Quad-tree construction for ordered dithering mask

Radu-Lucian Lupşa[1]

ABSTRACT Rendering grayscale images on printers requires transforming images form grayscale to bi-level black and white. Ordered dithering is the fastest algorithm for this transformation. It has been made popular with the blue noise mask algorithm. We propose here a novel algorithm for producing the mask for ordered dithering.

1 Introduction

Rendering a grayscale image on a printing device requires that the image be transformed from grayscale to black and white. Therefore, the gray levels must be transformed into a larger or smaller density of black points on the white background (paper). The process is commonly known as dithering or halftoning.

In spite of the increased computing speed, the quest for good halftoning algorithms is still open, as ever larger images are required to be printed at ever higher resolutions.

The main approaches to the problem are:

1. Fixed pattern for each gray level. That means that each pixel is converted into a pre-defined pattern of 8x8 or 16x16 pixels. The advantage is simplicity, the disadvantage is that the resolution is decreased 8 or 16 times, and the pattern size is a fixed compromise between the resolution loss and the number of available gray levels (65 or 257).

2. Error diffusion. Basically, the ideea is to compute, for a given region of the image, the sum of original image pixel values, and the sum of the halftoned picture value, and to set the next pixel so that that

[1]Babeş-Bolyai University of Cluj-Napoca

difference is kept between 0 and 1. The main drawback is that at least some gray levels tend to lead to visually-disturbing low-frequency patterns (artifacts) in the halftoned image. There is a huge number of algorithms in this class, with different approaches to avoiding the artifacts. Approaches include:

- choosing an elaborate path through the image
- distributing part of the difference to the neighbour pixels, and choosing either pseudo-random diffusion coefficients, or well crafted in order to avoid (read minimize) the artifacts

3. Ordered dithering. The dithering algorithm is simple, there is a mask (either the size of the image, or tiled to that size), and each pixel in the image is compared to the corresponding mask value. If the pixel value is greater, the corresponding pixel in the output image is set to 1, otherwise to 0. The halftoning itself is very simple and very fast; on the other hand, generating the mask is a difficult problem, that had been unsolved for ten. Now, the most popular solution is the so-called *blue noise mask*, which consist in a direct filter of the Fourier transform with a blue noise filter. Unfortunately, the method is patented. We propose here an alternate algorithm.

2 The algorithm

The algorithm constructs a mask for order-dithering, using an ideea taken from the quad-tree error-diffusion of [4]. Given any gray level, the points where that gray level is larger than the mask values of those elements will form a dithering pattern as the quad tree dithering algorithm would have constructed.

The mask consist of a squre, the sides length n being a power of 2. The mask is divided into four sub-squares, recursively divided again until the size of one pixel. Those sub-squares form a quad-tree.

Now, the mask construction goes as follows: For each value from 0 to $1 - \frac{1}{n^2}$ in increments of $\frac{1}{n^2}$, choose a path (see below) from the top of the quad-tree towards a leaf. The value is then written to that leaf. For each node the path pass through, a random permutation of the 4 sons is choosen, and each time the path reaches that node the next son is choosen as the next node for that path. Each time a cycle of four path is completed, a new random permutation is generated for that node.

Let's take as an example a mask of size 4x4 (16 pixels). The quad tree has 3 levels. The first value is 0. For the root node, suppose the random permu-

Figure 1: Original grayscale image

tation is NE (north-east), SE, NW, SW. Then, the north-eastern subsquare is choosen, and a new permutation is choosen for that node. Suppose it is SW, NW, NE, SE this time. Then the value is written to the south-western pixel. For the next value, 1/16, the path starts again from the root, this time towards the second (south-eastern) node, and so on. After the fourth value, the matrix may look like

?	?	?	?
2/16	?	0	?
3/16	?	1/16	?
?	?	?	?

where question marks denote not yet initialized elements.

When the next value is about to be entered, a new permutation is generated for the root node, but not for its sons.

3 Conclusion and future work

The algorithm creates a mask for the ordered dithering. The algorithm is very fast. The results are good: a good spreadout of the mask values is guaranteed by the permutations for each level. For a given gray level, it will result that, for each node, the number of white pixels of each sub-square is balanced, up to a difference of at most one pixel. The randomness ensures that no regular pattern is formed.

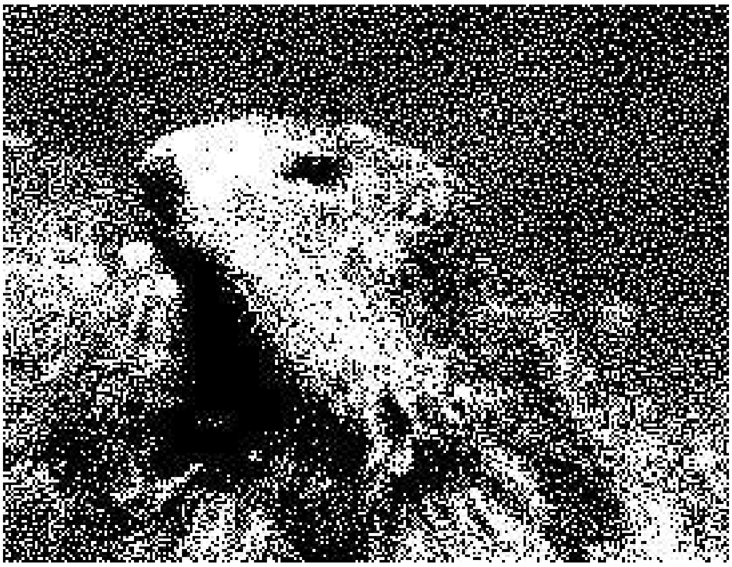

Figure 2: Halftoned image

There is however some room for improuvment, as the ratio between the white pixels and the total number of pixels for a square that does not correspond to a node of the quad-tree is no quaranteed to be equal to the gray level within one pixel error. A study of how to overcome that problem is under way.

References

[1] VICTOR OSTROMOUKHOV *A Simple and Efficient Error-Diffusion Algorithm* Computer Graphics Proceedings, Annual onference Series, 2001, p. 567–572

[2] T. MITSA, K. J. PARKER *Digital halftoning using a blue noise mask* Journal of the Optical Society of America, 1992

[3] R. W. FLOYD, L. STEINBERG *An adaptative algorithm for spatial grey scale* Proc. Soc. Inf. Display, 17:75–77, 1976

[4] RADU LUPSA, DANA LUPSA *A Color Dithering Algorithm* Proc. of ICAM3, publ. in Buletinul Stiintific al Universitatii Baia Mare, vol. XVIII, nr. 2, 2002