# Automatic Conductance Estimation Methods for Anisotropic Diffusion ITK Filters

*Release 0.1*

Antonio Carlos da S. Senra Filho[1]

May 3, 2017

[1]University of Sao Paulo, Brazil

**Abstract**

The anisotropic diffusion algorithm has been intensively studied in the past decades, which could be considered as a very efficient image denoising procedure in many biomedical applications. Several authors contributed many clever solutions for diffusion parameters fitting in specific imaging modalities. Furthermore, besides improvements regarding the image denoising quality, one important variable that must be carefully set is the conductance, which regulates the structural edges preservation among the objects presented in the image. The conductance value is strongly dependent on image noise level and an appropriate parameter setting is, usually, difficult to find for different images databases and modalities. Fortunately, thanks to many efforts from the scientific community, a few automatic methods have been proposed in order to set the conductance value automatically. Here, it is presented an ITK class which offers a simple collection of the most common automatic conductance setting approaches in order to assist researchers in image denoising procedures using anisotropic-based filtering methods (such as well described in the `itk::AnisotropicDiffusionFunction` class).



## Contents

## 1 Introduction

The anisotropic diffusion algorithm, firstly proposed by Perona and Malik [7], has been intensively studied in the past decades, which could be considered as a very efficient image denoising procedure in many biomedical applications [5, 8, 1, 12]. Following the initial proposal, several authors contributed many clever solutions for diffusion parameters fitting in specific imaging modalities, being magnetic resoncane imaging (MRI) one common imaging technique that has been greatly enhanced by this image filtering method [3, 6, 11, 1, 4]. The ITK library already offer a wide family of anisotropic diffusion based image filtering methods, being well presented in the `itk::AnisotropicDiffusionFunction` class.

Furthermore, besides improvements regarding the image denoising quality, other issues were also investigated such as the automatic diffusion variables adjustments approaches. One important variable that must be carefully set is the conductance (usually denoted as $\kappa$), which regulates the structural edges preservation among the objects presented in the image [7]. As discussed in the original paper [7] and other studies [10, 1, 12], the $\kappa$ value is strongly dependent on image noise level and an appropriate parameter setting is, usually, difficult to find for different images databases and modalities. In general, a manual parameter setting is adopted, as presented in the `itk::GradientAnisotropicDiffusionImageFilter` and `itk::VectorGradientAnisotropicDiffusionImageFilter` documentation, which creates practical difficulties for optimum results using anisotropic diffusion based algorithms.

Fortunately, thanks to many efforts from the scientific community, a few automatic methods have been proposed in order to set the $\kappa$ value automatically, using image features such as local variation and morphological indexes [9, 10, 1, 7]. Here, it is presented an ITK class in order to offer a simple application of the most common automatic conductance setting approaches. Firstly, a brief explanation of the anisotropic diffusion filtering procedure is presented in section 2 (but many more details could be found in [7, 1, 10]. Secondly, the methods adopted in the `AutomaticConductanceImageCalculator` ITK class are presented in section 3 and, finally, a short example is illustrated in section 4 in order to facilitates the usage of this ITK class.

## 2 Brief presentation of the anisotropic diffusion based methods

In summary, the diffusion equation could be represented by Equation (1), being a general formulation of partial differential equations that describes the classical diffusion regime in physical and engineering problems.

$$\frac{\partial I(\overrightarrow{r},t)}{\partial t} = \nabla.\Big(D.\nabla I(\overrightarrow{r},t)\Big) \tag{1}$$

Where $D$ is the diffusion coefficient that, in the case of anisotropic regime, it can be adopted as a spatial function, i.e. $D(\overrightarrow{r})$. The diffusion coefficient is responsible for regulating the diffusion intensity through the objects borders of the image, where a high value of $D(\overrightarrow{r})$ means that the diffusion process is strong (providing blurring) and, otherwise, the local diffusion is restrict (preserving objects edges). An example of a possible function that modulates the $D(\overrightarrow{r})$ parameters is given in Equation (2), being commonly applied in `itk::GradientAnisotropicDiffusionImageFilter` class. Many other functions could be adopted in order to modulate the $D(\overrightarrow{r})$ parameters [9, 7, 1, 10], however, in general, those functions also depend in $\kappa$ parameter.

$$D(\overrightarrow{r}) = e^{-(\frac{|\nabla I(\overrightarrow{r},t)|}{\kappa})^2} \tag{2}$$

As seen in Equation (2), the $D(\overrightarrow{r})$ parameters depends of the proper adjustment of $\kappa$, i.e. the conductance parameters. Usually, the $\kappa$ parameter is manually adjusted, being values between 0.5 to 2.0 commonly adopted as a reasonable range to be used in the majority of situations (as seen in `itk::AnisotropicDiffusionFunction` documentation). However, the choice of $\kappa$ value is often not straightforward, depending of the type of data that is inserted. In general, a manual $\kappa$ adjusting returns suboptimal results, which creates practical limitations in the final image filtering quality. For this reason, several studies were developed in the previous years in order to implement automatic conductance adjusting methods, being interesting to be added in the ITK library.

The code presented here is intended to offer a few of these automatic conductance adjusting methods, where it could be applied in the majority of the `itk::AnisotropicDiffusionFunction` classes (scalar and vector). The goal here is offer a simple tool to assist researcher for reducing the manual interaction in image denoising procedures.

## 3 Automatic approaches for conductance parameter adjustment

In general, the methods adopted here are based on image features extract at the initial time step, i.e. $I(\overrightarrow{r},0)$, which represents the highest noise level image in the image filtering iterative pipeline. The following methods are also well discussed in [9, 10]. Figure 1 illustrates some examples using the methods implemented in this ITK class.

### 3.1 Canny Noise Estimator

This method was proposed in the original anisotropic diffusion filtering paper, i.e. by Perona and Malik [7]. In this method, the histogram of absolute gradient values is calculated, where the $\kappa$ value is estimated as the 90% value of the integral of the gradient histogram. This strategy was previously described as a noise estimator method in [2], being considered as a classical image edge detection approach. A definition of this process is given in Equation (3)

$$\kappa = \int_{(M.N)} H[|\nabla I(\overrightarrow{r},t)|].d\overrightarrow{r} \tag{3}$$

### 3.2 Median Absolute Deviation

This formulation was proposed by Black et al. [1] which is based on Tukey's biweight robust estimator that preserves sharper boundaries than previous formulations and improves the automatic stopping of the diffusion. In summary, this method has been discussed as an efficient calculation for determining a robust statistic in order to avoid outliers influence in the conductance parameter estimation. In other words, using the gradient of input image, it is possible to infer an objective statistical representation that avoids global ”edge outliers” (gradient intensity extrema) in the conductance inference. A simple representation of this method is given in Equation (4), where *MAD* stands for Median Absolute Deviation.

$$\kappa = 1.4826.MAD(\nabla I(\overrightarrow{r},t)) = 1.4826.\text{median}_I[|\nabla I(\overrightarrow{r},t) - \text{median}_I(|\nabla I(\overrightarrow{r},t)|)|] \tag{4}$$

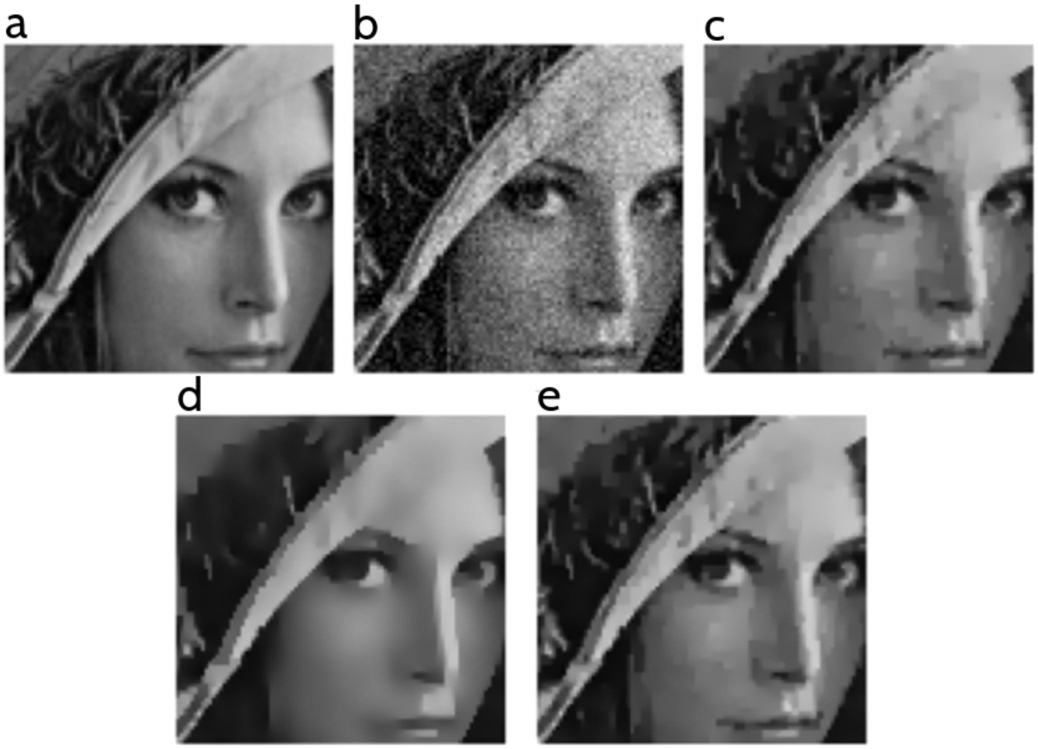


Figure 1: Some zoomed images to illustrate the effect of each automatic conductance methods. a) A zoomed part o the original Lena image. b) A noisy version of the image in a) with Gaussian noise with $\sigma = 0.07$. The output of the scalar gradient anisotropic diffusion filter ( itk::GradientAnisotropicDiffusionImageFilter) using 15 iterations and the c) Canny Noise Estimator (section 3.1), d) Median Absolute Variation (section 3.2) and e) Morphological Feature Estimation (section 3.3) methods. Figure adapted from [9]

Using this strategy, the authors concludes that the MAD method converges to a stable smoothing solution in a fast and precise way [1].

### 3.3 Morphological Feature Estimation

The idea of using a morphological approach to estimate the $\kappa$ parameter derives from the fact that morphology can be used for an estimation of noise intensity in the image. For this purpose, Voci et al. [10] considered the difference openingclosing morphological methods as a conductance quantification function. Furthermore, the $\kappa$ value should be at least the same as the averaged noise amplitude value [10]. Using those concepts, the Equation (5) can be defined, where the image $I(\overrightarrow{r})$, with $m$ lines and $n$ columns, $\Theta$ as a isotropic morphological kernel (defined as a $3\times3\times3$, for a 3D image) $\circ$ and $\bullet$ as the opening and closing morphological operators, respectively.

$$\kappa = \sum_{\overrightarrow{r}\in I} \frac{(I(\overrightarrow{r})\circ\Theta)}{(m\times n)} - \sum_{\overrightarrow{r}\in I} \frac{(I(\overrightarrow{r})\bullet\Theta)}{(m\times n)} \tag{5}$$

## 4 Example

The following example illustrates a simple usage of the AutomaticConductanceImageCalculator class. Initially, we start declaring the basic classes that will be used for data reading and processing

```
#include "itkImage.h"
#include "itkImageFileReader.h"
#include "itkAutomaticConductanceImageCalculator.h"
```

After that, the main() function could be initiated

```
int main(int argc, char* argv[])
{
    if ( argc < 3 )
    {
        std::cerr << "Missing parameters. " << std::endl;
        std::cerr << "Usage: " << std::endl;
        std::cerr << argv[0]
                << " inputImageFileName optimizationFunction (1,2,3)"
                << std::endl;
        return -1;
    }
    const unsigned int Dimension = 3;

    typedef float                       PixelType;
    typedef itk::Image<PixelType, Dimension>    InputImageType;
    typedef itk::Image<PixelType, Dimension>    OutputImageType;

    typedef itk::ImageFileReader<InputImageType> ReaderType;

    ReaderType::Pointer reader = ReaderType::New();
    reader->SetFileName(argv[1]);
    reader->Update();
```

In this example, the insertion of three parameters is needed in order to run the code, where the first parameter is regarding the input image, the second one is the automatic approach listed in section 3 and, finally, the last parameter is just a number selection for a simple switch-case C++ code[1].

After the usual data reading process, it could be inserted the automatic conductance estimation class declaration, as follow

```
    typedef itk::AutomaticConductanceImageCalculator<InputImageType>  CalculatorType;
    CalculatorType::Pointer calculator = CalculatorType::New();
    calculator->SetImage(reader->GetOutput());
    int optFunction = atoi(argv[2]);
    switch (optFunction) {
        case 1:
           calculator->SetOptimizationMethod(CalculatorType::CANNY);
```

[1]this last parameters is not linked with the AutomaticConductanceImageCalculator internal process and it is only used for this example

```
            break;
        case 2:
            calculator->SetOptimizationMethod(CalculatorType::MAD);
            break;
        case 3:
            calculator->SetOptimizationMethod(CalculatorType::MORPHOLOGICAL);
            break;
    default:
        break;
    }
```

Hence, after the input parameters being added in the `calculator` object (by `SetImage()` and `SetOptimizationMethod()` functions), the automatic conductance algorithm can be called by the following statement

```
    std::cout<<"Kappa: "<<calculator->GetKappa()<<std::endl;
```

## References


[1] M J Black, G Sapiro, D H Marimont, D Heeger, and D Michael J. Black. Robust Anisotropic Diffusion. *IEEE Trans. Image Process.*, 7(3):421, jan 1998. 1, 2, 3.2, 3.2

[2] J Canny. A computational approach to edge detection. *IEEE Pattern Analysis and Machine Intelligence*, (6), 1986. 3.1

[3] Antonio Carlos da Silva Senra Filho, Luiz Otávio Murta Junior, and Antonio Carlos dos Santos. Anisotropic Anomalous Filter Applied to Multimodal Magnetic Resonance Image in Multiple Sclerosis. In *I Transatlantic Workshop on Methods for Multimodal Neurosciences Studies*, São Pedro, SP, 2014. 1

[4] Ron Kikinis. Guido Gerig and Ferenc A Jolesz. Nonlinear Anisotropic Filtering of MRI Data. *IEEE Trans. Med. Imag.*, 11(2):221, 1992. 1

[5] Mohammad Reza Hajiaboli. An Anisotropic Fourth-Order Diffusion Filter for Image Noise Removal. *International Journal of Computer Vision*, 92(2):177–191, mar 2010. 1

[6] Karl Krissian and Santiago Aja-Fernàndez. Noise-driven anisotropic diffusion filtering of MRI. *IEEE transactions on image processing*, 18(10):2265–2274, oct 2009. 1

[7] P Perona and J Malik. Scale-space and edge detection using anisotropic diffusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(7):629–639, jul 1990. 1, 2, 3.1

[8] V B Pires and C A Z Barcelos. Edge Detection of Skin Lesions Using Anisotropic Diffusion. *Seventh International Conference on Intelligent Systems Design and Applications ISDA 2007*, 0:363–370, 2007. 1

[9] Chourmouzios Tsiotsios and Maria Petrou. On the choice of the parameters for anisotropic diffusion in image processing. *Pattern Recognition*, 46(5):1369–1381, may 2013. 1, 2, 3, 1

[10] Francesco Voci, Shigeru Eiho, N. Sugimoto, and H. Sekiguchi. Estimating the gradient threshold in the perona-malik equation. *IEEE Signal Processing Magazine*, 23(3):39–46, may 2004. 1, 2, 3, 3.3

[11] Qing Xu, Adam W Anderson, John C Gore, and Zhaohua Ding. Efficient anisotropic filtering of diffusion tensor images. *Magnetic resonance imaging*, 28(2):200–11, feb 2010. 1

[12] Y L You, W Xu, A Tannenbaum, and M Kaveh. Behavioral analysis of anisotropic diffusion in image processing. *IEEE transactions on image processing : a publication of the IEEE Signal Processing Society*, 5(11):1539–53, jan 1996. 1