

Object Recognition Inspiring HVS

Mohammadesmaeil Akbarpour, Nasser Mehrshad, Seyyed-Mohammad Razavi

Electrical Engineering, University of Birjand, Birjand, Iran

Article Info

Article history:

Received Apr 9, 2018

Revised May 20, 2018

Accepted Jul 11, 2018

Keywords:

Object recognition

HMAX

HVS

Hierarchical model

ABSTRACT

Human recognize objects in complex natural images very fast within a fraction of a second. Many computational object recognition models inspired from this powerful ability of human. The Human Visual System (HVS) recognizes object in several processing layers which we know them as hierarchically model. Due to amazing complexity of HVS and the connections in visual pathway, computational modeling of HVS directly from its physiology is not possible. So it considered as a some blocks and each block modeled separately. One models inspiring of HVS is HMAX which its main problem is selecting patches in random way. As HMAX is a hierarchical model, HMAX can enhanced with enhancing each layer separately. In this paper instead of random patch extraction, Desirable Patches for HMAX (DPHMAX) will extracted. HVS for extracting patch first selected patches with more information. For simulating this block patches with more variance will be selected. Then HVS will chose patches with more similarity in a class. For simulating this block one algorithm is used. For evaluating proposed method, Caltech 5 and Caltech101 datasets are used. Results show that the proposed method (DPMAX) provides a significant performance over HMAX and other models with the same framework.

Copyright © 2018 Institute of Advanced Engineering and Science.
All rights reserved.

Corresponding Author:

Mohammadesmaeil Akbarpour

Electrical Engineering, University of Birjand,

Birjand, Iran.

Email: esmaeil.akbarpour@birjand.ac.ir

1. INTRODUCTION

The Human Visual System (HVS) is able to recognize objects easily in a cluttered scene in less than a second. Recent works inspiring the Human Visual System (HVS) in image processing are such as image enhancement [1], data hiding [2, 3], digital image fusion [4], robust object recognition [5]. HVS processes images easily, while the most powerful computer systems are generally not capable of doing so. Due to the tremendous complexity of HVS and amazing connections in visual pathway, computational modeling of HVS for image processing applications directly from its overall anatomy and physiology is not possible [6]. One of way to overcome the limitation is the input-output modeling of the visual system (i.e. the saliency map) [7, 8].

Another way is modeling of the simple subsystems and their systematically combination based on the HVS structure (i.e. edge and line detection, contour extraction and texture diagnose) [9, 10]. It seems that the manner of the HVS in the object description stage and object recognizing (matching) process is optimized. In the first step of modeling the visual system behavior in object recognition, an appropriate object descriptor should be presented. This descriptor must be independent to scale and rotation [5, 11, 12].

HVS for object description uses processes such as saliency map, edge detection, line detection, contour extraction and texture diagnose. The saliency map [7, 8] is the first topographically arranged map that represents visual saliency of a corresponding visual scene. For edges detection [13, 14] the retina and LGN cells are inspired. They don't have directional selection because of the circular receptive field.

For line detection [15] the human primary visual cortex (V1) simple cells are inspired because of their directional receptive field. A set of adaptive filters is derived by learning mechanism which emulates the V1 simple cells. These filters are applied at every position of the input image to get a line feature representation. Non-classical receptive field (N-CRF) inhibition mechanism is an example to design physiologically plausible contour detection models [13, 16, 17]. N-CRF mechanism suppresses edges which make part of the texture, while it does not suppress edges that belong to the contours of objects.

Since HVS at first isolates contour of objects from scene images in its early stages of visual cortex, it is able to distinguish the texture edges and boundary of objects in scene images, known as the contrast [18, 19].

After object description, its recognition can be modeled by using the extracted features [20, 21]. For this purpose, in addition to simple cells, complex cells of the cortex must be modeled. Thus, the models are consist of two kinds of layers, each of which emulates the functions of V1 simple and complex cells. Thus, hierarchical models are created. One cases inspired by the hierarchical nature of primate visual cortex, is HMAX hierarchical model [22] (the neural network model for image classification). The HMAX model can be described as a four-level architecture with a first level consisting of multi-scale and multi-orientation local filters (i.e. Gaussian derivatives or Gabor filters). These networks combine the low level representations into object level representations suitable for recognition tasks [20].

Serre et al. [5] extended the original HMAX model to add multi-scale representations as well as more complex visual features. Huang et al. [38] also improved the HMAX model with constraints, a different pooling strategy and a feedback mechanism to improve feature learning. David et al. [23] has shown how HMAX filters can outperform state-of-the-art filters such as SIFT under various controlled invariance tasks on synthetic images.

Theriat et al. [22] First, HMAX was improved by integrating the local filters at the first level into more complex filters at the last level, providing a flexible description of object regions and combining local information of multiple scales and orientations. Second, a multi-resolution spatial pooling was introduced. This pooling encodes both local and global spatial information to produce discriminative image signatures.

Itti et al. [7] has introduced the hierarchical approach based on three parameters: intensity, orientation and colors for image saliency map diagnosis. The final model obtained by combining the output models of these features. In [34,35] saliency map are affected by the properties of the object. Poursad [24] proposed a Modified HMAX model based on combined with the visual featured model for HMAX (henceforth referred to as MHMAX) by calculating optimum patches based on their information. Ghodrati et al. [25] found the better patches for HMAX by using genetic algorithm (henceforth referred to as GMAX).

2. REVIEW OF BACKGROUND WORKS

In this section we introduce a brief review of HMAX models. The original HMAX model is inspiring the hierarchical theory of visual processing. Its architecture is derived from the well-known model introduced by Hubel & Wiesel [3,18sabouri4]. HMAX models the ventral visual pathway from first processing part in the visual cortex (V1) to higher levels of visual cortex (e. g. IT and PFCA). Schematic of the HMAX model is shown in Figure 1. As shown: The basic architecture of the HMAX model has four modules called and . The selectivity and invariance increase as the layers progress along the hierarchical structure of model. These layers imitate the behavior of cells from V1 to IT cortex. As the main problem of HMAX is random patch extraction and in this paper we solve it, it is considered as a separate module. Patches are extracted in training step and used in testing step. In the following, we explain these modules. Let the number of training image be denoted by N_{tr} , for each module we will discuss about input and output of module.

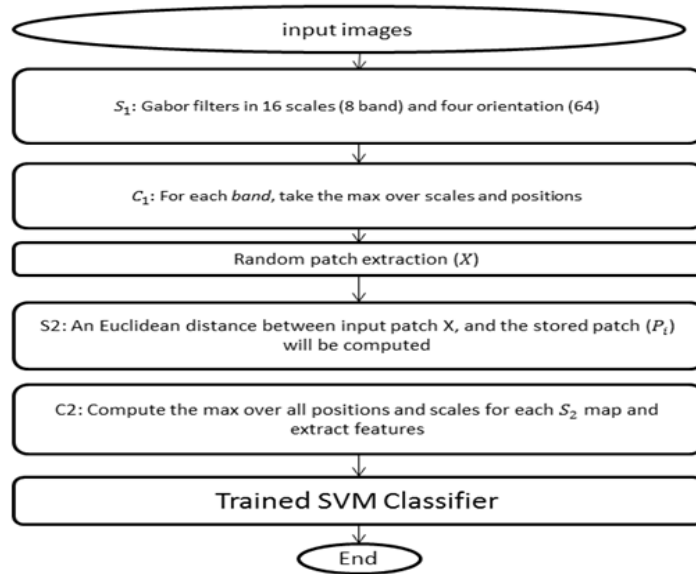


Figure 1. Schematic of the HMAX model

2.1. S1 Module

The first layer of the HMAX model called S_1 , receives images as its input. Afterward, these images are used to a set of edge detector filters to detect their edges. These filters are built based on the Gabor function. These Gabor filter formula is mentioned as Equation 1 and Equation 2:

$$g_{\sigma,\lambda,\theta}(x,y) = \exp\left(\frac{x_0^2 + \gamma y_0^2}{2\sigma^2}\right) \cdot \cos\left(\frac{2\pi}{\lambda} x_0\right) \tag{1}$$

$$x_0 = x \cos \theta + y \sin \theta, y_0 = y \cos \theta - x \sin \theta \tag{2}$$

Where the parameters γ , σ , θ , and λ are aspect ratio, effective width, orientation and the wavelength of the Gabor filter, respectively. The output of this module obtained by convolving the input image by Gabor filters.

This layer parameters (Gabor filter parameters) are shown in Table 1. As shown in Table 1, Gabor filter parameters are in 16 rows. Important note is these parameter are for each orientation ($\theta = \{0^\circ, 45^\circ, 90^\circ, 135^\circ\}$). For an input image, 64 images are produced such that edges are extracted with different sizes and orientations. Gabor filters mentioned in Table 1 (for 16 sizes and 4 orientations) are shown in Figure 2.

Table 1. S1 and C1 Parameters for each orientation (θ)

	S_1 module		C_1 module		O
	Gabor Filter parameter		MAX parameter		
Filter size	σ	λ	$N_s \times N_s$	Δ_s	
7 × 7	2.8	3.5	8 × 8	4	Band-1
9 × 9	3.6	4.6			
11 × 11	4.5	5.6	10 × 10	5	Band-2
13 × 13	5.4	6.8			
15 × 15	6.3	7.9	12 × 12	6	Band-3
17 × 17	7.3	9.1			
19 × 19	8.2	10.3	14 × 14	7	Band-4
21 × 21	9.2	11.5			
23 × 23	10.2	12.7	16 × 16	8	Band-5
25 × 25	11.3	14.1			
27 × 27	12.3	15.4	18 × 18	9	Band-6
29 × 29	13.4	16.8			
31 × 31	14.6	18.2	20 × 20	10	Band-7
33 × 33	15.8	19.7			
35 × 35	17.0	21.2	22 × 22	11	Band-8
37 × 37	18.2	22.8			

Note that the size of output and input are equal in this layer. As S1 module produces 64 outputs, the inputs of this module are Ntr training images and the outputs are 64Ntr images ($S_{1l,\theta}$).

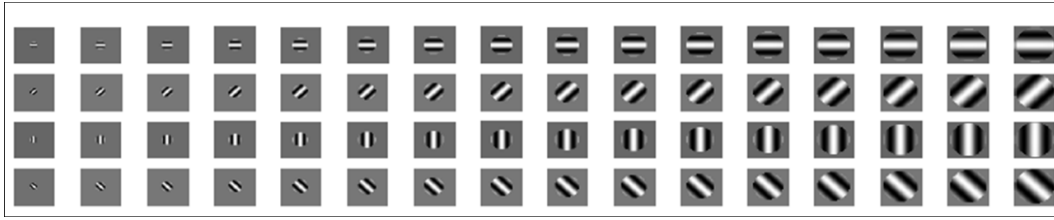


Figure 2. 64 Gabor filters (16 scales [7×7 to 37×37] by 4 orientations [0°, 45°, 90°, 135°]) [18]

2.2. C1 Module:

This module is the second layer in HMAX model which emulates the complex cells activity in the cortex. This layer parameter is shown in Table 1. As shown Band-7 contains scales 13,14 which are produced by Gabor filters with sizes 31×31 and 33×33. Let two consecutive scales be denoted by $S_{1l,\theta}$ and $S_{1l+1,\theta}$. These images are segmented into blocks with $N_s \times N_s$ size. Output of this layer is calculated as Equation 3:

$$C_{1b,\theta}(i,j) = \text{Max}\{S_{1l,\theta}(i,j), S_{1l+1,\theta}(i,j)\} \quad (3)$$

$$1 \leq i, j \leq N_s$$

where b is band, θ is orientation, and l are the number of scales. This process is for all orientations and all bands independently, so 32 matrices with $K \times L$ size, will produce for any input image (of size $K \times L$).

2.3. RPE Module:

This module is just for training step, this module extracts patches from each of the 32 output images ($C_{1b,\theta}$) produced by C1 module. For $C_{1b,\theta}$ from i -th training image ($i=1, \dots, N_{tr}$), patches with four sizes ($4h \times 4h$; $h=1,2,3,4$) are extracted. For each size, m patches are extracted. Hence for $C_{1b,\theta}$ from i -th training image, $4m$ patches will be extracted. Let this extracted patches be denoted by $P_{i,b,\theta,h,n}$, where i is the number of training image, b is the number of the band, θ is orientation, h is the patch size definer and n is the number of patch. These patches will be set as patches. The inputs of this module are $C_{1b,\theta}$ from C1 which for each band and orientation the number of them are N_{tr} , hence $4mN_{tr}$ patches for each band and orientation will be produced in total.

2.4. S2 Module:

The inputs of this module are taken from C1 and RPE modules. This module calculates the template matching between $C_{1b,\theta}$ from i -th ($i=1, \dots, N_{tr}$) training image and all patches in band b and orientation θ . As mentioned in RPE, the number of patches in band b and orientation θ are $4mN_{tr}$. Hence $4mN_{tr}$ matrices will be produced for each band and orientation and $32 \times 4mN_{tr} = 128mN_{tr}$ matrices for i -th training image. For all training image, S2 output is a cell with $N_{tr} \times 128mN_{tr}$ matrices. In this module, each patch $P_{i,b,\theta,h,n}$ is slid across an intermediate output matrix $C_{1b,\theta}$, and the template matching is calculated in a Gaussian-like way on the Euclidean distance between the local $C_{1b,\theta}$ block and $P_{i,b,\theta,h,n}$. Assume that the size of the patch $P_{i,b,\theta,h,n}$ is $W \times W$. Let the $W \times W$ block from e th $C_{1b,\theta}$, starting at coordinate (p,q) be denoted by X . The output $S_{2i,b,\theta,h,n}$ is then calculated as Equation 4:

$$S_{2i,b,\theta,h,n}(p,q) = \exp(-\beta \|X - P_{i,b,\theta,h,n}\|^2) \quad (4)$$

$$1 \leq p \leq K - W + 1, 1 \leq q \leq L - W + 1$$

Where i is the number of training image ($i=1, \dots, N_{tr}$) the parameter ($\beta > 0$) defines the sharpness of the exponential function and $\| \cdot \|$ is an Euclidian norm.

2.5. C2 Module:

The inputs of this module are S2 output matrices ($S_{2i,b,\theta,h,n}$). In this module, to find best matching, the maximum of these matrices are computed as Equation 5.

$$C2_{i,b,\theta,h,n} = \text{Max}\{S2_{i,b,\theta,h,n}(p, q)\} \quad (5)$$

$$1 \leq p, q \leq 140$$

HMAX in testing step:

The HMAX testing model is very similar to the training model except that, The testing model does not have Random Patch Extraction (RPE) module. As testing model does not have RPE module, S2 module in testing model uses the stored patch extracted in training model. In S2 module, the template matching is calculated in a Gaussian-like way on the Euclidean distance between the C1b,θ from testing images and stored patches in training model (Pi,b,θ,h,n). Hence in the same way, S2 has 128mNtr matrices for each testing image. Let the total number of testing images be denoted by Ntest. The output of S2 module will include is a cell with Ntest× 128mNtr matrices. C2 calculates the max of S2 output matrices. Hence the testing features will be a matrix with Ntest×128mNtr elements. Total tasks of HMAX model is shown in Figure 3.

Simple linear classifier: For image classification application, the output image features may be passed through a classifier (e.g. SVM) to classify an image.

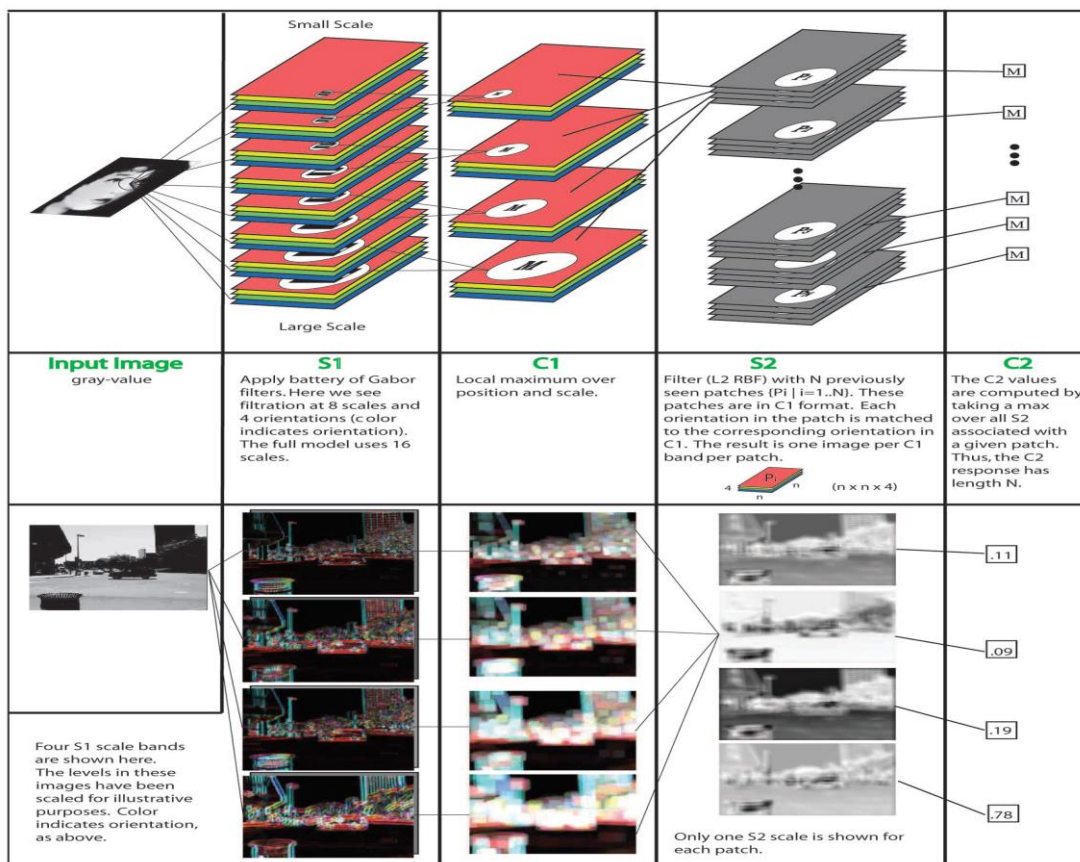


Figure 3. Total task of HMAX model [5]

3. PROPOSED TECHNIQUE

The basic architecture of the proposed model has six modules in training step and four modules in testing step (testing model is like HMAX testing model). The architecture is based on HMAX model. We introduce two main contributions to HMAX model that improve the recognition rate of HMAX model. First contribution is eliminating of background to avoid patch extraction from background that there is no appropriate information, second contribution is using optimum patch for HMAX model instead of random patch, third contribution is using 12 orientations instead of 4 orientations (the third contribution have done before). The proposed training step is different and has six modules: EB, S1, C1, OPE, S2, C2. EB is a new module, OPE is instead of RPE, S1 and C1 are based on HMAX model but they are slightly different, finally

other modules (S2, C2) are as the same as the HMAX modules. In the following we explain our modules separately:

3.1. EB:

In HMAX we extracted Random patches from image, first step for extracting optimum patch is eliminating background. When our system use patches from background there will be two important problems: first we will force system to do additional computational work second we will direct our system to the wrong result, because of our wrong information. An example of eliminating background on airplane from CALTECH101 dataset is shown in Figure 4. In this module we first, eliminate some rows of pictures from up and down of it, because almost there is no image information in these places but there is some information like explain about images that they are not useful. Then we use eliminate background algorithm [26]. After using eliminate algorithm we will extract four points: minimum row, maximum row, minimum column and maximum column. We extracted windowing image with rows from minimum row to maximum row and column from minimum column to maximum column.



Figure 4. CALTEC101 airplane image after eliminating background

3.2. S1:

In this module we use Gabor filter with Table 1. Parameters, like original HMAX model. But we use Gabor filters in 12 orientation as Equation 6.

$$\theta = \frac{k\pi}{12}, k = \{0,1, \dots, 11\} \quad (6)$$

3.3. C1:

In original HMAX model the patches are extracted from C1 module across all four orientations, patches sizes are $n \times n$ ($n=4,8,12,16$). In our model we extracted patches with one size for every band. As we explain before, we get max with windows from Table 1. So the output of C1 module is different, for example the size of output of C1 module for image with size of 140×140 is from 30×30 to 10×10 . This should be mentioned that if after eliminating background, if the size of output is small we should resize it to prepare size (i.e, 140×140).

3.4. OPE:

In HMAX model patches are extracted randomly. In our model, first backgrounds are eliminated, then among the image the optimum patches are extracted. We extracted optimum e in two steps:

Step1:

As HVS is sensitive to edges and lines, so it seems that optimum patches should be involve more edges and lines. Hence for our model, patches with more edges and lines are extracted instead of the dense inputs and blind patch selection in HMAX model. As seen in the Algorithm1, choosing optimum patch is based on choosing mm random patch and arrange them based on variance. Suppose the number of optimum patch is m and in this algorithm we find less optimum patch than m. So it is the problem of this algorithm. For solving this problem, first we select m random patches and save their value as an array. We arrange them according to their variance from worst to best. We repeat patch selection in random mode for mm times. If variance of new patch is more than the variance of the best random it will be one step for this algorithm. We will shift all patches and replace the best patch with new patch. In this step we will extract patches with more information (with more lines and edges).

Algorithm 1 choosing optimum patch

- 1) Extract m random patches (like HMAX), arrange them from worst to best and save them as m best patches.
- 2) Repeat random patch extraction for mm times, If $var(new\ patch) > var(old\ patch)$ then:
 - (I) shift all saved patches from best to worst.
 - (II) replace the best patch with new patch.

We sure that there will be m best patches in $(m+mm)$ iterations.
- 3) These m patches are input of $f(P_{i,j})$ according to Equation 8. and we will find the minimum of this function. Then $P_{i,j}$ will be the best patches.

Step2:

In this step we will extract patches with the best performance, among the patches that were extracted in step1. HVS in classification finds a patches that are the same in one category and has a most difference with other category. So we will extract patches that have most similarity with other patches in the same category and in other hand have the most difference with patches in other categories. For doing this with define a formula like Equation 8. With the best patches this formula will be minimum. So in simulation if we find minimum of this formula, the patches will be the best patches that have the maximum similarity with other patches in the same category and have most difference with other patches in other categories. We use equation for each band and orientation separately. We show the class number of patch with i and the j is the number of training image that patch is extracted ($P_{i,j}$). Then the function definition is as Equation 7:

$$f(P_{i,j}) = \frac{\sum_{l=1}^{CN} \sum_{k=1}^{TN} \sum_{l=k+1}^{TN} \|P_{l,k} - P_{l,i}\|^2}{\sum_{i=1}^{TN} \sum_{k=1}^{CN} \sum_{l=k+1}^{CN} \|P_{k,i} - P_{l,i}\|^2} \times a \tag{7}$$

$$a = \frac{CN(CN-1) \times TN}{TN(TN-1) \times CN}$$

Where CN in the number of class and TN is the number of Training image in each class. The total number of pair patches in each class are $TN(TN-1)$, so for CN class we will have $TN(TN-1) \times CN$ pair patches in same category. In the same way we will have $CN(CN-1) \times TN$ pair patches in different category. It is important that value of pairs be equal so we multiple the result with second term. This module activity is explained in Algorithm 1. As explained before, in HMAX model patches are extracted randomly in RPE module, which may not be optimal. The extracted patches by the HMAX model are from random position. So, these patches may come from background or other irrelevant objects rather than the target object. In the other hand they may come from target object but all the pixels are the same, so there is no information in it. The C2 features that are obtained from these patches are not very useful for recognizing a target object. They may also make the feature space more complex for classification [22]. The proposed module (OPE) selects patches with more information by employing optimum patch selection method which is explained in algorithm 1. As a result, less patches in training step in our model have the same result in training step in HMAX model that is good for system speed.

3.5. S2 and C2:

Two modules S2 and C2 are as the same as original HMAX model and we have explained it before.

4. EXPERIMENTAL RESULTS

We have tested our model on the CalTech5 and CalTech101 dataset of images. We have compared the proposed model (DPHMAX) with SIFT [?], HMAX [?], GMAX [] and MHMAX models. Our results show that there are significant improvements to classification in our model. The proposed algorithm is implemented with MATLAB.

4.1. Image Datasets

To evaluate our model on classification tasks, we use CalTech5, Caltech101 and GRAZ-01 datasets. CalTech5: This dataset [26] contains five classes of objects: the frontal-face, motorcycle, rear-car, airplane and leaf. The sample images of each category is shown in Figure 5. CalTech101: This dataset [27] contains 101 classes of objects such as: boat, car-side, bike, airplane, etc. The sample images are shown in Figure 6. GRAZ-01: This dataset [28] contains many different object like: bikes, people, plant, building, shoes, etc.

people and bike considered as a positive images and other images are as background. The sample images are shown in Figure 7.

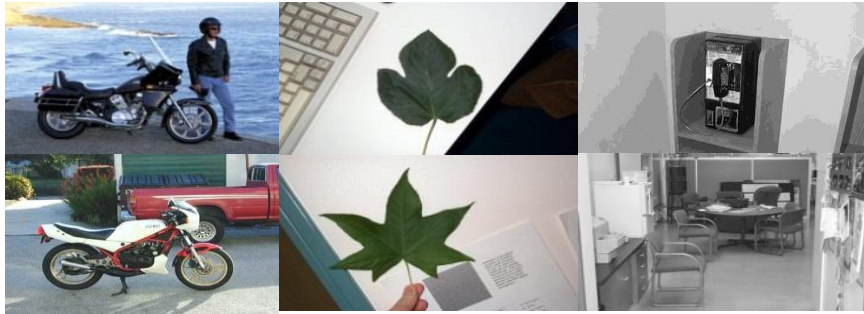


Figure 5. Sample images from Caltech5 database. From left to right: motorbike, leaf and background



Figure 6. Sample images from Caltech101. From left to right: airplane, car-side and background



Figure 7. Sample images from GRAZ-01. From left to right, the classes are bikes, people and backgrounds

4.2. Performance Measures

After classification, the images categorize in 4 group: True Positive (TP) is a correct classification of a positive (object), a True Negative (TN) is a correct classification of a negative (background), False Positive (FP) is an incorrect object classification and False Negative (FN) is an incorrect background classification. We chose the evaluation metrics of classification rate defined as Equation 8:

$$Classification\ Accuracy\ (A_c) = \frac{TP+TN}{TP+TN+FP+FN} \tag{8}$$

4.3. Classification results

We randomly chose 10, 20, 30, 40 and 50 images from each category of the datasets as training images, other images are considered as testing images. for Caltech5 and Caltech101 datasets, objects are positive images and background is negative images. for GRA-01 dataset, people and bike are positive images and other images are negative images (we make a negative images from combining other categories). All results reported were generated with 10 random splits. The results are shown in Figure 8. As shown, except in car-side from Caltech101, the proposed method provides a significant performance improvement over other methods.

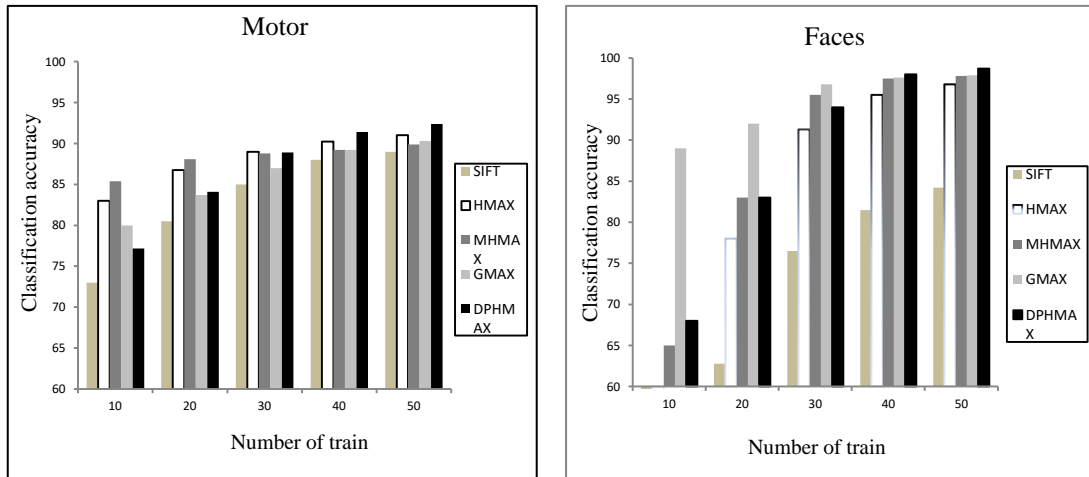


Figure 8(a). Classification accuracy for each class: Caltech5 dataset classification accuracy (motor and face)

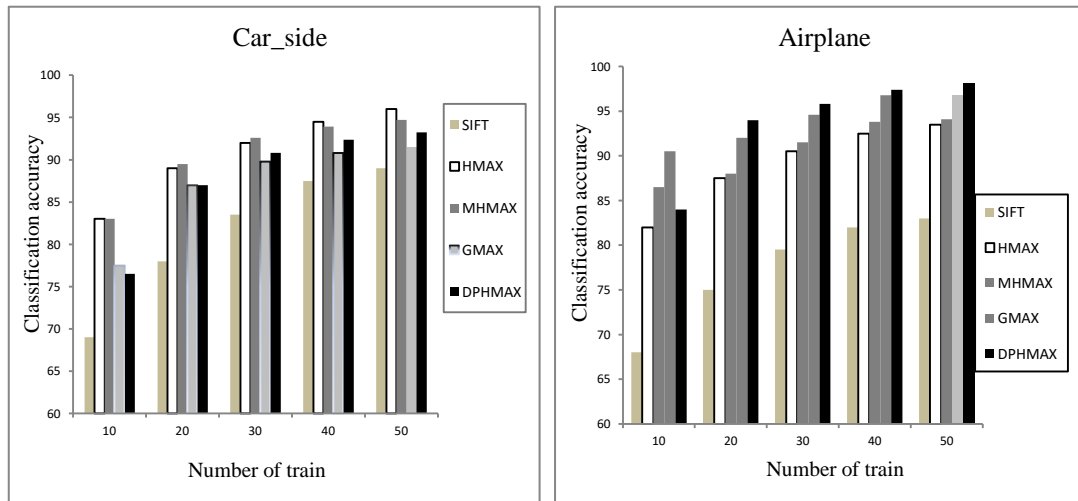


Figure 8(b). Classification accuracy for each class: Caltech101 dataset classification accuracy (car-side and airplane)

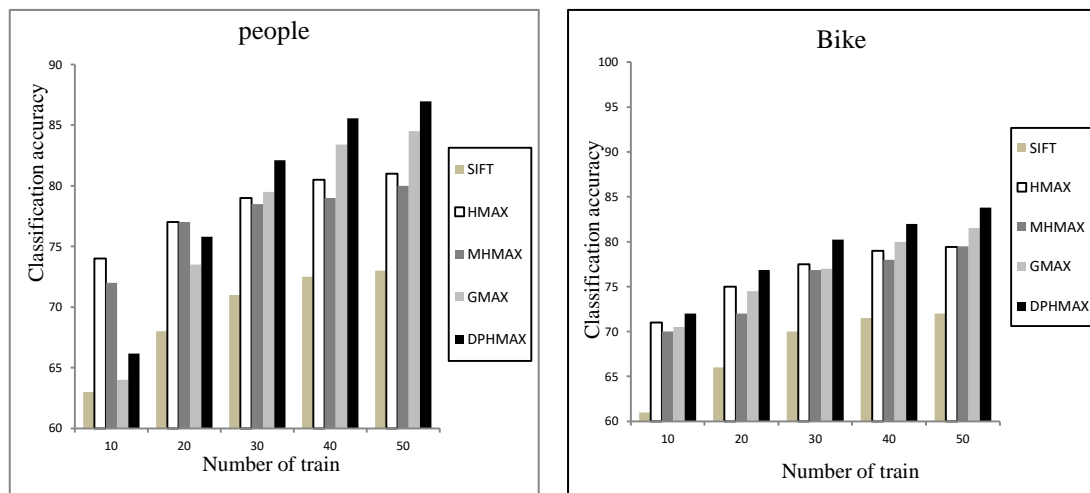


Figure 8(c). Classification accuracy for each class: GRAZ-01 dataset classification accuracy (people and bike)

5. CONCLUSION

In this paper a new framework for robust object recognition based on previous hierarchical model is proposed. A novel Desirable Patch extraction method for HMAX hierarchical model (OPHMAX) is proposed. For this purpose, first we eliminate the background, then we extracted the best patches that they are the same in the same category and they are more different in the different category. This improvement is only in training stage so it do not need more time in recognition stage. Optimum patches are the best patches to represent discriminative and invariant features. They solve the main limitation in the HMAX “dense inputs and blind feature selection”. Experiments based on three different kinds of datasets have demonstrated that the DPHMAX performs much better than the other methods in various visual recognition tasks.

REFERENCE

- [1] H. Yin, G. Lyu, X. Luo, and C. Li, "A segmentation-based adaptive image enhancement method inspired by the self-adjust features of HVS," *International Journal of Machine Learning and Cybernetics*, vol. 8, pp. 1895-1905, 2017.
- [2] R. Kumar, S. Chand, and S. Singh, "An Improved Histogram-Shifting-Imitated reversible data hiding based on HVS characteristics," *Multimedia Tools and Applications*, pp. 1-13, 2017.
- [3] S. Shafee and B. Rajaei, "A secure steganography algorithm using compressive sensing based on HVS feature," in *Emerging Security Technologies (EST), 2017 Seventh International Conference on*, 2017, pp. 74-78.
- [4] V. Radhika, K. Veeraswamy, and S. S. Kumar, "Digital Image Fusion Using HVS in Block Based Transforms," *Journal of Signal Processing Systems*, pp. 1-11, 2017.
- [5] T. Serre, L. Wolf, S. Bileschi, M. Riesenhuber, and T. Poggio, "Robust object recognition with cortex-like mechanisms," *IEEE transactions on pattern analysis and machine intelligence*, vol. 29, 2007.
- [6] M. Mishkin and L. G. Ungerleider, "Contribution of striate inputs to the visuospatial functions of parieto-preoccipital cortex in monkeys," *Behavioural brain research*, vol. 6, pp. 57-77, 1982.
- [7] L. Itti and C. Koch, "A saliency-based search mechanism for overt and covert shifts of visual attention," *Vision research*, vol. 40, pp. 1489-1506, 2000.
- [8] W. Zhang, Q. J. Wu, G. Wang, and H. Yin, "An adaptive computational model for salient object detection," *IEEE Transactions on Multimedia*, vol. 12, pp. 300-316, 2010.
- [9] D. Wiesel and T. Hubel, "Receptive fields of single neurones in the cat's striate cortex," *J. Physiol*, vol. 148, p. 2, 1959.
- [10] B. A. Olshausen, "Emergence of simple-cell receptive field properties by learning a sparse code for natural images," *Nature*, vol. 381, pp. 607-609, 1996.
- [11] F. J. Huang, Y.-L. Boureau, and Y. LeCun, "Unsupervised learning of invariant feature hierarchies with applications to object recognition," in *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, 2007, pp. 1-8.
- [12] M. Riesenhuber and T. Poggio, "Hierarchical models of object recognition in cortex," *Nature neuroscience*, vol. 2, pp. 1019-1025, 1999.
- [13] J. C. Bezdek, R. Chandrasekhar, and Y. Attikouzel, "A geometric approach to edge detection," *IEEE Transactions on Fuzzy Systems*, vol. 6, pp. 52-75, 1998.

- [14] P. Meer and B. Georgescu, "Edge detection with embedded confidence," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 23, pp. 1351-1365, 2001.
- [15] G. Papari and N. Petkov, "Edge and line oriented contour detection: State of the art," *Image and Vision Computing*, vol. 29, pp. 79-103, 2011.
- [16] C. Grigorescu, N. Petkov, and M. A. Westenberg, "Contour detection based on nonclassical receptive field inhibition," *Image Processing, IEEE Transactions on*, vol. 12, pp. 729-739, 2003.
- [17] Z. Li, "A neural model of contour integration in the primary visual cortex," *Neural computation*, vol. 10, pp. 903-940, 1998.
- [18] M. S. Landy and N. Graham, "73 Visual Perception of Texture," *The visual neurosciences*, vol. 1, p. 1106, 2004.
- [19] Fakoor, M., Kosari, A., Jafarzadeh, M. Revision on fuzzy artificial potential field for humanoid robot path planning in unknown environment, *International Journal of Advanced Mechatronic Systems*, Volume 6, Issue 4, page 174-183, 2015.
- [20] Fakoor, M., Kosari, A., Jafarzadeh, M. *Journal of Applied Research and Technology*, *Journal of Applied Research and Technology*, Volume 14, Issue 5, 2016, 300-310, 2016.
- [21] Z. Li, Z. Hu, X. Du, S. Fu, and Y. Ran, "Feature extraction for object recognition based on visualcortex-like mechanisms," *Energy Procedia*, vol. 13, pp. 3499-3506, 2011.
- [22] C. Theriault, N. Thome, and M. Cord, "Extended coding and pooling in the hmax model," *IEEE Transactions on Image Processing*, vol. 22, pp. 764-777, 2013.
- [23] N. Pinto, Y. Barhomi, D. D. Cox, and J. J. DiCarlo, "Comparing state-of-the-art visual features on invariant object recognition tasks," in *Applications of computer vision (WACV), 2011 IEEE workshop on*, 2011, pp. 463-470.
- [24] Y. Pourasad, "Developing a Modified HMAX Model Based on Combined with the Visual Featured Model," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 7, pp. 773-785, 2017.
- [25] M. Ghodrati, S.-M. Khaligh-Razavi, R. Ebrahimpour, K. Rajaei, and M. Pooyan, "How can selection of biologically inspired features improve the performance of a robust object recognition model?," *PloS one*, vol. 7, p. e32357, 2012.
- [26] <http://vision.caltech.edu/archive.htm>.
- [27] [http://www.vision.caltech.edu/Image_Datasets/Caltech101/.](http://www.vision.caltech.edu/Image_Datasets/Caltech101/)"
- [28] http://www.emt.tugraz.at/~pinz/data/GRAZ_01/