Radius-Optimized Efficient Template Matching for Lesion Detection from Brain Images

Subhranil Koley, Pranab K. Dutta, *Member, IEEE,* and Iman Aganj

***Abstract*–Computer-aided detection of brain lesions from volumetric magnetic resonance imaging (MRI) is in demand for fast and automatic diagnosis of neural diseases. The template-matching technique can provide satisfactory outcome for automatic localization of brain lesions; however, finding the optimal template size that maximizes similarity of the template and the lesion remains challenging. This increases the complexity of the algorithm and the requirement for computational resources, while processing large MRI volumes with three-dimensional (3D) templates. Hence, reducing the computational complexity of template matching is needed. In this paper, we first propose a mathematical framework for computing the normalized cross-correlation coefficient (NCCC) as the similarity measure between the MRI volume and approximated 3D Gaussian template with linear time complexity, $\mathcal{O}(a_{max}N)$, as opposed to the conventional fast Fourier transform (FFT) based approach with the complexity $\mathcal{O}(a_{max}N\log N)$, where N is the number of voxels in the image and a_{max} is the number of tried template radii. We then propose a mathematical formulation to analytically estimate the optimal template radius for each voxel in the image and compute the NCCC with the location-dependent optimal radius, reducing the complexity to $\mathcal{O}(N)$. We test our methods on one synthetic and two real multiple-sclerosis databases, and compare its performance in lesion detection with FFT and a state-of-the-art lesion prediction algorithm. We demonstrate through our experiments the efficiency of the proposed method for brain lesion detection and its comparable performance with existing techniques.**

***Index Terms*–Brain lesion detection, computational complexity, FFT, MRI, NCCC, template matching.**

I. Introduction

A lesion in the brain represents a diseased area or one with abnormal tissue due to many factors such as infection, traumatic brain injury, tumors, multiple sclerosis (MS), stroke, etc. [1]. The complications resulting from brain lesions are numerous including neurodegeneration, memory or vision loss, behavioral changes, etc. [2]. The presence of a lesion in the brain often demands prompt clinical investigation to initiate the treatment process. MRI contributes to the diagnosis significantly by visualizing the neuroanatomy with high soft-tissue contrast *in vivo*. Detection of the exact location of lesion is essential for clinical implications and thus assists in diagnosis and treatment planning. Manual assessment (i.e. visual screening) of individual two-dimensional (2D) slices from the whole MRI volume is the conventional practice of identifying the presence of lesions. This is a cumbersome task, especially in the presence of multiple lesions (e.g., MS, metastasis, etc.) or when performing it on a large dataset. Inter- and intra-rater variabilities are other caveats of manual lesion detection. The limitations of manual screening and volumetric analysis by trained experts have led to the development of semi-automatic and automatic segmentation algorithms in the last two decades, which can assist the clinician in identifying the region of interest (ROI) from the magnetic resonance (MR) image. The study conducted by Egger *et al.* [3] on fluid attenuation inversion recovery (FLAIR) MRI of MS lesions over 50 cases, reported that the segmentation methods such as lesion growth [4] and lesion prediction algorithm (LPA) [5] underperformed for quantifying the number of lesions on the subjects with high lesion load. According to them, the difficulties in the evaluation of the confluent lesions in periventricular region might be the cause of poor performance, which emphasizes the importance of prior detection of lesions in MRI volume, especially when multiple lesions are present in brain.

Machine learning, especially deep learning, has been successfully applied for automatic segmentation of brain lesions [6]. Deep neural networks are popular because of their high performance over conventional machine-learning techniques. The present-day deep learning requires a large amount of training data as well as high computational power. In the context of MRI, different scanners use different acquisition parameters leading to variations in image characteristics. This might create difficulty for supervised approaches, for instance when testing on a previously unseen image that deviates from the training data. The key advantage of using unsupervised approaches is that they do not require any prior knowledge or training data. Cabezas *et al.* [7] developed a two-stage

Support for this research was provided in part by a Fulbright-Nehru Doctoral Research Fellowship (Award no: 2098/DR/2015-2016) and Research Associateship grant funded by Centre for Scientific and Industrial Research, Govt. of India (File no. 09/81(1326)/18, EMR-1). IA was supported by the National Institutes of Health (NIH), specifically the National Institute of Diabetes and Digestive and Kidney Diseases (K01DK101631), and the BrightFocus Foundation (A2016172S). Computational resources were provided by the NIH Shared Instrumentation Grants (S10RR023401, S10RR019307, and S10RR023043).

S. Koley and P. K. Dutta are with the School of Medical Science and Technology and Electrical Engineering department, respectively, at Indian Institute of Technology Kharagpur, Kharagpur, WB-721302, India (e-mails: subhranil.bmi.smst@gmail.com; pkd@ee.iitkgp.ac.in).

I. Aganj is with the Athinoula A. Martinos Center for Biomedical Imaging, Radiology Department, Massachusetts General Hospital, Harvard Medical School, at 149 13th St., Suite 2301, Charlestown, MA 02129, USA, and with the Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology, at 32 Vassar St., Cambridge, MA 02139, USA (e-mail: iman@nmr.mgh.harvard.edu).

unsupervised methodology consisting of modified expectation-maximization (EM) for initial soft-tissue segmentation from T1-weighted (T1), T2-weighted (T2), and proton-density (PD) images and thresholding of FLAIR images for hyperintense-lesion segmentation. They reported 55% accuracy as the best mean sensitivity obtained in hyperintense lesion detection. Generally, thresholding-based approaches depend on intensity histogram to find the cutoff intensity, based on which the target lesion (hyper- or hypo-intense) could be differentiated. This technique faces difficulties in classifying non-lesion regions having similar intensity profile as lesions, where voxel intensity values are influenced by intensity inhomogeneity or artifacts [8]. *Template matching*, a fully unsupervised technique, has been employed for automatic lesion detection from medical images because of its satisfactory performance in localization of lesions. A general choice for the template could be a radially-symmetric shape such as a circle in 2D or a sphere in 3D, where the template with varying radii is matched at each image voxel. In other words, the image is locally compared with the template by some similarity measure, and this operation is repeated for a set of radii until the optimal radius is found [9, 10]. Tourassi *et al.* [11] developed a template-matching technique with a mutual-information similarity measure to detect mass from digital mammograms. Later on, Lochanambal *et al.* [12] defined the template based on the shape and the brightness of target lesion and micro calcifications for lesion detection in mammographic images. Template matching has also been employed for detection of lung nodules; for instance, Farag *et al.* [13] proposed deformable 2D and 3D templates for spiral CT scan images, Osman *et al.* [14] developed a 3D template to perform convolution with 3D ROI image for serial CT images, and Wang *et al.* [15] proposed a 3D spherical template with varying size to detect lesions having a diameter of 4-20 mm for typical CT scan images. Later, Moltz *et al.* [16] developed a generalized methodology based on template matching for detection of various types of lesions, such as lung nodules, liver metastasis, and lymph nodes, from CT images. Next, we discuss the applications of template matching in MRI.

Warfield *et al.* [17] developed an "adaptive, template moderated, spatially varying statistical classification" framework for segmentation of brain anatomy of neonates, MS lesions, brain tumors, and damaged knee cartilage from MR images by using an anatomical template to moderate the statistical classification. Ambrosini *et al.* [8] proposed a 3D spherical template and used the normalized cross-correlation coefficient (NCCC) as the similarity measure between templates (with varying radii) and the brain volume for automatic detection of metastatic brain tumor from post-contrast T1 MRI (spoiled gradient echo pulse sequence). Later, Farjam *et al.* [18] used 3D post-Gd-DTPA T1 volumes to localize hyperintense metastatic lesions with a 3D spherical-shell template. In a different study, Yang *et al.* [19] considered contrast-enhanced MRI black-blood pulse sequence for metastatic lesion detection using 3D template. Wang *et al.* [20] performed another study with an adaptive template using NCCC as similarity measure for tumor detection using T1 MR images. We found that an exhaustive search was a common choice to find the optimal size of the template in the literature. Such an exhaustive search is responsible for the elevated computational cost of the algorithm, as it increases the runtime proportionally to the number of tried template sizes [9]. Another factor affecting the computational complexity is the choice of FFT for convolution to compute the NCCC; however, its overall complexity of $\mathcal{O}(a_{max} N \log N)$ is still costly (a_{max} is the number of tried template radii), especially when processing large MRI datasets.

In this work, we develop a new computational framework for the NCCC with the 3D Gaussian template, where we approximate the convolution with the Gaussian as multiple convolutions with a box kernel, per the principle of central limit theorem (inspired by a fast method of computing continuous wavelet transform [21]), which follows linear time, $\mathcal{O}(N)$, for a fixed radius with respect to the number of voxels in the MRI volume, N. This method is fast with the overall complexity of $\mathcal{O}(a_{max} N)$, and produces equivalent results compared to the FFT-based algorithm (Eqs. (1) – (15) have been presented in our preliminary work [9]). We then extend this approach by developing a new template-matching algorithm to compute the NCCC with the optimal-size template in linear time, $\mathcal{O}(N)$, for automatic detection of brain lesions from volumetric MR images. To accomplish this, we propose a mathematical derivation to analytically optimize the maximum template radius for each voxel in the MRI volume, leading to a computational cost that is independent of the radius of template (lesion size). The addition of such a radius-optimized template-matching approach is possible via extending our previous algorithm, but not with the FFT-based method. We tested the performances of our two methods (with overall complexities of $\mathcal{O}(a_{max} N)$ and $\mathcal{O}(N)$) on synthetic data and real MR image databases of MS lesions, and then compared them with those of the conventional FFT-based method and the LPA. Experimental results over synthetic data validate the proposed approach, and lesion-detection results on real MRI volumes show its effectiveness and applicability.

II. Methods

A. Template Matching in $\mathcal{O}(a_{max} N)$

Let $f(\vec{X})$ be the D-dimensional input image the lesions in which we intend to detect. In this work, the D-dimensional Gaussian is proposed as the template, which is approximated – following the central limit theorem – by convolving a D-dimensional symmetric and normalized box kernel, $h_D(\cdot)$, with itself $n-1$ times, denoted as $h_D^{(n)}(\cdot)$ (aka B-splines [22]). In our experiments, a small value of $n = 2$ or $n = 3$ turned out to be sufficient. The sizes of the box kernel and the engulfing template are $2a$ and $2b$, respectively, in all dimensions; i.e. $h_D(\vec{X})$ is $1/(2a)^D$ if $\vec{X} \in \Omega_a$, and 0 otherwise, where $\Omega_l := \{\vec{X} \big| |X_1|, \dots, |X_D| \le l\}$. To ensure that the box kernel fits in the engulfing template after the convolutions, we restrict its size as: $0 < a \le a_{max} < b/n$. The similarity between the given image and the symmetric template with a varying a can be computed from the following formula for the NCCC:

$$\mathrm{NCCC}(\vec{X}) := \frac{\int_{\Omega_b}\left(f(\vec{X}+\vec{X'})-\bar{f}(\vec{X})\right)\left(h_D^{(n)}(\vec{X'})-\overline{h_D^{(n)}}\right)\mathrm{d}\vec{X'}}{\sqrt{\int_{\Omega_b}\left(f(\vec{X}+\vec{X'})-\bar{f}(\vec{X})\right)^2\mathrm{d}\vec{X'}}\sqrt{\int_{\Omega_b}\left(h_D^{(n)}(\vec{X'})-\overline{h_D^{(n)}}\right)^2\mathrm{d}\vec{X'}}}, \tag{1}$$

where $\bar{f}(\vec{X})$ and $\overline{h_D^{(n)}}$ represent the mean of the image inside the template centered at $\vec{X}$, and the mean of the template, respectively. Our goal is to maximize NCCC with respect to both a and $\vec{X}$, to accurately localize the lesions. Since by definition $\int_{\Omega_b}\left(h_D^{(n)}(\vec{X'}) - \overline{h_D^{(n)}}\right)\mathrm{d}\vec{X'} = 0$, we can omit $\bar{f}(\vec{X})$ from the numerator of Eq. (1). Noting that $h_D^{(n)}(-\vec{X}) = h_D^{(n)}(\vec{X})$, we compute $\int_{\Omega_b} f(\vec{X}+\vec{X'})h_D^{(n)}(\vec{X'})\mathrm{d}\vec{X'} = \int_{\mathbb{R}^D} f(\vec{X}+\vec{X'})h_D^{(n)}(\vec{X'})\mathrm{d}\vec{X'} = \left(f * h_D^{(n)}\right)(\vec{X})$. Thanks to the separability property of the template, i.e. $h_D^{(n)}(\vec{X}) = \prod_{j=1}^{D} h_1^{(n)}(X_j)$, we can first find the solution to this convolution for $D = 1$ and then apply it sequentially for each dimension. For $D = 1$, we note that:

$$f * h_1^{(n+1)} = \left(f * h_1^{(n)}\right) * h_1. \tag{2}$$

We assume (and then verify) the following solution for the convolution:

$$\left(f * h_1^{(n)}\right)(X) = \frac{1}{(2a)^n}\sum_{k=-n}^{n}\gamma_{n,k}F_n(X+ak), \tag{3}$$

where $F_n(X) := \int_{-\infty}^{X} F_{n-1}(X')\mathrm{d}X'$, with $F_0 := f$. Let $\gamma_{n,k} := 0$ for $|k| > n$, and also $\gamma_{0,0} := 1$ for the case with no convolution. Now we substitute Eq. (3) in Eq. (2):

$$\begin{aligned}\left(f * h_1^{(n+1)}\right)(X) &= \left(\frac{1}{(2a)^n}\sum_{k=-n}^{n}\gamma_{n,k}F_n(X+ak)\right) * h_1(X)\\ &= \frac{1}{(2a)^n}\sum_{k=-n}^{n}\gamma_{n,k}\int_{-\infty}^{\infty}F_n(X+ak+X')h_1(X')\mathrm{d}X'\\ &= \frac{1}{(2a)^{n+1}}\sum_{k=-n}^{n}\gamma_{n,k}\{F_{n+1}(X+a(k+1))\\ &\qquad - F_{n+1}(X+a(k-1))\}\\ &= \frac{1}{(2a)^{n+1}}\sum_{k=-n+1}^{n+1}\gamma_{n,k-1}F_{n+1}(X+ak)\\ &\qquad - \frac{1}{(2a)^{n+1}}\sum_{k=-n-1}^{n-1}\gamma_{n,k+1}F_{n+1}(X+ak)\\ &= \frac{1}{(2a)^{n+1}}\sum_{k=-(n+1)}^{n+1}(\gamma_{n,k-1}-\gamma_{n,k+1})F_{n+1}(X+ak).\end{aligned} \tag{4}$$

According to Eq. (3), we should have:

$$\left(f * h_1^{(n+1)}\right)(X) = \frac{1}{(2a)^{n+1}}\sum_{k=-(n+1)}^{n+1}\gamma_{n+1,k}F_{n+1}(X+ak). \tag{5}$$

This validates our assumption in Eq. (3) by induction, as Eq. (3) is true for the base case $n = 0$, and provided that it is true for n, it holds for $n + 1$with the following recursive relationship for γ, which is obtained by coefficient matching between Eqs. (4) and (5):

$$\gamma_{n+1,k} = \gamma_{n,k-1} - \gamma_{n,k+1}. \tag{6}$$

For example, $\gamma_{1,\{-1,0,1\}} = \{-1,0,1\}$ and $\gamma_{2,\{-2,\dots,2\}} = \{1,0,-2,0,1\}$. Being closely related to Pascal's triangle, Eq. (6) is solved as follows:

$$\gamma_{n,k} = \begin{cases}(-1)^{(n+s)}\binom{n}{s} & n+k = 2s\\ 0 & n+k = 2s+1\end{cases}, \quad s \in \mathbb{Z}. \tag{7}$$

We now extend this to D dimensions and solve the first numerator term of Eq. (1):

$$\int_{\Omega_b} f(\vec{X}+\vec{X'})h_D^{(n)}(\vec{X'})\mathrm{d}\vec{X'} = \frac{1}{(2a)^{nD}}\sum_{\vec{k}=-n}^{n}\left(\prod_{j=1}^{D}\gamma_{n,k_j}\right)\mathcal{S}_{n,D}\{f\}(\vec{X}+a\vec{k}), \tag{8}$$

Where $\sum_{\vec{k}=-n}^{n}(\cdot)$ is short for $\sum_{k_1,\dots,k_D=-n}^{n}(\cdot)$ and the operator $\mathcal{S}_{n,D}\{f\}$ is defined as:

$$\begin{aligned}\mathcal{S}_{n,D}\{f\} &:= \mathcal{S}_{n,D}^{(n)}\{f\},\\ \mathcal{S}_{m,j}^{(n)}\{f\}(\vec{X}) &:= \int_{-\infty}^{X_j}\mathcal{S}_{m,j-1}^{(n)}\{f\}(X_1,\dots,X_j',\dots,X_D)\mathrm{d}X_j',\\ \mathcal{S}_{0,j}^{(n)}\{f\} &:= \begin{cases}\mathcal{S}_{n,j-1}^{(n)}\{f\} & , \ j > 1\\ f & , \ j = 1\end{cases}.\end{aligned} \tag{9}$$

Using the fact that $\overline{h_D^{(n)}} = 1/(2b)^D$ (see below), the remainder of the numerator of Eq. (1) can be computed similarly to Eq. (8) (by fixing $n = 1$):

$$-\overline{h_D^{(n)}}\int_{\Omega_b} f(\vec{X}+\vec{X'})\mathrm{d}\vec{X'} = -\frac{1}{(2b)^D}\sum_{\vec{k}=-1}^{1}\left(\prod_{j=1}^{D}\gamma_{1,k_j}\right)\mathcal{S}_{1,D}\{f\}(\vec{X}+b\vec{k}). \tag{10}$$

Next, we rewrite the first factor of the denominator of Eq. (1) using the common expansion of the variance as $\int_{\Omega_b}\left(f(\vec{X}+\vec{X'}) - \bar{f}(\vec{X})\right)^2\mathrm{d}\vec{X'} = (2b)^D\left(\overline{f^2}(\vec{X}) - \bar{f}(\vec{X})^2\right)$, where $\overline{f^p}(\vec{X})$, $p = 1,2$, is calculated similarly to Eq. (8) as:

$$\overline{f^p}(\vec{X}) = \frac{1}{(2b)^D}\sum_{\vec{k}=-1}^{1}\left(\prod_{j=1}^{D}\gamma_{1,k_j}\right)\mathcal{S}_{1,D}\{f^p\}(\vec{X}+b\vec{k}), \tag{11}$$

Lastly, we calculate the second factor in the denominator of Eq. (1) by following: $\int_{\Omega_b}\left(h_D^{(n)}(\vec{X'}) - \overline{h_D^{(n)}}\right)^2\mathrm{d}\vec{X'} = (2b)^D\left(\overline{h_D^{(n)2}} - \overline{h_D^{(n)}}^2\right)$. The Fourier transform of the box function is $\mathcal{F}\{h_D\} = \prod_{j=1}^{D}\mathrm{sinc}(a\omega_j)$, which leads to the Fourier transform of the kernel $\mathcal{F}\left\{h_D^{(n)}\right\} = \mathcal{F}\{h_D\}^n = \prod_{j=1}^{D}\mathrm{sinc}^n(a\omega_j)$ via the convolution theorem. The integral of the template is $\int_{\Omega_b} h_D^{(n)}(\vec{X'})\mathrm{d}\vec{X'} = \int_{\mathbb{R}^D} h_D^{(n)}(\vec{X'})\mathrm{d}\vec{X'} = \mathcal{F}\left\{h_D^{(n)}\right\}\Big|_{\vec{\omega}=0} = 1$, from which the template mean is computed as $\overline{h_D^{(n)}} = 1/(2b)^D$. As for the mean of the square of the template, we use Parseval's theorem as follows:

$$\begin{aligned}\int_{\Omega_b} h_D^{(n)2}(\vec{X'})\mathrm{d}\vec{X'} &= \int_{\mathbb{R}^D} h_D^{(n)2}(\vec{X'})\mathrm{d}\vec{X'}\\ &= \frac{1}{(2\pi)^D}\int_{\mathbb{R}^D}\mathcal{F}\left\{h_D^{(n)}\right\}^2(\vec{\omega})\mathrm{d}\vec{\omega} = \prod_{j=1}^{D}\frac{1}{2\pi}\int_{-\infty}^{\infty}\mathrm{sinc}^{2n}(a\omega_j)\,\mathrm{d}\omega_j\\ &= \left(\frac{1}{2\pi a}\int_{-\infty}^{\infty}\mathrm{sinc}^{2n}\,\omega\,\mathrm{d}\omega\right)^D.\end{aligned} \tag{92}$$

We now define and compute [23, 24]:

$$\beta_n \coloneqq \frac{1}{2\pi}\int_{-\infty}^{\infty} \operatorname{sinc}^{2n}\omega \,\mathrm{d}\omega = n\sum_{i=0}^{n-1}\frac{(-1)^i (n-i)^{2n-1}}{i!\,(2n-i)!}. \tag{10}$$

Therefore, $\overline{h_D^{(n)^2}} = \left(\frac{\beta_n}{2ab}\right)^D$, and subtracting $(2b)^D\overline{h_D^{(n)}}^2$ from it leads to:

$$\int_{\Omega_b}\left(h_D^{(n)}(\overrightarrow{X'}) - \overline{h_D^{(n)}}\right)^2 \mathrm{d}\overrightarrow{X'} = \left(\frac{\beta_n}{a}\right)^D - \frac{1}{(2b)^D}. \tag{14}$$

Substituting all the above in Eq. (1) will result in the following formula for the proposed fast approach of NCCC computation:

$$\mathrm{NCCC}(\vec{X}) = \frac{\left(\frac{2b}{(2a)^n}\right)^D \sum_{\vec{k}=-n}^{n}\left(\prod_{j=1}^{D}\gamma_{n,k_j}\right)\mathcal{S}_{n,D}\{f\}(\vec{X}+a\vec{k}) - \sum_{\vec{k}=-1}^{1}\left(\prod_{j=1}^{D}\gamma_{1,k_j}\right)\mathcal{S}_{1,D}\{f\}(\vec{X}+b\vec{k})}{\sqrt{(2b)^D\sum_{\vec{k}=-1}^{1}\left(\prod_{j=1}^{D}\gamma_{1,k_j}\right)\mathcal{S}_{1,D}\{f^2\}(\vec{X}+b\vec{k}) - \left(\sum_{\vec{k}=-1}^{1}\left(\prod_{j=1}^{D}\gamma_{1,k_j}\right)\mathcal{S}_{1,D}\{f\}(\vec{X}+b\vec{k})\right)^2}\sqrt{\left(\frac{2\beta_n b}{a}\right)^D - 1}}. \tag{115}$$

NCCC values range from -1 to 1. In this subsection, we compute the NCCC for several different values of a. However, note that $\mathcal{S}_{n,D}\{f^p\}$ are volumes independent of a and can be pre-computed, along with the second term in the numerator and the first factor in the denominator of Eq. (115). Furthermore, the second factor in the denominator is a scalar that is computed fast in $\mathcal{O}(1)$ for each a. Thus, in the proposed approach to estimate NCCC with the Gaussian template, the bulk of the computational cost is only due to the first term in the numerator of Eq. (15), which can be computed in $\mathcal{O}(N)$ for each a, with N the number of voxels in the image. On the contrary, the computational complexity of FFT is $\mathcal{O}(N\log N)$ for each a [25], making the overall computational cost of lesion detection noticeably higher for the FFT-based algorithm (template has to be zero-padded to the size of the image) than the proposed approach. This difference in computational cost is particularly amplified given that the NCCC needs to be repeatedly computed for many values of $a = 1,2,\dots,a_{max}$.

B. Template Matching in $\mathcal{O}(N)$

a. Defining NCCC

We intend to optimize the NCCC with respect to the template radius; however, analytical maximization of Eq. (15) with respect to the template radius is difficult. Therefore, we aimed at redefining NCCC, which would be suitable for voxel-wise maximization of template radius. Let us assume the template, $g_D(\vec{X}) \coloneqq \exp\left(-\frac{1}{2a^2}\|\vec{X}\|^2\right)$ and the ROI are both Gaussian, where $\|\cdot\|$ denotes the Euclidean norm. As in the previous section, we solve NCCC by considering all terms of numerator and denominator, individually. We begin by focusing on the first term of the numerator of Eq. (1). For a Gaussian-shaped lesion, it can be solved as:

$$\begin{aligned}\int_{\Omega_b} f(\vec{X}+\overrightarrow{X'})g_D(\overrightarrow{X'})\mathrm{d}\overrightarrow{X'} &= \int_{\Omega_b} e^{-\frac{1}{2\sigma^2(\vec{X})}\left\|\overrightarrow{X'}-\vec{\mu}(\vec{X})\right\|^2} e^{-\frac{1}{2a^2}\left\|\overrightarrow{X'}\right\|^2}\mathrm{d}\overrightarrow{X'} \\ &= \int_{\Omega_b} e^{-\frac{1}{2}\left(\frac{\left\|\overrightarrow{X'}-\vec{\mu}(\vec{X})\right\|^2}{\sigma^2(\vec{X})}+\frac{\left\|\overrightarrow{X'}\right\|^2}{a^2}\right)}\mathrm{d}\overrightarrow{X'},\end{aligned} \tag{16}$$

where, $\vec{\mu}(\vec{X})$ and $\sigma(\vec{X})$ are the mean (with respect to $\vec{X}$) and standard deviation (SD) of the ROI, respectively. We now simplify Eq. (16) by reducing the power of exponential as,

$$\begin{aligned}&\frac{\left\|\overrightarrow{X'}-\vec{\mu}(\vec{X})\right\|^2}{\sigma^2(\vec{X})} + \frac{\left\|\overrightarrow{X'}\right\|^2}{a^2} = \frac{\left\|\overrightarrow{X'}\right\|^2}{\sigma^2(\vec{X})} - 2\frac{\overrightarrow{X'}\cdot\vec{\mu}(\vec{X})}{\sigma^2(\vec{X})} + \frac{\|\vec{\mu}(\vec{X})\|^2}{\sigma^2(\vec{X})} + \frac{\left\|\overrightarrow{X'}\right\|^2}{a^2} = \\ &\left(\frac{1}{\sigma^2(\vec{X})}+\frac{1}{a^2}\right)\left[\left\|\overrightarrow{X'}\right\|^2 - 2\frac{\overrightarrow{X'}\cdot\vec{\mu}(\vec{X})}{1+\frac{\sigma^2(\vec{X})}{a^2}} + \frac{\|\vec{\mu}(\vec{X})\|^2}{\left(1+\frac{\sigma^2(\vec{X})}{a^2}\right)^2}\right] + \frac{\|\vec{\mu}(\vec{X})\|^2}{\sigma^2(\vec{X})} - \\ &\frac{\|\vec{\mu}(\vec{X})\|^2}{\sigma^2(\vec{X})+\frac{\sigma^4(\vec{X})}{a^2}} = \left(\frac{1}{\sigma^2(\vec{X})}+\frac{1}{a^2}\right)\left\|\overrightarrow{X'} - \frac{\vec{\mu}(\vec{X})}{1+\frac{\sigma^2(\vec{X})}{a^2}}\right\|^2 + \frac{\|\vec{\mu}(\vec{X})\|^2}{a^2+\sigma^2(\vec{X})}.\end{aligned} \tag{17}$$

Now we can rewrite Eq. (16) as,

$$\begin{aligned}&\int_{\Omega_b} f(\vec{X}+\overrightarrow{X'})g_D(\overrightarrow{X'})\mathrm{d}\overrightarrow{X'} \\ &= e^{-\frac{\|\vec{\mu}(\vec{X})\|^2}{2(a^2+\sigma^2(\vec{X}))}}\int_{\Omega_b} e^{-\frac{1}{2}\left(\frac{1}{\sigma^2(\vec{X})}+\frac{1}{a^2}\right)\left\|\overrightarrow{X'}-\frac{\vec{\mu}(\vec{X})}{1+\frac{\sigma^2(\vec{X})}{a^2}}\right\|^2}\mathrm{d}\overrightarrow{X'}.\end{aligned} \tag{18}$$

We now define $\vec{t} \coloneqq \frac{\overrightarrow{X'} - \frac{\vec{\mu}(\vec{X})}{1+\frac{\sigma^2(\vec{X})}{a^2}}}{a\sigma(\vec{X})\sqrt{\frac{2}{a^2+\sigma^2(\vec{X})}}}$; hence, $\mathrm{d}\overrightarrow{X'} = \left(a\sigma(\vec{X})\sqrt{\frac{2}{a^2+\sigma^2(\vec{X})}}\right)^{\mathrm{D}}\mathrm{d}\vec{t}$. After the change of variables, and assuming the radius of engulfing template to be sufficiently large, as $b\to\infty$, Eq. (18) becomes,

$$\begin{aligned}&\int_{\Omega_b} f(\vec{X}+\overrightarrow{X'})g_D(\overrightarrow{X'})d\overrightarrow{X'} \\ &= \left(\sqrt{\frac{2a^2\sigma^2(\vec{X})}{a^2+\sigma^2(\vec{X})}}\right)^D e^{-\frac{\|\vec{\mu}(\vec{X})\|^2}{2(a^2+\sigma^2(\vec{X}))}}\int_{\mathbb{R}^D} e^{-\|\vec{t}\|^2}\mathrm{d}\vec{t} \\ &= \left(a\sigma(\vec{X})\sqrt{\frac{2\pi}{a^2+\sigma^2(\vec{X})}}\right)^D e^{-\frac{\|\vec{\mu}(\vec{X})\|^2}{2(a^2+\sigma^2(\vec{X}))}}.\end{aligned} \tag{19}$$

We now focus on the first term of denominator, i.e. image variance inside the template. We use the expansion of variance, $\int_{\Omega_b}\left(f(\vec{X}+\overrightarrow{X'}) - \bar{f}(\vec{X})\right)^2\mathrm{d}\overrightarrow{X'} = (2b)^D\left(\overline{f^2}(\vec{X}) - \bar{f}(\vec{X})^2\right)$, and compute $\overline{f^2}(\vec{X})$ and $\bar{f}(\vec{X})^2$ as follows.

$$\begin{aligned}\overline{f^2}(\vec{X}) &= \frac{1}{(2b)^D}\int_{\Omega_b} f(\vec{X}+\overrightarrow{X'})^2\,\mathrm{d}\overrightarrow{X'}, \\ (2b)^D\overline{f^2}(\vec{X}) &= \int_{\Omega_b} e^{-\frac{\left\|\overrightarrow{X'}-\vec{\mu}(\vec{X})\right\|^2}{\sigma^2(\vec{X})}}\mathrm{d}\overrightarrow{X'}.\end{aligned} \tag{20}$$

Again, for a large $b\to\infty$, Eq. (20) can be simplified as:

$$(2b)^D\overline{f^2}(\vec{X}) = \left(\sqrt{\pi}\sigma(\vec{X})\right)^D. \tag{21}$$

Similarly, one can see that,

$$\lim_{b\to\infty}(2b)^D\bar{f}(\vec{X})^2 = 0. \tag{22}$$

Subtracting Eq. (22) from Eq. (21), we get an approximation for the first term of denominator of NCCC for a large b as,

$$\sqrt{\int_{\Omega_b}\left(f(\vec{X}+\overrightarrow{X'})-\bar{f}(\vec{X})\right)^2 \mathrm{d}\overrightarrow{X'}}=\left(\sqrt{\pi}\sigma(\vec{X})\right)^{\frac{D}{2}}. \quad (23)$$

Next, we took on the second term of denominator, the template variance. We followed the expansion solution to initiate its computation. First, the mean of the square of the template can be computed as,

$$\overline{g_D{}^2}=\frac{1}{(2b)^D}\int_{\Omega_b} g_D{}^2(\overrightarrow{X'})\mathrm{d}\overrightarrow{X'}=\frac{1}{(2b)^D}\int_{\Omega_b} e^{-\left\|\frac{\overrightarrow{X'}}{a}\right\|^2}\mathrm{d}\overrightarrow{X'}. \quad (24)$$

Like previous steps, for a large $b\to\infty$, Eq. (24) can be simplified as, $(2b)^D\overline{g_D{}^2}=\left(\sqrt{\pi}a\right)^D$. Similarly, the mean of the template is zero, and we obtain the approximate solution (for large b) of the second term of denominator of NCCC as,

$$\sqrt{\int_{\Omega_b}\left(g_D(\overrightarrow{X'})-\overline{g_D}\right)^2\mathrm{d}\overrightarrow{X'}}=\sqrt{(2b)^D\left(\overline{g_D{}^2}-\overline{g_D}^2\right)}=\left(\sqrt{\pi}a\right)^{\frac{D}{2}}. \quad (25)$$

Next, we compute the second term of numerator of NCCC. We have, $\overline{g_D}=0$; therefore, $\int_{\Omega_b} f(\vec{X}+\overrightarrow{X'})\overline{g_D}\mathrm{d}\overrightarrow{X'}=0$.

Substituting the obtained solutions of all terms associated with NCCC in Eq. (1) and simplifying it further, results in the following formula as the final solution for computing NCCC, when both image and template are Gaussian.

$$\mathrm{NCCC}(\vec{X})=\left(\frac{2a\sigma(\vec{X})}{a^2+\sigma^2(\vec{X})}\right)^{\frac{D}{2}} e^{-\frac{\|\vec{\mu}(\vec{X})\|^2}{2(a^2+\sigma^2(\vec{X}))}}. \quad (26)$$

b. Optimization of Template Radius

The present form of NCCC is suitable for analytical maximization with respect to the template radius, a. The voxel-wise maximizer, a, would allow the computation of NCCC in a single step, thus reducing the computational complexity to the linear time, given that the NCCC no longer needs to be computed for a set of template radii. Let us assume, $\alpha=a/\sigma\,(\vec{X})$. With this notation, Eq. (26) can be simplified as,

$$Y\coloneqq\mathrm{NCCC}(\vec{X})=\left(\frac{2}{\alpha+\frac{1}{\alpha}}\right)^{\frac{D}{2}} e^{-\frac{\rho}{\alpha^2+1}}, \quad (27)$$

where, $\rho\coloneqq\frac{\|\vec{\mu}(\vec{X})\|^2}{2\sigma^2(\vec{X})}$. After substitution, we maximize Eq. (27) by equating its derivative w.r.t. α, $\frac{\mathrm{d}Y}{\mathrm{d}\alpha}=e^{-\frac{\rho}{\alpha^2+1}}\left(\frac{2\alpha}{\alpha^2+1}\right)^{D/2}\left(\frac{2\alpha\rho}{(\alpha^2+1)^2}-\frac{D(\alpha^2-1)}{2\alpha(\alpha^2+1)}\right)$, to zero, resulting in the following analytically optimized voxel-wise template radius,

$$a^*(\vec{X})=\sqrt{\frac{\|\vec{\mu}(\vec{X})\|^2}{D}+\sqrt{\frac{\|\vec{\mu}(\vec{X})\|^4}{D^2}+\sigma^4(\vec{X})}}. \quad (28)$$

c. Local Statistics of Image Volume

The local statistics, mean and SD, are the key parameters to compute the voxel-wise optimized radius. We measured these two parameters from Gaussian smoothed image to counter the small intensity variations in MRI. Gaussian smoothing was performed on the MRI volume with a 3D Gaussian filter of size $W\times W\times W$, where $W=2b'+1$ and $b'<b$, and an SD of 2 voxels. We then computed the background minimum image, f_{BG}, via moving minimum filtering over the smoothed image with a kernel of the same size to record the local minimum value inside the sliding window, which we then used in computing local statistics. The time complexity of this step is still kept $\mathcal{O}(N)$ by sequentially applying 1D filters in each dimension in both filtering steps.

Mean

We compute the local (moving) mean at each voxel in the smoothed image. The moving minimum is locally subtracted from the local image, in order to remove the effect of the background in the local neighborhood. We define the D-dimensional mean as described below.

$$\begin{aligned}\vec{\mu}(\vec{X})&=\frac{\int_{\Omega_{b'}}\left(f_s(\vec{X}+\overrightarrow{X'})-f_{BG}(\vec{X})\right)\overrightarrow{X'}\mathrm{d}\overrightarrow{X'}}{\int_{\Omega_{b'}}\left(f_s(\vec{X}+\overrightarrow{X'})-f_{BG}(\vec{X})\right)\mathrm{d}\overrightarrow{X'}}\\&=\frac{\int_{-b'}^{b'}\dots\int_{-b'}^{b'}\overrightarrow{X'}f_s(\vec{X}+\overrightarrow{X'})\mathrm{d}\overrightarrow{X'}-f_{BG}(\vec{X})\int_{-b'}^{b'}\dots\int_{-b'}^{b'}\overrightarrow{X'}\mathrm{d}\overrightarrow{X'}}{\int_{-b'}^{b'}\dots\int_{-b'}^{b'}f_s(\vec{X}+\overrightarrow{X'})\mathrm{d}\overrightarrow{X'}-f_{BG}(\vec{X})\int_{-b'}^{b'}\dots\int_{-b'}^{b'}\mathrm{d}\overrightarrow{X'}},\end{aligned} \quad (29)$$

where $f_s(\vec{X})$ is the Gaussian-smoothed image. The second term of the numerator is zero due to symmetry. To simplify the above expression, assume, $\overrightarrow{X'}=\vec{t}-\vec{X}$, so $\mathrm{d}\overrightarrow{X'}=\mathrm{d}\vec{t}$. Hence, we can rewrite Eq. (29) as,

$$\begin{aligned}&\vec{\mu}(\vec{X})\\&=\frac{\int_{X_1-b'}^{X_1+b'}\dots\int_{X_D-b'}^{X_D+b'}\vec{t}\,f_s(\vec{t})\,\mathrm{d}\vec{t}-\vec{X}\int_{X_1-b'}^{X_1+b'}\dots\int_{X_D-b'}^{X_D+b'}f_s(\vec{t})\,\mathrm{d}\vec{t}}{\int_{X_1-b'}^{X_1+b'}\dots\int_{X_D-b'}^{X_D+b'}f_s(\vec{t})\mathrm{d}\vec{t}-(2b')^D f_{BG}(\vec{X})}.\end{aligned} \quad (30)$$

The above equation can be computed using our proposed solution for convolution in D dimensions (refer to Eq. (8)) to solve all the integrals (with $n=1$) as shown below.

$$\begin{aligned}\vec{\mu}(\vec{X})&=\begin{pmatrix}\sum_{\vec{k}=-1}^{1}\left(\prod_{j=1}^{D}\gamma_{1,k_j}\right)\mathcal{S}_{1,D}\{\vec{u}f_s(\vec{u})\}(\vec{X}+b'\vec{k})\\-\vec{X}\sum_{\vec{k}=-1}^{1}\left(\prod_{j=1}^{D}\gamma_{1,k_j}\right)\mathcal{S}_{1,D}\{f_s\}(\vec{X}+b'\vec{k})\end{pmatrix}\\&/\left(\sum_{\vec{k}=-1}^{1}\left(\prod_{j=1}^{D}\gamma_{1,k_j}\right)\mathcal{S}_{1,D}\{f_s\}(\vec{X}+b'\vec{k})-(2b')^D f_{BG}(\vec{X})\right)\end{aligned} \quad (31)$$

Standard Deviation

SD is computed for each voxel by using a $W\times W\times W$ kernel in Gaussian-smoothed background-subtracted image. From the definition of variance, the SD (σ) can be written as,

$$\sigma(\vec{X})=\sqrt{\frac{1}{D}\left(\overline{\|\vec{X}\|^2}-\|\vec{\mu}(\vec{X})\|^2\right)}. \quad (32)$$

The covariance of a vector is defined as a $D\times D$ matrix. However, given that the template is a radially symmetric Gaussian, we consider a scalar variance here as $1/D$ of the trace of the covariance matrix. We proceed by computing $\overline{\|\vec{X}\|^2}$ as,

$$\begin{aligned}\overline{\|\vec{X}\|^2}&=\frac{\int_{\Omega_{b'}}\left(f_s(\vec{X}+\overrightarrow{X'})-f_{BG}(\vec{X})\right)\|\overrightarrow{X'}\|^2\mathrm{d}\overrightarrow{X'}}{\int_{\Omega_{b'}}\left(f_s(\vec{X}+\overrightarrow{X'})-f_{BG}(\vec{X})\right)\mathrm{d}\overrightarrow{X'}}\\&=\frac{\int_{-b'}^{b'}\dots\int_{-b'}^{b'}\|\overrightarrow{X'}\|^2 f_s(\vec{X}+\overrightarrow{X'})\mathrm{d}\overrightarrow{X'}-f_{BG}(\vec{X})\int_{-b'}^{b'}\dots\int_{-b'}^{b'}\|\overrightarrow{X'}\|^2\mathrm{d}\overrightarrow{X'}}{\int_{-b'}^{b'}\dots\int_{-b'}^{b'}f_s(\vec{X}+\overrightarrow{X'})\mathrm{d}\overrightarrow{X'}-f_{BG}(\vec{X})\int_{-b'}^{b'}\dots\int_{-b'}^{b'}\mathrm{d}\overrightarrow{X'}}.\end{aligned} \quad (33)$$

The denominator of mean and $\overline{\|\vec{X}\|^2}$ are the same, which is computed in Eq. (31). We compute the first term of numerator

of $\overline{\|\vec{X}\|^2}$, by a change of variables, $\overrightarrow{X'} = \vec{t} - \vec{X}$, so $\mathrm{d}\overrightarrow{X'} = \mathrm{d}\vec{t}$,

$$\int_{\Omega_{b'}} \|\overrightarrow{X'}\|^2 f_s(\vec{X} + \overrightarrow{X'})\mathrm{d}\overrightarrow{X'} = \int_{X_1-b'}^{X_1+b'} \cdots \int_{X_D-b'}^{X_D+b'} \|\vec{t}\|^2 f_s(\vec{t})\mathrm{d}\vec{t}$$
$$-2\vec{X} \cdot \int_{X_1-b'}^{X_1+b'} \cdots \int_{X_D-b'}^{X_D+b'} \vec{t} f_s(\vec{t})\mathrm{d}\vec{t} \tag{34}$$
$$+ \|\vec{X}\|^2 \int_{X_1-b'}^{X_1+b'} \cdots \int_{X_D-b'}^{X_D+b'} f_s(\vec{t})\mathrm{d}\vec{t}\,.$$

Similarly, to the case of the mean, we apply our approach to computation of convolution in the above equation (with $n = 1$) as described below,

$$\int_{\Omega_{b'}} \|\overrightarrow{X'}\|^2 f_s(\vec{X} + \overrightarrow{X'})\mathrm{d}\overrightarrow{X'}$$
$$= \sum_{\vec{k}=-1}^{1} \left(\prod_{j=1}^{D} \gamma_{1,k_j} \right) \mathcal{S}_{1,D}\{\|\vec{u}\|^2 f_s(\vec{u})\}(\vec{X} + b'\vec{k})$$
$$-2\vec{X} \cdot \sum_{\vec{k}=-1}^{1} \left(\prod_{j=1}^{D} \gamma_{1,k_j} \right) \mathcal{S}_{1,D}\{\vec{u} f_s(\vec{u})\}(\vec{X} + b'\vec{k}) \tag{35}$$
$$+\|\vec{X}\|^2 \sum_{\vec{k}=-1}^{1} \left(\prod_{j=1}^{D} \gamma_{1,k_j} \right) \mathcal{S}_{1,D}\{f_s\}(\vec{X} + b'\vec{k}).$$

We compute the second term in the numerator of Eq. (33) the same way as in Eq. (34), by using $f_s(\vec{t}) \leftarrow 1$. Finally, we have the solution of the terms necessary to calculate local SD.

d. Template Matching with Maximized Radius

In this step, the mean and SD are substituted in Eq. (28) to obtain the "a"-maximized volume. Further, the optimized a^* volume should be thresholded as $a^* \leftarrow \min(a^*, b/n - 1)$ before being substituted in the proposed solution of NCCC as described in Eq. (15). This is how the NCCC computational complexity can be reduced form $\mathcal{O}(a_{max}N)$ to $\mathcal{O}(N)$, i.e. we no longer need to use a set of templates with varying radii. In contrast, the FFT approach cannot make use of an optimal radius map to reduce its complexity any lower than $\mathcal{O}(a_{max}N \log N)$, given that FFT-based convolution cannot be performed with a locally varying kernel.

III. Results

In this section, we present experimental results comparing the proposed methods and other existing techniques, on 3D synthetic and real MRI data ($D = 3$). We first describe the FFT-based NCCC computation, to which we will compare our methods next.

A. FFT-Based Template Matching

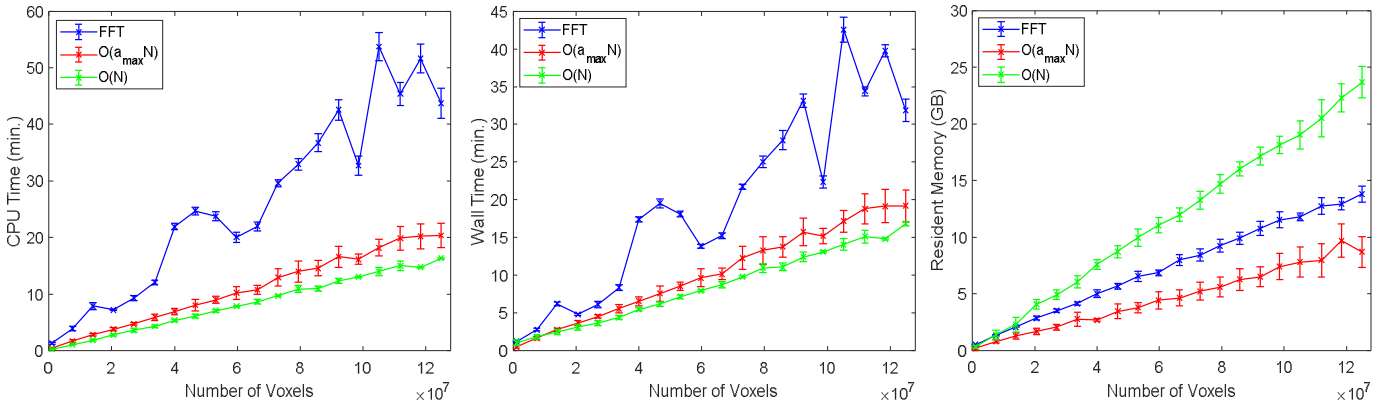
We describe here how to use FFT to calculate the NCCC via Eq. (1). We define a truncated Gaussian kernel ($n \to \infty$) as, $h_D^{(\infty)}(\vec{X}) = exp\left(-\|\vec{X}\|^2/(2a^2)\right) \times Box_{kernel}$ for $a = 1, \ldots, a_{max}$, which is used as the template. Box_{kernel} is a volume of equal size to the image, with the value of 1 inside Ω_b and 0 outside. We solve the convolution in the frequency domain by multiplying the FFT of f by that of $h_D^{(\infty)}$ and inverse Fourier transforming (IFFT) it, resulting in the first term of numerator of Eq. (1). The remainder of the numerator is computed similarly by using Box_{kernel} instead of $h_D^{(\infty)}$. The mean of the template is calculated as, $\overline{h_D^{(\infty)}} = \int_{\Omega_b} exp\left(-\|\vec{X}\|^2/(2a^2)\right) d\vec{X}/|\Omega_b|$. The denominators can be similarly computed. In the FFT-based approach, NCCC needs to be computed for a set of template radii, resulting in the computational complexity of $\mathcal{O}(a_{max}N\, log\, N)$.

B. Experiments on Synthetic Data

We first evaluated the proposed methods on synthetic data and compared their performances with that of FFT. We compared the detection accuracy and runtime of the three methods. We created twenty artificial volumes of size $513 \times 513 \times 513$ voxels, where each volume contained an enhancing sphere located at a random location with a random radius (varying in between 8 to 21 pixels) to generate the synthetic data. In our first experiment, we aimed to determine how accurately the enhancing spheres could be detected with their approximated radii from the respective volumes. During the experiments, we used the same input parameters for all methods such as $b = 50$ and $a_{max} = 24$ (except for the $\mathcal{O}(N)$ method, instead $b' = 24$). Note that n is ∞ for FFT and 2 for our methods. a_{max} was calculated as $a_{max} = b/n - 1$. During this experiment, we observed that both the FFT-based and our $\mathcal{O}(a_{max}N)$ template matching accurately identified the centers of the enhancing spheres in the respective volumes. For the proposed $\mathcal{O}(N)$ method, we calculated the analytically optimized a^* for each voxel, using which we computed the NCCC in linear time over the entire image. We then simply looked for the voxel with maximal NCCC value to identify the center of the detected enhancing sphere, and the a^* at that location was considered to be the detected radius. Since the SD of the Gaussian is proportional – but not necessarily equal – to the radius of the detected binary sphere, we conducted a linear regression analysis to reveal the ratio of the ground-truth binary radius to the detected Gaussian SD. From the experiment, we found their ratio to be 1.61 ± 0.05 and 1.74 ± 0.06 for the $\mathcal{O}(a_{max}N)$ and the FFT methods, respectively, both values significant ($p = 9 \times 10^{-18}$) [9].

As for the $\mathcal{O}(N)$ method, we first obtained the above mentioned ratio as 1.96 ± 0.11 ($p = 5 \times 10^{-16}$). The centers of the enhancing spheres, however, were not detected by the $\mathcal{O}(N)$ method with pinpoint accuracy; the mean distance between the ground-truth and detected centers was 1.6 ± 1.8 voxels. Consequently, we looked up the values of a^* at the known centers of the spheres, and recomputed the ground-truth-to-detected ratio as 2.12 ± 0.11. Next, we repeated the $\mathcal{O}(N)$ experiment with the corrected $a^* \leftarrow 2.12 \times a^*$. This helped us to reach absolute localization accuracy, with the new ground-truth-to-detected ratio of 1.01 ± 0.03 ($p = 8 \times 10^{-21}$).

We then performed another experiment on the synthetic data to analyze the runtime. The data were generated by creating 20 different volumes with linearly increasing number of voxels from $N = 10^6$ ($100 \times 100 \times 100$) to $N = 1.25 \times 10^8$ ($500 \times$



(a) **(b)** **(c)**

Fig. 1. Analysis of runtime (CPU and wall time) and resident memory of $\mathcal{O}(N)$ (green), $\mathcal{O}(a_{max}N)$ (red), and FFT-based (blue) approaches on synthetic data.

500×500), each of which containing a homogeneous bright sphere (all ones) of radius 17 pixels at the center. We ran the scripts of the algorithms 10 times for each synthetic volume for sphere detection with the same parameters as above. We measured the mean and SD of the 10 runtimes and compared them among the different methods, as depicted in Fig. 1. We ran the experiments in MATLAB on a Linux computing cluster with each node including 8 processors with 56 GB shared virtual memory (cluster allots 1 CPU with 7 GB of virtual memory to a single submitted job). To run the scripts on single thread for an unbiased comparison, we used the `singleCompThread` option, but we found that MATLAB still multithreaded the codes on multiple cores. Therefore, we report both the wall time (the real-world time elapsed between the start and end of the code) and the CPU time (the total time spent by all cores to execute the script). The $\mathcal{O}(\mathrm{N})$ method (green curve in Fig. 1) was respectively 3.3 ± 0.7 and 1.4 ± 0.1 times faster than the FFT and $\mathcal{O}(a_{max}N)$ methods in terms of CPU time, and 2.2 ± 0.6 and 1.2 ± 0.2 times faster in terms of wall time. Additionally, the $\mathcal{O}(a_{max}N)$ approach was 2.5 ± 0.4 and 1.9 ± 0.4 times faster than the FFT method in terms of CPU and wall times, respectively. The non-monotonicity seen in the FFT runtime is partially because the radix-2 implementation is faster when each dimension is a power of 2. We also observed that $\mathcal{O}(a_{max}N)$ method required 1.7 ± 0.3 and 2.5 ± 0.3 times less resident memory then the FFT and $\mathcal{O}(N)$ methods, respectively, and that the FFT-based template matching used 1.5 ± 0.3 times less memory than the $\mathcal{O}(N)$ method.

C. Experiments on Clinical MRI

We tested all three methods on *in vivo* MRI data of human brain for automatic detection of multiple sclerosis (MS) lesions. MS is a chronic demyelinating disease that generally affects the central nervous system, with lesions appearing in various regions of the brain, predominantly the white matter (WM) [4, 26]. Given that the cerebrospinal fluid (CSF) is attenuated in FLAIR (though sometime lesions are iso-intense to the hyper-intense gray matter), FLAIR MRI is suitable for understanding the lesion burden and measuring the amount of damage due to demyelination [3, 6]. We tested the methods over the FLAIR sequence of two different MS lesion MRI databases; the first one is the training data of 'MS lesion segmentation challenge 2008' (MSLS 2008), which was conducted in conjunction with MICCAI 2008 (http://www.ia.unc.edu/MSseg), and the other dataset is the training data of the 'MS segmentation challenge using a data management and processing infrastructure' (MSSEG 2016), which was conducted in MICCAI 2016 (https://portal.fli-iam.irisa.fr/msseg-challenge/overview).

a. MRI Scans of MS Subjects

We used 20 cases of training data of MSLS 2008 to evaluate the methods using the ground-truth annotation from the Boston Children's Hospital (BCH) rater as the reference. Isotropic-voxel (0.5×0.5×0.5 mm^3) FLAIR images of size 512×512×512 [29] were preprocessed through N4 bias field correction (https://www.slicer.org), brain extraction from T1 images using the volBrain tool (http://volbrain.upv.es), and then the resulting brain masks were multiplied with bias-corrected FLAIR images to remove the skulls.

The 15 cases of preprocessed (nonlocal means (NLM) filtering, rigid registration, skull stripping, and N4 bias field correction) MSSEG 2016 training data, as provided in the challenge website, were used in this study [6]. We sampled the 3D FLAIR images in the slice direction of each volume (with linear interpolation for FLAIR and nearest-neighbor for the ground-truth lesion and brain masks) to obtain isotropic-voxel (1×1×1 mm^3) data. After sampling, images were transformed into the new sizes, including 315×512×512, 154×224×256, and 245×336×336. The consensus segmentation obtained from all seven delineations (by seven expert raters) was used as the ground truth for the performance evaluation of all methods implemented in this study [6].

We min-max normalized the images of both databases before all experiments, and also kept the same input parameters during the experiments, such as $b = 18$ and $a_{max} = 8$ for the $\mathcal{O}(a_{max}N)$ and FFT-based methods, whereas the $\mathcal{O}(N)$ method dealt with all similar parameters, except a_{max}; instead we had $b' = 8$. We chose $n = \infty$ for the FFT methods and $n = 2$ for the other two methods, but eventually following a similar approach to find a_{max} in the FFT and $\mathcal{O}(a_{max}N)$ algorithms.

b. Performance Evaluation on Clinical MRI Data

The performance of all methods applied for lesion detection were quantified with respect to the expert-made manual annotation of MS lesions in MRI volumes by evaluating the popular statistical metrics such as the true positive fraction

(TPF, or sensitivity) and false positive fraction (FPF, or false discovery rate) [7] as defined below,

$$TPF = \frac{|L_A \cap L_M|}{|L_M|};\ FPF = \frac{|L_A \cap \neg L_M|}{|L_A|}, \tag{36}$$

where, L_A and L_M are the label volumes corresponding to automatic detection and manual annotation, respectively. A TPF value close to 1 signifies that all voxels representing lesions in the annotated ground truth are correctly detected, while the 0 value indicates absolutely no detection. Conversely, An FPF value of 1 means no detection, and a value of 0 means that all detected voxels were correctly inside the label.

We now discuss how we used the above metrics to evaluate the performance of the $\mathcal{O}(a_{max}N)$, $\mathcal{O}(N)$, and FFT approaches. We first computed the a-maximized NCCC from the FLAIR volume using each method. Many methods in the literature threshold the NCCC to keep putative lesion voxels, on which various performance metrics are evaluated [8, 15, 18]. In this work, instead of selecting a threshold, we considered N_{Top} detected voxels with the highest NCCC value. We found the voxel with maximal NCCC from the resulting a-maximized NCCC volume, masked a sphere around it with a radius twice the optimal value of a, updated the NCCC volume by filling the sphere with the value of -∞, then repeated this process to find all N_{Top} maxima.

Next, as described in Algorithm 1, we created a sphere of all ones with the radius 1.61 times the optimally detected a, centered at the location of top detection, inside an all-zero volume, M, with the same size as the input image. We compared M with the ground-truth label to calculate the TPF and FPF for the case of $N_{Top} = 1$. We then added an enhancing sphere in M for the next detection to compute the TPF and FPF for $N_{Top} = 2$ and so on.

Algorithm 1: Performance evaluation of algorithms for automatic detection of brain lesions from images
Input: - Sets of coordinates, $\mathbf{c}_i$, with corresponding detected radii, a_i, $i = 1, \ldots, 5000$, having maximum NCCC recorded from top 5000 detection.
- Manually annotated ground-truth binary image (L_M).
Output: TPF and FPF
Create an all-0 volume (M) of equal size to the input image.
for $i = 1, \ldots, 5000$ **do**
1. Create an enhancing sphere (E) at $\mathbf{c}_i$ with the radius $1.61a_i$
2. Update M by performing logical 'OR' operation; $M \leftarrow M \vee E$.
3. Measure the overlap between the automatic and manual label via the logical 'AND' operation; $M \wedge L_M$.
4. Calculate TPF and FPF for $N_{Top} = i$.
end for

The mean TPF vs. FPF graphs over the 20 subjects of MSLS 2008 and the 15 subjects of MSSEG 2016 are presented in Fig. 2 for the three methods. We also recorded the computation times and resident memory usage of the three algorithms while running them over the real data as summarized in Table I. The runtimes and required memory while executing the entire MATLAB scripts (termed as *Algorithm*) and those of the script till before the NCCC peak detection step (termed as *NCCC*), are shown in Table I. All the experiments over real data were performed in the same computational setup as used in the synthetic data analysis. The MATLAB scripts of all three methods were run only once on each image. The visualization of the top 30 detected lesions resulting from all three template matching algorithms are represented in Figs. 3-4.

TABLE I
SUMMARY OF COMPUTATIONAL ANALYSIS OVER REAL MRI DATABASE

SUMMARY			FFT	$\mathcal{O}(a_{max}N)$	$\mathcal{O}(N)$
MSLS 2008	CPU TIME (MIN.)	ALGORITHM	194 ± 52	147 ± 15	163 ± 18
		NCCC	16.6 ± 0.3	8.8 ± 0.8	17.0 ± 0.8
	WALL TIME (MIN.)	ALGORITHM	195 ± 52	180 ± 15	164 ± 18
		NCCC	23 ± 3	21 ± 1	19 ± 1
	RES. MEM. (GB)	ALGORITHM	17.1 ± 0.4	12.14 ± 0.02	24 ± 1
		NCCC	16.8 ± 0.8	9.91 ± 0.03	23.7 ± 0.9
MSSEG 2016	CPU TIME (MIN.)	ALGORITHM	45 ± 37	48 ± 39	46 ± 38
		NCCC	5 ± 4	3 ± 2	5 ± 4
	WALL TIME (MIN.)	ALGORITHM	45 ± 37	49 ± 39	47 ± 38
		NCCC	5 ± 4	3 ± 2	8 ± 5
	RES. MEM. (GB)	ALGORITHM	6 ± 4	4 ± 3	7 ± 5
		NCCC	5 ± 4	3 ± 2	7 ± 6

c. Comparison with a State-of-the-Art Technique

We compared our methods with the lesion prediction algorithm (LPA) [5], which is a logistic-regression based supervised learning technique (fundamentally different from our methods) and requires only the FLAIR image with no input parameters for MS lesion detection. We performed binary thresholding over the resulting volume of lesion probability map to compute the TPF and FPF. To maintain the parity with the proposed method in the performance evaluation, we chose the threshold in between 0 to 1 with an interval of 2.0×10^{-4}. The LST toolbox (https://www.applied-statistics.de/lst.html) was utilized in statistical parametric mapping (SPM12; https://www.fil.ion.ucl.ac.uk/spm) for the implementation of LPA. The mean TPF vs. FPF plots over the MS lesion databases are illustrated in Fig. 2. We can see the better performance of LPA especially on the MSSEG 2016 data compared to the proposed and all other methods used in this study.

IV. DISCUSSION

This paper focuses on fast template-matching based automatic brain lesion detection from MRI volumes. The proposed methods have complexities of $\mathcal{O}(N)$ and $\mathcal{O}(a_{max}N)$, compared to $\mathcal{O}(a_{max}N \log N)$ of the FFT approach. From the results of time complexity experiment over synthetic data (see Fig. 1(a-b)), it is clear that $\mathcal{O}(N)$ method is the fastest of the three template-matching algorithms for sphere detection. We derived a new analytical solution to obtain the voxel-wise optimal template radius from local image statistics. The proposed analytical solution is a computational alternative to exhaustive search for the template size. The $\mathcal{O}(N)$ method could detect the bright spheres at their exact centers in the synthetic data (after using the appropriate multiplier). On the other hand, the FFT and $\mathcal{O}(a_{max}N)$ based template matching with exhaustive search possess similar accuracy for detection of the bright spheres as expected. The only small deviation persisting between them for the prediction of the radius of the sphere is due to employment of approximated Gaussian template in the $\mathcal{O}(a_{max}N)$ technique. The results on synthetic data validates the theoretical prediction of achieving the linear complexity in template matching by our methods (Fig. 1(a-b)). Regarding the resident memory usage, the $\mathcal{O}(a_{max}N)$ method appears to be quite efficient holding less memory than other two methods. The $\mathcal{O}(N)$ method, however, uses high memory. The

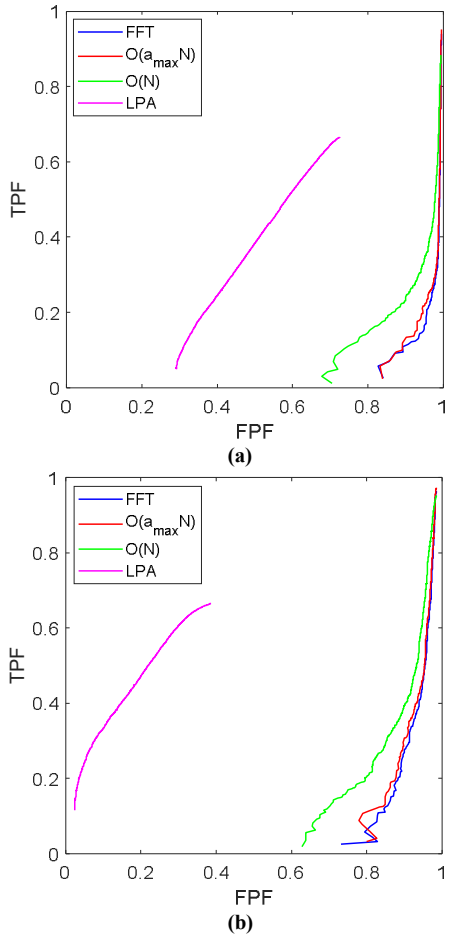


Fig. 2. Performance analysis (TPF vs. FPF) of proposed $\mathcal{O}(N)$ (green), $\mathcal{O}(a_{max}N)$ (red), and other FFT-based (blue) and LPA (magenta) approaches over real MRI image database of (a) MSLS 2008 and (b) MSSEG 2016.

analytically optimized template radius can be computed from mean and SD computed from the local 3D neighborhood, requiring large memory. In the exhaustive-search cases, we performed *loop* operation inside the MATLAB script to execute the similarity measure computation step for a set of template radii. We reduced the memory requirement by updating NCCC with the voxel-wise maximized values, thereby avoiding the storage of a separate NCCC volume for each tried radius. Fig. 1(c) shows that that the memory requirement increases linearly with the number of voxels in the volumes. In case of real data, the results follow a similar trend. The FFT method requires more memory than $\mathcal{O}(a_{max}N)$ does, because during template matching with the set of varying radii (*loop* operation) in the FFT method, we need to create a convolution kernel (template) that is as large as the input image, for every tried radius (so they can be multiplied elementwise in the frequency domain).

Regarding the runtimes of the methods on real MR images, Table I shows that the CPU time of the $\mathcal{O}(N)$ algorithm is higher than that of the $\mathcal{O}(a_{max}N)$ method, and close (till NCCC computation) to that of the FFT-based template matching on the MSLS 2008 data, and that $\mathcal{O}(a_{max}N)$ is the fastest of the three on the MSLS 2008 data. These results, which contradict those on the synthetic data, are because NCCC was computed for a set of only 8 different radii for both the $\mathcal{O}(a_{max}N)$ and FFT methods and this *loop*-based operation was proven to be faster than the computation of 10 more convolutions for optimizing template radius in the $\mathcal{O}(N)$ algorithm. On the contrary, for a large set of template radii as considered in the experiments on synthetic data, the $\mathcal{O}(N)$ method outperformed the others. The CPU times of the MSSEG 2016 experiments were very close for the three algorithms, since for small images, the gap between $N \log N$ and N becomes narrow. Overall, the $\mathcal{O}(N)$ method is expected to run faster for large MRI volumes, or for the detection of larger lesions. *Ex vivo* imaging of human brain, which has attracted researchers to explore microstructural neuroanatomy, can produce brain images with $(0.1\text{mm})^3$ isometric voxel resolution [27]. As the resolution increases, the ROI volume size increases proportionally, necessitating the reduction in computational complexity of the algorithms for processing very large MRI volumes to automatically detect lesions. Note that in real clinical scenario, we do not need to test the detection for top 5000 lesions, instead it would be for few numbers depending on the characteristics of the lesions and the clinician's needs. Also note that the computation of the a-maximized NCCC can be seen to take up only a small fraction of the entire runtime (see Table I), and has more variability.

Prior studies on MS lesions, as discussed in the next section, however, generally segment the lesions and report the detection accuracy from their automatic delineation. Therefore, our approach of lesion detection is different from most available tools. During the performance evaluation, we observed the monotonically increasing characteristics of both TPF and FPF with N_{Top} for all three methods (except LPA). Thus, a high sensitivity can be achieved with large N_{Top}, which can be interpreted quite easily from Fig. 2. The mean TPF vs. FPF plots (Fig. 2) show very similar performance by FFT and our $\mathcal{O}(a_{max}N)$ methods over both MRI databases, similar to synthetic data. We can also infer from Fig. 2 that the performance graph of the $\mathcal{O}(N)$ method is slightly better than FFT and $\mathcal{O}(a_{max}N)$, making this method a coherent assistive tool. In addition, the practicality of linear-time template matching can be well appreciated in processing of large 3D volumes with a high b (e.g., b = 50 showed the efficiency of $\mathcal{O}(N)$ in synthetic data). Throughout the experiment with the $\mathcal{O}(a_{max}N)$ and $\mathcal{O}(N)$ methods, we used an approximated 3D Gaussian template, which is almost radially symmetric, and expected to work robustly with spherical or close-to-spherical lesions. On the contrary, the MS lesions are arbitrary in shape and size (largely varying) [28]. Therefore, the algorithms might not detect the exact centers of such arbitrarily shaped lesions in FLAIR images. This reduced the overlap between the detected region and the ground truth, even when the lesions were correctly localized. This may be one of the reasons behind the rise of false positives. In the context of having large false positives, we would like to mention the work of Cabezas *et al.* [7]. Their unsupervised methodology deeply depends on tissue priors to estimate the mean and SD of the gray matter (GM) distribution in FLAIR images, for defining a threshold in order to segment the MS lesions. They reported a mean FPF higher than 90% (with close to 40% of mean TPF), as accuracy of lesion detection, without using any false detection refinement step. So, the presence of high FPF seems to be common among the performance of the algorithms, especially targeted to lesion

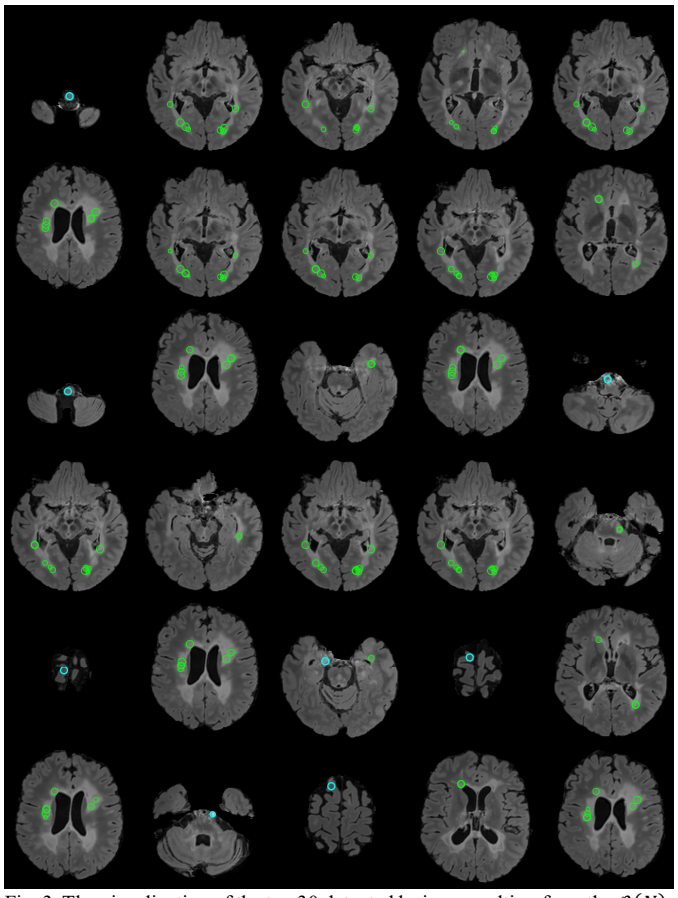


Fig. 3. The visualization of the top 30 detected lesions resulting from the $\mathcal{O}(N)$-based template matching algorithm on the first case of MSSEG 2016 training data. The green and cyan circles indicate the true and false positives, respectively. Thick and thin circles are detected in the shown and nearby slices, respectively.

detection from MS subjects, which we will discuss more in next section. Contrary to the unsupervised approaches, the prior knowledge learnt by LPA during the training phase seems to have been effective in improving the performance of this supervised-learning method. This analysis helps in tallying the performance of supervised and unsupervised techniques (Fig. 2). In template-matching works (see Section I), a general trend is to fix a threshold over the spectrum of NCCC values so as to evaluate the performance. However, the optimal NCCC threshold may depend on acquisition parameters, and fixing it might lead to the detection performance strongly depending on the threshold. In contrast, our strategy of using maximum NCCC from N_{Top} detection is not threshold-dependent.

A. Visualization of the Detected Lesions

We display the results for top 30 out of 5000 detections in Figs. 3-4. Blue, red, and green circles represent the FFT-, $\mathcal{O}(a_{max}N)$-, and $\mathcal{O}(N)$- based methods, respectively, on the 2D axial slices of the FLAIR volumes. We present the slices in order of lesion detection. (There may be slice repetition due to multiple detections in the same slice.) Thick circles indicate that the centers of the detected spheres are located in the shown slice. Thin circles show the intersections between the shown slice and the spheres representing top-30 detected lesions centered in nearby slices. Therefore, the lack of a circle around a lesion means that it has not been in the top 30 (but the algorithm may still have detected it). A limitation that we observed for all techniques is the multiple detection of parts of a large lesion, such as those scattered around the lateral ventricles or on the corpus callosum are common in MS [28]. When lesions of various sizes and shapes appear mainly in the white matter (WM) [28], multiple lesions were detected over the same large lesion because of the small b that we used. This issue may not be expected to arise in metastatic-lesion detection. The selection of a large b, on the other hand, may cause the algorithm to miss smaller lesions. We can observe the presence of false lesions (cyan circles in Figs. 3-4) included in the top 30 detection for all methods. False positives are mostly in the GM, which is hyper-intense in FLAIR images and sometimes confused by the algorithms with lesion. This is the main cause of high FPF in template matching algorithms. We also found false positives in other brain regions such as the splenium of corpus callosum, hippocampus, and medulla oblongata. The estimated lesion radius in the $\mathcal{O}(N)$ method is comparatively less than FFT- and $\mathcal{O}(a_{max}N)$-based methods, which is expected from the ground-truth-to-detected ratio radius (1.96 ± 0.11) obtained on synthetic data experiment.

B. Comparison with Existing Literature

In this section, we compare the performance among the existing state-of-the-art methods dedicated for MS lesion segmentation with MSLS 2008 data. We have considered some of the well performing techniques of the last decade (summarized in Tables II and III). Challenge organizers evaluated true positive rate (TPR) and false positive rate (FPR) (with the same definition as TPF and FPF used here) to determine the lesion detection ability [29] on the test data. Note that the FPR is defined differently in [30] from its standard definition. The lesion voxels are only a small fraction of the MRI volume, thus the standard formula makes the FPR negligible (impactful for small N_{Top}) due to uninformatively high true negatives. In this study, our method was evaluated on the MSLS 2008 training data with respect to the BCH ground-truths only (provided in Table III) due to the unavailability of UNC (University of North Carolina, NC, USA) delineations for all 20 cases. The positive predictive value (PPV) signifies the ratio of the number of true detections to all detections [31].

TABLE II
COMPARISON OF METHODS ON MICCAI MSLS TRAINING DATA [32].

Group	Method	MRI sequence	Performance	
			TPR	PPV
Souplet *et al.* [33]	EM algorithm followed by binary thresholding	T1, T2, FLAIR	21.0	30.0
Geremia *et al.* [34]	Context-rich Random Forest (RF)	T1, T2, FLAIR	40.0	40.0
Weiss *et al.* [35]	Unsupervised dictionary learning and sparse coding	FLAIR	33.0	37.0
Brosch *et al.* [36]	Convolutional encoder network	T1, T2, FLAIR	39.71	41.38
Manjon *et al.* [32]	Ensemble of neural networks	FLAIR	45.0	47.0

In this study, we intended to develop an algorithm for fast and automatic detection of brain lesions, as opposed to determining lesion contours (but still estimating the radius of the detected lesions). In most of the existing works, multi-contrast MR images were considered to delineate lesions, with

the exception of some methods that used only FLAIR images (like we did). Using multiple MRI sequences is attractive as it provides a larger feature space than using a single sequence. Weiss *et al.* and Manjon *et al.* avoided computing the whole image to possibly reduce the computational burden, instead they downsampled the image [35] or used candidate ROIs [32]. We observe from Table II that both TPR and PPV are low for all methods, which emphasizes the presence of high false positive with poor lesion detection ability. Table III presents a review of the performance of existing techniques on the test dataset. Note that the values presented in table were retrieved from the leaderboard results (http://www.ia.unc.edu/MSseg/results_table.php). This helps to comprehend that obtaining high FPR is very common among the methods even when the computation is done with multi-contrast MRI. We found that MS lesions were detected as regions with outlier intensity values compared to the healthy brain tissue in many works [37, 38]. Some research groups designed two-stage segmentation tasks; they first determine the putative lesion voxels with the help of various techniques such as unsupervised mixture models [39], estimation of lesion and tissue label using Markov random field (MRF) [40], cascaded convolutional neural network (CNN) [41], etc., and then segment the lesions by classifying the voxels. In another two-step approach, brain tissues, mainly WM, GM, and CSF, were segmented using different techniques such as atlas-based EM [33], Gaussian mixture model (GMM) [42], SPM8/12 segmentation algorithm [43], etc., before delineating the lesion while making use of the segmented tissues. These multistage approaches generally require multi-contrast MRI. Our proposed techniques compute the image at the original resolution and no spatial prior information on the tissue or lesion or candidate ROIs needs to be extracted; instead, it can directly localize the lesion region based on the NCCC measures, making it a single-stage method. Another key point we observed in the existing literature is the inclusion of post-processing operations, which is a common practice to refine the results and reduce false positives. The presence of flow artifacts in the FLAIR sequence creates hyperintensity in the CSF space, thus sometimes mimicking the lesion [44, 49]. In addition, the hyperintense border voxels at the GM-WM junction may result in some false positives [44, 50]. As a remedy, various post-processing operators have been adopted, such as morphological operations [33], setting a minimum lesion volume [43, 44, 51], connectivity rules, neighborhood similarity [43], etc., to reduce false detections and improve FPR and PPV. In this study, we did not perform any post-processing, and only evaluated the raw outcome of the proposed algorithms.

The performance summary of few of the existing works on 15 training images of MSSEG 2016 is presented in Table IV. In this challenge, the lesion detection capability was evaluated using F1-score, which can be defined from lesion sensitivity (SE_L) and lesion PPV (PPV_L) as, $F1 = 2\times SE_L\times PPV_L /(SE_L+PPV_L)$ [6]. From the table, we see different groups explored both multi-channel and single FLAIR MRI for processing. Beaumont *et al.* [51] did not consider lesions appearing at the edges of the brain mask, not dominantly located in the WM, etc., for final refinement of lesion boundaries. Knight and Khademi [52] showed 3D Gaussian filtering with 0.5 SD to be efficient for noise removal in FLAIR. In this work, we also used the Gaussian filter, however with a higher SD (2.0) in computing local image statistics. They also used three false-positive reduction strategies based on lesion volume, distance from the edge of the brain, and distance from midline of the brain. In other works [53, 54], handcrafted features and post-processing were used for MS lesion segmentation. Most groups favored working with multi-contrast MRI [6] for either designing machine learning algorithms or obtaining tissue prior, and followed with rule-based or morphological-operator-driven post-processing. From Tables II-IV, the sensitivity of lesion detection seems moderate with a high false detection. The organizers of MSSEG 2016, Commowick *et al.* [6], summarized that the deep-learning based approaches showed promise in narrowing the gap between manual and automatic segmentation of MS lesions. As our proposed method is a fully automatic unsupervised technique, the comparison with machine learning (deep neural networks) is not so informative due to the fundamental differences with supervised methods.

TABLE III
COMPARISON OF METHODS ON MICCAI MSLS 2008 TEST DATA.

Group	Method	MRI sequence	Performance	
			TPR	FPR
Geremia *et al.* [34]	Context-rich RF	T1, T2, FLAIR	58.08	70.01
Sudre *et al.* [44]	Model selection via hierarchical GMM	T1, FLAIR	24.6	18.2
Guizard *et al.* [45]	Rotation-invariant NLM	T2, FLAIR	52.7	42.0
Jesson & Arbel [40]	MRF and regional RF	T1, T2, FLAIR	53.5	24.2
Tomas-Fernandez and Warfield [46]	Local and global GMM tissue model	T1, T2, FLAIR	51.8	45.1
Roura *et al.* [43]	SPM8 and FLAIR thresholding	T1, FLAIR	55.4	40.5
Brosch *et al.* [47]	Deep 3D convolutional encoder	T1, T2, FLAIR, PD	56.0	49.8
Strumia *et al.* [48]	Geometric brain model	T1, FLAIR	42.9	30.5
Valverde *et al.* [41]	Cascaded CNN	T1, T2, FLAIR	68.7	46.0
		T1, FLAIR	60.2	41.8
Ghribi *et al.* [42]	Atlas-based GMM and lesion expansion	T1, FLAIR	84.0	69.7
Manjon *et al.* [32]	Ensemble of neural networks	FLAIR	68.0	68.6

TABLE IV
COMPARISON OF METHODS ON MICCAI MSSEG 2016 TRAINING DATA.

Group	Method	MRI sequence	Performance		
			SE_L	PPV_L	F1-score
Beaumont *et al.* [51]	Multimodal graph cut, EM, and post-processing	T1, T2, FLAIR	0.35	0.52	0.38
Beaumont *et al.* [55]	Voxel-wise comparison and post-processing	FLAIR, T2 ∩ FLAIR	0.45	0.28	0.32
		FLAIR	0.53	0.18	0.25
Knight & Khademi [52]	Fuzzy classification, thresholding, post-processing	FLAIR	0.53	0.80	-
Mahbod *et al.* [53]	Multilayer perceptron with morphology-based filtering	3D FLAIR	-	-	-
			Dice: 0.10–0.84		
Vera-Olmos *et al.* [54]	RF classifier and MRF based post-processing	T1, FLAIR	0.68	-	-
			FPR = 0.48		

In summary, the proposed template matching algorithms do not require tissue prior or any lesion-specific information. They are fully automatic and unsupervised models that can detect lesions from the FLAIR sequence with a lower computational complexity than the conventional FFT-based template

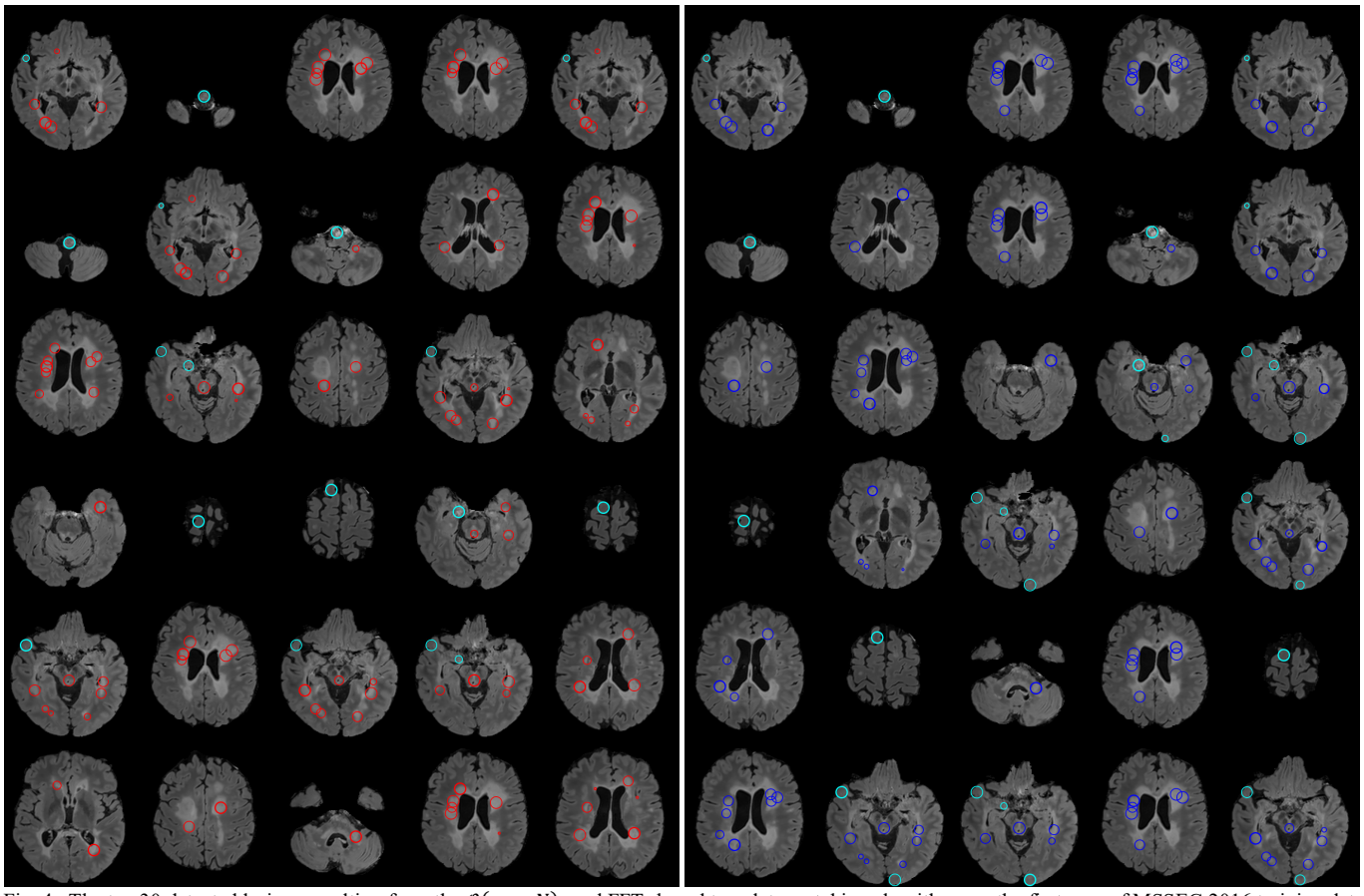


Fig. 4. The top 30 detected lesions resulting from the $\mathcal{O}(a_{max}N)$- and FFT- based template matching algorithms on the first case of MSSEG 2016 training data. The red ($\mathcal{O}(a_{max}N)$) and blue (FFT) circles indicate the true lesions, and the false lesions are identified with cyan colored circles. Thick lesions are those centered at the shown slice, whereas thin circles represent the intersections with other detected spheres in the nearby slices.

matching. We also did not apply any post-refinement approaches to remove the false positives; however, false lesions are ruled out during the review by the clinician. In lesion detection, we give a higher priority to avoiding false negatives than false positives.

V. Conclusion

We have proposed a new mathematical framework to compute a radius-optimized normalized cross-correlation coefficient (NCCC) similarity measure for 3D template matching, with application for automatic lesion detection. The key feature of our method is its low *linear-time* computational complexity, allowing for fast detection. Our alternative way to compute the NCCC exploits the approximation of Gaussian smoothing with multiple convolutions with a D-dimensional box kernel. We further proposed an analytical solution for the template radius that replaces the costly exhaustive search, thereby achieving the computational complexity of $\mathcal{O}(N)$, which is not possible with the FFT-based method. The proposed technique provides a new approach to implementing fast template matching for object detection from images. We tested our lesion-detection method on synthetic images as well as two real MRI databases of multiple sclerosis lesions, and compared its performance with existing state-of-the-art methods and the conventional FFT-based template matching algorithm. Future work includes extending our approach for anisotropic-voxel images, and also imaging modalities beyond MRI.

References

[1] C. Clinic. (2019, Sept. 12, 2019). *Brain Lesions*. Available: https://my.clevelandclinic.org/health/diseases/17839-brain-lesions

[2] WebMD. (2019, Sep. 12, 2019). *Brain Lesions: Causes, Symptoms, Treatments*. Available: https://www.webmd.com/brain/brain-lesions-causes-symptoms-treatments#1

[3] C. Egger, R. Opfer, C. Wang, T. Kepp, M. P. Sormani, L. Spies, *et al.*, "MRI FLAIR lesion segmentation in multiple sclerosis: Does automated segmentation hold up with manual annotation?," *NeuroImage: Clinical,* vol. 13, pp. 264-270, 2017.

[4] P. Schmidt, C. Gaser, M. Arsic, D. Buck, A. Förschler, A. Berthele, *et al.*, "An automated tool for detection of FLAIR-hyperintense white-matter lesions in multiple sclerosis," *NeuroImage,* vol. 59, pp. 3774-3783, 2012.

[5] P. Schmidt, "Bayesian inference for structured additive regression models for large-scale problems with applications to medical imaging," PhD thesis, Ludwig-Maximilians-Universität München, 2017.

[6] O. Commowick, A. Istace, M. Kain, B. Laurent, F. Leray, M. Simon, *et al.*, "Objective evaluation of multiple sclerosis lesion segmentation using a data management and processing infrastructure," *Scientific Reports,* vol. 8, p. 13650, 2018.

[7] M. Cabezas, A. Oliver, E. Roura, J. Freixenet, J. C. Vilanova, L. Ramió-Torrentà, *et al.*, "Automatic multiple sclerosis lesion detection in brain MRI by FLAIR thresholding," *Comput. Methods Programs Biomed.,* vol. 115, pp. 147-161, 2014.

[8] R. D. Ambrosini, P. Wang, and W. G. O'Dell, "Computer-Aided Detection of Metastatic Brain Tumors Using Automated 3-D Template Matching," *J. Magn. Reson. Imaging,* vol. 31, pp. 85-93, 2010.

[9] S. Koley, C. Chakraborty, C. Mainero, B. Fischl, and I. Aganj, "A Fast Approach to Automatic Detection of Brain Lesions," in *Proc. MICCAI 2016 BrainLes Workshop*, 2016, pp. 52-61.

[10] R. Narasimha, I. Aganj, A. E. Bennett, M. J. Borgnia, D. Zabransky, G. Sapiro, *et al.*, "Evaluation of denoising algorithms for biological electron tomography," *Journal of Structural Biology,* vol. 164, pp. 7-17, 2008.

[11] G. D. Tourassi, R. Vargas-Voracek, D. M. Catarious, and C. E. Floyd, "Computer-assisted detection of mammographic masses: A template matching scheme based on mutual information," *Medical Physics,* vol. 30, pp. 2123-2130, 2003.

[12] K. Lochanambal, M. Karnan, and R. Sivakumar, "Identifying masses in mammograms using template matching," in *Proc. Second Int. Conf. Communi. Soft. Net.*, 2010, pp. 339-342.

[13] A. A. Farag, A. El-Baz, G. Gimel'farb, and R. Falk, "Detection and recognition of lung abnormalities using deformable templates," in *Proc. 17th Int. Conf. Pattern Recognit. 2004. ICPR 2004.*, 2004, pp. 738-741.

[14] O. Osman, S. Ozekes, and O. N. Ucan, "Lung nodule diagnosis using 3D template matching," *Comput. Biol. Med.,* vol. 37, pp. 1167-1172, 2007.

[15] P. Wang, A. DeNunzio, P. Okunieff, and W. G. O'Dell, "Lung metastases detection in CT images using 3D template matching," *Med. Phys.,* vol. 34, pp. 915-922, 2007.

[16] J. H. Moltz, M. Schwier, and H.-O. Peitgen, "A general framework for automatic detection of matching lesions in follow-up CT," in *Proc. IEEE ISBI*, 2009, pp. 843-846.

[17] S. K. Warfield, M. Kaus, F. A. Jolesz, and R. Kikinis, "Adaptive, template moderated, spatially varying statistical classification," *Med. Image Anal.,* vol. 4, pp. 43-55, 2000.

[18] R. Farjam, H. A. Parmar, D. C. Noll, C. I. Tsien, and Y. Cao, "An approach for computer-aided detection of brain metastases in post-Gd T1-W MRI," *Magn. Reson. Imaging,* vol. 30, pp. 824-836, 2012.

[19] S. Yang, Y. Nam, M.-O. Kim, E. Y. Kim, J. Park, and D.-H. Kim, "Computer-aided detection of metastatic brain tumors using magnetic resonance black-blood imaging," *Investigative Radiology,* vol. 48, pp. 113-119, 2013.

[20] X.-F. Wang, J. Gong, R.-R. Bu, and S.-D. Nie, "A 3D adaptive template matching algorithm for brain tumor detection," in *Proc. Int. Conf. Life Sys. Model. Simu.*, 2014, pp. 50-61.

[21] A. Muñoz, R. Ertlé, and M. Unser, "Continuous wavelet transform with arbitrary scales and O (N) complexity," *Signal Processing,* vol. 82, pp. 749-757, 2002.

[22] H. Hou and H. Andrews, "Cubic splines for image interpolation and digital filtering," *IEEE Trans. Acoust., Speech, Signal Process,* vol. 26, pp. 508-517, 1978.

[23] http://mathworld.wolfram.com/SincFunction.html, "Sinc Function," 2016.

[24] S. Kogan, "A note on definite integrals involving trigonometric functions," *Available in http://www. mathsoft. com/asolve/constant/pi/sin/sin. html,* 1999.

[25] J. W. Cooley and J. W. Tukey, "An Algorithm for the Machine Calculation of Complex Fourier Series," *Math. Comput.,* vol. 19, pp. 297-301, 1965.

[26] B. Johnston, M. S. Atkins, B. Mackiewich, and M. Anderson, "Segmentation of multiple sclerosis lesions in intensity corrected multispectral MRI," *IEEE Trans. Med. Imag.,* vol. 15, pp. 154-169, 1996.

[27] B. L. Edlow, A. Mareyam, A. L. Horn, J. Polimeni, M. D. Tisdall, J. Augustinack*, et al.*, "7 Tesla MRI of the ex vivo human brain at 100 micron resolution," *BioRxiv,* p. 649822, 2019.

[28] F. Barkhof and P. Scheltens, "Imaging of white matter lesions," *Cerebrovascular Diseases,* vol. 13, pp. 21-30, 2002.

[29] M. Styner, J. Lee, B. Chin, M. S. Chin, O. Commowick, H.-H. Tran*, et al.*, "3D Segmentation in the Clinic: A Grand Challenge II: MS lesion segmentation," in *Proc. MICCAI-Multiple Sclerosis Lesion Segmentation Challenge Workshop,*, New York, NY, USA, 2008.

[30] D. S. Burke, J. F. Brundage, R. R. Redfield, J. J. Damato, C. A. Schable, P. Putman*, et al.*, "Measurement of the false positive rate in a screening program for human immunodeficiency virus infections," *N. Engl. J. Med.,* vol. 319, pp. 961-964, 1988.

[31] R. H. Fletcher, S. W. Fletcher, and G. S. Fletcher, *Clinical epidemiology: the essentials*: Lippincott Williams & Wilkins, 2012.

[32] J. V. Manjón, P. Coupé, P. Raniga, Y. Xia, P. Desmond, J. Fripp*, et al.*, "MRI white matter lesion segmentation using an ensemble of neural networks and overcomplete patch-based voting," *Computerized Medical Imaging and Graphics,* vol. 69, pp. 43-51, 2018.

[33] J.-C. Souplet, C. Lebrun, N. Ayache, and G. Malandain, "An Automatic Segmentation of T2-FLAIR Multiple Sclerosis Lesions," in *Proc. MICCAI-Multiple Sclerosis Lesion Segmentation Challenge Workshop*, New York, NY, USA, 2008.

[34] E. Geremia, O. Clatz, B. H. Menze, E. Konukoglu, A. Criminisi, and N. Ayache, "Spatial decision forests for MS lesion segmentation in multi-channel magnetic resonance images," *NeuroImage,* vol. 57, pp. 378-390, 2011.

[35] N. Weiss, D. Rueckert, and A. Rao, "Multiple sclerosis lesion segmentation using dictionary learning and sparse coding," in *Proc. Int. Conf. MICCAI*, 2013, pp. 735-742.

[36] T. Brosch, Y. Yoo, L. Y. Tang, D. K. Li, A. Traboulsee, and R. Tam, "Deep convolutional encoder networks for multiple sclerosis lesion segmentation," in *Proc. Int. Conf. MICCAI*, 2015, pp. 3-11.

[37] K. Van Leemput, F. Maes, D. Vandermeulen, A. Colchester, and P. Suetens, "Automated segmentation of multiple sclerosis lesions by model outlier detection," *IEEE Trans. Med. Imag.,* vol. 20, pp. 677-688, 2001.

[38] M. Prastawa and G. Gerig, "Automatic MS lesion segmentation by outlier detection and information theoretic region partitioning," *Grand Challenge Work.: Mult. Scler. Lesion Segm. Challenge,* pp. 1-8, 2008.

[39] T. Jerman, A. Galimzianova, F. Pernuš, B. Likar, and Ž. Špiclin, "Combining unsupervised and supervised methods for lesion segmentation," in *Proc. MICCAI 2015 BrainLes Workshop*, Munich, Germany 2015, pp. 45-56.

[40] A. Jesson and T. Arbel, "Hierarchical MRF and random forest segmentation of MS lesions and healthy tissues in brain MRI," in *Proc. 2015 Longitudinal Multiple Sclerosis Lesion Segmentation Challenge,* pp. 1-2, 2015.

[41] S. Valverde, M. Cabezas, E. Roura, S. González-Villà, D. Pareto, J. C. Vilanova*, et al.*, "Improving automated multiple sclerosis lesion segmentation with a cascaded 3D convolutional neural network approach," *NeuroImage,* vol. 155, pp. 159-168, 2017.

[42] O. Ghribi, L. Sellami, M. B. Slima, A. B. Hamida, C. Mhiri, and K. B. Mahfoudh, "An Advanced MRI Multi-Modalities Segmentation Methodology Dedicated to Multiple Sclerosis Lesions Exploration and Differentiation," *IEEE Trans. Nanobiosci.,* vol. 16, pp. 656-665, 2017.

[43] E. Roura, A. Oliver, M. Cabezas, S. Valverde, D. Pareto, J. C. Vilanova*, et al.*, "A toolbox for multiple sclerosis lesion segmentation," *Neuroradiology,* vol. 57, pp. 1031-1043, 2015.

[44] C. H. Sudre, M. J. Cardoso, W. H. Bouvy, G. J. Biessels, J. Barnes, and S. Ourselin, "Bayesian model selection for pathological neuroimaging data applied to white matter lesion segmentation," *IEEE Trans. Med. Imag,* vol. 34, pp. 2079-2102, 2015.

[45] N. Guizard, P. Coupé, V. S. Fonov, J. V. Manjón, D. L. Arnold, and D. L. Collins, "Rotation-invariant multi-contrast non-local means for MS lesion segmentation," *NeuroImage: Clinical,* vol. 8, pp. 376-389, 2015.

[46] X. Tomas-Fernandez and S. K. Warfield, "A model of population and subject (MOPS) intensities with application to multiple sclerosis lesion segmentation," *IEEE Trans. Med. Imag.,* vol. 34, pp. 1349-1361, 2015.

[47] T. Brosch, L. Y. Tang, Y. Yoo, D. K. Li, A. Traboulsee, and R. Tam, "Deep 3D convolutional encoder networks with shortcuts for multiscale feature integration applied to multiple sclerosis lesion segmentation," *IEEE Trans. Med. Imag.,* vol. 35, pp. 1229-1239, 2016.

[48] M. Strumia, F. R. Schmidt, C. Anastasopoulos, C. Granziera, G. Krueger, and T. Brox, "White matter MS-lesion segmentation using a geometric brain model," *IEEE Trans. Med. Imag.,* vol. 35, pp. 1636-1646, 2016.

[49] K. H. Ong, D. Ramachandram, R. Mandava, and I. L. Shuaib, "Automatic white matter lesion segmentation using an adaptive outlier detection method," *Magn. Reson. Imaging,* vol. 30, pp. 807-823, 2012.

[50] T. Samaille, O. Colliot, D. Dormont, and M. Chupin, "Automatic segmentation of age-related white matter changes on flair images: Method and multicentre validation," in *Proc. IEEE ISBI*, 2011, pp. 2014-2017.

[51] J. Beaumont, O. Commowick, and C. Barillot, "Multiple sclerosis lesion segmentation using an automated multimodal graph cut," in *Proc. MICCAI MSSEG Challenge*, 2016, pp. 1-8.

[52] J. Knight and A. Khademi, "MS Lesion Segmentation Using FLAIR MRI Only," in *Proc. MICCAI MSSEG Challenge*, 2016, pp. 21-28.

[53] A. Mahbod, C. Wang, and O. Smedby, "Automatic multiple sclerosis lesion segmentation using hybrid artificial neural networks," in *Proc. MICCAI MSSEG Challenge*, 2016, pp. 29-36.

[54] F. Vera-Olmos, H. Melero, and N. Malpica, "Random forest for multiple sclerosis lesion segmentation," in *Proc. MICCAI MSSEG Challenge*, 2016, pp. 81-86.

[55] J. Beaumont, O. Commowick, and C. Barillot, "Automatic Multiple Sclerosis lesion segmentation from Intensity-Normalized multi-channel MRI," in *Proc. MICCAI MSSEG Challenge*, 2016, pp. 9-15.