International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:12, No:10, 2018

Optimizing the Capacity of a Convolutional Neural Network for Image Segmentation and Pattern Recognition

Yalong Jiang, Zheru Chi

***Abstract*—**In this paper, we study the factors which determine the capacity of a Convolutional Neural Network (CNN) model and propose the ways to evaluate and adjust the capacity of a CNN model for best matching to a specific pattern recognition task. Firstly, a scheme is proposed to adjust the number of independent functional units within a CNN model to make it be better fitted to a task. Secondly, the number of independent functional units in the capsule network is adjusted to fit it to the training dataset. Thirdly, a method based on Bayesian GAN is proposed to enrich the variances in the current dataset to increase its complexity. Experimental results on the PASCAL VOC 2010 Person Part dataset and the MNIST dataset show that, in both conventional CNN models and capsule networks, the number of independent functional units is an important factor that determines the capacity of a network model. By adjusting the number of functional units, the capacity of a model can better match the complexity of a dataset.

***Keywords*—**CNN, capsule network, capacity optimization; character recognition, data augmentation; semantic segmentation.

I. Introduction

RECENTLY, deep learning aiming to discover and automatically learn good representations from raw data with a complex hierarchical model, has attracted great attention. The benefits are brought by the high capacity or Vapnik-Chervonenkis dimension [1]-[4] of models. Despite the improvements in accuracy, there are some major challenges in learning with CNNs: the requirement for high computational resources, the heavy reliance on training data and the lack of theoretical understanding of CNNs.

The major contribution of this paper is the development of a method to partition a CNN layer into independent functional units. The partition is based on the similarity among convolutional kernels and the Expectation-Maximization Algorithm (EM) proposed in [5]. Both conventional CNN models and the capsule network proposed in [6] can be partitioned into functional units, as is addressed in Section III. The contributions of functional units to the overall performance are evaluated and only units that are useful for the task are kept. The proposed method improves efficiency while maintaining the performance. In addition, a method for increasing the complexity of the dataset by enlarging variances is proposed to match the datasets with different complexity to the models with different capacity. Apart from the reduction in over-fitting and the improvement in efficiency, the method also provides a novel way to understand CNNs.

Our previous work on semantic segmentation [7] is based on the PASCAL VOC 2010 Person Part dataset [8] and Deeplab proposed in [9]. The proposed method for adjusting capacity is conducted on the Deeplab model.

We perform experiments on semantic segmentation using the model optimized by the proposed method. The experiments reported are conducted on the PASCAL VOC 2010 Person Part dataset and the MNIST dataset [10]. Training accuracy and test accuracy of models with different numbers of functional units are evaluated. Comparisons on the optimized network and some benchmark methods have shown that the model with the appropriate number of functional units performs the best on a specified task. Moreover, experiments are conducted to adjust the capacity of the capsule network to match to datasets with different capacity.

The rest of the paper is organized as follows. Section III discusses a method for partitioning a conventional CNN into functional units. Adjusting the capacity of a conventional CNN model or a capsule network by only using the most useful functional units is also discussed in this section. In Section IV, we propose a method for increasing the complexity of a dataset and the way of matching the capacity of a model to the complexity of a dataset. Section V reports and discusses our experimental results. The concluding remarks are drawn in Section VI.

II. Related Work

Matching the capacity of deep neural architectures to the complexity of tasks has become an area of active research.

Current work can be grouped into five categories. The first category focuses on increasing the capacity of a model by increasing the number of trainable parameters. The second type of algorithms focuses on enriching the types of operations in a CNN model. The methods of the third type focus on computing the mutual information among hidden activations in a CNN model. Related research topics include developmental learning [11]-[13] and lifelong learning [14]-[17]. The fourth category involves network pruning [18] and optimization on connectivity [19], [20]. The fifth category focuses on designing architectures of neural networks with genetic algorithms or reinforcement learning [21]. However, the first three methods can only increase models' capacity. Therefore, over-fitting is

Yalong Jiang is with the Department of Electronic and Information Engineering, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong (e-mail: yalong.jiang @ connect.polyu.hk).

Zheru Chi is with the Department of Electronic and Information Engineering, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong.

easy to occur. Also these methods cannot even qualitatively evaluate the capacity of a model or match the capacity of a model to the complexity of a dataset. There is the need for an effective way to find the optimal capacity of a CNN for a specific task. The fourth method requires significant computational resources during training, and the fifth method produces networks that are quite complex and the process of designing networks is time-consuming.

In this paper, we provide a point of view to understand CNNs and match the capacity of CNNs to the complexity of tasks. The proposed method improves the efficiency of inference, as compared to the first three methods. Also the method requires less time and computational resources during training, as compared to the fourth and fifth methods.

III. Methodology

A. Conventional CNNs

In this section, we propose to adjust the number of functional units to control the capacity of the Deeplab model proposed in [9]. In a CNN, different kernels within one layer correspond to different clues for the task, while different layers correspond to the compositions of clues. The clues and compositions in a CNN can be grouped based on similarity. We have developed a method to partition kernels into groups based on the EM algorithm [5], as is shown in Fig. 1.

	Operation
1	Initialize K to be 1. Also initialize the means $\boldsymbol{\mu}=\{\boldsymbol{\mu}_1,\boldsymbol{\mu}_2,...,\boldsymbol{\mu}_K\}$, covariances $\boldsymbol{\Sigma}=\{\boldsymbol{\Sigma}_1,\boldsymbol{\Sigma}_2,...,\boldsymbol{\Sigma}_K\}$ and mixing coefficients $\boldsymbol{\pi}=\{\pi_1,...,\pi_K\}$ with K-means algorithm.
2	(Outer loop) Increase K
3	(Inner loop) Perform E step to evaluate the responsibilities using existing parameters of groups $\gamma(z_{nk})=\dfrac{\pi_k N(\mathbf{x}_n\|\boldsymbol{\mu}_k,\boldsymbol{\Sigma}_k)}{\sum_{j=1}^{K}\pi_j N(\mathbf{x}_n\|\boldsymbol{\mu}_j,\boldsymbol{\Sigma}_j)}$
4	Re-estimate the parameters of different groups with updated responsibilities $\boldsymbol{\mu}=\{\boldsymbol{\mu}_1,\boldsymbol{\mu}_2,...,\boldsymbol{\mu}_K\}, \boldsymbol{\Sigma}=\{\boldsymbol{\Sigma}_1,\boldsymbol{\Sigma}_2,...,\boldsymbol{\Sigma}_K\}$ and $\boldsymbol{\pi}=\{\pi_1,...,\pi_K\}$: $\mu_k^{new}=\frac{1}{N_k}\sum_{n=1}^{N}\gamma(z_{nk})\mathbf{x}_n$ $\Sigma_k^{new}=\frac{1}{N_k}\sum_{n=1}^{N}\gamma(z_{nk})(\mathbf{x}_n-\mu_k^{new})(\mathbf{x}_n-\mu_k^{new})^{\mathrm{T}}$ $\pi_k^{new}=\frac{N_k}{N}$ where $N_k=\sum_{n=1}^{N}\gamma(z_{nk})$
5	Check the convergence of $\ln p(\mathbf{X}\|\boldsymbol{\mu},\boldsymbol{\Sigma},\boldsymbol{\pi})$ where $\ln p(\mathbf{X}\|\boldsymbol{\mu},\boldsymbol{\Sigma},\boldsymbol{\pi})=\sum_{n=1}^{N}\ln\{\sum_{k=1}^{K}\pi_k \mathrm{N}(\mathbf{x}_n\|\mu_k,\Sigma_k)\}$ If the convergence criterion (reaching maximum) is not satisfied, return to Step 3, else proceed
6	Evaluate the influence of halving all groups within one layer on the performance by comparing the test accuracy before and after halving the layer. If the drop in accuracy is below a threshold (3%), stop, else return to Step 2.

Fig. 1 The algorithm of partitioning a CNN into functional units

The process is carried out layer by layer. The channels are grouped based on similarity. Suppose that Layer $L,(L>=1)$ has M output feature channels, while Layer $L+1$ has N output feature channels. The kernels in Layer $L+1$ are with size 3×3. In this way, there are M kernels connecting input channels to each of the N output channels in Layer $L+1$. We flatten each kernel into a 9×1 vector, and concatenate the M kernels to construct a $9M\times1$ vector. Finally, we model the N vectors with size $9M\times1$ as a mixture of Gaussians and group the N vectors based on similarity.

Let the number of groups K increase until all the necessary clues can be expressed by different groups. In the beginning, K = 1. Removing the group will cause a great loss in accuracy. For a proper K, these groups cover all K necessary functional units for the task. This method is evaluated on the Deeplab model [9]. By keeping the useful functional units only, similar performance can be achieved and therefore, the capacity of the model matches well to the complexity of the dataset. Experimental results will show that a smaller network with a right capacity can perform better than a larger network of overcapacity.

B. Capsule Networks

Different from a conventional CNN, the capsule network proposed in [6] represents one identity with one capsule that is composed of several neurons. The neurons within each capsule form a vector whose norm shows the probability of existing an identity and whose angle shows the variance of the identity. All capsules in one-layer vote for each capsule in the layer above by multiplying their own matrices by transformation matrices. Each of these votes is weighted by an assignment coefficient, and the coefficients are updated using the EM algorithm.

Fig. 2 shows the structure of one layer in the capsule network. The dimension of each capsule is 16. Convolutions in different dimensions are independent, and different feature representations are provided by different dimensions. The function of a dimension in the capsule network is similar to that of a functional unit in the CNN introduced in 2.1. We propose to adjust the capacity of a capsule network by adjusting the number of dimensions.

IV. Method to Increase the Complexity of a Dataset

In this section, a method is proposed to increase the complexity of a dataset through adding the types of variances in each class. The proposed algorithm is based on Bayesian GAN proposed in [22]. Different θ_g are sampled from $p(\theta_g|\theta_d)$ to obtain different generators. θ_g and θ_d denote the parameters of the generator and the discriminator. Different θ_g can generate samples with an approximate level of entropy but with different styles. The differences between the images generated by different generators are determined by the distances between the parameters θ_g of the generators.

In our proposed method, we sample from $p(\theta_g|\theta_d)$ to obtain different generators and select several generators with their

parameters θ_g that are sufficiently dissimilar. The images generated by the generators are added to the training data to enrich the variances in all classes. Each time, the generator and the discriminator are trained on images of one class.

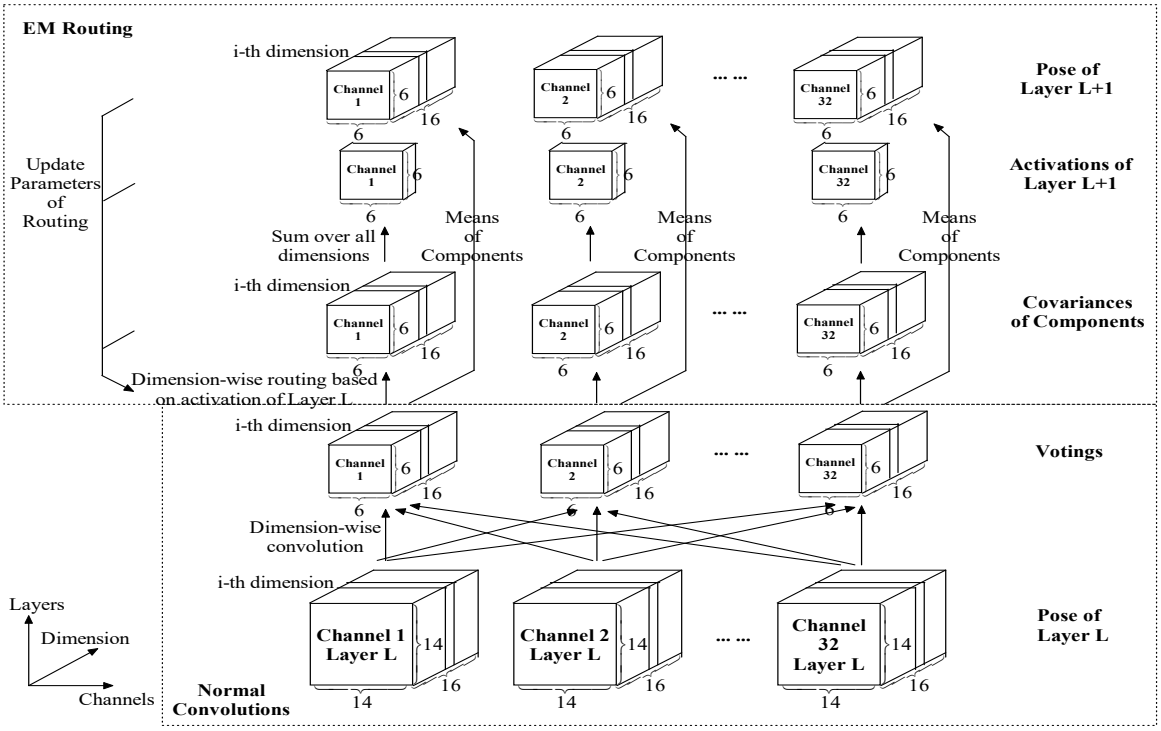


Fig. 2 One layer in the capsule network proposed in [6]

V. Experimental Results

A. Introduction to the Datasets

The performance of the Deeplab model proposed in [9] and that of our proposed model is evaluated on the PASCAL Person Parts Dataset [23]. The dataset contains 3,533 images with annotations on Head, Torso, Upper/Lower Arms and Upper/Lower Legs, resulting in six person parts classes and one background class. The performance evaluation of the capsule network is conducted on the MNIST dataset proposed in [10] which has 60,000 training examples and 10,000 test examples.

B. Adjust the Capacity of a Conventional CNN

Whether a group of functional unit is useful for prediction, it is evaluated by comparing the accuracy before and after removing these units. The evaluation is conducted on the PASCAL Person Parts Dataset [23]. The necessity of a certain functional unit is determined by comparing the accuracy before and after removing the unit. The removal of a group of channels involves dropping the connections between adjacent layers. Table I shows the influences on accuracy brought by removing each of the eight functional units in layers conv6_2 and conv6_3 from the network shown in Fig. 3 which is proposed in [9].

As is shown in Table I, there is one functional unit in each layer whose removal brings no harm to test accuracy as well as training accuracy. Removing the 5th functional unit from conv6_2 or removing the 4th functional unit from conv6_3 keeps training accuracy unchanged and increases test accuracy. So it can be inferred that the above two functional units are over-fitted to features that only appear in training data. Moreover, the removal of the 8th functional unit from conv6_2 and the removal of the 3rd functional unit from conv6_3 keep both training and test accuracy unchanged. Re-training with 5,000 iterations is conducted after removing functional units. In comparison, methods based on directly reducing channels, such as [18], requires over 15,000 iterations before convergence. The process of removing functional units which is followed by re-training is repeated until no functional unit can be removed.

TABLE I
The Change in Training and Test Accuracy (%) when Dropping One Functional Unit in Layers conv6_1, conv6_2 and conv6_3

	conv6_1		conv6_2		conv6_3	
	Train (%)	Test (%)	Train (%)	Test (%)	Train (%)	Test (%)
Complete	0.00	0.00	0.00	0.00	0.00	0.00
Remove Unit 1	-0.33	-0.87	-0.14	-0.10	-0.14	-3.03
Remove Unit 2	-0.02	-0.08	-0.09	-0.97	-0.09	-0.64
Remove Unit 3	-0.05	-0.01	-1.04	-3.19	+0.00	+0.00
Remove Unit 4	-0.58	-1.71	+0.00	-0.32	+0.00	+1.02
Remove Unit 5	-1.11	-4.20	-0.01	+0.18	-0.29	-1.04
Remove Unit 6	+0.00	+0.00	-0.36	-1.16	-0.03	-0.83
Remove Unit 7	-0.01	-0.52	-0.01	-0.33	-0.44	-1.31
Remove Unit 8	-0.14	+0.00	+0.00	+0.00	-0.41	-1.66

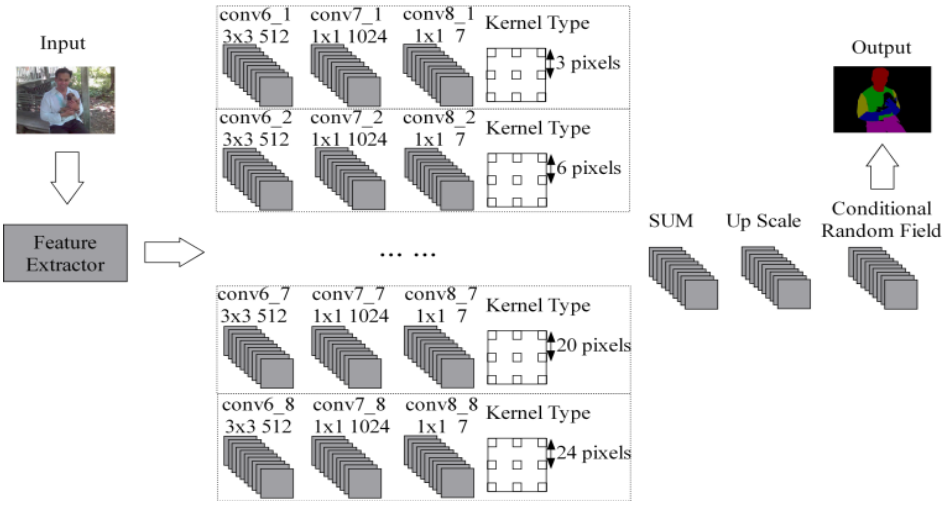


Fig. 3 Fully-convolutional model for semantic segmentation. Eight parallel filters with different field-of-views are adopted to extract features for pixel classification. The Kernel Type refers to the types of the kernels in layers from conv6_1 to conv6_8. The kernels in the eight layers are with size 3×3 but differ in the distance between weights in the kernels. The heat-maps generated by the eight parallel convolutional layers are summed up to generate the final heat-map. The feature extractor is composed of the first 13 layers shown in [9]

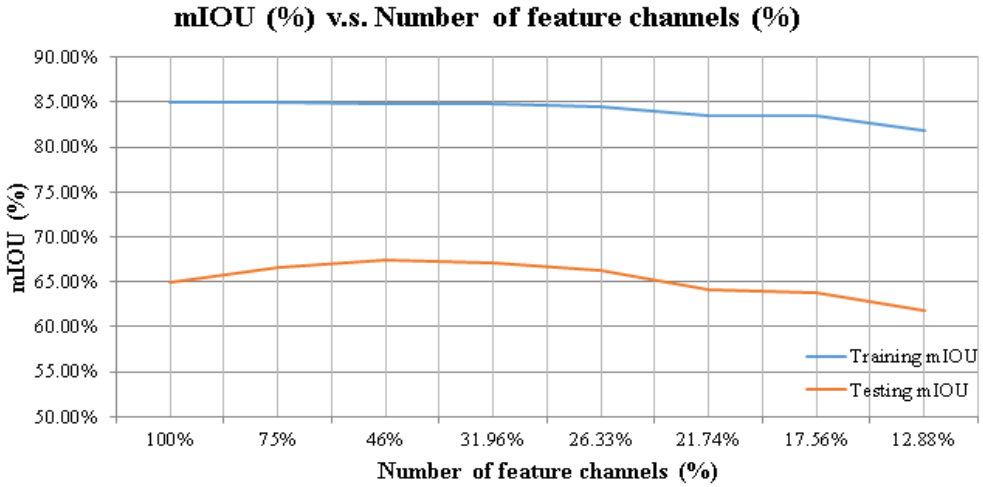


Fig. 4 The changes in training and test mIOU (%) versus the reduction in the number of feature channels. The horizontal axis denotes the portion of the number of feature channels remained in the model

Fig. 4 shows the influences on mean Intersection Union (mIOU, %) brought by reducing functional units from all layers in the model shown in Fig. 3. mIOU (%) is the ratio of the number of true positives over the sum of true positives, false negatives and false positives. The 100 percent in the horizontal axes in Fig. 4 corresponds to the original Deeplab model proposed in [9]. By continuously reducing channels through dropping functional units, the training accuracy keeps decreasing, the test accuracy increases first and then drops.

C. Performance Comparison on Benchmark Datasets

From Fig. 4, it can be seen that the model with reduced feature channels generalizes better than the original model shown in Fig. 3. The performance of the original model, the optimized model, and other related models is shown in Table II.

TABLE II
A COMPARISON IN MIOU (%) BETWEEN OUR MODEL AND BENCHMARK MODELS

Method	mIOU (%)
Attention [24]	56.39%
HAZN [25]	57.54%
LG-LSTM [26]	57.97%
Graph LSTM [27]	60.16%
Deep Lab-V2 [9]	64.94%
Deep Lab-V3 [28]	68.17%
Our optimized model	67.43%

TABLE III
A COMPARISON ON ACCURACY BETWEEN OUR METHOD AND DEEPLAB [9], [28]

Method	Accuracy (%)
Deep Lab-V2 [9]	77.69%
Deep Lab-V3 [28]	80.79%
Our optimized model	79.35%

The measure in Table II adopted for evaluating segmentation performance is mean Intersection Over Union (mIOU) proposed in [29], while the measure in Table III is accuracy. It is a metric for evaluating semantic segmentation tasks. It is calculated by dividing the number of true positive samples by the summation of true positive, false negative and false positive samples:

$$mIOU = \frac{1}{N}\sum_{i=1}^{N}\frac{n_{ii}}{t_i + \sum_{j\neq i} n_{ji}} \quad (1)$$

where n_{ji} is the number of pixels of class j which are predicted to class i, and $t_j = \sum_i n_{ji}$ is the total number of pixels belonging to class j. The measure mIOU takes into account both false positives and false negatives. The accuracy defined as $\sum_i n_{ii} / \sum_i t_i$ which divides the total number of correctly classified pixels by the number of pixels in the image.

It can be seen from Table II that the proposed framework significantly outperforms the existing methods. The improvement is about 2.49% over Deeplab-V2. Although Deeplab-V3 performs best, the complexity of Deeplab-V3 is significantly larger than the proposed model.

D.Adjust the Complexity of the MNIST Dataset

The MNIST dataset [10] contains handwritten digits of different styles. In Section III, a method was proposed to add variances to a dataset using Bayesian GAN. The generator and the discriminator were trained for 18,000 iterations in a semi-supervised fashion. 4000 images with labels and other 56,000 images without labels are used for training. The training process was conducted for 10 times. Each time, only the images of the same character are used for training. The number of θ_g's sampled from $p(\theta_g|\theta_d)$ is 5 and the number of stochastic gradient Hamiltonian Monte Carlo sampling is chosen to be 5 according to [22].

Fig. 5 shows the distribution of original images and the generated images. It can be seen that the images generated by different generators (in red, green and blue) are sufficiently dissimilar, and the generated images are different from the original images (in purple). 10000 images including all characters were generated to enrich the original training data.

E. Matching the Capacity of a Model to the Complexity of a Dataset

The MNIST dataset is augmented to a larger dataset containing 70,000 training images and 11,600 test images. We adjusted independent functional units in the capsule network discussed in Section II to make the network better matched to the complexity of both MNIST and the augmented dataset. Fig. 6 shows the test accuracy on both MNIST and the augmented MNIST datasets of the three networks with different dimensions. It can be seen from Fig. 6 (a) that for the MNIST dataset, the network with 16 functional units performs the best; the network with nine units suffers from under-fitting; and the network with 25 units suffers from over-fitting. As the dataset becomes more complex as the augmented MNIST dataset, the networks with 9 and 16 units suffer from under-fitting, while the network with 25 units performs better (Fig. 6 (b)). In conclusion, a model with a larger capacity is matched to a dataset that is more complex. The necessary capacity of a model is determined by the complexity of the dataset.

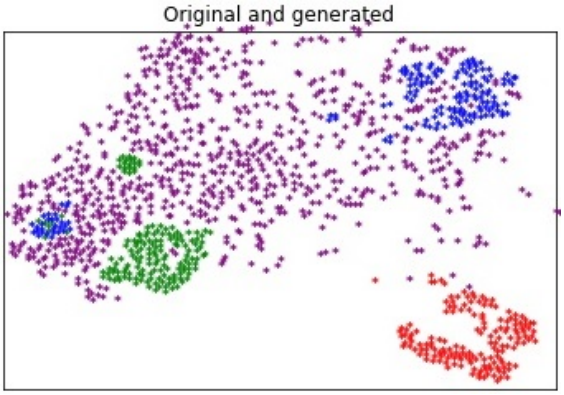
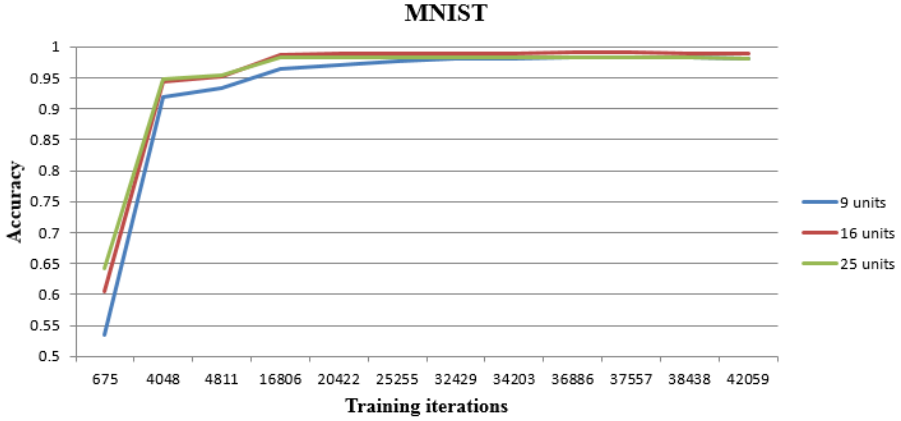
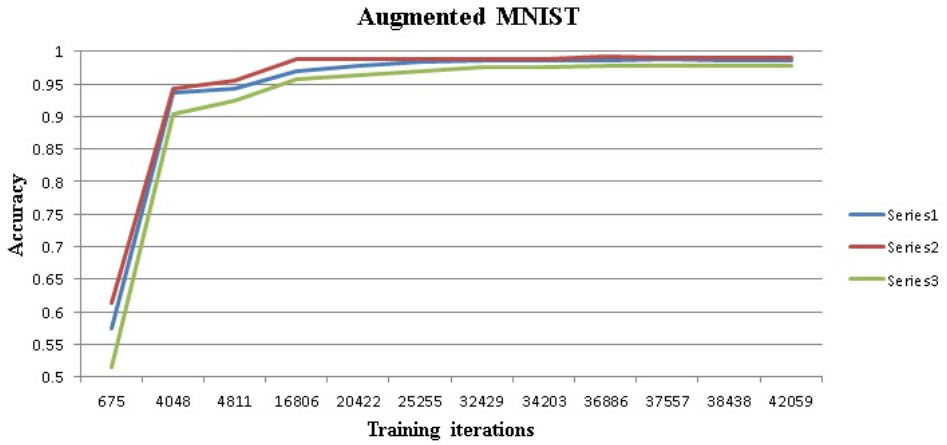


Fig. 5 The images from the original MNIST dataset (purple points) and generated images (in red, green and blue). The figure is based on tSNE [30]



(a) Test accuracy on the MNIST dataset



(b) Test accuracy on the augmented MNIST dataset

Fig. 6 Test accuracy of networks with 9, 16, and 25 functional units on both MNSIT and the augmented MNIST dataset

VI. Conclusion

In this paper, a method is proposed to adjust the number of independent functional units in a conventional CNN to control the capacity of the model for better matching to the complexity of the task. We also propose a method of adjusting the number of functional units in a capsules network to optimize its capacity for a training dataset. In addition, an approach for enriching the variances of the MNIST dataset is proposed based on Bayesian GAN. Experimental results have shown that, by adjusting the number of functional units, over-fitting can be avoided and the capacity of the model can match better to the complexity of the dataset.

Acknowledgment

This work was partially supported by a research grant from The Hong Kong Polytechnic University (Project Code: 4-BCCJ) and a Natural Science Foundation of China (NSFC) grant (Project Code: 61473243). Mr. Yalong Jiang would like to acknowledge the financial support from PolyU for his PhD study.

References

[1] V. N. Vapnik, and A. Ya. Chervonenkis, "On the uniform convergence of relative frequencies of events to their probabilities," Theory of Probability and Its Applications, vol. 16, no. 2, pp. 264-280, 1971.
[2] V. N. Vapnik, Estimation of Dependences Based on Empirical Data. New York: Springer-Verlag, 1982, vol. 40.
[3] Blumer, A. Ehrenfeucht, D. Haussler, and M. K. Warmuth, "Learnability and the Vapnik–Chervonenkis dimension," Journal of the ACM (JACM), vol. 36, no. 4, pp. 929-965, Oct. 1989.
[4] V. N. Vapnik, The Nature of Statistical Learning Theory. New York: Springer science & business media, 2013.
[5] Todd K. Moon, "The expectation-maximization algorithm.," IEEE Signal processing magazine, vol. 13, no. 6, pp. 47-60, 1996.
[6] Under review, "Matrix Capsules with EM routing," in International Conference on Learning Representations, 2018.
[7] Y. Jiang, and Z. Chi, "A Fully-Convolutional Framework for Semantic Segmentation," in International Conference on Digital Image Computing: Techniques and Applications, 2017.
[8] Chen, X., Mottaghi, R., Liu, X., Fidler, S., Urtasun, R., and Yuille, A., "Detect what you can: Detecting and representing objects using holistic models and body parts," in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014, pp. 1971-1978.
[9] L. C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and L. Alan, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. PP, no. 99, pp. 1-1, April 2017.
[10] Li Deng, "The MNIST database of handwritten digit images for machine learning research," IEEE Signal Processing Magazine, vol. 29, no. 6, pp. 141-142, 2012.
[11] Nelson, M. L. Collins, and M. Luciana, Handbook of developmental cognitive neuroscience.: MIT Press, 2001.
[12] W. Huitt, and J. Hummel., "Piaget's theory of cognitive development," Educational psychology interactive, vol. 3, no. 2, pp. 1-5, 2003.
[13] Sigaud, O., and Droniou, A., "Towards deep developmental learning.," IEEE Transactions on Cognitive and Developmental Systems, vol. 8, no. 2, pp. 99-114, 2016.
[14] S., Thrun, "Lifelong learning algorithms.," Learning to learn, pp. 181-209, 1998.
[15] T. M. Mitchell, W. Cohen, E. Hruschka, P. Talukdar, and J. Welling, "Never-ending learning," in AAAI, 2015, pp. 2302-2310.
[16] Tessler, C., Givony, S., Zahavy, T., Mankowitz, D. J., and Mannor, S., "A Deep Hierarchical Approach to Lifelong Learning in Minecraft.," in AAAI, 2017, pp. 1553-1561.
[17] Pickett, M., Al-Rfou, R, Shao, L., and Tar, C., "A Growing Long-term Episodic & Semantic Memory.," in arXiv preprint arXiv: 2016, p. 1610.06402.
[18] Guo, Y., Yao, A., and Chen, Y., "Dynamic network surgery for efficient dnns.," in Annual Conference on Neural Information Processing Systems, 2016, pp. 1379-1387.
[19] Zhang, X., Zhou, X., Lin, M., and Sun, J., "Shufflenet: An extremely efficient convolutional neural network for mobile devices.," in arXiv preprint arXiv, 2017, p. 1707.01083.
[20] Veniat, T., and Denoyer, L., "Learning time-efficient deep architectures with budgeted super networks." in arXiv preprint arXiv, 2017, p. 1706.00046.
[21] Xie, L., and Yuille, A., "Genetic CNN," in arXiv preprint arXiv, 2017, p. 1703.01513.
[22] S. Yunus and G. W. Andrew, "Bayesian GAN," in Conference on Neural Information Processing Systems, 2017.
[23] Everingham, M., Eslami, S.A., Van Gool, L., Williams, C.K., Winn, J., and Zisserman, A., "The pascal visual object classes challenge a retrospective," International journal of computer vision, vol. 111, no. 1, pp. 198-136, 2015.
[24] Chen, L.C., Yang, Y., Wang, J., Xu, W., and Yuille, A.L., "Attention to scale: Scale-aware semantic image segmentation," in Proc. IEEE CVPR, Jun., 2016, pp. 3640 - 3649.
[25] Xia, F., Wang, P., Chen, L.C., and Yuille, A.L., "Zoom better to see clearer: Human part segmentation with auto zoom net," in European

conference on Computer Vision, 2016, pp. 648-663.
[26] Liang, X., Shen, X., Xiang, D., Feng, J., Lin, L., and Yan, S., "Semantic object parsing with local-global long short-term memory," in Proc. IEEE CVPR, Jun., 2016, pp. 3185-3193.
[27] Liang, X., Shen, X., Feng, J., Lin, L., and Yan, S., "Semantic object parsing with graph lstm," in Proc. ECCV, Oct., 2016, pp. 125-143.
[28] Chen, Liang-Chieh, George Papandreou, Florian Schroff, and Hartwig Adam, "Rethinking atrous convolution for semantic image segmentation.," in arXiv preprint arXiv: 2017, p. 1706.05587.
[29] Oliveira, G.L., Valada, A., Bollen, C., Burgard, W., and Brox, T., "Deep Learning for human part discovery in images," in Proc. IEEE ICRA, May., 2016, pp. 1634-1641.
[30] Maaten, Laurens van der, and Geoffrey Hinton, "Visualizing data using t-SNE," Journal of machine learning research, vol. 9, pp. 2579-2605, 2008.

Yalong Jiang was born in 1989. This author received the BEng from Harbin Engineering University and MEng from Beijing Institute of Technology, in 2012 and 2015, respectively.

He is currently working toward the PhD degree in the Department of Electronic and Information Engineering, Hong Kong Polytechnic University. His research interests include pattern recognition, computer vision, and machine learning.

He has published papers in conferences such as the 2017 IEEE International Conference on System, Man and Cybernetics. 2017 Digital Image Computing: Techniques and Applications.

Zheru Chi was born in 1962. This author received the BEng and MEng degrees from Zhejiang University, in 1982 and 1985, respectively, and the PhD degree from the University of Sydney, in March 1994, all in electrical engineering.

Between 1985 and 1989, he was on the faculty of the Department of Scientific Instruments, Zhejiang University. He worked as a senior research assistant/research fellow in the Laboratory for Imaging Science and Engineering, University of Sydney, from April 1993 to January 1995. Since February 1995, he has been with Hong Kong Polytechnic University, where he is now an associate professor in the Department of Electronic and Information Engineering. Since 1997, he has served on the organization or program committees for a number of international conferences. He was an associate editor of the IEEE Transactions on Fuzzy Systems between 2008 and 2010, and is currently an editor of the International Journal of Information Acquisition.

His research interests include image processing, pattern recognition, and computational intelligence. He has authored/co-authored one book and 11 book chapters, and published more than 190 technical papers. He is a member of the IEEE.