

Project 3 Writeup

Instructions

- Describe any interesting decisions you made to write your algorithm.
- Show and discuss the results of your algorithm.
- Feel free to include code snippets, images, and equations.
- Use as many pages as you need, but err on the short side. If you feel you only need to write a short amount to meet the brief, then
- **Please make this document anonymous.**

Introdução

Neste trabalho vamos criar um algoritmo de correspondência de recursos locais e tentar combinar várias visualizações de cenas do mundo real. Há centenas de artigos na literatura de visão computacional que abordam cada etapa. Vamos implementar uma versão simplificada do SIFT.

Implementação

Neste trabalho apenas o método de combinação *matching* foi implementado. Para realizar a combinação o método NNDR (*Nearest Neighbor Distance Ratio Test*) foi implementado, conforme o especificado em (*Szeliski 4.1.3; equation 4.18 in particular*).

Código utilizado para realização do NNDR:

```
1 function [matches, confidences] = match_features_new(  
    features1, features2, limit)  
2  
3     %calcula o metodo de "nearest neighbor distance ratio  
    test"  
4     %implementado sem as funcoes proibidas  
5  
6     for f1index = 1 : size(features1, 1)  
7  
8         %computa as diferencas entre todas as features1 e  
            features2 retiradas da imagem  
9         differences = zeros(size(features2, 1), 1);  
10  
11         for f2index = 1 : size(features2, 1)
```

```
12         differences(f2index) = sum(abs(features1(
13             flindex,:) - features2(f2index,:)));
14     end
15     %encontra o primeiro e o segundo vizinho mais
16     proximo
17     [mindiff1val, mindiff1index] = min(differences(
18         differences>0));
19     differences2 = differences .* (differences>
20         mindiff1val);
21     [mindiff2val, mindiff2index] = min(differences2(
22         differences2>0));
23
24     %calcula a semelhanca entre os dois pontos, mais
25     proximo de 1 mais semelhante
26     % o limite da semelhanca esta sendo passado como
27     0.7
28     semelha = mindiff1val / mindiff2val;
29
30     %se for semelhanca adiciona ao vetor match e
31     confidence
32     %incrementa o vetor de match indice
33     if semelha < limit
34         matches(matchindex, :) = [flindex;
35             mindiff1index];
36         confidences(matchindex) = 1 / semelha;
37         matchindex = matchindex + 1;
38     end
39
40     %truncate match and confidence vector
41     matches = matches(1:matchindex-1,:);
42     confidences = confidences(1:matchindex-1);
43 end
```

Dificuldades

Não consegui implementar uma solução para a obtenção das *features* das imagens sem a utilização das funções proibidas.

Outra dificuldade foi encontrar a solução para a questão Q2 no livro de Szeliski, utilizei o artigo *Invariant Representations and Learning for Computer Vision*¹ para sua resolução.

¹<https://www.robots.ox.ac.uk/~vedaldi/assets/pubs/vedaldi08invariant.pdf>