Color VQ-based Image Compression by Manifold Learning

Christophe Charrier, Olivier Lezoray

▶ **To cite this version:**

Christophe Charrier, Olivier Lezoray. Color VQ-based Image Compression by Manifold Learning. International Conference on Image and Signal Processing, 2010, Trois-Rivières, Canada. pp.79-85, 10.1007/978-3-642-13681-8_10 . hal-00521069

HAL Id: hal-00521069

https://hal.science/hal-00521069

Submitted on 13 Feb 2014

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Color VQ-Based Image Compression by Manifold Learning

Christophe Charrier[1,2] and Olivier Lézoray[1]

[1] Université de Caen Basse-Normandie
Laboratoire GREYC, Unité Mixte de Recherche CNRS 6072
6 Bd. Maréchal Juin, 14050 Caen, France
[2] Université de Sherbrooke
Dept. d'informatique, laboratoire MOIVRE
2500 Bd. de l'Université, Sherbrooke, Qc., J1K 2R1, Canada

Abstract. When the amount of color data is reduced in a lossy compression scheme, the question of the use of a color distance is crucial, since no total order exists in $\mathbb{R}^n, n > 1$. Yet, all existing color distance formulae have severe application limitation, even if they are widely used, and not necesseraly within the initial context they have been developed for. In this paper, a manifold learning approach is applied to reduce the dimension of data in a Vector Quantization approach to obtain data expressed in $\mathbb{R}$. Three different techniques are applied before construct the codebook. Comparaisons with the standard LBG-based VQ method are performed to judge the performance of the proposed approach using PSNR, MS-SSIM and VSNR measures.

1 Introduction

Compression is a commonly used process to reduce the amount of initial data to be stored or transmited by a channel to a receiver. To reach this reduction goal, two compression families exist : 1) lossless compression approach (usually entropy-based schemes) and 2) lossy compression techniques. In those latters, compression operates as a nonlinear and noninvertible operation and is applied on individual pixel (scalar quantization) or group of pixels (Vector Quantization – VQ). VQ corresponds to the coding structure developed by Shannon in his theoretical development of source coding [1]. Conceptually, it is an extension of scalar quantization to a multidimensional space.

In VQ, the input image is parsed into a sequence of groups of pixels, referred to as *input vectors*. This sequence is termed the *test set*, and the *training set* corresponds to a set of training images. VQ maps a vector $\boldsymbol{x}$ of dimension k to another vector $\boldsymbol{y}$ of dimension k belonging to a finite set $\mathcal{C}$ (*codebook*) containing n output vectors, called *code vectors* or *codewords*.

When one tries to reduce the amount of data in a multidimensional space, the question of the used distance to measure existing difference between two candidate vectors is crucial. Actually no total order exists in spaces whose dimension is greater than 1. For example, within the color domain, many distance formulae

© Springer-Verlag Berlin Heidelberg 2010

have been introduce to counterbalance this main drawback. Furthermore, those formulae tend to measure the color perception as done by the Human Visual System. For now, all distances fails to measure the distance between two vectors, specially for chromatic data, represented by a 3D vector (C_1, C_2, C_3). In addition, when one tries to apply VQ techniques on 3D data (*i.e.*, colorimetric data), one has to measure the distance betweeen multidimensional color vectors to construct the final codebook. In that case, no satisfactory distance formulae are available to be apply on those data. Instead of developping a dedicated distance formula, an investigation about reduction dimensionnality methods is performed. The aim of such an approach is to be able applying Euclidean distance on scalar obtained after performind a dimensionnality reduction process.

In this paper, a manifold learning process is applied prior a VQ compression scheme in order to reduce the data dimensionality to generate the codebook. Comparison with standard VQ scheme is performed to measure the effeciency of the proposed approach.

2 Dimensionality Reduction

Given a set of visual features describing an image, a Manifold Learning method is used to project the data onto a new low-dimensional space. Thus, nonlinear new discriminant features of the input data are yielded. The obtained low dimensional sub-manifold is used as a new representation that is transmitted to classifiers. When data objects, that are the subject of analysis using machine learning techniques, are described by a large number of features (i.e. the data is high dimensional) it is often beneficial to reduce the dimension of the data. Dimensionality reduction can be beneficial not only for reasons of computational efficiency but also because it can improve the accuracy of the analysis. Indeed, traditional algorithms used in machine learning and pattern recognition applications are often susceptible to the well-known problem of the curse of dimensionality, that refers to the degradation in the performance of a given learning algorithm as the number of features increases. To deal with this issue, dimensionality reduction techniques are often applied as a data pre-processing step or as part of the data analysis to simplify the data model. This typically involves the identification of a suitable low-dimensional representation of the original high-dimensional data set. By working with this reduced representation, tasks such as classification or clustering can often yield more accurate and readily interpretable results, while computational costs may also be significantly reduced. Dimensionality reduction methods can be divided into two sets wether the transformation is linear or nonlinear. We detail here the principles of three well-known linear and nonlinear dimensionality reduction methods: Principal Components Analysis (PCA)[2], Laplacian Eigenmaps (LE)[3] and Manifold Parzen Window (MPW) [4]. Let $X = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n\} \in \mathbb{R}^p$ be n sample vectors. Dimensionality reduction consists in finding a new low-dimensional representation in $\mathbb{R}^p$ with $q \ll p$.

2.1 Principal Components Analysis

The main linear technique for dimensionality reduction, principal components analysis (PCA), performs a linear mapping of the data to a lower dimensional space in such a way, that the variance of the data in the low-dimensional representation is maximized. Traditionally, principal component analysis is performed on the symmetric covariance matrix C_{cov} or on the symmetric correlation matrix C_{cor}. We will denote C one of these two matrices in the sequel. From such a symmetric matrix, we can calculate an orthogonal basis by finding its eigenvalues and eigenvectors. Therefore, PCA simply consists in computing the eigenvectors and eigenvalues of the matrix C: $C = U\Lambda U^T$ where $\Lambda = diag(\lambda_1, \cdots, \lambda_n)$ is the diagonal matrix of the ordered eigenvalues $\lambda_1 \leq \cdots \leq \lambda_n$, and U is a $p \times p$ orthogonal matrix containing the eigenvectors. Dimensionality reduction is then obtained by the following operator $h_{PCA} : \mathbf{x}_i \rightarrow (y_1(i), \cdots, y_q(i))$ where $y_k(i)$ is the i^{th} coordinate of eigenvector $\mathbf{y}_k$. In the rest of this paper, we will denote h^{Cov}_{PCA} and h^{Cor}_{PCA}, dimensionality reduction performed with PCA of the covariance or the correlation matrix.

2.2 Laplacian Eigenmaps

Given a neighborhood graph G associated to the vectors of X, one considers its adjacency matrix W where weights W_{ij} are given by a Gaussian kernel $W_{ij} = k(\mathbf{x}_i, \mathbf{x}_j) = e\left(-\frac{||\mathbf{x}_i - \mathbf{x}_j||^2}{\sigma^2}\right)$. Let D denote the diagonal matrix with elements $D_{ii} = \sum_j W_{ij}$ and Δ denote the un-normalized Laplacian defined by $\Delta = D - W$. Laplacian Eigenmaps dimensionality reduction consists in searching for a new representation $\{\mathbf{y}_1, \mathbf{y}_2, \cdots, \mathbf{y}_n\}$ with $\mathbf{y}_i \in \mathbb{R}^n$, obtained by minimizing $\frac{1}{2}\sum\limits_{ij} \left\|\mathbf{y}_i - \mathbf{y}_j\right\|_2 W_{ij} = Tr(\mathbf{Y}^T \Delta \mathbf{Y})$ with $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \cdots, \mathbf{y}_n]$. This cost function encourages nearby sample vectors to be mapped to nearby outputs. This is achieved by finding the eigenvectors $\mathbf{y}_1, \mathbf{y}_2, \cdots, \mathbf{y}_n$ of matrix Δ. Dimensionality reduction is obtained by considering the q lowest eigenvectors (the first eigenvector being discarded) with $q \ll p$ and is defined by the following operator $h_{LE} : \mathbf{x}_i \rightarrow (y_2(i), \cdots, y_q(i))$ where $y_k(i)$ is the i^{th} coordinate of eigenvector $\mathbf{y}_k$.

2.3 Manifold Parzen Window

Considering X, one can associate an unknown probability density function $p_X(.)$. Let a training set contain l samples of that random variable, collected in a $l \times n$ matrix X whose row x_i is the i-th sample. Then, the goal is to estimate the density $p_X(.)$. Let us consider a small region $\mathcal{R}$ centered on the point $\mathbf{x}$ at which we wish to determine the probability density. In order to count the number K of points falling within this region, a commonly used way is the use of the following function

$$k(u) = \begin{cases} 1 \; |u_i| \leq 1/2, i = 1, \ldots, D \\ 0 \text{ otherwise} \end{cases} \tag{1}$$

that represents a unit cube centered on the origin. The function $k(u)$ is known as a *Parzen window*. From this, the quantity $k(\mathbf{x}-\mathbf{x}_n)/h$ will be one if the data point $\mathbf{x}_n$ lies inside the cube, and zero otherwise. Thus the total number of data points lying inside this cube is given by

$$K = \sum_{n=1}^{N} k\left(\frac{\mathbf{x}-\mathbf{x}_n}{h}\right) \tag{2}$$

Thus, the estimate density at data point $\mathbf{x}$ is given by

$$p(\mathbf{x}) = \frac{1}{N}\sum_{n=1}^{N} \frac{1}{h^p} k\left(\frac{\mathbf{x}-\mathbf{x}_n}{h}\right) \tag{3}$$

where $V = h^p$ is the volume of the cube of side h in p dimensions.

Nevertheless, this kernel density estimator suffers from one of the same problems encountered using the histogram method, namely the presence of artificial discontinuities at the cube boundaries.

To prevent this, one uses a smoother kernel function based on a Gaussian kernel defined as follows:

$$p(\mathbf{x}) = \frac{1}{N}\sum_{n=1}^{N} \frac{1}{\sqrt{2\pi h^2}} \exp\left(-\frac{\|\mathbf{x}-\mathbf{x}_n\|^2}{2h^2}\right) \tag{4}$$

where h represents the standard deviation of the Gaussian components, *i.e.* can be interpreted as the covariance matrix C. In that case, h^2 represents the determinant $|C|$.

In order to obtain a more compact representation of the inverse Gaussian, one stores only the eigenvectors associated with the first few largest eigenvalues of C_i, as described below. The eigen-decomposition of a covariance matrix$C$ can be expressed as: $C = VDV^T$, where the columns of V are the orthonormal eigenvectors and D is a diagonal matrix with the eigenvalues $\lambda_1, \ldots, \lambda_n$, that we will suppose sorted in decreasing order, without loss of generality. The first q eigenvectors with largest eigenvalues correspond to the principal directions of the local neighborhood, i.e. the high variance local directions of the supposed underlying q-dimensional manifold (but the true underlying dimension is unknown and may actually vary across space). Dimensionality reduction is obtained by considering the q lowest eigenvectors (the first eigenvector being discarded) with $q \ll p$ and is defined by the following operator $h_{PW} : \mathbf{x}_i \to (y_2(i), \cdots, y_q(i))$ where $y_k(i)$ is the i^{th} coordinate of eigenvector $\mathbf{y}_k$.

3 Performance Measure Protocol

To analyze the performance of the proposed approach, a comparison with the standard VQ compression scheme (based on the use of the LBG algorithm [5]) is computed.

The comparison operates using computable metrics. Three computable metrics are selected: 1) the PSNR due to its commonly use in the image processing community, 2) the Image Quality Assessment (IQA) algorithm Visual Signal-to-Noise Ratio (VSNR) introduced by CHANDLER and HEMAMI [6] and 3) the IQA algorithm Multiscale Structural Similarity Index (MS-SSIM). The VSNR has been developed to uantify the visual fidelity of natural images based on near-threshold and suprathreshold properties of human vision. In [7] SHEIKH *et al.* have shown that MS-SSIM is highly competitive with all other existing IQA algorithms.

3.1 Experimental Setup

To judge how the proposed method outperforms the standard VQ algorithm, 25 initial images in the LIVE image database are used [8]. From those images, two datasets are generated: 1) 12 images are used as a training set and the 2) the 13 remaining images serve as test set.

From the training set, 12 codebooks $(C_i)_{i\in[1,\dots,12]}$ of size from 32 to 560 are generated without performing the dimension reduction approach and 12 codebooks are computed after performing each one of the three manifold learning algorithm described in section 2. The size of the used vectors to construct both the training set and the test set is 8×8. Those codebook sizes yields us to have image quality from very bad to excellent.

4 Results

Table 1 presents the obtained results using the three computed metrics to measure the quality of VQ-based reconstructed images and the manifold learning VQ-based compressed images. In this latter only the first eigenvector is used to construct the codebook at different sizes using monodimensionnal data d_m projected on that new axis. One can observe that the obtained results are slightly better that those obtained using the initial spatial vectors to construct the codebook. The differences are not really significant to claim that the proposed method definitely outperforms the standard VQ-based compressed images. Nevertheless, the construction of the codebooks applying the proposed approach has been realized only using monodimensional data d_m instead of $8 \times 8 \times 3$ color data. By the way, a simple Euclidean distance is used instead of a complex color distance. From all tested manifold learning methods, the Laplacian Eigenmaps gives better results than the two others, in terms of IQA measures.

Table 2 presents the quality measure results taking into account more and more dimensions during the construction of the codebooks. One notes that for a number of dimension greater than 3, no significant improvement of the quality is observed. This tends to prove that no more 3D data are needed to construct codebook of quality when the distance between data is measure using the Euclidean formula. By the way, we do not need to use any specific colorimetric distance to generate a codebook of quality.

Table 1. Computable metrics applied on VQ-based and each one of the manifold learning VQ-based compressed images (in that case, only data provided by the first eigenvector are used)

Codebook size	32	80	128	176	224	272	320	368	416	464	512	560
	Standard VQ											
PSNR	20.3	21.2	21.0	22.1	22.3	22.8	23.9	24.1	25.3	26.7	27.2	28.2
MS-SSIM	0.919	0.948	0.956	**0.963**	0.965	0.969	0.970	0.971	0.972	0.973	**0.974**	0.976
VSNR	1.051	0.952	0.871	0.672	**0.596**	0.518	0.445	0.431	0.392	0.331	0.298	0.221
	h_{PCA}^{Cov} and VQ											
PSNR	23.4	25.1	26.2	26.8	26.8	**27.9**	**28.1**	**28.3**	28.4	**28.6**	**28.8**	29.1
MS-SSIM	0.917	0.946	0.952	0.960	0.962	0.964	0.968	0.970	0.972	0.972	0.973	0.974
VSNR	**1.021**	0.943	0.856	0.664	0.600	0.489	0.421	0.401	**0.391**	0.330	0.289	0.213
	h_{PCA}^{Cor} and VQ											
PSNR	23.2	25.4	26.3	26.8	26.7	27.5	23.9	26.1	26.5	27.0	27.1	27.9
MS-SSIM	0.912	0.943	0.954	0.961	0.960	0.957	0.965	0.968	0.970	0.972	0.973	0.974
VSNR	1.045	0.946	**0.869**	0.678	0.610	0.489	0.433	0.412	0.392	0.331	0.291	0.220
	h_{LE} and VQ											
PSNR	**23.9**	**26.1**	**26.5**	27.0	27.1	**27.9**	**28.1**	28.2	**28.5**	**28.6**	**28.8**	**29.3**
MS-SSIM	**0.921**	**0.951**	**0.957**	**0.963**	**0.967**	**0.969**	**0.971**	**0.973**	**0.973**	**0.974**	**0.974**	**0.978**
VSNR	1.032	0.946	0.869	0.678	0.610	0.509	**0.401**	**0.398**	**0.391**	**0.328**	**0.264**	**0.208**
	h_{PW} and VQ											
PSNR	23.6	25.2	25.8	**27.7**	**27.2**	**27.5**	27.8	28.1	28.2	28.3	28.3	28.5
MS-SSIM	0.916	0.942	0.951	0.958	0.964	0.970	0.969	0.969	**0.971**	0.972	0.972	0.973
VSNR	1.043	**0.923**	0.857	**0.654**	0.602	**0.487**	0.436	0.421	0.409	0.350	0.287	0.212

Table 2. Evolution of each computable metrics applied Laplacian Eigenmaps VQ-based compressed images when the number of eigenvectors increases

Codebook size		32	80	128	176	224	272	320	368	416	464	512	560
	PSNR	23.9	26.1	26.5	27.0	27.1	27.9	28.1	28.2	28.5	28.6	28.8	29.3
λ_1	MS-SSIM	0.921	0.951	0.957	0.963	0.967	0.969	0.971	0.973	0.973	0.974	0.974	0.978
	VSNR	1.032	0.946	0.869	0.678	0.610	0.509	0.401	0.398	0.391	0.328	0.264	0.208
	PSNR	24.22	26.32	26.67	27.5	27.6	28.3	28.6	28.8	29.2	29.6	29.5	29.7
λ_2	MS-SSIM	0.923	0.953	0.960	0.965	0.968	0.972	0.973	0.974	0.973	0.976	0.974	0.980
	VSNR	1.036	0.949	0.874	0.682	0.618	0.512	0.421	0.408	0.398	0.333	0.275	0.214
	PSNR	24.33	26.41	26.70	27.8	28.0	28.4	28.9	29.2	29.5	29.8	29.7	30.2
λ_3	MS-SSIM	0.924	0.954	0.962	0.965	0.968	0.972	0.974	0.974	0.974	0.977	0.975	0.981
	VSNR	1.037	0.951	0.876	0.684	0.619	0.515	0.424	0.414	0.402	0.335	0.277	0.217
	PSNR	24.35	26.44	26.71	27.9	28.1	28.45	29.1	29.4	29.7	29.8	29.9	30.3
λ_4	MS-SSIM	0.924	0.955	0.963	0.966	0.969	0.973	0.974	0.975	0.974	0.978	0.977	0.982
	VSNR	1.037	0.952	0.876	0.686	0.621	0.516	0.427	0.416	0.404	0.338	0.281	0.219

Fig. 1 shows an example of reconstruted image when a dimension reduction process is applied prior the VQ (a) or not (b).

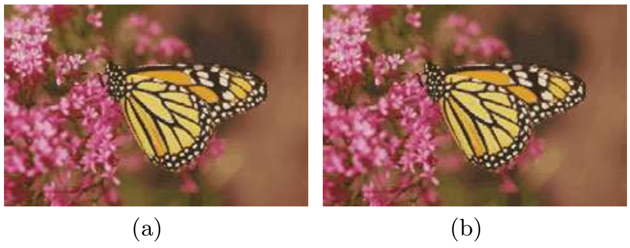

Fig. 1. Example of h_{LE} + VQ reconstructed image (a) and standard VQ reconstruted image (b) using a codebook size equal to 100

5 Conclusion

In this paper a dimensionnality reduction is applied before performing a VQ-based compression technique. Using such an approach, problems concerning the selection of a good colororimetric distance to construct the codebook is evacuated, since obtained data from manifold learning method are not color data. In that case, an Euclidean distance can be used. In a first part, three manifold learning methods have been compared to the standard VQ compression scheme in terms of PSNR, MS-SSIM and VSNR values. Laplacian Eigenmaps applied before the construction of the codebook give best results. In a second part, it has been shown that no more than 3 eigenvectors need to be use to improve the quality of the results. By the way, one can reach best quality using 3D data than initial spatial color vectors.

References

1. Gersho, A., Gray, R.M.: Vector Quantization and Signal Compression. Kluwer Academic Publishers, Dordrecht (1991)
2. Saul, L.K., Weinberger, K.O., Ham, J., Sha, F., Lee, D.D.: Spectral methods for dimensionnality reduction. In: Semi-supervised Learning, pp. 279–294. MIT Press, Cambridge (2006)
3. Belkin, M., Niyogi, P.: Laplacien eigenmaps for dimensionality reduction and data representation. Neural Computing 15(6), 1373–1396 (2003)
4. Bengio, Y., Vincent, P.: Manifold parzen windows. Tech. Rep., CIRANO (2004)
5. Linde, Y., Buzo, A., Gray, R.R.: An algorithm for vector quantizer design. IEEE Transactions on Communications 28, 84–94 (1980)
6. Chandler, D.M., Hemami, S.S.: VSNR: A wavelet-based visual signal-to-noise ratio for natural images. IEEE Transactions on Image Processing 16(9), 2284–2298 (2007)
7. Sheik, H.R., Sabir, M.F., Bovik, A.C.: A statistical evaluation of recent full reference image quality assessment algorithms. IEEE Transactions on Image Processing 5(11), 3441–3452 (2006)
8. Laboratory for Image & Video Engineering, University of Texas (Austin) LIVE Image Quality Assessment Database (2002), `http://live.ece.utexas.edu/research/Quality`