Journal of Uncertain Systems
Vol.3, No.1, pp.11-30, 2009
Online at: www.jus.org.uk

Direct and Inverse Fuzzy Transforms for Coding/Decoding Color Images in YUV Space

Ferdinando Di Martino[1,2], Vincenzo Loia[2,*], Salvatore Sessa[1]
[1]Università degli Studi di Napoli Federico II, Dipartimento di Costruzioni e Metodi Matematici in Architettura, Via Monteoliveto 3, 80134 Napoli, Italy
[2] Università degli Studi di Salerno, Dipartimento di Matematica e Informatica Via Ponte Don Melillo, 84084 Fisciano, Italy

Received 26 May 2008; Accepted 29 July 2008

Abstract

In some previous works the authors showed the advantages in coding and decoding images in the YUV space by using fuzzy relation equations. Indeed the images in the Y band were less compressed than in the U, V bands and a better Peak Signal to Noise Ratio was obtained with respect to that deduced by coding and decoding the same images in the RGB space. In another foregoing paper we used the fuzzy transform compression method for gray images and we compared the results with those ones obtained by using the fuzzy relation equations and JPEG compression methods: we concluded that the fuzzy transform method produces good results with respect to the fuzzy relation equations method under any compression rate and with respect to the DCT method (used in JPEG) for high compression rates. In this stream of investigations, here we test the fuzzy transform method for coding and decoding color images in the YUV space under high compression rates in the U, V bands. We compare the results with those ones obtained by using the fuzzy transforms and the standard JPEG compression method in the RGB space.

Keywords: fuzzy relation, fuzzy transform, PSNR, YUV space, RGB space, DCT, JPEG

1 Introduction

The YUV model defines a color space in terms of the brightness component (the Y band) and the two chrominance components (the U and V bands). The YUV color model is used in the JPEG color images compression process and in the NTSC, PAL, and SECAM composite color-video standards.

The study of the YUV space is interesting because the resolution of an image in the Y band is visible to the human eye much more than that one visible in the bands U, V while there is no difference of perception in the classical three color bands R, G, B. In [15, 16] the compression method based on fuzzy relation equations (for short, FRE) in the YUV space was applied to gray and color images. Indeed any image was divided in blocks of equal sizes and each block was coded with a low (resp. high) compression rate in the band Y (resp. U, V).

Since we also work with the standard JPEG image compression method [24] which manages color images in the YUV space, here we schematize in Figure 1 this coding/decoding process.

If the source image is represented in the RGB space, it is converted in the YUV space. In the coding process the source image is divided into blocks of sizes 8×8 and each block is transformed, via the forward Discrete Cosine Transform (for short, DCT), into a set of 64 values called DCT coefficients. Each coefficient is then transformed by using only one of the 64 corresponding values from a quantization table which is carried along with the compressed file. After this quantization, the coefficients are ordered for increasing frequency, prepared for the entropy coding process and hence converted into a one-dimensional zig-zag sequence.

In the decompression procedure each step performs essentially the inverse of its corresponding process realized during the coding procedure. Indeed the entropy decoder transforms the zig-zag sequence of the quantized DCT coefficients. After the de-quantization, the DCT coefficients are transformed in a block of sizes 8×8 with the inverse DCT.

[*] Corresponding author. Email: loia@unisa.it (V. Loia).

In this work we take advantage of the properties of the YUV space above described by using the Fuzzy TRansform (for short, FTR) method [18, 19, 20, 21, 23] for compressing images strongly in the bands U, V and softly in the band Y.

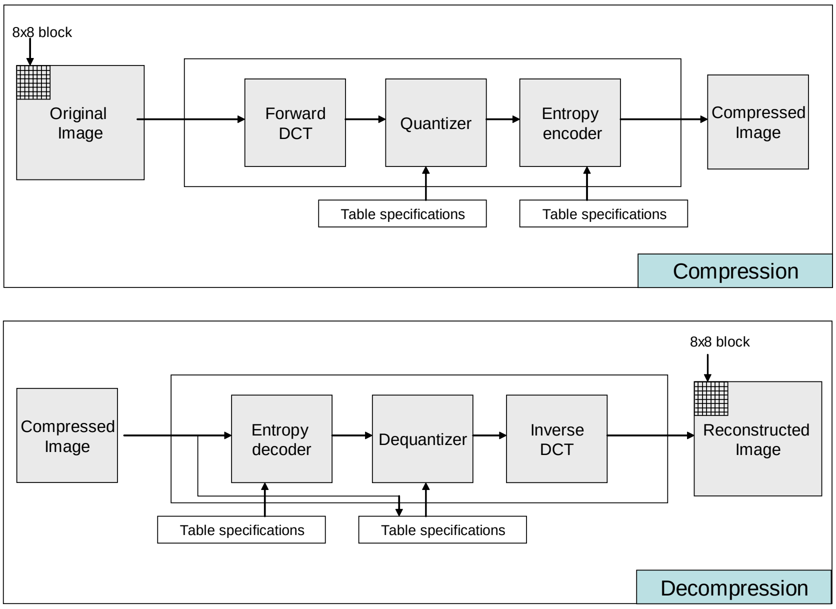

Figure 1. The JPEG coding/decoding process

An FTR [19] is an operator which transforms a continuous function over the interval [a,b] in a *n*-dimensional vector. Viceversa, an inverse FTR operator converts an *n*-dimensional vector into a continuous function which approximates the original function up to a small quantity ε. Thus it is possible to avoid complex computations since we translate the functional problem into the respective linear problem which is more simple to manipulate because one is faced with numerical vectors. By discretizing these processes, in [5,6] the authors showed that the FTR method gives better results with respect to the FRE and DCT methods and it is comparable with the coding/decoding JPEG standard method for high compression rates. In Figure 2 we show the schema of the process used for coding/decoding color images.

In the coding process we use a compression rate $\rho_U = \rho_V$ in the planes *U*, *V* and a compression rate $\rho_V > \rho_U$ in the plane *Y*. After the decompression process we obtain a decoded image with components $\tilde{Y}$, $\tilde{U}$,$\tilde{V}$ in the YUV space, converted into an image with components $\tilde{R}$, $\tilde{G}$, $\tilde{B}$ in the RGB space.

We analyze the quality of our results by evaluating the Peak Signal to Noise Ratio (for short, PSNR) obtained by using the FTR method in RGB and YUV spaces for several values of the compression rate. In the RGB space we practically assume $\rho_R = \rho_G = \rho_B$. In RGB (resp. YUV) space we define as compression rate the quantity $\rho_{RGB} = (\rho_R + \rho_G + \rho_B)/3 = \rho_R$ (resp. $\rho_{YUV} = (\rho_U + \rho_V + \rho_Y)/3 = (\rho_U + 2\cdot\rho_V)/3$). Further we assume and we operate in such a way the difference $|\rho_{RGB} - \rho_{YUV}|$ assumes a small negligible value (which achieves 0.002296 as maximum value as shown in Table 3), so that we can suppose $\rho_{RGB} \cong \rho_{YUV}$ without loss of generality. In Section 2 we recall the concepts of FTR of a function in one and in two variables in the continuous and discrete cases. In Section 3 we show how the techniques based on the discrete FTR and its inverse are used for coding/decoding processes of images. In Section 4 we describe how to convert our images from RGB to YUV space and conversely, furthermore we give the several compression rates used in our tests. In Section 5 we present the results of our tests by comparing them with the analogous ones deduced by adopting the FTR method over images in RGB space and the standard JPEG method under several compression rates. The final Section 6 contains the concluding comments.

2 FTR in One and Two Variables

Let $\{x_1, x_2,\ldots,x_n\}$ be a set of points of [a,b] such that $x_1 = a < x_2 <\ldots< x_n = b$ and $A_1,\ldots,A_n : [a,b] \rightarrow [0,1]$ be fuzzy sets forming a fuzzy partition of [a,b], that is the following conditions hold [19]:

(1) $A_i(x_i)=1$ for every i = 1,2,…,n;
(2) $A_i(x) = 0$ if $x\notin(x_{i-1}, x_{i+1})$ for each i = 2,…, n-1;
(3) $A_i(x)$ is a continuous function over [a,b];
(4) $A_i(x)$ is strictly increasing in $[x_{i-1}, x_i]$ for i = 2,…, n+1 and strictly decreasing in $[x_i, x_{i+1}]$ for i = 1,…, n-1;
(5) $\sum_{i=1}^{n} A_i(x)=1$ for every x∈[a,b].

Moreover the fuzzy partition $\{A_1,…, A_n\}$ is called uniform if
(6) n≥3 and $x_i = a+h\cdot(i-1)$, for every i = 1, 2, …, n, where h = (b-a)/(n-1);
(7) $A_i(x_i - x) = A_i(x_i + x)$ for every x ∈ [0,h] and i = 2,…, n-1;
(8) $A_{i+1}(x) = A_i(x - h)$ for every $x\in[x_i, x_{i+1}]$ and i = 1,2,…, n-1.

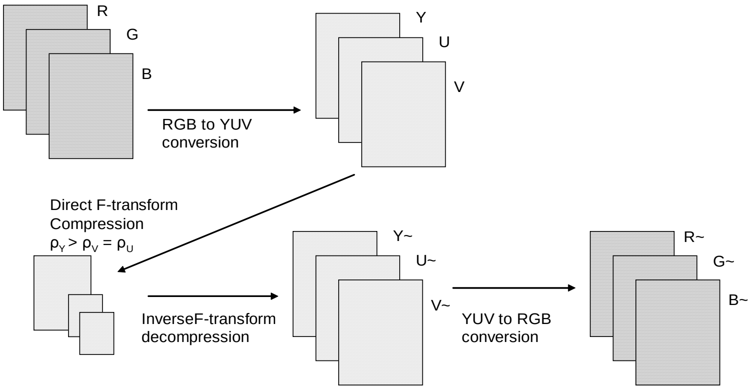

Figure 2. Schema of coding/decoding process from RGB to YUV space and conversely

Let f(x) be a continuous function over [a,b] and $\{A_1, A_2, …, A_n\}$ be a fuzzy partition of [a,b]. The *n*-tuple F = $[F_1, F_2,..., F_n]$ is called the FTR of f with respect to $\{A_1, A_2, …, A_n\}$ if the following holds for every *i* = 1,…,*n*:

$$F_i = \frac{\int_a^b f(x)A_i(x)dx}{\int_a^b A_i(x)dx} \quad . \tag{1}$$

The F_k's are called the components of the FTR of f and if $\{A_1, A_2, …, A_n\}$ is uniform, then we have that (cfr. [19, Lemma 1]):

$$F_i = \begin{cases} \frac{2}{h}\int_{x_1}^{x_2} f(x)A_1(x)dx & \text{if} \quad \mathrm{i=1} \\ \frac{1}{h}\int_{x_{i-1}}^{x_{i+1}} f(x)A_i(x)dx & \text{if} \quad \mathrm{i=2,...,n\text{-}1} \\ \frac{2}{h}\int_{x_{n-1}}^{x_n} f(x)A_n(x)dx & \text{if} \quad \mathrm{i=n.} \end{cases} \tag{2}$$

We can also define the following function $f_{F,n}$ by setting for every x∈[a,b]:

$$f_{F,n}(x) = \sum_{i=1}^{n} F_i A_i(x) \tag{3}$$

defined as the inverse FTR of f with respect to {A_1, A_2, …, A_n} and the following theorem (cfr. [19, Theorem 2]) holds.

Theorem 1. Let f(x) be a continuous function over [a,b]. For every ε > 0, then there exist an integer *n*(ε) and a related fuzzy partition {A_1, A_2, …, $A_{n(\varepsilon)}$} of [a,b] such that for all x ∈ [a, b],

$$| f(x) - f_{F,n(\varepsilon)}(x) | < \varepsilon \tag{4}$$

holds, $f_{F,\,n(\varepsilon)}$ (x) being the inverse FTR of *f* with respect to {A_1, A_2,…, $A_{n(\varepsilon)}$}.

We note that such a fuzzy partition {A_1, A_2, …, $A_{n(\varepsilon)}$} of [a,b] is not necessarily uniform. Now we discretize the continuous case, that is we assume that the function *f* assumes determined values in a finite number of points p_1,...,p_m ∈[a,b], which are sufficiently dense with respect to the fixed partition, that is for every i = 1,…,n there exists an index $j \in \{1,\ldots,m\}$ such that $A_i(p_j) > 0$. Thus we can define the *n*-tuple $F = [F_1, F_2, ..., F_n]$ as the discrete FTR of *f* with respect to {A_1, A_2, …, A_n }, being

$$F_{\mathrm{i}} = \frac{\sum_{j=1}^{m} f(p_j)A_i(p_j)}{\sum_{j=1}^{m} A_i(p_j)} \tag{5}$$

for every *i* = 1,…,*n*. Then we can also define the discrete inverse FTR of *f* with respect to {A_1, A_2,…, A_n} as the function $f_{F,n}$ by setting for every p_1,..., $p_m \in [a,b]$:

$$f_{F,n}(p_j) = \sum_{i=1}^{n} F_i A_i(p_j) \,. \tag{6}$$

Of course we have the following “discrete” approximation theorem (cfr. [19, Theorem 5]).

Theorem 2. Let f(x) be a function assuming values over a set of points P={p_1,..., p_m}⊆[a,b]. Then for every ε > 0, there exist an integer *n*(ε) and a related fuzzy partition {A_1, A_2, …, $A_{n(\varepsilon)}$} of [a,b] with respect to which *P* is sufficiently dense and such that the inequality

$$| f(p) - f_{F,n(\varepsilon)}(p) | < \varepsilon \tag{7}$$

holds for every *j = 1,…,m*.

By extending the above concepts to functions in two variables and limiting ourselves to the discrete case, let *n*, *m* ≥ 2, x_1, x_2, …, x_n ∈ [a,b] and y_1,y_2, …, y_m ∈[c,d] be n + m assigned points such that $x_1 = a < x_2 < \ldots < x_n = b$ and $y_1 = c < y_2 < \ldots < y_m = d$. Moreover, let A_1,…,A_n: [a,b] → [0,1] be a fuzzy partition of [a,b], B_1,…,B_m: [c,d] → [0,1] be a fuzzy partition of [c,d] and f(x,y) be a function assuming values in some given points (p_j,q_j)∈[a,b]×[c,d], where i = 1,…,N and j = 1,…,M. Suppose that the sets P={p_1, … , p_N} and Q={q_1, … ,q_M} are sufficiently dense with respect to the chosen partitions and then we can define the matrix [F_{kl}] as the discrete FTR of f with respect to {A_1,…, A_n} and {B_1,…,B_m} by setting for each *k* = 1,…,*n* and *l* = 1,…,*m*:

$$F_{\mathrm{kl}} = \frac{\sum_{j=1}^{M}\sum_{i=1}^{N} f(p_i,q_j)A_k(p_i)B_l(q_j)}{\sum_{j=1}^{M}\sum_{i=1}^{N} A_k(p_i)B_l(q_j)} \,. \tag{8}$$

Now we define the discrete inverse FTR of *f* with respect to {A_1, A_2, …, A_n} and {B_1,…,B_m} as the function f_{nm}^{F} by setting for any (i,j)∈{1,…,N}× {1,…,M} as

$$f_{nm}^{F}(p_i,q_j) = \sum_{k=1}^{n}\sum_{l=1}^{m} F_{kl}A_k(p_i)B_l(q_j) \,\cdot \tag{9}$$

It is plain that the following generalization of Theorem 2 holds.

Theorem 3. Let f(x,y) be a function assigned over the sets P×Q ⊆ [a,b] ×[c,d], where P={p_1, … , p_N} and Q={q_1, … ,q_M}. Then for every ε > 0, there exist two integers n(ε), m(ε) and related fuzzy partitions {A_1, A_2, …, $A_{n(\varepsilon)}$} of [a,b] and {B_1, B_2, …, $B_{m(\varepsilon)}$} of [c,d] with respect to which P and Q are sufficiently dense and such that the inequality

$$\left| f(p_i,q_j) - f_{n(\varepsilon)m(\varepsilon)}^{F}(p_i,q_j) \right| < \varepsilon \tag{10}$$

holds for every i∈{1,…,N} and j∈{1,…,M}.

3 The FTR for Coding/Cecoding Images

Let R be a gray image divided in N × M pixels. By normalizing the value of each pixel P(i, j) in [0,1] (for example, R(i,j) = P(i,j)/255 for 256 gray levels), we can consider R as a fuzzy relation R: (i, j)∈{1,…,N} ×{1,…,M} → R(i,j)∈[0,1]. In [5, 6] the image R is compressed by using a discrete FTR [F_{kl}] defined as:

$$F_{\mathrm{kl}}=\frac{\sum_{j=1}^{M}\sum_{i=1}^{N}R(i,j)A_k(i)B_l(j)}{\sum_{j=1}^{M}\sum_{i=1}^{N}A_k(i)B_l(j)} \qquad (11)$$

for every (k,l)∈{1,…,n}×{1,…,m}, where we have assumed that $p_i = i$, $q_j = j$, a = c = 1, b = N, d = M. Of course, {A_1,…,A_n} (resp. B_1,…,B_m)), with n « N (resp. m « M) forms a fuzzy partition of [1,N] (resp. [1,M]). The compressed image is decoded by using an inverse discrete FTR defined as

$$R^F_{nm}(i,j)=\sum_{k=1}^{n}\sum_{l=1}^{m}F_{kl}A_k(i)B_l(j) \qquad (12)$$

for every (i, j)∈{1,…,N}×{1,…, M}. The image R is subdivided in small images called blocks [5, 6] and let B any such block. Let R_B the corresponding fuzzy relation (submatrix of R) of sizes N(B)×M(B), coded to a block F_B of sizes n(B) × m(B) (with 3 ≤ n(B) « N(B), 3 ≤ m(B) « M(B)) via the discrete FTR $[F^B_{kl}]$ defined as

$$F^B_{kl}=\frac{\sum_{j=1}^{M(B)}\sum_{i=1}^{N(B)}R_B(i,j)A_k(i)B_l(j)}{\sum_{j=1}^{M(B)}\sum_{i=1}^{N(B)}A_k(i)B_l(j)} \qquad (13)$$

for every (k,l)∈{1,…,n(B)}×{1,…,m(B)}. As in [5], here we use the following functions {A_1,…,$A_{n(B)}$} and {B_1,…,$B_{m(B)}$} defined as

$$A_1(x)=\begin{cases}0.5(1+\cos\frac{\pi}{h}(x-x_1)) & \text{if } x\in[x_1,x_2]\\ 0 & \text{otherwise}\end{cases}$$

$$A_k(x)=\begin{cases}0.5(1+\cos\frac{\pi}{h}(x-x_k)) & \text{if } x\in[x_{k-1},x_{k+1}]\\ 0 & \text{otherwise}\end{cases} \qquad (14)$$

$$A_n(x)=\begin{cases}0.5(1+\cos\frac{\pi}{h}(x-x_n)) & \text{if } x\in[x_{n-1},x_n]\\ 0 & \text{otherwise,}\end{cases}$$

where n = n(B), *k* = 2,…, *n*, h = (N(B) - 1)/(n - 1), x_k = 1+ h·(k-1) and

$$B_1(y)=\begin{cases}0.5(1+\cos\frac{\pi}{s}(y-y_1)) & \text{if } y\in[y_1,y_2]\\ 0 & \text{otherwise}\end{cases}$$

$$B_t(y)=\begin{cases}0.5(1+\cos\frac{\pi}{s}(y-y_t)) & \text{if } y\in[y_{t-1},y_{t+1}]\\ 0 & \text{otherwise}\end{cases} \qquad (15)$$

$$B_m(y)=\begin{cases}0.5(1+\cos\frac{\pi}{s}(y-y_m)) & \text{if } y\in[y_{m-1},y_m]\\ 0 & \text{otherwise,}\end{cases}$$

where $m = m(B)$, $t = 2,\ldots, m$, $s = (M(B)-1)/(m-1)$, $y_t = 1+ s\cdot(t-1)$. The fuzzy sets (14) and (15) form an uniform fuzzy partition of [1,N(B)] and [1,M(B)], respectively. Each compressed block F_B is decoded to a block $R^F_{n(B)m(B)}$ of sizes $N(B) \times M(B)$ by using the inverse discrete FTR defined as

$$R^F_{n(B)m(B)}(i, j) = \sum_{k=1}^{n(B)}\sum_{l=1}^{m(B)} F^B_{kl} A_k(i) B_l(j) \tag{16}$$

for every (i, j)∈{1,…,N(B)}×{1,…,M(B)}. For every block B and for every ε, Theorem 3 guarantees the existence of two integers (say) n(B) = n(B,ε) « N(B), m(B) = m(B,ε) « M(B) and of two fuzzy partitions $\{A_1,\ldots,A_{n(B)}\}$ and $\{B_1,\ldots,B_{m(B)}\}$ such that the inequality $\left|R_B(i, j) - R^F_{n(B)m(B)}(i, j)\right| < \varepsilon$ holds true. Unfortunately Theorem 3 is not constructive, that is it does not give a practical method for building such integers n(B,ε) and m(B,ε) and the above fuzzy partitions. After some preliminary tests, we have seen that the functions defined from (14) and (15) give the best results and, by simplicity, we have considered all the images as square matrices R (hence N=M), subdivided in square submatrices R_B with $N(B) = M(B)$, in turn compressed to square blocks F_B with sizes $n(B) = m(B)$ and decoded to blocks $R^F_{n(B)m(B)}$. For every block B, then the related compression rate ρ(B) is given from ρ(B) = (n(B)×n(B))/(N(B)×N(B)), thus we can evaluate the quality of the reconstructed image via the PSNR given by

$$PSNR = 20\log_{10}\frac{255}{RMSE}, \tag{17}$$

where RMSE (Root Mean Square Error) is given by

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}\sum_{j=1}^{M}(R(i, j) - R^F_{NM}(i, j))^2}{N \times M}} \tag{18}$$

being R^F_{NM} the image obtained from the composition of the submatrices $R^F_{n(B)m(B)}$ calculated with (16). In the sequel, the above concepts shall be used in each band R, G, B and Y, U, V of the corresponding spaces with the symbols $\rho_R = \rho_R(B)$, $(PSNR)_R$, $(PSNR)_G$, $(PSNR)_B$, $(PSNR)_Y$, $(PSNR)_U$, $(PSNR)_V$. Of course we define the overall $(PSNR)_{RGB}$ and $(PSNR)_{YUV}$) with the following formulas:

$$(PSNR)_{RGB} = \frac{(PSNR)_R + (PSNR)_G + (PSNR)_B}{3} \tag{19}$$

$$(PSNR)_{YUV} = \frac{(PSNR)_Y + (PSNR)_U + (PSNR)_V}{3}. \tag{20}$$

Finally the PSNR's calculated via the DCT and JPEG methods are denoted with $(PSNR)_{DCT}$ and $(PSNR)_{JPEG}$, respectively.

4 By Coding/Decoding in RGB and YUV Spaces

We suppose to have a color image in the RGB space with pixel normalized in [0,1]. Then we convert it from RGB to YUV space via the following formula [24]:

$$\begin{bmatrix} Y \\ U \\ V \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.169 & -0.332 & 0.500 \\ 0.500 & -0.419 & -0.0813 \end{bmatrix}\begin{bmatrix} R \\ G \\ B \end{bmatrix} + \begin{bmatrix} 0 \\ 0.5 \\ 0.5 \end{bmatrix}\cdot$$

In accordance to [15, 16] we compress the converted image by using two compression rates: ρ_Y in the plane Y, and $\rho_U = \rho_V < \rho_Y$ in the planes U, V. We use the discrete FTR method in coding/decoding images in the YUV space and we convert the decoded image in the RGB space by using the following formula [24]:

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1.4075 \\ 1 & -0.3455 & -0.7169 \\ 1 & 1.7790 & 0 \end{bmatrix}\begin{bmatrix} Y \\ U \\ V \end{bmatrix} + \begin{bmatrix} -1.4075\cdot 0.5 \\ 1.0624\cdot 0.5 \\ -1.7790\cdot 0.5 \end{bmatrix}\cdot$$

In our tests we compare the results obtained with the FTR method on images in the YUV space with those ones obtained by using the same method on images in the RGB space and by using the standard JPEG compression method.

We compare also our results with the images obtained from the DCT coding/decoding method, which is used in the first phase of the JPEG coding/decoding process.

In these comparisons we use a compression rate $\rho \approx \rho_{YUV}=(\rho_Y + 2 \cdot \rho_{U.})/3$, ρ_{YUV} being the average of the compression rates ρ_Y, ρ_U, ρ_V used in the single bands of the YUV space, respectively. In Table 1 (resp. Table 2) are reported the sizes $N(\mathrm{B})\times N(\mathrm{B})$ of each square block R_B, the sizes $n(\mathrm{B})\times n(\mathrm{B})$ of the compressed square block F_B and the relative compression rates used by applying the FTR method in the YUV (resp. RGB) space.

Table 1. Sizes of the blocks used for the images in YUV space

Band Y			Bands U and V		
N(B) × N(B)	n(B) × n(B)	ρ_Y	N(B) × N(B)	n(B) × n(B)	$\rho_U = \rho_V$
11 × 11	10 × 10	0.826446	4 × 4	2 × 2	0.250000
14 × 14	13 × 13	0.862245	16 × 16	2 × 2	0.015625
4 × 4	3 × 3	0.562500	16 × 16	2 × 2	0.015625
8 × 8	5 × 5	0.395625	16 × 16	2 × 2	0.015625
5 × 5	2 × 2	0.160000	16 × 16	2 × 2	0.015625
8 × 8	2 × 2	0.062500	16 × 16	2 × 2	0.015625

Table 2. Sizes of the blocks used for the images in RGB space

N(B) × N(B)	n(B) × n(B)	$\rho_R = \rho_G = \rho_B$
3 × 3	2 × 2	0. 444444
11 × 11	6 × 6	0. 297521
9 × 9	4 × 4	0. 197531
8 × 8	3 × 3	0. 140625
13 × 13	3 × 3	0. 06250
16 × 16	2 × 2	0. 033058

In Table 3 are reported the compression rates used in our experiments by applying the FTR method in YUV and RGB spaces and the JPEG/DCT compression method. The quality of the reconstructed images are evaluated by determining the PSNR mean value for the image decoded in each band. In our comparisons we calculate the percent gain, denoted by Gain(YUV/RGB) (resp. Gain(YUV/DCT)) of the PSNR obtained by using the FTR method in the YUV space with respect to the PSNR obtained with the FTR method in the RGB space (resp. the DCT method) and the percent gain, denoted by Gain(JPEG/YUV), of the PSNR obtained by using the JPEG method with respect to the PSNR obtained by using the FTR method in the YUV space. Thus we have

$$\text{Gain(YUV/RGB)} = \frac{[(\text{PSNR of FTR in YUV space}) - (\text{PSNR of FTR in RGB space})]\cdot 100}{(\text{PSNR of FTR in RGB space})}, \tag{21}$$

$$\text{Gain(YUV/DCT)} = \frac{[(\text{PSNR of FTR in YUV space}) - (\text{PSNR in DCT})]\cdot 100}{(\text{PSNR in DCT})}, \tag{22}$$

$$\text{Gain(JPEG/YUV)} = \frac{[(\text{PSNR in JPEG}) - (\text{PSNR of FTR in YUV space})]\cdot 100}{(\text{PSNR of FTR in YUV space})}. \tag{23}$$

Table 3. Compression rates used in the coding processes

ρ_Y	ρ_U	$\rho_{YUV}=(\rho_Y+2\cdot\rho_U)/3$	$\rho_{RGB}=\rho_R$	$\rho_{JPEG} = \rho_{DCT}$
0.826446	0.250000	0.442148	0.444444	0.444444
0.862245	0.015625	0.297832	0.297521	0.297521
0.562500	0.015625	0.197917	0.197531	0.197531
0.395625	0.015625	0.140625	0.140625	0.140625
0.16000	0.015625	0.063750	0.06250	0.06250
0.062500	0.015625	0.031250	0.033058	0.035160

In order to have an exhaustive point of view, we have considered 100 color images of sizes 256 × 256 extracted from the well known image database of the University of Southern California at http://sipi.usc.edu/database but, for brevity, we only give the results for the following four images downloaded from the volume "Miscellaneous" of this database: "Girl 4.1.01" (Figure 3), "Couple 4.1.02" (Figure 4), "Girl 4.1.04" (Figure 5) and "Tree 4.1.06" (Figure 6).

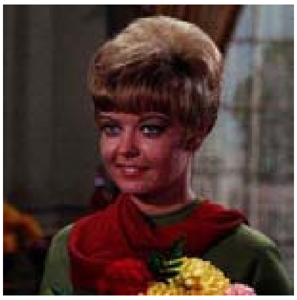

Figure 3. Girl 4.1.01

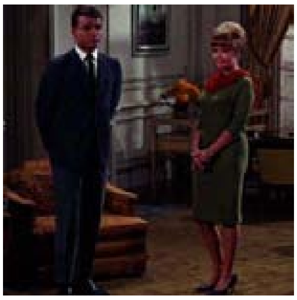

Figure 4. Couple 4.1.02

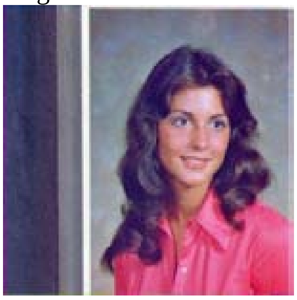

Figure 5. Girl 4.1.04

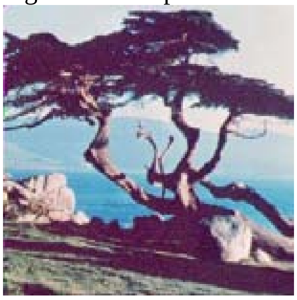

Figure 6. Tree 4.1.06

5. Comparisons

In Table 4 we show the compression rates used in the bands *Y*, *U*, *V*, the PSNR's obtained for these bands, the mean PSNR obtained for the image "Girl 4.1.01" decoded via FTR method in YUV space and the mean PSNR obtained for the images decoded after the conversion in RGB space.

Table 4. Values of PSNR for the image "Girl 4.1.01" via the FTR method in YUV space

ρ_Y	$(PSNR)_Y$	$\rho_U = \rho_V$	$(PSNR)_U$	$(PSNR)_V$	ρ_{YUV}	$(PSNR)_{YUV}$	$(PSNR)_{RGB}$
0.826446	38.7531	0.250000	48.0623	45.9190	0.442148	44.2448	38.3390
0.862245	39.2747	0.015625	38.9115	34.9604	0.297832	37.7155	34.9274
0.562500	36.2639	0.015625	38.9115	34.9604	0.197917	36.7119	33.3630
0.395625	34.3987	0.015625	38.9115	34.9604	0.140625	36.0902	32.1001
0.160000	30.0654	0.015625	38.9115	34.9604	0.063750	34.6458	29.5185
0.062500	27.1773	0.015625	38.9115	34.9604	0.031250	33.6831	27.4558

In Table 5 we show the compression rate used for the three bands *R*, *G*, *B*, the PSNR obtained by using the FTR method for the images decoded in each band and the mean PSNR. Moreover, in Table 6 we present the compression rate and the related PSNR obtained by using the DCT and the JPEG methods.

In Figure 7 we show the trends of the PSNR with respect to the compression rate ρ in all the methods considered here.

Figure 8 contains the percent gains Gain(YUV/RGB), Gain(YUV/DCT)) and Gain(JPEG/YUV) defined from (21) ÷ (23).

Table 5. Values of PSNR for the image “Girl 4.1.01” via the FTR method in RGB space

ρ_{RGB}	$(PSNR)_R$	$(PSNR)_G$	$(PSNR)_B$	$(PSNR)_{RGB}$
0.444444	34.2886	34.6155	34.9373	34.6138
0.297521	33.1792	33.4456	33.7657	33.4635
0.197531	31.0877	31.5139	31.8775	31.4930
0.140625	29.5117	29.9524	30.3874	29.9505
0.06250	26.8332	27.4686	28.1287	27.4768
0.033058	24.7486	25.6123	26.3812	25.6807

Table 6. Values of PSNR for the image “Girl 4.1.01” via the DCT and JPEG methods

$\rho_{DCT} = \rho_{JPEG}$	$(PSNR)_{DCT}$	$(PSNR)_{JPEG}$
0.444444	32.9555	39.4559
0.297521	30.9780	39.0397
0.197531	30.0026	37.9548
0.140625	28.2056	37.4974
0.06250	26.1011	34.4941
0.035160	23.8803	32.4817

Table 7. Values of PSNR for the image “Couple 4.1.02” via the FTR method in YUV space

ρ_Y	$(PSNR)_Y$	$\rho_U = \rho_V$	$(PSNR)_U$	$(PSNR)_V$	ρ_{YUV}	$(PSNR)_{YUV}$	$(PSNR)_{RGB}$
0.826446	38.0049	0.250000	49.3317	48.2004	0.442148	45.1790	38.4378
0.862245	38.4138	0.015625	40.0967	38.0575	0.297832	39.1528	35.6056
0.562500	35.9046	0.015625	40.0967	38.0575	0.197917	38.6164	34.3362
0.395625	33.7564	0.015625	40.0967	38.0575	0.140625	37.6003	32.9114
0.16000	29.5091	0.015625	40.0967	38.0575	0.063750	36.1845	29.5677
0.062500	26.9052	0.015625	40.0967	38.0575	0.031250	35.3166	27.4240

Table 8. Values of PSNR for the image “Couple 4.1.02” via the FTR method in RGB space

ρ_{RGB}	$(PSNR)_R$	$(PSNR)_G$	$(PSNR)_B$	$(PSNR)_{RGB}$
0.444444	34.0102	32.1927	34.3828	34.1952
0.297521	32.3370	32.4518	32.5935	32.4608
0.197531	30.6446	30.8797	31.0866	30.8703
0.140625	28.9953	29.2564	29.4748	29.2422
0.06250	26..220	27.0635	27.4077	27.0311
0.033058	24.8723	25.4277	25.8977	25.3992

The PSNR trends of Figure 7 and the percent gain trends of Figure 8 show that the results obtained by using the FTR method in YUV space are better than the results obtained by using the FTR method in the RGB space and the DCT method essentially for strong compressions while the Gain(JPEG/YUV) trend shows that the results obtained by using the FTR method in YUV space are well comparable with those ones obtained by using the JPEG method for low compression rates. Tables 7, 8, 9 show the PSNR obtained for the color image “Couple 4.1.0.2” compressed by using the FTR method in YUV space, in RGB space, in DCT and JPEG methods, respectively.

In Figure 9 we show the trends of the PSNR with respect to the compression rate ρ in all the methods and in Figure 10 we presents the percent gains Gain(YUV/RGB), Gain(YUV/DCT)) and Gain(JPEG/YUV). Then the results, in terms of PSNR and percent gains for the image “Couple 4.1.02”, confirm those ones obtained for the image “Girl 4.1.01”.

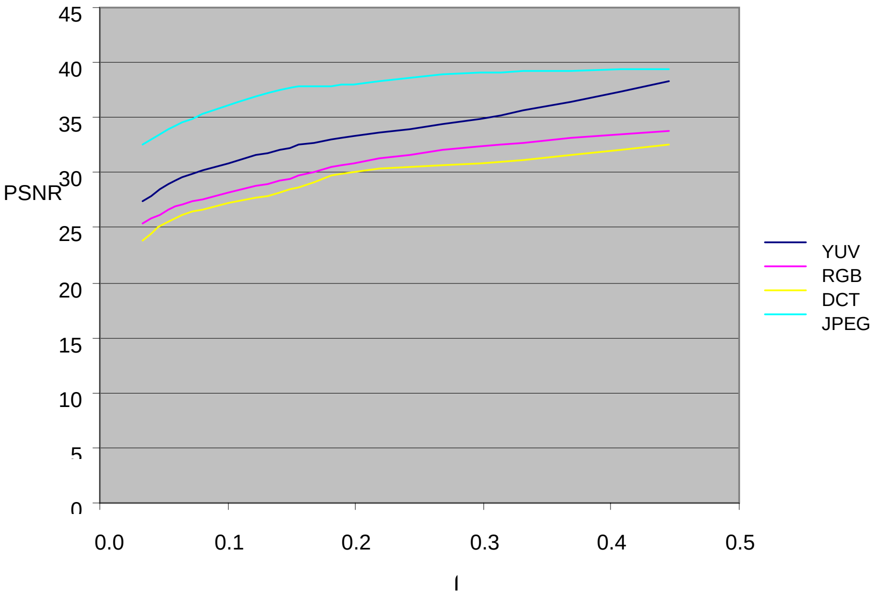

Figure 7. PSNR trends obtained in all methods for the image “ Girl 4.1.01”

Table 9. Values of PSNR for the image “Couple 4.1.02” via the DCT and JPEG methods

$\rho_{DCT} = \rho_{JPEG}$	$(PSNR)_{DCT}$	$(PSNR)_{JPEG}$
0.444444	33.0293	42.1601
0.297521	31.0977	41.9259
0.197531	29.3218	41.4370
0.140625	28.5088	39.8658
0.062500	25.8877	36.7028
0.035160	24.0588	34.2571

In Figure 9 we show the trends of the PSNR with respect to the compression rate ρ in all the methods and in Figure 10 we presents the percent gains Gain(YUV/RGB), Gain(YUV/DCT)) and Gain(JPEG/YUV). Then the results, in terms of PSNR and percent gains for the image “Couple 4.1.02”, confirm those ones obtained for the image “Girl 4.1.01”.

In Tables 10, 11, 12 we show the PSNR obtained for the color image “Girl 4.1.04” compressed by using the FTR method in the YUV space, in RGB space and in the DCT and JPEG methods, respectively.

Figure 11 contains the trends of the PSNR with respect to the compression rate ρ in all the methods. Figure 12 contains the percent gains Gain(YUV/RGB), Gain(YUV/DCT)) and Gain(JPEG/YUV). The results are similar to those ones obtained for the images “Girl 4.1.01” and “Couple 4.1.02”.

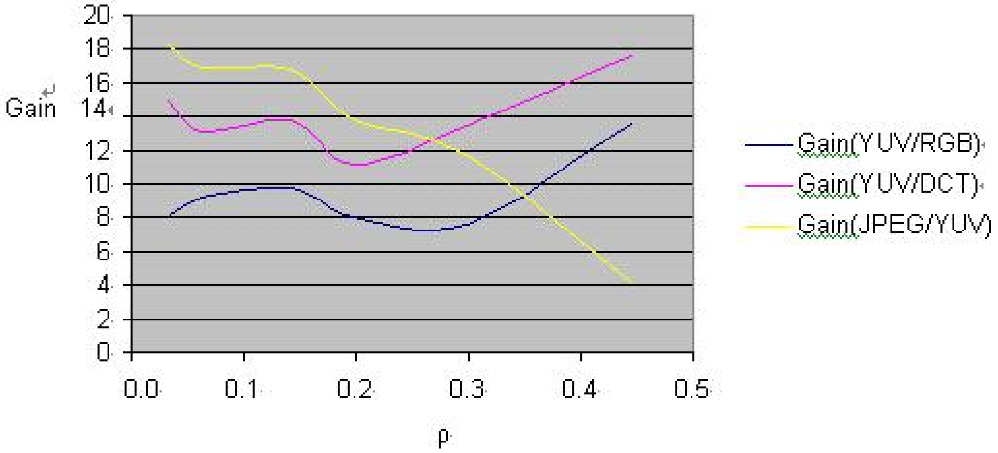

Figure 8. Trends of the percent gains obtained for the image "Girl 4.1.01"

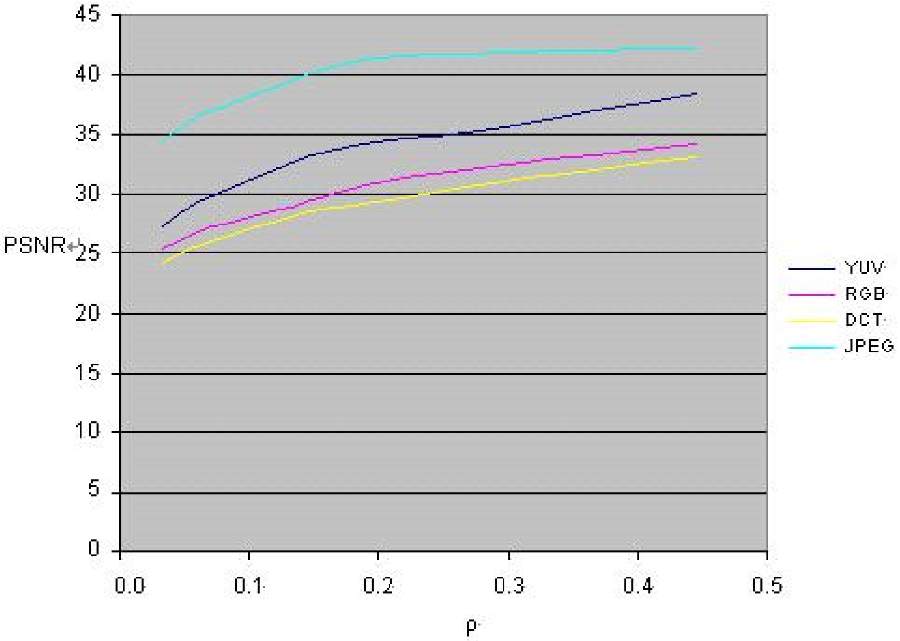

Figure 9. PSNR trends obtained in all methods for the image "Couple 4.1.02"

Table 10. Values of PSNR for the image "Girl 4.1.04" via the FTR method in YUV space

ρ_Y	$(PSNR)_Y$	$\rho_U = \rho_V$	$(PSNR)_U$	$(PSNR)_V$	ρ_{YUV}	$(PSNR)_{YUV}$	$(PSNR)_{RGB}$
0.826446	39.4809	0.250000	47.1007	44.6199	0.442148	43.7338	37.8675
0.862245	39.6136	0.015625	36.3692	33.4217	0.297832	36.4682	36.4682
0.562500	36.2842	0.015625	36.3692	33.4217	0.197917	35.3584	35.3584
0.395625	33.6135	0.015625	36.3692	33.4217	0.140625	34.4681	34.4681
0.16000	29.4108	0.015625	36.3692	33.4217	0.063750	33.0672	33.0672
0.062500	25.7537	0.015625	36.3692	33.4217	0.031250	32.1815	32.1815

Table 11. Values of PSNR for the image “Girl 4.1.04” via the FTR method in RGB space

ρ_{RGB}	$(PSNR)_R$	$(PSNR)_G$	$(PSNR)_B$	$(PSNR)_{RGB}$
0.444444	34.6447	34.4998	35.0771	34.7405
0.297521	33.1316	33.0268	33.3919	33.1834
0.197531	30.9807	30.9112	31.4261	31.1060
0.140625	28.5130	28.1548	28.9723	28.5467
0.06250	26.9491	26.8349	27.7917	27.1919
0.033058	23.7524	23.5312	24.6985	23.9940

Table 12. Values of PSNR for the image “Girl 4.1.04” via the DCT and JPEG methods

$\rho_{DCT} = \rho_{JPEG}$	$(PSNR)_{DCT}$	$(PSNR)_{JPEG}$
0.444444	32.4751	35.2751
0.297521	32.2877	35.2460
0.197531	30.2129	35.0274
0.140625	27.6067	34.1589
0.062500	25.3262	32.8795
0.035160	22.7105	31.4897

Table 13. Values of PSNR for the image “Tree 4.1.06” via the FTR method in YUV space

ρ_Y	$(PSNR)_Y$	$\rho_U = \rho_V$	$(PSNR)_U$	$(PSNR)_V$	ρ_{YUV}	$(PSNR)_{YUV}$	$(PSNR)_{RGB}$
0.826446	32.7784	0.250000	47.3167	43.0456	0.442148	41.0469	32.6760
0.862245	33.0496	0.015625	36.6848	31.2216	0.297832	36.4682	29.7371
0.562500	29.7449	0.015625	36.6848	31.2216	0.197917	35.3584	28.1302
0.395625	27.6596	0.015625	36.6848	31.2216	0.140625	34.4681	27.0294
0.16000	24.0636	0.015625	36.6848	31.2216	0.063750	33.0672	24.2024
0.062500	20.8604	0.015625	36.6848	31.2216	0.031250	32.1815	21.6241

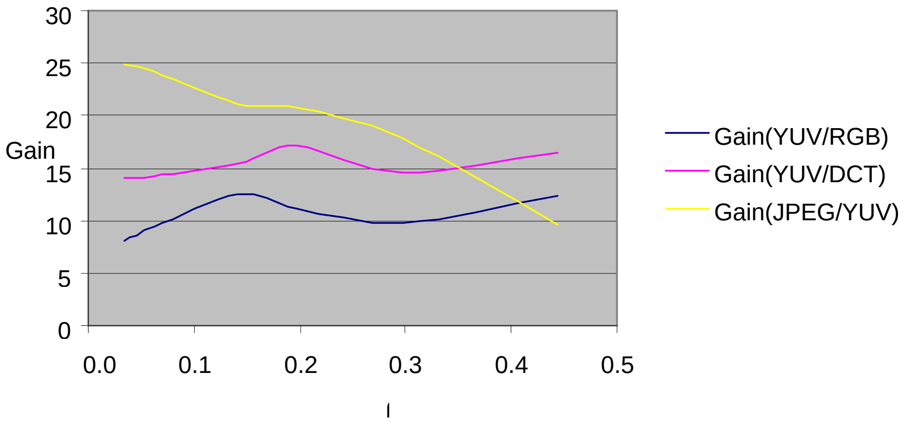

Figure 10. Trends of the percent gains obtained for the image “Couple 4.1.02”

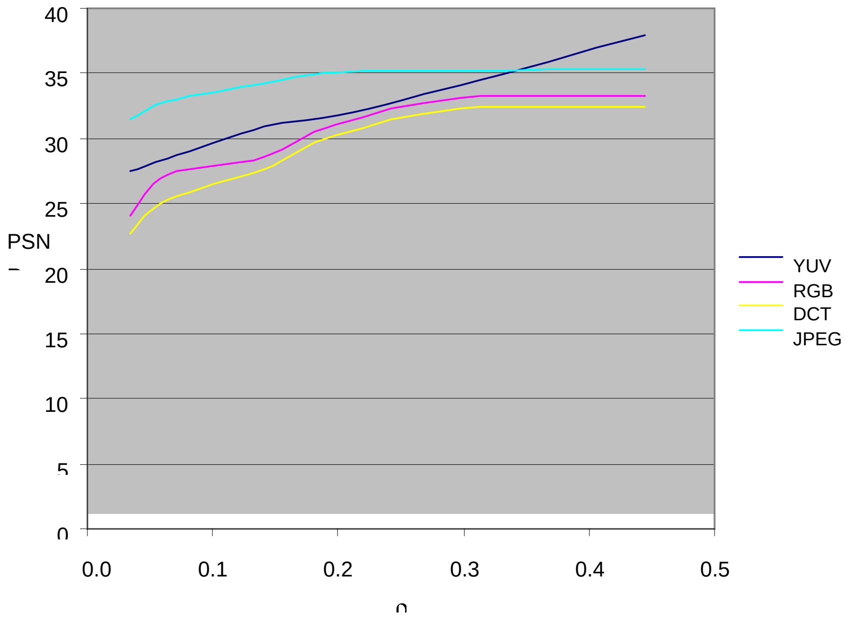

Figure 11. PSNR trends obtained in all methods for the image "Girl 4.1.04"

Table 14. Values of PSNR for the image "Tree 4.1.06" via the FTR method in RGB space

ρ_{RGB}	$(PSNR)_R$	$(PSNR)_G$	$(PSNR)_B$	$(PSNR)_{RGB}$
0.444444	28.8505	28.4319	28.7840	28.6888
0.297521	27.4117	27.1262	27.3625	27.3001
0.197531	25.6100	25.1470	25.5391	25.4320
0.140625	23.6062	23.0617	23.5330	23.4003
0.06250	21.9725	21.0418	21.8198	21.6114
0.033058	20.3843	19.1253	20.2157	19.9084

In Tables 13, 14, 15 we show the PSNR obtained for the color image "Tree 4.1.06" compressed by using the FTR method in YUV space, in RGB space and in DCT and JPEG methods, respectively.

In Figure 13 we show the trends of the PSNR with respect to the compression rate ρ in all the methods. Figure 14 contains the percent gains Gain(YUV/RGB), Gain(YUV/DCT)) and Gain(JPEG/YUV). The results are similar to those ones obtained for the images "Girl 4.1.01", "Couple 4.1.02" and "Girl 4.1.04".

For sake of completeness, we limit ourselves to show only the image "Girl 4.1.01" in Figures 15 ÷ 26 and the image "Tree 4.1.06" in Figures 27 ÷ 38 reconstructed under the FTR method in the YUV space (and converted to RGB space) and JPEG method. using various compression rates. The reconstructed color images show that the FTR method in the YUV space is well comparable with the JPEG method for weak compressions whereas the images compressed under the JPEG method with compression rates 0.035 and 0.062 have better quality in comparison to the those ones compressed using the FTR method in the YUV space.

Table 15. Values of PSNR for the image "Tree 4.1.0.6" via the DCT and JPEG methods

$\rho_{DCT} = \rho_{JPEG}$	$(PSNR)_{DCT}$	$(PSNR)_{JPEG}$
0.444444	27.6782	35.9372
0.297521	26.3944	35.6501
0.197531	24.1773	34.9764
0.140625	22.2296	32.9186
0.062500	20.3810	29.3296
0.035160	17.8499	27.1698

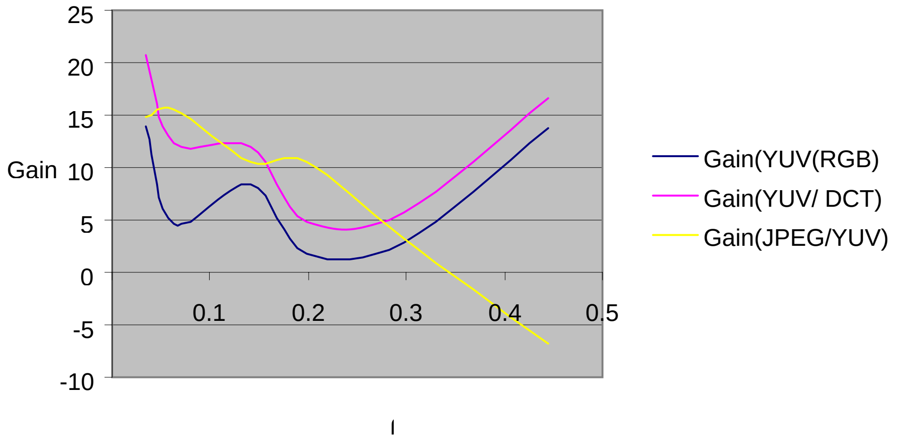

Figure 12. Trends of the percent gains obtained for the image “Girl 4.1.04”

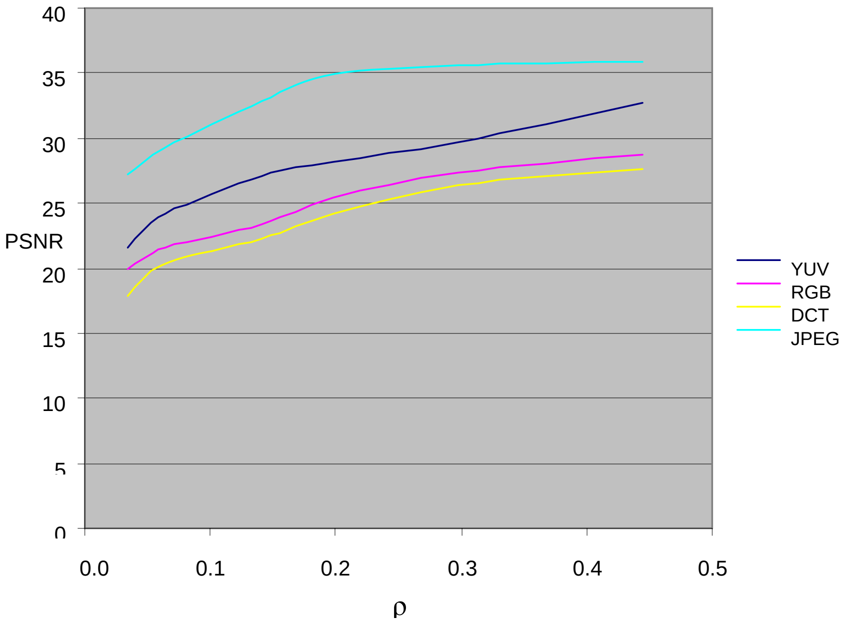

Figure 13. PSNR trends obtained in all methods for the image “Tree 4.1.06”

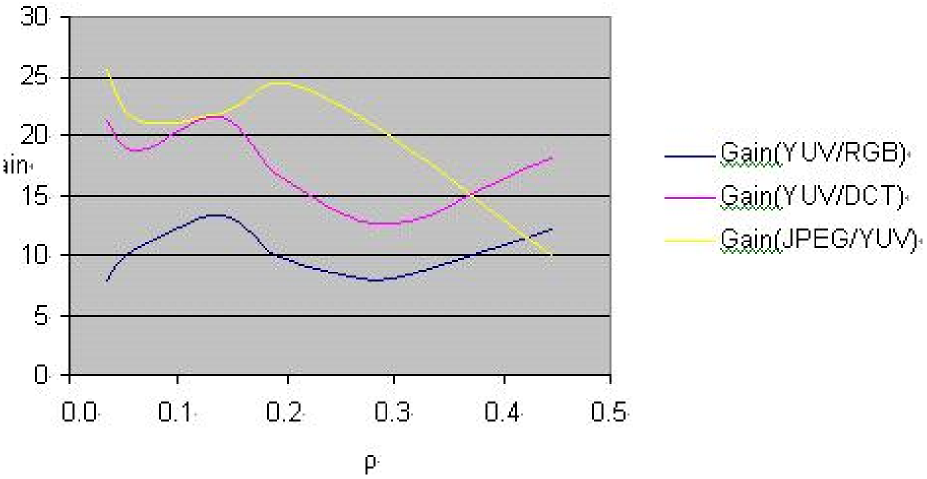

Figure 14. Trends of the percent gains obtained for the image “Tree 4.1.06”

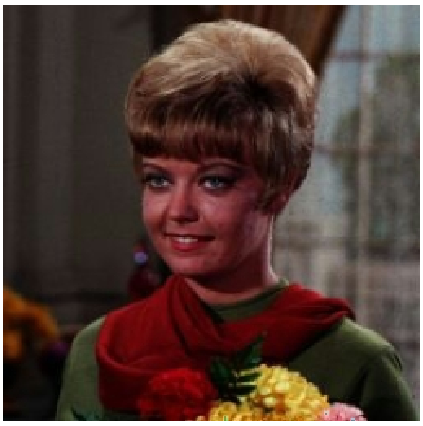

Figure 15. FTR in YUV space, ρ = 0.444

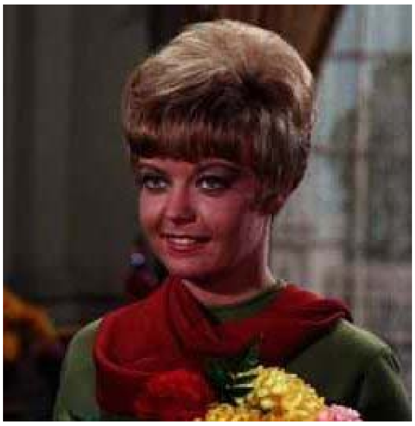

Figure 16. JPEG, ρ = 0.444

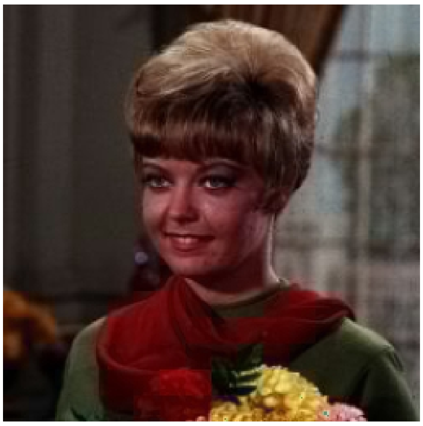

Figure 17. FTR in YUV space, ρ = 0.297

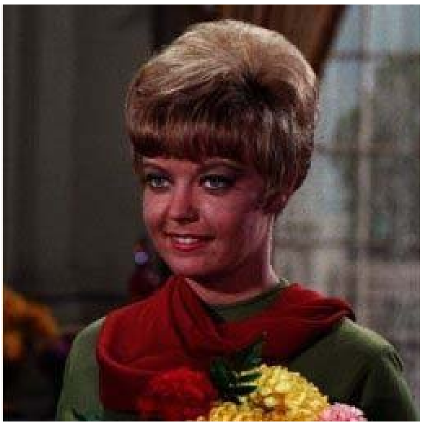

Figure 18. JPEG, ρ = 0.297

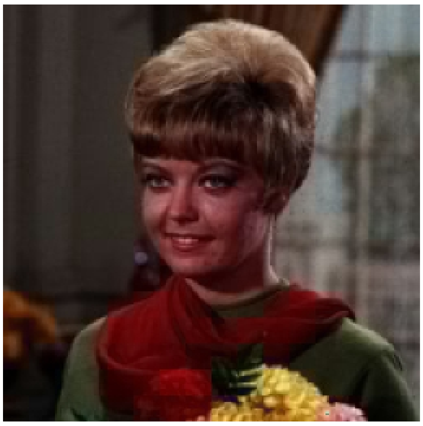

Figure 19. FTR in YUV space, ρ = 0.197

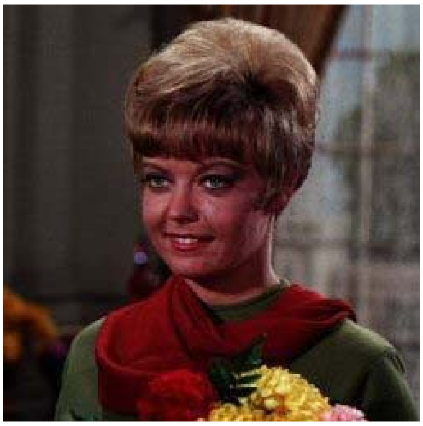

Figure 20. JPEG, ρ = 0.197

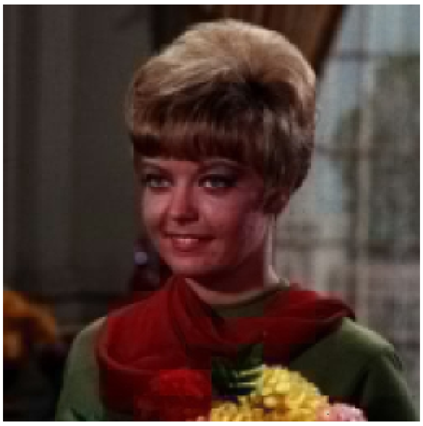

Figure 21. FTR in YUV space, ρ = 0.140

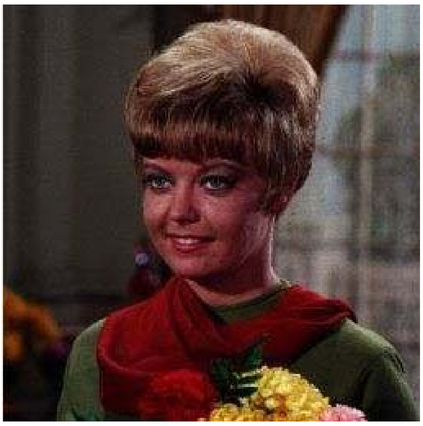

Figure 22. JPEG, ρ = 0.140

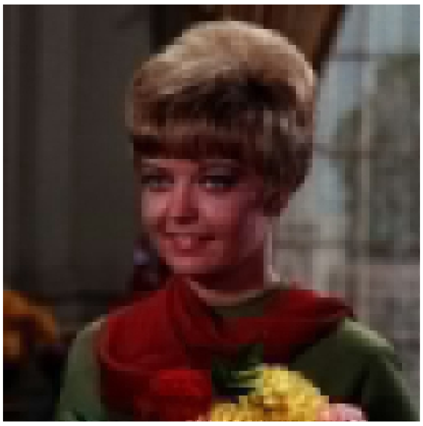

Figure 23. FTR in YUV space, ρ = 0.062

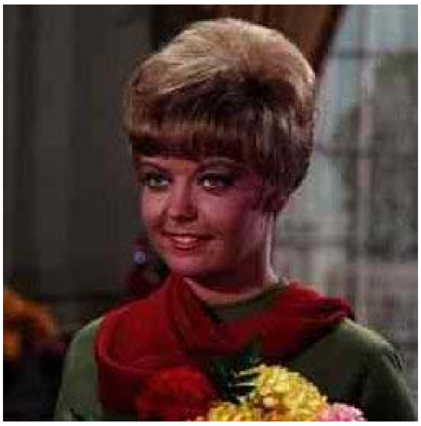

Figure 24. JPEG, ρ = 0.062

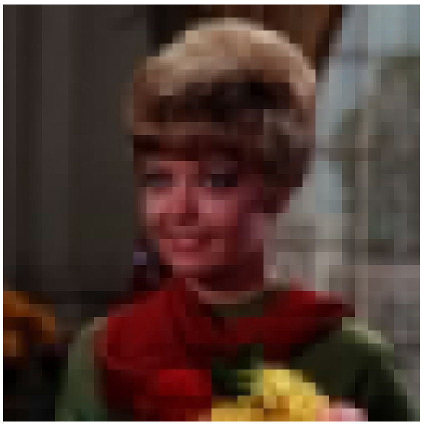

Figure 25. FTR in YUV space, ρ = 0.035

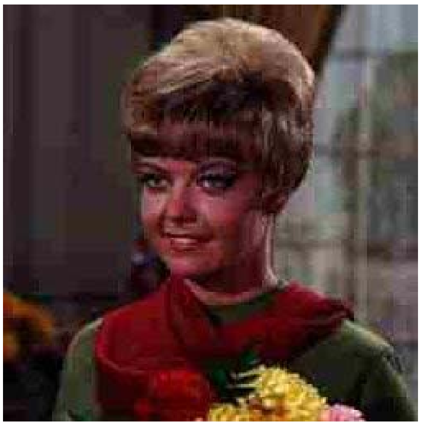

Figure 26. JPEG, ρ = 0.035

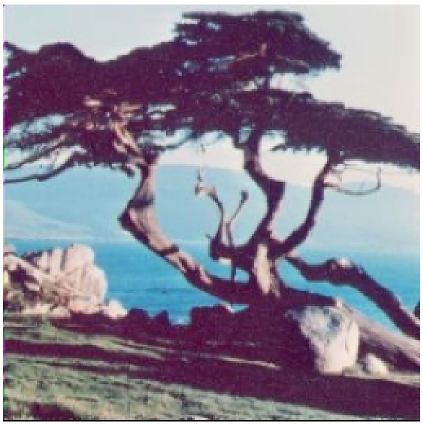

Figure 27. FTR in YUV space, ρ = 0.444

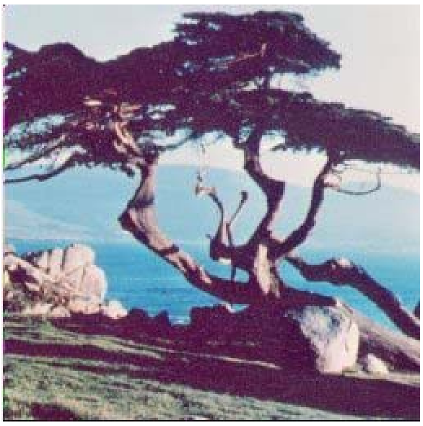

Figure 28. JPEG, ρ = 0.444

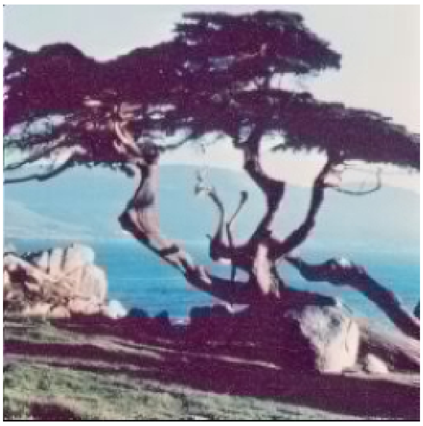

Figure 29. FTR in YUV space, ρ = 0.297

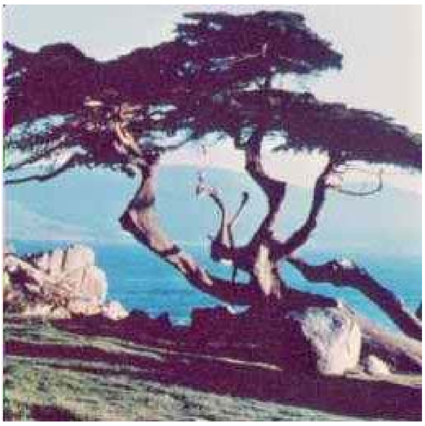

Figure 30. JPEG, ρ = 0.297

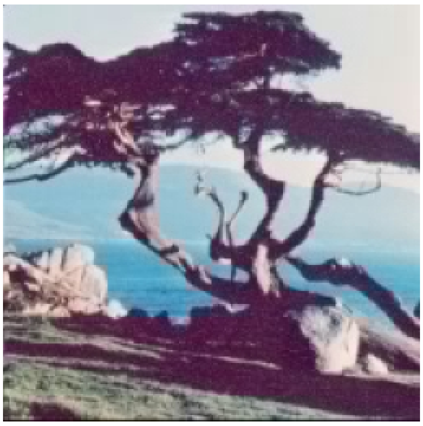

Figure 31. FTR in YUV space, ρ = 0.197

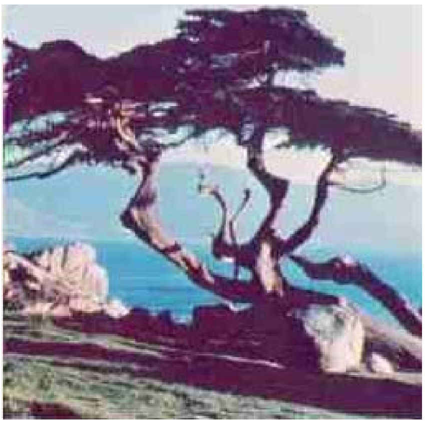

Figure 32. JPEG, ρ = 0.197

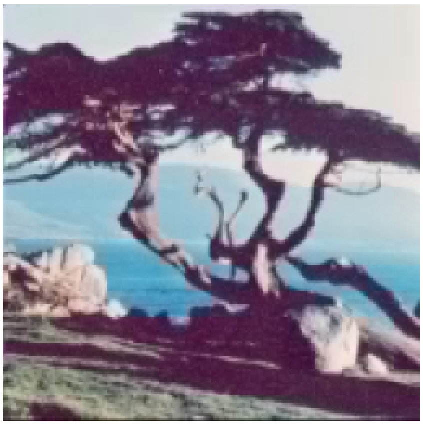

Figure 33. FTR in YUV space, ρ = 0.140

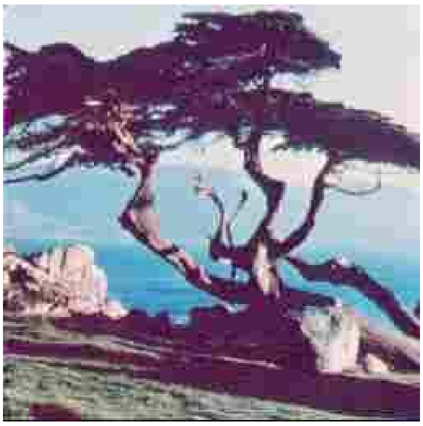

Figure 34. JPEG, ρ = 0.140

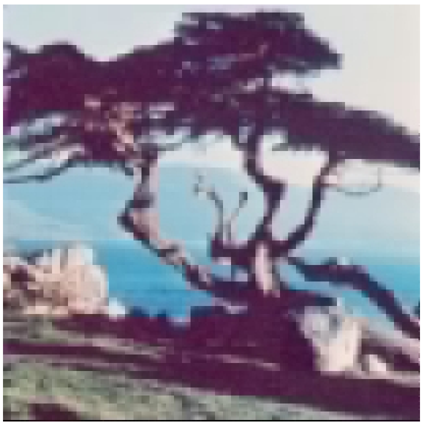

Figure 35. FTR in YUV space, ρ = 0.062

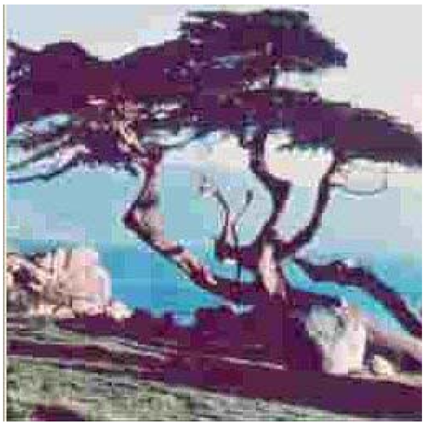

Figure 36. JPEG, ρ = 0.062

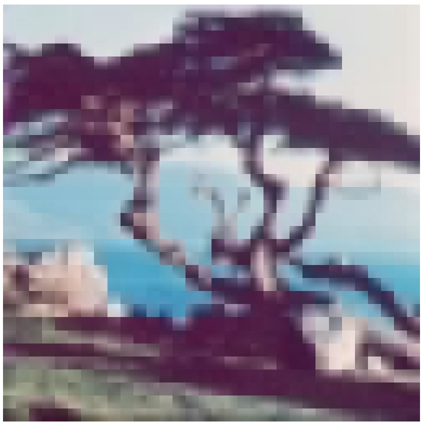

Figure 37. FTR in YUV space, ρ = 0.035

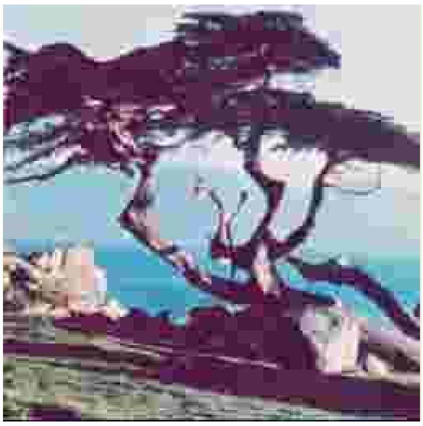

Figure 38. JPEG, ρ = 0.035

Figure 39 shows how the mean percent gains, that is the mean of all the Gain(YUV/RGB), Gain(YUV/DCT) and Gain(JPEG/YUV) calculated for each image of the sample considered from the above database, vary with respect to the compression rate ρ: the color images compressed by using the FTR method in YUV space are always better than those ones obtained by using the FTR method in RGB space and the DCT method, essentially for low and high compression rates, and they are comparable with those ones obtained by using the JPEG compression method for high compression rates (ρ > 0.3).

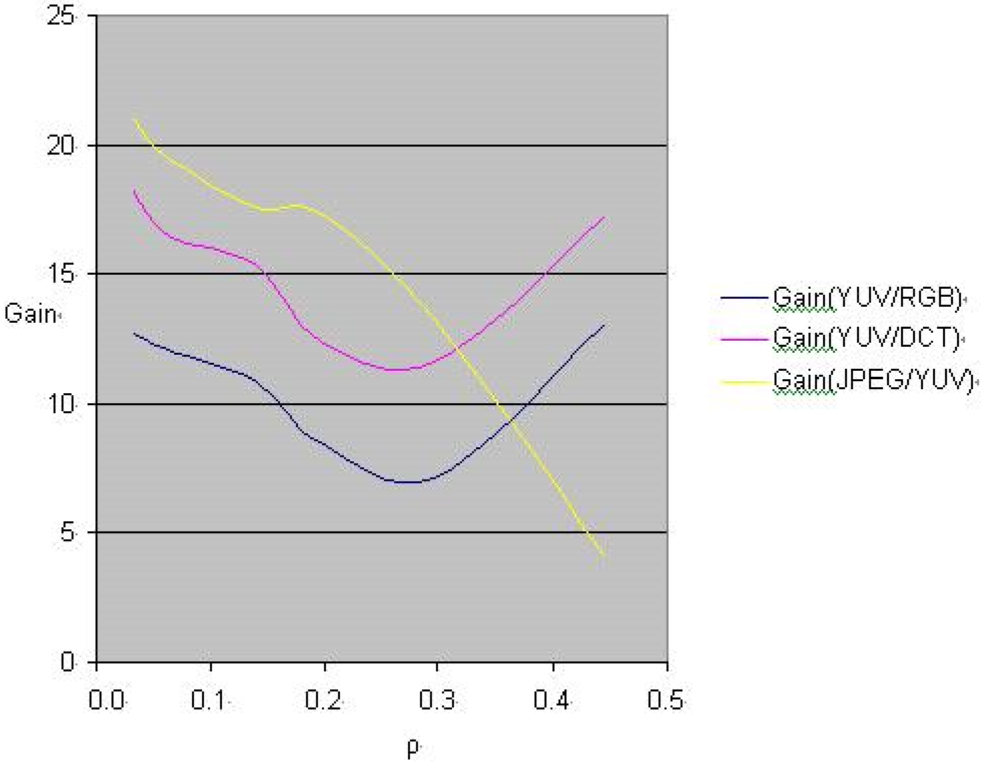

Figure 39. Trends of the percent gains of the images in the sample

6 Conclusions

In previous works [15, 16] the authors showed the advantages in coding/decoding images in YUV space by using fuzzy relation equations. The authors [5] showed that gray image decoded after a compression made by using the FTR method are better than those ones obtained with the FRE method [1, 2, 7, 8, 11, 12, 14, 17] and well comparable with the same images obtained by using the DCT method.

In this work we show that the color images reconstructed after a compression obtained by using the FTR method in YUV space are better then those ones obtained with the FTR method in RGB space and DCT method; furthermore the PSNR of the image deduced with the FTR method in YUV space gives PSNR values close to the PSNR obtained using the standard JPEG method under high compression rates.

Future researches on the usage of the FTR method in YUV space shall be made in other contexts like high resolution of very large images, image information retrieval [3], watermarking [4] and video compression [8, 9, 10, 22, 25].

References

[1] Di Martino, F., V. Loia, and S. Sessa, A method for coding/decoding images by using fuzzy relation equations, *Lecture Notes in Artificial Intelligence*, vol.2715, pp.436–441, 2003.

[2] Di Martino, F., V. Loia, and S. Sessa, A method in the compression/decompression of images by using fuzzy equations and fuzzy similarities, *Proceedings of the 10th IFSA World Congress*, Istanbul, pp.524–527, 2003.

[3] Di Martino, F., H. Nobuhara, and S. Sessa, Eigen fuzzy sets and image information retrieval, *Proceedings of the International Conference on Fuzzy Information Systems*, Budapest, vol.3, pp.1385–1390, 2004.

[4] Di Martino, F., and S. Sessa, Digital watermarking in coding/decoding processes with fuzzy relation equations, *Soft Computing*, vol.10, no.3, pp.238–243, 2006.

[5] Di Martino, F., V. Loia, I. Perfilieva, and S. Sessa, An image coding/decoding method based on direct and inverse fuzzy transform, *International Journal of Approximate Reasoning*, vol.48, no.1, pp.110–131, 2008.

[6] Di Martino, F., and S. Sessa, Compression and decompression images with discrete fuzzy transforms, *Information Sciences*, vol.117, no.11, pp.2349-2362, 2007.

[7] Di Nola, A., W. Pedrycz, E. Sanchez, and S. Sessa, *Fuzzy Relation Equations and Their Applications to Knowledge Engineering*, Kluwer Academic Publishers, Dordrecht, 1989.

[8] Hirota, K., and W. Pedrycz, Data compression with fuzzy relational equations, Fuzzy *Sets and Systems*, vol.126, no.3, pp.325–335, 2002.

[9] Kong, X., L. Wang, and G. Li, Fuzzy clustering algorithms based on resolution and their application in image compression, *Pattern Recognition*, vol.35, pp.2439–2444, 2002.

[10] Loe, K.F., W.G. Gu, and H.K. Phua, Speed-up fractal image compression with a fuzzy classifier, *Signal Processing: Image Communication*, vol.10, no.4, pp.303–311, 1997.

[11] Loia, V., W. Pedrycz, and S. Sessa, Fuzzy relation calculus in the compression and decompression of fuzzy relations, *International Journal of Image and Graphics*, vol.2, no.4, pp.617-632, 2002.

[12] Loia, V., and S. Sessa, Fuzzy relation equations for coding/decoding processes of images and videos, *Information Sciences*, vol.171, pp.145–172, 2005.

[13] Makrogiannis, S., G. Economou, and S. K. Fotopoulos, Region-oriented compression of color images using fuzzy inference and fast merging, *Pattern Recognition*, vol.35, pp.1807–1820, 2002.

[14] Nobuhara, H., W. Pedrycz, and K. Hirota, Fast solving method of fuzzy relational equation and its application to lossy image compression, *IEEE Transactions of Fuzzy Systems*, vol.8, no.3, pp.325–334, 2000.

[15] Nobuhara, H., W. Pedrycz, and K. Hirota, Relational image compression: optimizations through the design of fuzzy coders and YUV color space, *Soft Computing*, vol.9, no.6, pp.471–479, 2005.

[16] Nobuhara, H., K. Hirota, *et al.*, Fuzzy relation equations for compression/decompression processes of colour images in the RGB and YUV color spaces, *Fuzzy Optimization and Decision Making*, vol.4, no.3, pp.235–246, 2005.

[17] Nobuhara, H., K. Hirota, *et al.*, A motion compression/ reconstruction method based on max t-norm composite fuzzy relational equations, *Information Sciences*, 176: 2526–2552, 2006.

[18] Perfilieva, I., Fuzzy transforms: application to reef growth problem, *Fuzzy Logic in Geology*, Academic Press, Amsterdam, pp.275–300, 2003.

[19] Perfilieva, I., Fuzzy transforms, *Fuzzy Sets and Systems*, vol.157, no.8, pp.993–1023, 2006.

[20] Perfilieva, I., and E. Chaldeeva, Fuzzy transformation, *Proceedings of the 9th IFSA World Congress and 20th NAFIPS Internat. Conf.*, Vancouver, pp.1946–1948, 2001.

[21] Perfilieva, I., V. Novak, A. Dvorak, Fuzzy transform in the analysis of data, *International Journal of Approximate Reasoning*, vol.38, no.1, pp.36–46, 2008.

[22] Russo, F., A nonlinear technique based on fuzzy models for the correction of quantization artifacts in image compression, *Measurement*, vol.32, no.4, pp.273–279, 2002.

[23] Štěpnička, M., and O. Polakovič, Fuzzy transform from a neural network point of view, *Proceedings of IPMU*, Paris, pp.1860–1867, 2006.

[24] The International Telegraph And Telephone Consultative Committee, *Information Technology – Digital Compression And Coding of Continuous-Tone Still Images–Requirements and Guidelines*, Recommendation T81, 1992, (www.w3.org/Graphics/JPEG/itu–t81.pdf).

[25] Tsekouras, G.E., A fuzzy vector quantization approach to image compression, *Applied Mathematics and Computation*, vol.167, no.5, pp.539–560, 2005.