A preliminary exploration into top-down and bottom-up deep-learning approaches to localising neuro-interventional point targets in volumetric MRI

Enora Giffard, Pierre Jannin, John S H Baxter

A Preliminary Exploration into Top-Down and Bottom-Up Deep-Learning Approaches to Localising Neuro-Interventional Point Targets in Volumetric MRI

Enora Giffard, Pierre Jannin and John S.H. Baxter *

LTSI - INSERM UMR 1099,Université de Rennes, F-35000, Rennes, France.

*Corresponding author(s). E-mail(s): john.baxter@univ-rennes.fr;

Acknowledgments

We would like to acknowledge SYNEIKA (Rennes, France) for the use of their TMS targeting dataset.

Abstract

Purpose: Point localisation is a critical aspect of many interventional planning procedures, specifically representing anatomical regions of interest or landmarks as individual points. This could be seen as analogous to the problem of *visual search* in cognitive psychology, in which this search is performed either: *Bottom-Up*, constructing increasing abstract and coarse-resolution features over the entire image; or *Top-Down*, using contextual cues from the entire image to refine the scope of the region being investigated. Traditional convolutional neural networks use the former, but it is not clear if this is optimal. This article is a preliminary investigation as to how this motivation affects 3D point localisation in neuro-interventional planning.

Methods: Two neuro-imaging datasets were collected: one for cortical point localisation for repetitive transcranial magnetic stimulation and the other for sub-cortical anatomy localisation for deep-brain stimulation. Four different frameworks were developed using Top-Down vs Bottom-Up paradigms as well as representing points as co-ordinates or heatmaps. These networks were applied to point localisation for transcranial magnetic stimulation and subcortical anatomy localisation. These networks were evaluated using cross-validation and a varying number of training datasets to analyse their sensitivity to quantity of training data. **Results:** Each network shows increasing performance as the amount of available training data increases, with the co-ordinate-based Top-Down network consistently outperforming the others. Specifically, the Top-Down architectures tend to outperform the Bottom-Up ones. An analysis of their memory consumption also encourages the Top-Down co-ordinate based architecture as it requires significantly less memory than either Bottom-Up architectures or those representing their predictions via heatmaps. **Conclusion:** This paper is a preliminary foray into a fundamental aspect of machine learning architectural design: that of the Top-Down / Bottom-Up divide from cognitive psychology. Although there are additional considerations within the particular architectures investigated that could affect these results and the number of architectures investigated is limited, our results do indicate that the less commonly used Top-Down paradigm could lead to more efficient and effective architectures in the future.

1 Introduction

As neural networks tend to grow deeper, often with better results, there seems to be a limit to what degree of depth actually improves accuracy [1]. The amount of RAM needed to store parameters but also feature maps from convolutional neural networks is greatly increased. Deep learning has reached a much larger public in the last few years and there's a demand for less resource-consuming neural network architectures for both economical and ecological reasons. These resources include annotated training data, but also computational resources, notably memory. For example, dead neurons are individual artificial neurons which require memory, but are rarely if ever active, and thus do not contribute to the model as a whole. This is even more of an issue for

convolutional neural networks, in which individual neurons are replaced with convolutions representing entire images [2].

Localisation problems are of particular interest due to their prevalence throughout the different sub-fields of computer vision. In these problems, a network is designed to identify a series of points or regions-of-interest in an image that represent pre-determined objects of interest. This is different from segmentation as only the position of the structure is needed rather than its precise delineation. One of the first widely-used localisation methods in deep learning was the Regions with CNN features (R-CNN) [3] , which first used an object-agnostic region detector to suggest a large number of potential objects, then resized them to a fixed size for a traditional convolutional neural network to determine what object they represented, if any. FastR-CNN [4] extended this method using a non-agnostic region detection method that, rather than using object-agnostic features to determine if a region was of interest, trained a series of object-specific features which can be directly used for the later network that classifies the potential regions. This was motivated by a desire to save both time and memory, allowing for convolutional features to be shared across the multiple disparate tasks performed in the network as a whole. FasterR-CNN [5] again changes the region detector to make it more time- and memory-efficient. These frameworks are designed for not only localisation, but also detecting a previously unknown number of regions of interest. For example, these frameworks have been originally motivated by the problem of detecting and localising cars in 2D images for autonomous driving applications. It could well be that the image doesn't feature any car at all, such as on a deserted road. If there are cars, the network doesn't know how many as this can vary heavily from image to image. In addition, some of them would appear bigger than others due to proximity and some others of them might not be fully

visible, and these mixed localisation (i.e. region detection) and identification algorithms have thus included bounding boxes to capture these notions.

The medical field is a particularly interesting field of application for volumetric image analysis. Processing three-dimensional convolutional neural networks can be prohibitively memory consuming even with a reduced depth and width and therefore aren't easy to implement in medical equipment such as neuronavigation software. Therefore, it is unsurprising that there are far fewer published methods for point localisation in neuro-imaging contexts, let alone volumetric images. Sugimori *et al.* [6] used an R-CNN to identify the anterior and posterior commissures, which are used to define a neurological co-ordinate system. One notable aspect of this system was that it did not use the full volumetric MRI, but instead the 2D slice containing the two commissures, which may limit its clinical utility. Yang *et al.* [7] also used specific 2D images for finding these points, although using a more traditional CNN architecture. The issue with knowing these slices in advance was pointed out by Gohel *et al.* [8] who developed a framework for estimating the commissure locations in 3D using the axial, sagittal, and coronal localisers which do not themselves contain said points. A different approach as well as different targets were used by Baxter *et al.* [9] who developed a multi-resolution convolutional neural network architecture taking its inspiration from the human psychological active visual search Top-Down strategy [10]. This architecture was designed for the localisation of cortical repetitive transcranial magnetic stimulation targets within volumetric MRI and may be applied to other localisation problems within images which require a lot of memory. Li *et al.* [11] took a different mixed approach in which a single network was used for rating whether or not a sagittal slice might contain the points of interest (the right and left internal

acoustic pores and posterior cavernous sinuses in their case) followed by a 2D hour-glass network to find said points within the identified slice.

The two key differences between these neurosurgical localisation problems and the types of localisation problems more generally addressed in computer vision, which have been discussed previously, are that the identity of the points is known ahead of time (i.e. there is a fixed number of them which are guaranteed to be present and should also be labelled or otherwise distinguished from each other); and that their locations are much more highly dependent on global context, specifically brain anatomy. There are other differences, for example, because the images are 3D and always include the whole brain, there is no problem of objects looking bigger than they are or being hidden behind other objects. So if these problems are so different, why not use different methods to address them? That's what our brain is doing, studies of human active visual search have shown. In the next section, we will get into more detail about the two different strategies, Bottom-up and Top-down, our brain is using to solve localisation problems and which one is more efficient for which kind of problem. We will see how these strategies of the human brain influence the way artificial neural networks are designed.

1.1 Human visual search

Artificial neural networks first took their inspiration from their counterparts in the human brain and it is thus interesting to look at cognitive strategies our brains use when performing various tasks in order to implement new kinds of artificial neural network architectures. When actively looking for a specific object surrounded by distractors, the human brain has different strategies regarding attention and gaze allocation, two of which being Bottom-Up and Top-Down attentional selection [10].

Bottom-Up attentional selection is feature-driven, using a parallel processing of all the items laying in the subject's field of view to locate the one they are looking for, based on a few varying features. This strategy works best when the number of features allowing the subject to differentiate distractors from the target is small or when the target is very salient. The target also has to be in the subject's field of view in order for this strategy to be successful. This has traditionally been the basis for localisation networks such as R-CNN and its variants [3, 4, 5, 6], which process an entire image at its full resolution then subsequently uses pooling or other forms of aggregation to merge these bottom-up features together into region proposals in a method analogous to how convolutional neural networks perform other separate tasks such as image classification. Basic CNNs also can be used to perform localisation tasks and use a bottom-up strategy to do so.

Top-Down attentional selection on the other hand tends to be favoured by the human brain when the target isn't in the subject's immediate field of view, isn't very salient compared to the distractors, or when the context of the search space itself provides information about the location (for example, knowing what areas of the image are a road can lead to easier localisation of cars). This strategy performs a serial allocation of attention, based on the subject's prior knowledge and cognition. The subject thus directs their gaze at a particular zone because they think the target is likely to be located there, or analyses the different dimensions of all items one after the other in a defined order. From an optimisation perspective, this allows for local minima in the search process to be avoided by first examining the whole problem at a coarse resolution, then iteratively refining both the solution and the search space. An example of this is deformable image registration which has long used a hierarchical approach to use the solution to simple, more coarse-grained problems, to initialise and

limit the search space for finer-grained refinements [12]This can also apply to convolutional neural networks. Baxter *et al.* 's [9] multi-resolution neural network architecture was designed to mimic Top-Down attentional selection, using low resolution images to determine what areas of the volume to continue searching in at a higher resolution, conserving memory of, but not processing, the whole image at a high resolution.

It is clear that human visual search can motivate different automatic point localisation architectures, but it is unclear if, given a localisation problem of a certain type, the strategy used by the human brain also is the best choice for artificial neural networks.

Contributions

In this paper we compare two paradigmatic Top-Down architectures with two paradigmatic Bottom-Up convolutional neural network architecture for 3D point localisation in neuro-imaging. This comparison is made in terms of memory consumption, accuracy, and impact of the training dataset size. This is done by using these architectures to solve two localisation tasks. The first one is the localisation of 12 repetitive Transcranial Magnetic Stimulation treatment points on 3D MR images and the second one is the localisation of 30 sub-cortical structures used in Deep Brain Stimulation preoperative planning on another dataset of 3D MR images.

2 Neuro-interventional Pointing Tasks

Several neuro-interventions rely on pointing tasks during their planning, either directly or as a part of a larger image processing pipeline. These tasks occur when there is a specific anatomical landmark that needs to be identified (such

as the anterior and posterior commissures which define the Talairach co-ordinate space [6]) or neuro-anatomical regions that are sufficiently small to represent as singular points.

2.1 Cortical Transcranial Magnetic Stimulation

Transcranial magnetic stimulation (TMS) is a relatively new treatment option for a variety of neurological and psychological disorders varying from clinical depression to chronic pain [13]. The key shared characteristic of these disorders is that they imply abnormal cortical behaviour which can be corrected through repeated disruption which is provided externally via the local application of a varying magnetic field. The key task in repetitive TMS is therefore to identify the cortical regions requiring stimulation based on the patient's symptomatology. For most treatment approaches, these target regions are specific small neuro-anatomic regions in the frontal cortex, such as the dorsolateral prefrontal for clinical depression [14], or particular subregions of the primary motor cortex for chronic pain in the corresponding bodily region [15]. Although these particular regions are unique, there may be some variability in terms of which are actively being targeted, especially for psychiatric disorders which are less clearly neurologically localised [14].

Thus, the primary goal for pre-operative TMS planning is the identification of these points as to inform the operator of where to position the external stimulator. This targeting was originally performed blindly, using landmarks on the patient's skull as well as finding a single calibration point associated with the hand area of the primary motor cortex [16]. This has a relatively low accuracy in terms of finding the precise anatomical area (on the order of 1-1.5 cm [16]). More recent structural MRI guided TMS interventions have lowered that error although the human expert accuracy of delineating the

functionally defined treatment points in MRI is still on the order of 5-10 mm [9]. Despite these errors, the stimulation can still produce a clinically-relevant effect as the stimulation region can be several square centimetres although with diminishing strength further from the centre [17]. Thus, improvement in these accuracies may lead to more visible improvements using a potentially weaker field, although this has yet to be confirmed in clinical trials.

2.2 Small Subcortical Anatomy Localisation for Deep Brain Stimulation

Deep brain stimulation (DBS) is a treatment for various neurological and neurodegenerative disorders, notably Parkinson's disease, in which an electrode is placed at or within a particular structure in the basal ganglia to provide continuous stimulation [18]. These structures tend to be very small, leading to difficulties in their accurate segmentation especially with modern machine learning approaches whose statistical nature favours larger, more well-defined regions [19]. One way to alleviate this difficulty is by automatically cropping the image to a smaller region-of-interest containing only the anatomical structure of interest. To do so, however, requires being able to estimate its position within the entire volumetric image, a point localisation task. The amount of cropping thus depends on the accuracy of the localisation, with better accuracies allowing for more aggressive cropping to be used which improves the speed and accuracy of the downstream segmentation algorithm. Given the size of these structures, accuracies on the order of 1 cm render this cropping procedure feasible (i.e. cropping the image to approximately 1% of its original size) however better accuracies are still desired to further inform the downstream segmentation model and reduce cropped image sizes.

3 Material and Methods

3.1 Imaging Data

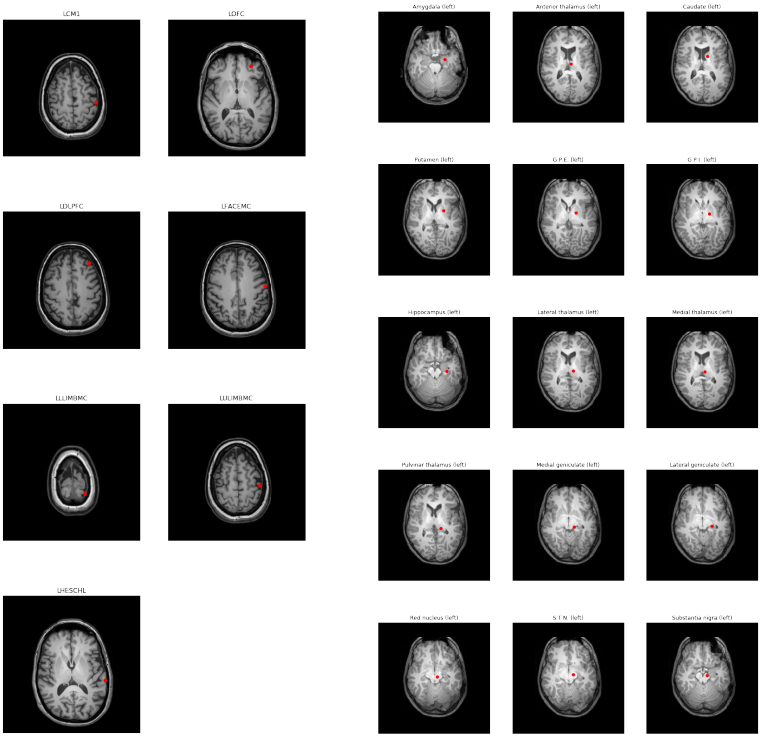
Two datasets were used for the comparison of the top-down and bottom-up architectures, each composed of a number of T1-weighted MR images and for each of these images the coordinates of a list of points, which the networks aimed at localising.

3.1.1 Cortical TMS Dataset

This dataset is the same as that used by Baxter *et al.* [9] for identifying cortical points-of-interest for repetitive Transcranial Magnetic Stimulation (TMS). It contains 26 T1-weighted MR images from 26 patients from various hospital centers. They were normalised, using approximate min-max normalisation (95% percentiles used instead of absolute max and min), and resampled to 256x256x256 voxels with an isotropic voxel size of 1x1x1 mm using the Convert3D tool[1]. 12 cortical points have been annotated by expert neurologists. The hand region of the left primary motor cortex (LCM1) used for calibration and the five points used for the treatment of psychiatric disorders (i.e. orbitofrontal cortex on both sides, or LOFC and ROFC, dorsolateral prefrontal cortex on both sides, or LDLPFC and RDLPFC, and left Heschl gyrus, or LHESCHL) have been annotated by a single expert. The 6 points used in the treatment of chronic pain (face, upper limb and lower limb regions of the motor cortex on both sides, or LFACEMC, RFACEMC, LULIMBMC, RULIMBMC, LLLIMBMC and RLLIMBMC) have been annotated by three experts (Tab. 1). These multiple annotations allow to measure expert variability [9] and to determine a consensus to improve the accuracy of the gold standard for each

[1] http://www.itksnap.org/pmwiki/pmwiki.php?n=Convert3D.Convert3D, The specific commands used were *swapdim*, *-resample-mm*, and *-pad-to*.

point. Three patients have missing annotations for the left Heschl's gyrus, with two also missing annotations for the hand region of the left primary motor cortex and the dorsolateral prefrontal cortices on both sides. The location of these points for the left side of the brain of one patient of the dataset is shown in Fig. 1a.



(a) Treatment points for the TMS dataset

(b) Sub-cortical anatomy centroids for the DBS dataset

Fig. 1: Location of targets in T1-weighted MR Images

Point	Annotation
Orbitofrontal cortices (left and right) a.k.a. LOFC and ROFC	Annotations by a single expert.
Dorsolateral prefrontal cortices (left and right) a.k.a. LDLPFC and RDLPFC	
Heschl's gyrus (left) a.k.a. LHESCHL	
Hand region of the primary motor cortex (left) a.k.a. LCM1	
Face regions of the motor cortices (left and right) a.k.a. LFACEMC and RFACEMC	Annotations by three experts.
Upper limb regions of the motor cortices (left and right) a.k.a. LULIMBMC and RULIMBMC	
Lower limb regions of the motor cortices (left and right) a.k.a. LLLIMBMC and RLLIMBMC	

Table 1: TMS target points annotated in TMS Database

3.1.2 Subcortical Anatomy DBS Dataset

The second database contains 216 T1-weighted MR images. They were deformably registered to the versions 2 and 3 of a Parkinson's disease specific atlas [20] in order to generate reference segmentations for the subcortical anatomy. The centroids of 30 subcortical structures (Tab. 2) could then be annotated for each image, constituting an approximate ground truth for our experiment. The location of these points for the left side of the brain of one patient of the dataset is shown in Fig. 1b.

3.2 Networks

We used four different artificial neural network architectures. Two of these are designed to use a top-down strategy and the other two a bottom-up strategy. They are all oriented towards the goal of localisation but render results in two

Point	Annotation
Subthalamic nucleus (left and right)	Registration using v2 and v3 of the ParkMedAtlis [20]
Caudate (left and right)	
Putamen (left and right)	
Amygdala (left and right)	
Anterior thalamus (left and right)	
Medial thalamus (left and right)	
Lateral thalamus (left and right)	
Pulvinar thalamus (left and right)	
Hippocampus (left and right)	
Medial geniculate (left and right)	
Lateral geniculate (left and right)	
Red nucleus (left and right)	
Substantia nigra (left and right)	
Globus pallidus externus (left and right)	
Globus pallidus internus (left and right)	

Table 2: Subcortical structures annotated in Database DBS.

different ways. Two of the architectures express their outputs as coordinates for each point. The other two use heatmaps to express their outputs as a spatial probability distribution. For each type of output, one architecture is designed to use a Top-Down strategy and the other a Bottom-Up strategy.

3.2.1 Co-ordinate-based Bottom-Up Architecture (Bc)

Our Co-ordinate-based bottom-up architecture, see Fig. 2, is a convolutional neural network architecture, composed of 6 convolutional layers followed by a 512 units linear layer. The first convolutional layer counts 32 kernels, increasing by two for each following convolutional layer. 2x2x2 Max-pooling is performed between convolutional layers. This architecture allows networks to focus on

individual features on the whole image, thus making it similar to bottom-up attentional selection. We used 2 as a batch size for training, validation and test. ReLU activation function was used between layers. The gradient descent optimizer chosen for this architecture was ADAM. During training, our top-down architecture used L2 loss function, while our top-down architecture used its own custom loss function since the cropping operation it performs is not differentiable (Baxter *et al.* , 2021). The mean error for a given patient and point on all repetitions,

$$Err_{i,p} = \frac{1}{n_{i,p}} \sum_{j=1}^{n_{i,p}} \|gt_{i,p,j} - pr_{i,p,j}\|_2 \tag{1}$$

, with $gt_{i,p,j}$ and $pr_{i,p,j}$ ground truth and prediction respectively for patient i, point p and repetition j and $n_{i,p}$ the number of repetitions for patient i and point p, was used for comparison of the two architectures.

3.2.2 Co-ordinate-based Top-Down Architecture (Tc)

The multi-resolution architecture implemented by Baxter *et al.* [9] allows networks to resample and crop images, always analysing images containing 8x8x8 voxels of increasingly fine resolution, until the native image resolution (1 mm) is achieved yielding the final position estimate. In this way, it resembles top-down attentional selection. A batch size of 8 was used for training, validation and test. The gradient descent optimizer chosen for this architecture was SGD.

3.2.3 Heatmap-based Bottom-Up Architecture (Bh)

To create a heatmap-based bottom-up architecture, we took inspiration from fully-convolutional neural networks for image segmentation [21], see Fig. 3. In order to maintain the output heatmap's resolution to be the same as the input

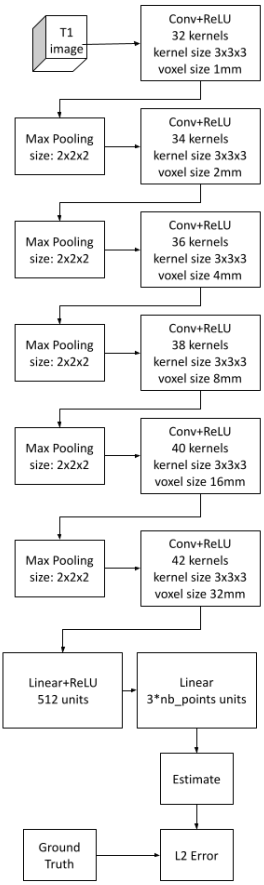


Fig. 2: Co-ordinate-based Bottom-Up network consisting of alternating between convolution and max-pooling layers until flattening and final processing with a series of linear layers.

images, convolution with spacing was used for the network to collect larger-scale, more-abstract features. Similar to the previous network, the L2 loss was used as the cost function and a batch size of 2 was used for training, validation, and testing.

3.2.4 Heatmap-based Top-Down Architecture (Th)

In our last architecture, see Fig. 4, the network trained a series of feature maps starting at a very coarse resolution. At each resolution level, the previous

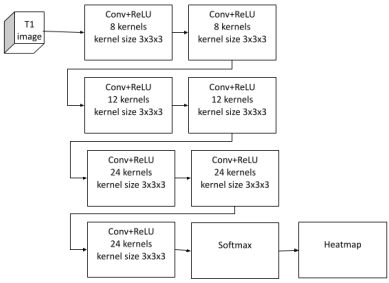


Fig. 3: Fully convolutional Heatmap-based Bottom-Up network.

coarser feature maps were upsampled and then combined with information extracted from the image at the same resolution level. This is analogous to how U-Nets [22] operate except that, in order to maintain a purely Top-Down approach, no convolutions were performed before any of the down-sampling, which could be interpreted as a Bottom-Up search for features. Again, the L2 loss was used as the cost function and a batch size of 2 was used for training, validation, and testing.

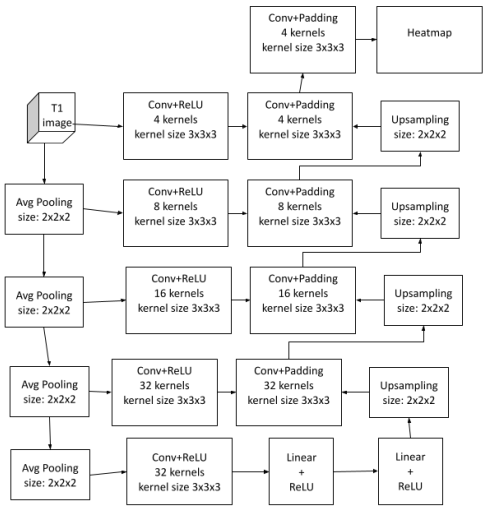


Fig. 4: Heatmap-based Top-Down network inspired by U-nets without convolution in the down-sampling.

	nb of folds	***Val. size***	***Training size***	***Test size***
TMS db $n = 26$	26	4	16	6
			11	11
			6	16
DBS db $n = 216$	24	9	162	45
			126	81
			90	117
			45	162
			7	200

Table 3: Number of samples q, validation set size v, training set size, test set size k and total number of patients n for the TMS and the DBS datasets.

3.3 Experiments

The experiment described in this section was performed on the TMS dataset for all architectures. It was performed on the DBS dataset for the two coordinate-based architectures for they were the best performing in the first experiment.

3.3.1 Cross-validation

We used repeated leave-k-out cross-validation, consisting in the creation of q different sets of training, validation and test samples. For a total number of n patients in the whole dataset, a size of v patients was chosen for validation samples, whereas test and training samples respectively contained k and n - v - k patients. To create each test sample, k patients were randomly selected from the groups of patients who had been included the least in previously created test samples. The n - k patients who were not selected in a particular test sample were randomly divided into corresponding validation and training samples. As a result, each patient was included $\frac{kq}{n} \pm 1$ times in a test sample.

In order to measure the way the number of patients available for training affects each of the two architectures, the aforementioned cross-validation was performed several times for different values of k. Tab. 3 shows the values of n, v and k for each dataset. The size of the DBS dataset allowed us to try 5 values for k instead of 3.

3.3.2 Data augmentation

Data augmentation was performed in validation and training samples in the form of left-right flipping and random rotations within the axial plane and translations.

4 Results

4.1 Memory usage

A convolutional neural network's memory usage depends on both parameters and feature maps. All parameters have to be saved in RAM in order to perform gradient descent. Our Bottom-up architecture is using approximately 1.1×10^7 parameters, which represent 45 MB, while our more complex Top-Down architecture is one order of magnitude larger with approximately 1.1×10^8 parameters, which represent 427 MB. As for feature maps, the total of their sizes doesn't matter as much as bottlenecks, as they don't need to be saved all the way through the network. For any given layer, networks need to store both its input data and output data simultaneously. Any other data a network needs to store through the whole learning process has to be added and a bottleneck is found when the need for storage reaches a maximum. This

	Tc	**Bc**	**Th**	**Bh**
Number of parameters	$1.1 * 10^8$	$1.1 * 10^7$	$1.1 * 10^9$	$4.7 * 10^4$
Feature map storage bottleneck	$1.2s$	$36s$	12s	48s
Total RAM usage bottleneck	504 MB	2.5 GB	5.1 GB	3.2 GB

Table 4: RAM usage for co-ordinate-based Top-Down **Tc**, co-ordinate-based Bottom-Up **Bc**, heatmap-based Top-Down **Th** and heatmap-based Bottom-Up **Bh** architectures. The number of parameters is for a whole network. As for feature maps storage, only the bottleneck (max. simultaneous memory usage) is expressed as a function of s, which is the memory usage for one input image. Total RAM usage bottleneck is computed using an input image size of 256x256x256 voxels and 4-Byte float numbers for both voxel values and parameters.

bottleneck is thus the minimum amount of memory necessary to run these architectures on a single testing set image, and can even be achieved in practice by deallocating tensors once they are no longer useful. Despite its need to store every input image resolution during learning, our Co-ordinate-based Top-Down architecture is far more efficient regarding memory usage by feature maps. Indeed, instead of using a whole image, the cropping step allows networks to use constant size images 8^5 times smaller than the original image. This way, its bottleneck regarding feature maps is only 1.2 times the size of an original input image, against 36 times for our Co-ordinate-based Bottom-up architecture. For the heatmap-based architectures, the bottlenecks were similar, occurring relatively close to the end of the network just before the high resolution heatmaps were created and used 12 times and 48 times the size of the image, respectively.. Using 4-Byte float numbers for voxel values and parameters and an input image size of 256x256x256 voxels, total memory usage for bottlenecks, adding feature maps and total parameters, was computed and weighs approximately 504 MB for our Top-Down co-ordinate based architecture, 2.5 GB for our Bottom-Up co-ordinate based architecture, 5.1 GB for the Top-Down heatmap-based architecture, and 3.2 GB for the Bottom-Up heatmap-based architecture (Tab. 4). The memory differences are most pronounced for the co-ordinate based architectures, largely because the Top-Down approach can effectively avoid doing any processing on feature maps with the same size and resolution as the entire image. This is unavoidable for heatmap-based methods, regardless of if they are Top-Down or Bottom-Up, as the output heatmaps themselves are feature maps with the same size and resolution as the entire image. In addition, the fully-connected layer in the Top-Down heatmap approach uses the vast majority of its parameters and contributes much to the memory required.

4.2 Cortical TMS Dataset

We present mean and standard deviation of the patient-point error $Err_{i,p}$ for $k = 6$ and a training sample size of 16 in Tab. 5. For the points which have been annotated by three experts, expert variability is also specified. A multifactorial ANOVA was performed fitting the functions $f_i(\cdot)$ in the following model:

$$E(P, D, T, C) = \tilde{E} + f_1(P) + f_2(D) + f_3(T) + f_4(C) + f_5(T, C) + \epsilon \quad (2)$$

where P is the patient, D is the dataset size, T is either "Top-Down" or "Bottom-Up", C is either "Coordinate-based" or "Heatmap-based", E is the error for a particular configuration of these variables, $\tilde{E}$ is the average error, and ϵ is the residual. A series of null hypotheses are then tested against in order to distinguish the parameters of each of the $f_i(\cdot)$ functions from 0. The patient and the training dataset size were used as additional factors (f_1 and f_2 respectively) to account for patient variability and the amount of data, both of which easily overwhelms the differences between methods. The p-values in Table 6 show the significance of the effects of the approach, Top-Down or Bottom-Up, the output type, co-ordinates or heatmap based, and the interaction between the two. All results are significant except for the effect of the Top-Down or Bottom-Up approach for Heschl's gyrus and the interaction between the effect of the approach and the effect of the output type for the right upper and lower limb areas. Qualitative results for all 4 methods are shown in Figure 5. This indicates that amongst the methods we investigated, Top-Down generally outperforms Bottom-Up although the degree depends on the specifics of the architectures used, hence the interaction effect.

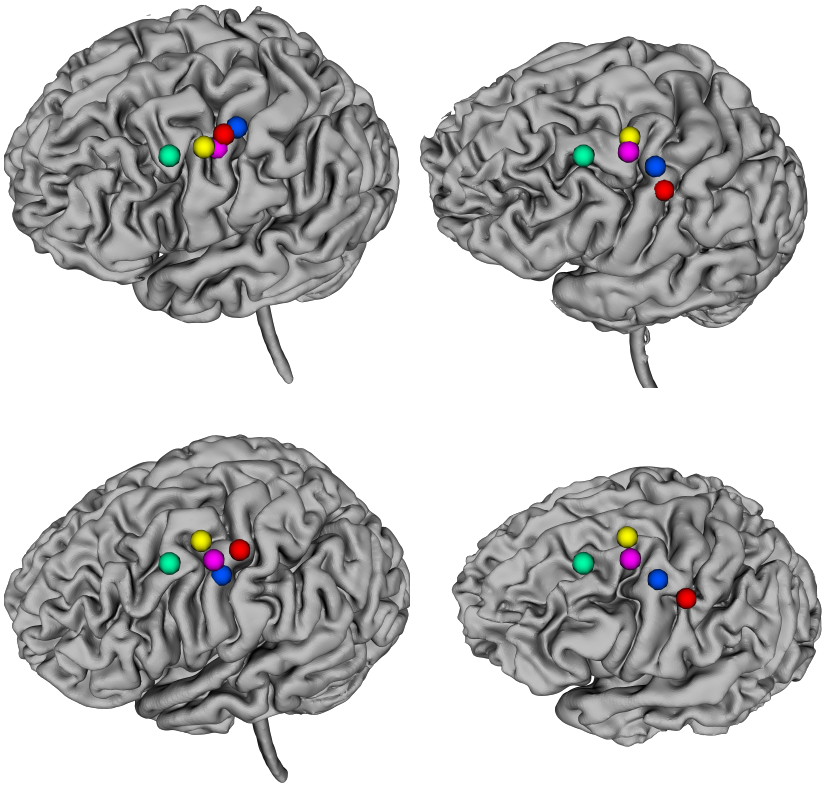


Fig. 5: Qualitative results showing the ground truth points for the left primary motor cortex in red, the Coordinate-based Top-Down results in blue, the Heatmap-based Top-Down results in magenta, the Coordinate-based Bottom-Up results in yellow, and the Heatmap-based Bottom-Up results in cyan.

4.3 DBS Dataset

The mean and standard deviation of the patient-point Error $Err_{i,p}$ for k = 45 and a 162 training sample size are listed in Tab. 7. A Wilcoxon signed rank test was performed following by Bonferroni correction for the 30 points and 5 subexperiments. All p-values are significant for all subexperiments, that is for all points at each training data set size. Since the mean error is consistently lower for the co-ordinate-based Top-Down architecture, we can safely conclude

Point	E.V. (mm)	Tc (mm)	Bc (mm)	Th (mm)	Bh (mm)
LOFC		5.99 ± 4.95	11.37 ± 5.34	13.24 ± 7.05	13.33 ± 5.83
ROFC		6.54 ± 4.95	11.36 ± 4.91	12.37 ± 5.57	12.45 ± 4.93
LDLPFC		9.20 ± 9.05	11.03 ± 5.34	13.15 ± 6.27	14.13 ± 6.81
RDLPFC		7.24 ± 4.38	11.07 ± 5.54	12.60 ± 5.75	12.53 ± 6.66
LHESCHL		6.11 ± 3.75	10.39 ± 4.10	12.94 ± 9.41	10.93 ± 5.10
LFACEMC	7.12 ± 4.54	6.45 ± 4.06	12.00 ± 5.30	13.06 ± 6.09	12.80 ± 5.14
RFACEMC	8.84 ± 5.45	8.30 ± 5.53	13.96 ± 6.15	13.80 ± 6.42	14.10 ± 6.14
LLLIMBMC	5.65 ± 3.95	8.77 ± 5.63	14.72 ± 6.40	14.80 ± 7.63	16.44 ± 7.62
RLLIMBMC	6.73 ± 6.30	10.02 ± 7.71	15.22 ± 7.71	15.31 ± 8.40	17.03 ± 8.85
LULIMBMC	6.85 ± 4.70	10.10 ± 5.47	13.84 ± 6.84	13.16 ± 5.92	15.55 ± 7.29
RULIMBMC	6.35 ± 4.79	9.63 ± 5.54	14.90 ± 6.26	15.25 ± 6.91	16.59 ± 7.94
LCM1		7.78 ± 3.62	13.15 ± 6.15	14.09 ± 7.05	17.49 ± 9.95

Table 5: TMS database: Mean and standard deviation for patient-point Errors $Err_{i,p}$, by point for Co-ordinates-based Top-Down Tc, Co-ordinates-based Bottom-Up Bc, Heatmap-based Top-Down Th and Heatmap-based Bottom-Up Bh networks, in mm. Experiment test sample size $k = 6$ and training sample size of $22 - k = 16$. Expert Variability (E.V.) as computed by Baxter *et al.* [9] for points for which three experts annotations are available.

	Factors		Interaction
Point	TD vs BU	Co vs Hm	(TD vs BU):(Co vs Hm)
LOFC	$\mathbf{p < 1.67 \times 10^{-35}}$	$\mathbf{p < 4.43 \times 10^{-154}}$	$\mathbf{p < 1.06 \times 10^{-13}}$
ROFC	$\mathbf{p < 1.71 \times 10^{-23}}$	$\mathbf{p < 1.11 \times 10^{-119}}$	$\mathbf{p < 1.90 \times 10^{-08}}$
LDLPFC	$\mathbf{p < 1.30 \times 10^{-05}}$	$\mathbf{p < 2.17 \times 10^{-75}}$	$\mathbf{p < 4.11 \times 10^{-05}}$
RDLPFC	$\mathbf{p < 4.50 \times 10^{-26}}$	$\mathbf{p < 2.54 \times 10^{-80}}$	$\mathbf{p < 5.35 \times 10^{-03}}$
LHESCHL	$p < 1$	$\mathbf{p < 2.46 \times 10^{-103}}$	$\mathbf{p < 5.14 \times 10^{-17}}$
LFACEMC	$\mathbf{p < 2.65 \times 10^{-39}}$	$\mathbf{p < 9.65 \times 10^{-99}}$	$p < 3.24 \times 10^{-01}$
RFACEMC	$\mathbf{p < 4.38 \times 10^{-50}}$	$\mathbf{p < 8.24 \times 10^{-93}}$	$\mathbf{p < 5.08 \times 10^{-05}}$
LLLIMBMC	$\mathbf{p < 9.81 \times 10^{-43}}$	$\mathbf{p < 7.43 \times 10^{-133}}$	$\mathbf{p < 2.91 \times 10^{-10}}$
RLLIMBMC	$\mathbf{p < 2.62 \times 10^{-65}}$	$\mathbf{p < 1.20 \times 10^{-92}}$	$p < 1$
LULIMBMC	$\mathbf{p < 1.53 \times 10^{-49}}$	$\mathbf{p < 4.06 \times 10^{-68}}$	$p < 2.04 \times 10^{-01}$
RULIMBMC	$\mathbf{p < 8.13 \times 10^{-61}}$	$\mathbf{p < 2.15 \times 10^{-85}}$	$p < 1$
LCM1	$\mathbf{p < 2.03 \times 10^{-27}}$	$\mathbf{p < 3.92 \times 10^{-89}}$	$\mathbf{p < 2.64 \times 10^{-18}}$

Table 6: p-values after Bonferroni correction are shown for multifactorial ANOVA. T.D. vs B.U. is Top-Down vs Bottom-Up. Co vs Hm is Co-ordinates-based vs Heatmap-based. Statistically significant results are shown in **bold**.

the co-ordinate-based Top-Down architecture performed better than the co-ordinate-based Bottom-Up architecture.

4.4 Training sample size effect on accuracy

4.4.1 TMS Dataset

As expected, the number of patients in the training sample seems to have an effect on accuracy for every architecture (Fig. 6). As this number increases, accuracy for our co-ordinate based Top-Down architecture seems to be getting closer to expert variability scores. We expect that as more data becomes available, our co-ordinate based Top-Down architecture network will perform as

	Tc (mm)	**Bc (mm)**	**Tc vs Bc**
Amygdala (L)	3.95 ± 3.03	8.83 ± 6.38	$\mathbf{p < 2.3 \times 10^{-33}}$
Amygdala (R)	3.62 ± 2.76	8.98 ± 6.58	$\mathbf{p < 1.5 \times 10^{-33}}$
Anterior thalamus (L)	4.13 ± 2.70	9.36 ± 6.95	$\mathbf{p < 1.7 \times 10^{-32}}$
Anterior thalamus (R)	4.19 ± 3.01	9.34 ± 6.81	$\mathbf{p < 3.8 \times 10^{-31}}$
Caudate (L)	5.09 ± 3.04	9.81 ± 7.07	$\mathbf{p < 4.5 \times 10^{-26}}$
Caudate (R)	4.58 ± 2.85	9.73 ± 6.90	$\mathbf{p < 6.0 \times 10^{-30}}$
Putamen (L)	4.48 ± 3.17	9.53 ± 7.07	$\mathbf{p < 9.9 \times 10^{-32}}$
Putamen (R)	4.23 ± 3.36	9.70 ± 7.70	$\mathbf{p < 3.8 \times 10^{-33}}$
GPE (L)	4.35 ± 2.75	9.34 ± 6.92	$\mathbf{p < 7.2 \times 10^{-30}}$
GPE (R)	4.24 ± 2.96	9.45 ± 8.32	$\mathbf{p < 7.5 \times 10^{-32}}$
GPI (L)	4.27 ± 2.65	9.17 ± 6.69	$\mathbf{p < 3.2 \times 10^{-30}}$
GPI (R)	4.22 ± 2.77	9.19 ± 7.23	$\mathbf{p < 5.0 \times 10^{-29}}$
Hippocampus (L)	4.64 ± 3.28	8.87 ± 7.96	$\mathbf{p < 3.0 \times 10^{-30}}$
Hippocampus (R)	4.12 ± 3.18	8.96 ± 6.80	$\mathbf{p < 5.4 \times 10^{-32}}$
Lateral thalamus (L)	4.48 ± 2.66	9.25 ± 7.44	$\mathbf{p < 2.4 \times 10^{-29}}$
Lateral thalamus (R)	4.45 ± 3.40	9.13 ± 6.68	$\mathbf{p < 5.7 \times 10^{-29}}$
Medial thalamus (L)	3.81 ± 2.70	9.13 ± 6.89	$\mathbf{p < 8.9 \times 10^{-32}}$
Medial thalamus (R)	3.56 ± 2.45	9.22 ± 6.72	$\mathbf{p < 9.9 \times 10^{-33}}$
Pulvinar thalamus (L)	4.72 ± 3.06	9.27 ± 7.42	$\mathbf{p < 3.0 \times 10^{-28}}$
Pulvinar thalamus (R)	4.28 ± 2.96	9.07 ± 7.00	$\mathbf{p < 7.2 \times 10^{-33}}$
Medial geniculate (L)	4.33 ± 2.89	8.79 ± 6.46	$\mathbf{p < 8.9 \times 10^{-31}}$
Medial geniculate (R)	5.03 ± 3.39	9.04 ± 6.85	$\mathbf{p < 8.0 \times 10^{-23}}$
Lateral geniculate (L)	4.16 ± 2.95	8.97 ± 6.79	$\mathbf{p < 1.7 \times 10^{-30}}$
Lateral geniculate (R)	4.41 ± 2.92	9.16 ± 6.86	$\mathbf{p < 1.8 \times 10^{-29}}$
Red nucleus (L)	4.04 ± 2.69	8.35 ± 6.65	$\mathbf{p < 9.2 \times 10^{-28}}$
Red nucleus (R)	4.31 ± 2.64	8.71 ± 7.10	$\mathbf{p < 4.3 \times 10^{-27}}$
STN (L)	4.29 ± 2.44	8.37 ± 6.32	$\mathbf{p < 9.5 \times 10^{-24}}$
STN (R)	3.91 ± 2.69	8.49 ± 6.71	$\mathbf{p < 5.3 \times 10^{-32}}$
Substantia nigra (L)	4.16 ± 2.75	8.32 ± 6.41	$\mathbf{p < 2.6 \times 10^{-28}}$
Substantia nigra (R)	3.79 ± 2.69	8.38 ± 6.46	$\mathbf{p < 4.8 \times 10^{-33}}$

Table 7: DBS database: Wilcoxon signed ranks test results on patient errors $Err_{i,p}$, by point for Co-ordinates-based Top-Down and Co-ordinates-based Bottom-Up networks. All p-values are shown after Bonferroni correction for multiple comparisons and are significant. The architecture which gets the lowest error is always the Top-Down one. Experiment test sample size $k = 45$ and training sample size of 162.

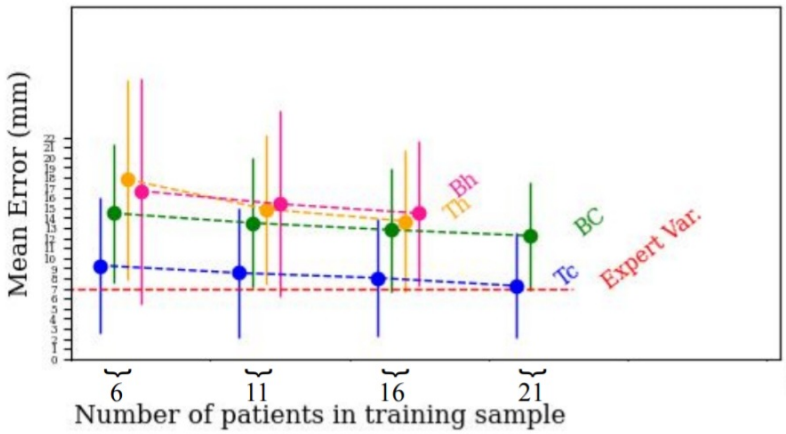


Fig. 6: TMS experiment - Evolution of all points mean error with respect to number of patients in training sample for all architectures. Mean Expert Variability on points for which three expert annotations are available is a constant and can be used for comparison.

well as experts on this task. The remaining architectures also improve with an increasing number of training datasets, indicating that significantly improved performance can be achieved through the acquisition of more data.

4.4.2 DBS Dataset

The first jump in training sample size seems to have an effect on accuracy for both architectures. However, past 45 patients in the training sample, the training sample size doesn't seem to make any significant difference. There seems to be a limit in how much the size of the training sample can improve accuracy for this problem.

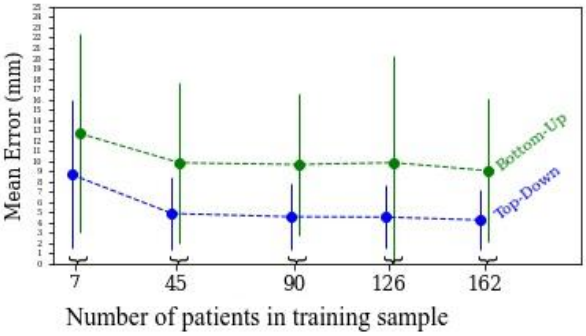


Fig. 7: DBS experiment - Evolution of all points mean error with respect to number of patients in training sample for both co-ordinate-based architectures.

5 Discussion

The concepts of "Top-Down" and "Bottom-Up" are only heuristics from the domain of cognitive psychology and don't have a simple one-to-one relationship with different computer vision architectures, let alone explain the differences between the plethora of architectures widely used in computer vision research today. For image segmentation, more modern architectures such as those derived from U-Nets [22] or vision transformers [23, 24] have distinctly Top-Down and Bottom-Up components working in tandem, demonstrating that the computer vision community is moving away from the purely Bottom-Up approaches it started with. Point localisation, being a less commonly researched problem, has yet to experience this.

Given this, this study should be interpreted as a preliminary exploration in this field. It is by no means an exhaustive exploration of all possible point localisation architectures, especially given their prevalence in 2D computer vision and the difficulty of translating them into a volumetric medical context. It is

possible in the future for a mostly Bottom-Up architecture to completely outperform all others, using some form of novel architecture or network component that has not yet been explored or developed.

The main limitation to our study is the selection and optimisation of our architectures. The plethora of CNN architectures proposed over the past decade for point localisation problems offers a large amount of choice within the Bottom-Up paradigm. The more traditional alternating convolution-plus-max-pooling approach was used, largely because it tends to involve fewer parameters and a lower memory usage than more recent 3D networks, which is a large design limitation. We also did not use any hyperparameter optimisation framework as learning a large number of networks (to investigate training dataset size dependence) is already quite time-consuming. Therefore, we cannot be certain that either architecture is fully optimized, although the prevalence of Bottom-Up architectures and the wide gap between models that we observed does suggest that Top-Down architectures are under-utilised.

Because of the difference in memory consumption, we had to use a smaller batch size for the Heatmap-based architecture and the co-ordinates-based Bottom-Up architecture (batch size of 2 volumes) than for the co-ordinates-based Top-Down architecture (batch size of 8 volumes), which may have affected training. If anything, it does further indicate the need we have for more memory-efficient architectures, which allow for bigger batch sizes as well as more complexity. Memory consumption is a pressing issue for point localisation in volumetric images as the task is not easily partitioned into patches which can be processed independently, which can facilitate the use of deep-learning for other types of volumetric image processing, notably segmentation. Architectures designed for natural images often cannot be used without significant re-parameterisation due to the memory constraints of current graphic cards.

It would be interesting to expand our work to other 3D point localisation problems to confirm it can be applied to various localisation problems beyond localisation of anatomical structures in brain MRI. Because of the focus on fully three-dimensional images, without simplifying assumptions that might not be clinically valid (such as being given the exact slice with the point of interest, as in Sugimori *et al.* [6]) the constraints on memory consumption remain a significant factor.

One last observation is that methods in the literature as well as in this study seem to take on two approaches based on how they represent points either via their coordinates [8, 9] or via heatmaps [6, 7, 11]. Our results suggest that the benefit of a Top-Down approach is more pronounced for coordinate-based frameworks moreso than heatmap based ones, especially in terms of reduced memory consumption. This is because for any heatmap based approach using volumetric images, a large amount of memory will have to be used to simply represent several heatmaps spanning the entire image volume. Heatmaps however could be used to leverage other architectures outside of point localisation, notably those for image segmentation which has received much more attention from the research community and provides a plethora of tools that blend both Top-Down and Bottom-Up types of image processing.

Future work

One immediate area of future work is to integrate these localisation frameworks into those designed for the segmentation of smaller anatomical structures. The motivation behind this would be to use more memory and time-efficient localisation networks to roughly estimate the location and extent of the anatomy, allowing for the image to be heavily cropped to only this region (plus a margin in case of error) [25]. This would allow for separate segmentation frameworks

to be orders of magnitude more efficient in terms of time and memory by processing images that are reduced in size by these orders of magnitude. Recent work has also suggested that such an approach can also be beneficial in terms of reducing the influence of high signal and contrast in other, unrelated regions of the image, which may "distract" the network from the specific segmentation task given limited data.

6 Conclusions

The distinction between Top-Down and Bottom-Up processing is a useful conceptual tool in cognitive psychology for explaining visual search which itself affects how one can structure machine learning architectures for performing the similar task of point localisation. This is particularly important for volumetric images, such as in neuro-interventional planning, which are subject to both technical limitations in terms of memory as well as accuracy requirements. This preliminary study takes motivation from this distinction and investigates in a quantitative manner the possibility for the less widely used Top-Down paradigm to improve performance with respect to these two considerations.

Our co-ordinate-based Top-Down network achieved a significantly better accuracy with a significantly lower memory bottleneck compared to its Bottom-Up counterpart on two neuro-interventional planning tasks involving both subcortical (DBS Dataset) and cortical (TMS Dataset) anatomy. We also have evidence that this improvement is robust to training dataset size at least in the small-to-middle range (i.e. 6 to 162 image volumes) which characterise many medical volumetric datasets. All of our networks' accuracy improve with the size of the training sample for small numbers of patients. However, it does seem that the improvement in accuracy than comes with increasing the number of patients is likely to reach a plateau and, based on our results, it is unclear

if given enough data the Bottom-Up method could begin to rival Top-Down performance. This is particularly important as other changes to the design of these networks, such as the representation of their output, also has a significant effect on their performance which can counter-balance the investigated design decision.

Although preliminary, we have found evidence for a critical re-appreciation of point localisation architectures specific to medical imaging that differs from computer vision and encourages Top-Down processing. These results may be potentially interesting to those developing new models for point localisation, encouraging a different paradigm from the one traditionally applied towards 2D images.

Declarations

- Funding for E. Giffard was received from the Institut des Neurosciences Cliniques de Rennes (INCR) and the Allocations de Recherche Doctorale (ARED) initiative from the Région Bretagne.
- We have no conflicts of interest to declare.
- All patient data was collected retrospectively with informed patient consent and approval from the institutional ethics review board.
- We consent to the publication of this article in IJCARS.
- Result tables for all experiments are available on request. However, we can not provide sensitive patient data.
- Our code will be available on request.
- Enora Giffard and John S.H. Baxter designed and carried out the experiments as well as wrote the manuscript. John Baxter and Pierre Jannin supervised the project.

References

[1] K. He and J. Sun, "Convolutional neural networks at constrained time cost," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 5353–5360.

[2] S. Qiao, Z. Lin, J. Zhang, and A. L. Yuille, "Neural rejuvenation: Improving deep network training by enhancing computational resource utilization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 61–71.

[3] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 580–587.

[4] R. Girshick, "Fast r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1440–1448.

[5] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *Advances in neural information processing systems*, vol. 28, 2015.

[6] H. Sugimori and M. Kawakami, "Automatic detection of a standard line for brain magnetic resonance imaging using deep learning," *Applied Sciences*, vol. 9, no. 18, p. 3849, 2019.

[7] X. Yang, W. T. Tang, G. Tjio, S. Y. Yeo, and Y. Su, "Automatic detection of anatomical landmarks in brain mr scanning using multi-task deep neural networks," *Neurocomputing*, vol. 396, pp. 514–521, 2020.

[8] B. Gohel, L. Kumar, and D. Shah, "Deep learning-based automated localisation of anterior commissure and posterior commissure landmarks in 3d space from three-plane 2d mri localiser slices of the brain," *Procedia Computer Science*, vol. 218, pp. 1027–1032, 2023.

[9] J. S. Baxter, Q. A. Bui, E. Maguet, S. Croci, A. Delmas, J.-P. Lefaucheur, L. Bredoux, and P. Jannin, "Automatic cortical target point localisation in mri for transcranial magnetic stimulation via a multi-resolution convolutional neural network," *International Journal of Computer Assisted Radiology and Surgery*, vol. 16, no. 7, pp. 1077–1087, 2021.

[10] T. Foulsham, C. Chapman, E. Nasiopoulos, and A. Kingstone, "Top-down and bottom-up aspects of active search in a real-world environment." *Canadian Journal of Experimental Psychology/Revue canadienne de psychologie expérimentale*, vol. 68, no. 1, p. 8, 2014.

[11] S. Li, Q. Gong, H. Li, S. Chen, Y. Liu, G. Ruan, L. Zhu, L. Liu, and H. Chen, "Automatic location scheme of anatomical landmarks in 3d head mri based on the scale attention hourglass network," *Computer Methods and Programs in Biomedicine*, vol. 214, p. 106564, 2022.

[12] H. Lester and S. R. Arridge, "A survey of hierarchical non-linear medical image registration," *Pattern recognition*, vol. 32, no. 1, pp. 129–149, 1999.

[13] J.-P. Lefaucheur, "Transcranial magnetic stimulation," *Handbook of clinical neurology*, vol. 160, pp. 559–580, 2019.

[14] M. Balconi, "Dorsolateral prefrontal cortex, working memory and episodic memory processes: insight through transcranial magnetic stimulation techniques," *Neuroscience bulletin*, vol. 29, pp. 381–389, 2013.

[15] P. Hamid, B. H. Malik, and M. L. Hussain, "Noninvasive transcranial magnetic stimulation (tms) in chronic refractory pain: a systematic review," *Cureus*, vol. 11, no. 10, 2019.

[16] R. Sparing, D. Buelte, I. G. Meister, T. Pauš, and G. R. Fink, "Transcranial magnetic stimulation and the challenge of coil placement: a comparison of conventional and stereotaxic neuronavigational strategies," *Human brain mapping*, vol. 29, no. 1, pp. 82–96, 2008.

[17] H. R. Siebner, G. Hartwigsen, T. Kassuba, and J. C. Rothwell, "How does transcranial magnetic stimulation modify neuronal activity in the brain? implications for studies of cognition," *cortex*, vol. 45, no. 9, pp. 1035–1042, 2009.

[18] E. Middlebrooks, R. Domingo, T. Vivas-Buitrago, L. Okromelidze, T. Tsuboi, J. Wong, R. Eisinger, L. Almeida, M. Burns, A. Horn *et al.*, "Neuroimaging advances in deep brain stimulation: review of indications, anatomy, and brain connectomics," *American Journal of Neuroradiology*, vol. 41, no. 9, pp. 1558–1568, 2020.

[19] J. S. Baxter and P. Jannin, "Validation in the age of machine learning: A framework for describing validation with examples in transcranial magnetic stimulation and deep brain stimulation," *Intelligence-Based Medicine*, p. 100090, 2023.

[20] C. Haegelen, P. Coupé, V. Fonov, N. Guizard, P. Jannin, X. Morandi, and D. L. Collins, "Automated segmentation of basal ganglia and deep brain structures in mri of parkinson's disease," *International journal of computer assisted radiology and surgery*, vol. 8, no. 1, pp. 99–110, 2013.

[21] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3431–3440.

[22] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*. Springer, 2015, pp. 234–241.

[23] R. Ranftl, A. Bochkovskiy, and V. Koltun, "Vision transformers for dense prediction," in *Proceedings of the IEEE/CVF International Conference*

on Computer Vision, 2021, pp. 12 179–12 188.

[24] A. Hatamizadeh, V. Nath, Y. Tang, D. Yang, H. R. Roth, and D. Xu, "Swin unetr: Swin transformers for semantic segmentation of brain tumors in mri images," in *Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries: 7th International Workshop, BrainLes 2021, Held in Conjunction with MICCAI 2021, Virtual Event, September 27, 2021, Revised Selected Papers, Part I.* Springer, 2022, pp. 272–284.

[25] J. S. Baxter and P. Jannin, "Combining simple interactivity and machine learning: a separable deep learning approach to subthalamic nucleus localization and segmentation in mri for deep brain stimulation surgical planning," *Journal of Medical Imaging*, vol. 9, no. 4, pp. 045 001–045 001, 2022.