Enhanced Biologically Inspired Model

Yongzhen Huang[1], Kaiqi Huang[1], Liangsheng Wang[1], Dacheng Tao[2], Tieniu Tan[1] and Xuelong Li[3]
[1]National Laboratory of Pattern Recognition, Institute of Automation
Chinese Academy of Sciences, Beijing, China
{yzhuang, kqhuang, lswang, tnt}@nlpr.ia.ac.cn
[2]Biometrics Research Centre, Department of Computing
The Hong Kong Polytechnic University, Hong Kong
dacheng.tao@gmail.com
[3]School of Computer Science and Information Systems, Birkbeck
University of London, London, UK
xuelong@dcs.bbk.ac.uk

Abstract

It has been demonstrated by Serre et al. that the biologically inspired model (BIM) is effective for object recognition. It outperforms many state-of-the-art methods in challenging databases. However, BIM has the following three problems: a very heavy computational cost due to dense input, a disputable pooling operation in modeling relations of the visual cortex, and blind feature selection in a feedforward framework. To solve these problems, we develop an enhanced BIM (EBIM), which removes uninformative input by imposing sparsity constraints, utilizes a novel local weighted pooling operation with stronger physiological motivations, and applies a feedback procedure that selects effective features for combination. Empirical studies on the CalTech5 database and CalTech101 database show that EBIM is more effective and efficient than BIM. We also apply EBIM to the MIT-CBCL street scene database to show it achieves comparable performance in comparison with the current best performance. Moreover, the new system can process images with resolution 128×128 *at a rate of 50 frames per second and enhances the speed 20 times at least in comparison with BIM in common applications.*

1. Introduction

Automatic object recognition and fast detection are key components for many applications, e.g., video surveillance, multimedia database management, web content analysis, human computer interactions, and biometrics. In general, object recognition is a difficult task because of the wide variety of objects potentially to be recognized and the complexity and variety of backgrounds. In particular, efficient learning and robust recognition are challenged by variations in lighting, geometric transformations, pose variations, occlusion and clutter. In addition, there is a difficulty of recognizing object categories under conditions of great intra-class variability. Beyond categorizing objects into distinct groups, the question of inter-category relationships remains largely unexplored.

The last three years have witnessed the significance of object recognition and a large number of object recognition algorithms have been proposed. The appearance-based approaches mainly utilize global low level visual features, e.g., color, shape, and texture histograms [26, 29]. These methods ignore local discriminative information and are sensitive to lighting conditions, object poses, clutter and occlusions. Local feature based approaches combine the interest point detectors and local descriptors with spatial information. Representative local features include Harris [34], scale-invariant feature transform (SIFT) [21], gradient location and orientation histogram (GLOH) [22], rotation invariant feature transform (RIFT) [19], shape context [4], and histogram of gradients (HOG) [12]. Although these features are effective in describing local discriminative information, they lack higher level information, e.g., relations of local orientations. Moreover, though bag-of-features [20] and bag-of-keypoints [11] are efficient, they abandon structure information. In summary, all of them have their strength and weakness and perform poor on difficult tasks, e.g., working on the CalTech101 database [13].

Visual object recognition is a fundamental, frequently performed cognitive task for the human vision system and recent research in computer vision demonstrates that visual cognitive models are valuable in promoting the performance of object recognition, especially for difficult tasks.

978-1-4244-2243-2/08/$25.00 ©2008 IEEE

For example, Serre et al. [31] developed a biologically inspired model (BIM) for object recognition and it strictly follows the organization of the human visual cortex. In experiments on the CalTech101 and the MIT-CBCL Street Scene database, it recognizes objects at a level competitive to state-of-the-art approaches.

However, BIM suffers from the following problems:

- First, to increase the selectivity in BIM, in the prophase, an image is convoluted with Gabor filters of various scales and orientations. Then, to increase the invariance, the convoluted image must be matched with a large number of stored prototypes at every position and scale to find the best match. Such a density of input has a very heavy computational cost and, further, because it retains some noise with high response values, this approach tends to produce lots of mismatches.

- Second, in the process that complex cells of the visual cortex pooling over the afferent responses of simple cells, BIM adopts for the way in which complex cells of the visual cortex pool over the afferent responses of simple cells. BIM uses the maximum pooling operation (MAX operation), which retains only the max response in a local area. This approach increases invariance but it has been found [6] that units (simples cells), which fire most strongly, will strengthen the responses of their neighbors. Therefore, MAX operation may lose some informative input, e.g., neighbors of the strongest response.

- Finally, BIM uses a feed-forward framework that blindly selects features for combination. In this method, a feature has the best match between a convoluted image and a prototype. As this prototype is randomly sampled from convoluted images of positive samples, the reliability of a match depends on using a large number of prototypes, which mean very high feature dimensions. Therefore, the computational cost for matching stage is very heavy. For example, to obtain a good recognition performance on the CalTech101 database, BIM requires 5000 feature dimensions and it takes around 600 seconds for patch extraction and 160 seconds for C2 layer generation in dealing with an image with size 128×128 in MATLAB.

In this paper, we develop an enhanced BIM (EBIM) that reduces the computational cost and the risk of mismatch by removing uninformative input by imposing on BIM with the sparsity constraints. It improves the sensitivity and informativeness of the pooling operation model by applying a novel local weighted pooling operation which weights and then sums the max response and its neighbors. Finally, rather than using the feed-forward procedure in BIM, it selects effective features for combination based on a feedback procedure. It is worth emphasizing that the feedback procedure is consistent with the biological theory [16]. Empirical studies on the CalTech5 and CalTech101 databases have shown that EBIM is much more effective and efficient than BIM. For object detection based on MIT-CBCL Street Scene database, the proposed EBIM also achieves competitive performance in comparison with state-of-the-art algorithms while its computational cost is much less than conventional ones and it can be applied to real time applications directly.

The organization of this paper is as follows. In Section 2, we briefly introduce BIM, describe its three problems in detail and review representative extensions. In Section 3, we develop the EBIM and show how problems in BIM can be solved in EBIM. Section 4 details empirical studies on CalTech5, CalTech101, and the MIT-CBCL Street Scene database and shows EBIM is a competitive model for object recognition and detection in comparing with state-of -the-art algorithms. Section 5 concludes the paper.

2. BIM: Problems and Extensions

BIM consists of four layers of computational units: S1, C1, S2, and C2, where S and C are respectively simple and complex cells in the visual cortex [31]. In the following, we first describe operations and problems associated with each of these four layers. Representative extensions of BIM are reviewed at the end of this Section.

2.1. Computational units in BIM and Problems

S1 units: The units in the S1 layer correspond to simple cells in the visual cortex. These units combine initial inputs using a group of Gabor filters and each of which is the product of an elliptical Gaussian envelope and a complex plane wave,

$$F(x,y) = \exp(-\frac{x_0 + \gamma^2 y_0^2}{2\sigma^2}) \times \cos(\frac{2\pi}{\lambda} x_0) \qquad (1)$$

where $x_0 = xcos\theta + ysin\theta$, $y_0 = -xsin\theta + ycos\theta$, the range of x and y decides the scales of Gabor filters and θ controls orientations. Gabor filters have good spatial localization, orientation selectivity, and frequency selectivity. However, this kind of dense input results in a heavy computational cost in the S1 units, because each image has to be convoluted with Gabor filters of different parameters in BIM. Apart from heavy computational cost, such dense inputs are more likely to produce mismatches, because noises responses are retained for later match in the S2 units.

C1 units: The C1 units correspond to complex cells in the visual cortex. C1 units pool over S1 units using a maximum operation which keeps only the max response of a local area of S1 units from the same orientation and scale. The size of local area is decided by the scale band index of

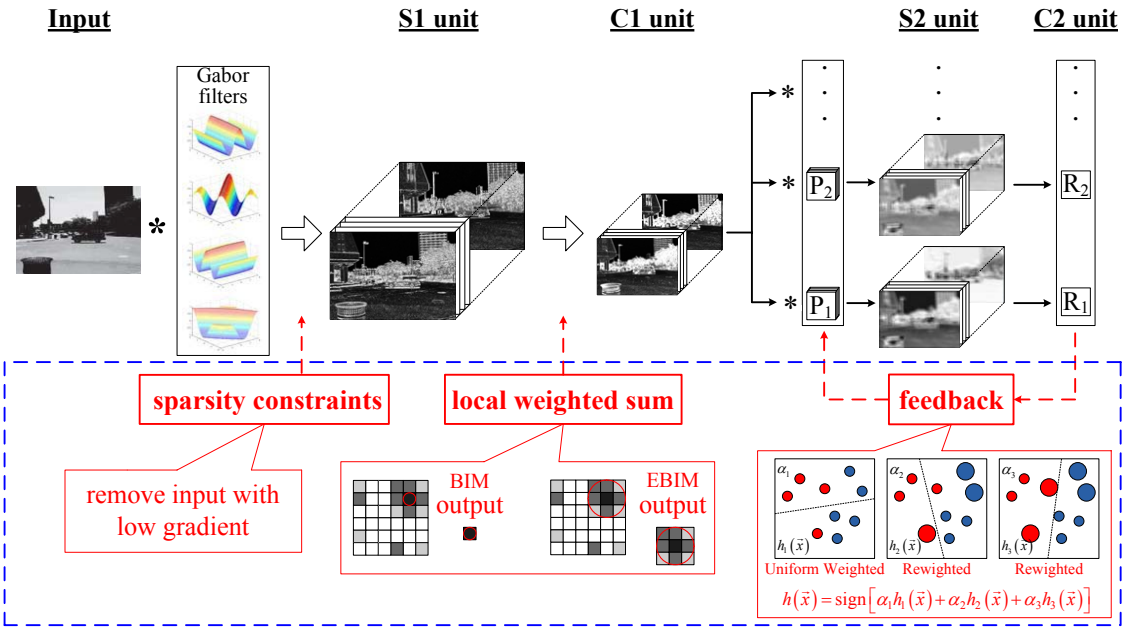

Figure 1. Framework of EBIM

the S1 units. Maximum operation shows some tolerance for shift and size but it loses some informative responses.

S2 units: The S2 units pool over C1 units in a Gaussian-like way between afferent C1 unit and a stored prototype,

$$\gamma = \exp(-\beta||X - P_i||^2) \tag{2}$$

where β defines the sharpness and P_i is one of the N features (prototypes). Eq. (2) reflects the similarity between an input image and stored prototypes. The prototypes are randomly sampled from C1 units of all positive samples. This process is computed in all orientations respectively. To achieve a good performance for recognition, a large number of prototypes should be sampled at random and this results in a very heavy computational cost.

C2 units: The C2 units are the global maximum over all scales and positions of S2 units in every direction. For an image, BIM will produce $N \times d$ maximum values, as the feature vector, where N is the number of stored prototypes and d is the number of orientations of Gabor filters.

Feed-forward scheme: BIM is a feed-forward procedure, which combines responses from S1, C1, S2, and C2 units. It blindly selects features because of the lack of a feedback stage to hint which features are important. Therefore, a large number of prototypes should be sampled at random for feature matching and thus the matching computational cost is very heavy.

2.2. Representative Extensions

Extensions of BIM have been developed recently. Wolf et al. [36] enumerated some of the alternative hierarchies for object recognition and empirically studied based on BIM showed that the strategy reverse-hierarchies for recognition could be effective. Mutch and Lowe [7] refined BIM in several biologically plausible ways using versions of sparsification, lateral inhibition, and feature selection. Stanley et al. [25] proposed four new image features, inspired by the gestalt principles of continuity, symmetry, closure, and repetition. All these extensions improved BIM in terms of effectiveness. However, all of them have very heavy computational cost for object recognition.

3. Enhanced Biologically Inspired Model

As discussed in Section 2, the conventional BIM has three particular drawbacks: very heavy computational cost associated with computing many inputs (input density), disputable MAX pooling operation in modeling relations of the visual cortex, and feed-forward framework based blind feature selection. In this section we describe our solutions to these drawbacks, the Enhanced Biologically Inspired Model (EBIM). EBIM responds to these problems by imposing sparse constraint, proposing a new pooling operation, and utilizing a feedback procedure for effective feature selection, as shown in Figure 1.

3.1. Sparsity of input information

BIM processes dense input, actually all pixels of an image, although only a very small part of the input is useful for classification tasks. The additional sparsity constraint can contribute to learning biologically plausible models from

the natural image statistics. Sparsity means that a random variable is far from the Gaussian-like distributions [18]. This constraint is important because first, it simplifies structures and reduces computational costs; second, it obtains a sparse estimation corresponding to performing feature or variable selection; and finally, it helps to enhance the generalization ability of learning machines, e.g., SVM [10, 32].

To take sparsity into account, we remove uninformative input, and retain only areas of interest. Unlike BIM, which computes every pixel of an image using all kinds of Gabor filters, the sparse approach in EBIM is concerned only with interesting points and associated neighbors. To implement this objective, we compute the horizontal and vertical gradients over outputs of S1 layer and retain special points, each of which satisfies the following condition:

$$|F_{x(i)}| + |F_{y(i)}| \geq \frac{\alpha}{n} \sum_{k=1}^{n} (|F_{x(k)}| + |F_{y(k)}|) \qquad (3)$$

where F_x and F_y are respectively horizontal and vertical gradients; n is the number of pixels in the image of the C1 layer; and α is a predefined constant for threshold control. We then dilate the filtered image several times so as to also retain the neighborhood around the interest points. This sparsity operation preserves informative pixels, which will be encountered in the later processes; and significantly reduces the number of pixels, which are uninformative. An example of sparsity constraint effect is shown in Figure 2.

3.2. A new pooling operation

Basically, there are three computational pooling models, which can be embedded in BIM. They are the maximum model, the energy model and the half-wave model [28], as shown in Figure 3. They connect simple cells with complex cells in the visual cortex of primates.

Maximum model: The response of complex cells is the maximum of the responses of all simple cells over a spatial neighborhood.

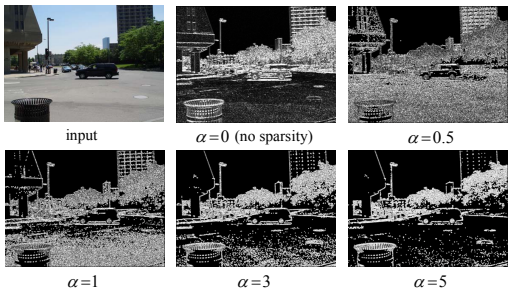

Figure 2. An example of sparsity constraint effect. α is the parameter in Eq.(3). The first image is input and others are different image of S1 layer of the input image.

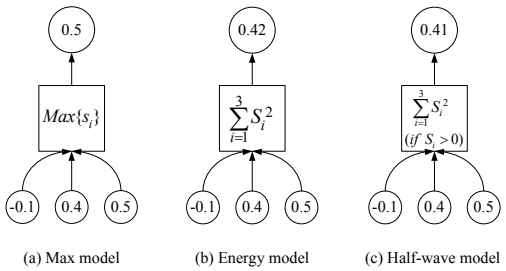

Figure 3. Three models of pooling operation.

Energy model: The response of complex cells is the linear summation of the energies (square values) of all simple cells in a certain area.

Half-wave model: The complex cells use the linear summation to pool outputs of energies of simple cells over a spatial neighborhood whose value is above a certain threshold.

Many theories about pooling functions for relationship modeling have been proposed. They have shown that each of these methods has its own strengths but describes partially the relationship between simple cells and complex cells. Lau et al. [23] found that the relation between the output of units (simple cells) and the output of the units (complex cells) is approximately quadratic (mean exponent 2.3±1.1). This proves the rationality of the energy model. Recently, Berkes and Wiskott [6] found that the most strongly activated units (simple cells) strengthen responses of their neighbors. Therefore, the weights of responses of simple cells are not identical in the pooling function. Based on above descriptions, we propose a new pooling function for EBIM.

In the EBIM model for pooling between simple and complex cells, we first find the maximal response and its neighbors; then other weak responses are removed due to the inhibition effect; and finally, we sum the energy of all responses remained by using different weights for S1 units. The weight of a point is defined as the inverse of the standard deviation of its neighbors. This definition has been shown to be effective in modeling feature relevance [9, 35, 27]. Therefore, the pooling function in EBIM is

$$C = \frac{1}{N_{I_0}} \sum_{x_i, y_i \in I_0} [w_i S^2(x_i, y_i)] \qquad (4)$$

where $S(x_i, y_i)$ is the response of the i^{th} simple cell; C is the responses of complex cells; I_0 is the neighborhood of the maximal response point in the local area of the domain of simple cells; N_{I_0} is the number of responses in I_0; and the weighting $W_i = \sqrt{D(I_i)}$. Here, D is deviation operation and I_i is the neighborhood of the i^{th} simple cell.

3.3. Feedback framework

BIM, a feed-forward system, consists of four levels and each level of the hierarchy is used only to produce the next level. That is once a level produces a new one, it can be discarded. However, a feedback system passes information from higher levels to lower levels for considering the combined information, as shown in Figure 1. This differs from the feed-forward mechanism that conforms to certain presumed constraints on high speed object recognition in the primate visual cortex, i.e. it takes place in the first 100-200 milliseconds. Feedback is important in a biologically inspired object recognition theory. Hochstein and Ahissar [16] developed reverse hierarchy theory (RHT). In this theory, visual information initially travels through the feed-forward visual hierarchy; and then reacts at higher levels. After that, visual information reaches lower levels via feedback connections, forming a reverse hierarchy. And in any case, the first 100-200 milliseconds reflects only one factor in recognition and need not to be a limiting factor to high level information, e.g., prior knowledge and inference. This is because the high level information serves as a significant role for classification tasks. The use of feedback is necessarily biological and it could be useful to combine high level and low level information in the training process. This inclusion of cognitive factors in object recognition is supported by Murphy and Medin [24] on similarity, which is not an absolute quality but rather a relative quality defined by feature selection and combination. Feedback is an important way to provide these varieties for different aspects. In any case, without feedback for feature selection and combination, BIM is left to randomly sample a large number of prototypes from C1 layer of positive samples with results that the computational cost is very heavy in the matching stage. Consequently, we utilize a feedback framework to modify random patch selection.

In EBIM, the feedback part consists of cascade-rejecters in the manner of Adaboost similar to [33]. Support vector machine (SVM) is selected as the weak classifier, although various classifiers, e.g., decision tree and neural networks, can be utilized as the same role. We choose SVM because SVM usually performs better than others and thus we can achieve a better convergence rate for AdaBoost in order to enhance efficiency. Apart from the efficiency obtained by combining AdaBoost and SVM, this combination reduces the classifier imbalance problem in SVM and generalizes better than a single SVM.

4. Experimental results

To prove the effectiveness and the efficiency of the proposed approach, we make comparisons on both object recognition and detection. For object recognition, we test the proposed approach on several public image databases, including CalTech5 [1] and CalTech101 [13], and baseline algorithms are BIM [30] and SIFT [21]. For object detection, we conduct comparison on MIT- CBCL street scene database and baseline algorithms are C1 [7], HoG [12], and BIM [30].All the experimental results are the average of ten independent tries using a PC with 3.4GHz CPU and 2GB memory. The final classifier we adopted is Lib-SVM [8].

4.1. CalTech5

Figure 4. Images from CalTech5 database. The last image is background.

This database contains 5 classes of objects: frontal-face, motorcycle, rear-car, airplane and leaf [1]. Examples of CalTech5 is shown in Figure 4. In this experiment, each category consists of several hundreds of images. We split each category into two parts for training and testing respectively and each part has 1/2 examples.

The performance measure reported is the classification accuracy at the equilibrium point, i.e., the classification accuracy at the point that the false positive rate equals to the miss rate (false negative rate).

To justify the effectiveness of each component of EBIM, three experiments are designed: first, we justify the effectiveness of the sparsity constraint by imposing it on BIM and performance curves shown in Figure 5 demonstrate that sparsity constraint is valuable to improve accuracy; second, we justify the effectiveness of the new pooling function by replacing the MAX with the new pooling function in BIM and performance curves shown in Figure 6 demonstrate that the new pooling function is useful to improve the classification accuracy; and finally, we justify the effectiveness of the feedback procedure by incorporating it into BIM and performance curves shown in Figure 7 demonstrate that feedback procedure is valuable for ameliorating the classification accuracy.

In Figure 5, results obtained by using BIM imposed with sparsity constraint are superior to BIM by setting the parameter α in Eq. (3) as a proper range, i.e. $0 < \alpha < 3.5$. Informative input may be removed if α is too large according to Eq. (3), which will result in the decrease of accuracy.

In Figure 6, results obtained by using a modified BIM, which MAX pooling function is replaced by the new pooling function, are superior to BIM.

In Figure 7, results obtained by BIM incorporated with a feedback procedure are superior to BIM, especially when

the dimension of C2 feature defined in section 2 is not too high. BIM is more sensitive to the number of C2 features because it samples patches at random to construct the final features. If the patches sampled in BIM are not enough, it is more possible that bad feature greatly affect the classification result. EBIM effectively chooses and combines features by feedback, which reduces the affection of bad features in BIM.

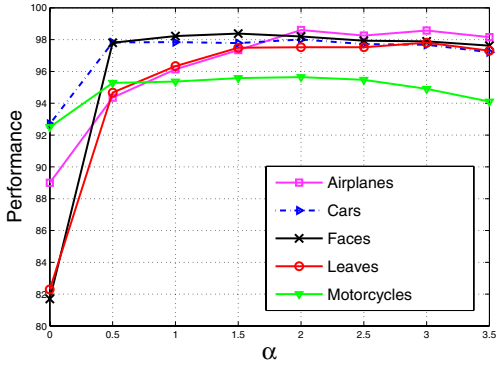

Figure 5. Comparison between BIM and EBIM according to sparsity of input information on CalTech5. α is the parameter in Eq. (3). $\alpha = 0$ denotes the performance of BIM

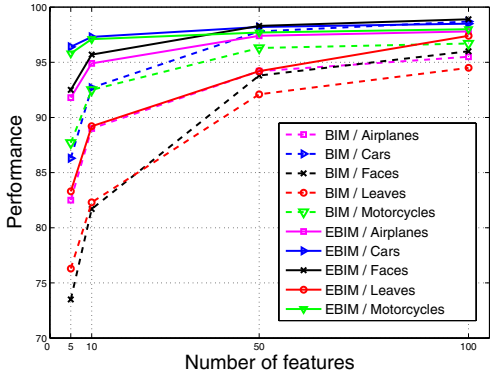

Figure 6. Comparison between BIM and EBIM on CalTech5 according to the new pooling function.

By combining all three new components together, we have EBIM and the performance is shown in Table 1. In accordance with this table, EBIM is superior to previous state-of-the-art algorithms.

The speed of EBIM is enhanced greatly especially when the number of feature is large. For example, BIM takes approximately 2 seconds to deal with an image (300×200) in the case of randomly sampling 1000 C2 features in the C++ version (completed by us) and more than 30 seconds with the Matlab version (completed by Serre et al. [2]). For the same task, EBIM needs only 0.1 second. Actually, for EBIM in common application, 100 features is enough.

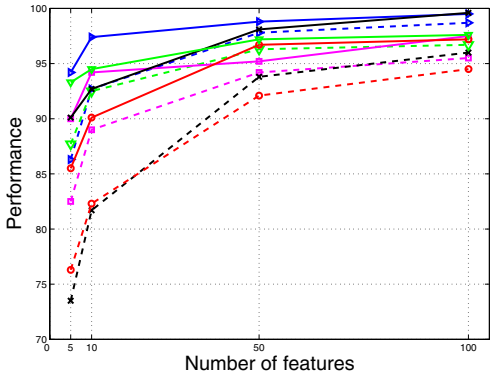

Figure 7. Comparison between BIM and EBIM on CalTech5 according to feedback framework.

Databases	Benchmark	SIFT	BIM	EBIM
Leaves	84.0	87.0	97.0	98.1
Cars	84.8	96.4	99.7	99.8
Faces	96.4	83.3	98.2	99.0
Airplanes	94.0	96.5	96.7	98.5
Motorcycles	95.0	99.5	98.0	98.3

Table 1. Results obtained with benchmark before BIM, SIFT with 1000 key-points, BIM with 1000 C2 features and EBIM with 1000 features.The results of Benchmark, SIFT and BIM are from [30]

4.2. CalTech101

Figure 8. Images from CalTech101 database

CalTech101 database contains 101 object classes plus a background class collected by Fei-Fei et al. [13]. There are about 40 to 800 images per category and most categories have more than 50 images. The size of each image is around 300×200. Example images are shown in Figure 8.

To conduct this experiment, we use 1000 features with best parameters learnt in CalTech5 test. The result reported here are the average and standard deviation, taken over all 101 classes, of the object recognition performance obtained from 10 independent trails. In each trial, 15 images are sampled at random for training and 50 images are sampled at random for testing. For classification, the pairwise SVM [15, 8] with majority voting rule is utilized.

Using this protocol, the performance of EBIM reaches

Dataset	Car		Pedestrian		Bicycle		
mearsure	tp@fp=fn	tp@fp=.01	tp@fp=fn	tp@fp=.01	tp@fp=fn	tp@fp=.01	time
BIM	90.0	51.0	82.5	45.0	88.5	51.0	$\approx 2s$
HoG	91.38	61.36	90.19	62.62	87.82	52.90	$\approx 0.5s$
C1	94.38	81.73	81.59	32.83	91.43	59.79	–
C1+Gestalt	96.40	90.90	95.20	85.20	93.80	84.70	$> 80s$
EBIM	98.54	96.02	85.33	70.00	96.49	93.43	$\approx 0.02s$

Table 2. Object detection results obtained by several state-of-the-art methods in the experiments of MIT-CBCL Street Scene database. "tp@fp=fn" denotes the true-positive-rate when false-positive-rate equals false-negative-rate. "tp@fp=.01" denotes the true-positive-rate when false-positive-rate is set to 1%. The last column is the averaged time cost to process a 128×128. Results of HOG, C1 and C1+Gestalt are obtained from [7]. Results of BIM are obtained from [30, 12] and EBIM is the proposed one.

above $49.8\% \pm 1.25\%$ correct classification rate. Some of the best performances achieved include: BIM [30] for $44\% \pm 1.14\%$, 51% in [25], 49.5% in [14], 44% in [17] and 45% in [5]. Although the training process takes several hours, the speed in test process is very fast: less than 30 second to deal with a test image with the resolution of 300×200 (Note that every test image needs to be classified more than 5000 times using all-pairs method in this experiment). As we know other methods are very time-consuming and nearly unbearable in this experiment, e.g., BIM combined with gestalt-like features [7] requires approximately 80 seconds to compute the feature vector for an image with size 128×128.

4.3. MIT-CBCL Street Scene database

MIT-CBCL Street Scene database [3], which is usually used for object detection, contains three kinds of objects: car, pedestrian and bicycle.

There are two groups of approaches for object detection: windowing based approaches and non-windowing based approaches. Windowing based approaches extract large number of (usually several thousands) image windows from an image at various scales and positions. Each of sampled windows has to be classified for a target object to be present or absent. Thus, windowing based approaches are time-consuming and not fit for fast practical applications. In this test, we use non-windowing strategy for detection based on MIT-CBCL street scene database, which equals to take the whole image as a window. Table 2 shows the performance comparison of EBIM with C1 [7, 30], HoG [12], and BIM [30].

In this object detection experiment, although EBIM uses non-windowing strategy, it achieves the best performance in car and bicycle detection and is comparable to C1+Gestalt [7] and HOG [12] in pedestrian detection. Moreover, EBIM significantly reduces the time complexity for detection, e.g., it works at a rate of about 50 frames per second and is ready for real-time applications. In comparison with C1+Gestalt, EBIM improves the detection speed around 4000 times. Demonstrations of detection are shown in Figure 9.

Figure 9. Demonstrations of detection in MIT-CBCL Street Scene database. Some best patches are selected from each of classes.

5. Conclusion

In this work, inspired by physiology and psychology, we have presented an enhanced biologically inspired model (EBIM) with three new components to improve Serre's biologically inspired model (BIM). These three particular components include: the sparsity constraint, a new pooling operation, and an AdaBoost based feedback procedure. They solve three popular problems in BIM respectively: a very heavy computational cost due to the dense input, a disputable pooling operation in modeling relations of the visual cortex, and blind feature selection in feed-forward frameworks. EBIM is superior to BIM in terms of both effectiveness and efficiency. Experiments on CalTech5 and CalTech101 have demonstrated that EBIM improves the accuracy and speed in object recognition. Further experiments in the MIT-CBCL street scene open database prove that it works effectively and efficiently for object detection application. It is worth emphasizing that EBIM accelerates biologically inspired methods 20 times at least in common applications.

In the future, we would like to further enhance EBIM in the following aspects: first, it is possible to combine some results in cognitive vision with image processing methods to reduce more redundant input in the proposed sparsity constrained input; second, there is a chance to consider weighting schemes in machine learning to improve the pooling function for recognition; and finally, variants of boosting schemes have been demonstrated to be more effective to enhance the classification accuracy, so to replace conventional AdaBoost with other machines will be reasonable to achieve some benefits, in terms of efficiency and effectiveness for object recognition and detection.

Acknowledgement

This work is supported by the National Basic Research Program of China (Grant No. 2004CB318110), the National Natural Science Foundation of China (Grant No. 60723005, 60605014, 60332010, 60335010 and 2004DFA06900), the CASIA Innovation Fund for Young Scientists and the Competitive Research Grants at the Hong Kong Polytechnic University (Project Number A-PC0A).

References

[1] http://www.robots.ox.ac.uk/vgg/data3.html. 5

[2] http://cbcl.mit.edu/software-datasets/index.html. 6

[3] http://cbcl.mit.edu/software-datasets/streetscenes. 7

[4] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Trans. on PAMI*, 24(4):509–522, 2002. 1

[5] A. Berg, T. Berg, and J. Malik. Shape matching and object recognition using low distortion correspondence. *CVPR*, 2005. 7

[6] P. Berkes and L. Wiskott. Slow feature analysis yields a rich repertoire of complex cell properties. *Journal of Vision*, 5:579–602, 2005. 2, 4

[7] S. Bileschi and L. Wolf. Image representations beyond histograms of gradients: The role of gestalt descriptors. *CVPR*, 2007. 3, 5, 7

[8] C. C. Chang and C. J. Lin. LIBSVM: a library for support vector machines, 2001. Software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm. 5, 6

[9] Y. Chen and J. Z. Wang. A region-based fuzzy feature matching approach to content-based image retrieval. *IEEE Trans. on PAMI*, 24(9):1252–1267, 2002. 4

[10] N. Cristianini and J. Shawe-Taylor. Support vector machines and other kernel-based learning methods. cambridge, 2000. U.K.: Cambridge Univ. Press. 4

[11] G. Csurka, C. Bray, C. Dance, and L. Fan. Visual categorization with bags of keypoints. *ECCV*, 2004. 1

[12] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. *CVPR*, 2005. 1, 5, 7

[13] L. Fei-Fei, R. Fergus, and P. Perona. Learning generative visual models from few training examples: an incremental bayesian approach tested on 101 object categories. *CVPR*, 2004. 1, 5, 6

[14] K. Grauman and T. Darrell. Pyramid match kernels: Discriminative classification with sets of image features, 2006. Technical Report MIT-CSAIL-TR-2006-020. 7

[15] T. Hastie and R. Tibshirani. Classification by pairwise coupling. *The annals of statistics*, 26(2):451–471, 1998. 6

[16] S. Hochstein and M. Ahissar. View from the top: Hierarchies and reverse hierarchies in the visual system. *J. Neuron*, 36(5):791–804, 2002. 2, 5

[17] A. Holub, M. Welling, and P. Perona. Exploiting unlabelled data for hybrid object classification. *Proc. Neural Information Processing Systems, Workshop Inter-Class Transfer*, 2005. 7

[18] A. Hyvarinen and P. O. Hoyer. A two-layer sparse coding model learns simple and complex cell receptive fields and topography from natural images. *Vision Research*, 41(18):2413–2423, 2001. 4

[19] S. Lazebnik, C. Schmid, and J. Ponce. A sparse texture representation using local affine regions, 2004. Technical Report, Beckman Institute, University of Illinois. 1

[20] T. Leung and J. Malik. Representing and recognizing the visual appearance of materials using three-dimensional textons. *International Journal of Computer Vision*, 43(1):29–44, 2001. 1

[21] D. G. Lowe. Distinctive image features from dcale-invariant key-points. *International Journal of Computer Vision*, 2(60):91–110, 2004. 1, 5

[22] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *IEEE Trans. on PAMI*, 27(10):1615–1630, 2005. 1

[23] P. Moreno and L. Wiskott. Slow feature analysis yields a rich repertoire of complex cell properties. *Journal of Vision*, 5:579–602, 2005. 4

[24] G. Murphy and D. Medin. The role of theories in conceptual coherence. *Psychological Review*, 92:289–316, 1985. 5

[25] J. Mutch and D. G. Lowe. Multiclass object recognition with sparse, localized features. *CVPR*, 2006. 3, 7

[26] W. Niblack, R. Barber, W. Equitz, M. Fickner, E. Glasman, D. Petkovic, and P. Yanker. The QBIC project: Querying images by content using color, texture and shape. *Proc. Storage and Retrieval for Image and Video Databases,SPIE*, 1993. 1

[27] Y. Rui, T. S. Huang, and S. Mehrotra. Content-Based image retrieval with relevance feedback in MARS. *IEEE Intl. Conf. on Image Processing*, 1997. 4

[28] K. Sakai and S. Tanaka. Spatial pooling in the second-order spatial structure of cortical complex cells. *Vision Research*, 40:855–871, 2000. 4

[29] B. Schiele and J. Crowley. Recognition without correspondence using multidimensional receptive field histograms. *International Journal of Computer Vision*, 36(1):31–35, 2000. 1

[30] T. Serre, L. Wolf, S. Bileschi, M. Riesenhuber, and T. Poggio. Robust object recognition with cortex-like mechanisms. *IEEE Trans. on PAMI*, 29(6), 1993. 5, 6, 7

[31] T. Serre, L. Wolf, and T. Poggio. Object recognition with features inspired by visual cortex. *CVPR*, 2005. 2

[32] V. N. Vapnik. Statistical learning theory, 1998. New York: John Wiley. 4

[33] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. *CVPR*, 2001. 5

[34] P. Viola and M. Jones. Robust real-time object detection. *Proc. Of IEEE Workshop on Statistical and Computa-tional Theories of Vision*, 2001. 1

[35] J. Z. Wang, G. Wiederhold, O. Firschein, and S. X. Wei. Content based image indexing and searching using daubechies wavelets. *J. Digital Libraries*, 1(4):311–328, 1998. 4

[36] L. Wolf, S. Bileschi, and E. Meyers. Perception strategies in hierarchical vision systems. *CVPR*, 2006. 3