20.4: Displaced Filtering for Patterned Displays

Claude Betrisey, James F. Blinn, Bodin Dresevic, Bill Hill, Greg Hitchcock, Bert Keely, Don P. Mitchell, John C. Platt, Turner Whitted

Microsoft Corporation, 1 Microsoft Way, Redmond, WA 98052, USA

Abstract

This paper describes the filtering used in Microsoft ClearType. ClearType is a software system than enhances the resolution and readability of fonts on displays that contain a repeating pattern of addressable colored sub-pixels. The filtering in ClearType is based on a perceptual model of human vision. The perceptual model leads to an optimization technique for finding the best output values. The results of the optimization can be approximated by pre-filtering each color channel of an input image and then sampling each filtered color image at the spatial locations of the same colored sub-pixels in the display. We refer to this filtering followed by displaced sampling as RGB decimation. RGB decimation eliminates the phase error caused by standard anti-aliasing. A further approximation of the optimal filter yields RGB decimation with displaced box filters. Fourier analysis demonstrates that both the optimization technique and the displaced box filter suppress frequencies that contribute most to color fringing in unfiltered displaced sampling.

1. Introduction

This paper describes the filtering used in Microsoft's ClearType™ system. ClearType enhances the resolution and readability of fonts on displays that contain a repeating pattern of addressable colored sub-pixels. The filtering in ClearType is based on a perceptual model of human vision that leads to an optimization technique for finding the best output values. We approximate the results of the optimization technique by first pre-filtering each color channel of a multi-color input image. Each resulting filtered color image is then sampled at the spatial positions of the same colored sub-pixels in the display. We refer to this form of sampling as *displaced sampling*. Displaced sampling eliminates the phase error caused by standard anti-aliasing, hence prevents blurring. Fourier analysis illustrates how pre-filtering followed by displaced sampling can suppress color errors.

1.1 Previous Work

There have been numerous efforts to characterize the effect of display patterns on image quality. User tests have consistently indicated a preference for striped patterns over alternatives for text and graphics. In some cases, researchers have described image quality in terms of an error signal due to display patterns [5], or in terms of spectral components of a visual model [6]. Hara, et al. [2] analyze the artifacts generated by displaced sampling for certain display patterns, but does not consider the use of pre-filtering to minimize these artifacts.

In the related field of halftoning, various researchers [4][7][8] have suggested using psychophysical models to minimize the perceived error introduced by halftoning. Feigenblatt [1] suggests using displaced sampling followed by error diffusion to generate a halftoned image.

By and large, the spatial position of sub-pixels has been ignored in the generation of images of text and non-halftoned graphics for color displays. Descriptions of anti-aliasing techniques all assume that the light for each pixel is emitted from a single spot on the screen [9]. Even when other physical properties of the display device are accommodated [3], the spatial position of sub-pixels is ignored.

2. Optimal Filtering

In this paper, we consider one common variety of patterned display: an LCD with repeating pattern of vertical stripes of red, green, and blue. The display consists of a repeating pattern of singly colored sub-pixels, which can be grouped for convenience into full-colored pixels. For the striped display, the potential horizontal luminance resolution is the sub-pixel resolution: three times the horizontal pixel resolution. However, treating each sub-pixel purely as a luminance source, while ignoring the color of the sub-pixel, creates large color fringing errors. Thus, the full resolution potential cannot be realized.

The trade-off between luminance resolution and color error can be best balanced by the use of a perceptual error metric, which estimates the perceived visual effect of luminance and color errors. As the starting point for ClearType filtering, Platt [9] introduces such a perceptual error metric. This perceptual error metric is then used to derive an optimal filter for mapping a scanline of full-color pixels, G_{kd}, into a scanline of single-color sub-pixels, A_k, where k indexes the individual sub-pixels and d indexes the color channels of the input image. The error between the single-color sub-pixel outputs and the full-color input image is broken into three perceptually relevant opponent color channels [11]: black/white, red/green, and blue/yellow. The error can thus be expressed as

$$E_{ck} = M_{ck} A_k - \sum_{d=0}^{2} C_{cd} G_{kd},$$

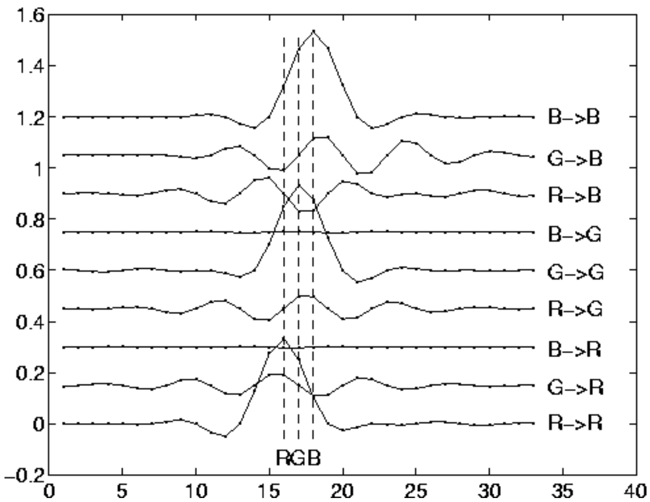
where C_{cd} and M_{ck} are matrices that transform A_k and G_{kd} into the opponent color space. The matrix M_{ck} contains information about the spatial pattern of the display. The error is then transformed into a Fourier basis:

$$\hat{E}_{cn} = \sum_{k=0}^{N-1} E_{ck} \phi_{kn}.$$

where ϕ_{kn} are the Fourier transformation functions. The optimal filter minimizes a weighted norm of the transformed error:

$$\sum_{c=0}^{2} \sum_{n=0}^{N-1} W_{cn} \hat{E}_{cn} \hat{E}^{*}_{cn},$$

where each frequency is weighted by W_{cn}, the frequency sensitivities of the human visual system to a grating with an opponent color pair [11]. Because this weighted error is a quadratic function of A_k and the derivative of a quadratic function



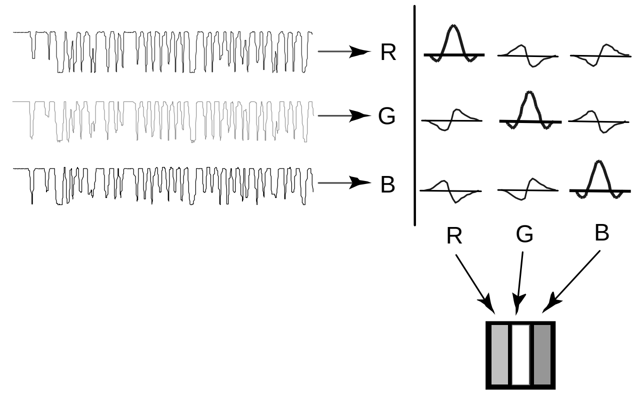


Figure 1. The optimal filtering consists of nine filters applied to three color channels (left). The filter coefficients (rights) plotted as a function of position along the horizontal axis. Dashed linear indicate the position of the R, G, and B sub-pixels. For the sake of clarity, the nine filters are offset vertically.

is linear, finding the sub-pixel outputs that minimize this weighted error is equivalent to solving a linear system for every scanline. The linear system depends only on the pattern of colors in the scanline and the human visual system frequency weightings. More details are available in [9].

Interestingly, solving the linear system is mathematically equivalent to applying nine filters to the scanline. The coefficients for these nine filters are shown in Figure 1. The x-axis of Figure 1-left is measured in sub-pixel units. For a particular color of sub-pixel, (e.g., R), the three filters (R→R, G→R, and B→R) apply to the R, G, and B input scanlines respectively. The output is the sum of these three filters and the sum is applied to the R sub-pixel whose location corresponds to the dashed line over the R in Figure 1-left. The application of these filters is shown schematically in Figure 1-right.

3. RGB Decimation

Based on the nine-filter method in [9], we seek a further simplification in order to produce a real-time implementation. This simplification also sheds light on why ClearType filtering works well. Examining Figure 1 shows that the filters that connect different colors (e.g., B→R, or G→B) have low amplitude. Although they are useful for sharpening primary-color fonts, we will speed up the filter by ignoring them. Further examination of Figure 1 shows that each of the remaining three filters (R→R, G→G, B→B) is centered underneath the corresponding location of the sub-pixel. These three filters are nearly identical low-pass filters that will be further analyzed below.

We therefore introduce a filtering technique called *RGB decimation*. RGB decimation starts with a multi-color input image and pre-filters each channel with a low-pass filter. The output of each of these filters undergoes displaced sampling, i.e. the samples are taken at spatial locations that correspond to the locations of the corresponding colored sub-pixels. For a real-time implementation, we can choose a very simple filter that approximates the low-pass same-color filters in Figure 1. In practice, we use one-pixel-wide box filters for our RGB decimation implementation, as shown in Figure 2.

Empirically, we find that box filter RGB decimation yields fonts that are almost as sharp as the optimal filters, but with substantial speed gains.

The effectiveness of RGB decimation also illustrates why standard anti-aliasing is not ideal for patterned displays. If a system uses standard anti-aliasing without displaced sampling on a striped display, the red and blue components will be computed as if they will be displayed at the same place as the green component. However, the red component will be shifted a third of a pixel to the left of where it should be, while the blue component will be shifted a third of a pixel to the right. If the input image has more than one primary color and conventional anti-aliasing is employed, the shifting of these primaries will lead to blurring. RGB decimation avoids the shifting, hence will form sharper fonts.

In other words, standard anti-aliasing causes a phase error when used with a patterned display. RGB decimation eliminates this phase error, which increases luminance resolution at the cost of some color error.

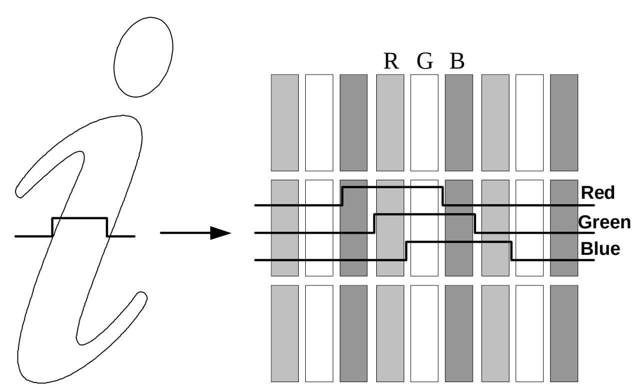


Figure 2. With RGB decimation using box filters, each filter is centered on the output sub-pixel and spans a complete pixel. Shown is a bi-level monochrome signal to be convolved with the three filters.

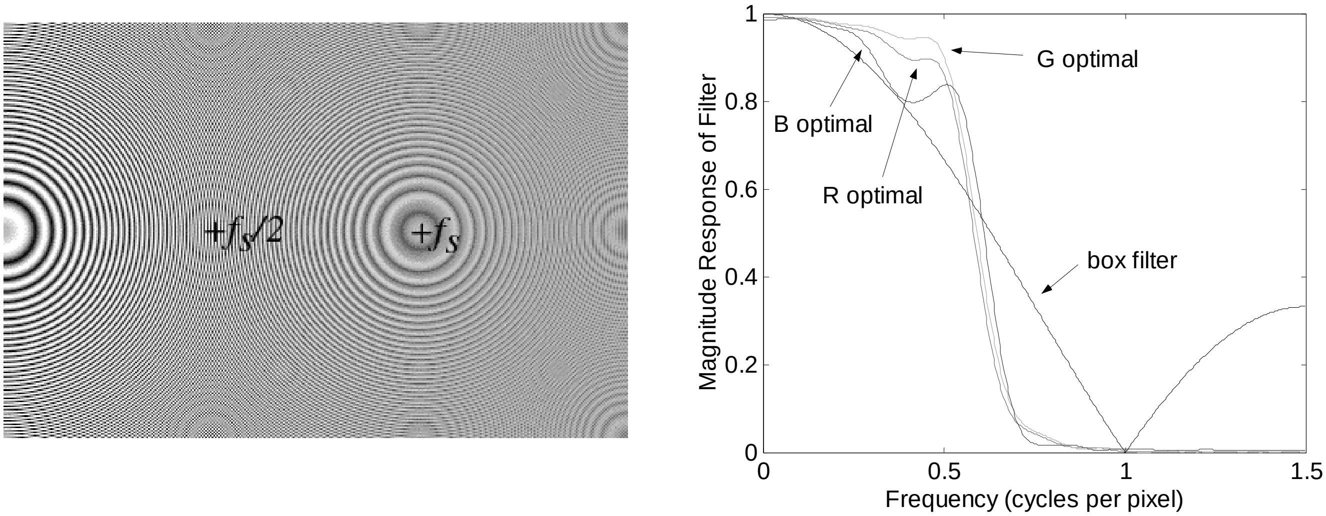


Figure 3. Black/white zone plate for displaced sampling with no pre-filtering (left). A color bull's-eye occurs at one cycle per pixel (f_s). The transfer response for various filters applied to a luminance signal (right).

4. Fourier Analysis

If the RGB decimation pre-filter is chosen to be near the optimal filter, it will produce minimal color errors. We can find a more general set of RGB decimation pre-filters by Fourier analysis. For brevity, we analyze the effect of displaced sampling on a striped display through the use of a zone plate.

The zone plate is a radially symmetric chirped black and white sinusoid. Displaced sampling of this zone plate, without pre-filtering, is shown on the left in Figure 3. DC is at the left hand side of figure 3-left: the horizontal frequency increases towards the right hand side of the zone plate, while the vertical frequency increases away from the vertical center of the zone plate. The cross labeled f_s occurs at the horizontal pixel sampling frequency.

There is a color bull's-eye in Figure 3-left at the horizontal pixel sampling frequency (twice the pixel Nyquist rate). This bull's-eye is the result of color aliasing: luminance frequencies at one cycle per pixel are aliased down to DC color components. However, because ClearType pre-filters before displaced sampling, we have an opportunity to suppress the color aliasing. The right side of Figure 3 shows the magnitude of the Fourier transform of the output for the optimal filter and for a box filter applied to a pure luminance signal.

Figure 3-right illustrates that both the optimal filter and the box filter will stop luminance signals near one cycle per pixel, so that the color bull's-eye is eliminated in the zone plate. Therefore, both RGB decimation and optimal filtering can be viewed as color anti-aliasing for patterned displays. Choosing the pre-filter to stop frequencies near one cycle per pixel minimizes color errors.

5. Implementation Details and Discussion

After the optimal filtering or generic RGB decimation is applied, there may be output values below zero or above one. For values below zero, we clip at zero. If any sub-pixel value is above one, we divide all three sub-pixel values from that pixel by the same value to maintain color correctness. A feature of box RGB decimation is that the filter never produces values below zero or above one.

It is also important to ensure that the output of the LCD truly matches the computed output of the filters. The outputs of the filters are values that are linear in the true brightness of the display. Therefore, the inverse of the non-linear response of the display must be applied to the filter outputs in order to produce the desired brightness. Failure to apply the inverse response may cause higher harmonics of the signal to produce frequencies near one cycle per pixel, which will cause excessive color errors.

In order to minimize aliasing caused by the displaced sampling, we use original images that are sampled at least six times horizontally per pixel. Both optimal filtering and RGB decimation can operate on these high-resolution input images.

Finally, these filtering techniques are only applicable to displays with individually addressable sub-pixels. Color CRTs or analog-input LCDs do not benefit to the same extent.

6. Results and Conclusions

We have applied box RGB decimation to a variety of fonts, with numerous foreground and background colors, with results that cannot be shown on the printed page. (Printing is a subtractive process that causes a color image of an LCD to appear too dark. Furthermore, we cannot control the non-linearity of the color reproduction.) Therefore, we illustrate our results in black-and-white. In Figure 4, the intensity of each sub-pixel is shown in a rectangle. A black rectangle is a dark sub-pixel, while a white (unseen) rectangle is a sub-pixel set to maximum brightness. Intermediate intensities are shown as gray rectangles. The top word in Figure 4 illustrates standard font rendering with no pre-filtering. There are many unpleasant jaggies if the font system fails to filter. The middle word in Figure 4 illustrates standard non-displaced box filter anti-aliasing. Standard anti-aliasing sets all three sub-pixels of a pixel to have the same intensity. Therefore, the resulting fonts are blocky and look smeared out. The bottom word in Figure 4 illustrates our box RGB decimation filter. The character shape is much smoother than either of the other alternatives. Also, because box RGB decimation stops frequencies near one cycle per pixel, there are minimal color fringes (which cannot be shown in Figure 4). In practice, fonts displayed with box RGB decimation are sharper and easier to read than non-displaced box filtered anti-aliasing.

Our analysis gives insight into the tradeoffs between sharpness and color fringing errors. It does not fully capture the subjective improvement in readability of ClearType text and graphics, nor does Figure 4. The readability of ClearType text is further enhanced through additional techniques, such as display-specific font hinting.

ClearType filtering is very general. Because it uses a full-color input image, it is applicable to text with arbitrary foreground and background colors. Indeed, ClearType filtering applies to any arbitrary RGB image, although the effects are most dramatic for text images. Also, both the optimal filtering and RGB decimation can easily be extended to two-dimensional patterns of colored sub-pixels, such as the Bayer pattern.

The impact of ClearType on the display industry will be significant. Without changing any manufacturing processes, or adding any cost to the display itself; displays will achieve higher effective resolution. Small fonts will become more readable. Microsoft has integrated ClearType into products such as the Microsoft Reader™ eBook software product and others.

7. Acknowledgements

We would like to thank Michael Duggan, Henrique Malvar, Chris Meek, and Geraldine Wade for help provided in the development of ClearType and in the preparation of this paper.

8. References

[1] Feigenblatt, R. I., "Full-color Imaging on Amplitude Color Mosaic Displays, *Proc. SPIE Digital Image Processing Applications*, **1075**, pp. 199-205, (1989).

[2] Hara, Z.-I., Terazaki, N., Shiramatsu, N., and Iwata, S. "Picture Quality of Different Pixel Arrangements for Large-Sized Matrix Displays," *Electronics and Communications in Japan*, Part 2, **77**, 7, pp 105-120, (1994). Translated from Denshi Joho Tshushin Gakkai Ronbunshi, **77-C-11**, 3, pp. 148-159, (1994).

[3] Kajiya, J. T., and Ullner, M., "Filtering High Quality Text for Display on Raster Scan Devices," *Computer Graphics* (Proceedings of SIGGRAPH 81) **15** 3, (1981).

[4] Kolpatzik, B. W., and Bouman, C. A., "Optimized Error Diffusion for High Quality Image Display," *J. Electronic Imaging*, **1**, 3, pp. 277-292, (1992).

[5] Krantz, J. H. and Silverstein, L. D., "Color Matrix Display Image Quality: The Effects of Luminance and Spatial Sampling," *SID Symp. Digest*, pp. 29-32, (1990).

[6] Martin, R. A., Ahumada, A. J. Jr., and Larimer, J. O., "Color Matrix Display Simulation based upon Luminance and Chromatic Contrast Sensitivity of Early Vision," *Proc. 1992 SPIE Symp. Electronic Imaging Sci. & Tech.*, **1666**, pp. 336-342, (1992).

[7] Mulligan, J. B., and Ahumada, A. J. Jr., "Principled Halftoning based on Human Vision Models," in *Proc. 1992 SPIE Symp. Electronic Imaging Sci. & Tech.*, **1666**, pp. 109-121, (1992).

[8] Neuhoff, D. L., Pappas, T. N., and Seshadri, N., "One-dimensional Least-squares Model-based Halftoning," in *Proc. ICASSP*, **3**, pp. 189-192, (1992).

[9] Platt, J. C., "Optimal Filtering for Patterned Displays," Microsoft Research Technical Report MSR-TR-2000-10, (2000).

[10] Warnock, J., "The Display of Characters using Gray Level Sample Arrays," *Computer Graphics* (Proceedings of SIGGRAPH 80) **14,** 3, pp. 302-307, (1980).

[11] Zhang, X. M., Wandell, B. A., "A Spatial Extension of CIELAB for Digital Color Image Reproduction," in *SID Symp. Digest*, pp. 731-734, (1996).

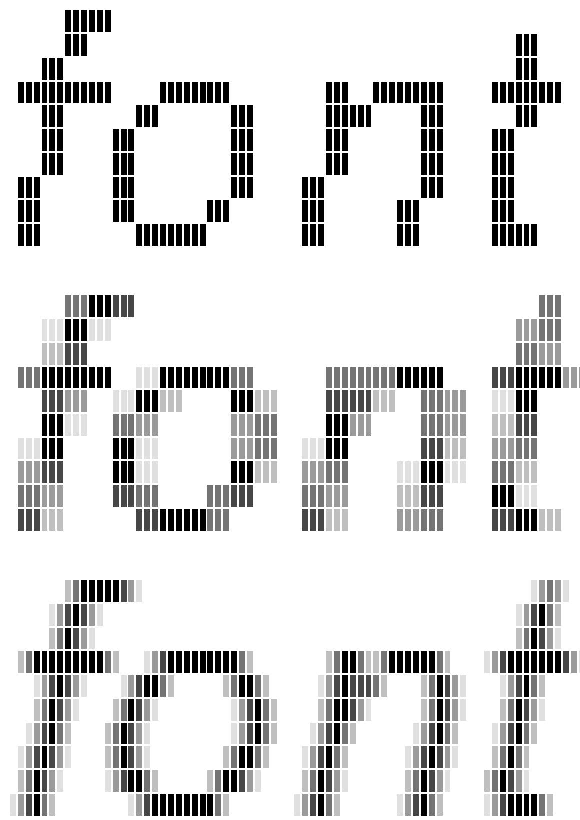


Figure 4. Examples of unfiltered (top), standard box filter anti-aliasing (middle), and box RGB decimation (bottom). All color sub-pixels are shown with identical luminance.

™ ClearType and Microsoft Reader are trademarks of Microsoft Corp.