Your Loss

Neural Cellular Automaton driven by adjustable loss function

Neural Cellular Automaton

A neural cellular automaton (NCA) is a 2D grid of cells updated by a shared local rule. As in classical cellular automata, each cell observes only its local neighborhood. The update rule is represented by a neural network.
Each cell stores a state

$$\mathrm{s}(x,y) = \big(v(x,y), h(x,y)\big),$$

where v is a visible grayscale value and h is a hidden channel.

Local Update Rule

At each step:

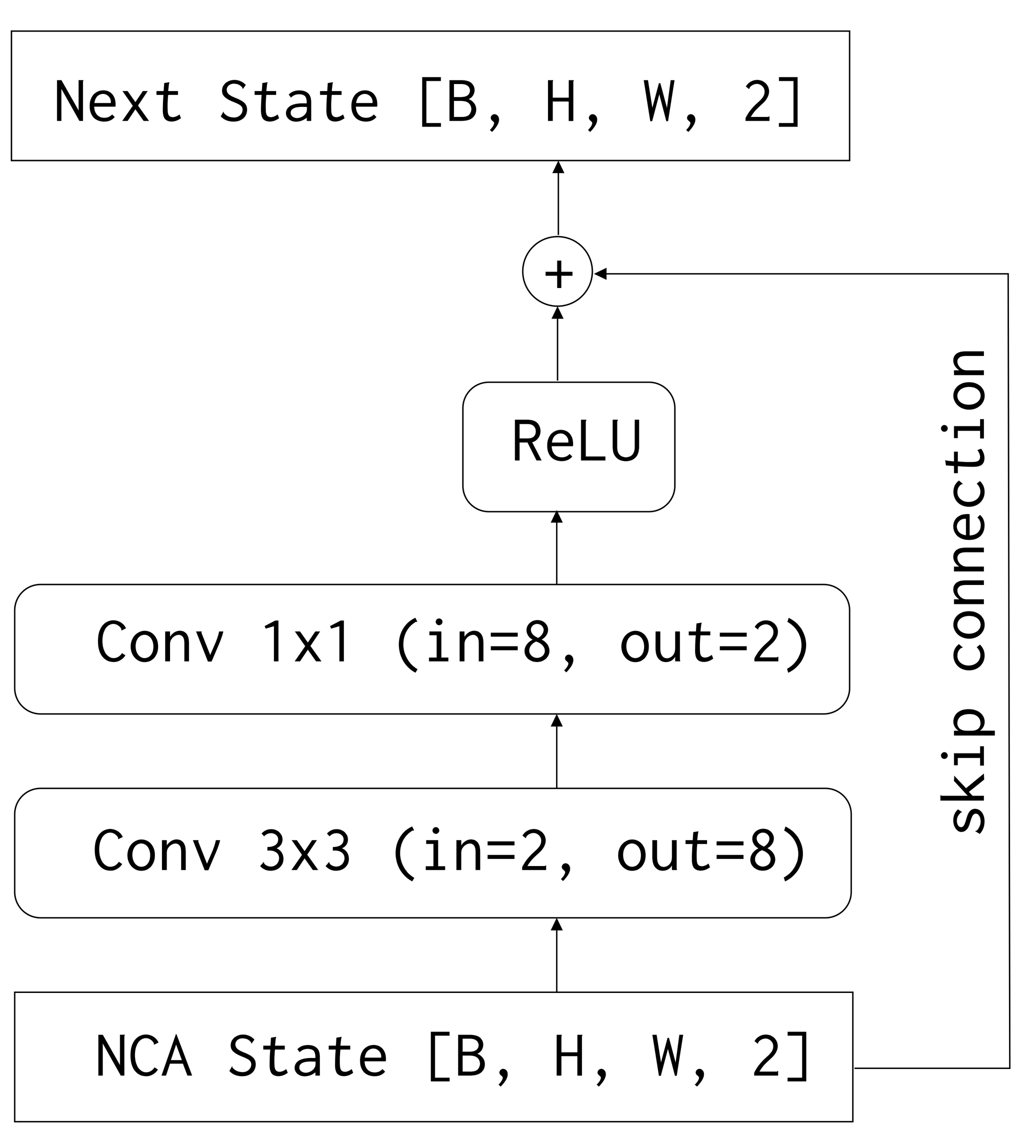
$$\mathrm{s}_{t+1} = \mathrm{s}_t + \Delta \mathrm{s}_t$$

$$\Delta \mathrm{s} = \mathrm{Conv}_{1\times1}\left(\mathrm{ReLU}\left(\mathrm{Conv}_{3\times3}(\mathrm{s})\right)\right)$$

- 3×3 convolution (8 kernels)
- ReLU nonlinearity
- 1×1 convolution
- residual (skip) connection

The network is shallow but applied repeatedly. Time acts as depth.

Network Architecture



Learning Process

The system is evolved for a fixed number of steps ($T = 10$ during training):

$$\mathrm{s}_0 \to \mathrm{s}_1 \to \cdots \to \mathrm{s}_T$$

A loss is evaluated only on the final visible state v_T. Gradients are backpropagated through all steps, making the system recurrent.
Although trained for a limited horizon, the automaton is intended to run indefinitely. Interesting structures often appear far beyond the training window.

Visible and Hidden Channels

The visible channel v is constrained to grayscale. The hidden channel h is unconstrained and acts as internal memory.
For visualization, the hidden state is compressed using a nonlinear mapping (tanh) and can be inspected interactively.

Compound Objective

The system is not trained on example images. Instead, it is optimized using a compound loss:

$$\mathcal{L} = \sum_i w_i \mathcal{L}_i$$

Each loss term measures a single visual property. Each contains a target parameter (Greek letters below), which can be adjusted interactively in the application.

Loss Functions

Notation

- $v(x,y)$ – visible grayscale value
- $\mathbb{E}[\cdot]$ – spatial average
- ∇v – spatial gradient
- Δv – discrete Laplacian
- G_σ – Gaussian blur kernel

Edge Density

$$\mathcal{L}_{\text{edge}} = \left(\mathbb{E}[\|\nabla v\|] - \rho\right)^2$$

Low ρ should yield smooth regions. High ρ encourages fine detail.

Laplacian Smoothness

$$\mathcal{L}_{\text{lap}} = \left(\mathbb{E}[|\Delta v|] - \lambda\right)^2$$

Controls surface roughness.

Mean Brightness

$$\mathcal{L}_{\text{mean}} = \left(\mathbb{E}[v] - \mu\right)^2$$

Sets the global brightness level.

Global Contrast (MAD)

$$\mathcal{L}_{\text{contrast}} = \left(\mathbb{E}[|v - \mathbb{E}[v]|] - c\right)^2$$

Controls overall contrast.

Neighbor Correlation

$$\mathcal{L}_{\text{corr}} = \left(\mathbb{E}[v(x,y)v(x+1,y)] - \kappa\right)^2$$

Controls local coherence.

Low-Pass Preference

$$\mathcal{L}_{\text{lp}} = \mathbb{E}\left[\left(v - G_\sigma * v\right)^2\right]$$

Penalizes energy above scale σ, implicitly favoring large-scale structure.