CISC7016/CISC8001

Advanced Topics in Computer Science

Department of Computer and Information Science
Faculty of Science and Technology

Neural Cellular Automata GAN(NCA-GAN)

Xuhang Chen, YC174911, yc17491@umac.mo
Yan Xian, MC14965, mc14965@umac.mo

澳門大學
UNIVERSIDADE DE MACAU
UNIVERSITY OF MACAU

Introduction

Neural Cellular Automata(NCA)

Definition:

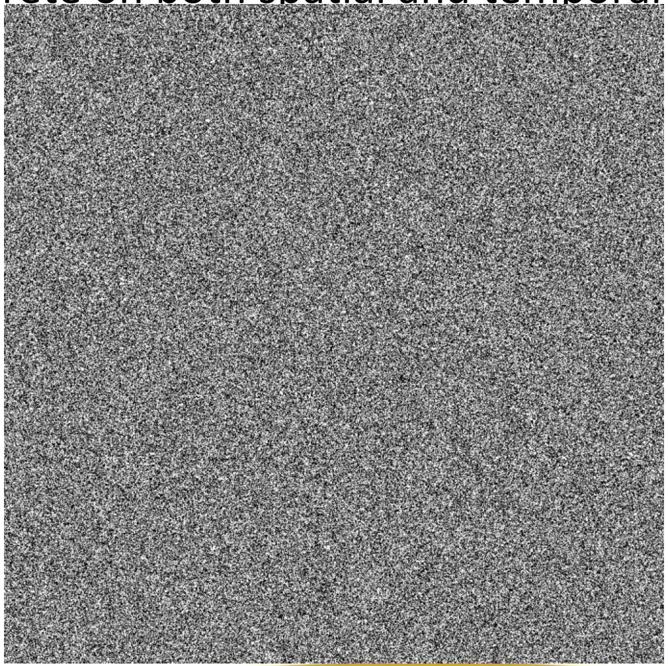
Computational systems characterized by being discrete on both spatial and temporal scales and abstract.

Four fundamental features:

1. discrete n-dimensional lattice of cells
2. discrete states
3. local interactions
4. discrete dynamics

Example:

Conway's game of life



Neural Cellular Automata(NCA)

One of the possible application of CAs is to biological systems, aiming at reproducing one of the most striking processes that characterize all the life forms: **morphogenesis**

The elemental building blocks of living organisms are able to cooperate with their neighbors in

1. deciding the shape, location and size of the organs and body parts
2. building all the necessary interconnection between them
3. knowing when, what and how to regenerate damaged portions of the organism

The Cellular Automaton will consist of a grid of cells that will be iteratively updated following the same update rule and in a way that the state of each cell depends only on those of its neighbors.
In this context, simulating this processes means identifying the rules at elemental unit level which give rise to the macroscopic regenerative behaviour

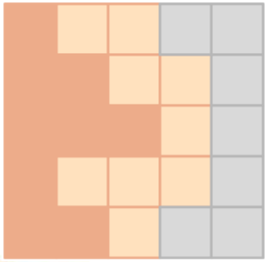
Cell States

Although not typical, continuous valued cell states have been used, which

- makes the update rule a differentiable function of the cell's neighbourhood's states
- better simulates the actual mechanism that rules living organisms' bodies

The cell state is a 16-dimensional real-valued vector

- the first three channels represent the cell color visible to humans (RGB)
- the 4-th channel, alpha channel α, has the role of defining living cells
    - $\alpha_{cell} > 0.1 \Rightarrow$ cell's neighbors are considered living, and the cell is mature
    - $\alpha_{cell} \leq 0.1 \Rightarrow$ cell's neighbors are dead ($\alpha_{cell} = 0$), and the cell is growing
- hidden channels do not have a specific meaning: they carry some information which can find several biological interpretations



Why

There are a couple of common feature between NCA and Convolutional Neural Networks(CNNs)

Neural networks can act as universal function approximator ⇒ each transition $\Sigma n \ni \sigma \rightarrow \Phi(\sigma) \in \Sigma$ can be approximated by a complex enough neural network

The single neighborhood operator Φ can be implemented as embedded in a convolutional neural network: it is applied to all pixels' (cells) neighborhood once the network has been fed with an input image

The transition rule must be the same for each cell and can depend solely on the 3x3 neighborhood of the cell ⇒ convolution operation

NCA rule

The update rule is divided into 4 phases

Perception

- implemented using three 3x3 convolutions with fixed kernels obtained by applying two Sobel filters and an identity filter
- 48-dimensional perception vector

Update rule

- using 1x1-convolutions and ReLU nonlinearity, each cell operate over the perception vector
- the update of the cells states are incremental, applied before each time step

Stochastic cell update

- single cell's updates are assumed as independent, separated by time intervals of random duration
- random per-cell mask

Living cell masking

- dead cells are forbidden from carrying out computations over the perception vector

$Sobel_x$

-1	0	+1
-2	0	+2
-1	0	+1

$Sobel_y$

-1	-2	-1
0	0	0
+1	+2	+1

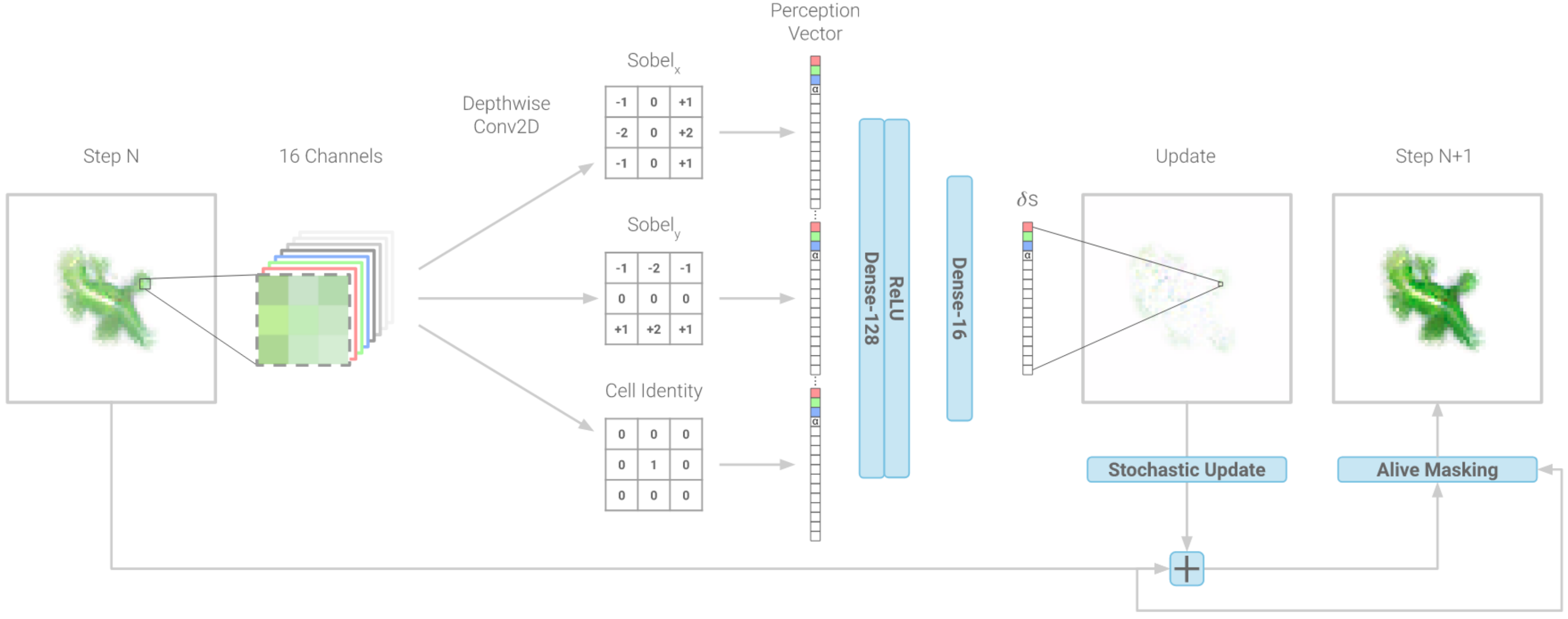
Cell Identity

0	0	0
0	1	0
0	0	0

Example

The output of a single update step is then fed back into the NCA

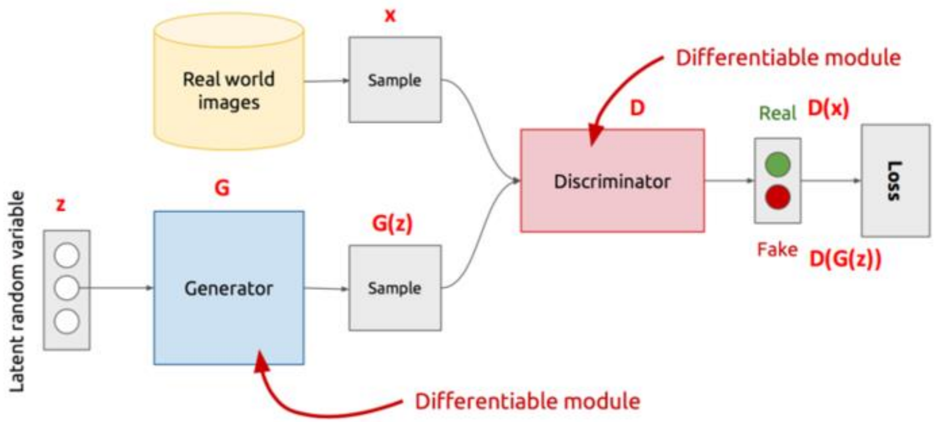
- the procedure is carried on for a few dozens iterations
- the same architecture with the same weights is used during this steps



Introduction(cont.)

Generative Adversarial Network (GAN)

Generator + Discriminator

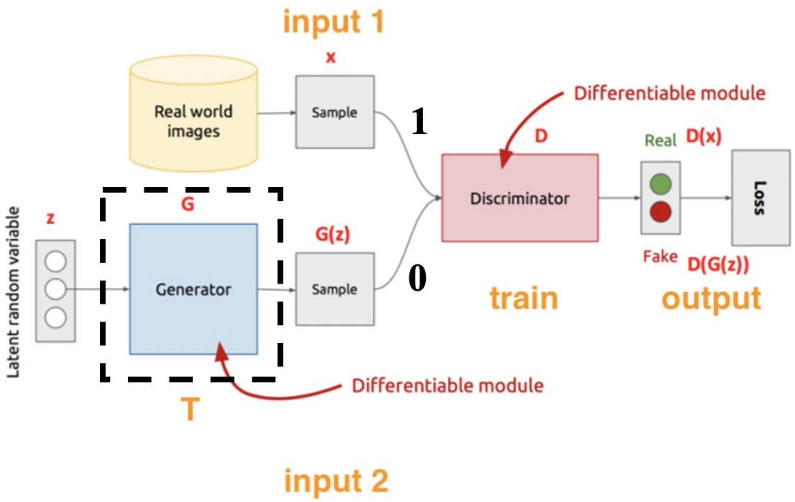


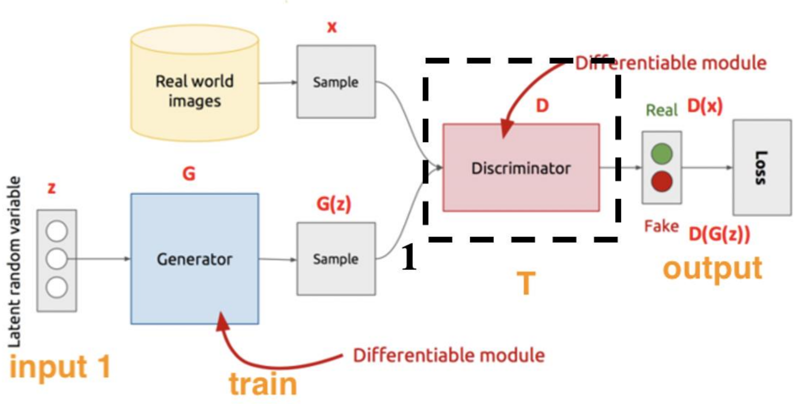
- **Z** is some random noise (Gaussian/Uniform).
- **Z** can be thought as the latent representation of the image.

Generator(G) Discriminator(D)

Generative Adversarial Network

Training Process





Related works

1. NCA can be regarded as a neural network[2]
2. NCA can generate images/patterns[3]
3. NCA with variational auto encoder can be a promising generative model[4]

Related works(Cont.)

1. VAE can be a generator in GAN[5]
2. NCA-GAN combination is present[6]

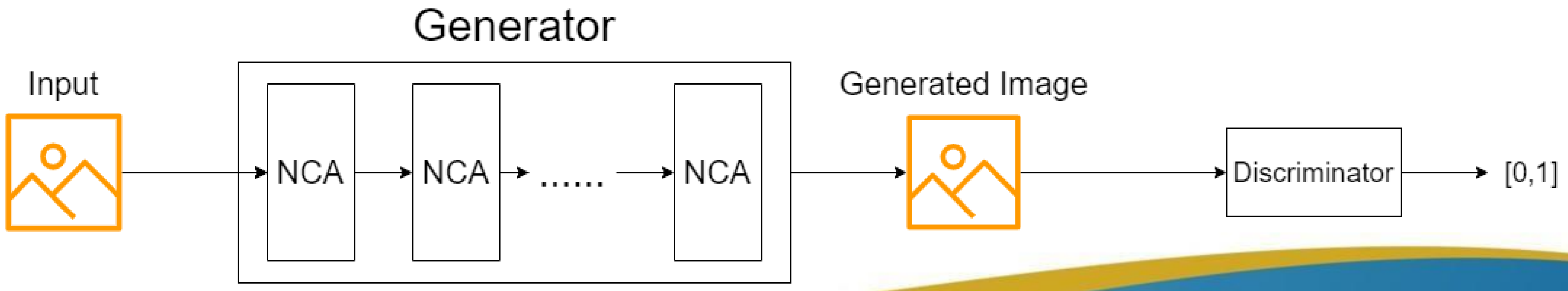
Challenges/Motivations

1. NCA
    - Need enhancement or other auxiliary components such as VAE
2. Discriminator
    - Match the generative capability of NCA
3. Loss function
    - Find a good loss function

Methodology

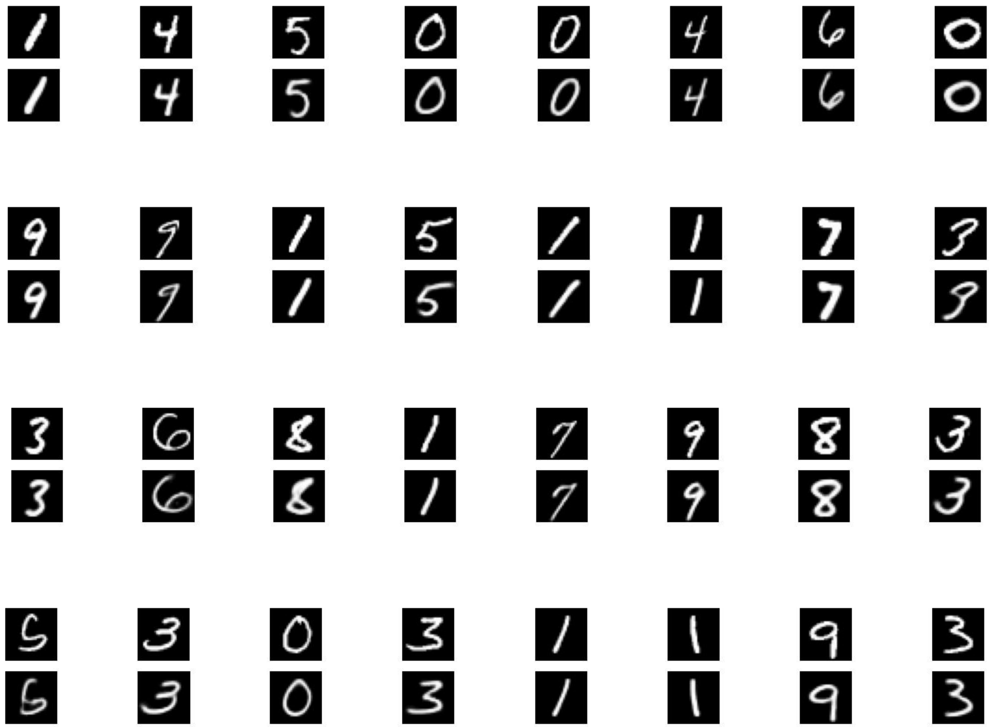
Our Generative Model

1. Typical GAN style, generator is replaced by NCA or its variants
2. Datasets
   - MNIST/CIFAR
   - CelebA
3. Metrics
   - PSNR/SSIM



Results

Some current results:



References

1. Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville A, Bengio Y. Generative adversarial networks. Communications of the ACM. 2020 Oct 22;63(11):139-44.
2. Gilpin W. Cellular automata as convolutional neural networks. Physical Review E. 2019 Sep 4;100(3):032402.
3. Mordvintsev A, Randazzo E, Niklasson E, Levin M. Growing neural cellular automata. Distill. 2020 Feb 11;5(2):e23.
4. Chen M, Wang Z. Image Generation With Neural Cellular Automatas. arXiv preprint arXiv:2010.04949. 2020 Oct 10.
5. Larsen AB, Sønderby SK, Larochelle H, Winther O. Autoencoding beyond pixels using a learned similarity metric. InInternational conference on machine learning 2016 Jun 11 (pp. 1558-1566). PMLR.
6. Otte M, Delfosse Q, Czech J, Kersting K. Generative Adversarial Neural Cellular Automata. arXiv preprint arXiv:2108.04328. 2021 Jul 19.

Q&A