2015 IEEE International Conference on Multimedia Big Data

Model of Human Visual Cortex Inspired Computational Models for Visual Recognition

Jinjun Wang, Qiqi Hou, Nan Liu, Shizhou Zhang
Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University
28 West Xianning Road, Xi'an, Shaanxi, China, 710049
jinjun@mail.xjtu.edu.cn

Abstract—**In this paper, we are mostly interested in investigating how the study and discovery of the human visual cortex could be utilised to improve the computational models for visual recognition by computer vision. Many of the brain perceptual abilities in vision have corresponding algorithms exist in computer vision, and in this paper we discuss three such models. First we present a model that has the ability for iterative bottom-up/top-down recognition, and experimental results on applying the model for facial landmark detection has shown improved accuracy over benchmark approaches. Second we introduce a new SOM model that could be deep and invariant, which could achieve significantly improved digit recognition accuracy over traditional SOM. And third we show how the convolutional neural network could be combined with linear coding based architecture, where experimental results show that the proposed model could outperform many existing algorithms for image classification.**

I. Introduction

Research of human brains has huge potential and key importance for the development of human society, economy, culture, education, science and technology. Numerous research projects of human brain have been founded from all over the world. DARPA's Mind-Controlled Prosthetic Arm works like a regular arm, with the ability to bend, rotate, and twist in 27 different ways. The Human Brain Project [1] funded by the European Union is by far the largest research project in neuroscience. It is a major step towards understanding the human brain by building a full computer model of a functioning brain and simulating the complete human brain on supercomputers. The US BRAIN Initiative [2] aims at mapping the activity of every neuron in the human brain. The Nature special, Turing at 100 [3], has pointed out that the fusion of human intelligence and machine intelligence is a new way of realising artificial intelligence.

In China, the study of human brain perception and brain-machine interaction keeps attracting attentions from government, academy and industry. The first research institute of cognitive science was founded in 1988. Research of cognitive brain science is one of the key directions in the 8∼11th Five-Year Plan for National Economic and Social Dev. The National Med. & Long-term Plan for Sci. & Tech Dev. (2006∼2020) has listed the brain and cognitive science as one of the key scientific issues. In 2001, China became participant of the global Human Brain Project.

In this paper, we are mostly interested in investigating how discoveries in human brain research, especially the model of human visual cortex, could be utilized to improve the computational models for visual recognition by computer vision. Many of the brain perceptual abilities in vision have corresponding algorithms exist in computer vision. Amongst those representative examples, the disparity is one of the first computational models of visual cortex [4], [6], [7]. Since then, a large number of models of the human visual cortex have been reported, including models of edge detection [5], spatio-temporal interpolation and approximation, computation of optical flow and direction selectivity [8], [9], computation of lightness and albedo, shape form contours/texture/shading, binocular stereo matching [4], structure from motion/stereo, surface reconstruction [10] and filling-in [11], surface color [12], the orientation tuning of simple cells in the primary visual cortex (V1) [13], the properties of motion sensitive neurons in visual area for motion processing (MT) [14]. Reader could refer to [15] for a more complete survey.

This paper introduces three additional computational models that borrows ideas and principles from the study in model of human visual cortex. Specifically, first we present a model that has the ability for iterative bottom-up/top-down recognition, second we introduce a new SOM model that could be deep and invariant, and third we show how the convolutional neural network could be combined with linear coding based architecture to improve classification accuracy.

II. Facial Landmark Detection Via Iterative Bottom-up/Top-down process

Most existing computer vision models for object recognition proceeds through a cascade of hierarchically layers with computations at each successive stage being feedforward (bottom-up). In fact, the top-down effects are key to normal everyday vision, and backprojections are also likely to be a key part of what cortex is computing and how [15]. In human visual cortex, the effect of cortical feedback will be seen after some time (¿100ms), which means that computational models that simulate only the bottom-up process are actually performing what human vision system is doing in the first 100ms, and therefore lacking the ability to deal with more complex scenes.

In this paper, we present a multi-channel convolutional neural network for facial landmark detection. The model works in an iterative fashion where the initial locations of multiple patches are fed into the bottom-up path of the model to

978-1-4799-8688-0/15 $31.00 © 2015 IEEE
DOI 10.1109/BigMM.2015.29

IEEE computer society

calculate the corrections of landmark coordinates. The top-down path then guesses new landmark coordinates, and such bottom-up/top-down process iterates until convergency.

Let $\mathbf{S} \in \mathbb{R}^{2*p}$ be the coordinates of facial landmarks in an image I, where p denote the number of facial landmarks. In this paper, we refer to the vector $\mathbf{S}$ as a shape. denote $\mathbf{S}^t$ the the estimate of $\mathbf{S}$ at stage t, denote $R^t$the regressor at stage t. The ground truth shape is $\hat{\mathbf{S}}$.

In training, with N training samples $\{I_i, \hat{\mathbf{S}}_i, \mathbf{S}_i^0\}_{i=1}^N$, we want to reduce the alignment errors on training set, The t stage regressor is formally learnt as follows

$$R^t = \underset{R^t}{argmin} \sum_{i=1}^{N} \left\| \hat{\mathbf{S}}_i - \mathbf{S}_i^{t-1} - R^t(I_i, \mathbf{S}_i^{t-1}) \right\|_2 \tag{1}$$

where $\mathbf{S}_i^{t-1}$ is the estimated shape in the previous stage $t-1$. Note all shape in our experiments are normalized by meanshape like ESR[18].

In testing, with a facial image I and an initial shape $\mathbf{S}^0$, we will update face shape in a top-down manner:

$$\mathbf{S}_i^t = \mathbf{S}_i^{t-1} + R^t(I_i, \mathbf{S}_i^{t-1}) \tag{2}$$

The stage regressor computes $\Delta\mathbf{S}^t$ based on the previous shape $\mathbf{S}^{t-1}$ and image I. In this framework, shape $\mathbf{S}$ should be more and more close to the ground truth shape $\hat{\mathbf{S}}$ though cascade regressing.

We take CNN as our regressor. The regressor takes the raw pixels as input and performs regression on the location of landmarks. Two convolutional layer are stacked after the input layer. Finally a fully connected layer connect all local convolutional layer together.

For convolutional layer, we use Rectified Linear Units (ReLUs) as our neurons. Then convolutional layer can be represented as follow:

$$O_{i,j,k} = max(\sum_{x=0}^{h-1} \sum_{y=0}^{w-1} \sum_{z=0}^{c-1} I_{i-x,j-y,z} \cdot \mathbf{W}_{x,y,z,k} + \mathbf{B}_k, 0) \tag{3}$$

where I is the input to the convolutional layer, O is the output, $\mathbf{W}$ is the weight and $\mathbf{B}$ is the bias. h, w, c denote the width, height and channel of filter. k means that is the k^{th} filter.

Pooling layers in the network can summarize the output of neighboring groups of neurons in the same kernel map. we used max pooling non-overlapping pooling regions. Finally, we use Euclidean-loss which computes:

$$loss = \frac{1}{2N} \sum_{i=1}^{N} \left\| O_i - \Delta\mathbf{S}_i \right\|_2 \tag{4}$$

where N is the batch-size, O is the network output, $\Delta\mathbf{S}$ is the difference between ground truth shape and current shape.

We conducted experiments on the 300 Face in-the-Wild Challenge dataset (300-W) [16]. Following the protocol suggested by [17], our training set consist of 3148 images in total. The testing set has two parts, specifically the common subset and the challenging subset. The former consists of 544 images in total, while the later has 135 images. As can be seen from Table II, our method outperformed these method by incorporating both the local feature in a bottom-up process and the global constraint in a top-down process into a generic regression framework. Figure II shows some examples.

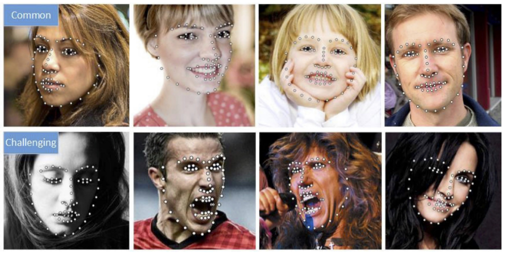

Fig. 1. Example results from the 300-W dataset

TABLE I
COMPARISON OF FACIAL LANDMARK DETECTION ERROR

Method	Full	Common	Challenging
ESR [18]	7.58	5.28	17.00
SDM [19]	7.52	5.60	15.40
LBF [17]	6.32	4.95	**11.98**
Our method	**6.30**	**4.91**	12.03

III. DEEP SELF-ORGANIZING MAP

The self-organizing map[20] has often been described as a form of non-linear principle component analysis [21]. It is able to map a structured, high-dimensional signal manifold onto a much lower-dimensional network in an orderly fashion. In addition, the obtained map tends to preserve the topological information, which very closely resembles what was found in the cortices of brains. However, one major limitation of SOM is due to its shallow structure that is not capable of performing high-level feature abstraction. Biology study has revealed that human could describe visual concepts in hierarchical ways, and the mammal brain is organized in a deep architecture, with given input percept represented at multiple levels of abstraction. In this paper, we introduce a novel Deep SOM (DSOM) algorithm which performs visual classification from directly raw image pixels input. Similar to other deep models, DSOM consists of layers of alternating self-organizing layer and sampling layer as elaborated next.

For an $M \times M$ sized input, either the gray image input (the first layer) or the sampling layer output, the self-organizing layer uses multiple maps to model the input pattern, each focusing on a $K \times K$ sub-region, called a patch, (the color frame on digit two in Figure 2) from the input. Assuming the stride is s, then there will be $N_{map} \times N_{map}$ maps and

$$N_{map} = ceil\big((M-K)/s\big) + 1 \tag{5}$$

DSOM uses T modes in each map to model the patterns from the corresponding image patch. To elaborate, assuming patches from position $\{p, q\}$ are to be modelled by a map with the set of neuron weights denoted as $\{w_{1,p,q}^l, ..., w_{T,p,q}^l\}$ where the superscript l indicates the layer index. The patch

$x_{p,q}$ extracted from the input updates the map by firstly find the winning neuron j^* by

$$j^* = \arg\min_j ||\mathbf{x}_{p,q} - \mathbf{w}^l_{j,p,q}(t)||^2 \,. \tag{6}$$

and then adjust the weights in the neighborhood of j^* by

$$\begin{aligned}\mathbf{w}^l_{j,p,q}(t+1) =& \mathbf{w}^l_{j,p,q}(t) + \eta(t)\alpha(t,j,j^*)\big(\mathbf{x}_{p,q} - \mathbf{w}^l_{j,p,q}(t)\big) \\ & \forall j \in \mathcal{N}_{j^*} \,, \end{aligned} \tag{7}$$

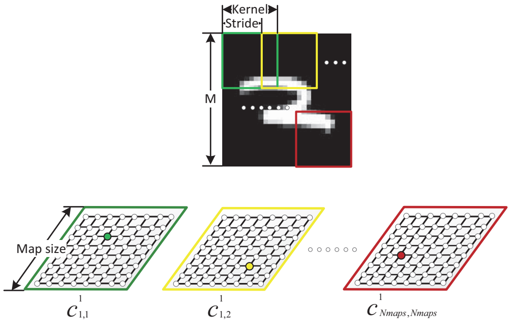

Fig. 2. Illustration of the self-organizing layer

Once the preceding self-organizing layer decomposes the input into $N_{map} \times N_{map}$ maps, next a sampling layer is applied to combine information from these $N_{map} \times N_{map}$ maps into another 2D grid of neurons, with another higher level of abstraction. In order to do that, the set of patch $\{x_{p,q}\}, \forall 1 \le p,q \le N_{map}$ will be used again to find each corresponding winning neuron index $\{j^l_{p,q}\}, \forall 1 \le p,q \le N_{map}$. These winning neuron index values are aligned onto another 2D grid with $N_{map} \times N_{map}$ neurons (Figure 3), and thus obtaining the sampling layer output.

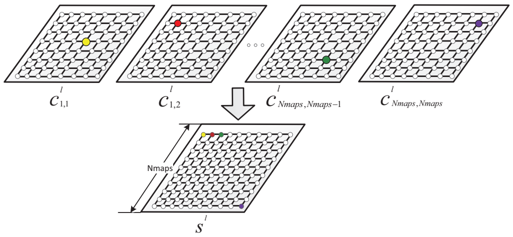

Fig. 3. Illustration of the sampling layer

The output of the sampling layer has identical form as the input gray image, and therefore a second self-organizing layers can be concatenated. In this way, the proposed DSOM algorithm allows a deep structured SOM to be constructed. During the training phase, the proposed DSOM is trained layer by layer starting from the bottom layer first.

We evaluated the performance of our proposed DSOM method on the MNIST [22] dataset and the birds dataset [23], and used the supervised SOM (SKN [24]) as the benchmark.The MNIST dataset is a handwritten digits databset which consists of a training set with 60,000 examples and a test set with 10,000 examples. Every digit in the dataset has been size-normalized to 28 × 28 pixels and centered. The birds dataset contains 600 images in six classes, specifically egret, mandarin duck, snowy owl, puffin, toucan and wood duck, each with 100 samples. Since the images have variable resolution, in the experiment, they are resized to 50 × 50 and the gray level image are used. It is seen from Table II that, our proposed DSOM outperforms SKN on both datasets.

TABLE II
COMPARISON OF OBJECT RECOGNITION ACCURACY

Method	MNIST	Birds
SKN	89%	42.57%
Our method	**96.17%**	**49.33%**

IV. Deep Sparse Coding Network

Conceptually, the convolution kernels in CNN can be considered on the analogy of a codebook used by BoW. A pixel generates large response only when it is highly correlated (similar) to the corresponding kernel. This process is similar to VQ but unfortunately, the zero-order nature of VQ may result in very low response when none of the kernel is close enough to the pixel. In case the response gets maxed out in pooling, the generated feature maps may not fully resemble the input and hence leads to information loss. In this paper, we propose a method by constructing Deep Sparse-coding Net (DeepSCNet) to marry the advantages of both CNN and sparse-coding techniques. The proposed algorithm has four type of building blocks in its deep architecture, as illustrated in Figure 4: *Sparse-coding layer*, *Pooling layer*, *Normalization layer* and *Map reduction layer*. Compared to CNN, training DeepSCNet is relatively easier even with training set of moderate size.

Sparse-coding layer: Denoting the receptive field at spatial location $i,\ j$ as $\mathbf{X}_{ij}$ which is an $C \times w \times h$ cube, where C is the number of channels, and $w \times h$ represents a small rectangular region centering at $i,\ j$. For the image input layer, C is the number of color channels, while for the feature map layer, C is the number of feature maps. For consistency, we columnize $\mathbf{X}_{ij}$ as $\mathbf{x}_{ij} \in \mathbb{R}^D$ where $D = C \times w \times h$. The sparse-coding layer embeds $\mathbf{x}_{ij}$ into $\mathbf{c}_{ij}$ through the following coding process

$$\begin{aligned}\mathbf{c}_{ij} = \arg\min_{\mathbf{c}} \frac{1}{2}||\mathbf{x}_{ij} - \mathbf{W}\mathbf{c}_{ij}||^2 \,, \\ s.t.\ ||\mathbf{c}_{ij}||_{L_0} \le \mathbf{K}\end{aligned} \tag{8}$$

where the weight $\mathbf{W} = [\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_M] \in \mathbb{R}^{D \times M}$ is a codebook which is similar to a collection of M convolution kernels in CNN. $\mathbf{c}_{ij}$ is M-dimensional, and we put the m^{th} element $\mathbf{c}^m_{ij}$ into the m^{th} feature map at location $i,\ j$. For input of size $W \times H$, the size of each feature map is also padded to $W \times H$. $\mathbf{K}$ controls the sparsity of $\mathbf{c}_{ij}$ by upper bounding the number of non-zero entries. In practise, the L_0 constraint can be relaxed by other coding schemes summarized in [25].

Pooling layer and Normalization layer: The pooling and the normalisation operation are fairly similar to that in CNN.

Map Reduction layer: The dictionary used during sparse-coding is usually over-complete, *i.e.* $M > D$. As the model becomes “deep”, the number of feature maps grows exponentially because D grows exponentially. In order to remedy

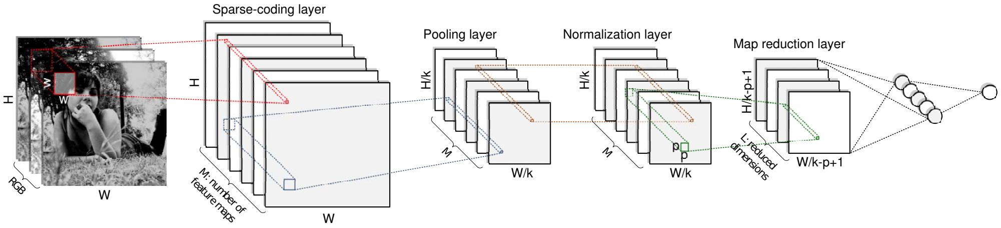

Fig. 4. Illustration of Deep Sparse-Coding Network (DeepSCNet)

the disadvantage, it is necessary to have a certain layer that reduces the dimensionality of input data. DeepSCNet consists of a map reduction layer which is a full connection layer that reduces the number of feature maps M (*i.e.* dimension of $\mathbf{n}$ from the previous normalization layer) to L, and $L < M$. This could be implemented using PCA, and instead we train a deep auto-encoder for the purpose and remain the "unsupervised" nature of the overall model.

We evaluate the performance of the proposed DeepSCNet with three public benchmarks: the MITScene-67 dataset [26], the UIUC Sports Event dataset [27], and the Scene-15 dataset [28]. As can be seen from Table III, DeepSCNet achieved promising accuracy for all datasets.

TABLE III
COMPARISON OF SCENE RECOGNITION ACCURACY

Method	MITScenes-67	UIUC Sports	15-scenes
KSPM [28]	-	-	81.40
ScSPM [29]	36.9	82.74	80.28
LScSPM [30]	-	85.31	-
RBoW [31]	37.9	-	-
HMP [32]	41.8	85.7	-
VC [33]	46.4	84.8	**83.4**
Our method	**49.4**	**87.1**	82.7

V. Conclusion

This paper introduces three computational models for visual recognition, inspired by the model of human visual cortex. First we present a model that has the ability for iterative bottom-up/top-down recognition. Second we introduce a new SOM model that could be deep and invariant. And third we show how the convolutional neural network could be combined with linear coding based architecture. Experimental results show that the proposed models could outperform many existing algorithms for visual recognition tasks.

VI. Acknowledgement

This work is supported by the National Science Foundation of China under Grant No. 61473219.

References

[1] http://en.wikipedia.org/wiki/Human_Brain_Project
[2] http://en.wikipedia.org/wiki/BRAIN_Initiative
[3] http://www.nature.com/news/specials/turing/index.html
[4] D. Marr and T. Poggio, "Cooperative computation of stereo disparity", Science 194(4262): 283-7, 1976
[5] D. Marr and T. Poggio, "A computational theory of human stereo vision." Proc. of R Soc Lond B Biol Sci 204(1156): 301-28, 1979
[6] N. Qian, "Computing stereo disparity and motion with known binocular cell properties", Neural Computation, 6(3), 390404, 1994
[7] J. Read, J. Parker and B. Cumming, "A simple model accounts for the response of disparity-tuned V1 neurons to anticorrelated images." Visual Neurosci 19(6): 735-753, 2002
[8] S. Ullman, "The interpretation of visual motion." MIT Press, 1979
[9] Marr, D. and S. Ullman, "Directional selectivity and its use in early visual processing." Proc. R Soc Lond B 211(1183):151-180, 1981
[10] W. Grimson, "A computational theory of visual surface interpolation", Philos Trans R Soc London Ser B 298:395-427, 1982
[11] S. Ullman, "Filling in the gaps: the shape of subjective contours and a model for their generation", Biol Cybern, 25:1-6, 1976
[12] A. Hurlbert, "The computation of color", PhD Thesis. Harvard Medical School and Massachusetts Institute of Technology, Department of Brain and Cognitive Sciences, 1989
[13] A. Teich and N. Qian, "Comparison among some models of orientation selectivity." J Neurophysiol 96(1): 404-19, 2006
[14] N. Rust, V. Mante, et al., "How MT cells analyze the motion of visual patterns." Nat Neurosci 9(11): 1421-31, 2006
[15] doi:10.4249/scholarpedia.3516
[16] C. Sagonas, et al, "A semi-automatic methodology for facial landmark annotation", CVPR'13, pp. 896–903, 2013
[17] S. Ren, X. Cao, Y. Wei, and J. Sun, "Face alignment at 3000 fps via regressing local binary features",
[18] X. Cao, Y. Wei, F. Wen, and J. Sun, "Face alignment by explicit shape regression", IJCV, vol. 107, no. 2, pp. 177-190, 2014
[19] X. Xiong and F. Torre, "Supervised descent method and its applications to face alignment", CVPR'13, pp. 532-539, 2013
[20] T. Kohonen, "The self-organizing map", Proceedings of the IEEE, vol. 78, no. 9, pp. 1464-1480, 1990
[21] I. Jolliffe, "Principal component analysis", Wiley Online Library, 2005
[22] E. Kussul and T. Baidyk, "Improved method of handwritten digit recognition tested on mnist database", Image and Vision Computing, vol. 22, no. 12, pp. 971-981, 2004
[23] S. Lazebnik, C. Schmid, and J. Ponce, "A maximum entropy framework for part-based object recognition", ICCV'05, pp. 832-838, 2005
[24] W. Melssen, R. Wehrens, and L. Buydens, "Supervised kohonen networks for classification problems", Chemometrics and Intelligent Laboratory Systems, vol. 83, no. 2, pp. 99-113, 2006
[25] J. Wang and Y. Gong, "Discovering Image Semantics in Codebook Derivative Space", TMM, 2012
[26] A. Quattoni and A. Torralba, "Recognizing indoor scenes", 2009
[27] L.-J. Li and L. Fei-Fei, "What, where and who? classifying events by scene and object recognition",ICCV'07, pp. 1-8, 2007
[28] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories",CVPR'06, pp. 2169-2178, 2006
[29] J. Yang, K. Yu, Y. Gong, and T. Huang, "Linear spatial pyramid matching using sparse coding for image classification", CVPR'09, pp. 1794-1801, 2009
[30] S. Gao, I. W. Tsang, L.-T. Chia, and P. Zhao, "Local features are not lonely–laplacian sparse coding for image classification",CVPR'10, pp. 3555-3561, 2010
[31] S. N. Parizi, J. G. Oberlin, and P. F. Felzenszwalb, "Reconfigurable models for scene recognition", CVPR'12, pp. 2775–2782, 2012
[32] L. Bo, X. Ren, and D. Fox, "Hierarchical matching pursuit for image classification: Architecture and fast algorithms", NIPS'11, pp. 2115–2123, 2011
[33] Q. Li, J. Wu, and Z. Tu, "Harvesting mid-level visual concepts from large-scale internet images", CVPR'13, pp. 851–858, 2013