Deep Morphological Neural Networks

Yucong Shen, Xin Zhong, and Frank Y. Shih,
presented by Nicolas Boutry[1] and Guillaume Tochon[1]

[1] Laboratoire de Recherche et Développement de l'EPITA (LRDE), France

30th September 2019

Outline

Outline

Let assume the sequence of morphological operators $(\phi_i)_{i \in [1,n]}$, like erosions or dilations, each one associated to a structuring element ρ_i.

We want this sequence to be able to go from an image I and to result in the image J by the following operation:

$$J = \phi_1(\ldots(\phi_n(I)))$$

Finding the good operators $(\Phi_i)_i$ and their associated SE's can be cumbersome and time-consuming task.

The authors propose then to make an adaptation of a CNN called Morphological Neural Networks (MNN's):

One (morphological) layer $=$ one morphological operator.

However, usual morphological operators are <u>non-differentiable</u>

$\Rightarrow$ no gradient descent method is possible

$\Rightarrow$ some adaptation is needed.

$\Rightarrow$ we will present this adaption and its benefits.

Weaknesses of preceding architectures:

- they can learn only flat structuring elements (SE),
- they cannot learn the morphological operations $(\phi_i)_i$.

Outline

Masci *et al.* proposed this approximation of erosions and dilations:

$$(f *_P \omega)(x) := \frac{(f^{P+1} * \omega)(x)}{(f^P * \omega)(x)},$$

where f is the grayscale image, x the position, ω the kernel, and P the power s.t.:

- $P \to \infty$ leads to the dilation,
- $P \to -\infty$ leads to the erosion,
- $P > 0$ leads to a pseudo-dilation,
- $P < 0$ leads to a pseudo-erosion,
- $P = 0$ leads to the usual convolution.

However, this approximations are often very inaccurate. Then, Shih *et al.* proposed to use the softmin/softmax functions instead:

$$-\ln\left(\sum_{i=1}^{n} e^{-\omega_i x_i}\right), \text{ and } \ln\left(\sum_{i=1}^{n} e^{\omega_i x_i}\right),$$

where i corresponds to the number of the pixel and n the number of weights of the SE ω.

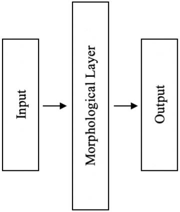
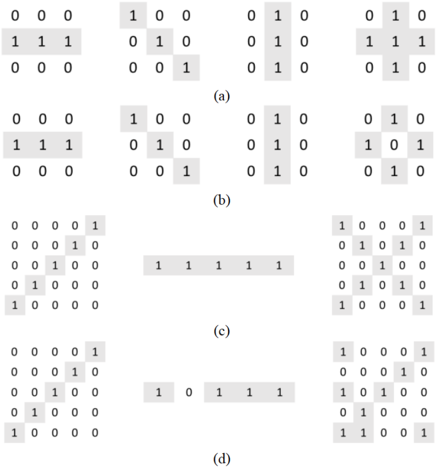


Fig. 1. Architecture of the single layer MNN.



Problem: failure of the estimation of the needed SE's due to rounding errors.

Shen *et al.* propose then the following definition:

Definition (Differentiable binary dilation)

$$Y_j = \ln\left(\sum_{i=1}^{n} e^{W_i X_i}\right) + b$$

Definition (Differentiable binary erosion)

$$Y_j = -\ln\left(\sum_{i=1}^{n} e^{-W_i X_i}\right) + b$$

Definition (Differentiable grayscale dilation)

$$Y_j = \ln\left(\sum_{i=1}^{n} e^{W_i + X_i}\right) + b$$

Definition (Differentiable grayscale erosion)

$$Y_j = -\ln\left(\sum_{i=1}^{n} e^{-W_i - X_i}\right) + b$$

Learning procedure: we can compute the gradient for the proposed layer like usual:

$$g^{\omega} := \frac{\partial J}{\partial \omega}$$

$$g^{b} := \frac{\partial J}{\partial b}$$

where J is the (objective) function to minimize. Assuming that the learning rate is η, we obtain then:

$$\omega_{i+1} := \omega_i - \eta\, g_t^{\omega}, \text{ and } b_{i+1} := b_i - \eta\, g_t^{b}.$$

Outline

Goal: how we learn the SE's of a **stack** of morphological layers where operations are already defined.

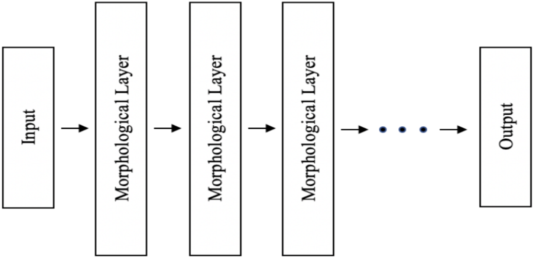


Fig. 3. Architecture of the multi-layer deep morphological neural network.

Concerning the computed feature maps, we have either a dilation:

$$Z^{(l)} = \omega^{l} \bigoplus z^{(l-1)} + b^{l},$$

or an erosion:

$$Z^{(l)} = \omega^{l} \ominus z^{(l-1)} + b^{l}.$$

Assuming we use some activation function σ, we obtain the new formula :

$$Z^{(l)} = \sigma\left(\omega^{l} \bigoplus z^{(l-1)} + b^{l}\right),$$

then:

$$g^{(l)} = \frac{\partial J}{\partial \omega} = \frac{\partial J}{\partial z^{(l)}} \frac{\partial z^{(l)}}{\partial \omega},$$

from which we will deduce:

$$\omega_{i+1}^{(l)} := \omega_{i}^{(l)} - \eta\, g_{t}^{(l)},$$

Outline

Application: detect the corners as residuals of some MM-based procedure (for feature extraction or shape classification).

Fig. 4. The morphological residual model. Applying opening on the original image with circle structuring elements, then subtraction of result image from original image can obtain the morphological residuals.

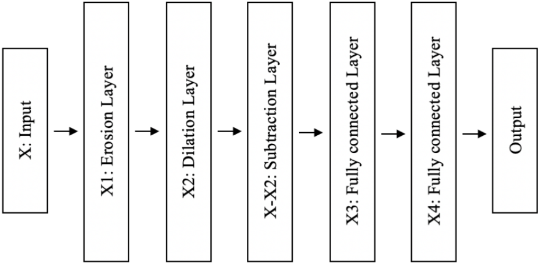


Fig. 5. The architecture of residual MNN.

Outline

Goal: Learn the transformations (erosion vs. dilation)

Proposed model:

$$z_j = sign(a)\ \ln\left(\sum_{i=1}^{N} \exp^{sign(a)\omega_i x_i}\right),$$

where j is the number of the N studied pixel in the feature map.

Idea: the trainable value $a \in \mathbb{R}$ induces the sign used in the expression.

Problem: we need an smooth sign function!

Propositions of Shen *et al.*:

$$f(x) = \frac{x}{1+|x|} \quad \text{(soft sign function)},$$

or:

$$g(x) = \frac{\exp^x - \exp^{-x}}{\exp^x + \exp^{-x}} \quad \text{(hyperbolic tangent function)}.$$

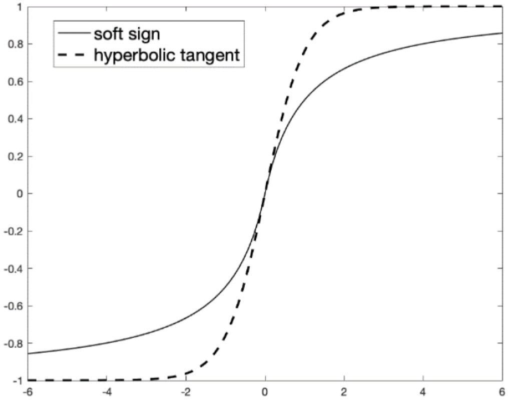


Fig. 6. The soft sign function and hyperbolic tangent function.

Then a is learnable and we obtain:

$$z_j = f(a)\ \ln\left(\sum_{i=1}^{N} \exp^{f(a)\omega_i x_i}\right),$$

and

$$z_j = g(a)\ \ln\left(\sum_{i=1}^{N} \exp^{g(a)\omega_i x_i}\right),$$

Now, there is the model to learn an adaptive morphological layer:

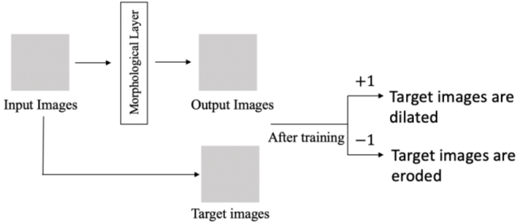


Fig. 7. The flow chart of detecting morphological operations by a single adaptive morphological layer MNN.

In brief, a set of input images with their eroded outputs will lead to a sign to -1 and a set of input images with their dilated outputs will lead to a sign to $+1$.

This is due to the learning procedure using the following chain rule:

$$g_a^{(l)} = \frac{\partial J}{\partial a^{(l)}} = \frac{\partial J}{\partial z}\frac{\partial z}{\partial a} = \frac{\partial J}{\partial z}\varphi'(a)$$

where φ is in $\{f, g\}$, and then we have the usual formula:

$$a_{t+1}^{(l)} = a_t^{(l)} - \eta\, g_a^{(l)}.$$

Outline

Shen *et al.* propose to proceed to shape features recognition with MNN's on these 4 families of images:

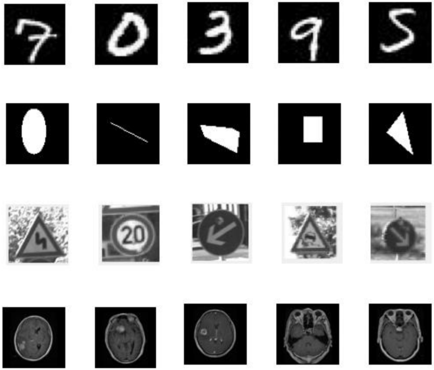


Fig. 8. The examples from the four datasets in the experiments. The first row is the images from MNIST dataset, the second row from SCGS dataset, the third row from GTSRB dataset, and the fourth row from brain tumor dataset.

Learning the SE's I

After 20 epochs: accuracy of 91%, and after 100 epochs, accuracy of 97%.

0	1	0
1	1	1
0	1	0

1	0	0	0	1
0	1	0	1	0
0	0	1	0	0
0	1	0	1	0
1	0	0	0	1

1	1	1	1	1

(a)

0	1	0
1	1	1
0	1	0

1	0	0	0	1
0	1	0	1	0
0	0	1	0	0
0	1	0	1	0
1	0	0	0	1

1	1	1	1	1

(b)

Fig. 9. (a) The diamond 3×3 structuring element, crossing 5×5 structuring element, and 1×5 structuring element, (b) the learned structuring elements by a single dilation layer MNN after improvement.

Learning the SE's II

With non-flat SE's, after 20 epochs, original and learned SE are very close:

(a)

0.2060	0.3234	0.6542
0.3551	0.5692	0.3950
0.6405	0.5834	0.5104

0.2086	0.3211	0.6521
0.3540	0.8055	0.4135
0.6261	0.5747	0.5097

(b)

0.8329	0.4865	0.9737
0.0440	0.8055	0.1752
0.6563	0.5816	0.0463

0.8361	0.4876	0.8951
0.0559	0.5054	0.1763
0.6585	0.5836	0.0573

Fig. 10. (a) The top box shows the original structuring element and the bottom box shows the learned structuring elements by a single dilation layer MNN. (b) The top box shows the original structuring elements and the bottom box shows the learned structuring elements by a single erosion layer MNN.

Learning the SE's III

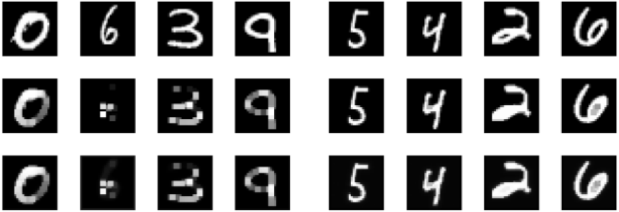
Learned erosions and dilations:

(a) (b)

Fig. 11. The results of learning grayscale (a) dilation and (b) erosion operations by MNN. The first row shows the original images, the second row shows the target images, and the third row shows the output of the network after training 20 epochs.

Learning the SE's IV

Learned openings and closings:



(a) (b)

Fig. 12. The results of learning (a) opening and (b) closing operations by DMNN. The first row shows the original images, the second row shows the target images, and the third row shows the output of the network after training 20 epochs.

Learning the morphological operators I

Estimation of the accuracy of the sign estimation:

TABLE III
DETECTION ACCURACY OF TWO SMOOTH SIGN FUNCTIONS

	Dilation	Erosion
Soft sign	100%	100%
Hyperbolic tangent	100%	100%

Classification I

MNN's are very competitive:

TABLE VI
COMPARISON OF RESIDUAL MNN WITH STATE-OF-ART CONVOLUTIONAL NEURAL NETWORKS

Classifier	Dataset	Testing accuracy	Number of parameters
MCDNN [3]	MNIST	99.77%	2,682,470
Residual MNN	MNIST	98.93%	104,181
MLeNet	SCGS	99.50%	10,493,795
Residual MNN	SCGS	98.89%	4,721,175
MLeNet	GTSRB (Grayscale)	97.94%	4,202,339
Residual MNN	GTSRB (Grayscale)	96.49%	1,594,903
MLeNet	Brain tumor	96.10%	10,493,795
Residual MNN	Brain tumor	95.43%	4,721,175

Classification II

TABLE IX
COMPARISON OF RESIDUAL MNN AND RESIDUAL CNN

	Residual MNN ($a = 1$)	Residual CNN ($b = 1$)	Residual MNN ($a = 16$)	Residual CNN ($b = 16$)
MNIST	98.93%	97.14%	97.78%	98.18%
SCGS	98.89%	98.25%	98.90%	98.91%
GTSRB	96.49%	90.60%	97.48%	93.39%
Brain tumor	95.43%	96.10%	96.75%	94.15%

Outline

Shen and his coworkers succeeded in:

- making differentiable their morphological feature extractors,
- obtaining adaptive morphological layer,
- developing new MNN's based on non-flat SE's,
- making MNN's almost as efficient as state-of-the-art CNN's but with much less weights,
- and then in making faster competitive NN's for geometrical/topological feature extraction, classification, etc.

Questions? :D