A patch-based framework for new ITK functionality: Joint fusion, denoising, and non-local super-resolution

Release 0.00

Nick Tustison[1] Brian Avants[2], Long Xie[3], Paul Yushkevich[3], and Jose Vicente Manjón[4]

March 9, 2017

[1]University of Virginia, Charlottesville, VA
[2]Biogen, Cambridge, MA
[3]University of Pennsylvania, Philadelphia, PA
[4]ITACA institute, Universidad Politécnica de Valencia, Valencia, Spain

Abstract

In an earlier Insight Journal article, we introduced an ITK implementation of the adaptive patch-based image denoising algorithm described in [3]. We follow-up up that offering with a generalized non-local, patch-based ITK class framework and a refactored denoising class. In addition, we provide two additional ITK implementations of related, well-known algorithms. The first is a non-local super resolution method described in [1, 2]. The second is the multivariate joint label fusion algorithm of [4, 5]. Accompanying these ITK classes are documented programming interfaces which use our previously introduced unique command line interface routines. Several 2-D examples on brain imaging data are provided to qualitatively demonstrate performance.

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

1 Introduction

Image filtering techniques are usually applied to increase the quality of MR images. Most of these techniques assume an equal noise distribution across the image. When this assumption is not met, the resulting filtering becomes suboptimal. This is the case of MR images with spatially varying noise levels, such as those obtained by parallel imaging (sensitivity-encoded), intensity inhomogeneity-corrected images or surface coil based acquisitions. In this work, we have adapted a recently proposed filter so-called Non-Local Means (NL-means) to deal with MR images with spatially varying noise levels (for both Gaussian and Rician distributed noise). With this new method, information regarding the local image noise level is used to adjust the amount of denoising strength of the filter. Such information is automatically obtained from the images by using the local noise estimation method described in [3].

This method has been validated and compared with the standard NL-means filter on simulated and real MR imaging data. This noise-adaptive NL-means method was demonstrated to outperform the standard NL-means based filter when spatially varying noise is present in the images. Although source code is available from Prof. Manjón Herrera's website (http://personales.upv.es/jmanjon/denoising/arnlm.html). This particular algorithm is currently not available in the Insight Toolkit which provides good performance with minimal parameter tuning. In this Insight Journal submission, we provide a brief overview of the ITK implementation and provide a couple examples.

2 ITK Implementation

The following files define the adaptive NL-means algorithm and is written in ITK coding style:

- itkAdaptiveNonLocalMeansDenoisingImageFilter.h
- itkAdaptiveNonLocalMeansDenoisingImageFilter.hxx

The following helper class

- itkVarianceImageFilter.h
- itkVarianceImageFilter.hxx

is the variance analog to the `itk::MeanImageFilter` which "Computes an image where a given pixel is the [variance] value of the the pixels in a neighborhood about the corresponding input pixel." Both the adaptive and variance filters are multi-threaded.

Additionally, we wrote an easy-to-use interface program, DenoiseImage.cxx, with an ANTs-based command line interface facilitated by the following helper classes:

- antsCommandLineOption.h
- antsCommandLineOption.cxx
- antsCommandLineParser.h
- antsCommandLineParser.cxx

These latter files are not essential for other applications or for inclusion into ITK. They are included simply to facilitate the generation of a stand-alone application that people can use. One can invoke the help menu via `$ DenoiseImage --help 1` at the command line:

```
$ DenoiseImage --help 1

COMMAND:
     DenoiseImage
          Denoise an image using a spatially adaptive filter originally described in J. V.
          Manjon, P. Coupe, Luis Marti-Bonmati, D. L. Collins, and M. Robles. Adaptive
          Non-Local Means Denoising of MR Images With Spatially Varying Noise Levels,
          Journal of Magnetic Resonance Imaging, 31:192-203, June 2010.
```

```
OPTIONS:
      -d, --image-dimensionality 2/3/4
            This option forces the image to be treated as a specified-dimensional image. If
            not specified, the program tries to infer the dimensionality from the input
            image.

      -i, --input-image inputImageFilename
            A scalar image is expected as input for noise correction.

      -n, --noise-model Rician/(Gaussian)
            Employ a Rician or Gaussian noise model.

      -x, --mask-image maskImageFilename
            If a mask image is specified, denoising is only performed in the mask region.

      -s, --shrink-factor (1)/2/3/...
            Running noise correction on large images can be time consuming. To lessen
            computation time, the input image can be resampled. The shrink factor, specified
            as a single integer, describes this resampling. Shrink factor = 1 is the
            default.

      -p, --patch-radius 1
                           1x1x1
            Patch radius. Default = 1x1x1

      -r, --search-radius 3
                            3x3x3
            Search radius. Default = 3x3x3.

      -o, --output correctedImage
                     [correctedImage,<noiseImage>]
            The output consists of the noise corrected version of the input image.
            Optionally, one can also output the estimated noise image.

      -v, --verbose (0)/1
            Verbose output.

      -h
            Print the help menu (short version).

      --help
            Print the help menu.
```

which should provide minimal detail regarding each parameter. The default parameters should provide adequate performance for a variety of cases thus requiring only the simple call (for a 3-D image)

```
$ DenoiseImage -d 3 -i ${inputImage} -o ${outputImage}
```

As mentioned, this filter is multi-threaded so one might want to set the environment variable to enable more than one thread, e.g.,

```
$ export ITK_GLOBAL_DEFAULT_NUMBER_OF_THREADS=4
```

To see the progress of the filter, one also needs to set the verbose flag, i.e., `-v 1`.

3 Examples

We included a 2-D brain slice and a 3-D hyperpolarized gas volume each with a corresponding mask. Below we include the call and resulting output for these two sets of data.

3.1 T1-weighted MRI brain slice

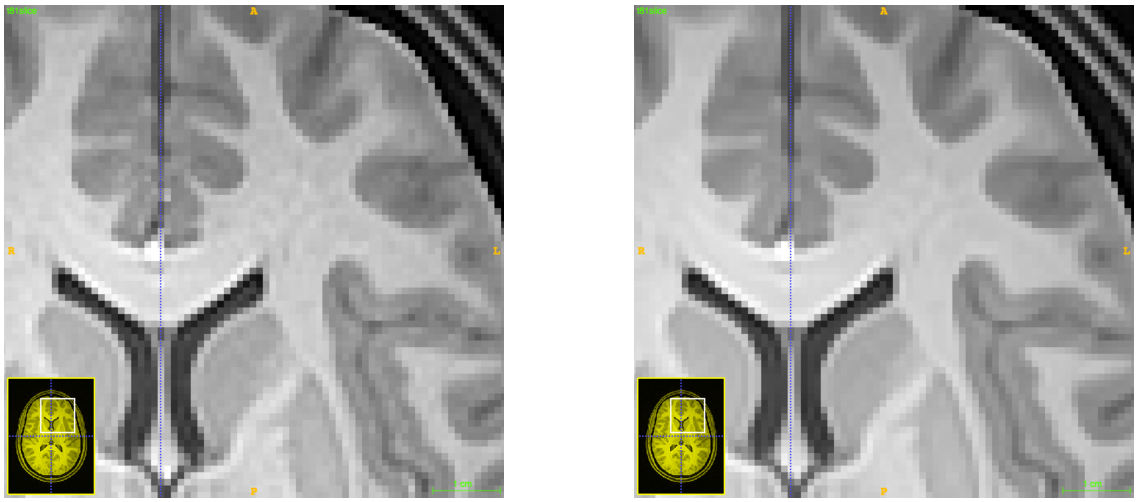
```
$ DenoiseImage -d 2 -i t81slice.nii.gz -x t81mask.nii.gz -o t81denoised.nii.gz -v 1

Running for 2-dimensional images.

*********10*********20*********30*********40*********50*********60*********70*********80*********90*********

    AdaptiveNonLocalMeansDenoisingImageFilter (0x7f9733997080)
      RTTI typeinfo:   itk::AdaptiveNonLocalMeansDenoisingImageFilter<itk::Image<float, 2u>, itk::Image<float, 2u>, itk
      Reference Count: 1
      Modified Time: 517
      Debug: Off
      Object Name:
      Observers:
        ProgressEvent (Command)
      Inputs:
        Primary: (0x7f9733996c10) *
        _1: (0x7f97339988f0)
      Indexed Inputs:
        0: Primary (0x7f9733996c10)
        1: _1 (0x7f97339988f0)
      Required Input Names: Primary
      NumberOfRequiredInputs: 1
      Outputs:
        Primary: (0x7f9733996a50)
      Indexed Outputs:
        0: Primary (0x7f9733996a50)
      NumberOfRequiredOutputs: 1
      Number Of Threads: 1
      ReleaseDataFlag: Off
      ReleaseDataBeforeUpdateFlag: Off
      AbortGenerateData: Off
      Progress: 1
      Multithreader:
        RTTI typeinfo:   itk::MultiThreader
        Reference Count: 1
        Modified Time: 299
        Debug: Off
        Object Name:
        Observers:
          none
        Thread Count: 1
        Global Maximum Number Of Threads: 128
        Global Default Number Of Threads: 1
      CoordinateTolerance: 1e-06
      DirectionTolerance: 1e-06
```

```
      Using Gaussian noise model.
      Epsilon = 1e-05
      Mean threshold = 0.95
      Variance threshold = 0.5
      Smoothing variance = 2
      Neighborhood radius for local mean and variance = [1, 1]
      Neighborhood search radius  = [3, 3]
      Neighborhood block radius = [1, 1]
Elapsed time: 0.0831451
```



(a) Original (b) Denoised

Figure 1: Effects of the NL-means filter on a 2-D T1-weighted MRI brain slice.

3.2 Hyperpolarized He3 Lung MRI volume

```
$ DenoiseImage -d 3 -i he3volume.nii.gz -x he3mask.nii.gz -o he3denoised.nii.gz -v 1

Running for 3-dimensional images.

*********10*********20*********30*********40*********50*********60*********70*********80*********90*********

   AdaptiveNonLocalMeansDenoisingImageFilter (0x7ff279e0a440)
     RTTI typeinfo:   itk::AdaptiveNonLocalMeansDenoisingImageFilter<itk::Image<float, 3u>, itk::Image<float, 3u>, itk
     Reference Count: 1
     Modified Time: 506
     Debug: Off
     Object Name:
     Observers:
       ProgressEvent (Command)
     Inputs:
       Primary: (0x7ff27a97ea20) *
       _1: (0x7ff27a97f040)
     Indexed Inputs:
       0: Primary (0x7ff27a97ea20)
       1: _1 (0x7ff27a97f040)
```

```
      Required Input Names: Primary
      NumberOfRequiredInputs: 1
      Outputs:
        Primary: (0x7ff279e0a710)
      Indexed Outputs:
        0: Primary (0x7ff279e0a710)
      NumberOfRequiredOutputs: 1
      Number Of Threads: 4
      ReleaseDataFlag: Off
      ReleaseDataBeforeUpdateFlag: Off
      AbortGenerateData: Off
      Progress: 1
      Multithreader:
        RTTI typeinfo:   itk::MultiThreader
        Reference Count: 1
        Modified Time: 294
        Debug: Off
        Object Name:
        Observers:
          none
        Thread Count: 4
        Global Maximum Number Of Threads: 128
        Global Default Number Of Threads: 4
      CoordinateTolerance: 1e-06
      DirectionTolerance: 1e-06
      Using Gaussian noise model.
      Epsilon = 1e-05
      Mean threshold = 0.95
      Variance threshold = 0.5
      Smoothing variance = 2
      Neighborhood radius for local mean and variance = [1, 1, 1]
      Neighborhood search radius  = [3, 3, 3]
      Neighborhood block radius = [1, 1, 1]
Elapsed time: 1.7134
```

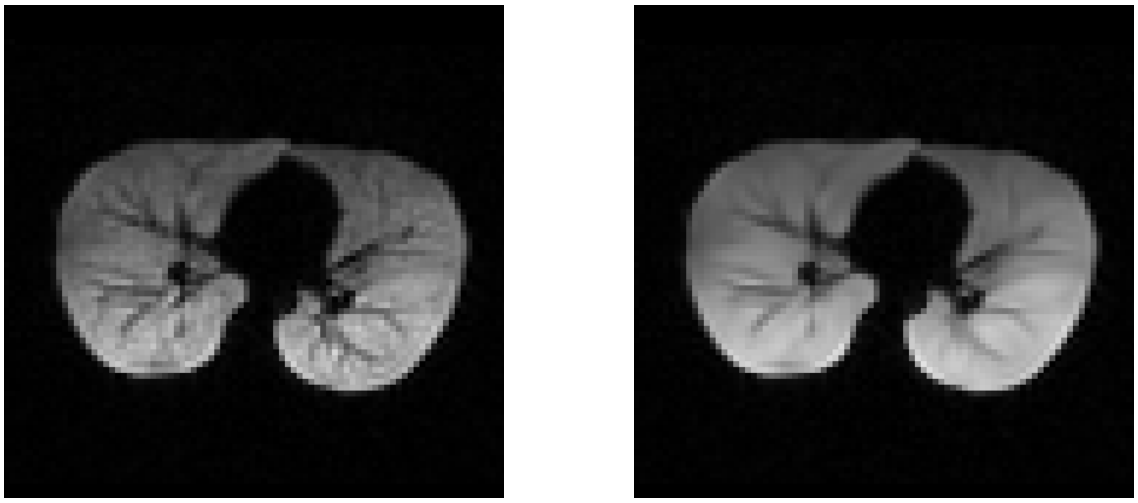
References

[1] José V Manjón, Pierrick Coupé, Antonio Buades, D Louis Collins, and Montserrat Robles. Mri super-resolution using self-similarity and image priors. *Int J Biomed Imaging*, 2010:425891, 2010. (document)

[2] José V Manjón, Pierrick Coupé, Antonio Buades, Vladimir Fonov, D Louis Collins, and Montserrat Robles. Non-local mri upsampling. *Med Image Anal*, 14(6):784–92, Dec 2010. (document)

[3] José V Manjón, Pierrick Coupé, Luis Martí-Bonmatí, D Louis Collins, and Montserrat Robles. Adaptive non-local means denoising of mr images with spatially varying noise levels. *J Magn Reson Imaging*, 31(1):192–203, Jan 2010. (document), 1

[4] Hongzhi Wang, Jung W Suh, Sandhitsu R Das, John B Pluta, Caryne Craige, and Paul A Yushkevich. Multi-atlas segmentation with joint label fusion. *IEEE Trans Pattern Anal Mach Intell*, 35(3):611–23, Mar 2013. (document)



(a) Original (b) Denoised

Figure 2: Effects of the NL-means filter on a 3-D hyperpolarized gas MRI lung volume.

[5] Hongzhi Wang and Paul A Yushkevich. Multi-atlas segmentation with joint label fusion and corrective learning-an open source implementation. *Front Neuroinform*, 7:27, 2013. (document)