# Incremental PCANet: A Lifelong Learning Framework to Achieve the Plasticity of both Feature and Classifier Constructions

Wang-Li Hao and Zhaoxiang Zhang(✉)

CAS Center for Excellence in Brain Science and Intelligence Technology, CASIA, Beijing, China
zhaoxiang.Zhang@ia.ac.cn

**Abstract.** The plasticity in our brain gives us promising ability to learn and know the world. Although great successes have been achieved in many fields, few bio-inspired methods have mimiced this ability. They are infeasible when the data is time-varying and the scale is large because they need all training data loaded into memory. Furthermore, even the popular deep convolutional neural network (CNN) models have relatively fixed structures. Through incremental PCANet, this paper aims at exploring a lifelong learning framework to achieve the plasticity of both feature and classifier constructions. The proposed model mainly comprises of three parts: Gabor filters followed by maxpooling layer offering shift and scale tolerance to input samples, cascade incremental PCA to achieve the plasticity of feature extraction and incremental SVM to pursue plasticity of classifier construction. Different from CNN, the plasticity in our model has no back propogation (BP) process and don't need huge parameters. Experiments have been done and their results validate the plasticity of our models in both feature and classifier constructions and further verify the hypothesis of physiology that the plasticity of high layer is better than the low layer.

**Keywords:** Plasticity · Lifelong learning · Incremental PCANet · Incremental SVM

## 1 Introduction

Recently, bio-inspired visual processing methods have drawn much more attention as they are favorable in guiding people to design effective models in surveillance, automotive safety and robotics application fields. However, few of them have investigated the plasticity like in primates' visual cortex.

HMAX [1], a popular bio-inspired shallow model, has two S layers, two C layers and a view-tuned units layer. Via convoluting with filters, the S layers can extract more and more abstract features. Through max pooling, the C layers can guarantee the scale and position invariant of the images. VisNet [2] has a similar architecture with HMAX. It comprises of Differential of Gaussian (DoG)


C.-L. Liu et al. (Eds.): BICS 2016, LNAI 10023, pp. 298–309, 2016.
DOI: 10.1007/978-3-319-49685-6_27

filters layer and 4 hierarchical layers that correspond to V2, V4, PIT and AIT respectively. Although they are capable of generating some physiological results, they need all data loaded into memory, which makes them insufficient when dealing with the large scale or flow data. Furthermore, they are lack of plasticity.

Drawn inspiration from the hierarchical structures of the visual cortex [6], combining the deep learning [7] and convolutional neural networks [8], the deep models [3–5] emerged. They generally consist of some continuous convolutional layers, pooling layers and full connection layers. From low to high, convolution layers tend to extract more and more complex and abstract features. Recently, deep convolutional neural networks (DCNNs) have been able to achieve impressive performances on very large and difficult object databases such as ImageNet [9]. Inspite the BP process of DCNNs can finetune the deep networks, their training are very expensive and need a lot of parameters.

Refering to convolution neural network, a simple unsupervised deep learning baseline, PCANet, is first proposed by Chan et al. [10]. It consists of cascade PCA and the binarization hashing followed by block-wise histograms. Despite simple architecture, PCANet obtains very well performance in many image classification tasks. Nevertheless, its training process also need all data available and it has no plasticity.

Above all, no matter shallow and deep bio-inspired models, they are not investigate the plasticity or have expensive finetuning cost. In this paper, we first propose the incremental PCANet, a lifelong learning framework to explore the plasticity of both feature and classifier constructions. Our model basically consists of three parts, Gabor convolutional filters with different scales and orientations followed by maxpooling layer, incremental PCANet and incremental SVM. In the view of machine learning, our model has well self-adaption and robustness which should be attributed to its lifelong learning framework. From the respect of brain-inspired fields, our model explores the plasticity of visual pathway from low to high levels.

The contributions of this paper are listed as follows: (1) Since PCANet suffers a performance degeneration when the input images exhibit diverse in scales or poses, the Gabor filters with several scales and orientations followed by max pooling layer are utilized to process the images. (2) The incremental PCANet is first proposed to investigate the plasticity of feature extraction like in the visual cortex. (3) To explore the plasticity of classifier construction, offline linear classifier utilized in the traditional PCANet is alternated with incremental SVM. (4) We first combine the plasticity of both feature extraction and classifier construction together, thus leading to an end to end lifelong learning framework. (5) Through our model, we do some experiments to validate the plasticity of which layer is crucial.

The remainder of this paper is organized as follows. Related works are described in Sect. 2, our method is shown in Sect. 3, experiments are listed in Sect. 4 and we conclude this paper in Sect. 5.

## 2 Related Works

Inspired by the hierarchical structure of visual cortex, hierarchical neural networks for object categorization have achieved a widespread success in a variety of domains. Neocognitron, one of the earliest hierarchical neural networks was proposed by Fukushima [11]. Other impressive hierarchical neural networks for object recognition contain LeNet [12] and HMAX [1].

In recent years, the deep hierarchical convolutional neural networks showed us a gradual deeper model. Starting from AlexNet [3], VGG [4], Inception [5] to Residual [13] networks. Furthermore, the performance become much better in many image recognition tasks. AlexNet [3], comprises of five convolutional and three fully connected layers, which wins the ILSVRC2012 competition. To investigate the function of the number of internal layers of DCNNs, Simonyan and Zisserman [4] develop deep convolutional networks with 11, 13, 16, and 19 layers. Their results have shown that the more the layer number is, the better the results will be. GoogLeNet, consists of several inception modules, which wins ILSVRC 2014 [5]. The latest residual networks (ResNet) [13] makes further breakthrough in many tasks, it has won ImageNet and COCO 2015 competition.

On the basis of scattering theory, scattering convolution network (ScatNet) [14] is proposed. Its filters are prefixed as they are derived from mathematical functions. Benefit from these filters, ScatNet shows the state-of-the-art performance in many fields, such as texture discrimination and handwritten recognition.

With the explosion of data, incremental techniques that don't need all data loaded into memory emerged to perform many tasks like principal component analysis (PCA), support vector machine (SVM) and so on. The basic principle of incremental PCA is to update the current PCA without recalculating it when the new data arrive. Many techniques have been proposed to realize it, such as perturbation techniques [15], incremental methods [16], and stochastic optimization [17]. The key principle of incremental SVM is to make use of the current SVM solution to simply figure out the quadratic program of the next search. Specifically, when the new data arrive, they are integrated into the quadratic program and the kernel and regularization parameters $(C, \sigma)$ are then modified correspondingly [18].

Lifelong learning, is very important to the flexible machines. Early studies on lifelong learning was mainly about sharing distance metrics through transferring invariances in neural networks [19] and task clustering [20]. It has also been extended for learning by reading [21].

## 3 Methods

Among bio-inspired methods, few of them have mimiced the plasticity in primates' visual cortex. To explore the plasticity of both feature extraction and classifier construction, a lifelong learning framework based on incremental PCANet proposed here.

### 3.1 MulOri_PCANet

Although PCANet has promising performance in some face, hand-written digital, texture and object recognition tasks, when the input data bear diverse scales or poses, the performance of PCANet may decrease. Inspired by the phenomenon that the primate vision cortex (V1) has some shift and scale invariance [1], its similar realization in machine learning, the Gabor filters with several scales and orientations followed by maxpooling layer are employed to process the images before they are sent to PCANet. They can be described as:

$$G(X,Y) = exp(-\frac{X^2+\gamma^2Y^2}{2\sigma^2}) \times \cos(\frac{2\pi}{\lambda}X) \tag{1}$$

Where $X = x\cos\theta + y\sin\theta$ and $Y = -x\sin\theta + y\cos\theta$, $\theta$ denotes the orientation, $\sigma$ represents effective width and $\lambda$ indicates the wavelength.

Assume $N$ input images are $\{I_i\}_{i=1}^N$. Each of them is first convoluted with the above Gabor filters and several feature maps are obtained subsequently. To mimic the shift and position invariances and orientation sensitivity of V1 in visual cortex, these feature maps are first maxpooled among different scales, then maxpooled in certain grid size $Q$ but not pooled in orientations. As a result, the samples flowed into the subsequent PCANet have multiple orientations, specifically, there are four orientations here. The PCANet with Gabor filters followed maxpooling layer is dubbed as MulOri_PCANet and illustrated in Fig. 1.

### 3.2 Incremental PCANet

Since the training process of traditional PCANet need all data, it's inefficient when dealing with the large scale or time-varying data. Furthermore, it has no

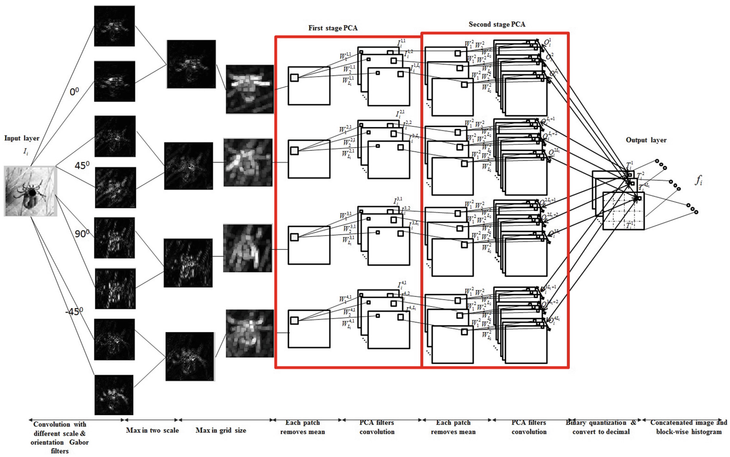



**Fig. 1.** Detailed structure of MulOri_PCANet.

plasticity. To this end, the incremental PCANet is introduced to explore the plasticity of feature extraction. For simplify, the incremental PCANet is dubbed as IncPCANet.

A. The first stage Incremental PCA: Assume the samples from the Gabor filters flow into the network batch after batch. Let $lth$ batch samples are $\{I_i^{o,l}\}_{i=1}^{N_l}\{o = 1, 2, 3, 4\}$ with size $m \times n$, $N_l$ is the number of input samples of $lth$ batch, $o$ indicts the $oth$ orientation. For each sample, around each pixel, patches with size $k_1 \times k_2$ pixels are extracted, converted into column vectors and concatenated together as $y_{i,1}^{o,l}, y_{i,2}^{o,l}, \cdots, y_{i,\tilde{m}\tilde{n}}^{o,l} \in \mathbb{R}^{k_1 k_2}$, each $y_{i,j}^{o,l}$ indicates the $jth$ patch in $I_i^{o,l}$, here, $\tilde{m} = m - \lceil k_1/2 \rceil$ and $\tilde{n} = n - \lceil k_1/2 \rceil$. Then the mean of patches are subtracted from themselves

$$\bar{Y}_i^{o,l} = [\bar{y}_{i,1}^{o,l}, \bar{y}_{i,2}^{o,l}, \cdots, \bar{y}_{i,\tilde{m}\tilde{n}}^{o,l}] \tag{2}$$

where $\bar{y}_{i,j}^{o,l}$ indicates a patch with its mean removed.

For each orientation samples, by applying the same operators for all input samples and concatenating them together, we obtain

$$Y^{o,l} = [\bar{Y}_1^{o,l}, \bar{Y}_2^{o,l}, \cdots, \bar{Y}_{N_l}^{o,l}] \in \mathbb{R}^{k_1 k_2 \times N_l \tilde{m}\tilde{n}} \tag{3}$$

Since the filter banks of PCANet in every layer are obtained via figuring out several eigenvalues of the convariance matrix $\Sigma$ correspond to the first largest eigenvalues. Here, $\Sigma$ is the convariance matrix of the mean-removed pathes of all training samples. To realize the incremental PCANet, instead of storing the mean-removed patches of all training samples of earlier batches, their convariance matrices are just stored and added into that of the current batch. This can significantly save the space and time.

Assume that the $pth$ layer has $L_p$ filters, the filters of the first stage incremental PCA layer can be expressed as

$$W_j^{o,l,1} = mat_{k_1 k_2}(q_p(\sum_{t=1}^{l} Y^{o,t} Y^{o,tT})) \in \mathbb{R}^{k_1 \times k_2}, j = 1, 2, \cdots, L_1 \tag{4}$$

which means the $jth$ filter bank of $lth$ batch of $oth$ orientation. The function $mat_{k_1 k_2}$ is to map the vector into a matrix with size $k_1 \times k_2$ and $q_p(XX^T)$ denotes the $pth$ principle eigenvector of $XX^T$. This step is crucial for realizing the incremental PCANet, we just need to add the matrix $Y^{o,t}Y^{o,tT}$ of the arriving batch to those of all earlier batches.

B. The second stage Incremental PCA: Repeating the same process of what we do in first stage incremental PCA, assume the $jth$ filter output of the $oth$ orientation of $lth$ batch is

$$I_i^{o,l,j} = I_i^{o,l} * W_j^{o,l,1}, i = 1, 2, \cdots, N_l \tag{5}$$

where * indicates 2D convolution. In order to keep the size of each output $I_i^{o,l,j}$ the same as $I_i^{o,l}$, the boundary of $I_i^{o,l}$ is zero-padded before it convolves with

$W_j^{o,l,1}$ . Like in the first stage, all of the overlapping patches of $I_i^{o,l,j}$ are collected and mean-removed to form

$$\bar{Z}_i^{o,l,j} = [\bar{z}_{i,j,1}^{o,l}, \bar{z}_{i,j,2}^{o,l}, \cdots, \bar{z}_{i,j,\tilde{m}\tilde{n}}^{o,l}] \in \mathbb{R}^{k_1 k_2 \times \tilde{m}\tilde{n}} \tag{6}$$

$\bar{z}_{i,j,k}^{o,l}$ means the $kth$ mean-removed patch in $I_i^{o,l,j}$.

Furthermore, let

$$Z^{o,l,j} = [\bar{Z}_1^{o,l,j}, \bar{Z}_2^{o,l,j}, \cdots, \bar{Z}_{N_l}^{o,l,j}] \in \mathbb{R}^{k_1 k_2 \times N_l \tilde{m}\tilde{n}} \tag{7}$$

represents all mean-removed patches of the $jth$ filter output of $oth$ orientation in $lth$ batch.

Concatenate $Z^{o,l,j}$ for all of the filter outputs as

$$Z^{o,l} = [Z^{o,l,1}, Z^{o,l,2}, \cdots, Z^{o,l,L_1}] \in \mathbb{R}^{k_1 k_2 \times L_1 N_l \tilde{m}\tilde{n}} \tag{8}$$

Further, gather $Z^{o,l}$ for all of the orientation outputs as

$$Z^l = [Z^{l,1}, Z^{l,2}, \cdots, Z^{l,4}] \in \mathbb{R}^{k_1 k_2 \times 4 L_1 N_l \tilde{m}\tilde{n}} \tag{9}$$

As in the first stage, the incremental PCA filters of $lth$ batch for second stage can be represented as

$$W_j^{l,2} = mat_{k_1 k_2}(q_l(\sum_{t=1}^{l} Z^t Z^{tT})) \in \mathbb{R}^{k_1 \times k_2}, j = 1, 2, \cdots, L_2 \tag{10}$$

For each input $I_i^{o,l,j}$ of the second stage, we can get $L_2$ output maps of size $m \times n$ via convoluting with $\{W_j^{l,2}\}_{j=1}^{L_2}$

$$O_i^{o,l,j} = \{I_i^{l,j} \times W_j^{l,2}\}_{j=1}^{L_2} \tag{11}$$

The number of output images at the second increment PCA stage is $4L1L2$. The subsequent binarization and histogram pooling can refer to the literature [10], and we don't describe it here.

### 3.3 Incremental PCANet with Incremental SVM

To analog the plasticity of classifier construction in primates' brain, instead of offline linear SVM classifier [22] used in PCANet [10], incremental SVM in literature [18] is adopted into our model. Since the original incremental SVM algorithm is utilized to deal with two classification problem, we extend it to solve multi classification via one vs one technique here. The IncPCANet with incremental SVM formed a lifelong learning framework that has the plasticity of both feature and classifier constructions.

## 4 Experiments

To validate the effectiveness of our model, several experiments have been done. They are detailedly depicted as follows.

### 4.1 Mulori_PCANet: Robust to Input Samples with Diverse Scales or Shifts

For purpose of enforcing the scale and shift invariance of input images flowed into PCANet, Gabor filters with 2 scales $s = \{3, 5\}$ and four orientations $\theta = \{-45, 0, 45, 90\}$ are utilized to convolute with them and their responses are maxpooled among different scales and certain gride size $Q = 4$.

To varify the performance of MulOri_PCANet, original PCANet [10], AlexNet [3] and ScatNet [14] are utilized as comparisons. For fair comparison, except the parameters of Gabor filters, the other parameters utilized in PCANet and MulOri_PCANet are set with the same values, the filter size $k_1 = k_2 = 7$, the number filter $L_1 = L_2 = 8$, the block size and the block overlap ration used in block-wise histogram pooling are set as $7 \times 7$, 0.5 respectively. In ScatNet, the number of scales and orientations are set as 3 and 8. The parameters of AlexNet are listed in Table 1 and the mean file of imagenet is utilized here.

**Table 1.** The parameters of AlexNet used on different databases.

| Parameters | Test_iter | Test_interval | Lr | Lr_policy | $\gamma$ | Stepsize | Max_iter | Momentum | Weight_decay |
|---|---|---|---|---|---|---|---|---|---|
| Mnist variations | 100 | 500 | 0.01 | INV | 0.1 | 10000 | 10000 | 0.9 | 0.0005 |
| Caltech101 | 100 | 1000 | 0.01 | STEP | 0.1 | 10000 | 45000 | 0.9 | 0.0005 |

Databases: MNIST variations [23], including background noise, rotations, and background images, to MNIST. The Caltech101 dataset consists of 101 objects and a background category, it can be downloaded from http://www.vision.caltech.edu. In each MNIST databases, there are 10000 train samples and 50000 test samples. For Caltech101, 30 images per category form the training samples and the rest images are used as test samples.

For fair comparison, the linear SVM classifier [22] is utilized for both MulOri_PCANet and PCANet [10].

The results can be found in Table 2. From it, we can see that the performance of our model uniformly better than PCANet and ScatNet-2 most cases, which shows the robustness of MulOri_PCANet. Although MulOri_PCANet performs not better than AlexNet sometimes, it has no data augmentation and the filter learning in MulOri_PCANet don't include adjustments of parameters, meanwhile, its unsupervised learning process with very simple structure is also attractive.

**Table 2.** The error rate of different algorithms based on several databases.

| Databases | bg-img-rot | bg-img | rot | Caltech101 |
|---|---|---|---|---|
| MulOri_PCANet | 32.48 | 9.88 | 5.89 | 29.52 |
| PCANet | 35.48 | 10.95 | 7.37 | 31.54 |
| ScatNet-2 [14] | 50.48 | 18.40 | 7.48 | 46.04 |
| AlexNet [3] | 19.26 | 4.40 | 7.06 | 39.30 |

### 4.2 IncPCANet: Achieve the Plasticity of Feature Extraction

Inspite great success achieved in many image processing fields, most bio-inspired methods are infeasible when handling the large scale or flow data. This is due to they need all samples available and lack of plasticity. To this end, IncPCANet is proposed, aims at exploring the plasticity of feature extraction.

To validate the effectiveness of IncPCANet, we compare the performance of IncPCANet and traditional PCANet on database with identical distribution, specifically, it is Caltech101. The number of training and testing samples are set the same as in Sect. 4.1. For each database, the whole training samples are divided into 10 batches, these batches are sent to the IncPCANet continuously, after each batch pass, the test samples are utilized to test the model.

To evaluate the effects of filter banks learned by IncPCANet and traditional PCANet, the same number training samples are utilized to train their filter banks. Specifically, the filter banks in PCANet are learned from all arriving batch samples, while those in incremental PCANet are updated just utilize the new arriving data based on earlier batches. For fair comparison, the features are used to train SVM classifier in IncPCANet are the features of all training samples extracted by the current filter banks every time.

The results can be found in Tables 3 and 4. With much less training time, IncPCANet can obtain the comparable results with traditional PCANet, specifically, the Training_Time here means the PCA filter banks' learning time. When dealing with large scale data, IncPCANet is a better choice.

To verify the plasticity of feature extraction of IncPCANet, we compare the performance of IncPCANet and PCANet on databases with different distributions. Here, the Mnist variations are utilized as a database with different distributions. The database is organized with the order of Mnist_basic, Mnist_bg_img, Mnist_bg_rand, Mnist_rot, each of them have 4000 training samples and 5000 test samples, further, the same distribution samples are arriving together and 1000 training samples are seen as a batch.

When the new batch samples arrive, the filter banks of PCANet are updated by utilizing all arriving batch samples no matter the distribution of data has changed or not. Nevertheless, the filter banks of IncPCANet are trained just using the

**Table 3.** The performance of Incremental PCANet based on Caltech101.

| Training numbers | 306 | 612 | 918 | 1224 | 1530 | 1836 | 2142 | 2448 | 2754 | 3060 |
|---|---|---|---|---|---|---|---|---|---|---|
| Accuracy | 66.64 | 60.43 | 51.26 | 39.77 | 38.18 | 34.56 | 31.55 | 30.78 | 30.45 | 29.81 |
| Training_Time | 26.09 | 26.03 | 26.82 | 26.72 | 26.74 | 27.48 | 29.13 | 26.95 | 27.09 | 26.95 |

**Table 4.** The performance of PCANet based on Caltech101.

| Training numbers | 306 | 612 | 918 | 1224 | 1530 | 1836 | 2142 | 2448 | 2754 | 3060 |
|---|---|---|---|---|---|---|---|---|---|---|
| Accuracy | 66.96 | 54.73 | 46.38 | 37.68 | 36.37 | 33.89 | 31.72 | 30.735 | 29.76 | 29.40 |
| Training_Time | 21.17 | 39.87 | 59.77 | 79.74 | 101.51 | 123.41 | 146.42 | 169.02 | 192.53 | 202.68 |

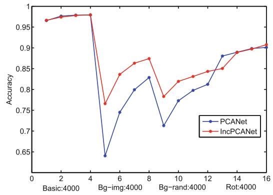



**Fig. 2.** The comparison of PCANet and IncPCANet based on Mnist database with different distributions.

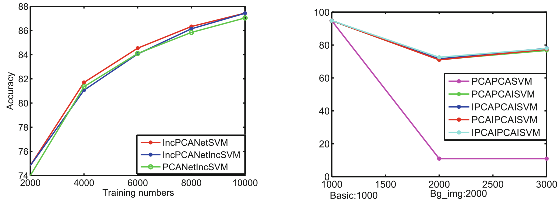
current arriving batch samples based on the earlier filter banks. The samples are used to train SVM classifier in PCANet are features of all training samples extracted by the current filter banks, while in IncPCANet are just features of all new distribution training samples extracted by the current filter banks.

The results are shown in Fig. 2. In Fig. 2, the name of arriving databases with different distributions and their training numbers are listed in X-axis. From it, we can see that when the distribution of data is changed, the accuracies of PCANet and IncPCANet first decrease and then increase gradually most times, and IncPCANet can obtain better performance than PCANet. This is because when the new data arrive, the filter banks obtained earlier are not suitable for the new situation, so the accuracy decreases. Further, with the number of new samples becomes more and more, the filter banks obtained can better represent the new data, lead to an increase accuracy. However, when Mnist_rot arrive, the performance of two models are not degenerate and IncPCANet first performs worse than PCANet while better than it at last. This is due to Mnist_rot samples have some similarities with the earlier databases, the filter banks of earlier samples can represent Mnist_rot at some extent, so the accuracy of Mnist_rot is not decrease. Furthermore, IncPCANet has less training samples than PCANet at first, which leads to a worse result compared to PCANet. Nevertheless, with the number of Mnist_rot samples becomes more, IncPCANet can quickly adjust the filter banks to better approximate the new data and obtain a better results. Above all, the experimental results in this section reveal IncPCANet has plasticity in feature extraction and owns better self-adaption and robustness compared to PCANet.

### 4.3 Validate the Effectiveness of the End to End Lifelong Learning Model Based on IncPCANet with Incremental SVM

The plasticity in primates' brain not only include the feature extraction but also classifier construction. Incremental PCANet with incremental SVM (dubbed as IncPCANetIncSVM) is expected here to analog this end to end plasticity. In this section, we first validate the effectiveness of IncPCANetIncSVM and in the next section further verify the plasticity in visual cortex.

To varify the performance of IncPCANetIncSVM, the incremental PCANet with offline SVM (IncPCANetSVM) and PCANet with incremental SVM (PCANetIncSVM) are employed as comparisons. Mnist_rot is utilized as their




(a) The comparison of different models based on Mnist_rot.

(b) The comparison of different models based on Mnist database with different distributions.

**Fig. 3.** The validations of the model's plasticities based on different levels.

databases and the numbers of training and testing samples are 10000 and 50000 respectively. For fair comparison, the offline SVM we also use the batch SVM offered in [18] other than linear SVM [22].

The result is shown in (a) subfigure of Fig. 3. Along with the number of training samples increased, the results of three models become better. Furthermore, with much less time and space cost, the IncPCANetIncSVM can obtain comparable or even better performance compared to other two models, which reveal the efficient and effectiveness of IncPCANetIncSVM.

### 4.4 Analog and Validate the Functions of Plasticity of Different Layers in Visual Cortex

Assume the databases with different distributions flow into the incremental PCANet, when the data distribution is changed, to validate the plasticity of which layer is crucial, we freeze some layer and adjust the rest layers. Here, the adjustment operations via incremental technique to realize.

- Feature layers and classifier layer with no adjustments (dubbed as PCAPCASVM)
- Feature layers with no adjustments and classifier layer with adjustment (dubbed as PCAPCAISVM)
- Low level feature layer and classifier layer with adjustments and high level feature layer with no adjustment (dubbed as IPCAPCAISVM)
- High level feature layer and classifier layer with adjustments and low level feature layer with no adjustment (dubbed as PCAIPCAISVM)
- Feature and classifier layers with adjustments (dubbed as IPCAIPCAISVM).

Mnist variations which include Mnist_basic and Mnist_bg_img are utilized as database with different distributions to validate the performance of above different models. Here the Mnist_basic has 1000 training samples and 5000 test samples, Mnist_bg_img has 2000 training samples and 5000 test samples.

The results are show in (b) subfigure in Fig. 3. From it, we can see that when the data distribution is changed, with no feature and classifier layers adjustments, the performance decreased heavily. As long as the classifier layer has adjusted, the accuracy increase dramatically and becomes better along with the number of training samples increase. This is because the high level classifier layer similar like the neural center, it's very effective by utilizing the supervised information to adjust the model. Furthermore, the feature layers adjustments also have effect in updating the model even the function of it is very smaller when compared to that of the classifier layer. Finally, the IPCAIPCAISVM has best results while the PCAPCAISVM has the worst among all compared models. These results validate the hypothesis of physiology that the plasticities exist in every layer of brain cortex and the plasticity in high level is more efficient and effective than the low level. However, it seems because the effect of the error accumulation, PCAIPCAISVM has no better results than IPCAPCAISVM. This is just the preliminary experiments and the further experiments are ongoing.

## 5 Conclusion

Inspired by the plasticity in primates' visual cortex, a lifelong learning framework based on incremental PCANet is proposed here to explore the plasticity of both feature extractor and classifier construction. Gabor filters with maxpooling layer are used to enforce the scale and shift tolerance to input samples. Incremental PCANet dedicated to investigate the plasticity of feature extraction and the incremental SVM devoted to validate the plasticity in classifier construction. Experiment results show that our model have better self-adaption and robustness compared to PCANet. Further, in the view of physiology, the incremental PCANet with incremental SVM has achieved the plasticity of both feature and classifier constructions. Finally, via our model, we validate the plasticity of high level classifier layer is much better than that of the low feature extraction layers, which is consist with the hypothesis of physiology.

## References

1. Jim, M., David, L.G.: Multiclass object recognition with sparse, localized features. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2006), vol. 1, pp. 11–18 (2006)
2. Rolls, E.T., Milward, T.: A model of invariant object recognition in the visual system: learning rules, activation functions, lateral inhibition, and information-based performance measures. Neural Comput. **12**(11), 2547–2572 (2000)
3. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems, pp. 1097–1105 (2012)
4. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
5. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–9 (2015)

6. Thorpe, S., Fize, D., Marlot, C.: Speed of processing in the human visual system. Nature **381**(6582), 520–522 (1996)
7. Schmidhuber, J.: Deep learning in neural networks: an overview. Neural Netw. **61**, 85–117 (2015)
8. LeCun, Y., Bengio, Y.: Convolutional networks for images, speech, and time series. In: The Handbook of Brain Theory and Neural Networks, vol. 3361, MIT Press, Cambridge (1995). (10)
9. LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. Nature **521**(7553), 436–444 (2015)
10. Chan, T.H., Jia, K., Gao, S., Lu, J., Zeng, Z., Ma, Y.: PCANet: a simple deep learning baseline for image classification? IEEE Trans. Image Process. **24**(12), 5017–5032 (2015)
11. Fukushima, K.: Neocognitron: a self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. Biol. Cybern. **36**(4), 193–202 (1980)
12. LeCun, Y., Boser, B., Denker, J.S., Henderson, D., Howard, R.E., Hubbard, W., Jackel, L.D.: Backpropagation applied to handwritten zip code recognition. Neural Comput. **1**(4), 541–551 (1989)
13. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. arXiv preprint arXiv:1512.03385 (2015).
14. Nguyen, V.L., Vu, N.S., Gosselin, P.H.: A scattering transform combination with local binary pattern for texture classification. In 2016 14th International Workshop on Content-Based Multimedia Indexing (CBMI), pp. 1–4, IEEE (2016)
15. Hegde, A., Principe, J.C., Erdogmus, D., Ozertem, U., Rao, Y.N., Peddaneni, H.: Perturbation-based eigenvector updates for on-line principal components analysis and canonical correlation analysis. J. VLSI Signal Process. Syst. Signal Image Video Technol. **45**(1–2), 85–95 (2006)
16. Weng, J., Zhang, Y., Hwang, W.S.: Candid: covariance-free incremental principal component analysis. IEEE Trans. Pattern Anal. Mach. Intell. **25**, 1034–1040 (2003)
17. Krasulina, T.: Method of stochastic approximation in the determination of the largest eigenvalue of the mathematical expectation of random matrices. Autom. Remote Control., 50–56 (1970)
18. Diehl, C.P., Cauwenberghs, G.: SVM incremental learning, adaptation and optimization. In: Proceedings of the International Joint Conference on Neural Networks, 2003, vol. 4, pp. 2685–2690. IEEE (2003)
19. Thrun, S.: Explanation-Based Neural Network Learning: A Lifelong Learning Approach. Kluwer Academic Publishers, Boston (1996)
20. Thrun, S., O'Sullivan, J.: Discovering structure in multiple learning tasks: the TC algorithm. ICML **96**, 489–497 (1996)
21. Carlson, A., Betteridge, J., Kisiel, B., Settles, B., Hruschka, E.R., Mitchell, T.M.: Toward an architecture for never-ending language learning. In: AAAI, vol. 5, p. 3 (2010)
22. Fan, R.E., Chang, K.W., Hsieh, C.J., Wang, X.R., Lin, C.J.: LIBLINEAR: a library for large linear classification. J. Mach. Learn. Res. 1871–1874 (2008)
23. Larochelle, H., Erhan, D., Courville, A., Bergstra, J., Bengio, Y.: An empirical evaluation of deep architectures on problems with many factors of variation. In: Proceedings of the 24th International Conference on Machine learning, pp. 473–480. ACM, June 2007