Foveated Gaze-Contingent Displays for Peripheral LOD Management, 3D Visualization, and Stereo Imaging

ANDREW T. DUCHOWSKI
Clemson University
and
ARZU ÇÖLTEKIN
Helsinki University of Technology and University of Zurich

Advancements in graphics hardware have allowed development of hardware-accelerated imaging displays. This article reviews techniques for real-time simulation of arbitrary visual fields over still images and video. The goal is to provide the vision sciences and perceptual graphics communities techniques for the investigation of fundamental processes of visual perception. Classic gaze-contingent displays used for these purposes are reviewed and for the first time a pixel shader is introduced for display of a high-resolution window over peripherally degraded stimulus. The pixel shader advances current state-of-the-art by allowing real-time processing of still or streamed images, obviating the need for preprocessing or storage.

Categories and Subject Descriptors: I.3.6 [**Computer Graphics**]: Methodology and Techniques—*Interaction techniques*; I.3.7 [**Computer Graphics**]: Three-Dimensional Graphics and Realism—*Virtual reality*

General Terms: Performance, Human Factors

Additional Key Words and Phrases: Eye tracking, foveation, gaze-contingent displays, level-of-detail

ACM Reference Format:

Duchowski, A. T. and Çöltekin, A. 2007. Foveated gaze-contingent displays for peripheral LOD management, 3D visualization, and stereo imaging. ACM Trans. Multimedia Comput. Comm. Appl. 3, 4, Article 24 (December 2007), 18 pages. DOI = 10.1145/1314303.1314309 http://doi.acm.org/10.1145/1314303.1314309

1. INTRODUCTION

Gaze-contingent displays (GCDs) degrade the resolution of peripheral image regions, generally in a manner consistent with human vision, such as by degrading resolution matching human visual acuity. In such applications, an eye tracker is used to track the user's gaze so that a foveal region moves with the user's (overt) focus of attention. GCDs are important for both the study of human vision and eye movements, as well as for the reduction of computational effort or bandwidth in peripheral regions during image transmission, retrieval, or display.

This work was supported in part by an NSF Career award (no. 9984278).
Authors' addresses: A. Duchowski (contact author), Department of Computer Science Clemson University, 100 McAdams Hall, Clemson, SC 29634; email: duchowski@acm.org or andrewd@cs.clemson.edu; A. Çöltekin, GIVA/Geography Department, University of Zurich, Winterthurstr. 190, CH-8057, Switzerland.

Gaze-contingent displays have been invaluable for the purpose of studying visual perception. By removing information beyond perceptual limits, GCDs match the resolvability of human vision. In vision research, GCDs extend the classic "moving window" experimental paradigm [McConkie and Rayner 1975] originally developed in reading studies. Current research efforts extend this classic but one-dimensional form of display to two and three dimensions through spatial degradation of images and spatiotemporal degradation of video.

In applied work, GCDs help increase display speed through compression of peripheral image information that is not resolvable by the user. Applications include flight and driving simulators, virtual reality, infrared and indirect vision, remote piloting, robotics and automation, teleoperation and telemedicine, image transmission and retrieval, and video teleconferencing [Baudisch et al. 2003]. In at least one instance GCDs inspired a nongaze-contingent approach to the application of a visual eccentricity model of the human visual system's contrast sensivitiy function, or CSF, to videophone compression based on face tracking [Daly et al. 2001].

In virtual environments three-dimensional (3D) stereoscopic displays are used, but are usually not gaze contingent. Viewers often experience eye strain and several other problems, including diplopia and accommodation-convergence conflict (also referred to as vergence-focus conflict). Diplopia occurs when the disparity is greater than what the human brain can fuse and is present in most stereoscopic material. The accommodation-convergence conflict is also a problem related to binocular (stereoscopic) vision. Human eyes typically converge and accommodate at the same point. In 3D displays, however, while accommodation occurs at a point on the flat surface of the projection screen, convergence occurs in the front or behind the screen where the image forms. This conflict creates eye strain and can be disturbing [Mon-Williams and Wann 1998]. Applying the human visual system's (HVS) principles to scene visualization, that is, utilizing GCDs to simulate the depth of field, is one of the suggested (though not extensively tested) methods to help solve problems such as accommodation-convergence conflict and diplopia (see Ware [2004], Linde [2003], and Luebke et al. [2003]).

Model-based manipulation of level-of-detail (LOD), where geometric models are reduced in detail at distance as well as eccentricity, has also been explored in the last ten years. Notable examples include the work of Levoy and Whitaker [1990], Ohshima et al. [1996], Luebke and Erikson [1997], Murphy and Duchowski [2001], O'Sullivan et al. [2003], and Parkhurst and Niebur [2004]. Note that geometric LOD approaches fall outside the scope of this article, which only addresses image-based techniques.

While gaze-contingent techniques have evolved substantially over the last three decades, the image degradation methods employed by researchers have generally remained based in software. Perhaps due to the interdisciplinary makeup of this community, its members may not be fully aware of graphical approaches that are both simple and elegant, yet exceptionally robust in terms of their capabilities for control of spatial as well as chromatic peripheral degradation. The latter is of particular importance, since peripheral chromatic degradation has not yet been fully explored, and any potential bandwidth savings have not yet been investigated.

Gaze-contingent displays are susceptible to two major sources of display lag. First, any eye tracker used in the enterprise will exhibit a delay in delivering its real-time gaze coordinates. State-of-the-art eye-tracking devices such as those from Tobii Technology [Tobii Technology AB 2003], relying on video scenes of the face and eyes, generally incur a delay inversely proportional to their sampling rates. This delay can range from 5–20 ms or more, depending on the type of camera used. Second, given the gaze coordinates (considered instantaneous), GCDs must incur an additional delay by reconstructing an image degraded by applying a degradation function to image regions peripheral to the center of gaze. Common software-based image decomposition approaches (e.g., pyramidal schemes such as Laplacian or wavelet) addressing this delay depend on preprocessing a given image, storing numerous degraded

```
xhe quick brown xxxxxxxxxxxxxxxxxxxxxxxxxxx
        *
xxxxxxxxk brown fox jumxxxxxxxxxxxxxxxxxxxxx
              *
xxxxxxxxxxxxxxxxcat jumped ovexxxxxxxxxxxxxxx
                       *
xxxxxxxxxxxxxxxxxxxxxxxxxd over the laxxxxxx
                                  *
```

Fig. 1. Four temporal instances during gaze-contingent display change: Note the change of word `fox` to `cat`; the asterisks indicate fixation locations.

images in memory, and calling up the one with its foveally processed region closest to the instantaneous point of gaze.

While the delay incurred by eye tracking is one that cannot be obviated without effective use of eye movement prediction schemes, the image reconstruction bottleneck can for the most part be eliminated through application of hardware-assisted display techniques. The contribution of this article is a fragment shader for real-time gaze-contingent image reconstruction, performed on the graphics processing unit (GPU) of a modern PC graphics card. Although the shader only requires 5 lines of `GLSL` code, hardly warranting more than a page of documentation, the purpose of this work is to promote awareness of this elegant method in the eye-tracking community as well as the vision science and perceptual graphics communities, and to suggest the potential for spatiochromatic peripheral degradation research, which has not yet been attempted.

The article is composed in two major sections. First, a brief historical review of GCDs is given, focusing on the display techniques used previously. Second, the GPU-based image reconstruction program is developed through an evolutionary exposition of hardware-assisted methods, beginning with a multitextured approach and culminating with the simplest yet most robust (in terms of spatiochromatic rendering) `GLSL` program.

2. BACKGROUND

2.1 Early Research

Two experimental paradigms, the *moving window* and the *foveal mask*, were developed in the mid-1970's to explore eye movements and human reading strategies. Since then, these paradigms have been adapted to other domains, such as vision research and computer graphics. In the moving window paradigm, or *gaze-contingent display change* paradigm, developed by McConkie and Rayner [1975], a window is sized to include a number of characters (e.g., 14) to the left and right of a fixated word. For example, the sentence

```
the quick brown fox jumped over the lazy dog
```

is presented in four temporal instances during gaze-contingent viewing in Figure 1. The assumption with this technique is that when the window is as large as the region from which the reader can obtain information, there is no difference between reading in that situation and when there is no window. A related but reverse method, developed by Rayner and Bertera [1979] (see also Bertera and Rayner [2000]), places a foveal mask over a number of fixated characters (e.g., 7).

```
the qxxxxxxxown fox jumped over the lazy dog
        *
the quick brown cat jumpedxxxxxxxhe lazy dog
                              *
```

(a) (b) (c) (d)

Fig. 2. Visual field simulation of age-related macular degeneration (AMD): (a) and (b) are from National Institutes of Health [2003]; (c) shows our real-time rendering approximation with (d) an inverted Gaussian degradation mask with $\sigma = 200$.

This situation creates an artificial foveal scotoma and eye movement behavior quite similar to the eye movement behavior of patients with real scotomas [Rayner 1998].

2.2 Medical Vision and GCDs

By manipulating the display in real time, GCDs can provide compelling visualizations of visual field defects. GCDs can thus be used to educate students, physicians, and patients' family members about the perceptual and performance consequences of vision loss [Geisler and Perry 2002]. Figure 2(b) shows a visualization of age-related macular degeneration (AMD) (versus normal vision shown in Figure 2(a)) from a pamphlet issued by the National Institutes of Health [2003]. To render the image, National Eye Institute (NEI) doctors ask their patients with visual impairments what they see and try to get an in-depth description from them. Simulations are then created by computer staff and the doctors have them make changes until they feel that the information is correct [National Eye Institute 2004]. Although the rendering appears somewhat implausible, as the degenerative area appears to be inverted, the GCD technique described herein could easily generate such a depiction given an appropriate degradation function and fragment program. A simpler but perhaps more plausible resolution degradation function, shown in Figure 2(d), was used to simulate AMD in Figure 2(c).

2.3 Perceptually Lossless GCDs

A GCD is said to be perceptually lossless for a specified viewing distance and (instantaneous) gaze direction if the reconstructed display and the original appear identical to human observers when viewed from the specified distance (compare with the perceptually lossless image compression of Hahn and Mathews [1997]). Prior research of GCDs has mostly focused on the perceptual or performance effects of reducing the spatial frequency (i.e., cycles per degree or bits per pixel) of peripheral image regions (for two excellent surveys on GCDs, see Parkhurst and Niebur [2002] and Reingold et al. [2003]). Due to hardware limitations, a good deal of prior work relied on image preprocessing. For real-time display, preprocessed images would be recalled from memory on a "just-in-time" basis, that is, usually in relation to the location of the user's eye-tracked point of gaze.

Loschky and McConkie [2000] evaluated user performance with gaze-contingent multiresolution displays by using a set of 15 complex, monochromatic, photographic scenes as stimuli. Images were transformed such that a circular, high-resolution foveal region was surrounded by a degraded peripheral region through use of the discrete wavelet transform. These transformations were accomplished by first carrying out a wavelet decomposition of each image into 4 bands of increasingly high spatial frequencies, each an octave apart, using a 9/7 symmetric biorthogonal wavelet basis function. Images were reconstructed by dropping varying numbers of higher-frequency subbands. The GCD utilized a ViewGraphics 2Gbyte semiconductor image memory in which were stored 330 precomputed versions of each image, each having a high-resolution area at a different spatial location, corresponding to a

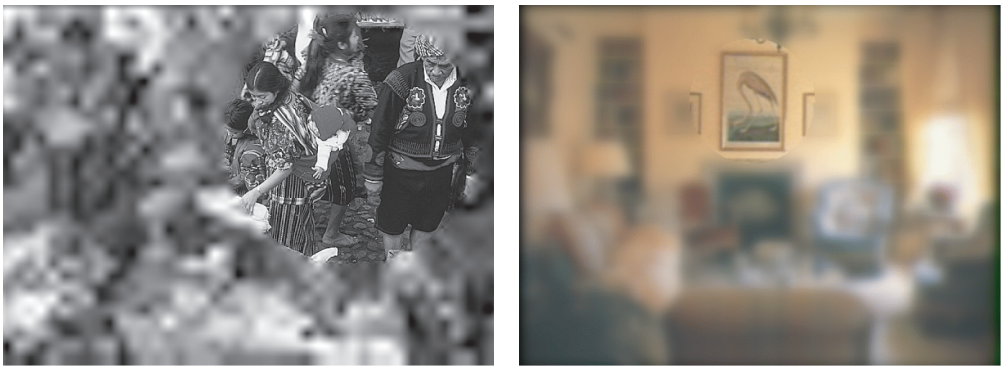


Fig. 3. Left: Loschky and McConkie's [2000] GCD © ACM. Right: Parkhurst et al.'s [2000] GCD © ACM.

15×22 array of screen locations, each less than 1° apart. Refreshing at 60 Hz, a new image was completely scanned onto the monitor within 17 ms of the time that an image change was requested (i.e., in response to a saccade).

Parkhurst et al. [2000] also evaluated variable-resolution displays by preprocessing images. Images were low-pass filtered by convolving with a Gaussian filter, yielding an attenuation of at least 70dB at and above the prescribed cut-off frequency.

For screen-based VR rendering, Watson et al.'s [1997] work is particularly relevant. Watson et al. studied the effects of level-of-detail (LOD) peripheral degradation on visual search performance. Both spatial and chrominance detail degradation effects were evaluated in head-mounted displays (HMDs), although in a head-contingent manner rather than a gaze-contingent one. To sustain acceptable frame rates, two polygons were texture mapped in real time to generate a high-resolution inset within a low-resolution display field. The authors suggested that visual spatial and chrominance complexity can be reduced by almost half without degrading search performance. More recently, Watson et al. [2004] used the same head-mounted display (but not head-tracked this time) to gain insights into peripheral LOD control beyond the perceptual threshold.

2.4 Space-Variant Imaging and Foveation

Most of the aforesaid approaches result in a biresolution GCD, with the demarcation between the foveal disk region's and peripheral resolution often purposefully made visible (thus without any inter-LOD blurring or averaging). Geisler and Perry [2002] proposed a method to generate completely arbitrary variable-resolution displays. Their display depends on pyramidal preprocessing of the images prior to display [Geisler and Perry 1998] (see Burt and Adelson [1983] for a detailed description of multiresolution pyramids with spatial filtering). Geisler and Perry's *space-variant imaging* software produces smooth, nearly artifact-free images at high frame rates, but is limited to manipulation of spatial resolution. The software implementing this method on Windows platforms is freely available online.[1] Geisler and Perry's work is particularly significant for its separation of resolution degradation from image source. In image compositing parlance, this *switchmatte* operation makes gaze-contingent rendering immediately obvious: Simply preserve high-resolution pixels only at matte locations with $\alpha = 1$ and map pixels at lower matte luminance levels to lower-resolution pixels (e.g., from a bank of preprocessed images).

[1] <http://fi.cvis.psy.utexas.edu>

Fig. 4. Geisler and Perry's [2002] GCD © ACM showing a scene from the movie *The Gladiator*. Here, the arbitrary visual field, inset bottom-right, simulates glaucoma over the original image, inset bottom-left. Original image © 2000 DreamWorks SKG and Universal Studios; gaze-contingent rendering and resolution map courtesy of Bill Geisler and Jeff Perry.

Since pyramidal reconstruction schemes draw pixel data from multiple levels of resolution, pyramidal image synthesis provides a smoothly degraded, convincing visualization of the human visual system, termed *foveation*. A particularly popular pyramidal approach relies on image decomposition via the discrete wavelet transform (DWT) with selective coefficient scaling and decimation prior to reconstruction [Chang et al. 2000; Duchowski 2000]. Provided that appropriate wavelet filters can be found, reconstruction exactly matches linear mipmapping.

Given an $N \times N$ image, assuming without loss of generality that N is a power of 2 with $n = \log_2 N$, the original image $f^n(x, y)$ is subsampled and smoothed into $n + 1$ subimages via

$$f^j\left(\left\lfloor \frac{x}{M} \right\rfloor, \left\lfloor \frac{y}{M} \right\rfloor\right) = \frac{1}{M^2} \sum_{k=0}^{M-1} \sum_{m=0}^{M-1} f^n(x + k, y + m), \quad 0 \le j \le n, \tag{1}$$

where M is a smoothing filter of size 2^{n-j}, and j is the resolution level. Eq. (1) generates projections of the original image onto $n + 1$ scaled subspaces equivalent to the subspaces generated by the scaling function of the DWT. The subspaces in this instance are scaled analogously to the DWT with resolution level $j = 0$ corresponding to the coarsest resolution level. Eq. (1) is a slightly different representation from the classical recursive pyramidal approach, since each subimage is subsampled directly from the original image f^n, not from the image at the next-finer resolution level f^{j+1}. The wavelet pyramid is formed by the union of the original image and the set of subsampled images. Reconstruction of the image at a given pixel location (x, y) depends on the desired resolution of the pixel. The desired resolution level is bandlimited to the number of decomposed resolution levels (typically the decomposition is dyadic in

nature) bounded by the two closest resolution subimages f^{j-1} and f^{j}. The final pixel value at location (x, y) is calculated as a linear combination of pixel intensities in the pyramid.

$$f(x, y) = (1-p)f^{j-1}\left(\left\lfloor \frac{x}{2^{n-(j-1)}} \right\rfloor, \left\lfloor \frac{y}{2^{n-(j-1)}} \right\rfloor\right) + (p)f^{j}\left(\left\lfloor \frac{x}{2^{n-j}} \right\rfloor, \left\lfloor \frac{y}{2^{n-j}} \right\rfloor\right) \tag{2}$$

Eq. (2) represents linear *intermap* interpolation.

To generate wavelet multiscale representations of a given image matching mipmap decomposition with the normalized box filter, the low-pass wavelet filter $\{h_k\}$ is set to $\{1/2, 1/2\}$. The detail filter is then the quadrature mirror of $\{h_k\}$, namely, $\{g_k\} = \{1/2, -1/2\}$. To guarantee perfect reconstruction, dual filters are required to satisfy biorthogonal conditions. In other words, $\langle h_k, \widetilde{g}_m \rangle = \langle g_k, \widetilde{h}_m \rangle = 0$ must hold simultaneously for $k, m \in \mathbf{Z}$, with $\langle \cdot, \cdot \rangle$ denoting the inner product. For filters of length 2, the following equations must hold.

$$h_0\widetilde{h}_0 + h_1\widetilde{h}_1 = 1, \quad g_0\widetilde{h}_0 + g_1\widetilde{h}_1 = 0 \tag{3}$$

Given $\{h_k\}$ and $\{g_k\}$, $\{\widetilde{h}_k\}$ is derived from (3), giving $\widetilde{h}_0 = \widetilde{h}_1 = 1$. Dual detail filter coefficients are derived similarly, producing $\widetilde{g}_0 = -\widetilde{g}_1 = 1$. These filters are unnormalized versions of the Haar filters; that is, they are semi-orthogonal Haar wavelets (or prewavelets). Normalized Haar filters will generate the same reconstruction as mipmapping at dyadic-resolution boundaries, but will lose luminance information between boundaries where linear interpolation is required. The benefit of the semi-orthogonal wavelets is that correct luminance values will be generated at any desired resolution level. Note that the coefficients of the low-pass filter $\{h_k\}$ match the aforementioned averaging box filter exactly. This can be easily verified by obtaining the tensor product of the scaling filter at any decomposition level. For example, at the first level ($j = n - 1$), the effective sampling filter is a 2×2 filter with cells equal to 1/4. At level $j = n - 2$, the filter is a 4×4 filter with cells equal to 1/16. Note that under the DWT, the finest resolution level (i.e., $j = n$, the original image) is not present in the pyramidal transformation.

To obtain interpolation results identical to mipmapping, an intuitive approach would be to maintain reconstructed scaled subimages produced by successive steps of the inverse DWT, and then to perform the interpolation step between the subimages. Assuming that equivalent subsampling filters guaranteeing perfect reconstruction are used, for example, orthogonal filters, this approach yields identical results to mipmapping, although it is memory intensive. What is perhaps not obvious is that identical interpolation results can be obtained by scaling wavelet coefficients prior to reconstruction. Wavelet coefficient scaling results in attenuation of the signal with respect to the average (low-pass) signal. Full decimation of the coefficients (scaling by 0) results in a lossy, subsampled reproduction of the original. Conversely, scaling wavelet coefficients by 1 preserves all detail information, producing lossless reconstruction. Selectively scaling the coefficients by a value in the range [0, 1], at appropriate levels of the wavelet pyramid, produces a variable-resolution image upon reconstruction. This approach is equivalent to mipmapping reconstruction with linear interpolation of pixel values.

In mipmapping, the value of interpolant p is determined by an arbitrary mapping function which specifies the desired resolution level l. The two closest pyramid resolution levels are then determined by rounding down and up to find subimage levels $j - 1$ and j. The interpolant value is obtained by the relation $p = l - \lfloor l \rfloor$. Note that the slope of the mapping function should match the resolution hierarchy of the pyramid, that is, if resolution decreases eccentrically from some reference point, the parameter l should also decrease eccentrically. If it does not, its value may be reversed by subtracting from the number of resolution levels, namely, $n - l$. To scale wavelet coefficients, p is set to 0, 1, or the interpolant

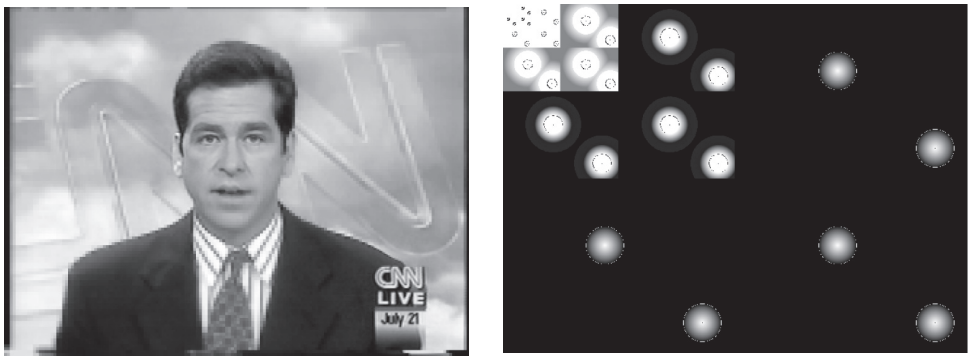


Fig. 5. Wavelet foveation.

value at particular subbands according to the following relation dependent on l.

$$p = \begin{cases} 1, & j \leq \lfloor l \rfloor \\ l - \lfloor l \rfloor, & j = \lceil l \rceil \\ 0, & j > \lceil l \rceil \end{cases} \tag{4}$$

For example, if at some pixel location $(x, y), l = 1.5$, then wavelet coefficients would be preserved (scaled by 1) at levels $j \leq 1$, scaled by .5 at level $j = 2$, and decimated (scaled by 0) at levels $j > 2$ at the appropriate pixel location in the subimages.

Given an n-length discrete 1D function f^j at resolution-level j, its DWT decomposition is given by the relations

$$f_\phi^{j-1}(x) = \sum_k h_k f_\phi^j(2x + k), \quad f_\psi^{j-1}(x) = \sum_k g_k f_\phi^j(2x + k), \quad k \in \mathbf{Z},$$

where $\{h_k\}, \{g_k\}$ are one-dimensional low- and high-pass filters, corresponding to the scaling and wavelet functions ϕ and ψ, respectively. Reconstruction with coefficient scaling is then written as

$$f_\phi^j(2x - t) = \sum_k \widetilde{h}_{t-2k} f_\phi^{j-1}(x - k) + p \sum_k \widetilde{g}_{t-2k} f_\psi^{j-1}(x - k), \quad t, k \in \mathbf{Z},$$

where $\{\widetilde{h}_k\}, \{\widetilde{g}_k\}$ are dual reconstruction filters, t is the filter length, and p is the resolution-level-dependent interpolant defined by Eq. (4). The aforesaid relations extend directly to two dimensions through 2D tensor product assembly of the 1D filters. It can be shown that wavelet coefficient scaling is equivalent to linear pixel interpolation under mipmapping [Duchowski 1998]. The proof is intuitive, since subimages in the mipmap pyramid correspond to the low-pass subimages recovered at each stage of the inverse DWT reconstruction. In fact, the low-pass subimages generated at each level of reconstruction are identical to the subsampled images used in mipmapping, provided both approaches use equivalent filters and the DWT is guaranteed to be lossless (e.g., orthogonal wavelets are used).

Duchowski [2000] provides an acuity-matching mapping modeling the human visual system derived from empirical MAR (minimum angle of resolution) data [Foster et al. 1989]. MAR data at the border of the projected foveal ROI (at 5° visual angle) is converted to expected maximum resolution in dots per inch (dpi). Expected resolutions at peripheral eccentricities are derived relative to this maximum. Depending on the viewing distance and resolution of the display device, relative resolvability values in dots per inch are then converted back to pixel units to give the diameters of resolution bands. Wavelet space coefficient degradation and a resultant image with 2 foveated ROIs are shown in Figure 5.

2.5 Stereoscopic Imaging

Recently, Çöltekin [2006] used a pyramid-based LOD management method for close-range stereo photogrammetric rendering. Stereoscopic media has been used for the purposes of amusement since the late 1800's. The first stereoscopic drawings are often attributed to Sir Charles Wheatstone in 1838, some years before the first stereo photographs—even though research regarding human stereoscopic vision and perception is much older; for example, Kepler's stereo projection theory goes as far back as 1611 (see Lipton [1982]).

Even though all GCDs are not 3D and all displays do not use stereoscopic principles, stereoscopy is an obvious and relatively mature technology to utilize for a number of purposes. Recently, mobile devices such as a mobile phone and a laptop computer were produced by Sharp, Inc., with optional autostereoscopic displays and successfully marketed.[2]

These developments and the potential for 3D television with a compelling sense of 3D presence have revived popular interest in stereoscopic displays. Stereoscopic displays may offer certain other advantages as well. These include potential for better relative depth judgment, spatial localization, camouflage breaking, surface material perception, and better judgment of surface curvature [Holliman 2005]. It is also thought that stereoscopic vision improves visual acuity when compared to monoscopic viewing (see Campbell and Green [1965] and Drascic and Milgam [1991]).

For stereoscopic display technology (e.g., head-mounted displays, panoramic displays, CAVEs), management of the visualization in a gaze-contingent manner has obvious bandwidth benefits, as the amount of data transmitted in a stereoscopic system is always double that of monoscopic ones. There are also indicated health benefits, to applying gaze-contingent thinking to stereoscopic displays. Research by Çöltekin [2006] and Linde [2003] are the two most recent works conducted on stereoscopic imaging and gaze-contingent rendering.

2.6 Limitations and Recent Developments

Böhme et al.'s [2006] GCD goes beyond spatial resolution degradation by degrading temporal as well as spatial content. Spatiotemporal degradation extends Geisler and Perry's [2002] pyramidal preprocessing approach into the temporal dimension.

Although recent GCD implementations have yielded new insights into perception, most of these approaches still rely on somewhat restrictive computational strategies, that is, either software-based pyramidal image reconstruction (as in spatiotemporal video degradation by Böhme et al.) or a texture-mapped rendering with a limited number of textures (as in Watson et al.'s dual viewport composition). The limitation of these techniques surfaces either in their limited speed or display characteristics. Böhme et al. claim 30 frames per-second performance but have previously reported an average system latency of 60 ms [Dorr et al. 2005]. Watson et al.'s dual viewports limited the display to two distinct regions. It is unlikely that this approach lends itself to the simulation of arbitrary visual fields. Indeed, Reddy [2001] noted that practically all perceptually-based work (up to that point) had used a small set of presimplified versions of an object from which to choose to render in a view-dependent manner. It appears that this is still the predominant approach, without apparent employment of the GPU for image synthesis. Reddy's *percept* visualization performs a per-pixel calculation of the pixel's spatial frequency based on angular velocity and eccentricity, while Çöltekin's *foveaglyph* builds a pyramid of scaled images.

Due to recent advancements in computer hardware, gaze-contingent imaging research has appeared where image processing operations are performed in real time, either by dedicated image processing

[2] <http://www.sharp3d.com>

hardware, or by more general-purpose graphics engines. In a recent example of dedicated hardware-accelerated eye-movement-controlled image coding, Bergström [2003] used a DCT-based image codec to achieve real-time image compression and display.

The GPU-based GCD, presented in the following section, evolved from several attempts at gaze-contingent image processing in graphics hardware. The resultant GLSL program allows simulation of arbitrary visual fields as inspired by Geisler and Perry [2002], but for the first time degraded chromatically as well as spatially. The approach has progressed from Nikolov et al.'s [2004] and Duchowski's [2004] independent introduction of multitexturing approaches.

3. EVOLUTION OF THE GPU-BASED GCD

Duchowski [2003] refers to gaze-contingent display processing as either *screen based* or *model based* where the former depends on image processing and the latter on processing graphics primitives (e.g., triangles). Texture-mapping (and shader programming) for gaze-contingent image display is a hybrid approach. Peripheral degradation of the image still relies on image processing, albeit the image is now considered a texture map. Rendering of the image relies on mapping the image onto a simple graphical object, in most cases a polygon (usually a screen-aligned quadrilateral) of the same dimension as the display window.

There are several tradeoffs between the texture-mapping and screen-based approaches, although both are now typically provided by graphics libraries such as OpenGL [Shreiner et al. 2006]. Advantages of the screen-based approach include the following.

—Image resolution is of minor importance. Provided that the viewing window is made to be the same size as the given image, the resultant display is generally shown at 1:1 pixel mapping, namely, the image is drawn to scale.

—Provided a graphics card that supports OpenGL's imaging subset in hardware is used, image processing operations can be performed quickly via hardware-accelerated convolution.

—Various blending operations are provided that enable simple image combinations to take place via an image's alpha channel.

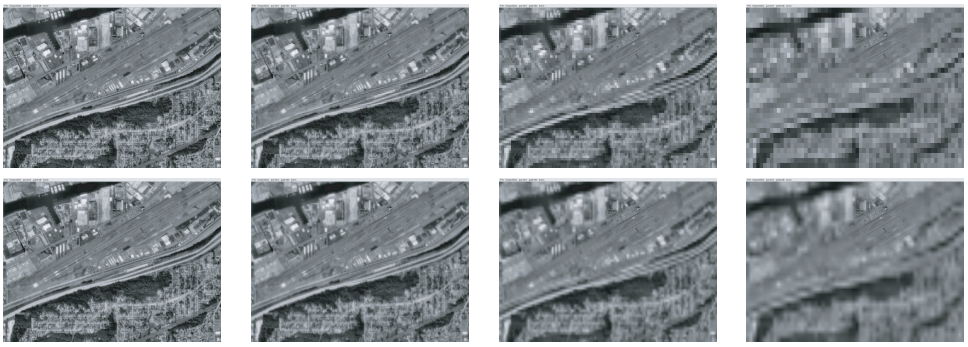
There are, however, disadvantages to the screen-based approach.

—Not all graphics cards support (or supported) the imaging subset in hardware. For example, the NVidia GeForce4 Ti 4600 card did not, but its more expensive cousin the NVidia Quadro4 (e.g., XGL 900) did. Lacking hardware support for the imaging subset, imaging operations such as convolution with the GeForce4 reverted to software implementation. This resulted in noticeable speed degradation.

—The most significant drawback of the screen-based approach for gaze-contingent display is that the required image-blending functions (for blending foveal and peripheral image portions) rely on the images' alpha channels. Thus, to provide a GCD, the image alpha channels would need to be translated in real time to match the foveal region, a potentially prohibitively expensive operation.

Texture-mapping, and in particular multitexturing and related fragment programming, solves the blending problem, since the alpha channel can be dissociated from either foveal or peripheral image and made into its own image. This is an important point since once so dissociated, the alpha mask can be manipulated independently. The manipulation that is most relevant to gaze-contingent display is translation of the foveal mask. Since mask translation is performed quickly in hardware, the result is real-time movement of the foveal region. There are, however, disadvantages to the texture-mapping approach.

Table I. Gaze-Contingent Rendering Results

Degradation level			
2^1	2^2	2^3	2^4
			

Here, the fovea is over the airplane in the upper-right quadrant. The top image row shows nearest-neighbor interpolation: a faster means of image reconstruction from multiple subimages at the cost of visible blocky artifacts. The bottom image row shows linearly interpolated reconstruction, leading to smoothly reconstructed images.

—In general, texture mapping is more complicated than simple image drawing, since it relies on the definition of the graphical object that is to be textured. Using a quadrilateral for this purpose is often the most simple and logical choice. Following geometry definition, textures need to be defined, bound, and loaded into memory. There are numerous options for doing so (this is somewhat of a blessing and a curse).

—Because texture-mapping generally relies on a geometric primitive and that primitive is subject to geometric transformations, the resultant display may or may not preserve the 1:1 pixel mapping between original image and final display. In contrast to the imaging subset, one usually needs to define the window size (as before), and also the polygon onto which the image will be texture-mapped. Care must be taken to properly display the polygon without inadvertently changing the polygon's size (which is quite easy to do, e.g., via viewing transformations).

—To display the texture-mapped primitive, texture coordinates are required. Care must be taken not to introduce inadvertent image scaling, shifting, etc., through improper coordinate use.

To summarize the distinction between screen-based and texture-based approaches, texture-mapping offers much greater flexibility in image display at the expense of additional complexity.

3.1 Mipmapping

For fully hardware-accelerated display as discussed here, GCDs can utilize in-hardware image degradation provided by built-in mipmapping functions. Mipmapping provides a method of prefiltering an image (texture) at multiple levels-of-detail [Williams 1983]. Mipmaps are dyadically (by powers of two) reduced versions of a high-resolution image. One can either create these images manually a priori, or have them created automatically by OpenGL.

Several filter options are available for generating coarsely subsampled or linearly interpolated images. Four texture minification options control combinations of inter- and intramap pixel interpolation. The effect of these commands generates a coarsely or smoothly degraded periphery for dyadic levels of degradation, exemplified in Table I for various levels of degradation. Real-time control of the texture environment and texture parameters allows on-the-fly switching of peripheral degradation.

Next, two recent approaches based on mipmapping are briefly reviewed for completeness and comparison to the subsequent newly introduced fragment programming technique. The former approach

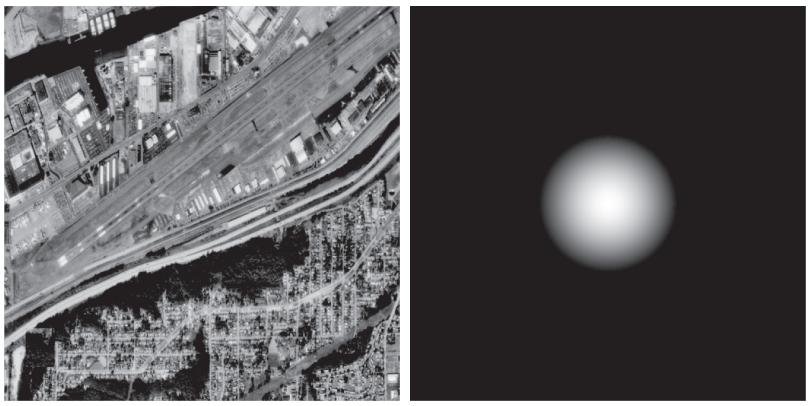


Fig. 6. Original runway image with Gaussian mask with $\sigma = 100$.

is suitable for implementations on third-generation graphics cards, while the latter requires fourth-generation cards (fourth-generation graphics cards are distinguished from earlier versions by their ability to compile and run so-called *fragment* as well as *vertex* shader programs).

3.2 Multitexturing

Real-time rendering of a biresolution GCD relies on two images. The first requirement is the source image for generating a high-resolution inset as well as a low-resolution background. The low-resolution background image is generated by dyadically degrading in hardware the source image via OpenGL's mipmapping facilities. Alternately, the source image (or another image altogether) may be preprocessed in some other way and can be substituted for the background image. The second required image is an arbitrary visual mask whose shape forms the foveal window. For example, an aerial image of a runway is shown in Figure 6, along with a Gaussian mask image ($\sigma = 100$). Both images are used in the following exemplar development of a GCD.

Using special-effects compositing terminology, the mask image simply constitutes the matte image which serves as the alpha mask for blending of the foreground (high-resolution) and background (low-resolution) images. The matte image is typically a normalized grey-scale image, where pixel values of 1 represent portions of the high-resolution image that show through while values of 0 are masked and therefore replaced by the corresponding background image pixels. Of course, any grey-scale image can be used instead to simulate an arbitrary visual field. Simply inverting a Gaussian 1-center, 0-surround map, for example, results in the "moving mask" paradigm used in perceptual vision research [Bertera and Rayner 2000] and can also simulate a form of glaucoma or AMD (as done for Figure 2(d) with larger σ).

To obtain a composited rendering of a foveal high-resolution window atop a low-resolution background, three textures are created for a quadrilateral. The first texture, assigned to texture-unit 0, or TU0, is the image mask. The second texture is the given image, which is assigned as the foreground image at texture-unit 1, or TU1. The third texture is the original image used for the foreground, also mipmapped, with the exception of the use of different LOD. It is the coarser LOD that generates the degraded background in the GCD.

During display, the mask texture at TU0 is translated to the real-time coordinates of the foveal position. The process is shown diagrammatically in Figure 7, with the callout showing the change in resolution between foveal and background regions. For printing considerations, a grey-scale image is

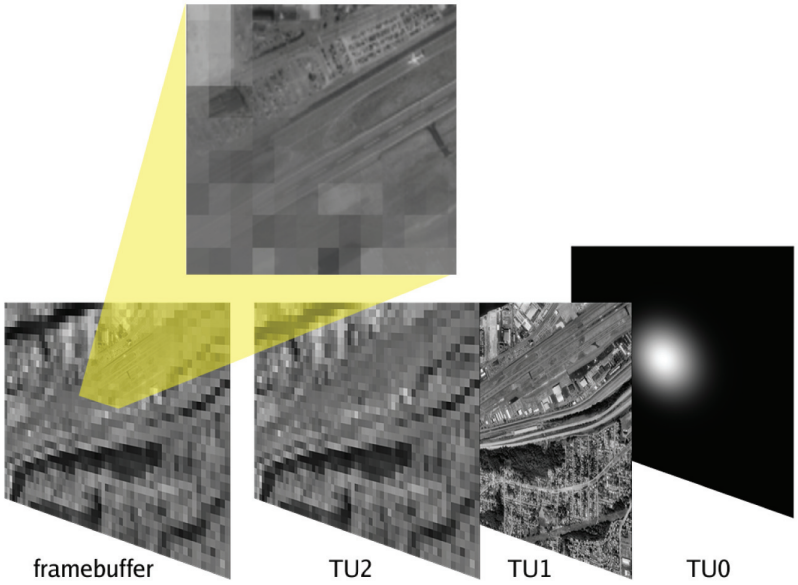


Fig. 7. Multitexture blending graphics pipeline.

used as the example stimulus although the texture-mapping methodology applies equally well to 24-bit (or 32-bit) color images.

An alternative approach, but also based on multitexturing, involves using two quadrilaterals instead of three, as shown by Nikolov et al. [2004], who use a similar approach to the preceding and apply their GCD to numerous applications, including gaze-contingent multiresolution displays, gaze-contingent multimodality displays (e.g., graphical maps overlayed on aerial photographs), and gaze-contingent image analysis.

3.3 Fragment Programming

The three-texture approach described previously leads to a biresolution display. For a more accurate representation of human visual acuity, multiple levels-of-detail are needed in the periphery, resulting in anisotropic peripheral degradation, otherwise known as a multiresolution gaze-contingent display (MRGCD). To provide multiple levels of resolution in the periphery, the aforementioned multitexturing approach would require the use of multiple texture units. What is required is schematically shown in Figure 8. At any given pixel concentrically related to the foveal position, a lookup is needed to a pixel at a specific level of resolution. Fragment programs provide just this type of flexibility by providing control of mipmap LOD bias at each fragment (pixel). The resulting sample is mapped to RGBA and written to the resulting vector. Unlike multitexturing, this rather elegant approach does not require explicit blending. Instead, the appropriate mipmap-level bias is obtained directly at each fragment. Note that if the degradation map is allowed to change dynamically, fragment programming allows dynamic visual field representation, such as, allowing multiple "regions of interest" (ROIs) which could be used for preattentive display purposes [Duchowski and McCormick 1995].

Source code for implementation in `GLSL` is given in Listing 1. A simple `GLUT` example is available on the web at <http://andrewd.ces.clemson.edu/gcd/>.

The GPU-based GCD code has been tested via both mouse- and eye-controlled foveal windows and runs well above hardware display rates (60 frames per second (fps); note that display updates as late

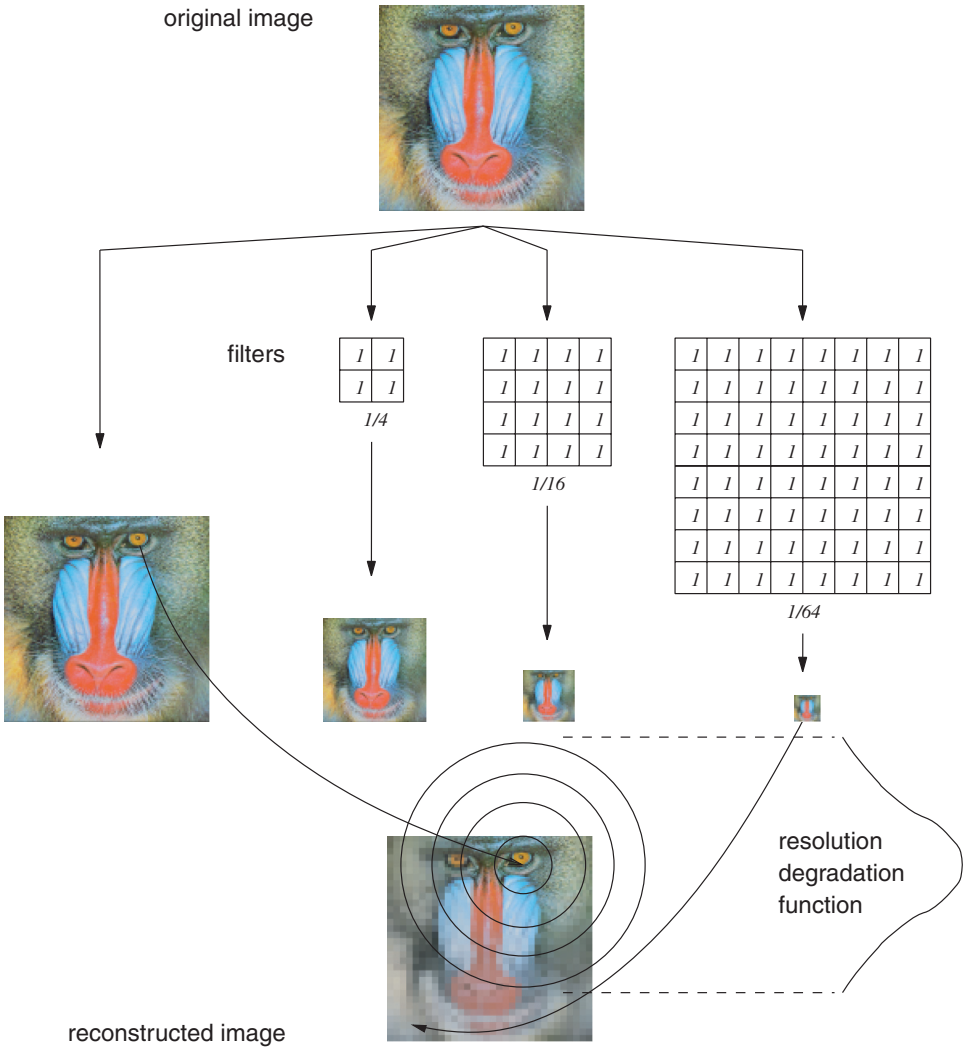


Fig. 8. Illustration of per-fragment mipmap LOD bias selection (filter sizes are suggestive of mipmap levels but are not drawn to scale).

as 60 ms after eye-movement completion do not significantly increase the detectability of image blur and/or motion transients due to the update [Loschky and Wolverton 2007]). The code has also been extended to display video streams by interfacing with a video loading library. Due to hardware-assisted subsampling of a given image, we have found that the GPU-based GCD is sufficiently fast for real-time video degradation (display rates have also informally been measured well above 60 fps). This suggests that for gaze-contingent display, image processing no longer poses a significant bottleneck, obviating the need for image preprocessing or storage.

4. CONCLUSION

This article reviewed and presented current hardware-accelerated techniques for real-time simulation of arbitrary visual fields over still images and video suitable for a GCD. The main goal of this contribution

```
uniform float min_lod;
uniform sampler2D img_tex, deg_tex;

void main (void)
{
   // rgb@→ @lum coefficient vector (from @Foley et al. [1990]@)
   vec4 lum = vec4 (0.299, 0.587, 0.114, 1.0);

   // fetch degradation texture sample
   vec4 deg = texture2D (deg_tex, gl_TexCoord[1].st);

   // invert lod mapping
   float lod = (1.0 − deg.w) * min_lod;

   // fetch lod biased image texture sample
   vec4 rgb =
     texture2D(img_tex, gl_TexCoord[0].st, lod);

   // return final composite
   gl_FragColor =
     vec4(rgb.xyz * deg.xyz, rgb.w) +
       dot(lum.xyz, (rgb.xyz * (1.0 − deg.xyz)));
}
```

Listing 1. GLSL for computing spatiochromatic degradation at gaze point from an arbitrary 4-channel degradation texture map.

Fig. 9. Real-time spatiochromatic degradation of Lena.

is to alert the eye-tracking, vision sciences, and perceptual graphics communities with available computer graphics techniques facilitating the investigation of fundamental processes of visual perception. The banality of the solution is a consequence of hardware catching up to functionality that has been anticipated for over a decade, when gaze-contingent video display was anything but trivial.

The given hardware-accelerated fragment programming technique offers considerable flexibility for future perceptual and graphics research. A rather powerful but as yet unexploited benefit of fragment programming in this context is the potential for gaze-contingent color degradation. This is achieved by the use of a 4-channel degradation mask. Since only one channel (the alpha channel) is needed for resolution degradation, it is natural to use the remaining RGB channels to represent color degradation maps. Color degradation can be independently controlled in RGB color-space, since each of the RGB channels is itself a normalized image. Image color can simply be degraded by scaling a given pixel's color by a scalar found in the corresponding degradation image RGB channels. A pixel's output color is then interpolated between the pixel's full color (original) and its luminance, where luminance is obtained from a constant conversion coefficient vector [Foley et al. 1990]. Due to the independence of the RGB degradation channels, this offers a rather powerful technique for exploring perceptual effects of peripheral color degradation (see Figure 9). While peripheral visual acuity (and contrast sensitivity, e.g., see Reddy [2001] and Luebke et al. [2000]) have been studied widely, peripheral color sensitivity (and degradation) has not. Newly developed color degradation metrics, for example, following the classic evaluation paradigm of Funkhouser and Séquin [1993], could affect perceptual rendering of images, games, and the like. It is plausible that a perceptually-based color degradation metric can be empirically derived to accelerate global illumination algorithms, in manners analogous to the adaptation or contrast-sensitivity examples of Ferwerda et al. [1996] and Ramasubramanian et al. [1999], respectively. Faster global illumination algorithms, for example, ray tracing or radiosity, may in turn lead to more efficient production of computer-generated imagery.

ACKNOWLEDGMENTS

We would like to extend a special thanks to Nathan Cournia for developing the original GCD `Cg` on which the final `GLSL` version is based.

REFERENCES

Baudisch, P., DeCarlo, D., Duchowski, A. T., and Geisler, W. S. 2003. Focusing on the essential: Considering attention in display design. *Commun. ACM 46*, 3.

Bergström, P. 2003. Eye-Movement controlled image coding. Ph.D. thesis, Linköping University, Linköping, Sweden.

Bertera, J. H. and Rayner, K. 2000. Eye movements and the span of the effective stimulus in visual search. *Percept. Psychophys. 62*, 3, 576–585.

Böhme, M., Dorr, M., Martinetz, T., and Barth, E. 2006. Gaze-Contingent temporal filtering of video. In *Proceedings of the Eye Tracking Research Applications (ETRA) Symposium*. ACM, San Diego, CA, 109–116.

Burt, P. J. and Adelson, E. H. 1983. The Laplacian pyramid as a compact image code. *IEEE Trans. Commun. 31*, 4 (Apr.), 532–540.

Campbell, F. and Green, D. 1965. Monocular versus binocular visual acuity. *Nature 208*, 191–192.

Çöltekin, A. 2006. Foveation for 3D visualization and stereo imaging. Ph.D. thesis, Helsinki University of Technology, Helsinki, Finland. ISBN 951-22-8017-5, also <http://lib.tkk.fi/Diss/2006/isbn9512280175/>.

Chang, E.-C., Mallat, S., and Yap, C. 2000. Wavelet foveation. *J. Appl. Comput. Harmonic Anal. 9*, 3, 312–335.

Daly, S., Matthews, K., and Ribas-Corbera, J. 2001. As plain as the noise on your face: Adaptive video compression using face detection and visual eccentricity models. *J. Electron. Imag. 10*, 1, 30–46.

Dorr, M., Böhme, M., Martinetz, T., and Barth, E. 2005. Visibility of temporal blur on a gaze-contingent display. In *Proceedings of the 2nd Symposium on Applied Perception in Graphics and Visualization (APGV)*. ACM.

Drascic, D. and Milgam, P. 1991. Positioning accuracy of a virtual stereographic pointer in a real stereoscopic video world. In *the Conference on Stereoscopic Displays and Applications II*, Bellingham, WA. SPIE, vol. 1457, 58–69.

DUCHOWSKI, A. T. 1998. 3D wavelet analysis of eye movements. In *Proceedings of the Wavelet Applications IV Conference*, Bellingham, WA. SPIE, vol. 3391.

DUCHOWSKI, A. T. 2000. Acuity-Matching resolution degradation through wavelet coefficient scaling. *IEEE Trans. Image Process. 9*, 8 (Aug.), 1437–1440.

DUCHOWSKI, A. T. 2003. *Eye Tracking Methodology: Theory and Practice*. Springer, London.

DUCHOWSKI, A. T. 2004. Hardware-Accelerated real-time simulation of arbitrary visual fields. In *Proceedings of the Eye Tracking Research and Applications (ETRA) Symposium*, San Antonio, TX, 59.

DUCHOWSKI, A. T. AND MCCORMICK, B. H. 1995. Pre-Attentive considerations for gaze-contingent image processing. In *Proceedings of the Conference on Human Vision, Visual Processing, and Digital Display VI*, Bellingham, WA. SPIE, vol. 2411, 128–139.

FERWERDA, J. A., PATTANAIK, S. N., SHIRLEY, P., AND GREENBERG, D. P. 1996. A model of visual adaptation for realistic image synthesis. In *Proceedings of the ACM SIGGRAPH International Conference on Computer Graphics and Interactive Techniques*. ACM, New York, 249–258.

FOLEY, J. D., VAN DAM, A., FEINER, S. K., AND HUGHES, J. F. 1990. *Computer Graphics: Principles and Practice*, 2nd ed. Addison-Wesley, Reading, MA.

FOSTER, D. H., GRAVANO, S., AND TOMOSZEK, A. 1989. Acuity for fine-grain motion and for two-dot spacing as a function of retinal eccentricity: Differences in specialization of the central and peripheral retina. *Vision Res. 29*, 8, 1017–1031.

FUNKHOUSER, T. A. AND SÉQUIN, C. H. 1993. Adaptive display algorithm for interactive frame rates during visualization of complex virtual environments. In *Proceedings of the ACM SIGGRAPH International Conference on Computer Graphics and Interactive Techniques*. ACM, New York.

GEISLER, W. S. AND PERRY, J. S. 1998. Real-Time foveated multiresolution system for low-bandwidth video communication. In *Conference on Human Vision and Electronic Imaging*, Bellingham, WA.

GEISLER, W. S. AND PERRY, J. S. 2002. Real-Time simulation of arbitrary visual fields. In *Proceedings of the Eye Tracking Research and Applications (ETRA) Symposium*, New Orleans, LA. ACM, 83–153.

HAHN, P. J. AND MATHEWS, V. J. 1997. Perceptually lossless image compression. In *Proceedings of the Data Compression Industry Workshop*, Snowbird, UT.

HOLLIMAN, N. 2005. *3D Display Systems*. IOP Press. ISBN: 0 7503 0646 7. also <http://www.dur.ac.uk/n.s.holliman/Presentations/3dv3-0.pdf>, last accessed January 2007.

LEVOY, M. AND WHITAKER, R. 1990. Gaze-Directed volume rendering. In *Proceedings of the ACM SIGGRAPH International Conference on Computer Graphics and Interactive Techniques*. ACM, New York, 217–223.

LINDE, I. V. D. 2003. Space-Variant perceptual image compression for gaze-contingent stereoscopic displays. Ph.D. thesis, Anglia Polytechnic University, UK.

LIPTON, L. 1982. *Foundations of the Stereoscopic Cinema, A Study in Depth*. Van Nostrand Reinhold. ISBN O-442-24724-9. also <http://www.stereoscopic.org>.

LOSCHKY, L. C. AND MCCONKIE, G. W. 2000. User performance with gaze contingent multiresolutional displays. In *Proceedings of the Eye Tracking Research and Applications Symposium*. Palm Beach Gardens, FL, 97–103.

LOSCHKY, L. C. AND WOLVERTON, G. S. 2007. How late can you update gaze-contingent multiresolutional displays without detection? *ACM Trans. Multimedia Comput. Commun. Appl. 3*, 4 (Nov.). this issue.

LUEBKE, D. AND ERIKSON, C. 1997. View-Dependent simplification of arbitrary polygonal environments. In *Proceedings of the ACM SIGGRAPH International Conference on Computer Graphics and Interactive Techniques*. ACM, New York.

LUEBKE, D., REDDY, M., COHEN, J. D., VARSHNEY, A., WATSON, B., AND HUEBNER, R. 2003. *Level of Detail for 3D Graphics*. Morgan-Kauffman/Elsevier, San Francisco, CA. ISBN 1-55860-838-9. also <http://www.lodbook.com/>.

LUEBKE, D., VARSHNEY, A., COHEN, J., WATSON, B., AND REDDY, M. 2000. Course 41: Advanced issues in level of detail. In *Proceedings of the ACM SIGGRAPH International Conference on Interactive Techniques.* New York, NY.

MCCONKIE, G. W. AND RAYNER, K. 1975. The span of the effective stimulus during a fixation in reading. *Percept. Psychophys. 17*, 578–586.

MON-WILLIAMS, M. AND WANN, J. 1998. Binocular virtual reality displays: When problems do and don't occur. *Hum. Factors 40*, 18–24. also <http://www.abdn.ac.uk/~psy359/dept/Papers/Human Facts.pdf>.

MURPHY, H. AND DUCHOWSKI, A. T. 2001. Gaze-Contingent level of detail. In *EuroGraphics Conference*, Manchester, UK.

NATIONAL EYE INSTITUTE. 2004. Office of Communication and Health Education. Personal communique.

NATIONAL INSTITUTES OF HEALTH. 2003. Age-Related macular degeneration: What you should know. Rep. 03-2294, National Eye Institute, National Institutes of Health. also <http://www.nei.nih.gov/health/maculardegen/webAMD.pdf>, last accessed January 2007.

NIKOLOV, S. G., NEWMAN, T. D., BULL, D. R., CANAGARAJAH, N. C., JONES, M. G., AND GILCHRIST, I. D. 2004. Gaze-Contingent display using texture mapping and OpenGL: System and applications. In *Proceedings of the Eye Tracking Research and Applications (ETRA) Symposium*, San Antonio, TX, 11–18.

OHSHIMA, T., YAMAMOTO, H., AND TAMURA, H. 1996. Gaze-Directed adaptive rendering for interacting with virtual space. In *Proceedings of the Virtual Reality Annual International Symposium (VRAIS)*. IEEE, 103–110.

O'SULLIVAN, C., DINGLIANA, J., GIANG, T., AND KAISER, M. K. 2003. Evaluating the visual fidelity of physically based animations. *ACM Trans. Graphics 22*, 3, 527–536.

PARKHURST, D., CULURCIELLO, E., AND NIEBUR, E. 2000. Evaluating variable resolution displays with visual search: Task performance and eye movements. In *Proceedings of the Eye Tracking Research and Applications Symposium*, Palm Beach Gardens, FL, 105–109.

PARKHURST, D. J. AND NIEBUR, E. 2002. Variable resolution displays: A theoretical, practical, and behavioral evaluation. *Hum. Factors 44*, 4, 611–629.

PARKHURST, D. J. AND NIEBUR, E. 2004. A feasibility test for perceptually adaptive level of detail rendering on desktop systems. In *Proceedings of the Applied Perception, Graphics and Visualization (APGV) Symposium*. ACM, New York, 49–56.

RAMASUBRAMANIAN, M., PATTANAIK, S. N., AND GREENBERG, D. P. 1999. A perceptually based physical error metric for realistic image synthesis. In *Proceedings of the ACM SIGGRAPH International Conference on Computer Graphics and Interactive Techniques*. ACM, New York, NY, 73–82.

RAYNER, K. 1998. Eye movements in reading and information processing: 20 years of research. *Psychol. Bull. 124*, 3, 372–422.

RAYNER, K. AND BERTERA, J. H. 1979. Reading without a fovea. *Sci. 206*, 468–469.

REDDY, M. 2001. Perceptually optimized 3D graphics. *Comput. Graphics Appl. 21*, 5, 68–75.

REINGOLD, E. M., LOSCHKY, L. C., MCCONKIE, G. W., AND STAMPE, D. M. 2003. Gaze-Contingent multi-resolutional displays: An integrative review. *Hum. Factors 45*, 2, 307–328.

SHREINER, D., WOO, M., NEIDER, J., AND DAVIS, T. 2006. *OpenGL Programming Guide: The Official Guide to Learning OpenGL, Version 2*, 5th ed. Addison-Wesley.

TOBII TECHNOLOGY AB. 2003. Tobii ET-17 eye-tracker product description (v1.1). <http://www.tobii.se/> (last accessed January 2007).

WARE, C. 2004. *Information Visualization—Perception for Design*. Morgan-Kauffman, Elsevier, San Francisco, CA.

WATSON, B., WALKER, N., AND HODGES, L. F. 2004. Supra-Threshold control of peripheral LOD. In *Proceedings of the ACM SIGGRAPH International Conference on Computer Graphics and Interactive Techniques*, 750–759.

WATSON, B., WALKER, N., HODGES, L. F., AND WORDEN, A. 1997. Managing level of detail through peripheral degradation: Effects on search performance with a head-mounted display. *ACM Trans. Comput.-Hum. Interact. 4*, 4 (Dec.), 323–346.

WILLIAMS, L. 1983. Pyramidal parametrics. *Comput. Graphics 17*, 3 (Jul.), 1–11.

Received August 2007; accepted August 2007