

## Reconstrução 3D de Blocos de Rochas Ornamentais Utilizando Visão Estéreo e SfM

Nicolas G. Cavalcante\* Clebeson Canuto\* Raquel F. Vassallo\*  
Daniel L. Cosmo\*\*

\* *Departamento de Engenharia Elétrica, Universidade Federal do Espírito Santo, ES, (e-mail: nicolas.cavalcante@edu.ufes.br, clebeson.canuto@gmail.com, raquel.vassallo@ufes.br).*

\*\* *MOGAI, Mogai Tecnologia de Informação*

---

**Abstract:** Traffic accidents involving the transport of ornamental stones are largely due to non-compliance with legislation. In addition, there is little inspection, which contributes to an increase in disobedience. However, recently, there has been a great advance in the field of computer vision, especially in the area of semantic analysis of objects. In this sense, this work proposes an alternative approach to the inspection of truck loading carrying ornamental stones. The approach involves a pipeline of 3D reconstruction from images, a semantic segmentation network, and a cuboid fitting algorithm. To reduce processing time, a keyframe selector algorithm is proposed. The presented pipeline has shown practical potential, although changes are still needed to bring it closer to a real-time operation.

**Resumo:** Acidentes de trânsito envolvendo o transporte de rochas ornamentais são em grande parte devido ao não cumprimento da legislação. Além disso, a fiscalização é pouca, o que colabora para o aumento da desobediência. Entretanto, houve um grande avanço nos últimos anos nas técnicas de visão computacional, em destaque para a área de análise semântica de objetos. Nesse sentido, este trabalho propõe uma abordagem alternativa para a fiscalização do carregamento de caminhões transportando rochas ornamentais. A abordagem envolve um pipeline de reconstrução 3D a partir de imagens, uma rede de segmentação semântica, e um algoritmo de ajuste de cuboide. Para diminuir o tempo de processamento, é proposto um algoritmo seletor de *Keyframes*. O pipeline exposto mostrou possuir potencial prático, embora ainda seja necessário alterações para torná-lo mais próximo de uma operação em tempo real.

**Keywords:** Semantic Segmentation; Deep Learning; Curve Fitting; Structure From Motion; 3D Reconstruction; Photogrammetry

**Palavras-chaves:** Segmentação Semântica; Aprendizado Profundo; Ajuste de Curva; Structure From Motion; Reconstrução 3D; Fotogrametria

---

### 1. INTRODUÇÃO

O Espírito Santo é o principal exportador de Rochas Ornamentais do Brasil. No ano de 2020, o estado acumulou US\$ 813 milhões de dólares e 187,3 mil toneladas em exportações, respectivamente 82,42% e 75,87% do total exportado pelo país (sindirochas, 2021). No mesmo período, o PIB do estado foi de R\$ 139,5 bilhões de reais (dos Santos Neves, 2020), o que mostra a relevância do setor no Espírito Santo.

No estado é transportada grande parte do volume de rochas ornamentais produzido no país e, por consequência, muitos acidentes envolvendo rochas ornamentais acabam por acontecer em suas rodovias. No dia 22 de junho de 2017, um acidente entre duas ambulâncias, um ônibus e

uma carreta carregada com rochas ornamentais, matou 22 pessoas em Guarapari, Km 343 da BR-101. No dia 10 de setembro de 2017, um outro acidente envolvendo um caminhão carregado de placas de granito levou a óbito outras 11 pessoas.

Diante deste contexto, medidas de prevenção e regulação são necessárias. De acordo com o presidente do Sindicato da Indústria de Rochas Ornamentais, Cal e Calcário do Espírito Santo (Sindirochas), Tales Machado, a fiscalização da Polícia Rodoviária Federal é insuficiente para prevenir os acidentes nas estradas. “A resolução (que regulamenta o transporte) é boa, mas a fiscalização tem que obrigar o uso. A fiscalização é fraca. A PRF não consegue fiscalizar sozinha” (Vaillant, 2017). Em entrevista à radio CBN Vitória<sup>1</sup>, Tales Machado fala sobre a imprecisão das medidas dentro das pedreiras, feita por estimativa, com margem de erro de 10% a 15% do peso. “Lá, na pedreira, você tira as medidas do bloco e faz uma estimativa. Para ter certeza, só levando o carro na balança. A Polícia Rodoviária Federal

---

\* Trabalho realizado com o apoio da FAPES - Fundação de Amparo a Pesquisa e Inovação do Espírito Santo, edital 10/2019, através do projeto “Sistema automático para pesagem de blocos de rochas ornamentais e distribuição do peso pelos eixos do caminhão utilizando tecnologia de câmera 3D e vídeo analytics”, da MOGAI - Mogai Tecnologia de Informação

---

<sup>1</sup> 30 de jun. de 2017

faz isso: uma fiscalização por amostragem. Os casos que ela pega com suposto excesso, ela leva para a balança”.

Diante da problemática exposta, este trabalho explora uma alternativa para a medida do peso da rocha, por uso de técnicas de visão computacional. Essa abordagem possui algumas vantagens em relação aos métodos tradicionais:

- Em Relação às medidas feitas dentro das pedreiras, por estimativa:
  - (1) é possível atingir maior precisão.
- Em Relação à balança, é um sistema de maior:
  - (1) Mobilidade;
  - (2) Agilidade;
  - (3) Praticidade;
  - (4) Oferece medida de volume.

Trabalhos na literatura como (Moulon et al., 2012; Janosek and Pajdla, 2011; Schönberger and Frahm, 2016; Schönberger et al., 2016) são métodos consolidados, de código aberto, que resolvem o problema de reconstrução 3D a partir de imagens. Todos eles aplicam o método chamado SfM (*Structure from Motion*) para realizar uma reconstrução 3D de cenas ou objetos de forma iterativa e acumulativa, à medida que novas imagens vão sendo adicionadas ao processo de reconstrução, resultando em modelos tridimensionais mais completos e precisos. O resultado final é uma nuvem de pontos densa ou uma malha 3D, contendo todos os pontos que conseguiram ser triangulados pelas câmeras.

No corte e transporte, as rochas ornamentais normalmente não são pesadas no local. Quando ocorre a pesagem, normalmente os blocos já estão colocados sobre o caminhão. Nesse caso, para o processo de reconstrução é interessante que apenas o bloco seja reconstruído e não todo o cenário, incluindo o caminhão. Além disso, para agilizar a fiscalização e aumentar o número de caminhões fiscalizados, seria muito interessante que a reconstrução dos blocos e sua estimativa de volume e peso, pudesse ser feita sem a necessidade de parada completa do veículo. Esse é outro ponto importante, já que normalmente as técnicas de reconstrução 3D atuais ainda não conseguem ser utilizadas nas cenas em que os elementos estão em movimento. Nesse caso, se o bloco de rocha ornamental pudesse ser segmentado do veículo e apenas suas imagens fossem usadas no processo de reconstrução, o conjunto de imagens apresentado ao algoritmo de reconstrução 3D seria similar ao da cena estática. Sendo assim, um número muito maior de inspeções poderia ser realizado, aumentando a segurança das rodovias, prevenindo acidentes e inibindo a sobrecarga no transporte.

Desta forma, o objetivo deste trabalho é, a partir de pares de imagens obtidas de um sistema estéreo, realizar a segmentação de um bloco de rocha ornamental, isolando-o do resto da imagem, e então realizar a sua reconstrução 3D, calculando o seu volume e peso. A intenção é que tal método possa ser utilizado tanto em imagens estáticas quanto em imagens obtidas com o bloco em movimento.

Uma vez que este trabalho ainda se encontra em desenvolvimento, nesse artigo, serão apresentados, além do método, os resultados obtidos até então com imagens onde o bloco se encontra parado sobre o caminhão ou no chão.

A aplicação do método nos casos do bloco em movimento ainda se encontra em avaliação e os resultados ainda não são conclusivos.

## 2. METODOLOGIA

Esta seção descreve uma visão geral do *pipeline* proposto, mostrado na Figura 1. A seção seguinte detalha os passos destacados.

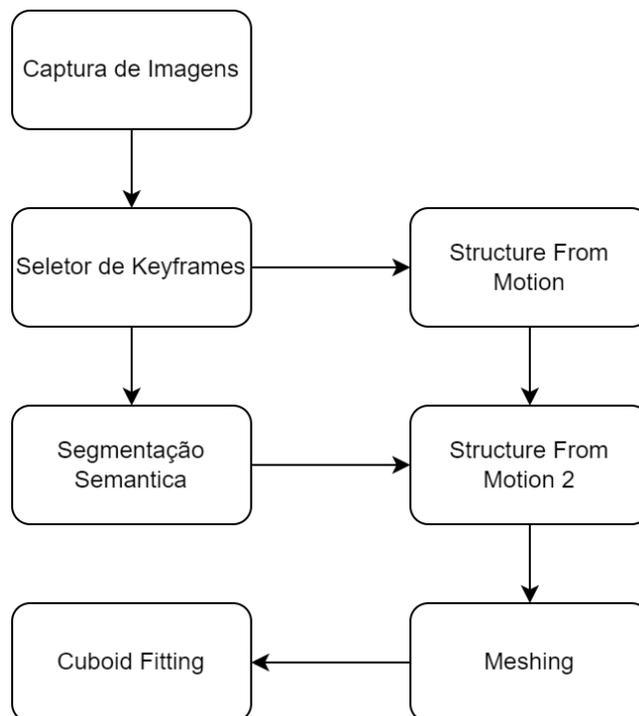


Figura 1. Pipeline de Reconstrução 3D de Rochas.

Inicialmente, um operador realiza a filmagem da rocha, dando uma volta completa de 360° ao redor do caminhão. A filmagem gera uma grande quantidade de imagens, normalmente além do necessário para uma boa reconstrução, consumindo um tempo excessivo de processamento caso todas elas fossem usadas na reconstrução. Para lidar com esse problema, foi proposto um algoritmo *Seleção de Keyframes* que visa reduzir a quantidade de imagens sem prejudicar a reconstrução.

Com esse conjunto de imagens selecionado, é executado um *pipeline* de *Structure from Motion*, de onde se obtém a posição das câmeras. Além disso, as imagens selecionadas são segmentadas, separando a rocha do fundo da imagem, com o uso de uma rede neural de *Segmentação Semântica*.

Um novo *Structure from Motion* é executado, dessa vez usando as imagens segmentadas e as posturas (posição e orientação) das câmeras recuperadas a partir do primeiro *Structure from Motion*. O resultado é uma nuvem de pontos esparsa contendo apenas a rocha.

Com a postura das câmeras e a nuvem de pontos da rocha, é executado um *pipeline* de *Meshing*, gerando uma malha de pontos da rocha. Finalmente, sobre essa

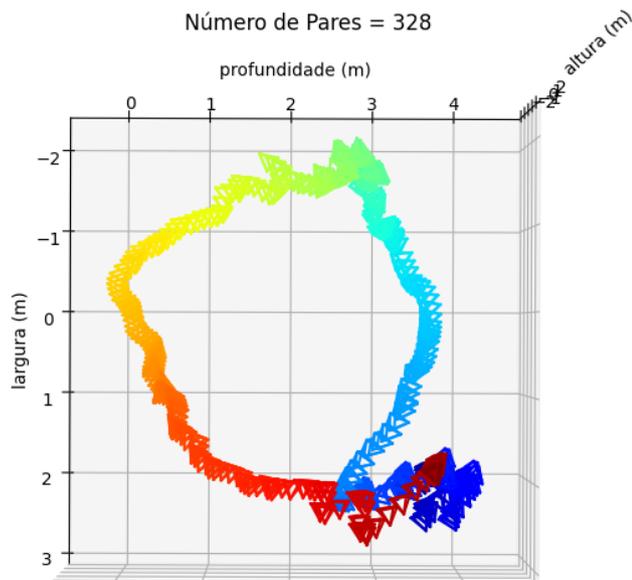


Figura 2. Posturas das câmeras sem *Seleção de Keyframes*

malha, é executado um algoritmo de *Cuboid Fitting* para obtenção das dimensões da rocha.

A seguir serão detalhadas cada uma das etapas descritas anteriormente e mostradas na Figura 1.

### 2.1 Seleção de Keyframes

Dada uma sequência de imagens  $I$ , ordenada cronologicamente, e um limiar mínimo  $T$  entre 0 e 1, e considerando o conjunto de imagens selecionadas como  $S$ , a seleção dos *keyframes* acontece conforme descrito a seguir.

- (1) Toma-se a primeira imagem do conjunto e extraem-se os pontos de interesse e descritores. Tal imagem é considerada o *keyframe* atual,  $F_k$ . A quantidade de pontos de interesse de  $F_k$  é  $N$ .
- (2) Enquanto houver imagens no conjunto  $I$ , retira-se a próxima imagem, extraíndo-se os pontos de interesse e descritores.
- (3) Faz-se, então, a correspondência de descritores da imagem retirada de  $I$  com  $F_k$ , utilizando verificação cruzada. É obtida uma quantidade  $M$  de correspondências. Se a proporção de correspondências  $\frac{M}{N} < T$ ,  $F_k$  é adicionado ao conjunto  $S$ , e a imagem retirada de  $I$  passa a ser o novo  $F_k$ . Caso contrário, tal imagem é descartada.

As Figuras 2 e 3 mostram, através das diferentes posturas das câmeras, um exemplo do número de *keyframes* a serem usados na reconstrução, com e sem o uso do *Seleção de Keyframes*. No exemplo apresentado, 72 pares são considerados suficientes para que a reconstrução seja realizada com sucesso, sem a necessidade de se utilizar todos os 328 pares originais de imagens estéreo, capturados no momento da filmagem.

O critério e métrica usados para selecionar os pares de imagens que compõem o conjunto  $S$  serão mostrados mais adiante na Seção 3.

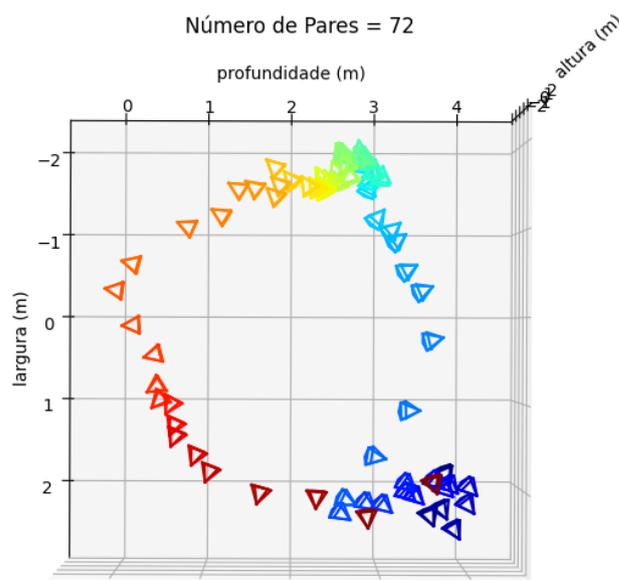


Figura 3. Posturas das câmeras com *Seleção de Keyframes*

### 2.2 Structure From Motion

O *Structure From Motion* é uma técnica para estimar estruturas tridimensionais a partir de imagens bidimensionais, além do movimento tridimensional ou postura das câmeras. Nesse trabalho foi utilizado o *pipeline* de SfM do Meshroom (Griwodz et al., 2021). Nesta abordagem, primeiramente é executado um algoritmo de *Feature Extraction*, para localizar pontos repetidos entre imagens. Depois, o *Image Matching* seleciona pares de imagens a serem utilizados na etapa posterior. Por fim *Feature Matching* realiza a correspondência de pontos entre os pares de imagens, e esses dados são utilizados no SfM, o qual retorna um modelo 3D do objeto reconstruído.

### 2.3 Segmentação Semântica

A segmentação semântica em visão computacional é uma técnica utilizada para separar regiões de interesse em imagens. Nessa parte do *pipeline*, é utilizada uma rede neural de segmentação semântica para separar a rocha do resto da imagem, como mostrado na Figura 4.

Foi utilizada uma arquitetura *encoder-decoder*, DeepLabv3 (Chen et al., 2017) com o ResNet-101 (He et al., 2015) no *encoder*, previamente treinado no banco de dados *ImageNet*, um banco de dados que possui 1.2 milhões de imagens e 1000 classes (Deng et al., 2009). Arquiteturas *encoder-decoder* são muito comuns em tarefas de segmentação semântica. O *encoder* é responsável por capturar *features* das imagens, e o *decoder* por agregar as relações espaciais.

Para o treinamento, foram coletadas imagens de rochas, tanto em cima do caminhão quanto no chão (ver Figura 5). Foi feita a anotação (Bréhéret, 2017) de 7.744 imagens, das quais 4.202 correspondem a 31 rochas no chão, e 3.542 correspondem a 7 rochas no caminhão. Além disso, uma pequena quantidade de imagens da internet em condições similares ao conjunto de dados coletado foi utilizada.

O conjunto de dados marcado possui um desequilíbrio na quantidade de imagens por bloco. A rocha mais frequente

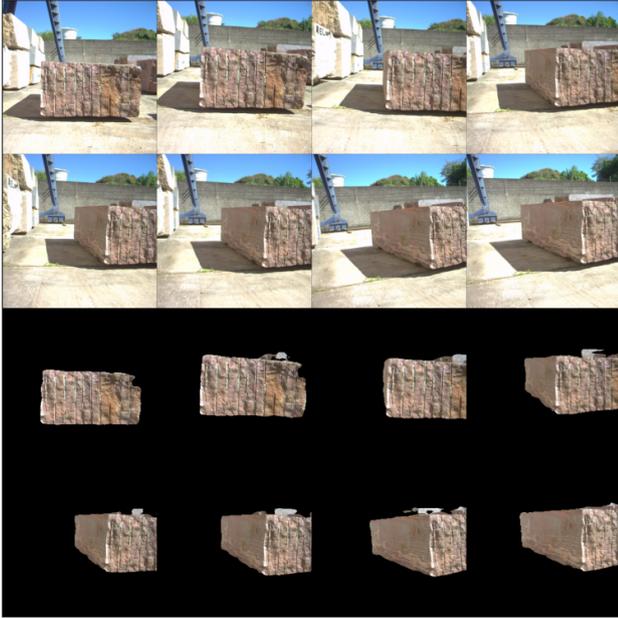


Figura 4. Conjunto de imagens com a rocha segmentada.



Figura 5. Amostra do conjunto de dados coletado

possui 2.225 imagens e a menos frequente, 48. A fim de mitigar esse problema, foram testadas alternativas para o balanceamento dos dados:

- *Undersample* - É utilizado um limite máximo de imagens por rocha, descartando aleatoriamente as imagens sobressalentes.
- *Oversample* - É utilizado um limite mínimo de imagens por rocha, amostrando e adicionando ao conjunto original até atingir o limite mínimo.
- *Equalsample* - União das técnicas de *Undersample* e *Oversample*
- *Ponderação de Classes* - São atribuídos pesos às imagens na função de perda, de forma a equalizar a contribuição das rochas na otimização da rede neural.

- *Nada* - para verificar a efetividade das técnicas de balanceamento, verificou-se o efeito do não uso de técnicas de balanceamento.

Por se utilizar uma rede pré-treinada, aplicou-se a técnica de *Transfer Learning*. Durante o *Transfer Learning*, os pesos do *backbone* (*encoder*) da rede não são alterados, e somente os pesos do *decoder* são modificados. Dos hiperparâmetros do modelo, também foi variado o número de épocas de *Transfer Learning*.

Pelo fato desse conjunto de dados ter sido coletado a partir de vídeos de um objeto em foco, existe uma similaridade muito grande entre imagens consecutivas. Caso essas imagens sejam separadas no treino e validação, a capacidade do sistema de medir a generalização da rede ficaria comprometida. Para evitar esse problema, foi feita a divisão dos dados de treino, validação e teste de maneira que não houvesse repetição de rochas entre os três conjuntos.

#### 2.4 Meshing

Essa etapa é responsável por gerar, a partir de uma nuvem esparsa de pontos e a posição das câmeras, uma malha triangulada. Para isso, primeiramente são gerados os mapas de profundidade para cada imagem. Depois, é realizada a fusão dos mapas. Nesse trabalho, para essa etapa, foi utilizado o *pipeline* de SfM do Meshroom (Griwodz et al., 2021).

#### 2.5 Cuboid fitting

Partindo de uma nuvem de pontos, é encontrado o cuboide com melhor ajuste aos pontos. Esse cuboide é descrito pelos seguintes parâmetros: os de rotação  $r_x, r_y$  e  $r_z$ , um ponto de origem  $o_x, o_y$  e  $o_z$ , e um ponto diagonalmente oposto a este  $d_x, d_y, d_z$ . Com essa parametrização, os comprimentos das arestas do cuboide são dados pela seguinte expressão:  $l_x, l_y, l_z = R [|d_x - o_x| |d_y - o_y| |d_z - o_z|]$ , onde  $R$  é a matriz de rotação formada pelos parâmetros  $r_x, r_y$  e  $r_z$ . Primeiro, os pontos são rotacionados pela matriz formada pelos autovetores da matriz de covariância da nuvem de pontos. Então, é subtraída a média. Obtêm-se com isso, os valores máximos e mínimos em cada um dos eixos  $x, y$  e  $z$ , representados por  $x_{min}, y_{min}, z_{min}$  e  $x_{max}, y_{max}, z_{max}$ . A estimativa inicial dos parâmetros do problema são:

$$\begin{aligned} o_x &= x_{min} & o_y &= y_{min} & o_z &= z_{min} \\ r_x &= 0.0 & r_y &= 0.0 & r_z &= 0.0 \\ d_x &= x_{max} & d_y &= y_{max} & d_z &= z_{max} \end{aligned} \quad (1)$$

Essa estimativa inicial é otimizada utilizando o algoritmo de Gauss-Newton. A função objetivo é dada por:

$$\min_{o,r,d} \sum_{i=1}^N \min((P(o, r, d) \begin{bmatrix} p_i \\ 1 \end{bmatrix})^2) \quad (2)$$

onde  $P(o, r, d)$  representa as equações dos 6 planos do cuboide que são dadas por:

$$P(o, r, d) = \begin{bmatrix} R & -R \begin{bmatrix} o_x \\ o_y \\ o_z \end{bmatrix} \\ R & -R \begin{bmatrix} d_x \\ d_y \\ d_z \end{bmatrix} \end{bmatrix} \quad (3)$$

### 3. EXPERIMENTOS E RESULTADOS

#### 3.1 Seletor de Keyframes

Para se testar o seletor, foi utilizada uma filmagem com a rocha no caminhão, capturada a uma taxa de 7 quadros por segundo. Inicialmente, executou-se o *pipeline* proposto até a etapa de *Structure From Motion*, sem a etapa do *Seletor de Keyframes*, e a posição das câmeras obtida foi utilizada como *baseline* de comparação.

Então, usando o seletor, foi variado o valor do limiar de 0.7 até 0.3, e medido o tempo de processamento somado do *Seletor de Keyframes* com o SfM, para cada um dos limiares. Para comparar a qualidade da reconstrução, a posição final dos *KeyFrames* (KF) selecionados pelo algoritmo de seleção, foi comparada com a posição das respectivas imagens do *baseline*. Então, foi medido o erro de translação, dado por (4):

$$\frac{1}{N} \sum_{n=1}^N \|t_n - t_{KF,n}\|^2 \quad (4)$$

onde  $t_n$  representa o vetor de translação da imagem calculada pelo *baseline* e  $t_{KF,n}$  representa o vetor de traslação da imagem calculada pelo *pipeline* com filtragem.

O erro de rotação foi medido por (5):

$$\frac{1}{N} \sum_{n=1}^N \arccos\left(\frac{\text{Tr}(R_n^T R_{KF,n}) - 1}{2}\right) \quad (5)$$

onde  $R_n$  representa a matriz de rotação da imagem calculada pelo *baseline* e  $R_{KF,n}$  representa a matriz de rotação da imagem calculada pelo *pipeline* com filtragem.

Os resultados são mostrados nas Tabela 1 e 2.

Tabela 1. Seletor de Keyframes:  
Custo de Processamento, seletor + SfM

Limiar	Nº de Imagens	Tempo(minutos)
1,00	1074	329,39
0,70	1016	304,58
0,60	776	196,11
0,55	616	146,21
0,50	466	96,27
0,45	344	66,87
0,40	254	42,90
0,35	190	67,28
0,30	118	25,60

Como esperado, conforme o valor do limiar diminui, o número de imagens e o tempo total de processamento diminui, e o erro de rotação e translação aumentam.

Uma exceção ocorre no limiar de 0,35 devido ao funcionamento da etapa de *Image Matching*. Até 200 imagens,

Tabela 2. Seletor de Keyframes: Erro Médio

Limiar	Erro de Rotação (rad)	Erro de Translação(m)
1,0000	0,0000	0,0000
0,7000	0,0102	0,0956
0,6000	0,0141	0,1403
0,5500	0,0132	0,0205
0,5000	0,0765	0,4182
0,4500	0,0329	0,2779
0,4000	0,3928	0,8216
0,3500	0,1819	0,6851
0,3000	0,3064	0,7710

todos os pares são elencados para o *Feature Matching*, em uma abordagem de força bruta. Acima dessa quantidade de imagens, é utilizada uma *Vocabulary Tree*(Nister and Stewenius, 2006), para encontrar subconjuntos de imagens compartilhando conteúdo similar e limitar a quantidade de pares correspondentes. Nos experimentos, foi estabelecido um máximo de 50 pares correspondentes por imagem. Sendo assim, como foram selecionadas 190 imagens para o valor de limiar de 0,35, foi realizada a correspondência de  $(190 * 189)/2 = 17955$  pares de imagens, enquanto que para o limiar de 0.40, foram selecionadas 254 imagens, resultando em até  $254*50=12700$  pares. Entretanto, no experimento, o algoritmo de *Image Matching* utilizado selecionou 8014 pares neste caso.

Uma outra exceção ocorre nos casos dos limiares de 0,45 e 0,55. Possivelmente esses filtros acabaram selecionando imagens com melhores características para a reconstrução, como por exemplo maior presença de pontos de interesse repetitivos ao longo de várias imagens.

De qualquer forma, a abordagem se mostrou simples e efetiva para obter uma diminuição no tempo de processamento, com um *Trade-off* de perda de qualidade. De acordo com esse experimento, um limiar entre 0,5 e 0,6 estabelece um bom compromisso entre a diminuição do tempo de processamento e o aumento dos erros de translação e rotação até níveis que não comprometem a qualidade da reconstrução.

Assim, foi aplicando-se esse critério e métricas, que se chegou, por exemplo, ao conjunto de 72 pares de imagens mostrado anteriormente na Figura 3.

#### 3.2 Segmentação Semântica

Para realizar o treinamento da rede, utilizou-se a linguagem de programação Python e a biblioteca de *Deep Learning fastai* (Howard et al., 2018), versão 2.5.3. Os experimentos foram executados em uma máquina Windows 10, em um ambiente virtualizado Linux, distribuição Ubuntu. Processador Intel Core i5-6400, 2.70GHz com 4 núcleos, 16GB de RAM, 1TB de unidade de armazenamento e placa gráfica Nvidia GTX 1060 6GB.

Para avaliar e comparar o desempenho dos modelos, foi utilizado o índice de *Jaccard*. Este índice, também conhecido como *Intersection Over Union* ou IoU, é uma das métricas mais utilizadas em tarefas de segmentação semântica.  $IoU = \frac{VP}{VP+FP+FN}$ , onde  $VP$  são os verdadeiros positivos (pixels verdadeiramente classificados como rocha),  $FP$  são os falso positivos (pixels erroneamente classificados como rocha) e  $FN$  são os falso negativos (pixels erroneamente classificados como não-rocha). Os

valores do índice de *Jaccard* no conjunto de validação para cada combinação de hiper-parâmetros são mostrados na Tabela 3.

Tabela 3. Índice de Jaccard da Rede de Segmentação

Freeze Epochs	Balance	Jaccard Valid
150	undersample	84,34%
150	weight	83,67%
5	none	83,13%
150	oversample	82,95%
5	weight	82,92%
5	oversample	82,85%
5	equalsample	82,73%
150	equalsample	82,23%
5	undersample	82,11%
150	none	80,57%

O modelo que obteve o melhor índice foi aquele em que os hiper-parâmetros são 150 épocas, com os pesos do *backbone* congelados, e com a estratégia de *undersample*. Testes iniciais deste modelo para a tarefa de medir as dimensões dos blocos não levaram a resultados satisfatórios. A rede de segmentação não foi suficientemente precisa, o que levou alguns blocos a não serem reconstruídos. Portanto, novas imagens foram coletadas e anotadas, totalizando 67894 imagens, das quais 50886 correspondem a 41 rochas no chão, e 17008 correspondem a 7 rochas no caminhão. Então, esse modelo foi re-treinado com a nova base de dados, e o resultado é mostrado na Tabela 4, onde o índice de *Jaccard* foi superado em 3%. Este modelo obteve o valor do índice de *Jaccard* no conjunto de teste de **90,98%**.

Tabela 4. Índice de Jaccard da Rede de Segmentação com Base de Dados Expandida

Freeze Epochs	Balance	Jaccard Valid
150	undersample	87,26%

### 3.3 Cálculo de Dimensões

Para base de comparação, foi coletado um conjunto de dados de rochas e coletada suas dimensões utilizando uma trena. No total, foram medidas 3 rochas. Cada rocha foi filmada 3 vezes, totalizando 9 filmagens. Dessas rochas, uma foi utilizada durante o treinamento da rede de segmentação juntamente com as demais imagens selecionadas para isso. As outras duas foram usadas unicamente para teste do método proposto.

Foram feitas 4 medidas por dimensão, igualmente espaçadas ao longo dos lados das rochas, totalizando 12 medidas por rocha. A Figura 6 e as Tabelas 5 e 6 resumem a comparação entre o volume calculado das rochas pelo método proposto e as medidas feitas com a trena, as quais foram tomadas como *groundtruth*.

Tabela 5. Volume Médio( $m^3$ ): Trena x Algoritmo.

Nome da Rocha	Algoritmo ( $m^3$ )	Trena ( $m^3$ )
2021-05-27_09-30-09	10,165173	10,369143
2022-03-18_10-32-25	12,308312	12,067241
2022-03-18_11-18-59	10,415394	9,890201

A rocha de nome “2021-05-27 09-30-09” foi utilizada durante o treinamento da rede de segmentação, e apresentou

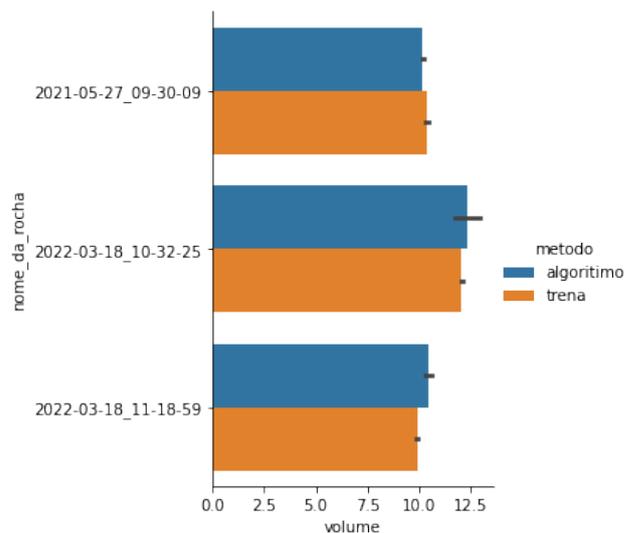


Figura 6. Comparação do Volume ( $m^3$ ) Trena x Algoritmo.

Tabela 6. Desvio Padrão do Volume ( $m^3$ ): Trena x Algoritmo.

Nome da Rocha	Algoritmo ( $m^3$ )	Trena ( $m^3$ )
2021-05-27_09-30-09	0,136790	0,315820
2022-03-18_10-32-25	1,697820	0,070062
2022-03-18_11-18-59	0,437817	0,192692

o resultado com menor desvio padrão entre os testes. A rocha de nome “2022-03-18 10-32-25” foi a que obteve o pior resultado. Essa filmagem acabou ficando com a exposição da câmera saturada durante a captura, o que prejudicou o bom funcionamento do *pipeline* proposto.

Mesmo assim, os resultados mostram que o método proposto fornece estimativas bem próximas dos valores de *groundtruth*. Acredita-se que para uma validação mais completa, novos exemplares de rochas deverão ser testados e a apreciação dos resultados por parte de profissionais da área de extração de rochas ornamentais deve ser realizada para a sua aprovação.

## 4. CONCLUSÃO E TRABALHOS FUTUROS

O trabalho apresentado neste artigo tem como objetivo isolar um bloco de rocha ornamental, através segmentação semântica em imagens estéreo, e depois realizar a sua reconstrução 3D, calculando o seu volume e peso.

O método proposto apresenta alguns pontos fortes para construir uma solução do problema. A rede de segmentação foi boa o suficiente para realizar a reconstrução 3D segmentada. Além disso, em todas as situações testadas, o algoritmo de ajuste de cuboide foi capaz de se adequar ao bloco da maneira esperada.

O maior gargalo atual é o tempo de processamento, que torna a solução inviável para fiscalização em tempo real. Porém, já existem algumas abordagens para se obter uma nuvem de pontos 3D a partir de um par de câmeras estéreo de forma mais rápida (Hirschmüller et al., 2012) (Tao et al., 2008). Tais métodos, complementados com o *pipeline* mostrado nesse trabalho, podem viabilizar o processamento em tempo real.

Para trabalhos futuros, existe o interesse de realizar a medida de volume da rocha sem realizar a parada completa do caminhão. Entretanto, o SfM utilizado foi idealizado para os casos onde a cena é estática e apenas a câmera se movimenta. Felizmente, o *pipeline* proposto pode ser adaptado para este caso, bastando realizar a segmentação das imagens antes de realizar a reconstrução 3D e localização da postura das câmeras. Portanto, é proposto o uso do *pipeline* adaptado para o caso em que a câmera está parada e a rocha no caminhão em movimento.

Tao, T., Koo, J.C., and Choi, H.R. (2008). A fast block matching algorithm for stereo correspondence. In 2008 IEEE Conference on Cybernetics and Intelligent Systems, 38–41. IEEE.

Vaillant, B. (2017). 'fiscalização é fraca', diz sindirochas sobre acidentes nas rodovias. [urlhttps://www.gazetaonline.com.br/noticias/cidades/2017/09/fiscalizacao-e-frac-a-diz-sindirochas-sobre-acidentes-nas-rodovias-1014099224.html](https://www.gazetaonline.com.br/noticias/cidades/2017/09/fiscalizacao-e-frac-a-diz-sindirochas-sobre-acidentes-nas-rodovias-1014099224.html).

## REFERÊNCIAS

- Bréhéret, A. (2017). Pixel Annotation Tool. <https://github.com/abreheret/PixelAnnotationTool>.
- Chen, L.C., Papandreou, G., Schroff, F., and Adam, H. (2017). Rethinking atrous convolution for semantic image segmentation. arXiv preprint arXiv:1706.05587.
- Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., and Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In 2009 IEEE conference on computer vision and pattern recognition, 248–255. Ieee.
- dos Santos Neves, I.J. (2020). PIB Trimestral do Espírito Santo. URL <http://www.ijns.gov.br/component/attachments/download/7290>.
- Griwodz, C., Gasparini, S., Calvet, L., Gurdjos, P., Castan, F., Maujean, B., De Lillo, G., and Lanthony, Y. (2021). Alicevision meshroom: An open-source 3d reconstruction pipeline. In Proceedings of the 12th ACM Multimedia Systems Conference, 241–247. ACM Press.
- He, K., Zhang, X., Ren, S., and Sun, J. (2015). Deep residual learning for image recognition. CoRR, abs/1512.03385. URL <http://arxiv.org/abs/1512.03385>.
- Hirschmüller, H., Buder, M., and Ernst, I. (2012). Memory efficient semi-global matching. ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 3, 371–376.
- Howard, J. et al. (2018). fastai. <https://github.com/fastai/fastai>.
- Jancosek, M. and Pajdla, T. (2011). Multi-view reconstruction preserving weakly-supported surfaces. In CVPR 2011. IEEE. doi:10.1109/cvpr.2011.5995693. URL <https://doi.org/10.1109/cvpr.2011.5995693>.
- Moulon, P., Monasse, P., and Marlet, R. (2012). Adaptive structure from motion with a contrario model estimation. In Proceedings of the Asian Computer Vision Conference (ACCV 2012), 257–270. Springer Berlin Heidelberg. doi:10.1007/978-3-642-37447-0\_20.
- Nister, D. and Stewenius, H. (2006). Scalable recognition with a vocabulary tree. In 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06), volume 2, 2161–2168. Ieee.
- Schönberger, J.L. and Frahm, J.M. (2016). Structure-from-motion revisited. In Conference on Computer Vision and Pattern Recognition (CVPR).
- Schönberger, J.L., Zheng, E., Pollefeys, M., and Frahm, J.M. (2016). Pixelwise view selection for unstructured multi-view stereo. In European Conference on Computer Vision (ECCV).
- sindirochas (2021). Informativo Mensal de Exportações de Rochas. URL <https://www.sindirochas.com/downloads/relatorios/Dezembro-2020-2.pdf>.