Neural Cellular Automata

EI9IT385 AI for video games

S9 - Year 2024-2025

Author: Mohamed Kherraz

February 2, 2025

Contents

0.1 Introduction

One of the major problems faced in modern biology is the lack of understanding of how complex organisms are formed from a single cell which evolute into larger and more complex structures. This process is called morphogenesis and it allows cells to know where to grow, what to form, and when to stop. The main difficulty in this field is understand how that large scale information is stored in one cell and how that process can be replicated and controlled. In this report we propose the use of neural cellular automata which uses the information present in neighboring cells to build new cells.

0.2 Neural Cellular Automata (NCA)

Cellular Automata (CA)

Define Cellular Automata (CA), a computational model used to study complex systems and pattern formation. Explain how CA relies on simple rules applied to cells in a grid, where the state of a cell depends on the state of its neighbors.

Neural Cellular Automata (NCA)

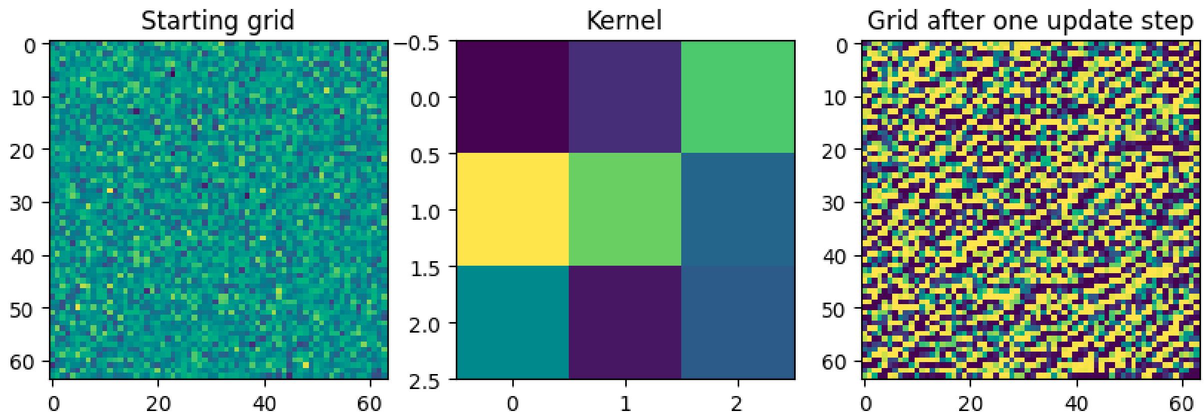

Figure 1: One step of NCA rule application in an example grid. The state of each cell is updated based on its neighbors and kernel rules.

Neural Cellular Automata (NCA) extend the classical CA framework by incorporating neural networks to define the rules defining cell behavior, as described in figure 1 and 2 . Unlike traditional CA, where rules are handcrafted, NCA rules are learned through gradient-based optimization. This allows NCA to model more complex behaviors. The key advantage of NCA is its differentiability, which enables the use of machine learning techniques to optimize the rules for specific tasks, such as simulating morphogenesis or generating textures.

The figure 3, shows an example of advanced simulation using neural cellular automata with a simple rule of inverse gaussian activation function.

Training NCA

Training NCA involves defining a loss function that measures the difference between the desired and actual states of the system. Gradient descent is then used to update the neural

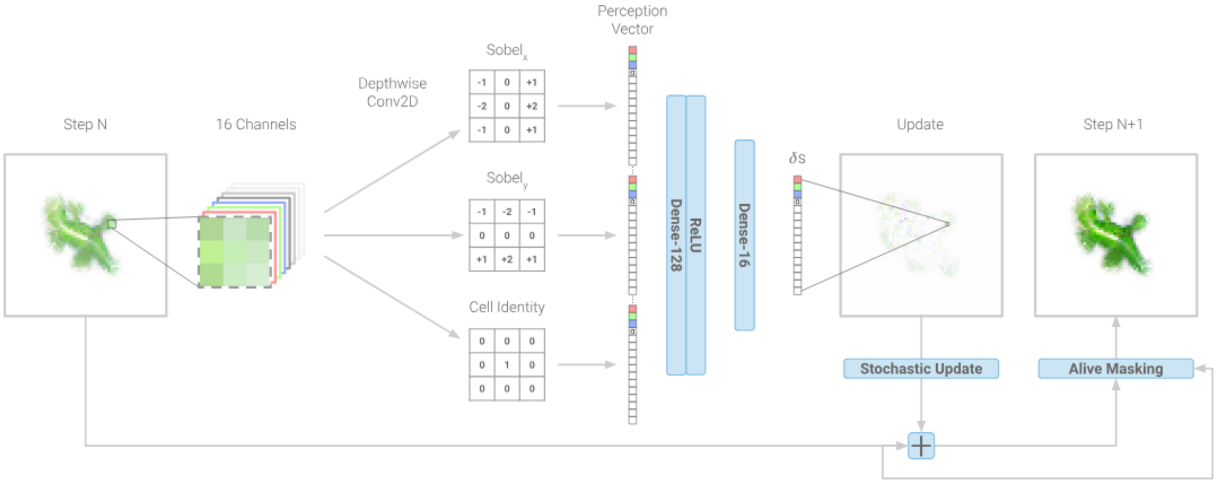

Figure 2: The structure of NCA according to paper [1]

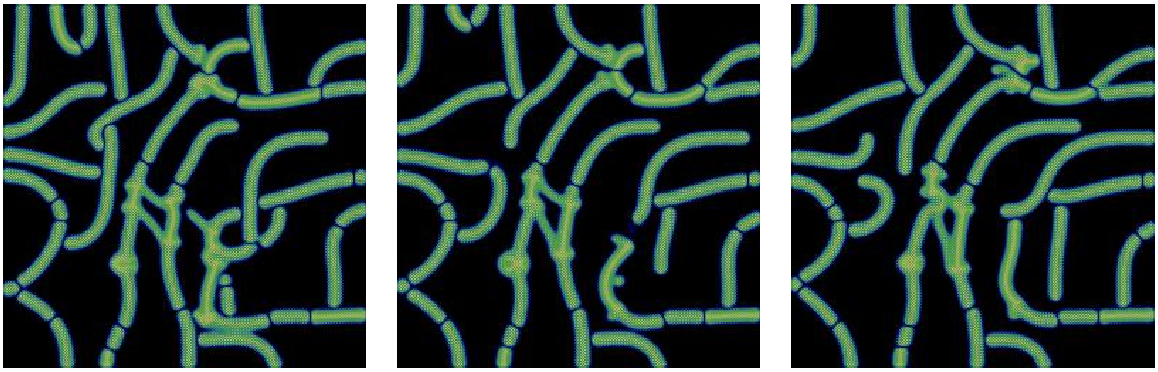

Figure 3: Worm simulation using the inverse gaussian activation function.

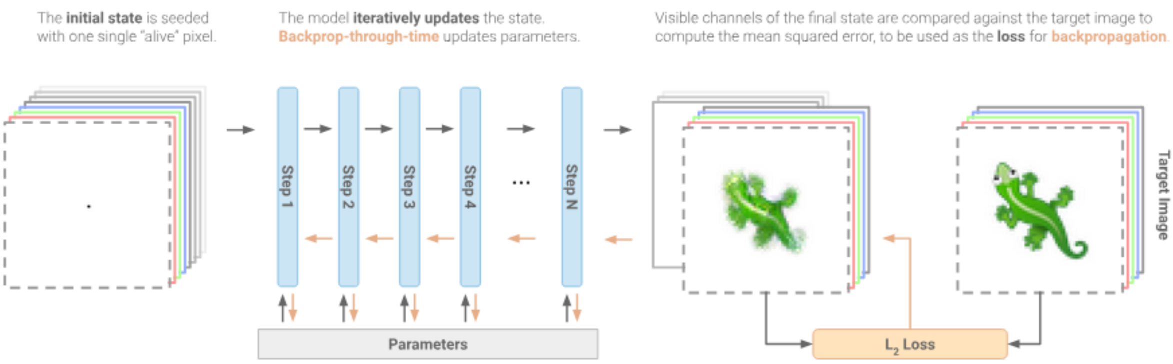

Figure 4: The structure of NCA according to paper [1]

network parameters to minimize this loss. For example, in the case of morphogenesis, the loss function might penalize deviations from a target shape or pattern. This training process was further improved by damaging states to teach the NCA to regrow it's shape back using an iterative approach as depicted in figure 4.

0.3 Experimentation

To demonstrate the capabilities of NCA, we conducted experiments on pattern regeneration and growth. The system was initialized with a simple seed pattern, and the NCA rules were applied iteratively to grow the structure of a golden horse.

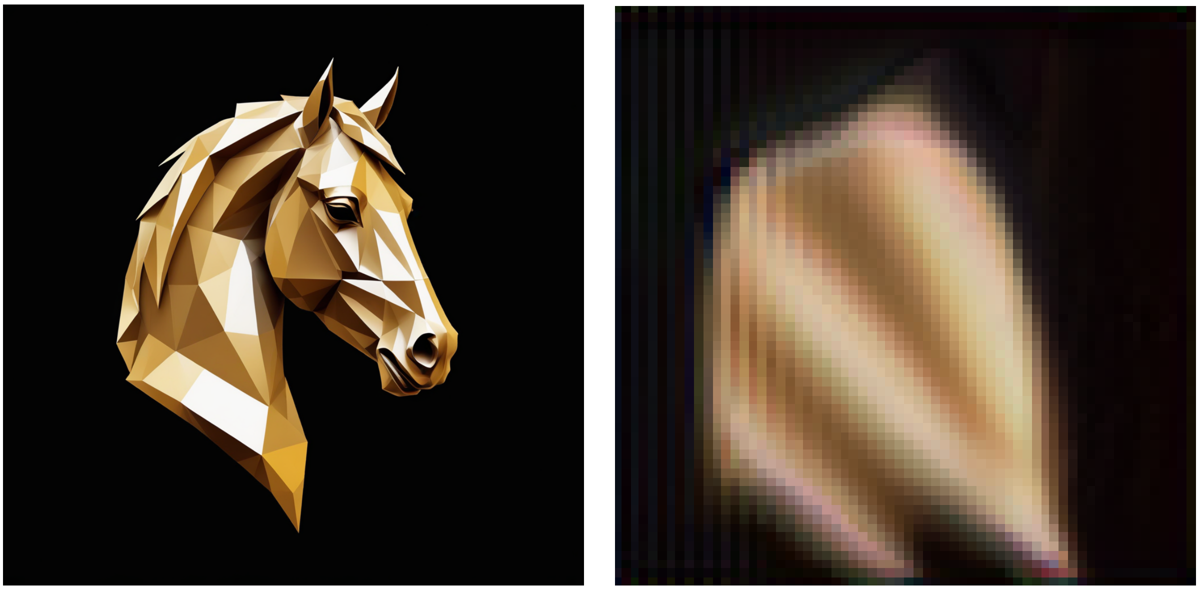

Figure 5: Golden horse image constructed using NCA

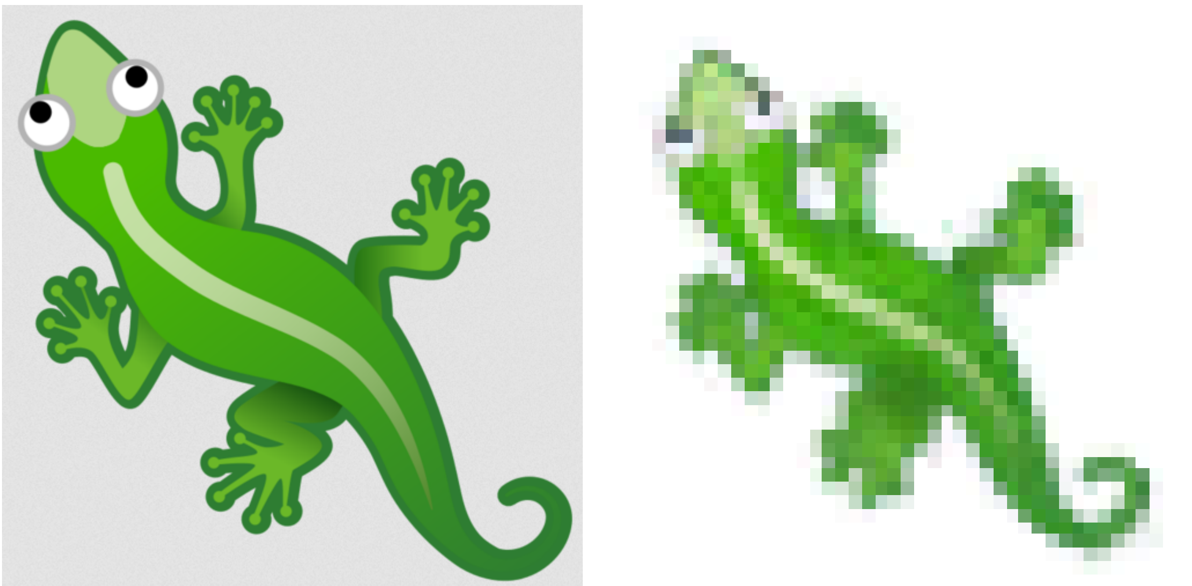

Figure 6: Lizard image constructed using NCA

This NCA was trained for 1000 epochs on a RTX 3060 with a pool size of 1024.

The figure 5, shows that our model is slowly converging to the target structure, but it doesn't match it exactly, this might be due to two main problems, the first is that the NCA needs more training on the golden horse image and the second is that the golden horse image is very complex compared to the NCA architecture.

In the best case scenario the model would be able to exactly reproduce the wanted structure as shown in figure 6. For more insight into neural cellular automata and an interactive experimentation we suggest the use of the demonstration provided by the paper [1] at link.

The results show that NCA can generate complex and stable patterns, even when parts of the system are damaged or removed. This highlights the potential of NCA for modeling biological processes and designing adaptive systems.

0.4 Conclusion

Neural Cellular Automata provide a powerful framework for modeling complex systems and understanding the principles of self-organization. By combining the simplicity of cellular automata with the flexibility of neural networks, NCA can simulate a wide range of phenomena, from biological growth to procedural generation.

Bibliography

[1] Mordvintsev, et al., "Growing Neural Cellular Automata", Distill, 2020.