

Image Recognition Based on Combined Filters with Pseudoinverse Learning Algorithm

Xiaodan Deng, Xiaoxuan Sun, Ping Guo, Qian Yin

▶ **To cite this version:**

Xiaodan Deng, Xiaoxuan Sun, Ping Guo, Qian Yin. Image Recognition Based on Combined Filters with Pseudoinverse Learning Algorithm. 15th IFIP International Conference on Artificial Intelligence Applications and Innovations (AIAI), May 2019, Hersonissos, Greece. pp.202-209, 10.1007/978-3-030-19823-7_16 . hal-02331298

HAL Id: hal-02331298

https://inria.hal.science/hal-02331298

Submitted on 24 Oct 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

Image Recognition Based on Combined filters with Pseudoinverse Learning Algorithm

Xiaodan Deng[1], Xiaoxuan Sun[1], Ping Guo[1*], and Qian Yin[2*]

[1] Image Processing and Pattern Recognition Laboratory,
School of systems science,
Beijing Normal University, Beijing 100875, China
[2] Image Processing and Pattern Recognition Laboratory,
College of information science and technology,
Beijing Normal University, Beijing, China
dengxiaodancyc@163.com, sunxiaoxuan922@163.com,
pguo@ieee.org, yinqian@bnu.edu.cn

Abstract. Deep convolution neural network (CNN) is one of the most popular Deep neural networks (DNN). It has won state-of-the-art performance in many computer vision tasks. The most used method to train DNN is Gradient descent-based algorithm such as Backpropagation. However, backpropagation algorithm usually has the problem of gradient vanishing or gradient explosion, and it relies on repeated iteration to get the optimal result. Moreover, with the need to learn many convolutional kernels, the traditional convolutional layer is the main computational bottleneck of deep CNNs. Consequently, the current deep CNN is inefficient on computing resource and computing time. To solve these problems, we proposed a method which combines Gabor kernel, random kernel and pseudoinverse kernel, incorporating with pseudoinverse learning (PIL) algorithm to speed up DNN training processing. With the multiple fixed convolution kernels and pseudoinverse learning algorithm, it is simple and efficient to use the proposed method. The performance of the proposed model is tested on MNIST and CIFAR-10 datasets without using GPU. Experimental results show that our model is better than existing benchmark methods in speed, at the same time it has the comparative recognition accuracy.

Keywords: Pseudoinverse learning autoencoder, Gabor kernel, random kernel, Image recognition, Ensemble learning.

1 Introduction

Recently, deep convolutional neural networks (CNNs) have been overwhelmingly successful across a variety of visual perception tasks. LeNet5 in [1], designed by Yann LeCun and Yoshua Benfio in 1998, is considered as the beginning of CNN. Over the past several years, many successful CNN architectures have emerged, such as AlexNet [2], VGG [3], GoogLeNet [4], ResNet [5, 6], MobileNet [7], and DenseNet [8], etc. Most deep neural networks are trained by the gradient descent (GD) based algorithms

and their variations [1, 3]. However, it is found that the gradient descent based algorithm in deep neural networks has inherent instability. This instability blocks the learning process of the previous or later layers. Though CNN has good performing result, it needs much professional knowledge to use and it takes a lot of time to train.

In this paper, we proposed a method combines Gabor kernel [9], random kernel and pseudoinverse kernel. It is corresponding to multiple convolutional kernels. Gabor feature from Gabor kernel is a kind of handcraft feature which is faster obtained than learned features. In paper [10], perturbation layer is an alternative of convolutional layer. Their theoretical analysis shows that the perturbation layer can approximate the response of a standard convolutional layer. Inspired by perturbative neural network, a kind of random kernel with the same size of input data was proposed. Pseudoinverse learning algorithm was proposed by Guo et al. [11-13]. It's a fast feedforward propagation algorithm. In our method, a random weight was used as the input weight of the pseudoinverse learning algorithm. As a result, the training time is reduced significantly and the random weight can regulate the whole model.

Our model combines multiple fixed convolutional kernels, such as Gabor kernel, random kernel and pseudoinverse kernel. The parameters of convolutional kernel can be obtained without iteration. Therefore, the training process is accelerated. Moreover, the variant kernels contribute to variant image features which facilitate the recognition task. Instead of using gradient-based algorithm, pseudoinverse learning algorithm was used to speed up the training process significantly. three base learner were trained, then feed their prediction to the meta learner to obtain the final result. Our model was tested on MNIST and CIFAR-10 datasets without using GPU. The experimental results show that our model is better than existing benchmark methods in speed, at the same time it has the comparative recognition accuracy.

2 Related Work

Recently, random feature has attracted researchers' attention. Random feature shows its significant success in many research fields. The test of time award paper in NIPS 2017[14], presented two method, random Fourier features and random binning features to map the input data to random features. Random feature mapping speeds up the training of large-scale kernel methods. Perturbative Neural Networks [10] presented a perturbative layer as the alternative of convolutional layer. The perturbative layer computes its response as a weighted linear combination of non-linearly activated additive noise perturbed inputs. The input data added a random and fixed noise is a kind of random features. The perturbation layer in [10] shows that maybe the convolutional layers are not necessary to be learned from input image. Perturbative Neural Networks performs as well as standard convolutional neural network.

Pseudoinverse learning algorithm was originally proposed by Guo *et al.* [11-13], which is a kind of fast feedforward training algorithm. As a variant of pseudoinverse learning algorithm, pseudoinverse learning autoencoder [15] is a useful method to train the multiplayer neural network.

Our previous works include combining handcraft features with pseudoinverse learning algorithm [16, 17]. These works perform well in terms of training time, however, it's not satisfactory in accuracy especially on complicated data sets. In this paper, we proposed a method combining multiple fixed convolutional kernels, using pseudoinverse learning algorithm to accelerate the training. Our method performs better than other baseline method in speed, and obtains comparable accuracy. Meanwhile, our proposed method does not need large compute resource. It can meet the need of edge learning.

3 Proposed Methodology

3.1 Gabor kernel

The base learner 1 was presented as shown in Figure 1. The input image is extracted features by Gabor kernels firstly and then trained by PIL1 [18]. PIL1 is original pseudoinverse adding Gaussian noise perturbation matrix. Gabor kernel is corresponding to convolutional kernel and Gabor feature is corresponding to convolutional feature as shown in formula (1),

$$\mathbf{I}_G = \mathbf{I} \oplus \mathbf{G}, \tag{1}$$

where $\mathbf{I}$ is the grayscale distribution of the image, $\mathbf{I}_G$ is the feature extracted from $\mathbf{I}$, "$\oplus$" stands for 2D convolution operator, $\mathbf{G}$ is the defined Gabor kernel. As a kind of handcraft feature, Gabor feature can be obtained faster than learned features. Meanwhile, multiple Gabor features will facilitate the recognition.

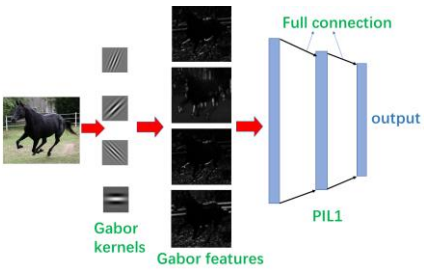


Figure 1. Gabor kernels are variant, the Gabor kernels is set in advance.

3.2 Random kernel

The second base learner was shown in Figure 2. The PIL1 part is as same as demonstrated in section 3.1. The difference is on the front part. The input image was added with a random kernel, which has the same size as the input image. The values in random kernel are derived from specific distribution. Gaussian distribution and uniform distribution both work well. It's better to control the mean of extracted values is zero [19,20].

At the same time, the noise value should be small, otherwise, the original information in input is covered by heavy noise. The features added noise are activated by RELU. Then features are combined by linear weight. The obtained feature is as follow,

$$F = \sum_{i=1}^{q} W_i * f_{relu}(X + R_i), \quad (2)$$

where, q is the number of random features, R is the random kernel matrix.

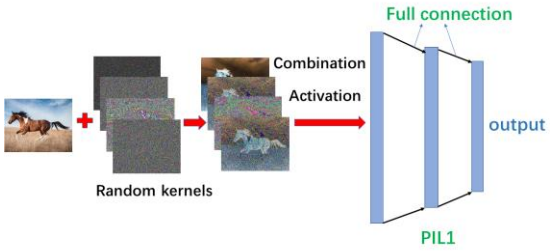


Figure 2. Random kernel is a random noise matrix with the same size of the input image.

Random features are obtained by adding random noise to the input image. This is the simplest and fastest way to get random features. Moreover, adding noise to the input data in neural network can regulate the performance.

3.3 Pseudoinverse kernel

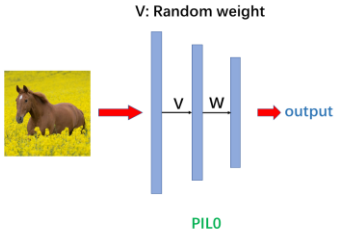


Figure 3. pseudoinverse kernel. The input weight V is obtained randomly.

The third base learner was shown in Figure 3. The input image is sent to PIL0[18]. The input weight of PIL0 is a random weight whose values are within a small scale, such as [-1,1]. The number of input data is n, and the number of hidden neurons is p (p<= n). The size of random input weight is n*p. The random input weight is usually gotten from the Gaussian distribution and uniform distribution, and it's necessary to restrict the mean value to be zero and variance to be limited [19,20].

3.4 Ensemble Model

Our proposed ensemble model is shown in Figure 4. The ensemble model contains three base learners, combining Gabor kernel, random kernel and pseudoinverse kernel. The training data set is extracted from the original sample data set. N_t $t$raining samples will be sampled from the original sample set using the Bootstrapping method. We obtain three training data sets which are independent of each other. Each training data set corresponds to a base learner. The prediction from the three base learners is fed to a meta learner to get the final result. The meta learner here is a multilayer neural network. The multilayer neural network was trained by pseudoinverse learning algorithm.

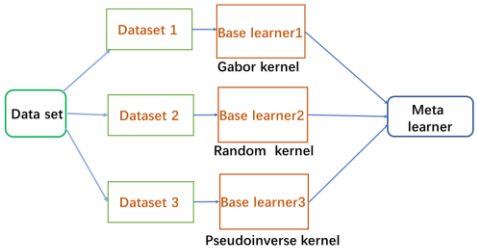


Figure 4. Ensemble model. The model contains three base learners, combining Gabor kernel, random kernel and pseudoinverse kernel.

4 Performance Evaluation

We use our proposed method to classify real world datasets, including MNIST dataset and CIFAR-10 dataset. All the experiments are conducted on the same hardware computer with Core i7 3.20 GHz processors.

4.1 MNIST Dataset

The Table 1 shows comparison results of our method and other benchmark methods on MNIST dataset. In this experiment, Three training data sets are extracted from original data. Each training data set has 50000 samples. In the first base learner, Four Gabor kernels chose from Gabor kernel bank are used to obtain Gabor features. The number of hidden neurons in PIL1 is 576. In the second base learner, the random noise is extracted from uniform distribution with scale (-0.05, 0.05). Four variant fixed noise matrixes are added to training data separately. The activation function is RELU. Then, they are added linearly with average weights. In the third base learner, the input weight of PIL0 is extracted from uniform distribution randomly. In this experiment, 5 and 10 convolution kernels are used in different layers in LeNet5. MLP uses one hidden layer and the number of hidden neurons is 300. PILAE has one encoder layer in this experiment. The same Gabor features are used in GF + PILAE and the method proposed in this paper.

From the experimental results shown in Table 1, we can see that the proposed method is faster than other methods, and the accuracy is comparable to other methods.

Table 1. Performance comparison on MNIST dataset.

Model	Training accuracy(%)	Testing accuracy(%)	Training time(s)
Ours	98.95	98.53	44.35
LeNet5	98.51	98.49	1270.8
PILAE	93.88	93.78	33.54
MLP	97.87	97.80	411.68
SVM	98.72	96.46	2593.28
GF+PILAE	98.86	98.42	103.25

4.2 CIFAR-10 Dataset

The Table 2 shows comparison results of our method and other benchmark methods on CIFAR-10 dataset. In this experiment, each training dataset has 50000 samples. In the first base learner, four Gabor filters are used. The number of hidden neurons in PIL1 is 789. In the second base learner, the random noise is extracted from uniform distribution with scale (-0.25, 0.25). In the third base learner, the input weight of PIL0 is extracted from uniform distribution randomly with scale (-0.5, 0.5). In this experiment, LeNet5 has 20 and 50 kernels in different layers. MLP uses one hidden layer and the number of hidden neurons is 2000. PILAE has three layers in this experiment. The same Gabor features are used in GF + PILAE and the method proposed in this paper.

Table 2. Performance comparison on CIFAR10 dataset.

Model	Training accuracy(%)	Testing accuracy(%)	Training time(s)
Ours	54.33	52.11	354.46
LeNet5	64.31	63.02	6743.82
PILAE	45.16	44.08	151.47
MLP	38.98	38.32	765.68
GF+PILAE	48.34	47.02	388.23

From the experimental results shown in Table 2, we can see that the proposed method is superior than other methods in speed, and is comparable to other methods in accuracy.

4.3 Discussion

From the results shown in Table1 and Table 2, we know that our method is better than other methods in speed. Our method combines multiple kernels including Gabor kernel,

random kernel and pseudoinverse kernel, which corresponded to Gabor convolutional kernel, random convolutional kernel, and pseudoinverse convolutional kernel. Gabor feature is handcraft feature which is easier to obtain than learned features in terms of time. Adding noise is the easiest way to get random features which will speed up the training. Moreover, adding noise to the input data is a way to regularize the whole network. The pseudoinverse algorithm is a feedforward way to train network. Instead of backpropagation algorithm, the pseudoinverse algorithm does not need repeated iteration. The input weight in pseudoinverse learning network was set as random weight, which will speed up and regularize the whole network. Our method performs well on MNIST dataset in speed and accuracy. On CIFAR-10 dataset, the method performs well on speed, however, the accuracy is not good enough. The reason may be that we only use three layers in pseudoinverse learning network. The network is not deep enough to get better results. In the future, we will design more complicated network architecture to improve the classification performance on the complicated image.

5 Conclusions

In this paper, a method is proposed to improve the performance of image classification task. The classification model contains three base learners, taking the advantages of Gabor convolutional kernel, random convolutional kernel and pseudoinverse convolutional kernel. The multiple convolutional kernels generate variant submodules. It satisfies the diversity of ensemble learning requirements. In the proposed model, the convolution kernels are set manually, without iteration. Instead of gradient descent based backpropagation algorithm which is time consuming, the full connection layer is trained by pseudoinverse learning algorithm. All the pseudoinverse algorithm in submodules use random input weight, which is extracted randomly from uniform distribution. The Gabor kernels, random kernels and random input weight all speed up the training process. The performance of our model was tested on some benchmark datasets such as MNIST, CIFAR-10 without using GPU. The results show that our model is superior to other models in learning speed and the learning accuracy can be compared with other models.

Acknowledgements

The research work described in this paper was fully supported by the Joint Research Fund in Astronomy (U1531242) under cooperative agreement between the NSFC and CAS, the National Key R&D Program of China (2017YFC1502505). Prof. Ping Guo and Qian Yin are the authors to whom all correspondence should be addressed.

References

1. LeCun Y, Bottou L, Bengio Y, & Haffner P (1998) Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11), pp 2278-2324

2. Krizhevsky A, Sutskever I, & Hinton G E (2012) Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems, pp 1097-1105
3. Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556
4. Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, & Rabinovich A (2015) Going deeper with convolutions. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1-9
5. He K, Zhang X, Ren S, & Sun J (2016) Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp 770-778
6. He K, Zhang X, Ren S, & Sun J (2016) Identity mappings in deep residual networks. In European conference on computer vision, pp 630-645. Springer, Cham
7. Howard A G, Zhu M, Chen B, Kalenichenko D, Wang W, Weyand T & Adam H (2017) Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861
8. Huang G, Liu Z, Van Der Maaten L, & Weinberger K Q (2017) Densely connected convolutional networks. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp 4700-4708
9. Lee T S (1996) Image representation using 2D Gabor wavelets. IEEE Transactions on Pattern Analysis & Machine Intelligence, (10), 959-971
10. Juefei-Xu F, Naresh Boddeti V, & Savvides M (2018) Perturbative neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 3310-3318
11. Guo P, Chen C P, & Sun Y (1995) An Exact supervised learning for a three-layer supervised neural network. In Proceedings of 1995 International Conference on Neural Information Processing, pp 1041-1044
12. Guo P, Lyu M R, & Mastorakis N E (2001) Pseudoinverse learning algorithm for feedforward neural networks. Advances in Neural Networks and Applications, pp 321-326
13. Guo P, & Lyu M R. (2004) A pseudoinverse learning algorithm for feedforward neural networks with stacked generalization applications to software reliability growth data. Neurocomputing, 56, pp 101-121
14. Rahimi A, Recht B (2008) Random features for large-scale kernel machines. In Advances in neural information processing systems, pp 1177-1184
15. Wang K, Guo P, Xin X, Ye Z (2017) Autoencoder, low rank approximation and pseudoinverse learning algorithm. 2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC), pp 948-953
16. Deng X, Feng S, Guo P, Yin Q (2018) Fast Image Recognition with Gabor Filter and Pseudoinverse Learning AutoEncoders. In International Conference on Neural Information Processing, pp 501-511. Springer, Cham.
17. Feng S, Li S, Guo P, Yin Q (2017) Image Recognition with Histogram of Oriented Gradient Feature and Pseudoinverse Learning AutoEncoders. In International Conference on Neural Information Processing, pp 740-749. Springer, Cham
18. Guo P (2018) A Vest of the Pseudoinverse Learning Algorithm. arXiv preprint arXiv:1805.07828
19. Nelder J A, Wedderburn R W (1972) Generalized linear models. Journal of the Royal Statistical Society: Series A (General), 135(3), pp 370-384
20. Moody J E (1992) The effective number of parameters: An analysis of generalization and regularization in nonlinear learning systems. In Advances in neural information processing systems, pp 847-854