A comparison of experimental results with an Evolution Strategy and Competitive Neural Networks for near real-time Color Quantization of image sequences

A. I. Gonzalez, M. Graña, A. D'Anjou, F.X. Albizuri, F.J Torrealdea

Dept. CCIA Univ. Pais Vasco/EHU[1],Aptdo 649, 20080 San Sebastián, España

e-mail: ccpgrrom@si.ehu.es

ABSTRACT

Color Quantization of image sequences is a case of Non-stationary Clustering problem. The approach we adopt to deal with this kind of problems is to propose adaptive algorithms to compute the cluster representatives. We have studied the application of Competitive Neural Networks and Evolution Strategies to the one-pass adaptive solution of this problem. One-pass adaptation is imposed by the near real-time constraint that we try to achieve. In this paper we propose a simple and effective Evolution Strategy for this task. Two kinds of Competitive Neural Networks are also applied. Experimental results show that the proposed Evolution Strategy can produce results comparable to that of Competitive Neural Networks.

1 INTRODUCTION

Evolution Strategies [1, 2, 3] have been developed since the sixties. They belong to the broad class of algorithms inspired by natural selection. The features most widely accepted as characteristic of Evolution Strategies are: (1) vector real valued individuals, (2) the main genetic operator is mutation[2] , (3) individuals contain local information for mutation so that adaptive strategies can be formulated to self-regulate the mutation operator. However, it is widely recognized [3] that a lot of hybrid algorithms can be defined, so that it is generally difficult to assign a definitive "label" for a particular algorithm. Nevertheless, we classify the algorithm

[1]This work is being supported by a research grant from the Dpto de Economía of the Excma. Diputación de Guipuzcoa, and a predoctoral grant and project UE96/9 of the Gobierno Vasco

[2] This assertion can be subject to discussion. The least that can be asserted is that, in the context of Evolution Strategies, mutation has a more sensible definition than recombination. The generation of mutated individuals as random (gaussian) perturbations of the parents has the clear meaning of a local search around the actual solutions, whereas the definitions of recombination are less clear. In fact, most of the formal works on convergence and adaptive autotuning of parameters focus on the role of mutation.For a recent revision see [24]

proposed here as an Evolution Strategy because it fits better in the above characterization than in the accepted characterizations of Genetic Algorithm or Genetic Programming. The population of the proposed algorithm is the set of cluster representatives, each individual is a color representative in the $[0,1]^3$ real subspace that defines a cluster in the color space of the image pixels via the Voronoi tesselation induced by the whole population. The main evolutive operator is mutation. The mutation operator is based on the local cluster covariance matrices, which guide and regulate its realization. We have not defined recombination operators.

Competitive Neural Networks [4, 5] are usually proposed as stochastic gradient descent algorithms on distortion like functions and have been extensively applied to Vector Quantization and Clustering problems. In this paper we will apply the basic Simple Competitive Learning scheme which minimizes the within cluster scattering of the data, and the Soft Competition algorithm [6, 7] that can be derived as a minimization of the Kullback-Leiber cross-entropy that measure the fitness of the empirical distribution of the data by a mixture of Gaussians.

Non-stationary Clustering problems assume a time varying population sampled at discrete times. Therefore the Clustering of the data must be recomputed at each time instant. A related problem is that of Adaptive Vector Quantization [8]. The works reported here belong to the class of shifting-mean Adaptive Vector Quantization [9]. Both the Evolution Strategy and the Competitive Neural Networks are applied as adaptive algorithms for the computation of the cluster representatives given by the cluster means at each time instant. Our work fits in the original proposition of Holland [10] of evolution as a mechanism for adaptation to uncertain and varying environment conditions

Both Evolution Strategies and Competitive Neural Networks fall in the broad family of stochastic algorithms. These algorithms are characterized by slow convergence and large computation times. As we are trying to apply them in a near real-time framework, we impose two computational restrictions in their application: (1) One-pass adaptation and (2) we use subsamples of the data to estimate the cluster representatives for the whole data set at each time instant. For the Evolution Strategy, restriction (1) implies that only one generation is computed at each time instant. For the Competitive Neural Networks, this restriction implies that the sample data is presented only once, therefore the learning parameters must have a very fast decrease.

Color Quantization [11, 12, 13, 14] is an instance of Vector Quantization (VQ) [8] in the space of colors. Color Quantization has application in visualization, color image segmentation, data compression and image retrieval [15]. In this paper we do not deal with the problem of finding the *natural* number of colors. This is a more involved problem than looking for a fixed number of color representatives, and some of the results discussed in section 5 recommend that it must approached cautiously, and after being satisfied with the results of quantization to a fixed number of colors. In summary, Color Quantization to a fixed number of colors of image sequences [16] is a case of Non-stationary Clustering, that we deal with by performing Adaptive Vector Quantization.

The paper is organized as follows. Section 2 introduces the framework of Non-stationary Clustering/VQ and the adaptive approach to solve it. Section 3 presents the Evolution strategy that we propose. Section 4 reviews the definitions of the Competitive Neural Networks. Section 5 presents the experimental results, and section 6 gives our conclusions and lines for further work.

2 ADAPTIVE APPROACH TO NON-STATIONARY CLUSTERING/VQ

Cluster analysis and Vector Quantization are useful techniques in many engineering and scientific disciplines [8,17,18,19,20,21]. In their most usual formulation it is assumed that the data is a sample of an stationary stochastic process, whose statistical characteristics will not change in time. Non-stationary Clustering and Vector Quantization assume a time varying population sampled at diverse time instants that can be modelled by a discrete time non-stationary stochastic process $\{X_t \;\; t = 0,1,..\}$. If a model is known (or assumed), a predictive approach [8] would reduce the problem to a stationary one. The general formulation of the Non-stationary Clustering problem does not assume any model of the process.

A working definition of the Non-stationary Clustering problem could read as follows: Given a sequence of samples $\aleph(t) = \{\mathbf{x}_1(t),...,\mathbf{x}_n(t)\}$ obtain a corresponding sequence of partitions of each sample given by a sequence of sets of disjoint clusters $\{\aleph_1(t),..,\aleph_c(t)\}$ that minimizes a criterium function $C = \sum_{t \geq 0} C(t)$ The similar Non-stationary Vector Quantization design problem can be stated as the search for a sequence of representatives $\mathbf{Y}(t) = \{\mathbf{y}_1(t),..,\mathbf{y}_c(t)\}$ that minimizes the error function (distortion) $E = \sum_{t \geq 0} E(t)$. The squared Euclidean distance is the dissimilarity measure most widely used to define criterium/error functions. The Non-stationary Clustering/VQ problem can be stated as an stochastic minimization problem:

$$\min_{\{\mathbf{Y}(t)\}} \sum_{t \geq 0} \sum_{j=1}^{n} \sum_{i=1}^{c} \left\| \mathbf{x}_j(t) - \mathbf{y}_i(t) \right\|^2 \delta_{ij}(t)$$

$$\delta_{ij}(t) = \begin{cases} 1 & i = \underset{k=1,..,c}{\arg\min} \left\{ \left\| \mathbf{x}_j(t) - \mathbf{y}_k(t) \right\|^2 \right\} \\ 0 & \text{otherwise} \end{cases}$$

The proposition of adaptive algorithms to solve this stochastic minimization problem is based in two simplifying assumptions: (1) The minimization of the sequence of time dependent error function can be done independently at each time step. (2) Smooth (bounded) variation of optimal set of representatives at successive time steps. Then the set of representatives obtained after adaptation in a time step can be used as the initial conditions for the next time step.

The adaptive application of Evolution Strategies, such as the one presented below, is done as follows: At time t the initial population is given by the set of representatives/ codevectors computed from the sample of the process at time t-1. A series of generations are computed starting from this initial population to compute the representatives for the clusters of the sample at time t. The fitness function is related to the distortion of the representatives, coded somehow in the population, relative to the sample at time t. This process is repeated for the sample at time t+1, and thereafter. A distinctive feature of our proposed Evolution Strategy is that only one generation is computed to perform the adaptive step.

The adaptive application of the Competitive Neural Networks has been done as follows. The initial cluster representatives are assumed to be the ones found for the previous data sample. Sample vectors are randomly extracted from the sample data and used to compute the adaptation rules. Only a small subsample is used once in the adaptation. In practice, we have extracted a subsample of each image in the sequence and this has been used as the data samples for both the Evolution Strategy and the Competitive Neural Networks.

3 THE EVOLUTION STRATEGY

A widely accepted pseudocode representation of the general structure of the algorithm of Evolution Strategies is as follows [2]:

```
t:= 0
initialize P(t)
evaluate P(t)
while not terminate do
    P'(t):= recombine P(t)
    P"(t):= mutate P'(t)
    evaluate P"(t)
    P(t+1):= select (P"(t) U Q)
    t:= t+1
end while
```

We have defined each individual as a single cluster centre, so that the entire population gives a single solution to the Clustering/VQ problem. From the discussion in the preceding section, follows that the generation number coincide with the time instant at which the sample is taken. The population at generation t is given by $P(t)=\{y_i(t);i=1..c\}$. The local fitness of each individual is, then, its local distortion $F_i(t)=\sum_{j=1}^{n}\left\|\mathbf{x}_j(t)-\mathbf{y}_i(t)\right\|^2\delta_{ij}(t)$ relative to the sample considered in this generation. The fitness of the population as a whole can be evaluated as $F(t)=\sum_{i=1}^{c}F_i(t)$ which corresponds to the objective function to be minimized. Our population fitness corresponds to the within cluster scatter S_W of the clustering specified by the population. The well known equation relating the within cluster and between cluster scattering

$$S = S_W + S_B$$

can be interpreted in the context of the above Evolution Strategy as:

$$S(t) = \sum_{j=1}^{n} \left\| \mathbf{x}_j(t) - \bar{\mathbf{y}}(t) \right\|^2 = \sum_{i=1}^{c} F_i(t) + \sum_{i=1}^{c} \left\| \mathbf{y}_i(t) - \bar{\mathbf{y}}(t) \right\|^2$$

where $S(t)$ remains constant as far as the same data sample is considered, and $\bar{\mathbf{y}}(t)$ denotes the centroid of the entire data sample $\aleph(t) = \{\mathbf{x}_1(t),...,\mathbf{x}_n(t)\}$ considered at time t. What we expect of the Evolution Strategy is that it will implicitly react through the above equation balancing the minimization of the population fitness, from the local optimization of individual cluster representatives, and the maximization of the between cluster scattering. This justifies our working hypothesis that the local optimization of individual cluster distortions will eventually lead to the global optimization of the entire set of cluster centres.

Our theoretical mutation operator is a random perturbation that follows a normal distribution of zero mean and whose covariance matrix is estimated from the data in the cluster associated with the individual to be mutated. There are three design questions to answer at this point: (1) Which individuals will be mutated? (2) How many mutations will be allowed? and (3) what information will be used to compute mutations?. Our proposed Evolution Strategy performs a guided selection of the individuals subjected to mutation. The set of mutated parents is composed of the individuals whose local distortion is greater than the mean of the local distortions in its generation . More formally:

$$S(t) = \left\{ i \middle| F_i(t) \geq \bar{F}(t) \right\}$$

As to the number of mutations we have decided to approach as much as possible to a fixed number of mutations *m*, so that the number of mutations per individual $m_i(t)$ will depend on the size of S(t), $m_i(t) = \lceil m / |S(t)| \rceil$. Regarding the information used to generate mutated individuals, we have decided to use the local covariance matrices of the sample partition associated with each individual. We apply a deterministic approximation to the theoretical random mutation operator in order to avoid the variability introduced by random sampling. Mutations are computed along the axes defined by the eigenvectors of the estimated local covariance matrix:

$$\hat{\Sigma}_i(t) = (n-1)^{-1}\sum_{j-1}^{n}\left(\mathbf{x}_j(t)-\mathbf{y}_i(t)\right)\left(\mathbf{x}_j(t)-\mathbf{y}_i(t)\right)^t \delta_{ij}(t)$$

Let $\Lambda_i = \operatorname{diag}\left(\lambda_{ij}, j = 1..3\right)$ and $\Phi_i = \left[e_{ij}, j = 1..3\right]$ denote, respectively, the eigenvalue and eigenvector matrices of $\hat{\Sigma}_i(t)$. Then the set of mutations generated along the axis of $\mathbf{e}_{ij}$ is:

$$P''_{ij}(t) = \left\{\mathbf{y}_i \pm \alpha_k \lambda_{ij} eij \,\middle|\, k = 1..m_{ij}(t), i \in S(t)\right\}$$

$$m_{ij}(t) = \operatorname{round}\left(m_i(t)\lambda_{ij} \Big/ 2\sum_{l=1}^{3}\lambda_{il}\right),\ \alpha_k = 1.96k \big/ m_{ij}(t)$$

The set of individuals generated by mutation is

$$P''(t) = \bigcup_{i,j} P''_{ij}(t)$$

Finally, to define the selection of the next generation individuals we pool together parents and children: $Q = P(t)$ Let $m' = \left|P''(t)\right|$ be the number of individuals effectively generated by mutation. The fitness function used for selection of an individual is the distortion when the sample is codified with the codebook given by $P''(t) \cup Q - \{y_i\}$, more formally:

$$F_i^s(t) = \sum_{k=1;i\neq k}^{c+m'}\sum_{j=1}^{n}\left\|\mathbf{x}_j(t)-\mathbf{y}_k(t)\right\|^2 \delta_{kj}^s(t)$$

$$\delta_{kj}^s(t) = \begin{cases} 1 & k = \underset{l=1,..,c+m';\, l\neq i}{\arg\min}\left\{\left\|\mathbf{x}_j(t)-\mathbf{y}_l(t)\right\|^2\right\} \\ 0 & \text{otherwise} \end{cases}$$

The selection operator selects the c best individuals according to the above fitness:

$$P(t+1) = \operatorname{select}\left(P''(t) \cup Q\right) = \left\{\mathbf{y}_i \in P^*(t);\ i = 1..c\right\}$$

$$P^*(t) = \left\{\mathbf{y}_{i_1}, .., \mathbf{y}_{i_{c+m'}} \,\middle|\, i_j < i_k \Rightarrow F_{i_j}^s(t) > F_{i_k}^s(t)\right\}$$

This definition of selection involves the fitness of the whole population with the addition of the mutations generated. This makes the algorithm sensitive to the number of mutations generated. A large number of mutations decrease the discriminatory power of $F_i^S(t)$. The number of allowed mutations must be carefully chosen. Our decision has been to allow as many mutations as individuals in the population.

4 COMPETITIVE NEURAL NETWORKS

Competitive Neural Network algorithms are derived to solve the Clustering problem as adaptive algorithms that perform stochastic gradient descent on a distortion like criterium function [4, 5]. The simplest algorithm, that we will call *Simple Competitive Learning (SCL)*, can be stated as:

$$\mathbf{y}_i(\tau+1) = \mathbf{y}_i(\tau) + \alpha_i(\tau)\,\delta_i(\mathbf{x}(\tau), \mathbf{Y}(\tau))\left[\mathbf{x}(\tau) - \mathbf{y}_i(\tau)\right] \qquad ;\mathbf{x}(\tau) \in \aleph\ ;\ i = 1,..,c$$

$$\delta_i(\mathbf{x}, \mathbf{Y}) = \begin{cases} 1 & i = \underset{k=1..c}{arg\,min}\left\{\left\|\mathbf{x} - \mathbf{y}_k\right\|^2\right\} \\ 0 & otherwise \end{cases}$$

Where $\mathbf{Y} = \{\mathbf{y}_1,..,\mathbf{y}_c\}$ is the set of cluster representatives and τ is the adaptation step. This expression is derived as the stochastic gradient search for the minimum distortion, that corresponds to the instantaneous distortion in the framework of non-stationary Clustering

$$\min_{\mathbf{Y}} \sum_{j=1}^{n}\sum_{i=1}^{c}\left\|\mathbf{x}_j - \mathbf{y}_i\right\|^2 \delta_{ij}$$

The $\alpha_i(\tau)$ denotes the (local) learning rate, in order to guarantee theoretical convergence the learning rate must cope with the conditions $\lim_{\tau\to\infty} \alpha(\tau) = 0$, $\sum_{\tau=0}^{\infty}\alpha(\tau) = \infty$, and $\sum_{\tau=0}^{\infty}\alpha^2(\tau) < \infty$. These conditions imply very lengthy adaptation processes, so that in practice they are often overlooked. In the experiments below, the learning rate follows the expression: $\alpha_i(\tau) = 0.1(1 - \tau_i/n)$ where $\tau_i = \sum_{k=1}^{\tau}\delta_i(\mathbf{x}(k))$ for the Simple Competitive

Another interesting competitive rule, often called *Soft Competition* [6, 7], can be derived from a parametrical approach to the clustering problem. Let us consider $\aleph = \{\mathbf{x}_1,..,\mathbf{x}_n\}$ as a sample of a random vector **X**, and let us consider the hypothesis of its probability density function being a mixture of Gaussian densities: $P_{\mathbf{X}}(\mathbf{x}) = \sum_{i=1}^{c} p(\omega_i)\Psi_{\mathbf{x}}(\mu_i, \Sigma_i)$ where $p(\omega_j)$ are the a priori probabilities of the classes (clusters) and $\Psi_{\mathbf{x}}(\mu_i, \Sigma_i)$ denotes a Gaussian density with mean μ_i and variance-covariance matrix Σ_i that models the conditional density $p(\mathbf{x}|\omega_i)$. Then, the clustering problem becomes a search for the parameters (μ_i ,Σ_i) that provide the best fit to the empirical distribution $\left\{P^*_{\mathbf{X}}(\mathbf{x}_i)\right\}$ computed from the sample. This search can be performed as an stochastic gradient minimization of the Kullback-Leiber cross-entropy: $C_{KL} = \sum_{i=1}^{n} P_{\mathbf{X}}^{*}(\mathbf{x}_i) log\left(P_{\mathbf{X}}^{*}(\mathbf{x}_i) / P_{\mathbf{X}}(\mathbf{x})\right)$ In the simplest case, when the covariance matrices are of the form $\Sigma_i = \sigma_i^2 \mathbf{I}$ the adaptation rule derived as the stochastic gradient descent of this measure has the shape of a Competitive Neural Network with a normalized Gaussian as the neighboring function:

$$\mathbf{y}_i(\tau+1) = \mathbf{y}_i(\tau) + \alpha_i(\tau)\, \vartheta_i\left(\mathbf{x}(\tau), \mathbf{Y}(\tau), \boldsymbol{\sigma}(\tau)\right) \left[\mathbf{x}(\tau) - \mathbf{y}_i(\tau)\right] \quad \mathbf{x}(\tau) \in \aleph; i = 1..c$$

$$\vartheta_i(\mathbf{x}, \mathbf{Y}, \boldsymbol{\sigma}) = \frac{\Psi_{\mathbf{x}}\left(\mathbf{y}_i, \sigma_i^2 \mathbf{I}\right)}{\sigma_i^2 \sum_{j=1}^{c} \Psi_{\mathbf{x}}\left(\mathbf{y}_j, \sigma_j^2 \mathbf{I}\right)}$$

An interesting feature of the Soft-Competition approach is that, applying the same reasoning, an adaptive rule can be derived equally for the assumed variance around the codevectors:

$$\sigma_i(\tau+1) = \sigma_i(\tau) + \beta_i(\tau)\, \vartheta_i\left(\mathbf{x}(\tau), \mathbf{Y}(\tau), \boldsymbol{\sigma}(\tau)\right) \frac{\left\|\mathbf{x}(\tau) - \mathbf{y}_i(\tau)\right\|^2 - d\sigma_i^2(\tau)}{\sigma_i^3(\tau)} \quad 1 \le i \le c$$

The joint application of the adaptive rules for the cluster representatives and the variances is what we have called Soft-Competition in the experiments reported below. We have found that this algorithm is very sensitive to the learning rate β in the adaptation rule for the variances. The best empirical results were found for $\beta_i(\tau) = 10^{-7} \alpha_i(\tau)$.

5 EXPERIMENTAL RESULTS

The sequence of images used for the experiment is a panning of the laboratory taken with an electronic camera. Original images have an spatial resolution of 480x640 pixels. Each two consecutive images overlap 50% of the scene. In figure 1 we represent each image in the experimental sequence as a set of points in the RGB unit cube. Each point corresponds to a pixel in the image, and the point coordinates are given by the pixel color components in the RGB color representation. This figure illustrates the unpredictable time varying nature of the pixel color population that justifies the categorization of the problem as Non-stationary Clustering. Most of the works dealing with image sequences are performed on the so-called “talking heads” that consist of recordings of the face of a talking person. These image sequences show very little, if any, variation of the color distribution and, despite their dynamic nature, their Color Qunatization is better categorized as an Stationary Clustering problem. The experimental data represented in figure 1 has been carefully designed to show a variability not found in “talking heads” image sequences.

As a bechmark non adaptive Clustering algorithm we have used a variation of the one proposed by Heckbert [11] as implemented in MATLAB following [22]. This algorithm partitions the RGB cube using an exhaustive minimum variance search. It is almost optimal, and its complexity is proportional to the discretization of the color space [22]. It has been applied to the entire images in the sequence in two ways. Figure 2 shows the distortion results of the Color Quantization of the experimental sequence to 16 and 256 colors based on both applications of the Heckbert algorithm. The curves denoted *Time Varying Min Var* are produced assuming the non-stationary nature of the data and applying the algorithm to each image independently. The curves denoted *Time Invariant Min Var* come from the assumption of stationarity of the data: the color representatives obtained for the first image are used for the Color Quantization of the remaining images in the sequence. The gap between those curves gives an indication of the non stationarity of the data. Also this gap defines the response space left for truly adaptive algorithms. To accept an algorithm as an adaptive solution its response could not be worse than the *Time Invariant Min Var* curve. The *Time Varying Min Var* defines the best response that we expect. These two curves are shown in the remaing figures as the reference responses that give an indication of the quality of the results obained with the adaptive algorithms.

In the experiments reported in this paper we have used samples of 1600 pixels to perform the adaptive computations, and, unless stated otherwise, the distortion results correspond to the Color Quantization of the whole images. We have selected the task of Color Quantization to 16 colors as representative of the general class of image segmentation tasks based on the color information. Color Quantization to 256 colors is representative of compression tasks. The experimentation with these two number of color representatives shows that the algorithms are sensitive to the number of clusters searched. As a general inspection of figures 3 to 6 will confirm, the qualitative performance of the algorithms (their error relative to the optimal application of the Heckbert algorithm) decreases as the number of clusters searched increases. This result must be hold in mind when trying to design adaptive algorithms that look for the *natural* number of clusters (color representatives).

The first set of results refer to the application of the Evolution Strategy with the theoretical random mutation operator. These results are shown in figure 3, and consist of the distortion of the Color Quantization of the 1600 pixels image samples. We have performed 30 replicas of the adaptive application of the Evolution Strategy. We give in the figure the average and 95% confidence interval of the results of these replicas. It can be seen that the random mutation operator introduces a high uncertainty on the quantization results. This uncertainty is greater in the images that show the greater distribution variation relative to their predecessor in the sequence. It can be also appreciated that the confidence intervals are more large in the case of 16 colors than in the case of 256 colors.

The random mutation operator produces some very bad results, sometimes much worse than the *Time Invariant* applicaton of the Heckbert algorithm. That is, the random mutation operator gives a significative probablity of having responses well far from the desired adaptive one. We propose the deterministic formulation of the mutation operator to avoid this uncertainty. The results of the application of the Evolution Strategy with the deterministic mutation operator on the experimental sequence are shown in figure 4 given by the curves of asterisks (*). Also shown in the figure are the results of the best replica found with the application of the random mutation operator, denoted by the curve of zeroes (o). In this and subsequent figures, the distortion results refer to the Color Quantization of the whole images. The figure shows that the deterministic operator gives a good approximation while reducing greatly the computational requirements. The Evolution Strategy with the deterinistic mutation operator performs adaptively almost all the time. As can be expected from the one generation schedule, it is not able to adapt to very big variations in the pixel distributions, such as it is the case in the transition from images #10 to #11. However it shows a quick recover after this sudden transition of distributions.

Figure 5 shows the results for the Competitive Neural Networks. They also show a good adaptive behavior. Surprisingly the Simple Competitive Learning algorithm seems to perform better than the Soft Competition. The difference between both algorithms decrease as the number of clusters searched increase, suggesting that the bad response in fig 5a of the Soft Competition is due to the fact that the empirical distribution is badly adjusted by a mixture of Gaussians in this case.

Finally, figure 6 compares the results obtained with the Simple Competitive Learning (SCL) and the Evolution Strategy (ES) with a deterministic mutation operator. For the case of 16 colors it can be seen that their behavior is quite similar. However the Evolutionary Strategy shows a quicker recover after sudden changes, improving over the Simple Competitive Learning after them (frames #12 and #13). The response of the SCL is smoother and has a kind of momentum that gives a slower but better recovery (frames #14 and #15). In the case of the 256 colors the responses are similar. Both approaches seem to be sensitive to the quality of the response (relative to that of the Heckbert algorithm) when the number of color representatives (clusters) increases. Both of them seem to behave adaptively most of the time, if the population changes are smooth enough.

6 CONCLUSIONS AND FURTHER WORK

We have proposed an Evolution Strategy for the adaptive computation of color representatives for Color Quantization that can be very efficiently implemented and reach almost real time performance for highly variable color populations. We have tested it on an experimental sequence of images. We have also tested two Competitive Neural Network algorithms against this data. Some general conclussions can be drawn from our experiments. The first is that the algorithms tested perform as desired. They profit on the previous time solutions to compute fast adaptations to the present time data. The second is the sensitivity of the adaptive algorithms to the number of clusters or color representatives searched. This sensitivity is demonstrated by the relative degradation (in front of the optimal application of the Heckbert algorithm) of the responses. This sensitivity must be taken into account when trying to design adaptive algorithms that look for the *natural* number of color representatives.

We have been able to propose a deterministic mutation operator that retain the adaptive nature of the Evolution Strategy proposed, while avoiding the uncertainty introduced by the random mutation operator. Our Evolution Strategy performance is comparable to that of some well known Competitive Neural Networks, validating it as an appropriate adaptive algorithm.

We are currently working on improving the Evolution Strategy looking for alternative definitions of the fitness function used in the selection operator, that could be as fast in their implementation as the currently used and give more optimal results. Also we are applying other Competitive Neural Networks architectures, such as the Kohonen Self Organizing Map to our data. We think that an interesting feature of our Evolution Strategy is that it can be easily extended to the search for the natural number of clusters through small modifications of the selection operator. However, these modifications must take into account the above mentioned sensitivity to the number of clusters. A sensible approach would be to include the desired order of magnitude of the number of color representatives in the selection operator.

REFERENCES

[1] T. Back, H.P. Schwefel "An overview of Evolutionary Algorithms for parameter optimization" *Evolutionary Computation* 1 pp.1-24 (1993)

[2] T. Back, H.P. Schwefel "Evolutionary computation: an overview" *IEEE ICEC'96*. pp.20-29 (1996)

[3] Z. Michalewicz "Evolutionary computation: practical issues" *IEEE ICEC'96*. pp.30-39 (1996)

[4] J.A. Hertz, A.S. Krogh, R.G. Palmer "Introduction to the theory of neural computation" Addison Wesly 1991

[5] B. Kosko "Stochastic compentitive learning" *IEEE trans Neural Networks* 2 pp522-529 (1991)

[6] Yair E., K. Zeger, A. Gersho "Competitive Learning and Soft Competition for Vector Quantization" *IEEE Trans. Sign. Proc.* 40(2) pp.294-308 (1992)

[7] Tomassini, L. "Apprentissage d´une representation statistique et topologique d´un environment" PhD Thesis, Ecole Nationale Superieure de l´Aeronautique et de l´Espace. (1992)

[8] A. Gersho, R.M. Gray "Vector Quantization and signal compression" *Kluwer Acad. Pub.* (1992)

[9] M. Goldberg, H. Sun "Image sequence coding using vector quantization" *IEEE trans. Communications* 34 pp703-710 (1986)

[10] J.H. Holland "Adaptation in Natural and Artificial Systems" *The Univ. of Michigan Press* 1975

[11] P. Heckbert "Color image quantization for frame-buffer display" *Computer Graphics* 16(3) pp.297-307 (1980)

[12] M.T. Orchard, C.A. Bouman "Color quantization of images" *IEEE trans. Signal Processing* 39(12) pp.2677-2690 (1991)

[13] T.S. Lin, L.W. Chang "Fast color image quantization with error diffusion and morphological operations" *Signal Processing* 43 pp.293-303 (1995)

[14] T. Uchiyama, M. A. Arbib "Color image segmentation using competitive learning" *IEEE . Patt. Anal. and Mach. Int.* 16(12) pp.1197-1206 (1994)

[15] M.S. Kankanhalli, B.M. Mehtre, J.K. Wu "Cluster based color matching for image retrieval" *Pattern Recognition* 29(4) pp.701-708 1996

[16] Y. Gong, H. Zen, Y. Ohsawa, M. Sakauchi "A color video image quantization method with stable and efficient color selection capability" *Int. Conf. Pattern Recognition* 1992 vol 3 pp.33-36

17] J. Hartigan "Clustering Algorithms" *Wiley* 1975

[18] E. Diday, J.C. Simon, "Clustering Analysis" In K.S. Fu (ed) *Digital Pattern Recognition* pp.47-94 Springer Verlag 1980

[19] R.D. Duda, P.E. Hart "Pattern Classification and Scene Analysis", Wiley 1973

[20] A. K. Jain, R.C. Dubes "Algorithms for clustering data" Prentice Hall 1988

[21] K. Fukunaga "Statistical Pattern Recognition" Academic Press 1990

[22] X. Wu "Efficient Statistical Computations for Optimal Color Quantization" In J. Arvo (ed) *Graphics Gems II* Academic Press Professional (1991) pp.126-133

[23] A. I. Gonzalez, M. Graña, A. d'Anjou, F.X. Albizuri "A near real-time evolutive strategy for adaptive Color Quantization of image sequences" *Joint Conference Information Sciences* 1997, Vol 1, pp. 69-72.

[24] T. Bäck, U. Hammel, H.P Schwefel, “Evolutionary Computation: comments on the history and current state” IEEE trans. Evol. Comp.1997 1(1) pp3-17

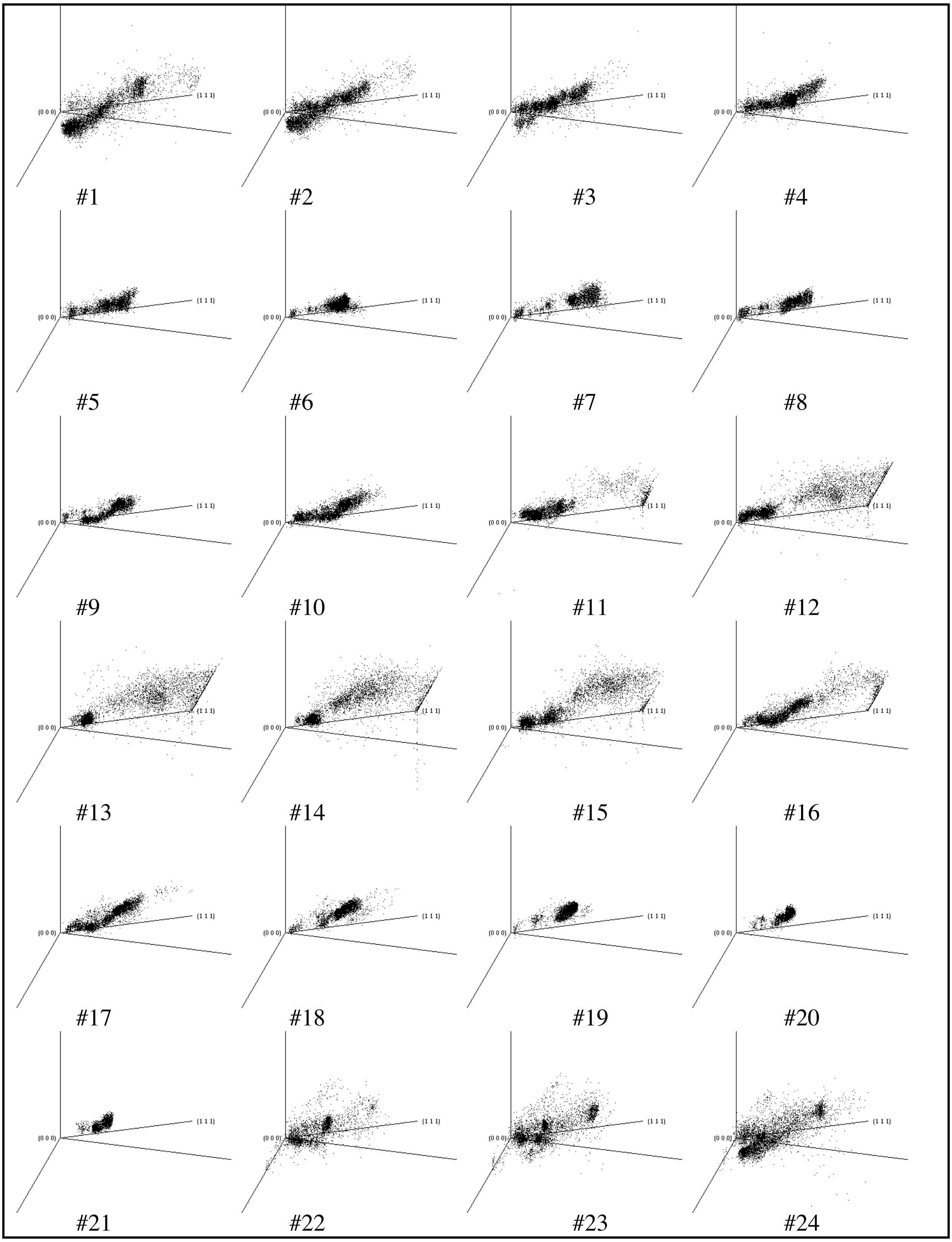
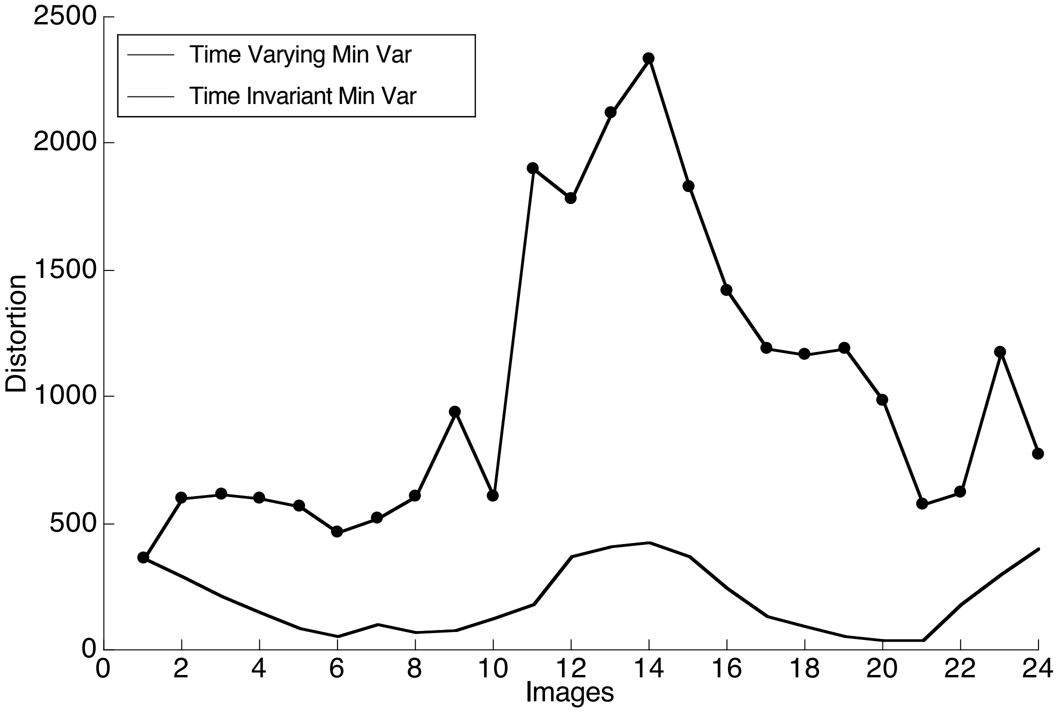
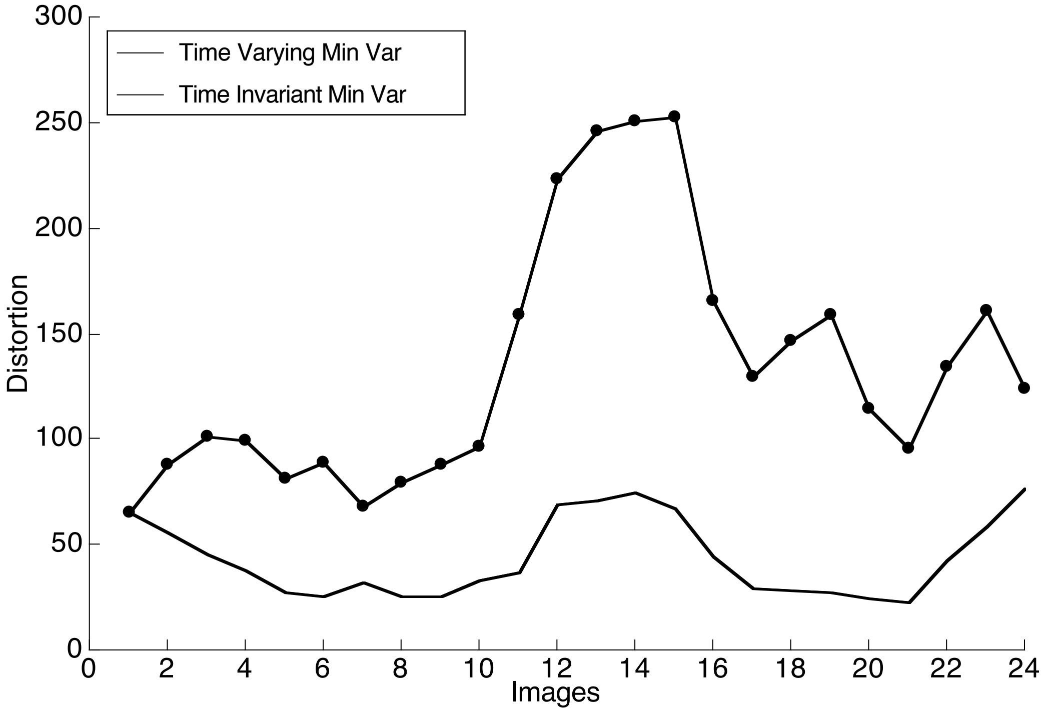


Figure 1 Representation of the pixels of the images in experimental sequence as points inside the RGB unit cube. The sequence shows a smooth but unpredictable variation of the distribution of the pixel colors that illustrates the case of Non-Stationary Clustering.

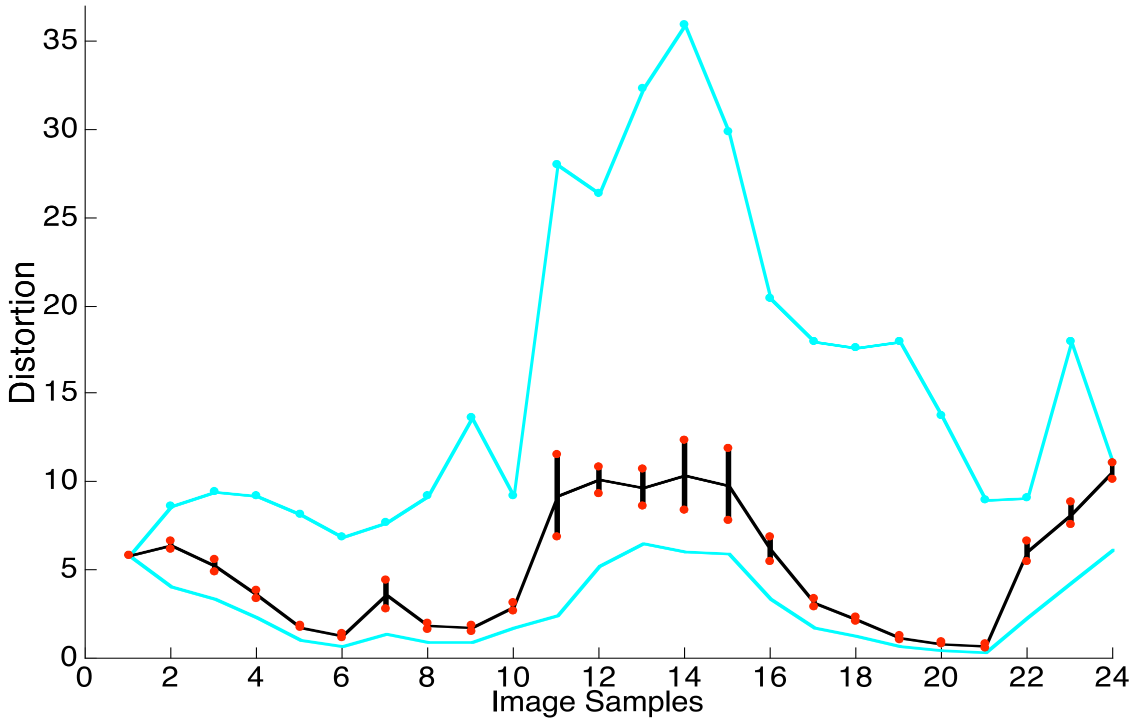


(a)

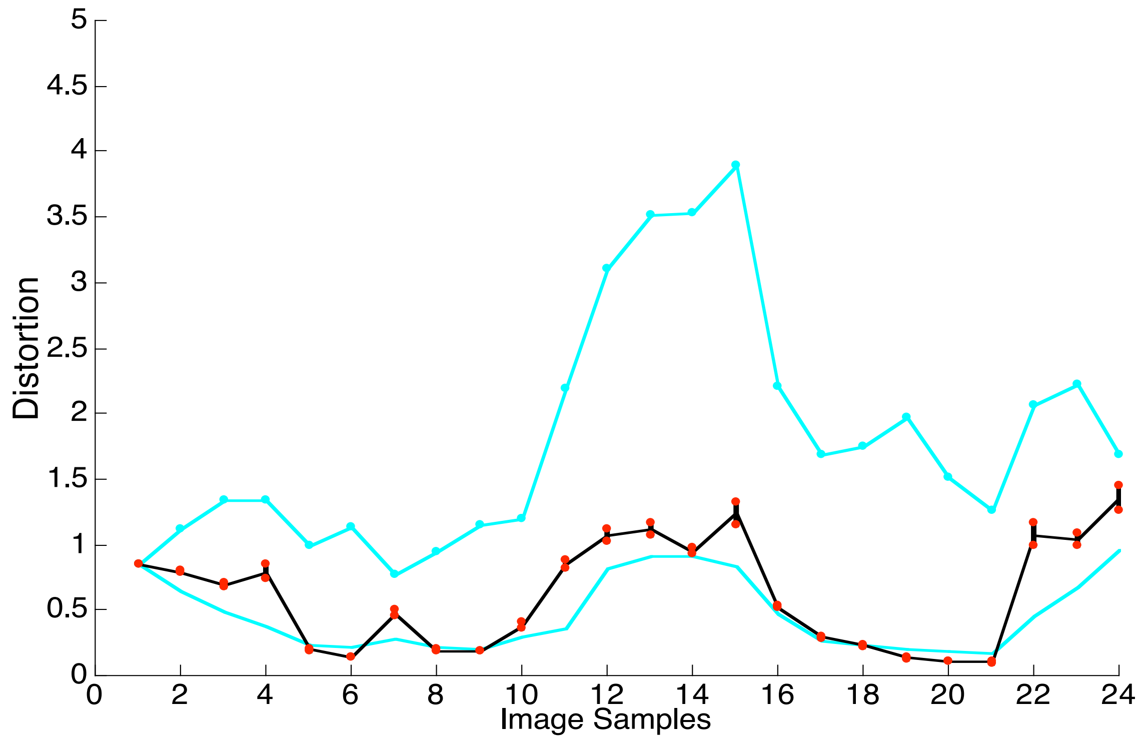


(b)

Figure 2. Benchmark distortion values obtained with the application of the Matlab implementation of the Heckbert algorithm to compute the color quantizers of (a) 16 and (b) 256 colors of the images in the experimental sequence.

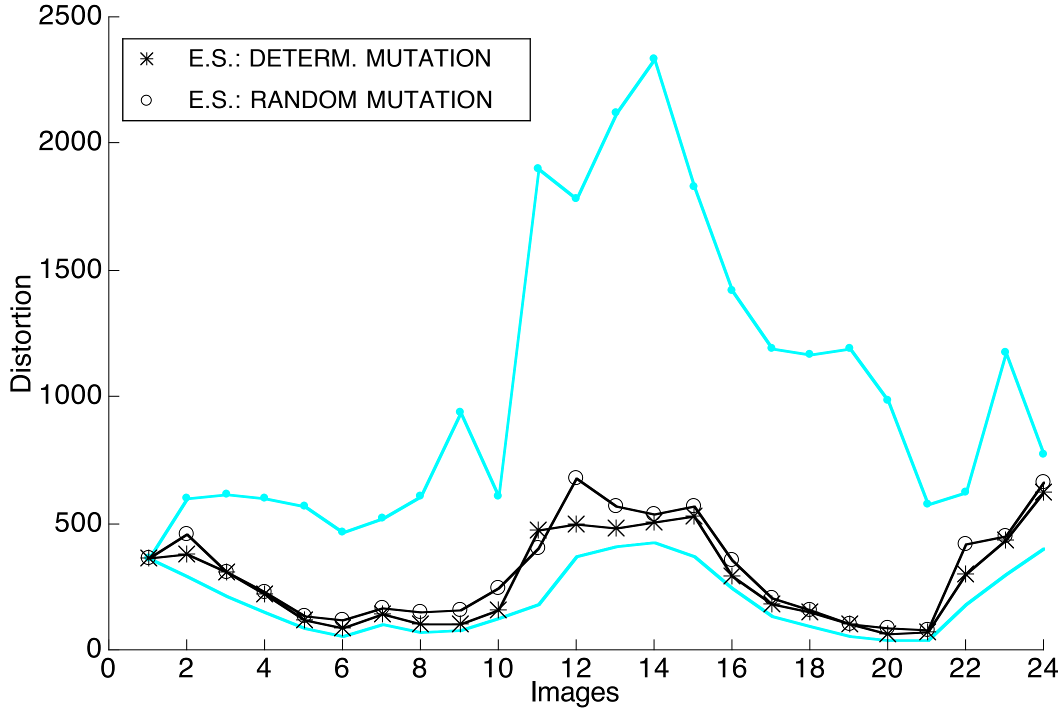


(a)

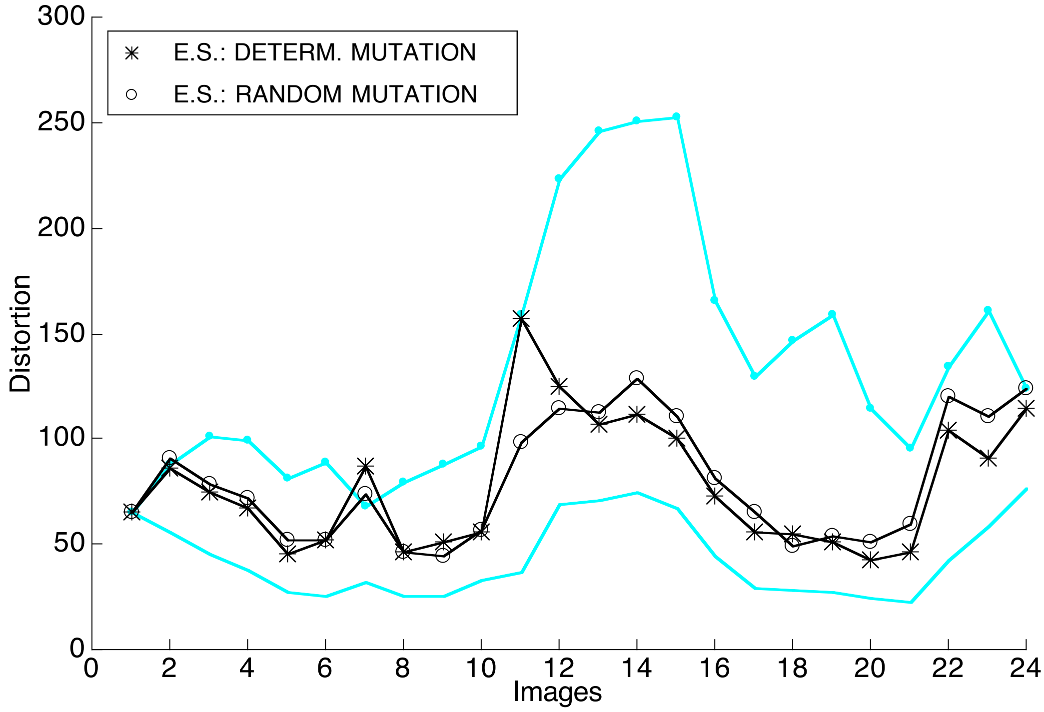


(b)

Figure 3: Mean distortion results and 95% confidence intervals of the application of the Evolution Strategy with the random mutation operator upon image samples of size n=1600 (a) with c=16, m=16,. (b) with c=256, m=256,

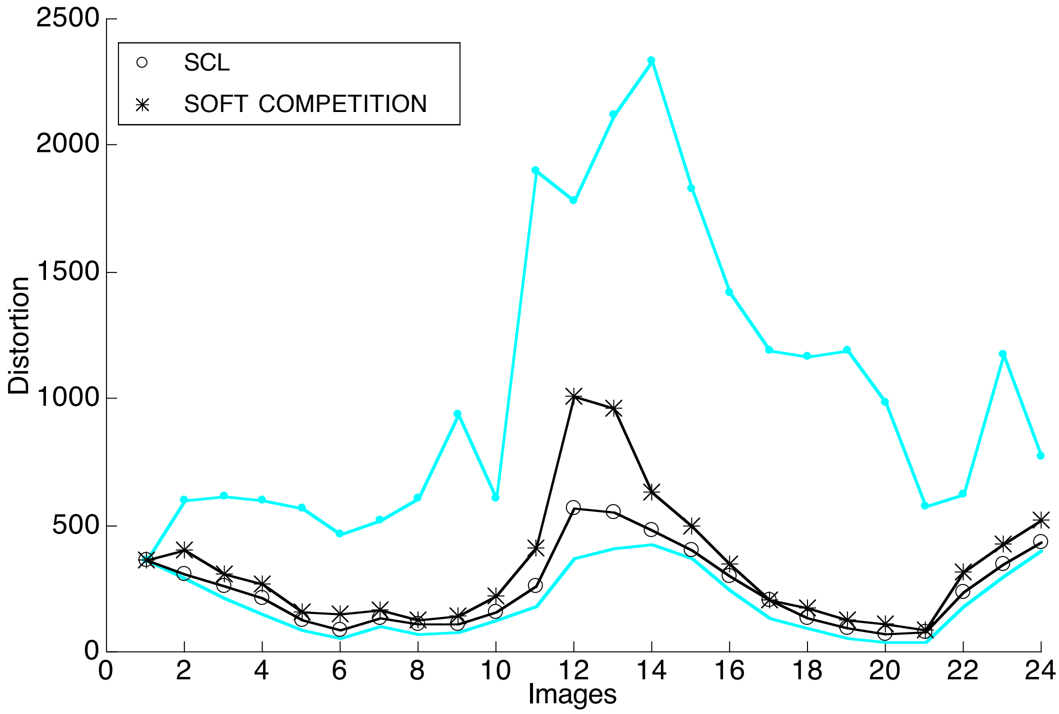


(a)

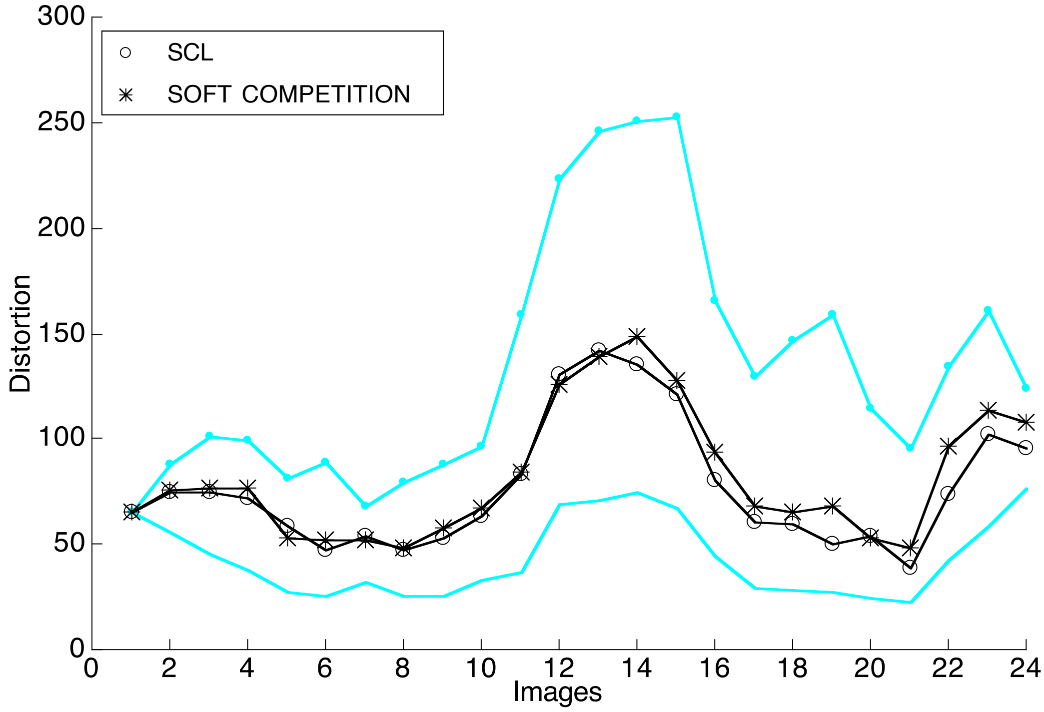


(b)

Figure 4. Distortion results on the image sequence from the Color Representatives computed by the Evolution Strategy with the best coddebooks found after 30 replica of the application using the random mutation operator, and the ones found with the deterministic operator. (a) c=16 and (b) c=256

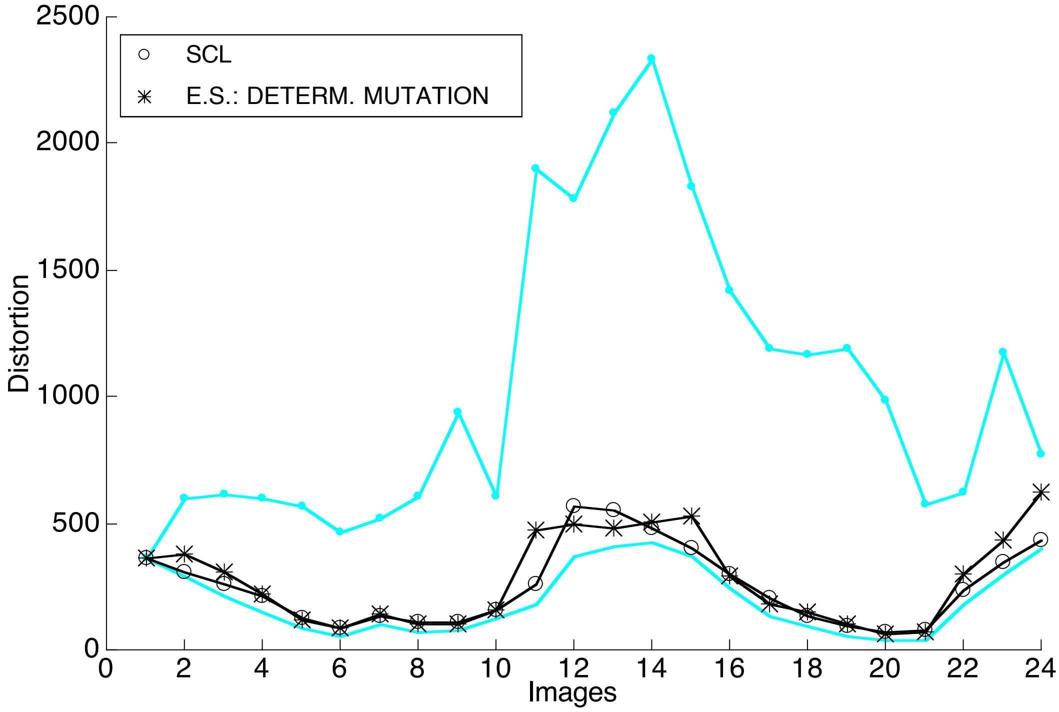


(a)

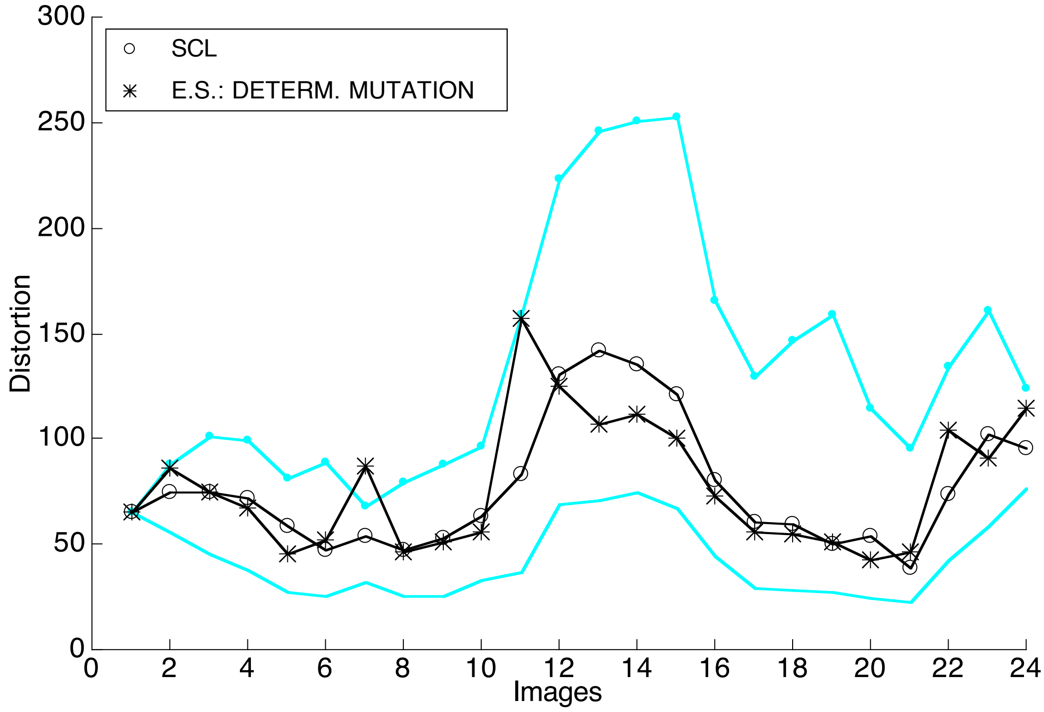


(b)

Figure 5 . Distortion results on the image sequence from the Color Representatives computed by the Simple Competitive Learning (SCL) and Soft Competition over image samples of size n=1600. (a) c=16 and (b) c=256



(a)



(b)

Figure 6. Distortion results on the image sequence from the Color Representatives computed by the Simple Competitive Learning (SCL) and the Evolution Strategy with a deterministic mutation operator over image samples of size n=1600. (a) c=16 and (b) c=256