ISSN 1870-4069

Improving Leukemia Image Classification by Extracting and Transferring Knowledge by Evolutionary Vision

Rocio Ochoa-Montiel[1,2], Gustavo Olague[3], Humberto Sossa[1,4]

[1] Instituto Politécnico Nacional,
Centro de Investigación en Computación,
Mexico

[2] Universidad Autónoma de Tlaxcala,
Facultad de Ciencias Básicas,
Mexico

[3] Centro de Investigación Científica y de Educación Superior de Ensenada,
EvoVision Laboratory Ensenada,
Mexico

[4] Tecnológico de Monterrey,
Escuela de Ingeniería y Ciencias,
Mexico

{ma.rocio.ochoa,gustavo.olague,
humbertososa}@gmail.com

Abstract. The aim of evolutionary vision is to address typical problems of artificial vision through techniques whose principles are based on the theory of biological evolution. This allows us to consider the visual problem as a goal-oriented vision problem. Although the current automatic approaches are widely used in diverse recognition tasks, these are unable to explain how the knowledge to solve the problem is derived. In this paper, we use an evolutionary vision technique called brain programming (BP) to extract the knowledge used for the classification task. This model allows us to know how the knowledge to solve the leukemia images classification problem is derived. In addition, we present two variants focused on transferring knowledge to improve the classification. Results show that classification on leukemia images is achieved successfully from the solutions obtained by the proposed variants.

Keywords: Evolutionary vision, leukemia type classification, knowledge transfer.

1 Introduction

Leukemia is a cancer of the early blood-forming cells that is among the top 10 types of cancer more frequent in the world. Its diagnosis requires of information derived from several modalities, including morphology, cell phenotyping, cytochemistry, cytogenetics, and molecular genetics. Particularly, the morphological analysis requires a manual microscopic examination that is time-consuming and is prone to human error [4, 10].

pp. 167–176; rec. 2021-07-31; acc. 2021-09-15

To overcome the limitations of the manual analysis, several computational methods have been proposed following both, handcraft and automatic approaches. A handcraft approach utilizes conventional image processing and machine learning techniques, in which properties as the shape, color, and the distribution of some elements in the image are meaningful details for the recognition task [1, 2]. The knowledge derived from this approach can be used for the development of explainable learning models.

On the other hand, although the automatic approaches such as the convolutional neural networks have shown satisfactory results in leukemia images recognition, these are not driven by human reasoning, and they are computationally expensive, which is a clear drawback [7]. In this regard, the evolutionary computing techniques as the genetic programming offer an important advantage because use a trees-based representation that can automatically perform feature extraction, feature derivation, feature selection, and classification. Nevertheless, the classification task is a challenging problem due to issues such as low classification accuracy and a long training time.

For this reason, the reuse of learned knowledge is helpful to address these limitations as is suggested in [3, 6]. In this paper, we use an evolutionary insight paradigm as a baseline method to address the leukemia classification problem and show that knowledge extraction and transfer are useful for improving classification accuracy even when the reused knowledge comes from classes other than those under study. In the next section, we recall the theoretical concepts. Section 3 presents the proposed methodology. In Section 4, experiments and results are shown. Conclusions are drawn in Section 5.

2 Materials and Methods

In this work, the recognition problem is addressed using a model of evolutionary computer vision named brain programming (BP) as the baseline method. The BP uses the genetic programming (GP) to discover a set of evolutionary visual operators (EVOs) that are included within a structure named artificial visual cortex (AVC) [9].

We introduce the image recognition problem from the standpoint of data modelling. A minimization problem requires finding a solution $\mathbf{L}_{min} \in S$ such that $f(\mathbf{L}_{min})$ is a global minimum on S, $\exists \mathbf{L}_{min} \in S : f(\mathbf{L}_{min}) \leq f(\mathbf{L})$. In GP for a recognition problem, the purpose is to find a function that satisfies the task of data modelling. Thus, image recognition is defined as:

$$\mathbf{y} = \min(f(\mathbf{x}, \mathbf{F}, \mathbf{T}, \mathbf{a})), \tag{1}$$

where the dataset is given by $(\mathbf{x}, \mathbf{y})$, $\mathbf{F}$ denotes the set of functions, $\mathbf{T}$ represents the terminal set, and $\mathbf{a}$ describes the parameters used for tuning the algorithm. Thus, we require a feature extraction method and a suitable criterion $\mathbf{C}$ for minimization. The methodology requires the definition of two parts: 1) the AVC is the algorithm in charge of feature extraction, and 2) BP is the algorithm used to tune $(\mathbf{F}, \mathbf{T}, \mathbf{a})$ for each visual operator into the AVC. We use a multilayer perceptron (MLP) as the classifier to learn a mapping $f(\mathbf{x})$ where the descriptors $\mathbf{x}_i$ are associated with labels $\mathbf{y}_i$. In this work, we address the given problem as a multiclass classification task.

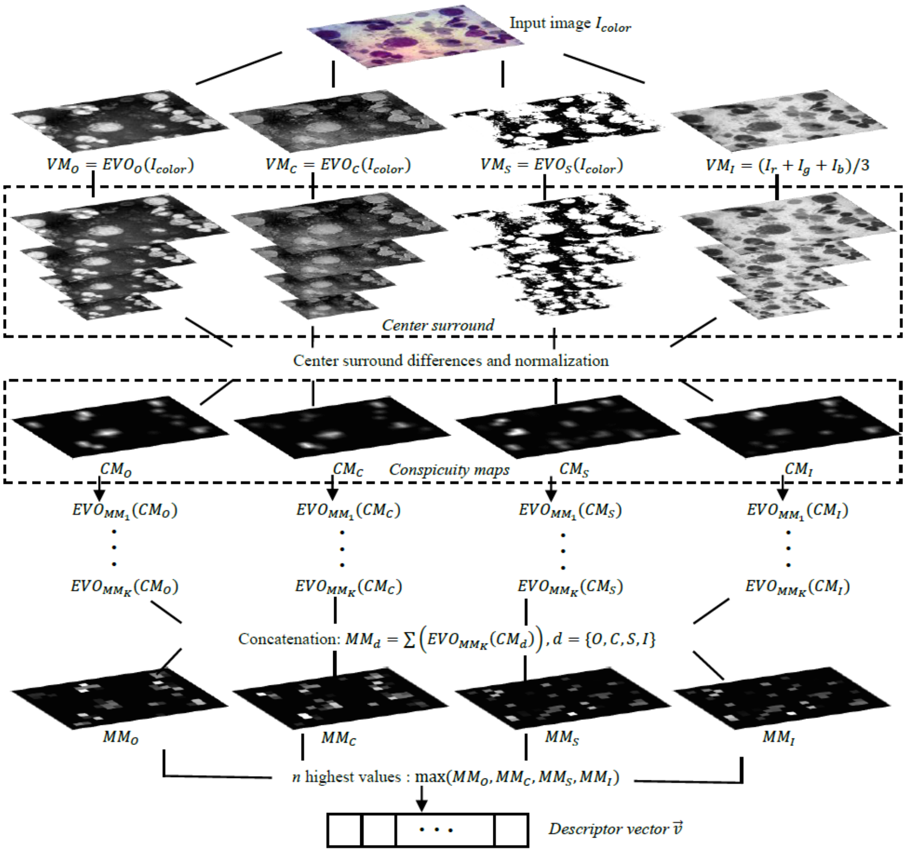

Input image I_{color}
$VM_O = EVO_O(I_{color})$
$VM_C = EVO_C(I_{color})$
$VM_S = EVO_S(I_{color})$
$VM_I = (I_r + I_g + I_b)/3$
Center surround
Center surround differences and normalization
CM_O
CM_C
Conspicuity maps
CM_S
CM_I
$EVO_{MM_1}(CM_O)$
$EVO_{MM_1}(CM_C)$
$EVO_{MM_1}(CM_S)$
$EVO_{MM_1}(CM_I)$
$EVO_{MM_K}(CM_O)$
$EVO_{MM_K}(CM_C)$
$EVO_{MM_K}(CM_S)$
$EVO_{MM_K}(CM_I)$
Concatenation: $MM_d = \sum\left(EVO_{MM_K}(CM_d)\right), d = \{O, C, S, I\}$
MM_O
MM_C
MM_S
MM_I
n highest values : max(MM_O, MM_C, MM_S, MM_I)
. . .
Descriptor vector $\vec{v}$

Fig. 1. Flowchart of the first stage of AVC.

Hence, it is assumed that in the minimization problem, the variables $((\mathbf{x}, \mathbf{y}), \mathbf{F}, \mathbf{T}, \mathbf{a}, \mathbf{C})$ are related in such a way that the objective is to associate the descriptors and the labels. Since the function is the base for understanding the transformations of the scene, then an image I is considered as the graph of a function [9].

2.1 Artificial Visual Cortex

The AVC is a hierarchical model that reproduces some aspects of the human visual cortex. First, a set of visual functions is defined according to the problem, and then each layer of the artificial visual cortex computes the mathematical operations pertaining to these functions. A representation of the object of interest is created from the image's visual features. This representation consists of salient points in the image to generate a descriptor used for classification. Figure 1 shows the first stage performed by the AVC. The AVC is composed of two stages. In the first stage, the features that describe the object are acquired and transformed, whereas in the second phase, the descriptor obtained in the previous stage is used to classify the object.

The input of the AVC is an RGB image I, and multiple colour channels are considered to create the set $I_{color} = \{I_r, I_g, I_b, I_c, I_m, I_y, I_k, I_h, I_s, I_v\}$, where the elements refer to the colour components of the RGB, HSV, and CMYK colour spaces. Later, each EVO is applied separately to generate a visual map (VM) to emphasize features based on orientation, color, and shape.

The EVOs are specialized functions evolved from a set of image operators such as color opponency or Gabor filters that, according to the knowledge from neuroscience-based on the way these functions are obtained in the biological visual cortex of the brain. In this way, the evolutionary process uses a set of functions and terminals to generate the operators for each dimension.

The dimensions d are the feature types used by the AVC, in this work we use the dimensions of orientation, color, shape, and intensity. This last, is the only one that performs a constant operator due to it only transforms the input color image into an intensity image. Thus, $VM_I = (I_r + I_g + I_b)/3$. Where I_r, I_g, and I_b are the color components of the RGB color model. The next step is to compute a centre surrounding process. First, the scale-invariant features are extracted and stored in a conspicuity map (CM).

The CM is calculated as the difference between the different scales that are obtained through a pyramid of 9 levels: $P_d^\sigma = \{P_d^{\sigma=0}, P_d^{\sigma=1}, P_d^{\sigma=2}, ..., P_d^{\sigma=8}\}$. A Gaussian smoothing filter is used on each VM to calculate each pyramid. This produces an image that is half the size of the input map. The process is repeated 8 times to obtain the 9-level pyramid. In the next step, the differences with respect to each pyramid level in P_d^σ are calculated using (2) as follows:

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

where $j = 1, 2..6$. Each level of P_d^σ is normalized and scaled to the dimensionality of the VM using polynomial interpolation. Finally, the six levels are combined into a single map with a summation operation. As a result, a CM is obtained for each dimension. The second stage of the AVC is description and classification. Here, a map is built to highlighting the most salient features from the CMs. For this, a mental map (MM) is built from the CMs using (3), where d is the dimensionality and k is the cardinality of the set EVO_{MM}. The MMs occupy the fourth position of the tree onward:

$$MM_d = \sum_{i=1}^{k} EVO_{MM_i}(CM_d). \tag{3}$$

Once the MMs are obtained and concatenated with the remainder of the syntactic trees, the generated program is applied to each image. The n highest values are used to define the descriptor vector $\overrightarrow{v}$ for each image. The next step is to train a classifier using the feature vectors. In this work, an MLP is trained to create a model $f(\mathbf{x})$ that maps a set of descriptor vectors $\mathbf{x}_i$ to their corresponding labels $\mathbf{y}_i$, satisfying (1).

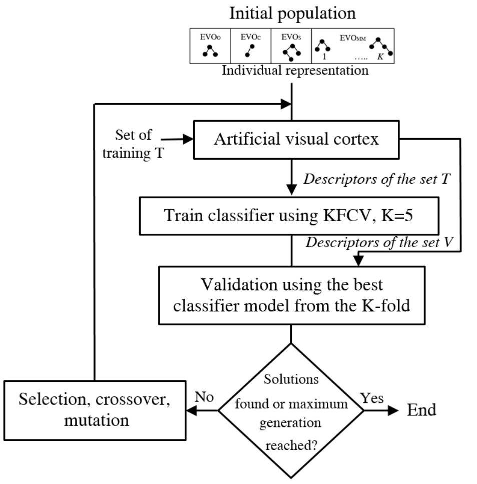

Initial population
EVO$_O$
EVO$_C$
EVO$_S$
EVO$_{MM}$
1
.....
K
Individual representation
Set of training T
Artificial visual cortex
Descriptors of the set T
Train classifier using KFCV, K=5
Descriptors of the set V
Validation using the best classifier model from the K-fold
Solutions found or maximum generation reached?
No
Yes
End
Selection, crossover, mutation

Fig. 2. Baseline method.

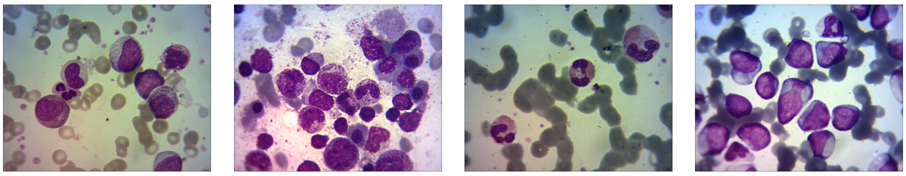
3 The Proposed Methodology

The Brain programming is the baseline method that we use to search for a set of mathematical expressions inspired on the functionality of specialized tissues in the brain in order to solve the classification problem. The evolutionary process of BP begins with a randomized initial generation. In this way, a set of initialization variables are defined, such as the population size, sizes of solutions or individuals, and crossing-mutation probabilities.

An individual represents a computer program written with a set of syntactic trees included in hierarchical structures. These individuals contain four kinds of functions that are defined in previous research by[9], one for each visual operator (VO). Each individual is encoded in a multitree representation. A variable number of syntactic trees, ranging from four to 10, compose each individual. The individuals are evaluated for each AVC performed, and a classifier determines their fitness for each generation.

Selection, crossover, and mutation processes are performed according the individual representation as suggested in [9], thus we consider that the whole individual is a chromosome, and each EVO within this is a gene. Finally, the stopping conditions are: (a) the algorithm reaches a maximum number of generations, or (b) the algorithm fitness reaches an optimal value; this means, all images are correctly classified. Fig. 2 presents the general scheme of the proposed approach.

Although in evolutionary computation it is common the use random sources to start the evolution, we propose to adapt the baseline method described above, in a similar way as in [8] to discover the best solutions for the classification task. To achieve this, we built a testbed with three experiments. In all experiments, we use the baseline method to classify the leukemia images M2, M3, M4, and M5. The first experiment uses only the line base method, whereas the second experiment uses the best solutions from the first experiment as the initial population.



(a) M2 (b) M3 (c) M4 (d) M5

Fig. 3. Types of acute myelocytic leukemia.

For the third experiment, in the first step, we use the baseline method to classify three different types of leukemia L1, L2, and L3. From the best solutions of this step, is created the initial population for the third experiment.

4 Experiments and Results

The implementation is carried out on a computer with an Intel i9-7900X, 64 GB of RAM, the Windows10 operating system, and MATLAB. Parameters initialization is as follow: population size 30, generations 30, method ramped half and half for generating of population, crossover rate 0.4, roulette wheel selection, mutation 0.1, tree max depth 50 levels, and a maximum length of genes 10. From the diagram in Figure 2, the evolutionary loop starts by computing the fitness of each AVC obtained from each individual using an MLP to calculate the classification rate using the training T and validation V sets.

The MLP has one intermediate layer with 50 neurons. Later, a set of AVCs is selected using roulette-wheel selection, and the best AVC is used for further processing. The new individual is created from the selected AVC by applying a crossover or mutation operation, as in [9]. The conditions to stop are (1) when the classification rate is equal to 100% or when (2) the number of generations reaches $N = 30$.

The dataset used is composed of 868 bone marrow smear color images from four subtypes of acute myeloid leukemia (AML): M2, M3, M4, and M5. We use 217 images per class, Fig. 3 shows a sample for each class. Image acquisition is performed by employing an optical microscope with a magnification of 1250 times and a camera coupled to the microscope. The images size is 1280×1024 pixels, which are resized to 256×320 pixels using bicubic interpolation.

For the experiments, we divide the dataset into three parts: a training set T, a validation set V, and testing set A. Each new individual is estimated by the classification error rate with the MLP from the best classifier model B of the k-folds. To select the best-performing solution, we test B on the validation set V. Later, we select one solution with the best validation error as the (near-) optimal feature descriptor, which is used for the model evaluation.

4.1 Results

From the proposed methodology, in order to assessment the model we test the best solution on the test A set using five-fold cross-validation. The average of the five test accuracies is the classification result.

Table 1. Results of classification.

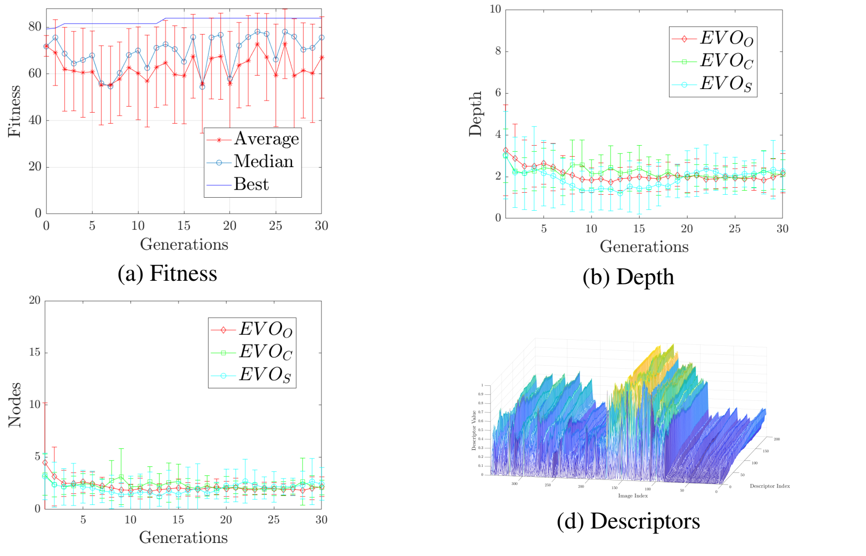
	Standard evolution			**Evolution from**			**Evolution from**		
	of four classes E1			four known classes E2			three unknown classes E3		
Run	Val. accuracy	Test accuracy		Val. accuracy	Test accuracy		Val. accuracy	Test accuracy	
		MLP	Random forest		MLP	Random forest		MLP	Random forest
1	77.31	72.67	75.82	81.92	76.34	79.49	78.84	79.38	**82.04**
2	73.46	66.07	69.71	83.84	80.31	**81.16**	77.69	70.05	73.76
3	72.69	72.24	74.15	84.61	79.83	80.71	76.92	74.01	78.08
4	81.92	77.36	**79.96**	81.53	77.91	79.99	78.84	70.50	69.32
5	80.00	73.25	76.53	83.07	72.18	75.76	77.69	76.77	80.35
6	71.92	68.90	72.14	81.92	76.92	79.83	-	-	-
7	78.07	75.35	76.93	81.92	78.15	79.68	-	-	-
8	75.38	72.08	74.22	81.92	75.36	79.89	-	-	-
Average	76.34	72.23	74.93	82.59	77.13	**79.57**	77.99	74.14	76.71
σ	±3.59	±3.51	±3.13	±1.12	±2.60	±1.63	±0.83	±4.01	±5.16
Outliers detected	0	0	0	0	0	1	0	0	0
Critical value Z	2.12	2.12	2.12	2.12	2.12	2.12	1.71	1.71	1.71

We repeat this process ten times and, the overall classification result is the accuracies' average of these repetitions. To make a more robust evaluation, we performed the independent test using two classifiers: an MLP and a random forest. Table 1 shows the results of the model evaluation. Seeing the results, we confirm the advantage of applying the best solutions discovered in previous experiments, despite using solutions from unknown leukemia types for the classification problem.

We consider the Grubbs statistical method for outlier detection [5].We run the experiments using Grubbs' test from the results of every experiment, with a significance level of 0.05 (two-sided). Values of average, standard deviation, outliers detected, and critical value Z are reported in Table 1. According to the results, there are no outliers, except in the experiment E2. Nevertheless, this is not significant because only one outlier was detected. In view of the best solutions are obtained by using the best individuals as the initial population using only the baseline method, we selected the best solution from run two to show the behavior during the evolutionary process.

The validation accuracy of this solution is 83.84%, and the test accuracy is 81.16% using random forest. This is a competitive result since the content of the images is visually similar. The fitness evolution, the average, standard deviation, and the best fitness thus far along the run are shown in Fig. 4 (a). Furthermore, in order to quantify the solution complexity we use the depth and the number of nodes. In this way, the Fig. 4 (b)-(c) depicts that the discovered solution has low complexity because the number of nodes and depth of trees decreases as the generations progress.

On the other hand, the properties of descriptor generated by the best solution in Fig. 4 (d) illustrate the range of descriptor values for this solution. The descriptor vector given by the AVC is a numerical vector of length 200. Since we use a balanced data set with the following percentages for each set: 30% for set T, 30% for set V, and 40% for set A; thus, set A contains 348 images.



(a) Fitness

(b) Depth

(c) Nodes

(d) Descriptors

Fig. 4. Behavior of the best solution. (a) Fitness, (b-c) Complexity, (d) Descriptors.

Table 2. Structure of best solutions.

No. Exp.	EVO_o	EVO_c	EVO_s	EVO_{MM_d}
$E1$	$log(K)$	S^2	S	$MM_1 = \sqrt{D_y(CM_d)}$
				$MM_2 = D_x(D_x(D_x(D_x(D_x(D_x(CM_d))))))$
				$MM_3 = CM_d/(D_x(D_x(CM_d)))$
$E2$	$log(K)$	S^2	$round(S)$	$MM_1 = D_y(CM_d)$
				$MM_2 = D_x(D_x(D_x(D_x(D_x(D_x(CM_d))))))$
				$MM_3 = CM_d/(D_x(D_x(CM_d)))$
$E3$	$G_{\sigma=1}(S)$	$Op_{b-y}(I_{rgb}) *$ $Op_{b-y}(I_{rgb})$	B	$MM_1 = Half(G_{\sigma=1}(G_{\sigma=1}(CM_d)))$
				$MM_2 = G_{\sigma=1}(CM_d)$
				$MM_3 = \frac{\sqrt{G_{\sigma=1}(G_{\sigma=1}(CM_d))} * G_{\sigma=1}(CM_d)}{G_{\sigma=1}(G_{\sigma=1}(CM_d))}$

Then, in Fig. 4 (d) the indices of images 1...87 correspond to class M2, and so on for classes M3, M4, and M5. Clearly, the descriptor values show the differences between the categories, which in some cases could be evaluated with simple techniques instead of using a classifier. Finally, in Table 2 we show the structure of the best solutions for each experiment. From these results, it is worth to note that the solutions can be read and are susceptible to simplifying.

In addition, as expected, experiments E1 and E2 give similar solutions due to the similarity of the images used in both cases. From results, we can see that the use of the best solutions from previous experiments like in E2 and E3, is useful to improve the classification results. Furthermore, the model used as baseline technique allows us to know how knowledge is derived to obtain the best solution.

This is a significant advantage, since experts need to recognize leukemia types from the characteristics of an image, as well as to know the process for learning to recognize them. Although we do not find a similar approach to our proposal, we provide experiments with a convolutional neural network for comparison. The network structure consists of four convolutional and max-pooling layers with 4, 8, 16, and 32 filters, respectively. Data augmentation is used on the training set, which includes horizontal and vertical reflexion.

Five-fold cross-validation is used to assess the training performance, which is stopped when the validation loss does not decrease for 20 epochs. Later, the net best model from the five-fold validation is selected and used with the test set. To evaluate the global classification performance, we repeat the experiment eight times, from which accuracy of $86.85\% \pm 5.88$ (mean ± s.d., runs=8) is obtained. From the results, it is worth noting that although the CNN achieves higher performance on the classification task, a critical problem is the lack of information about the learning process.

5 Conclusions

This paper presents an evolutionary vision model named brain programing, as a baseline method to solve the problem of leukemia recognition. An important advantage of this model is its capability for explaining how the knowledge to solve the problem is derived, which opens the possibility of answer the question: What is the visual task for?

Since a hierarchical structure and a functional composition perform the visual extraction and description, then the evolutionary cycle discovers the functions embedded in this structure. In two of the experiments shown, we use the best set of solutions from previous experiments as the initial population for a new set of experiments. The results showed that knowledge transfer through these good solutions improves the classification rate despite the fact that the solutions used as the initial population correspond to different types of leukemia from those to be classified.

Thus, it is to note that the use of knowledge transfer can improve the classification performance on the complex image classification task, in both similar domains and different domains. Due to the recognition problem requires a clear explanation to better understand the studied subject, the presented model can be helpful in the medical area for the recognition of diverse pathologies. Finally, future work from these results is to widen the number of different domains for knowledge transfer and address the issue of explainability in this knowledge transfer.

Acknowledgments. The authors would like to acknowledge the support provided by the Instituto Politécnico Nacional under projects 20200630 and 20210788, CONACYT under projects 65 (Fronteras de la Ciencia) and 6005 (FORDECYT-PRONACES), and CICESE through project 634-135 to carry out this research. The first author thanks the Autonomous University of Tlaxcala, Mexico for their support. The authors also express their gratitude to the Applied Computational Intelligence Network (RedICA).

References

1. Aljaboriy, S., Sjarif, N., Chuprat, S., Abduallah, W.: Acute lymphoblastic leukemia segmentation using local pixel information. Pattern Recognition Letters, vol. 125 (2019) doi: 10.1016/j.patrec.2019.03.024
2. Bodzas, A., Kodytek, P., Zidek, J.: Automated detection of acute lymphoblastic leukemia from microscopic images based on human visual perception. Frontiers in Bioengineering and Biotechnology, vol. 8, pp. 1005 (2020) doi: 10.3389/fbioe.2020.01005
3. Dinh, H., Chu, T., Nguyen, Q. U.: Transfer learning in genetic programming. pp. 1145–1151 (2015) doi: 10.1109/CEC.2015.7257018
4. Ferlay, J., Ervik, M., Lam, F., Colombet, M., Mery, L., Piñeros, M., Znaor, A., Soerjomataram, I., Bray, F.: Global cancer observatory: Cancer today. lyon, france: International agency for research on cancer. https://gco.iarc.fr/today
5. Grubbs, F. E.: Sample criteria for testing outlying observations. Ann. Math. Statist., vol. 21, no. 1, pp. 27–58 (1950) doi: 10.1214/aoms/1177729885
6. Iqbal, M., Xue, B., Al-Sahaf, H., Zhang, M.: Cross-domain reuse of extracted knowledge in genetic programming for image classification. IEEE Transactions on Evolutionary Computation, vol. 21, no. 4, pp. 569–587 (2017) doi: 10.1109/TEVC.2017.2657556
7. Ochoa-Montiel, R., Olague, G., Sossa, H.: Expert knowledge for the recognition of leukemic cells. Applied Optics, vol. 59, no. 14, pp. 4448–4460 (2020) doi: 10.1364/AO.385208
8. Olague, G., Chan Ley, M.: Hands-on artificial evolution through brain programming, pp. 227–253 (2020)
9. Olague, G., Clemente, E., Dozal, L., Hernández, D.: Evolving an artificial visual cortex for object recognition with brain programming. In: Schuetze, O., Coello, C. A., Tantar, A. A., Tantar, E., Bouvry, P., Moral, P. D., Legrand, P. (eds) EVOLVE - A Bridge between Probability, Set Oriented Numerics, and Evolutionary Computation III, Studies in Computational Intelligence, vol. 500, pp. 97–119 (2014)
10. Rodak, B. F., Carr, J. H.: Clinical Hematology Atlas (2016)