An Empirical Optimization to Logistic Classification Model

Release 0.1

Antonio Carlos da S. Senra Filho[1]

September 7, 2019

[1]University of Sao Paulo, Brazil

Abstract

Recently, the scientific community has been proposing several automatic algorithms to biomedical image segmentation procedure, being an interesting and helpful approach to assist both technicians and radiologists in this time-consuming and subjective task. One of these interesting and widely used image segmentation method could be the voxel intensity-based algorithms, e.g. image histogram threshold methods, which have been intensively improved in the past decades. Recently, an interesting approach that gained focus is the logistic classification (LC) for object detection in biomedical images. Even though the general concept behind the LC method is fairly known, the proper method's optimization still commonly adjusted by hand which naturally adds a level of uncertainty and subjectivity in the general segmentation performance. Therefore, an empirical LC optimization is presented, offering a ITK class that performs the LC parameters optimization based on empirical input data analysis. It is worth mentioning that the `LogisticContrastEnhancementImageFilter` class showed here is also applied on others computational problems, being briefly explained in this document.

Contents

1 Introduction

Recently, the scientific community has been proposing several automatic algorithms to biomedical image segmentation procedure, being an interesting and helpful approach to assist both technicians and radiologists in this time-consuming and subjective task. There are many ingenious approaches proposed in the literature which could be highlighted the automatic segmentation techniques based on image patterns analysis [7, 20, 1, 16, 18], statistical models [4, 6, 9] and machine learning approaches [3, 19, 11]. Much more details are presented in recent scientific reviews, being a few examples here suggested [13, 5, 12].

One of the most commonly adopted segmentation strategy is the voxel intensity-based algorithms, e.g. image thresholding, which have been intensively improved in the past decades. Some examples of such technique could be FAST [21] and Atropos [**?**], where both of them recruited a Gaussian Mixture Model (GMM) algorithm in order to define the signal peaks regarding each object of the image. Another interesting approach being raised recently is the usage of logistic classification (LC) for object detection in biomedical images [2]. Hence, grouping both GMM and LC methods, a common conclusion is that the image gray level is an important feature to be explored in object segmentation procedure.

Even though the general concept behind the LC method is fairly known, it proper method's optimization still commonly adjusted by hand which naturally adds a level of uncertainty and subjectivity in the general segmentation performance. Therefore, an empirical LC optimization is presented.

2 The empirical optimization

To classify an object from the background, it must be certified that both entities present a minimum signal difference between them. In other words, a reasonable signal contrast should exist in order to distinguish an object from the surrounding background region. This is usually true in many biomedical imaging analysis, such as hyperintense white matter lesion in Multiple Sclersis (MS) [5], being a common first assumption given to the input data.

Following the previous concept, an object segmentation algorithm based on a LC optimization strategy can be developed. It is worth commenting that the LC optimization is focused on a voxel-wise intensity modulation, applying the original voxel contrast as a "degree of belief" of being a true object. Let $I(r)$ as the digital image with a certain voxel gray level given at the position $r = (x, y, z)$, the LC method can be denoted in Equation (1) (also known as a sigmoid function). In addition, the representation given in Equation (1) is conveniently scaled to $S[I(r)] \in \Re, [0, 1]$, which can be used as a weighted map for further segmentation manipulation.

$$S[I(r)] = \frac{1}{1 + e^{-\frac{I(r)-\beta}{\alpha}}} \tag{1}$$

An interesting characteristic that can be highlighted in the sigmoid function is the possibility to obtain a fuzzy classification [10]. This versatile property is directly modulated by the (α, β) parameters, as seen in Equation 1. Using the previous assumption regarding the signal contrast between object and background gray level, the (α,β) parameters can be optimized. Hence, a parameter optimization that maximizes the brightest voxels and also suppresses those less intense ones can be useful to detect an object in the image.

The strategy chosen to fit the sigmoid function was based on a image histogram (denoted as function H), representing the signal distribution of the objects ensemble presented in the image. The β parameter could be defined using Equation 2. As noticed, the β adjustment simply takes the central point delimited between the maximum ($p_{max} = max\{H[I(r)]\}$) and the tissue threshold ($p_{min} = \eta_M\{H[I(r)]\}$) values, being η_M a first approximation of the object gray level threshold. The object gray level threshold referred as the p_{min} value is a key factor in delineating the first approximation between object/background boundary, which can be modulated by widely known automatic image threshold methods such as the maximization of Shannon entropy ($p_{Sh} = \eta_{Sh}\{H[I(r)]\}$) [8], maximization of variance ($p_{Ot} = \eta_{Ot}\{H[I(r)]\}$) [14] or image moments ($p_{Mo} = \eta_{Mo}\{H[I(r)]\}$) [17].

After β have been optimized, the α parameter can be calculated. However, in order to properly define α, another parameter should be also assumed: the transition tolerance (T). Since α modulates the ”level of detection” of the sigmoid function (Equation 1), the transition tolerance T is necessary to define the amount of crossing signals evidenced in the image histogram that may lead to misclassification. In other words, T modulates how conservative (or generative) the sigmoid function will behave and increase the flexibility of the initial p_{min} that is adopted automatically.

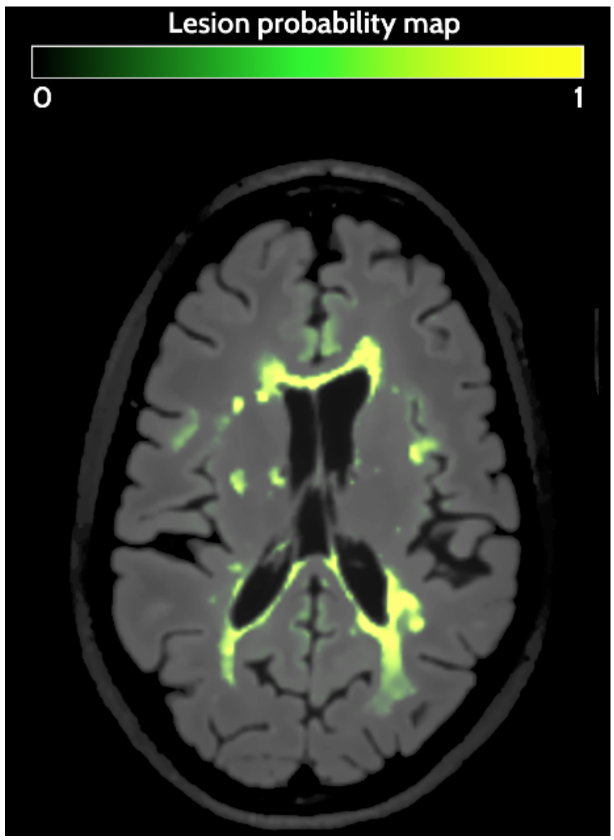

Figure 1: Example of LC model applied on the hyperintense brain white matter Multiple Sclerosis lesion segmentation. The fuzzy segmentation resulted from the LC model can be interpreted as a lesion probability map, helping to evaluate a patient-specific lesion pattern.

$$\beta = \frac{(p_{max} - p_{min})}{2} + p_{min} \tag{2}$$

$$
\begin{aligned}
S(p_{min}) = T &= \frac{1}{1+e^{-\frac{(p_{min}-\beta)}{\alpha}}} \\
e^{-\frac{(p_{min}-\beta)}{\alpha}} &= \frac{1-T}{T} \\
\beta - p_{min} &= ln\left(\frac{1-T}{T}\right).\alpha \\
\alpha &= \frac{\beta - p_{min}}{ln\left(\frac{1-T}{T}\right)}
\end{aligned} \tag{3}
$$

Assuming a conservative classification, defined by a low tolerance, the sigmoid function returns a sharp transition between NAWM and MS lesions. As seen in Equation 3, taking the $\lim_{T\to 0}\alpha = 0$ means a sharp thresholding method, i.e. a binary classification. On the other hand, a generative classification is more flexible, allowing a higher amount of signal merging between object/background voxels. The tolerance T is manually adjusted. After the (α,β) being optimized, the LC model can be applied to the biomedical image, which results in a fuzzy lesion segmentation (i.e. a weighted map representing the object and it's surrounding background), see a brief example in Section 3.

3 Useful applications

The following subsections describe the application of LC optimization, being often refined depending on the specific input details of each study. In addition, further citation details of these methods is also provided.

3.1 Hyperintense Multiple Sclerosis Lesion Segmentation

A recent study applies the LC model for hyperintense brain white matter Multiple Sclerosis lesion segmentation [2], which Figure 1 illustrates a simple example. As noticed, the hyperintense brain lesions represents the desired "object" and the white matter tissue the "background". Hence, the restricted image histogram, applied only over the white matter tissue, informs the initial conditions used in the empirical LC segmentation model presented here.

The original paper was published with the following details:

Senra Filho, A. C. (2018) "A hybrid approach based on logistic classification and iterative contrast enhancement algorithm for hyperintense multiple sclerosis lesion segmentation", Medical & Biological Engineering & Computing, Volume 56, Issue 6, pp 1063-1076. doi: 10.1007/s11517-017-1747-2.

Furthermore, a full segmentation pipeline is also freely provided as a 3D Slicer extension [15] to the scientific community, which uses the previous hyperintense brain lesion MS segmentation algorithm (called *LesionSpotlight*). More details are given in the documentation page at `https://www.slicer.org/wiki/Documentation/Nightly/Extensions/LesionSpotlight`

4 Code example

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

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

#include "itkLogisticContrastEnhancementImageFilter.h"
```

After that, the `main()` function could be initiated

```
int main(int argc, char* argv[])
{
    if ( argc < 4 )
    {
        std::cerr << "Missing parameters. " << std::endl;
        std::cerr << "Usage: " << std::endl;
        std::cerr << argv[0]
                << " inputImageFileName outputImageFileName [flipObject] thrMethod"
                << 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;
    typedef itk::ImageFileWriter<OutputImageType> WriterType;

    ReaderType::Pointer reader = ReaderType::New();
    reader->SetFileName(argv[1]);
    try
    {
        reader->Update();
    }
    catch ( itk::ExceptionObject &err)
    {
        std::cerr << "ExceptionObject caught !" << std::endl;
        std::cerr << err << std::endl;
        return -1;
    }
```

In this example, the insertion of three parameters are needed in order to run the code. An optional `flipObject` flag can be informed which simply invert the LC model by a negative α value. The first parameter is regarding the input image, the second one is the output weighted map provided by the LC application and, finally, the last parameter is just the image threshold function that is applied in the image, which can be adjust by a simple set of options (see the `itkLogisticContrastEnhancementImageFilter.h` class for more details).

After the usual data reading process, it could be inserted the `LogisticContrastEnhancementImageFilter` class declaration, as follows

```
typedef itk::LogisticContrastEnhancementImageFilter<InputImageType, OutputImageType>  FilterType;
```

```
FilterType::Pointer filter = FilterType::New();
filter->SetInput(reader->GetOutput());
filter->SetNumberOfBins(128);
if (atof(argv[3])==1) {
   filter->FlipObjectAreaOn();
}
filter->SetThresholdMethod(FilterType::ThresholdMethod(atoi(argv[4])));
filter->Update();

 std::cout<<"Alpha: "<<filter->GetAlpha()<<std::endl;
std::cout<<"Beta: "<<filter->GetBeta()<<std::endl;
```

Note that the `LogisticContrastEnhancementImageFilter` class has the following parameters set:

SetInput() Method to insert the input image. **(Required)**

SetMaximumOutput() Adjust the maximum value given in the weighted map. The default value is set as equal to 1.0. **(Optional)**

GetMaximumOutput() Returns the value given in the previous function. **(Optional)**

SetMinimumOutput() Adjust the minimum value given in the weighted map. The default value is set as equal to zero. **(Optional)**

GetMinimumOutput() Returns the value given in the previous function. **(Optional)**

SetNumberOfBins() Set the number of bins used in the image threshold calculation. The default value is set as equal to 128. **(Optional)**

GetNumberOfBins() Returns the value given in the previous function. **(Optional)**

SetTolerance() Set the tolerance level used in the LC optimization. This value must be given between zero to 100, which represents the percentage of voxels that are partially assumed in the classification process. The default value is set as equal to 1.0 (i.e. 1%). **(Optional)**

GetTolerance() Returns the value given in the previous function. **(Optional)**

SetFlipObjectArea() Informs if the LC optimization should privilege the contrary image histogram regions, i.e. the background instead of the object signals. The default value is `FlipObjectAreaOff()` (i.e. the object should be detected). **(Optional)**

GetFlipObjectArea() Returns the value given in the previous function. **(Optional)**

SetThresholdMethod() This is a list of image histogram threshold methods that can be applied in the LC model. A total of seven methods are listed. More details are found in the `itkLogisticContrastEnhancementImageFilter.h` class. The default value is `MAXENTROPY` which uses the classic Shannon entropy maximization method (`itk::MaximumEntropyThresholdCalculator`). **(Optional)**

Finally, after the LC method finishes, the resulted weighted map can be saved using

```
typename WriterType::Pointer writer = WriterType::New();
writer->SetFileName(argv[2]);
writer->SetInput(filter->GetOutput());
try
{
   writer->Update();
}
catch ( itk::ExceptionObject &err)
{
    std::cerr << "ExceptionObject caught !" << std::endl;
    std::cerr << err << std::endl;
    return -1;
}

return EXIT_SUCCESS;
}
```

Acknowledgments

The authors would like to thank the financial support provided by Conselho Nacional de Pesquisa (CNPq), grant number 405574/2017-7.

References

[1] Mariano Cabezas, Arnau Oliver, Eloy Roura, Jordi Freixenet, Joan C. Vilanova, Lluís Ramió-Torrentà, Àlex Rovira, and Xavier Lladó. Automatic multiple sclerosis lesion detection in brain MRI by FLAIR thresholding. *Computer Methods and Programs in Biomedicine*, 115(3):147–161, 2014. 1

[2] Antonio Carlos da Silva Senra Filho. A hybrid approach based on logistic classification and iterative contrast enhancement algorithm for hyperintense multiple sclerosis lesion segmentation. *Medical & Biological Engineering & Computing*, nov 2017. 1, 3.1

[3] Colm Elliott, Douglas L Arnold, D Louis Collins, and Tal Arbel. Temporally consistent probabilistic detection of new multiple sclerosis lesions in brain MRI. *IEEE transactions on medical imaging*, 32(8):1490–503, aug 2013. 1

[4] Onur Ganiler, Arnau Oliver, Yago Diez, Jordi Freixenet, Joan C. Vilanova, Brigitte Beltran, Lluís Ramió-Torrentà, Àlex Rovira, and Xavier Lladó. A subtraction pipeline for automatic detection of new appearing multiple sclerosis lesions in longitudinal studies. *Neuroradiology*, 56(5):363–374, 2014. 1

[5] Daniel García-Lorenzo, Simon Francis, Sridar Narayanan, Douglas L. Arnold, and D. Louis Collins. Review of automatic segmentation methods of multiple sclerosis white matter lesions on conventional magnetic resonance imaging. *Medical Image Analysis*, 17(1):1–18, 2013. 1, 2

[6] Daniel García-Lorenzo, Sylvain Prima, Douglas L. Arnold, D. Louis Collins, and Christian Barillot. Trimmed-likelihood estimation for focal lesions and tissue segmentation in multisequence MRI for multiple sclerosis. *IEEE Transactions on Medical Imaging*, 30(8):1455–1467, aug 2011. 1

[7] Saurabh Jain, Diana M. Sima, Annemie Ribbens, Melissa Cambron, Anke Maertens, Wim Van Hecke, Johan De Mey, Frederik Barkhof, Martijn D. Steenwijk, Marita Daams, Frederik Maes, Sabine Van Huffel, Hugo Vrenken, and Dirk Smeets. Automatic segmentation and volumetry of multiple sclerosis brain lesions from MR images. *NeuroImage: Clinical*, 8:367–75, may 2015. 1

[8] J.N. Kapur, P.K. Sahoo, and A.K.C. Wong. A new method for gray-level picture thresholding using the entropy of the histogram. *Computer Vision, Graphics, and Image Processing*, 29(3):273–285, 1985. 2

[9] Rasoul Khayati, Mansur Vafadust, Farzad Towhidkhah, and Massood Nabavi. Fully automatic segmentation of multiple sclerosis lesions in brain MR FLAIR images using adaptive mixtures method and markov random field model. *Computers in Biology and Medicine*, 38(3):379–390, mar 2008. 1

[10] L Kuncheva. *Fuzzy Classifier Design*. Studies in Fuzziness and Soft Computing. Physica-Verlag HD, 2000. 2

[11] Zhiqiang Lao, Dinggang Shen, Dengfeng Liu, Abbas F. Jawad, Elias R. Melhem, Lenore J. Launer, R. Nick Bryan, and Christos Davatzikos. Computer-Assisted Segmentation of White Matter Lesions in 3D MR Images Using Support Vector Machine. *Academic Radiology*, 15(3):300–313, mar 2008. 1

[12] Xavier Lladó, Arnau Oliver, Mariano Cabezas, Jordi Freixenet, Joan C. Vilanova, Ana Quiles, Laia Valls, Lluís Ramió-Torrentà, and Àlex Rovira. Segmentation of multiple sclerosis lesions in brain MRI: A review of automated approaches. *Information Sciences*, 186(1):164–185, mar 2012. 1

[13] Daryoush Mortazavi, Abbas Z Kouzani, and Hamid Soltanian-Zadeh. Segmentation of multiple sclerosis lesions in MR images: a review. *Neuroradiology*, 54(4):299–320, apr 2012. 1

[14] Nobuyuki Otsu. A Threshold Selection Method from Gray-Level Histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, 9(1):62–66, jan 1979. 2

[15] S Pieper, B Lorensen, and W Schroeder. The NA-MIC Kit: ITK, VTK, pipelines, grids and 3D slicer as an open platform for the medical image computing community. *3rd IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, pages 698–701, 2006. 3.1

[16] J C Souplet, C Lebrun, N Ayache, and G Malandain. An Automatic Segmentation of T2-FLAIR Multiple Sclerosis Lesions. *Multiple Sclerosis*, pages 1–11, 2008. 1

[17] Wen-Hsiang Tsai. Moment-preserving thresolding: A new approach. *Computer Vision, Graphics, and Image Processing*, 29(3):377–393, mar 1985. 2

[18] K Van Leemput, F Maes, D Vandermeulen, A Colchester, and P Suetens. Automated segmentation of multiple sclerosis lesions by model outlier detection. *IEEE transactions on medical imaging*, 20(8):677–88, aug 2001. 1

[19] Daisuke Yamamoto, Hidetaka Arimura, Shingo Kakeda, Taiki Magome, Yasuo Yamashita, Fukai Toyofuku, Masafumi Ohki, Yoshiharu Higashida, and Yukunori Korogi. Computer-aided detection of multiple sclerosis lesions in brain magnetic resonance images: False positive reduction scheme consisted of rule-based, level set method, and support vector machine. *Computerized Medical Imaging and Graphics*, 34(5):404–413, 2010. 1

[20] Byung Il Yoo, Jung Jae Lee, Ji Won Han, San Yeo Wool Oh, Eun Young Lee, James R. MacFall, Martha E. Payne, Tae Hui Kim, Jae Hyoung Kim, and Ki Woong Kim. Application of variable threshold intensity to segmentation for white matter hyperintensities in fluid attenuated inversion recovery magnetic resonance images. *Neuroradiology*, 56(4):265–281, apr 2014. 1

[21] Y Zhang, M Brady, and S Smith. Segmentation of brain MR images through a hidden Markov random field model and the expectation-maximization algorithm. *IEEE transactions on medical imaging*, 20(1):45–57, jan 2001. 1