Graph Cuts Based Super Pixel Segmentation for VTK

Release 0.00

David Doria

March 16, 2010

Rensselaer Polytechnic Institute, Troy NY

Abstract

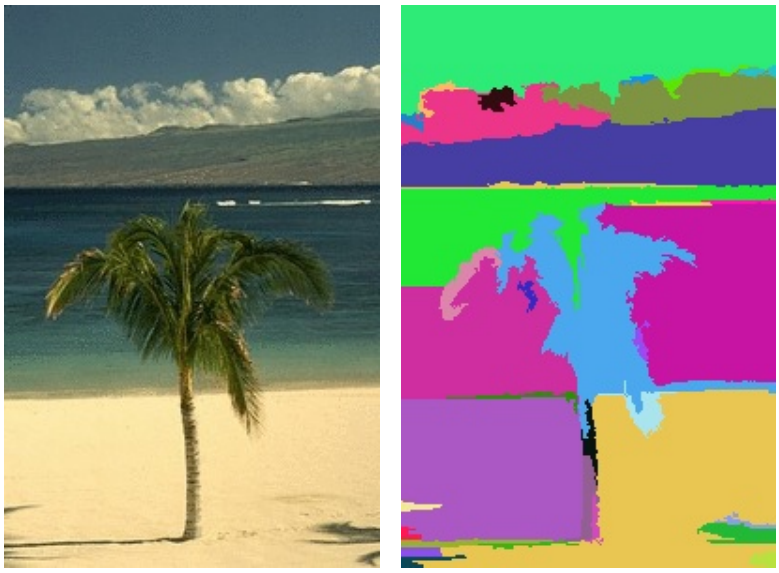
Segmenting images into "super pixels" is lately a very hot topic. One implementation of such a technique is [1]. We propose a new class, vtkSuperPixelSegmentation, which has wrapped Felzenszwalb's code (with permission) to provide this functionality in the VTK framework.

Contents

1 Introduction

Segmenting images into "super pixels" is lately a very hot topic. One implementation of such a technique is [1]. We propose a new class, vtkSuperPixelSegmentation, which has wrapped Felzenszwalb's code (with permission) to provide this functionality in the VTK framework.

2 Demonstration



(a) Input Image (b) Segmented Image

Figure 1: Segmentation demonstration.

3 Options

3.1 Scale of Observation, k

Sets the component size preference. Larger k causes a preference for larger components. The default is 500.

This paramter can be set with:

```
filter->SetK(400);
```

3.2 Gaussian Smoothing, σ

The input image is smoothed with a Gaussian filter before segmentation.

This paramter can be set with:

```
filter->SetSigma(400);
```

The default is 0.5.

3.3 Small Component Removal, *MinSize*

Small components are merged with neighboring components if they are smaller than *MinSize*.

This paramter can be set with:

```
filter->SetMinSize(400);
```

The default is 20.

4 Code Snippet

This example reads a jpeg image, segments it, and writes the output image.

```
vtkSmartPointer<vtkJPEGReader> reader =
  vtkSmartPointer<vtkJPEGReader>::New();
reader->SetFileName(argv[1]);
reader->Update();

vtkSmartPointer<vtkSuperPixelSegmentation> superPixelSegmentation =
    vtkSmartPointer<vtkSuperPixelSegmentation>::New();

superPixelSegmentation->SetSigma(.5);
superPixelSegmentation->SetK(500);
superPixelSegmentation->SetMinSize(50);

superPixelSegmentation->SetInputConnection(reader->GetOutputPort());
superPixelSegmentation->Update();

vtkSmartPointer<vtkJPEGWriter> writer =
    vtkSmartPointer<vtkJPEGWriter>::New();
writer->SetFileName(argv[2]);
writer->SetInputConnection(superPixelSegmentation->GetOutputPort());
writer->Write();
```

References

[1] Pedro Felzenszwalb and Daniel Huttenlocher. Efficient graph-based image segmentation. *International Journal of Computer Vision*, 2004. (document), 1