African Journal of Science and Technology (AJST)
Science and Engineering Series Vol. 12, No. 1, pp. 7 - 13

SUPERVISED COLOR IMAGE SEGMENTATION, USING LVQ NETWORKS AND K-MEANS. APPLICATION : CELLULAR IMAGE

M. Messadi, A. Feroui and A. Bessaid

Laboratory of Biomedical Engineering, Electronic Department, Faculty of Science Engineering, University of Abou Bekr Belkaid, Tlemcen BP 119, 13000, Algeria,

Email: m_messadi@mail.univ-tlemcen.dz; a_bessaid@mail.univ-tlemcen.dz

ABSTRACT: *TThis paper proposes a new method for supervised color image classification by the Kohonen map, based on LVQ algorithms. The sample of observations, constituted by image pixels with 3 color components in the color space, is at first projected into a Kohonen map. This map is represented in the 3-dimensional space, from the weight vectors resulting of the learning process . Image classification by kohonen is a low-level image processing task that aims at partitioning an image into homogeneous regions. How region homogeneity is defined depends on the application. In this paper color image quantisation by clustering is discussed. A clustering scheme, based on learning quantisation vector (LVQ), is constructed and compared to the K-means clustering algorithm. It is demonstrated that both perform equally well. However, the former performs better than the latter with respect to the known number of although class. Both depend on their initial conditions and may end up in local optima. Based on these findings, an LVQ scheme is constructed which is completely independent of initial conditions; this approach is a hybrid structure between competitive learning and splitting of the color space. For comparison, a K-means approach is applied; it is known to produce global optimal results, but with high computational load. The clustering scheme is shown to obtain near-global optimal results with low computational load*

Keywords: *color image, kohonen, LVQ, classification, K-means*

INTRODUCTION

In this paper the problem of color image quantisation is discussed. Color quantisation consists of two steps: template design, in which a reduced number of template colors (typically 8-256) is specified, and pixel mapping in which each color pixel is assigned to one of the colors in the template. From the pattern recognition point of view, color quantisation can be regarded as a supervised classification of the (2D) color space, each class being represented by one color template. Since an RGB image can contain up to (256)3 distinct colors, the classification problem involves a large number of data points in a low dimensional space. Several techniques exist for color quantisation. First, there is the class of splitting algorithms that divide the color space into disjoint regions, by consecutive splitting up of the space. From each region a color is chosen to represent the region in the color template. Another class of quantisation techniques performs clustering of the color space, and cluster representatives are chosen as template colors. A frequently used clustering algorithm is the K-means clustering algorithm. Here, an iterative updating of the cluster representatives and an assignment of color pixels to clusters takes place. Clustering algorithms are commonly accepted as optimal quantisation approaches, but are also known as very time consuming. Moreover, although optimal, the above clustering algorithms suffer from their dependence on initial conditions. In most applications, one specific initial condition is chosen to present the results. However, using other initial conditions can change the performance of the algorithm dramatically. In this paper, the problem of local optima in color image quantisation is studied, by applying several clustering techniques. First of all, K-means is compared to a Competitive Learning Vector technique (LVQ). LVQ is very similar to K-means, in the sense that it minimizes the same objective function and the numbers of class are known. The main difference is that when using LVQ, cluster centers are updated sequentially [1].

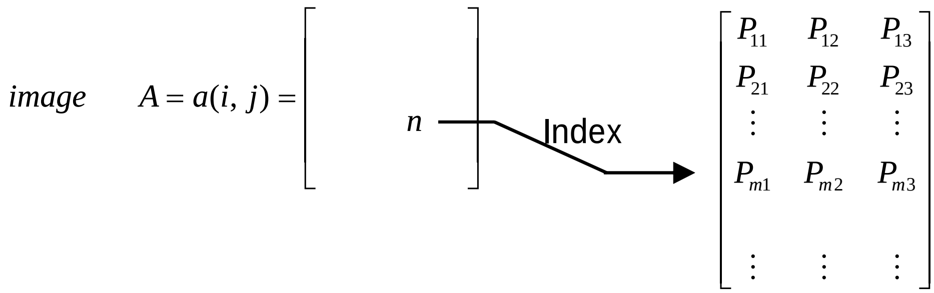

Figure 1: Connection between the image and the template

CLUSTERING ALGORITHMS

Principle

The separation of color from a topographic map is a problem of image color segmentation [4]. Information, which we can extract from this type of map, is represented with the colors:

Images that we have in entrance are coded on three plans: red, green and blue. The color of each pixel is given by a vector with three components. Generally, this vector is given in the RGB space color. The three elements of each vector corresponding to the RGB components of every pixel of the image to be reprocessed are presented at the entrance of a classifier [10] with the aim of doing a color separation. An example of a data file with the three components, R, G, B is given below (Table 1).

Table 1

R	G	B	color
60	193	111	Blue
166	182	109	Green
226	48	242	Brown
0	0	0	Black
12	17	16	Black
200	190	140	Brown
161	24	255	Green
…	…	…	….

The use of the color in image segmentation is a relatively recent research topic. Although one finds several algorithms of color segmentation, but the literature is not rich enough than that for images in grey level [9].

The method we propose to solve the problem of the color classification uses the Kohonen model LVQ, the results we obtained using this approach are compared to the k-means classifier.

The K-Means clustering Algorithm

K-means algorithm [7] is a post-clustering technique that is widely used in image coding and pattern recognition. A sequence of iterations starts with some initial set $\overline{C}^{(0)}$. At each iteration t set all data points $c \in C$ are assigned to one of the clusters $\overline{S_k}^{(t)}$ as defined in (2). A new centred $\overline{C_{(k)}}^{(t)}$ for a cluster is computed as follows:

$$\overline{c_j}^{(t+1)} = \frac{1}{t}\sum_{i=1}^{t}\left(c_i \middle| c_i \in \overline{S_k}^{(t)}\right) \qquad (1)$$

and

$$S_k = \left\{c \in C : q(c) = \overline{c_k}\right\} \qquad (2)$$

$\overline{S_k}^{(t)}$: The quantisation mapping defines a set of clusters

The algorithm is known to converge to a local minimum. The K-means algorithm was used to quantize images in [6]. For the test images it produced smaller average errors $\in_{q(C,I)} = \frac{1}{M}\sum_{(x,y)\in I}\left\| c_{(x,y)} - q(c_{(x,y)})\right\|$ than the median cut and variance-based pre-clustering algorithms. Unfortunately, the high cost of computation makes K-means impractical for image quantisation.

The Learning Vector Quantisation (LVQ)

This model corresponds to a layer of neurons and an entrance layer (Fig. 2).

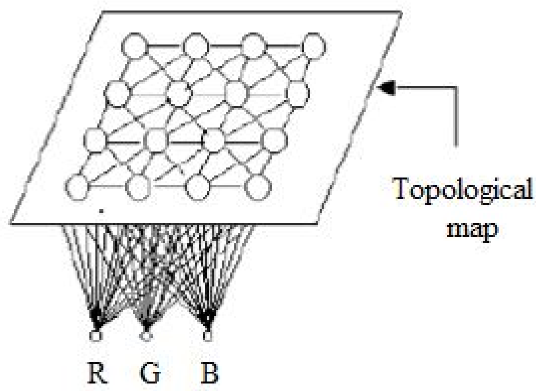

Figure 2: Topological map

- The entrance layer serves only for the presentation of the entrance vectors (components R G B of the different pixels).
- An adaptation layer formed by a neuron network. These neurons are some simple linear and are connected to all components R,G,B of the entrance layer.

Every neuron j of the topological map calculates a distance between the x example presented to the entrance and its weight vector Wj (entrance vectors x and weight vectors W of the neurons of the map have the same dimension). The neuron j* (winner) is then the one that has the minimum distance [9].

$$\left\| x - W_{j*} \right\| = \min \left\| x - W_j \right\| \quad (3)$$

The Euclidean distance d_j is very often used in the domain of the classification. It is calculated as follows:

$$d_j = \sum_{i=0}^{N} (x_i - W_{ij})^2 \quad (4)$$

N is the dimension of the entrance vector, it is equal to 3 (RGB components) in our case.

Therefore, we determine the neuron whose weight vector is nearest to the sense of the Euclidean distance, of the presented vector x. It then adjusts its weight in order to come closer to the example presented again. In its topological version, every winning neuron incites these neighbors to also modify their weight in the same sense. If one notes j* the winning neuron, Vj* its neighborhood and Wj the vector of weight of a j neuron, modifications that are going to take place after presentation of the vector of x entrance to the iteration at time "t" are going to be:

The LVQ1

Assume that a number of 'codebook vectors' W (free parameter vectors) are placed into the input space to approximate various domains of the input vector x by their quantized values. Usually several codebook vectors are assigned to each class of x values, and x is then assumed to belong to the same class to which the nearest W belongs. Let

$$c = \arg\min \left\{ \left\| x - W \right\| \right\}$$

define the nearest W to x, denoted by Wc. Values for the W that approximately minimize the misclassification errors in the above nearest-neighbor classification can be found as asymptotic values in the following learning process. Let x(t) be a sample of input and let the W represent sequences of the W in the discrete-time domain. Starting with properly defined initial values, the following equations define the basic LVQ1 process [8]:

$$\begin{aligned} W_i(\tau+1) &= W_i(\tau) + \alpha(\tau)\left(X - W_k(\tau)\right) \\ W_j(\tau+1) &= W_j(\tau) - \alpha(\tau)\left(X - W_k(\tau)\right) \end{aligned} \quad (5)$$

$\alpha^{(t)}$ is a gain term $(0 < \alpha^{(t)} < 1)$ that decreases in time.

Algorithm: [10]

Step 1: Initialize Weights from N inputs to M output-nodes shown in Fig. 2 to small random values.

Step 2: Present New Input

Step 3: Compute Distance to all Nodes Compute distances dj between the input and each output node j using

$$d_j = \sum_{i=0}^{N-1} (x_i^{(t)} - W_{ij}^{(t)})^2$$

where $x_i^{(t)}$ is the input to node i at time t and W_{ij} is the weight from input node i to output node j at time t.

Step 4: Select output Node With Minimum Distance.
Select node j* as that output node with minimum distance dj.

Step 5: Update Weights to Node j* and Neighbors
Weights are updated for node j* and all nodes in the neighborhood defined by $NE_{i*}^{(t)}$. New weights are changed by supervised update called LVQ1.

$$W_i(\tau+1) = W_i\ (\tau) + \alpha(\tau)\big(X - W_k(\tau)\big)$$
$$W_j(\tau+1) = W_j\ (\tau) - \alpha(\tau)\big(X - W_k(\tau)\big)$$

Step 6: Repeat by Going to Step 2

The data file dedicated to the training includes three columns of the RGB (Figure 3) components and a fourth column. In this fourth column the label of the color correspondsto the RGB and is given (Figure 4). Note that a preliminary manual operation must be done by an operator where he must select maps to study zones corresponding to different colors (brown, black, green, blue, etc.) in order to construct a data file prototype.

$$x = \begin{bmatrix} R & G & B \\ 11 & 9 & 3 \\ 172 & 196 & 27 \\ 146 & 178 & 25 \\ & \vdots & \end{bmatrix}$$

Figure 3: Example of input image vector In RGB space

$$\mathrm{W} = \begin{bmatrix} 11 & 9 & 3 & black \\ 172 & 196 & 27 & yellow \\ 146 & 178 & 25 & green \\ & \vdots & & \end{bmatrix}$$

Figure 4: example of vector weight

APPLICATION

In this section, two experiments are carried out and discussed to demonstrate the performance of the different clustering algorithms LVQ and K-means. The images used are RGB color images of 100x100 pixels. In the LVQ1, the experiment related to the dependence of the method on initial conditions is investigated. Several strategies are possible to obtain an initial set of template colors for starting a clustering algorithm. An obvious choice is a random initial set. Applying the quantised on different initial sets independently, allows one to study in a statistical way the influence of the initial conditions on the behaviour of the algorithm. A statistically representative number of initial sets is constructed and the algorithm is applied on each set independently. The distribution obtained in Figure 5 is shown for quantisation of color image 'test', quantised to 4 colors, after 10000 independent runs, using random initial conditions, after K-means.

In Figure 6, a few discrete local optima are clearly visible, which indicates that K-means converges to a local optimum. However, the distance between different local optima is large. Finally in Figure 6, a narrow distribution on the left hand side demonstrates that the classification approach, which is independent of initial conditions, converges to a solution near the real global optimum. The distribution shows the dependence of the competitive learning algorithm on the learning order of the presentation of the color pixels. The effect of this dependence is only a few percent. In the second experiment, the algorithms K-means is compared to LVQ1 when fixed initial conditions are applied.

	A number of pixels of the weights	A number of pixels after classification		Recognition pixels		Rate of Recognition by colors		Total rate	
		LVQ1	K-mean	LVQ1	K-mean	LVQ1	K-mean	LVQ1	K-mean
Brown	715	712	718	712	715	99.58%	100%	Correct total rate 99.79%	Correct total rate 99.38%
Green	8145	8127	8123	8127	8123	99.78%	99.73%		
Bleu	448	458	461	448	430	100%	95.98%	Harm total rate 0.21%	Harm total rate 0.62%
Mauve	692	703	698	692	670	100%	96.82%		

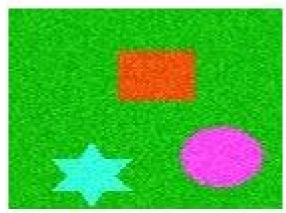

Figure 5a: Images test

Comparisons:

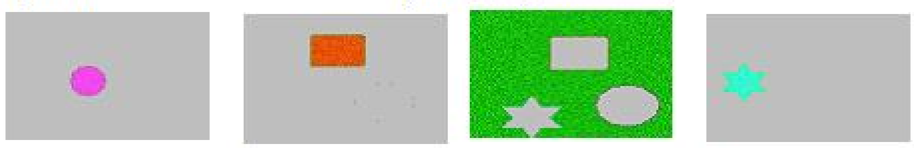

Figure 5b: separation the mauve, brown colors, green and blue colors

Figure 5: LVQ1 classification

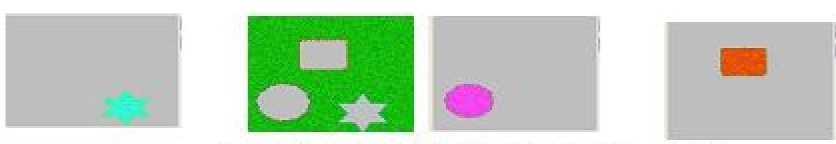

Figure 6: Classification by K-means with 4 weights

CLASSIFICATION OF CELLULAR IMAGE

This article discussed an effective algorithm for cell segmentation and showed its integration that supports decision-making in clinical pathology. The nonparametric nature of the segmentation and its robustness to noise allowed the use of a fixed resolution for the processing of hundreds of digital specimens captured under different conditions. The classification has been indirectly evaluated which demonstrated satisfactory overall performance. As a broader conclusion, however, this research proved that the segmentation, although a very difficult task in its general form.

With an aim of testing the robustness of our algorithm, we applied it to a cellular image. This image is coded on 8 bits (Figure 7). Each pixel of the image is represented by its three dimensions components RGB. The construction of the training file is carried out in the following way: The first stage consists in specifying the number of color present in the image. Then, for each weights color, one selects, using the mouse, a small area comprising of the pixels having similar points of colors. The last stage consists in giving labels to each color introduced into the training file. In order to have a good illustration of the projection of these observations on a of Kohonen map, we used a map of size 200 neurons, and at the time of the phase of training, the observations are presented sequentially one by one at the entry of the random network.

The images which we have to process are thus color images. On these images cells are present; these must be extracted.

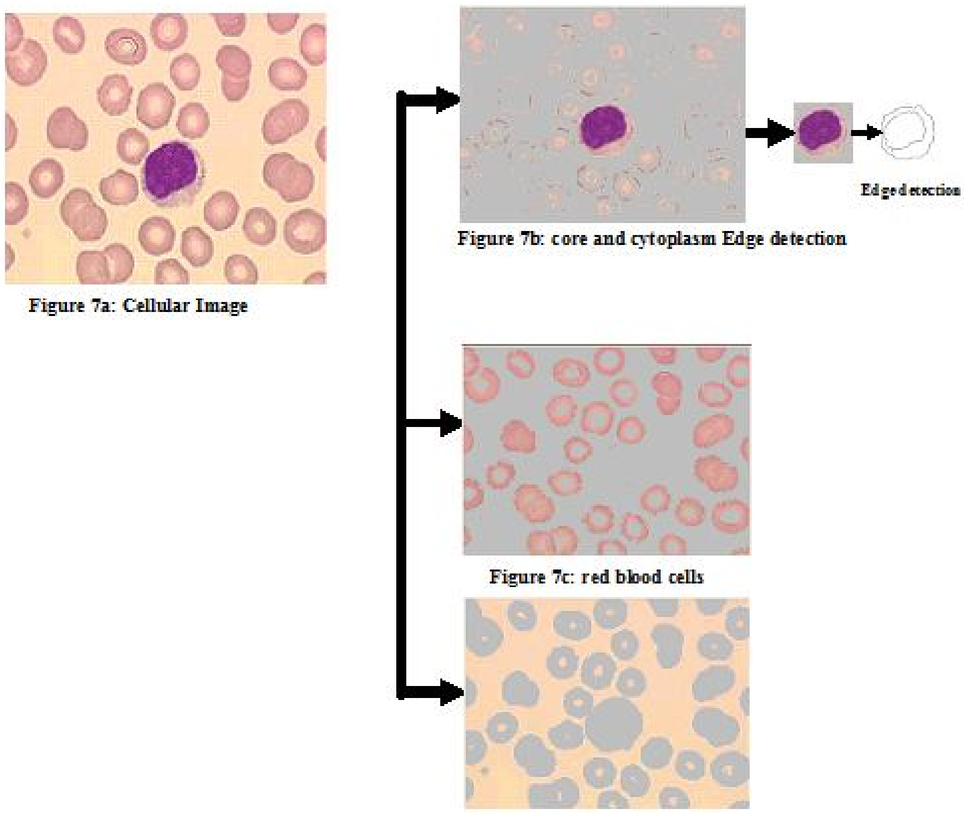

Figure 7a: Cellular Image

Figure 7b: core and cytoplasm Edge detection

Figure 7c: red blood cells

Figure 7d: white blood cells

Figure 7: Classification of cellular image

We should insulate at the same time their cytoplasm and their core. Two bits of information must bring to recognition the various cellular types. Before starting the first stage of segmentation, we must precisely know the nature and the context of the images. Our images are color images presenting the cells coming from cytology from cereuses and colored by the international coloring standard. The cells have a brown core and a yellow cytoplasm except for the red blood cells. The strategy of segmentation we adopt is an ascending strategy. We extract the core and cytoplasm of the cells at the same time, and then we extract the red blood cells [11].

Segmentation is realised using colors. Classification is achieved with a supervised self-organizing LVQ1 and by using the operators of edge detection (canny, sobel) to finalize the work.

CONCLUSION

Few traditional neural network algorithms have been meant to directly operate on raw data such as pixels of an image or samples of speech waveforms picked up from the time domain. Most pattern recognition tasks are preceded by a pre-processing transformation that extracts invariant features from the raw data such as spectral components of acoustical signals or elements of co-occurrence matrices of pixels. Selection of a proper pre-processing transformation for a particular task usually requires careful consideration and no general rules can be given here. It is cautioned that if this LVQ is used for benchmarking against other methods a proper pre-processing should always be used. In performing statistical experiments a separate data set for training and another separate data set for testing must be used. If the number of required learning steps is bigger than the number of training samples available, the samples must be used reiteratively in training either in a cyclical or in a randomly sampled order.

In this work, we proposed automatic approach of classification of the colored images, based on the association of a Kohonen map to an algorithm of supervised training LVQ. This approach consists at first step to represent the samples of observations representative of the pixels of an image in space 3d of the colors components. The next step is the training phase, the weights vectors corresponding to the extracted modal areas taken as prototypes of the classes present in the image, and are used for the assignment of each pixel of the image to one of the classes identified at the time of the phase of classification. This approach shows, that in a supervised context, the Kohonen map allows a good automatic classification of the color image.

REFERENCES

[1] P. Scheunders. A comparison of clustering algorithms applied to color image quantisation, Vision Lab, Dept. of Physics, RUCA University of Antwerp,

[2] Kohonen T. *Self-Organizing Maps*, Springer Series in Information Sciences, 30, Springer-Verlag, New York, 1995.

[3] Kotropoulos C., E. Augé and I. Pitas. Two-layer learning vector quantizer for color image quantisation. In: Eds., *Signal Processing IV: Theories and Applications*, Elsevier, 1177-1180, 1992.

[4] N. Ebi, B. Lauterbach, and W. Anheier, An image analysis system for automatic data acquisition from colored scanned maps, Machine Vision and Application, pp 148-164, 1994.

[5] Dekker A.H. Kohonen neural networks for optimal color quantisation. *Network: Computation in Neural Systems*, 5 351-367, 1994.

[6] S. J. Wan, P Prusinkewicz, and S. K. M. Wong. Variance-based color image quantisation for frame buffer display. Color Research and Application, February 1990

[7] Y. Linde, A. Buzo, and R. M. Gray. An algorithm for vector quantizer design. IEEE Transactions on Communication, January 1980.

[8] Teuvo Kohonen LVQ PAK The Learning Vector Quantisation Helsinki University of Technology Laboratory of Computer and Information Science Rakentajanaukio 2 C, SF-02150, 1995.

[9] Abdelhafid Bessaid, Hassane Bechar, M. Karim Fellah, “Image analysis and pattern recognition as tools in map interpretation”, Electronic Journal «Technical Acoustics», *published ,2003*.

[10] Richard P. Lippman, *An Intoduction to Computing with Neural Nets*, IEEE ASSP Magazine April 1987.

[11] O. Lezoray, A. Elmoataz, H. Cardot and M. Revenu, A.R.C.T.I.C, Un système automatique de Tri Cellulaire par Analyse d’Images. Laboratoire Universitaire des Sciences Appliquées de Cherbourg E.I.C, Octeville, France, 1999 ;