An ITK implementation of the Normalized Gradient Field Image to Image Metric

Release 0.00

Gert Wollny, Maria J. Ledesma-Carbayo, and Andres Santos

February 5, 2010

Ciber BBN, Spain;
Biomedical Imaging Technologies, Department of Electronic Engineering, ETSIT, Universidad Politcnica de Madrid, Spain

Abstract

This article describes the ITK implementation of a *Normalized Gradient Fields* (NGF) based image-to-image metric. Some properties of the metric are discussed and example registrations are presented.

Contents

1 Introduction

In an image registration framework, the choice of the right image similarity measure is important to achieve a good image registration. Image similarity measures like the *Sum of Squared Differences* (SSD), *Cross Correlation*, and *(Normalized) Mutual Information* [2, 3], are global measures that best describe the similarity of the image content, if the material-intensity mapping is consistent over the whole image domain. In image series, where the material-intensity mapping changes locally over time, like e.g., in perfusion magnetic resonance imaging, these measures may not perform well. Here, a similarity measure that draws its value only from a local vicinity of each pixel may be more appropriated.

2 A normalized gradient fields based image to image metric

Haber and Modersitzki [1] proposed *Normalized Gradient Fields* NGF as an alternative image registration metric: Given a domain Ω, an intensity range $\mathbb{V}$, an image as a mapping $I : \Omega \to \mathbb{V}$ and its noise level η , a measure ε for boundary "jumps" (locations with a high gradient) can be defined as

$$\varepsilon := \eta \frac{\int_\Omega |\nabla I(\mathbf{x})| d\mathbf{x}}{\int_\Omega d\mathbf{x}}, \tag{1}$$

and with

$$\|\nabla I(\mathbf{x})\|_\varepsilon := \sqrt{\sum_{i=1}^{d} (\nabla I(\mathbf{x}))_i^2 + \varepsilon^2}, \tag{2}$$

the NGF of an image I is defined as follows:

$$\mathbf{n}_\varepsilon(I, \mathbf{x}) := \frac{\nabla I(\mathbf{x})}{\|\nabla I(\mathbf{x})\|_\varepsilon}. \tag{3}$$

NGF based similarity measures for the image registration of a test image S to a reference image R have been formulated using kernels based either on the scalar product $\langle \cdot, \cdot \rangle$ or the cross product $\cdot \times \cdot$ of the vectors of the NGF [1]:

$$F_{\text{NGF}}^{(\cdot)}(S,R) := -\frac{1}{2}\int_\Omega \langle \mathbf{n}_\varepsilon(R), \mathbf{n}_\varepsilon(S)\rangle^2 d\mathbf{x} \tag{4}$$

$$F_{\text{NGF}}^{(\times)}(S,R) := \frac{1}{2}\int_\Omega \|\mathbf{n}_\varepsilon(R,\mathbf{x}) \times \mathbf{n}_\varepsilon(S,\mathbf{x})\|^2 d\mathbf{x} \tag{5}$$

However, both similarity measures exhibit problems when it comes to their application: Even though the gradient of the scalar product based cost function $F_{\text{NGF}}^{(\cdot)}$ (4) is analytically zero at the minimum, for practical implementations of the gradient evaluation, like e.g., finite differences, the gradient evaluates to non-zero values (i.e. even if $S = R$), thus making the optimization using gradient based methods difficult. On the other hand, when using the cross product based version (5), $F_{\text{NGF}}^{(\times)}(\mathbf{x})$ is not only zero when $\mathbf{n}_\varepsilon(R,\mathbf{x})(\mathbf{x}) \parallel \mathbf{n}_\varepsilon(S,\mathbf{x})(\mathbf{x})$ (as desired), but also when either $\mathbf{n}_\varepsilon(R,\mathbf{x})$, or $\mathbf{n}_\varepsilon(S,\mathbf{x})$ have zero norm.

Therefore, we tested other evaluators, specifically:

$$F_{\text{NGF}}^{(\Delta\cdot)}(S,R) := \frac{1}{2}\int_\Omega \left(\|\mathbf{n}_\varepsilon(R)\|^2 - \frac{\langle \mathbf{n}_\varepsilon(R), \mathbf{n}_\varepsilon(S)\rangle^2}{\|\mathbf{n}_\varepsilon(R)\| \|\mathbf{n}_\varepsilon(S)\|} \right)^2 d\mathbf{x}, \tag{6}$$

$$F_{\mathrm{NGF}}^{(\Delta)}(S,R) := \frac{1}{2}\int_{\Omega} \langle \mathbf{n}_{\varepsilon}(R) - \mathbf{n}_{\varepsilon}(S), \mathbf{n}_{\varepsilon}(R)\rangle^2, \tag{7}$$

and

$$F_{\mathrm{NGF}}^{(\Delta^2)}(S,R) := \frac{1}{2}\int_{\Omega} \left(\langle \mathbf{n}_{\varepsilon}(R), \mathbf{n}_{\varepsilon}(R)\rangle - \langle \mathbf{n}_{\varepsilon}(S), \mathbf{n}_{\varepsilon}(S)\rangle\right)^2 \tag{8}$$

$F_{\mathrm{NGF}}^{(\Delta\cdot)}$ is always differentiable and its evaluation as well as the evaluation of its derivatives are straightforward, making it easy to use for image registration. $F_{\mathrm{NGF}}^{(\Delta\cdot)}(S,R)|_{\mathbf{x}}$ is minimized when $\mathbf{n}_{\varepsilon}(R,\mathbf{x}) \parallel \mathbf{n}_{\varepsilon}(S,\mathbf{x})$. In the optimal case, $S = R$ the cost function and its first order derivatives are zero, and their evaluation is numerically stable. However, in homogeneous areas of the reference image, where $\mathbf{n}_{\varepsilon}(R,\mathbf{x})$ has zero norm, $F_{\mathrm{NGF}}(S,R)$ has also a zero value and a zero gradient. Therefore, in these areas the measure does not contribute to the all-over cost measure, making its application difficult in non-rigid registration.

In addition, it as to be noted that $F_{\mathrm{NGF}}^{(\Delta)}$ (7) is only minimized when the gradients in fixed and moving image are parallel *and* point in the same direction, making it not usable in most cases of multi-modal registration, and $F_{\mathrm{NGF}}^{(\Delta^2)}$ (8) is only minimized when both gradients have the same magnitude. Because of these limitations, these measures will be omitted from the further discussion although their implementation is provided for test purposes.

3 Implementation Details

The implementation of the metric has been split into three parts:

- the metric `NormalizedGradientFieldImageToImageMetric` ,
- the metric evaluator kernels
  - $F_{\mathrm{NGF}}^{(\times)}$: `NGFCrossKernel` ,
  - $F_{\mathrm{NGF}}^{(\cdot)}$: `NGFScalarKernel` ,
  - $F_{\mathrm{NGF}}^{(\Delta\cdot)}$: `NGFScaledDeltaKernel` ,
  - $F_{\mathrm{NGF}}^{(\Delta)}$: `NGFDeltaKernel` ,
  - $F_{\mathrm{NGF}}^{(\Delta^2)}$: `NGFDelta2Kernel` ,
- a filter to evaluate the NGF from an image (`ImageToNGFFilter`),
- and a helper function (`GetImageNoise`) to estimate an approximation on the noise present in an image of an image.

3.1 The Metric

The class `NormalizedGradientFieldImageToImageMetric` is the one implementing the metric and its use is straightforward: As it doesn't require additional parameters, it can be used as a drop-in replacement for, e.g., the sum of squared differences without further changes to the code. In its default implementation it will use `NGFScaledDeltaKernel` (6) as the evaluation kernel. However, it is possible to select another kernel for the image metric evaluation by calling `SetEvaluator`.

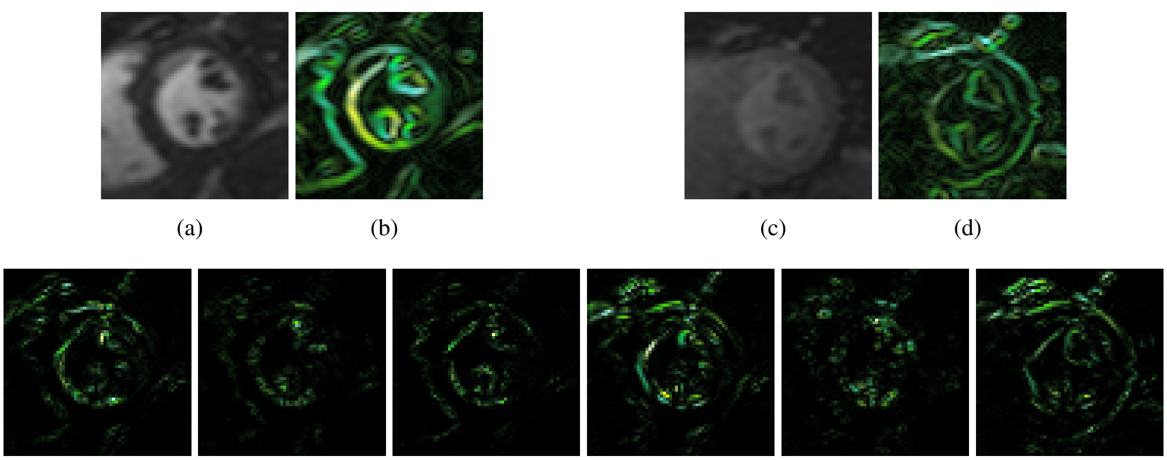
3.2 Metric evaluators

As described in section 2, image-to-image metrics that are based on normalized gradient fields can be defined in various ways by changing the evaluation kernel. The class `NGFKernel` defines the abstract base class for such evaluators. As derivatives of this class, the five evaluators described above are currently implemented. Note, however, that the cross product based evaluator can only be used for two- and three-dimensional images, since otherwise the cross-product is not defined.

3.3 Image to NGF filter

`ImageToNGFFilter` is implemented as a helper class that evaluates the normalized gradient field of an image according to (3). The filter is implemented as a specialization of the `GradientImageFilter` and does not require further initialization. In this case, the image noise η will be estimated by calling `GetImageNoise`, which evaluates the noise as the median intensity of the output image of the filter `NoiseImageFilter`. This automatic noise estimation is done only once during the first call to the filter, and the estimated value is then used in subsequent calls. In order to (re-)trigger the noise estimation, one can set the noise to a negative value using the `SetNoise` method before calling the `Update` method of the filter. Alternatively, setting the noise to a positive value calling `SetNoise` this value is used as a noise estimate no automatic estimation is done.

4 Some metric properties



(a) (b) (c) (d)

(e) $F^{(\cdot)}_{\text{NGF}}$ (f) $F^{(\times)}_{\text{NGF}}$ (g) $F^{(\Delta\cdot)}_{\text{NGF}}$ (h) $F^{(\cdot)}_{\text{NGF}}$ (i) $F^{(\times)}_{\text{NGF}}$ (j) $F^{(\Delta\cdot)}_{\text{NGF}}$

Figure 1: The two images (a) and (c) with their respective normalized gradient fields (b) and (d) are taken from a series of myocardial perfusion images where the contrast changes over time due to the contrast agent passing through. The images (e)-(g) show the gradients when image (a) is used as fixed image, and the images (h)-(j) illustrate the gradient when image (c) is used as fixed image. The image where created with the program `NGFGradientMap.cpp` from the `Examples` directory.

In order to assess some of the properties of this metric, the two images from cardiac perfusion imaging were used (Fig. 1 (a), and (c)). As can be seem from the Fig. 1 (f) and (i), $F^{(\times)}_{\text{NGF}}$ (5) provides less gradient information. $F^{(\cdot)}_{\text{NGF}}$ (4), and $F^{(\Delta\cdot)}_{\text{NGF}}$ (6) on the other hand provide more gradient information, and they should, therefore, perform better in image registration. Additionally, $F_{\text{NGF}}^{(\Delta\cdot)}$ (6) shows and asymmetric behavior: Although Fig. 1 (a) as a reference results in less gradient information then using (c) (Fig. 1 (g) and (j)).

However, as discussed above, $F_{\text{NGF}}^{(\cdot)}$ (4) is unstable considering gradient evaluation: For example, when finite differences are used to evaluate the gradient $\nabla F_{\text{NGF}}^{(\cdot)}(S,S)$ (as it is implemented), the gradient that is analytically zero may evaluate to non-zero in certain regions (Fig. 2 (a)). $F_{\text{NGF}}^{(\times)}$ (5), and $F_{\text{NGF}}^{(\Delta\cdot)}$ (6), on the other hand, are stable in that regard (see Fig. 2 (b) and (c) respectively).

(a) $\nabla F_{\text{NGF}}^{(\cdot)}(S,S)$ (b) $\nabla F_{\text{NGF}}^{(\times)}(S,S)$ (c) $\nabla F_{\text{NGF}}^{(\Delta\cdot)}(S,S)$

Figure 2: Gradients of the cost function at the analytical minimum $R = S$ using Fig. 1(a) as image. Note, that the finite difference implementation of $\nabla F_{\text{NGF}}^{(\cdot)}$ doesn't provide a zero gradient.

5 Testing rigid 2D registration

In the following we present some test results of the rigid registration of a pair of images. As Haber and Modersitzki comment in [1], a Gauss-Newton method would be the natural choice for the optimizer, but since such optimizer is not available in the registration framework of ITK, we settled for the `RegularStepGradientDescentOptimizer`, and used `Rigid2DTransform` as transformation.

The optimization parameters were set to

- optimizer scales = [$\sqrt{n_x^2 + n_y^2}$, 1.0, 1.0] for an image of size $n_x \times n_y$.
- maximum number of iterations = 1000
- Relaxation Factor = 0.8
- step size $\in [\frac{1}{100}, \frac{1}{10}]$ at the lowest multiresolution level, the step size boundaries will be multiplied by 2 with each change to a higher resolution,

Note however, that the ITK implementation of the `RegularStepGradientDescentOptimizer` is highly sensitive to a change of these parameters. Therefore, the results given below are by no means representative but just a proof of concept.

A 3-level multi-resolution scheme was employed, and the program to obtain the registration results is implemented as `MultiResImageRegistration2D.cxx` in the `Examples` directory.

In the first case (Table 1), we selected image Fig. 1 (a) as reference and the Fig. 1 (b) as moving image. As the colored overlay of the registered moving and the fixed image show, all three cost functions perform equally well in this setting and with the given optimization parameters.

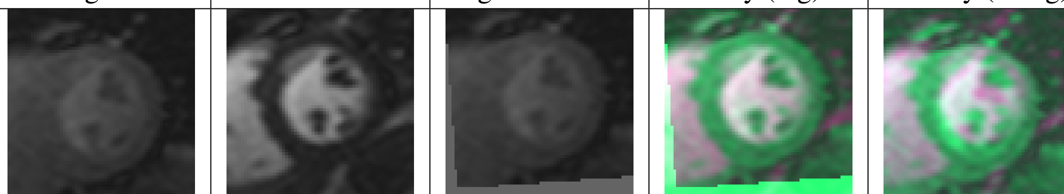
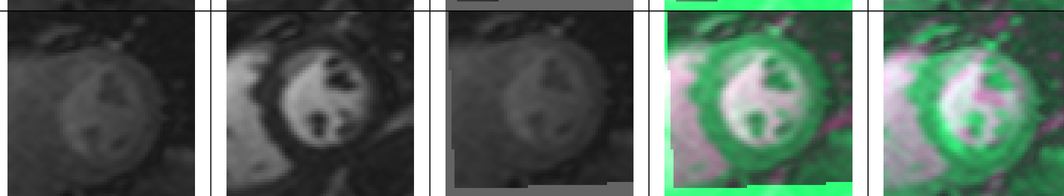
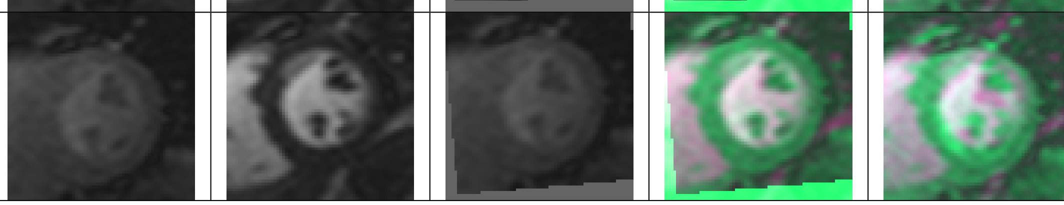
Cost	Moving	Fixed	Registered	Overlay (reg)	Overlay (unreg)
$F_{\text{NGF}}^{(\Delta\cdot)}$					
$F_{\text{NGF}}^{(\cdot)}$					
$F_{\text{NGF}}^{(\times)}$					

Table 1: Rigid registration using different evaluation kernels and the high contrast image as reference.

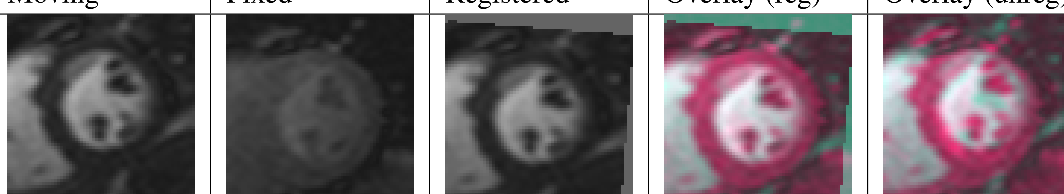
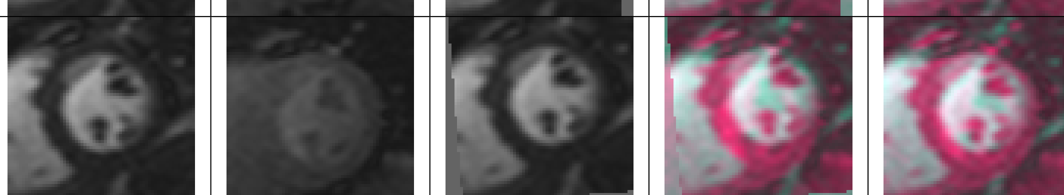
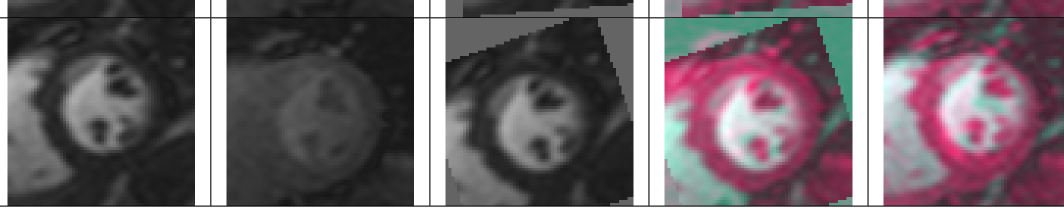
Cost	Moving	Fixed	Registered	Overlay (reg)	Overlay (unreg)
$F_{\text{NGF}}^{(\Delta\cdot)}$					
$F_{\text{NGF}}^{(\cdot)}$					
$F_{\text{NGF}}^{(\times)}$					

Table 2: Rigid registration using different evaluation kernels and the low contrast image as reference.

However, using Fig. 1 (b) image as fixed and the high contrast image as moving image results in a different picture: Applying $F_{\text{NGF}}^{(\Delta\cdot)}$ as registration criterion performs well, but using $F_{\text{NGF}}^{(\times)}$ or $F_{\text{NGF}}^{(\cdot)}$ as evaluators kernels doesn't result in proper registration (Table 2).

6 Software

6.1 Requirements

To only use the cost function and run the registration examples the following software is required:

- Insight Toolkit $\geq$ 3.10,
- CMake $\geq$ 2.6,
- A C++ compiler. It is highly recommended to use a compiler that supports partial template specialization.

For running the tests, one will also need

- BOOST $\geq$ 1.34.1

for its testing framework.

Further software is required to create this document:

- LaTeX
- GNU Make

7 Closing notes

Currently, the image metric does not support masks. In addition, some of the code needs to be reworked to use facilities provided by ITK to support multi-threading of the metric evaluation and properly triggering the `Update` function. Finally, with the current generic implementation using a B-Spline based transformation model results in very long run-times.

Acknowledgment

This study was partially supported by research projects TIN2007-68048-C02-01, CDTI-CDTEAM and SINBAD (PS-010000-2008-1) from Spain's Ministry of Science and Innovation. Image data was provided with the support of the Intramural Research Program of the NIH, National Heart, Lung and Blood Institute.

References

[1] Eldad Haber and Jan Modersitzki. Beyond mutual information: A simple and robust alternative. In Alexander Horsch Hans-Peter Meinzer, Heinz Handels and Thomas Tolxdorff, editors, *Bildverarbeitung für die Medizin 2005*, Informatik Aktuell, pages 350–354. Springer Berlin Heidelberg, March 2005. 2, 2, 5

[2] F. Maes, A. Collignon, D. Meulen, G. Marchal, and P. Suetens. Multi-modality image registration by maximization of mutual information. *IEEE Trans. on Med. Imaging*, 16:187–198, 1997. 1

[3] P. Viola and W. M. Wells III. Alignment by maximization of mutual information. *IJCV*, 24(2):137–154, 1997. 1