# Neural Cellular Automata

**Finn Radatz, Hannah Willkomm, Laurenz Kammeyer, Marlon May**

TECHNISCHE UNIVERSITÄT DARMSTADT

## Introduction

Neural Cellular Automata (NCA) combine cellular automata with neural networks. They function like normal cellular automata with update rules learned by neural networks to achieve specific goals. NCAs can model complex emergent behaviors and have applications in image generation and pattern formation. As self-organizing systems, they can grow and regenerate patterns, inspired by biological processes like morphogenesis. They promise to improve our understanding of intercellular organisation and communication patterns. For this work, our goal is to reproduce a target image from a seed-pixel. Efforts are also made to introduce transformers to this domain.

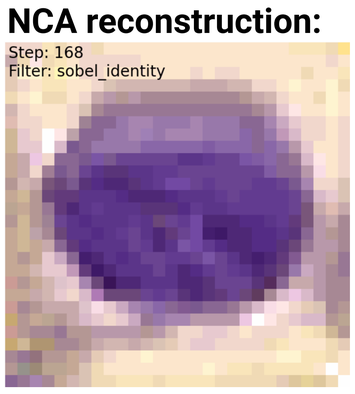
## Training Dataset

MedMNIST is a standardized collection of biomedical image datasets for machine learning research and education, inspired by MNIST.

- 18 datasets (12 x 2D, 6 x 3D)
- Multiple imaging modalities: X-Ray, CT, MRI, etc.
- Sample sizes: 100 to 100,000+ images

**Example images:**

**NCA reconstruction:**



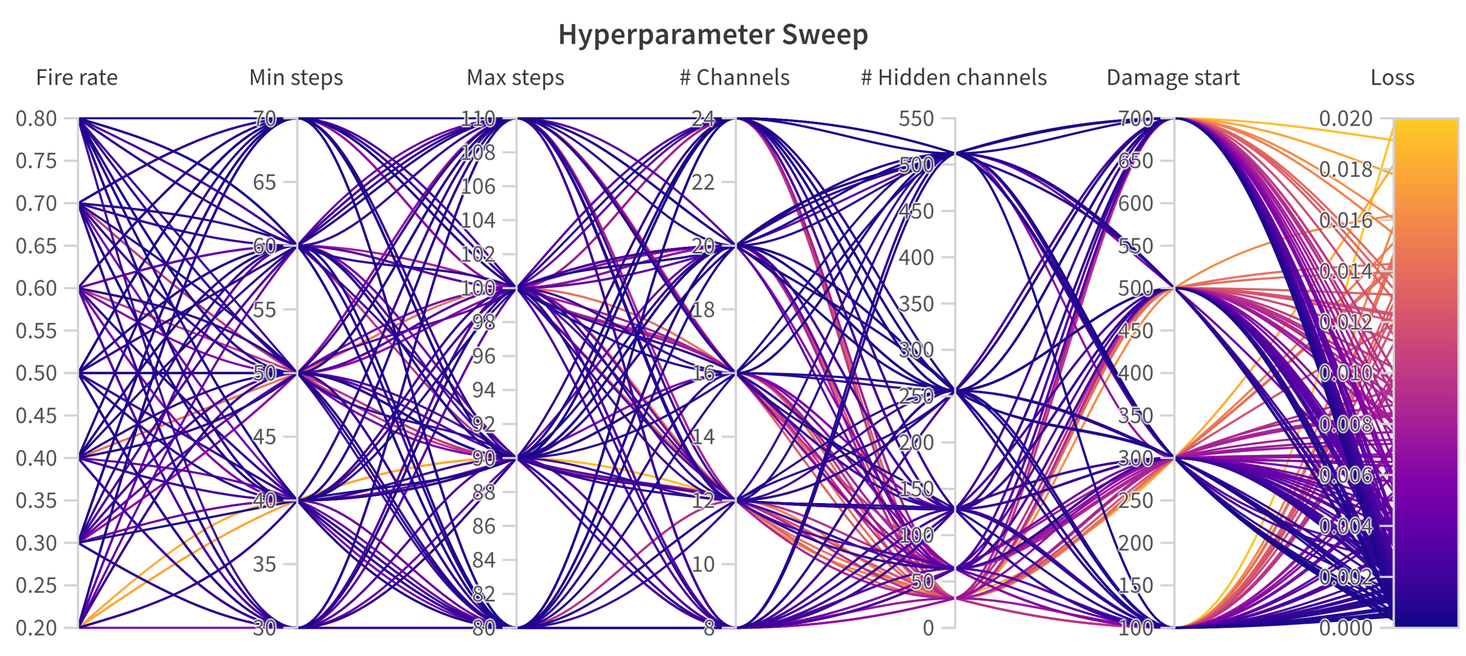

## Methods

1. Regular RGB state-space is expanded by variable number of additional values per pixel (to $\sim 16$). The extra dimensions have no predefined purpose and can be used by the model for anything.

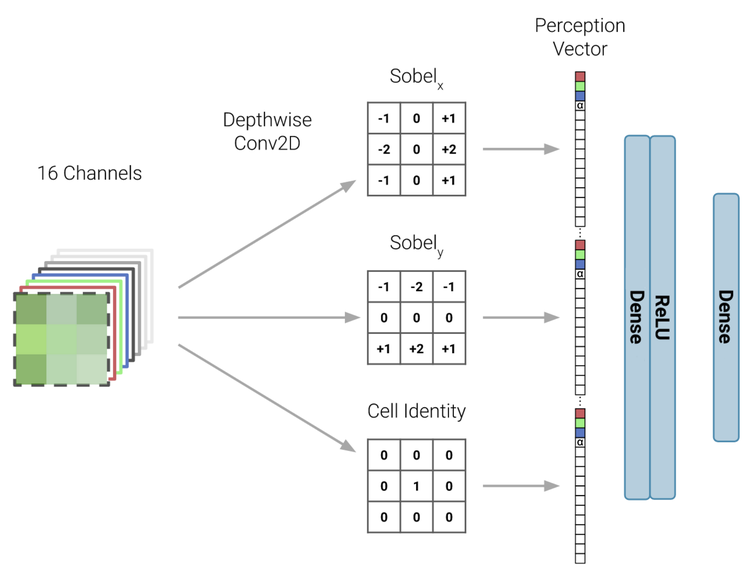
2. Several filters are used as Kernels for a convolution with each cell state. Filters are set before training, but could also be learnied.

3. Resultant vectors of step 2 are concatenated and extended using the cell's own state to form the cell's perception

4. Neural Network (NN) is used to convert percept input into new cell state (correspond to update rule of NN). Updates are subject to random dropout to mimic asynchronous development of biological sytems.




## Model Architecture




## Experiments

Our hyperparameter investigation shows that the number of hidden channels is the most important hyperparameter with a negative correlation of -0.63.

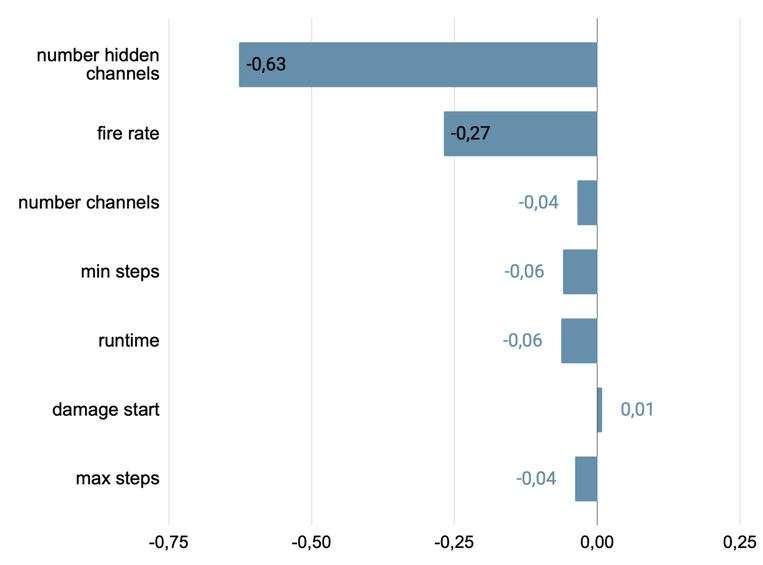
Our baseline model uses the Sobel-Identity filter and a MSE loss.

Further, we analyzed the loss of our NCA using different **loss functions**: *Hinge, MSE, Manhattan, SSIM, Combined SSIM & L1* and the following **filters**: *Sobel, Identity, Laplacian, Gaussian, Sobel - Identity*, resulting in 25 different combinations.

To our surprise none of these filter-loss combinations were able to come close to the performance of the base model. The only combination that delivered reasonable results were the Manhattan loss and the Sobel-Identity filter.

## Hyperparameters

The plot above illustrate the results of the hyperparameter sweep for our Baseline model with each line representing one of the 194 runs. The correlation between each modified hyperparameter and the loss is visualized below.




## Conclusion

As with all machine learning approaches, their level of capability scales with their complexity. NCAs offer nice computational properties, e.g., good parallelizability and the ability for continuous operation. Due to their ability to naturally mimic the behaviour of concentration gradients, they could also prove very useful for simulating various physical, chemical and biological processes. As far as we can tell, they do not scale as quickly as other approaches, resulting in even larger model architectures. To our knowledge, there were no experiments for NCAs with parameter counts on the scale of modern architectures, so it is not possible to say anything about their capabilities using billions of parameters.