Visual Attention Simulation in RGB and HSV Color Spaces

***Frédérique Robert-Inacio*[1,2], *Quentin Stainer*[2], *Rémy Scaramuzzino*[2], *Edith Kussener-Combier*[1,2]**
[1] ***Institut Matériaux Microélectronique et Nanosciences de Provence UMR CNRS 6242, Toulon, France***
[2] ***Institut Supérieur de l'Électronique et du Numérique, Toulon, France***

Abstract

In computer vision several methods are directly inspired by the human visual system. Visual attention is one of the main abilities a human eye uses to discover a new scene. This ability is based on the focus of attention principle, which enables to look at a particular point. The works presented in this paper describe how to simulate such an ability in computer vision. Color images are resampled according to this principle, in order to considerably decrease the amount of data to be processed. This resampling is done by using a concentric distribution of hexagonal cells instead of the rectangular cell grid generally provided by uniform sensors such as CCD cameras. Such a distribution is derived from the cone distribution on the retina and the result is encoded by using polar coordinates. In this way the information is more and more blurred in an isotropic way, when getting far and far from the focusing point. This resampling method then allows decreasing data number while it globally keeps all the information included in the image. In this way a new scene can be explored by focusing successively at a sequence of focusing points. This allows reproducing the human eye behavior that explores a new scene by saccades. Furthermore the resampled images can be used in order to set up image preprocessing. For example this resampling can be achieved before a preliminary step of segmentation. The resampled image is segmented in order to coarsely determine regions. Afterwards the segmentation step can be refined by combining the segmented resampled image and the original image. A comparison between results obtained in RGB and HSV spaces is also given on a set of images.

Introduction

In computer vision, several mechanisms of the human eye are simulated in order to reproduce the human behavior when acquiring visual data. For example the human eye moves by saccades for discovering a new scene [1]. Such a behaviour induces differences on image acquisition depending on the observer sensitivity. Furthermore the roles played by central and peripheral visions are studied to determine what the human eye is sensitive to [2].

Several studies on attention focusing were achieved in the nineties and later on grey-level images [3][4]. These studies showed how to simulate the cone behavior in the fovea. The fovea is a small region of retina where visual acuity is maximal (Fig. 1).

Cones and rods are light receptors located on the retina. Cones allow color perception whereas rods sense brightness and can be sensitive at low light levels [6][7]. As a matter of fact, the role played by cones in diurnal vision is preponderant. Cones are much less numerous than rods in most parts of the retina, but greatly outnumber rods in the fovea. Furthermore cones are located in a concentric way inside the fovea (Fig. 2) [8].

This paper shows how to resample an image by simulating the cone behavior inside the fovea. A point called focusing point is chosen on the image and then the whole image is covered with hexagonal-shaped cells. Each cell integrates information data on the image region it covers. This allows to undersample the image and reduce data amount. This study is led in both RGB and HSV spaces.

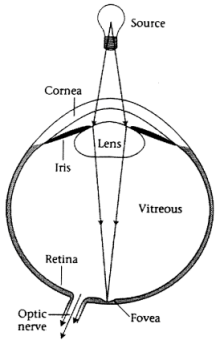

Figure 1. *Concentric distribution of cone cells in the fovea [5].*

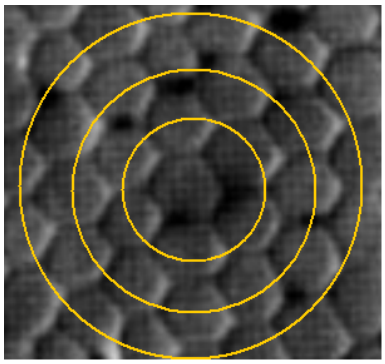

Figure 2. *Concentric distribution of cone cells in the fovea [9].*

Focus of Attention

The basic principle of attention focusing is the following. Generally the focusing point is chosen as a point of great interest. The eye focuses on this point and sees the small region surrounding the focusing point very clearly as it corresponds to photoreceptors of the fovea. Outside the fovea visual data are blurred (Fig. 3).

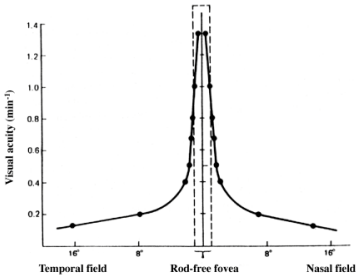

Figure 3. *Visual acuity according the vision angle [10].*

In this way several kinds of photoreceptors can be used to redefine the radial sampling outside the fovea. Actually works from [3] show how this resampling method works with circular cells. Unfortunately using disks to modelize retina photoreceptors prevents from defining a pavement, but only a covering of retina. In other words some parts of retina are covered by several photoreceptors (up to 4) and then information data belonging to these areas are taken into account several times. As well, using square cells with a non homogeneous distribution requires to take into account data more than a single time [11]. That is why hexagonal cells with size evolving according to the distance to the fovea have been chosen as they define a pavement of the whole space under study.

Hexagonal pavement

On this basis, the resampling method we propose enables to preserve data around this point without totally losing surrounding data. Hexagonal cells are arranged around the focusing point as a projection of the foveal cones on the image. This covering is isotropic and cells get larger and larger according to their distance to the focusing point (Fig. 4). The cell size increases according to the distance to the focusing point because the cone cells are located on a sphere whereas the resampling cells are projected on a plane.

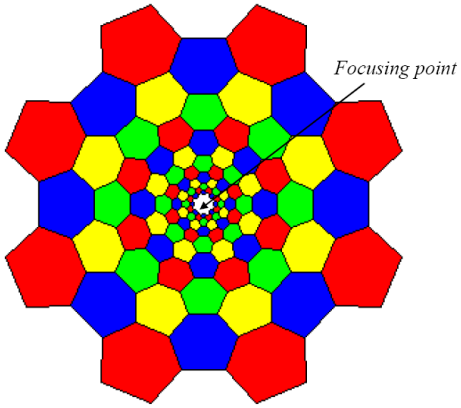

Figure 4. *Distribution of hexagonal cells around the focusing point.*

In order to summarize, this process works by firstly determining a point to focus on and a radius for the foveal area. The receptor distribution is concentric around the particular point on which the eye is supposed to focus on. Then the image is resampled in order to simulate blurring increasing on the scene when getting far from the focusing point.

Concentric Distribution of Hexagonal Cells

On Fig. 4 concentric rings of hexagons are made of either red and blue hexagons or green and yellow hexagons [12]. Then, as shown on Fig. 5 vertices of hexagonal cells belong to concentric circles of radius R_n. These radii R_n define a sequence, for $n \geq 1$:

$$R_n = R_0 \left(1 + 4.cos\frac{\alpha}{2}.sin\frac{\pi}{N}\right)^n \tag{1}$$

where N is the number of hexagons crossing the circle of radius R_n and R_0 the radius of the central cell.

Furthermore, the vertex coordinates are given by the following formulae:

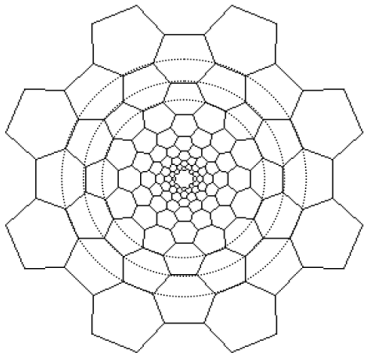

Figure 5. *Concentric location of hexagon vertices.*

$$\begin{aligned}
&A\left(R_n.cos0, R_n.sin0\right)\\
&B\left(R_{n+1}.cos\tfrac{\pi}{N}, R_{n+1}.sin\tfrac{\pi}{N}\right)\\
&C\left(R_{n+1}.cos\tfrac{3\pi}{N}, R_{n+1}.sin\tfrac{3\pi}{N}\right)\\
&D\left(R_n.cos\tfrac{4\pi}{N}, R_n.sin\tfrac{4\pi}{N}\right)\\
&E\left(R_{n-1}.cos\tfrac{3\pi}{N}, R_{n-1}.sin\tfrac{3\pi}{N}\right)\\
&F\left(R_{n-1}.cos\tfrac{\pi}{N}, R_{n-1}.sin\tfrac{\pi}{N}\right)
\end{aligned} \tag{2}$$

if A, B, C, D, E and F are the vertices of a given hexagon (Fig.6). Other hexagons are obviously obtained by rotation.

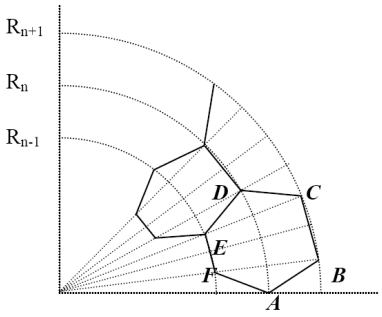

Figure 6. *Description of hexagon vertices.*

The angle α shown on Fig. 7 enables to control the flatness degree of hexagons.

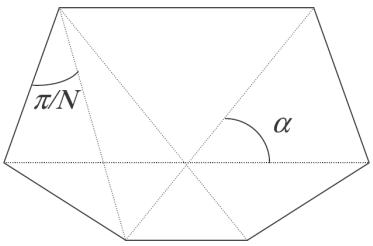

Figure 7. *Angle α defining the flatness degree.*

In order to ensure the most regularity on hexagon distribution and shape, α is set to $\frac{\pi}{3}$ and N is chosen as a power of 2, $N = 2^p$, to preserve symmetry around the focusing point. In this case, for $n \geq 1$, the R_n sequence is defined by:

$$R_n = R_0 \left(1 + 2.\sqrt{3}.sin\frac{\pi}{N}\right)^n \tag{3}$$

Data Integration

Data integration can be achieved by different ways. For example, the cell value can be the mean value (average filter), the maximal value (dilation filter), the minimal value (erosion filter) or the median value (median filter) reached on the cell [13]. There is no difficulty to set up a median filter as the encoding process works separately on each component (marginal processing), whatever the color space. The encoding step is then achieved in grey-level.

Data Encoding

Encoded data can be stored as a rectangular image, called encoded image, in which each hexagonal cell is defined by its polar coordinates.

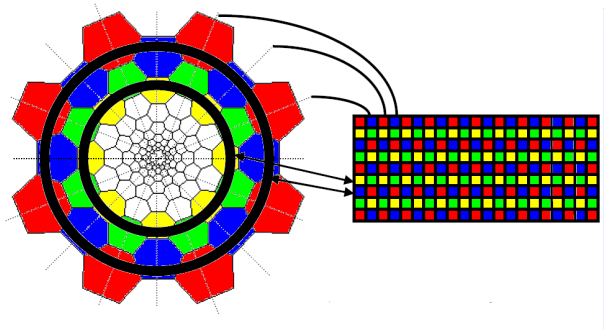

Figure 8. *Data encoding scheme.*

As shown on Fig. 8, each value of hexagonal cells is located at (θ, R_n) in the rectangular image. θ is the angle between the horizontal line and the one passing by the hexagon center. In this way the amount of data to process has considerably decreased whereas the information is globally preserved. This method can be compared to data compression with loss, but not for example with Gaussian filtering that also blurres data without decreasing their amount. Furthermore, data bordering each other on the original image still border on on the encoded image, except for those around $\theta = 0$.

Full Image Resampling on an Example

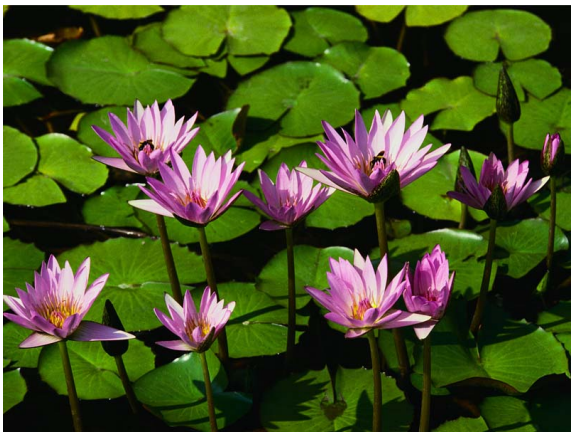

Figure 9. *Original image.*

The original image selected to show how the image resampling works is presented on Fig. 9. This image is 800×600 pixels. Fig. 10 shows the components corresponding to this original image in RGB space (Fig. 10a, b and c) and in HSV space (Fig. 10d, e and f).

The encoding parameters are the following: the focusing point coordinates are $(400, 280)$ (heart of the central flower), the radius of the central cell is set to 20 whereas the hexagon number per ring, N, is equal to 64.

Fig. 11 shows the encoded images of each component presented on Fig. 10. Each encoded image is 64×370 pixels, that means that the amount of data has been divided by 20.27. The disk at the top represents the central cell and the bottom part is the outside of this central cell. This part is not rectangular as there is a lack of points on the original image, in other words, the hexagon number in θ direction varies according to this angle value. Futhermore the four peaks at the very bottom are the four vertices of the original image.

Fig. 12 shows the recomposed images from the encoded images of Fig. 11. Data have been more and more blurred when getting far from the focusing point.

Finally, on Fig. 13, the recomposed color images are shown in RGB and HSV. These images are obtained by drawing a flood-filled hexagon at each cell location. The color value is the one computed when encoding the image. It seems that brightness and hue are much well preserved in HSV. Furthermore, the color number is of 117520 on the original image versus 15092 on the RGB recomposed image and 14484 on the HSV recomposed image. Fig. 14 shows the color histograms for the original image and the two recomposed images. The following table gives several error measures [14] between the original image and the two recomposed ones (MAE: Mean Absolute Error, MSE: Mean Square Error, NCD: Normalized Color Difference, PSNR: Peak Signal-to-Noise Ratio). We can see that error values for the RGB and HSV recomposed images are very close to each other.

Error measures between the original image (Fig. 9) and the recomposed images in RGB and HSV.

	MAE	MSE	NCD	PSNR
RGB	6.264	119.738	0.161	36.604
HSV	6.357	120.868	0.162	36.672

Comparison on a Set of Images

The following images (Fig. 15) are extracted from the Kodak image database (http://r0k.us/graphics/kodak/). They contain 768×512 pixels. The following table gives the encoding parameters:

Encoding parameters for images of Fig. 15.

	Color number	Focusing point	Description
a	25964	340, 360	T of boat name
b	37552	410, 230	flower heart
c	39784	90, 320	wood on riverside
d	14096	400, 220	island center

The focusing point location is arbitrarily chosen at a spot of interest. Color numbers are determined from the histograms. For the four images, N is set to 64 (Note that N is also the column number for the encoded image) and the radius of the central cell to 10. N is the column number of the encoded images. Fig. 16 and 17 show respectively the recomposed images in RGB and HSV after resampling. The following table gives the features of the encoded images in RGB and HSV:

Features of the encoded images of Fig. 16 and 17.

	Size	Color number in RGB	Color number in HSV
a	64×428	10780	11767
b	64×416	11663	12690
c	64×457	11461	13259
d	64×417	9451	10145

The encoded image size varies according to the focusing point position. When the distance between the focusing point and the image center increases, the row number increases as well, because for several pairs $(\theta, \theta+\pi)$ the hexagon number in direction θ is greater than the one in direction $\theta+\pi$. The image size has been divided respectively by 14.36 (a), 14.77 (b), 13.44 (c) and 14.73 (d).

The following table shows that error values are close to each other for each of the four selected images, as well as for the image of Fig. 9. Then working in the RGB space, which is the one provided by most sensors, gives results as satisfying as in the HSV space. Actually the image information is only arranged in another way when changing from RGB to HSV. More tests on a larger database have to be achieved to confirm this.

Error measures between the original images (Fig. 15) and the recomposed images in RGB and HSV.

a	MAE	MSE	NCD	PSNR
RGB	8.233	221.271	0.059	34.335
HSV	8.286	223.584	0.059	34.267
b	MAE	MSE	NCD	PSNR
RGB	10.450	466.500	0.070	26.269
HSV	10.578	472.347	0.070	26.194
c	MAE	MSE	NCD	PSNR
RGB	23.349	1212.109	0.102	17.575
HSV	23.397	1211.579	0.102	17.588
d	MAE	MSE	NCD	PSNR
RGB	9.044	221.169	0.082	34.487
HSV	9.072	220.116	0.081	34.552

Conclusion

The resampling method presented above enables to considerably decrease the amount of data included into an image, while preserving the main information. Resampled data are stored according to their polar coordinates and the encoded image can be considered as a compressed image with loss. But the encoded image contains very precisely the information inside the central cell and the compression is adaptive according to the distance to this particular cell.

Such an image encoding can be used in order to achieve preprocessing. For example a pre-segmented image can be computed from the encoded image. That enables to draw coarsely primary boundaries between the different regions of a given image. In a second time these boundaries can be refined according to the original image in order to obtain the final segmented image.

References

[1] T. Hernandez, C. Levitan, M. Banks, and C. Schor, "How does saccade adaptation affect visual perception?" *Journal of Vision*, vol. 8(8):3, pp. 1–16, 2008.

[2] A. Larson and L. Loschky, "The contributions of central versus peripheral vision to scene gist recognition," *Journal of Vision*, vol. 9(10):6, pp. 1–16, 2009.

[3] H. Yamamoto, Y. Yeshurun, and M. Levine, "An active foveated vision system: Attentional mechanisms and scan path convergence measures," *CVGIP:Image Understanding*, vol. 63(1), pp. 50–65, January 1996.

[4] F. Robert, E. Dinet, and B. Laget, "Adaptive image sampling based on the human visual system," in *Proceedings of the IS&T 50th Annual Conference*, Boston, USA, May 18-23 1997, pp. 437–442.

[5] B. Wandell, *Foundations of vision*. Sunderland: Sinauer Associates, 1995.

[6] J. Gibson, *Perception of the visual world*. Houghton, 1950.

[7] D. Marr, "Early processing of visual information," in *Proc. Royal Society of London*, vol. B275, 1976, pp. 483–519.

[8] ——, *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information*. San Francisco: W. H. Freeman, 1982.

[9] C. Curcio, K. Sloan, R. Kalina, and A. Hendrickson, "Human photoreceptor topography," *J Comp Neurol.*, vol. 292(4), pp. 497–523, February 1990.

[10] M. Levine, *Vision in man and machine*. New York: Mc Graw-Hill Publishing Company, 1985.

[11] F. Robert and E. Dinet, "A relation between distance maps and radial sampling," *Acta Stereologica*, vol. 185/3, pp. 351–360, November 1999.

[12] ——, "Biologically inspired pavement of the plane for image encoding," in *Proceedings of International Conference on Computational Intelligence for Modelling, Control and Automation (CIMCA'99)*, Vienna, Austria, February 17-19 1999, pp. 1–6.

[13] J. Roerdink, "Computer vision and mathematical morphology," *Computing, Supplement*, vol. 11, pp. 131–148, 1996.

[14] L. Jin and D. Li, "A switching vector median filter based on the cielab color space for color image restoration," *Signal Processing*, vol. 87, pp. 1345–1354, 2007.

Author Biography

Frédérique Robert-Inacio received her Master degree in applied mathematics in 1988 and her PhD in image processing in 1992 from University Jean Monnet, Saint-Étienne, France. She worked as a researcher at the University of Sherbrooke, Canada, from 1993 to 1994, and since September 1994, she has been a lecturer at the ISEN. Her present spots of interest concern pattern recognition, color imaging and computer vision, and more accurately image processing furthering the estimation of parameters characterizing geometrical features for shape classification.

Quentin Stainer and Rémy Scaramuzzino are presently students in Master Degree in Electronics and Computer Science at the ISEN. Their fields of interest are directly related with image processing and computer vision.

Edith Kussener received the diploma in microelectronics engineering from the Institute Superior Electronic et du Numeric (ISEN), Toulon, France. She received the M.S in telecom and Ph.D degrees in microelectronics from the University of Sciences and Technologies, Lille in 1998 and 2002, respectively. Her doctoral research was on the on chip DC-DC voltage converter for smartcard application, work realized at STMicroelectronics, Rousset, France. In 2001, she joined ISEN high school and the IM2NP laboratory , as an Analog Designer, where she works on analog power conversion and ultra low power.

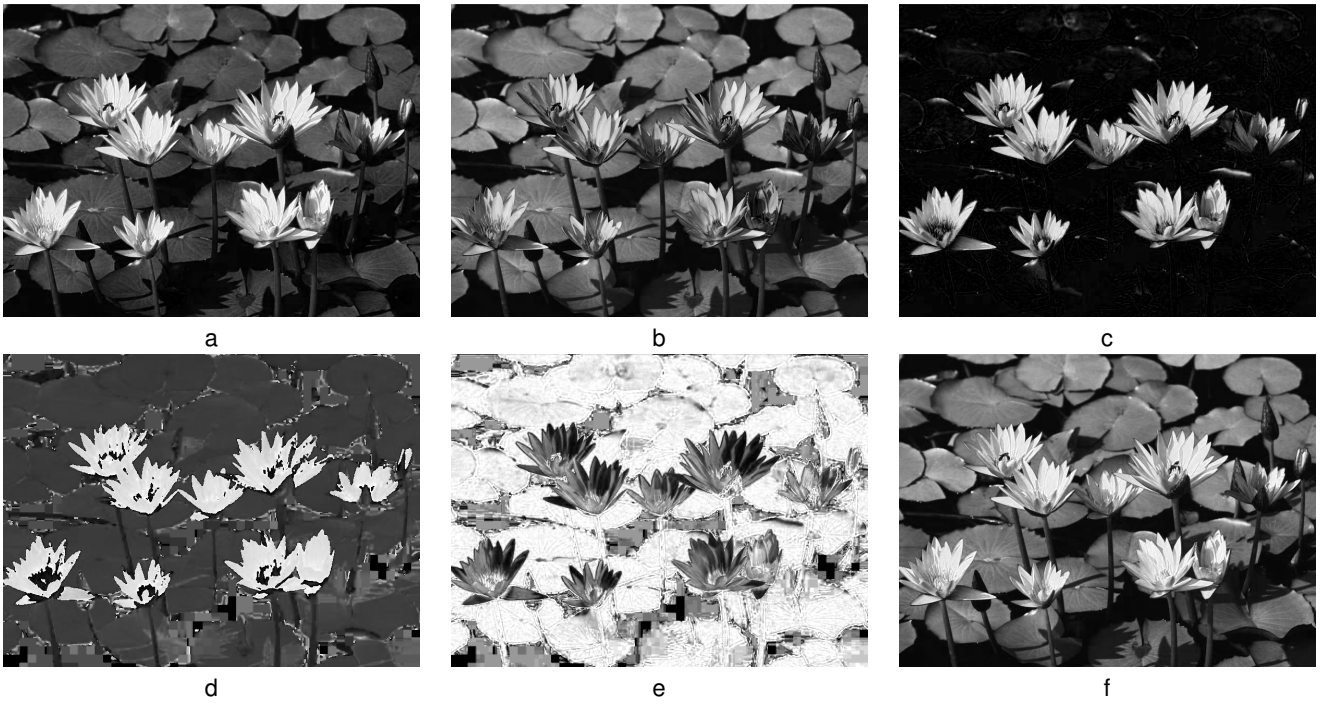

Figure 10. *Components in Red (a), Green (b) and Blue (c), and in Hue (d), Saturation (e) and Value (f) of the original image (Fig. 9).*

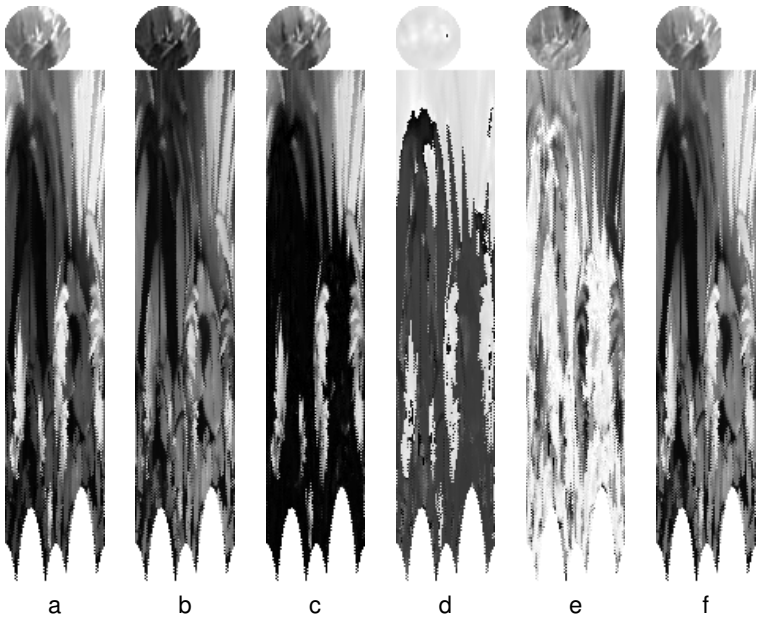

Figure 11. *Encoded components in Red (a), Green (b) and Blue (c), and in Hue (d), Saturation (e) and Value (f).*

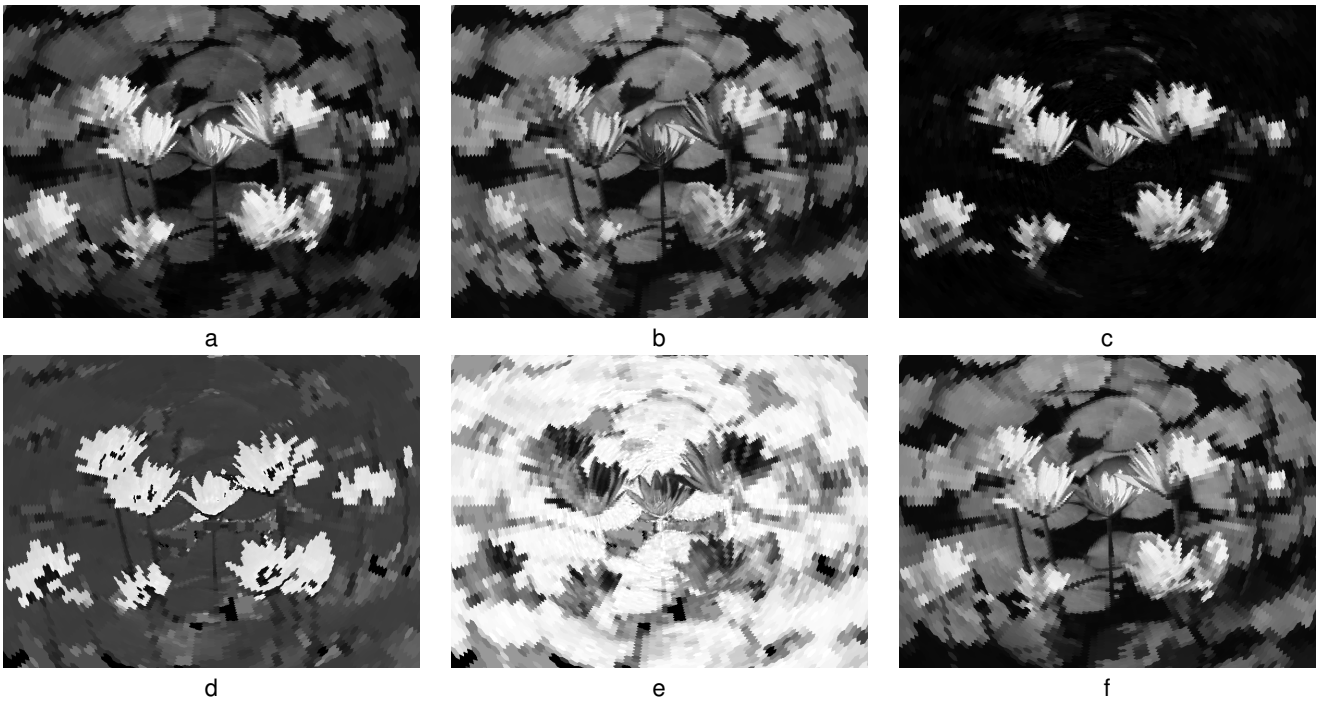

Figure 12. *Decoded components in Red (a), Green (b) and Blue (c), and in Hue (d), Saturation (e) and Value (f) of the encoded images (Fig. 11).*

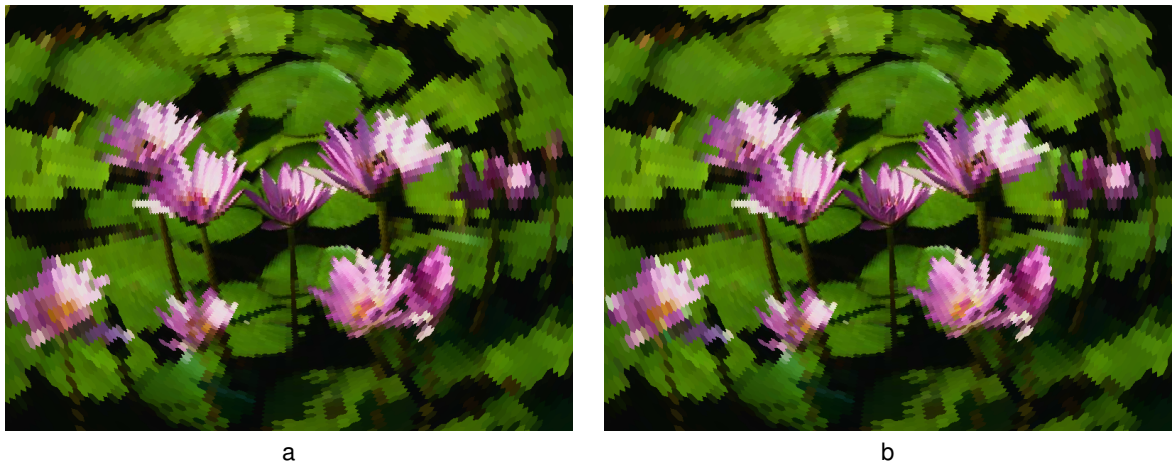

Figure 13. *Recomposed images in a) RGB and b) HSV.*

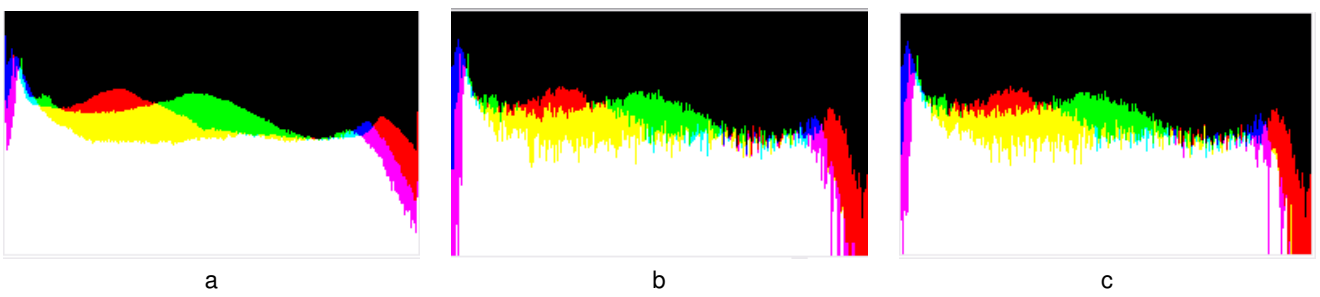

Figure 14. *Color histograms of the a) original image, b) recomposed RGB image and c) recomposed HSV image.*

Figure 15. *Original images from the Kodak database.*

Figure 16. *Recomposed images in RGB.*

Figure 17. *Recomposed images in HSV.*