Pseudo-Vector Error Diffusion Using Cross Separation Threshold Imprints

Reiner Eschbach
Xerox Digital Imaging Technology Center
800 Phillips Rd. 128-27E, Webster, New York
eschbach@wrc.xerox.com

Error Diffusion is a common halftoning algorithm that converts a contone input image signal into a binary or multi-level output signal for printing and display. Different methods exist to expand error diffusion to color, ranging from a purely scalar approach, all the way to vector error diffusion. This paper describes an intermediate approach that maintains some of the positive aspects of vector error diffusion without some of its computational drawbacks.

Introduction:

Reproducing an image on an output device commonly encompasses the reduction of the image data from the original 8 or more bit, to a smaller set of reproducible output levels. For a large number of printers, the reproducible number of output levels is "2", describing a binary printer. It should be noted that real printers commonly have interactions between neighboring pixels so that the actual number of output levels for a "binary" printer might actually be quite larger than two.[1,2,3] One common method to reduce the number of levels from the original number to the number required for the output device is error diffusion,[4] In that method, an error feedback loop is used to compensate for all errors that are made during the quantization step. Figure 1 shows a 1-dimensional version of error diffusion.

The error diffusion concept can easily be generalized to more than the 1-dimensional case shown in Figure 1 and to more than two output levels*. Extending the algorithm to more than two output levels immediately poses the possibility that the output levels can describe a higher-dimensional space. Simple examples of such higher-dimensional spaces are complex numbers and color values. A full extension of error diffusion would then result in vector calculations for the output and for the error. A simplistic extension would be to use a scalar algorithm for each of the components, provided that the actual set of output levels includes all possible combinations of the scalar output levels.

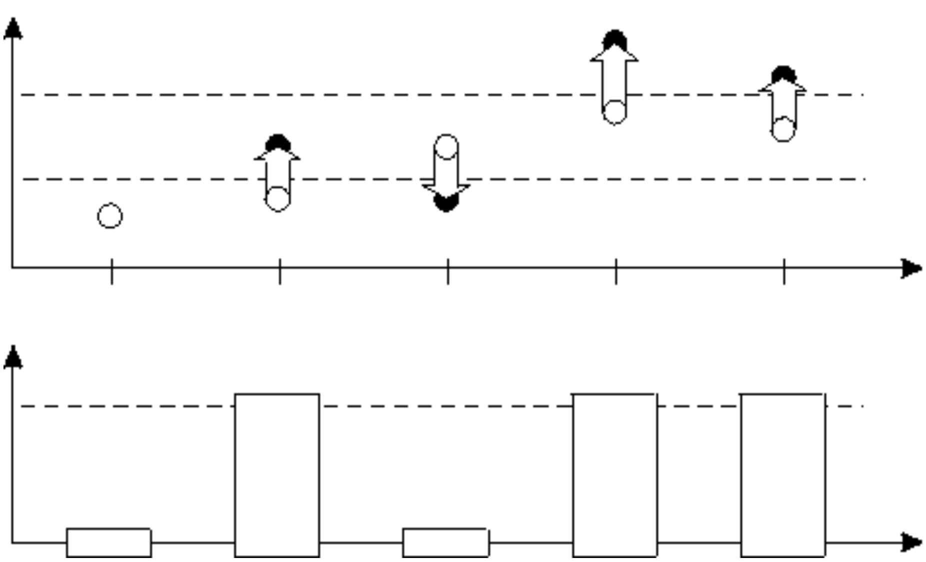

Figure 1. The continuous tone input data (gray circles top) is quantized to the binary output (gray rectangles bottom) generating a quantization error (white arrows). This error is used to create a modified input (black circles top) for subsequent pixels.

Vector Error Diffusion vs. Scalar Error Diffusion

It is obvious that the vectorial extension of error diffusion allows better control over the algorithm than the scalar approximation, however, there are some disadvantages in the vectorial approach that might warrant a scalar algorithm. One obvious advantage of the scalar approach is processing speed. Scalar processing can, dependent on the actual output device, offer some additional benefits.

Assume a theoretical print engine that deposits the "red", "green" and "blue" components of a color pixel simultaneously. In this case, changing from vector error diffusion to scalar error diffusion only reduces the computational complexity. Now assume a theoretical print engine that deposits the "red" components of an entire page, before it deposits the "green" and subsequent "blue" components. In this case, the entire image data has to be stored in order to process the image. For color images, this can easily amount to a memory requirement of larger than 20 or 30 MB. This memory requirement can severely impact the overall systems performance, giving a strong advantage to scalar processing.

The major disadvantage of the scalar processing is that the different components are processed independently and that interdependencies between the components can therefore not be easily controlled. Consider the simple example of a 50% cyan patch. This patch can be generated by using a "blue", "green", "blue", "green" sequence, by a "cyan", "white", "cyan", "white" sequence or any random or non-random pattern similar to the mentioned sequences. Obviously, the different sequences will show different visual noise, e.g.: a cyan-white transition has higher visual contrast than a blue-green transition, and different color behavior,[5] In real data, some image noise will cause a "random" switching between the different sequences, resulting in a system that is hard to calibrate and that potentially exhibits strong visual boundaries.

It is therefore desired to have a system that controls the correlation between the individual separations, while being computationally as independent as possible.

Pseudo-Vector Error Diffusion

Pseudo-vector error diffusion is a common name attached to methods that are predominantly scalar in nature, but that maintain some control over the correlation of the individual separations. In the methods described here, the correlation control is achieved using threshold imprints,[6]

Standard error diffusion uses a constant threshold, in the past, several modifications to this constant threshold have been used to improve the quality of the output.[7,8] A similar approach can be used to control the correlation between different separations. In this case, the binary output of one separation is used to guide the threshold imprint of the subsequent separations. If an "on" pixel in the first separation raises the threshold in subsequent separations, the "on" pixels in the different separations will tend to be at different locations; if an "on" pixel in the first separation lowers the threshold in the subsequent separations, the "on" pixels will preferably fall on the same locations.

A simple implementation of the above mentioned idea can be obtained by using

$$T_{i+1}(m,n) = \begin{cases} T_i(m,n) + \alpha & \text{if} \quad B_i(m,n) = \text{"off"} \\ T_i(m,n) + \beta & \text{if} \quad B_i(m,n) = \text{"on"} \end{cases}$$

where T_{i+1} is the threshold of separation $i+1$, and α and β are constants that control the correlation between the separations. Table 1 shows the results for using different α and β on a constant color input, where "Zero States" refers to "black" output, i.e.: none of the three separations r,g,b is set, "One State" refers to the output colors red, green, or blue, "Two States" refers to cyan, magenta and yellow, obtained by superposing two of the primary r,g,b colors, and "Three States" refers to white.

As can be seen, from Table 1, the correlation between the separations can strongly be influenced as expected, varying from a strong in-phase, to a strong out-of-phase correlation. It should be noted that the "random" correlation is not really random but strongly influenced by the correlation of the input data.

Table 1. Changing the constants α and β can alter the correlation of the binary output pixels from an out-of-phase correlation to an in-phase correlation.

	Correlation	Zero States	One State	Two States	Three States
$\alpha<0, \beta>0$	out-of-phase	77	323	0	0
$\alpha=0, \beta=0$	random	149	181	66	4
$\alpha>0, \beta<0$	in-phase	261	49	2	88

Conclusions

A pseudo vector error diffusion algorithm can be implemented using a threshold imprint to control the correlation between different color separations. For this purpose, the output of one separation is used to modify the threshold of subsequent separations, directly influencing the probability with which pixels are set to "on" or "off" in the subsequent separations. This method maintains some advantageous aspects of full vector processing, but at reduced computational costs.

References

* The 1976 error diffusion article showed a 2-dimensional version and already gave the extension to more than 2 levels.

1. Shen-ge Wang, "Two-by-two Centering Printer Model and Yule-Nielsen Equation", *Proc. NIP14: International Conference on Digital Printing Technologies*, 1998.
2. T.N.Pappas, "Model-based halftoning", Human Vision, Visual Processing and Digital Display II, *SPIE* **1453**, 1991.
3. C.J.Rosenberg, "Measurement-based evaluation of a printer dot model for halftone algorithm tone correction", *J. of Electronic Imaging,* **2/3** 1993.
4. R. W. Floyd and L. Steinberg, "An adaptive algorithm for spatial grey-scale," *Proc. Soc. Inf. Disp*. **17,** 75-77 (1976).
5. R. Victor Klassen, Reiner Eschbach and Krishna Bharat, "Vector Error Diffusion in a Distorted Colour Space", *IS&T Annual Meeting 1994,*
6. US-A 5,565,994, "Multiple Separation Error Diffusion with Cross Separation Control for Color Images", R. Eschbach
7. R. Eschbach and K. T. Knox, "Error-diffusion algorithm with edge enhancement," *J. Opt. Soc. Am. A* **8,** 1844-1850(1991).
8. T. Knox and R. Eschbach , "Threshold modulation in error diffusion", *Journal of Electronic Imaging*, **2,** (1993) 185-192