Graphical image classification combining an evolutionary algorithm and binary particle swarm optimization

Beibei Cheng*, Renzhong Wang*, Sameer Antani, R. Joe Stanley*, George R. Thoma

U.S. National Library of Medicine, National Institutes of Health, Bethesda, Maryland
*Department of Electrical and Computer Engineering,
Missouri University of Science and Technology, Rolla, Missouri

ABSTRACT

Biomedical journal articles contain a variety of image types that can be broadly classified into two categories: regular images, and graphical images. Graphical images can be further classified into four classes: diagrams, statistical figures, flow charts, and tables. Automatic figure type identification is an important step toward improved multimodal (text + image) information retrieval and clinical decision support applications. This paper describes a feature-based learning approach to automatically identify these four graphical figure types. We apply Evolutionary Algorithm (EA), Binary Particle Swarm Optimization (BPSO) and a hybrid of EA and BPSO (EABPSO) methods to select an optimal subset of extracted image features that are then classified using a Support Vector Machine (SVM) classifier. Evaluation performed on 1038 figure images extracted from ten BioMedCentral® journals with the features selected by EABPSO yielded classification accuracy as high as 87.5%.

Keywords: image processing, feature selection, Binary Particle Swarm Optimization (BPSO), Evolutionary Algorithm (EA), Support Vector Machine (SVM), graphical image

1. INTRODUCTION

A variety of biomedical images needed for instructional purposes or in support of clinical decisions are often found in biomedical articles, but are not easily accessible to retrieval tools. Broadly, the images found as figures in the articles can be classified into two categories: regular and graphical images, respectively. Regular images are those that are acquired through an imaging device and include MRI, CT, X-ray, photographs, etc. Graphical images (henceforth, graphics) are those that are created by authors to illustrate biomedical processes or content or biomedical data analyses. These images can be further classified into four classes: diagrams, statistical figures, flow-charts, and tables. Although tables are often represented in XHTML form in online articles today, older issues still provide them as images. It is necessary to annotate these images to support multimodal (image + text) medical information retrieval and clinical decision support systems. Graphical figure type identification is a key step toward such automatic annotation for figures extracted from scientific publications. The task of separating regular images from graphics is also a goal of the project and has been reported earlier [1].

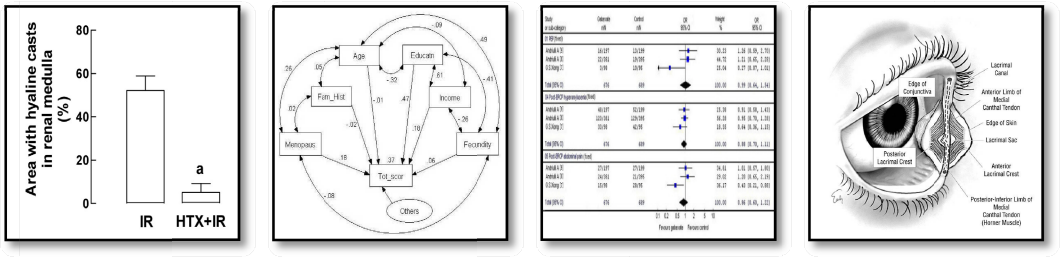
Graphics used in medical articles often appear in a variety of formats such as tables, graphs, flow charts, and diagrams as illustrated in Figure 1. For this paper, we use 1038 graphical images selected from ten BioMedCentral journals (Cancer, Cardio, Urology, Gastroenterology, Musculoskeletal Disorders, Nephrology, Ophthalmology, Pulmonary Medicine, Surgery and Dermatology) available in the Open Access dataset from the PubMedCentral® repository of the National Library of Medicine, part of the U.S. National Institutes of Health. The objective of this project is to develop a feature-based learning approach to identify the four graphics types, viz., table, graph, flow chart, and diagram.

This article describes our feature-based learning approach applying Evolutionary Algorithm (EA), Binary Particle Swarm Optimization (BPSO) and a hybrid of EA and BPSO (EABPSO) methods to an optimal subset of extracted image features which are then classified using a Support Vector Machine (SVM) [2] classifier. The proposed approach can address complex, hybrid, and composite graphics, which existing approaches [3, 4, 5] fail to identify satisfactorily. For example, the widely used Hough transform [3] can identify chart types such as pie charts and bar charts by detecting arc and line components inside the image based on the fact that arcs only appear in pie charts while vertical lines with similar length often exist in bar charts.

We also address other graphics types such as tables and flow charts with more complicated compositions. For example, the X-axis in a graph does not have to be present (Figure 1(a)); a flow chart can contain curved lines (Figure.1(b)); some

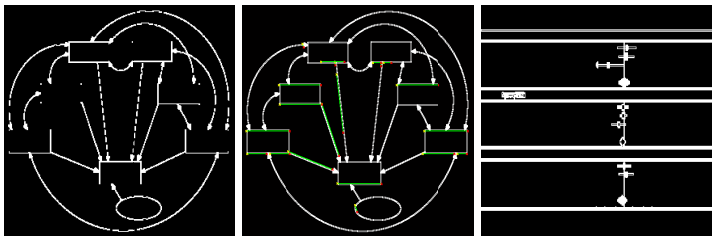
Document Recognition and Retrieval XIX, edited by Christian Viard-Gaudin, Richard Zanibbi,
Proc. of SPIE Vol. 8297, 829703 · © 2012 SPIE · CCC code: 0277-786X/12/$18 · doi: 10.1117/12.910533

blocks in a table may be missing (Figure 1(c)); and a diagram may have pictures within it (Figure.1(d)). Classifying such figures can be more challenging. To address these challenges, multiple features, including textural features [6, 7], region property features [8, 9], weighted density distribution (WDD) features [8, 10], Hough features [3] and hole object features, associated with the chart types are extracted. A hybrid of Evolutionary Algorithm [11] and binary Particle Swarm Optimization [12] (EABPSO) is employed to remove irrelevant features using the SVM classifier's output as feedback for evaluating the merits of a generated feature subset.



(a) (b) (c) (d)

Figure 1. Graphical image type examples. (a) Graph, (b) Flow chart, (c) Table, and (d) Diagram



(a) (b) (c)

Figure.2. Examples illustrating problems, discussed in Section 2, in existing approach after preprocessing. (a) Flow chart. (b) Flow chart with marked straight lines. (c) Table image.

2. RELATED WORK

Several approaches to chart type identification have been proposed with varying degrees of success. Zhou et al proposed a method of graphics type identification based on the Hough transform [3]. Huang et al presented a classification method that uses the revised diverse density algorithm [4]. Liu et al developed an approach using vectorized graphical information extraction from an image [5]. There are two common drawbacks in these existing approaches. First, these approaches are developed to identify a particular graphic, viz., statistical charts, such as pie, bar and line charts. Second, only a small number of charts have been examined, which make the performance results published for these algorithms inapplicable for the wide variety of graphical image types.

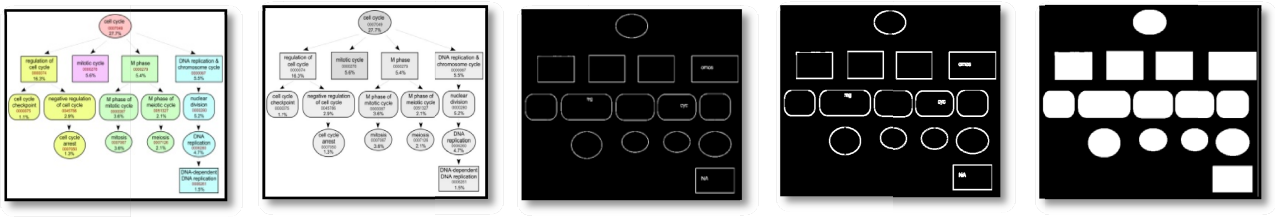
Techniques from existing research are implemented to detect flow charts. Take Figure 1(b) as an example. Text is first removed from an image and an edge map is obtained through edge detection (Figure 2(a)); Next, straight lines are detected by using the Hough transform (Figure 2(b)). There are three problems with this approach. First, it cannot detect all the lines if there are a large number of line segments of various lengths in a chart. Second, even if all of the lines and arcs are precisely detected by the Hough transform or by the Line net global vectorization [13], it is still difficult to determine the chart type due to complex composition. For example, both flow charts and diagrams can contain lines and arcs. Third, removing text from an image may also remove important information in the chart. For example, only lines are left after preprocessing the table (Figure 2(c)) using this approach.

To solve the first and second problems, in addition to lines and arcs, more features that are associated with the shape of a chart need to be considered, for example, the region property features and weighted density distribution features. For the third problem, textural features are extracted before removing the text. However, feature extraction may yield some features that are not relevant to classification. Incorporation of these unrelated features may have an adverse effect on the

classifier's performance. Feature selection [14], the process of selecting the best feature subset that contains the least number of features, contributes most to accuracy and efficiency. Both EA and PSO are stochastic search procedures and are generally suitable for solving this problem. However, EA has a slower convergence rate so that it usually takes longer to reach the global optimum while PSO has a higher convergence speed and is easily trapped in local optimum, as shown in Section 5. Therefore, a hybrid of EA and PSO is developed to combine the advantages of both and is described in Section 4.

3. METHODS

In our approach, an image is preprocessed in four steps with various features extracted at each step. This process is illustrated in Figure 3. The first step is to convert the original image (Figure 3(a)) to a gray scale image (Figure 3(b)) and extract textural features [6, 7]. Second, the gray scale image is binarized using Otsu's method [15] and then small objects with short length are removed so that it contains only frame-like objects (Figure 3(c)). Shape features, weighted density distribution features can be extracted. Third, the Sobel edge filter [16] is applied to the binary image and Hough features are computed (Figure 3(d)). Finally, the hole inside the binary image is filled [17] and features from these holes are extracted (Figure 3(e)).



(a) (b) (c) (d) (e)

Figure 3: Image preprocessing example. (a) Original image. (b) Gray image. (c) Binary image with chart frame. (d) Binary image with chart frame edge. (e) Binary image with hole objects.

The features thus obtained are then evaluated and subsets are selected by applying feature selection algorithms, i.e., EABPSO. The results from the classifier are used as feedback for assessing the merits of the candidate feature subset. This process proceeds iteratively until an optimized feature subset is obtained. The optimized feature subset is used as the input to the classifier, which determines the type of the chart.

3.1 Feature extraction

Relevant features extracted from the gray scale image and the three object mask images can be grouped into five sets, viz., textural features, shape features, weighted density distribution (WDD) features, Hough features and hole object features. Besides Hough features are used in previous research, textural features measure smoothness, coarseness, and regularity [-7] of an image; Shape features extract edge (object shape) characteristics in the image; WDD features measure the image's symmetry; hole object features compare the objects in an image. Table 1 summarizes these features and further explanations of these are given as follows.

A. Textural features: The textural features are items 1 to 6 in Table 1. However, these features use only histograms, which carry no information regarding the relative positions of pixels with respect to each other. To solve this problem, a co-occurrence matrix [7] is used which considers pixel position. Additional textural features from these co-occurrence matrices are labeled 7 to 30 in Table 1.

B. Shape features: These features are based on region properties and are obtained by applying the MATLAB® function *regionprops* [18] to the entire image. Ten region property features extracted in this project are listed in Table 1 labeled 31~40.

C. Weighted Density Distribution (WDD) features: The third set of 24 WDD features (rows 41 to 64 in Table 1) is obtained by computing a one dimensional shape profile of each object in the binary chart frame and correlating those profiles with WDD functions [19].

D. Hough features: After the straight lines inside an image are identified by the Hough transform algorithm, five Hough features are generated as shown in Table 1 rows 65 to 73.

E. Hole object features: This set of features is generated from the hole objects inside an image, as illustrated in Figure 4(d). The hole objects are obtained by applying the hole filling algorithm [17]. Five hole object features are generated for the sample image in this paper. They are shown in Table 1 rows 74 to 79.

3.2 Classifier

SVM classifier is chosen since it delivers a deterministic solution. The key features of SVMs are the use of kernels, the absence of local minima, the sparseness of solution, and the capacity control achieved by optimizing the margin.

Table 1. Extracted Features

Feature set	Label	Measure	Description
Textural features	1	Mean of histogram	The first moment of gray image
	2	Variance of histogram	The second moment of gray image
	3	Skewness of histogram	The third moment of gray image
	4	Flatness of histogram	The fourth moment of gray image
	5	Maximum of histogram	Uniformity of gray image
	6	Entropy of histogram	Average entropy of gray image
	7~10	Contrast	The intensity contrast of correlation matrices
	11~14	Correlation	The correlation of correlation matrices
	15~19	Uniformity	The uniformity of correlation matrices
	20~23	Closeness	The homogeneity of correlation matrices
	24~27	Strongest response	The maximum probability of correlation matrices
	28~30	Randomness	The average entropy of correlation matrices
Shape features	31	MajorAxisLength	Length (in pixels) of the major axis of the ellipse that has the same normalized second central moments as the region.
	32	MinorAxislength	Length (in pixels) of the minor axis of the ellipse that has the same normalized second central moments as the region .
	33	Axis ratio	Ratio of MajorAxisLength to MinorAxislength.
	34	Normalized area	Area of the region divided by the whole image.
	35	Solidity	Area of the region divided by the convex hull area.
	36	EulerNumber	The number of objects in the region minus the number of holes in those objects.
	37	EquiDiam	The diameter of a circle with the same area as the region.
	38	Extent	Ratio of area to bounding box area
	39	Horizontal MinPixelNo	The minimum number of intersection area for the object and its bounding box horizontally.
	40	Vertical MinPixelNo	The minimum number of intersection area for the object and its bounding box vertically.
WDD features	41~64	WDD	Correlation of binary chart frame and WDD function.
Hough features	65	Line number	Number of straight lines
	66	Longest line length	Longest line's length
	67	Longest line slope	Longest line's slope
	68	2nd Longest line length	Second longest line's length
	69	2nd Longest line slope	Second longest line's slope
	70	Line slope	Average value of lines' slope
	71	Line length	Average value of lines' length
	72	Variance of line slope	Variance of lines' slope
	73	Variance of line length	Variance of lines' length
Hole object features	74	Hole number	Number of hole objects
	75	Largest hole area	Area of largest hole object
	76	Hole area	Average hole objects' area,
	77	Area variance	Variance of hole objects' area,
	78	Area ratio	Average ratio of hole object's area to its bounding box's area
	79	Area ratio variance	Variance of ratio of hole object's area to its bounding box's area.

4. OPTIMAL FEATURE SELECTION

After the 79 features listed in Table 1 are extracted from the images, Evolutionary Algorithm (EA), Binary Particle Swarm Optimization (BPSO) and a combined Evolutionary Algorithm and Binary Particle Swarm Optimization (EABPSO) are then applied to obtain the optimal feature subset. They all use the same scheme for candidate (feature subset) representation, where each individual in a population is an N-dimensional binary vector with each element of the vector representing a feature and N being the total number of features. For each element of the binary vector, '1' means that the corresponding feature is selected. The initial population is randomly initialized in the sense that each element in a vector is randomly picked as 0 or 1. The fitness values for EA, BPSO and EABPSO are set to the accuracy of the SVM classifier applied to the selected feature set. The algorithms for generating the candidate feature subsets in EA, BPSO and EABPSO are described below.

4.1 Evolutionary Algorithm (EA)

The offspring of the Evolutionary Algorithm are generated as follows: 1) randomly select two parents from the parent pool of M initial candidates; 2) generate two offspring by applying a uniform [20] crossover operator; 3) offspring are then altered by performing a mutation operation. A random parameter ranging from 0 to 1 is generated for each bit of the candidate vector, which will flip once the parameter is greater than a predefined threshold. The next parent pool is selected based on whether the parents or their offspring maximize the classification accuracy. The same process is used for obtaining the next generation of offspring and this process is repeated for N epochs. From the final parent pool, the parent which maximizes the classification accuracy is selected as the final result. Since EA evaluates many points simultaneously in the search space it is more likely to find the global solution but at the cost of higher computation time.

4.2 Binary Particle Swarm Optimization (BPSO)

In implementing BPSO, the velocity and position of a candidate are computed using Eq.1 and Eq. 2 respectively. A sigmoid transformation of the velocity component is applied to keep the velocity values constrained in the range (0, 1). However, the BPSO algorithm can be easily trapped into a local minimum and may lead to premature convergence. It has been observed that when BPSO reaches a local optimal solution, all particles tend to gather around it making it difficult to find a global optimum.

$$V_{mn}(t+1) = wV_{mn}(t) + c_1 rand_1\left(Pbest_{mn} - X_{mn}(t)\right) + c_2 rand_2\left(Gbest_{mn} - X_{mn}(t)\right) \quad (1)$$

$$X_{mn}(t+1) = \begin{cases} 1 & \text{if } (rand < \frac{1}{1+e^{-V_{mn}(t+1)}}) \\ 0 & \text{else} \end{cases} \quad (2)$$

where t is the iteration index (time step), m is the current particle ($1 \leq m \leq M$) in a population of M, n is the attribute element ($1 \leq n \leq \boldsymbol{N}$), $\boldsymbol{V_{mn}(t)}$ is the particle's current velocity, $\boldsymbol{V_{mn}(t+1)}$ is the particle's new velocity, $\boldsymbol{X_{mn}(t)}$ is the particle's current position, and $\boldsymbol{X_{mn}(t+1)}$ is the particle's new position, $\boldsymbol{Gbest_{mn}}$ is the global best position, $\boldsymbol{Pbest_{mn}}$ is the previous best position, $\boldsymbol{c_1}$ and $\boldsymbol{c_2}$ are the random value from 0 to 1, $\boldsymbol{w}$ is the learning weight, selected from 0 to 1.

4.3 EABPSO

To address the individual shortcomings of these two algorithms, we design EABPSO combining the feature evolution idea of EA and BPSO into a hybrid solution appropriate for discrete (binary) problems. EABPSO is an improvement over prior hybrid evolutionary algorithms [21, 22] that solved continuous problems.

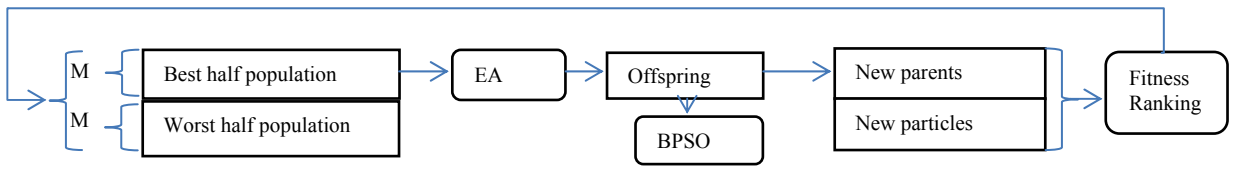


Figure 4. Overview of EABPSO procedure.

Figure 4 shows an overview of the EABPSO procedure. As can be seen, EA and BPSO both work with the same initial population. To solve an M-dimensional problem, 2M individuals are randomly generated in the sense that each element

in an individual is randomly picked from 0 or 1. These individuals may be considered analogous to chromosomes in the case of EA, or as particles in the case of BPSO. The 2M individuals are sorted by fitness, and the top M individuals are fed into EA to create M new offspring by crossover and mutation operations, as described in section 4.1. The new offspring are used as the input to BPSO to compute M new particles as described in section 4.2. The new parent and new particles are combined and sorted in preparation for repeating the entire run.

5. EXPERIMENTAL SETUP AND RESULTS

The experimental data set consists of 1038 medical images annotated by type including 306 diagrams, 329 graphs, 154 tables, and 249 flow charts, which are selected from the ten BioMedCentral journals mentioned in Section 1. 79 features are extracted from these images and optimal features selection step is applied. In the BPSO algorithm, the inertia weight w is empirically set to 0.8, the cognitive acceleration constant c_1, 1, and the social acceleration constant c_2, 1. In EA, the uniform crossover operator evaluates each bit in the parent strings for exchange with a probability of 0.5. The predefined mutation threshold is set as 0.8. EABPSO shares the same parameters with BPSO and EA. In addition, the dimension size (N) is the same as the number of features. For training the SVM, Platt's sequential minimal optimization algorithm [23] was implemented. It globally replaces all missing values and transforms nominal attributes into binary ones. It also normalizes all attributes and uses the polynomial kernel by default. A three-fold cross validation is used to set up the training and testing data sets. The data set is divided into three parts where 2/3rd is used for training and the rest is used for testing. This procedure is repeated three times. Therefore, for each time, the training set is 692 and the representative test set is 346 images. The accuracy of the classifier presented in the next section is based on averaging the accuracy of the three test sets.

Seven different schemes for feature subset selections are used. They are as follows: (i) **Case 1**: EA feature selection with uniform crossover operator; (ii) **Case 2:** BPSO feature selection; (iii) **Case 3**: EABPSO; (iv) **Case 4**: voting algorithm based on the selected feature set. The voting algorithm selects features based on the frequency of their occurrence in the three feature selection algorithms; (v) **Case 5**: Chi-square statistic [24]; (vi) **Case 6**: information gain [25] also used in order to compare classifier performance against EA, BPSO, EABPSO; and, finally, (vii) **Case 7**: uses all features as the input.

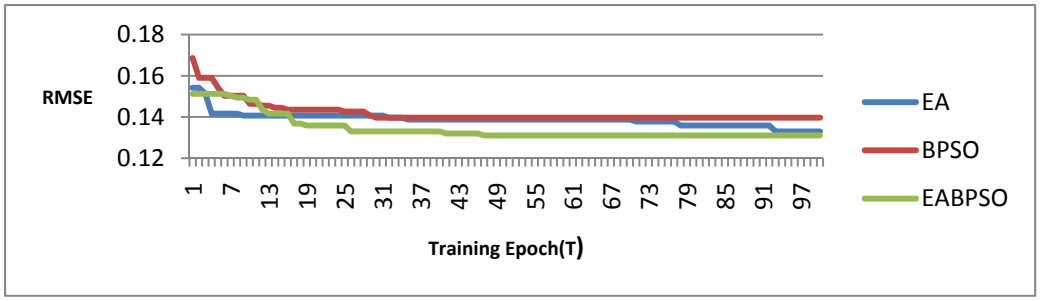


Figure 5. Root Mean Squared Error (RMSE) versus iteration number for each of the feature selection schemes.

In Figure 5, the root mean square error (RMSE) performance measures (1-Accuracy) for cases 1, 2 and 3 are shown as the training progresses for one run. The population size (M) is 30 and the total training epoch (T) is 100. Table 2 shows the final accuracy of the SVM for feature subset from case 1 to case 7. We choose the particle size (M) to be 20 and 30, the total training epoch (T) to be 50, 100 and 150. Within the different combinations of the particle size (M) and the total training epoch (T), for case 1 to case 3, the best and averaged accuracy for the ten runs are listed in Table 2. The number of features in the subset is listed after the best accuracy. The accuracy of cases 4, 5 and 6 is also listed in Table 2 based on the feature combination that gives the best accuracy for case 1, 2 and 3.

As can be seen in Figure 5, for one hundred epochs, the accuracy ranking from high to low is 1) EABPSO, 2) EA, and 3) BPSO. BPSO stops converging at epoch 31; EA keeps converging until epoch 94; EABPSO achieves the global minimum at epoch 48. Figure 5 shows that BPSO algorithm gets trapped at the local minimum although it has a very fast convergence speed. EA does a good job to reach the lowest RMSE value but it takes a long time. Since EABPSO combines the evolutionary ideas of both BPSO and EA where BPSO helps to enhance the offspring created by EA in order to generate fitter feature combinations (elites) in each epoch, EABPSO has the ability to attain the best RMSE at a

higher speed. Also, with N features, the computational complexity is on the order of $O(N!)$ if every possible combination of features is explored. The PSO/EA and EABPSO reduce the complexity to $O(N)$.

5.1 Discussion

As seen in Table 2, the best overall classification accuracy of 87.5% was achieved with feature subset selected by BPSO (case 2) and EABPSO (case 3) for population size 20 and training epoch 100/150. For six different combinations of population size (M) and training epoch (T), EABPSO has the best accuracy. The average overall classification accuracy, 87.0% was achieved by both EA and EABPSO. This shows that EABPSO keeps delivering good and consistent results. Further, with the same particle size (M =30) as the epoch increases from 100 to 150 the average accuracy increases in EA (case 1) while staying the same in both BPSO (case 2) and EABPSO (case 3). This is due to longer convergence time for EA also shown in Figure 5. For six different combinations of population size (M) and training epoch (T), the number of features for the best subset obtained by EABPSO (case 3) stays around 45, while EA and BPSO have a larger range. Also, the advantage of using the voting algorithm for feature selection is seen through the consistently good overall performance of case 4 since the voting algorithm's input features are the best selected features of cases 1,2,3. Justification for the usefulness of BPSO, EA and EABPSO is found through the observation that the highest classifier accuracies are achieved from cases 1-4.

Table 2. Performance comparisons for different feature combinations. (M = population size; T = maximum #iterations, Performance/XX: Here XX = the number of features used in computing the performance). Best Performance in **bold**.

Case No.	Accuracy	M=20, T=50	M=20, T=100	M=20, T=150	M=30, T=50	M=30, T=100	M=30, T=150
1. EA	(Average)	0.863	0.867	0.867	0.866	0.868	0.870
	(Best)	0.866/49	0.870/38	0.868/46	0.869/51	0.871/45	**0.872/49**
2.BPSO	(Average)	0.864	0.869	0.869	0.867	0.867	0.867
	(Best)	0.866/48	**0.875/51**	**0.875/51**	0.870/46	0.870/52	0.870/50
3.EABPSO	(Average)	0.870	0.870	0.870	0.868	0.869	0.869
	(Best)	**0.872/48**	**0.875/45**	**0.875/46**	**0.871/48**	**0.872/43**	**0.872/43**
4.Voting algorithm	Accuracy	0.870	0.872	0.872	0.869	0.870	0.870
5.Chi square	Accuracy	0.846	0.851	0.851	0.846	0.837	0.837
6.Information gain	Accuracy	0.846	0.850	0.850	0.846	0.836	0.836
7.All features	Accuracy	0.863					

6. CONCLUSION

This paper proposes a framework for graphical image type identification based on image feature analysis and computational intelligence techniques [26, 27]. Several feature extraction techniques are applied to the preprocessing of the images. Multiple features associated with the chart types are then extracted. EA and binary PSO are employed to find the optimal subset of features since both are stochastic search procedures and are generally suitable for solving the optimization problem. PSO has a higher convergence speed but easily trapped in local optimum while EA usually takes longer time to reach the global optimum although it has a mutation operator that can keep it out of local minimum. Thus EABPSO is proposed to combine the new individual generation functions of both EA and PSO, to attain the global minimum at high speed. The experimental results demonstrate that integration of various image processing techniques, feature extraction techniques, and computational intelligence methods for optimal feature selection as proposed in this paper can achieve high classification accuracy.

ACKNOWLEDGMENT

This work was supported by NLM under contract number 276200800413P and the Intramural Research Program of the National Institutes of Health (NIH), NLM, and Lister Hill National Center for Biomedical Communications (LHNCBC).

REFERENCES

1. M. Simpson, M.M. Rahman, S. Singhal, D. Demner-Fushman, S. Antani, G. Thoma. Text- and Content-based Approaches to Image Modality Detection and Retrieval for the ImageCLEF 2010 Medical Retrieval Track. CLEF 2010 LABs and Workshops Notebook Papers. ISBN 978-88-904810-2-4. ISSN 2038-4963. September 2010.
2. D. Meyer, F. Leisch, K. Hornik, The support vector machine under test. Neurocomputing, vol. 55, pp. 169-186, 2003.
3. Y. P. Zhou, C. L. Tan, Hough technique for bar charts detection and recognition in document images. International Conference on Image Processing, pp. 494-497, 2000.
4. W. Huang, Chart Image Classification Using Multiple-instance Learning. IEEE workshop on Applications of Computer Vision, pp. 27, 2007.
5. R. Liu, W. Huang , C. L. Tan, Extraction of Vectorized Graphical Information from Scientific Chart Images, Document Analysis and Recognitio, ICDAR, Ninth International Conference on, vol.1, pp.521-525, 2007.
6. R. C. González, R. E. Woods, Digital Image Processing, Pearson/Prentice Hall, 2008.
7. R.M.Haralick, K. Shanmugan, I. Dinstein, Textural Features for Image Classification, IEEE Transactions on Systems, Man, and Cybernetics, vol. SMC-3, pp. 610-621, 1973.
8. B. Cheng, R. J. Stanley, S. Antani, G. Thoma, A Novel Computational Intelligence based Approach for Medical Image Artifacts Detection, Proceedings of the 2010 International Conference on Artificial Intelligence and Pattern Recognition. Orlando, FL, pp.113-120, 2010.
9. B. Cheng, S. Antani, R. J. Stanley, Automatic segmentation of subfigure image panels for multimodal biomedical document retrieval, Proceedings of the SPIE Electronic Imaging, vol. 7874, pp. Z-1-11, 2011.
10. R.J. Stanley, J. Keller, P. Gader, C.W. Caldwell, Data-driven homologue matching for chromosome identification. IEEE Trans Med Imaging, vol. 17, no.3, pp, 451-462, 1998.
11. J. H. Holland, Adaptation in Natural and Artificial Systems. University of Michigan Press, Ann Arbor, 1975.
12. J. Kennedy, R. Eberhart, Particle swarm optimization. Proceedings of the IEEE International Conference on Neural Networks, Piscataway, NJ, pp. 1942-1948, 1995.
13. J. Song, F. Su, J. Chen, C. L. Tai, and S. Cai, Line net global vectorization: an algorithm and its performance analysis. IEEE Conference on Computer Vision and Pattern Recognition, South Carolina, pp. 383-388, 2000.
14. H.C. Peng, F. Long, C. Ding, Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy. IEEE Trans Pattern Anal and Machine Intel, 27(8), pp. 1226–1238, 2005.
15. M. Sezgin, B. Sankur, Survey over image thresholding techniques and quantitative performance evaluation, Journal of Electronic Imaging, vol. 13, no. 1, pp. 146-165, 2003.
16. I. Sobel, G. Feldman, A 3x3 Isotropic Gradient Operator for Image Processing. Presented at a talk at the Stanford Artificial Project in 1968, unpublished but often cited, orig. in Pattern Classification and Scene Analysis, Duda R and Hart P, John Wiley and Sons, pp. 271-272, 1973.
17. Soille, P., Morphological Image Analysis: Principles and Applications, Springer-Verlag, 1999, pp. 173-174.
18. D. C. Hanselman, B. L. Littlefield, Mastering MATLAB 7. Prentice Hall, 2004.
19. J. Piper, Granum E. On fully automatic feature measurement for banded chromosome classification. Cytometry, vol.10, pp.242-255, 1989.
20. G. Sywerda, Uniform crossover in genetic algorithms. Proceedings of the third international conference on genetic algorithms, pp.2-9, 1989.
21. Y. Kao, E.Zahara, A hybrid genetic algorithm and particle swarm optimization for multimodal functions, Elsevier, Applied Soft Computing, vol: 8(2), pp. 849-857, 2008.
22. C. Juang, A hybrid of genetic algorithm and particle swarm optimization for recurrent network design, Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on , 34(2), pp. 997- 1006, 2004.
23. J. Platt, Machines using Sequential Minimal Optimization. In B. Schoelkopf and C. Burges and A. Smola, editors, Advances in Kernel Methods - Support Vector Learning, 1998.
24. H. Liu, and R. Setiono, Chi2: Feature selection and discretization of numeric attributes, Proc. IEEE 7th International Conference on Tools with Artificial Intelligence, pp.338-391, 1995.
25. T.M. Mitchell, Machine Learning, the Mc-Graw-Hill Companies, Inc, 1997.
26. B. Cheng, R. J. Stanley, T. Szalapski, W. V. Stoecker, A Hybrid Computational Intelligence Algorithm for Automatic Skin Lesion Segmentation in Dermoscopy Images, Proceedings of Artificial Neural Networks in Engineering (ANNIE), pp. 379-386, 2010.
27. B. Cheng, R.J. Stanley, W.V. Stoecker, Automatic Telangiectasia Analysis in Dermoscopy Images Using Adaptive Critic Design, Skin Research and Technology, in Press, 2011.