Morphological Network: How Far Can We Go with Morphological Neurons?

Ranjan Mondal
Samsung Research, Bangalore, India
ranjan.rev@gmail.com

Sanchayan Santra
IDS, Osaka University, Osaka, Japan
sanchayan.santra@gmail.com

Soumendu Sundar Mukherjee
Indian Statistical Institute, Kolkata, India
soumendu041@gmail.com

Bhabatosh Chanda
Indian Statistical Institute, Kolkata, India
bchanda57@gmail.com

Motivation

Mathematical morphology provides powerful nonlinear tools that are suitable for visual tasks. However, devising a sequence of operation and their parameters for a given problem is not straightforward and requires expert knowledge. So, efforts have been made to automatically learn their parameters from data. These learnable structures are termed morphological neurons. It has been shown, that the applicability of morphological neurons is not just limited to visual tasks, they are also effective for classification [1,2] and regression [3]. But the literature lacks their thoretical analysis for practical purposes. So, in this work we have theoretically analysed the properties of morphological neurons and shown that a specific arrangement of the neurons can approximate any continuous function. To be more precise, we have defined a structure called a Morphological Block and shown that a sequence of two morphological blocks can work as a universal approximator. However, to facilitate the theoretical analysis, we have restricted ourselves to the 1D version of the morphological operators, where the operators work over the whole input at once, not locally.

Dilation and Erosion Neuron

Given an input $\boldsymbol{x} \in \mathbb{R}^d$ and a structuring element $\boldsymbol{s} \in \mathbb{R}^d$, the operation of **dilation** ($\oplus$) and **erosion** ($\ominus$) neurons are defined, respectively, as

$$\boldsymbol{x} \oplus \boldsymbol{s} = \max_k \{x_k + s_k\}, \tag{1}$$

$$\boldsymbol{x} \ominus \boldsymbol{s} = \min_k \{x_k - s_k\}, \tag{2}$$

where x_k denotes k^{th} element of input vector $\boldsymbol{x}$. In these neurons, the structuring element ($\boldsymbol{s}$) is learned in the training phase.

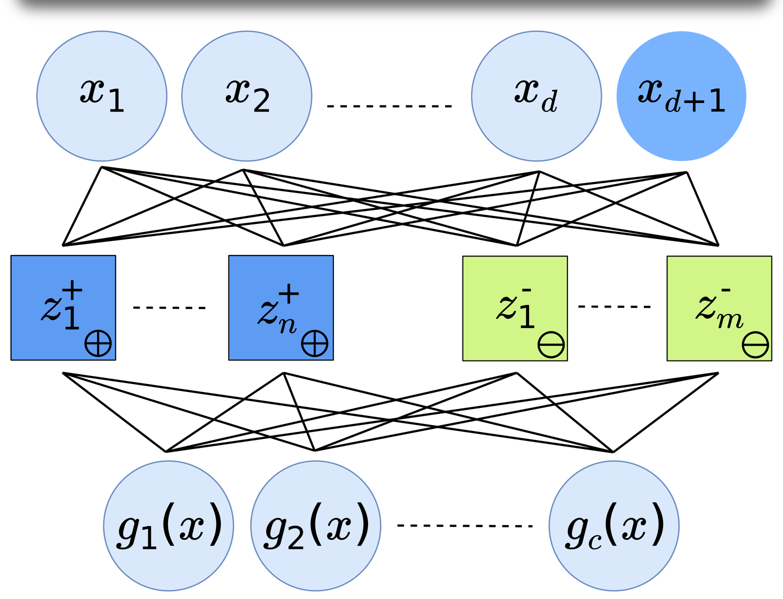
The max and min operators used in the dilation and erosion neurons are only piece-wise differentiable. To overcome this problem we propose to use the soft version of max and min to define *soft dilation* and *soft erosion* neurons as follows.

$$\boldsymbol{x}' \hat{\oplus} \boldsymbol{s}' = \frac{1}{\beta} \log \left(\sum_{k+1} e^{(x'_k + s'_k)\beta} \right), \tag{3}$$

$$\boldsymbol{x}' \hat{\ominus} \boldsymbol{s}' = -\frac{1}{\beta} \log \left(\sum_{k+1} e^{(s'_k - x_k)\beta} \right), \tag{4}$$

where $\hat{\oplus}$ and $\hat{\ominus}$ denote the soft dilation and soft erosion, respectively, and β is the "hardness" of the soft operations.

Morphological Block



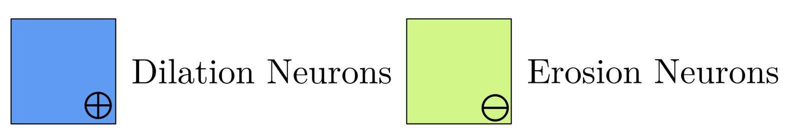


Fig 1. Architecture of a single layer morphological block. It contains an input layer, a dilation-erosion layer with n dilation and m erosion neuron and a linear combination layer with c neurons producing the output.

Equivalence of configurations

Theorem 1. *If we denote $D_{m_1}E_{m_2}$ as a layer with m_1 dilation neurons and m_2 erosion neurons and L as a linear combination layer, the following may be said about their configurations.*

(i) The architecture $D_{m_1}E_0 \to D_{m_2}E_0 \to \cdots \to D_{m_\ell}E_0$ consisting only of dilation layers is equivalent to the architecture $D_{m_\ell}E_0$ with a single dilation layer. A similar statement is true if one considers architectures with only purely erosion layers.

(ii) The architecture $D_1E_1 \to D_1$ is not equivalent to D_1E_0. Similarly, it is not equivalent to D_0E_1, and, consequently, the architectures $D_1E_1 \to D_1E_1$ and D_1E_1 are not equivalent.

(iii) The architecture $D_1E_1 \to D_1 \to L$ is not equivalent to $D_1E_0 \to L$.

(iv) The architecture $D_2E_0 \to D_0E_2 \to D_1$ is not equivalent to $D_2E_0 \to D_1$.

Morphological block : A sum of hinge functions

Definition 1 (k-order Hinge Function [7]). *] A k-order hinge function $h^{(k)}(\boldsymbol{x})$ consists of $(k+1)$ hyperplanes continuously joined together. It may be defined as*

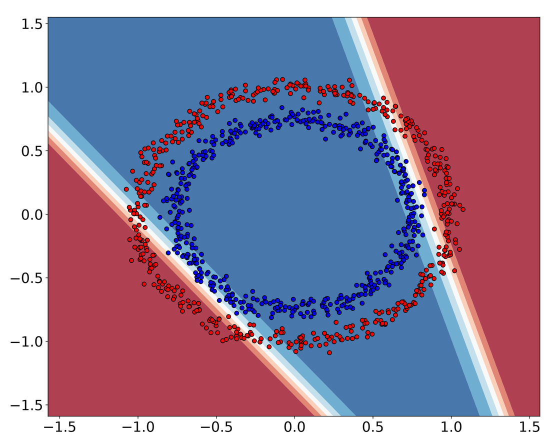
$$h^{(k)}(\boldsymbol{x}) = \pm \max\{\boldsymbol{w}_1^T\boldsymbol{x} + b_1, \boldsymbol{w}_2^T\boldsymbol{x} + b_2, \ldots, \boldsymbol{w}_{k+1}^T\boldsymbol{x} + b_{k+1}\} \tag{1}$$

Proposition 1. *The function computed by a Morphological Block (denoted by $\mathcal{M}(\boldsymbol{x})$) with n dilation and m erosion neurons followed by their linear combination, is a sum of multi-order hinge functions.*

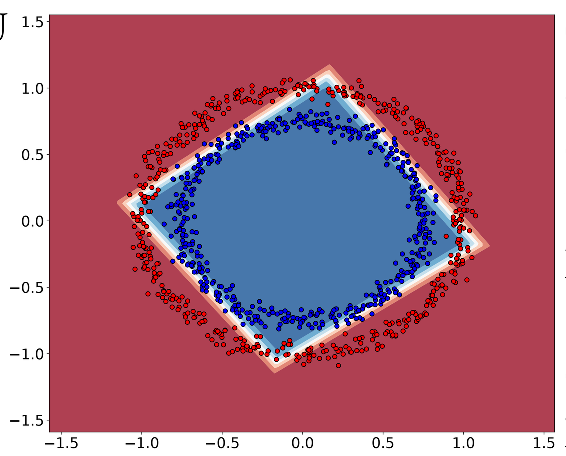
In fact, we can show that

$$\mathcal{M}(\boldsymbol{x}) = \sum_{i=1}^{l} \alpha_i h_i^{(d)}(\boldsymbol{x}), \tag{2}$$

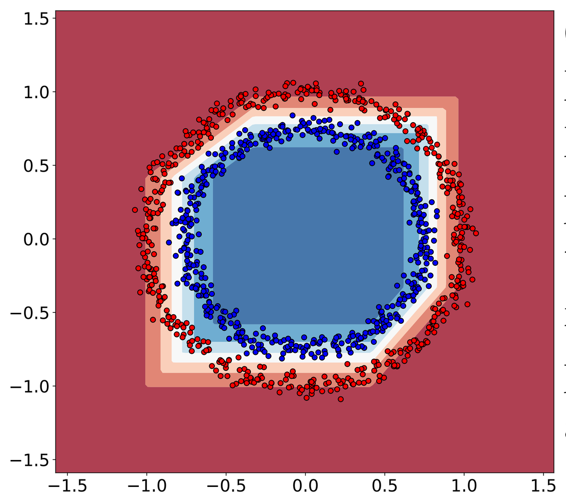
where $l = m+n, \alpha_i \in \{1, -1\}$ and $h_i^{(d)}(\boldsymbol{x}), 1 \leq i \leq l$, are d-order hinge functions. The proof is given in the supplementary material.



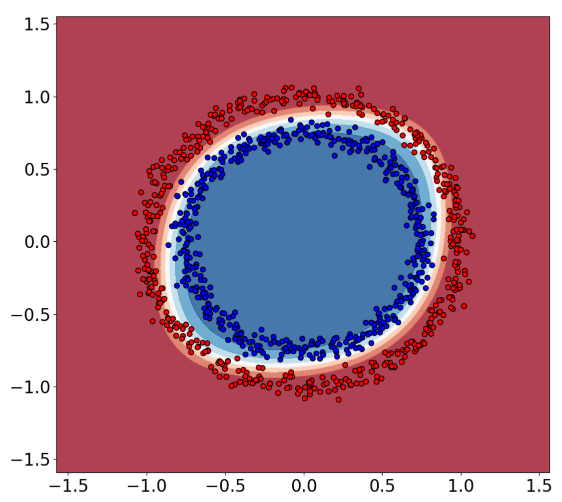
(a) NN-ReLU network is able to learn only two planes.



(b) Maxout network is able to learn two more planes with the help of additional parameters. .



(c) Morphological Block learns more planes with the same number of parameters as NN-ReLU.



(d) Soft Morpho-logocal block, smooths the learned decision boundary.

Fig 2. Decision boundaries learned by different networks with two hidden neurons.

Two Morphological Blocks: An universal approximator

Lemma 1. *Any linear combination of hinge functions $\sum_{i=1}^{m} \alpha_i h^{(k_i)}(\boldsymbol{x})$ can be represented over an arbitrary compact set K as a two sequential morphological block consisting of dilation neurons only.*

Theorem 2 (Universal approximation). *Two morphological blocks applied sequentially, can approximate continuous functions over arbitrary compact sets.*

Proof. Continuous functions can be approximated over compact sets by sums of hinge functions (Theorem 3.1 of [7]). Therefore, by Lemma 1, it follows that any continuous function can be approximated over arbitrary compact sets by two-layer Morph-Nets. □

Results

Dataset	Test Accuracy		
	Morph-Net	Soft Morph-Net ($\beta = 8$)	Similar Network
MNIST	98.39	98.90	**99.79** [4]
Fashion-MNIST	89.87	**89.84**	89.70 [5]

Table 1: Accuracy on MNIST and Fashion-MNIST Datasets using a single hidden layer with 400 morphological neurons.

Architecture	l=200		l=400		l=600	
	CIFAR10	SVHN	CIFAR10	SVHN	CIFAR10	SVHN
NN-tanh	46.6 ± 0.06	73.9 ± 0.12	46.9 ± 0.04	73.9 ± 0.23	48.0 ± 0.05	75.6 ± 0.14
NN-ReLU	47.2 ± 0.11	64.2 ± 0.88	48.0 ± 0.05	76.2 ± 0.32	48.1 ± 0.02	$\mathbf{79.5 \pm 0.11}$
Maxout-Network ($k = 2$) [6]	46.9 ± 0.05	69.4 ± 0.10	48.0 ± 0.10	74.1 ± 0.22	46.4 ± 0.33	37.8 ± 3.15
Our	52.0 ± 0.02	73.4 ± 0.03	53.6 ± 0.01	76.9 ± 0.03	54.0 ± 0.02	78.2 ± 0.03
Our (Soft: $\beta = 12$, 20)	$\mathbf{53.5 \pm 0.04}$	$\mathbf{74.1 \pm 0.06}$	$\mathbf{55.8 \pm 0.05}$	$\mathbf{77.0 \pm 0.05}$	$\mathbf{56.9 \pm 0.04}$	78.5 ± 0.05

Table 2: Test accuracy achieved on CIFAR-10 and SVHN dataset by different networks when the number of neurons (l) in the hidden layer is varied. The value of β is 12 and 20 for CIFAR10 and SVHN respectively.

References

[1] G. X. Ritter and P. Sussner. An introduction to morphological neural networks. In Proceedings of 13th International Conference on Pattern Recognition, volume 4, pages 709–717 vol.4, August 1996.

[2] Peter Sussner and Estevão Laureano Esmi. Morphological perceptrons with competitive learning: Lattice-theoretical framework and constructive learning algorithm. Information Sciences, 181(10):1929–1950, May 2011. ISSN 0020-0255.

[3] Ricardo de A. Araújo. A morphological perceptron with gradient-based learning for Brazilian stock market forecasting. Neural Networks, 28:61–81, April 2012. ISSN 0893-6080.

[4] Li Wan, Matthew Zeiler, Sixin Zhang, Yann Le Cun, and Rob Fergus. Regularization of neural networks using dropconnect. In International Conference on Machine Learning, pages 1058–1066, 2013.

[5] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. arXiv preprint arXiv:1708.07747, 2017.

[6] Ian J. Goodfellow, David Warde-Farley, Mehdi Mirza, Aaron Courville, and Yoshua Bengio. Maxout Networks. In Proceedings of the 30th International Conference on International Conference on Machine Learning - Volume 28, ICML'13, pages III–1319–III–1327, Atlanta, GA, USA, 2013.

[7] Leo Breiman. Hinging hyperplanes for regression, classification, and function approximation. IEEE Transactions on Information Theory, 39(3):999–1013, May 1993.