



# Med-NCA: Robust and Lightweight Segmentation with Neural Cellular Automata

John Kalkhof<sup>(✉)</sup> , Camila González , and Anirban Mukhopadhyay 

Darmstadt University of Technology, Karolinenplatz 5, 64289 Darmstadt, Germany  
john.kalkhof@gris.tu-darmstadt.de

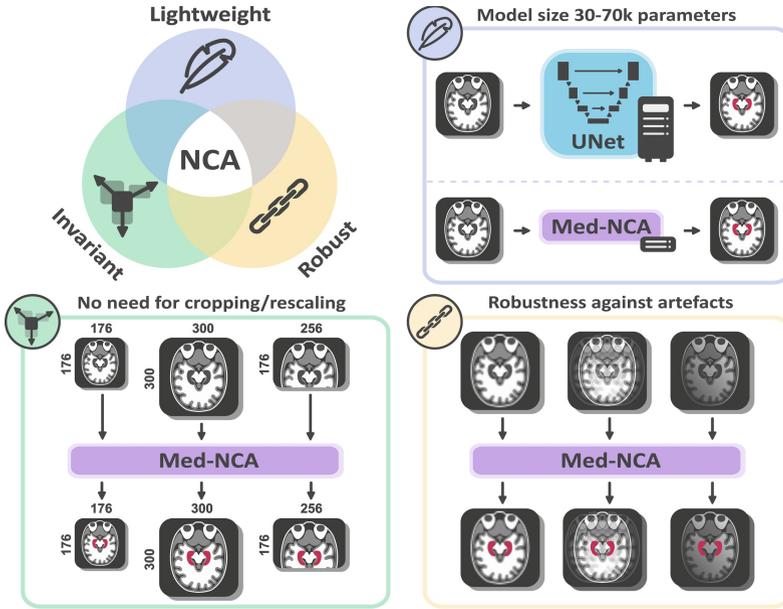
**Abstract.** Access to the proper infrastructure is critical when performing medical image segmentation with Deep Learning. This requirement makes it difficult to run state-of-the-art segmentation models in resource-constrained scenarios like primary care facilities in rural areas and during crises. The recently emerging field of Neural Cellular Automata (NCA) has shown that locally interacting *one-cell* models can achieve competitive results in tasks such as image generation or segmentations in low-resolution inputs. However, they are constrained by high VRAM requirements and the difficulty of reaching convergence for high-resolution images. To counteract these limitations we propose Med-NCA, an end-to-end NCA training pipeline for high-resolution image segmentation. Our method follows a two-step process. Global knowledge is first communicated between cells across the downsampled image. Following that, patch-based segmentation is performed. Our proposed Med-NCA outperforms the classic UNet by 2% and 3% Dice for hippocampus and prostate segmentation, respectively, while also being **500 times smaller**. We also show that Med-NCA is by design invariant with respect to image scale, shape and translation, experiencing only slight performance degradation even with strong shifts; and is robust against MRI acquisition artefacts. Med-NCA enables high-resolution medical image segmentation even on a Raspberry Pi B+, arguably the smallest device able to run PyTorch and that can be powered by a standard power bank.

**Keywords:** Neural Cellular Automata · Medical Image Segmentation · Robustness

## 1 Introduction

State-of-the-art medical image segmentation is dominated by UNet-style architectures [18], which still perform at the top of most grand challenges in its various forms [12]. This trend of task-specific optimizations of UNet-style models is usually accompanied by diminishing returns regarding model size versus performance. The increase in model complexity raises serious concerns that machine learning cannot be leveraged in resource-constrained environments [1]. In settings

such as primary care facilities in rural areas, only minimal computing infrastructure is available [4], so it is challenging to deploy models requiring large GPUs. UNet-style models are also particularly susceptible to the *domain shift* problem [7] and have difficulty generalising to other input resolutions. To mitigate this, shifts are often brute-forced into the training pipeline by adding augmentations like translation or simulated acquisition artefacts [5].



**Fig. 1.** NCA models are *lightweight* due to their small size and asynchronous inference, and can be run on low-powered systems. They are also, by design, *invariant* to the input scale and field of view. Further, they are *robust* against image artefacts.

Unlike most state-of-the-art methods that rely on optimising UNet-based frameworks, such as the nnUNet pipeline [12], we investigate a fundamentally different learning system that is by design **lightweight**, **robust** and **input-invariant**, yet achieves **reliable performance**. We introduce Neural Cellular Automata (NCA) as our base architecture, which due to the *one-cell model size*, can be *distributed across any image size* during deployment. The minimal size and asynchronous inference *requires significantly less computing power* than classical models. Additionally, due in part to the limited amount of parameters, a rule has to be learnt that *generalises well* across the problem space, which renders it *robust* by design (illustrated in Fig. 1).

Heavily inspired by the interaction between cells in living organisms, NCAs are minimal models that look at a single cell at a time and can only communicate with their direct neighbours. Global knowledge can be transmitted by deploying

the model on each cell and iteratively applying the same rules. Each iteration increases the perceptive range by one cell in each direction. Recently, NCAs have made advances in tasks like robust image generation [13] and even segmentation for small-resolution natural images [20], all while learning a single local update rule that is applied incrementally to each cell.

Despite its advantages such as lightweight inference, training NCAs requires *exponentially higher video ram (VRAM)* depending on the input size during training, which quickly reaches 20 GB for a single sample with a resolution of  $256 \times 256$ . In medical image processing, this is prohibiting as data is typically high in resolution. The local interaction makes inference on big images difficult (greater than  $100 \times 100$ ), as many steps are required to communicate global knowledge. In addition, *high-resolution images increase the convergence difficulty*. Due to these constraints, previous works on NCAs have focused on small-resolution computer vision benchmarks [2, 9, 13–15, 17]. We solve these limitations with *Med-NCA*, a two-step NCA model. The model distributes global knowledge across a downsampled image in the first step. In the second step, Med-NCA combines the resulting information with high-resolution image patches to perform high-quality segmentations.

We evaluate our proposed Med-NCA on T1-weighted hippocampus and T2-weighted prostate MRI datasets. We first compare the segmentation performance of Med-NCA to classic and efficient UNet-style architectures, where Med-NCA outperforms them by at least 2% for the hippocampus and 3% for the prostate, with a *90 to 500 times smaller model size*, although there is still a 2% and 10% performance gap to the auto ML pipeline nnUNet. Secondly, we perform an in-depth analysis of three types of **input invariances**: scale, shape and translation. Med-NCA shows *consistent performance* in comparison to UNet-style models and can even outperform the nnUNet for strong shifts in shape and translation. We then investigate the influence of synthetic MRI acquisition artefacts of increasing severity on Med-NCA and UNet. Despite the vastly different one-cell local interaction setup of Med-NCA, our experiments show similar **robustness** to the UNet in terms of anisotropy and bias field and even slightly better robustness to ghosting artefacts. Lastly, we demonstrate that **deployment in low-resource environments** is possible, due to the asynchronous inference and the one-cell model size of NCAs, on a Raspberry Pi Model B+(US\$ 35). There is currently a slightly stronger successor, the Raspberry Pi Zero, which costs US\$ 5.

To ensure reproducibility and drive further research on NCA segmentation, *we make our complete framework available* under [github.com/MECLabTUDA/Med-NCA](https://github.com/MECLabTUDA/Med-NCA).

## 2 Related Work

Recent publications have shown the applicability of NCAs to different tasks, such as robust image generation from a single cell [13] and foreground-background segmentation [20]. In this section, we review relevant related work on NCAs and medical image segmentation.

**Neural Cellular Automata:** NCA models are a one-cell model architecture, recently adapted to convolutional neural networks by Gilpin [6]. NCAs do not look at the whole image globally but instead only interact locally. Each cell can exclusively communicate with its direct neighbours, and all inherit the same learnt rule. By performing multiple iterations, global knowledge can be conveyed between cells. Despite their small size, NCAs have shown robustness in tasks such as image generation [13, 15], where models display a high degree of resilience against perturbations. To the best of our knowledge, only one previous work explores image segmentation with NCAs [20]. The proposed method focuses on foreground-background segmentation on small images of  $64 \times 64$ . While it provides a simple up/downscaling solution for high-resolution images, performance is insufficient for medical image segmentation (see Table 1).

**Medical Image Segmentation:** With the improvements in graphics cards and VRAM availability [12], machine learning models are growing significantly in size. Models like the state-of-the-art nnUNet define 4GB of VRAM as their minimum requirement [12], and thus require proper infrastructure for inference. There have been several attempts to create minimal segmentation models, mainly by modifying the well-established UNet [18]. The ‘Segmentation Models’ python package [10] provides a collection of UNet-style models where the encoder has been replaced with smaller architectures like *EfficientNet* [22], *MobileNetV2* [19], *DenseNet* [11], *ResNet18* [8] and *VGG11* [21]. However, computational requirements are still significant (see Fig. 3).

UNets and other state-of-the-art segmentation models typically have a pyramid-like structure with multiple up- and downscaling blocks. NCAs stand in strong contrast as they are tiny models acting on a single pixel and communicating global knowledge through iteratively applying the same rule. This intrinsic change in the design allows NCAs to maintain a number of parameters several orders of magnitude lower, and consequently to be run on minimal hardware.

### 3 Methodology

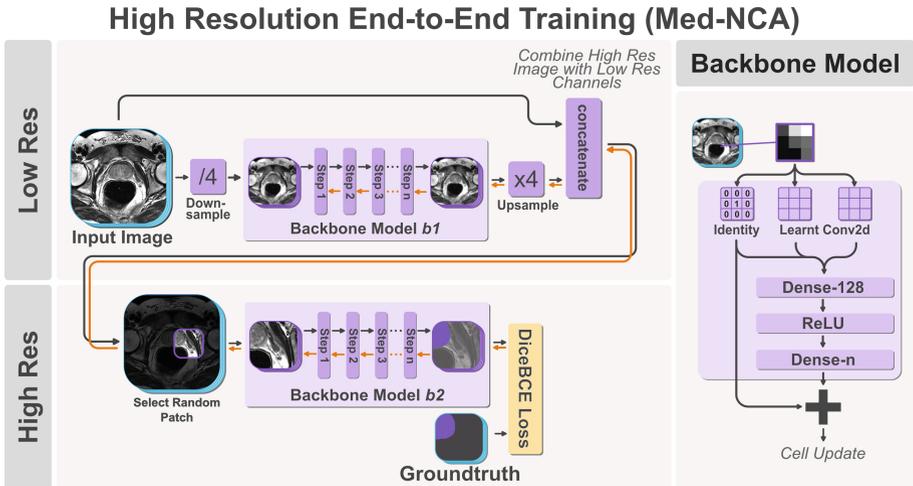
The local architecture of NCAs, where the model deals only with a single cell and its surroundings, allows them to be *lightweight* in terms of storage space and inference time, but this does not come without limitations. Training NCAs end-to-end on images of size  $256 \times 256$  can easily require 20GB of VRAM (for batch size 1). This is because NCAs are replicated across all input image pixels, and backpropagation is performed through all the iterative steps of the model, increasing VRAM needs. VRAM requirements are therefore dependent on model size, input size and number of iterations/steps.

An additional consideration is that increasing the difficulty of the problem (e.g. directly learning on the full-scale high-resolution image) *can result in NCAs not converging*. Med-NCA reduces VRAM needs and simplifies the segmentation problem by separating it into two steps (illustrated in Fig. 2). Its standard configuration reduces the required VRAM for training by a factor of 16 compared to a full-resolution learning setup and enough steps to increase the perceptive range

of each cell to a global scale. This could be improved even further by adding one or more downsampling steps into the pipeline if the need arose.

### 3.1 Med-NCA

Med-NCA is our main methodological contribution defining a pipeline for training NCAs on high-resolution images. It is optimised to reduce VRAM as well as simplify training for bigger images. We illustrate the training procedure in Fig. 2. We start by describing the backbone NCAs we use in Med-NCA, and then describe the training and inference processes.



**Fig. 2.** The *Med-NCA* training strategy relies on cropped images in the ‘High Res’ step of segmentation to limit VRAM requirements. The final inference is performed on the full image and does not require patchification.

### 3.2 Backbone NCA

Med-NCA consists of two identical backbone NCA models that iterate over different scales of the input image. Our models are inspired by the architecture presented in *Growing Neural Cellular Automata* [13].

The backbone model is constructed of  $n$  input channels, where the first  $x$  channels are reserved for the image. The NCA can freely set the remaining  $n - x$  channels. Instead of growing from a single pixel, we adapt it to the segmentation task by immediately distributing it across the whole image.

The proposed model consists of two  $3 \times 3$  learnt convolutional layers and is concatenated with the current cell state, resulting in a state vector of size  $3 * n$ , as illustrated in Fig. 2. It is important that the learnt convolutional layers

use reflect padding, as the model otherwise learns to use the image borders for ‘spatial orientation’, which worsens input invariance capabilities. The state vector is connected to a dense layer with hidden size  $h$ , following a ReLU and another dense layer with the output size  $n$ . The standard configuration of our model uses the following parameters:  $n = 32$ ,  $h = 128$ . These parameters are set to the maximum value at which the model still converges stably, as this provides cells with more memory and allows them to learn a more advanced update rule. For a version of the model that requires 2.5 times fewer parameters, the channel size can be set to  $n = 16$ , resulting in a slight performance degradation, as shown in the ablation study in Table 1. Similar to other NCA approaches [13, 20], we randomly activate  $x = 50\%$  of the cells in each step to simulate asynchronous activation.

### 3.3 Training of Med-NCA

Med-NCA uses multiple-level NCAs for performing the segmentation. We do that by training two backbone NCAs  $b1$  and  $b2$  on different image scales so that training remains stable and the model learns to consider global and detail-rich information.

The pipeline is executed as follows: in the first step, the image is downsampled by a factor of four. Then,  $b1$  is iteratively applied for  $s$  steps on the input image  $x$ . Afterwards, the output  $\bar{x}$  is upscaled back to the original image size. In the next step, we replace the first channel of the output, still containing the upscaled low-resolution image, with its high-resolution counterpart, thus allowing  $b2$  to refine the outputs further. We then take a random patch  $p$  that is of similar size as  $x$  and iteratively apply  $b2$   $s$  times. Lastly, we perform backpropagation with a loss of Dice and binary cross-entropy on the patch prediction and the corresponding ground truth segmentation.

This two-step process lowers VRAM requirements by a factor of 16 during training, making it possible to train Med-NCA on high-resolution images.

### 3.4 Inference

While training has to be specifically adapted to work well with big images, inference is very simple. Extracting a prediction is extremely lightweight and can be carried out on nearly any device that runs PyTorch. The asynchronous and iterative nature of NCAs makes this possible. Inference is only limited by the RAM size, where Med-NCA has a much smaller footprint than, e.g., a UNet. Since NCAs itself are very small, only the size of the image and number of channels are relevant. Further, *during inference patchification is not necessary* as the trained model can be applied directly to the whole image.

## 4 Experimental Setup

The evaluation of our Med-NCA focuses on the three main aspects *robustness*, *input invariance* and *model size*. The evaluation is performed on two segmenta-

tion tasks, namely hippocampus and prostate. We compare our proposed method with UNet-style architectures as well as the auto ML pipeline nnUNet.

**Data:** We use the hippocampus data released for the Medical Segmentation Decathlon (MSD) [2]. The prostate data is a mixture of two datasets, the MSD (CC-BY-SA 4.0 licence) and ISBI 2013 challenge (CC BY 3.0) [3]. The prostate images range from  $320 \times 320$  to  $384 \times 384$ . We scale the image to a training size of  $256 \times 256$ , allowing us to perform out-of-distribution experiments for smaller and higher-resolution scales.

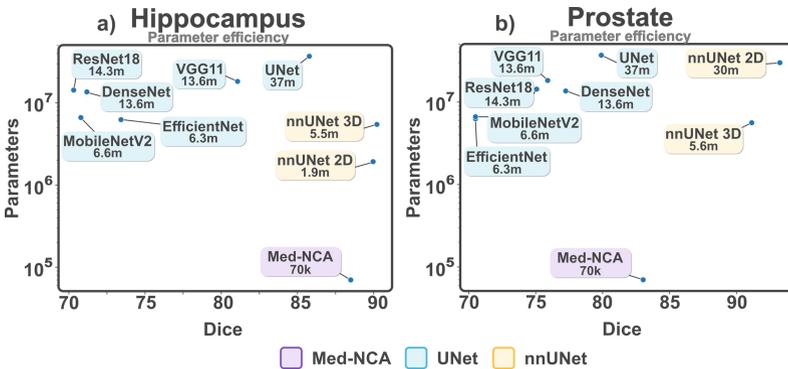
**Evaluated Architectures:** We compare the performance of our proposed Med-NCA to the well-established UNet [18] and different resource-efficient versions. In the latter case, the encoder is replaced by *EfficientNet* [22], *MobileNetV2* [19], *DenseNet* [11], *ResNet18* [8] and *VGG11* [21]. We use the version of each encoder with the least amount of parameters provided in the *Segmentation Models* repository [10]. We also compare our approach to 2D and 3D full-resolution versions of the nnUNet [12].

## 5 Results

The evaluation of our proposed Med-NCA model shows that it deals well with MRI acquisition artefacts and input size variations while reaching state-of-the-art performance on the hippocampus and prostate segmentation datasets.

### 5.1 Performance and Resource Consumption

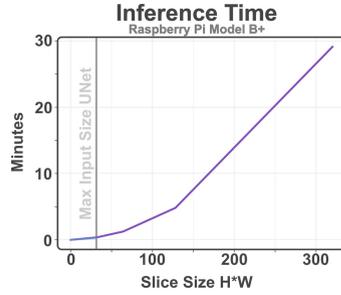
We perform a thorough performance analysis of Med-NCA compared to other UNet-style architectures, and place it in the context to resource consumption. This relation is illustrated in Fig. 3.



**Fig. 3.** We compare Med-NCA to other efficient UNet setups as well as the nnUNet in terms of performance vs. the number of parameters.

We see a general trend of UNet-style models suffering in performance from a decrease in model size. Med-NCA, on the other hand, reaches higher Dice scores than the classic UNet, outperforming it by 2% on the hippocampus data and 3% on the prostate dataset, while also requiring 500 times less trainable parameters.

To show how lightweight Med-NCA is we deploy it on a Raspberry Pi B+ and perform inference up to an image size of  $320 \times 320$  (see Fig. 4). While inference is rather slow, with 30 min per slice for the maximal image size of  $320 \times 320$ , using a more recent system with an *RTX 2060 Super* and an *AMD Ryzen 5600X* inference only takes seconds for a whole MRI. In comparison, the UNet can only be executed on the Raspberry Pi up to an image size of  $32 \times 32$ .



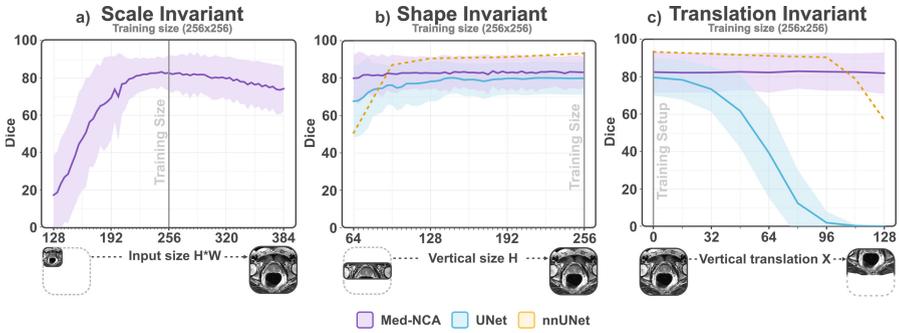
**Fig. 4.** Med-NCA single slice segmentation time on a Raspberry Pi B+ by image size.

## 5.2 Input Invariance

When dealing with medical images, the scale or input size may change after training. A problem with UNet-style segmentation models is that they are not good at adapting to such variability, which our experiments in Fig. 5 show. When UNet faces strong shifts in vertical shape, the Dice drops by 13%, whereas Med-NCA only loses 3%. While the auto ML pipeline nnUNet performs robustly until strong severity of shape changes, it then loses even more performance and drops to 50% Dice. We experience similar results for translational changes. While Med-NCA performs consistently across all introduced shifts, the UNet drops to 0% Dice, indicating strong overfitting on the position of the image. As for shape variations, the nnUNet performs stable until strong translational shifts appear, where it again drops to 56% Dice. The third experiment we conduct is Med-NCA's capability of dealing with different image sizes as it can be arbitrarily distributed across any image size due to its one-cell model setup. Med-NCA shows only slight performance losses of maximal 3% when dealing with images that are in the range of 0.8 up to 1.5 times the training size. When working on MRIs, input invariance is especially relevant as only scanning a partial region or lowering the resolution can *greatly increase capturing speed*. Med-NCA has the considerable advantage that it can be trained on a full upper body scan but perform equally on partial images.

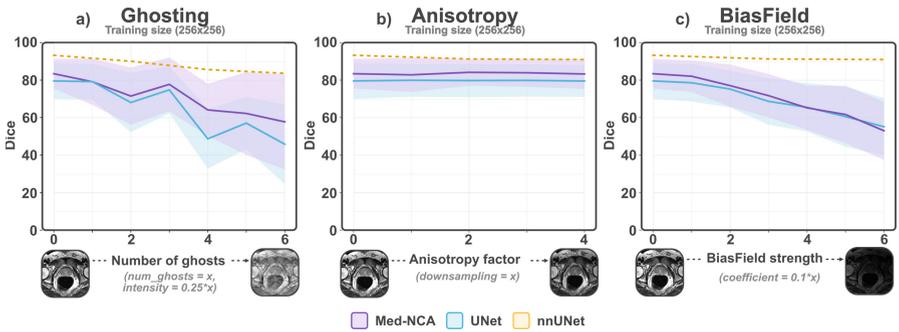
## 5.3 Robustness

Robustness against MRI acquisition artefacts is an important trait in medical image segmentation. In our robustness analysis, illustrated in Fig. 6, we show that Med-NCA can handle anisotropy and bias field artefacts similarly well to



**Fig. 5.** Comparison of our proposed Med-NCA with UNet and the nnUNet in of out-of-distribution (a) scale, (b) shape and (c) translation scenarios.

a classic UNet. Both models experience no drop in performance with anisotropy artefacts up to a severeness factor of 4. In the case of bias field, Med-NCA and UNet drop more than 25% in performance when the severity of the artefacts becomes too strong. In cases of severe ghosting artefacts, we can see the performance of Med-NCA suffers less than a classic UNet, where Med-NCA performs 12% better in the most severe cases of ghosting.



**Fig. 6.** Robustness analysis of our proposed Med-NCA in comparison to UNet and the nnUNet. nnUNet results are collected for fewer data points, indicated by the dashed line, as the non-flexible pipeline requires manual evaluation. Analysis is performed for synthetic MRI acquisition artefacts of increasing severity: (a) Ghosting, (b) Anisotropy and (c) Bias Field (using TorchIO [16]).

As our comparison to the nnUNet demonstrates, an auto ML pipeline can increase the robustness as nnUNet only suffers from slight performance losses across the MRI acquisition artefacts. It is plausible to include a similar setup in the training of Med-CNA and thus improve robustness.

**Table 1.** Comparison of different Med-NCA setups (where  $c = \text{channels}$ ,  $h = \text{hidde nsize}$ ), as well as the Backbone-NCA, previous work on NCA segmentation and a standard 2D UNet.

Model	Hippocampus		Prostate	
	Dice $\uparrow$	# Param. $\downarrow$	Dice $\uparrow$	# Param. $\downarrow$
Med-NCA	<b>0.886 <math>\pm</math> 0.042</b>	70016	<b>0.838 <math>\pm</math> 0.083</b>	70016
Med-NCA $c = 16$	0.873 $\pm$ 0.033	<b>25920</b>	0.822 $\pm$ 0.087	<b>25920</b>
Med-NCA $h = 64$	0.858 $\pm$ 0.047	47530	0.808 $\pm$ 0.125	47530
Backbone-NCA $64 \times 64$	0.871 $\pm$ 0.870	35008	0.789 $\pm$ 0.132	35008
Seg. NCA [20]	0.805 $\pm$ 0.045	39472	0.634 $\pm$ 0.190	39472
UNet	0.858 $\pm$ 0.044	36951555	0.799 $\pm$ 0.099	36951555

## 5.4 Ablation Study

Lastly, we perform an ablation on our approach in Table 1. Our results show that a different trade-off of performance vs. model size can make Med-NCA 2.5x more lightweight while only sacrificing 1% of performance. Further, we can see that the previous NCA segmentation model *Seg. NCA* [20] is not suitable for the task of medical image segmentation as it performs 7.4% worse for hippocampus and even 20.2% worse for the prostate segmentation task.

## 6 Discussion

We have shown that NCA-based architectures are not only suitable for low-resolution imaging tasks, but can also be leveraged for high-resolution image segmentation with our proposed Med-NCA. Since standard NCAs require exponential amounts of VRAM, determined by the input size, this requires an adapted training pipeline. We enable the training with high-resolution images by incorporating patches in the second segmentation step during training. Inference can be performed directly on the full-resolution image.

When comparing Med-NCA to the classical UNet and resource-efficient variations, Med-NCA outperforms them by 2% and 3% Dice for hippocampus and prostate image segmentation, respectively. Med-NCA is also more robust to changes in image scale, shape and translation, which is directly inherited by its one-cell model size. The local interaction prevents the model from knowing where in space information is located and therefore eliminates biases based on position. Further, despite the vastly different one-cell model size with 500 times fewer parameters, Med-NCA shows similar or better robustness than a UNet to MRI acquisition artefacts. Due to the one-cell model size and asynchronous inference, NCAs are lightweight enough to be executed on any hardware that runs PyTorch, which we demonstrate by deploying Med-NCA on a Raspberry Pi B+, that can be powered by any 5 W power source like a standard power bank.

While we have solved the problem of NCAs for high-resolution inputs, one limitation of the present approach is that it only admits 2D image slices. Further

optimisations could make Med-NCA work for 3D inputs by adapting the perceptive field and integrating VRAM improvements based on the three-dimensional space. 3D inputs would make Med-NCA suitable for more datasets and might further improve performance. In addition, although Med-NCA is inherently robust, we experience performance degradation when the severity of the acquisition artefacts becomes too strong, which could be improved upon in future work.

A further limitation is that there is a 10% performance difference between Med-NCA and nnUNet on the prostate task. For future work, we propose the development of a training pipeline similar to the nnUNet, which uses NCA as the core architecture and includes data augmentations and post-processing steps during training. We expect that such a framework could achieve equivalent performance while profiting from the inherited benefits of the NCA architecture like size and scale invariance and general robustness.

## 7 Conclusion

In this work we introduce the Med-NCA segmentation model, which solves the VRAM limitations inherited by NCAs for high-resolution images and is therefore suitable for medical images. Our approach can be run on minimal hardware (e.g. a Raspberry Pi B+), is inherently robust against scale, shape, translation and image artefacts, and reaches near state-of-the-art segmentation performance. We compare our model to UNet-style models optimised for low parameter size and show that Med-NCA not only achieves a higher Dice score, but also outperforms them in terms of input invariance, robustness and resource use. This makes Med-NCA a perfect candidate for primary care scenarios with limited infrastructure and highly variable imaging equipment.

## References

1. Ajani, T.S., Imoize, A.L., Atayero, A.A.: An overview of machine learning within embedded and mobile devices-optimizations and applications. *Sensors* **21**(13), 4412 (2021)
2. Antonelli, M., et al.: The medical segmentation decathlon. *Nat. Commun.* **13**(1), 1–13 (2022)
3. Bloch, N., et al.: NCI-ISBI 2013 challenge: automated segmentation of prostate structures (2015). <https://doi.org/10.7937/K9/TCIA.2015.zF0vIOPv>
4. Boppart, S.A., Richards-Kortum, R.: Point-of-care and point-of-procedure optical imaging technologies for primary care and global health. *Sci. Transl. Med.* **6**(253), 253rv2 (2014)
5. Chlap, P., Min, H., Vandenberg, N., Dowling, J., Holloway, L., Haworth, A.: A review of medical image data augmentation techniques for deep learning applications. *J. Med. Imaging Radiat. Oncol.* **65**(5), 545–563 (2021)
6. Gilpin, W.: Cellular automata as convolutional neural networks. *Phys. Rev. E* **100**(3), 032402 (2019)
7. González, C., et al.: Distance-based detection of out-of-distribution silent failures for Covid-19 lung lesion segmentation. *Med. Image Anal.* **82**, 102596 (2022)

8. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)
9. Hernandez, A., Vilalta, A., Moreno-Noguer, F.: Neural cellular automata manifold. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 10020–10028 (2021)
10. Iakubovskii, P.: Segmentation models PyTorch (2019). [https://github.com/qubvel/segmentation\\_models.pytorch](https://github.com/qubvel/segmentation_models.pytorch)
11. Iandola, F., Moskewicz, M., Karayev, S., Girshick, R., Darrell, T., Keutzer, K.: DenseNet: implementing efficient convnet descriptor pyramids. arXiv preprint [arXiv:1404.1869](https://arxiv.org/abs/1404.1869) (2014)
12. Isensee, F., Jaeger, P.F., Kohl, S.A., Petersen, J., Maier-Hein, K.H.: nnU-Net: a self-configuring method for deep learning-based biomedical image segmentation. *Nat. Methods* **18**(2), 203–211 (2021)
13. Mordvintsev, A., Randazzo, E., Niklasson, E., Levin, M.: Growing neural cellular automata. *Distill* **5**(2), e23 (2020)
14. Niklasson, E., Mordvintsev, A., Randazzo, E., Levin, M.: Self-organising textures. *Distill* **6**(2), e00027–003 (2021)
15. Palm, R.B., Duque, M.G., Sudhakaran, S., Risi, S.: Variational neural cellular automata. In: International Conference on Learning Representations (2021)
16. Pérez-García, F., Sparks, R., Ourselin, S.: TorchIO: a python library for efficient loading, preprocessing, augmentation and patch-based sampling of medical images in deep learning. *Comput. Methods Programs Biomed.* 106236 (2021). <https://doi.org/10.1016/j.cmpb.2021.106236>. <https://www.sciencedirect.com/science/article/pii/S0169260721003102>
17. Randazzo, E., Mordvintsev, A., Niklasson, E., Levin, M., Greydanus, S.: Self-classifying MNIST digits. *Distill* **5**(8), e00027–002 (2020)
18. Ronneberger, O., Fischer, P., Brox, T.: U-Net: convolutional networks for biomedical image segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) MICCAI 2015. LNCS, vol. 9351, pp. 234–241. Springer, Cham (2015). [https://doi.org/10.1007/978-3-319-24574-4\\_28](https://doi.org/10.1007/978-3-319-24574-4_28)
19. Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L.C.: MobileNetv 2: inverted residuals and linear bottlenecks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4510–4520 (2018)
20. Sandler, M., Zhmoginov, A., Luo, L., Mordvintsev, A., Randazzo, E., et al.: Image segmentation via cellular automata. arXiv preprint [arXiv:2008.04965](https://arxiv.org/abs/2008.04965) (2020)
21. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint [arXiv:1409.1556](https://arxiv.org/abs/1409.1556) (2014)
22. Tan, M., Le, Q.: EfficientNet: rethinking model scaling for convolutional neural networks. In: International Conference on Machine Learning, pp. 6105–6114. PMLR (2019)