Image vectorization using blue-noise sampling

Jiaojiao Zhao^a, Jie Feng^{ab} and Bingfeng Zhou^{ab*} ^aThe institute of computer science and technology, Peking University; ^bThe Key Laboratory of Machine Perception (Ministry of Education), Peking University; Beijing 100871, P.R. China

ABSTRACT

Current image vectorization techniques mainly deal with images with simple and plain colors. For full-color photographs, many difficulties still exist in object segmentation, feature line extraction, and color distribution reconstruction, etc.

In this paper, we propose a high-efficiency image vectorization method based on importance sampling and triangulation. A set of blue-noise sampling points is first generated on the image plane by an improved error-diffusion sampling method. The point set well preserves the features in the image. Then after triangulation on this point set, color information can be recorded on the mesh vertices to form a vector image. After certain image editing, e.g. scaling or transforming, the whole image can be reconstructed by color interpolating inside each triangle.

Experiments show that the method has high performing efficiency and abilities in feature-preserving. It will bring benefits to many applications, e.g. image compressing, editing, transmitting and resolution enhancement.

Keywords: Image vectorization, blue-noise Sampling

1. INTRODUCTION

In computer graphics, images are usually represented in two different ways: pixel-based raster images and geometrybased vector images.

Raster image is a fundamental form of 2D image representation. A raster image is composed of a colored pixel matrix which may be directly mapped onto displaying devices, and hence it has high rendering speed. However, it would also seriously suffer from aliasing or loss of details when the image is scaled.

In recent years, vector image, which is made up of a group of colored geometry primitives (points, curves or polygons), becomes an effective alternative for raster image in many circumstances. Vector image is a compact representation, requires less bit rate, and is independent to the displaying resolution. Therefore high-quality images can be rendered in any scale without aliasing. Because of the advantages of more compact, scalable, editable and easier to animate, vector images have been increasingly adopted in personal computers and on the Internet, and high-efficiency image vectorization becomes an important issue in both academic and commercial research.

Early image vectorization techniques mainly deal with cartoon arts or line drawings that contain only simple and plain colors [1][2][3][4]. In recent years, researchers also proposed some methods to perform vectorization on full-color photographs [5][6][7]. However, many difficulties still lie in object segmentation, extraction of the feature lines, the fitting of the controlling curves to the features, and the reconstruction of the color distribution inside each primitive or between controlling curves.

In general, an ideal image vectorization method should possess the following properties:

- 1) Accurate approximation for both sharp features and smooth areas of color variation in a full-color image.
- 2) Constructing high-quality vector representation with less geometry primitives.
- 3) Automatic image vectorization that depends on less user interaction and parameter adjustment.
- 4) High rendering speed which is comparable to raster images.

^{*} Email: {zhaojiaojiao, feng_jie, cczbf}@pku.edu.cn

In this paper we propose a new and efficient image vectorization method. Here we adopt a triangle-mesh-based representation in image vectorization, for triangle is the most flexible primitive and advantageous in preserving features and fast rendering. The triangulation is performed in the image plane, on a set of *blue-noise sampling points* generated by an adaptive sampling algorithm, e.g. an improved error-diffusion [8]. Such a sampling point set has excellent distribution property and can preserve both the edge and the internal features in the image, and consequently may result in an ideal Delaunay triangulation. Then the colors in the image can be recorded on the corresponding sampling points to form a vector image. After certain image transformations, e.g. scaling, the topology of the triangle mesh does not change, hence the features of the image are maintained, and the whole image can be reconstructed by color interpolating inside each triangle.

In the follow of the paper, we first give a brief review of the related work in image vectorization. Then in Section 3, an overview of our method is described. The details of the method are given in Section 4 and 5, followed by some experimental results in Section 6. Finally there is a summary and some discussion in the last section.

2. RELATED WORK

In prior research work, a common way of 2-D full-color image vectorization is to convert the image plane into a mesh, which may be a triangular, quadric, gradient or some other form of mesh. Then the corresponding information in the image, including colors and gradients etc. can be stored in the mesh vertices to constitute a vector image.

Currently, existing image vectorization methods may be classified into three main categories:

2.1 Triangle mesh based methods

Triangle mesh is the most basic 2-D mesh, for triangle is the simplest and most flexible 2-D primitive. Hence a straightforward way is to construct vector image on the basis a proper triangulation of the image plane.

This category of methods decomposes the image into a group of triangles and stores the corresponding pixel colors on the mesh vertices. Inside each triangle, the color of each pixel can be recalculated by interpolation. Due to the limited representation power of the color functions defined over each triangle, triangulation-based methods usually have to generate a large number of small triangles to describe the detailed color variations and the feature lines in complex full-color images, e.g. photographs.

In the work of Xia et al. [6], they made an improvement on the basic triangle mesh by replace the triangles with a group of parametric triangular patches which have curved boundaries. The boundary curves are fitted to the extracted feature lines of the image. Therefore, this method may produce more adaptive meshes, support flexible topology and more smooth color distribution inside each patch. There are still some shortcomings lies in this approach. First, there is no scale information associated with the detected edge features, which is highly related to the vectorization quality. Second, the thin-plate spline fitting in this algorithm is not fully optimized.

2.2 Gradient mesh based methods

Another category of methods use higher-order parametric surface mesh for vectorization representation, such as quadric mesh, gradient mesh, etc. In these object-based methods, the whole image is segmented into several objects, and each one is represented by respective mesh.

Sun et al. presented a semi-automatic vectorization algorithm [9]. With the exploitation of gradient mesh, the number of quad is reduced. However, since non-linear optimization is included, this algorithm is time-consuming and is highly depended on the manual initialization.

Then, Lai et al. [7] introduced a topology-preserving gradient mesh representation. This method preserves the topology, and the objects with holes are not required to be cut into sub-objects. On another aspect, it uses the concepts of image manifolds and surface parameterization to generate gradient meshes in a fully automatic way and no user's manual guide is needed. Hence the method works much faster than previous approaches. However, the method requires rectangular arrangement of patches, which brings unnecessary restrictions to achieve a highly adaptive spatial layout, and makes it difficult to automatically align internal patch boundaries with detailed features inside an object layer. Moreover, the mesh grid may be distorted at complicated boundaries and features. Hence it is not suitable for fine and rapidly changing details. The object segmentation is also a great challenge for complex images.

2.3 Diffusion curves

Different from the mesh-based methods mentioned above, Orzan et al. [10] proposed a novel mesh-free representation which is called diffusion curves.

A diffusion curve consists of three parts: a cubic Bezier spline specified by a set of control points; two sets of color control points corresponding to color constraints on two half space of the curve; and a set of blur control points defining the smoothness of the transition between the two halves. Given a set of diffusion curves, a final image is calculated by solving a Poisson equation. Comparing with the traditional vectorization method, the diffusion curves are allowed to intersect and un-closed. Hence, relying on certain edge detection algorithms, the original image can be converted into diffusion curve representation.

However, problems with the Poisson equation solver may occur in regions with relatively sparse features. Secondly, auto-detected edges may have undesirable spatial distributions: some regions may have overly dense edges, while others may have too few. This will increase the complicity of the algorithm. Moreover, edge-based representation does not guarantee closed regions, making it infeasible to perform region-based editing.

3. OVERVIEW OF OUR METHOD

In this paper, we propose an efficient image vectorization method for full-color images, based on a triangle mesh representation and importance sampling.

Here we adopt the triangulation-based representation because triangles are the most flexible primitives. They are advantageous in preserving features in the image and improving the rendering speed. Therefore, the number and the distribution of the mesh vertex will significantly affect the error of image vectorization and its visual quality. This is guaranteed by importance sampling in our method.

Our method has three important features:

First, the density of the triangle mesh vertices is controlled by an *importance function* which is defined by the structure and color features in the image. In the areas with significant features, the importance value and the vertex density will be higher, while in the smooth areas, the density will be lower.

Second, on the basis of the importance function, an improved variable-coefficient error-diffusion method [8] is performed on the image plane, and generates a sampling point set with ideal blue-noise distribution. These sampling points will perfectly preserve the edge and internal features of the image.

Finally, using the sampling points produced by the error-diffusion method as the vertex set, an optimized triangle mesh can be generated by a Delaunay triangulation method. The optimized triangle mesh, along with the color value corresponds to each vertex, constitute a vector representation of the original raster image.

The main steps and the workflow of our method are illustrated in the chart in Fig.1.



Figure 1. The workflow of our image vectorization method based on importance sampling and triangulation.

4. IMPORTANCE SAMPLING

As mentioned in the last section, the distribution and the density of the mesh vertex will directly affect the quality of the image vectorization. In our method, we obtain an optimized vertex distribution by importance sampling. An importance function is defined to preserve features in the original image, and an improved error-diffusion sampling method is adopted to generate the vertex set according to the importance function.

Here we first give a brief review of the adopted error-diffusion sampling method, and then describe the definition of our importance function.

4.1 Improved error-diffusion sampling method

Error diffusion [11] is originally a halftoning technique, which quantizes a multilevel image into a binary image while preserving its visual appearance. It put image pixels one by one through a quantization function and the residual error is distributed to the unprocessed neighboring pixels. The method is invented for gray scale image display and printing, but also frequently used in many other areas as a discrete domain sampling algorithm because of its high efficiency.

In 2001, Ostromoukhov [12] improved the original error diffusion algorithm by using variable coefficients for each different grayscale level. This greatly increased the quality of halftoning, especially in the light-tone.

Later, Zhou and fang [8] made further improvement on the basis of Ostromoukhov's work. In this method, the quantization threshold, which is a fixed value in the original algorithm, is modulated by a random perturbation. Controlling the amplitude of the perturbation for each grayscale level, high-quality halftoning output with 'blue-noise' property [13] will be obtained, and the visual artifacts are eliminated especially in mid-tone areas.

In this paper, we mainly use Zhou and Fang's threshold modulated error diffusion [8] as the sampling method to generate the vertex distribution for the triangle mesh. For convenience, we refer this method as the *improved error diffusion*. For color images, the sampling of the mesh vertices are performed in the three color channels of the color image separately. We also try to taken into account the correlations between the three channels during the sampling, so that the resulting point set in different channels would not overlap, and maintain ideal blue-noise property as a whole. This is achieved by using another method called *multi-class error-diffusion* [14].

4.2 The importance function

The sampling point distribution generated by error diffusion should be modulated according to the features of the image to make the feature lines and boundaries in the image better retained. More sampling points should be created in the areas with dense and sharp features, and the smooth areas should have less sampling points. This is accomplished by defining an importance function to control the local density of the sampling points in the error-diffusion sampling process.

The value of the importance function is determined according to the structure and color features in the original image. For this purpose, we first detect the boundaries and feature lines in the image. In this paper, we adopt an improved Sobel operator to detect these features and calculate the absolute gradient value on each pixel.

-1	-2	-1		-1	0	1		2	1	0		0	1	2
0	0	0		-2	0	2		1	0	-1		-1	0	1
1	2	1		-1	0	1		0	-1	-2		-2	-1	0
(a)			1	(b)			1	(c)			1	(d)		

Figure 2. The four templates of the improved Sobel operator.

The standard Sobel operator includes only two templates: one for detecting vertical lines and another for horizontal lines (Fig. 2(a)-(b)). As shown in Fig. 2(c)-(d), the improved Sobel operator contains two more templates for 45° and 135° feature lines. Hence the detected features will be more accurate and more details can be maintained.

For each template, an approximating derivative G_i (i = 1,...,4), is calculated by convolution. Then, the gradient value for the current pixel is calculated as:

$$|G| = \max(|G_1|, |G_2|, |G_3|, |G_4|)$$
(1)

The gradient values reflect the saliency of the features in the image. Hence we may define the importance value (i.e. the sampling density) on the basis of these values. However, the calculated gradient values are non-negative integers, while the input range of the error-diffusion algorithm is [0, 255]. Therefore, the importance function F is in fact a mapping from $[0, \max]$ to [0, 255].

In a straightforward way, the function may be simply defined as a linear mapping:

$$F(x) = \frac{x}{\max} * 255, \ x \in [0, \max] \text{ and } x \in Z^+.$$
(2)

But from experiments we found that the gradient value distribution on [0, max] is extremely uneven. Hence a large part (about 20%-30%) of the values will result in 0 after such a linear mapping, and consequently too few sampling points will be generated (as show in Fig. 3(a)).

To achieve an appropriate number of sampling points, we adopt an un-linear mapping in our method:

$$F(x) = \left(\frac{x}{\max}\right)^{\frac{1}{\gamma}} * 255, \quad \gamma > 0, x \in [0, \max] \text{ and } x \in Z^+,$$
(3)

where γ is a constant that controls the sensitivity of the function. Experiments prove that this function works better than the linear one. Only less than 15% of the pixels will have an importance value of 0 after the mapping. Besides, the number of the resulting sampling points can be well controlled between 10%-15% of the total number of the pixels, which is an acceptable proportion. A comparison of the sampling results produced by the two functions is given in Fig. 3.



Figure 3. The importance sampling results and their corresponding histograms using different importance functions: (a)&(b) Linear mapping produces too few sampling points; (c)&(d) Un-linear mapping produces appropriate number of sampling points.

4.3 The sampling of the vertex set

To better preserve the feature lines and boundaries in the original image, we do not sample the image directly. Instead, we firstly use the importance function value that calculated from the gradient information to generate a feature image,

namely a *salience map*. The salience is computed in the three color planes separately, and the resulting salience map is also an RGB image, as shown in Fig. 4(b). Then, the error-diffusion sampling is performed on the salience map and produces the final sampling point set that will be used as the vertices of the triangle mesh (Fig. 4(c)).



Figure 4. Building and rendering vector image. (a) Original raster image; (b) Salience map; (c) Importance sampling point set; (d) Triangulation on the sampling point set; (e) Rendered vector image.

5. BUILDING AND RENDERING VECTOR IMAGES

When the sampling points on the image plane are generated by the error-diffusion method according to the importance function, a Delaunay triangulation algorithm will be performed by taking these sampling points as the mesh vertices (Fig. 4(d)). Since the blue-noise distribution property of the point set is guaranteed by the sampling method, the resulting triangulation also possesses good quality.

Then, the colors, gradients or other information of the image can be recorded in the corresponding sampling points and form a vector image representation. It is notable that, when the image is put through certain transformations such as scaling or tilting, the topology of the sampling point set and the triangle mesh do not change. That means the features of the image will be well preserved.

Finally, with proper texture interpolation inside each triangle, we can reconstruct the image in any scales. In this paper, we use a bilinear interpolation method to calculate the color of each pixel inside a triangle, according to the colors of its three vertices (Fig. 4(e)).

6. EXPERIMENTAL RESULTS

We apply our importance sampling based vectorization method on a group of input raster images. Firstly, we use the improved error-diffusion [8] as the sampling algorithm, and some experimental results are demonstrated in Fig. 5. In each example, the calculated salience map and the sampled vertex set are also given. The parameters and the resulting sampling point numbers are listed in Table 1, from which we can also find that our vector representation obtains satisfying compression rate and will save a great deal of storage space.

Because of the simplicity of the triangle mesh and the high computing efficiency of the error-diffusion method, the building and the rendering of the vector images can be completed in real-time. Furthermore, our method can work automatically. Only one optional parameter, i.e. the value of γ in Eq.(3) is needed to be determined in advance, which will affect the sampling rate of the image: smaller γ will result in more sampling points.

From the examples in Fig. 6 we can see that our method is able to produces comparable result with other prior works. The quality of the reconstructed images is satisfying while our method is much more concise and efficient. It is notable that there are obvious color blocks existing in the results from Lai et al. 2009 [7], while our rendered vector images have smoother color distribution that is closer to the original image.

As mentioned in Section 4, the improved error-diffusion [8] algorithm is performed in the three channels of the salience map separately. Some problems may occur when using this strategy. For instance, sampling point in different channels may overlap and reduce the final valid point number. Hence, we also try to utilize a multi-class error-diffusion method [14] to solve this issue.



Figure 5. Some experimental results generated by our vectorization method. In each group, (a) is the input raster image, (b) is the calculated salience map, (c) Sampling point set by improved error-diffusion, and (d) is the rendered vector image.

Name	Original size	Value of y	No. of points	Compression rate
Bronze	500x500	1.0	21532	20.1%
Flower	500x500	0.8	23795	22.21%
Lily	500x500	0.8	18457	17.23%
Candle	500x500	0.6	26845	25.06%

Table 1. Experimental Parameters of Figure 5.



Figure 6. Comparison of reconstructed images by our method with some prior works.

Fig. 7 shows an example of using multi-class error-diffusion. The sampling processes in the three channels are performed simultaneously and no sampling point will overlap with others. Hence, from the same salience map, more valid sampling points are generated. Consequently, rendering results with higher quality can be achieved and more sharp details are preserved after reconstruction.

In Fig. 8 we adjust the value of γ and make the point numbers generated by the two sampling algorithms nearly equal. In these examples we can see that the multi-class method still produce better rendering results, especially in the areas with sharp edges.

7. CONCLUSION AND FUTURE WORK

In this paper, we proposed an image vectorization method based on importance sampling and triangulation-based representation. Experimental results show that the method has high performing efficiency, ability in feature preserving and satisfying reconstruction quality. It can produce comparable results with other prior works, while the algorithm is much more concise, efficient and automatic.

The vector image representation is compact. Hence it may also bring benefits for reducing storage, computing and transmitting costs, especially in the environments with limited computational resources.

From the experiments we found that for color images, the multi-class error diffusion method can produce sampling point set with higher quality than simply apply the improved error diffusion. The utilizing of this new sampling method is not fully explored in this paper, and that will be an important work for us in the future.



(b) Salience map



(d) Improved E-D



(f) Multi-class E-D

Figure 7. An example of using multi-class error-diffusion sampling. (c) & (d): Sampling and rendering result of improved errordiffusion; (e) & (f): Result of multi-class error-diffusion. In both results $\gamma=0.8$.

Improved error-diffusion Multi-class error-diffusion Improved error-diffusion Multi-class error-diffusion (y=1.0, 73425 points) (y=0.28, 85085 points) (y=0.7, 85075 points) $(\gamma = 0.48, 73434 \text{ points})$

Figure 8. Comparison of the rendering results generated by the two sampling algorithms, at similar number of sampling points. Multi-class error-diffusion method preserves more sharp details.

Furthermore, for better visual effect, hierarchical / progressive sampling and triangulation may also be considered in our vectorization and reconstruction process. For some images with obvious directional feature lines, anisotropic sampling may also bring benefit in feature preserving and point number controlling.

REFERENCES

- [1] Dori, D., and Liu, W. "Sparse pixel vectorization: an algorithm and its performance evaluation". IEEE Transactions on Pattern Analysis and Machine Intelligence 21, 3, 202–215 (1999).
- [2] Zou, J. J., and Yan, H. "Cartoon image vectorization based on shape subdivision". Proc. Computer Graphics International, 225–231 (2001).
- [3] Hilaire, X., and Tombre, K. "Robust and accurate vectorization of line drawings". IEEE Transactions on Pattern Analysis and Machine Intelligence 28, 6, 890–904 (2006).
- [4] Zhang, S.-H., Chen, T., Zhang, Y.-F., Hu, S.-M., and Martin, R. R. "Vectorizing cartoon animations". IEEE Transactions on Visualization and Computer Graphics, IEEE computer Society Digital Library, doi:10.1109/TVCG. (2009.9).
- [5] Swaminarayan, S., and Prasad, L. "Rapid automated polygonal image decomposition". In AIPR '06: Proceedings of the 35th Applied Imagery and Pattern Recognition Workshop, IEEE Computer Society, 2006, 28.
- [6] Xia, T., Liao, B., Yu, Y. "Patch-Based Image Vectorization with Automatic Curvilinear Feature Alignment". ACM Transactions on Graphics, 28(5): Article 115, 10 pages (2009).
- [7] Lai, Y.-K., Hu, S.-M., Martin, R. "Automatic and Topology-Preserving Gradient Mesh Generation for Image Vectorization". ACM Transactions on Graphics, 28(3): Article 85 (2009).
- [8] Bingfeng Zhou and Xifeng Fang. "Improving Mid-Tone Quality of Variable-Coefficient Error Diffusion Using Threshold Modulation". Proc. ACM SIGGRAPH 2003, ACM Transactions on Graphics, 22(3), pp. 437-444 (2003).
- [9] Sun, J. Liang, L., Wen, F., and Shum, H.-Y. "Image vectorization using optimized gradient meshes". ACM Transactions on Graphics, 26(3), Article 11 (2007).
- [10] Orzan, A., Bousseau, A., Winnemoller, H., Barla, P., Thollot, J., and Salesin, D. "Diffusion curves: a vector representation for smooth-shaded images". ACM Transactions on Graphics, 27(3), Article 92 (2008).
- [11] Floyd, R. W., and Steinberg, L. "An Adaptive Algorithm for Spatial Greyscale". Proceedings of the Society for Information Display 17(2), 75–77 (1976).
- [12] Ostromoukhov, V. "A simple and efficient error-diffusion algorithm". In Proceedings of SIGGRPAH, New York, E. Fiume, Ed., Computer Graphics Proceedings, Annual Conference Series, ACM, 567-572 (2001).
- [13] Ulichney, R. A. "Dithering with blue noise". Proceedings of the IEEE 76 (Jan.), 56–79 (1988).
- [14] Haoli Fan, Jie Feng and Bingfeng Zhou. "Multi-Class Error-diffusion with Blue-Noise Property", Technical report, ICST, Peking University (2012).