IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received June 9, 2019, accepted June 30, 2019, date of publication July 3, 2019, date of current version July 23, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2926697

Quantitative Analysis of Patch-Based Fully Convolutional Neural Networks for Tissue Segmentation on Brain Magnetic Resonance Imaging

JOSE BERNAL, KAISAR KUSHIBAR, MARIANO CABEZAS, SERGI VALVERDE, ARNAU OLIVER, AND XAVIER LLADÓ, (Senior Member, IEEE)
Department of Computer Architecture and Technology, Universitat de Girona, 17003 Girona, Spain

Corresponding author: Jose Bernal (jose.bernal@udg.edu)

This work was supported in part by the La Fundació la Marató de TV3 and in part by the Retos de Investigació under Grant TIN2014-55710-R, Grant TIN2015-73563-JIN, and Grant DPI2017-86696-R from the Ministerio de Ciencia y Tecnología. The work of J. Bernal and K. Kushibar was supported by the Catalan Government under Grant FI-DGR2017, Grant 2017FI B00476, and Grant 2017FI B00372. The work of M. Cabezas was supported by the Juan de la Cierva–Incorporación Grant from the Spanish Government under Grant IJCI-2016-29240.

ABSTRACT Accurate brain tissue segmentation in magnetic resonance imaging (MRI) has attracted the attention of medical doctors and researchers since variations in tissue volume and shape permit diagnosing and monitoring neurological diseases. Several proposals have been designed throughout the years comprising conventional machine learning strategies as well as convolutional neural networks (CNNs) approaches. In particular, in this paper, we analyze a sub-group of deep learning methods producing dense predictions. This branch, referred in the literature as fully CNN (FCNN), is of interest as these architectures can process an input volume in less time than CNNs. Our study focuses on understanding the architectural strengths and weaknesses of literature-like approaches. We implement eight FCNN architectures inspired by robust state-of-the-art methods on brain segmentation related tasks and use them within a standard pipeline. We evaluate them using the IBSR18, MICCAI2012, and iSeg2017 datasets as they contain infant and adult data and exhibit different voxel spacing, image quality, number of scans, and available imaging modalities. The discussion is driven in four directions: comparison between 2D and 3D approaches, the relevance of multiple imaging sequences, the effect of patch size, and the impact of patch overlap as a sampling strategy for training and testing models. Besides the aforementioned analysis, we show that the methods under evaluation can yield top performance on the three data collections. A public version is accessible to download from our research website to encourage other researchers to explore the evaluation framework.

INDEX TERMS Quantitative analysis, brain MRI, tissue segmentation, fully convolutional neural networks.

I. INTRODUCTION

Automatic brain Magnetic Resonance Imaging (MRI) tissue segmentation continues being an active research topic in medical image analysis as it provides doctors with meaningful quantitative information, such as tissue volume and shape measurements. This information is widely used to diagnose brain pathologies and evaluate progression through regular MRI analysis over time [1]–[3]. Thus, the more accurate, reliable, and consistent these quantities, the more solid subsequent investigation. Hence, the study of MRI is crucial to comprehend the nature of brain diseases and the effectiveness of new treatments.

A plethora of tissue segmentation algorithms has been proposed throughout the years. Many supervised machine learning methods existed before the Convolutional Neural Network (CNN) era. A clear example of that situation is the approaches that participated in the MRBrainS13 challenge [4]. Commonly, intensity-based methods assumed that each tissue could be represented by its intensity

The associate editor coordinating the review of this manuscript and approving it for publication was Yonghong Tian.

values [5] (e.g. through Gaussian mixture models). As noise and intensity inhomogeneities degraded them, they were later equipped with spatial information [6]–[9]. Four main strategies were distinguished in the literature: (i) impose local contextual constraints using Markov Random Fields [10], (ii) include penalty terms accounting for neighborhood similarity in clustering objective functions [11], (iii) use Gibbs prior to model spatial characteristics of the brain [12] and (iv) introduce spatial information using probabilistic atlases [13], [14]. Of note, some of these methods, like FAST [10] and SPM [13], [14], are still being used in medical centers due to their robustness and adaptability [15].

Nowadays, CNNs have become appealing to address brain segmentation as they have achieved record-shattering performances in various fields in computer vision, and they discover classification-suitable representations directly from the input data. However, unlike traditional approaches, these methods still present two main issues when placed in real life scenarios: (i) lack of sufficiently labeled data and (ii) domain adaptation issues – also related to generalization problems. Seminal work on CNN for brain tissue segmentation date back to 2015 when Zhang *et al.* [16] proposed a CNN to address infant brain tissue segmentation on MRI where tissue distributions overlap and, hence, the GMM assumption does not hold. The authors showed that their CNN was suitable for the problem and could outperform techniques, such as random forest, support vector machines, level sets, and majority voting. From thereon, more sophisticated proposals have been devised [17], [18].

Former CNN strategies for tissue segmentation were trained to provide a single label given an input patch [16], [19]–[21]. Naturally, both training and testing can be time-consuming and computationally demanding. Moreover, the relationship between neighboring segmented voxels is not encoded in principle and, consequently, additional layers (such as in [22]) or post-processing may be needed to smooth results. These drawbacks can be diminished by adapting the network to perform dense prediction. The prevailing approach consists of replacing fully connected layers by 1×1 convolutional layers – $1 \times 1 \times 1$ if processing 3D data. This particular group is known in the literature as Fully CNN (FCNN) [23].

Regarding input dimensionality, three main streams are identified: 2D, 2.5D, and 3D. At the beginning of the CNN era, most of the state-of-the-art CNN techniques were 2D, in part, due to their initial usage on natural images, and computation limitations of processing 3D volumes directly. Evidently, three independent 2D models can be arranged to handle patches from axial, sagittal and coronal at the same time, hence improving acquired contextual information. These architectures are referred to as 2.5D [24]–[26]. With advances in technology, more 3D approaches have been developed and attracted more researchers as they tend to outperform 2D architectures [18]. Intuitively, the improvement of 3D and 2.5D over 2D lies on the fact that more contextual information coming from the three orthogonal planes is integrated into the network. However, this does not imply that they always perform better [27]. To the best of our knowledge, 2.5D FCNN networks are not widespread.

Several public brain MR datasets are available to the community, especially those organized by Medical Image Computing and Computer-Assisted Intervention (MICCAI) society,[1] actively encouraging research and publications in the field. Each one of these evaluation frameworks has been proposed to quantitatively compare segmentation algorithms under the same directives: common training and testing data sets and evaluation metrics. Although they have indeed carried out their mission successfully, the algorithms are generally tweaked to perform the best. Hence, it is possible that the top-performing algorithm on a specific dataset does not achieve excellent scores on another one using the same pipeline (i.e. pre-processing, data preparation, and post-processing). Moreover, a direct comparison of architectures cannot be set up as each pipeline varies. Thus, hindering understanding the underlying properties of the different networks.

In this paper, we analyze quantitatively 4×2 FCNN architectures for tissue segmentation on brain MRI. We aim at comparing various method more fairly by fixing training and test sets, processing pipeline (e.g. skull stripping, data normalization, and reconstruction), training and optimization schemes (e.g. epochs, early stopping policy, loss function, learning rate, optimizer, hardware), and performance evaluation metrics. The considered networks, comprising 2D and 3D implementations, are inspired in four recent works [28]–[31]. The models are tested on three well-known datasets of infant and adult brain scans, with different spatial resolution, voxel spacing, and image modalities. In this work, we (i) compare different FCNN strategies for tissue segmentation; (ii) quantitatively analyze the effect of network dimensionality (2D or 3D) and the impact of fusing information from single or multiple modalities; (iii) study the influence of patch size on the segmentation performance; and (iv) investigate the effects of extracting patches with a certain degree of overlap as a sampling strategy in both training and testing. We made the repository available to the public as we intend to provide a ready-to-use framework for exploring various state-of-the-art methods, valuable for newcomers to the topic. To the best of our knowledge, this is the first work providing a comprehensive evaluation of FCNNs for the task mentioned above on different datasets. Furthermore, as all architectures are part of a standard pipeline, a direct comparison can be established, allowing us to understand the advantages and disadvantages of one architecture over another.

The rest of the paper is organized as follows. In Section II, we present our evaluation framework: assessed networks, aspects to analyze, pipeline and implementation details. We describe the selected measures and datasets and the obtained results in Section III and analyze them in Section IV. We discuss final remarks in Section V.

[1] http://www.miccai.org/

II. METHODOLOGY

A. FCNNS FOR BRAIN MRI SEGMENTATION TASKS

The proposed works using FCNN for brain MRI segmentation tasks are listed in Table 1. Proposals comprise single or multiple flows of information – referred in the literature as single-path and multi-path architectures, respectively. While single-path networks process input data faster than multi-path, knowledge fusion occurring in the latter strategy may lead to better segmentation results: various feature maps from different interconnected modules and superficial layers are used to produce the final verdict [19]. Under this scheme, networks are provided with contrast, fine-grained, and implicit contextual information. Furthermore, proposals apply successive convolutions only or convolutions and de-convolutions in the so-called u-shaped models. The latter approach commonly considers connections from high-resolution layers to up-sampled ones to retain location and contextual information [28], [32], [33].

TABLE 1. Relevant information of state-of-the-art FCNN approaches for brain segmentation tasks. The reference articles are listed in the first column. The following columns outline information regarding dimensionality of the input, high-level architectural details and segmentation problem addressed by the authors. U-shaped architectures are denoted by "[U]".

Article	Architecture	Target
Brosch *et al.* [34]	3D multi-path [U]	Lesion
Kleesiek *et al.* [35]	3D single-path	Skull stripping
Nie *et al.* [36]	2D single-path [U]	Tissue
Shakeri *et al.* [37]	2D single-path	Sub-cortical structure
Kamnitsas *et al.* [31]	3D multi-path	Lesion/tumour
Dolz *et al.* [29]	3D multi-path	Sub-cortical structure
Guerrero *et al.* [30]	2D multi-path [U]	Lesion
Moeskops *et al.* [38]	2D single-path	Structure
Xu *et al.* [39]	3D multi-path [U]	Lesion
Hashemi *et al.* [40]	3D multi-path [U]	Lesion
Clèrigues *et al.* [41]	3D multi-path [U]	Lesion

From the papers indexed in Table 1, we built four multi-path architectures inspired by the works of Kamnitsas *et al.* [31], Dolz *et al.* [29], Çiçek *et al.* [28], and Guerrero *et al.* [30] (i.e. two convolution-only and two u-shaped architectures). The networks were implemented in 2D and 3D to investigate the effect of the network dimensionality on tissue segmentation. All these architectures were implemented from scratch following the architectural details given in the original work and are publicly available at our research website.[2] Although we made slight architectural changes, we retained the core idea of the original proposals. More details of the networks are given in the following sections.

1) NETWORKS INCORPORATING MULTI-RESOLUTION INFORMATION

Kamnitsas *et al.* [31], proposed a two-path 3D FCNN for brain lesion segmentation. This approach achieved top performance on two public benchmarks, BRATS 2015 and ISLES 2015. By processing information of the targeted area from two different scales simultaneously, the network incorporated local and larger contextual information, providing a more accurate response [19]. A high-level scheme of the architecture is depicted in Fig. 1a. Initially, two independent feature extractor modules extracted maps from patches from normal and downscaled versions of an input volume. Each module consisted of eight 3 × 3 × 3 convolutional layers using between 30 and 50 kernels. Afterwards, two intermediate 1 × 1 × 1 convolutional layers with 150 kernels fused and mined resulting features maps. Finally, a classification layer (another 1 × 1 × 1 convolutional layer) produced the segmentation prediction using a softmax activation.

Dolz *et al.* [29] presented a multi-resolution 3D FCNN architecture for sub-cortical structure segmentation. A general illustration of the architecture is shown in Fig. 1. The network consisted of 13 convolutional layers: nine 3 × 3 × 3, and four 1 × 1 × 1. Each one of these layers was immediately followed by a Parametric Rectified Linear Unit (PReLU) layer, except for the output layer which activation was softmax. Multi-resolution information was integrated into this architecture by concatenating feature maps from shallower layers to the ones resulting from the last 3 × 3 × 3 convolutional layer. As explained by Hariharan *et al.* [42], these kinds of connections grant networks to learn semantic – coming from deeper layers – as well as fine-grained localization information – coming from superficial layers.

2) U-SHAPED NETWORKS

In the u-shaped network construction scheme, feature maps from higher resolution layers are commonly merged to the ones on deconvolved maps to keep localization information. Merging has been addressed in the literature through concatenation [28], [34] and addition [30], [41]. In this paper, we consider networks using both approaches. A general scheme of our implementations inspired in both works is displayed in Fig. 1c.

Çiçek *et al.* [28] proposed a 3D u-shaped FCNN, known as 3D u-net. The network is formed by four convolution-pooling layers and four deconvolution-convolution layers. The number of kernels ranged from 32 in its bottommost layers to 256 in its topmost ones. In this design, maps from higher resolutions were concatenated to upsampled maps. Each convolution was immediately followed by a Rectified Linear Unit (ReLU) activation function.

Guerrero *et al.* [30] designed a 2D u-shaped residual architecture for lesion segmentation, referred as u-ResNet. The building block of this network was the residual module which (i) added feature maps produced by 3 × 3- and 1 × 1-kernel convolution layers, (ii) normalized resulting features using batchnorm, and, finally, (iii) used a ReLU activation. The network consisted of three residual modules with 32, 64 and 128 kernels, each one followed by a 2 × 2 max pooling operation. Then, a single residual module with 256 kernels was applied. Afterwards, successive deconvolution-and-residual-module pairs were employed to enlarge the networks' output

[2] http://github.com/NIC-VICOROB/tissue_segmentation_comparison

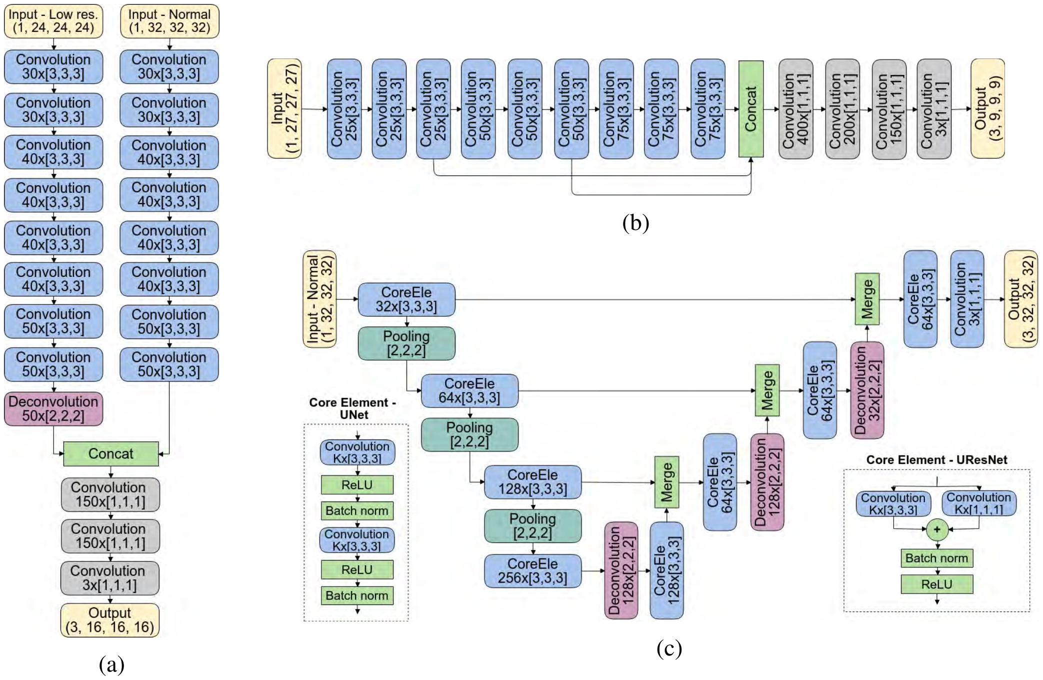


FIGURE 1. Diagram of the considered networks. Our implementations are inspired by the works of (a) Kamnitsas *et al.* [31], (b) Dolz *et al.* [29], (c) Çiçek *et al.* [28], and (d) Guerrero *et al.* [30]. Only 3D versions are shown. Notation is as follows: four-element tuples indicate number of channels and patch size in x, y and z, in that order; triples in brackets indicate kernel size. In (c), merging is either concatenation or addition; CoreEle stands for core elements of the models (both of them are detailed on the bottom left and right corner of the (c)); the letter K on the core elements is the number of kernels at a given stage.

size. The number of filters went from 256 to 32 in the layer before the prediction one. Maps from higher resolutions were merged with deconvolved maps through addition.

B. ASPECTS TO EVALUATE

This paper aims at analyzing (i) overlapping patch extraction in training and testing, (ii) single and multi-modality architectures, (iii) patch size, and (iv) 2D and 3D strategies. Details on these four evaluation cornerstones are discussed in the following sections.

1) OVERLAPPING SAMPLING IN TRAINING AND TESTING

One of the drawbacks of networks performing dense-inference is that – under similar conditions – the number of parameters increases. This issue implies that more samples should be used during training to obtain acceptable results. A common approach consists of augmenting the input data through transformations – e.g. translation, rotation, scaling. However, if the output dimension is not equal to the input size, other options can be considered. Although the main advantage of patch-based FCNNs is their dense prediction, a single pass on a particular area may produce inaccurate outputs as (i) block boundary artifacts may appear – direct consequence of tiling volumes up – and (ii) patches may not contain sufficient information to produce an accurate verdict – e.g. on the boundaries of the input. For instance, patches can be extracted from the input volumes with a certain extent of overlap and, thus, the same voxel would be seen several times surrounded by different neighborhoods. As each patch contains a specific part of the region of interest, each voxel would be classified according to the information it contains. An example of patch extraction with three extents of overlap is depicted in Fig. 2. In such a way, more information would be taken into account to produce a more consented and smoother response. Summarizing, the strategy is beneficial as (i) more samples are gathered, and (ii) networks are provided with information that may improve spatial consistency as illustrated in Fig. 3. Of note, the overlap degree is determined by the overlap between adjacent output patches and not input ones.

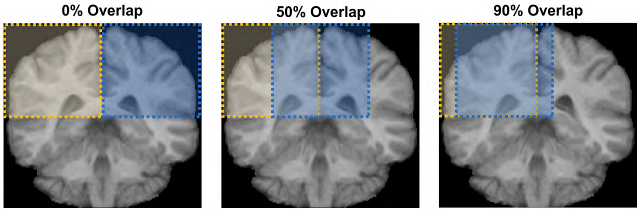


FIGURE 2. Patch extraction with null, medium and high overlap. Yellow and blue areas corresponds to the first and second blocks to consider. When there is overlap among patches, voxels are seen in different neighborhoods each time.

The sampling strategy aforementioned can be enhanced by overlaying predictions, i.e. obtain a consented prediction per voxel from the segmentation of different overlapping patches.

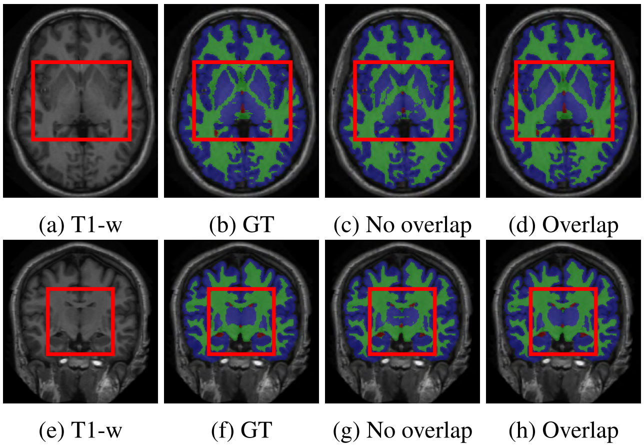


FIGURE 3. Segmentation using overlapping patch extraction in training (a-d) and testing (e-h). From left to right, T1-w volume (a)-(e), ground truth (b)-(f), segmentation without overlap (c)-(g) and with overlap (d)-(h). The basal ganglia area (inside the red box) depicts notable changes between strategies. Notice that results obtained with overlapping sampling appear more similar to the ground truth. Colors for CSF, GM and WM, are red, blue and green, respectively.

Unlike sophisticated post-processing techniques, the network itself is used to improve its segmentation. As depicted in Fig. 3 (e-h), the leading property of this post-processing technique is that small segmentation errors – e.g. holes and block boundary artifacts – are corrected. The consensus among outputs can be addressed through majority voting, for instance.

2) INPUT MODALITIES

Depending on the number of modalities available in a dataset, approaches can be either single- or multi-modality. If many modalities were acquired, networks could be adapted to process them all at the same time either using different channels or various processing paths – also referred in the literature as early and late fusion schemes [27], respectively. Naturally, the former strategy is desirable regarding computational resources, but the latter may extract more valuable features. In this work, we consider the early fusion only. Regardless of the fusion scheme, merging different sources of information may provide models with complementary features and, hence, lead to enhanced outputs [16].

3) PATCH SIZE

A pivotal hyperparameter of CNNs is the input patch size. Experiments in this regard have shown that the larger the input patch, the more contextual information the network can mine to produce the final response. Nevertheless, the greater the patch, the more resources needed to train the network successfully and the more parameters to be optimized during training. Thus, a trade-off between these factors is needed to obtain the best response.

4) NETWORK DIMENSIONALITY

There are two main streams of FCNN regarding its input dimensionality: 2D and 3D. On the one hand, 2D architectures are fast, flexible, and scalable; however, they ignore completely data from neighboring slices, i.e. implicit information is reduced compared to 3D approaches. On the other hand, 3D networks acquire valuable implicit contextual information from orthogonal planes. Even though labeling is carried out slice-by-slice, these strategies tend to lead to better performance than 2D. Nevertheless, they are computationally demanding due to the exponential increase in parameters and resource consumption and may require larger training sets. Therefore, depending on the data itself, one approach would be more suitable than the other.

C. IMPLEMENTATION DETAILS

1) GENERAL PIPELINE

General tissue segmentation pipelines contemplate four essential components: pre-processing, data preparation, classification, and post-processing. Specific implementations of each one of these elements can be plugged and unplugged as required to achieve the best performance. First, pre-processing is carried out by (i) removing skull, and (ii) normalizing intensities between scans. We use the ground truth masks to address the former tasks and standardize our data to have zero mean and unit variance. Second, data is prepared by extracting useful and overlapping patches – containing information from one of the three tissues. Third, each patch is classified. Fourth, no post-processing is considered.

2) NETWORK TRAINING

The steps to train a model on a given dataset are as follows. First, for each dataset, the training set is split into training and validation at random (80% and 20% of the volumes, respectively). Both training and validation sets are fixed for all networks to ensure they were trained under similar conditions. Second, the networks are trained in batches of 32 elements for a maximum of 20 epochs. In this particular case, we observed experimentally that the loss function of all networks converged to their lowest values for both training and validation collections within 20 epochs and overfitted afterwards. Third, at the end of each epoch, the loss function value on the validation set is computed. The training stopping criterion is no improvement in validation accuracy after n epochs, which is monitored using an early stopping policy with patience n equal to 2. We adopted this strategy to guarantee that all deep networks were trained in the best way possible while avoiding over-fitting to the training set and increasing the chances of achieving the best performance on unknown collections. The models are optimized for the categorical cross-entropy loss function using the Adam [43] optimization method with an initial learning rate of 1×10^{-3}, a decay of 0.0, $\beta_1 = 0.9$, and $\beta_2 = 0.999$ (i.e. default parameter values, as suggested in the original paper). Of note, we considered this particular optimizer as it showed empirically improved performance in comparison to other stochastic optimization methods and favorable performance in problems with noisy gradients and, also, we used its default hyperparameter values since the authors found that little tuning was needed to reach acceptable

TABLE 2. Details per implemented architecture. The items into consideration appear on the first column. Note that there are two inputs for KK as the network has two processing branches.

	Item	DM_{2D}	DM_{3D}	KK_{2D}	KK_{3D}	UN_{2D}	UN_{3D}	URN_{2D}	URN_{3D}
General	Input size	27 × 27	27 × 27 × 27	32 × 32 20 × 20	32 × 32 × 32 20 × 20 × 20	32 × 32	32 × 32 × 32	32 × 32	32 × 32 × 32
	Output size	9 × 9	9 × 9 × 9	16 × 16	16 × 16 × 16	32 × 32	32 × 32 × 32	32 × 32	32 × 32 × 32
	Number of parameters	547 278	3 333 270	569 678	7 101 038	1 931 620	5 606 308	995 108	2 623 844
No. components	Convolutional	13	13	19	19	18	18	18	18
	Batchnorm	0	0	0	0	18	18	9	9
	Max pooling	0	0	0	0	3	3	3	3
	Deconvolution	0	0	1	1	3	3	3	3
	Residual connections	0	0	0	0	0	0	12	12
	Concatenations	1	1	1	1	3	3	0	0

results in most of the cases. All voxels laying on the background region are given a weight of zero to avoid considering them in the optimization process. This decision was taken as non-brain regions were removed during pre-processing.

3) NETWORK TESTING

The steps to test a trained model on a given input MR volume are as follows. First, the whole volume is divided into patches. These patches are extracted from the entire input and not from specific regions. Second, the different patches are passed through the network to obtain a segmentation. Third, as there might be a degree of overlap between output probability maps, the final segmentation is provided through means of majority voting. The mode of the votes for each voxel is selected as consensed classification value. Convolutional-only networks classify only a subset of voxels. Commonly, networks dispense with outermost voxels and predict centermost ones only. For instance, the DM2D model receives a 27 × 27 patch and outputs classification values for voxels within a 9 × 9 rectangular region delimited by $(9, 9) - (9, 18) - (18, 18) - (18, 9)$. Thus, patches must be extracted with a step in between them of at most the output size of the network to be able to produce a valid whole brain segmentation, i.e. an incoming MR volume is tiled up so that the resulting output maps are adjacent to each other. Once patches are extracted from the scan, they are passed through the network and rearranged to reconstruct the segmentation volume.

4) SOFTWARE AND HARDWARE

All the architectures were implemented from scratch in Python, using the Keras library. From here on, our implementations of [28]–[31] are denoted by *DM*, *KK*, *UN* and *URN*, respectively. Relevant information per architecture is summarized in Table 2. All the experiments were run on a GNU/Linux machine box running Ubuntu 16.04, with 128GB RAM. CNN training and testing were carried out using a single TITAN-X PASCAL GPU (NVIDIA corp., United States) with 8GB RAM. The developed framework for this work is currently available to download at our research website. The source code includes architecture implementation and experimental evaluation scripts.

TABLE 3. Relevant information from the considered datasets. In the table, the elements to be considered are presented in the first column and the corresponding information from IBSR18, MICCAI 2012 and iSeg2017 are detailed in the following ones. In the row related to the number of scans (with GT), the number of training and test volumes is separated by a + sign. For both IBSR18 and iSeg2017, the evaluation is carried out using leave-one-out cross-validation.

Item	IBSR18	MICCAI 2012	iSeg2017
Target	Adult	Adult	Infant
Number of scans	18	15 + 20	10
Bias-field corrected	Yes	Yes	Yes
Intensity corrected	No	Yes	No
Skull stripped	No	No	Yes
Voxel spacing	0.8 × 0.8 × 1.5 0.9 × 0.9 × 1.5 1.0 × 1.0 × 1.5	0.5 × 0.5 × 0.5	1.0 × 1.0 × 1.0
Modalities	T1-w	T1-w	T1-w, T2-w

III. EXPERIMENTAL RESULTS

A. CONSIDERED DATASETS

We consider one publicly available repository and two challenges: Internet Brain Segmentation Repository 18 (IBSR18),[3] MICCAI Multi-Atlas Labeling challenge 2012 (MICCAI 2012)[4] and 6-month infant brain MRI segmentation (iSeg2017) [44],[5] respectively. The datasets were chosen since they have been widely used in the literature to compare different methods and, also, they contain infants and adults data, with different voxel spacing and a different number of scans. We believe that these two factors allow us to see how robust, general, and useful in different scenarios can be the algorithms. The organizers of the MICCAI 2012 challenge split the data into training and testing (10 and 13 volumes, respectively). To be consistent with the challenge and allow comparison with other strategies, we followed the same evaluation procedure. To use annotations of MICCAI 2012, we mapped all the labels to form the three tissue classes. Specific details of these datasets are presented in Table 3.

B. EVALUATION MEASUREMENTS

We used the Dice similarity coefficient (DSC) [45], [46] and the modified Hausdorff distance [47] to compare segmentation outputs against the ground truths. The DSC is used to

[3] http://www.nitrc.org/projects/ibsr

[4] http://masi.vuse.vanderbilt.edu/workshop2012

[5] http://iseg2017.web.unc.edu

determine the extent of overlap between a given segmentation and the ground truth. Given an input volume V, its corresponding ground truth $G = \{g_1, g_2, \ldots, g_n\}$, $n \in \mathbb{Z}$ and obtained segmentation output $S = \{s_1, s_2, \ldots, s_m\}$, $m \in \mathbb{Z}$ the DSC is mathematically expressed as

$$DSC\,(G, S) = 2\,\frac{|G \cap S|}{|G| + |S|}, \tag{1}$$

where $|\cdot|$ represents the cardinality of the set. The values for DSC lay within [0, 1], where the interval extremes correspond to null or exact similarity between the compared surfaces, respectively.

The MHD evaluates the distance between the sets of points forming the segmented and ground truth surfaces. Using the same notation as in Eq. 1, the MDH is calculated as follows

$$MHD\,(G, S) = \max\left\{{}^{95}K^{th}_{g_i \in G} d(g_i, S), {}^{95}\, K^{th}_{s_i \in S} d(s_i, G)\right\}, \tag{2}$$

where $d(a, \mathcal{B})$ corresponds to the minimum Euclidean distance between the point a and all the points in set $\mathcal{B}$ and ${}^{x}K^{th}_{b \in \mathcal{B}}$ represents the K-th ranked distance such that $K/|\mathcal{B}| = x\%$ [47]. For example, $x = 50$ corresponds to the median of the distances. We use the 95-th percentile MHD calculation over the original HD ($x = 100$) as the former is more robust to outliers in the segmentation. The values for MHD are positive decimal numbers greater or equal to zero, where zero indicates that the two surfaces exactly coincide – neglecting eccentric observations.

We consider the Wilcoxon signed-rank test to assess and report the statistical differences among architectures.

C. EVALUATION RESULTS

The evaluation conducted in this paper is four-fold. First, we investigate the effect of overlapping patches in both training and testing stages. Second, we assess the improvement of multi-modality architectures over single-modality ones. Third, we study whether patch size has any influence on the performance. Fourth, we compare the different models on the three considered datasets. Note that, for the sake of simplicity, the network' dimensionality is shown as a subscript (e.g. URN_{2D} denotes the 2D version of the URN architecture). The exact evaluation results are attached as Supplementary Material.

1) OVERLAPPING

To evaluate the effect of extracting overlapping patches in training and testing, we ran all the architectures on the three datasets contemplating three levels: null, medium and high (approximately 0%, 50% and 90%, respectively). On IBSR18 and iSeg2017, we carried out the evaluation using a leave-one-out cross-validation scheme. On MICCAI2012, we used the given training and testing sets.

The number of patches and average processing times for training, validating and testing each architecture in MICCAI2012, IBSR18, and iSeg2017 are condensed in Table 4. The average response time per voxel for DM_{2D}, DM_{3D}, KK_{2D}, KK_{3D}, UN_{2D}, UN_{3D}, URN_{2D}, and URN_{3D} was $0.14\mu s$, $0.11\mu s$, $0.16\mu s$, $0.09\mu s$, $1.70\mu s$, $0.81\mu s$, $0.79\mu s$, and $0.48\mu s$, respectively. On the one hand, 3D architectures output more voxels at a time and, hence, their voxel-wise classification response time is lower than their 2D analogues. On the other hand, the latter set of networks provides a considerably faster whole volume segmentation compared to their counterpart, in accordance with the literature [18]. Additionally, the fact that the overlapping policy led to a vast amount of training patches could explain why the networks converged in a few epochs: the more the patches, the longer the epochs, but the more the information provided to the network in a single pass.

The first test consisted of quantifying improvement between networks trained with either null or high degrees of overlap on training. The distribution of segmentation scores obtained on the three datasets is depicted in Fig. 4. In general, the models trained with patches extracted with a high extent of overlap yielded higher DSC and lower MHD values compared to when they were not. On the one hand, the sampling technique led to significantly higher DSC scores (p-value < 0.05) in 58 out of the 72 comparisons. On the other hand, overall, the precision of the method (measured in terms of inter-quartile range) regarding MHD increases but improvements were not significant in most of the cases (p-value > 0.05 in 51 out of 71 comparisons). Of note, there are enhancements in the boundaries but without taking into account most eccentric observations, MHD values are fairly similar. These three observations imply that the methods improve their segmentation, are more precise, but, in general, the borders of the segmentation masks do not change dramatically. In IBSR18, most of the models exhibited low DSC and notably high MHD scores when segmenting CSF. This outcome might be a consequence of the reduced number of samples available for this class (only the ventricular region). For iSeg2017, although the models trained with the overlapping sampling strategy yielded high DSC scores for the three classes, the MHD values show that the models had problems with delineating the limits between GM and WM accurately. The two groups of architectures exhibited opposite behaviors. U-shaped networks exhibited topmost improvements. This outcome is related to the fact that non-overlap may mean not enough samples. Instead, convolutional-only models evidenced the least increase. Since output patches are smaller, additional data can be extracted and used during training. Therefore, they can provide already accurate results. This fact is illustrated by the results of DM_{2D} and KK_{2D}.

The second test contemplated quantifying the improvement of extracting patches using combinations of the three considered degrees of overlap during training and testing. As mentioned previously, results were fused using a majority voting technique. We noted that the general trend was that the difference between results using null and high extends of overlap on testing time was not significant (p-values > 0.05). Also, the interquartile range remained similar regardless of

TABLE 4. Number of patches and average processing time for training, validating and testing each model in each dataset. The values for training and validation are of each one of the epochs and the ones for testing are of each volume.

		Overlap	Item	DM_{2D}	DM_{3D}	KK_{2D}	KK_{3D}	UN_{2D}	UN_{3D}	URN_{2D}	URN_{3D}
MICCAI2012	Training	Null (0%)	Patches	2 666 496	291 648	835 584	52 224	196 608	6 144	196 608	6 144
			Time (s)	176	360	104	170	21	29	19	31
		Intermediate (50%)	Patches	8 930 304	1 779 084	3 440 640	430 080	835 584	52 224	835 584	52 224
			Time (s)	591	2 195	428	1 397	90	244	79	267
		High (90%)	Patches	56 229 888	28 114 944	56 229 888	28 114 944	13 959 168	3 489 792	13 959 168	3 489 792
			Time (s)	3 720	34 684	7 000	91 356	1 504	16 335	1 316	17 820
	Validation	Null (0%)	Patches	666 624	72 912	208 896	13 056	49 152	1 536	49 152	1 536
			Time (s)	44	90	26	42	5	7	5	8
		Intermediate (50%)	Patches	2 232 576	444 771	860 160	107 520	208 896	13 056	208 896	13 056
			Time (s)	148	549	107	349	23	61	20	67
		High (90%)	Patches	14 057 472	7 028 736	14 057 472	7 028 736	3 489 792	872 448	3 489 792	872 448
			Time (s)	930	8 671	1 750	22 839	376	4 084	329	4 455
	Testing	Null (0%)	Patches	222 208	24 304	69 632	4 352	16 384	512	16 384	512
			Time (s)	15	30	9	14	2	2	2	3
		Intermediate (50%)	Patches	744 192	148 257	286 720	35 840	69 632	4 352	69 632	4 352
			Time (s)	49	183	36	116	8	20	7	22
		High (90%)	Patches	4 685 824	2 342 912	4 685 824	2 342 912	1 163 264	290 816	1 163 264	290 816
			Time (s)	310	2 890	583	7 613	125	1 361	110	1 485
IBSR18	Training	Null (0%)	Patches	1 304 576	142 688	425 984	26 624	106 496	3 328	106 496	3 328
			Time (s)	86	176	53	87	11	16	10	17
		Intermediate (50%)	Patches	4 243 200	845 325	1 703 936	212 992	425 984	26 624	425 984	26 624
			Time (s)	281	1 043	212	692	46	125	40	136
		High (90%)	Patches	27 262 976	13 631 488	27 262 976	13 631 488	6 815 744	1 703 936	6 815 744	1 703 936
			Time (s)	1 804	16 817	3 394	44 294	734	7 976	642	8 701
	Validation	Null (0%)	Patches	401 408	43 904	131 072	8 192	32 768	1 024	32 768	1 024
			Time (s)	27	54	16	27	4	5	3	5
		Intermediate (50%)	Patches	1 305 600	260 100	524 288	65 536	131 072	8 192	131 072	8 192
			Time (s)	86	321	65	213	14	38	12	42
		High (90%)	Patches	8 388 608	4 194 304	8 388 608	4 194 304	2 097 152	524 288	2 097 152	524 288
			Time (s)	555	5 174	1 044	13 629	226	2 454	198	2 677
	Testing	Null (0%)	Patches	100 352	10 976	32 768	2 048	8 192	256	8 192	256
			Time (s)	7	14	4	7	1	1	1	1
		Intermediate (50%)	Patches	326 400	65 025	131 072	16 384	32 768	2 048	32 768	2 048
			Time (s)	22	80	16	53	4	10	3	10
		High (90%)	Patches	2 097 152	1 048 576	2 097 152	1 048 576	524 288	131 072	524 288	131 072
			Time (s)	139	1 294	261	3 407	56	614	49	669
iSeg2017	Training	Null (0%)	Patches	602 112	65 856	193 536	12 096	43 008	1 344	43 008	1 344
			Time (s)	40	81	24	39	5	6	4	7
		Intermediate (50%)	Patches	1 906 688	379 848	774 144	96 768	193 536	12 096	193 536	12 096
			Time (s)	126	469	96	314	21	57	18	62
		High (90%)	Patches	12 386 304	6 193 152	12 386 304	6 193 152	3 096 576	774 144	3 096 576	774 144
			Time (s)	820	7 640	1 542	20 124	334	3 624	292	3 953
	Validation	Null (0%)	Patches	172 032	18 816	55 296	3 456	12 288	384	12 288	384
			Time (s)	11	23	7	11	1	2	1	2
		Intermediate (50%)	Patches	544 768	108 528	221 184	27 648	55 296	3 456	55 296	3 456
			Time (s)	36	134	28	90	6	16	5	18
		High (90%)	Patches	3 538 944	1 769 472	3 538 944	1 769 472	884 736	221 184	884 736	221 184
			Time (s)	234	2 183	441	5 750	95	1 035	83	1 129
	Testing	Null (0%)	Patches	75 264	8 232	24 192	1 512	5 376	168	5 376	168
			Time (s)	5	10	3	5	1	1	1	1
		Intermediate (50%)	Patches	238 336	47 481	96 768	12 096	24 192	1 512	24 192	1 512
			Time (s)	16	59	12	39	3	7	2	8
		High (90%)	Patches	1 548 288	774 144	1 548 288	774 144	387 072	96 768	387 072	96 768
			Time (s)	102	955	193	2 515	42	453	36	494

the method or dataset. Nevertheless, the general trend was an improvement of mean DSC of at least 1% in the overlapping cases. Another important observation from our experiments is that zero impact or slight degradation of the DSC and MHD values was noted when training with null overlap and testing with high overlap. Naturally, this situation is a consequence of merging predictions of a poorly trained classifier.

Medium level of overlap patch extraction, in both training and testing, led to improvement with respect to null degree cases but yielded lower values than when using a considerable extent of overlap. The general trend is: the more the extent

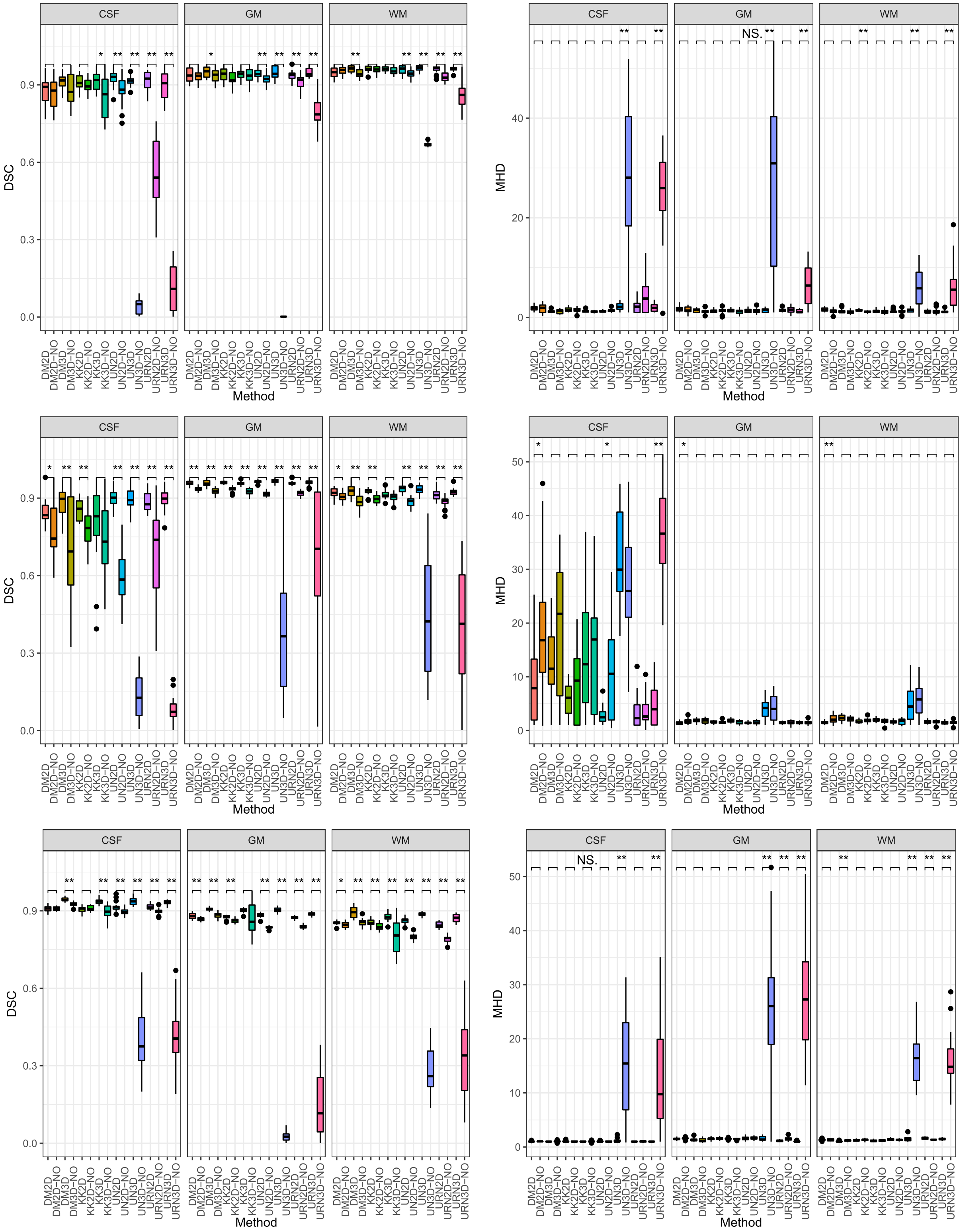


FIGURE 4. DSC (left column) and MHD (right column) values obtained using the null and high overlapping sampling in training. The suffix "-NO" on the name of the method means that the architecture was not trained using the sampling strategy. From top to bottom, boxplots for MICCAI2012, IBSR18, and iSeg2017, respectively. Differences between both versions of the same baseline architecture are highlighted with NS, *, and ** indicating a p-value > 0.1, < 0.05 and < 0.01, respectively.

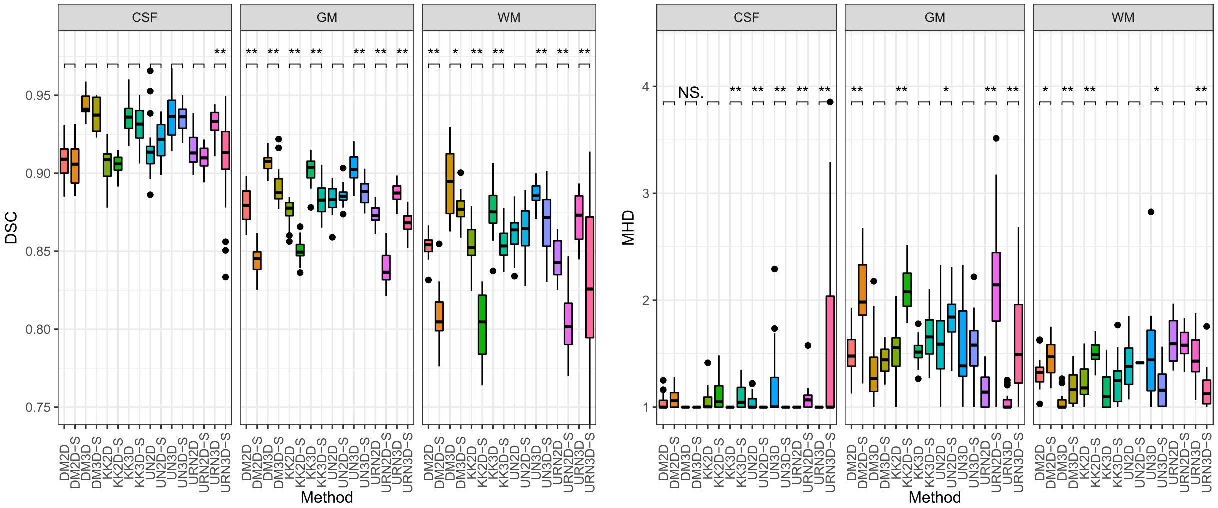


FIGURE 5. DSC (left) and MHD (right) values obtained using single and multiple input modalities. The suffix "-S" on the name of the method means that the architecture was single modality. Differences between both versions of the same baseline architecture are highlighted with NS, *, and ** indicating a p-value > 0.1, < 0.05 and < 0.01, respectively.

of overlap, the higher the overall performance of the method. The price to pay for using further levels of overlap is computational time and power since the number of samples to process increases exponentially. For example, given an input volume with dimensions 256 × 256 × 256 and a network producing output size of 32 × 32 × 32, the number of possible patches to be extracted following the null, medium and high overlap policies are 512, 3 375 and 185 193, respectively.

As overlapping sampling proved useful, the results showed in following sections correspond to the ones obtained using a high overlap in both training and testing.

2) SINGLE AND MULTIPLE MODALITIES

We performed leave-one-out cross-validation on the iSeg2017 dataset using the implemented 2D and 3D architectures to assess the effect of single and multiple imaging sequences on the final segmentation. The results of this experiment are shown in Fig. 5. Overall, the more the input modalities, the better the segmentation. In this case, two modalities not only allowed the network to achieve higher mean but also to reduce the IQR, i.e. networks are more accurate and precise. This behavior was evidenced regardless of architectural design or tissue type. For instance, while the best single modality strategy scored 0.937 ± 0.011, 0.891 ± 0.010 and 0.868 ± 0.016 for CSF, GM and WM, respectively; its multi-modality analogue yielded 0.944 ± 0.008, 0.906 ± 0.008 and 0.887 ± 0.017 for the same classes. Furthermore, in most of the cases, the strategies using both T1-w and T2-w obtained significantly higher DSC and lower MHD values compared to their single-modality counterparts. These results imply that multi-modality architectures obtained enhanced segmentation maps similar to the ground truth compared to the single-modality analogues as a direct consequence of providing the network with additional tissue contrast information (e.g. DSC increased and MHD decreases for CSF due to the contrast between this class and the other two in T2-w).

3) EFFECT OF PATCH SIZE

The effect of patch size in the overall performance has been investigated previously [48]–[51] and the overall trend has been that the larger the patch size, the more the contextual information provided to the network and, thus, the more enhanced the segmentation per se. Nonetheless, this particular experiment has not been carried out on 2D and 3D networks for tissue segmentation to the knowledge of the authors. We modified the baseline architectures by changing the input – and, consequently, output – patch size to study this matter. The size of the patches was selected in light of computational requirements (namely, the larger the patch, the more resources needed) and conditions imposed by the architectures (e.g. u-shaped networks may require input patch dimensions to be multiple of two due to max pooling modules). Information regarding the resulting designs is condensed in Table 5.

We performed a leave-one-out cross-validation on the iSeg2017 dataset using the various architectures to study the effect of patch size. The averaged DSC and MHD results of this trial are displayed in Fig. 6. On the one hand, the large u-shape architectures performed better than their medium-size counterpart (improved DSC and MHD mean and, in some cases, standard deviation as well) and significantly better than their small analogues (p-value < 0.05). On the other hand, convolutional-only networks did not exhibit the same pattern. In some cases, the small *DM* and *KK* architectures outperformed their medium and large versions, but improvements were not statistically significant. In some other cases, the medium variants led to the best segmentation outcomes. Overall, the large convolutional-only

TABLE 5. Implemented architectures to test patch size influence. The items into consideration appear on the first column. Note that there are two inputs for KK as the network has two processing branches.

	Item	DM_{2D}	DM_{3D}	KK_{2D}	KK_{3D}	UN_{2D}	UN_{3D}	URN_{2D}	URN_{3D}
Small	Input size	23 × 23	23 × 23 × 23	28 × 28 22 × 22	28 × 28 × 28 22 × 22 × 22	8 × 8	8 × 8 × 8	8 × 8	8 × 8 × 8
	Output size	5 × 5	5 × 5 × 5	14 × 14	14 × 14 × 14	8 × 8	8 × 8 × 8	8 × 8	8 × 8 × 8
	Number of parameters	458 254	1 835 654	466 830	4 345 966	1 931 620	5 606 308	995 108	2 623 844
Medium	Input size	27 × 27	27 × 27 × 27	32 × 32 20 × 20	32 × 32 × 32 20 × 20 × 20	16 × 16	16 × 16 × 16	16 × 16	16 × 16 × 16
	Output size	9 × 9	9 × 9 × 9	16 × 16	16 × 16 × 16	16 × 16	16 × 16 × 16	16 × 16	16 × 16 × 16
	Number of parameters	547 278	3 333 270	569 678	7 101 038	1 931 620	5 606 308	995 108	2 623 844
Large	Input size	37 × 37	37 × 37 × 37	36 × 36 26 × 26	36 × 36 × 36 26 × 26 × 26	32 × 32	32 × 32 × 32	32 × 32	32 × 32 × 32
	Output size	19 × 19	19 × 19 × 19	20 × 20	20 × 20 × 20	32 × 32	32 × 32 × 32	32 × 32	32 × 32 × 32
	Number of parameters	938 398	13 773 790	695 054	11 116 014	1 931 620	5 606 308	995 108	2 623 844

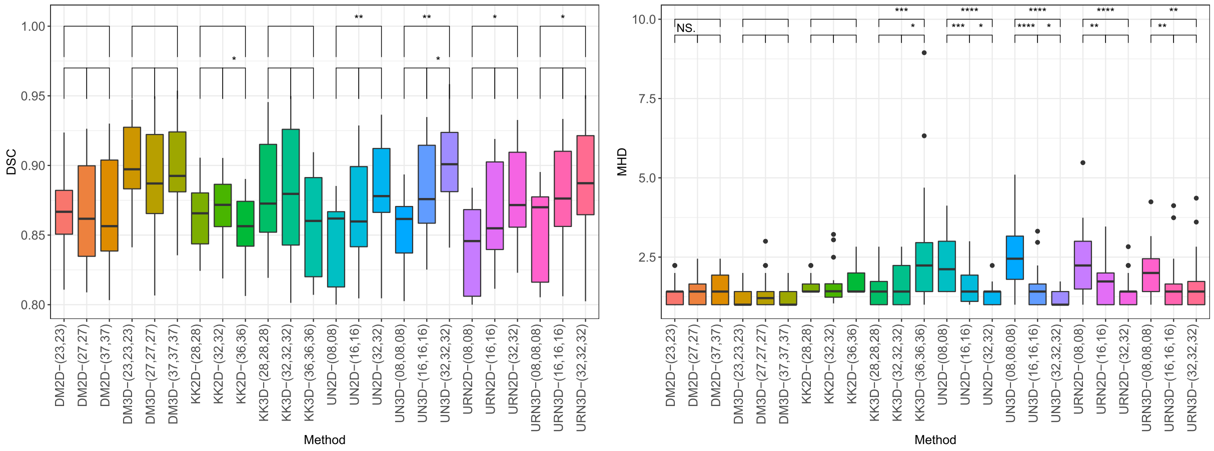


FIGURE 6. DSC (left) and MHD (right) values obtained by three variations of the baseline architectures concerning input and output patch size on the iSeg2017 dataset. Displayed results correspond to the average of scores obtained per class. The suffix indicates the input block dimensions. Differences between variations of the same baseline architecture are highlighted with ****, ***, **, *, and NS indicating a *p*-value < 0.0001, < 0.001, < 0.01, < 0.05, and > 0.10, respectively.

architectures led to inferior performance. This situation might have to do with the fact that the number of trainable parameters increases substantially between network adaptations. For instance, there is an increase in the number of parameters of approximately 256% between the smallest and the largest implementations of KK_{3D}. Consequently, these sizeable networks require more training samples to surpass their tinier versions.

We opted for using the largest u-shaped designs (i.e. patch dimensions equal to 32) and the intermediate convolutional-only networks (i.e. patch dimensions equal to 27 for DM and 32-20 for KK).

4) COMPARISON BETWEEN 2D AND 3D FCNN ARCHITECTURES

The eight architectures were evaluated using their best parameters according to the previous sections on the three different datasets: MICCAI2012, IBSR18, and iSeg2017. The distribution of segmentation scores for DSC and MHD is shown in Fig 7. The observations for each dataset vary. In MICCAI2012, the difference between 2D and 3D methods can be mostly perceived in the distance between data points, forming the CSF segmentation masks. In IBSR18, 3D algorithms yielded similar or lower performance than their 2D analogues. Taking into account the information in Table 3, 3D architectures might be slightly more affected by heterogeneity in voxel spacing. One of the reasons explaining this outcome is the lack of sufficient data which prevents 3D networks from understanding spacing and resolution variations, i.e. 3D networks might lack enough information to generalize properly. In iSeg2017, the 2D architectures displayed lower performance than their 3D counterparts, mostly concerning DSC. The networks performing the best on MICCAI2012, IBSR18, and iSeg2017 were UN_{3D}, URN_{2D} and UN_{2D}, and DM_{3D}, respectively.

Segmentation outputs obtained by the different methods on one of the volumes of the IBSR18 dataset are displayed in Fig. 8. Note that architectures using 2D information were trained with axial slices. Since 2D networks process each slice independently, the final segmentation is not necessarily accurate nor consistent: (i) subcortical structures exhibit unexpected shapes and holes, and (ii) sulci and gyri are not segmented finely. Thus, even if segmentation was carried out slice-by-slice, 3D approaches exhibit a smoother

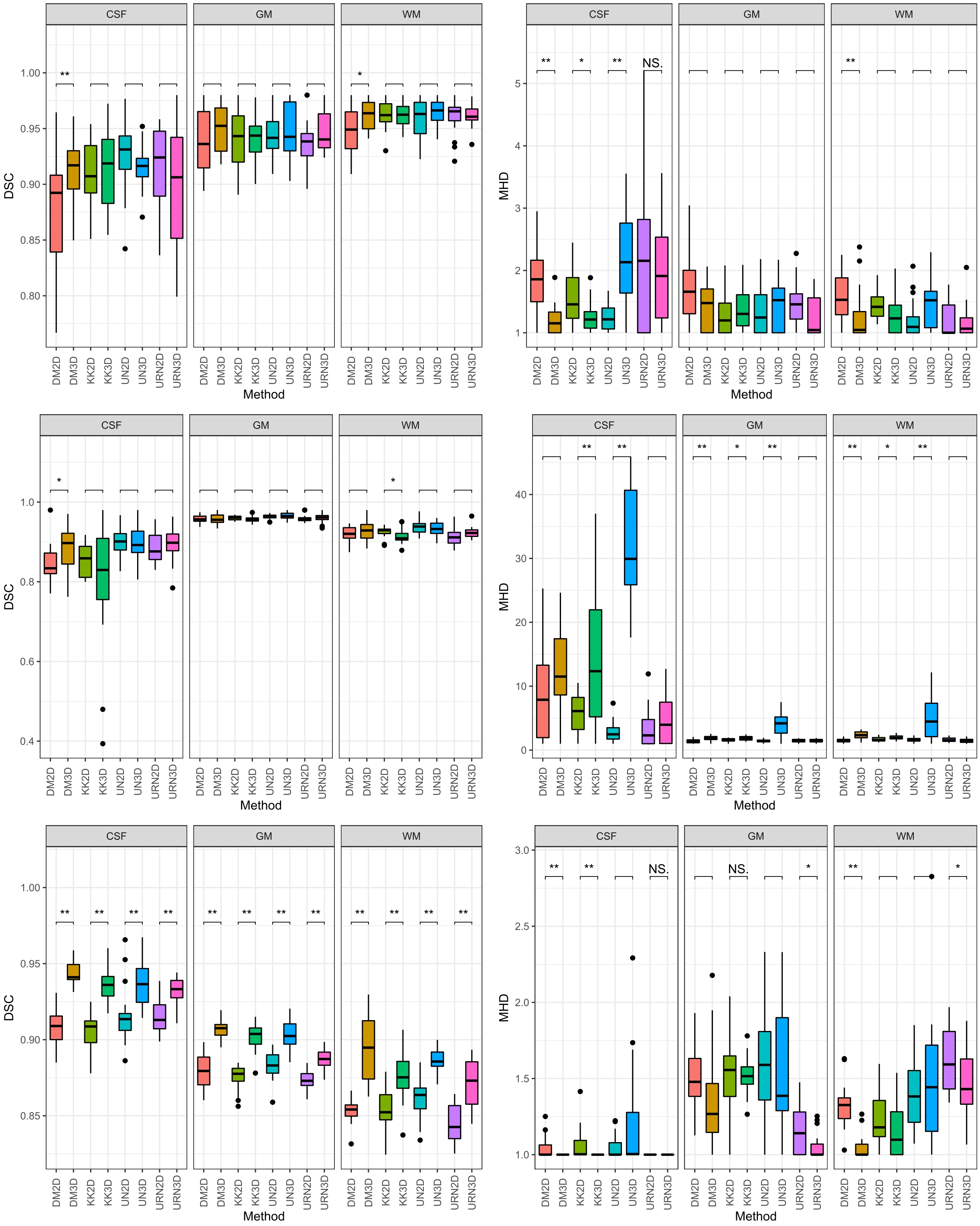
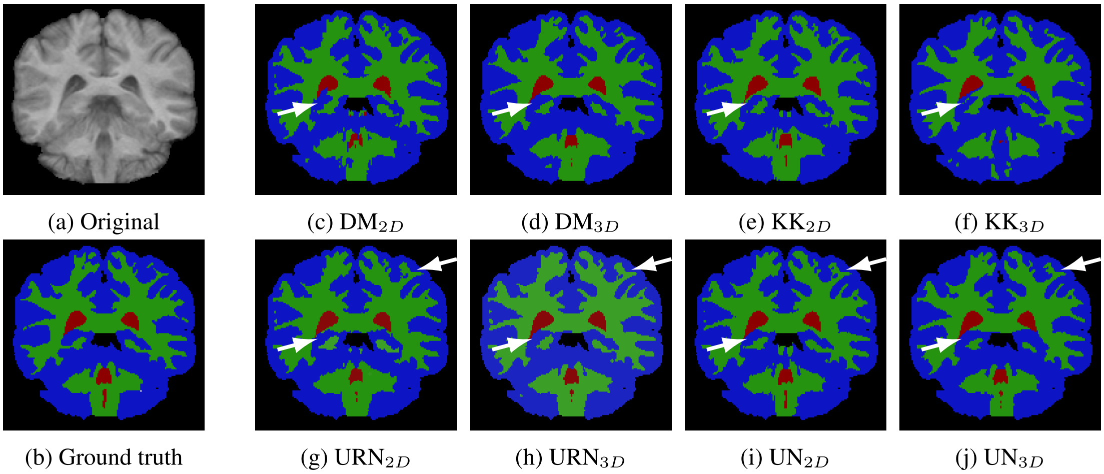


FIGURE 7. DSC (left column) and MHD (right column) values obtained using 2D and 3D versions of the same architecture. From top to bottom, boxplots for MICCAI2012, IBSR18, and iSeg2017, respectively. Differences between both versions of the same baseline architecture are highlighted with NS, *, and ** indicating a *p*-value > 0.1, < 0.05 and < 0.01, respectively.

segmentation presumably as they exploit the 3D nature of the MR volumes directly.

Another thing to note in Fig. 8f is that segmentation provided by KK_{3D} seems worse than the rest – even than its 2D analogue. The problem does not appear to be related to the number of parameters since KK_{3D} has less trainable elements compared to DM_{3D} and UN_{3D}, according to Table 2. This issue might be a consequence of the architectural design


(a) Original (c) DM$_{2D}$ (d) DM$_{3D}$ (e) KK$_{2D}$ (f) KK$_{3D}$

(b) Ground truth (g) URN$_{2D}$ (h) URN$_{3D}$ (i) UN$_{2D}$ (j) UN$_{3D}$

FIGURE 8. Segmentation output of the eight considered methods. The ground truth is displayed in (a) and the corresponding segmentation in (b-i). The colors for CSF, GM and WM, are red, blue and green, respectively. White arrows point out areas, where differences compared to the ground truth, are more noticeable. Architectures using 2D information were trained with axial slices.

itself. Anisotropic voxels and heterogeneous spacing may be affecting the low-resolution path of the network considerably. Hence, the overall performance is degraded.

5) COMPARISON WITH THE STATE OF THE ART AND CONVENTIONAL METHODS

We compared our best results for each dataset against two commonly used methods: SPM and FAST. In testing time, SPM, FAST and our models could reach a whole brain segmentation within 6 min. More importantly, SPM and FAST did not require GPUs as deep learning methods do. The results are shown in Supplementary Material. Overall, SPM and FAST led to significantly lower segmentation results compared to our best model (p-value < 0.001). Nonetheless, it is essential to understand the pros and cons of each strategy. On the one hand, conventional methods are suitable for many domains, but noise, intensity inhomogeneities [6]–[9], overlap between tissue distributions, and variations in shape (baby brain vs adult brain) and labeling protocols, hinder obtaining accurate outputs. On the other hand, the accuracy of CNN methods tends to decrease when the distribution of the test set differs significantly from one of the training set due to variations in imaging and labeling protocols. For example, a network trained on one of the datasets would not yield top results if tested on any of the other two since the voxel spacing, image quality and delineation of the different tissues would be different; a workaround would be to adapt the weights of the network to the new domain through transfer learning or, in a practical scenario, to map all the volumes to a standard template (e.g. MNI). We believe that fusing different approaches into a single framework (e.g., convolutional neural networks with tissue segmentation priors [26], [52]) is a promising area to explore to reach robustness.

In comparison with the state of the art, our methods showed similar or enhanced performance. First, the best DSC scores for IBSR18 were collected by Valverde *et al.* [9]. The highest values for CSF, GM and WM were 0.83 ± 0.08, 0.88 ± 0.04 and 0.81 ± 0.07; while our best approach scored 0.90 ± 0.03, 0.96 ± 0.01 and 0.93 ± 0.02, for the same classes. Second, the best-known values for tissue segmentation using the MICCAI 2012 dataset, were reported by Moeskops *et al.* [19]. Their strategy – a multi-path CNN – obtained 0.85 ± 0.04 and 0.94 ± 0.01 for CSF and WM, respectively; while our best approach yielded 0.92 ± 0.03 and 0.95 ± 0.02. In this case, we cannot establish a direct comparison of GM scores since in Moeskops' case, this class was subdivided into (a) cortical GM and (b) basal ganglia and thalami. Third, based on the results displayed in Fig 7, our pipeline using DM$_{3D}$ led to the best segmentation results on the iSeg2017 leave-one-out cross-validation. Hence, we submitted our approach to the online challenge under the team name "nic_vicorob".[6] The mean DSC values were 0.951, 0.910 and 0.885 for CSF, GM and WM, correspondingly; and we also ranked top-5 in six of the nine evaluation scenarios (three classes, three measures).

IV. DISCUSSION

In this paper, we analyzed quantitatively eight FCNN architectures inspired by the literature of brain segmentation related tasks. The networks were assessed through three experiments studying the importance of (i) overlapping patch extraction, (ii) multiple modalities, and (iii) network dimensionality. To ensure that all networks were evaluated under similar and favorable conditions, we used exactly the same pipeline (i.e. pre-processing, data preparation, segmentation, and post-processing), same optimizer, and same training and validation collections, and controlled overfitting by monitoring the network performance on the validation sets.

Our first experiment evaluated the impact of overlapping as sampling strategy at training and testing stages. This

[6] Results can be viewed at http://iseg2017.web.unc.edu/rules/results/

overlapping sampling is explored as a workaround to the commonly used data augmentation techniques in medical image tasks. This procedure can be used in this case as none of these networks processes a whole volume at a time, but patches of it. Based on our results, the technique proved beneficial as most of the strategies obtained significantly higher values than when not considered. In particular, the four u-shaped architectures exhibited a remarkable influence of this approach, presumably since more samples are used during training and the same area is seen with different neighboring regions, enforcing spatial consistency. Overlapping sampling in testing acted as a de-noising technique. We observed that this already-incorporated tool led to better performance than when absent as it helped filling small holes in areas expected homogeneous. The improvement was found to be at least 1%. Naturally, the main drawback of this technique was the expertise of the classifier itself, since it could produce undesired outputs when poorly trained.

Our second experiment assessed the effect of single and multiple imaging sequences on the final segmentation. We observed that regardless of the segmentation network, the inclusion of various modalities led to significantly better segmentations that when using a single imaging sequence. This situation may be a consequence of networks being able to extract valuable contrast information. Improvements were noted concerning the mean as well as the dispersion of the values yielded by the methods. Although this outcome is aligned with the literature [16], further trials on more datasets should be carried out to draw stronger conclusions. Future work should consider evaluating tissue segmentation in the presence of pathologies and using more imaging sequences such as FLAIR and PD.

Our third experiment examined the influence of patch size on the final segmentation. Although the literature reports that the larger the patch size, the better the segmentation due to additional contextual information [48]–[51], we observed that this trend is only followed when there are enough training samples to train such a larger network. This outcome is expected as the number of parameters increases substantially as the input patch dimensions augment. Unexpectedly, small-scale versions of the u-shaped networks were able to distinguish between classes and even though the performance was significantly lower than the large variants, the median DSC and MHD values were above 80% and below 2.50 pixels, respectively. However, it is crucial to recognize that this outcome might not hold on other tasks where tissues are split into sub-classes (e.g. whole brain parcellation or subcortical structure segmentation) as more contextual information might be needed to distinguish one class from another.

Our fourth experiment evaluated significant differences between 2D and 3D methods on the three considered datasets. Although 3D architectures tend to outperform their 2D analogues, the differences may not be significant. Moreover, in one of our datasets, IBSR18, 2D versions of the same baseline architecture could reach better segmentation scores than their 3D analogues. This outcome is a consequence of the heterogeneity of the data in IBSR18, i.e. 2D methods seem to be more resilient to issues regarding voxel spacing than 3D ones. Naturally, the immediate workaround to this issue is to re-sample during pre-processing. Additionally, the situation is likely to worsen when processing highly anisotropic volumes as there is less information in the third dimension.

According to our evaluation results, the segmentation performance is not strictly conditioned by the number of trainable parameters. For example, in IBSR18, 2D networks performed better than 3D networks due to issues of 3D networks to adapt to voxel spacing variations and image quality; in MICCAI2012, the differences between the performance of 2D networks in comparison to 3D networks were not significant overall; in MICCAI2012 and IBSR18, DM3D performed almost similar or worse than u-shaped networks even though it has at least 120% additional parameters. These outcomes suggest that some inherent architectural weaknesses and strengths define the overall performance of a network. Instead, we noted that specific modules allowed some networks to outperform some others. First, we observed that models using information from shallower layers in deeper ones achieved higher performance than those using multi-resolution information directly from the input volume, namely KK_{2D} and KK_{3D}. The difference was far more evident in datasets with heterogeneous volumes, e.g. in IBSR18 where scans vary in voxel spacing and image quality, where the latter strategy performed worse on average. This situation underlines the relevance of internal connections (e.g. residual connections and concatenation) for fusing multi-resolution information to segment more accurately. Second, we observed that concatenation and residual layers are present in all of the state-of-the-art networks. This might be related to the fact that these types of connections help in dealing with the degradation problem (i.e. deep networks tend to saturate and degrade rapidly) [53]. As the residual layers reduce the number of parameters to optimize, they should be preferred over concatenation modules. In fact, our experiments showed that two similar u-shaped networks using both approaches achieved similar results. Third, although u-shaped networks tended to outperform convolutional-only networks, no significant/remarkable difference was seen between both design patterns, except for processing times. In both training and testing, u-shaped networks segmented faster than convolutional-only networks: u-shaped models require extracting less number of patches and provide a more prominent output at a time.

Regarding general performance, two methods, DM_{3D} and UN_{3D}, obtained the best results. Of note, our specific implementation of the latter architecture required 30% fewer parameters to be set than the former and classified $\approx$ 32K voxels more at a time and completed a whole volume segmentation in half of the time or less. Although URN networks use slightly fewer parameters than UN architectures, both of them have comparable response times. In general, should the priority be overall processing time (training and testing), u-shaped networks are a suitable and recommended approach to address tissue segmentation instead of convolutional-only approaches.

Taking into account results reported in the literature, we achieved top performance for IBSR18, MICCAI2012 and iSeg2017 with our implemented architectures. Three important things to note in this work. First, none of these networks has explicitly been tweaked to the scenarios; a typical pipeline has been used. Hence, it is possible to compare them under similar conditions. Approaches expressly tuned for challenges may win, but it does not imply they will work identically – using the same set-up – on real-life scenarios. Second, although these strategies have shown acceptable results, more development on domain adaptation and transfer learning (zero-shot or one-shot training) should be carried out to implement them in medical centers. Third, we did not intend to compare the original works. The original works inspired our implementations, but general pipelines were not taken into account in here. In short, our study focused on understanding the architectural strengths and weaknesses of literature-like approaches.

V. CONCLUSIONS

In this paper, we have quantitatively analyzed 4×2 FCNN architectures, 2D and 3D, for tissue segmentation on brain MRI. These networks were implemented inspired by four recent works [28]–[31]. Among other characteristics, these methods comprised (i) convolutional-only and u-shaped architectures, (ii) single- and multi-modality inputs, (iii) 2D and 3D network dimensionality, (iv) varied implementation of multi-path schemes, and (v) different number of parameters. The networks were compared under a common evaluation framework: same training and test sets, processing pipeline, training and optimization schemes, and performance evaluation metrics. We believe that this setup allows us to establish a direct comparison between the different methods and, consequently, understand the underlying properties of the various architecture directives.

The eight networks were tested using three different well-known datasets: IBSR18, MICCAI2012, and iSeg2017. These datasets were considered since they were taken from infants and adults and acquired with diverse configuration parameters. To establish a direct architecture comparison, we fixed a common processing pipeline consisting of skull stripping, patch extraction, patch-wise segmentation, and voxel-wise majority voting. The testing scenarios evaluated the effect of overlapping sampling on both training and testing, patch size, multiple modalities, and 2D and 3D inputs on the final segmentation outputs. First, we observed that extracting patches with a certain degree of overlap among themselves led consistently to improved performance. The same approach on testing did not show a relevant improvement (around 1% in DSC), but it is a de-noising tool that comes along with the trained network. Second, we noted that using multiple modalities – when available – could provide the method with relevant tissue contrast information leading to significantly enhanced segmentation scores (p-value < 0.01) on average. Third, we observed that the larger the patch the network could process, the better the segmentation. However, the overall improvement is subject to computational resources and training sample availability: the larger the network, the more parameters to be tuned up and, hence, the more resources and training samples needed. Fourth, 3D methods tend to outperform their 2D counterpart. Nonetheless, the former group is more affected by variations in image resolution and voxel spacing.

In terms of architectural design, we found that specific modules allow some networks to perform better than others. First, multi-resolution information should be incorporated into the model by considering internal connections (namely, residual and concatenation connections) instead of explicitly providing the network with local and larger contextual information since voxel spacing heterogeneity affects the latter more. Second, residual and concatenation layers appear to be a popular design strategy as they help networks to cope with the degradation problem arising from building deep architectures. We implemented four u-shaped networks, two using residual connections and other two using concatenations, and observed that the accuracy of both of them was similar. Thus, residual connections should be preferred over concatenations as they reduce the number of parameters to optimize. Third, although u-shaped networks reached higher DSC and lower MHD values than convolutional-only architectures overall, no significant variation was evidenced between the best approaches on each group, except for response time. Evidently, the fact that the input and output sizes for former were the same contributed to obtaining whole brain segmentation in less time compared to the latter.

The networks implemented in this paper were able to deliver state-of-the-art results on IBSR18 and MICCAI2012. Our best approach on the iSeg2017 leave-one-out cross-validation assessment achieved top-5 performance on the first round of the real challenge in most of the online testing scenarios.

To encourage other researchers to use the implemented evaluation framework and FCNN architectures, we have released a public version of it at our research website.

ACKNOWLEDGMENT

The authors were with the Computer Vision and Robotics Institute, Universitat de Girona. The authors gratefully acknowledge the support of the NVIDIA Corporation with their donation of the TITAN-X PASCAL GPU used in this research. The authors would like to thank the organizers of the iSeg2017 challenge for providing the data.

REFERENCES

[1] Á. Rovira, M. P. Wattjes, M. Tintoré, C. Tur, T. A. Yousry, M. P. Sormani, N. De Stefano, M. Filippi, C. Auger, M. A. Rocca, F. Barkhof, F. Fazekas, L. Kappos, C. Polman, D. Miller, and X. Montalban, and On Behalf of the MAGNIMS Study Group, "Evidence-based guidelines: MAGNIMS consensus guidelines on the use of MRI in multiple sclerosis-clinical implementation in the diagnostic process," *Nature Rev. Neurol.*, vol. 11, no. 8, pp. 471–482, 2015.

[2] M. D. Steenwijk, J. J. G. Geurts, M. Daams, B. M. Tijms, A. M. Wink, L. J. Balk, P. K. Tewarie, B. M. J. Uitdehaag, F. Barkhof, H. Vrenken, and P. J. W. Pouwels, "Cortical atrophy patterns in multiple sclerosis are nonrandom and clinically relevant," *Brain*, vol. 139, no. 1, pp. 115–126, 2016.

[3] M. Filippi, M. A. Rocca, O. Ciccarelli, N. De Stefano, N. Evangelou, L. Kappos, A. Rovira, J. Sastre-Garriga, M. Tintorè, J. L. Frederiksen, C. Gasperini, J. Palace, D. S. Reich, B. Banwell, X. Montalban, F. Barkhof, and MAGNIMS Study Group, "MRI criteria for the diagnosis of multiple sclerosis: MAGNIMS consensus guidelines," *Lancet Neurol.*, vol. 15, no. 3, pp. 292–303, 2016.

[4] A. M. Mendrik *et al.*, "MRBrains challenge: Online evaluation framework for brain image segmentation in 3T MRI scans," *Comput. Intell. Neurosci.*, vol. 2015, Jan. 2015, Art. no. 813696.

[5] M. J. Cardoso, A. Melbourne, G. S. Kendall, M. Modat, N. J. Robertson, N. Marlow, and S. Ourselin, "AdaPT: An adaptive preterm segmentation algorithm for neonatal brain MRI," *NeuroImage*, vol. 65, pp. 97–108, Jan. 2013.

[6] L. P. Clarke, R. P. Velthuizen, M. A. Camacho, J. J. Heine, M. Vaidyanathan, L. O. Hall, R. W. Thatcher, and M. L. Silbiger, "MRI segmentation: Methods and applications," *Magn. Reson. Imag.*, vol. 13, no. 3, pp. 343–368, 1995.

[7] T. Kapur, W. E. L. Grimson, W. M. Wells, and R. Kikinis, "Segmentation of brain tissue from magnetic resonance images," *Med. Image Anal.*, vol. 1, no. 2, pp. 109–127, 1996.

[8] A. W.-C. Liew and H. Yan, "Current methods in the automatic tissue segmentation of 3D magnetic resonance brain images," *Current Med. Imag. Rev.*, vol. 2, no. 1, pp. 91–103, 2006.

[9] S. Valverde, A. Oliver, M. Cabezas, E. Roura, and X. Lladó, "Comparison of 10 brain tissue segmentation methods using revisited IBSR annotations," *J. Magn. Reson. Imag.*, vol. 41, no. 1, pp. 93–101, 2015.

[10] Y. Zhang, M. Brady, and S. Smith, "Segmentation of brain MR images through a hidden Markov random field model and the expectation-maximization algorithm," *IEEE Trans. Med. Imag.*, vol. 20, no. 1, pp. 45–57, Jan. 2001.

[11] D. L. Pham, "Robust fuzzy segmentation of magnetic resonance images," in *Proc. 14th IEEE Symp. Comput.-Based Med. Syst. (CBMS)*, Jul. 2001, pp. 127–131.

[12] D. W. Shattuck, S. R. Sandor-Leahy, K. A. Schaper, D. A. Rottenberg, and R. M. Leahy, "Magnetic resonance image tissue classification using a partial vol. model, " *NeuroImage*, vol. 13, no. 5, pp. 856–876, 2001.

[13] J. Ashburner and K. J. Friston, "Unified segmentation," *NeuroImage*, vol. 26, no. 3, pp. 839–851, 2005.

[14] J. Ashburner, G. Barnes, and C. Chen. (2012). *SPM8 Manual*. Accessed: May 18, 2017. [Online]. Available: http://www.fil.ion.ucl.ac.uk/

[15] S. Valverde, A. Oliver, E. Roura, S. González-Villà, D. Pareto, J. C. Vilanova, L. Ramió-Torrentà, À. Rovira, and X. Lladó, "Automated tissue segmentation of MR brain images in the presence of white matter lesions," *Med. Image Anal.*, vol. 35, pp. 446–457, Jan. 2017.

[16] W. Zhang, R. Li, H. Deng, L. Wang, W. Lin, S. Ji, and D. Shen, "Deep convolutional neural networks for multi-modality isointense infant brain image segmentation," *NeuroImage*, vol. 108, pp. 214–224, Mar. 2015.

[17] G. Litjens, T. Kooi, B. E. Bejnordi, A. A. A. Setio, F. Ciompi, M. Ghafoorian, J. A. van der Laak, B. van Ginneken, and C. I. Sánchez, "A survey on deep learning in medical image analysis," *Med. Image Anal.*, vol. 42, pp. 60–88, Dec. 2017.

[18] J. Bernal, K. Kushibar, D. S. Asfaw, S. Valverde, A. Oliver, R. Martí, and X. Lladó, "Deep convolutional neural networks for brain image analysis on magnetic resonance imaging: A review," *Artif. Intell. Med.*, vol. 95, pp. 64–81, Apr. 2019.

[19] P. Moeskops, M. A. Viergever, A. M. Mendrik, L. S. de Vries, M. J. N. L. Benders, and I. Išgum, "Automatic segmentation of MR brain images with a convolutional neural network," *IEEE Trans. Med. Imag.*, vol. 35, no. 5, pp. 1252–1261, May 2016.

[20] H. Chen, Q. Dou, L. Yu, J. Qin, and P.-A. Heng, "VoxResNet: Deep voxelwise residual networks for brain segmentation from 3D MR images," *NeuroImage*, vol. 170, pp. 446–455, Apr. 2017.

[21] K. Kushibar, S. Valverde, S. González-Villà, J. Bernal, M. Cabezas, A. Oliver, and X. Lladó, "Supervised domain adaptation for automatic subcortical brain structure segmentation with minimal user interaction," *Sci. Rep.*, vol. 9, pp. 1–15, Dec. 2019.

[22] M. F. Stollenga, W. Byeon, M. Liwicki, and J. Schmidhuber, "Parallel multi-dimensional LSTM, with application to fast biomedical volumetric image segmentation," in *Proc. 28th Int. Conf. Neural Inf. Process. Syst. (NIPS)*, Cambridge, MA, USA: MIT Press, 2015, pp. 2998–3006.

[23] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 3431–3440.

[24] M. Lyksborg, O. Puonti, M. Agn, and R. Larsen, "An ensemble of 2D convolutional neural networks for tumor segmentation," in *Proc. Scand. Conf. Image Anal.* Cham, Switzerland: Springer, 2015, pp. 201–211.

[25] A. Birenbaum and H. Greenspan, "Longitudinal multiple sclerosis lesion segmentation using multi-view convolutional neural networks," in *Proc. Int. Workshop Large-Scale Annotation Biomed. Data Expert Label Synth.* Cham, Switzerland: Springer, 2016, pp. 58–67.

[26] K. Kushibar, S. Valverde, S. González-Villà, J. Bernal, M. Cabezas, A. Oliver, and X. Lladó, "Automated sub-cortical brain structure segmentation combining spatial and deep convolutional features," *Med. Image Anal.*, vol. 48, pp. 177–186, Aug. 2018.

[27] M. Ghafoorian, N. Karssemeijer, T. Heskes, I. W. M. van Uden, C. I. Sanchez, G. Litjens, F.-E. de Leeuw, B. van Ginneken, E. Marchiori, and B. Platel, "Location sensitive deep convolutional neural networks for segmentation of white matter hyperintensities," *Nature Sci. Rep.*, vol. 7, no. 1, 2017, Art. no. 5110.

[28] Ö. Çiçek, A. Abdulkadir, S. S. Lienkamp, T. Brox, and O. Ronneberger, "3D u-net: Learning dense volumetric segmentation from sparse annotation," in *Proc. Int. Conf. Med. Image Comput.-Assist. Intervent*. Cham, Switzerland: Springer, 2016, pp. 424–432.

[29] J. Dolz, C. Desrosiers, and I. B. Ayed, "3D fully convolutional networks for subcortical segmentation in MRI: A large-scale study," *NeuroImage*, vol. 170, pp. 456–470, Apr. 2018.

[30] R. Guerrero, C. Qin, O. Oktay, C. Bowles, L. Chen, R. Joules, R. Wolz, M. Valdés-Hernández, D. Dickie, J. Wardlaw, and D. Rueckert, "White matter hyperintensity and stroke lesion segmentation and differentiation using convolutional neural networks," *NeuroImage, Clin.*, vol. 17, pp. 918–934, Jan. 2018.

[31] K. Kamnitsas, C. Ledig, V. F. J. Newcombe, J. P. Simpson, A. D. Kane, D. K. Menon, D. Rueckert, and B. Glocker, "Efficient multi-scale 3D CNN with fully connected CRF for accurate brain lesion segmentation," *Med. Image Anal.*, vol. 36, pp. 61–78, Feb. 2017.

[32] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput.-Assist. Intervent*. Cham, Switzerland: Springer, 2015, pp. 234–241.

[33] F. Milletari, N. Navab, and S.-A. Ahmadi, "V-Net: Fully convolutional neural networks for volumetric medical image segmentation," in *Proc. 4th Int. Conf. 3D Vis. (3DV)*, 2016, pp. 565–571.

[34] T. Brosch, L. Y. W. Tang, Y. Yoo, D. K. B. 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, no. 5, pp. 1229–1239, May 2016.

[35] J. Kleesiek, G. Urban, A. Hubert, D. Schwarz, K. Maier-Hein, M. Bendszus, and A. Biller, "Deep MRI brain extraction: A 3D convolutional neural network for skull stripping," *NeuroImage*, vol. 129, pp. 460–469, Apr. 2016.

[36] D. Nie, L. Wang, Y. Gao, and D. Sken, "Fully convolutional networks for multi-modality isointense infant brain image segmentation," in *Proc. IEEE 13th Int. Symp. Biomed. Imag. (ISBI)*, Apr. 2016, pp. 1342–1345.

[37] M. Shakeri, S. Tsogkas, E. Ferrante, S. Lippe, S. Kadoury, N. Paragios, and I. Kokkinos, "Sub-cortical brain structure segmentation using F-CNN's," in *Proc. IEEE 13th Int. Symp. Biomed. Imag. (ISBI)*, Apr. 2016, pp. 269–272.

[38] P. Moeskops, M. Veta, M. W. Lafarge, K. A. Eppenhof, and J. P. Pluim, "Adversarial training and dilated convolutions for brain MRI segmentation," in *Proc. Deep Learn. Med. Image Anal. Multimodal Learn. Clin. Decis. Support*. Cham, Switzerland: Springer, 2017, pp. 56–64.

[39] B. Xu, Y. Chai, C. M. Galarza, C. Q. Vu, B. Tamrazi, B. Gaonkar, L. Macyszyn, T. D. Coates, N. Lepore, and J. C. Wood, "Orchestral fully convolutional networks for small lesion segmentation in brain MRI," in *Proc. IEEE 15th Int. Symp. Biomed. Imag. (ISBI)*, Apr. 2018, pp. 889–892.

[40] S. R. Hashemi, S. S. M. Salehi, D. Erdogmus, S. P. Prabhu, S. K. Warfield, and A. Gholipour, "Asymmetric loss functions and deep densely-connected networks for highly-imbalanced medical image segmentation: Application to multiple sclerosis lesion detection," *IEEE Access*, vol. 7, pp. 1721–1735, 2019.

[41] A. Clèrigues, S. Valverde, J. Bernal, J. Freixenet, A. Oliver, and X. Lladó, "Acute and sub-acute stroke lesion segmentation from multimodal MRI," *coRR*, vol. abs/1810.13304, pp. 1–11, Oct. 2018.

[42] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik, "Hypercolumns for object segmentation and fine-grained localization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 447–456.

[43] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. 3rd Int. Conf. Learn. Represent. (ICLR)*, May. 2015, pp. 1–15.

[44] L. Wang *et al.*, "Benchmark on automatic 6-month-old infant brain segmentation algorithms: The iSeg-2017 challenge," *IEEE Trans. Med. Imag.*, to be published.

[45] L. R. Dice, "Measures of the amount of ecologic association between species," *Ecology*, vol. 26, no. 3, pp. 297–302, 1945.

[46] W. R. Crum, O. Camara, and D. L. G. Hill, "Generalized overlap measures for evaluation and validation in medical image analysis," *IEEE Trans. Med. Imag.*, vol. 25, no. 11, pp. 1451–1461, Nov. 2006.

[47] M.-P. Dubuisson and A. K. Jain, "A modified Hausdorff distance for object matching," in *Proc. 12th IAPR Int. Conf. Pattern Recognit. (ICPR)*, vol. 1, Oct. 1994, pp. 566–568.

[48] C. Farabet, C. Couprie, L. Najman, and Y. LeCun, "Learning hierarchical features for scene labeling," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1915–1929, Aug. 2013.

[49] P. O. Pinheiro and R. Collobert, "Recurrent convolutional neural networks for scene labeling," in *Proc. 31st Int. Conf. Int. Conf. Mach. Learn.*, vol. 32, 2014, pp. 82–90.

[50] H. Li, R. Zhao, and X. Wang, "Highly efficient forward and backward propagation of convolutional neural networks for pixelwise classification," *coRR*, vol. abs/1412.4526, pp. 1–10, Dec. 2014.

[51] J. Hamwood, D. Alonso-Caneiro, S. A. Read, S. J. Vincent, and M. J. Collins, "Effect of patch size and network architecture on a convolutional neural network approach for automatic segmentation of OCT retinal layers," *Biomed. Opt. Express*, vol. 9, no. 7, pp. 3049–3066, 2018.

[52] J. Bernal, M. Salem, K. Kushibar, A. Clèrigues, S. Valverde, M. Cabezas, S. Gonzáles-Villa, J. W. Salvi, A. Oliver, and X. Lladó. (2018). *MR Brain Segmentation Using an Ensemble of Multi-Path U-Shaped Convolutional Neural Networks and Tissue Segmentation Priors*. Accessed: Feb. 20, 2019. [Online]. Available: http://mrbrains18.isi.uu.nl/wp-content/uploads/2018/11/nic_vicorob.pdf

[53] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 770–778.

JOSE BERNAL received the B.Sc. degree in computer engineering from the Universidad del Valle, in 2014, and the M.Sc. degree in computer vision and robotics from the Université de Bourgogne, the Universitat de Girona, and the Heriot-Watt University, in 2017. He is currently pursuing the Ph.D. degree in technology with the Universitat de Girona.

KAISAR KUSHIBAR received the B.Sc. degree in computer engineering from International IT University, in 2015, and the M.Sc. degree in computer vision and robotics from the Université de Bourgogne, the Universitat de Girona, and the Heriot-Watt University, in 2017. He is currently pursuing the Ph.D. degree in technology with the Universitat de Girona.

MARIANO CABEZAS received the B.Sc. degree in computer engineering, the M.Sc. degree in automation, computation and systems, and the Ph.D. degree in technology from the Universitat de Girona, in 2009, 2010, and 2013, respectively. He is currently a Postdoctoral Researcher with the VICOROB Research Institute, Universitat de Girona.

SERGI VALVERDE received the B.Sc. degree in computer engineering from the Universitat de Girona, in 2010, the M.Sc. degree in computer vision and robotics from the Université de Bourgogne, the Universitat de Girona, and Heriot-Watt University, in 2012, and the Ph.D. degree in technology from the Universitat de Girona, in 2016. He is currently a Postdoctoral Researcher with the VICOROB Research Institute, Universitat de Girona.

ARNAU OLIVER received the B.Sc. degree in physics from the Universitat Autònoma de Barcelona, in 1999, and the B.Sc. degree in computer science and the Ph.D. degree in information technology from the Universitat de Girona, in 2002 and 2007, respectively. Dr. Oliver coordinated the Erasmus + Joint Master's Degree in Medical Imaging and Applications (MAIA). Also, he is the responsible in UdG for the Interuniversity Doctoral Programme in Bioinformatics and member of the academic board of the Doctoral Programme in Technology.

XAVIER LLADÓ received the B.Sc. degree in computer science and the Ph.D. degree in computer engineering from the Universitat de Girona, in 1999 and 2004, respectively. From 2004 to 2006, he was a Postdoctoral Research Assistant with the Department of Computer Science, Queen Mary University of London, with Dr. L. Agapito. He is currently a Full Professor with the Department of Computer Architecture and Technology, Universitat de Girona. He is also currently the Head of the Department of Computer Architecture and Technology, Universitat de Girona. He has published more than 200 papers in peer-reviewed journals and conferences. He is also a Reviewer of more than 20 JCR journals and 15 international conferences and he has been a member of the Program Committee and the organization of several conferences.