Neural Networks 23 (2010) 770–781

Contents lists available at ScienceDirect

Neural Networks

journal homepage: www.elsevier.com/locate/neunet

2010 Special Issue

Classification of 2-dimensional array patterns: Assembling many small neural networks is better than using a large one

Liang Chen [a,*], Wei Xue [b], Naoyuki Tokuda [c,1]

[a] Computer Science Department, University of Northern British Columbia, Prince George, B.C., Canada V2N 4Z9

[b] Wenzhou University, Wenzhou City, Zhejiang Province, China

[c] SunFlare R & D Center, Shinjuku Hirose Bldg, Yotsuya 4-7, Shinjuku-ku, Tokyo 160-0004, Japan

ARTICLE INFO

Article history:
Received 17 October 2009
Revised and accepted 24 March 2010

Keywords:
Neural network
Robust
Task decomposition
Noisy environment
Pattern classification
Image and signal processing

ABSTRACT

In many pattern classification/recognition applications of artificial neural networks, an object to be classified is represented by a fixed sized 2-dimensional array of uniform type, which corresponds to the cells of a 2-dimensional grid of the same size. A general neural network structure, called an undistricted neural network, which takes all the elements in the array as inputs could be used for problems such as these. However, a districted neural network can be used to reduce the training complexity. A districted neural network usually consists of two levels of sub-neural networks. Each of the lower level neural networks, called a regional sub-neural network, takes the elements in a region of the array as its inputs and is expected to output a temporary class label, called an individual opinion, based on the partial information of the entire array. The higher level neural network, called an assembling sub-neural network, uses the outputs (opinions) of regional sub-neural networks as inputs, and by consensus derives the label decision for the object. Each of the sub-neural networks can be trained separately and thus the training is less expensive. The regional sub-neural networks can be trained and performed in parallel and independently, therefore a high speed can be achieved. We prove theoretically in this paper, using a simple model, that a districted neural network is actually more stable than an undistricted neural network in noisy environments. We conjecture that the result is valid for all neural networks.

This theory is verified by experiments involving gender classification and human face recognition. We conclude that a districted neural network is highly recommended for neural network applications in recognition or classification of 2-dimensional array patterns in highly noisy environments.

1. Introduction

1.1. Problem definition

Our aim in a typical pattern classification application of artificial neural networks is to approximate a mapping $f : \mathcal{R}^{l\times w} \rightarrow \mathcal{L}$, where $\mathcal{L}$ is a set of class labels, l and w are integers. The object, $\mathcal{R}$, to be classified is represented by an $l \times w$ array of uniform type which corresponds to cells in a $l \times w$ grid. We assume here that $l \times w$ is a large number so that our discussion of large neural networks makes sense.

An example of such applications is the classification of images, where an object is an $l \times w$ array of which the elements are the intensity values of pixels in an image. Another application is the nucleotide sequence promoter predication problem, where $l = 1$ and w is a large integer.

The neural network used for these classification problems for 2 dimensional array patterns is usually a multi-layer neural network with $l \times w$ inputs and some hidden neurons. This network might be very complex, such as the interesting RSONFIN in Juang and Lin (1999) and Wu and Lin (2001), which consists of 6 layers. In this paper, we are not interested in the internal structure, such as the number of hidden neurons or the relations among neurons. Therefore, we represent the neural network as a "black box", illustrated by Fig. 1. We suppose that although it may sometimes come up with errors, the output of the neural network with any input array $\mathbf{x}^{l\times w}$ is expected to agree with the label of the object the input array represents. We call this an undistricted neural network, as opposed to a districted neural network, which we will explain later. An undistricted neural network is usually large, due to the size of the input arrays. Many methods for training, and also for avoiding over-training, have been proposed for undistricted large neural networks. We shall not discuss the details of training in this paper; all we need to emphasize is that the

* Corresponding author. Tel.: +1 250 9605838; fax: +1 250 9605544.
E-mail addresses: lchen@ieee.org (L. Chen), xw@wzu.edu.cn (W. Xue), tokuda_n@sunflare.co.jp (N. Tokuda).
[1] Tel.: +81 3 3355 1383; fax: +81 3 3355 1204.

0893-6080/$ – see front matter
doi:10.1016/j.neunet.2010.03.006

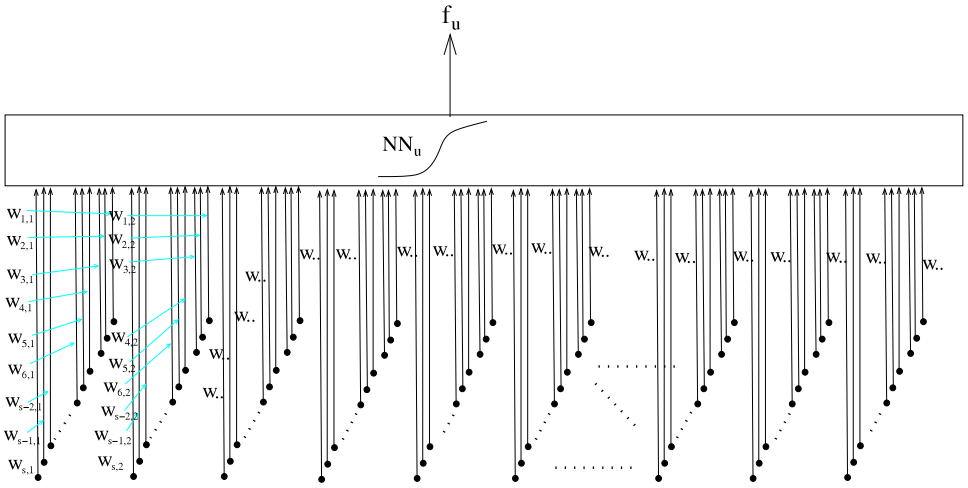

Fig. 1. Undistricted neural network.

training of a neural network with N inputs, when N is reasonably large, is definitely much more difficult than the training of K neural networks of similar structure but with only N/K inputs each.

Following the divide-and-conquer idea, we can partition the grid of input array into non-overlapping regions so as to employ a neural network structure, called a districted neural network, as shown in Fig. 2. A districted neural network consists of two levels of neural networks, some lower level networks and a higher level network. Each of the lower level neural networks, called regional sub-neural networks, takes a block (region) of cells as its inputs; the higher level network, called an assembling sub-neural network, takes the outputs of the regional sub-neural networks as inputs, and derives the final label decision for the object. Each of the sub-neural networks can be trained separately and thus the training is less expensive. The regional sub-neural networks can be trained and performed in parallel and independently, therefore a high speed can be achieved.

We expect that, while it is most likely to come up with many more errors in comparison to the original undistricted large neural network, each regional sub-neural network can independently determine a class label for any input array. That is, for an object denoted as $\mathbf{x}^{l\times w}$ belonging to class C, we expect the output of any regional sub-neural network whose input array is part of $\mathbf{x}^{l\times w}$ *can* agree with the class label C. Of course, this might have many errors because of limited input sizes. But we can also expect that *not all* the regional sub-neural networks come to the wrong conclusion at the same time, therefore the assembling sub-neural network is able to combine the "opinions" of these individual regional sub-neural networks to derive the correct answer.

In this paper, we will discuss the stability of districted and undistricted neural networks, and we will forego the requirement that the region size should be "large" enough, since there is no formal definition of "too small".

Although we believe that the validity of our method does not depend on the particulars of neural networks used as sub-neural networks in the districted neural network, we do not expect that we are able to set up a model for *arbitrary* neural networks. We use only the simplest Feed-Forward Back-Propagation neural network as the sub-neural networks in the analysis. We further only use a simple bi-classification problem for theoretical analysis of the stabilities of districted and undistricted neural networks.

1.2. Relation with previous work in neural networks

It is easy to realize that, after the self-organization is performed, the neocognitron (Fukushima & Miyake, 1982) can be viewed as a special distributed neural network. It has been approved by experiences that the neocognitron has shown improved recognition rates in many applications (Fukushima, 2003). The advantage of the neocognitron can be seen from the neurophysiological findings of the visual nervous system. Neocognitron reflects the hierarchical structure of the general organization of the visual cortex in a series of layers from simple cells to complex cells, to complex feature cells, to complex composite cells and then to view-tuned cells (Riesenhuber & Poggio, 1999). We can see that the distributed neural network shares the same physiological background.

This paper is motivated to show mathematically, but not physiologically, that districted neural networks are actually more stable than undistricted neural networks. We do realize that the idea of ensembling regional sub-neural networks into a large neural network has been successfully applied in face detection. In Rowley, Baluja, and Kanade (1998), the authors proposed a 3 level retinally connected neural network looking at windows of 20×20 pixels for upright face detection. In their implementation, they employ three types of hidden neurons: 4 which look at 10×10 pixel sub-regions, 16 which look at 5×5 pixel sub-regions, and 6 which look at overlapping 20×5 pixel horizontal strips of pixels. Each of these types was chosen to allow the hidden neurons to detect local features that might be important for face detection.

More closely related to this work, Tan, Chen, Zhou, and Zhang (2005) recently developed a face recognition system by partitioning the upright face images into regions of 5 by 5 pixels, using an SOM network in each region, for calculating the "distance between unknown picture blocks and known picture blocks, and then using a simple weighted voting approach. They showed that their system is able to outperform the standard PCA approach for face recognition in AR and FERET face collections. It is easy to see, of course, that the "weighted voting" can be taken as a simple neural network with the results from the SOM network as inputs.

Before any further discussion, we should emphasize here that our analysis is *not* valid for the situations when the elements of an input array are of different data types, e.g., some are the numbers of electrons and some are the values of temperature degrees, nor when the elements are the values of cells distributed in the space without any neighboring relation, e.g., one element is the value of the electron current of a lamp in the White House, another is that of a fan on my desk. The object we are concerned with should be able to be represented by an array of uniform type which corresponds to the cells in a grid. We can see that both the image classification and the nucleotide sequence promoter recognition

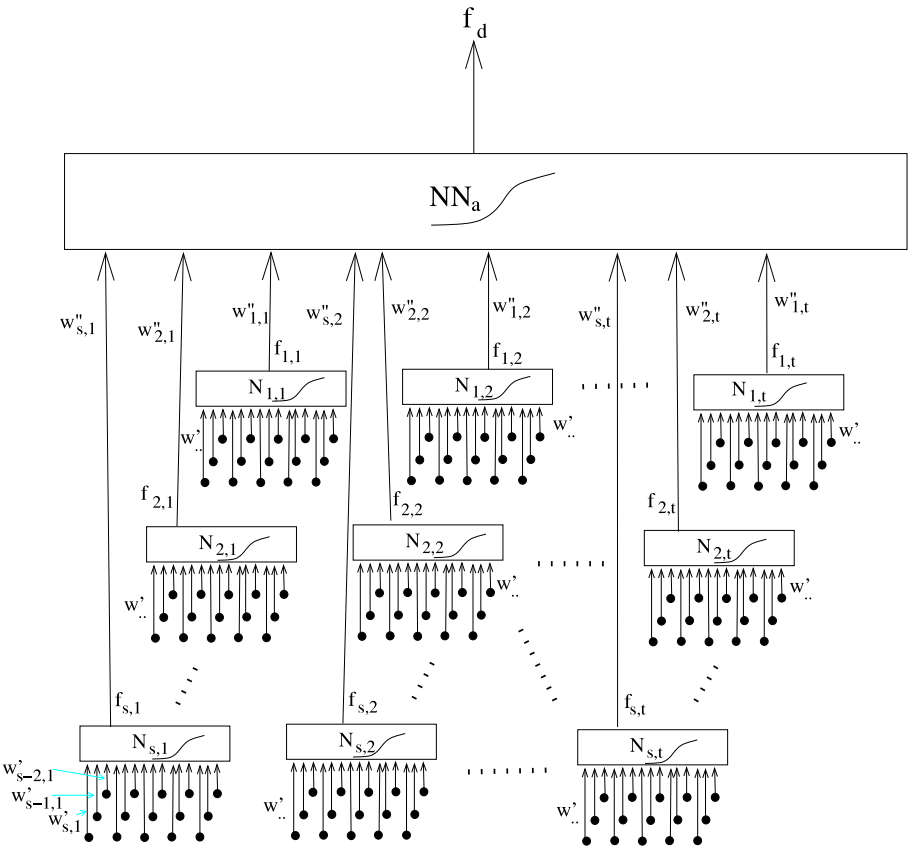

Fig. 2. Districted neural network.

problems mentioned above are good examples of applications of our method.

The districted neural network structure is very similar to the US presidential election system, also called the Electoral College, where the winner of the nation is selected according to a weighed sum of winning states while the winner of a state is decided by a simple majority. An undistracted approach, however, is like the popular election vote system where a *win* simply means getting more votes. In Chen and Tokuda (2005a, 2003b), Chen and etal proved that the Electoral College (referred to as regional voting in these papers) is more stable than the popular election (referred to as national voting then), subject to the restriction that the size of regions is large enough to accommodate the (weak) average distribution assumption. Of course the districted and undistricted neural networks are much more complex than both the regional and national voting systems. In this paper, we will discuss the stability of districted and undistricted neural networks, and we will forego the requirement that the region size should be "large" enough, since there is no formal definition of "too small".

While a districted neural network can be viewed as a divide-and-conquer approach, it is radically different form the M^3 neural network proposed by Lu and Ito (1999), where a multi-label classification problem is partitioned into a series of bi-classification problems, each of which is solved by a neural network. We divide the input vectors while Lu and Ito (1999) partitions the output label sets. But these two approaches are not mutually exclusive; they can be mixed together by first partitioning a class label set with Lu and Ito's approach to generate many bi-classification problems, then solving each of the bi-classification problems using the districted neural network we propose in this paper.

It is worth noting that the districted neural network we propose and discuss is substantially different from the neural network committee machines (Haykin, 1999), sometimes called hierarchical modular neural networks, such as the ensemble-averaging/boosting based committee machine, mixtures of experts (ME) model and hierarchical mixture of experts (HME) model. A neural network committee machine can always be regarded as a divide and conquer strategy, where each input date is taken as a point in high dimensional space, and each of those lower-level sub-neural networks is expected to work best in a certain part of the space. The districted neural network we discuss, on the other hand, partitions the input array into sub-arrays, such that the size of input of each of the lower level sub-neural networks is much smaller than the original input array. The neural network committee machines may reduce the training time complexity because each of the sub-neural network is only required to work best in a certain part of the space. The districted neural network can remarkably reduce the training time complexity, because it is well known that the training of one neural network with large input size is always much more difficult (and time consuming) than the training of many neural network with small sized inputs.

Actually, the neural network ensembling method (Ho, 1998; Kittler, Hatef, Duin, & Matas, 1998; Zhou, Wu, & Tang, 2002) also generates and combines multiple neural networks to improve the accuracy of individual neural networks. Usually these multiple neural networks are obtained through manipulating or partitioning the training data set. The districted neural network we propose in this paper can be taken as a special ensemble neural network, where the constituting neural networks used are obtained by manipulating or partitioning the input arrays. Therefore the sizes of these neural networks are always smaller. The major advantage of the districted neural network is the time complexity reduction in training, as we mentioned before. We do *not* claim, of course, that our approach can be used to replace the general ensembling

method, as the input object we require is special; that is, the input object should be able to be represented by an array of uniform type which corresponds to the values of cells in a grid.

The districted neural network is also substantially different from neural network regression. In the neural network regression approach, although the neural networks used for regression may have different internal structures, the size of input array remains the same for each of the neural networks. The input size of the constituting neural networks for our districted neural network, on the other hand, is always much smaller, as we claimed before.

The remainder of the paper is organized as follows: We will first define a simple Bi-label classification problem for the purpose of neural network performance analysis, then provide a simple model for noise, undistricted neural networks, and districted neural networks for stability analysis in Section 2. We will derive our main theoretical results in Section 3. The theory will be verified by two experiments in Section 4, and the concluding remarks will be given in Section 5.

2. Basic model and assumption

2.1. Bi-label classification problem, undistricted and districted neural network models

2.1.1. Bi-label classification problem description

For the convenience of discussion, we only discuss a bi-label classification problem. But note that the following derivation can also be generalized to M-label classification problems for $M > 2$. We assume the elements of an array representing an object either take the value "+1" or "−1", and suppose the class labels are "+" and "−". We suppose that we have one data set for training and another for testing.

2.1.2. Undistricted and districted neural networks

We discuss only the following two simplified neural networks:
(1) An undistricted neural network as shown in Fig. 1, where the block NN_u denotes a single output neuron with the sign function as its activation function. So, the output of the undistricted neural network on input array $\mathbf{x}^{l\times w}$ is:

$$f_u(\mathbf{x}^{l\times w}) = \begin{cases} +1, & \text{if } \sum_{1\le i\le l, 1\le j\le w} w_{i,j}x_{i,j} > 0, \\ -1, & \text{if } \sum_{1\le i\le l, 1\le j\le w} w_{i,j}x_{i,j} < 0, \\ 0 & \text{otherwise;} \end{cases}$$

where $x_{i,j}$ $(1 \le i \le l,\ 1 \le j \le w)$ is an element $\mathbf{x}^{l\times w}$, and $f_u(\mathbf{x}^{l\times w}) = +1$, or -1, or 0, represents the label for $\mathbf{x}^{l\times w}$ is "+", or "−", or "?" (a tied-up case) respectively.
(2) A districted neural network as shown in Fig. 2, where all the blocks NN_a and $N_{u,v}$'s denote single neurons with sign functions as their activation functions. We suppose the region sizes are $r_l \times r_w$, and assume r_l and r_w divide l and w respectively. Thus, given an input array $\mathbf{x}_{u,v}^{r_l\times r_w}$, which consists of the elements $x_{i,j}$ in $\mathbf{x}^{l\times w}$, $(u-1)\times r_l+1 \le i \le u\times r_l$ and $(v-1)\times r_w \le j \le v\times r_w$, the output of a regional sub-neural network $\mathrm{NN}_{u,v}$ $(1 \le u \le l/r_l,\ 1 \le v \le w/r_w)$ should be:

$$f_{u,v}(\mathbf{x}_{u,v}^{r_l\times r_w}) = \begin{cases} +1 & \text{if } \sum_{\substack{(u-1)\times r_l+1\le p\le u\times r_l \\ (v-1)\times r_w\le q\le v\times r_w}} w'_{p,q}x_{p,q} > 0, \\ -1 & \text{if } \sum_{\substack{(u-1)\times r_l+1\le p\le u\times r_l \\ (v-1)\times r_w\le q\le v\times r_w}} w'_{p,q}x_{p,q} < 0, \\ 0 & \text{otherwise.} \end{cases}$$

With a $\frac{lw}{r_l r_w}$ tuple $(f_{1,1}, \ldots, f_{u,v}, \ldots, f_{l/r_l,w/r_w})$ as its input,[2] the output of the assembling sub-neural network NN_a, should be:

$$f_d(f_{1,1}, \ldots, f_{u,v}, \ldots, f_{l/r_l,w/r_w}) = \begin{cases} +1 & \text{if } \sum_{\substack{1\le p\le l/r_l \\ 1\le q\le w/r_w}} w''_{p,q}f_{p,q}(\mathbf{x}_{p,q}^{r_l\times r_w}) > 0, \\ -1 & \text{if } \sum_{\substack{1\le p\le l/r_l \\ 1\le q\le w/r_w}} w''_{p,q}f_{p,q}(\mathbf{x}_{p,q}^{r_l\times r_w}) < 0, \\ 0 & \text{otherwise.} \end{cases}$$

Notice that $f_{1,1}, \ldots, f_{u,v}, \ldots$, and $f_{l/r_l,w/r_w}$ are outputs of lower level regional sub-neural networks, f_d $(f_{1,1}, \ldots, f_{u,v}, \ldots, f_{l/r_l,w/r_w}) = f_d(f_{1,1}(\mathbf{x}_{1,1}^{r_l\times r_w}), \ldots, f_{u,v}(\mathbf{x}_{u,v}^{r_l\times r_w}), \ldots, f_{l/r_l,w/r_w}(\mathbf{x}_{l/r_l,w/r_w}^{r_l\times r_w})) = +1$, or -1, or 0, which represent the label for $\mathbf{x}^{l\times w}$, which is "+", "−", or "?" (a tied-up case) respectively.

To simplify the terminology, we call a neural network positive definite (or negative define, or indefinite, respectively) for an input if and only if the output of the network is positive (or negative, or indefinite, respectively) with this input.

2.1.3. Training data

(1) Training Data Set for the Undistricted Neural Network

We assume that we have a very large and perfect data set for training purposes so that it covers all the possible situations.[3] We assume an ideal situation here, that is, all $2^{l\times w}$ samples covering all different patterns are included in the training set.[4]

We further assume that the inner relationship between each sample and its label and each sample follows the following equation[5]:

$$L = \begin{cases} \text{“+”}, & \text{if } \sum_{i,j} d_{i,j}x_{i,j} > 0, \\ \text{“−”}, & \text{if } \sum_{i,j} d_{i,j}x_{i,j} < 0, \\ \text{“?”}, & \text{if } \sum_{i,j} d_{i,j}x_{i,j} = 0; \end{cases} \tag{1}$$

where $x_{\cdot,\cdot}$ is an element of array $\mathbf{x}^{l\times w}$, "?" represents the case that the problem is tied-up so that the exact label is unknown[6] and $d_{\cdot,\cdot}$ $(d_{\cdot,\cdot} > 0)$ is the weight of a cell in determining a class label. Although we do not have pre-knowledge of $d_{\cdot,\cdot}$'s and they should not be equivalent, we should assume that they are well distributed in the whole array in the following ways:

- for any subset S of $\{(i,j)|1 \le i \le l, 1 \le j \le w\}$, if $\sum_{(i,j)\in S} x_{i,j} > 0$ (<0, respectively), then most likely, $\sum_{(i,j)\in S} d_{i,j}x_{i,j} > 0$ (<0, respectively); there are more (or less, respectively) samples satisfying $\sum_{(i,j)\in S} d_{i,j}x_{i,j} > 0$ than those satisfying $\sum_{(i,j)\in S} d_{i,j}x_{i,j} < 0$

[2] We can take it as an $l/r_l \times w/r_w$ array, of course.

[3] As mentioned in Introduction, we suppose our neural networks are trained in a noise free laboratory environment.

[4] Theoretically, there are a total of $2^{l\times w}$ different samples. Practically, we can only require that the training set consist of a large part of these samples, in which the following feature is satisfied: these samples in the training set are chosen such that, for any arbitrarily chosen subset of the perfect sample set, the number of elements included in the training set is proportional to the size of the subset. The training set with this feature is called the perfect training set.

[5] It is what the districted and the undistricted neural networks are required to learn from the sample.

[6] Note that, "?" is *NOT* a third label. It simply indicates that an object array (vector) is on the boundary of two classes.

- for any subset S of $\{(i,j)|1 \le i \le l, 1 \le j \le w\}$, $\sum_{(i,j)\in S} d_{i,j} = \sum_{1\le i\le l, 1\le j\le w} d_{i,j} \times \frac{|S|}{lw}$
- any portion of the elements in the input array is *decisive*, that is, for any subset S of $\{(i,j)|1 \le i \le l, 1 \le j \le w\}$, for each $\{x_{i,j}|(i,j) \in S\}$, $x_{i,j} = 1$ or -1, which satisfies $\sum_{(i,j)\in S} d_{i,j}x_{i,j} > 0$ (or <0, respectively), there exists $\{x_{i,j}|(i,j) \notin S\}$, $x_{i,j} = 1$ or -1, such that $\sum_{(i,j)\notin S} d_{i,j}x_{i,j} + \sum_{(i,j)\notin S} d_{i,j}(-x_{i,j}) < 0$ (or >0, respectively), and $\sum_{(i,j)\in S} d_{i,j}x_{i,j} + \sum_{(i,j)\notin S} d_{i,j}x_{i,j} > 0$ (or <0, respectively).

Therefore, we say $d_{i,j}$'s follow the well distribution assumption.

We denote this sample set $\mathcal{U}$. We will use all the data in $\mathcal{U}$ for the training of the undistricted neural network. A sample of the training data should be in the form $(\mathbf{x}^{l\times w}; *)$ where $*$ can be either $+1$, -1, or 0, representing either "+", "−", or "?" (the labels for $\mathbf{x}^{l\times w}$) respectively.

As we assume we have a large and perfect data set as described above, we can prove the following Lemma.

Lemma 1. 1. *The undistricted neural network can be trained perfectly so that the accuracy on the training data set $\mathcal{U}$ is 100%.*
2. *After the undistricted neural network is trained perfectly, all the weights of the connections $w_{i,j} > 0$.*
3. *Suppose, after the undistricted neural network is trained perfectly, for any (i_1, j_1) and (i_2, j_2), if $d_{i_1,j_1} > d_{i_2,j_2}$, then $w_{i_1,j_1} > w_{i_2,j_2}$.*

Brief proof

1. This conclusion can be obtained by setting all $w_{i,j}$ to $d_{i,j}$.
2. This is true because each element $x_{i,j}$ is decisive.
3. When we take $x_{i_1,j_1} = 1$, $x_{i_2,j_2} = -1$, $d_{i_1,j_1}x_{i_1,j_1} + d_{i_2,j_2}x_{i_2,j_2} = d_{i_1,j_1} - d_{i_2,j_2} > 0$. Because the subset $S = \{(i_1, j_1), (i_2, j_2)\}$ of the input array elements is decisive, there exists $\{x_{i,j}|(i,j) \notin S\}$, $x_{i,j} = 1$ or -1, such that $\sum_{(i,j)\notin S} d_{i,j}x_{i,j} + d_{i_1,j_1}x_{i_1,j_1} + d_{i_2,j_2}x_{i_2,j_2} > 0$, and $\sum_{(i,j)\notin S} d_{i,j}x_{i,j} - d_{i_1,j_1}x_{i_1,j_1} - d_{i_2,j_2}x_{i_2,j_2} < 0$. Therefore, $\sum_{(i,j)\notin S} w_{i,j}x_{i,j} + w_{i_1,j_1}x_{i_1,j_1} + w_{i_2,j_2}x_{i_2,j_2} > 0$, and $\sum_{(i,j)\notin S} w_{i,j}x_{i,j} - w_{i_1,j_1}x_{i_1,j_1} - w_{i_2,j_2}x_{i_2,j_2} < 0$. Therefore, $w_{i_1,j_1}x_{i_1,j_1} + w_{i_2,j_2}x_{i_2,j_2} = w_{i_1,j_1} - w_{i_2,j_2} > 0$.

Lemma 1 implies that the weight of a connection is very closely, if not exactly proportional to $d_{i,j}$, especially when $l \times w$ is a large number. Thus, without losing generality, we assume $w_{i,j} = d_{i,j}$ after the undistricted neural network is trained.[7]

(2) Training Data Set for Districted Neural Network

The training set $\mathcal{R}_{u,v}$ for a regional sub-neural network $\mathrm{NN}_{u,v}$ $(1 \le u \le l/r_l, 1 \le v \le w/r_w)$ is constructed as follows: for any $(\mathbf{x}^{l\times w}; *) \in \mathcal{U}$, where $*$ can be either $+1$, or -1 or 0, we obtain a sample $(\mathbf{x}_{u,v}^{r_l\times r_w}; *)$ where $\mathbf{x}_{u,v}^{r_l\times r_w}$ consists of the elements $x_{i,j}$, for all $(u-1)\times r_l + 1 \le i \le u \times r_l$ and $(v-1)\times r_w \le j \le v \times r_w$.

The training set $\mathcal{A}$ for the assembling sub-neural network NN_a is constructed as follows: Firstly, we have all the regional sub-neural networks $\mathrm{NN}_{u,v}$ trained using the training sets $\mathcal{R}_{u,v}$ described above; Then, for any $(\mathbf{x}^{l\times w}; *) \in \mathcal{U}$, where $*$ can be either $+1$, -1 or 0, we place a sample $(f_{1,1}(\mathbf{x}_{1,1}^{r_l\times r_w}), \ldots, f_{u,v}(\mathbf{x}_{u,v}^{r_l\times r_w}), \ldots, f_{l/r_l,w/r_w}(\mathbf{x}_{l/r_l,w/r_w}^{r_l\times r_w}); *)$ in $\mathcal{A}$, where $f_{u,v}(\mathbf{x}_{u,v}^{r_l\times r_w})$ $(1 \le u \le l/r_l, 1 \le v \le w/r_w)$ is the output of the trained regional sub-neural network $\mathrm{NN}_{u,v}$ with input $\mathbf{x}_{u,v}^{r_l\times r_w}$.

It is easy to see that the training set $\mathcal{R}_{u,v}$ for the regional sub-neural network $\mathrm{NN}_{u,v}$ consists of $2^{l\times w}$ samples, and there are many contradictory samples in $\mathcal{R}_{u,v}$. It is easy to prove that for any array $\mathbf{x}_{u,v}^{r_l\times r_w}$, if $\sum_{x_{i,j}\in\mathbf{x}_{u,v}} d_{i,j}x_{i,j} > 0$ (or <0, or $= 0$, respectively), then there are more (or less, or the same number of) samples of the form $(\mathbf{x}_{u,v}^{r_l\times r_w}; +1)$ in $\mathcal{R}_{u,v}$ than those of form $(\mathbf{x}_{u,v}^{r_l\times r_w}; -1)$ in $\mathcal{R}_{u,v}$.[8]

We know that if there are contradictory samples, the output of a well trained neural network follows the majority principle.[9] Thus, we have the following lemma using similar techniques employed for proving Lemma 1.

Lemma 2. 1. *A regional sub-neural network $\mathrm{NN}_{u,v}$ can be trained perfectly so that $f_{u,v}(\mathbf{x}_{u,v}^{l\times w}) = +1$ (or -1, or 0, respectively) if $d_{i,j}\mathbf{x}_{u,v}^{l\times w}$ is larger than (or smaller than, or equal to, respectively) 0.*[10]
2. *After a regional sub-neural network $\mathrm{NN}_{u,v}$ is trained perfectly, all the weights of the connections $w'_{i,j} > 0$.*
3. *After the undistricted neural network is trained perfectly, for any (i_1, j_1) and (i_2, j_2), $w_{i_1,j_1} > w_{i_2,j_2}$ iff $d_{i_1,j_1} > d_{i_2,j_2}$.*

It is also easy to see that the training set $\mathcal{A}$ for the assembling sub-neural network consists of $2^{l\times w}$ samples, and there are again many contradictory samples in $\mathcal{A}$. This can be seen in the following example.

Example: Suppose we have an undistricted neural network with $l = w = 3$, $r_l = 3$, $r_w = 1$ and all the weights $w_{i,j}$ are equal to 1. Let's consider two samples $(\mathbf{a}^{3\times 3}; +1)$, $(\mathbf{b}^{3\times 3}; -1) \in \mathcal{U}$, where all the weights $d_{i,j}$s are equal to 1, and

$$a_{i,j} = \begin{cases} +1, & \text{if } (1 \le i \le 2 \& 1 \le j \le 2) \text{ or } (i = 1 \& j = 3); \\ -1, & \text{otherwise;} \end{cases}$$

and

$$b_{i,j} = \begin{cases} +1, & \text{if } 1 \le i \le 2 \& 1 \le j \le 2; \\ -1, & \text{otherwise.} \end{cases}$$

We know these two samples will come up with two samples $(+1, +1, -1; +1) \in \mathcal{A}$ and $(+1, +1, -1; -1) \in \mathcal{A}$ for the training of the assembling sub-neural network NN_a. These two samples are contradictory, however; we can not expect a neural network to satisfy both. But we can find that there are more samples of the form $(+1, +1, -1; +1)$ than those of form $(+1, +1, -1; -1)$, simply because $+1 + 1 - 1 > 0$. Theoretically then, we have the following lemma.

Lemma 3. *For a given $\frac{lw}{r_l r_w}$ tuple $(f_{1,1}, \ldots, f_{u,v}, \ldots, f_{l/r_l,w/r_w})$, where*

$$\sum_{\substack{1\le u\le l/r_l \\ 1\le v\le w/r_w}} D_{u,v}f_{u,v} \tag{2}$$

>0 (or <0, or $=0$, respectively), the number of samples of the form $(f_{1,1}, \ldots, f_{u,v}, \ldots, f_{l/r_l,w/r_w}; +1)$ is greater than (or less than, or equal to, respectively) that of samples of the form $(f_{1,1}, \ldots, f_{u,v}, \ldots, f_{l/r_l,w/r_w}; -1)$. Here, $D_{u,v} = \sum_{\substack{(u-1)\times r_l+1\le p_{u,v}\le u\times r_l \\ (v-1)\times r_w\le q_{u,v}\le v\times r_w}} d_{p_{u,v},q_{u,v}}$.

This lemma is easily understood: First, all $D_{u,v}$ should be roughly equivalent for all u, v, because of the well distribution feature of $d_{i,j}$. Therefore, Eq. (2) > 0 (or <0, or $= 0$, respectively) means that the number of regions of which the weighted sum of values

[7] Carefully checking the results in our theorems in the later sections, we could know that it actually does *not matter* if $w_{i,j} = d_{i,j}$ is not true for all i and j, as long as the conclusion of item 3 of Lemma 1 is valid.

[8] Basically, when $\sum_{x_{i,j}\in\mathbf{x}_{u,v}} d_{i,j}x_{i,j} > 0$, for each sample of which $\sum_{0\le i\le l, 0\le j\le w} d_{i,j}x_{i,j} < 0$, we can find a sample $\mathbf{x}$, where $x_{i,j} = -x_{i,j}$ for all $x_{i,j} \in \mathbf{x}_{u,v}$, $x_{i,j} = x_{i,j}$ for all other $x_{i,j}$, such that $\sum_{0\le i\le l, 0\le j\le w} d_{i,j}x_{i,j} > 0$.

[9] Alternatively, we can view it as the removal of minority samples before training, when contradiction happens.

[10] In practice, for the case $d_{i,j}\mathbf{x}_{u,v}^{l\times w}$ is equal to 0, the neural network may output either $+1$ or -1 because of the possible errors. But this does not influence the analysis below.

of all the cells is greater than 0 is larger than (or smaller than, or equal to) the number of regions of which the weighted sum of values of all the cells is greater than 0. Then, it is most likely that, in the whole grid, the weighted sum of all the cells is larger than (or smaller than, or equal to, respectively) 0. Notice that a sample in $\mathcal{U}$ in the form of $(\cdots\cdots; +1)$ (or $(\cdots\cdots; -1)$, or $(\cdots\cdots; 0)$, respectively), will come up with a sample of the form $(\cdots\cdots; +1)$ (or $(\cdots\cdots; -1)$, or $(\cdots\cdots; 0)$, respectively) for the training of the assembling neural network NN_a. Therefore, the lemma is correct.

The detailed proof is omitted.

Using Lemma 3, and the techniques used in proving Lemmas 1 and 2, we can come up with.

Lemma 4. 1. *An assembling sub-neural network can be trained perfectly so that* $f_d(f_{1,1}, \ldots, f_{u,v}, \ldots, f_{l/r_l,w/r_w}) > 0$ *(or* <0*, or* $= 0$*, respectively) iff* $f_{1,1} + \cdots + f_{u,v} + f_{l/r_l,w/r_w} > 0$ *(or* <0*, or* $= 0$*, respectively).*

2. *After an assembling sub-neural network is trained perfectly, all the weights of the connections* $w''_{i,j} > 0$.
3. *After the assembling neural network is trained perfectly, for any* (u_1, v_1) *and* (u_2, v_2), $w_{u_1,v_1} > w_{u_2,v_2}$ *iff* $D_{u_1,v_1} > D_{u_2,v_2}$.

Because of Lemmas 2 and 4, without losing generality, we can assume $w''_{u,v} = D_{u,v}$ and $w'_{i,j} = 1$ for all possible $w''_{u,v}$ and $w'_{i,j}$. Because all $D_{u,v}$ are roughly equivalent, according to the well distribution assumption, we further assume $w''_{u,v} = 1$.

2.2. Stability criterion for stability analysis

As we are estimating stability, we suppose, without losing generality, that an object $\mathbf{y}^{l\times w}$ to be classified is a "+" object, and it is very close to the critical point. That is, the number of elements in $\mathbf{y}^{l\times w}$ taking the value "+1" is very close to the number of elements taking the value "−1". We let α and β denote the proportions of elements taking the values "+1" and "−1" respectively in the absence of noise. Immediately, we have

(1) $\alpha + \beta = 1$;
(2) $1 > \alpha > \beta > 0$.[11]
(3) $\alpha - \beta$ is close to 0.

We will provide the formal definition of noise in Section 2.3. We are to find the upper bound of the "amount" of noise that can change the label selections of districted and undistricted neural networks.

Because we consider the input array to have a reasonably large number of elements, we could regard α and β as the probability that an element has the value of "+1" and "−1" respectively, when we arbitrarily choose an element in the array $\mathbf{y}^{l\times w}$. This implies that in a large subset consisting of elements in $\mathbf{y}^{l\times w}$, the ratio of "+1" elements should remain the same throughout the whole vector.

The two label classification system we adopt in this paper is not as restrictive as it looks, it is not difficult to see that the conclusions are still valid for a multi-label model involving three or more class labels.

2.3. Noise definition

1. The noise is defined as a change of environment that forces a change of classification label. When subjected to noise, the values of some of the cells will undergo a change from "+1" to "−1", some from "−1" to "+1", and others may remain unchanged. The noise that influences values of cells to change from "+1" to "−1" (or "−1" to "+1") is called anti-positive-noise (or anti-negative-noise). A cell whose value undergoes a change from "+1" to "−1" (or "−1" to "+1") is called an anti-positive-noise-contaminated cell (or anti-negative-noise-contaminated cell).
2. Two types of noise caused by independent, known or unknown, sources are considered: concentrated noise, which influences the values of cells within a concentrated block of the grid, and white noise, which is distributed uniformly and randomly over the whole grid.
3. A set of anti-positive-white noise (or anti-negative-white noise, respectively) is dispersed uniformly over the grid, producing a uniform chance of converting values of cells from "+1" to "−1" (or "−1" to "+1", respectively). The result of white noise thus could be regarded as a change in the probability of a cell taking value "+1" from α to a new value, and a change in probability of a cell taking value "−1" from β to a new value accordingly.[12]
4. We call a region concentrated/white noise polluted *if and only if* there is at lease one cell is concentrated/white noise contaminated. We call it concentrated/white noise free otherwise. When a region is concentrated/white noise polluted (or free) we call the regional sub-neural network for it a concentrated/white noise polluted (or free) regional sub-neural network.
5. A set of anti-positive-concentrated noise (or anti-negative-concentrated noise, respectively) is defined as the union of non-overlapping rectangle blocks of size $n_l \times n_w$, on each of which each cell taking the value "+1" (or "−1", respectively) will be changed to "−1" (or "+1"). The corresponding union of these rectangle areas is called a noise concentrated area and $n_l \times n_w$ is called the size of noise blocks.[13]
6. In accordance with the above two types of noise, the anti-positive-noise-contaminated cells (or anti-negative-noise-contaminated cells, respectively) comprise two different types depending on the noise type, namely anti-positive-concentrated-noise-contaminated cells (or anti-negative-concentrated-noise-contaminated cells, respectively) and anti-positive-white-noise-contaminated cells (or anti-negative-white-noise-contaminated cells, respectively).

   Notice that when both of white noise and concentrated noise coexist, some noise-contaminated cells may belong to both of these types, as will be seen in the proof of theorems in Section 3.

   It is reasonable that the noise concentrated area should be viewed as reasonably large. Since white noise is dispersed uniformly over the grid, a ratio of white noise contaminated cells in the noise concentrated area should not change from that in the whole grid.
7. For the interests of the lower bounds of the stabilities of the neural networks, we throughout this paper consider only the anti-positive-noise in the analysis. Thus when we refer to noise, concentrated noise, white noise, or contaminated cells hereafter, anti-positive-noise, anti-positive-concentrated noise, anti-positive-white noise, anti-positive-noise contaminated cells respectively are implied.

[11] In a later section, we will see that $f_u = f_d = +$, which means that both districted and undistricted neural networks shall classify $\mathbf{y}^{l\times w}$ as a positive object in a noise free environment.

[12] It is obvious that the union of a set of white noise is also a set of white noise.

[13] Intuitively, the "white noise" is isolated and scattered randomly over discrete "points" of the grid while "concentrated noise" is distributed over connected, continuous areas which may be randomly distributed across the grid.

2.4. Assumption

We always assume that the number of cells in the whole grid is large and the amount of noise is large so that both the total number of noise contaminated cells and the noise concentrated area are large.

Basic Assumption: In the absence of concentrated noise, the proportion of positive (negative) definite regional sub-neural networks among all the regional sub-neural networks, or among a set of a large number of arbitrarily chosen regional sub-neural networks, which may or may not be neighbors, is equivalent to the chance that more cells take values of "+1" than those of "−1" (or more cells take values of "−1" than those of "+1") in a region.

This assumption implies.

Lemma 5. *In the absence of both white and concentrated noise, among a set of a large number of arbitrarily chosen regional neural networks, the proportions of positive definite, negative definite and indefinite regional sub-neuron networks, denoted by P_+, P_-, and $P_?$, can be computed by:*

$$P_+ = \sum_{y=0}^{\left\lfloor \frac{r_l r_w - 1}{2} \right\rfloor} \binom{r_l r_w}{y} \beta^y \alpha^{r_l r_w - y};$$

$$P_- = \sum_{y=0}^{\left\lfloor \frac{r_l r_w - 1}{2} \right\rfloor} \binom{r_l r_w}{y} \alpha^y \beta^{r_l r_w - y};$$

$$P_? = \begin{cases} 0, & \text{if } r_l r_w \text{ is an odd number}, \\ \binom{r_l r_w}{r_l r_w/2} \alpha^{r_l r_w/2} \beta^{r_l r_w/2}, & \text{otherwise}. \end{cases}$$

Lemma 6. *In the presence of only white noise, in a set of a large number of arbitrarily chosen regional sub-neural networks, the ratios of positive definite, negative definite and indefinite regional sub-neural networks, denoted as P'_+, P'_- and $P'_?$ can be computed by:*

$$P'_+ = \sum_{y=0}^{\left\lfloor \frac{r_l r_w - 1}{2} \right\rfloor} \binom{r_l r_w}{y} \beta'^y \alpha'^{r_l r_w - y};$$

$$P'_- = \sum_{y=0}^{\left\lfloor \frac{r_l r_w - 1}{2} \right\rfloor} \binom{r_l r_w}{y} \alpha'^y \beta'^{r_l r_w - y};$$

$$P'_? = \begin{cases} 0, & \text{if } r_l r_w \text{ is an odd number}, \\ \binom{r_l r_w}{r_l r_w/2} \alpha'^{r_l r_w/2} \beta'^{r_l r_w/2} & \text{otherwise}; \end{cases}$$

where α' and β' are the ratios of the cells taking value of " +1 " or " −1 " respectively, in the presence of white noise.

Of course, we have $P_+ + P_- + P_? = 1$, and $P'_+ + P'_- + P'_? = 1$.

The fact that $P_+ > P_-$ iff $\alpha > \beta$, simply indicates that in a noise free environment, both the undistricted neural network and the districted neural network select the same label "+" for $\mathbf{x}^{l\times w}$.

We should regard the set of concentrated noise polluted regions as a set of "arbitrarily" chosen regions discussed in the above Basic Assumption.

3. Theorems, conclusions and conjecture

We shall let $\aleph_c$ and $\aleph_w$ denote the number of concentrated-noise contaminated cells and the number of white-noise contaminated cells respectively.

3.1. The stability of undistricted neural network

Theorem 1. *In the presence of white and concentrated noise, the output of the undistricted neural network will be:*

$$+1, \quad \text{if } \aleph_c + \aleph_w - \frac{\aleph_c \aleph_w}{\alpha N} < \frac{\alpha - \beta}{2} \times N;$$

$$0, \quad \text{if } \aleph_c + \aleph_w - \frac{\aleph_c \aleph_w}{\alpha N} = \frac{\alpha - \beta}{2} \times N;$$

$$-1, \quad \text{if } \aleph_c + \aleph_w - \frac{\aleph_c \aleph_w}{\alpha N} > \frac{\alpha - \beta}{2} \times N.$$

Proof. Since the "+1" cells constitute α portion of all cells in the grid, the total number of "+1" cells is αN. When the number of concentrated-noise-contaminated cells is $\aleph_c$, the portion of concentrated-noise-contaminated cells among all "+1" cells is $\frac{\aleph_c}{\alpha N}$; When the number of white-noise-contaminated cells is $\aleph_c$, the portion of white-noise-contaminated cells among all "+1" cells is $\frac{\aleph_c}{\alpha N}$. Considering Basic Assumption, we know that $\frac{\aleph_c}{\alpha N} \times \frac{\aleph_c}{\alpha N} \times \alpha N = \frac{\aleph_c \aleph_w}{\alpha N}$ cells are overlapped between the set of concentrated-noise-contaminated cells and the set of white-noise-contaminated cells. The undistricted neural network outputs "+1" ("0", or "+1") if and only if the number of overall noise-contaminated cells is *less than* (*equal to*, or *greater than*) $\frac{\alpha-\beta}{2} \times N$.

3.2. The stability of districted neural networks

Theorem 2. *The districted neural network will output*

(1) *" +1 ", if:*

$$\aleph_c < \frac{\frac{n_l n_w}{r_l r_w}}{\left(\left\lceil \frac{n_l - 1}{r_l} \right\rceil + 1\right)\left(\left\lceil \frac{n_w - 1}{r_w} \right\rceil + 1\right)} \times \frac{P_+(\aleph_w) - P_-(\aleph_w)}{1 + P_+(\aleph_w) - P_-(\aleph_w)} \cdot \alpha \cdot N,$$

and

$$\aleph_w < (\alpha - \beta)/2 \times N;$$

(2) *" −1 " if:*

$$\aleph_c > \frac{P_+(\aleph_w) - P_-(\aleph_w)}{1 + P_+(\aleph_w) - P_-(\aleph_w)} \cdot \alpha \cdot N,$$

or

$$\aleph_w > (\alpha - \beta)/2 \times N;$$

where $P_+(\aleph_w)$ and $P_-(\aleph_w)$ denote the proportions of positive definite and negative definite neural networks in the presence of white noise, and can be calculated as

$$P_+(\aleph_w) = \sum_{y=0}^{\left\lfloor \frac{r_l r_w - 1}{2} \right\rfloor} \binom{r_l r_w}{y} (\beta + \aleph_w/N)^y (\alpha - \aleph_w/N)^{r_l r_w - y},$$

$$P_-(\aleph_w) = \sum_{y=0}^{\left\lfloor \frac{r_l r_w - 1}{2} \right\rfloor} \binom{r_l r_w}{y} (\alpha - \aleph_w/N)^y (\beta + \aleph_w/N)^{r_l r_w - y}.$$

Proof. Let S_r denote the total size of concentrated noise polluted regions within the grid.

Noticing that a noise block of size $n_l \times n_w$ can be partitioned into, at most $(\lceil \frac{n_l - 1}{r_l} \rceil + 1)(\lceil \frac{n_w - 1}{r_w} \rceil + 1)$ different regions, at least $\frac{n_l n_w}{r_l r_w}$, we have:

$$1 \le \frac{S_r}{\aleph_c/\alpha} \le \left(\left\lceil \frac{n_l - 1}{r_l} \right\rceil + 1\right)\left(\left\lceil \frac{n_w - 1}{r_w} \right\rceil + 1\right) \frac{r_l r_w}{n_l n_w}. \tag{3}$$

Let $X\,(0 \le X \le 1)$ denote the proportion of concentrated noise polluted regional sub-neural networks among all regional sub-neural networks.

(1) If $\aleph_w < (\alpha-\beta)/2 \times N$, it is easy to see that $P_+(\aleph_w) > P_-(\aleph_w)$, and thus the districted neural network will remain positive definite if concentrated noise is not present.

Assume

$$\aleph_c < \frac{\frac{n_l n_w}{r_l r_w}}{\left(\left\lceil \frac{n_l-1}{r_l}\right\rceil+1\right)\left(\left\lceil \frac{n_w-1}{r_w}\right\rceil+1\right)} \times \frac{P_+(\aleph_w)-P_-(\aleph_w)}{1+P_+(\aleph_w)-P_-(\aleph_w)} \cdot \alpha \cdot N,$$

the inequality (3) indicates:

$$S_r < \frac{P_+(\aleph_w)-P_-(\aleph_w)}{1+P_+(\aleph_w)-P_-(\aleph_w)}.$$

We have

$$X = \frac{S_r}{N} < \frac{P_+(\aleph_w)-P_-(\aleph_w)}{1+P_+(\aleph_w)-P_-(\aleph_w)}.$$

According to Lemma 6, we know that a $P_+(\aleph_w)$ portion of all the concentrated-noise free regional sub-neural networks are positive definite and a $P_-(\aleph_w)$ portion are negative definite. In the presence of an X portion of concentrated noise polluted regional sub-neural networks, by regarding all the concentrated-noise polluted regional sub-neural networks as negative definite,[14] the total number of positive definite regional sub-neural networks should be

$$(1-X)P_+(\aleph_w) > \frac{P_+(\aleph_w)}{1+P_+(\aleph_w)-P_-(\aleph_w)};$$

the total number of negative defined regional sub-neural networks should be

$$\begin{aligned} X+(1-X)P_-(\aleph_w) &= P_-(\aleph_w)+X(1-P_-(\aleph_w)) \\ &< \frac{P_-(\aleph_w)(1+P_+(\aleph_w)-P_-(\aleph_w))+(P_+(\aleph_w)-P_-(\aleph_w))(1-P_-(\aleph_w))}{1+P_+(\aleph_w)-P_-(\aleph_w)} \\ &= \frac{P_+(\aleph_w)}{1+P_+(\aleph_w)-P_-(\aleph_w)}. \end{aligned}$$

It is clear here that the number of positive definite regional sub-neural networks should be greater than the number of negative definite regional sub-neural networks. Therefore the districted neural network should be able to remain positive definite. This ends the proof of item (1) of Theorem 2.

(2) If $\aleph_w > (\alpha-\beta)/2 \times N$, it is easy to see that $P_+(\aleph_w) < P_-(\aleph_w)$; thus the districted neural network will be negative positive definite if concentrated noise is not present.

Assume

$$\aleph_c > \frac{P_+(\aleph_w)-P_-(\aleph_w)}{1+P_+(\aleph_w)-P_-(\aleph_w)} \cdot \alpha \cdot N,$$

considering the total size of concentrated noise polluted regions within the grid cannot be less than the total size of the noise concentrated area, the left inequality in inequality (3) holds.

Thus

$$S_r > \frac{P_+(\aleph_w)-P_-(\aleph_w)}{1+P_+(\aleph_w)-P_-(\aleph_w)},$$

and

$$X = \frac{S_r}{N} < \frac{P_+(\aleph_w)-P_-(\aleph_w)}{1+P_+(\aleph_w)-P_-(\aleph_w)}.$$

In this situation, in the presence of an X portion of concentrated noise polluted regional sub-neural networks, all concentrated-noise polluted regional sub-neural networks will be negative definite. Therefore, the total number of positive definite regional sub-neural networks should be

$$(1-X)P_+(\aleph_w) < \frac{P_+(\aleph_w)}{1+P_+(\aleph_w)-P_-(\aleph_w)};$$

the total number of negative defined regional sub-neural networks should be

$$\begin{aligned} X+(1-X)P_-(\aleph_w) &= P_-(\aleph_w)+X(1-P_-(\aleph_w)) \\ &> \frac{P_-(\aleph_w)(1+P_+(\aleph_w)-P_-(\aleph_w))+(P_+(\aleph_w)-P_-(\aleph_w))(1-P_-(\aleph_w))}{1+P_+(\aleph_w)-P_-(\aleph_w)} \\ &= \frac{P_+(\aleph_w)}{1+P_+(\aleph_w)-P_-(\aleph_w)}. \end{aligned}$$

Therefore the districted neural network will be negative definite. This ends of the proof of the conclusion of item (2) of Theorem 2.

3.3. Average stability of the districted neural network

In item (1) of Theorem 2 above, the ceiling operations are used to develop a sufficient condition of stability that accounts for the worst possible condition, whereby each of the noise blocks pollutes a maximum number of regions and regional sub-neural networks. We shall see that, by average, a Districted Neural Network can accommodate more concentrated-noise contaminated cells than the lower boundary in item (1) of Theorem 2. However, it is unlikely that the worst situation for each noise block will happen at the exactly same time. Some appropriate averaging will be introduced here by shifting the partitions.

The following theorem shows the averaged result.

Theorem 3. *By average, a districted neural network will output "+1" if:*

$$\aleph_c < \frac{n_l n_w}{(r_l+n_l-1)(r_w+n_w-1)} \cdot \frac{P_+(\aleph_w)-P_-(\aleph_w)}{1+P_+(\aleph_w)-P_-(\aleph_w)} \cdot \alpha \cdot N$$

and

$$\aleph_w < \frac{1}{2}(\alpha-\beta)N$$

where $P_+(\aleph_w)$ and $P_-(\aleph_w)$ are calculated as they are in Theorem 2.

Proof. Geometrically, we regard the pair of opposing edges along the outer boundary of the rectangular grid as "glued" together, so that they are able to glide onto the other end as we move across the boundary, allowing a total of $r_l \times r_w$ different partitions by merely shifting the horizontal and vertical boundaries of the regions. Now, with all the $r_l r_w$ different partitions, we have a total of $r_l r_w \times N/(r_l r_w) = N$ different regions.

Let S_r denote the average of the total size of concentrated noise polluted regions within the grid, $X\,(0 \le X \le 1)$ be the average of the proportion of concentrated noise polluted regions among all the regions for a districted neural network.

For each $n_l \times n_w$ sized noise block, we enumerate all the possible $r_l \times r_w$ partitions by the positions of their intersections relative to the noise block. For a partition through the intersection $p\,(0 \le p \le r_l-1)$ cells above the bottom edge of the noise block and q $(0 \le p \le r_l-1)$ cells right to the left edge of the noise block, it divides the block into $(\lceil \frac{p}{r_l}\rceil + \lceil \frac{n_l-p}{r_l}\rceil) \times (\lceil \frac{q}{r_w}\rceil + \lceil \frac{n_w-q}{r_w}\rceil)$ regions.

[14] Indeed, it is quite possible that some of them still remain positive definite or only turn out to be indefinite in practice.

It is not difficult to prove that,

$$\sum_{p=0}^{r_l-1}\sum_{q=0}^{r_w-1}\left(\left\lceil\frac{p}{r_l}\right\rceil+\left\lceil\frac{n_l-p}{r_l}\right\rceil\right)\times\left(\left\lceil\frac{q}{r_w}\right\rceil+\left\lceil\frac{n_w-q}{r_w}\right\rceil\right)$$
$$=(n_l+r_l-1)(n_w+r_w-1).$$

Therefore, a noise block of size $n_l \times n_w$ will be divided into $(n_l + r_l - 1)(n_w + r_w - 1)$ different regions for all the $r_l \times r_w$ different partitions. By average, the block will be divided into $\frac{(n_l+r_l-1)(n_w+r_w-1)}{r_l\times r_w}$ different regions. Accordingly, we have

$$\frac{S_r}{\aleph_c/\alpha}=\frac{(n_l+r_l-1)(n_w+r_w-1)}{n_l\times n_w},$$

and

$$X=S_r/N=\frac{(n_l+r_l-1)(n_w+r_w-1)}{n_l\times n_w}\times\aleph_c/\alpha N.$$

When

$$\aleph_c<\frac{n_ln_w}{(r_l+n_l-1)(r_w+n_w-1)}\cdot\frac{P_+(\aleph_w)-P_-(\aleph_w)}{1+P_+(\aleph_w)-P_-(\aleph_w)}\cdot\alpha\cdot N,$$

immediately we have

$$X<\frac{P_+(\aleph_w)-P_-(\aleph_w)}{1+P_+(\aleph_w)-P_-(\aleph_w)}.$$

As shown in the proof for Theorem 2, the above inequality indicates the number of positive definite regional sub-neural networks should be greater than the number of negative definite regional sub-neural networks. Thus, the districted neural network should be able to remain positive definite, and we have proved the theorem.

3.4. Conclusions

Considering a near equilibrium case of $\alpha - \beta = 0.03$ as it is assumed (see Section 2.2), and using the conclusion of Theorem 3 for districted neural network so as to see an averaged situation, we have Fig. 3 to illustrate the number of noise-contaminated cells that districted and undistricted neural networks can accommodate before the original label selection "+" is reversed.

We see that the number of noise contaminated cells that a districted neural network can accommodate increases continuously, as the size of subdivided regions decreases. Until up to a certain limit, beyond which the stability margin starts to decrease, becoming asymptotic to an undistricted neural network limit where the improvement in stability from localizing the effects of noise into regional sub-neural networks is minimized.

From Fig. 3, it seems that for very large regions with little white noise, the stability margin for the districted neural network looks smaller than that of the undistricted neural network for concentrated noise.

We shall see that, in fact, Theorem 2 leaves the case of

$$\frac{\frac{n_ln_w}{r_lr_w}}{\left(\left\lceil\frac{n_l-1}{r_l}\right\rceil+1\right)\left(\left\lceil\frac{n_w-1}{r_w}\right\rceil+1\right)}\cdot\frac{P_+(\aleph_w)-P_-(\aleph_w)}{1+P_+(\aleph_w)-P_-(\aleph_w)}\cdot\alpha\cdot N$$
$$\le\aleph_c\le\frac{P_+(\aleph_w)-P_-(\aleph_w)}{1+P_+(\aleph_w)-P_-(\aleph_w)}\cdot\alpha\cdot N;$$

and

$$\aleph_w<(\alpha-\beta)/2\times N;$$

undecided.

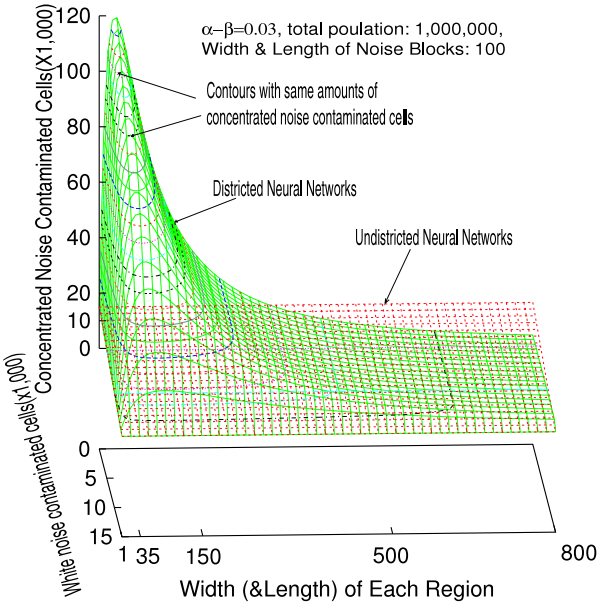

Fig. 3. Numbers of white & concentrated noise contaminated cells that the districted and undistricted neuron networks can accommodate.

Even taking into consideration of the "averaged case" as demonstrated by Theorem 3, the case that

$$\frac{n_ln_w}{(r_l+n_l-1)(r_w+n_w-1)}\cdot\frac{P_+(\aleph_w)-P_-(\aleph_w)}{1+P_+(\aleph_w)-P_-(\aleph_w)}\cdot\alpha\cdot N$$
$$\le\aleph_c\le\frac{P_+(\aleph_w)-P_-(\aleph_w)}{1+P_+(\aleph_w)-P_-(\aleph_w)}\cdot\alpha\cdot N;$$

and

$$\aleph_w<(\alpha-\beta)/2\times N;$$

is still undecided.

In such a situation, whether the districted neural network remains positive definite depends on the number of concentrated-noise polluted regional sub-neural networks that still remain positive. We can find that, in the proofs of Theorems 2 and 3, we always regard all the concentrated noise polluted regional sub-neural network as negative definite so that only the regional sub-neural networks that remain entirely *clean* are counted as keeping their positive definite state. In fact, many of the concentrated noise polluted regional sub-neural networks still remain positive definite or only turn out to be indefinite. Evidently this is most serious when the size of regions is large, as we can see that a positive definite regional sub-neural network has very little chance to be transformed to negative definite when only a few cells of the inputs change their signs. We could see that if the size of regions is close to the size of the grid, the districted neural network will be close to an undistricted neural network again. This implies that if the effect of over-estimation of the concentrated noise polluted regional sub-neural networks is properly taken into account, the stability margin will increase so that the surface representing the districted neural network closing to the right-back corner of Fig. 3 will move up slightly from the peak point to the right end. Thus, we conclude that the districted neural network is always more stable than the undistricted neural network, even when the size of the regions is very large. In addition, the districted neural network and the undistricted neural network will become identical when the size of regions is as small as 1 or as large as that of the grid.

3.5. Conjecture

In the previous sections, we used a simple BP neural network as the undistricted neural network, and also as the sub-neural networks that constitute the districted neural network; we only use a simple 2-classification problem as the pattern recognition/classification to be solved. We believe, however, that the above conclusion about districted and undistricted neural networks still remains valid, even when more complicated neural network structures are involved, and more complicated pattern recognition/classification problems are to be solved, so long as the objects to be classified are represented as 2-dimensional fixed sized arrays of uniform type which correspond to cells in a 2-dimensional grid.

4. Experiments

The experiments we show here are gender classification and human face recognition, where neural network approach has been extensively used (e.g. Garcia & Delakis, 2004; Mäkinen & Raisamo, 2008; Rowley et al., 1998; Tan et al., 2005). We should emphasize here that although there are many approaches, e.g. Turk and Pentland (1991) and Gordon, Chervonenkis, Gammerman, Shahmuradov, and Solovyev (2003), that can work better than neural networks for these applications, we in this paper only intend to show districted neural networks are better than their undistricted versions. It is *not* our purpose to compare our approach to other methods. We use these experiments simply for validating our theory, and showing that our conjecture is correct.

4.1. Gender classification

4.1.1. Data set

We use the FERET subsets[15] used in Mäkinen and Raisamo (2008) for experiment: The Training Set consists of 304 FERET face images, an equal number of both genders; Two sets are used for testing, The Test Set includes 107 FERET forefront face images, 60 males and 47 females; The Pose Test Set contains the face images of 112 subjects, an equal number of both genders, each with 9 images representing the poses of the subject rotating $-60°$, $-45°$, $-30°$, $-15°$, $0°$, $15°$, $30°$, $45°$ and $60°$ out-of-plane. In our experiment, the images are first resized into 24×24 pixels, where the line between two eyes has fixed length and is parallel to the horizontal line.

4.1.2. Detailed methods and results

Two types of neural networks, multilayer feedforward networks and linear networks, are used for the experiments.

(1) Feedforward network structure.

We implemented an undistricted neural network with 3 layers, 24×24 inputs, 2 hidden units[16] and 1 output. A positive output represents "male", and a negative "female".

The districted neural network was constructed as follows: we divided the each image into K ($K = (24/r)^2$) regions of size $r \times r$ ($r = 2, 3, 4, 6, 8, 12$). We implemented a 3-layer neural network (regional sub-neural network) for each region. A regional sub-neural network had $r \times r$ inputs, 2 hidden neurons and 1 outputs. The assembling sub-neural network used all the outputs of regional sub-neural networks as inputs, that is, it had K inputs; the number of the hidden neurons for a assembling sub-neural network was always 2 again.

Table 1
Accuracy of gender classification for test set.

	Undistricted neural network	Size of regions for districted NNs				
		2×2	3×3	4×4	6×6	8 × 8
Linear network	0.523	0.888	0.850	0.876	0.720	0.710
Feedforward network	0.822	0.879	0.822	0.888	0.757	0.832

Table 2
Accuracy of gender classification for pose test set.

	Undistricted neural network	Size of regions for districted NNs				
		2×2	3×3	4×4	6×6	8 × 8
Linear network	0.551	0.657	0.650	0.665	0.629	0.641
Feedforward network	0.568	0.705	0.622	0.640	0.573	0.622

We used the linear transfer function (purelin) as the activation function for each neuron of hidden layers of the undistricted neural network, of the regional sub-neural networks and of the assembling sub-neural network; The hyperbolic tangent sigmoid transfer function (tansig) was used as the activation function for the output neuron of undistricted and districted neural networks, and those of regional sub-neural networks. The undistricted neural network, as well as each of the sub-neural networks in the districted neural network, was trained by the implementation of the Levenberg–Marquart algorithm (trainlm). 80% of the training images were used for training and 20% for validation; an "early stopping" technique was used so that the training was terminated if the network performance on the validation vectors fails to improve or remains the same for 5 epochs in a row. The "early stopping" technique was aimed to improve the generalization and avoid over-fitting. Each of undistricted neural networks, regional sub-neural networks, and assembling sub-neural networks is trained 100 times, we choose only the one with highest performance on the validation set.

The trained undistricted and districted neural networks were used on the test set and the pose test set, the results are shown in Tables 1 and 2.

(2) Linear network structure.

Two layer linear networks (newlind) of Matlab were used directly as the undistricted neural network, regional sub-neural network and assembling sub-neural network. The undistricted neural network has 24×24 inputs and 1 output where a positive output represents "male", and a negative "female". A regional sub-neural network had $r \times r$ inputs ($r = 2, 3, 4, 6, 8, 12$, $r \times r$ corresponds with the region sizes) and 1 outputs. The assembling sub-neural network used all the outputs of regional sub-neural networks as inputs, that is, it had K ($K = (24/r)^2$) inputs; it has 1 output too.

The trained undistricted and districted neural networks were used on the test set and the pose test set, the results are also shown in Tables 1 and 2.

We can clearly see that the districted neural network approach performs much better than the undistricted matching version, which verifies our theorems, and roughly, as the size of regions gets smaller, the performance gets better up to a limit, after which the performance starts to decrease.

4.2. Face recognition

These experiments were done for the human face recognition problem. Usually, mainly because of the registration problem, face recognition is based on features extracted from facial images

[15] http://www.cs.uta.fi/hci/mmig/vision/datasets/.

[16] We tested different number of hidden units and found that 2 is always the best for undistricted neural network for this experiments in terms of accuracy.

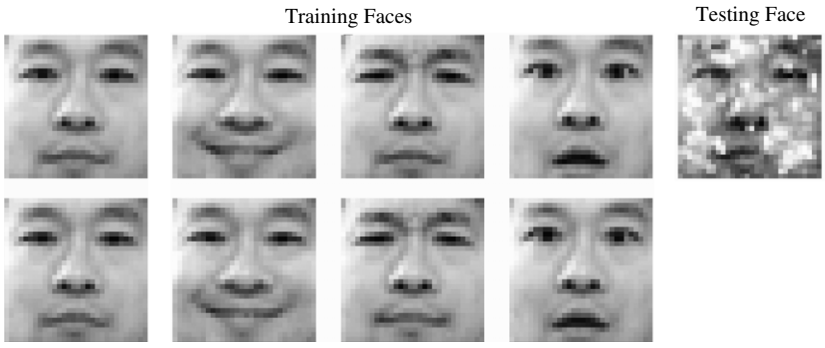

Fig. 4. Examples of training and testing faces.

rather than using pixels directly. However, half-automatic face recognition approaches based on pixel information directly do work well, as long as the face images are pre-processed by cropping each picture into a fixed size and rotating each face into a standard position. Successful works using pixel based matching can be found in Ikeda, Watta, Artiklar, and Hassoun (2001) and Mu, Artiklar, Hassoun, and Watta (2003).

4.2.1. Data set

A picture collection of 20 people is used for the experiment. Each people has 2 sets of 4 gray-scale pictures showing different facial expressions: blank, smile, angry and surprised. We cropped each picture into a face of size 30×30 pixels where the line between two eyes has fixed length and is parallel to the horizontal line, as shown in Fig. 4, and stored them in raw format. We used all these photos as the training data. We obtained by adding noise using Photoshop 6.0, a set of 20 noise polluted pictures of these 20 people, then cropped them into faces of size 30×30 satisfying the requirement on the line between two eyes. Several typical training faces and a testing face are shown in Fig. 4.

4.2.2. Detailed methods and results

We implemented an undistricted neural network with 3 layers, 30×30 inputs, 60 hidden units and 20 outputs. These 20 outputs represent 20 different people. Each of them is used for predicating whether the photo to be classified belongs to a certain person in the data set. The output of the network is interpreted by believing the output neuron with the highest score. The districted neural network was constructed as follows: we divided the each image into $K = (30/r)^2$ regions of size $r \times r$ ($r = 3, 5, 10, 15$) each. We implemented a 3-layer neural network (regional sub-neural network) for each region. A regional sub-neural network had $r \times r$ inputs, $2r$ hidden neurons and 20 outputs. The assembling sub-neural network used all the outputs of regional sub-neural networks as inputs, that is, it had $20 \times K$ inputs; the number of the hidden neurons for a assembling sub-neural network was always $2 \times r$.

We used the hyperbolic tangent sigmoid transfer function (tansig) as the activation function for each neuron of hidden layers of the undistricted neural network, of the regional sub-neural networks and of the assembling sub-neural network. The log-sigmoid transfer function (logsig) was used as the activation function for each output neuron of undistricted and districted neural networks, and those of regional sub-neural networks.

The undistricted neural network, as well as each of the sub-neural networks in the districted neural network, was trained by the implementation of the back-propagation algorithm with adaptive learning rate (traingda). During the training process, an "early stopping" technique was used to improve the generalization and avoid over-fitting. If the error did not decrease in 10 consecutive epochs, the training of the neural network was terminated to avoid over fitting.

Table 3
Matching result of human faces.

Undistricted neural network	Size of regions for districted NNs				
	2×2	3×3	5×5	10×10	15×15
7	7	8	19	15	12

The undistricted and districted neural networks were trained 20 times, and the trained networks were used on the test set, the best results are shown in Table 3.

We see that the experiments showed a phenomenon that is very close to our theory.

5. Further work and open problems

Some interesting neural network schemes may emerge from our districted neural network to extend the applicability of the method to a wider range of decision making processes involving 2D array patterns. For example, it seems to be a highly exciting subject to pursue the analysis of a multi-level districted neural network where the regional sub-neural networks in a districted neural network are recursively replaced by a districted neural network by recursively partitioning each of the regions into smaller (sub) regions.

Recently, Cao, Murata, ichi Amari, Cichocki, and Takeda (2003, 2002) have successfully used a PCA-based pre-whitening technique to reduce noise before further separation of different signal components for Magnetoencephalography (MEG) data analysis. It may be very useful, and substantially increase accuracy to use a similar technique to reduce the noise of a pattern before employing a districted neural network to determine its class label. As the advantage of a districted neural network comes from fact that it can localize the effects of concentrated noise into a restricted number of regional sub-neural networks, and it does not show any advantage if only white noise is present. A careful analysis of the features of remaining component should be interesting and important if we want to use a noise reduction as the first stage of classification.

Acknowledgement

The authors appreciate the help of Mr. R. Chen at Jilin University in implementing the experiments in Section 4.2. A preliminary version of this paper was published in Proceedings of 2005 International Joint Conference on Neural Network, Montréal, Québec, Canada. (Chen, 2005). The research of the first author is supported by a Discovery Grant of NSERC, Canada.

References

Cao, J., Murata, N., ichi Amari, S., Cichocki, A., & Takeda, T. (2003). A robust approach to independent component analysis of signals with high-level noise measurements. *IEEE Transactions on Neural Networks*, *14*(3), 631–645.

Cao, J., Murata, N., ichi Amari, S., Cichocki, A., & Takeda, T. (2002). Independent component analysis for unaveraged single-trial meg data decomposition and single-dipole source localization. *Neurocomputing*, *49*(1–4), 255–277.

Chen, L. (2005). Pattern classification by assembling small neural networks. In *Proceedings of 2005 international joint conference on neural networks*. (pp. 1947–1952).

Chen, L., & Tokuda, N. (2005a). A general stability analysis on regional and national voting schemes against noise. *Artificial Intelligence*, *163*(1), 47–66.

Chen, L., & Tokuda, N. (2003b). Stability analysis of regional and national voting schemes by a continuous model. *IEEE Transactions on Knowledge and Data Engineering*, *15*(4), 1037–1042.

Fukushima, K. (2003). Neocognitron for handwritten digit recognition. *Neurocomputing*, *51*(April), 161–180.

Fukushima, K., & Miyake, S. (1982). Neocognitron: a new algorithm for pattern recognition tolerant of deformations and shifts in position. *Pattern Recognition*, *15*(6), 455–469.

Garcia, C., & Delakis, M. (2004). Convolutional face finder: a neural architecture for fast and robust face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *26*(11), 1408–1423.

Gordon, L., Chervonenkis, A. Y., Gammerman, A. J., Shahmuradov, I. A., & Solovyev, V. V. (2003). Sequence alignment kernel for recognition of promoter regions. *Bioinformatics*, *19*, 1964–1971.

Haykin, S. (1999). *Neural networks C A comprehensive foundation*. Prentice Hall.

Ho, T. K. (1998). The random subspace method for constructing decision forests. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *20*(8), 832–844.

Ikeda, N., Watta, P., Artiklar, M., & Hassoun, M. H. (2001). A two-level Hamming network for high performance associative memory. *Neural Networks*, *14*(9), 1189–1200.

Juang, C. F., & Lin, C. T. (1999). A recurrent self-organizing neural fuzzy inference network. *IEEE Transactions on Neural Networks*, *10*(4), 828–845.

Kittler, J., Hatef, M., Duin, R. R. W., & Matas, J. (1998). On combining classifiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *20*(3), 226–239.

Lu, B.-L., & Ito, M. (1999). Task decomposition and module combination based on class relations: a modular neural network for pattern classification. *IEEE Transactions on Neural Networks*, *10*(5), 1244–1256.

Mäkinen, E., & Raisamo, R. (2008). Evaluation of gender classification methods with automatically detected and aligned faces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *30*(3), 543–547.

Mu, X., Artiklar, M., Hassoun, M. H., & Watta, P. (2003). An RCE-based associative memory with application to human face recognition. In *Proceedings of the international joint conference on neural networks 2003*: *Vol. 4* (pp. 2552–2557). Portland, Oregon: IEEE Press, July.

Riesenhuber, M., & Poggio, T. (1999). Hierarchical models of object recognition in cortex. *Nature Neuroscience*, *2*, 1019–1025.

Rowley, H., Baluja, S., & Kanade, T. (1998). Neural network-based face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *20*(1), 23–38.

Tan, X., Chen, S., Zhou, Z.-H., & Zhang, F. (2005). Recognizing partially occluded, expression variant faces from single training image per person with SOM and soft knn ensemble. *IEEE Transactions on Neural Networks*, *16*(4), 875–886.

Turk, M., & Pentland, A. (1991). Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, *3*(1), 71–86.

Wu, G. D., & Lin, C. T. (2001). A recurrent neural fuzzy network for word boundary detection in variable noise-level environments. *IEEE Transactions on Systems, Man, and Cybernetics Part B*, *31*(1), 84–97.

Zhou, Z.-H., Wu, J., & Tang, W. (2002). Ensembling neural networks: many could be better than all. *Artificial Intelligence*, *137*(1–2), 239–263.