Eurographics Italian Chapter Conference (2006)
G. Gallo and S. Battiato and F. Stanco (Editors)

Color Palette Images Re-indexing by Self Organizing Motor Maps

S. Battiato[1], F. Rundo[2], F. Stanco[1]

[1]Dipartimento di Matematica e Informatica, University of Catania
Viale A. Doria, 6 - 95125 Catania, Italy
{battiato, fstanco} @ dmi.unict.it
[2]Software Competence Center of Imaging Division
STMicroelectronics of Catania
Stradale Primosole, 50 - 95121 Catania, Italy
francesco.rundo@st.com

Abstract

Palette re-ordering is a well known and very effective approach for improving the compression of color-indexed images. If the spatial distribution of the indexes in the image is smooth, greater compression ratios may be obtained. As known, obtaining an optimal re-indexing scheme is not a simple problem. In this paper we provide a novel algorithm for palette re-ordering problem showing the advantages of using a neural network instead of classical heuristic methods. We propose to apply the Motor Map neural network which is considered an extension of the well-known SOM Kohonen neural network.
Experiments confirm the effectiveness of the proposed technique.

Categories and Subject Descriptors (according to ACM CCS): I.4.2 [Image processing and Computer Vision]: Compression (Coding)

1. Introduction

Indexed images encode colors using a fixed look up table or *palette* where each entry is a triplet of RGB values. For each pixel in the image only the index of the corresponding color needs to be stored. The efficiency of a compression algorithm for indexed images depends on the assignment of indexes in the relative look up table. In particular, a palette which assigns consecutive indexes to colors sharing many adjacent pixels in the image will provide better compression ratios. Since the number of possible color indexing is $M!$ for an image with M colors, the methods to find the optimal ordering is intrinsically difficult (NP-hard). In literature, different algorithms have been proposed to address this issue. They search for a specific correlation between the pixels in the images proposing heuristic solutions. A survey describing almost all of them is [PN04]. A subsequent section will provide some brief details about the main strategy underlying the most effective algorithms. We have proposed a reindexing technique in [BGIS04] where the entropy was reduced by using an approximation of the Travelling Salesman Problem. In this paper we propose a method to solve the reindexing problem by means of Motor Maps neural network [MRS92] which basically represent a specific behavior of human brain which can be suitable to solve many kinds of complex problems. We tested our algorithm using a subset of synthetic images used by Pinho et al [PN04] with different size and number of colors. Some of them have been used to evaluate the performance of the Motor Map neural network as Optimum Palette generator. Experimental results show that the bit per pixel (bpp) is reduced sensibly using our approach. Moreover, the results are better than other algorithms known in literature as well as the results showed on [PN04].

The paper is structured as follows. Section 2 introduces the re-indexing problem, while Section 3 reviews some related works. Section 4 describes the Motor Maps theory. Our technique is described in Section 5, while experimental results are presented in Section 6. Conclusions are drawn in Section 7.

2. The Palette re-indexing problem

The re-indexing problem can be stated as follows [BGIS04]. Let I be an image of $m \times n$ pixels, and M be the number of distinct colors. I can be represented as $I(x,y) = P(I'(x,y))$, where $P = \{S_1, S_2, \dots, S_M\}$ is the set of all the colors in I, and I' is a $m \times n$ matrix of indexes in $\{1, 2, \dots, M\}$. An image represented in such a fashion is called *indexed image* and P is its *palette*. Typical values for M are 16, 64 or 256.

Most of the compression engines proceed by scanning in some sequential order the indexes in I'. Once an ordered scan has been performed the pixels encountered may be named $p_1, \dots, p_{m \times n}$. If a differential approach to coding and compression is adopted the information needed to reconstruct the original image is:

i) the colour of pixel p_1 ;
ii) a table providing the correspondence between colours $S_1, S_2, \dots, S_M$ with index $i_1, i_2, \dots, i_M$;
iii) the sequence of differences:
$d_h = |(index\,of\,colour\,in\,pixel\,h+1) - (index\,of\,colour\,in\,pixel\,h)|$.

Let $D(I')$ be the set of all differences d_j with $j = 1, 2, \dots, (n \times m) - 1$. Information theory states that any lossless scheme to encode the set of differences $D(I')$ requires a number of bits per pixel (bpp) greater or equal to the zero-order entropy of the statistical distribution of $D(I')$. The related entropy of the sequence of differences is one of the main parameters that guides the optimization process as described in the next Section. We claim that alternative sequences processing of the local differences could give improvement.

If indexes $i_1, i_2, \dots, i_M$ are ordered so as to produce an almost uniform distribution of values d_h the entropy value will be large. Conversely, a zero-peaked distribution in $D(I')$ gives a lower entropy value. Hence, finding an optimal indexing scheme is a crucial step for differential lossless compression of indexed images.

3. Related works

The existing solutions to the re-indexing problem may be classified into two main groups, according to the particular model/strategy adopted. The first group of solutions performs the re-indexing of color indexes according to perceptive similarity between different colors. In [ZL93, SM01, PT94, HS94] consecutive symbols are assigned to visually similar colors. Several perceptual similarity measures can be adopted: "closest pairs ordering" [PT94] with the aim of assigning close indexes to colors that are close in three-dimensional (3-D) color space; combinatorial optimization [HS94] which aims at finding minimum (or maximum) values of a cost (or objective) function, usually nonlinear and of many independent variables; distances between colors in 3-D color space [SM01]. Although several different measures are proposed in this group, the most widely used is the reordering luminance based [ZL93]. The second group of re-indexing algorithms is guided by both information theory and local adaptive considerations. Memon investigated the problem of ordering the palette with respect to the compression ratio obtained with a suite of different compression algorithms [MV96]. The bottleneck of this group of solutions is the relative inefficiency of running a simulated annealing algorithm to optimize the palette re-indexing. To overcome this problem a *Pairwise Merge* (PM) heuristic has been proposed in [MV96]. Some techniques specifically devoted to work on a bit plane basis are presented in [FV98] and [Gor95], while [WR94] introduces a color correlation sorting algorithm. The most widely known representative in this class is the solution proposed by Zeng et alii [ZLL00]. This technique is based on a greedy algorithm to maximize a suitable potential function. The potential function has been heuristically selected in such a way that large values correspond to more peaked distributions of the set $D(I')$. A modified version of this algorithm is proposed in [PN04], where the potential function in optimized to improve the speed of the method. In [BGIS04] the re-indexing problem is translated into an optimization problem over a weighted graph and solved in an approximate fashion.

4. The Motor Maps theory

The brain cannot limit itself, however, to the representation of sensory input signals alone, but must also solve the complementary task of sending appropriate signals to the muscle system to react to the sensory input. The brain regions that are responsible for these tasks, such as the motor cortex and the superior colliculus, appear in many cases to be organized in a way similar to the sensory areas, i.e., as maps that react to localized excitation by triggering a movement. This movement varies in a regular way with the focus of the excitation in the layer. Therefore, the layer can be considered as a motor map in which movement commands are mapped to two-dimensional locations of excitation [MRS92]. In this work as abstraction that can also serve as a model for such motor maps, we consider the Artificial Motor Map SOM as extension of Kohonen's original Self Organizing Maps (SOM) in which an output nerual layer will be added. The importance of topology-preserving maps in the brain relies on both the representation of sensory input signals and the ability to perform an action in response to a given stimulus. Neurons in the brain are organized in local assemblies which are able to perform a given task such as sending appropriate signals to muscles. These neural assemblies constitute two-dimensional layers in which the locations of the excitation are mapped into movements. Topology-preserving structures are able to classify input signals inspired by the paradigm of Kohonen's Networks [MRS92]. These artificial neural networks formalize the self-organizing process in which a topographic map is created. Neighboring neurons are thus excited by similar inputs. Successful applications of these maps have been found

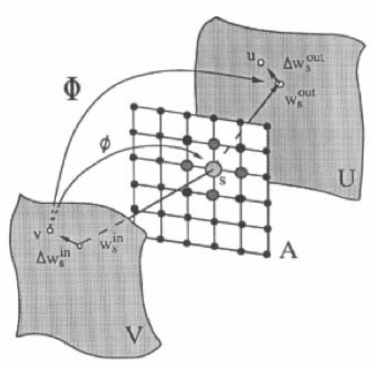

Figure 1: *The extended Kohonen'SOM model with the inclusion of output values is showed. Each formal neuron s of the neuron layer (lattice A) has, in addition to its pre-existing weight vector w_s^{in} , a vector w_s^{out} of output values assigned to it. A learning step requires, for each presentation of an input vector v, the specification of a corresponding output value u. The adaptation of the output values w_s^{out} is completely analogous to the scheme used for the "input side": all neurons in the neighborhood of the selected neuron by the input value shift their output vectors towards the specified output value u.*

in the field of pattern generation, chaos control, clustering and so on [PLM02, PLMG04, Koh72]. An extension of these neural structures is represented by motor maps. These are networks able to react to localized excitation by triggering a movement (like the motor cortex or the superior colliculus in the brain). To do this, motor maps, unlike Kohonen's networks, should include storage of an output specific to each neuron site. This is achieved by considering two layers: one devoted to the storage of input weights and one devoted to output weights. The plastic characteristics of the input layer should also be preserved in the assignment of output values, so the learning phase deals with updating both the input and the output weights. These considerations led to the idea of using Motor Maps as adaptive self-organizing controllers. Formally, a Motor Map can be defined as an array of neurons mapping the space V of the input patterns onto the space U of the output actions:

$$\Phi : V \rightarrow U \tag{1}$$

A schematic representation of a motor map is given in Fig. 1. The learning algorithm is the key to obtain a spatial arrangement of both the input and output weight values of the map. This is achieved by considering an extension of the winner-take-all algorithm. At each learning step, when a pattern is given as input, the winner neuron is identified: this is the neuron that best matches the input pattern. Then, a neighborhood of the winner neuron is considered and an update involving both the input and output weights for neurons belonging to this neighborhood is performed. Even though both supervised and unsupervised learning can be applied, only unsupervised learning should be considered if an autonomous self-organizing system for optimum palette scheme generator has to be defined. In this case, there is no *a priori* information on the appropriate optimum palette scheme and no "teacher" is available. The algorithm has to find the correct palette re-indexing by itself. The only source of information is provided by the so-called reward function, introduced below, which indicates how well the palette scheme generated is being performed. Weight updating takes place only if the corresponding palette re-indexing leads to an improvement in terms of zero-order entropy of local differences in the image processed; otherwise, the neuron weights are not updated. In this framework a fundamental role is taken by the reward function. The definition of this function is perhaps the most crucial point in the whole network design.

4.1. Unsupervised learning for Motor Maps

The unsupervised learning algorithm for the motor map can be described in the following five steps.
Step 1. The topology of the network is established. The number of neurons needed for a given task is chosen by a trial-and-error strategy, thus once numerical results indicate that the number of neurons is too low, one must return to this step modifing the map dimensions. At this step the map's weights are randomly fixed.
Step 2. An input pattern is presented and the neuron whose input weight best matches it is established as the winner. Therefore, to establish the winner neuron, the distance between the neuron input weight and the input pattern is computed for each neuron, considering the absolute value of the difference between these two vectors.
Step 3. Once the winner neuron has been chosen, its output weight is used to perform the local re-indexing of the palette image . Typically, this is not used directly, but a random variable is added to the value to guarantee a random search for possible solutions, as follows:

$$f(t) = w_{winner,out} + a_s\lambda \tag{2}$$

where $w_{winner,out}$ is the output weight of the winner neuron, a_s is a parameter determining the mean value of the search step for the neuron s, and λ is a Gaussian random variable with zero mean. Then the increase ΔR in the reward function is computed and, if this value exceeds the average increase b_s gained at the neuron s, the next step (updating of the neuron weights) is performed; otherwise this step is skipped. The average increase in the reward function is updated as follows:

$$b_s^{new} = b_s^{old} + \rho(\Delta R - b_s^{old}) \tag{3}$$

where ρ is a positive value. Moreover, a_s is decreased as more experience is gained (this holds for the winner neuron

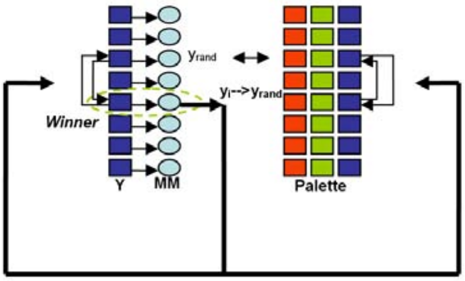

Figure 2: *New palette scheme provided by Motor Map.*

and for the neighboring neurons), according to the following rule:

$$a_i^{new} = a_i^{old} + \eta_a \xi_a (a - a_i^{old}) \quad (4)$$

where i indicates the generic neuron to be updated (the winner and its neighbors), a is a threshold the search step should converge to, η_a is the learning rate, while ξ_a takes into account the fact that the parameters of the neurons to be updated are varied by different amounts, defining the extent and the shape of the neighborhood.
Step 4. If $\Delta R \geq b_s$ the weights of the winner neuron and those of its neighbors are updated following the rule:

$$\begin{aligned} w_{i,in}(t+1) &= w_{i,in}(t) + \eta(t)\xi(t)(v(t) - w_{i,in}(t)) \quad (5) \\ w_{i,out}(t+1) &= w_{i,out}(t) + \eta(t)\xi(t)(f(t) - w_{i,out}(t)) \quad (6) \end{aligned}$$

where $\eta(t)$ is the learning rate, $\xi(t)$, $v(t)$, $w_{i,in}$, and $w_{i,out}$ are the neighborhood function, the input pattern, the input weights and the output weights, respectively, and the index i takes into account the neighborhood of the winner neuron. In *supervised learning*, $f(t)$ is the target, while in *unsupervised learning* it is varied, as discussed above.
Step 5. Steps 2)-4) are repeated. If one wishes to preserve a residual plasticity for a later re-adaptation, by choosing $a \neq 0$ in step 3), the learning is always active and so steps 2)-4) are always repeated. Otherwise, by setting $a = 0$, the learning phase stops when the weights converge.

5. The Re-indexing algorithm provided by Motor Maps

The idea proposed in this paper is based on the ability of the Motor Map Neural Network to learn the "features" of the input pattern (a still image in this case) and providing an appropriate output stimulus. At this point, given a color-indexed image, the question is: *Is there a specific mathematical correlation between the optimum color indexing scheme of the palette and the relative color shape?* The only way to give an answer to this question is to use a Motor Map which provides, during the learning process, a palette shape clustering for searching (in the output stage of the network) the optimum indexing scheme. The learning process can be described as follow.
Step.1 Let Y the luminance vector computed starting from the palette P of the image I. In the case of RGB color space, the luminance can be approximated by the lightness factor computed for each color $S_i(r_i, g_i, b_i)$ by using the well known expression:

$$y_i = 0.299 r_i + 0.587 g_i + 0.114 b_i \quad i = 1, 2, \ldots, M \quad (7)$$

where M is the number of palette colors.

The first step in the Motor Map initialization is to define the number of neurons to be used in the network which can be easily fixed equal to the number of elements of the vector Y. Let N be a number of neurons of the Motor Map. Each neuron will be composed by an input weight $w_{i,input}$ and output weight $w_{i,output}$ and a field which store the average increasing of the reward function b_i. The variable range of the $w_{i,output}$ values is $[1, \ldots, M]$. A crucial parameter to be carefully chosen is the reward function. In our case, the following reward function has been chosen:

$$Reward = -(Entropy)^2 \quad (8)$$

In a few words, the selection of the above reward function leads the Motor Map to find an optimum palette index scheme which minimize the entropy of the image and then the related compression ratio. Regarding the output stimulus $f(t)$ produced by the Motor Map during the learning phase, for sake of simplicity, has been forced equal to $w_{i,output}$. The $w_{i,output}$ will be equal a random index generated during the learning process when the corresponding neuron wins. Before to start the learning phase, the Motor Map (both input layer and output layer) will be initialized randomly.
Step.2 The vector Y will be presented to the input layer of the Motor Map searching the winner neuron i.e., the neuron which has the minimum value of the following distance:

$$d_i = |y_i - w_{i,input}| \quad i = 1, 2, \ldots, M \quad (9)$$

At this point, the winner neuron provides an updating of the output stimulus *f(t)* which is, in this case, a new index y_{rand} (for the "winner" luminance) on the luminance vector i.e. a new index for the related color on the corresponding palette and finally the related swaps on the image matrix. The new index is generated randomly according to the upper bound fixed by the number of colors (or indexes).

The zero-order entropy of the new palette image will be computed (to be precise, the zero-order entropy of the set of differences $D(I'_{new})$) and then the $\Delta Reward$:

$$\begin{aligned} \Delta Reward &= Reward^{new} - Reward^{old} \quad (10) \\ &= -(Entropy^{new})^2 + (Entropy^{old})^2 \end{aligned}$$

Step.3 The average increasing of the reward function is weighted by the b_{winner}:

$$b_{winner}^{new} = b_{winner}^{old} + \rho(\Delta Reward - b_{winner}^{old}) \qquad (11)$$

Step.4 If the $\Delta Reward \geq b_{winner}$ the new index scheme will be accepted and then the next learning step (steps from 1 to 4) will be repeated on the new image. Conversely, the new index scheme will be rejected and the previous ones will be restored (Fig. 2).

In the first architecture of the Motor Map proposed in this work, the neuron has not an adaptive neighboring and the learning rate remains constant during all the learning phase. Moreover, a particular modification on the learning process has been added. By taking into account the neuro-biology similarity between the Motor Map and motor cortex on the human brain, the concept of "neural perturbation" has been implemented artificially. When the human brain tries to solve a problem (the mathematical law is often unknown) in such a case provide an action which show a significantly perturbation on the current heuristic set of actions being applied in order to solve the problem. To think of problem to balance a vertical pen placed on centre of the palm of the human hand. The human brain does not know the mathematical motion equations but it tries to solve the problem with heuristic hand movements based on the input signal provided by eyes. When the slope of the pen is large or when the equilibrium is very unstable, the brain can try to search the solution after a perturbation of the neurons involved in the problem (which shows some perturbation movements). In the same manner, when the Motor Map realizes that the current entropy minimizing process is motionless (likely a local minima), provides a perturbation of the neurons and it continues on the learning process according to the following expression:

$$w_{i,input} = w_{i,input} + \lambda_{Y_i} \qquad (12)$$

where λ_{Y_i} is a gaussian random variable normalized with upper bound equal to maximum value of the luminance associated to the image to be re-indexed. The stop of the learning process of the Motor Map can be reached when the entropy computed is less or equal to a specific lower bound value.

6. Experimental Results

In order to check the performance of the Motor Map as palette re-indexing algorithm, we propose the comparison between our method and the most important reordering methods described in Section 3. In particular, we compare our method (called MMap) with classical Luminance re-ordering; with Memon [MV96] technique; with modified Zeng algorithm proposed by [PN04]; and with Battiato et alii. algorithm [BGIS04]. For these comparisons, we use some images that have been used by Pinho at al. in [PN04]. The software implementation has been realized in MATLAB 7.0.1 using "Image Processing" toolbox. The average computational time in the learning process of the Motor Map is about 900 (time unit is the second). Such timing results is only a preliminar estimation of the overall computation time.

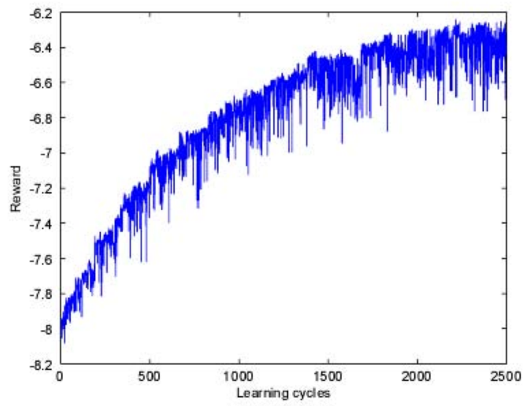

Figure 3: *Reward function values with respect to the learning cycles.*

Table 1 shows the bits per pixel obtained with JPEG-LS. The values relative to our approach are the lower than the others. This is independent from the number of colors in the images. We claim that this trend will remain unchanged even if the number of colors are increased. This proof is part of our future works.

Fig. 3 shows a plot of the reward function value used in our network with respect to the learning cycles. Fig. 4 reports the indices matrix before and after the palette reordering. In particular, the indices in Fig. 4(b) are clearly smoother than the indices in Fig. 4(a). This is a visual confirmation that the MMap approach works in the right direction.

7. Conclusion and future work

Palette reordering is a very effective approach for improving the compression of color-indexed images. In this paper, we described a technique that shows a good performance on the optimum palette scheme generation without any initial hypothesis on the palette index scheme or on the pixel distribution. In fact, it is interesting to note that a lot of palette re-indexing algorithm proposed in literature are based on the assumption that the differences of neighboring pixels of well-reordered images should follow a Laplacian distribution. This is in accordance with the JPEG-LS image coding standard, which also assumes a Laplacian model for the prediction residuals and, therefore, may provide a justification for the good performance of both methods. The Motor

Images	**Colors**	**Luminance**	**Memon**	**mZeng**	**Battiato**	**MMap**
Gate	84	2.930	2.548	2.566	3.116	2.339
Benjerry	48	1.423	1.133	1.137	1.186	1.114
Netscape	32	1.918	1.745	1.752	1.907	1.05

Table 1: *Lossless compression results in bit per pixel, obtained with JPEG-LS applied to the indexed images after using the palette reordering methods presented in the paper.*

(a) (b)

Figure 4: *(a) The positive of the indices of the input image; (b) the positive of the indices after the MMap palette reordering.*

Map algorithm does not need any hypothesis on the initial image's pixel distribution as well as on the re-ordered final image. The algorithm proposed is only a first preliminary realization of the Motor Map used as optimum palette indexes scheme generator. Future research aims at finding a better realization of the algorithm defining in a suitable way, the neighboring interaction between the neurons as well as an adaptive updating of the learning rate. Also the possibility to work on the profile color space CIE XYZ, where Y is the real luminance factor, will be considered.

References

[BGIS04] BATTIATO S., GALLO G., IMPOCO G., STANCO F.: An efficient re-indexing algorithm for color-mapped images. *IEEE Transactions on Image Processing 13*, 11 (November 2004), 1419–1423.

[FV98] FOJTÍK J., VÁCLAV H.: Invisible modification of the palette colour image enhancing lossless compression. In *Joint IAPR Workshops SSPR98* (1998), no. LNCS Vol. 1451, pp. 1029–1036.

[Gor95] GORMISH M. J.: Compression of palletized images by colour. In *IEEE Int. Conf. on Image Processing* (1995).

[HS94] HADENFELDT A. C., SAYOOD K.: Compression of color-mapped images. *IEEE Trans. Geosci. Remote Sens. 32* (May 1994), 534–541.

[Koh72] KOHONEN T.: Self organized formation of topologically correct feature maps. *Bio.Cybern. 43* (1972).

[MRS92] MARTINETZ T., RITTER H., SCHULTEN K.: *Neural Computing and Self-Organizing Maps*. Reading, MA: Addison-Wesley, 1992.

[MV96] MEMON N., VENKATESWARAN A.: On ordering colour maps for lossless predictive coding. *IEEE Trans. Image Proc. 5*, 11 (November 1996), 1522–1527.

[PLM02] P.ARENA, L.FORTUNA, M.FRASCA: Chaos control by using motor maps. *Chaos 12*, 3 (2002).

[PLMG04] P.ARENA, L.FORTUNA, M.FRASCA, G.SICURELLA: An adaptive self organizing dynamical system for hierarchical control of bio-inspired locomotion.

[PN04] PINHO A., NEVES A.: A survey on palette reordering methods for improving the compression of color-indexed images. *IEEE Transactions on Image Processing 13*, 11 (November 2004).

[PT94] PO L. M., TAN W. T.: Block address predictive color quantization image compression. *Electron. Lett. 30*, 2 (January 1994), 120–121.

[SM01] SPIRA A., MALAH D.: Improved lossless compression of color-mapped images by an approximate solution of the traveling salesman problem. In *IEEE Int. Conf. Acoustics, Speech, Signal Processing* (May 2001), no. III, pp. 1797–1800.

[WR94] WALDEMAR P., RAMSTAD T. A.: Subband coding of color images with limited palette size. In *IEEE Int. Conf. on Acoustics, Speech, and Signal Processing ICASSP-94* (April 1994), no. V, pp. 353–356.

[ZL93] ZACCARIN A., LIU B.: A novel approach for coding color quantized images. *IEEE Trans. Image Processing 2* (1993), 442–453.

[ZLL00] ZENG W., LI J., LEI S.: An efficient colour re-indexing scheme for palette-based compression.