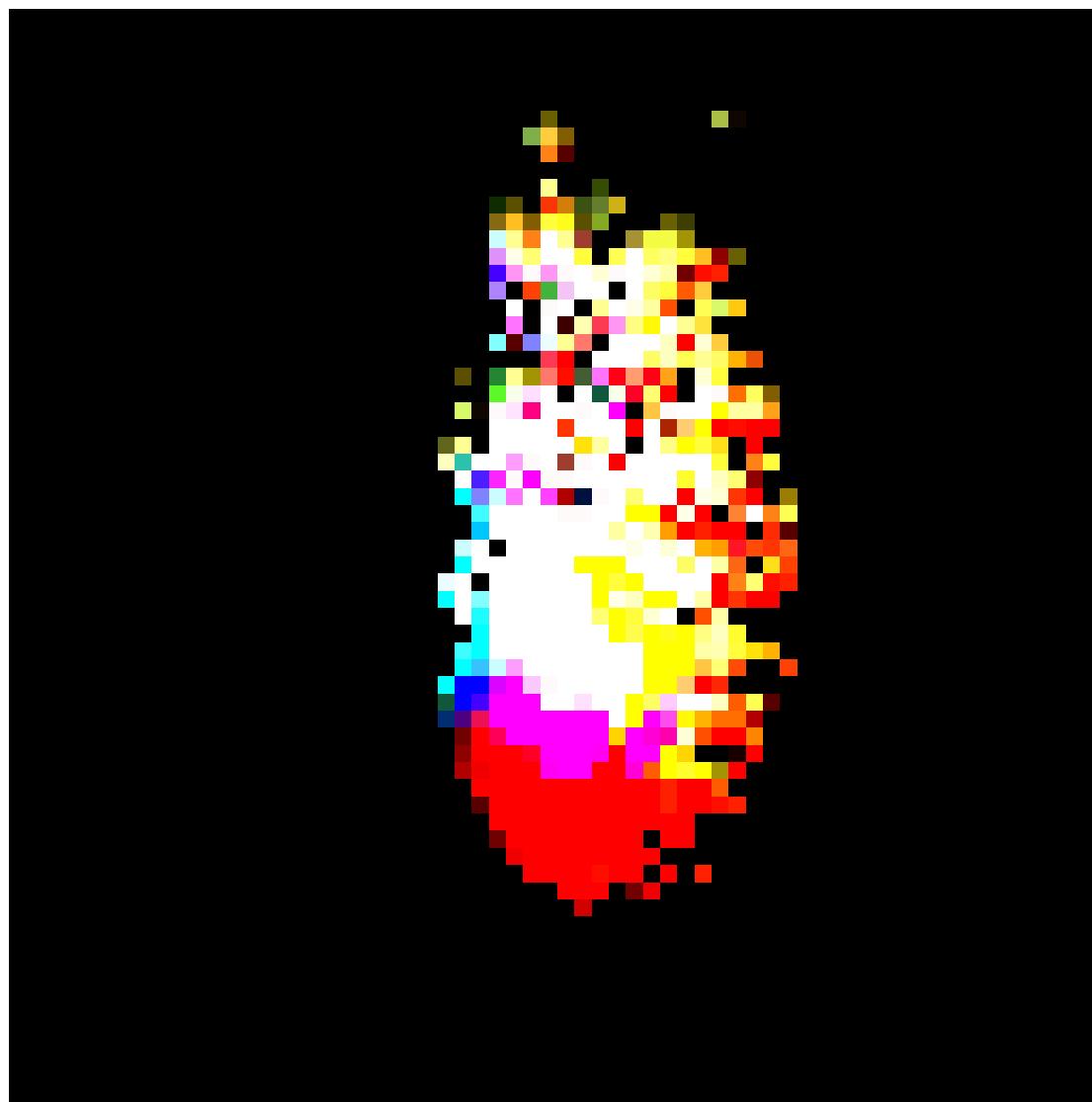
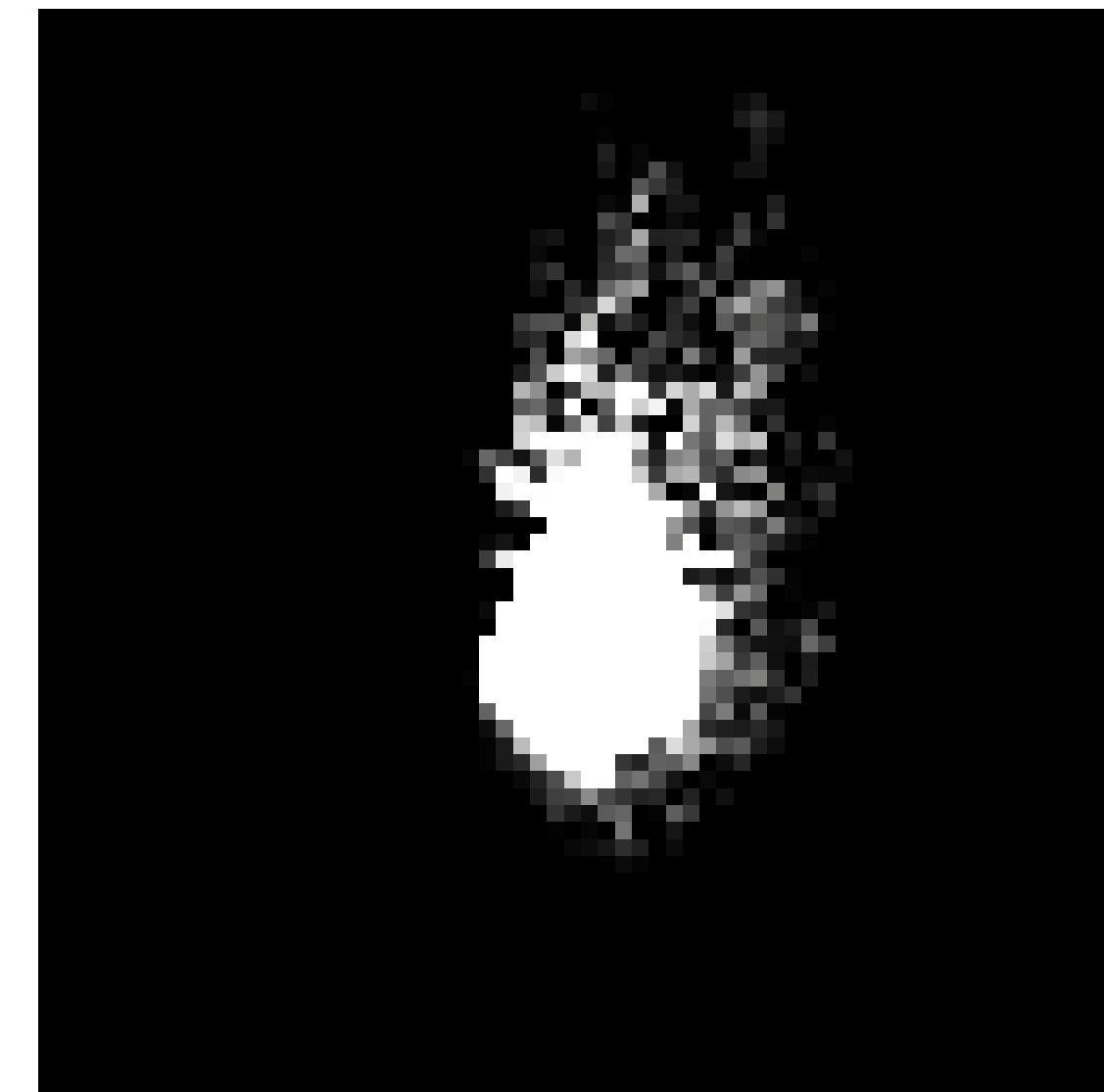


Neural Cellular Automata (NCA) to learn Physics simulations



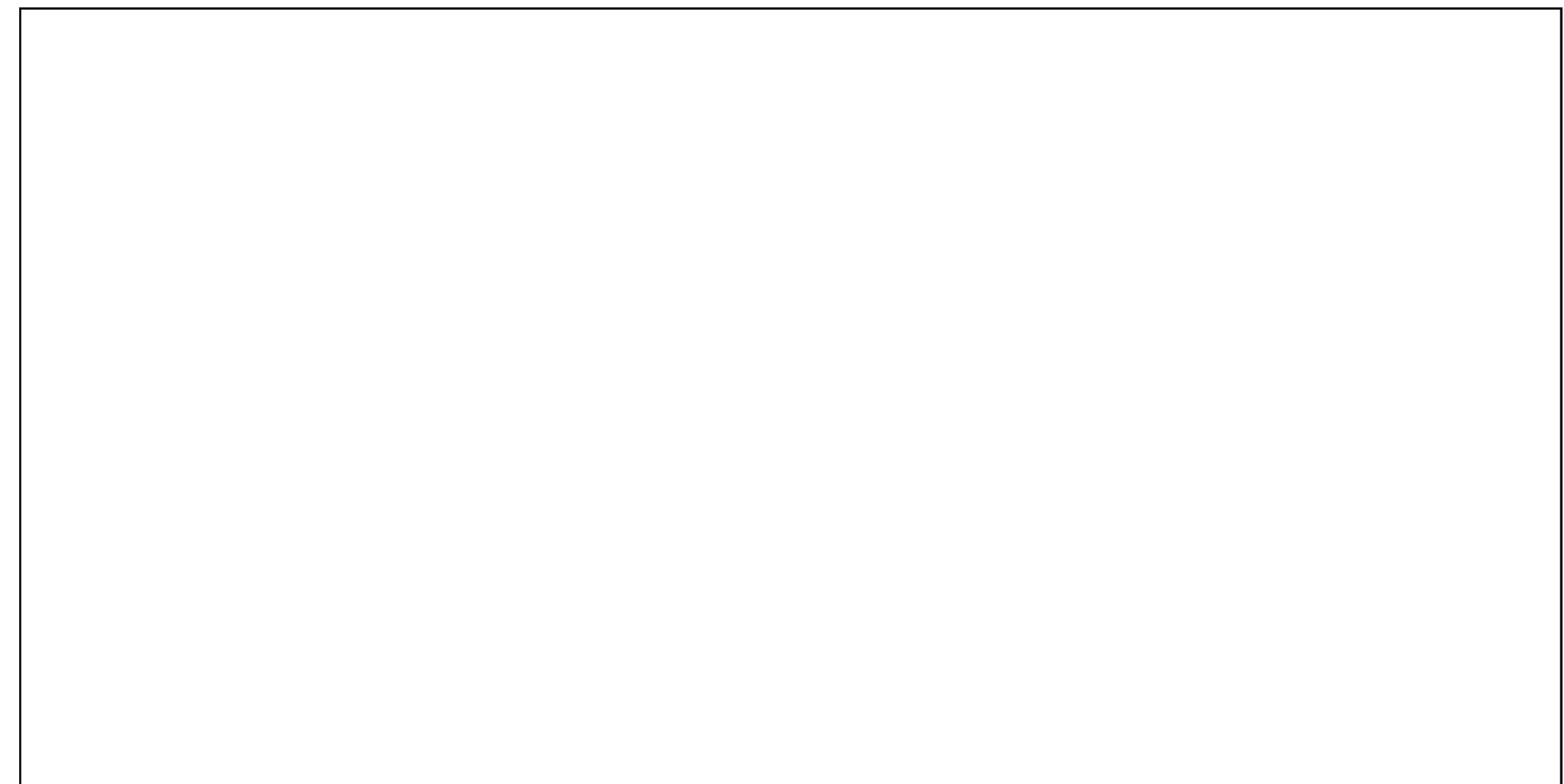
Andrin , Etienne
team : I just want to do ai



Cellular Automata

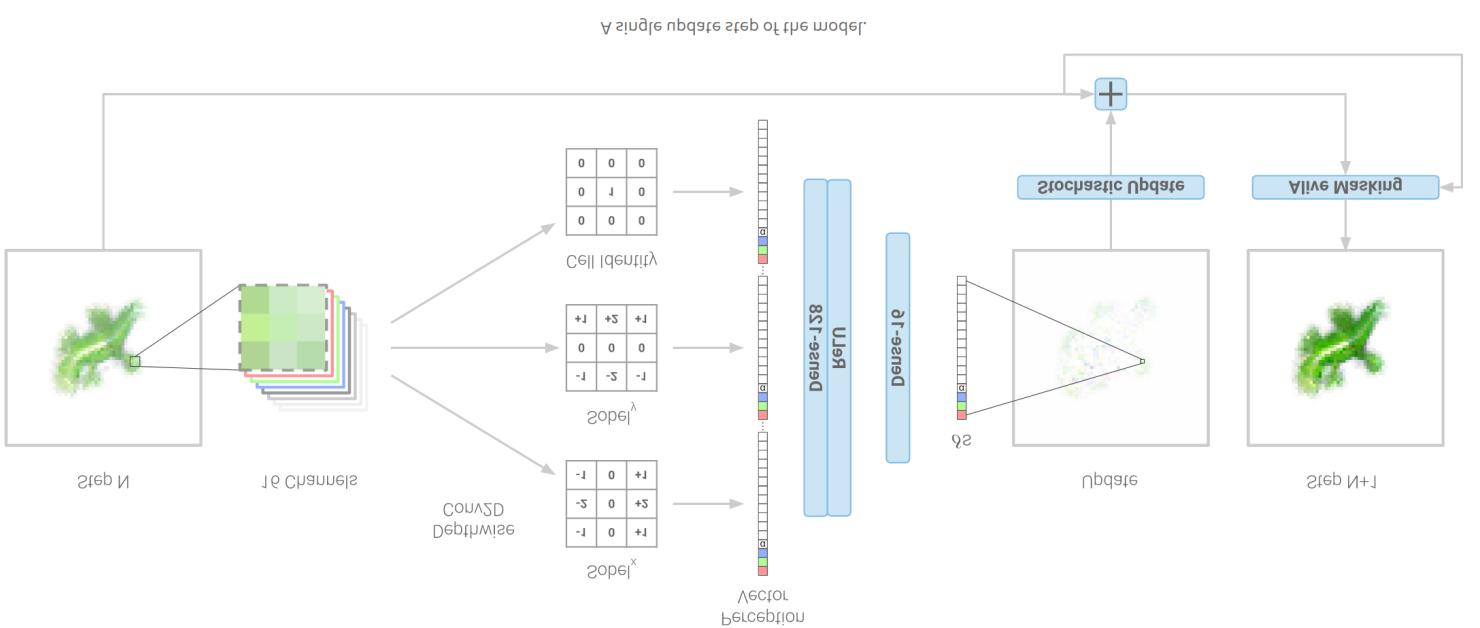
- generate complex behaviour from simple local intercation rules

Conways Game of Life

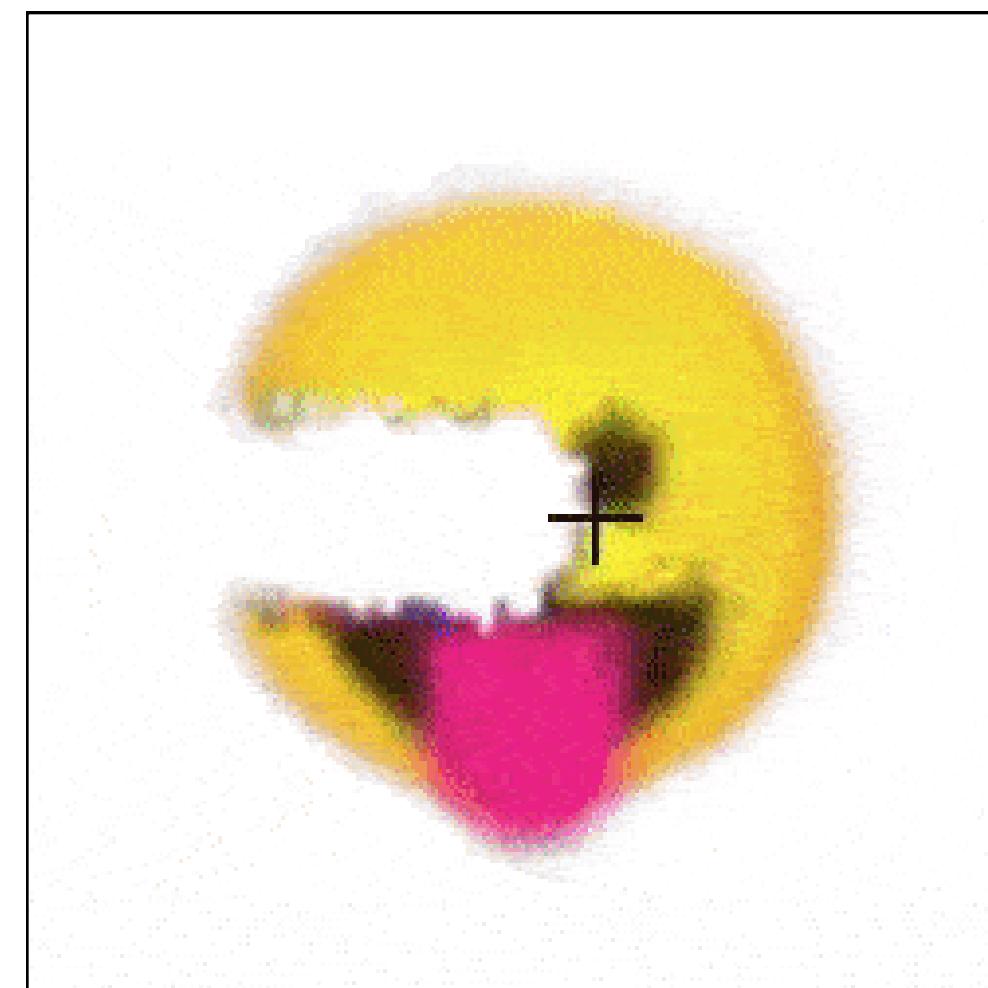


Neural Cellular Automata

- learn the CA rules to grow a target shape given a seed

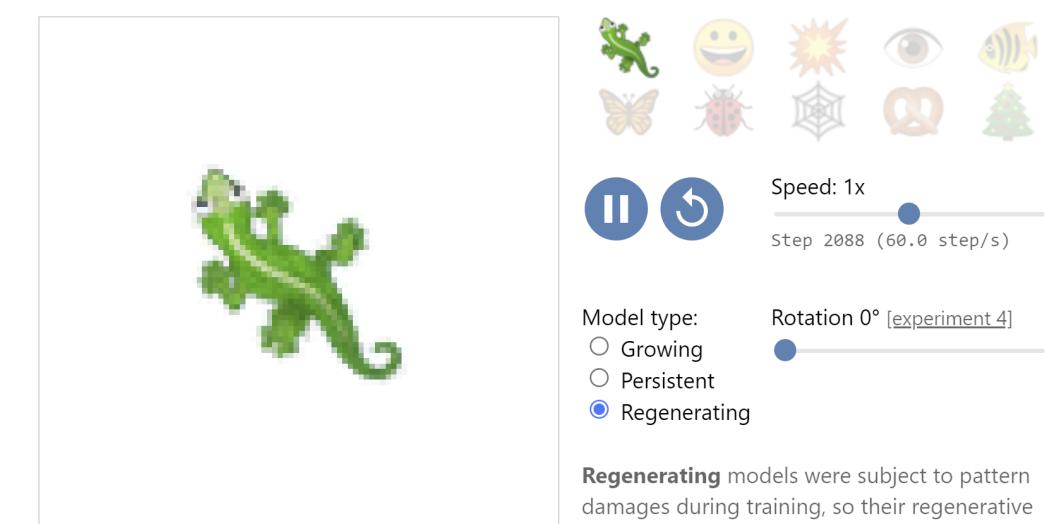


NCA



Growing Neural Cellular Automata

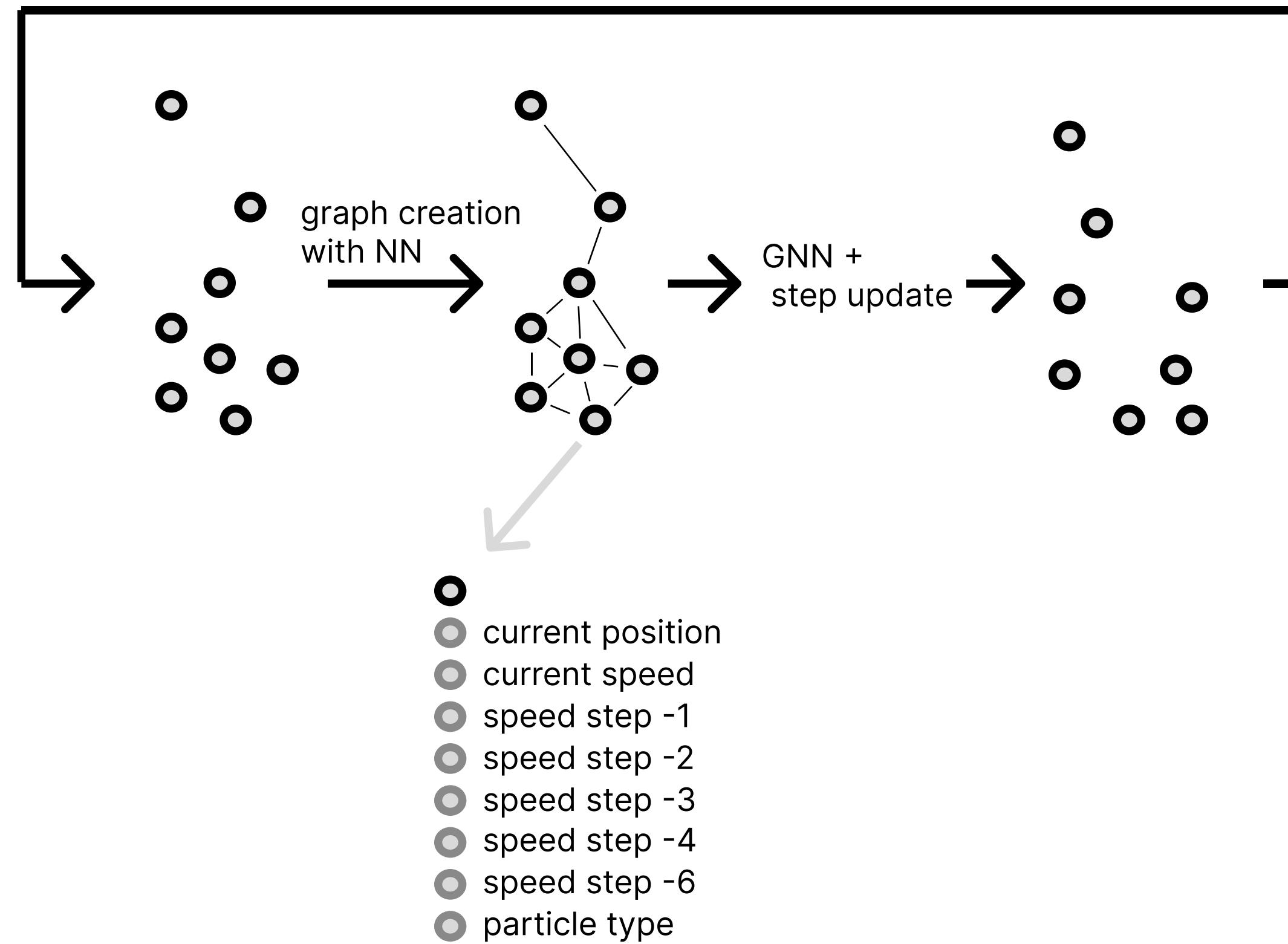
Differentiable Model of Morphogenesis



Click or tap the image to erase the part of the pattern and see it regenerate. Double clicking places a new seed cell on the grid.

AUTHORS
Alexander Mordvintsev
Ettore Randazzo
Eyvind Niklasson
Michael Levin
AFFILIATIONS
Google
Google
Google
Allen Discovery Center at Tufts University
PUBLISHED
Feb. 11, 2020
DOI
10.23915/distill.00023

Learning Message passing on graphs for physics sim.



Learning to Simulate Complex Physics with Graph Networks

Alvaro Sanchez-Gonzalez ^{*1} Jonathan Godwin ^{*1} Tobias Pfaff ^{*1} Rex Ying ^{*1,2} Jure Leskovec ²
Peter W. Battaglia ¹

Abstract

Here we present a machine learning framework and model implementation that can learn to simulate a wide variety of challenging physical domains, involving fluids, rigid solids, and deformable materials interacting with one another. Our framework—which we term “Graph Network-based Simulators” (GNS)—represents the state of a physical system with particles, expressed as nodes in a graph, and computes dynamics via learned message-passing. Our results show that our model can generalize from single-timestep predictions with thousands of particles during training, to different initial conditions, thousands of timesteps, and at least an order of magnitude more particles at test time. Our model was robust to hyperparameter choices across various evaluation metrics: the main determinants of long-term performance were the number of message-passing steps, and mitigating the accumulation of error by corrupting the training data with noise. Our GNS framework

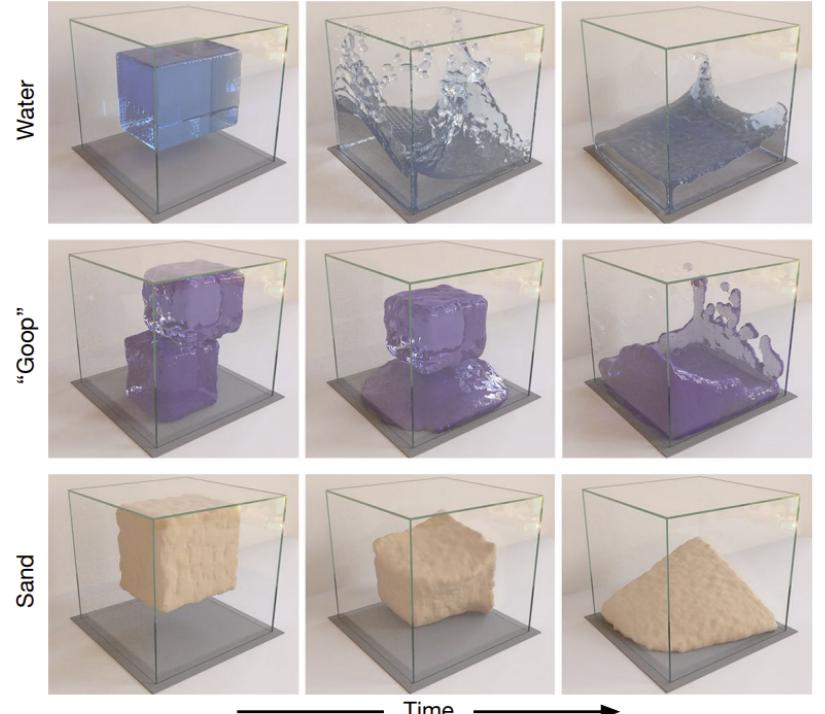
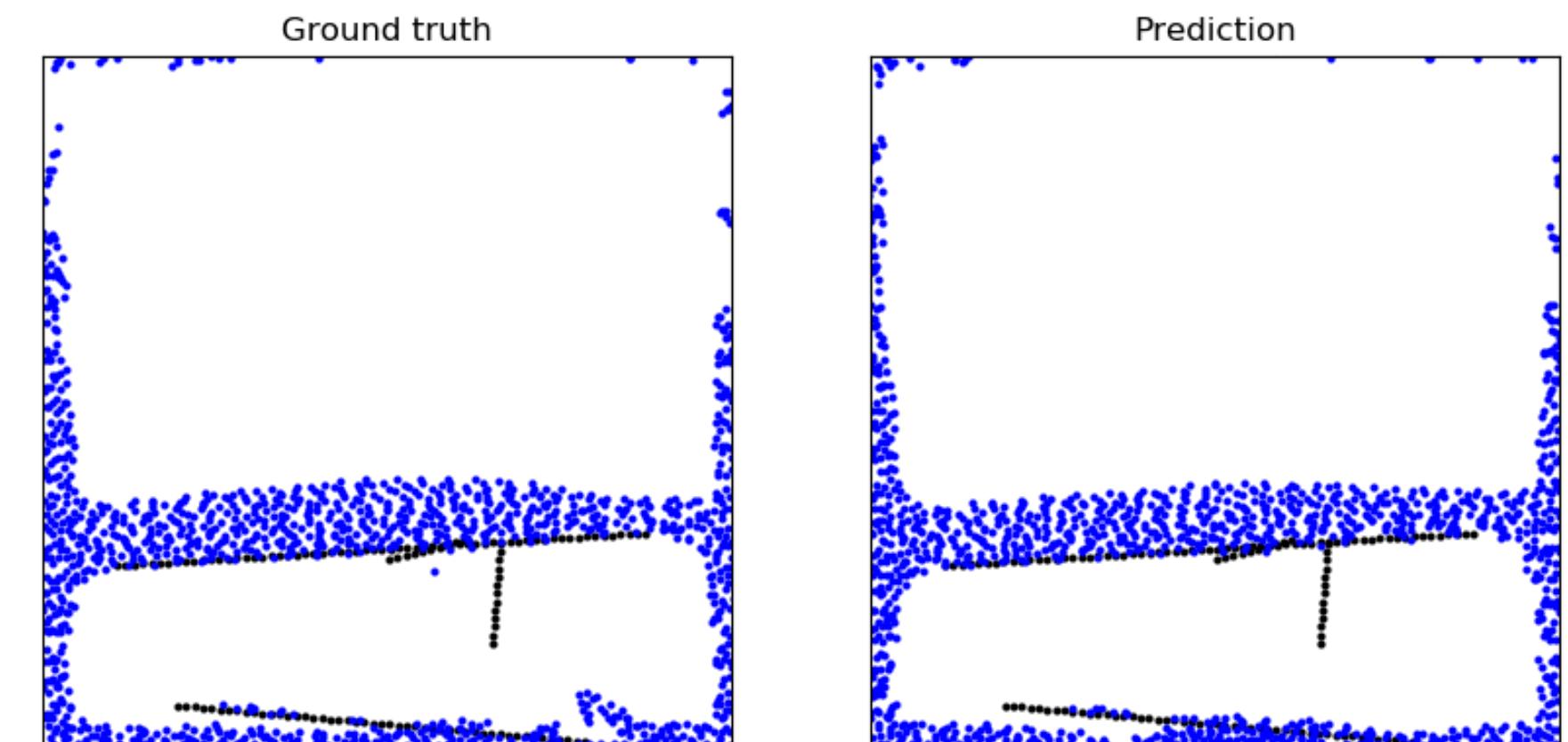
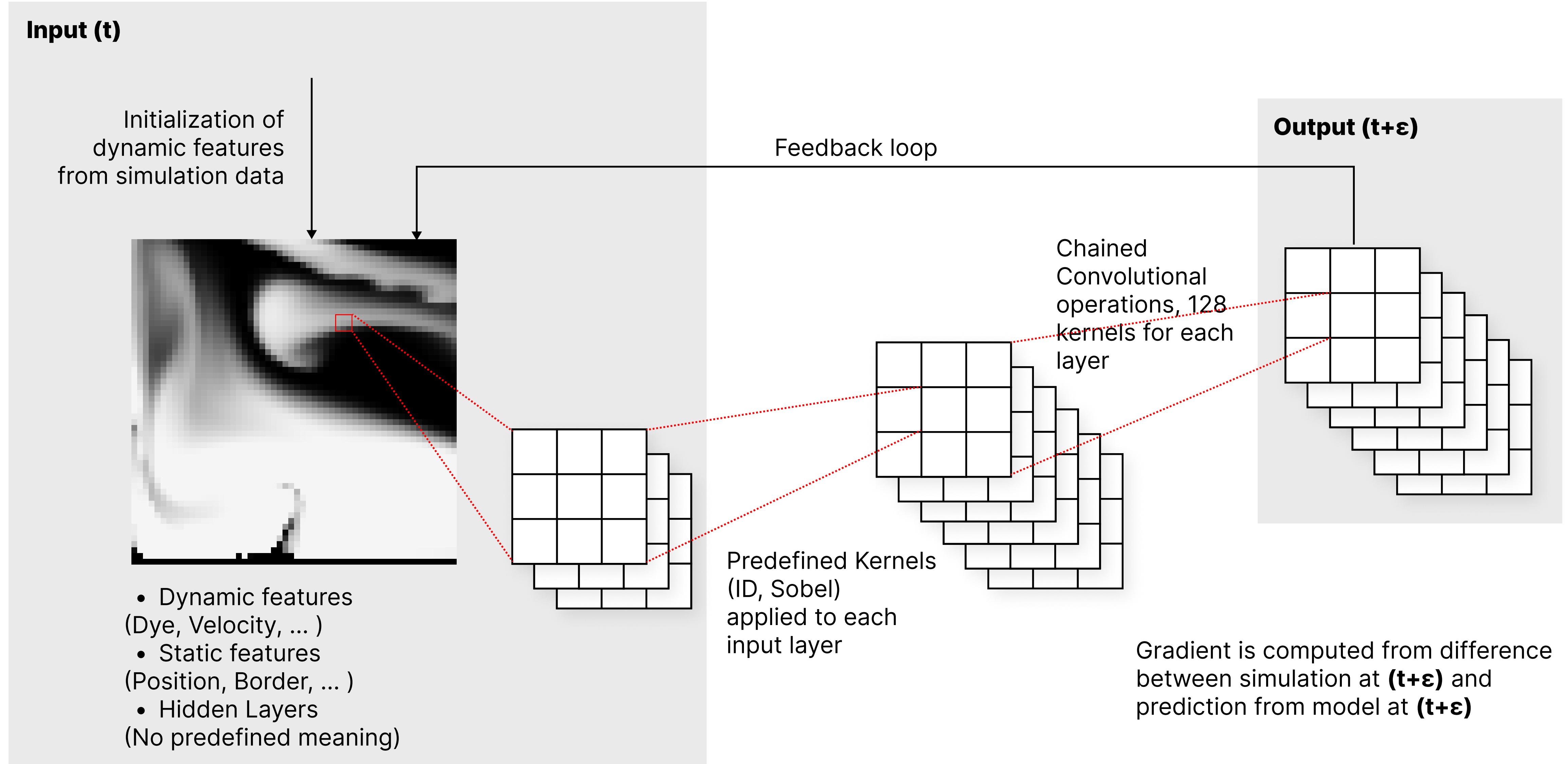


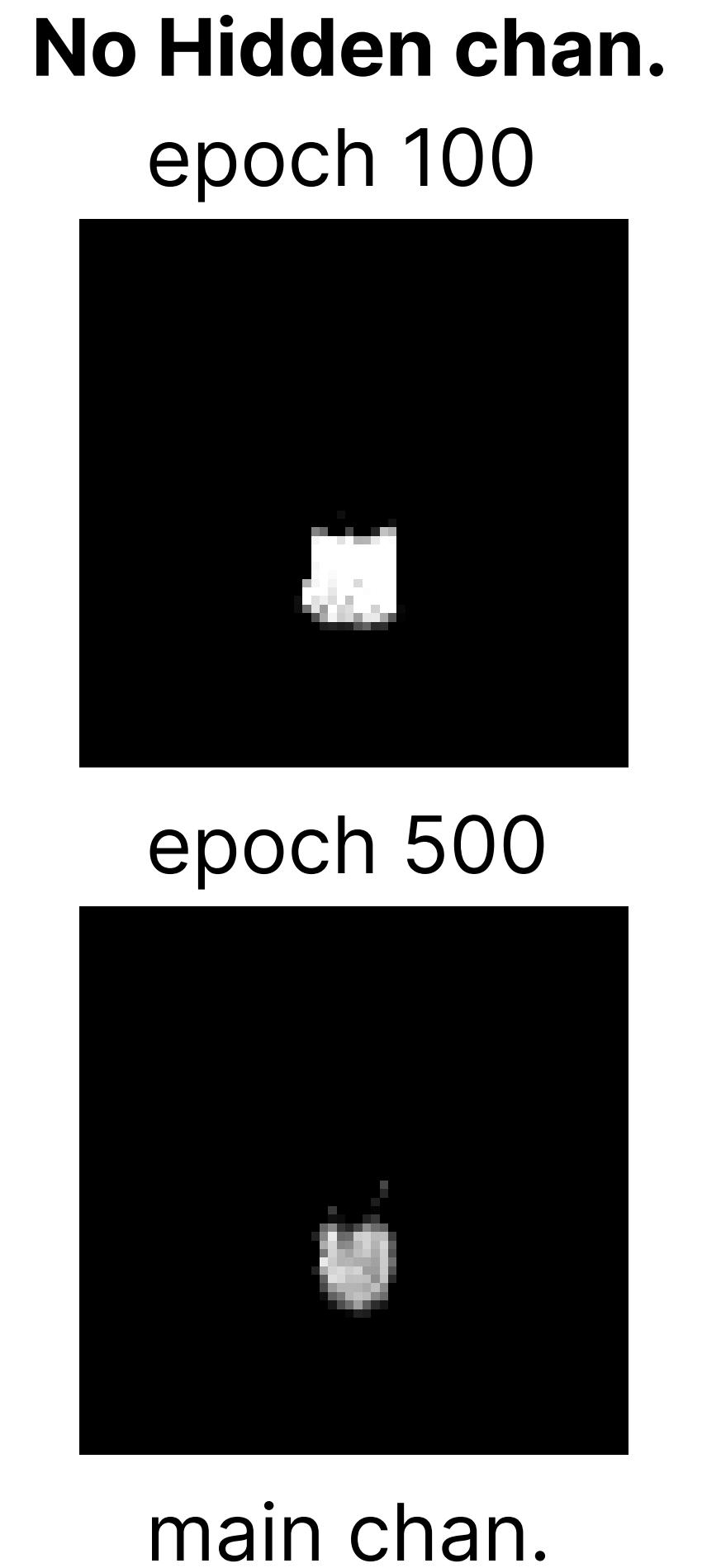
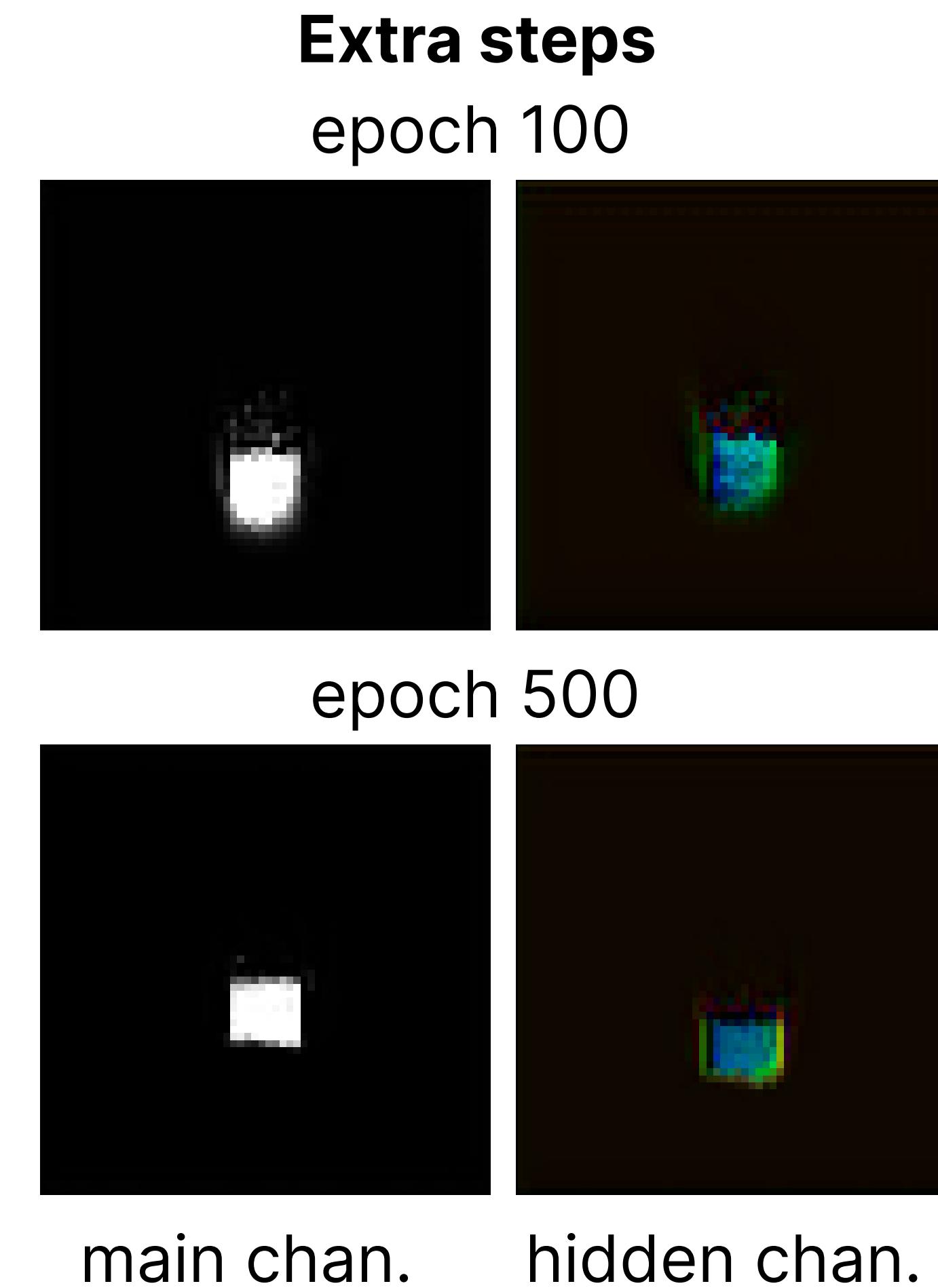
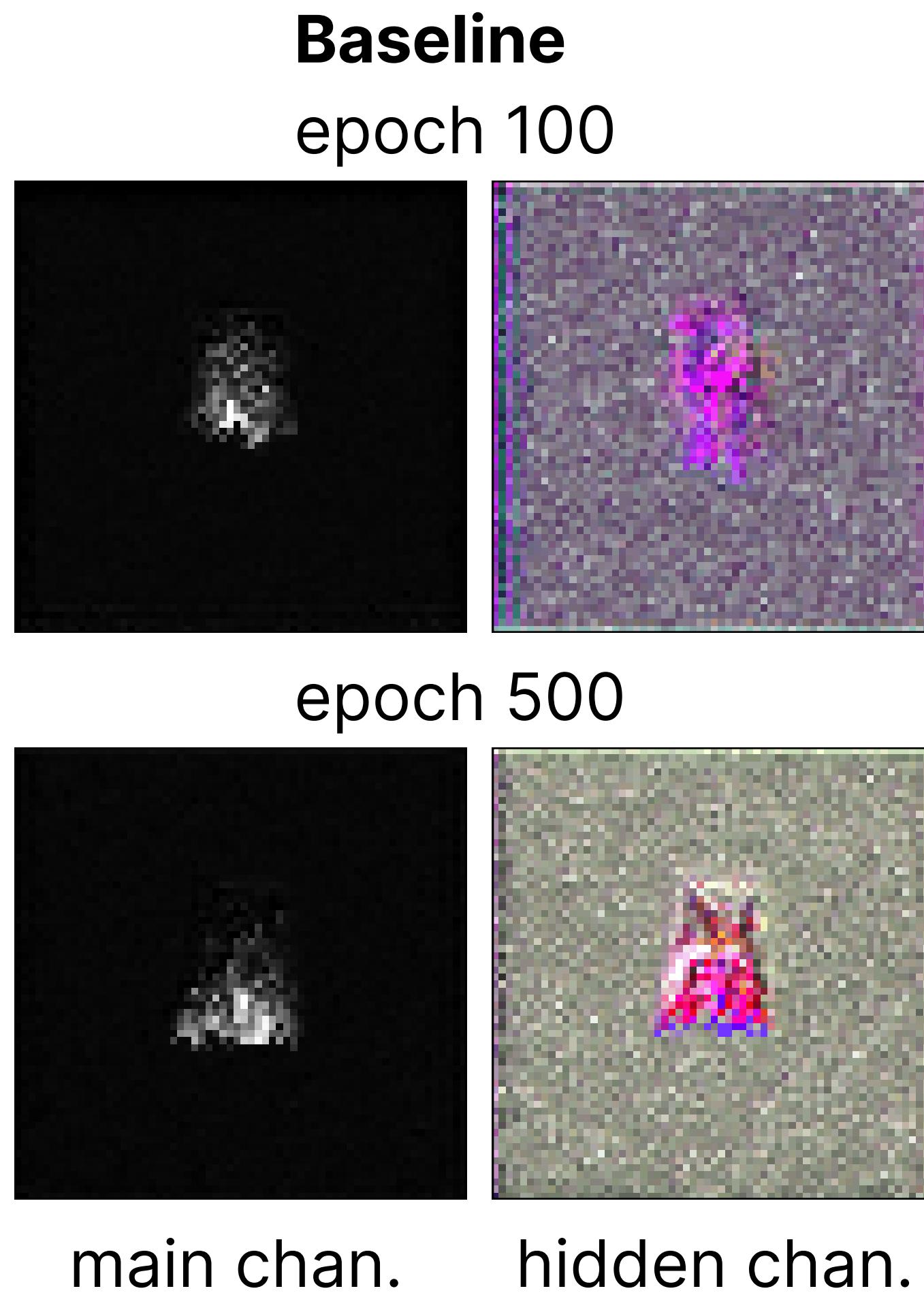
Figure 1. Rollouts of our GNS model for our WATER-3D, GOOP-3D and SAND-3D datasets. It learns to simulate rich materials at resolutions sufficient for high-quality rendering [video].



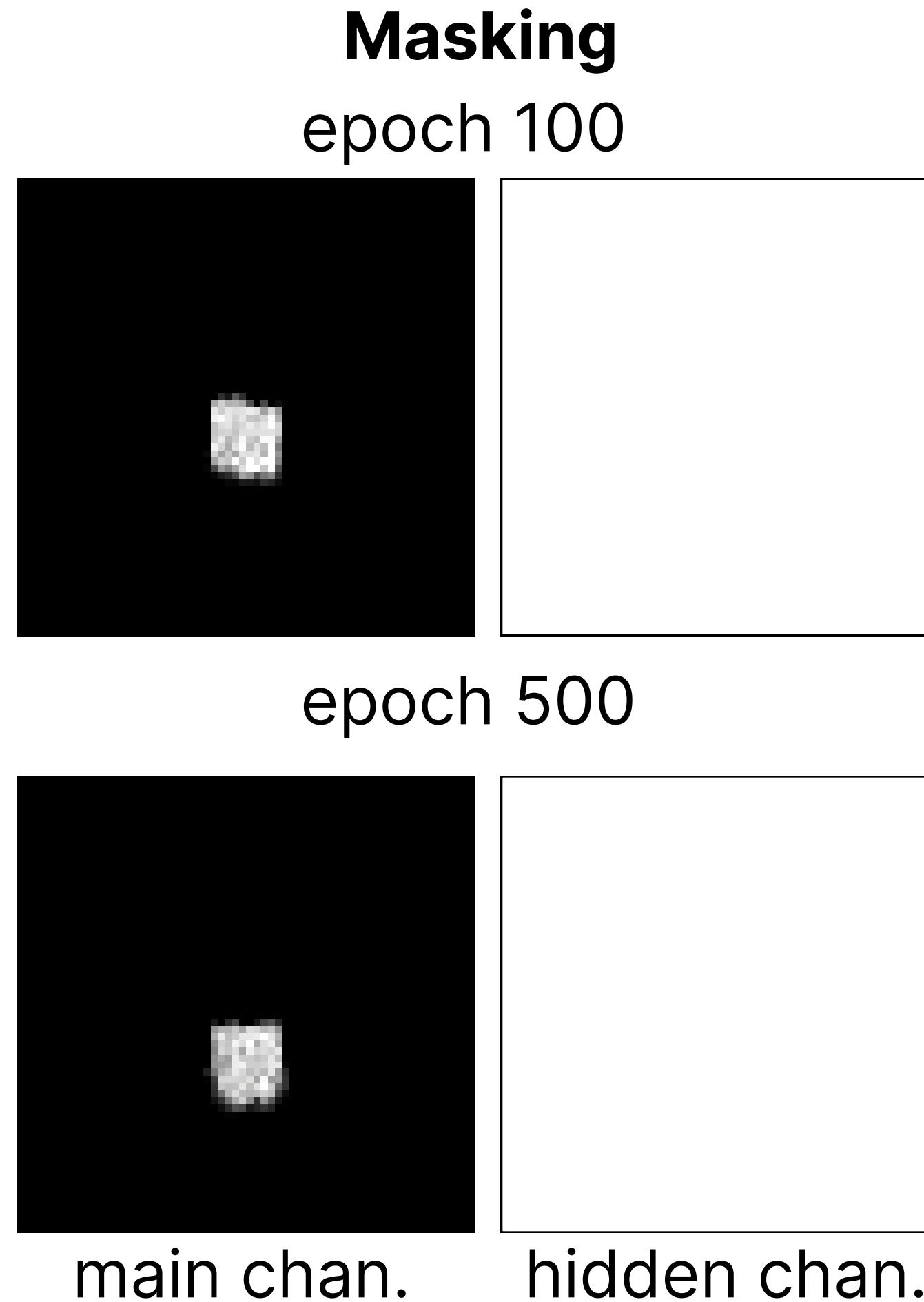
Our Model



1st scenario, falling square



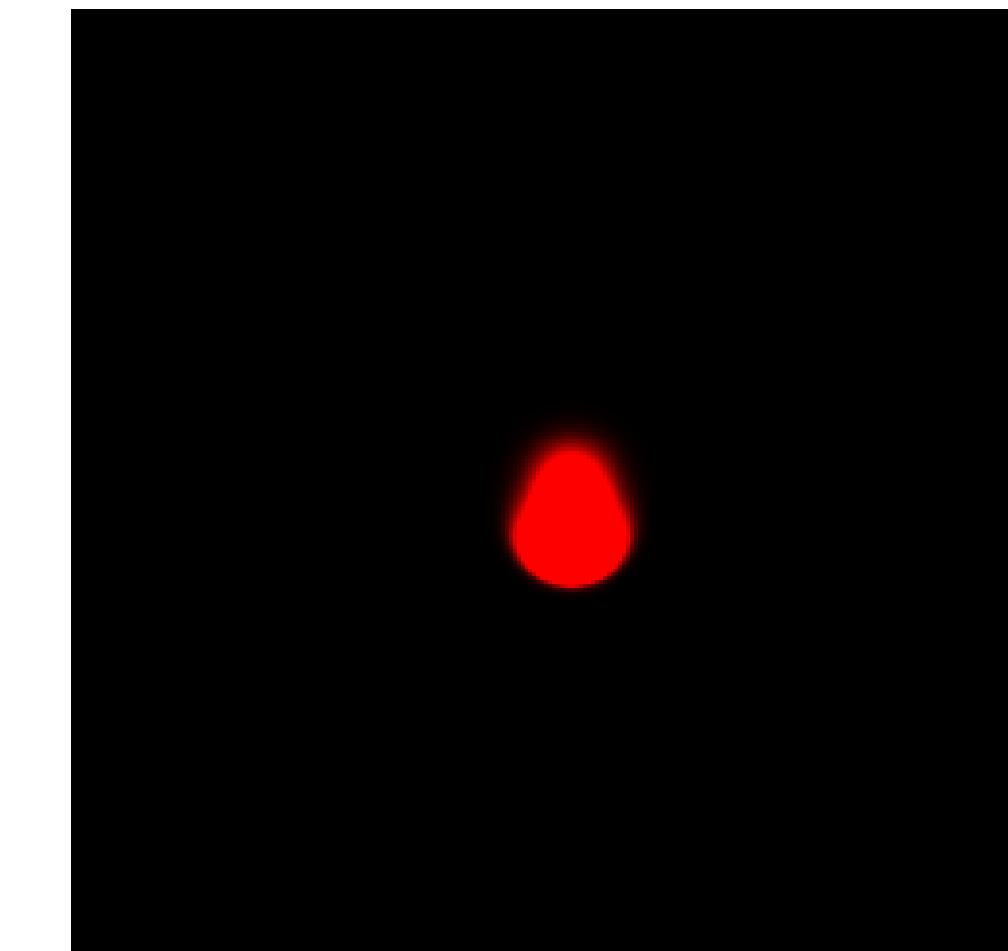
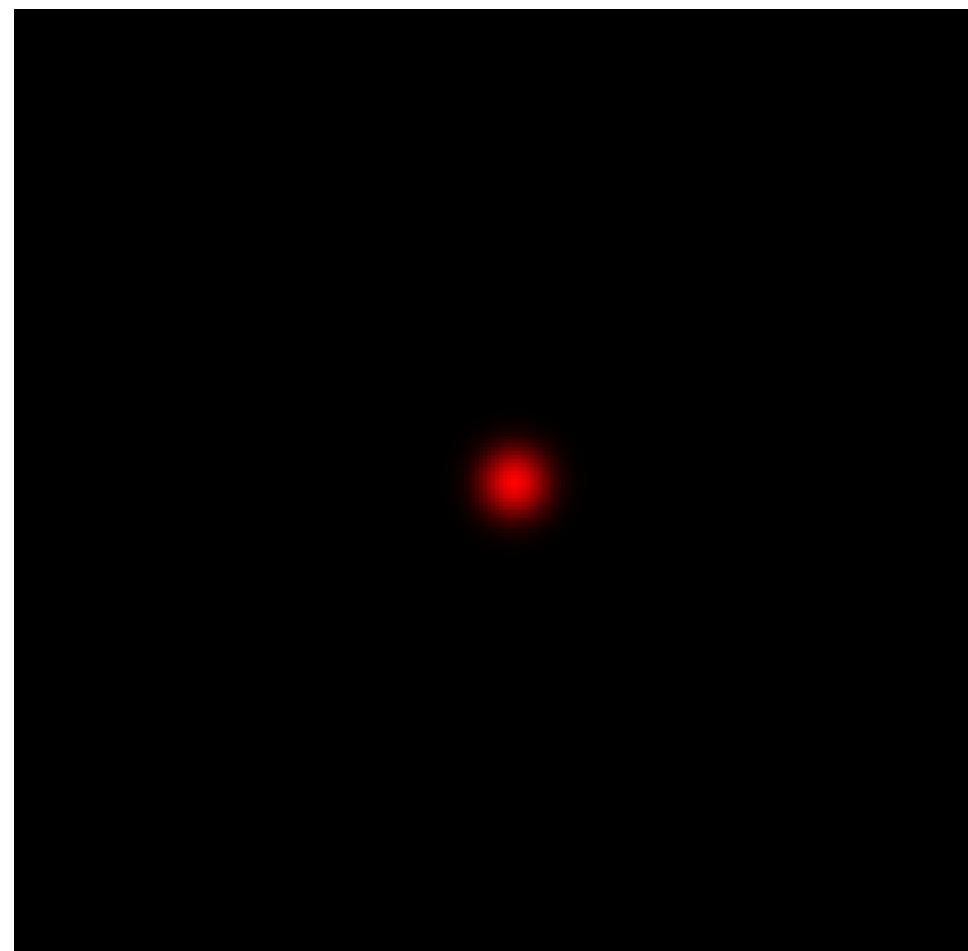
1st scenario, falling square



- only update cells that were alive in previous step or that pass a threshold on the NCA updated step
- this adaptive update is done through masking

2nd scenario, fluid sim

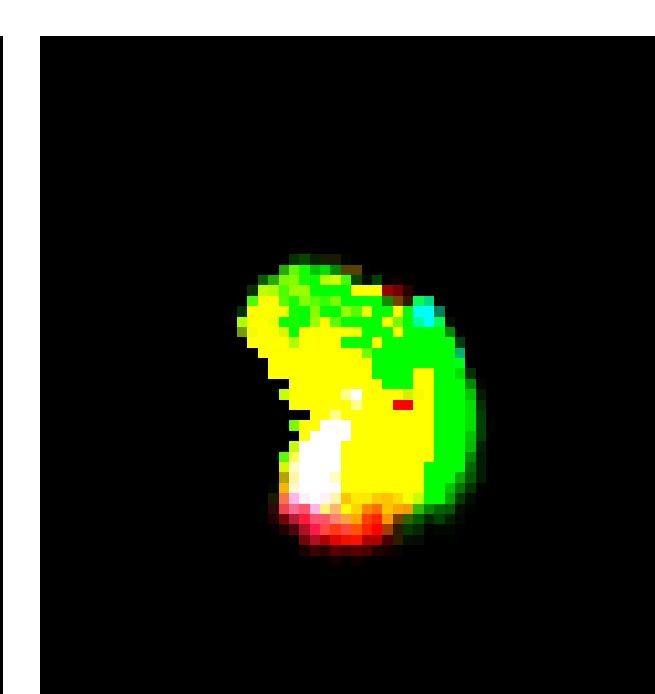
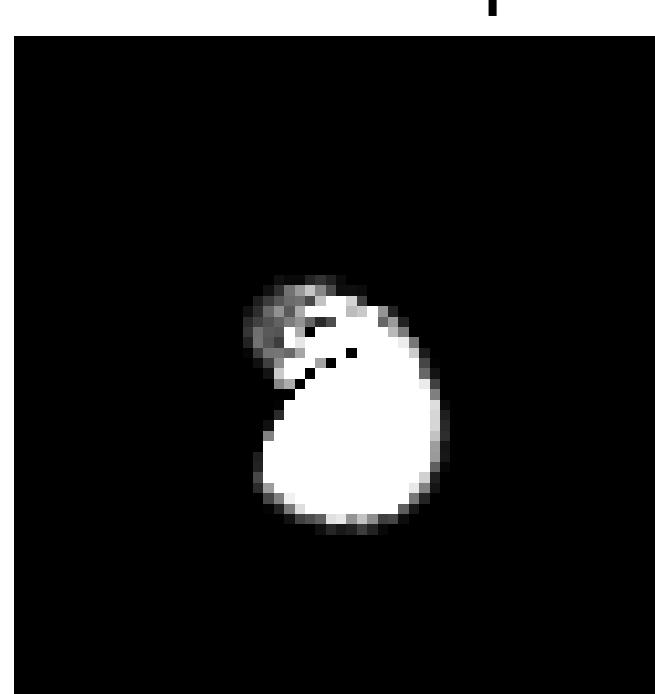
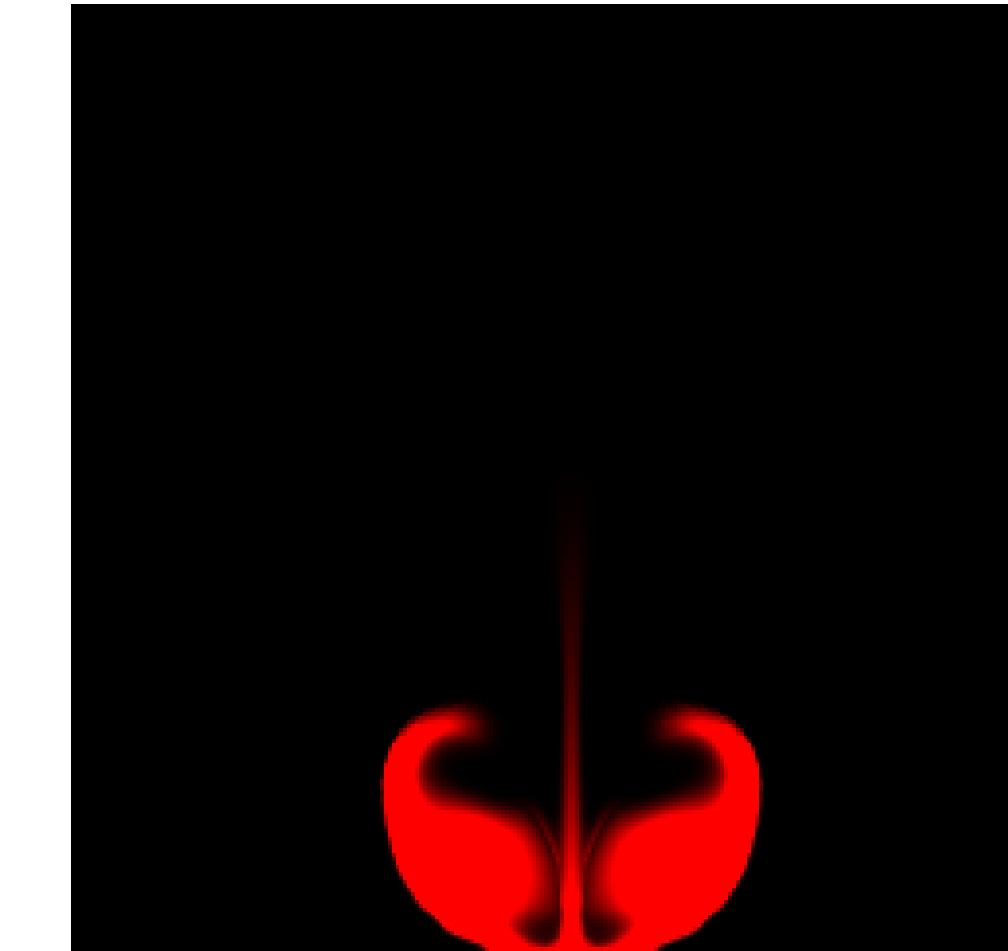
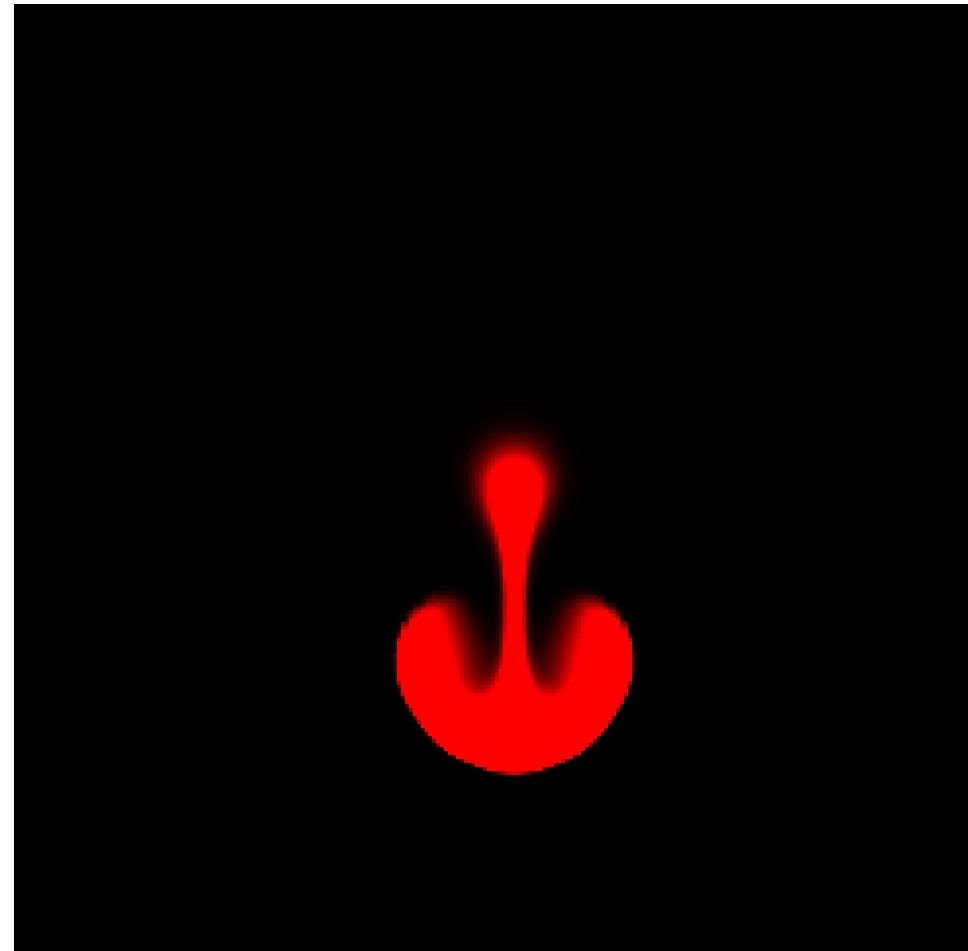
Ground truth simulation



Masking
epoch 100



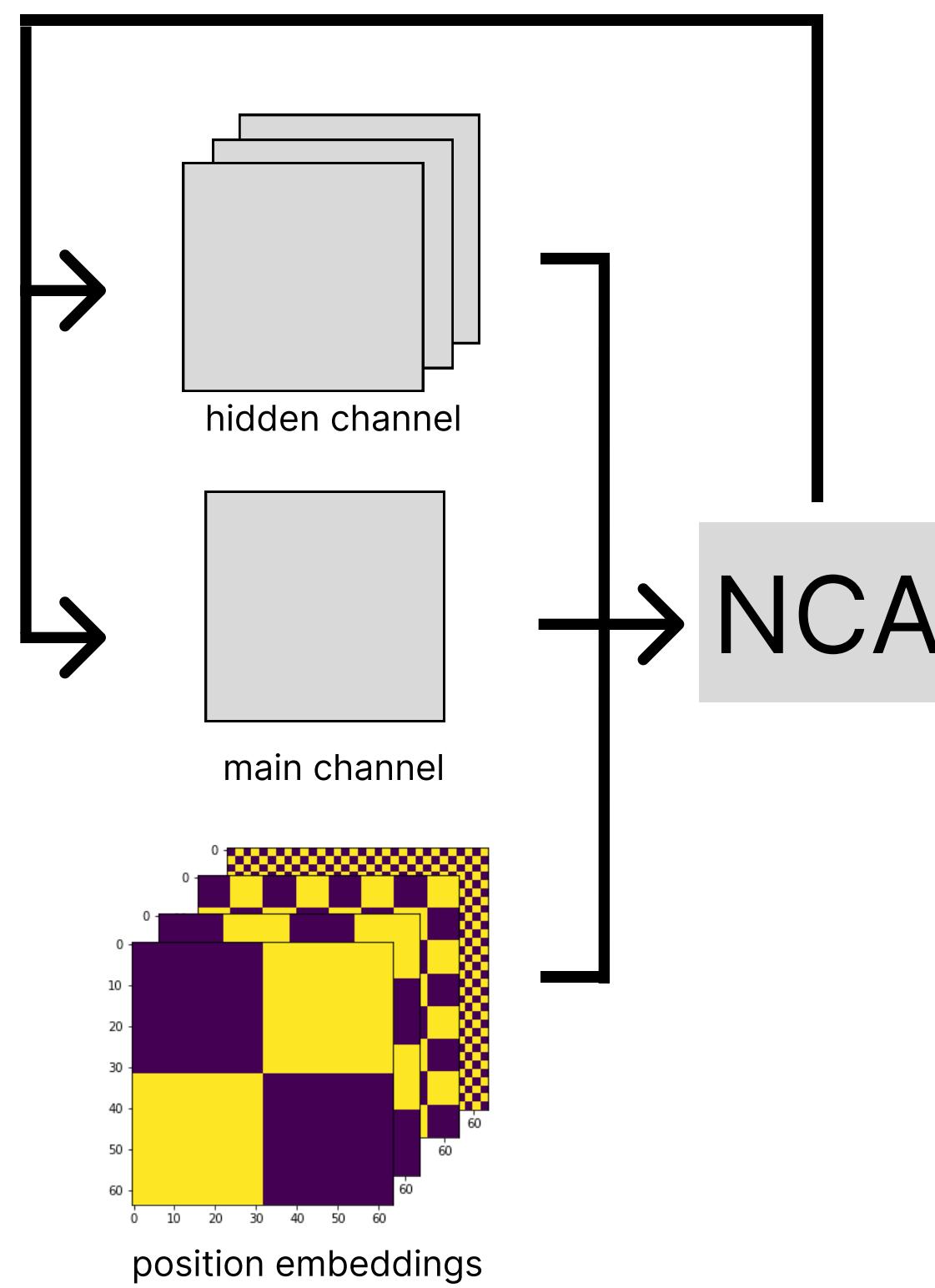
epoch 500



main chan.

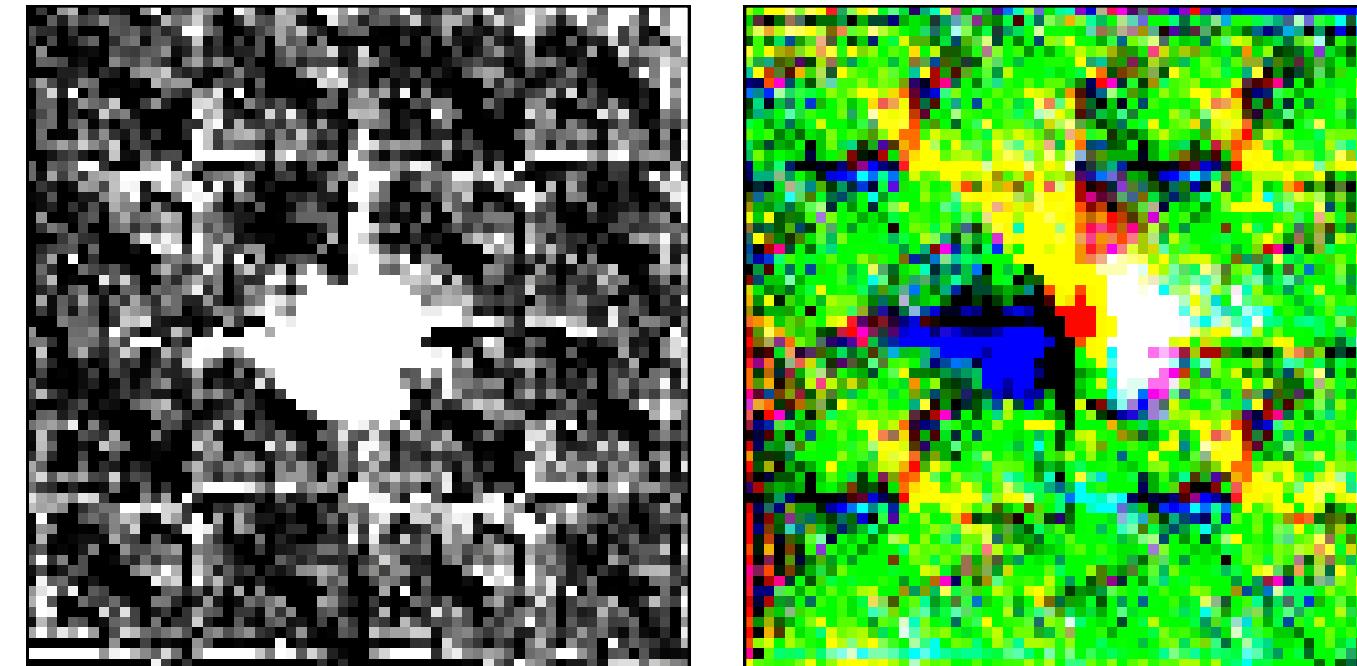
hidden chan.

2nd scenario, fluid sim

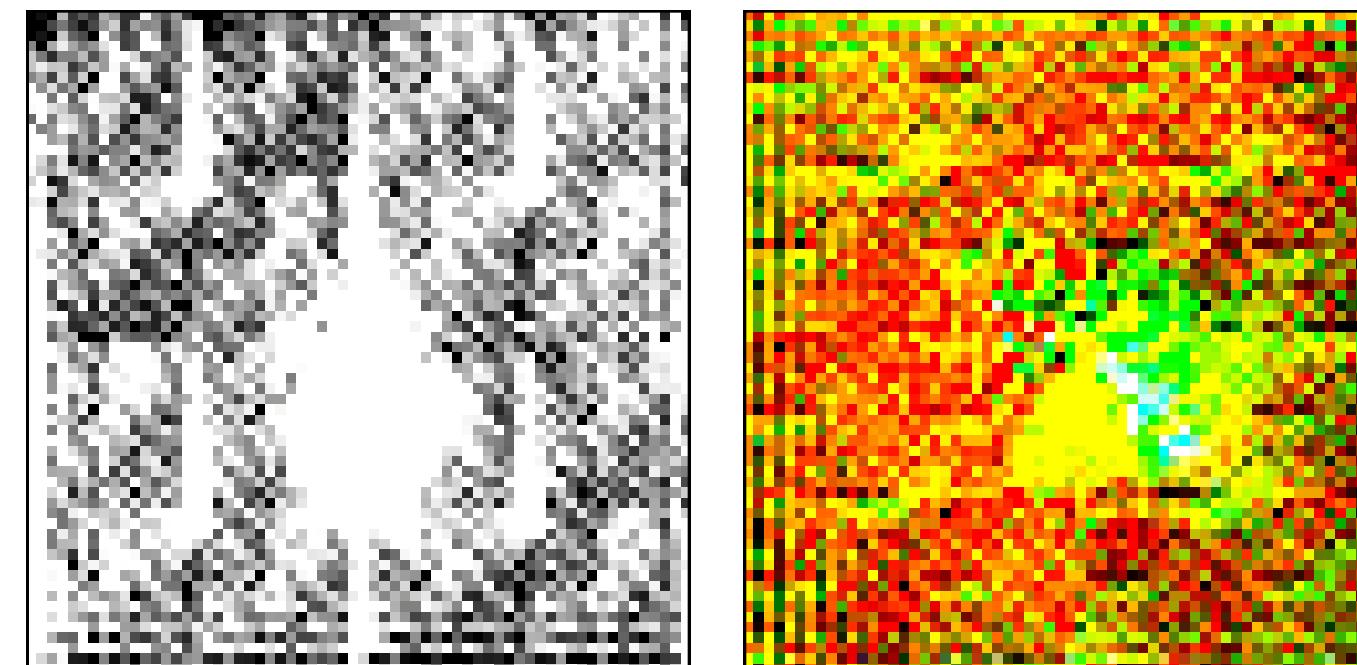


Position Encoding

epoch 100



epoch 500

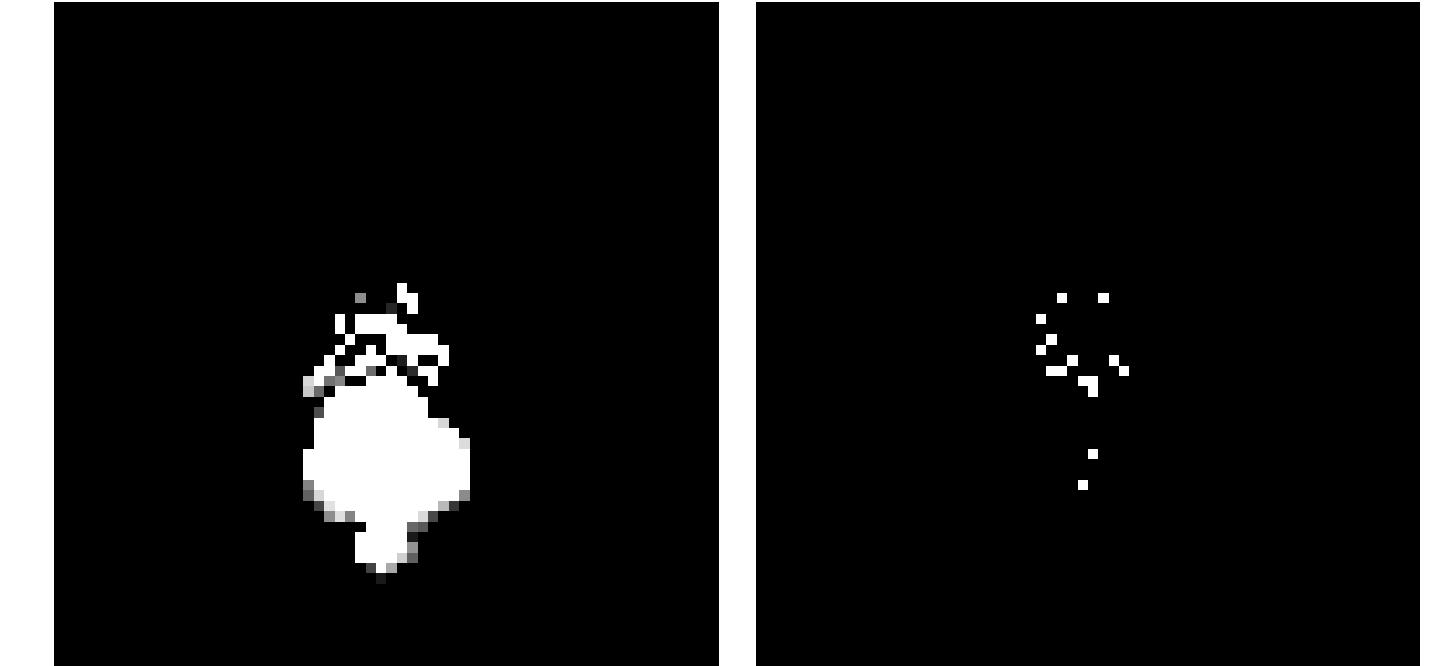


main chan.

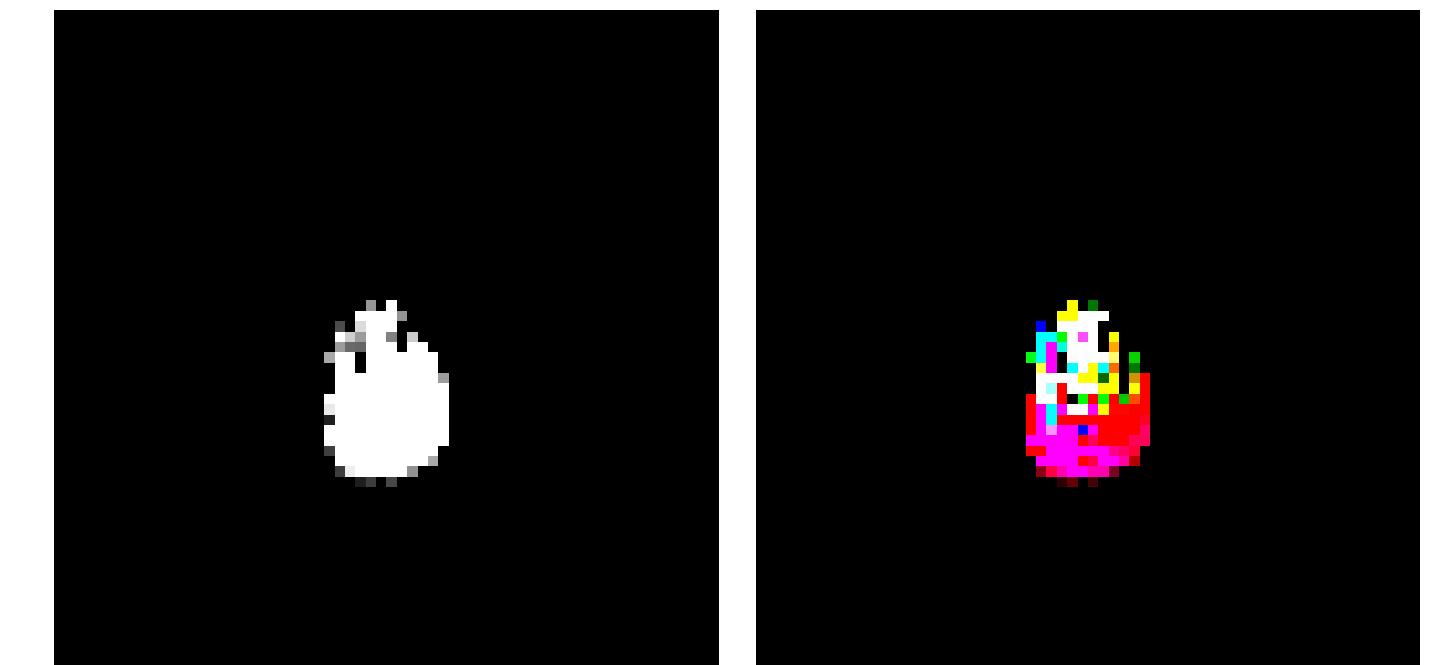
hidden chan.

**Position Encoding
Masking**

epoch 100



epoch 500

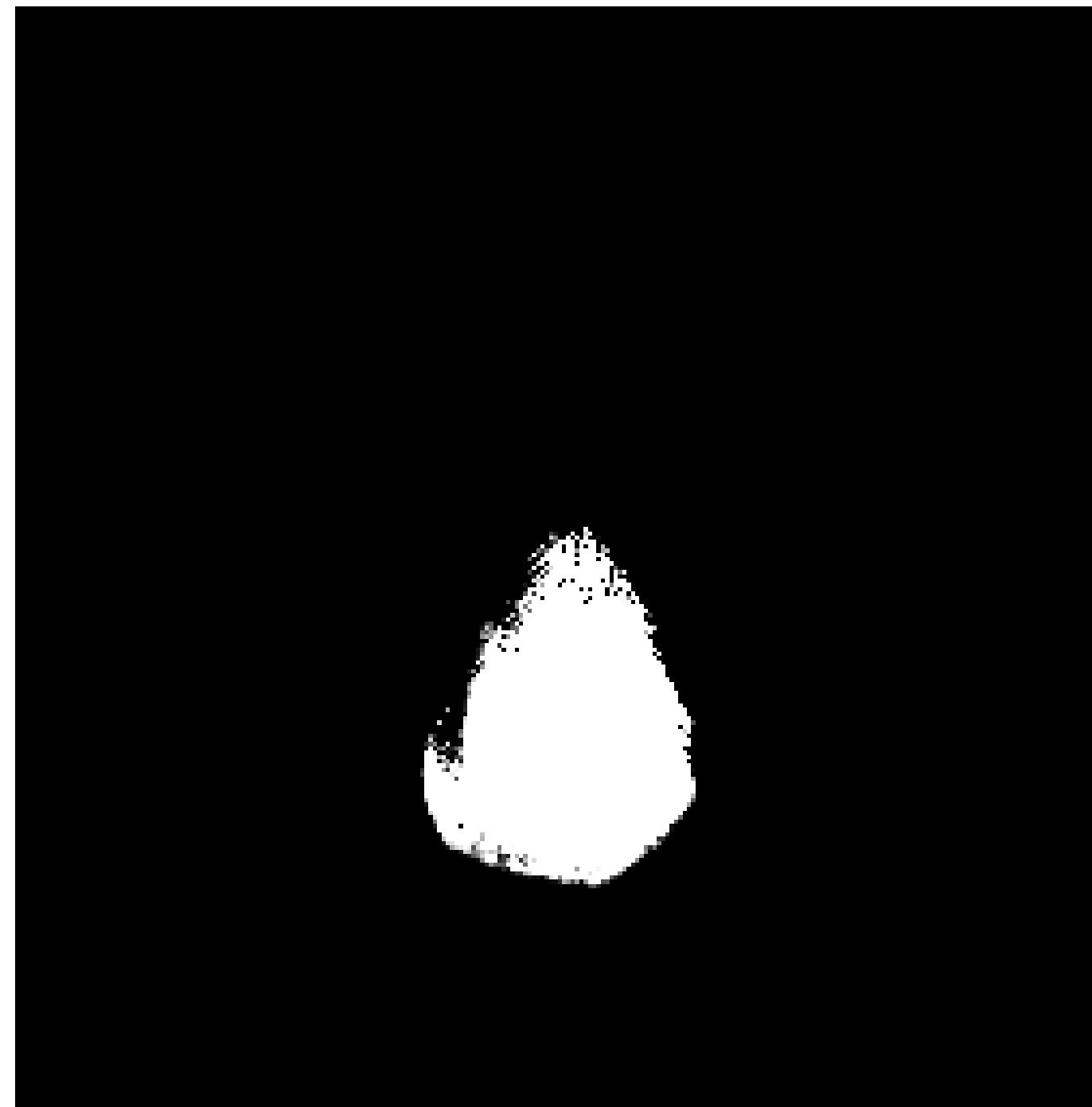


main chan.

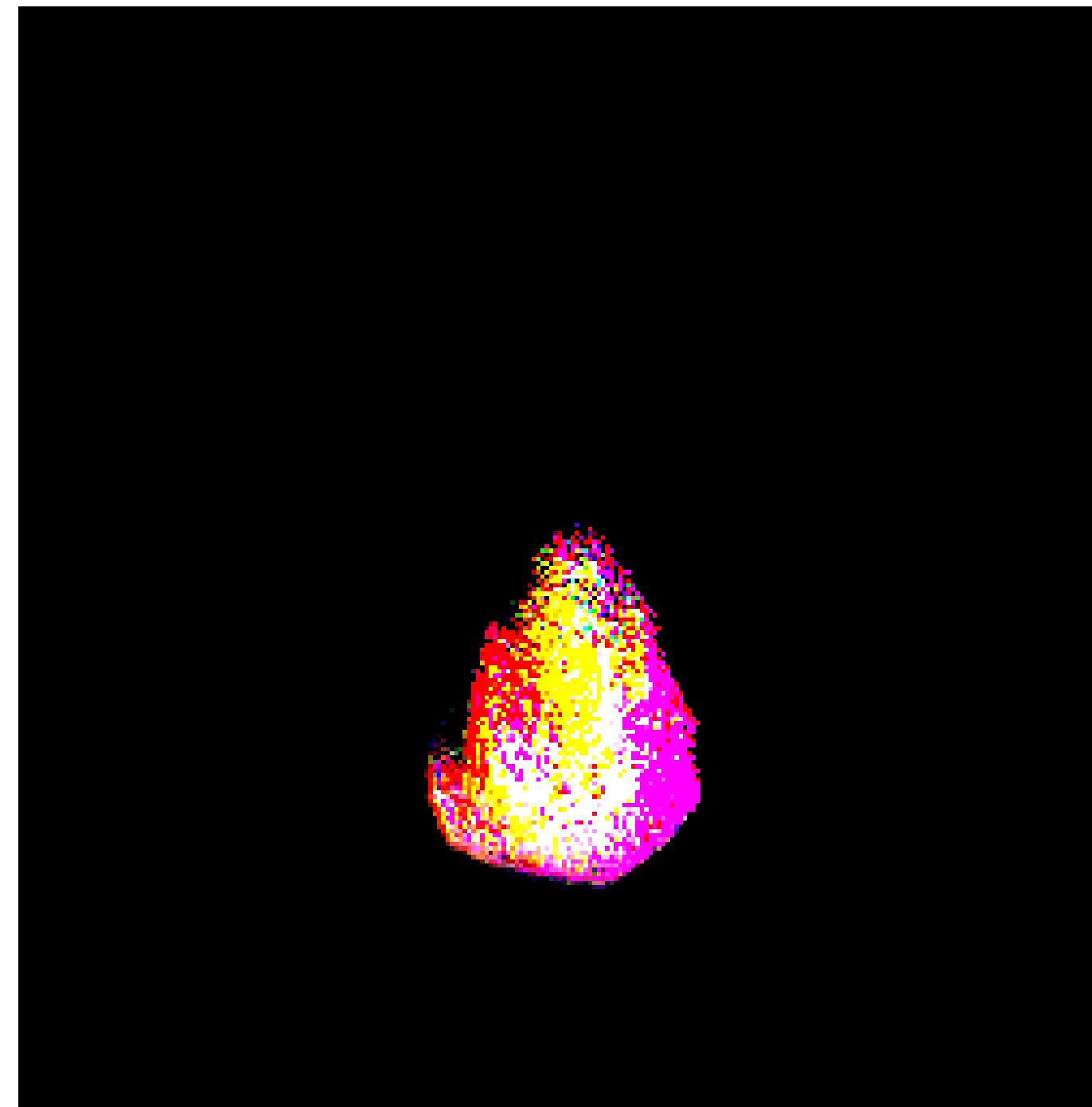
hidden chan.

2nd scenario, fluid sim

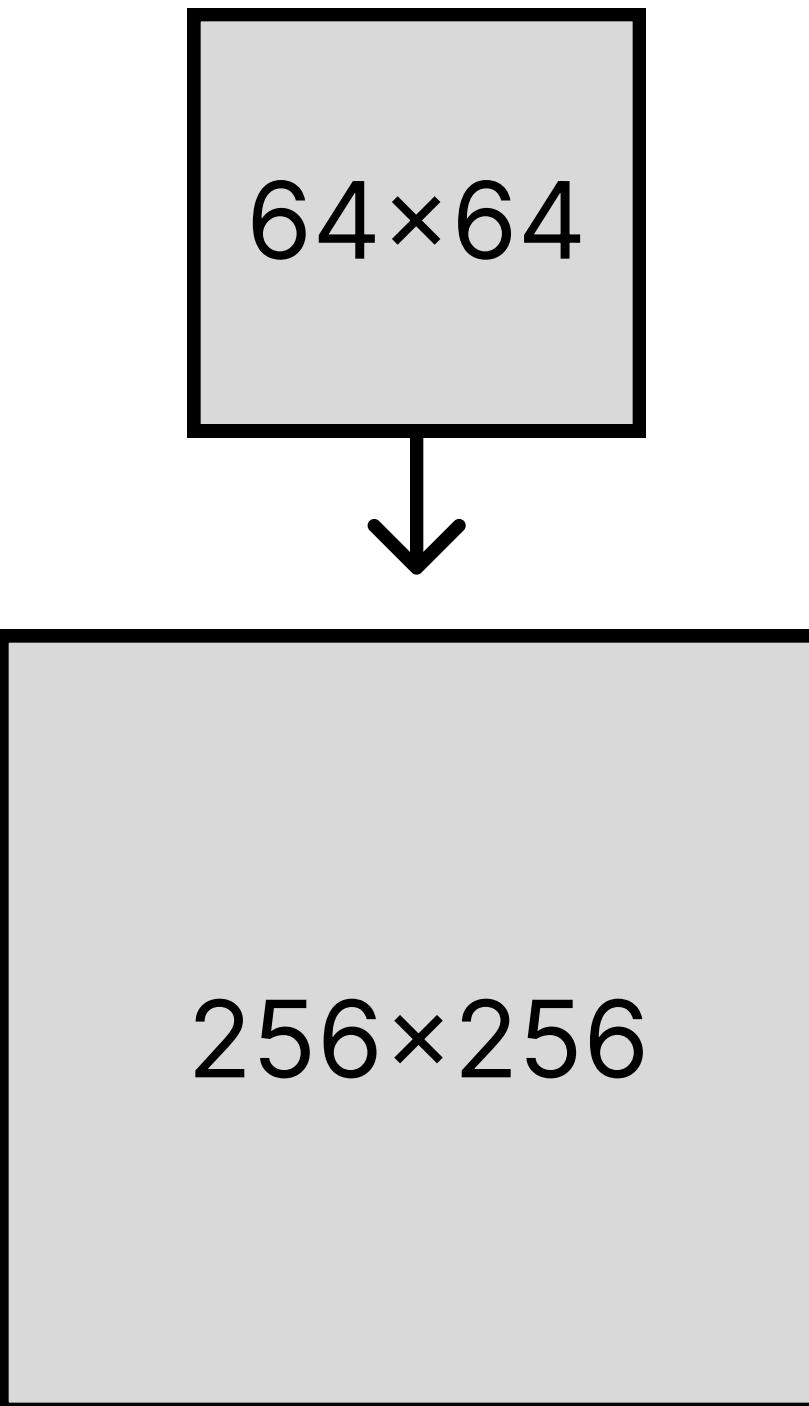
Does the CA scale with domain size ?



main chan.

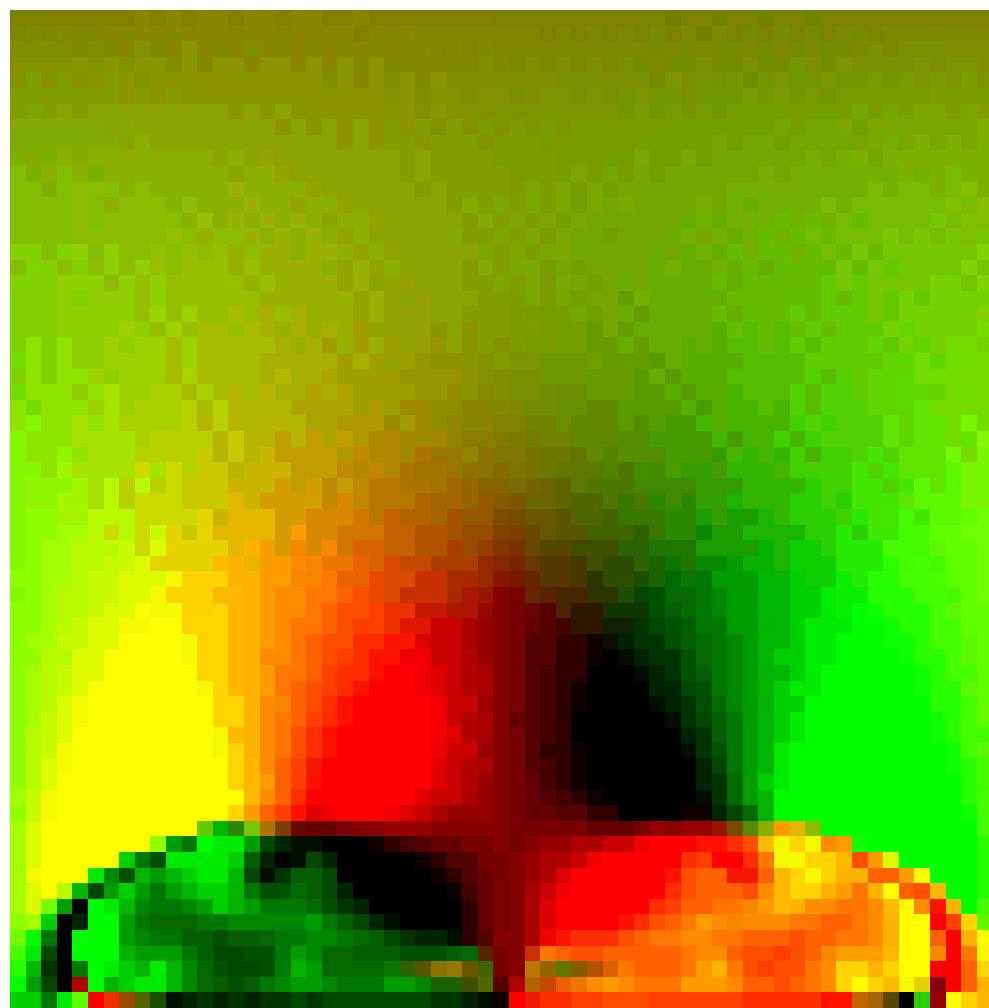


hidden chan.

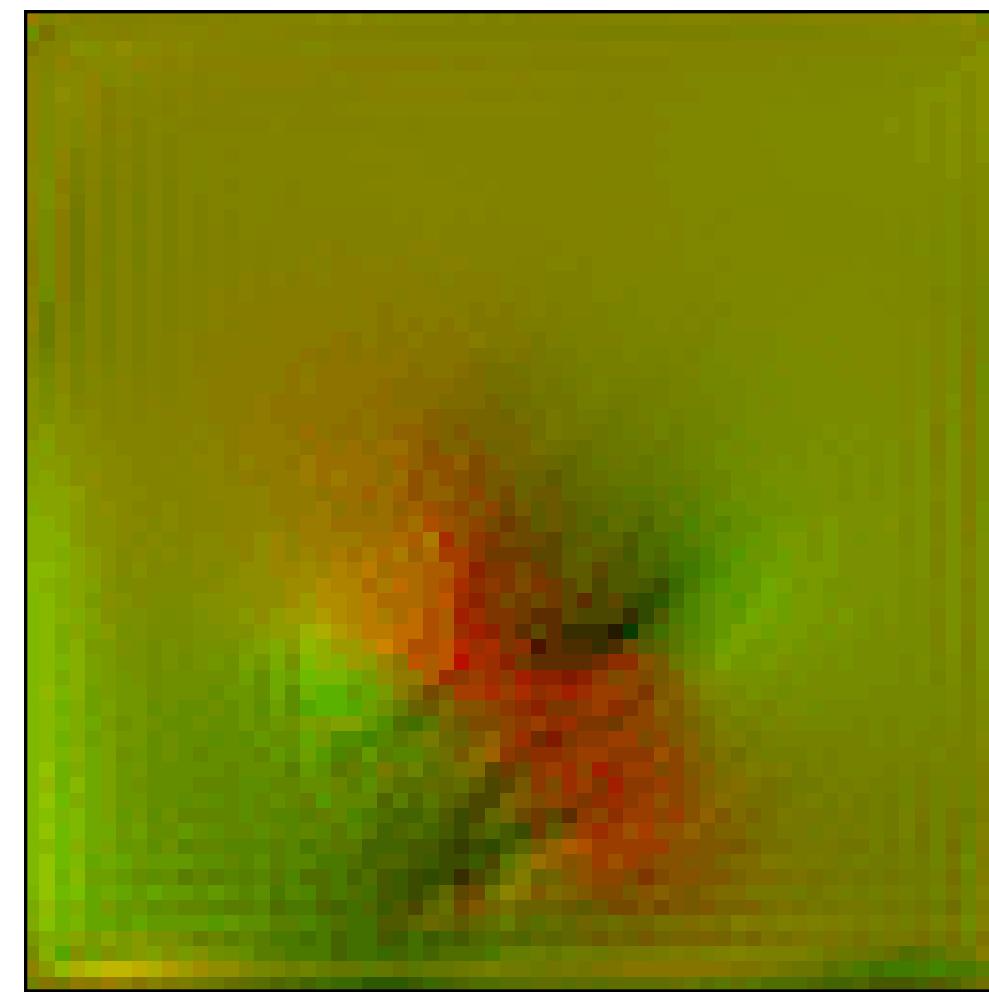


Many attempts failed

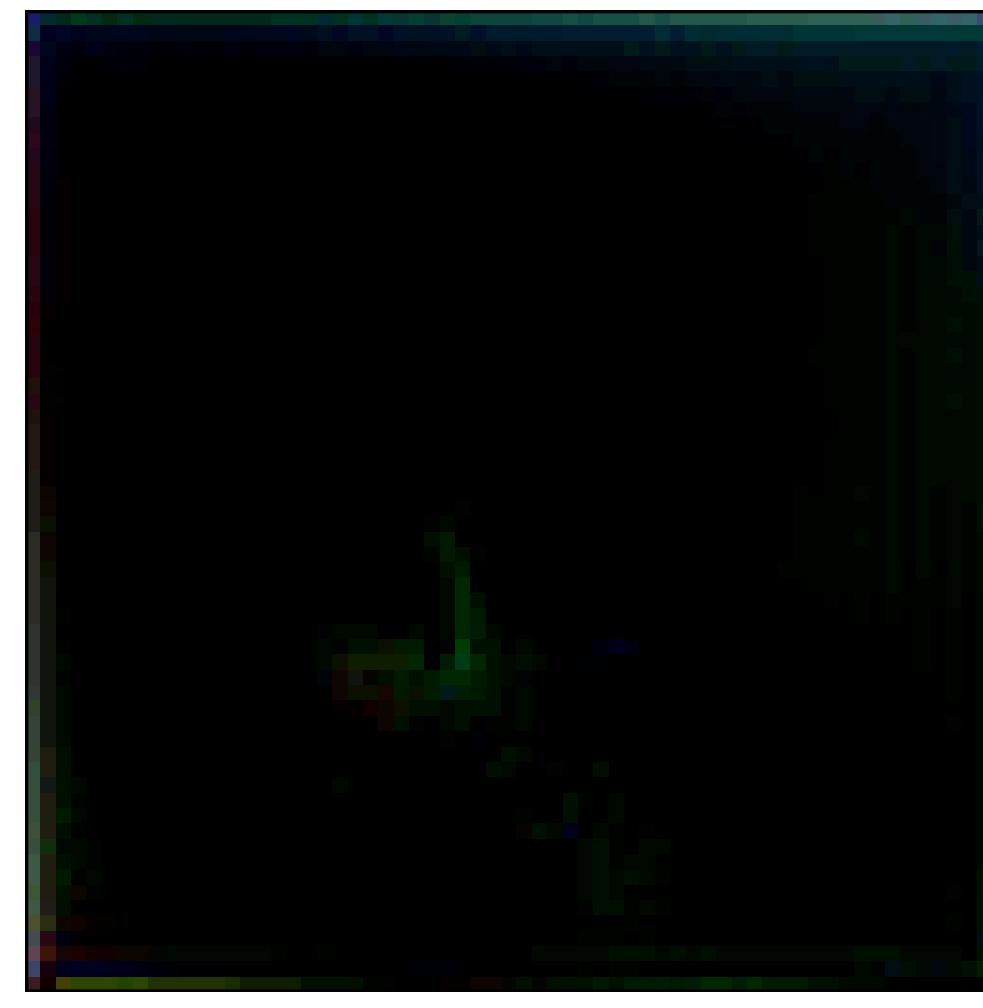
- using vel as feature



target

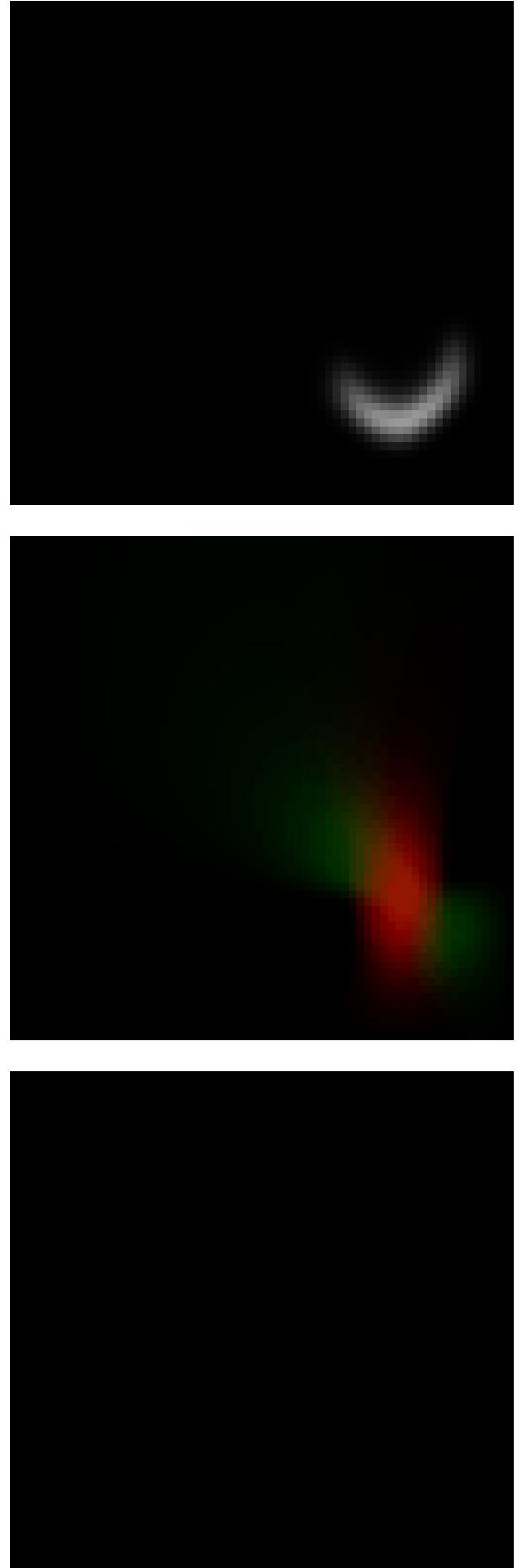


main chan.

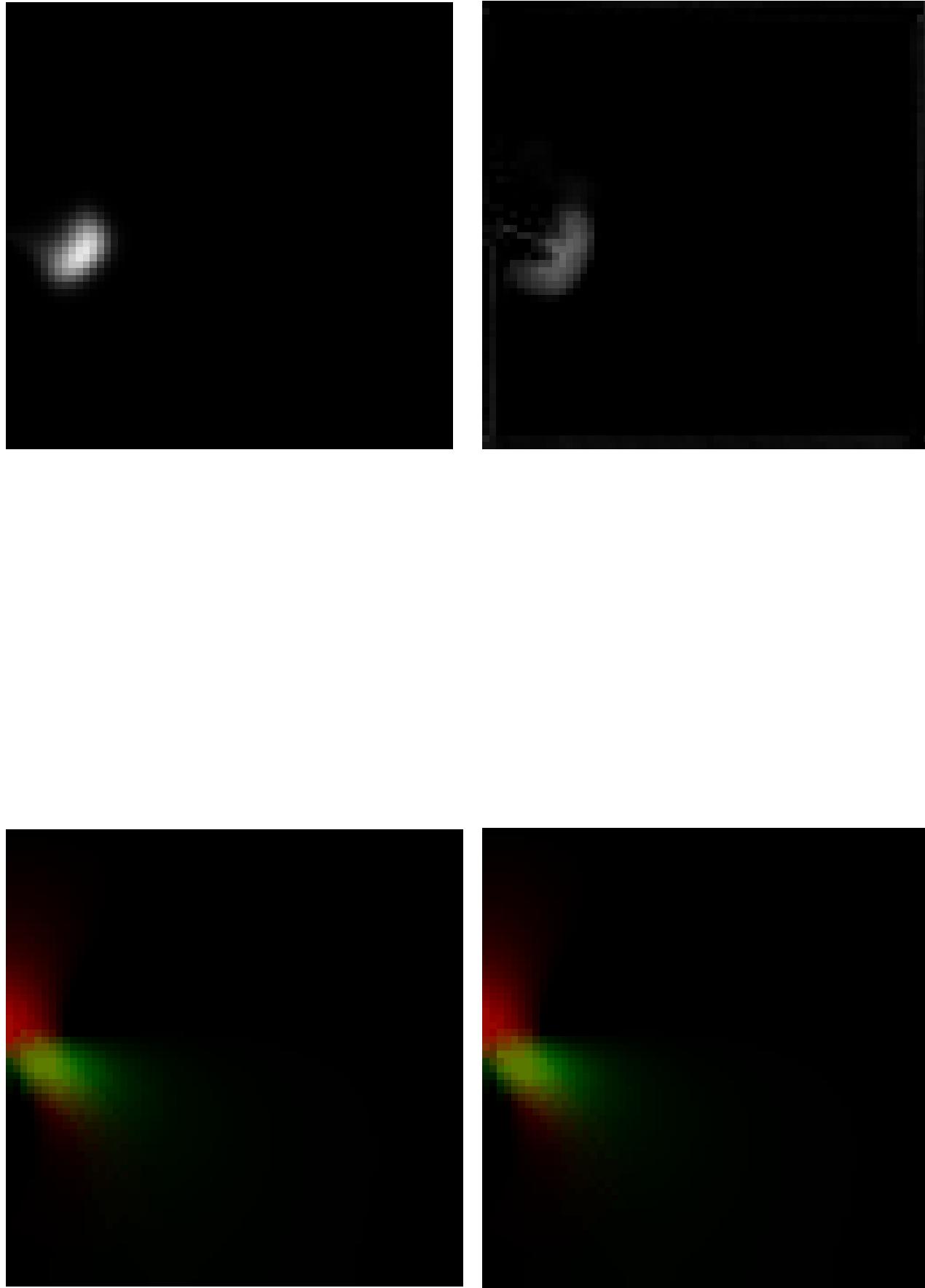


hidden chan.

Fluids

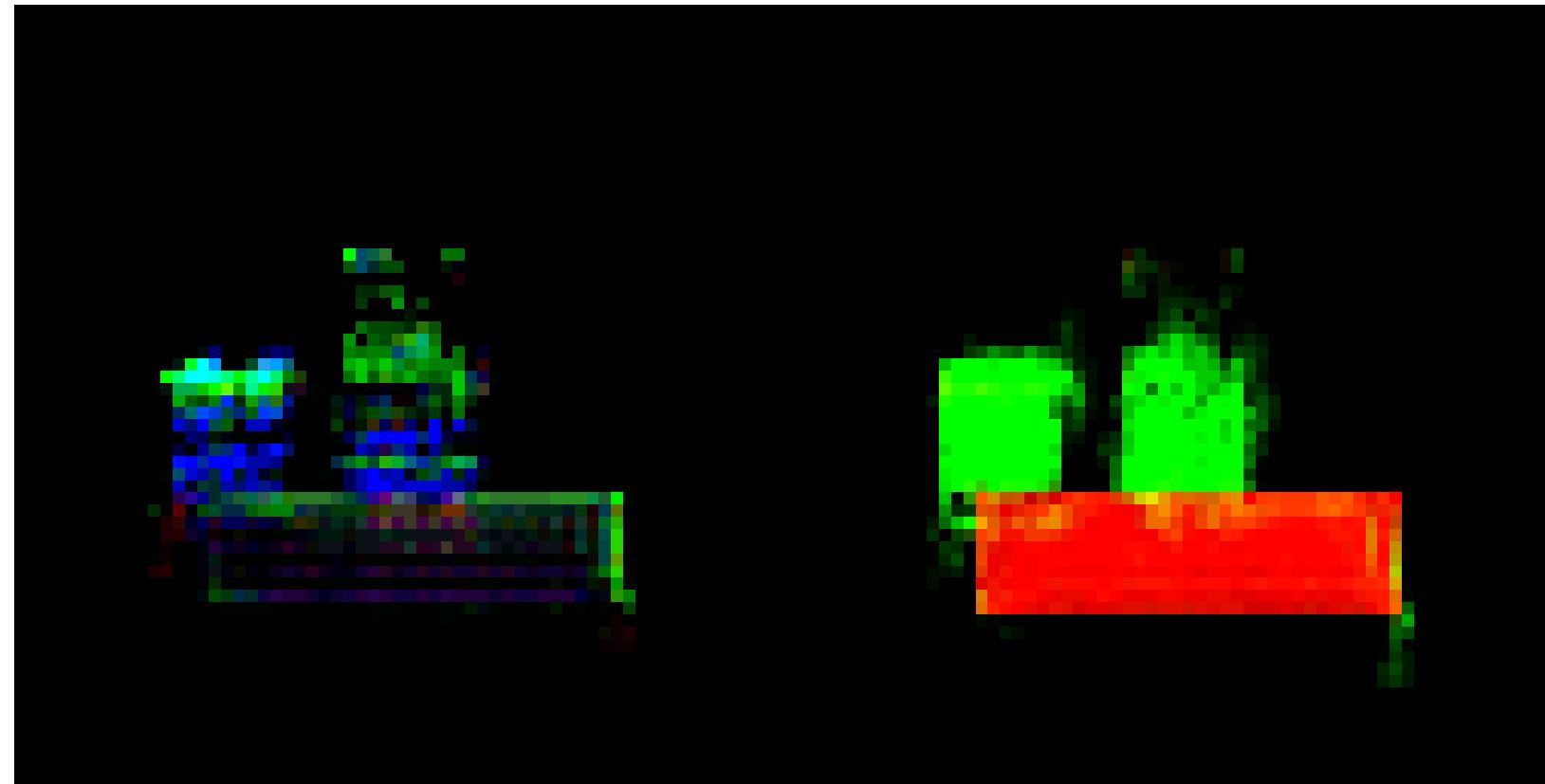


- More diverse and bigger dataset
- Greatly decreased delta time
- Deeper convolution network
- Dye transport and advection appears to work
- Velocity transport not really...
- Model does not make use of hidden channels → velocity channel is enough

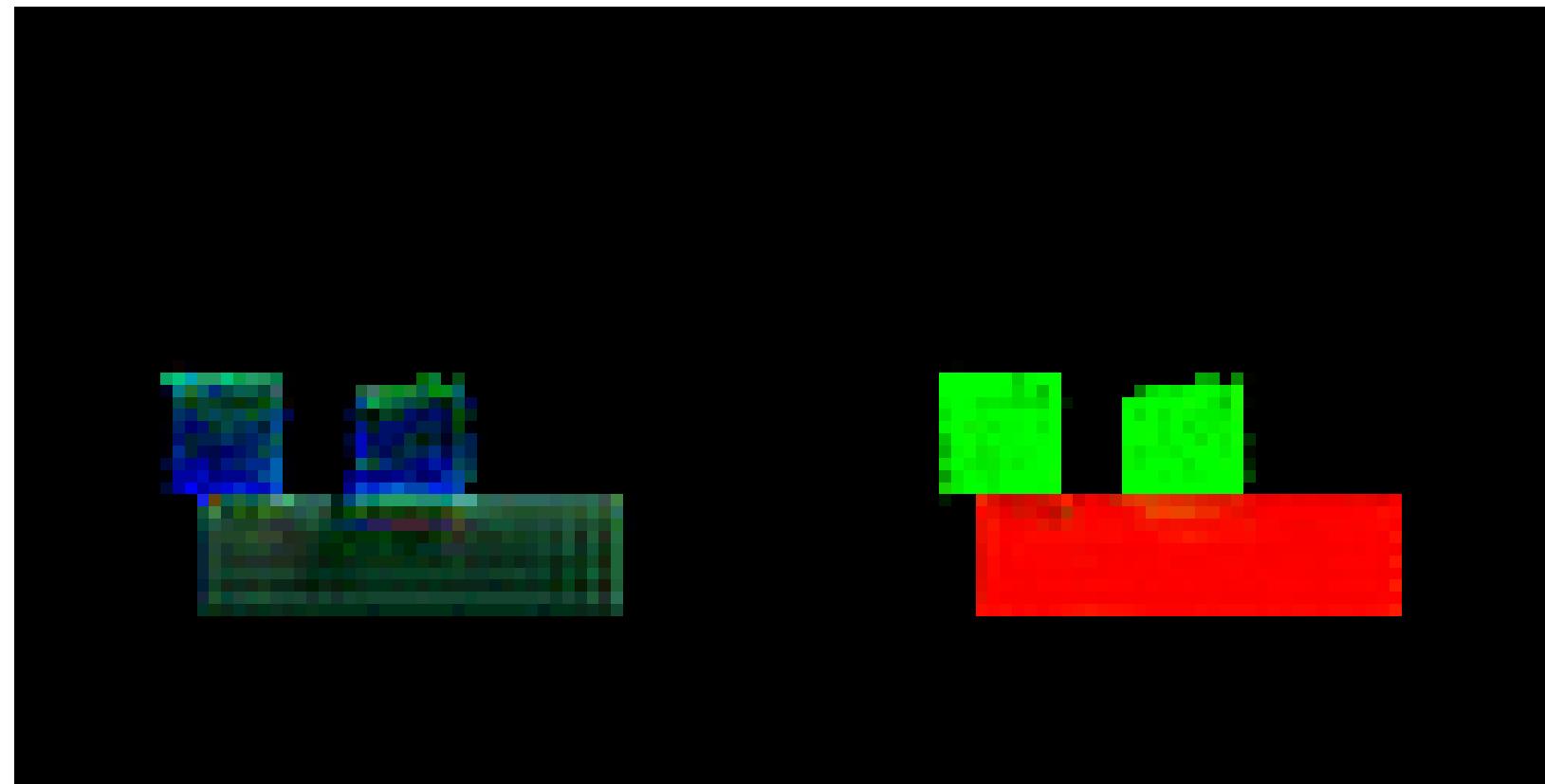


Materials interactions

epoch 100



epoch 500



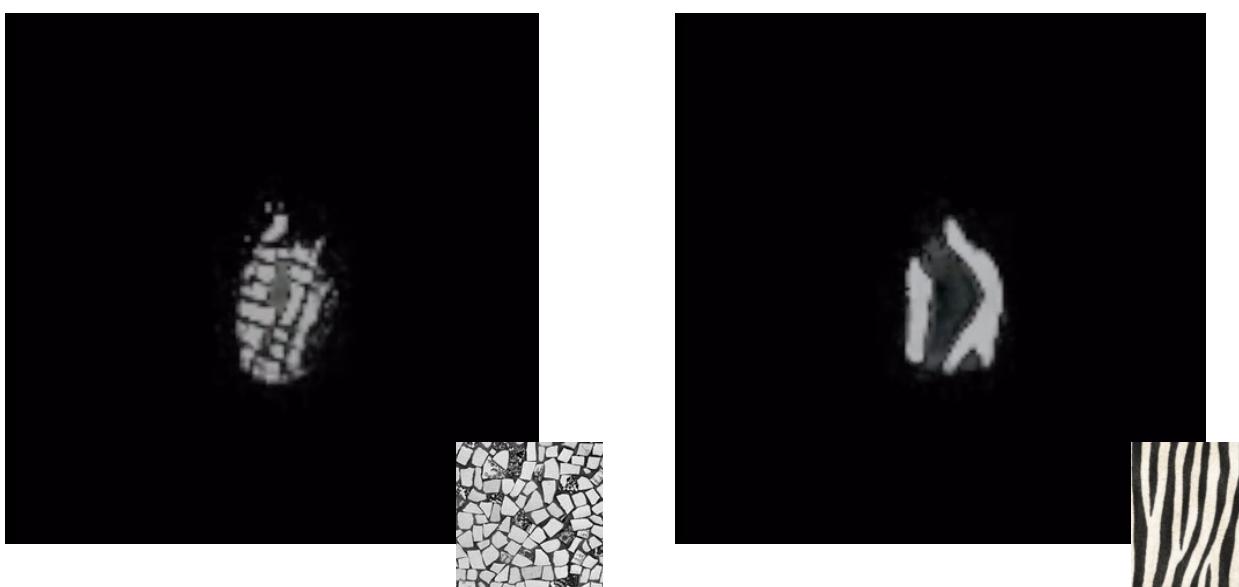
- try to learn to stop a falling green square
- use an extra channel per material type (here 2: green, red)
- look at the message passing at the moment of impact

Styling the simulation

1.



2.



μ NCA: Texture Generation with Ultra-Compact Neural Cellular Automata

Alexander Mordvinsev* Eyvind Niklasson*
Google Research
{moralex, eyvind}@google.com



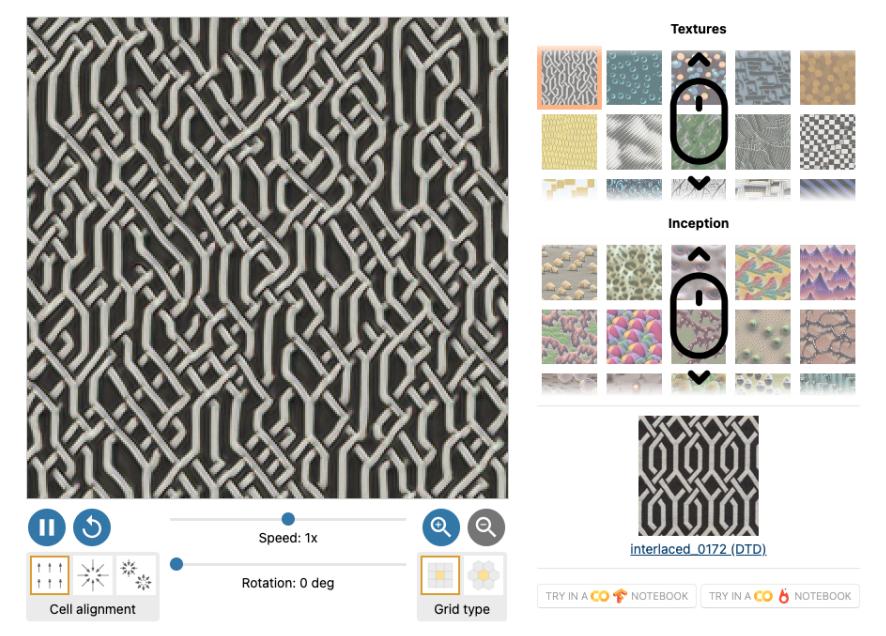
Abstract

We study the problem of example-based procedural tex-

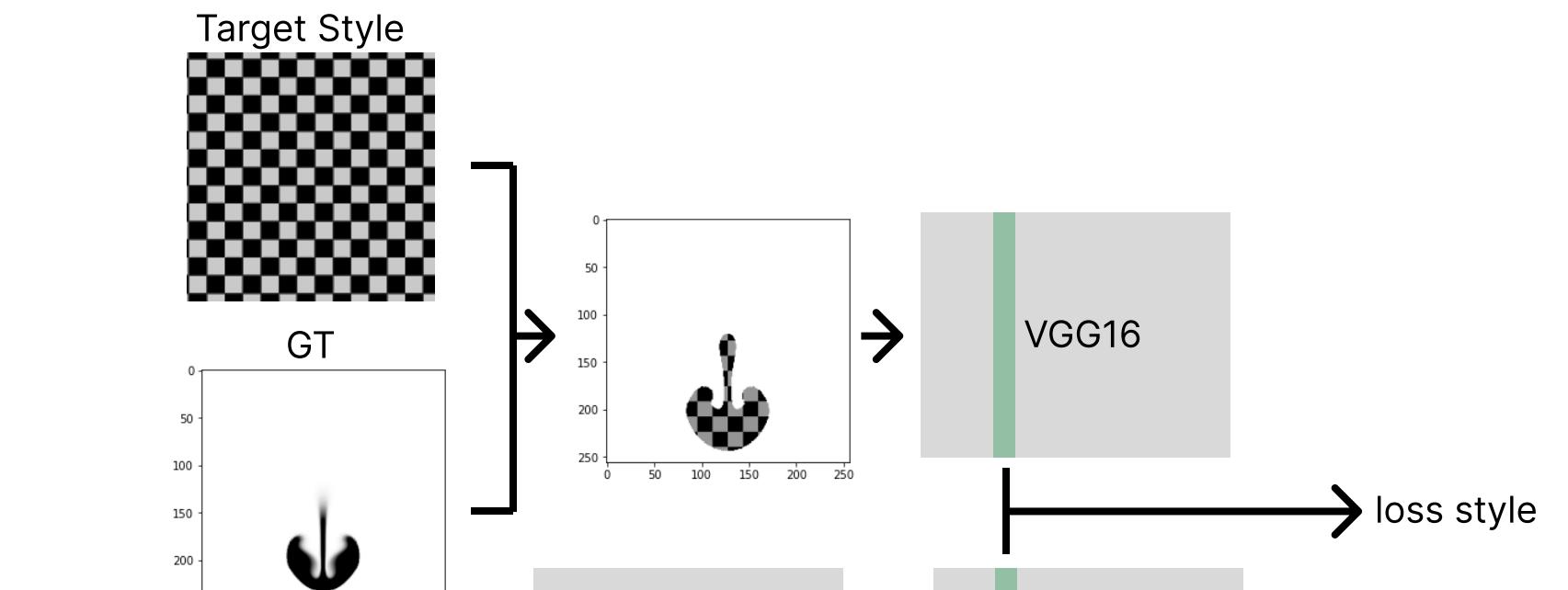
ture generation in general computing for almost as long as general purpose computing has been available to the public. Algorithmic information theory defines Kolmogorov complex-

Self-Organising Textures

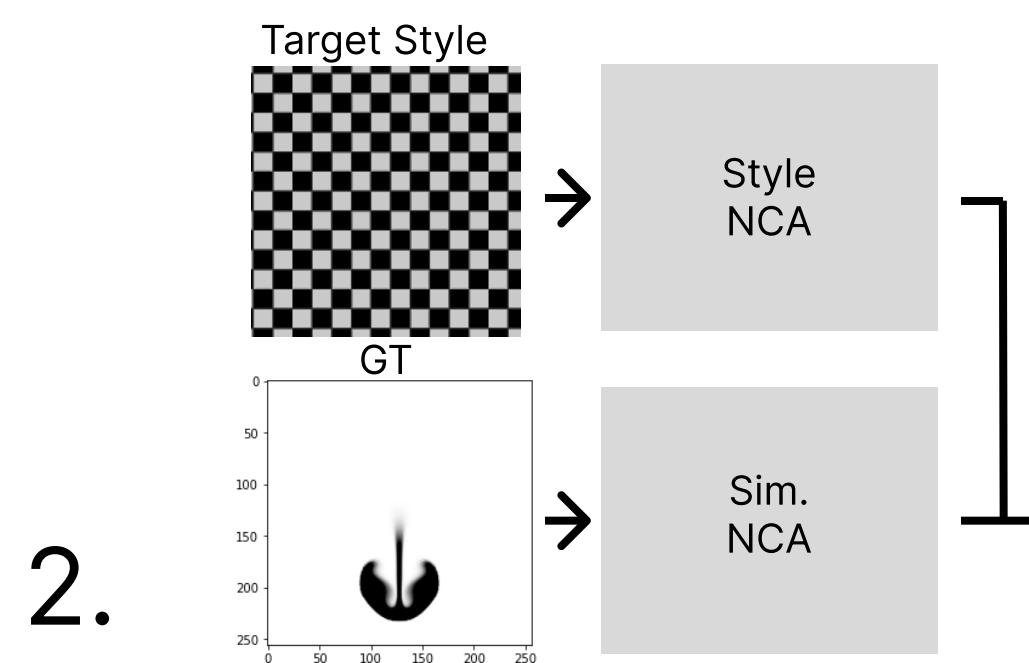
Neural Cellular Automata Model of Pattern Formation



AUTHORS
Eyvind Niklasson
Alexander Mordvinsev
Ettore Randazzo
Michael Levin
AFFILIATIONS
Google
Google
Google
Tufts
PUBLISHED
Feb. 11, 2021
DOI
10.23915/distill.00027.003



1.



2.

Conclusion & Next