

Random Paintbrush Transformation

Tamás Szirányi

*Analogical Computing Laboratory, Comp. & Automation Inst., Hungarian Academy of Sci., H-1111 Budapest, Kende u. 13-17, Hungary
E-mail: sziranyi@sztaki.hu*

Zoltán Tóth

University of Veszprém, Department of Image Processing and Neurocomputing, H-8200 Veszprém, Egyetem u. 10, Hungary

Abstract

A paintbrush-like image transformation is proposed in this paper. It is based on a random searching to insert brush-strokes into a generated image at decreasing scale of brush-sizes, without predefined models or interaction. One of the goals of the method is to transform the image into a representation that is very similar to the human sensation of artistic images. We introduce a sequential multiscale image decomposition method, based on simulated rectangular-shaped paintbrush strokes. The resulting images look like good-quality paintings with well-defined contours, at an acceptable distortion compared to the original image. The image can be described with the parameters of the consecutive paintbrush strokes, resulting in a parameter-series that can be used for compression. The painting process can be applied for scale-space image representation, segmentation and contour detection, and image representation for retrieval purposes.

1. Introduction

Images can be interpreted in several ways by decomposition into basic functions: strokes [4, 5], fractals [1], etc. Each of these is natural in some sense: strokes are good representations of letters or shapes, Gabor-functions are natural for human sensation, fractals originate from the self-similarity. When we are looking at some images or image scenes, we usually search for familiar features. This is exploited in fine arts: small details are sometimes neglected while the main features are enhanced.

Nonlinear partial differential equations [6] can be used for enhancing the main image structure. When compressing the image, anisotropic diffusion, based on the scale-space paradigm, can enhance the basic image features to get visually better quality [8].

When observing an image, our visual system [11] is searching for small image-fragments to describe the scene: contours described by lines and curves, patches, simple textures, strokes. How can we interpret an image?

There is an offering answer for this question from an artistic point of view: ask a talented painter and he/she will give a painted interpretation of the word: the scene as the artist sees it. Certainly, paintings and drawings are very different in their nature. A good artist can sketch the picture with one curved line making a drawing. In this paper we shall deal with images painted detail by detail only. Such an image is made of brush-strokes of different sizes and colors put on the screen one after the other in a sequence by the painter. Small articles are elaborated with fine brushes, while plain surfaces are painted with greater strokes.

In this paper we try to follow a painter's process by using simplified artificial strokes. It is not an image-filtering to get a painting-like isotropic transformation. Other image-painting methods of computer graphics deal with model- or edge-controlled methods [e.g. 3,7]. These methods can serve interesting effects, but they are far from the quality of a careful painter (e.g. old-fashion portrait painters or S. Dali). Our first goal is to simulate the painting process to get a picture similar to a real painting, where the purpose of the painter is to portray something, which looks like to be a real scenery. We deal with a method copying the real visual world into a pleasant form rather than with the artistic interpretation of some special style. On the other hand, the sequential parameters of the consecutive strokes can be applied for image description or some moderate compression as well. We can see that our method enhances the main features, which can easily be followed by the eye. The strokes guide our sight.

2. The basic concepts

Here we'd like to take into considerations the human behavior and sensation when looking at and reproducing the visual word. When looking at an image, then there is a very usual way of understanding the visual information:

- Looking at the main outline of the image,

- Looking for the objects,
- Finding small details,
- Wandering and roaming on the image.

When interpreting the image content in a visual form, like representational painting, the re-creation of the image can proceed a similar way:

- Outlining the main areas,
- Elaborating objects,
- Refining the small details.

In both cases (understanding and reproducing) there is a well-defined scale-space line. First the large details, later the finer details are proceeded. In this aspect the representational painting is similar to the anisotropic diffusion that is based on the scale-space theory [6] based on PDEs. However, there is a very important difference between the two scale-space approaches, namely:

Anisotropic diffusion enhances the main edges but smoothes the others, while the present paintbrush transformation has sharp edges at any stage. We get sharp details even in the case of small or low-contrast areas. In the latter sense representational painting gives better segmentation than the anisotropic diffusion. The effect is similar to that of Markov Random Field segmentation [2]. However, MRF has the drawback that the possible number of “colors” could be maximum cc. 10 to achieve good segmentation in a finite time.

The other important viewpoint is the question of the finest details of an image. When the image contains too many fine details (e.g. sharp photo), our sight may be perturbed by the unnecessary small details. If the image is compressed before by a function-set of e.g. Cosine or Wavelet functions, then the image results in very annoying artifacts in the range of fine details. So, we have fine details, sometimes they are over-defined, sometimes they are over-informed. We can only remove the annoying fine information contents at the cost of smoothing the sharp segment-contours. A very practical effect is the case when we are looking at an image from a distance:

- In case of artwork painting, the visual effect could be perfect, giving high contrast to represent the objects;
- Getting closer to the painting, we see sharp edges (paintbrush-strokes), but there is a point, where there are no more fine details behind it. We automatically stop to get closer, and it is not annoying;
- We can enjoy and relax when looking at the image, since there are no details, which are vibrating when scanning through.

When defining what we hope from a new process that follows the main features of representational painting, we can define the main concepts of the new algorithm:

1. It should have sharp edges at any level of image-construction;

2. There are no fine details below a limit;
3. There are sharp edges at the finest level as well;
4. From a given distance the image must give the same visual scenery as the original.

In the following we can see that the above constraints can be fulfilled by a method, which follows the generation of painting by using different sizes of paintbrush strokes.

3. The algorithm

Main steps of the algorithm are following to generate paintbrush strokes:

- 1) *Starting with the rawest brush-size;*
- 2) *Selecting the next, smaller brush-size (δ);*
 - *If the finest scale is over then Goto 14;*
- 3) $C_{\varphi\delta}$: *Convolving the I input image by brush-distributions depending on orientation φ and brush-size δ . At δ brush-size we have at least 8 $C_{\varphi\delta}$ maps due to the φ orientations. This map-series is necessary to estimate the brush-color anywhere in the image without further brush-stroke tuning.*
- 4) D : *Difference image between the original I and the present iterated stage X ;*
- 5) A : *Absolute or square values of the D difference image to get a distortion-map;*
- 6) *If the error-summation over A is smaller then a limit*
 - *then Goto 2;*
- 7) E : *Convolving the A error image by a smoothing due to the diameter of the δ brush-size;*
- 8) *Calculating histogram of the distortion-map E , defining a threshold value where the probability of higher errors is ε ;*
- 9) *Randomly choose a x,y position in the image and a φ brush-orientation;*
- 10) *If the distortion value of E at the pixel position of x,y is in the upper region of the distortion-histogram with probability ε then*
 - *give the color of $C_{\varphi\delta}(x,y)$ to the brush-stroke centered at position (x,y) with the actual φ orientation and δ brush-size;*
 - *or, in case of large strokes, give the color of the majority vote in the brush-area of the original I image to the brush-stroke centered at position (x,y) ;*
- 11) *Cover the painted image by the pattern of this brush-stroke;*
- 12) *If the error-summation between the original and present stage images over the area of this stroke is smaller then a limit*
 - *then accept the new stroke;*
 - *else reject and restore the previous stage in the stroke-area;*

- 13) If the counter of the brush-strokes is smaller than a limit-number
 - then Goto 9 ;
 - else Goto 4 ;
- 14) Ready.

We can see that our algorithm is stochastic, error-controlled and multiscale. In our present experiments brush-strokes are simple rectangles. Since strokes need relatively great convolutions, the $C_{\phi\delta}$ maps are generated in the Fourier domain.

The method has been tested for several parameter sets and test images. Some of the resulted images can be found in Figure 1. The “painted” images seemed to have high

quality at the last phase with the finest brush. However, at every stage the edges are sharp and the patterns and textures are appropriate. It is interesting to note that the sharp edges and patterns can be generated without any a priori definitions of contours or textured places.

This method demonstrates that a fully random searching process generating brush-strokes can result in a high quality and pleasant-to-see image. This method takes about 10-30 minutes on a Pentium III PC when generating a 512x512 image. However, it can be easily implemented in parallel processor-arrays at high speed, like MRF-based algorithms in [9].



Original image Intermediate with larger brush-size Finished image with small brush-size

Figure 1. *Painting stages at different brush-sizes of image “Lena” and “Goldhill”*

Number of the scales of brush-strokes was up to 10

4. Segmentation by the transformation

The transformation in the present form (rectangular uniform brush-strokes) generates pattern-formations of sharp edges. Figure 2 shows the effect for test image “Rowboat”, where we get good contours for the object-

edges, even in the places of the original image where it is hard to find contours. In Figure 1 we find that the final brush-size of the painting process determines the scaling of the contour-detection. In this sense, this transformation is a scale-space feature-detection method.

We can modify the algorithm to support color-selection considering the neighborhood of a brush-stroke. Constituting two error-measures, namely: distortion from the original area and difference around the perimeter, we can run an MRF-like optimization by using brush-stroke patches. In this way we have got larger contiguous areas and good segmentation.

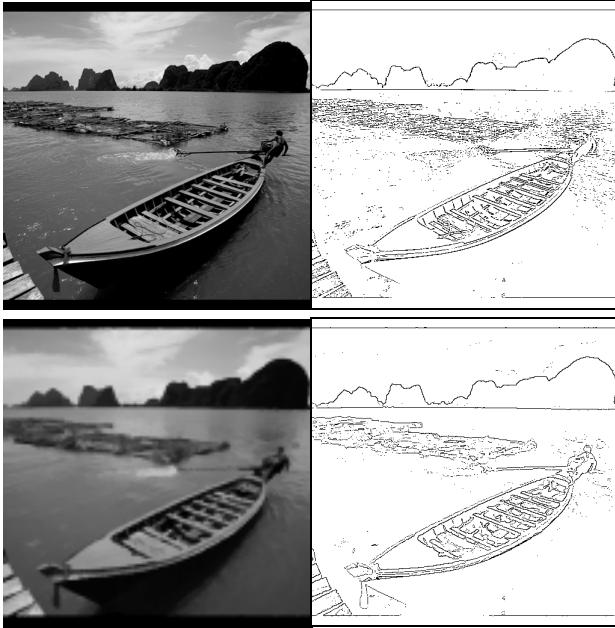


Figure 2.

The original image “Rowboat” (up), its painted copy (down) and their edge maps

5. Compression

When generating the “painted” image consisting of a series of paintbrush strokes, the resulting image can be a good quality one with some unique image structure. This can be applied to describe the image. The main steps of the compression-like algorithm are:

1. Series of paintbrush strokes
 - Grouping parameter: δ brush-size
 - Placement parameters: position x and y and ϕ orientation
2. Reverse checking to delete covered strokes, to eliminate redundancies
3. ZIP code (statistical compression) of the parameter series

The diagram in Figure 3 shows that the above method is a progressive coding. However, the present first-attempt coder has poorer compression quality in PSNR than other well-known methods [e.g. 1].

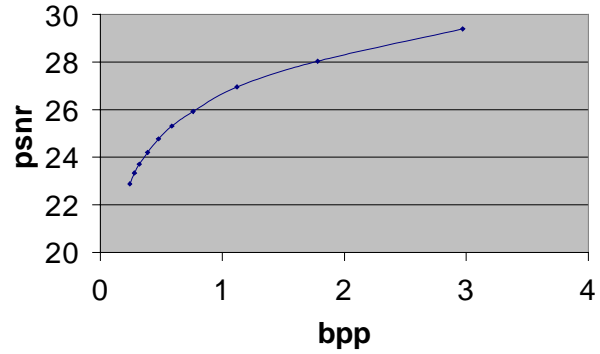


Figure 3.

Compression ratio (depending on the size of the final brush-strokes) vs. SNR when “Lena” is transformed into brush-strokes

6. References

- [1] Y. Fisher. (ed.), *Fractal Image Compression*, Springer Verlag, 1994.
- [2] S. Geman, D. Geman, “Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images,” *IEEE Tr. PAMI*, Vol. 6, 1984, pp. 721-741.
- [3] P.E. Haeberli, “Paint by numbers: Abstract image representations”, *Computer Graphics*, (Proc. SIGGRAPH’ 1990), V.24, pp.207-214, 1990.
- [4] H. H.S. Ip, H. T. F. Wong, “Generation of Brush Written Characters with fractal Characteristics from True-type Fonts”, *ICCPOL, 17th Int. Conf. on Computer Processing of Oriental Languages*, pp. 156-161, Hong Kong, April, 1997.
- [5] H. H S Ip, H. T F Wong, “Calligraphic Character Synthesis using Brush Model”, *CGI97, Computer Graphics International conference*, pp. 13-21, Hasselt-Diepenbeek, Belgium, June 23-27 1997.
- [6] T. Lindeberg, B. M. Haar Romeny, “Linear scale-space I-II”, *Geometry - Driven Diff. In Computer Vision, Kluwer Academic Publishers*, 1992, pp.1-72.
- [7] P. Litwinowicz, ‘Processing Images and Video for An Impressionist Effect’, *Computer Graphics*, (Proc.SIGGRAPH’ 1997), pp.407-414, 1997
- [8] T. Szirányi, I. Kopilovic, B. P. Tóth, “Anisotropic Diffusion as a Preprocessing Step for Efficient Image Compression”, *Proc. of the 14th ICPR*, Brisbane, *IAPR&IEEE*, Australia, pp. 1565-1567, August 16-20, 1998.
- [9] T. Szirányi, J. Zerubia, “Markov Random Field Image Segmentation using Cellular Neural Network”, *IEEE CAS I*, Vol. 44, pp. 86-89, January, 1997.
- [10] P. Teo, D. Heeger, “Perceptual Image Distortion”, *First IEEE Int.Conf. Image Proc.*, Vol.2, 1994, pp.982-986
- [11] F. Werblin, A.Jacobs, J.Teeters, “The computational eye”, *IEEE Spectrum*, pp.30-37, 1996