Self Organizing Motor Maps for Color-Mapped Image Re-Indexing

Sebastiano Battiato, *Senior Member, IEEE*, Francesco Rundo, and Filippo Stanco, *Member, IEEE*

***Abstract*—Palette re-ordering is an 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 is already known, obtaining an optimal re-indexing scheme is not a trivial task. In this paper, we provide a novel algorithm for palette re-ordering problem making use of a motor map neural network. Experimental results show the real effectiveness of the proposed method both in terms of compression ratio and zero-order entropy of local differences. Also, its computational complexity is competitive with previous works in the field.**

***Index Terms*—Indexed images, motor map (MM).**

I. Introduction

COLOR-MAPPED images make use of an index map to store the different involved colors maintaining for each pixel the location of the corresponding index. Local index redundancy can be considered to improve the compression performances of any compression technique. Re-indexing techniques try to find the optimal reordering avoiding to consider all possible color indexing ($M!$ for an image with M colors).

The existing re-indexing algorithms may be classified into two main groups: color and index based. These approaches are devoted to obtain respectively color and index similarity.

In color-based solutions [1]–[4], consecutive symbols are assigned to visually similar colors according to some heuristic measures. Typically, color mapped images present an indexes map sorted by luminance order [1]. Alternatively, index-based methods are guided by both information theory and local adaptive considerations [5]–[8]. The bottleneck of this group of solutions is the intrinsic inefficiency to numerically optimize the palette re-indexing. To overcome this problem, different heuristics have been proposed [9]–[11]. A survey describing almost all methods with related details and experimental results is available in [11].

Unfortunately, a common public repository for re-indexed images obtained by collecting results of each method is not available. Moreover, experimental results are often reported in terms of bits per pixel (bpp) by using standard compression encoding. An effective re-indexing scheme should be able to reduce the residual entropy of local differences by considering also different configuration patterns. The overall performance of each method should be also based on numerical comparisons of such entropy values.

Recently, further improvements have been obtained properly preprocessing the indexed input image with respect to the available encoder [12], [13]. The final image is obtained not only by decoding, but making use of the corresponding postprocessing phase. In some sense, such methods provide a much more sophisticated "coded" representation and, therefore, cannot be classified as methods for re-indexing.

Some authors have recently pointed out that, by using advanced quantization strategy during the original generation of color-mapped images, it is possible to produce sub-optimal indexing scheme. In particular, in [14], a self-organizing Kohonen feature map, is used to properly guide the color reduction. The obtained results are in some cases very effective but the method requires, of course, the availability of the original true-color images.

In this paper, we propose a method to solve the re-indexing problem by means of a motor map (MM) neural network [15]. The plasticity as well as the self-organization property of the MM allows it to be applied in different scientific fields such as nonlinear dynamics control [16] and bio-inspired robot control [17]. The main advantage of the MM is related to its learning algorithm [15] which is unsupervised, application independent, highly adaptable due to self-organizing features. The learning algorithm, which is an extension of the well-known "winner-take-all" (WTA) learning is driven by the so called "reward" function which no longer directly specifies the optimum palette indexing but only indicates "how well" the current palette indexing complies with the target to be reached (i.e., minimize the zero-order entropy of the original color image). The ability of MMs to find an optimal solution without requiring the knowledge of the underlying model has been crucial in this context. The overall performances have been evaluated by considering the same repository used in [11] that contains synthetic and natural images with different size and number of colors. Experimental results show effective results if compared with previous results in the field. It represents a good tradeoff between overall performances and computational complexity.

The paper is structured as follows. Section II introduces the re-indexing problem, while MM reindexing technique is presented in Section III. Experimental results are presented in Section IV. Conclusions are drawn in Section V.

II. Problem Formulation

The re-indexing problem can be stated as expressed in [10] and [11]. 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 $P = \{S_1, S_2, \ldots, S_M\}$ is the set of all the colors in I, and I' is a $m \times n$ matrix of indexes in $\{1, 2, \ldots, M\}$

Manuscript received December 21, 2006; revised September 11, 2007. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Bruno Carpentieri.

The authors are with the Dipartimento di Matematica e Informatica, University of Catania, 95125 Catania, Italy (e-mail: battiato@dmi.unict.it; rundo@dmi.unict.it; fstanco@dmi.unict.it).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2007.909415

with M typically equal to 16, 64, 256, or 512. An image represented in such a fashion is called *indexed image* (or *color mapped image*) and P is its *palette*.

Most of the compression engines proceed by coding the data attacking the local spatial redundancy. For indexed images, an ordered scan of the indexes in I' named $p_1, \ldots, p_{m \times n}$ is usually performed. The residual entropy of local differences can be considered to estimate the overall "energy" of the signal. The information needed to reconstruct the original image is:

i) the color of pixel p_1;

ii) a table providing the correspondence between colors $S_1, S_2, \ldots, S_M$ with index $i_1, i_2, \ldots, i_M$;

iii) the set of differences

$$D(I') = \{d_{x,y} | x = 1, 2, \ldots, m \; y = 1, 2, \ldots, n\}$$

where each $d_{x,y}$ is a local differencee obtained by considering some specific patterns as better specified as follows.

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 entropy of the sequence of differences is one of the main parameters that guides the proposed optimization process as described in the Section III-B.

If indexes $i_1, i_2, \ldots, i_M$ are properly ordered to produce an almost uniform distribution of values $d_{x,y}$ 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 any lossless compression of indexed images.

III. MM Re-Indexing Algorithm

A. Theory

MM extend Kohonen's self organizing maps (SOMs) [18] by adding a neural output layer. The key idea in the MM is to represent artificially some typical features of the cerebral cortex placed in the human brain. The human cerebral cortex is 2–4 mm (0.08–0.16 in) thick. It has a crucial role in many complex brain functions including memory, attention, perceptual awareness, language and so on. The "topology-preserving" feature as well as the "self organization" ability of the human cerebral cortex are suitable to leads our brain to manage appropriately the input signals detected by human sensory such as eyes or ears, producing, if necessary, an appropriate output "stimulus." Most of the sensory stimulation arrives at the cerebral cortex indirectly through different thalamic nuclei. The areas in the human brain that receive such information are called sensory areas. There is also another area called primary sensory area. Vision, audition, and touch are managed respectively by the primary visual cortex, primary auditory cortex and primary somato-sensory cortex. The classical Kohonen SOM represents artificially the features of the sensory areas placed within the cerebral cortex. The Kohonen SOM tries to map in a typical 2-D lattice structure the input patterns (which represents the input signal captured by human sensory). In the cerebral cortex of the human brain, there is also a motor areas (or motor cortex) located in hemispheres of the cortex. These areas are responsible to the control of voluntary movements of the human body such as a fine fragmented movements performed by the hand, the moving of the legs, etc. The motor areas are organized in a way similar to the sensory areas (i.e., as maps that react to

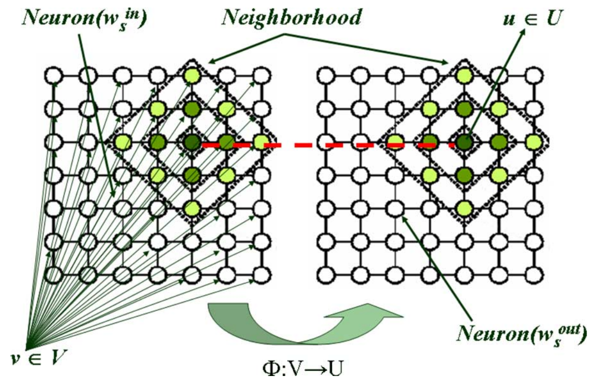

Fig. 1. Schematic representation of a typical MM.

localized excitation by triggering a stimulus through the motor neurons). The sensory inputs mapped in the sensory areas of the human cerebral cortex are used as "excitations" of the motor areas which is able to solve the complementary task of sending appropriate signals to the muscle system, by means of motor neurons, to react to the sensory input. The produced output stimulus (tipically a movement) varies depending on the location in the motor areas where the excitation happens. By taking into account what previously mentioned, the motor areas in the human cerebral cortex can be considered as a MM which provide a reaction depending on sensory input mapped in the sensory cortex. The human brain uses the sensory areas and the motor areas to solve complex real problems without specific knowledge about the specific involved details (i.e., mathematic formulation of the underlying problem) by means of a learning process which involves both sensory neurons and motor neurons. Clearly, within both sensory and motor cortex, the neurons are arranged in a lattice structure in which its neurons are interconnected with its neighborhood by means of synaptic contacts.

In the MM neural networks, an algorithm able to solve complex problem by means of a learning process is reproduced artificially. Its structure includes the capability both of the sensory area and the motor area of the human cerebral cortex. From a mathematical point of view, an artificial MM can be defined as an application

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

where V is a space of the input signals and U represents the space of the output stimulus generated by the MM. The mapping between each element of the space V with an element of the space U occurs through the synaptic weights (which represent the real synaptic contact placed in the human neural cortex) as showed in Fig. 1, where a schematic representation of a typical MM is presented. The term w_s^{in} represents the input weight whereas the term w_s^{out} represents the output weight. All the weights are arranged in a MM with lattice structure s. The term v represents an input pattern, while the term u is the output reaction triggered by the MM according to the information learned from the input pattern being presented to the network. In analogy with the human brain, the artificial MM needs a target to be reached typically defined as an optimization process where a function is minimized by making use of the involved learning of the whole network. The learning process both for sensory layer (input layer) and motor layer (output

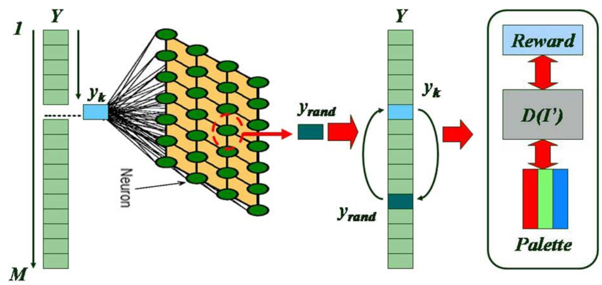

Fig. 2. New palette scheme provided by MM.

layer) of the artificial MM is the well-known WTA algorithm typically used for training Kohonen SOM [15]. According to the learning process, the functional used for describing the problem to be solved is usually called "reward," just to highlight that the winner neurons which provide a reaction to improve the progressive solving of the problem, will be "rewarded." Further details are properly presented in [16] and [17], where MM have been successfully applied in the field of bio-inspired control system, chaos control, etc. In our case, the unsupervised learning updates input and output weights of the neighborhood without *a priori* information about the optimum palette scheme. A crucial point in the whole network design is the definition of the so-called reward function useful to drive the learning process. As better specified in the following, we have used the zero-order entropy of the local index differences of the input image.

The MM unsupervised learning can be characterized by specifying the following network parameters: network topology, input/output initial weights, reward function, and comparison metric. Also, some further details about the smoothing action, the learning rate, the output stimulus and the neighborhood's shape have to be properly defined. Major details about MM network design and related applications can be found in [16] and [17].

B. MM for Re-Indexing

The proposed algorithm is based on the ability of the MM neural network to learn the "features" of the input pattern (a lightness factor of a still image, in this case) providing an appropriate output stimulus. We propose to use a MM which provides, during the learning process, a palette shape clustering for searching (in the output stage of the network) the optimum indexing scheme. The WTA algorithm allows the network to cluster the features of the input patterns in the training set. The MM is trained by using the lightness factor vector associated to the palette (of the source image to be re-indexed) with the aim to cluster the common input features.

The learning process has been modified respect to the classic algorithm in order to adapt itself to solve the palette re-indexing issue. The unsupervised learning mechanism of MM can be adapted to general purpose systems. In our case, the overall approach can be described by the following steps.

Step 1) The topology of the MM has been established by making use of some heuristic considerations so that a lattice structure of 64 $\times$ 64 neurons appears suitable for this kind of applications. In particular, the lattice structure of 64 $\times$ 64 neurons shows a good compromise between the neural network

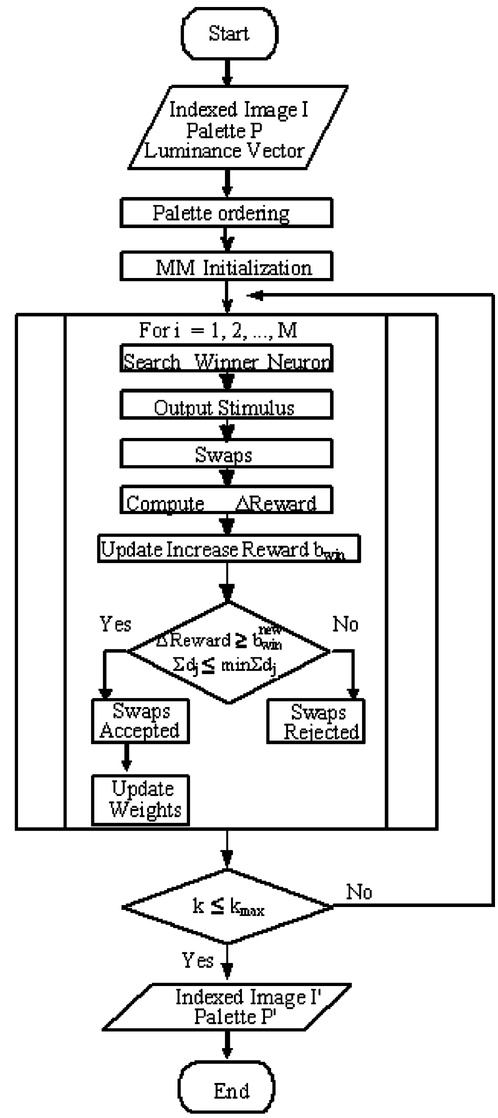

Fig. 3. Flowchart of the proposed MM algorithm.

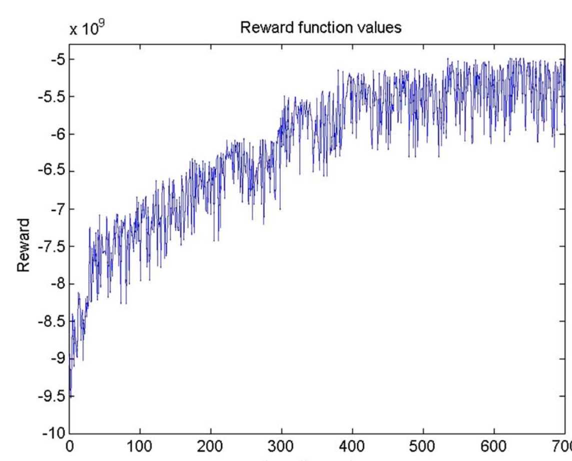

Fig. 4. Reward function.

capability to learn input information (the so called plasticity of the neural network) and the computational resource consumption. The self-organizing structure of the MM provides a ro-

TABLE I
ENTROPY COMPUTED ON COLOR PALETTE IMAGES ALREADY RE-INDEXED WITH BATTIATO'S APPROACH [10], AND, THEN, PROCESSED WITH THE PROPOSED MM ALGORITHM

Images	Colours	Luminance	Battiato	Battiato + MM	% Epoch less
sea_dusk	46	0,103	0.099	**0.096314**	98%
winaw	10	0,601	0.556	**0.529667**	80%

TABLE II
RESIDUAL ENTROPY OF LOCAL DIFFERENCES COMPUTED BY USING PATTERN V_i, $i = 1, \ldots, 4$

Images	Colours	V_1	V_2	V_3	V_4
clegg	256	4,600	**4,563**	4,586	4,700
cwheel	256	**2,743**	3,289	3,032	3,085
fractal	256	**5,035**	5,095	5,067	5,118
frymire	256	**2,851**	2,896	2,877	2,941
house	256	**4,059**	4,271	4,168	4,286
serrano	256	**2,764**	2,825	2,799	2,877
yahoo	229	**1,545**	1,632	1,603	1,696
sunset	204	**1,730**	2,784	2,285	2,535
descent	122	**2,370**	2,722	2,555	2,708
gate	84	**1,945**	1,945	1,953	2,034
benjerry	48	**0,572**	0,602	0,591	0,631
sea_dusk	46	**0,094**	0,993	0,596	0,750
netscape	32	0,622	**0,611**	0,618	0,662
party8	12	0,340	**0,289**	0,316	0,342
winaw	10	**0,530**	0,707	0,622	0,670
music	8	1,090	**1,074**	1,089	1,120
books	7	**1,421**	1,457	1,442	1,463
pc	6	**0,741**	0,777	0,762	0,881
Average	-	**1,947**	*2,141*	*2,053*	*2,139*
Natural 1	256	**4,623**	*4,673*	*4,653*	*4,740*
	128	**3,847**	*3,869*	*3,861*	*3,932*
	64	*3,133*	**3,101**	*3,138*	*3,193*
Average	-	**3,868**	*3,881*	*3,874*	*3,955*
Natural 2	256	**3,999**	*4,192*	*4,104*	*4,243*
	128	**3,226**	*3,402*	*3,322*	*3,449*
	64	**2,534**	*2,693*	*2,621*	*2,735*
Average	-	**3,253**	*3,429*	*3,349*	*3,476*

bust mechanism to prune all not useful neurons (i.e., neurons always losing in the extended WTA process). The MM during the learning process, after a transient, will use only the needed neurons, according to the input image shape and complexity. Let Q be the number of neurons of the MM. Each neuron is composed by an input weight w_i^{in}, $i = 1, \ldots, Q$ with $w_i^{\text{in}} \in [0, 1]$ and output weight w_i^{out}, $i = 1, \ldots, Q$ and a variable b_i, $i = 1, \ldots, Q$ which stores the average increasing of the reward function. The range of the w_i^{out} values is $[1, \ldots, M]$, where M is the overall number of colors (index) of the input image I. In our case, the following reward function has been chosen:

$$Reward = -\left(\sum D(I')\right)^2. \tag{2}$$

The above reward function is strictly proportional to the zero-order entropy of the differences matrix $D(I')$. Moreover, the selection of the above reward function leads the MM to find an optimum palette index scheme which minimize the entropy of the image and then the related compression ratio. In the MM design, the output stimulus u has to be derived according to the specific task to be solved; it can be defined as a function f of the output weights w_i^{out} (and eventually of the time t) or by some heuristics [15], [19]. In the proposed approach, the MM output stimulus has been forced equal to w_i^{out}. The w_i^{out} will be equal to a random index generated during the learning process when the corresponding neuron wins. Before to start the learning phase, the MM (both input layer and output layer) is initialized randomly. Just before learning, a palette preprocessing is applied (i.e., the corresponding lightness factor of the input colors are sorted in increasing order).

Step 2) Let Y be 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

$$y_i = 0.299r_i + 0.587g_i + 0.114b_i \quad i = 1, 2, \ldots, M. \tag{3}$$

Step 3) Each element of the vector Y, normalized in the range [0,1], is fed to the input layer of the MM, one element at each iteration, searching the winner neuron (i.e., the neuron with the minimum distance value)

$$d_i = \left|y_k - w_i^{\text{in}}\right| \quad i = 1, 2, \ldots, Q; k \in [1, 2, \ldots, M]. \tag{4}$$

The winner neuron provides an updating of the output stimulus 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). Also, the relative swap of the involved indexes is performed. The new index is generated randomly in the range $[1, 2, \ldots, M]$. After the index updating, the new reward function can be computed; it is possible update only the elements in $D(I')$ involved in the indexes swapping, just to speed up the overall execution time. The $\Delta Reward$ will be computed as

$$\Delta Reward = (Reward^{\text{new}} - Reward^{\text{old}})^2. \tag{5}$$

The average increasing of the reward function is weighted by the $b_{\text{win}}^{\text{new}}$

$$b_{\text{win}}^{\text{new}} = b_{\text{win}}^{\text{old}} + \rho\left(\Delta Reward - b_{\text{win}}^{\text{old}}\right) \tag{6}$$

where ρ is a positive value related to the smoothing action.

Step 4) If the $\Delta Reward \geq b_{\text{win}}^{\text{new}}$ and the new current entropy (really the new current sum of absolute differences) is better than the already ones processed until now, the new index scheme will be accepted and the weights of the winner neurons will be updated as follows:

$$w_{\text{win}}^{\text{in}}(t+1) = w_{\text{win}}^{\text{in}}(t) + \eta\left(y_k - w_{\text{win}}^{\text{in}}(t)\right) \tag{7}$$
$$w_{\text{win}}^{\text{out}}(t+1) = y_{\text{rand}} \tag{8}$$

where η is the learning rate factor.

TABLE III
RESIDUAL ZERO-ORDER ENTROPY OF IMAGES BEFORE AND AFTER USING THE PALETTE RE-ORDERING METHODS

Images	Colours	Random	Luminance	Memon	Zeng	mZeng	Battiato	MM
clegg	256	5,948	5,318	**4,572**	4,693	4,712	4,915	4,600
cwheel	256	4,336	3,504	**2,396**	2,405	2,501	2,413	2,743
fractal	256	6,941	6,104	5,109	5,307	5,164	5,612	**5,035**
frymire	256	3,680	3,421	**2,792**	2,809	2,919	2,852	2,851
ghouse	256	5,682	5,170	**3,814**	3,917	3,956	4,061	4,059
serrano	256	3,865	3,399	**2,538**	2,601	2,710	2,622	2,764
yahoo	229	2,142	2,026	**1,502**	1,507	1,539	1,559	1,545
sunset	204	2,448	2,043	1,789	1,785	1,774	1,831	**1,730**
descent	122	3,250	2,893	2,382	2,371	2,376	2,505	**2,370**
gate	84	3,274	3,086	2,557	2,720	2,566	2,599	**2,542**
benjerry	48	1,307	1,281	**1,078**	1,090	1,081	1,113	1,079
sea_dusk	46	0,101	0,103	**0,094**	0,096	0,099	0,099	**0,094**
netscape	32	2,415	2,100	1,881	1,895	1,884	1,927	**1,879**
party8	12	0,377	0,391	0,341	0,342	0,342	0,344	**0,340**
winaw	10	0,644	0,601	**0,530**	**0,530**	**0,530**	0,556	**0,530**
music	8	1,234	1,234	1,086	**1,075**	1,086	1,087	1,090
books	7	1,852	1,782	**1,421**	1,425	1,425	**1,421**	**1,421**
pc	6	0,808	0,836	0,746	**0,741**	0,748	0,741	0,741
Average	-	*2,795*	*2,516*	**2,035**	*2,073*	*2,078*	*2,125*	*2,078*
Natural 1	256	5,829	4,988	**3,764**	3,981	3,956	4,143	3,999
	128	4,715	4,008	**3,098**	3,230	3,195	3,319	3,226
	64	3,616	3,101	**2,444**	2,491	2,491	2,576	2,534
Average	-	*4,720*	*4,032*	**3,102**	*3,234*	*3,214*	*3,346*	*3,253*
Natural 2	256	6,589	5,879	**4,556**	4,875	4,750	5,083	4,623
	128	5,487	4,884	**3,789**	4,025	3,954	4,171	3,847
	64	4,414	3,944	**3,100**	3,236	3,226	3,370	3,133
Average	-	*5,497*	*4,902*	**3,815**	*4,045*	*3,977*	*4,208*	*3,868*

After that, the learning steps 2)–4) are repeated until the stop criteria is verified. Conversely, the new index scheme will be rejected and the previous ones will be restored. Fig. 2 shows the MM re-indexing mechanism while, in Fig. 3, a schematic representation of the overall process is reported. In the architecture of the MM proposed in this work, the neuron has not an adaptive neighboring and the learning rate remains constant during all the learning phase. The stop of the learning process is reached when the computed entropy is less or equal to a specific lower bound value or after a fixed number of epochs k_{max} (an epoch is a number of cycles needed for presenting all the input patterns to neural network).

C. Algorithm Parameters

The MM parameters are chosen according to trial-and-error policies as well as heuristic considerations. In particular, we have chosen $\eta = 0.01$ and $\rho = 0.90$. Fig. 4 shows a plot of typical reward function with respect to the learning cycles (epochs). Typical number of learning cycle k is in the range of [350, 700] in almost all involved experiments.

To evaluate the learning speed process we have applied the proposed method to two synthetic re-indexed images obtained by [10] (without the luminance ordering step). Table I shows how in both cases there is a real speed-up of about 90% in terms of learning rate and a substantial improvement in terms of related entropy. In some other cases, no improvement was achieved. The initial luminance sorting proceeds disposing the overall indexes according to the "natural smoothnes" of real world. Starting from other configuration the MM methods could be not able to find the solution due to a sort of suboptimality of the initial ordering.

The set of differences $D(I')$ has been computed by using different local pattern configurations [20]. The considered patterns are

$$D_{V_1}(I') = \left\{ I'_{x,y} - I'_{x,y-1} \right\} \tag{9}$$

$$D_{V_2}(I') = \left\{ I'_{x,y} - I'_{x-1,y} \right\} \tag{10}$$

$$D_{V_3}(I') = \left\{ \frac{2 * I'_{x,y} - I'_{x,y-1} - I'_{x-1,y}}{2} \right\} \tag{11}$$

$$D_{V_4}(I') = \left\{ \frac{3 * I'_{x,y} - I'_{x,y-1} - I'_{x-1,y} - I'_{x-1,y-1}}{3} \right\} \tag{12}$$

where x, y are the corresponding valid indexes in the original $m \times n$ image I.

Table II reports the final residual entropy of the local differences by making use of pattern V_i con $i = 1, \ldots, 4$ applied to a specific dataset. The dataset is organized in three groups: synthetic, a set of natural images also known as the "Kodak" database, and a set of popular natural images. The last two sets contain quantized version of the same images with 256, 128, and 64 colors, respectively. In almost all cases, the pattern V_1 gives better results also considering the different number of involved colors. For this reason, according to analog consideration in [20], we have adopted the pattern V_1 as basis for our experiments.

TABLE IV
LOSSLESS COMPRESSION RESULTS IN BIT PER PIXEL, OBTAINED WITH JPEG2000 APPLIED TO THE INDEXED IMAGES AFTER USING THE PALETTE REORDERING METHODS PRESENTED IN THE PAPER

Images	**Colours**	**Palette Size**	**Random**	**Luminance**	**Index-based methods**				
					Memon	**Zeng**	**mZeng**	**Battiato**	**MM**
clegg	256	0,008	6,961	6,184	**5,951**	6,497	6,126	7,166	**5,951**
cwheel	256	0,013	4,876	4,686	**3,068**	3,398	3,243	3,661	3,777
fractal	256	0,016	7,244	6,229	5,974	6,406	6,038	7,522	**5,768**
frymire	256	0,005	5,176	5,377	**4,093**	4,369	4,217	4,721	4,132
ghouse	256	0,013	7,050	5,488	**4,760**	5,402	5,091	6,083	5,130
serrano	256	0,012	5,508	5,148	**4,038**	4,416	4,323	4,901	4,519
yahoo	229	0,202	3,673	4,378	**2,220**	2,229	2,253	2,500	2,302
sunset	204	0,016	4,193	3,328	3,424	3,579	3,263	3,859	**3,096**
descent	122	0,046	4,715	4,536	3,499	3,504	**3,424**	4,510	3,470
gate	84	0,033	4,152	3,460	2,987	3,088	3,037	3,538	**2,944**
benjerry	48	0,041	2,416	2,825	**1,745**	1,761	1,769	2,008	**1,745**
sea_dusk	46	0,007	0,159	0,314	0,255	0,129	0,129	**0,120**	0,220
netscape	32	0,012	2,836	2,653	2,382	2,420	2,390	2,611	**2,381**
party8	12	0,002	0,586	0,970	**0,411**	0,414	0,413	0,432	0,421
winaw	10	0,000	0,860	0,919	0,611	0,611	0,611	0,634	**0,610**
music	8	0,015	1,708	2,142	1,374	1,388	1,374	1,428	**1,366**
books	7	0,003	2,239	2,169	1,598	1,601	1,601	1,604	**1,596**
pc	6	0,000	1,082	0,953	0,748	0,736	0,749	0,736	**0,673**
Average	-	-	*3,438*	*3,141*	**2,611**	*2,794*	*2,677*	*3,078*	*2,783*
Natural1	256	0,016	6,574	5,274	**4,575**	5,281	4,870	5,938	4,860
	128	0,008	5,358	4,308	**3,792**	4,203	3,914	4,717	3,927
	64	0,004	4,208	3,399	**2,981**	3,146	3,046	3,539	3,088
Average	-	-	*5,380*	*4,327*	**3,783**	*4,210*	*3,943*	*4,731*	*3,958*
Natural2	256	0,026	6,959	5,429	**5,062**	5,904	5,358	6,532	5,181
	128	0,013	5,903	4,552	**4,203**	4,769	4,447	5,201	4,216
	64	0,006	4,596	3,523	**3,213**	3,493	3,349	3,814	3,329
Average	-	-	*5,819*	*4,501*	***4,159***	*4,722*	*4,385*	*5,182*	*4,242*

D. Computational Complexity

Let M the overall number of involved colors of an input image I having $N = m \times n$ pixels. The proposed technique requires a preprocessing phase devoted to sort the input according to their lightness factor.

Each learning cycle have to compute the reward function, by considering just a single index swap in the matrix index $D(I')$. The overall computational complexity is $O(M \log M) + O(kNM)$ where k is the number of learning cycles.

We remember that previously published works in the field of re-indexing of color mapped images have been often also compared with respect to their asymptotic complexity but only considering the number of colors M. According to this criteria, the most effective and used methods have the following time complexity: $O(M \log M)$ for luminance order, $O(M^2 \log M)$ for Battiato's approach [10], $O(M^3)$ for Zeng's [9] and its modification [20], and $O(M^4)$ for Memon. The proposed MM re-indexing is clearly one of the most efficient methods also when the above complexity is computed considering the input image resolution (i.e., the number of involved pixels N). When the input resolution size is much greater than M (and k) the asymptotic complexity of our system is just linear. Of course, there is no way to design a re-indexing scheme without accessing (at least once) to the original indexes stored in the input image. As reported in [11], the evaluation (and comparisons) of timing performances of the various methods cannot be done: most of the implementations have not been optimized for speed. Pei *et al.* [14] have found that, if the palette size dominates the image size, although the reindexing consumes ignorable time when the palette is small, the complexity of the almost all methods goes apparent as more as colors as used. For our system, we simply remember that a discrete components electronic realization is presented in [19].

IV. EXPERIMENTAL RESULTS

In order to check the performances of the MM as palette re-indexing algorithm, we propose the comparison between our method and the most important reordering methods. In particular, we compare our MM method with classical luminance re-ordering, Memon's technique [8], Zeng's algorithm [9], modified Zeng (mZeng) algorithm [11], and with our previous solution proposed in [10]. For sake of comparison, as above specified, the dataset used is the same as in [11] and [14]. The source code for the other re-indexing methods [8], [9], [11], [20] has been kindly provided by the authors of [11]. We present only results by considering nondithered version of *Natural1* and *Natural2* groups because the performances (in terms of bpp) of any re-indexing technique are clearly affected

TABLE V
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

Images	Colours	Palette Size	Random	Luminance	Index-based methods				
					Memon	Zeng	mZeng	Battiato	MM
clegg	256	0,008	6,333	5,330	5,220	5,863	5,456	6,075	**5,188**
cwheel	256	0,013	4,134	4,087	**2,724**	3,058	2,878	3,374	3,297
fractal	256	0,016	7,036	6,086	5,763	6,193	5,828	7,234	**5,549**
frymire	256	0,005	4,083	3,759	**3,259**	3,619	3,376	3,946	3,291
ghouse	256	0,013	6,399	4,890	**4,229**	4,841	4,541	5,418	4,476
serrano	256	0,012	4,087	3,583	**3,001**	3,393	3,273	3,779	3,249
yahoo	229	0,202	2,701	2,787	1,743	1,798	1,789	2,008	**1,648**
sunset	204	0,016	2,854	2,328	2,407	2,570	2,307	2,647	**2,185**
descent	122	0,046	3,762	3,485	2,817	2,943	2,854	3,531	**2,792**
gate	84	0,033	3,490	2,930	2,548	2,587	2,566	3,116	**2,538**
benjerry	48	0,041	1,379	1,423	1,133	1,154	1,137	1,186	**1,118**
sea_dusk	46	0,007	0,194	0,191	0,197	0,189	0,189	0,189	**0,195**
netscape	32	0,012	2,040	1,918	**1,745**	1,791	1,752	1,907	1,748
party8	12	0,002	0,346	0,367	0,321	**0,318**	**0,318**	0,319	**0,318**
winaw	10	0,000	0,552	0,546	**0,450**	**0,450**	**0,450**	0,464	**0,450**
music	8	0,015	1,171	1,143	**1,051**	1,060	**1,051**	1,071	1,067
books	7	0,003	2,046	1,884	**1,453**	1,469	1,469	**1,453**	**1,453**
pc	6	0,000	0,812	0,768	0,748	**0,743**	0,745	**0,743**	0,751
Average	-	-	*2,854*	*2,526*	**2,243**	*2,452*	*2,333*	*2,650*	*2,306*
Natural1	256	0,016	6,077	4,897	**4,203**	4,907	4,475	5,530	4,440
	128	0,008	4,838	3,926	**3,441**	3,844	3,552	4,287	3,526
	64	0,004	3,661	3,002	**2,641**	2,804	2,709	3,144	2,710
Average	-	-	*4,859*	*3,942*	**3,428**	*3,852*	*3,579*	*4,320*	*3,558*
Natural2	256	0,026	6,433	5,008	**4,607**	5,491	4,881	5,983	4,779
	128	0,013	5,318	4,117	**3,771**	4,339	3,979	4,729	3,785
	64	0,006	3,967	3,069	**2,793**	3,060	2,912	3,281	2,921
Average	-	-	*5,239*	*4,065*	**3,724**	*4,297*	*3,924*	*4,664*	*3,828*

by a slightly degradation due to the controlloed "noise" inserted by the specific applied dithering.

Our proposed solution has been implemented in Ansi C. For each re-indexed image, a log file shows the number of epochs performed for re-indexing the corresponding palette.

As stated above, the first evaluation of a re-indexing scheme have to be done by using the residual zero order entropy of local differences $D_{V_i}(I')$ by using the pattern V_1. A useful comparisons of such values between the proposed method MM and the others is reported in Table III.

The Memon's approach on the average confirms its good performances but the proposed MM algorithm typically provides results almost similar often outperforming all the other methods, especially for the synthetic group.

Tables IV–VI report the bits rate in terms of bpp (bit per pixels) obtained by lossless compression of the dataset after palette reordering by using JPEG2000,[1] JPEG-LS,[2] and PNG,[3] respectively. The tables also show the size (in bpp) of the corresponding palette just to evaluate the relative impact on the overall results. By using the PNG codec, the values of bpps of the proposed approach are considerable lower than the other methods; in some cases, the differences are substantial. This trend is maintained independently from the number of colors and it does not change if we compare synthetic or natural images.

The filtering process used by PNG codec is mainly devoted to preprocess the input data to reduce the local spatial redundancy. Four different predictors (i.e., Sub, Up, Average, and Paeth), based on the local neighbourhood allows to properly reduce the residual entropy [21], [22]. More specifically, in *libpng*, the free reference library used by most PNG supporting applications, the filter that minimizes the sum of absolute differences is chosen for each raw under processing [22]. In some sense, the PNG-codec makes use of a local (raw-based) optimization criterion that is almost identical to the reward function (2) allowing to the proposed approach to reach the best results just for PNG coding. In any case, there is no a clear evidence of the various performances of JPEG2000, JPEG-LS, and PNG codec engines (lossless) also considering different image categories as reported in [23] and [24].

With JPEG-LS and JPEG2000 compression, the proposed approach is not able to sensibly improve the Memon's results especially for the *Natural1* group. We remember that Memon's approach is, however, strongly limited by its computational complexity [11], [20]. The group *Natural2* has been also re-indexed by considering the corresponding dithered version. Table VII reports the overall bit-rate obtained for the three considered codec engine. As expected, there is a common degradation although MM seems to be more robust (i.e., in terms of additional bpp

[1]FastStone vers. 2.6(B4),(www.FastStone.org).

[2]SPMG/JPEG-LS Encoder vers. 1.0

[3]The Gimp vers. 2.2.9

TABLE VI
LOSSLESS COMPRESSION RESULTS IN BIT PER PIXEL, OBTAINED WITH PNG APPLIED TO THE INDEXED IMAGES AFTER USING THE PALETTE REORDERING METHODS PRESENTED IN THE PAPER

Images	**Colours**	**Palette Size**	**Random**	**Luminance**	**Index-based methods**				
					Memon	**Zeng**	**mZeng**	**Battiato**	**MM**
clegg	256	0,008	5,494	5,185	**4,564**	4,622	4,700	4,704	4,597
cwheel	256	0,013	3,217	3,042	2,630	2,651	2,687	**2,605**	2,734
fractal	256	0,016	7,085	6,841	5,842	6,097	5,888	6,465	**5,677**
frymire	256	0,005	3,065	3,007	2,770	2,735	2,848	**2,714**	2,863
ghouse	256	0,013	5,387	5,563	**4,423**	4,585	4,596	4,522	4,642
serrano	256	0,012	3,415	3,293	**2,835**	2,872	3,007	2,874	2,975
yahoo	229	0,202	2,230	2,256	**1,861**	1,890	1,927	1,873	**1,861**
sunset	204	0,016	2,524	2,267	2,165	2,170	2,147	2,116	**2,049**
descent	122	0,046	3,768	3,356	3,205	3,205	3,218	2,962	**2,862**
gate	84	0,033	3,572	3,263	3,295	3,045	3,219	2,923	**2,747**
benjerry	48	0,041	1,391	1,396	1,362	1,404	1,372	1,232	**1,222**
sea_dusk	46	0,007	0,092	0,081	0,093	0,086	0,085	**0,080**	0,092
netscape	32	0,012	2,629	2,290	2,514	2,400	2,523	2,149	**2,100**
party8	12	0,002	0,364	0,369	0,368	0,368	0,368	0,349	**0,345**
winaw	10	0,000	0,639	0,620	0,633	0,618	0,633	0,586	**0,580**
music	8	0,015	1,240	1,304	1,310	1,285	1,310	1,221	**1,203**
books	7	0,003	2,062	2,038	2,181	2,215	2,242	**1,782**	1,811
pc	6	0,000	0,516	0,532	0,507	0,507	0,508	**0,505**	0,506
Average	-	-	2,705	2,595	2,364	2,375	2,404	2,315	**2,270**
Natural1	256	0,016	5,786	5,611	**4,404**	4,656	4,625	4,688	4,605
	128	0,008	5,604	4,674	4,041	4,175	4,193	3,975	**3,776**
	64	0,004	4,234	3,613	3,438	3,497	3,529	3,116	**3,009**
Average	-	-	*5,208*	*4,633*	*3,961*	*4,109*	*4,116*	*3,926*	**3,797**
Natural2	256	0,026	6,757	6,685	**5,305**	5,592	5,482	5,667	5,350
	128	0,013	6,721	5,845	5,127	5,081	5,216	5,071	**4,541**
	64	0,006	5,396	4,715	4,558	4,545	4,671	4,130	**3,772**
Average	-	-	*6,291*	*5,749*	4,997	*5,073*	5,123	*4,956*	**4,554**

TABLE VII
LOSSLESS COMPRESSION RESULTS IN BIT PER PIXEL, OBTAINED WITH JPEG2000 AND JPEG-LS APPLIED TO THE INDEXED IMAGES WITH DITHERING AFTER USING THE PALETTE REORDERING METHODS PRESENTED IN THE PAPER

Images	**Colours**	**Palette Size**	**Random**	**Luminance**	**Index-based methods**				
					Memon	**Zeng**	**mZeng**	**Battiato**	**MM**
Natural2	256	0,026	6,923	5,363	**5,063**	6,006	5,322	5,455	5,074
JPEG-LS	128	0,013	5,844	4,495	**4,205**	4,766	4,423	5,314	4,213
	64	0,006	4,712	3,643	**3,417**	3,712	3,548	4,127	3,418
Average	-	-	*5,826*	*4,500*	**4,228**	*4,828*	*4,431*	*4,965*	*4,235*
Natural2	256	0,026	7,234	5,651	5,408	6,289	5,706	6,867	**5,383**
JPEG2000	128	0,013	6,164	4,754	**4,482**	5,049	4,720	5,607	4,494
	64	0,006	5,037	3,908	**3,694**	3,999	3,831	4,447	3,698
Average	-	-	*6,145*	*4,771*	**4,528**	*5,112*	*4,752*	*5,640*	*4,529*
Natural2	256	0,026	6,757	6,685	**5,305**	5,592	5,482	5,667	5,350
PNG	128	0,013	6,721	5,845	5,127	5,081	5,216	5,071	**4,541**
	64	0,006	5,396	4,715	4,558	4,545	4,671	4,130	**3,772**
Average	-	-	*6,291*	*5,749*	4,997	*5,073*	5,123	*4,956*	**4,554**

required) with respect to the general decreasing of the compression ratio obtained with all other methods.

If we consider the cumulative results of the three codec engine used, the proposed approach has the best performances both in terms of bpp and computational complexity. Finally, we show the final re-ordered palette for a single image just to visually evaluate the smoothness obtained with the different methods (Fig. 5). We also report (Fig. 6) the histogram of the corresponding $D(I')$ with respect to the various re-indexed images of Fig. 5. The MM histogram provides a distribution clearly more "peaked" (see the peak value at 0 and the lower tails), and, hence, it is amenable to any involved compression engine than all others.

All re-indexed images can be downloaded at the following web address: http://www.dmi.unict.it/~iplab/MMap_reindexing/.

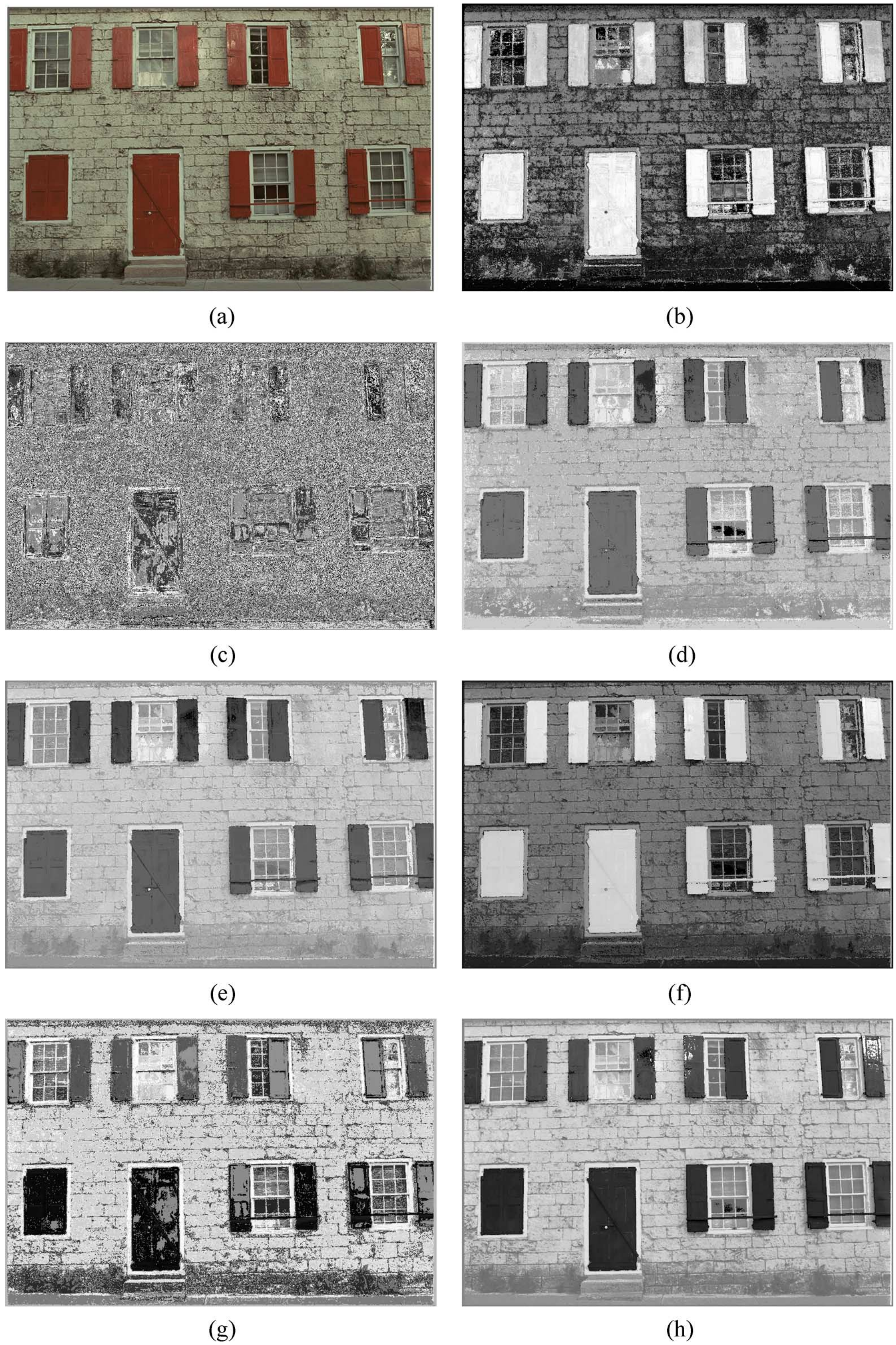

Fig. 5. Different re-indexing schemes. (a) Original color image; (b) original indexes; (c) random; (d) Zeng's re-indexing; (e) modified Zeng's re-indexing; (f) Memon's re-indexing; (g) Battiato's re-indexing; (h) proposed MM approach. (a), (b) $\mathrm{Entropy} = 5,762$; (c) $\mathrm{Entropy} = 6,860$; (d)$\mathrm{Entropy} = 5,010$; (e)$\mathrm{Entropy} = 4,928$; (f) $\mathrm{Entropy} = 4,824$; (g) $\mathrm{Entropy} = 5,397$; (h) $\mathrm{Entropy} = 4,806$.

V. 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 Lapla-

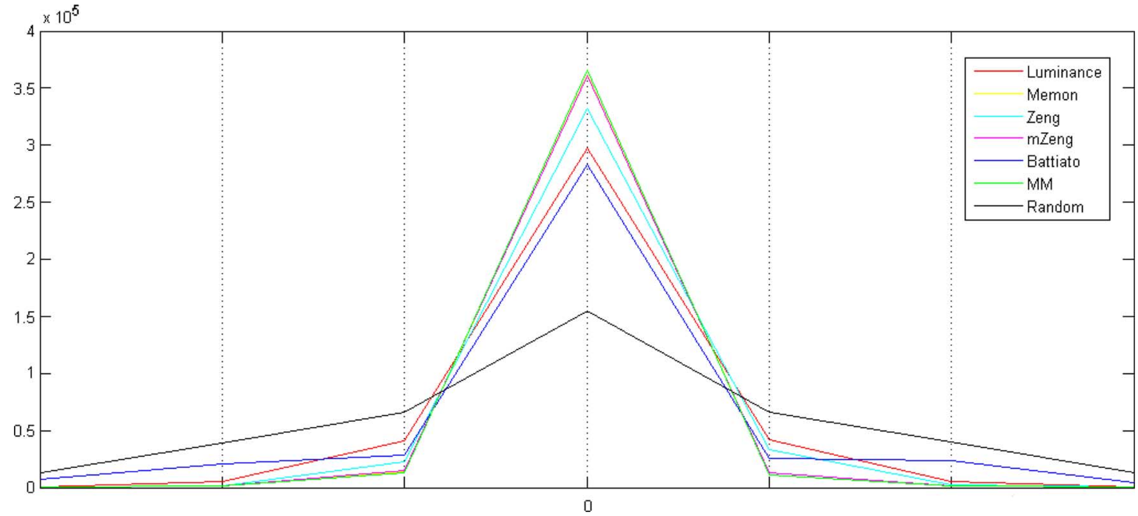

Fig. 6. Histogram of the corresponding $D(I')$ with respect to the various re-indexed images of Fig. 5.

cian model for the prediction residuals and, therefore, may provide a justification for the good performance of both methods. The MM algorithm does not need any hypothesis on the initial image's pixel distribution, as well as on the re-ordered final image.

ACKNOWLEDGMENT

The authors would like to thank the reviewers for helpful comments. The authors would also like to thank Prof. A. Pinho for having provided the software used for comparisons.

REFERENCES

[1] A. Zaccarin and B. Liu, "A novel approach for coding color quantized images," *IEEE Trans. Image Process.*, vol. 2, no. 4, pp. 442–453, Oct. 1993.

[2] A. Spira and D. Malah, "Improved lossless compression of color-mapped images by an approximate solution of the traveling salesman problem," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing*, May 2001, no. III, pp. 1797–1800.

[3] L. M. Po and W. T. Tan, "Block address predictive color quantization image compression," *Electron. Lett.*, vol. 30, no. 2, pp. 120–121, Jan. 1994.

[4] A. C. Hadenfeldt and K. Sayood, "Compression of color-mapped images," *IEEE Trans. Geosci. Remote Sens.*, vol. 32, no. 5, pp. 534–541, May 1994.

[5] J. Fojtik and H. Vaclav, "Invisible modification of the palette colour image enhancing lossless compression," in *Proc. Joint IAPR Workshops SSPR*, 1998, vol. 1451, LNCS, pp. 1029–1036.

[6] P. Waldemar and T. A. Ramstad, "Subband coding of color images with limited palette size," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, Apr. 1994, no. V, pp. 353–356.

[7] M. J. Gormish, "Compression of palletized images by color," presented at the IEEE Int. Conf. Image Processing, 1995.

[8] N. Memon and A. Venkateswaran, "On ordering color maps for lossless predictive coding," *IEEE Trans. Image Process.*, vol. 5, no. 11, pp. 1522–1527, Nov. 1996.

[9] W. Zeng, J. Li, and S. Lei, "An efficient color re-indexing scheme for palette-based compression," in *Proc. 7th IEEE Int. Conf. Image Processing*, Sep. 2000, pp. 476–479.

[10] S. Battiato, G. Gallo, G. Impoco, and F. Stanco, "An efficient re-indexing algorithm for color-mapped images," *IEEE Trans. Image Process.*, vol. 13, no. 11, pp. 1419–1423, Nov. 2004.

[11] A. Pinho and A. Neves, "A survey on palette reordering methods for improving the compression of color-indexed images," *IEEE Trans. Image Process.*, vol. 13, no. 11, pp. 1411–1418, Nov. 2004.

[12] K.-S. You, D.-S. Han, E. S. Jang, S.-Y. Jang, S.-K. Lee, and H.-S. Kwak, "Ranked image generation for arithmetic coding in indexed color image," in *Proc. 7th Int. Workshop on Enterprise networking and Computing in Healthcare Industry, HEALTHCOM*, Jun. 2005, pp. 299–302.

[13] A. J. R. Neves and A. J. Pinho, "A bit-plane approach for lossless compression of color-quantized images," presented at the IEEE Int. Conf. Acoustics, Speech, and Signal Processing, May 2006.

[14] S.-C. Pei, Y.-T. Chuang, and W.-H. Chuang, "Effective palette indexing for image compression using self-organization of kohonen feature map," *IEEE Trans. Image Process.*, vol. 15, no. 9, pp. 2493–2498, Sep. 2006.

[15] T. Martinetz, H. Ritter, and K. Schulten, *Neural Computing and Self-Organizing Maps*. Reading, MA: Addison-Wesley, 1992.

[16] P. Arena, M. Bucolo, L. Fortuna, and M. Frasca, "Motor map for nonlinear control," in *Proc. 6th Experimental Chaos Conf.*, 2001, pp. 22–26.

[17] P. Arena, L. Fortuna, M. Frasca, and G. Sicurella, "An adaptive, self-organizing dynamical system for hierarchical control of bio-inspired locomotion," *IEEE Trans. Syst., Man, Cybern. B*, vol. 34, no. 4, pp. 1823–1837, Aug. 2004.

[18] T. Kohonen, "Self organized formation of topologically correct feature maps," *Bio.Cybern.*, vol. 43, 1972.

[19] P. Arena, L. Fortuna, and M. Frasca, "Chaos control by using motor maps," *Chaos*, vol. 12, no. 3, 2002.

[20] A. J. Pinho and A. J. R. Neves, "On the relation between memon's and the modified zeng's palette reordering methods," *Image Vis. Comput.*, vol. 24, no. 5, pp. 534–540, 2006.

[21] G. Roleof, *PNG: The Definitive Guide*, 2nd ed. San Jose, CA: Greg Roelof, 2003 [Online]. Available: http://www.libpng.org/pub/png/book/

[22] K. Sayood, *Lossless Compression Handbook*. New York: Academic, 2002.

[23] R. Starosolski, "Performance evaluation of lossless medical and natural continuous tone image compression algorithms," in *Proc. SPIE Medical Imaging Conf.*, 2005, vol. 5959.

[24] D. Santa-Cruz, R. Grosbois, and T. Ebrahimi, "Jpeg 2000 performance evaluation and assessment," *Signal Process.: Image Commun.*, vol. 17, no. 1, pp. 113–130, Jan. 2002.

Sebastiano Battiato (M'05–SM'06) was born in Catania, Italy, in 1972. He received the degree in computer science (*summa cum laude*) in 1995 and the Ph.D. degree in computer science and applied mathematics in 1999.

From 1999 to 2003, he lead the "Imaging" team c/o STMicroelectronics, Catania. Since 2004, he has been a Researcher with the Department of Mathematics and Computer Science, University of Catania. He has published more than 80 papers in international journals and conference proceedings. He is the co-inventor of about 15 international patents. He is a reviewer for several international journals and has participated in many international and national research projects. His research interests include image enhancement and processing and image coding.

Francesco Rundo received the degree in computer science engineering from the University of Catania, Catania, Italy, in 2000, where he is currently pursuing the Ph.D. degree and is a member of IPLAB Research Group, Department of Mathematics and Informatics.

Since December 2000, he has been with the R&D Division of ST Microelectronics, Italy, as a Senior System Engineer. Since 2003, he has been a member of the Research Group in Applied Mathematics, University of Cagliari, Cagliari, Italy. His research interests include neural computing for image processing, complex systems, adaptive zooming algorithms, cellular neural networks, and evolutionary robotics.

Mr. Rundo is a member of the Italian National Institute of High Mathematics.

Filippo Stanco (M'04) was born in Nicosia, Italy, in 1975. He received the degree in computer science (*summa cum laude*) and the Ph.D. degree in computer science from the University of Catania, Catania, Italy, in 1999 and 2003, respectively.

From 2003 to 2006, he was a Research Assistant at the University of Trieste, Trieste, Italy, and the University of Catania. Since 2006, he has been a Researcher with the Department of Mathematics and Computer Science, University of Catania. He has published several papers in international journals and conference proceedings, and he is a reviewer for several leading international conferences and journals. His research interests include digital restoration, artifacts removal, filtering, super-resolution, interpolation, and texture analysis.