

Brain Programming and the Random Search in Object Categorization

Gustavo Olague¹(✉), Eddie Clemente², Daniel E. Hernández³,
and Aaron Barrera¹

¹ Centro de Investigación Científica y de Educación Superior de Ensenada,
Ensenada, BC, Mexico

`gustavo.olague@gmail.com, abarrera@cicese.edu.mx`

² Instituto Tecnológico de Ensenada, Ensenada, BC, Mexico
`eclemente@ite.edu.mx`

³ Instituto Tecnológico de Tijuana, Tijuana, BC, Mexico
`daniel.hernandezm@tectijuana.edu.mx`

Abstract. Computational neuroscience lays the foundations of intelligent behavior through the application of machine learning approaches. Brain programming, which derives from such approaches, is emerging as a new evolutionary computing paradigm for solving computer vision and pattern recognition problems. Primate brains have several distinctive features that are obtained by a complex arrangement of highly interconnected and numerous cortical visual areas. This paper describes a virtual system that mimics the complex structure of primate brains composed of an artificial dorsal pathway – or “where” stream – and an artificial ventral pathway – or “what” stream – that are fused to recreate an artificial visual cortex. The goal is to show that brain programming is able to discover numerous heterogeneous functions that are applied within a hierarchical structure of our virtual brain. Thus, the proposal applies two key ideas: first, object recognition can be achieved by a hierarchical structure in combination with the concept of function composition; second, the functions can be discovered through multiple random runs of the search process. This last point is important since is the first step in any evolutionary algorithm; in this way, enhancing the possibilities for solving hard optimization problems.

Keywords: Object recognition · Random search · Brain programming

1 Introduction

Object recognition is a fundamental task for humans and all living beings endowed with the sense of sight, since it allows the interaction of the organism with the surrounding environment and the understanding of a given object. In the last two decades computer vision and evolutionary algorithms have seen a growing interest from the scientific community [1]. In general, the human visual system is able to recognize and classify an object according to its category with ease.

Both tasks consider that the set of attributes or features extracted from the images are general enough to classify the object as part of the category, while maintaining in memory the features that serve to identify that particular object within a given scene [2–4]. The aim of this paper is to present a new model inspired by the transformations that take place within the visual cortex for the solution of the object classification task.

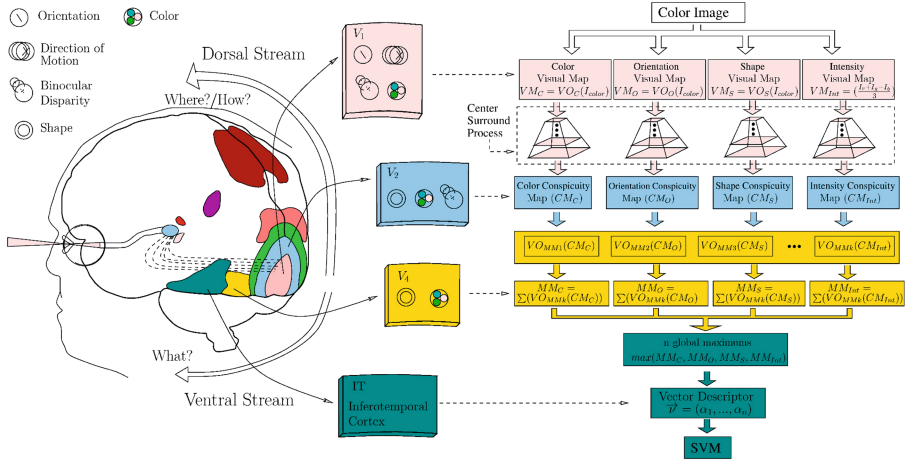


Fig. 1. Conceptual Model of the Artificial Visual Cortex. The color image is decomposed into four dimensions (color, orientation, shape and intensity). Then, a hierarchical structure is charged of solving the object classification problem through a function driven paradigm.

Ungerleider and Mishkin in 1982 proposed the existence of two routes in the visual cortex. These pathways have been called dorsal and ventral streams, the functionality of the dorsal stream provides the location of an object within the scene, while the ventral stream is dedicated to the task of object recognition. Thus, efficient visual functionality is achieved by a high interchange of information between the two streams [5–10]. In this way, object recognition involves processes performed along the dorsal stream such as selectivity that is defined as the ability to filter unwanted information, and those performed in the ventral stream in charge of describing the objects. Thus, the approach proposed in this work is suggested by a computational model based in these two information streams of the visual cortex. This approach differs from those of the state-of-the-art where a data-driven principle is applied using a set of patches – image regions – while creating a dictionary of visual words like in a bag-of-words approach [11–15]. In our work, the first hypothesis is that the dictionary of visual words can be replaced by a set of visual operators which are built with a group of mathematical functions. The second idea is based on the integration of properties responsible for the visual attention process – or selectivity – that is related to

creation of conspicuity maps (*CMs*) and the center surround process together with description and combination of maximum responses executed by a *max* operation of the functions that select features, which categorize the object; see Fig. 1. In this way, this model sees the brain as a collection of functions joined into a structural unit that serves to the purpose of object recognition, where the functionality of each area or layer in the visual cortex is represented by a kind of mathematical function, and the interconnection among them is given by the hierarchical structure of the model. Hence, each compound mathematical function mimics the functionality of its natural counterpart as a way of designing a set of virtual brain areas, called Artificial Visual Cortex (*AVC*). Therefore, the object categorization for the presence/absence problem is achieved through the application of the correct combination of functions within the *AVC*; an approach that we are calling brain programming [16–18].

In this manner, brain programming is defined by an evolutionary cycle. In this paradigm, the optimization problem is defined as the search of multiple parts embedded on a hierarchical system known as the *AVC* model, which plays a key role on the representation of the solutions that are more complex than a single syntax tree. In this kind of systems, we must encapsulate the key parts of the hierarchical structure in order to evolve them. Hence, by integrating the evolved operations within the complex structure we were able to synthesize solutions for difficult problems; in this case, the object recognition problem [31].

The applicability and efficiency of this methodology has been described in several works [16–21], nevertheless, an individual analysis of the *AVC* model and the brain programming methodology has not been done. As in many optimization paradigms, we propose to compare brain programming with a random search approach, in order to characterize the benefits of the *AVC* model by itself and the improvement brought by the evolutionary approach. This paper focus on the random search and we will explore the whole algorithm in a future article.

1.1 Research Contributions

This paper outlines the following research contributions:

- First, in the proposed approach the total number of visual operators made of mathematical functions and embedded within the hierarchical structure can be discovered through a small number of random trials, while actually achieving outstanding results on an standard testbed. This article provides evidence that the hierarchical structure plays a significant role on the solution of visual problems.
- Second, a comparison of the random search with state-of-the-art algorithms and the whole evolutionary cycle gives us a clear picture of the benefit of applying brain programming.

1.2 Related Work

Most of the works are divided in two basic approaches, the first is regarding visual attention conducted along the dorsal stream, while a second approach is related

on object recognition held in the ventral stream. Nevertheless, there are a few works that attempted to integrate the two approaches. For example, Fukushima in 1987 implemented a hierarchical neural network that serves as a model for selective attention and object recognition [22]. In this case, when several patterns are presented simultaneously, the model performs a selective attention to each one, segmenting it from the rest and recognizing it separately. Afterwards, Olshausen *et al.* in 1993 defined a biologically plausible model that combines an attentional mechanism with an object recognition process to form position and scale invariant representations in the visual world [23]. Then, Walther *et al.* suggested a combined model for spatial attention and object recognition [24]. In their work, visual attention follows the computational model proposed by Itti and Koch [15] and object recognition is achieved through the *HMAX* model of Riesenhuber and Poggio [25]. Their information stream follows the whole visual attention process and the final saliency map is fed into the S2 layer of the *HMAX* model to accomplish the task of object detection. This model was applied to the problem of recognizing artificial paperclips. Next, Walther and Koch in 2007 suggested, with a computational model, that features learned by the *HMAX* model used for the recognition of a particular object category may also serve for top-down attention tasks [26]. Finally, Heinke and Humphreys applied a model called SAIM for visual search involving simple lines and letters [27]. This model, in a first stage selects the object within the image and subsequently performs an object identification step using a template matching technique.

In our work, we propose a hierarchical model following the preattentive stage of visual attention described in [28] in order to locate the conspicuity regions within the image. Then, a description process is performed using the max operator in combination with a series of functions that emulate the functionality of the V4 area in the visual cortex. This approach differs from traditional models for object recognition [4, 12, 13, 25–27] where a set of patches – or visual words – are used to identify the object. In contrast, in our proposed approach the discovered functions provide the functionality of multiple patches; hence, helping in the creation of a straightforward process as will be shown in the experimental results.

2 The *AVC* Algorithm

In the natural system the interrelation between the layers of the visual cortex is not fully understood; nevertheless, the functionality at each stage has been described on previous works. Figure 1 depicts the proposed model based on these processes. The *AVC* is divided in two main parts. In the first stage the proposed system executes the acquisition and transformation of features. Then, in a second stage the *AVC* performs description and classification associated to the studied object.

2.1 Acquisition and Transformation of Features

The first step of our algorithm is represented by the image acquired with the camera, whose natural counterpart is the retina. Here, the system considers

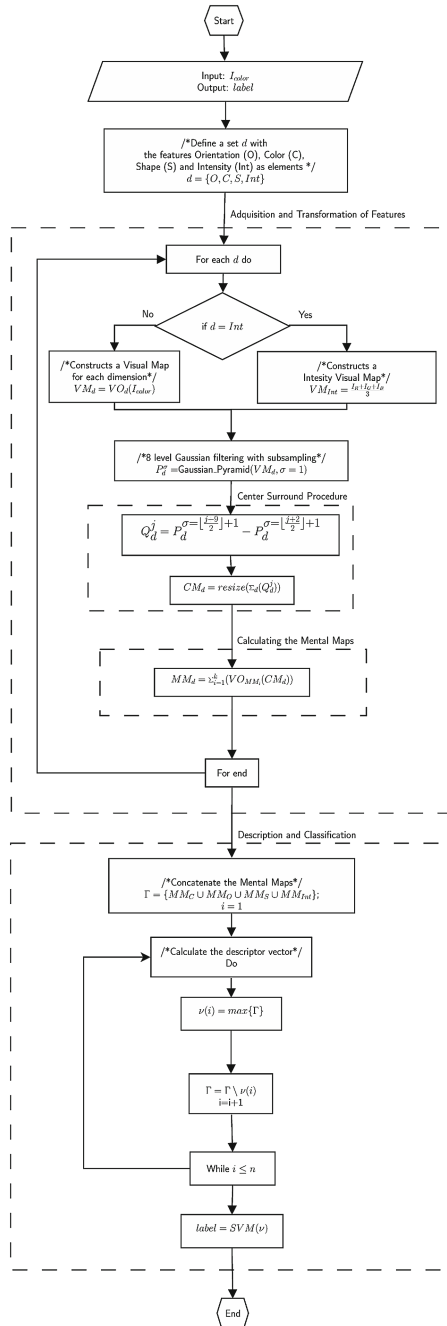


Fig. 2. Schematic representation of the computational algorithm whose output is a label that represents the membership to a specific class.

digital color images in the RGB color model, which are later transformed into the CMYK and HSV color models; see Fig. 2. In this way, the color image is decomposed into multiple color channels. The idea is to build the set $I_{color} = \{I_r, I_g, I_b, I_c, I_m, I_y, I_k, I_h, I_s, I_v\}$, which corresponds to the *red, green, blue, cyan, magenta, yellow, black, hue, saturation, and value* components of their respective color models and which are used to provide the initial representation of the scene. In our work, the input images in I_{color} are transformed by four visual operators (VOs) applied independently to emphasize specific image features. The transformations are performed to recreate the feature extraction process of the brain; resulting into a visual map (VM) per dimension [28].

2.2 Feature Dimensions

The VOs are defined with the aim of classifying specific image features along several dimensions: color, shape, orientation and intensity; hence, $d \in \{C, S, O, Int\}$. Figure 2 shows that features are extracted sequentially one at a time by applying the corresponding operator VO_d .

2.3 Center Surround Process

The center surround method is based on the functionality of the ganglion cells that measure the difference between the firing rates at the center and surrounding areas of their receptive fields. The goal of this process is to generate a conspicuity map (CM) per dimension according to the model proposed in [29]. The algorithm consists of a two step process where the information is built to emulate its natural counterpart as follows; see Fig. 2. First, the computation of the CMs is modeled as the difference between fine and coarse scales, which are computed through a pyramid of nine levels $P_d^\sigma = \{P_d^{\sigma=0}, P_d^{\sigma=1}, P_d^{\sigma=2}, P_d^{\sigma=3}, \dots, P_d^{\sigma=8}\}$. Each pyramid is calculated from its corresponding VM_d using a Gaussian smoothing filter resulting in an image that is half of the input map size and the process is repeated recursively eight times to complete the nine level pyramid. Second, the pyramid P_d^σ is used as input to a center surround procedure to derive six new maps that result from the difference between some of the pyramid levels that are calculated as follows.

$$Q_d^j = P_d^{\sigma=\lfloor \frac{j+9}{2} \rfloor + 1} - P_d^{\sigma=\lfloor \frac{j+2}{2} \rfloor + 1},$$

where $j = \{1, 2, \dots, 6\}$. Note that the levels of P_d^σ have different size and are scaled down to the size of the top level to calculate their difference. Next, each of these six maps are normalized and combined into a unique map through the summation operation, which is then normalized and scaled up to the VM_d maps' original size using a polynomial interpolation to define the final CM_d .

2.4 Description and Classification Stage

After the construction of the CMs , the next stage along the AVC is to define a descriptor vector that will be used as input to a support vector machine (SVM) model for classification purposes.

2.5 Computation of the Mental Maps

In this stage of the process a single set of visual operators is used to produce a mental map (MM_d) per dimension; see Fig. 2. After the computation of the conspicuity maps a set of visual operators VO_{MM} is applied with the aim of describing the image content. Note that the proposed visual operators are homogeneous and independently applied to each feature dimension. This operation is defined as follows:

$$MM_d = \sum_{i=1}^k (VO_{MM_i}(CM_d)), \quad (1)$$

where d is the dimension index and k represents the cardinality of the set VO_{MM} . Each summation is applied to integrate the output of all operations VO_{MM_k} to produce a MM_d per dimension. Thereafter, the four Mental Maps are concatenated into a single array and the n highest values are selected to define the vector \vec{v} that describes the image.

In contrast to our proposal, the state-of-the-art methodologies [11–15] are based on a template matching paradigm with the goal of learning a set of prototype image patches. Traditionally, the idea is to learn such a set by using what is known as the bag-of-words model, which is applied to identify a given object category. In this way, our approach substitutes the set of templates with the set of visual operators to characterize one object class with excellent results as we will show in the experiments.

3 Experiments and Results

We use the CalTech 5 and CalTech 101 image databases, despite many serious concerns raised about them [30, 31]. Nevertheless, that test is still widely used in the object recognition community and thus most state-of-the-art algorithms report their classification results [12, 13, 32–36].

3.1 Methodology to Obtain an AVC Solution

The methodology that was used to generate the *AVC* programs followed the algorithm of Sect. 2, where an important step is the construction of *VOs*. These operators consist of syntax trees made of internal and leaf nodes, which are defined by a set of primitive elements also called function set (see Table 1) and the terminal set defined by the domain of each function. In our work, each tree has its own sets of functions and terminals that were carefully chosen according to the desired functionality that we attempt to emulate within the *AVC*. All *VOs* were generated through a random procedure with a maximum depth of 5 levels, where half of the trees were balance trees and the other half were constructed as arbitrary trees adding nodes until the maximum depth is reached.

The proposed methodology for designing *AVCs* to study the absent/present classification problem is divided in three steps. The first two steps define the training stage while the last one is devoted to the testing stage. In this way, all image databases were randomly divided into three subsets for each class; one per step. This process is detailed next.

Table 1. Functions for the visual operators (*VOs*).

Function	Description
$A + B, A - B, A \times B, A/B$	Arithmetic functions between two images A and B
$\log(A), \exp(A)$	Transcendental functions over the image A
$(A)^2$	Square function over the image A
\sqrt{A}	Square root function over the image A
$(A)^c$	Image complement over the image A
$Op_{r-g}(I), Op_{b-y}(I)$	Color opponency Red - Green and Blue - Yellow
$thr(A)$	Dynamic threshold function over the image A
$k + A, k - A, k \times A, A/k$	Arithmetic functions between an image A and a constant k
$round(A), half, \lfloor A \rfloor, \lceil A \rceil$	Round, half, floor and ceil functions over the image A
$A \oplus SE_d, A \oplus SE_s, A \oplus SE_{dm}$	Dilation operator with disk, square, and diamond structure element (SE)
$A \ominus SE_d, A \ominus SE_s, A \ominus SE_{dm}$	Erosion operator with disk, square, and diamond structure element (SE)
$Sk(A)$	Skeleton operator over the image A
$Perim(A)$	Find perimeter of objects in the image A
$A \otimes SE_d, A \otimes SE_s, A \otimes SE_{dm}$	Hit or miss transformation with disk, square, and diamond structures
$T_{hat}(A), B_{hat}(A)$	Performs morphological top-hat and bottom-hat filtering over the image A
$A \odot SE_s, A \oslash SE_s$	Opening and closing morphological operators on A
$ A , A + B , A - B $	Absolute value applied to A , and the addition and subtraction operators
$\inf(A, B), \sup(A, B)$	Infimum and supremum functions between the images A and B
$G_{\sigma=1}(A), G_{\sigma=2}(A)$	Convolution of the image A and a Gaussian filter with $\sigma = 1$ or 2
$D_x(A), D_y(A)$	Derivative of the image A along direction x and y

1. The first step starts by randomly generating a set of *VOs* to be used inside the *AVC* structure. Then, it proceeds to the training stage of the *SVM* using the images from the first subset, called training-A. As a constraint, if the *SVM* achieves a given threshold in classification accuracy during its training, the process continues to step 2; on the other hand, the *VOs* together with the *SVM* are discarded and the process is restarted.
2. At step 2 the system uses the set of *VOs* found in step 1 but it trains a new *SVM* with the second image subset, called training-B. Once again, if the *SVM* scores the given threshold in classification accuracy the process continues to step 3 and the *AVC* structure is considered as the solution; on the other hand, both *VOs* and *SVM* are discarded and the search continues at step 1.

- In the last step, the best *AVC* structures are tested by classifying the third image subset. The testing is performed with the estimated *SVM* from step 2 and the *VOs* from step 1. The whole process is repeated until the best set of solutions are discovered.

Finally, all experimental results are provided in the following sections.

Table 2. Total number of random runs needed to discover 100 solutions per class for all subset sizes.

Class	Size of the training set						
	10	20	30	40	50	60	70
Airplanes	2501	3219	1916	2088	1993	1971	3253
Cars	7294	11032	6652	10674	4447	4845	22037
Faces	1811	3041	1489	2148	1556	1462	1940
Leaves	1781	1960	1392	1843	1893	1355	1763
Motorcycles	10419	23131	12871	20214	9386	6662	39470

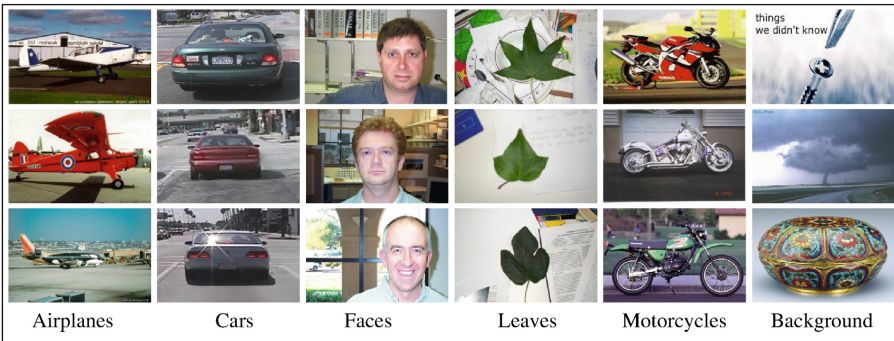


Fig. 3. Sample images from CalTech-5 database, and the category background from CalTech-101 database.

3.2 Experimental Evaluation of the *AVC* for Classification of Color Images

In a first experiment, the performance of the proposed model was evaluated through a binary test using five classes from the Caltech-5 database in combination with the Google background of Caltech-101, see Fig. 3. The goal is to analyze the effect on the recognition performance by using training sets of different sizes. Thus, the *AVC* model was trained with randomly selected positive images used to define the training-A subset of size: 1, 10, 20, 30, 40, 50, 60, and 70; while using a constant subset of 50 negative images for all experiments. In the

case of one positive training image and after 7500 randomly possible evaluations an *AVC* was never found; hence, it was discarded from further tests. Thus, the numbers of images selected for training-B were set to 50 positive images and 50 negative images. In this way, Table 2 provides the number of random runs that were necessary to discover the solutions. This experiment was repeated until 100 *AVCs* were found for each training-B subset producing a total of 700 solutions with 100% accuracy during the training stage per class. All these solutions were tested and the mean and standard deviation are reported in the following section.

3.3 Testing the Performance of the Random Search

Table 3 presents the summary of the experiment showing the average, standard deviation, maximum and minimum performance for the testing stage. All results were normalized between 0 and 1 in such a way that 1 represents 100% of classification accuracy. The best solutions were obtained for the airplanes, faces and leaves classes scoring 95%, 99% and 97% respectively; while in the case of cars and motorcycles classes the best solutions scored a classification accuracy of 77% and 75% respectively. Note that these final scores are similar regardless of the subset size that is applied during the training stage. Moreover, the solutions whose scores are highlighted in bold at Table 3 are provided with their corresponding formulae in Table 4.

Table 3. This table shows a summary of the results of the *AVC* testing which were obtained with a random search and using the color category background from CalTech-101 database.

Images on training	Airplane class				Cars class				Faces class			
	Mean	Std	Max	Min	Mean	Std	Max	Min	Mean	Std	Max	Min
10	0.64	0.08	0.87	0.50	0.57	0.06	0.76	0.46	0.66	0.15	0.98	0.48
20	0.61	0.08	0.83	0.50	0.55	0.05	0.77	0.45	0.67	0.14	0.99	0.45
30	0.62	0.09	0.88	0.45	0.56	0.05	0.71	0.47	0.65	0.13	0.98	0.50
40	0.61	0.09	0.95	0.50	0.57	0.06	0.75	0.45	0.64	0.12	0.98	0.45
50	0.61	0.08	0.88	0.49	0.55	0.05	0.74	0.46	0.64	0.14	0.98	0.50
60	0.57	0.07	0.83	0.49	0.54	0.05	0.74	0.46	0.58	0.10	0.99	0.46
70	0.61	0.08	0.92	0.50	0.55	0.04	0.73	0.47	0.62	0.12	0.98	0.50
	Leaves class				Motorcycle class							
	Mean	Std	Max	Min	Mean	Std	Max	Min				
10	0.68	0.14	0.97	0.48	0.58	0.05	0.75	0.46				
20	0.67	0.12	0.95	0.48	0.58	0.06	0.75	0.46				
30	0.64	0.12	0.95	0.50	0.56	0.05	0.70	0.44				
40	0.65	0.13	0.97	0.49	0.55	0.05	0.71	0.47				
50	0.59	0.09	0.93	0.48	0.54	0.04	0.74	0.47				
60	0.62	0.11	0.97	0.48	0.54	0.04	0.70	0.46				
70	0.61	0.09	0.94	0.50	0.57	0.06	0.74	0.47				

Table 4. This table shows the best solutions that were discovered after a random process.

Name	VO	VO_{MMk}	Evaluation
AVC_{A1}	$VO_O = \text{round}(D_{xy}(I_k))$ $VO_C = I_r$ $VO_S = \lfloor I_r \rfloor$	$VO_{MM1} = D_{yy}(CM_d)$ $VO_{MM2} = G_{\sigma=2}(D_{xx}(CM_d))$ $VO_{MM3} = \sqrt{D_{xy}(CM_d)}$ $VO_{MM4} = D_{xx}(CM_d)$ $VO_{MM5} = D_y(CM_d)$ $VO_{MM6} = D_{xy}(CM) - D_x(CM_d) $ $VO_{MM7} = G_{\sigma=2}(D_{yy}(CM_d))$ $VO_{MM8} = D_{xx}(CM_d)$ $VO_{MM9} = \log(D_{xx}(CM_d))$	$Tr = 100\%$ $Tst = 95\%$
AVC_{C1}	$VO_O = D_y(I_v)$ $VO_C = \frac{I_b}{I_r}$ $VO_S = \lfloor I_c \rfloor$	$VO_{MM1} = D_x(CM_d)$ $VO_{MM2} = \log(CM_d)$ $VO_{MM3} = D_{xx}(CM_d)$ $VO_{MM4} = D_{xy}(CM_d)$ $VO_{MM5} = D_{xy}(CM_d)$ $VO_{MM6} = D_{yy}(CM_d)$ $VO_{MM7} = G_{\sigma=1}(D_y(CM_d))$ $VO_{MM8} = D_y(CM_d)$ $VO_{MM9} = CM_d$ $VO_{MM10} = D_{xy}(CM_d)$ $VO_{MM11} = \log(D_y(CM_d))$ $VO_{MM12} = G_{\sigma=2}(D_{yy}(CM_d))$	$Tr = 100\%$ $Tst = 77\%$
AVC_{F1}	$VO_O = \text{round}(D_{xx}(I_m))$ $VO_C = \sqrt{I_k}$ $VO_S = 0.45 * (I_b)$	$VO_{MM1} = (D_{xx}(CM_d))^2$	$Tr = 100\%$ $Tst = 99\%$
AVC_{L1}	$VO_O = \sqrt{D_x(I_b)}$ $VO_C = I_b * I_g$ $VO_S = \lfloor I_h \rfloor$	$VO_{MM1} = 0.5 * (D_y(CM_d))$ $VO_{MM2} = D_{yy}(CM_d) - D_y(CM_d)$	$Tr = 100\%$ $Tst = 97\%$
AVC_{M1}	$VO_O = G_{\sigma=1}(D_x(I_y)^2)$ $VO_C = \frac{I_b^c}{I_g - I_r}$ $VO_S = (I_v - 0.21) \oplus SE_{dm}$	$VO_{MM1} = \frac{D_{xx}(CM_d)}{D_{xx}(CM_d)} - \log(D_x(CM_d))$	$Tr = 100\%$ $Tst = 75\%$

3.4 Comparison Between the AVC and HMAX Models

The *HMAX* model was used in a second series of tests based on the experimental design proposed in [12], in order to compare our results with the state-of-the-art. Thus, once again, the solutions from the first experiment were tested, in the object present/absent experiment, with a new random set of images considering 50 positive images for the object classes selected earlier, as well as 50 negative images selected from the Caltech-5 background database. The aim is to investigate the effect on the 700 final solutions per class using the recognition performance based on the accuracy. Note that for this test the background images are in gray scale; hence, the color components of the image were initialized

Table 5. This table summarizes the classification results achieved on testing using the background Caltech-5 database as the negative class. Note that the performance is better than the previous experiment, since the background is built by gray tone images.

Images on training	Airplane class				Cars class				Faces class			
	Mean	Std	Max	Min	Mean	Std	Max	Min	Mean	Std	Max	Min
10	0.64	0.10	0.95	0.50	0.59	0.07	0.78	0.47	0.66	0.15	0.97	0.43
20	0.63	0.09	0.90	0.49	0.61	0.12	0.97	0.45	0.67	0.14	0.98	0.47
30	0.63	0.10	0.86	0.48	0.60	0.09	0.97	0.50	0.66	0.13	0.99	0.47
40	0.62	0.09	0.97	0.50	0.59	0.09	0.92	0.45	0.64	0.13	0.97	0.46
50	0.62	0.10	0.96	0.49	0.61	0.11	0.96	0.47	0.65	0.14	1.00	0.48
60	0.59	0.10	0.95	0.50	0.57	0.09	0.92	0.46	0.60	0.13	0.97	0.42
70	0.64	0.11	0.96	0.50	0.60	0.10	0.98	0.46	0.63	0.12	0.99	0.47
	Leaves class				Motorcycle class							
	Mean	Std	Max	Min	Mean	Std	Max	Min				
10	0.70	0.14	0.98	0.50	0.62	0.12	0.95	0.43				
20	0.67	0.13	0.97	0.49	0.61	0.12	0.98	0.48				
30	0.65	0.12	0.91	0.47	0.58	0.08	0.97	0.46				
40	0.66	0.13	0.95	0.46	0.59	0.10	0.96	0.47				
50	0.59	0.11	0.97	0.45	0.59	0.11	0.96	0.45				
60	0.63	0.13	0.96	0.46	0.55	0.07	0.90	0.45				
70	0.61	0.11	0.92	0.47	0.61	0.12	1.00	0.41				

with the same value. The results summary is shown in Table 5. The comparison between our model and the *HMAX* model is provided in Table 6. We report the error rate at equilibrium point as the measure performance in these experiments. For the sake of showing that the differences between the performances of the proposed *AVC* and the *HMAX-SVM* models are statistically significant, we used two non-parametric statistical tests: the Wilcoxon rank sum [37] and a two-sample Kolmogorov-Smirnov test [38]. These last experiments were tested on the 30 best random solutions out of the 700 found for each class.

Table 6. This table shows a comparison of the performance achieved among the *HMAX* model, considering the boost and *SVM* classifiers, and the *AVC* model. Note that in the case of the *HMAX* model a learning process was applied in order to identify the best patches. However, for the *AVC* model only a random sampling was used to discover the best solution.

Datasets	Performance of <i>HMAX</i>		Artificial V. C	Statistical Significance	
	boost	<i>SVM</i>		K-S test	Wilcoxon test
Airplanes	96.7	94.9	98.6	4.9×10^{-15}	1.1×10^{-7}
Cars	99.7	99.8	98.1	2.1×10^{-13}	3.9×10^{-8}
Faces	98.2	98.1	100	1.8×10^{-15}	1.3×10^{-10}
Leaves	97.0	95.9	96.2	7.9×10^{-17}	6.3×10^{-12}
Motorcycles	98.0	97.4	100	5.4×10^{-13}	9.6×10^{-6}

Using the whole evolutionary process, we present in Table 7 the solutions that were discovered by the brain programming. The selection process was implemented following the *roulette-wheel* strategy, which consists in assigning to each individual a probability of selection proportional to its fitness value. Termination criteria was defined using a maximum number of generations; 30 in this case and 30 solutions per generation. Thus, the aim is that each evolutionary process reaches an optimal *AVC* program at each single run. Note that the performance of each solution in testing is 100% in classification accuracy.

Table 7. This table shows some solutions that were discovered after of evolutionary process of the brain programming.

Solution	<i>EVO</i>	<i>EVO_{MMk}</i>	Accuracy
Airplanes	$EVO_O = D_{xy}(I_r)$ $EVO_C = Op_{b-y}(I)$ $EVO_S = \frac{I_r \oplus SE_{dm}}{K}$	$EVO_{MM_1} = 0.5 * (D_x(MC))$	<i>Tr.</i> = 100% <i>Val.</i> = 100% <i>Tst.</i> = 100%
Cars	$EVO_O = \sqrt{0.33 * D_{yy}(I_y)}$ $EVO_C = ((Op_{r-g}(I))^c)^2$ $EVO_S = Sk(I_v) - 0.33 * I_m$	$EVO_{MM_1} = 0.5 * (D_x(MC)) $	<i>Tr.</i> = 100% <i>Val.</i> = 100% <i>Tst.</i> = 100%
Faces	$EVO_O = G_{\sigma=2}(0.5 * D_{yy}(I_r))$ $EVO_C = \frac{Op_{r-g}(I)}{I_k * \sqrt{I_m}}$ $EVO_S = T_{hat}(I_h \ominus SE_s)$	$EVO_{MM_1} = \log(D_{xx}(MC))$	<i>Tr.</i> = 100% <i>Val.</i> = 100% <i>Tst.</i> = 100%
Leaves	$EVO_O = D_x(\log(I_h - I_s))$ $EVO_C = (I_k)^2$ $EVO_S = 0.13 * ((I_h \oplus SE_s) \oplus SE_s)$	$EVO_{MM_1} = 0.5 * \sqrt{MC} - (D_{yy}(MC) - D_x(MC)) $ $EVO_{MM_2} = \frac{ D_y(MC) - D_y(MC) }{D_y(MC) - MC}$	<i>Tr.</i> = 100% <i>Val.</i> = 100% <i>Tst.</i> = 100%
Motorcycles	$EVO_O = \sup(\sqrt{D_x(I_r)}, thr(D_x(I_r)))$ $EVO_C = \sqrt{\sqrt{(I_r)^2}}$ $EVO_S = I_g - I_m$	$EVO_{MM_1} = \log(D_{xy}(MC) + D_x(MC))$ $EVO_{MM_2} = 0.5 * (G_{\sigma=1}(D_{xy}(MC)))$	<i>Tr.</i> = 99.5% <i>Val.</i> = 99.66% <i>Tst.</i> = 100%

4 Conclusions and Future Work

This paper presented a computational model of the visual cortex following the hierarchical structure of previous visual attention and object recognition proposals. As a result, the proposed methodology replicates the initial stages of the artificial dorsal stream using four dimensions: color, shape, orientation and intensity, in combination with the final stages of an artificial ventral stream, which are used to approach the task of object categorization. The overall approach considers that the process of visual information extraction and description can be enforced by function composition through a set of mathematical operations that are used in the aforementioned stages. According to the results all functions embedded within the hierarchical structure of the *AVC* can be easily discovered through random search while achieving excellent results on the Caltech database. In this sense, we presented examples that illustrate the behavior of the

discovered *AVCs* for the problems of faces and motorcycles of the Caltech categorization problem. Finally, we provide a comparison with the *HMAX* model that is considered as the reference for these kind of approaches and found that the *AVC* model was superior according to the results obtained for the Caltech testbed.

As a conclusion, we can say that the *AVC* methodology offers a new perspective to study the development of artificial brains since the structural complexity can be improved because the approach is susceptible of being framed as an optimization problem. This article provides some results of the whole evolutionary cycle for the studied database. Indeed, the results score perfect accuracy except for one case. In this way, we can synthesize new structures according to the task at hand. In particular, as future research we would like to test the approach with more complex datasets, such as the GRAZ and the VOC challenge datasets. Additionally, since the methodology is computationally costly, we propose to undergo a change towards a parallel computing implementation of the *AVC* model through the application of GPGPU technology [39]. Finally, we would like to explore the application of this new paradigm to problems of humanoid robotics.

Acknowledgments. This research was funded by CONACyT through the Project 155045 - “Programación cerebral aplicada al estudio del pensamiento y la visión”. This work is also supported by ITE-TecNM through the project 5748.16-P, “Optimización de controladores aplicados a la navegación de un robot móvil, utilizando cómputo evolutivo”.

References

1. Olague, G.: *Evolutionary Computer Vision: The First Footprints*. Springer, Heidelberg (2016)
2. Logothetis, N.K., Sheinberg, D.L.: Visual object recognition. *Ann. Rev. Neurosci.* **19**, 577–621 (1996)
3. DiCarlo, J.J., Zoccolan, D., Rust, N.C.: How does the brain solve visual object recognition? *Neuron* **73**(3), 415–434 (2012)
4. Riesenhuber, M., Poggio, T.: Models of object recognition. *Nat. Neurosci.* **3**, 1199–1204 (2000)
5. Rees, G., Frackowiak, R., Frith, C.: Two modulatory effects of attention that mediate object categorization in human cortex. *Science*. **275**(5301), 835–8 (1997)
6. Desimone, R., Duncan, J.: Neural mechanisms of selective visual attention. *Ann. Rev. Neurosci.* **18**, 193–222 (1995)
7. Kastner, S., Ungerleider, L.G.: Mechanisms of visual attention in the human cortex. *Ann. Rev. Neurosci.* **23**, 315–341 (2000)
8. Milner, A.D., Goodale, M.A.: *The Visual Brain in Action*, 2nd edn. Oxford University Press, Oxford (2006)
9. Creem, S.H., Proffitt, D.R.: Defining the cortical visual systems: “what”, “where”, and “how”. *Acta Psychol.* **107**(1–3), 43–68 (2001)
10. Farivar, R.: Dorsal-ventral integration in object recognition. *Brain Res. Rev.* **61**(2), 144–153 (2009)

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. Serre, T., Kouh, C., Cadieu, M., Knoblich, G., Kreiman, U., Poggio, T.: Theory of object recognition: computations and circuits in the feedforward path of the ventral stream in primate visual cortex. Technical report, Massachusetts Institute of Technology Computer Science and Artificial Intelligence Laboratory (2005)
13. Mutch, J., Lowe, D.G.: Object class recognition and localization using sparse features with limited receptive fields. *Int. J. Comput. Vis.* **80**(1), 45–57 (2008)
14. Mel, B.W.: Seemore: combining color, shape, and texture histogramming in a neurally inspired approach to visual object recognition. *Neural Comput.* **9**(4), 777–804 (1997)
15. Itti, L., Koch, C.: Computational modeling of visual attention. *Nat. Rev. Neurosci.* **2**(3), 194–203 (2001)
16. Clemente, E., Olague, G., Dozal, L., Mancilla, M.: Object recognition with an optimized ventral stream model using genetic programming. In: Chio, C., et al. (eds.) *EvoApplications 2012*. LNCS, vol. 7248, pp. 315–325. Springer, Heidelberg (2012). doi:[10.1007/978-3-642-29178-4_32](https://doi.org/10.1007/978-3-642-29178-4_32)
17. Clemente, E., Olague, G., Dozal, L.: Purposive evolution for object recognition using an artificial visual cortex. In: Schuetze, O., Coello, C.A.C., Tantar, A.-A., Tantar, E., Bouvry, P., Del Moral, P., Legrand, P. (eds.) *EVOLVE - A Bridge between Probability, Set Oriented Numerics, and Evolutionary Computation II*, pp. 355–370. Springer, Heidelberg (2013)
18. Olague, G., Clemente, E., Dozal, L., Hernández, D.E.: Evolving an artificial visual cortex for object recognition with brain programming. In: Schuetze, O., Coello, C.A.C., Tantar, A.-A., Tantar, E., Bouvry, P., Del Moral, P., Legrand, P., et al. (eds.) *EVOLVE - A Bridge between Probability, Set Oriented Numerics, and Evolutionary Computation III*. Springer, Heidelberg (2014)
19. Dozal, L., Olague, G., Clemente, E., Hernandez, D.E.: Brain programming for the evolution of an artificial dorsal stream. *Cognit. Comput.* **6**(3), 528–557 (2014)
20. Hernandez, D.E., Clemente, E., Olague, G., Briseño, J.L.: Evolutionary multi-objective visual cortex for object classification in natural images. *J. Comput. Sci.* **17**(1), 216–233 (2016)
21. Clemente, E., Chavez, F., Fernandez de Vega, F., Olague, G.: Self-adjusting focus of attention in combination with a genetic fuzzy system for improving a laser environment control device system. *Appl. Soft Comput.* **32**, 250–265 (2015)
22. Fukushima, K.: Neural network model for selective attention in visual pattern recognition and associative recall. *Appl. Opt.* **26**(23), 4985–4992 (1987)
23. Olshausen, B.A., Anderson, C.H., Van Essen, D.C.: A neurobiological model of visual attention and invariant pattern recognition based on dynamic routing of information. *J. Neurosci.* **13**(11), 4700–4719 (1993)
24. Walther, D., Itti, L., Riesenhuber, M., Poggio, T., Koch, C.: Attentional selection for object recognition — a gentle way. In: Bülthoff, H.H., Wallraven, C., Lee, S.-W., Poggio, T.A. (eds.) *BMCV 2002*. LNCS, vol. 2525, pp. 472–479. Springer, Heidelberg (2002). doi:[10.1007/3-540-36181-2_47](https://doi.org/10.1007/3-540-36181-2_47)
25. Riesenhuber, M., Poggio, T.: Hierarchical models of object recognition in cortex. *Nat. Neurosci.* **2**(11), 1019–1025 (1999)
26. Walther, D., Koch, C.: Attention in hierarchical models of object recognition. *Progr. Brain Res.* **165**, 57–78 (2007)

27. Heinke, D., Humphtey, G.W.: Attention, spatial representation, and visual neglect: simulating emergent attention and spatial memory in the selective attention for identification model (SAIM). *Psychol. Rev.* **110**(1), 29–87 (2003)
28. Treisman, A.M., Gelade, G.: A feature-integration theory of attention. *Cogn. Psychol.* **12**(1), 97–136 (1980)
29. Walther, D., Koch, C.: Modeling attention to salient proto-objects. *Neural Netw.* **19**(9), 1395–407 (2006)
30. Pinto, N., Cox, D.D., DiCarlo, J.J.: Why is real-world visual object recognition hard? *PLoS Comput. Biol.* **4**(1), 151–156 (2008)
31. Ponce, J., et al.: Dataset issues in object recognition. In: Ponce, J., Hebert, M., Schmid, C., Zisserman, A. (eds.) *Toward Category-Level Object Recognition*. LNCS, vol. 4170, pp. 29–48. Springer, Heidelberg (2006). doi:[10.1007/11957959_2](https://doi.org/10.1007/11957959_2)
32. Wang, Z., Feng, J.: Multi-class learning from class proportions. *Neurocomputing* **119**, 273–280 (2013)
33. Ji, Z., Wang, J., Su, Y., Song, Z., Xing, S.: Balance between object and background: object-enhanced features for scene image classification. *Neurocomputing* **120**, 15–23 (2013)
34. Chen, B., Polatkan, G., Sapiro, G., Blei, D., Dunson, D., Carin, L.: Deep learning with hierarchical convolutional factor analysis. *IEEE Trans. Pattern Anal. Mach. Intell.* **8**(35), 1887–1901 (2013)
35. Xu, B., Hu, R., Guo, P.: Combining affinity propagation with supervised dictionary learning for image classification. *Neural Comput. Appl.* **22**(7–8), 1301–1308 (2013)
36. Chandra, S., Kumar, S., Jawahar, C.V.: Learning hierarchical bag of words using naive bayes clustering. In: Lee, K.M., Matsushita, Y., Rehg, J.M., Hu, Z. (eds.) *ACCV 2012*. LNCS, vol. 7724, pp. 382–395. Springer, Heidelberg (2013). doi:[10.1007/978-3-642-37331-2_29](https://doi.org/10.1007/978-3-642-37331-2_29)
37. Wilcoxon, F.: Individual comparison by ranking methods. *Biometr. Bull.* **1**(6), 80–83 (1945)
38. Massey, F.J.: The Kolmogorov-Smirnov test for goodness of fit. *J. Am. Stat. Assoc.* **46**(253), 68–78 (1951)
39. Hernandez, D.E., Olague, G., Hernandez, B., Clemente, E.: CUDA-based parallelization of a bio-inspired model for fast object classification. *Neural Comput. Appl.* (2017). doi:[10.1007/s00521-017-2873-3](https://doi.org/10.1007/s00521-017-2873-3)