

Combining Registration Errors and Supervoxel Classification for Unsupervised Brain Anomaly Detection*

Samuel Botter Martins^{1,2,3}[0000-0002-2894-3911], Alexandre Xavier Falcão¹[0000-0002-2914-5380], and Alexandru Cristian Telea⁴[0000-0003-0750-0502]

¹ Laboratory of Image Data Science (LIDS), University of Campinas, Brazil

² Bernoulli Institute, University of Groningen, the Netherlands

³ Federal Institute of São Paulo, Campinas, Brazil

⁴ Dep. of Information and Computing Sciences, Utrecht University, the Netherlands
{sbmmartins, afalcao}@ic.unicamp.br, a.c.telea@uu.nl

Abstract. Automatic detection of brain anomalies in MR images is challenging and complex due to intensity similarity between lesions and healthy tissues as well as the large variability in shape, size, and location among different anomalies. Even though discriminative models (supervised learning) are commonly used for this task, they require quite high-quality annotated training images, which are absent for most medical image analysis problems. Inspired by groupwise shape analysis, we adapt a recent fully *unsupervised* supervoxel-based approach (SAAD) — designed for abnormal asymmetry detection of the hemispheres — to detect brain anomalies from registration errors. Our method, called BADRESC, extracts supervoxels inside the right and left hemispheres, cerebellum, and brainstem, models registration errors for each supervoxel, and treats outliers as anomalies. Experimental results on MR-T1 brain images of stroke patients show that BADRESC outperforms a convolutional-autoencoder-based method and attains similar detection rates for hemispheric lesions in comparison to SAAD with substantially fewer false positives. It also presents promising detection scores for lesions in the cerebellum and brainstem.

Keywords: Brain Anomaly Detection · Supervoxel Segmentation · One-class Classification · Registration Errors · MRI.

1 Introduction

Quantitative analysis of MR brain images has been used extensively for the characterization of brain disorders, such as stroke, tumors, and multiple sclerosis. Such methods rely on delineating objects of interest — (sub)cortical structures or lesions to solve detection and segmentation simultaneously. Results are usually

*The authors thank CNPq (303808/2018-7), FAPESP (2014/12236-1) for the financial support, and NVIDIA for supporting a graphics card.



Fig. 1: Axial slices of three stroke patients from the ATLAS dataset [19] with lesions (ground-truth borders in red) that significantly differ in location, shape, and size. Figure referenced from [23].

used for tasks such as quantitative lesion assessment (*e.g.*, volume), surgical planning, and overall anatomic understanding [7, 17, 35]. Note that *segmentation* corresponds to the exact delineation of the object of interest, whereas *detection* consists of finding the rough location of such objects (*e.g.*, by a bounding box around the object), in case they are present in the image.

The simplest strategy to detect brain anomalies consists of a visual slice-by-slice inspection by one or multiple specialists. This process is very laborious, time-consuming, easily prone to errors, and even impracticable when a large amount of data needs to be processed. Continuous efforts have been made for automatic anomaly detection that delineates anomalies with accuracy close to that of human experts. However, this goal is challenging and complex due to the large variability in shape, size, and location among different anomalies, even when the same disease causes these (see, *e.g.*, Fig. 1). These difficulties have motivated the research and development of automatic brain anomaly detection methods based on *machine learning* algorithms.

Most automatic methods in the literature rely on supervised machine learning to detect or segment brain anomalies. They train a classifier from training images — which must be previously labeled (*e.g.*, lesion segmentation masks) by experts — to delineate anomalies by classifying voxels or regions of the target image. Traditional image features (*e.g.*, edge detectors and texture features) and deep feature representations (*e.g.*, convolutional features) are commonly used [3, 13, 30, 34, 35]. Some works propose a groupwise shape analysis based on estimating the deformation field between a target image and a template (reference image) after image registration [13, 34].

However, these methods commonly have three main limitations. First, they require a large number of high-quality annotated training images, which is absent for most medical image analysis problems [1, 15, 40]. Second, they are only designed for the lesions found in the training set. Third, some methods still require weight fine-tuning (retraining) when used for a new set of images due to image variability across scanners and acquisition protocols, limiting its application into clinical routine.

All the above limitations of supervised methods motivate research on *unsupervised* anomaly detection approaches [4, 8, 14, 23–25, 33]. From a training set

with images of *healthy* subjects *only*, these methods encode general knowledge or assumptions (*priors*) from healthy tissues, so that an outlier who breaks such general priors is considered anomaly [14]. As unsupervised brain anomaly detection methods do not use labeled samples, they are less effective in detecting lesions from a specific disease than supervised approaches trained from labeled samples for the same disease. For the same reason, however, unsupervised methods are generic in detecting any lesions, *e.g.*, coming from multiple diseases, as long as these notably differ from healthy training samples.

Since many neurological diseases are associated with abnormal brain asymmetries [43], an *unsupervised* method called Supervoxel-based Abnormal Asymmetry Detection (SAAD) [24] was recently proposed to detect abnormal asymmetries in MR brain images. SAAD presents a mechanism for asymmetry detection that consists of three steps: (i) it registers all images to the same symmetric template and then computes asymmetries between the two hemispheres by using their mid-sagittal plane (MSP) as reference; (ii) a supervoxel segmentation method, named SymmISF, is used to extract pairs of symmetric supervoxels from the left and right hemispheres for each test image, guided by their asymmetries. Supervoxels define more significant volumes of interest for analysis than regular 3D patches; and (iii) each pair generates a local one-class classifier trained on control images to find supervoxels with abnormal asymmetries on the test image. SAAD was further extended to detect abnormal asymmetries in the own native image space of each test image [25].

Although SAAD claims to obtain higher detection accuracy even for small lesions compared to state-of-the-art detection methods, its analysis is limited to asymmetric anomalies in the brain hemispheres, ignoring lesions in the cerebellum and brainstem. Moreover, if the same lesion is localized in both hemispheres roughly in the same position (*e.g.*, some cases of multiple sclerosis), it is not detected due to the lack of asymmetries.

Inspired by groupwise shape analysis, in this work, we present BADRESC, an *unsupervised* method for Brain Anomaly Detection based on Registration Errors and Supervoxel Classification in 3T MR-T1 images of the brain. After registering a target image to a standard template with *only* healthy tissues by deformable registration, BADRESC assumes that registration errors for anomalies are considerably different from the registration errors for healthy tissues. Thus, BADRESC adapts the SAAD framework as follows. First, it replaces the asymmetry maps with registration errors. A robust preprocessing is considered to improve the quality of image registration. Second, it then analyses four macro-objects of interest — right and left hemispheres, cerebellum, and brainstem — by extracting supervoxels for each one *separately*. Finally, each supervoxel generates a local one-class classifier for healthy tissues to detect *outliers* as anomalies.

This work is an extension of a previous one presented in [23], which originally introduced BADRESC. While considering the same macro-objects of interest and datasets, our contributions include:

- a more detailed explanation of BADRESC’s steps, especially supervoxel segmentation (Section 2 and Appendix A);

- an extended evaluation that considers another unsupervised baseline (like ours), which consists of a convolutional-autoencoder-based approach; and
- more evaluation metrics: Dice, mean recall, and three other false-positive metrics.

Experimental results on 3D MR-T1 brain images of stroke patients confirm the accuracy of BADRES-C to detect hemispheric lesions with only a few false positives. Additionally, BADRES-C presents promising results for the detection of lesions in the cerebellum and brainstem.

This paper is organized as follows. Section 2 introduces preliminary concepts on supervoxel segmentation and the considered framework used by BADRES-C. Section 3 presents BADRES-C. Section 4 presents experiments, while Section 5 reports and discusses results. Section 6 concludes the paper and discusses some possible future directions.

2 Iterative Spanning Forest (ISF)

One crucial step of our proposed method consists of extracting supervoxels inside each macro-object of interest for subsequent analysis. *Supervoxels* are groups of voxels with similar characteristics resulting from an oversegmentation of a 3D image or region of interest. We call *superpixels* for 2D images. They preserve intrinsic image information (*e.g.*, the borders of tissues and lesions) and are used as an alternative to patches to define more meaningful VOIs for computer-vision problems [37, 45] and some medical image applications [35, 44]. Supervoxels are a better alternative than 3D regular patches for our target problem, as (i) they better fit lesions and tissues, and (ii) their voxels contain minimum heterogeneous information.

In this work, we rely on the Iterative Spanning Forest (ISF) framework [42] for supervoxel segmentation. ISF is a recent approach for both superpixel and supervoxel segmentation that has shown higher effectiveness than several state-of-the-art counterparts, especially when used for 3D MR image segmentation of the brain [42]. ISF consists of three key steps: (i) seed sampling followed by multiple iterations of (ii) connected supervoxel delineation based on the image foresting transform (IFT) algorithm [11] (Appendix A), and (iii) seed recomputation to improve delineation. We next present the theoretical background for ISF as well as its algorithm.

2.1 Theoretical Background

Let the pair $\hat{I} = (D_I, \vec{I})$ be a d-dimensional multi-band image, where $D_I \subset Z^d$ is the *image domain*, and $\vec{I} : D_I \rightarrow \mathbb{R}^c$ is a mapping function that assigns a vector of c real-valued intensities $\vec{I}(p)$ — one value for each band (channel) of the image — to each element $p \in D_I$. For simplicity, assume that the term *voxels* represents the d-dimensional-image elements.

An image can be interpreted as a graph $G_I = (D_I, \mathcal{A})$, whose *nodes* are the voxels, and the *arcs* are defined by an *adjacency relation* $\mathcal{A} \subset D_I \times D_I$,

with $\mathcal{A}(p)$ being the *adjacent set* of a voxel p . In this work, we consider the 6-neighborhood adjacency for ISF. We refer to Appendix A for more details about image graphs, paths, and connectivity functions.

For a given *initial seed set* \mathcal{S} , labeled with consecutive integer numbers $\{1, 2, \dots, |\mathcal{S}|\}$, and a *connectivity function* f , ISF computes each supervoxel as a *spanning tree* rooted at a *single* seed, such that seeds compete among themselves by offering lower-cost paths to conquer their most strongly connected voxels. We use the following connectivity function f given by

$$f(\langle q \rangle) = \begin{cases} 0, & \text{if } q \in \mathcal{S}, \\ +\infty, & \text{otherwise,} \end{cases} \quad (1)$$

$$f(\pi_p \cdot \langle p, q \rangle) = f(\pi_p) + \left[\alpha \cdot \|\vec{I}(q) - \vec{I}(R(p))\| \right]^\beta + \|q - p\|,$$

where $\|\vec{I}(t) - \vec{I}(R(p))\|$ is the Euclidean distance between the intensity vectors at voxels $R(p)$ and q , $\|q - p\|$ the Euclidean distance between the voxels p and q , $\langle q \rangle$ is a trivial path, $\pi_p \cdot \langle p, q \rangle$ the extension of a path π_p with terminus q by an arc $\langle p, q \rangle$, and $R(p)$ the starting node (seed) of π_p . The factors α and β serve to control a compromise between supervoxel boundary adherence and shape regularity. Although the authors of ISF have fixed $\alpha = 0.5$ and $\beta = 12$ during the experiments [42], such factors are problem-dependent and should be optimized to yield more accurate supervoxels. Fig. 2 shows the impact of α and β for the superpixel segmentation of a 2D brain image.

2.2 The ISF Algorithm

Algorithm 1 presents a pseudo code for the Iterative Spanning Forest framework. At each iteration (Lines 2–4), ISF performs connected supervoxel delineation on the image I based on IFT (Line 3) — as described by Algorithm 2 (Appendix A) — from a given seed set \mathcal{S}' , adjacency relation \mathcal{A} , and the connectivity function f described by Eq. 1. The seed set at Iteration 1 is the initial seed set \mathcal{S} (Line 1). Next, the seed set is recomputed by the function *SeedRecomputation* to improve delineation (Line 4). This process continues until reaching N iterations. The algorithm returns the optimum-path forest (predecessor map), root map, path-cost map, and the supervoxel label map. Fig. 3 illustrates the execution of ISF.

In this work, we adopted a seed-recomputation strategy proposed by Vargas-Muñoz *et al.* [42], as detailed next. At each iteration, we promote the *centroids* from the obtained supervoxels — *i.e.*, their geometric centers — to be the seeds of the next iteration. If a given centroid c_i is out of its supervoxel L_i — due to the non-convex shape of L_i — we select the voxel of L_i that is the closest to c_i . We refer to Vargas-Muñoz *et al.* [42] for more specific details.

A crucial step for the success of ISF consists of performing a robust *initial seed estimation*. This step, however, is problem-dependent, so that simple and general strategies — *e.g.*, a grid sampling in the input image — can provide unsatisfactory results (*e.g.*, undersegmentating a lesion). Section 3.3 details our strategy to select the initial seed for our problem.

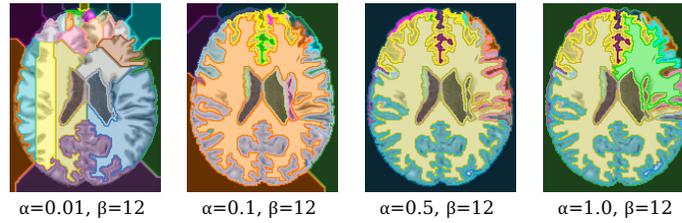
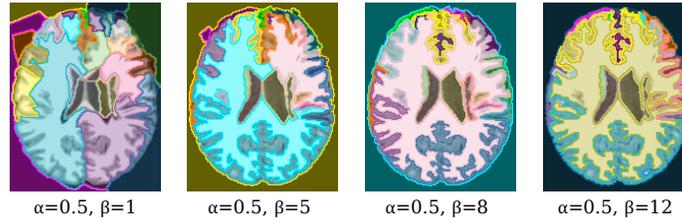
(a) Different values of α and $\beta = 12$.(b) $\alpha = 0.5$ and different values of β .

Fig. 2: The impact of the factors α and β for superpixel segmentation by ISF. Each superpixel is represented by a different color. For all cases, we performed ISF on the same 2D brain image with 10 iterations and identical 30 initial seeds selected by grid sampling.

Algorithm 1: Iterative Spanning Forest

Input: Image $\hat{I} = (D_I, \vec{I})$, adjacency relation \mathcal{A} , connectivity function f , initial seed set $\mathcal{S} \subset D_I$, and the maximum number of iterations $N \geq 1$.

Output: Optimum-path forest P , root map R , path-cost map C , and supervoxel label map L .

Aux: Seed set \mathcal{S}' , and the variable i .

```

1  $\mathcal{S}' \leftarrow \mathcal{S}$ 
2 for  $i \leftarrow 0$  to  $N - 1$  do
3    $(P, R, C, L) \leftarrow IFT(\hat{I}, \mathcal{A}, f, \mathcal{S}')$  /* see Alg. 2 */
4    $\mathcal{S}' \leftarrow SeedRecomputation(\hat{I}, \mathcal{S}', P, R, C)$ 
5 return  $(P, R, C, L)$ 

```

3 Description of BADRESC

Fig. 4 presents the pipeline of BADRESC which consists of five steps: 3D image preprocessing, image registration, registration error computation, supervoxel segmentation, and classification. We next describe all these steps to detect anomalies in the brain hemispheres, cerebellum, and brainstem.

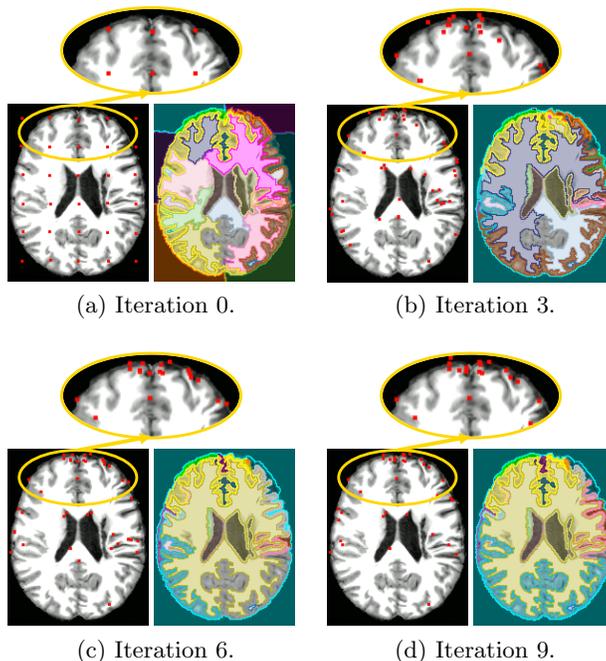


Fig. 3: Example of the ISF execution (10 iterations with $\alpha = 0.5$ and $\beta = 12$) on a 2D brain image. (a)–(d) Four iterations of ISF. For each iteration, we show its input seeds (red points) and the resulting obtained superpixels (each color represents a different superpixel). Iteration 0 shows the initial seed set obtained by grid sampling; the other seed sets are obtained by seed recomputation. As the insets show, most seeds do not change positions over iterations.

3.1 3D Image Preprocessing and Registration

MR images are affected by image acquisition issues such as noise and intensity heterogeneity. This makes their automated analysis very challenging since intensities of the same tissues vary across the image. To alleviate these and make images more similar to each other, we use typical preprocessing steps known in the literature [16, 21, 24], as shown in Fig. 5.

For each 3D image (Fig. 5a), we start performing noise reduction by median filtering, followed by MSP alignment, and bias field correction by N4 [41]. As voxels from irrelevant tissues/organs for the addressed problem (*e.g.*, neck and bones) can negatively impact the image registration and intensity normalization, we use AdaPro [22] to segment the regions of interest: right and left hemispheres, cerebellum, and brainstem (Fig. 5b).

To attenuate differences in brightness and contrast among images, we apply a histogram matching between the segmented images and the template. This operation only considers the voxels inside the regions of interest (Fig. 5d). We

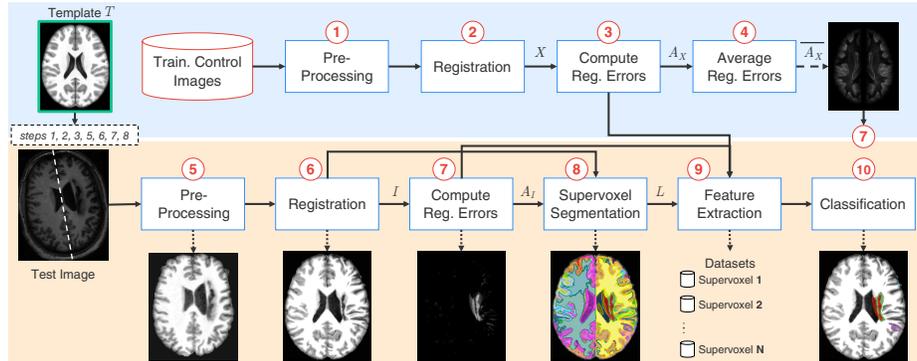


Fig. 4: Pipeline of BADRES-C [23]. The upper blue part is computed offline. The bottom orange part is computed for each test image. The template (reference image) is used in both parts (Steps 1, 2, 3, 5, 6, 7, and 8).

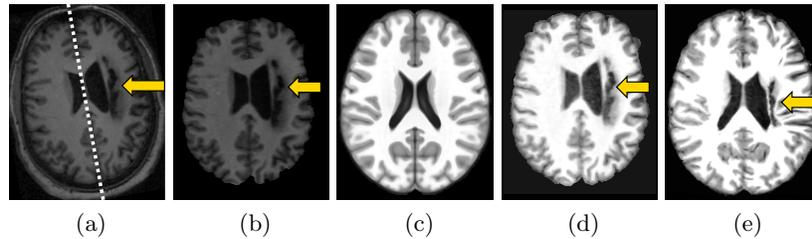


Fig. 5: 3D image preprocessing and registration steps. (a) Axial slice of a raw test 3D image. The dashed line shows its mid-sagittal plane (MSP) and the arrow indicates a stroke lesion. (b) Test image after noise filtering, MSP alignment, bias field correction, and brain segmentation. (c) Axial slice of the symmetric brain template (reference image). (d) Histogram matching between (b) and the template (intensity normalization). (e) Final preprocessed image after non-rigid registration and histogram matching with the template.

then perform deformable registration to place all images in the coordinate space of the ICBM 2009c Nonlinear Symmetric template [12]. Since the image registration technique has a critical impact on the analysis, we use Elastix [18], a popular and accurate image registration method.⁵ Finally, we perform another histogram matching between the registered images and the template (Fig. 5e).

3.2 Registration Error Computation

When registering images to a standard template with *only* healthy tissues, we expect that registration errors (REs) — *i.e.*, voxel-wise absolute differences be-

⁵We used the *par0000* files available at <http://elastix.bigr.nl/wiki/index.php>

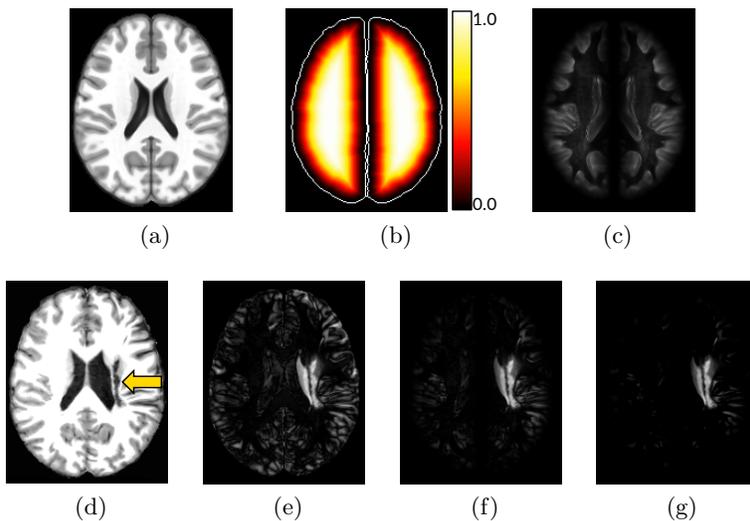


Fig. 6: Registration error computation. **(a)** Axial slice of the brain template. **(b)** Euclidean Distance Transform (EDT) normalized within $[0, 1]$ computed for the brain segmentation mask defined for the template. Brain borders are shown only for illustration purposes. **(c)** Common registration errors for control images. **(d)** Axial slice of a test stroke image after preprocessing and registration in **(a)**. The arrow indicates the stroke lesion. **(e)** Registration errors. **(f)** Attenuation of **(e)** for the cortex based on the EDT. **(g)** Final registration errors for the test image: positive values of the subtraction between **(f)** and **(c)**. Figure referenced from [23].

tween the registered image and the template — are lower and present a different pattern compared to anomalies (Fig. 6e). However, some healthy structures in the cortex, such as gyri and sulci, present high REs due to their complex shapes and very large variability between subjects — observe the cortex of the template and the registered image in Figs. 6a and 6d; note its resulting REs in Fig. 6e. As such, we need to apply some attenuation-process to avoid detecting false positives in this region.

Let T be the template (Fig. 6a) and M_T its predefined brain segmentation mask for the right hemisphere, left hemisphere, cerebellum, and brainstem (background voxels have label 0 and each object has a different label). Let $X = \{X_1, \dots, X_k\}$ be the set of k registered training images (output of Step 2 in Fig. 4) and I the test image after preprocessing and registration (output of Step 6 in Fig. 4; see also Fig. 6d).

Firstly, we compute the Euclidean Distance Transform (EDT) for each object of M_T and normalize the distances within $[0, 1]$ to build the map E (Fig. 6b). Next, we obtain the set of registration errors R_X for all X by computing the voxel-wise absolute differences between X and T (Fig. 4, Step 3; see also Fig. 6e).

For each training image $X_i \in X$, we attenuate REs in its cortex such that for each voxel $v \in X_i$,

$$\begin{aligned} f(v) &= 1 - (E(v) - 1)^\lambda \\ A_{X_i}(v) &= R_{X_i}(v) \cdot f(v), \end{aligned} \tag{2}$$

where $E(v)$ is the euclidean distance for the voxel v , $f(v)$ is its attenuation factor within $[0, 1]$, λ is the exponential factor of the function, and A_{X_i} is the map with the attenuated REs for X_i . In this work, we considered $\lambda = 4$. Thus, REs of voxels close to the brain borders are extremely attenuated, whereas those from voxels far from the borders are slightly impacted (Fig. 6f). A downside of this approach is that subtle lesions in the cortex tend to be missed due to the lack of REs.

In order to even ignore REs caused by noises or small intensity differences in regions/tissues far from the cortex, we create a *common registration error map* \bar{A}_X by averaging the attenuated REs from A_X (output of Step 4 in Fig. 4; see also Fig. 6c). Finally, we repeat the same steps to compute the attenuated REs for the test image I and then subtract \bar{A}_X from them. Resulting positive values form a final attenuated registration error map A_I for I (output of Step 7 in Fig. 4; see also Fig. 6g).

3.3 Supervoxel Segmentation

The direct comparison between the registered image and its template, or even between large 3D regular patches, is not useful as it will not tell us where *small-scale* REs occur — a similar parallel is done for asymmetries in [25]. Conversely, a voxel-wise comparison is risky, since individual voxels contain too little information to capture REs. These difficulties motivate the use of *supervoxels* as the unit of comparison (Step 8 in Fig. 4).

Inspired by the SymmISF method [24] used in SAAD for symmetrical supervoxel segmentation, we propose a new technique that extracts supervoxels in the brain guided by registration errors, as shown in Fig. 7. Our supervoxel segmentation is also based on the recent Iterative Spanning Forest (ISF) framework [42] for superpixel segmentation (Section 2) and has three steps: (i) seed estimation; (ii) connected supervoxel delineation (multiple iterations); and (iii) seed recomputation to improve delineation, as follows.

Recall a template T , its predefined brain segmentation mask M_T (objects of interest), a preprocessed 3D test image I registered on T , and its attenuated registration error map A_I . Equivalently to SymmISF, we find initial seeds by selecting one seed per local maximum in A_I (see the seeds in Fig. 7). We compute the local maxima of the foreground of a binarized A_I at $\gamma \times \tau$, where τ is Otsu’s threshold [28]. The higher the factor γ is, the lower is the number of components in the binarized A_I . We extend the seed set with a fixed number of seeds (*e.g.*, 100) by uniform grid sampling the regions with low REs of the binarized image, resulting in the final seed set \mathcal{S} .

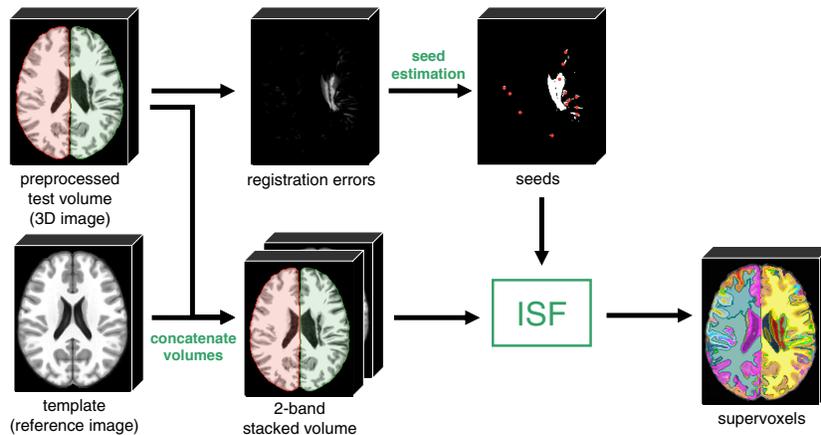


Fig. 7: Pipeline of the proposed supervoxel segmentation. The method stacks the input preprocessed test 3D image (segmented objects are colored) with the template to build a 2-band volume. An initial seed set is obtained from the registration errors of the test image. For each object of the segmentation brain mask, the ISF framework [42] estimates supervoxels inside the object from the initial seeds. Resulting supervoxels are combined and relabeled to form the final label map.

By stacking I and T as the input 2-band volume (Fig. 7), we perform ISF inside each object of interest in M_T , *separately*, from the initial seeds. The results are label maps wherein each supervoxel is assigned to a distinct number/color. We then combine and relabel the resulting supervoxels to build the final supervoxel map L (output of Step 8 in Fig. 4).

3.4 Feature Extraction and Classification

Our feature extraction and classification steps are very similar to those of SAAD [24], as detailed next. BADRESC relies on an outlier detection approach that designs a set of *specialized* one-class classifiers (OCCs) specific for each test 3D image, as shown in Fig. 8. For each 3D test image, each supervoxel in L is used to create a *specialized* one-class classifier (OCC) using as feature vector the *normalized histogram* of the attenuated registration errors (REs) in A_I (Step 9 in Fig. 4). Classifiers are trained from *control images* only, thus locally modeling normal REs for the hemispheres, cerebellum, and brainstem. BADRESC uses the one-class linear Support Vector Machine (oc-SVM) for this task [20]. Finally, BADRESC uses the trained OCCs to find supervoxels with abnormal REs in I (Step 10 in Fig. 4). Fig. 9 illustrates the supervoxel classification.

When *dynamically* designing specialized one-class per-supervoxel classifiers for each test image, BADRESC implicitly considers the *position* of the supervoxels when deciding upon their registration errors. The central premise for this

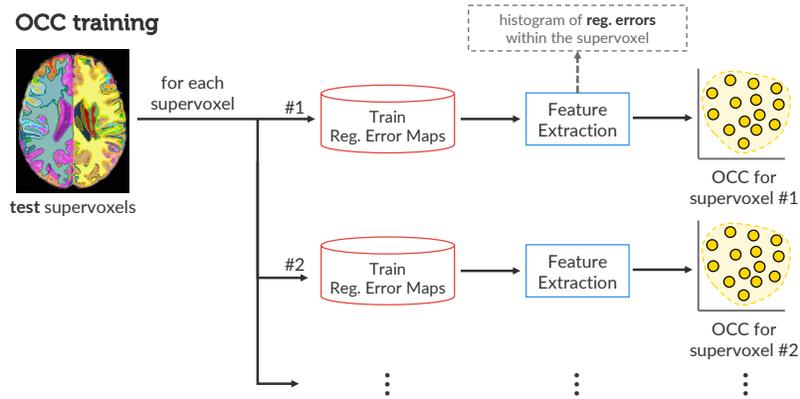


Fig. 8: One-class classifier (OCC) training to detect abnormal registration errors. For each supervoxel from a given test 3D image, BADRESC trains an OCC from the training normal registration errors previously computed.

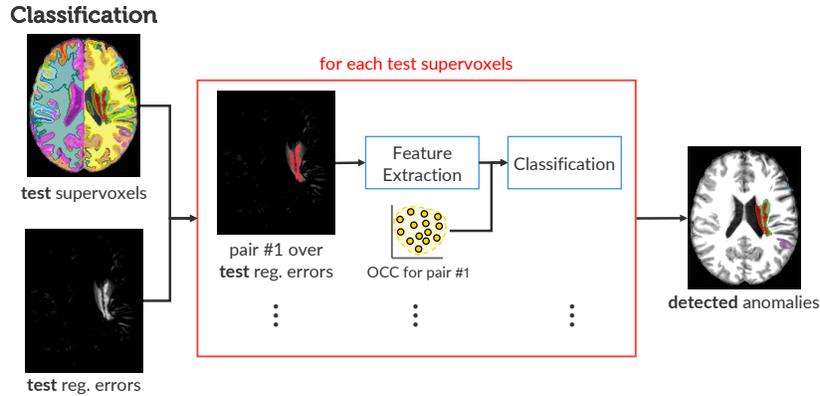


Fig. 9: Detection of abnormal registration errors of a test 3D image by supervoxel classification. For each supervoxel, BADRESC uses the corresponding OCC to classify the registration errors inside it.

is that a *single* global classifier cannot to separate normal and anomalous tissues by only using texture features.

4 Experiments

To assess the performance of BADRESC, we conducted a set of experiments. This section describes the MR-T1 image datasets, baselines, and the evaluation protocol considered for the experiments. All computations were performed on the same Intel Core i7-7700 CPU 3.60GHz with 64GB of RAM.

4.1 Datasets

To evaluate BADRESC, we need datasets with *volumetric* MR-T1 brain images (i) from *healthy* subjects for training, and (ii) with lesions of different appearance (especially small ones) and their segmentation masks. First, we considered the CamCan dataset [39], which has 653 MR-T1 images of 3T from healthy men and women between 18 and 88 years. As far as we know, CamCan is the largest public dataset with 3D images of healthy subjects acquired from different scanners. To avoid noisy data in the training set, we removed some images with artifacts or bad acquisition after a visual inspection in all MR-T1 images, yielding 524 images.⁶

For testing, we chose the Anatomical Tracings of Lesions After Stroke (ATLAS) public dataset release 1.2 [19] in our experiments. ATLAS is a challenging dataset with a large variety of manually annotated lesions and images of 1.5T and 3T acquired from different scanners. It contains heterogeneous lesions that differ in size, shape, and location (see Fig. 1). All images only have a mask with the stroke region, ignoring other possible anomalies caused by those lesions. Current state-of-the-art segmentation results [31] for ATLAS from a *supervised* method based on U-Net are inaccurate yet (Dice score of 0.4867).

Since the considered training images have a 3T field strength, we selected all 3T images from ATLAS for analysis (total of 269 images). All images were registered into the coordinate space of ICBM 2009c Nonlinear Symmetric template [12] and preprocessed as outlined in Section 3.1.

4.2 Baselines

We compared BADRESC against two baselines: (i) the SAAD method proposed in [24], which in turn was also evaluated with the ATLAS dataset as reported in [24], and (ii) the convolutional-autoencoder-based approach (CAE) from Chen *et al.* [8], which is, as far as we know, the current state-of-the-art *unsupervised* method for the ATLAS dataset.

We considered the 2D axial slices of all preprocessed training images to train CAE, which has the following architecture: three 2D convolutional layers with 16, 8, and 8 filters of patch size 3×3 , respectively, followed by *ReLU* activation and 2D max-pooling in the *encoder*, and the corresponding operations in the *decoder*. The Nadam gradient optimizer [38] minimized the mean squared error between reconstructed and expected 2D axial slices during training. The method detects anomalies by thresholding the residual image of the input image *vs* its reconstruction to obtain a binary segmentation, similarly to Baur *et al.* [4] and Chen *et al.* [8]. We followed Baur *et al.* [4] by selecting two thresholds as the 90th and 95th percentile from the histogram of reconstruction errors on the considered training set, resulting in the brightness of 194 and 282, respectively. For simplicity, we call CAE-90 and CAE-95 for the versions with the 90th and 95th percentile, respectively.

⁶A link to all these images will be added in the camera-ready paper.

For a fair comparison, we evaluated SAAD for all 3T images that only contain lesions in the hemispheres. Additionally, we evaluated BADRESC and CAE for all considered testing images, including the ones with stroke lesions in the cerebellum and brainstem. We used the following parameters for BADRESC, empirically obtained from the observation on a few training control images: $\alpha = 0.06$, $\beta = 5.0$, $\gamma = 3$, histograms of 128 bins, and $\nu = 0.01$ for the linear oc-SVM.

4.3 Quality Metrics

Although BADRESC detects anomalies regardless of their types or diseases, we can compute quantitative scores only over those lesions that are labeled in ATLAS, which are a subset of what BADRESC can detect. Thus, we propose a set of metrics to evaluate detection quality, as follows. We start computing the detection rate based on at least 15% overlap between supervoxels detected by the methods and lesions labeled in ATLAS (Tables 1 and 2, row 1). We then computed the *true positive rate* (recall) that measures the percentage of lesion voxels correctly classified as abnormal (Tables 1 and 2, row 2). Although our focus is on *detecting* abnormal asymmetries, we also measured the *Dice score* between lesions and the detected supervoxels to check BADRESC’s potential as a *segmentation method* (Tables 1 and 2, row 3). However, observe that true anomalies detected by our method that are not annotated as lesions in the ground-truth masks will be incorrectly considered as false-positive and, thus, underestimating the Dice score. We could then consider only supervoxels overlapped with the annotated lesions to compute Dice scores, but this would be unfair to the considered baselines.

We provided false-positive (FP) scores in terms of both voxels and supervoxels regarding the ground-truth stroke lesions of ATLAS. Hence, some anomalies with no labeled masks in ATLAS are considered FP. This is the best we can do in the absence of labeled masks for all kinds of abnormalities in this dataset. We computed the *mean number of FP voxels*, *i.e.*, incorrectly classified as abnormal (Tables 1 and 2, row 4). We normalized this count with respect to all classified voxels (Tables 1 and 2, row 5), *i.e.*, the total number of voxels inside the right hemisphere for SAAD and all voxels from the hemispheres, cerebellum, and brainstem for BADRESC and CAE.

At the next level, we estimated FP supervoxels as those whose voxels overlap less than 15% with ground-truth lesion voxels. We computed the mean number of FP supervoxels and their proportions to the total number of supervoxels (Tables 1 and 2, rows 6 and 7). The first metric gives us an estimation of the visual-inspection user effort. The second metric checks how imprecise detection is regarding the total number of regions that the user has to analyze visually.

When visually analyzing FP supervoxels, it is harder to check many disconnected supervoxels spread across the brain than a few connected ones. Hence, we gauge visual analysis user-effort by evaluating the two metrics outlined above on the level of connected FP supervoxel components (Tables 1 and 2, rows 8 and 9). Finally, we also computed the mean processing times of each method

(Tables 1 and 2, row 10) for preprocessed images, thus excluding the mean time of the preprocessing step (Section 3.1), which is 90 seconds on average.

5 Evaluation Results

Table 1 summarizes quantitative results of the baselines for stroke lesions in the hemispheres, while Fig. 10 presents some corresponding visual results. CAE-90 presents considerably higher detection scores (0.953) than SAAD (0.845) and BADRESC (0.82). However, these impressive results are misleading as CAE reports considerably more false-positive voxels than SAAD (about 6.75x), being drastically worse than BADRESC (about 48x) — compare rows 4 and 5 in Table 1. For instance, CAE-90 misclassifies 23.7% of the hemispheres as abnormal, which is far from being reasonable and hinders the visual analysis (we expect just a small portion of the brain, *e.g.*, 1%). These high FP rates explain the poor Dice scores for CAE in Table 1, which in turn are compatible with the ones reported in [8].

Additionally, CAE is speedy (running time about 2s per image) and yields very noisy disconnected regions, especially in regions with transitions between white and gray matter (*e.g.*, the cortex), that hinder the subsequent visual inspection (see the results in Fig. 10). Even though the number of FP voxels decrease for a higher threshold, the detection score can be hugely impacted; for example, the threshold at the 95th percentile approximately halves both the detection score and FP voxels rates compared with the results for the 90th percentile in Table 1. CAE might present better results by using a considerable large training set and/or some additional post-processing, but this is not considered in [4, 8]. CAE presents better results for other medical imaging modalities, such as CT and T2 [4, 8].

SAAD reports a better detection rate and mean recall for hemispheric lesions than BADRESC, although the difference between such scores is not accentuate — *e.g.*, SAAD has a detection rate of 0.845 while BADRESC has 0.82. BADRESC, in turn, reports a better Dice score (0.17) than SAAD (0.12). However, as outlined in Section 4.3, this score is underestimated since real unlabeled anomalies detected by the methods are considered false-positive. If we considered only supervoxels overlapped with the annotated lesions, such a Dice score leverages to 0.42. While still low, this score is not far from state-of-the-art results (Dice score 0.4867) on the ATLAS dataset from a *supervised* method based on U-Net [31]. Interestingly, our method is noticeably superior to CAE, which is an *unsupervised* method (like ours), reporting Dice scores of 0.015.

When analyzing supervoxels, both SAAD and BADRESC output more meaningful regions for visual analysis than CAE — compare the detected regions in Fig. 10. They can accurately detect small asymmetric lesions in the hemispheres (Fig. 10, Image 1) since asymmetries and registration errors successfully emphasize such lesions (see these for Image 1 in Fig. 10). SAAD cannot detect lesions with low asymmetries, while BADRESC does not have this limitation — compare the results, asymmetries, and registration errors for Image 2 in Fig. 10. However,

Table 1: Quantitative comparison between the baselines for images from the ATLAS dataset with stroke lesions in the hemispheres. **Top part:** higher values mean better accuracies. **Bottom part:** lower values mean better accuracies. The abbreviation k denotes thousands.

	CAE-90	CAE-95	SAAD	BADRES
1 Detection rate	0.953	0.567	0.845	0.82
2 True positive rate (mean recall)	0.34 ± 0.15	0.21 ± 0.13	0.44 ± 0.25	0.39 ± 0.26
3 Dice	0.015 ± 0.023	0.015 ± 0.024	0.12 ± 0.15	0.17 ± 0.15
4 # FP voxels	432k ± 186k	207k ± 45k	64k ± 37k	9k ± 11k
5 FP voxel rate	0.237 ± 0.102	0.113 ± 0.025	0.08 ± 0.05	0.005 ± 0.01
6 # FP supervoxels	method does not use supervoxels		58.87 ± 22.45	21.46 ± 13.86
7 FP supervoxel rate			0.2 ± 0.06	0.1 ± 0.07
8 # FP connected supervoxels			53 ± 17.31	16.61 ± 9.21
9 FP connected supervoxel rate			0.18 ± 0.05	0.08 ± 0.042
10 Mean processing time (in secs)	2.09 ± 0.08	2.04 ± 0.16	63.03 ± 6.73	54.17 ± 1.3

both methods are ineffective in detecting tiny anomalies (Fig. 10, image 3) since asymmetries and registration errors cannot highlight such anomalies.

BADRES is a bit faster and reports seven times fewer false-positive (FP) voxels than SAAD (Table 1, rows 4, 5, and 10): an average of 9000 FP voxels against approximately 64000, respectively. Concerning FP supervoxel scores, BADRES is consistently better than SAAD (scores roughly three times higher) — compare rows 6–9 in Table 1. For instance, SAAD incorrectly classifies 58.87 supervoxels on average, which consists of 20% of all analyzed supervoxels and 8% of the analyzed voxels in the hemisphere respectively. BADRES, in turn, reports an average of 21.46 FP supervoxels, which corresponds to 10% of analyzed supervoxels and only 0.5% of voxels in the whole brain. When grouping connected detected supervoxels, BADRES reports only 16 FP supervoxels. Hence, a user will need far less effort and time to visually analyze results from BADRES than from SAAD.

BADRES is less accurate when detecting lesions in the cerebellum and brainstem (detection rate of 0.683) than in the hemispheres, as shown in Table 2 and Fig. 11. Some of these lesions are indeed challenging, especially in the cerebellum, whose appearances are similar to their surrounding tissues (Fig. 11, Image 6). BADRES reports similar FP scores to those of hemispheric lesions, which confirms the stability of the method (compare rows 4–9 for BADRES in Tables 10 and 11). The considered registration-error attenuation (Eq. 2 with

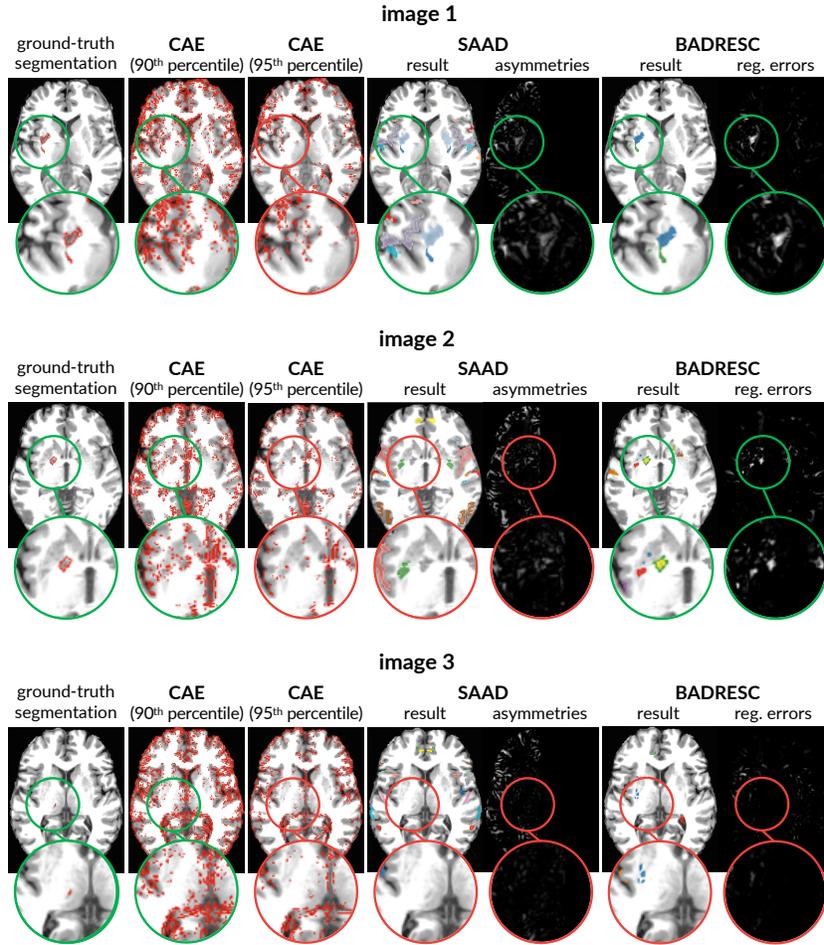


Fig. 10: Comparative results between the baselines for stroke lesions in the hemispheres. For each image (axial slice), we present an inset surrounding the lesion whose border color indicates if the lesion was detected (green) or missed (red).

$\alpha = 4$) seems to be very strong for the cerebellum and brainstem, which impairs the representation of the lesions. However, as SAAD cannot detect lesions in the cerebellum and brainstem, BADRESC is a more interesting solution to be further investigated and improved, especially in such macro-objects of interest.

6 Conclusion

We presented a new *unsupervised* method for brain anomaly detection that combines registration errors and supervoxel classification. Our approach, named

Table 2: Quantitative comparison between CAE and BADRESK for images from the ATLAS dataset with stroke lesions in the cerebellum and brainstem. **Top part:** higher values mean better accuracies. **Bottom part:** lower values mean better accuracies. The abbreviation k denotes thousands.

	CAE-90	CAE-95	BADRESK
1 Detection rate	0.878	0.365	0.683
2 True positive rate (mean recall)	0.3 ± 0.145	0.17 ± 0.14	0.26 ± 0.26
3 Dice	0.01 ± 0.02	0.01 ± 0.01	0.1 ± 0.15
4 # FP voxels	434k ± 68k	225k ± 67k	8.7k ± 7.9k
5 FP voxel rate	0.238 ± 0.038	0.124 ± 0.037	0.005 ± 0.004
6 # FP supervoxels	method does not use supervoxels		23.43 ± 15.25
7 FP supervoxel rate			0.09 ± 0.05
8 # FP connected supervoxels			18.41 ± 10.19
9 FP connected supervoxel rate			0.08 ± 0.03
10 Mean processing time (in secs)	2.09 ± 0.08	2.04 ± 0.16	54.17 ± 1.3

BADRESK, adapts a recent supervoxel-based approach (SAAD) to detect *outliers* as anomalies from registration errors in the hemispheres, cerebellum, and brainstem. This work is an extension of a previous one, which originally introduces BADRESK. Its main contributions include a more detailed explanation of the method, especially concerning supervoxel segmentation, and an extended evaluation (more baselines and evaluation metrics).

BADRESK was validated on 3T MR-T1 images of stroke patients with annotated lesions, outperforming a convolutional-autoencoder-based approach, and attaining similar detection accuracy to SAAD for lesions in the hemispheres and substantially fewer false positives. BADRESK also detects lesions in the cerebellum and brainstem with promising results.

For future work, we intend to improve BADRESK by optimizing its parameters and using additional visual analytics techniques to improve seeding and further investigate other anomaly features and classifiers to yield better detection rates, especially for the cerebellum and brainstem.

References

1. Akkus, Z., et al.: Deep learning for brain MRI segmentation: state of the art and future directions. *J Digit Imaging* **30**(4), 449–459 (2017)

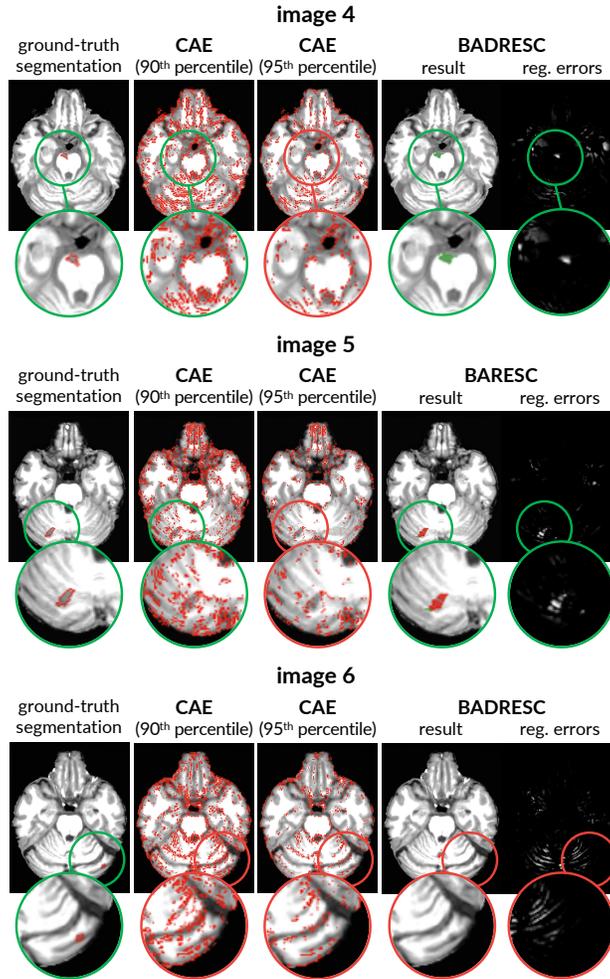


Fig. 11: Comparative results between CAE and BADRESC for stroke lesions in the cerebellum and brainstem. For each image (axial slice), we show an inset surrounding the lesion whose border color indicates if the lesion was detected (green) or missed (red).

2. Amorim, W.P., Falcão, A.X., Papa, J.P., Carvalho, M.H.: Improving semi-supervised learning through optimum connectivity. *Pattern Recog* **60**, 72–85 (2016)
3. Aslani, S., et al.: Deep 2D encoder-decoder convolutional neural network for multiple sclerosis lesion segmentation in brain MRI. In: *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*. pp. 132–141 (2018)
4. Baur, C., Wiestler, B., Albarqouni, S., Navab, N.: Deep autoencoding models for unsupervised anomaly segmentation in brain MR images. In: *International MICCAI Brainlesion Workshop*. pp. 161–169 (2018)

5. Belém, F., Melo, L., Guimaraes, S.J.F., Falcão, A.X.: The importance of object-based seed sampling for superpixel segmentation. In: Conf. on Graphics, Patterns and Images (SIBGRAPI). pp. 108–115 (2019)
6. Bragantini, J., Martins, S.B., Castelo-Fernandez, C., Falcão, A.X.: Graph-based image segmentation using dynamic trees. In: Iberoamerican Congress on Pattern Recognition (CIARP). pp. 470–478 (2018)
7. Chen, H., et al.: VoxResNet: Deep voxelwise residual networks for brain segmentation from 3D MR images. *Neuroimage* **170**, 446–455 (2018)
8. Chen, X., et al.: Deep generative models in the real-world: An open challenge from medical imaging. arXiv preprint arXiv:1806.05452 (2018)
9. Ciesielski, K.C., Falcão, A.X., Miranda, P.A.V.: Path-value functions for which dijkstra’s algorithm returns optimal mapping. *Journal of Mathematical Imaging and Vision* **60**(7), 1025–1036 (2018)
10. Falcão, A.X., Cunha, B.S., Lotufo, R.A.: Design of connected operators using the image foresting transform. In: SPIE Medical Imaging. vol. 4322, pp. 468–479 (2001)
11. Falcão, A.X., Stolfi, J., de Alencar Lotufo, R.: The image foresting transform: Theory, algorithms, and applications. *IEEE T Pattern Anal* **26**(1), 19–29 (2004)
12. Fonov, V.S., et al.: Unbiased nonlinear average age-appropriate brain templates from birth to adulthood. *Neuroimage* **47**, S102 (2009)
13. Gao, Y., Riklin-Raviv, T., Bouix, S.: Shape analysis, a field in need of careful validation. *Human Brain Mapping* **35**(10), 4965–4978 (2014)
14. Guo, D., et al.: Automated lesion detection on MRI scans using combined unsupervised and supervised methods. *BMC Medical Imaging* **15**(1), 50 (2015)
15. Havaei, M., et al.: Brain tumor segmentation with deep neural networks. *Med Image Anal* **35**, 18–31 (2017)
16. Juan-Albarracín, J., et al.: Automated glioblastoma segmentation based on a multiparametric structured unsupervised classification. *PLoS One* **10**(5), e0125143 (2015)
17. Kamnitsas, K., Ledig, C., Newcombe, V.F.J., Simpson, J.P., Kane, A.D., Menon, D.K., Rueckert, D., Glocker, B.: Efficient multi-scale 3D CNN with fully connected CRF for accurate brain lesion segmentation. *Med Image Anal* **36**, 61–78 (2017)
18. Klein, S., Staring, M., Murphy, K., Viergever, M.A., Pluim, J.P.W.: elastix: A toolbox for intensity-based medical image registration. *IEEE T Med Imaging* **29**(1), 196–205 (2010)
19. Liew, S.L., et al.: A large, open source dataset of stroke anatomical brain images and manual lesion segmentations. *Scientific Data* **5**, 180011 (2018)
20. Manevitz, L.M., Yousef, M.: One-class SVMs for document classification. *Journal of Machine Learning Research* **2**(Dec), 139–154 (2001)
21. Manjón, J.V.: MRI preprocessing. In: *Imaging Biomarkers*, pp. 53–63. Springer (2017)
22. Martins, S.B., Bragantini, J., Yasuda, C.L., Falcão, A.X.: An adaptive probabilistic atlas for anomalous brain segmentation in MR images. *Medical Physics* **46**(11), 4940–4950 (2019)
23. Martins, S.B., Falcão, A.X., Telea, A.C.: BADRESC: Brain anomaly detection based on registration errors and supervoxel classification. In: *Biomedical Engineering Systems and Technologies: BIOIMAGING*. pp. 74–81 (2020), best student paper awards
24. Martins, S.B., Ruppert, G., Reis, F., Yasuda, C.L., Falcão, A.X.: A supervoxel-based approach for unsupervised abnormal asymmetry detection in MR images of the brain. In: *IEEE Intern. Symp. on Biomed. Imaging (ISBI)*. pp. 882–885 (2019)

25. Martins, S.B., Telea, A.C., Falcão, A.X.: Extending supervoxel-based abnormal brain asymmetry detection to the native image space. In: IEEE Engineering in Medicine and Biology Society (EMBC). pp. 450–453 (2019)
26. Miranda, P.A.V., Mansilla, L.A.C.: Oriented image foresting transform segmentation by seed competition. *IEEE Transaction on Image Processing* **23**(1), 389–398 (2014)
27. Montero, A.E., Falcão, A.X.: A divide-and-conquer clustering approach based on optimum-path forest. In: Conf. on Graphics, Patterns and Images (SIBGRAPI). pp. 416–423 (2018)
28. Otsu, N.: A threshold selection method from gray-level histograms. *IEEE Trans. on systems, man, and cybernetics* **9**(1), 62–66 (1979)
29. Papa, J.P., Falcão, A.X., Suzuki, C.T.N.: Supervised pattern classification based on optimum-path forest. *International Journal of Imaging Systems and Technology* **19**(2), 120–131 (2009)
30. Pinto, A., Pereira, S., Correia, H., Oliveira, J., Rasteiro, D.M., Silva, C.A.: Brain tumour segmentation based on extremely randomized forest with high-level features. In: IEEE Engineering in Medicine and Biology Society (EMBC). pp. 3037–3040 (2015)
31. Qi, K., Yang, H., Li, C., Liu, Z., Wang, M., Liu, Q., Wang, S.: X-net: Brain stroke lesion segmentation based on depthwise separable convolution and long-range dependencies. In: Medical Image Computing and Computer-Assisted Intervention (MICCAI). pp. 247–255 (2019)
32. Rocha, L.M., Cappabianco, F.A.M., Falcão, A.X.: Data clustering as an optimum-path forest problem with applications in image analysis. *International Journal of Imaging Systems and Technology* **19**(2), 50–68 (2009)
33. Sato, D., Hanaoka, S., Nomura, Y., Takenaga, T., Miki, S., Yoshikawa, T., Hayashi, N., Abe, O.: A primitive study on unsupervised anomaly detection with an autoencoder in emergency head ct volumes. In: SPIE Medical Imaging. p. 105751P (2018)
34. Shakeri, M., et al.: Statistical shape analysis of subcortical structures using spectral matching. *Computerized Medical Imaging and Graphics* **52**, 58–71 (2016)
35. Soltaninejad, M., Yang, G., Lambrou, T., Allinson, N., Jones, T.L., Barrick, T.R., Howe, F.A., Ye, X.: Automated brain tumour detection and segmentation using superpixel-based extremely randomized trees in FLAIR MRI. *International Journal of Computer Assisted Radiology and Surgery* **12**(2), 183–203 (2017)
36. Sousa, A.M., Martins, S.B., Falcão, A.X., Reis, F., Bagatin, E., Irion, K.: ALTIS: A fast and automatic lung and trachea CT-image segmentation method. *Medical Physics* **46**(11), 4970–4982 (2019)
37. Stutz, D., Hermans, A., Leibe, B.: Superpixels: An evaluation of the state-of-the-art. *Computer Vision and Image Understanding* **166**, 1–27 (2018)
38. Sutskever, I., Martens, J., Dahl, G., Hinton, G.: On the importance of initialization and momentum in deep learning. In: International Conference on Machine Learning (ICML). pp. 1139–1147 (2013)
39. Taylor, J.R., Williams, N., Cusack, R., Auer, T., Shafto, M.A., Dixon, M., Tyler, L.K., Henson, R.N., et al.: The cambridge centre for ageing and neuroscience (CamCAN) data repository: structural and functional mri, meg, and cognitive data from a cross-sectional adult lifespan sample. *Neuroimage* **144**, 262–269 (2017)
40. Thyreau, B., Sato, K., Fukuda, H., Taki, Y.: Segmentation of the hippocampus by transferring algorithmic knowledge for large cohort processing. *Med Image Anal* **43**, 214–228 (2018)

41. Tustison, N.J., Avants, B.B., Cook, P.A., Zheng, Y., Egan, A., Yushkevich, P.A., Gee, J.C.: N4ITK: improved N3 bias correction. *IEEE T Med Imaging* **29**(6), 1310–1320 (2010)
42. Vargas-Muñoz, J.E., Chowdhury, A.S., Alexandre, E.B., Galvão, F.L., Miranda, P.A.V., Falcão, A.X.: An iterative spanning forest framework for superpixel segmentation. *IEEE Transaction on Image Processing* **28**(7), 3477–3489 (2019)
43. Wang, L., Joshi, S.C., Miller, M.I., Csernansky, J.G.: Statistical analysis of hippocampal asymmetry in schizophrenia. *Neuroimage* **14**(3), 531–545 (2001)
44. Wu, W., Chen, A.Y.C., Zhao, L., Corso, J.J.: Brain tumor detection and segmentation in a CRF (conditional random fields) framework with pixel-pairwise affinity and superpixel-level features. *International Journal of Computer Assisted Radiology and Surgery* **9**(2), 241–253 (2014)
45. Yan, J., Yu, Y., Zhu, X., Lei, Z., Li, S.Z.: Object detection by labeling superpixels. In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. pp. 5107–5116 (2015)

A Image Foresting Transform

The *Image Foresting Transform (IFT)* is a methodology for the design of image operators based on optimum connectivity [11]. For a given connectivity function and a graph derived from an image, the IFT algorithm minimizes (maximizes) a connectivity map to partition the graph into an *optimum-path forest* rooted at the minima (maxima) of the resulting connectivity map. The image operation resumes to a post-processing of the forest attributes, such as the root labels, optimum paths, and connectivity values. IFT has been successfully applied in different domains, such as image filtering [10], segmentation [6, 22, 36], superpixel segmentation [5, 24, 42], pattern classification [2, 29], and data clustering [27, 32]. This appendix presents preliminary concepts and introduces the IFT algorithm.

A.1 Preliminary Concepts

Image Graphs: A *d-dimensional multi-band image* is defined as the pair $\hat{I} = (D_I, \vec{I})$, where $D_I \subset Z^d$ is the *image domain* — *i.e.*, a set of elements (pixels/voxels) in Z^d — and $\vec{I} : D_I \rightarrow \mathbb{R}^c$ is a mapping function that assigns a vector of c intensities $\vec{I}(p)$ — one value for each band (channel) of the image — to each element $p \in D_I$. For example, for 2D RGB-color images: $d = 2$, $c = 3$; for 3D grayscale images (*e.g.*, MR images): $d = 3$, $c = 1$. We represent a *segmentation* of \hat{I} by a label image $\hat{L} = (D_I, L)$, wherein the function $L : D_I \rightarrow \{0, 1, \dots, M\}$ maps every voxel of \hat{I} to either the background (label 0) or one of the M objects of interest.

Most images, like the ones used in this paper, typically represent their intensity values by *natural numbers* instead of *real numbers*. More specifically, $\vec{I} : D_I \rightarrow [0, 2^b - 1]$, where b is the *number of bits* (pixel/voxel depth) used to encode an intensity value.

An image can be interpreted as a graph $G_I = (D_I, \mathcal{A})$, whose *nodes* are the voxels and the *arcs* are defined by an *adjacency relation* $\mathcal{A} \subset D_I \times D_I$, with

$\mathcal{A}(p)$ being the *adjacent set* of a voxel p . A *spherical adjacency relation* of radius $\gamma \geq 1$ is given by

$$\mathcal{A}_\gamma : \{(p, q) \in D_I \times D_I, \|q - p\| \leq \gamma\}. \quad (3)$$

The image operators considered in this paper use two types of adjacency relations: \mathcal{A}_1 (6-neighborhood) and $\mathcal{A}_{\sqrt{3}}$ (26-neighborhood), as illustrated in Fig. 12.

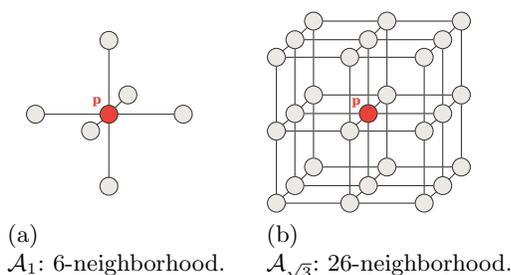


Fig. 12: Examples of adjacency relation for a given voxel p (red).

Paths: For a given image graph $G_I = (D_I, \mathcal{A})$, a *path* π_q with terminus q is a sequence of distinct nodes $\langle p_1, p_2, \dots, p_k \rangle$ with $\langle p_i, p_{i+1} \rangle \in \mathcal{A}$, $1 \leq i \leq k - 1$, and $p_k = q$. The path $\pi_q = \langle q \rangle$ is called *trivial path*. The concatenation of a path π_p and an arc $\langle p, q \rangle$ is denoted by $\pi_p \cdot \langle p, q \rangle$.

Connectivity Function: A *connectivity function* (path-cost function) assigns a value $f(\pi_q)$ to any path π_q in the image graph $G_I = (D_I, \mathcal{A})$. A path π_q^* ending at q is optimum if $f(\pi_q^*) \leq f(\tau_q)$ for every other path τ_q . In other words, a path ending at q is optimum if no other path ending at q has lower cost.

Connectivity functions may be defined in different ways. In some cases, they do not guarantee the optimum cost mapping conditions [9], but, in turn, can produce effective object delineation [26]. A common example of connectivity function is f_{max} , defined by

$$f_{max}(\langle q \rangle) = \begin{cases} 0 & \text{if } q \in \mathcal{S}, \\ +\infty & \text{otherwise.} \end{cases} \quad (4)$$

$$f_{max}(\pi_p \cdot \langle p, q \rangle) = \max\{f_{max}(\pi_p), w(p, q)\},$$

where $w(p, q)$ is the arc weight of $\langle p, q \rangle$, usually estimated from \hat{I} , and \mathcal{S} is the labeled seed set.

A.2 The General IFT Algorithm

For multi-object image segmentation, IFT requires a labeled seed set $\mathcal{S} = \mathcal{S}_0 \cup \mathcal{S}_1 \cup \dots \cup \mathcal{S}_M$ with seeds for object i in each set \mathcal{S}_i and background seeds in \mathcal{S}_0 , as illustrated in Fig. 13. The algorithm then promotes an optimum seed competition so that each seed in \mathcal{S} conquers its most closely connected voxels in the image domain. This competition considers a connectivity function f applied to any path π_q .

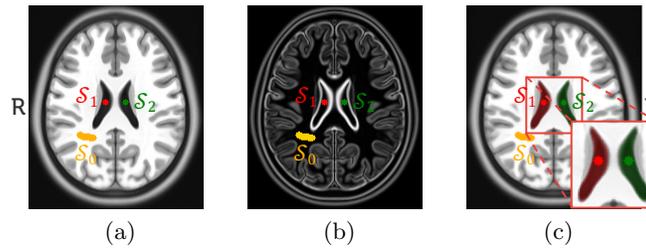


Fig. 13: Multi-object image segmentation by IFT. **(a)** Axial slice of a brain image with seeds \mathcal{S}_0 for the background (orange), \mathcal{S}_1 for the right ventricle (red), and \mathcal{S}_2 for the left ventricle (green). **(b)** Gradient image for (a) that defines the arc weights for seed competition. Arcs have high weights on object boundaries. **(c)** Resulting segmentation mask for the given seeds and arc weights. Red and green voxels represent object voxels, whereas the remaining ones are background.

Defining Π_q as the set of all possible paths with terminus q in the image graph, the IFT algorithm minimizes a path cost map

$$C(q) = \min_{\forall \pi_q \in \Pi_q} \{f(\pi_q)\}, \quad (5)$$

by partitioning the graph into an optimum-path forest P rooted at \mathcal{S} . That is, the algorithm assigns to q the path π_q^* of minimum cost, such that each object i is defined by the union between the seed voxels of \mathcal{S}_i and the voxels of D_I that are rooted in \mathcal{S}_i , *i.e.*, conquered by such object seeds.

Algorithm 2 presents the general IFT approach. Lines 1–7 initialize maps, and insert seeds into the priority queue Q . The state map U indicates by $U(q) = White$ that the voxel q was never visited (never inserted into Q), by $U(q) = Gray$ that q has been visited and is still in Q , and by $U(q) = Black$ that q has been processed (removed from Q).

The main loop (Lines 8–20) performs the propagation process. First, it removes the voxel p that has *minimum path cost* in Q (Line 9). Ties are broken in Q using the first-in-first-out (FIFO) policy. The loop in Lines 11–20 then evaluates if a path with terminus p extended to its adjacent q is cheaper than the

Algorithm 2: The General IFT Algorithm

Input: Image $\hat{I} = (D_I, I)$, adjacency relation \mathcal{A} connectivity function f , and seed set $\mathcal{S} \subset D_I$ labeled by λ .

Output: Optimum-path forest P , root map R , path-cost map C , and label map L .

Aux: Priority queue Q , state map U , and variable tmp .

```

1 foreach  $q \in D_I$  do
2    $P(q) \leftarrow \emptyset, R(q) \leftarrow q$ 
3    $C(q) \leftarrow f(\langle q \rangle), L(q) \leftarrow 0$ 
4    $U(q) \leftarrow White$ 
5   if  $q \in \mathcal{S}$  then
6      $\text{insert } q \text{ into } Q$ 
7      $L(q) \leftarrow \lambda(q), U(q) \leftarrow Gray$ 
8 while  $Q \neq \emptyset$  do
9   Remove  $p$  from  $Q$  such that  $C(p)$  is minimum
10   $U(p) \leftarrow Black$ 
11  foreach  $q \in \mathcal{A}(p)$  such that  $U(q) \neq Black$  do
12     $tmp \leftarrow f(\pi_p^* \cdot \langle p, q \rangle)$ 
13    if  $tmp < C(q)$  then
14       $P(q) \leftarrow p, R(q) \leftarrow R(p)$ 
15       $C(q) \leftarrow tmp, L(q) \leftarrow L(p)$ 
16      if  $U(q) = Gray$  then
17         $\text{update position of } q \text{ in } Q$ 
18      else
19         $\text{insert } q \text{ into } Q$ 
20         $U(q) \leftarrow Gray$ 
21 return  $(P, R, C, L)$ 

```

current path with terminus q and cost $C(q)$ (Line 13). If that is the case, p is assigned as the predecessor of q , and the root of p is assigned to the root of q (Line 14), whereas the path cost and the label of q are updated (Line 15). If q is in Q , its position is updated; otherwise, q is inserted into Q . The algorithm returns the optimum-path forest (predecessor map), root map, path-cost map, and the label map (object delineation mask).