



FuzzyConnectedness

This interactive command line application demonstrates the use of ITK components for performing object delineation using fuzzy connectedness.

What is Fuzzy Connectedness?

Fuzzy connectedness captures the idea of “hanging-togetherness” of image elements in an object. Fuzzy connectedness can be used for object segmentation as follows: a user specifies a seed point within the object they wish to delineate. Using dynamic programming, a “connectedness” value with respect to the seed point is computed for each pixel. The connectedness value represents the largest “strength” of all paths going from one pixel to another. The strength depends on how close pixels are spatially and in terms of intensity and intensity-based properties.

A binary mask of the delineated object can be obtained by thresholding the connectedness value map.

See the following article for more information on fuzzy connectedness based segmentation:

J. K. Udupa and S. Samarasekera, "Fuzzy connectedness and object definition: theory, algorithms, and application in image segmentation", *Graphical Models in Image Processing*, Vol 58, pp. 246-261, 1996

Application overview:

This application reads in a raw 3D image volume. The user specifies an initial seed point and an initial threshold value. A binary mask of the segmentation is computed. Each 2D slice from the input volume and from binary mask is written out as PGM files – facilitating viewing with simple 2D image viewers.

On startup the application prompts the user to type in:

1. the filename of the input volume
2. specify the endian-ness of the file

3. the size of the input volume
4. the directory where the PGM files are to be written (NB the directory must already exist otherwise no images are written out)
5. the initial seed index position
6. the initial threshold value

An initial segmentation is performed and the results are written to the specified directory. The application then goes into command mode. There are four valid commands:

| Command | Description |
|--------------------------|---|
| s <i>xx yy zz</i> | Sets the seed to index (xx,yy,zz) in column-row-slice order. Segmentation is then performed using this new seed and existing threshold value. The results are written to the specified directory. |
| t <i>ww</i> | Sets the threshold value to ww. Segmentation is then performed using the existing seed and new threshold value. The results are written to the specified directory. |
| d | Display the current seed and threshold values. |
| x | Exit the application. |

What components of ITK does this application use?

This application makes use of `itk::FuzzyConnectednessImageFilter`.

Example run:

This is an example using dataset in the `InsightData` repository. Start the application as follows:

```
F:\Insight\Insight-VC++\Examples\FuzzyConnectedness> FuzzyConnectedness
```

The application then prompts the users to provide input and output information:

```
Input file name: n:\upenn\tumor\4121\raw\tumor1.flair
Input image big endian? [y|n]: y
Input image size: 256 256 30
PGM output directory: pgms
Writing PGM files of the input volume.
```

Each 2D slice from the input volume is written out as PGM files into directory `pgms`. The output files have prefix `input` followed by the slice number and then extension `.pgm`. Fig 1. shows slice 14 of the input volume for this example. The application then prompts for the initial seed and threshold values:

```
Set initial seed index: 151 112 13
Set initial threshold value: 0.8
Generating the fuzzy connectedness map.
```

Segmentation is then performed using the initial seed and threshold values. Each 2D slice from the computed segmentation mask is written out as PGM files into directory `pgms`.

The output files have prefix `seg` followed by the slice number and then extension `.pgm`. Fig 2. shows slice 14 of the segmentation mask for this example.

The user can then interactively change the seed point and/or threshold value. The following command changes the threshold to 0.5.

```
Command [s|t|d|x]: t 0.5  
Re-thresholding the map.
```

The segmentation mask is recomputed. Fig. 3 shows the effect of reducing the threshold value. Command 'd' display the current seed and threshold values and 'x' exits the application.

```
Command [s|t|d|x]: d  
Seed: [151, 112, 13]  
Threshold: 0.5
```

```
Command [s|t|d|x]: x  
Goodbye.
```

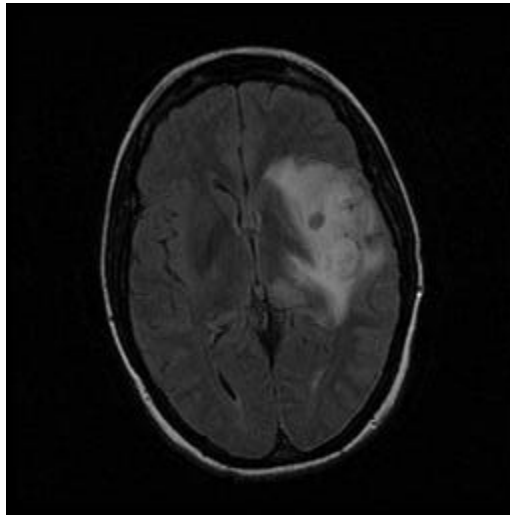


Fig 1: Slice 14 of input volume (pgms/input014.pgm)

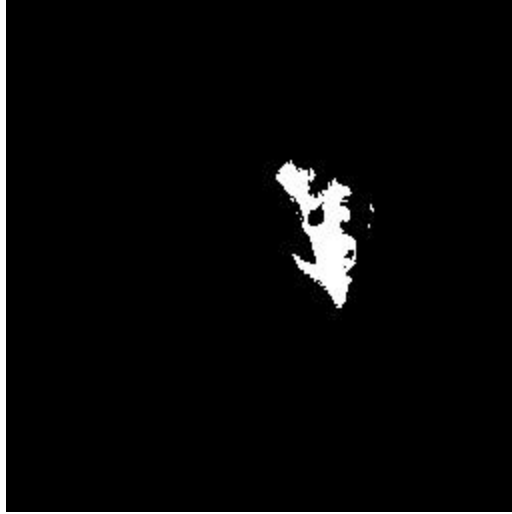


Fig 2: Slice 14 of segmentation mask with seed (151,112,13) and threshold of 0.8
(pgms/seg014. pgm)

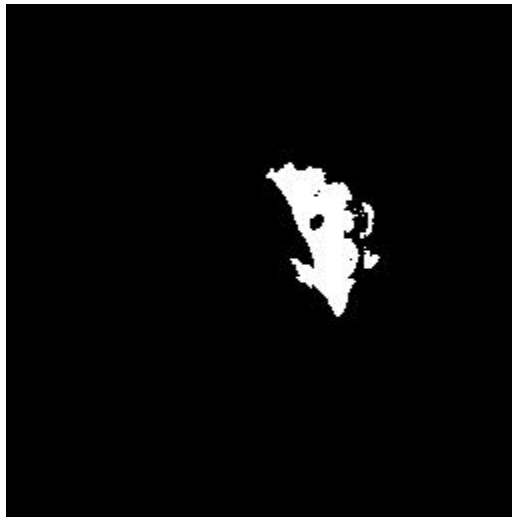


Fig 3: Slice 14 of segmentation mask with seed (151,112,13) and threshold of 0.5
(pgms/seg014. pgm)