Entropy Image Filter

Release 0.01

Robert Tamburo[1]

December 13, 2010

[1]robert.tamburo@gmail.com

Abstract

This paper describes an intensity image filter http://www.itk.org/Doxygen318/html/group__IntensityImageFilters.html for computing the entropy of pixel values contained within a neighborhood centered at each input pixel. The output image contains the calculated entropy for each input pixel location. This paper is accompanied with source code for the filter and test, test images and parameters, and expected output images.

Latest version available at the Insight Journal [http://hdl.handle.net/10380/XXXXXX]

Contents

1 Implementation of Algorithm

The entropy image filter uses the same entropy computation as found in `Examples/Statistics/ImageEntropy1.cxx`.

Figure 1: The input image `sf4.png` for this example.

2 Example

This filter requires setting an input image `SetInput()` and the radius of the neighborhood with `SetRadius()`. The radius defaults to 1. The marginal scale and number of bins for the histogram can also be set. Example usage is shown below.

```
typedef itk::EntropyImageFilter<ImageType, ImageType> EntropyFilterType;
EntropyFilterType::Pointer entropyFilter = EntropyFilterType::New();
entropyFilter->SetInput(inputImage); // set the input image
ImageType::SizeType radius;
radius.Fill(10);
entropyFilter->SetRadius(radius); // set the desired neighborhood radius
entropyFilter->SetNumberOfBins(64); // set the number of histogram bins
entropyFilter->SetMarginalScale(100); // set the marginal scale
entropyFilter->Update();
```

For this example, `sf4.png` (Fig. 1) from the `Data` directory is used as the input image. Shown in Figure 2 is the output with a radius of 10, 64 histogram bins, and a marginal scale of 100.

3 Testing

The test code included with this article will generate images in Figures 2 with the folloiwing arguments.

```
EntropyImageFilter.exe images/sf4 .png 10 64 100
```

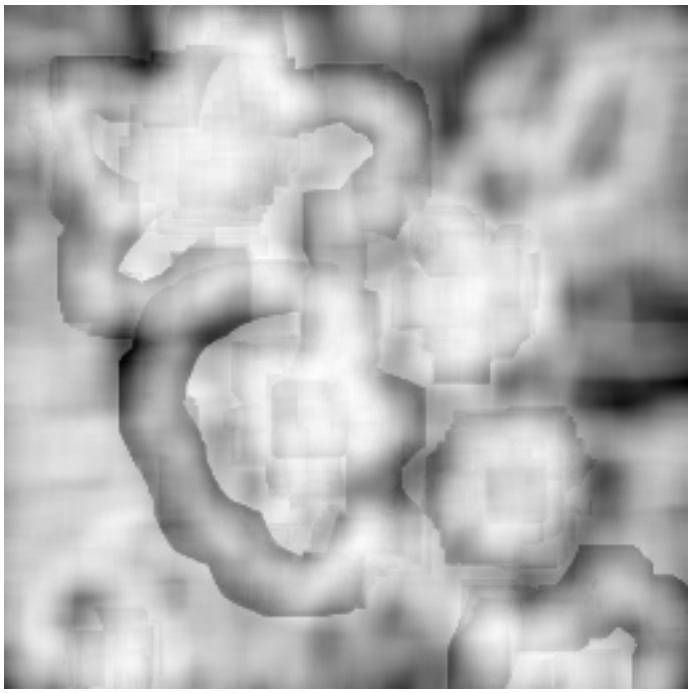

Figure 2: Output image with `radius = 10`.

4 Software Used

This filter was developed on a Windows 7 64-bit computer. It has been successfully tested with ITK version 3.18.0, MinGW version 5.1.6, and CMake version 2.8.2 (Windows binary), and gcc version 4.3.4 20090804 release 1 under cygwin.