Resource-Constrained Volumetric Data Estimation From MRI Images

1071068

University of Oxford

Abstract. Processing 3D MRI images is computationally intensive and requires substantial RAM resources. Models like 3D-UNet have millions of parameters, making it difficult to deploy them in resource-limited environments. We explore simpler models that process 2D-slices such as 2D-UNet, SegNet and Fully Convolutional Network to test the hypothesis that there are techniques that help to achieve sufficient performance and accuracy of calculating volumes of structures when using 2D data with smart data preprocessing. We show that increasing complexity of models by adding more parameters and increasing complexity of the dataset by adding more data does not automatically translate into better performance. We investigate the effects of data augmentation, approaches to 2D slicing, various model architectures and components. In our experiments, we evaluate performance using DICE score and volume estimation error to show that quality of data and suitable model architecture are more important than quantity of data and number of model parameters.

1 Introduction

Neural networks are useful in the biomedical domain in areas such as MRI imaging, drug discovery, and many others. Due to constraints on computational resources which are the norm in hospitals and research facilities, deploying models with fewer training parameters is preferable [1]. Simpler models require less computational and memory resources, and are faster to train, which is essential for quick iterations of diagnosis and treatment. Models of lower complexity generalize better to unseen data, an important aspect given the diversity of medical data and its limited availability. Hospitals require smaller networks that are more explainable, giving healthcare practitioners confidence to trust these models. Additionally, simpler models better adapt to new diagnostics protocols, and can be deployed on mobile or hospital devices [2].

Despite these requirements, processing 3D MRI images is computationally expensive and requires a lot of RAM memory due to the size of inputs. Models which tackle this task have large numbers of parameters, such as 3D-UNet (20 million)[3] or V-Net (70 million)[4], making them difficult to deploy and scale. This motivates the need for less computationally heavy models and better approaches to data preprocessing. Our research question explores how does the quality and diversity of data, along with the choice of model architecture, impact the performance of deep learning models in comparison to the quantity of data and the number of model parameters.

2 Related work

Numerous techniques have been developed to overcome the computational challenges, including altering preprocessing steps to compress the sizes of inputs, slicing images into 2D and 2.5D [5], data augmentation, and reducing model architectures [6]. Ronneberger et al. [7] presents applying elastic deformations and transformations to the training data which improves test performance on MRI image segmentation. Isensee et al. [8] uses combination of geometric transformations and intensity-based augmentations on the task of brain tumor segmentation. Patch-based slicing is investigated by Roth et al. [9] that uses this technique for segmenting 2D slices extracted from 3D images. Zhou et al. [10] introduces a 2D U-Net architecture with fewer parameters than the 3D version, while maintaining a high level of segmentation accuracy. Badrinarayanan et al. [11] presents even lighter SegNet architecture, which is based on an encoder-decoder architecture with skip connections, and has fewer parameters compared to 2D U-Net. Long et al. [12] presents Fully Convolutional Network, which removes the fully connected layers, significantly reducing the number of parameters.

In this paper, we explore the effects of data augmentation, 2D slicing techniques, adding more data and parameters, and varying model architectures on the performance of models for MRI image segmentation. We investigate the hypothesis that combining smart data preprocessing and augmentation techniques inspired by the works above with architectures of proposed 2D models such as 2D UNet, SegNet and Fully Convolutional Network can lead to good performance on the tasks of image segmentation and volume calculation using MRI scan images from the IXI dataset.

3 Data & Proposed Approach

Preprocessing pipeline: We split data into 3 independent datasets. Data from Guy's Hospital site and Hammersmith Hospital (HH) site are merged, randomly shuffled, we produce a split of ratio 0.85 / 0.15 to obtain the training and validation datasets. Data from Institute of Psychiatry (IOP) site are left as testing dataset (to simulate the real-life scenario that incoming data at test-time might come from a different hospital than the data which are used during training which uses a different system to produce MRI images). Data are centred, and one-hot encoding of the channel of the labels is applied. We extract 2D data by slicing the 3D images along the depth dimension to produce 40 stacked 2D images with dimensions 1 x 128 x 128. Images are sampled to obtain 2D-slices from each 3D image for the training and evaluation purposes. The sampled 2D slices are visually checked to ensure they contain appropriate structures for segmentation. Note this preprocessing pipeline is used on both types of labeled datasets - brain and subcortical structures. The final 'base' split used in most experiments (unless specified otherwise) contains 419 training samples, 74 validation samples and 71 testing samples (in total 564) for brain data and 430 training samples, 76 validation samples and 71 testing samples for subcortical data (in total 577). Experiments E1-E4 use brain labels, E5 uses subcortical structure labels.

Design of experiments: To answer our research question, we designed 5 independent experiments. Each experiment focuses on exploring 1 aspect of our research question in isolation that will be evaluated. Experiments are repeated 10 times (where relevant) and we report the mean statistics.

- **E1: Data quality/diversity.** We train UNet model on different augmented datasets (brain labels) and evaluate test DICE score and brain volume estimates. Dataset variations include horizontal and vertical flip, rotations, Gaussian noise, contrast changes, and mixed transformations. Performance is compared to a baseline with no transformations.
- **E2: Number of training parameters.** We vary the number of encoder and decoder layers in UNet and FCN models, evaluate training time, test DICE score, and measure the number of model parameters.
- **E3: Data quantity.** We explore different sample sizes of slices (1-20) from each 3D image and the effect on the DICE score and training time.
- **E4: Model complexity.** We compare the performance of different models (with varying complexities) on brain image segmentation tasks by examining DICE score and brain volume estimate errors.
- **E5: Task complexity.** We explore the relationship between task complexity and performance by comparing the UNet model's performance on brain segmentation and subcortical structures segmentation (more challenging). We report test DICE score and volume calculations for brain and subcortical structure datasets.

Model specifications: In our experiments, we use the following architectures:

SegNet [11] The encoder consists of Conv2d layers, batch normalization and ReLU activations, followed by decoder consisting of max-pooling layers, Conv2d layers, batch normalisation and ReLU activations. The final output is passed through a Conv2d layer with sigmoid activation. In total we use 10 convolutional layers (kernel size=3 and padding=1). The number of output channels of each convolutional layer increases from 64 to 512 in the encoder, and decreases back to 64 in the decoder. The max-pooling and max-unpooling layers have kernel size 2 and stride 2. We use Adam optimiser, $lr = 1 \times 10^{-3}$, ReduceLROnPlateau scheduler and Focal loss function.

UNet [7] Encoder and a decoder network are connected through skip connections. The encoder consists of 4 downsampling convolutional layers, where each downsampling layer contains two 3x3 convolutional layers followed by ReLU and batch normalization. The decoder consists of 4 upsampling layers that upsample the feature maps using bilinear interpolation with a scale factor of 2, followed by two convolutional layers. The final layer of the decoder is a Conv2d layer (kernel size=1) and a sigmoid activation function. We use BCE loss (combining binary cross-entropy and the DICE coefficient), Adam optimizer with $lr = 1 \times 10^{-3}$ and ReduceLROnPlateau scheduler.

FCN [12] Encoder contains a series of repetitions (we use 6) of a single convolutional layer (kernel size=3, padding=1), followed by ReLU and a max pooling layer (with downsampling factor 2). The kernel size of the convolutional layer is 3 and padding 1. The middle bottleneck layer consists of 2 convolutional

layers with a ReLU activation function, a dropout layer, and a 1x1 convolutional layer. Decoder contains a series of transposed convolutional layers (with upsampling factor 2), ReLU and a dropout layer. The final layer is a convolutional layer with a 1x1 kernel size and sigmoid activation. We train the model using BCE loss function, RMSprop optimizer with $lr = 1 \times 10^{-4}$, ReduceLROnPlateau scheduler and dropout probability 0.5.

Hyperparameter tuning: For each model we perform systematic grid search to tune hyperparameters and model components, including learning rate, number of filters, number of layers, dropout rate, batch size, loss function (we explore DICE loss, Binary DICE loss, Focal loss, Tversky loss, Soft DICE loss), optimizer, scheduler and activation functions. The model which performs best on the validation dataset is selected for experiments. We use early stopping after validation DICE score does not improve for 5 consecutive epochs.

Performance evaluation: To evaluate performance of a model on the image segmentation task we use DICE score to measure the overlap between a predicted segmentation mask and the true segmentation mask [13]. Value of this metric ranges between 0 (no overlap) to 1 (perfect overlap). It can be computed as $DICE = \frac{2*|A \cap B|}{|A|+|B|}$. The score is calculated by applying DICE to each pixel in the segmentation masks and then averaging the result across all pixels. To evaluate how well the model extracts volumetric data from MRI images we use our designed metric volume calculation which calculates volume of each structure (i.e. brain) by stacking the 2D predictions to reconstruct 3D image and summing up all predicted regions alongside the "depth" dimension.

Volume calculation: We first train the models to perform image segmentation on randomly sampled 2D slices from each 3D image. At test time, we evaluate the accuracy of volume calculation by taking a 3D image from the testing dataset (40x128x128) and make 40 predictions (1 prediction per slice 1x128x128). We then calculate the area of the brain/subcortical structure region in each slice by counting the number of pixels corresponding to that region. We estimate the total volume by summing up the areas of all 40 slices of the 3D image.

4 Experiments

E1 Results: Table 1 shows that applying a mix of augmentation techniques results in the highest test DICE score of 0.959, the smallest average brain volume estimation error of 1.54% and the smallest estimation error standard deviation of 0.41%. From all transformations applied in isolation, changing contrast (brightness factor 1.4) and Gaussian noise (mu=0, std=0.1) applied with probability of 0.4 to the inputs yield the greatest improvement. Applying rotations (max rotation 30°, p=0.3), horizontal and vertical flips (p=0.5) also result in better model performance. Applying random equalize in isolation does not improve test DICE performance compared to the baseline (for tested values of p in range 0.1-0.8,

we report the best 0.4), with test DICE score of 0.931 and an average estimate error 2.20%. Overall we see that increased data diversity improves performance.

Dataset augmentation	Test DICE	Avg. estimate error	Estimate error std.
No transform	0.939	2.13 %	0.98%
Contrast	**0.957**	**1.56** %	**0.52** %
H/V flip	**0.942**	**1.67**%	**0.57** %
Gaussian noise	**0.955**	**1.56** %	**0.48**%
Rotation	**0.941**	**1.81** %	**0.89**%
Random equalize	0.931	2.20 %	1.32%
Mix	**0.959**	**1.54**%	**0.41**%

Table 1: Dataset augmentation performance

E2 Results: Tables 2 and 3 display change in the number of model parameters, test DICE score and time taken to train on a GPU (until triggering early stopping) as we add more encoder and decoder blocks in selected models. As model complexity increases, the number of parameters grows exponentially, leading to significant increase in training time. More complex UNet model shows an improvement in test DICE score from 0.872 to 0.939. We observe that further increases in complexity (4+ ED blocks) do not result in any performance gains, while training time increases significantly. In contrast, as the complexity of simpler FCN model increases, there is a consistent improvement in test DICE score from 0.685 to 0.927 which does not plateau - suggesting that a model of lower complexity benefits from adding more parameters and layers. In contrast, adding any more than 4 blocks in the case of UNet for this task is only wasting computational resources. We noted limiting factor of 7 ED blocks for this experiment, the maximum that could fit system RAM 51GB in Colab without crashing.

ED blocks	Params	Test DICE	Time
2	403k	0.872	91s
3	4.3M	0.921	97s
4	8.8M	0.939	100s
5	17.3M	0.939	117s
6	69.2M	0.939	177s
7	276.8M	0.939	371s

Table 2: UNet model study

ED blocks	Params	Test DICE	Time
2	99.3k	0.685	7s
3	501.2k	0.796	27s
4	2.1M	0.853	38s
5	8.5M	0.890	39s
6	34.2M	0.905	42s
7	137M	0.927	105s

Table 3: FCN model study

E3 Results: Table 4 shows the effect of increasing the number of 2D slices sampled from each 3D MRI scan on the test performance and training time. Increasing the amount of data points from 419 to 8380 by sampling more 2D slices from each 3D image only results in a marginal 1.8% improvement in performance on the test data. Meanwhile, the training time (until early stopping trigger) increased by over 1600% (from 2m38s to 45m16s). This suggests that increasing quantity of data without consideration of correlations in data might contribute towards using a lot of computational resources without any performance benefits.

N. slices	N. data points	Test DICE	Training time
1	419	0.939	2m38s
2	838	0.949	5m40s
5	2095	0.952	12m50s
8	3352	0.954	20m19s
10	4190	0.956	23m12s
20	8380	0.957	45m16s

Table 4: Effect of increasing training data

E4 Results: Table 5 compares 3 model architectures (we fixed 5 ED blocks) on the brain segmentation task in terms of their performance. UNet, being the most complex model out of tested models, with the largest number of parameters, achieved the highest test DICE score of 0.939, closely followed by SegNet 0.934 and lastly FCN 0.890. UNet and FCN, on average, overestimate the volume of brain by 2.13% and 3.76% respectively, while SegNet underestimates the volume by 3.09%. UNet has the lowest average error standard deviation indicating most stable performance across test samples, suggesting that increasing complexity of architecture of encoder-decoder blocks benefits performance.

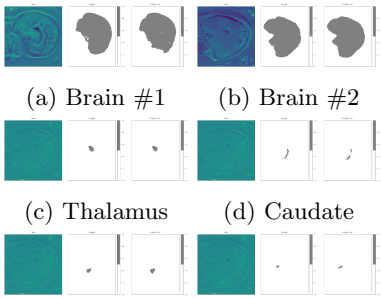
Model	Train DICE	Val DICE	Test DICE	Avg. vol. error	Avg. std	Estimate
UNet	0.995	0.962	0.939	2.13%	0.98%	over
SegNet	0.987	0.958	0.934	3.09%	1.92%	under
FCN	0.947	0.945	0.890	3.76%	3.95%	over

Table 5: Model architecture comparison

E5 Results: Table 6 shows that % error of volume estimation increases on a more challenging segmentation task, replacing brain extraction with subcortical labels. In comparison to the average volume estimation error of 2.13% (measured in E4), we observe larger errors in volume estimation for classes 1-4, representing the Thalamus, Caudate, Putamen, and Amygdala regions. There is large variance, with some regions having overestimated and underestimated volumes. This can be attributed to the very small size of regions and the increased difficulty of segmentation. We increased the number of encoder-decoder blocks to 7 (maximum that could fit into RAM) to achieve average test DICE score of 0.924 - which is despite using more powerful model lower than base UNet performance of 0.939 on the brain segmentation data. Figure 1 shows that the predicted masks for subcortical regions are not as well-defined as for brain regions which explains why the volume estimates are not accurate for subcortical regions.

Region	Avg. estimate error	Under/Over
background	0.08%	over
Thalamus	-17.18%	under
Caudate	13.32%	over
Putamen	-4.59%	under
Amygdala	9.68%	over

Table 6: Subcortical regions



(a) Brain #1 (b) Brain #2

(c) Thalamus (d) Caudate

(e) Putamen (f) Amygdala

Fig. 1: UNet test data segmentation

5 Discussion

In this section, we evaluate results and link our findings to the research question.

Data quality. E1 results support our hypothesis that using augmentation such as adding Gaussian noise, changing contrast, and flipping images can increase the diversity of the data and improve performance without the use of more resources. Augmentation enables the model to learn spatial and rotational invariances, become robust towards noise, reduce over-fitting and improve accuracy in volume estimation. The downsides include the need to fine-tune parameters such as probability of each transformation, rotational angle, noise distribution, and intensity adjustment values, which are specific for each dataset and model. This 'tuning' can be time-exhaustive and may require domain knowledge.

Number of training parameters. Increasing model complexity by adding layers results in improved performance, but also increases training time and possibly reaches plateau in performance gains, as demonstrated by the UNet model in E2. Starting with simpler model and progressively increasing complexity as needed, rather than deploying model with a large number of parameters from the start, is the approach which allows for identifying the most optimal model complexity for given task while minimizing the use of computational resources. Although the results support our hypothesis, an aspect which we did not consider is that the relationship between model complexity and performance is not always monotonic. The "double descent" phenomenon [14] suggests that there are points in the model complexity spectrum where performance initially improves, worsens, and then improves again.

Data quantity. E3 supports our hypothesis by demonstrating the importance of data quality over quantity. Sampling more 2D slices from the same 3D image does not provide much additional information due to the high correlation between slices. Although the test DICE score improves as the number of sampled slices increases, this performance improvement plateaus, with only a 1.8% increase when moving from 419 to 8380 data points, while training time increases significantly, by over 1600%. This is a marginal performance improvement which might not be worth the 1600% increase in training time.

Comparison of models. We find that UNet is the most suitable architecture for segmenting brain structures compared to SegNet and FCN, demonstrating that architecture selection is a crucial factor affecting performance. UNet has greater ability for retaining spatial information and capturing contextual information at multiple scales due to hierarchical feature extraction [7]. This enables the model to learn local and global patterns in the data, which is required for the task of MRI image segmentation.

Complexity of task. Our approach of using UNet and sampled 2D slices for estimating volumes of regions performs reasonably well for brain segmentation task, however, there is considerably large error and variance in the volume estimation errors for the smaller subcortical regions. It shows the need for a more powerful model and data preprocessing which can capture these intricate patterns, handle multiple class labels and high level of variability. For more complex segmentation tasks, 2D slices might not fully capture the patterns in 3D structures and we might require architectures with greater expressive capabilities.

6 Conclusion

Our study has shown that data preprocessing techniques such as data augmentation, which increase the diversity of the training dataset without impacting computational requirements, improve the test performance and generalization capabilities of a model. In contrast, increasing the amount of data without careful consideration negatively impacts the training time and computational requirements, while it does not result in significantly improved performance, as shown in our experiments. We found that architecture selection can improve performance without introducing extra computational needs and that only increasing the number of parameters is not guaranteed to improve the model's performance.

Our objective was to use a minimal approach and show that simple solutions such as using 2D models and 2D slices can be reasonably accurate, although as we demonstrated, it depends on the complexity of the required task. The performance of simple models deteriorates if the task becomes more complex.

This study has limitations as we do not consider 3D model architectures which might achieve higher performance within reasonable computational budget or alternative slicing techniques. Our study only covers MRI images from 3 hospitals and focuses on brain and subcortical structures, but this might not apply to other anatomical structures or different MRI scanners. Additionally, the data used for our investigation did not cover a wide enough range of population to be a representative sample.

Further work could explore the transferability of our findings to other medical images, such as CT scans and ultrasound images, evaluate additional model architectures, analyze the relationship between model complexity and performance in more detail to better understand the optimal model complexity required for different tasks. Investigating the performance of 3D models and comparing their volume estimates to simpler 2D models could provide further insights into the trade-offs between performance and computational requirements.

References

1. I. Tobore, J. Li, L. Yuhang, Y. Al-Handarish, A. Kandwal, Z. Nie, and L. Wang, "Deep Learning Intervention for Health Care Challenges: Some Biomedical Domain Considerations," *JMIR Medical Informatics*, vol. 9, no. 8, p. e21032, 2021.
2. C. Baumgartner, K. Petersen, L. M. Koch, and E. Konukoglu, "The Importance of Being Small: Improved Inference in the Inference Phase of Deep Learning for Medical Image Computing Applications," in *International Workshop on Simulation and Synthesis in Medical Imaging*. Springer, 2018, pp. 55–63.
3. Ö. Çiçek, A. Abdulkadir, S. S. Lienkamp, T. Brox, and O. Ronneberger, "3D U-Net: Learning Dense Volumetric Segmentation from Sparse Annotation," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2016, pp. 424–432.
4. N. Lin, H. Lu, J. Gao, S. Qiao, and X. Li, "VNet: A Versatile Network for Efficient Real-Time Semantic Segmentation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 15. AAAI Press, 2021, pp. 13 474–13 482. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/17826
5. E. R. Schreiber, S. E. Olabarriaga, and A. Moelker, "Slicing 3D objects into multi-resolution 2D images for medical applications: A comparative review," *Journal of Digital Imaging*, vol. 23, no. 6, pp. 692–721, 2010.
6. M. Heinrich, M. Blendowski, J. Köster, J. Lindner, and C. Botha, "Reducing model complexity for segmentation of low-contrast structures," in *Medical Imaging 2021: Image Processing*, vol. 11596. International Society for Optics and Photonics, 2021, p. 115962S.
7. O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, vol. 9351, pp. 234–241, 2015.
8. F. Isensee and K. H. Maier-Hein, "nnU-Net: Self-adapting Framework for U-Net-Based Medical Image Segmentation," *Nature Methods*, vol. 18, pp. 203–211, 2021.
9. H. R. Roth, H. Oda, X. Zhou, N. Shimizu, Y. Yang, Y. Hayashi, M. Oda, M. Fujiwara, K. Misawa, and K. Mori, "New Techniques for Semantic Segmentation of 3D MRI using 2D U-Nets," *Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pp. 645–652, 2014.
10. Z. Zhou, M. M. R. Siddiquee, N. Tajbakhsh, and J. Liang, "UNet++: A Nested U-Net Architecture for Medical Image Segmentation," *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*, pp. 3–11, 2018.
11. V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 12, pp. 2481–2495, 2017.
12. 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.
13. K. H. Zou, S. K. Warfield, A. Bharatha, C. M. Tempany, M. R. Kaus, S. J. Haker, W. M. Wells III, F. A. Jolesz, and R. Kikinis, "Statistical validation of image segmentation quality based on a spatial overlap index," *Scientific reports*, vol. 4, p. 6658, 2004.
14. M. Belkin, D. Hsu, S. Ma, and S. Mandal, "Reconciling modern machine-learning practice and the classical bias–variance trade-off," *Proceedings of the National Academy of Sciences*, vol. 116, no. 32, pp. 15 849–15 854, 2019.