REAL-TIME VIEW-DEPENDENT VISUALIZATION OF REAL WORLD GLOSSY SURFACES

Claus B. Madsen, Bjarne K. Mortensen and Jens R. Andersen
Computer Vision and Media Technology Lab, Aalborg University, Aalborg, Denmark

Keywords: HDR, radiance, Image-Based Rendering, reflectance, real-time, non-isotropic textures.

Abstract: A technique for real-time visualization of glossy surfaces is presented. The technique is aimed at recreating the view-dependent appearance of glossy surfaces under some fixed illumination conditions. The visualized surfaces can be actual real world surfaces or they can be surfaces for which the appearance is precomputed with a global illumination renderer. The approach taken is to image to surface from a large number of viewpoints distributed over the viewsphere. From these images the reflected radiance in different directions is sampled and a parameterized model is fitted to these radiance samples. Two different models are explored: a very low parameter model inspired by the Phong reflection model, and a general Spherical Harmonics model. It is concluded that the Phong-based model is best suited for this type of application.

1 INTRODUCTION

The use of images as textures in computer graphics is extremely common. These images, acquired with a camera under controlled illumination, or synthetically generated (procedurally or manually), are typically/traditionally used to modulate the diffuse reflection of light from surfaces. Therefore albedo maps could be a more proper term than the term texture map normally used, (Yu et al., 1999). Increasingly more visual realism is achieved by combining such diffuse textures (appearance is independent on viewing direction) with a global glossiness term (for the entire surface), normal maps, displacement maps, etc.

A special case application area for textures is for image-based illumination of augmented objects in Augmented Reality applications, (Debevec, 2005; Madsen et al., 2003; Havran et al., 2005; Barsi et al., 2005; Jensen et al., 2006; Madsen and Laursen, 2007; Debevec, 1998; Debevec, 2002). Here, fully omni-directional images of the environment are used for determining the illumination conditions at the location where the augmented object will be inserted into the scene. These environment maps are acquired in High Dynamic Range (HDR, where pixel values are floating point numbers) in order to fully capture the range of illumination in a real scene. Using omni-directional environment maps have the clear disadvantage that the technique assumes that the surrounding scene is distant, i.e., that distances from the augmented object to the surfaces of the rest of the scene are large compared to the size of the augmented object. Pasting the omni-directional environment map onto a coarse model of the scene, (Gibson et al., 2003) alleviates the distant scene assumption, but introduces a new assumption (or problem): since the environment map was a acquired with a camera from a certain position in the scene it is assumed that the reflected light from the surfaces of the scene is view-independent (diffuse reflectors). This poses a severe problem in scenes containing windows and/or non-isotropic light fixtures. For a survey of illumination in mixed reality see (Jacobs and Loscos, 2004).

This paper addresses the general issue of exploring techniques for view-dependent, glossy textures. Our long term goal is to use such textures for improving illumination of augmented objects (as described above). Our short term goal, as reported in the present paper, is simply to develop techniques addressing the following problem: How can we, by taking many images of a surface from a large number of viewpoints, reconstruct, in real-time, the view-dependent appearance of a glossy surface, faithfully recreating the illumination conditions present during image acquisition.

We are thus not interested in being able to change the illumination conditions, and therefore not interested in explicitly modelling reflectance functions (BRDFs). Rather, we are interested in being able to

B. Madsen C., K. Mortensen B. and R. Andersen J. (2008).
REAL-TIME VIEW-DEPENDENT VISUALIZATION OF REAL WORLD GLOSSY SURFACES.
In *Proceedings of the Third International Conference on Computer Graphics Theory and Applications*, pages 231-240
DOI: 10.5220/0001095602310240

synthesize novel views of real world glossy surfaces in real-time. Figure 1 illustrates the type of results of this work.

2 RELATED WORK

Clearly, this work falls within the field of Image-Based Rendering, IBR. Overviews of the field can be found in (Kang, 1997; Oliveira, 2002). Related work can be grouped in three categories: 1) BRDFs estimated from multiple images, 2) Light Fields, and 3) View-Dependent Texture Maps.

Estimation of reflectance functions from images has been formulated as inverse rendering, (Yu et al., 1999; Boivin and Gagalowicz, 2001; Boivin and Gagalowicz, 2002), where one or a small number images of a surface are used to estimate the parameters of low-parameter BRDFs. The advantage of such an approach is that the surface can subsequently be subjected to novel illumination conditions, but the clear disadvantage is that the estimation process requires knowledge of the light sources in the scene. Conversely, the Bidirectional Texture Function, (Dana et al., 1997), represents a technique for storing surface appearance indexed by viewing and illumination directions. Similarly, there is a large body of research into capturing reflectance maps, e.g., (McAllister et al., 2002; Lensch et al., 2003; Zickler et al., 2005). By capturing BRDFs from systematic and controlled imaging under varying view direction and illumination direction enormous flexibility is achieved and the appearance can subsequently be reconstructed under arbitrary illumination conditions. The disadvantage of this approach is that acquisition requires sampling the illumination direction as well as the viewing direction, which is impossible given our long term goal of being able to capture the view-dependent appearance of for instance a room under totally normal illumination conditions.

Light Fields and Surface Light Fields (Levoy and Hanrahan, 1996; Gortler et al., 1996; Miller et al., 1998; Wood et al., 2000) represent techniques for appearance reconstruction using more or less knowledge about scene geometry. Appearance from novel views is based on interpolation of acquired appearances from sample viewing directions. These techniques either impose limitations on the allowable set of viewing directions for novel views, or require very high quality geometric models of the scene. Furthermore, the Surface Light Field approach is vertex based and appearance reconstruction is a complicated process ill suited for rendering using acceleration.

View-Dependent Texture maps (Debevec et al., 1998) use geometric information to re-project input images to any novel view, blending among the input images based on considerations of view direction and sampling rate to avoid aliasing. A disadvantage of this technique is that is based simply on interpolation on input images and does not represent any kind of condensation of appearance samples into a compact representation.

In the present paper we explore techniques for compacting view-dependent appearance into a single parametric representation per surface point, suitable for accelerated (GPU-based) visualization in real-time.

3 OVERVIEW OF APPROACH

The chosen approach is conceptually very simple. We mount a particular surface, e.g., a poster or piece of card board with various material samples glued onto it, in a setup which allows for photographing the surface from a large number of viewpoints distributed across the upper hemisphere. Figure 1 demonstrates an OpenGL application with special purpose pixel shaders developed to reconstruct the view-dependent appearance based on models fitted to the measurements from the photographs of the surface.

In the example in figure 1 the surface was photographed from approximately 100 different viewpoints keeping illumination conditions fixed. For every point on the surface we therefore have 100 RGB measurements of that point's appearance from different view directions. For every point we then fit a low-parameter model to those 100 measurements. In the demonstrated case we have used 10 parameters per surface point. These parameters are stored in a texture which is mapped to quadrilateral in an OpenGL based application. A purpose-built pixel shader then reconstructs the view-dependent appearance of the surface. As seen, highlights move across the surface as the view changes.

4 CAPTURING APPEARANCE

In order to model and reconstruct the appearance of a surface as a function of viewing direction we need to sample (measure) the appearance from a number of viewing directions. Furthermore, since the appearance changes across the surface, e.g., due to texture, and changes in glossiness, we need to sample the appearance at a large number of distinct locations across the surface. This section describes how images

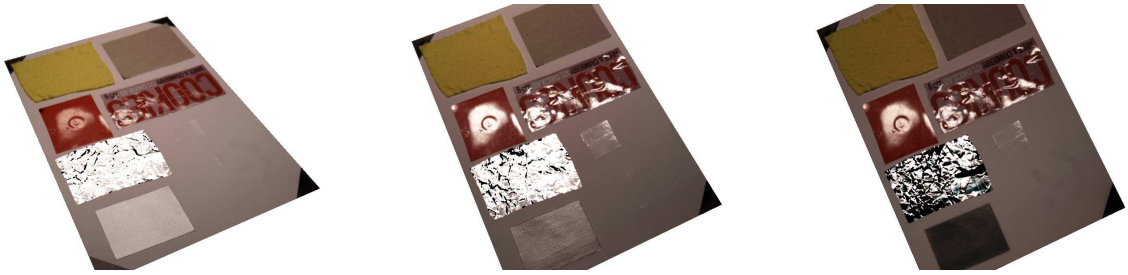


Figure 1: Three different views of a surface with different materials (in the foreground some duct tape, alu-foil and various plastics). The view-dependent appearance of the surface is reconstructed in a pixel shader in real-time (60 fps) based on a per-pixel appearance model fitted to data acquired from a large number of photos of the real surface.

are used in order to collect information about view-dependent appearance changes for a given surface under given illumination conditions.

The proper radiometric quantity to use for describing appearance is the outgoing radiance. Imaging sensors, cameras and human eyes alike, produce responses that are proportional to the incident radiance and the proportionality constant depends on the geometry of the sensor, (Dutré et al., 2003). Radiance is a radiometric quantity measured in $W/(m^2 \cdot sr)$ and describes the power leaving (or arriving at) a surface point, per solid angle, and per projected area. Radiance does not attenuate with distance and hence a certain surface looks (to a camera or to a human) equally bright independent on the distance between the observer and the surface, (provided that interaction with participating media is disregarded).

4.1 Measuring Real Surface Radiances

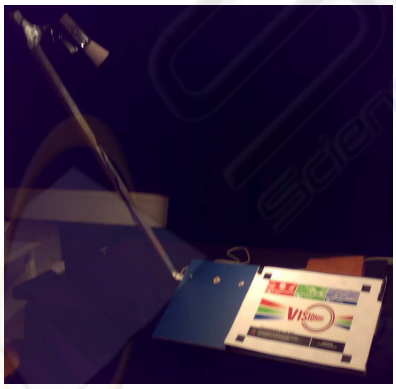


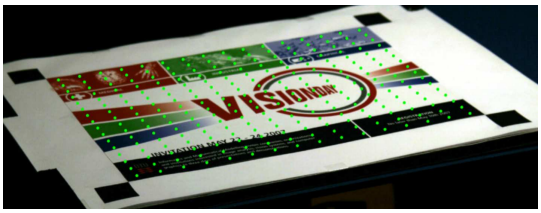
Figure 2: The setup used to acquire photographs of the surface from a all directions. The surface sample and the light source are mounted together on a setup which can pan and tilt relative to a static camera on a tripod. For these controlled experiments the light source is the only illumination in the room.

A high quality digital camera is used to take images of a surface. Figure 2 shows images of a simple setup used to acquire measurements of surface radiance. To be able to properly represent the actual dynamic range in reflected radiance from real specular surfaces we capture multiple images from each viewing position by varying the exposure. These multiple exposures are fused into a single High Dynamic Range (HDR) floating point image using the free software tool HDRShop, (Debevec et al., 2007), which is based on the techniques presented in (Debevec and Malik, 1997). In this work we have not taken steps to radiometrically calibrate the camera (Canon EOS 1Ds MARK II 16 Mega pixel digital SLR fitted with a 28 - 135 mm zoom lens) we use for acquiring images of real surfaces. Therefore the constant of proportionality between HDR pixel value and reflected surface radiance is unknown. We simply operate on the pixel values directly in further processing, acknowledging that there is an unknown scale factor. Nevertheless, the pixel values will be referred to as measured radiances, a radiance for each of the three color channels.

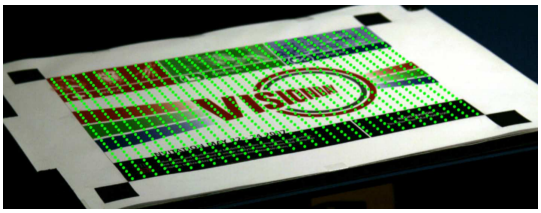
4.2 Surface Sample Points

A rectilinear sampling grid is imposed on the surface patch. Figure 3 illustrates how the surface is sampled at high density (typically 512x512 or 1024x1024). To ensure that the sample points have the same location on the surface for all viewpoints a camera calibration process is run in order to compute the position and orientation of the camera relative to the surface for all views. When combined with an a priori internal camera calibration of the lens and camera parameters this allows for projecting world coordinate points (the sample grid points) to the image plane. The black squares at the corners of the surface patch in figure 3 are used to estimate camera position and orientation. A camera calibration toolbox for MATLAB has been used, (Bouguet, 2007), which also exist as an OpenCV toolbox for implementation in a C program, (SourceForge.net, 2007).

Since the camera by no means is infinitely far away from the surface being imaged the direction vec-



(a) Resolution = 16x16.



(b) Resolution = 32x32.

Figure 3: Two different low sampling resolutions for illustrative purposes. In reality higher resolutions are required to reconstruct text and detailed figures on the surface. The black squares at the corners are used for calibrating the camera position and orientation to the surface coordinate system.

tor to the focal point of the camera is not the same for all sample grid locations. That is, for a given image the viewing direction for all the surface points is not the same. Therefore, alongside with storing the RGB radiances for a given surface point for a given image it is also necessary to store the viewing direction which the radiance measurement corresponds to.

4.3 Constructing an Observation Map

An Observation Map (OM) is a large multi-dimensional texture. It contains all the information accumulated from the sequence of images taken of the surface. For a given surface point each image (each viewpoint) produces three radiance values (RGB) and two parameters describing the viewing direction in spherical coordinates (the distance to the camera is irrelevant as radiance is independent on distance). The OM thus contains $S \cdot T \cdot N$ times five floating point values, where S and T give the resolution of the sampling grid, and N is the number of viewpoints from which the surface has been imaged.

The viewing directions (camera positions relative to surface coordinate system) used are found by subsampling an icosahedron in order to get uniformly distributed positions across the upper hemisphere, (Ballard and Brown, 1982). Between 100 to 300 directions are used for the real surfaces shown in this paper, which in reality is not quite enough for highly glossy surfaces. Our rig for acquiring images (figure 2) is manual. An automated (motorized and controllable) rig would make acquisition much easier. Figure 4 shows the measured radiances for a specific sample point on a surface.

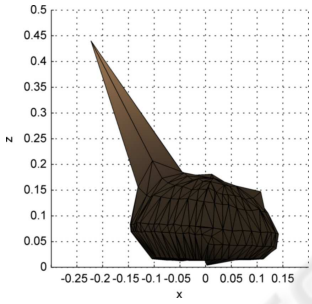


Figure 4: Measured radiance from 278 views of a point on a glossy poster. Notice the clear highlight spike caused by the single light source illuminating the surface during acquisition. The colors in the figure correspond to the balance between the measured RGB radiances, but since the highlight is so strong most of the radiances appear black. This figure illustrates the content of the Observation Map at each surface sample point.

A brute force Image-Based Rendering approach using the OM directly for subsequent real-time visualization would involve creating a pixel shader which used the view vector to interpolate between the observed radiances for a given texel. Apart from being cumbersome this has three obvious disadvantages: 1) the number of observations may vary from surface to surface complicating the shader code, 2) the observations per texel would have to be sorted some how in order to make it possible to efficiently search for nearest neighbors among the observation directions for a given view vector, and 3) the shear size of the OM is prohibitive (1.5 GByte for a 512x512 map, with 300 observation directions). Obviously, some kind of compression is needed and in this work we have opted for fitting a parameterized model to the measured radiances in order to be able to reconstruct the radiation diagram, an example of which was illustrated in figure 4. Section 5 addresses the modelling issue.

4.4 Synthesized Surface Radiances

Experimentally we have for this paper worked primarily with trying to re-visualize real world surfaces but all techniques are equally applicable to basing the modelling process on synthesized surface radiances computed with a Global Illumination solver, for example a ray tracer/path tracer. It would be quite straight forward to alter a path tracer to directly produce the OM data needed per surface point. This would make it possible to visualize precomputed view-dependent global illumination effects in real time.

5 PARAMETRIC MODELLING OF RADIANCES

We are trying to model the outgoing radiance distribution, e.g., figure 4, for a surface point as a function of viewing direction. Depending on the surface material properties and the illumination conditions of the environment this radiance distribution can be arbitrarily complex. The worst case situation is a mirror surface which reflects the environment and the outgoing radiance from a point on the surface can thus change with very high frequency as a function of view direction change. Below we describe two models that can be used to approximate the outgoing radiance distribution for surfaces with some degree of specularity. We shall return briefly to the issue of frequency content of the outgoing radiance distribution in section 5.3.

5.1 The Multiple Highlight Model

This model is inspired by the Phong reflection model, although it should be kept in mind that this work does not attempt to model reflection properties. We are trying to model reflected radiance, which obviously is a *combination* of reflection properties and illumination conditions as typically formulated by the Rendering Equation, (Jensen, 2001; Dutré et al., 2003).

The Multiple Highlight model (henceforth the MH model) is inspired by the Phong reflection model in the sense that it models the reflected radiance from a material which can be described by the Phong reflection model, and which is illuminated by N point/directional light sources. The MH model can be formulated as a combination of an ambient term and a sum of highlight cosine lobes:

$$L(\vec{v}) \quad = \quad K_d + \sum_{i=1}^{N} K_{si} \cdot \left(\vec{d_i} \cdot \vec{v}\right)^m \qquad (1)$$

Here $L(\vec{v})$ is the outgoing radiance in the viewing direction, $\vec{v}$. N is a number of highlights that the model must encompass. K_d is a view-independent, "ambient", term and can be represented with 3 parameters (one for each color channel). K_{si} is a specular highlight "amplitude" for each of the N highlights. Again, since there are three color channels, 3 parameters are needed to represent each K_{si}. The dot product between $\vec{d_i}$ and $\vec{v}$ (both vectors are assumed to be unit length) modulates the specular highlight with the cosine of the angle between the viewing direction and a main "highlight direction", $\vec{d_i}$. The width of this cosine lobe is controlled by raising the cosine to some power, controlled by the shininess parameter, m. The shininess is a characteristic of the surface and is therefore the same for all the N highlights, i.e., does not depend on i. The highlight direction for the ith highlight, $\vec{d_i}$, can be parameterized by two spherical directions, ϕ_i and θ_i. When fitting the MH model to acquisition data we have opted to parameterize it by its three cartesian coordinate components, though, in order to have simpler expressions and less non-linear behaviour. In total the MH model requires 3 (ambient term), plus N times 6 (three for specular amplitude plus three for highlight direction) plus 1 (shininess) parameters, equalling $4 + N \cdot 6$ parameters.

A collection of cosine lobes somewhat similar to eq. 1 were used in (Gibson et al., 2001) for modelling non-isotropic radiation from virtual point light sources, but in that work the exponent was a fixed number not estimated during fitting.

The Phong reflection model is based on point or directional light sources which in reality do not exist. Even the sun subtends a non-infinitesimal solid angle as seen from the Earth (a 0.53 degree diameter disc). The reflected radiance from a specular surface modeled by a Phong reflection model and illuminated by an area light source will exhibit a thicker highlight lobe for a given shininess m than the same surface illuminated by a point light source. Therefore, by using the Phong-like specular highlight term we are not delimiting ourselves from modelling highlights caused by area light sources. The estimated shininess, m, would just be smaller than the actual shininess of the surface, in order to encompass the thicker highlight lobe.

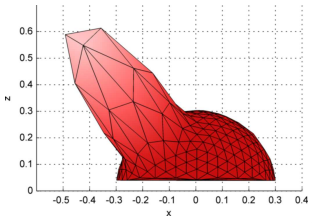
As stated the MH model requires $4 + N \cdot 6$ parameters, e.g., 10 to model one highlight, 16 to model 2 highlights. Each image (viewing direction) of a surface point provides three radiance sample values (one for each of the three color channels). Therefore, at a minimum four viewing directions for each point is needed to provide enough equations to solve for a model with one highlight, and six directions are needed to solve for two highlights. In practice many more observations are needed to ensure a good model fit the critical part being to ensure getting direction samples that clearly describe the highlight(s).

Model fitting has been implemented in MATLAB using the built-in **lsqcurvefit** function. This function allows specification of validity ranges of the parameters to be estimated, in order to bound the search space. K_d and K_{si} are constrained to the interval $[0,1]$ for all color channels, since the measured radiance values are normalized to this range prior to performing the model fitting. The highlight direction vector, $\vec{d_i}$, is constrained to the upper hemisphere, and the shininess, m, must be larger than 1.

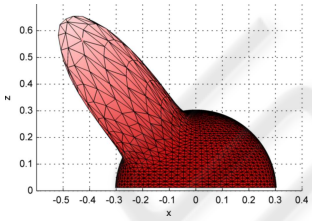
To ensure fast convergence it is important with reasonable initial estimates for the parameters to be

fitted. Initial estimates for K_d are set to the average of all direction radiance samples. The initial estimates for K_{si} are set to zero, whereas the highlight direction is initialized to the direction of the strongest radiance sample. Shininess, m, is initialized to 10. Convergence is obtained within 5 to 10 iterations for synthetic data and within 10 to 20 iterations for real data.

Figure 5 shows results of fitting a 10 parameter (one highlight) MH model to synthetically generated radiance samples of a surface. Synthetic radiance samples are generated in the following manner. A square surface is modelled in Autodesk 3DS Max 8 and illuminated by a single, white point light source. 289 uniformly distributed viewpoints are generated from the upper hemisphere of a subsampled icosahedron and the surface is rendered from each of these views. Using these images an observation map is generated as described in section 4.3, and the MH model is fitted to each of the surface sample points in the observation map. Figure 5 concerns an arbitrarily chosen sample point on the surface.



(a) Input radiances.



(b) Fitted MH model.

Figure 5: Top: 289 synthetically generated reflected radiance samples uniformly distributed across the upper hemisphere. Bottom: One-highlight MH model fitted to the samples. The mean percentage residuals are $[0.94, 2.34, 2.34]$ for R, G, and B, respectively.

5.2 The Spherical Harmonics Model

As an alternative to the MH model described above we have also experimented with the much more general Spherical Harmonics framework. Spherical Harmonics (SH) comprise a linear set of basis function which can be used to represent spherical functions, similarly to how Fourier basis functions are used describe normal N-dimensional functions. (Green, 2003) provides an excellent introduction to practical use of SH based approaches. It is beyond the scope of the present paper to give an exhaustive description of the fundamentals of the SH approach. We shall focus on the two important issues of 1) how to compute the SH coefficients for a given data input, and 2) how to handle the issue that the SH approach in principle assumes signals that are periodical over the sphere and the data we are fitting to are only valid for the upper hemisphere.

In the continuous case the ith SH coefficient, c_i, is found by projecting the signal (the reflected radiance distribution) onto the ith SH basis function, $y_i(\theta,\phi)$. This in turn is done through convolution over the entire sphere. In the present case we have a set of radiance samples, L_k, $k = 1 \ldots K$, where K is the number of views stored in the OM for the given surface point. These samples of the radiance distribution function can be projected onto the SH basis vectors using numerical integration:

$$c_i \approx \sum_{K}^{k=1} L_k \cdot y_i(\theta_k, \phi_k) A_k \qquad (2)$$

where (θ_k, ϕ_k) are the spherical angles of the viewing direction of the kth sample point, and A_k is the solid angle subtended by the viewing direction (each sample represents a certain potion of the sphere of directions). These solid angles are computed by projecting all the viewing directions (on a unit sphere) onto the plane, performing a Voronoi tessellation, and computing the area of each Voronoi polygon. These areas are in turned weighted by $1/cos(\theta_k)$ to compensate for the projection onto the plane.

The other non-trivial issue concerns handling the missing samples for the lower hemisphere. Several different approaches have been tried: 1) only integrating over the upper hemisphere (the Zero Hemisphere, ZH, approach), (Sloan et al., 2002), 2) mirroring the upper hemisphere, (Westin et al., 1992), and 3) the Least Squares Optimal Projection (LSOP) method, (Sloan et al., 2003). Experimentally the LSOP method performs better, confirming results in (Sloan et al., 2003).

5.3 Discussion On Frequency Content

Both models can be made arbitrarily complex, and can thus in principle be used to model the outgoing radiance from a surface point even for mirror surfaces. The cost is obviously an increase in the number of parameters. For there to be any “space savings”, or compression factor, in modelling the radiance distribution with a function, it must be reasonable to assume that the radiance distribution is sufficiently smooth that it

can be faithfully represented by the chosen parameterization. In the case of the Spherical Harmonics model there is also an aliasing issue to consider if the number of SH coefficients is too small compared to the frequency content of the radiance distribution.

In the experiments section (section 7) we demonstrate that the SH model cannot faithfully model the reflected radiance from a glossy poster even with 64 parameters, and this is for a case where only one light source is illuminating the material. Obviously, the more light sources the more highlights, which in turn leads to a higher frequency content in the radiance distribution. A 64 parameter (per color channel) SH model for a 512x512 surface texture requires 192 MByte memory with no compression.

6 REAL-TIME VISUALIZATION

A demonstrator system has been implemented in C++ using OpenGL. A multi-dimensional texture containing the parameters estimated for the chosen model (beit the MH model or the SH model) is mapped to a surface through uv coordinates. Using the mouse the user can rotate the surface using a world-in-hand scheme combined with moving toward or away from the surface. In a vertex program the viewing direction is computed and transformed to tangent space. This viewing direction is interpolated over the surface and passed to a fragment shader.

The fragment shader depends on the chosen model. Assuming that the number of parameters used for the given model is passed to the shader there is no significant overhead in implementing the shader generally enough that it can encompass an arbitrary number of parameters. It is quite straight forward to reconstruct reflected radiance given viewing direction for both models, especially for the MH model, where the reconstruction expression is given by equation 1. For the SH model reconstruction is found by computing the values of all the basis functions in the given viewing direction and finding the dot product with the coefficient vector:

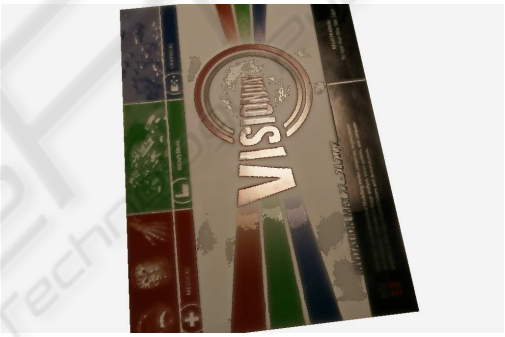
$$L(\vec{v}) \approx \sum_{i=0}^{n^2-1} c_i y_i(\vec{v}) \qquad (3)$$

where n is the number of bands used. An SH model with 64 parameters encompasses 8 bands, 0 through 7. In our implementation the $y_i(\vec{v})$ values are computed directly in the shader instead of storing them in high resolution tables and interpolating.

The fragment shader performs all computations in floating point, and as a result the reconstructed radiance for a given fragment is floating point. Prior to display the radiance values are tone-mapped to LDR 8 bit RGB brightness values as brightness $= 1 - exp(-\gamma \cdot L(\vec{v}))$, where γ is an interactively adjustable exposure value.

7 EXPERIMENTAL RESULTS

The described techniques have been implemented and tested on a single processor 2.2 GHz AMD athlon 64 machine with 3 GByte RAM. The computer is running 32 bit Windows XP (emulating 32 bit). The graphics card is an NVIDIA GeForce 6800 series card. On this machine the visualization application is running at 60 fps (limited to refresh rate) with a 10 parameter MH model, and 20 and 10 fps with a 36 and a 64 parameter (per color) SH model, respectively. There is much room for improvement on efficiency of the shader code.



(a) 10 parameter MH model

(b) 3x64 parameter SH model

Figure 6: A glossy poster reconstructed using the MH and the SH models, respectively. The MH models highlights much better, but has problems with the white areas on the poster. See text for explanation.

Figure 6 compares the visual performance of the MH and SH models. At first glance the SH model performs much more pleasing, since the MH model appears to give a very spotty result. In reality the SH model basically fails to represent the highlight areas.

The 64 parameter SH projection simply cannot handle the narrow highlight lobe correctly. On the other hand the MH model (the simple Phong inspired model) actually recreates the highlights very accurately, only there are some areas where it does not catch them (large regions of the white part of the poster). The problem does not lie in the MH model, though. When inspecting the data in the Observation Map (OM), it is seen that in the white areas the highlight lobe is simply so narrow that it is not captured from any viewpoint. Figure 7 shows the OM data for a sample point in a white poster area.

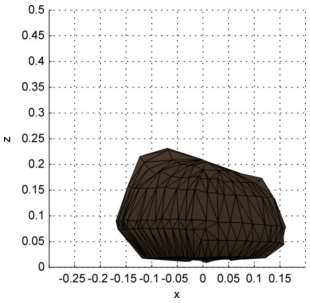


Figure 7: For most of the white area on the poster the highlight is not captured in the radiance samples and consequently does not appear in the subsequent visual reconstruction. Compare to figure 4 which illustrates a sample point in one of the few white poster areas where the highlight actually *is* properly captured.

Two lessons are learned immediately from this: 1) roughly 300 viewpoint samples distributed over the viewsphere is not enough to provide good samples for a material such as a glossy poster, and 2) the SH model requires more than 64 parameters to represent highlights from such a material.

Another experimental result is illustrated in figure 8. When looking at the yellow cloth in the foreground it is seen that the visual reconstruction method is capable of handling effects that are usually achieved through displacement mapping. More specifically the depth discontinuity along the edges of the yellow cloth correctly exhibits occlusion/dis-occlusion and shadows appear and vanish consistently with a visual interpretation of a non-planar surface.

More generally the demonstrated approach with modelling and reconstructing appearance embeds illumination, reflectance, normal, and depth-discontinuity information into a single compact representation. The advantage being that it can represent and reconstruct all these phenomena, and the disadvantage being that it is just an appearance model, which does allow for a change in any of the embedded information.

(a) View 1

(b) View 2

Figure 8: Two different reconstructed views of a surface with multiple materials. Notice the correct handling of the displacement and shadow at the edges of the thick yellow cloth material in the foreground.

8 DISCUSSIONS AND FUTURE WORK

This work has explored the possibilities for creating non-diffuse textures which can be reconstructed (visualized) in real-time based on direction to the viewer. With the current size of GPU memory it is clear that the parameter maps used to represent these non-diffuse textures must be as economical as possible in their memory requirements. As described a 10 parameter (one highlight) MH model for a 512x512 texture requires 10 MBytes. A three highlight MH map requires 22 MBytes. In comparison a 64 parameter (per color channel) SH map of the same resolution requires a staggering 192 MBytes.

We therefore believe the MH model to be the more viable approach. In addition the MH model captures the nature of highlights from a real glossy surface better than the SH model (given that it is unrealistic to increase the number of SH parameters). It could be said that the MH model in this context is a more “model-based model” than the SH model, which is completely general. The drawback of the MH model is that it is necessary to a priori determine how many highlight

to include when fitting the model to data. We have successfully fitted two-highlight MH models to data (not reported in this paper), and we have fitted one-highlight models to data with two highlights. In the latter case, if highlights are close (on the hemisphere) the resulting fit is a broad, soft highlight. If the highlights are further apart the smaller of the two highlights is basically ignored in the fit. An important area for future research is to explore robust techniques for fitting the MH model to general data with no a priori knowledge, e.g., by gradually increasing the number of highlights until the optimal fit is achieved.

Another approach to combating the memory requirements of the parameter maps is to switch from a texel-based to a vertex-based approach. For some applications (when surfaces are not highly textured) it may be sufficient to store radiance distribution models per vertex in a high resolution mesh, and then perhaps use SH models with higher number of parameters.

As hinted in section 1 we also wish to continue this work in a direction where the view-dependent textures are used for illuminating augmented objects in Augmented Reality applications. We are exploring the use of an Irradiance Volume, (Greger et al., 1998), approach, where the Irradiance Volume is computed in real-time based on the view-dependent textures. This would alleviate the assumption that the environment is infinitely distant, an assumption typically made when using image-based approaches to illumination in Augmented Reality, (Debevec, 2005; Madsen et al., 2003; Havran et al., 2005; Cohen and Debevec, 2001; Barsi et al., 2005).

9 CONCLUSIONS

It has been demonstrated that it is possible to model real-world glossy surface appearance with a low parameter model fitted to measured reflected radiance sampled over the viewsphere. The parameters of the fitted models are stored in texture maps which are used by pixel shaders for hardware accelerated real-time visualization.

Two different modelling schemes have been compared, one being a specific highlight model (the Multiple Highlight model) inspired by the Phong reflection model, and the other being the completely general Spherical Harmonics framework. Experiments demonstrated that the MH model is the only viable approach due to the fact that the SH models require too many coefficients to faithfully represent highlights from glossy materials, resulting in parameter textures that cannot fit into contemporary texture memory.

It was also seen that the number of view samples needs to be very high (several hundreds) in order to properly capture highlights if uniform sampling is used.

The advantage of the approach is that photo-realistic appearance of a real-world glossy surface can be achieved, which compresses the combined effects of illumination, reflectance, surface normal and moderate height differences into a single appearance model which can be reconstructed in real-time.

ACKNOWLEDGEMENTS

This research is funded by the CoSPE project (26-04-0171) under the Danish Research Agency.

REFERENCES

Ballard, D. H. and Brown, C. M. (1982). *Computer Vision*. Prentice-Hall.

Barsi, A., Szirmay-Kalos, L., and Szécsi, L. (2005). Image-based illumination on the gpu. *Machine Graphics and Vision*, 14(2):159 – 169.

Boivin, S. and Gagalowicz, A. (2001). Image-based rendering of diffuse, specular and glossy surfaces from a single image. In *Proceedings: ACM SIGGRAPH 2001*, Computer Graphics Proceedings, Annual Conference Series, pages 107–116.

Boivin, S. and Gagalowicz, A. (2002). Inverse rendering from a single image. In *Proceedings: First European Conference on Color in Graphics, Images and Vision, Poitiers, France*, pages 268–277.

Bouguet, J.-Y. (2007). *Camera Calibration Toolbox for Matlab, www.vision.caltech.edu/bouguetj/calib_doc/*.

Cohen, J. M. and Debevec, P. (2001). *The LightGen HDRShop plugin*. www.hdrshop.com/main-pages/plugins.html.

Dana, K. J., van Ginneken, B., Nayar, S. K., and Koenderink, J. J. (1997). Reflectance and texture of real-world surfaces. *IEEE Conference on Computer Vision and Pattern Recognition*.

Debevec, P. (1998). Rendering synthetic objects into real scenes: Bridging traditional and image-based graphics with global illumination and high dynamic range photography. In *Proceedings: SIGGRAPH 1998, Orlando, Florida, USA*.

Debevec, P. (2002). Tutorial: Image-based lighting. *IEEE Computer Graphics and Applications*, pages 26 – 34.

Debevec, P. (2005). A median cut algorithm for light probe sampling. In *Proceedings: SIGGRAPH 2005, Los Angeles, California, USA*. Poster abstract.

Debevec, P. and Malik, J. (1997). Recovering high dynamic range radiance maps from photographs. In *Proceedings: SIGGRAPH 1997, Los Angeles, CA, USA*.

Debevec, P. E., Borshukov, G., and Yu, Y. (1998). Efficient view-dependent image-based rendering with projective texture-mapping. In *9th Eurographics Rendering Workshop,*.

Debevec et al., P. (2007). *www.hdrshop.com*.

Dutré, P., Bekaert, P., and Bala, K. (2003). *Advanced Global Illumination*. A. K. Peters.

Gibson, S., Cook, J., Howard, T., and Hubbold, R. (2003). Rapic shadow generation in real-world lighting environments. In *Proceedings: EuroGraphics Symposium on Rendering, Leuwen, Belgium*.

Gibson, S., Howard, T., and Hubbold, R. (2001). Flexible image/based photometric reonstruction using virtual light sources. In Chalmers, A. and Rhyne, T.-M., editors, *Proceedings: Annual Conference of the European Association for Computer Graphics, EUROGRAPHICS 2001, Manchester, United Kingdom*.

Gortler, S. J., Grzeszczuk, R., Szeliski, R., and Cohen, M. F. (1996). The lumigraph. In *SIGGRAPH '96: Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 43–54, New York, NY, USA. ACM Press.

Green, R. (2003). Spherical harmonic lighting: The gritty details. Technical report, Sony Computer Entertainment America.

Greger, G., Shirley, P., Hubbard, P. M., and Greenberg, D. P. (1998). The irradiance volume. *IEEE Computer Graphics and Applications*, 18(2):32–43.

Havran, V., Smyk, M., Krawczyk, G., Myszkowski, K., and Seidel, H.-P. (2005). Importance Sampling for Video Environment Maps. In Bala, K. and Dutré, P., editors, *Eurographics Symposium on Rendering 2005*, pages 31–42,311, Konstanz, Germany. ACM SIGGRAPH.

Jacobs, K. and Loscos, C. (2004). State of the art report on classification of illumination methods for mixed reality. In *EUROGRAPHICS*, Grenoble, France.

Jensen, H. W. (2001). *Realistic Image Synthesis Using Photon Mapping*. A. K. Peters.

Jensen, T., Andersen, M., and Madsen, C. B. (2006). Real-time image-based lighting for outdoor augmented reality under dynamically changing illumination conditions. In *Proceedings: International Conference on Graphics Theory and Applications, Setúbal, Portugal*, pages 364–371.

Kang, S. B. (1997). A survey of image-based rendering techniques. Tehcnical report series CRL 97/4, Cambridge Research Laboratory, Digital Equipment Corporation, One Kendall Square, Building 700, Cambridge, Massachusetts 02139.

Lensch, H. P. A., Kautz, J., Goesele, M., Heidrich, W., and Seidel, H.-P. (2003). Image-based reconstruction of spatial appearance and geometric detail. *ACM Transactions on Graphics*, 22(2):234–257.

Levoy, M. and Hanrahan, P. (1996). Light field rendering. In *SIGGRAPH '96: Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 31–42, New York, NY, USA. ACM Press.

Madsen, C. B. and Laursen, R. (2007). A scalable gpu-based approach to shading and shadowing for photo-realistic real-time augmented reality. In *Proceedings: International Conference on Graphics Theory and Applications, Barcelona, Spain*, pages 252 – 261.

Madsen, C. B., Sørensen, M. K. D., and Vittrup, M. (2003). Estimating positions and radiances of a small number of light sources for real-time image-based lighting. In *Proceedings: EUROGRAPHICS 2003, Granada, Spain*, pages 37 – 44.

McAllister, D. K., Lastra, A. A., and Heidrich, W. (2002). Efficient rendering of spatial bi-directional reflectance distribution functions. In *Graphics Hardware 2002*, pages 79–88.

Miller, G. S. P., Rubin, S. M., and Ponceleon, D. B. (1998). Lazy decompression of surface light fields for pre-computed global illumination. In *Rendering Techniques*, pages 281–292.

Oliveira, M. M. (2002). Image-based modelling and rendering: A survey. *RITA - Revista de Informatica Teorica a Aplicada*, 9(2):37 – 66. Brasillian journal, but paper is in English.

Sloan, P.-P., Hall, J., Hart, J., and Snyder, J. (2003). Clustered principal components for pre-computed radiance transfer. In *Proceedings: SIGGRAPH 2003, New York, New York, USA*, pages 382 – 391.

Sloan, P.-P., Kautz, J., and Snyder, J. (2002). Precomputed radiance transfer for real-time rendering in dynamic, low-frequency lighting environments. In *Proceedings: SIGGRAPH 2002, San Antonio, Texas, USA*, pages 527 – 536.

SourceForge.net (2007). *OpenCV Computer Vision Library, www.sourceforge.net/projects/opencv/*.

Westin, S., Arvo, J., and Torrance, K. (1992). Predicting reflectance functions from complex surfaces. In *Proceedings: SIGGRAPH 1992, New York, New York, USA*, pages 255 – 264.

Wood, D. N., Azuma, D. I., Aldinger, K., Curless, B., Duchamp, T., Salesin, D. H., and Stuetzle, W. (2000). Surface light fields for 3D photography. In *Proceedings: SIGGRAPH 2000*, pages 287–296.

Yu, Y., Debevec, P., Malik, J., and Hawkins, T. (1999). Inverse global illumination: Recovering reflectance models of real scenes from photographs. In *Proceedings: SIGGRAPH 1999, Los Angeles, California, USA*, pages 215 – 224.

Zickler, T., Enrique, S., Ramamoorthi, R., and Belhumeur, P. (2005). Reflectance sharing: Image-based rendering from a sparse set of images. In *Rendering Techniques 2005: 16th Eurographics Workshop on Rendering*, pages 253–264.