

É interresante observar que, para a métrica do *ARAP* em 2D e 3D, houveram variações nos resultados, com métodos como SuperPoint [4] tendo resultados melhores no 2D, enquanto métodos como o DELF [18] tiveram melhor desempenho no 3D. Isto ocorre devido aos processos de projeção intrínsecos a ambas as métricas, que em alguns casos suaviza erros moderados no 2D durante o processo de deformação do ARAP, pois considera a conectividade da malha formada. Por outro lado, *matches* que poderiam ser considerados corretos em um ambiente 2D, quando vistos em profundidade podem corresponder a superfícies diferentes, afetando assim a métrica.

4.5.2. Tolerância

Neste ponto, deseja-se analisar a sensebilidade dos resultados em relação aos diferentes valores de tolerância adotados para ambas as medidas. Esta analise busca destacar como a escolha do *threshold* afeta a interpretação do desempenho. Na Figura 4, é possivel observar a variação dos resultados dos diferentes algoritmos em relação aos diferentes níveis de tolerância para a tarefa de correspondência em objetos com deformações leves.

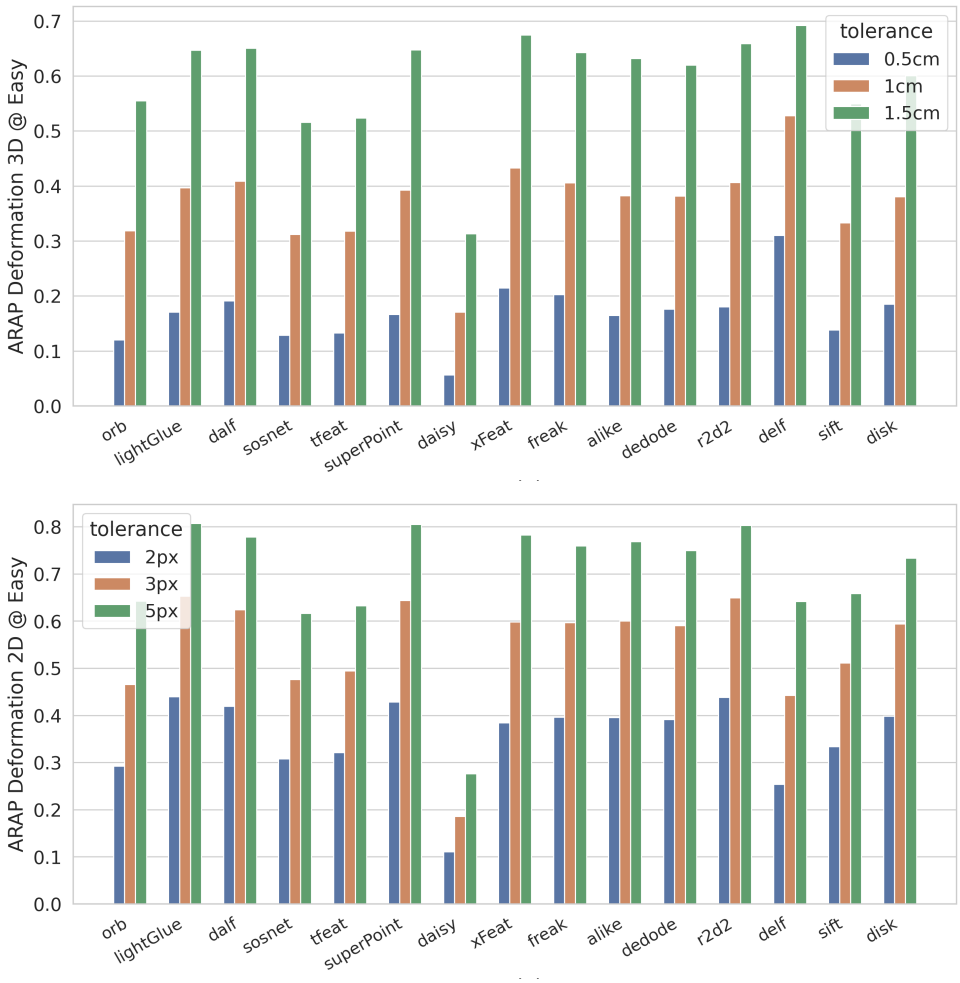


Figura 4. Gráfico com os resultados obtidos com diferentes métodos para a métrica ARAP-based surface registration em 3D e em 2D em um nível de deformação facíl para diferentes tolerâncias de erro

Como esperado, todos os métodos se beneficiaram com a flexibilização da tolerância e tiveram ganhos em suas acurácias. Na medida em 2D, o método que mais obteve ganhos foi o DELF [18] saindo de 0.253 com tolerância de 2px para 0.641 com tolerância de 5px, um ganho de 2.53 vezes, enquanto isso, o LightGlue [14] e o R2D2 [23] tiveram os menores ganahos com 1.835 e 1.831 vezes, respectivamente. Já nas medidas em 3D, os ganhos foram bem altos para todos os metodos, com destaque para o Daisy [29] que teve um ganho de 5.572 vezes, saindo de 0.056 para 0.313, em contra partida foi o DELF [18], que apresentou o menor ganho com apenas 2.229 vezes, que já é um ganho considerável.

4.5.3. Explorando outras configurações

Por fim, nesta etapa deseja-se explorar as variações de ponto de vista e iluminação para identificar seus impactos no desempenho dos métodos avaliados e por consequência sua robustez a este tipo de transformação. Além disso, este é provavelmente o ambiente mais desafiador possível dentro do *dataset* e também aquele que mais replica situações de captura de imagens reais no dia a dia. Na Figura 5, mostra-se os resultados das medidas em 3D de cada método para variações de iluminação e ponto de vista simultaneamente, além dos três níveis de deformação com um *threshold* de 0.5 cm.

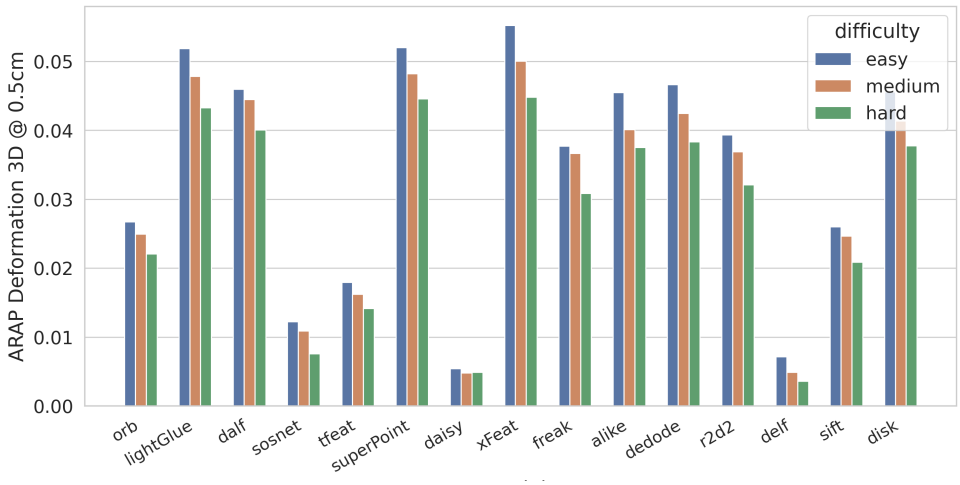


Figura 5. Gráfico com os resultados obtidos com diferentes métodos para a métrica ARAP-based surface registration em 3D com threshold de 0.5 cm e variação de iluminação e ponto de vista simultâneos

Observando a Figura 5 em conjunto com a Figura 3, nota-se que a queda dos resultados com estas transformações foi altíssima com os melhores resultados próximos de 0.05, demonstrando de fato a dificuldade deste ambiente. Estes resultados se devem a uma alta taxa de falhas nos processos de avaliação com o *ARAP*, com falta de pontos para a geração das malhas ou malhas desconexas que não conseguem ser deformadas pelo algortimo. Um exemplo disto é o SOSNet [28], que conseguiu gerar malhas validas para avaliação em apenas 11% dos pares comparados.

5. Considerações finais

Ao longo deste trabalho, avanços significativos foram feitos, não somente pelo autor desta monografia mas pelos demais integrantes da pesquisa. Avanços os quais não foram contemplados neste trabalho. Sendo estes, a melhoria das simulações do *dataset*, tornando as deformações mais fisicamente plausíveis e a adição de abordagens modernas como RoMa [32] e OmniGLue [12] no baseline do projeto.

Os experimentos executados durante a pesquisa, indicam que há um espaço considerável para o aprimoramento dos métodos da literatura atual dentro desta tarefa, visto que nenhum modelo apresentou um bom desempenho nas métricas avaliadas, principalmente o ARAP-based surface registration, e mais ainda em suas tolerâncias mais restritivas. Para além, validou-se a relevância de um *dataset* como o *Nonrigid Image Matching Benchmark* como um desafio a ser superado por outros pesquisadores.

Como passo futuros para a pesquisa, busca-se melhorar as deformações geradas no *dataset*, com o intuito de torná-las mais realistas, apesar dos já recentes esforços realizados neste sentido durante o desenvolvimento desta monografia. Para isso, estuda-se a possibilidade da utilização da plataforma Genesis [9]. Esta é uma plataforma de simulação física originalmente desenvolvida para aplicações de robótica e semelhantes, contudo os resultados preliminares de deformações realizadas na plataforma foram promissores e se apresentou como um caminho viavél de melhoria.

Outros trabalhos futuros são: a avaliação de modelos densos como o RoMa [32] nos experimentos descritos na secção de Trabalho Desenvolvido, o *fine-tuning* de parte dos métodos dentro do conjunto de treinamento e posterior avaliação, a expansão dos experimentos para os conjuntos de testes *Multi-Object* e *Scale* e a execução de testes cruzados com outros datasets de deformação não rígida.

6. Agradecimentos

Gostaria de expressar meus agradecimentos ao meu orientador, Prof. Dr. Erickson Rangel do Nascimento, pela orientação desta monografia e pelas aulas ministradas de Visão Computacional, que foram fundamentais para o trabalho. Agradeço também ao Aluno de Doutorado, Felipe Cadar Chamone pela colaboração, troca de ideias e aconselhamentos durante toda a execução do projeto.

Gostaria de prestar minha gratidão com Universidade Federal de Minas Gerais (UFMG) e o Departamento de Ciências da Computação (DCC), por todo o conhecimento adquirido durante o curso. Esta gratidão se extende também para os meus chefes e colegas de trabalho da Invent Vision, que foram cruciais no baleceamento entre a vida acadêmica e profissional.

Por fim, mas não menos importante, agradeço a minha familia pelo apoio e compreensão durante todo este processo.

Referências

[1] Alexandre Alahi, Raphael Ortiz, and Pierre Vandergheynst. Freak: Fast retina keypoint. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 510–517. IEEE, 2012. 2, 6

[2] Vassileios Balntas, Edgar Riba, Daniel Ponsa, and Krystian Mikolajczyk. Learning local feature descriptors with triplets and shallow convolutional neural networks. pages 119.1–119.11, 2016. 2, 6

[3] Aljaž Božič. Deepdeform: Learning non-rigid rgb-d reconstruction with semi-supervised data. *CVPR*, 2020. 2, 3, 4

[4] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superpoint: Self-supervised interest point detection and description. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 224–236, 2018. 2, 3, 6, 7

[5] Gianluca Donato and Serge Belongie. Approximate thin plate spline mappings. 2001. 3

[6] Laura Downs, Anthony Francis, Nate Koenig, Brandon Kinman, Ryan Hickman, Krista Reymann, Thomas B. McHugh, and Vincent Vanhoucke. Google scanned objects: A high-quality dataset of 3d scanned household items, 2022. 4

[7] Johan Edstedt, Georg Bökman, Mårten Wadenbäck, and Michael Felsberg. Dedode: Detect, don't describe – describe, don't detect for local feature matching, 2023. 2, 6

[8] Ibrahim El Rube'. Two-fold and symmetric repeatability rates for comparing keypoint detectors. *Computers, Materials & Continua*, 73(3):6495–6511, 2022. 4, 5, 6

[9] Equipe Genesis World. Genesis world documentation, 2025. Acesso em: 03 dez. 2025. 9

[10] P. Haven. Poly haven. `https://polyhaven.com/`, 2024. Acessado em 03/05/2024. 4

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015. 3

[12] Hanwen Jiang, Arjun Karpur, Bingyi Cao, Qixing Huang, and Andre Araujo. Omniglue: Generalizable feature matching with foundation model guidance, 2024. 9

[13] Yang Li, Hiroshi Takehara, Takafumi Taketomi, Bo Zheng, and Matthias Niessner. 4dcomplete: Non-rigid motion estimation beyond the observable surface. In *ICCV*, 2021. 3, 4

[14] Philipp Lindenberger, Paul-Edouard Sarlin, and Marc Pollefeys. Lightglue: Local feature matching at light speed, 2023. 3, 6, 8

[15] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004. 2, 6

[16] Krystian Mikolajczyk, Tinne Tuytelaars, Cordelia Schmid, Andrew Zisserman, Jiri Matas, Frederik Schaffalitzky, Timor Kadir, and Luc Van Gool. A comparison of affine region detectors. *International Journal of Computer Vision*, 2005. 4, 5, 6

[17] Erickson R. Nascimento, Guilherme Potje, Renato Martins, Francisco Chamone, Mario Campos, and Ruzena Bajcsy. Geobit: A geodesic-based binary descriptor invariant to non-rigid deformations for rgb-d images. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 10003–10011. IEEE, 2019. 3, 4

[18] Hyeonwoo Noh, Andre Araujo, Jack Sim, Tobias Weyand, and Bohyung Han. Large-scale image retrieval with attentive deep local features, 2018. 3, 6, 7, 8

[19] Guilherme Potje, Renato Martins, Felipe Cadar, and Erickson R. Nascimento. Deal: Extracting deformation-aware local features by learning to deform. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 10759–10771, 2021. 3, 4, 5, 6

[20] Guilherme Potje, Felipe Cadar, André Araujo, Renato Martins, and Erickson R. Nascimento. Enhancing deformable local features by jointly learning to detect and describe keypoints. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 3, 5, 6

[21] Guilherme Potje, Felipe Cadar, Andre Araujo, Renato Martins, and Erickson R. Nascimento. Xfeat: Accelerated features for lightweight image matching, 2024. 2, 6

[22] David Recasens, Martin R. Oswald, Marc Pollefeys, and Javier Civera. The drunkard's odometry: Estimating camera motion in deforming scenes. In *arXiv preprint arXiv:2306.16917*, 2023. 3

[23] Jerome Revaud, Philippe Weinzaepfel, César De Souza, Noe Pion, Gabriela Csurka, Yohann Cabon, and Martin Humenberger. R2d2: Repeatable and reliable detector and descriptor, 2019. 2, 6, 8

[24] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation, 2015. 2

[25] Edward Rosten, Reid Porter, and Tom Drummond. Faster and better: A machine learning approach to corner detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(1):105–119, 2010. 2

[26] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. Orb: An efficient alternative to sift or surf. In *2011 International Conference on Computer Vision*, pages 2564–2571. IEEE, 2011. 2, 6

[27] Olga Sorkine and Marc Alexa. As-rigid-as-possible surface modeling. *Eurographics Symposium on Geometry Processing*, 2007. 1, 3, 5

[28] Yurun Tian, Xin Yu, Bin Fan, Fuchao Wu, Huub Heijnen, and Vassileios Balntas. Sosnet: Second order similarity regularization for local descriptor learning, 2019. 2, 6, 9

[29] Engin Tola, Vincent Lepetit, and Pascal Fua. Daisy: An efficient dense descriptor applied to wide baseline stereo. *IEEE transactions on pattern analysis and machine intelligence*, 32:815–30, 2010. 2, 6, 8

[30] Michał J. Tyszkiewicz, Pascal Fua, and Eduard Trulls. Disk: Learning local features with policy gradient, 2020. 2, 6

[31] Xiaoming Zhao, Xingming Wu, Jinyu Miao, Weihai Chen, Peter C. Y. Chen, and Zhengguo Li. Alike: Accurate and lightweight keypoint detection and descriptor extraction, 2022. 3, 6

[32] Tianwei Zhou, Xingyu Wang, Yuhang Xu, Li Zhang, and Shengyu Zheng. Roma: Robust matching via dense feature

graph optimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 2, 9