Tutorial Notes for the DAGM 2001

A Framework for the Acquisition, Processing and Interactive Display of High Quality 3D Models

Research Report

MPI-I-2001-4-005 September 2001

Hendrik P. A. Lensch, Michael Goesele, Hans-Peter Seidel

Max-Planck-Institut für Informatik
Stuhlsatzenhausweg 85
66123 Saarbrücken
Germany
lensch,goesele,hpseidel@mpi-sb.mpg.de

Abstract

This tutorial highlights some recent results on the acquisition and interactive display of high quality 3D models. For further use in photorealistic rendering or object recognition, a high quality representation must capture two different things: the shape of the model represented as a geometric description of its surface and on the other hand the appearance of the material or materials it is made of, e.g. the object's color, texture, or reflection properties.

The tutorial shows how computer vision and computer graphics techniques can be seamlessly integrated into a single framework for the acquisition, processing, and interactive display of high quality 3D models.

1 Introduction

The rapid advances of consumer level graphics hardware make it possible to render increasingly complex and accurate models in real time. Computer-generated movies are getting more and more realistic and users will soon demand a similar level of realism in a wide range of every day applications such as computer games, digital libraries and encyclopedias, or e-commerce applications. Being able to efficiently generate, process and display the necessary models will become a more and more important part of computer vision and computer graphics.

To fulfill these requirements a high quality representation must capture two different things: the shape of the model represented as a geometric description of its surface and the appearance of the material or materials it is made of, e.g. the object's color, texture, or reflection properties. Subsequently, geometry and surface appearance data must be integrated into a single digital model which must then be stored, processed, and displayed, trying to meet several conflicting requirements (such as realism versus interactive speed).

As more and more visual complexity is demanded, it is often infeasible to generate these models manually. Automatic and semi-automatic methods for model acquisition are therefore becoming increasingly important.

A system built to acquire and to process the necessary data relies on computer vision techniques as well as on computer graphics techniques. To obtain the geometry of an object, a 3D scanner is used. The output is transformed into a mesh representation and further processed to reduce noise and complexity. The surface properties of the object are acquired by taking a number of images with constrained lighting. These images have to be registered to the 3D geometry by use of camera calibration techniques. By inspecting the images, the object's texture, the spatially varying reflection properties and microstructure (normal maps) can be extracted.

Combining all the data, a compact representation of the object can be obtained that allows for accurately shaded, photorealistic rendering from new viewpoints under arbitrary lighting conditions. In addition, the high quality 3D model may be used for object recognition and material investigation.

This tutorial highlights some recent results on the acquisition and interactive display of high quality 3D models. It shows how computer vision and computer graphics techniques can be seamlessly integrated into a single framework for the acquisition, processing, and interactive display of high quality 3D models. Some examples will illustrate the approach. Finally, we point out some remaining questions and important areas for future research concerning both computer graphics and computer vision.

2 3D Object Acquisition Pipeline

In this tutorial we focus on the generation of high quality 3D models containing the object's geometry and the surface appearance Such a model contains information needed for many computer graphics or computer vision applications. However, there are also other types of high quality models such as volumetric or image-based models (e.g., computer tomography data sets, light fields [31]) that are suitable for different applications.

In our case, the generation of a high quality 3D model for a real world object includes several, partially independent steps. Figure 2.1 shows an overview of these steps.

First, the geometry and the texture of the object are acquired. Typically, different techniques and acquisition devices for the geometry and the texture are applied which makes it necessary to align both data sets in a separate registration step. However, it is also possible to derive geometry information from texture data and vice versa. Various subsequent processing steps are necessary to extract information such as reflection properties or normal maps from the input data.

Once a complete model is created it can be resampled, converted to a different data representation, or compressed to make it suitable for a particular application scenario. Finally, the target application should be able to display the model interactively without omitting any important information.

In the following sections we give a detailed description of all the steps of the 3D object pipeline. In Section 3 we start with image-based acquisition techniques followed by acquisition techniques for appearance properties in Section 4. We give an overview over the acquisition of 3D geometry in Section 5 and describe a technique to register texture and image data in Section 6. Section 7 introduces several methods to display the acquired models interactively. We present some examples of acquired models in Section 8 before we conclude with Section 9.

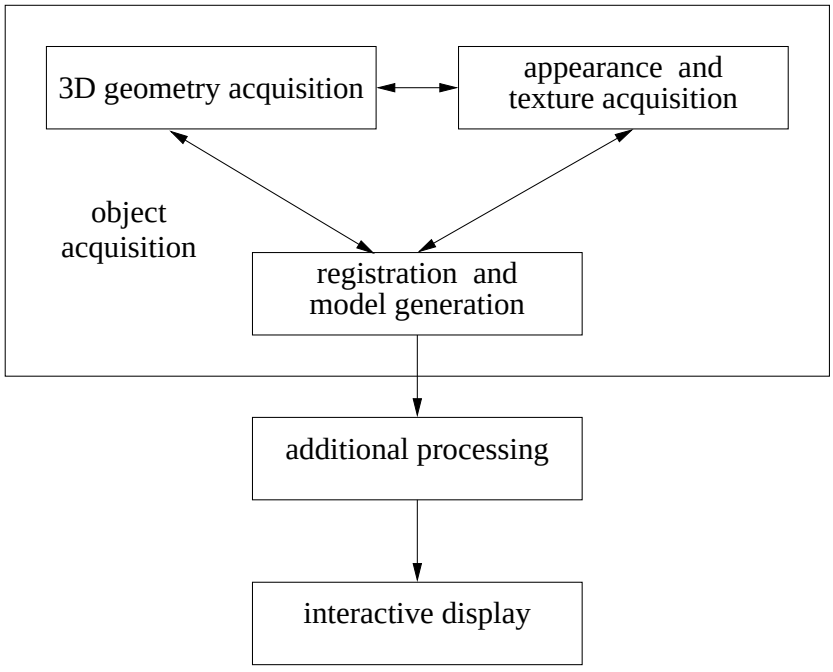

Figure 2.1: *The 3D object pipeline. Depending on the applied techniques geometry acquisition, texture and appearance acquisition, and registration depend on each other in different configurations.*

3 Image-Based Acquisition Techniques

Today, image-based techniques become more and more popular to acquire models of real world objects (see Section 4). A key element of these methods is a digital camera to capture images of the object from which various properties of the object can be derived. The large number of measurements that can be made in parallel (i.e. one per pixel for a digital camera) lead to efficient methods to sample complex functions such as four-dimensional BRDFs. However, these measurements can only be meaningful if the equipment used is appropriate for the measurements, if the properties of the devices are known, and if the relevant parts are calibrated.

3.1 Photographic Equipment

Both analog and digital cameras can be used for measurement purposes. The advantages of analog photography include the high resolution of analog film (especially in combination with commercial high quality digitization services as the Kodak Photo CD), its comparably large dynamic range, and the huge selection of available cameras, lenses and film types. However, the development and scanning of film can take quite long and the resulting images are not naturally registered against the camera lens system.

In contrast to that, the position of the imaging sensor in a digital camera remains fixed with respect to the lens system which makes it easy to capture several aligned images from the same position under different lighting conditions. If the digital camera is capable of returning the raw image sensor data it is possible to calibrate the individual sensor elements to account for variations on the sensor [1, 14].

Most consumer quality digital cameras use the lossy JPEG compression format to store their images although more recent cameras are often also capable of producing images in a lossless compressed format. The lossy JPEG compression

introduces compression artifacts which makes them rather unsuitable for measurement purposes. Additional artifacts can occur due to various steps in the image processing chain of digital cameras such as sharpening operations or the color reconstruction in single chip cameras. The imaging community developed a large number of methods to characterize various aspects of a digital camera such as the modulation transfer function (MTF) [52]. These methods are not only helpful to choose an appropriate camera but can also be used to debug a measurement setup when an error occurs.

3.2 Lighting Equipment

For most algorithms that reconstruct the appearance properties of an object from images, it is important to control the lighting conditions exactly. Although this is also true for images taken by a regular photographer, the requirements differ strongly. A point light source, i.e. a light source where all light is emitted from a single point is ideal for many of the techniques mentioned above but is rarely used in photography as it casts very hard shadows. A perfectly constant and diffuse lighting is ideal to capture the color of an object but leads from a photographers point of view to very flat looking images due to the absence of shadows.

Figure 3.1: *A view of our photo studio with black, diffuse reflecting material on the floor, walls, and ceiling. This image was generated from a High Dynamic Range image to which a tone-mapper has been applied.*

The surrounding of an object has also a huge influence on the lighting situation, especially if the object has a specular reflecting surface. In order to minimize this influence the measurement region should be surrounded with dark material that absorbs as much light as possible. Furthermore, the light that is not absorbed

should be reflected in a very diffuse way. Figure 3.1 shows a view of our photo studio whose floor, walls, and ceiling are covered with black, diffuse reflecting material to reduce the influence of the environment on the measurements as much as possible.

A more technical and in-depth discussion of camera and lighting issues can be found in [12].

3.3 Camera Calibration

When using a camera as a measurement device various aspects should be calibrated in order to guarantee high-quality results and the repeatability of the measurements.

3.3.1 Geometric Calibration

The properties of the camera transformation which describes how an object is projected onto the camera's image plane should be recovered e.g. using [49, 54, 18]. These methods generally use an image or a set of images of a calibration target (e.g. a checkerboard pattern) to determine camera parameters such as the focal length of the lens, the location of the optical axis relative to the imaging sensor (principal point), and various distortion coefficients. Once this information is known, a ray in space can be assigned to each pixel in an image.

3.3.2 High Dynamic Range Imaging

The dynamic range of a camera, i.e. the ratio between the brightest and the darkest luminance sample that can be captured in a single image, is for most cameras quite small (on the order of $10^2 - 10^3$). As the dynamic range of a scene can be much higher (e.g., about 10^6 between highlight and shadow regions), some techniques have to be used to capture the full dynamic range of a scene.

Several manufacturers have developed CMOS cameras that are capable of capturing a sufficiently large dynamic range by either combining multiple exposures or by the use of special imaging sensors. These cameras are typically video cameras and provide only a limited resolution. Furthermore, the measured values are quantized to 8–12 bits per pixel and color channel leading to a rather low precision.

In the computer graphics community, several authors proposed methods to extend the dynamic range of digital images by combining multiple images of the same scene that differ only in exposure time. Madden [34] assumes linear response of the imaging sensor and selects for each pixel an intensity value from the

brightest non-saturated image. Debevec and Malik [10] and Robertson et al. [41] recover the response curve of the imaging system and linearize the input data before combining them into a single high dynamic range image. In [13], Goesele et al. proposed a technique to combine high dynamic range imaging with color management techniques (see Section 3.3.3).

3.3.3 Color Issues

Accurately recording the continuous spectrum of the visible light is difficult – especially if the spectrum is not smooth but contains sharp peeks such as the spectrum of a discharge lamp or even a laser. Likewise, the spectral response curve that describes the way light is reflected by an object is not always smooth. Measurement devices such as a spectrophotometer perform therefore a very dense sampling of the spectrum and output large data sets.

In contrast to that, most analog and digital cameras record only three color values per pixel (tristimulus values). Each sensor in a digital camera integrates the amount of incoming light weighted by its response curve over the whole visible spectrum. This is inspired by the human visual system that also contains three types of sensors behaving in a similar way [19]. A camera can record the colors of objects as perceived by a human observer most accurately if the corresponding response curves are identical [33], but the true spectrum of the light hitting the sensor can never be reconstructed and different spectra can result in the same tristimulus values (metamerism). Color measurements done with a tristimulus device are therefore always an incomplete representation of the actual spectrum.

White Balance

The human visual system can adapt to a wide range of illumination conditions. Within this range, colored objects look roughly the same even if the spectrum of the light source changes and therefore the spectrum of the reflected light hitting the retina is different. A digital camera can mimic this behavior with a white balancing step: the tristimulus values are multiplied with constant factors so that the color of the light source is recorded as white. The influence of the light source on the recorded color of an object is hereby minimized.

Color Management Systems

For a digital camera, the recorded color of an object depends not only on the light source but also on several other factors including the properties of the optical system, the sensor, and the image processing steps applied by the camera itself or other software.

In order to relate the recorded color to well defined standards, color management systems have become a standard tool. An image of a well known test target such as the IT8.7/2 target (see Figure 3.2) is taken and processed in the same way all later images are processed. The relation between the color values of the test target patches and the color values reported by the camera is analyzed and used as calibration data. The International Color Consortium (ICC) introduced the so called ICC profiles [21, 50] as a standard way to store this information.

Figure 3.2: *IT8.7/2 target used to capture the color properties of an imaging system in order to generate an ICC profile.*

The basic mechanism behind ICC based color management systems is to use a well defined color space as profile connection space (PCS). All input data is converted into the PCS using an ICC input profile associated with the input device. Other profiles are used to convert data from the PCS into the color space of display or output devices such as monitors and printers.

One of the color spaces used as PCS is the linear CIEXYZ space [7]. In [13], Goesele et al. have shown that this color space can be used to generate color calibrated high dynamic range images which are a tool to improve the color fidelity of appearance acquisition methods.

4 Appearance Acquisition

The appearance of an object consists of several surface properties including color, texture, reflection properties, and normal directions or the local tangent frame in the case of anisotropic materials. Due to their large number they are difficult to acquire but nevertheless necessary to generate a convincing looking representation of an object. It is therefore justifiable to put a lot of effort into this acquisition step.

Traditionally the appearance of an object is captured using a variety of special devices [20]. But many surface properties can be acquired by the use of a photographic camera – preferably a digital camera – in a controlled lighting setup. Captured images can for example be used to color the 3D geometry model during rendering. The digital pictures are simply projected onto the model as image textures using texture mapping [15]. To ensure that each part of the object is colored, a sufficient number of images must be taken from different view points [37, 46]. During the projection a perspective correction must be performed to gain a seamless transition between textures of different images (see also Section 6). To obtain more precise surface properties than just a single color value, further processing is needed.

4.1 Reflection Properties

Constant, diffuse lighting during the acquisition phase would reproduce only the object's color. More realistic models can be obtained by considering further aspects of a material's appearance, for example the reflection properties. The intensity and color of any material typically varies if viewed from different directions or under different illumination (see Figure 4.1).

When light interacts with a perfectly reflective surface, i.e. a mirror, the reflected light leaves the surface at the same angle it hits the surface. However, perfect mirrors do not exist in reality. In contrast, most surface have a very complex micro-structure. This micro-structure makes different materials appear differently.

When light hits such a surface, it is not reflected toward a single direction, but

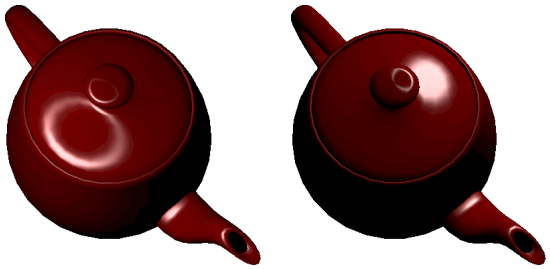

Figure 4.1: *A teapot with complex reflection properties illuminated from two different directions.*

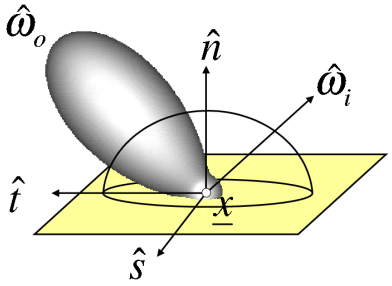

Figure 4.2: *Here you can see the values of a BRDF (depicted as a lobe) for one incident light direction $\hat{\omega}_i$ and every possible outgoing direction $\hat{\omega}_o$.*

rather to a cone of directions. If the surface is perfectly diffuse (e.g. for a piece of chalk), light even scatters equally in all directions.

In computer graphics the *bidirectional reflectance distribution function* (BRDF or also reflectance model) is used to describe the way a surface reflects light. The BRDF yields the fraction of light arriving at a point from one direction to the light that is reflected off the surface at the same point into an exitant direction.

Hence a BRDF is a four-dimensional function $f_r(\hat{\omega}_o, \hat{\omega}_i)$ that depends on the incident light direction $\hat{\omega}_i$ and the viewing direction $\hat{\omega}_o$ (see Figure 4.2). It should be noted, that it also depends on the wavelength, which is usually represented by three samples (RGB) only. In the following, the wavelength dependency is not stated explicitly.

A number of analytical BRDF models have been developed to approximate the reflection properties of real materials (e.g. [48, 51, 28, 2]).

4.2 Measuring Reflection Properties

In addition to these analytical models, it is possible to measure real-world BRDFs directly. There are special devices available to accomplish this task: The most general approach is to use a gonioreflectometer which measures the light that is emitted in every direction when the object is illuminated from a given direction. However, this measurement procedure can be very time consuming and captures only the properties of a single point on the surface of an object. If the surface is not uniform, this is not very helpful.

One way to overcome the ”single point” constraint for appearance measurements is the use of a digital camera. When an image is taken with such a camera it corresponds to millions of parallel measurements of radiance samples hitting the sensor. The main challenge is to recover the appearance information from images taken from different positions under controlled lighting conditions.

Marschner [35] used this approach to determine a single BRDF for an object by combining all the pixel data. Compared to a gonioreflectometer this technique is considerably faster, but it still assumes that the entire object consists of a single material, represented by a large number of tabulated BRDF samples. A specific BRDF model can be fitted to these BRDF samples by optimizing for the parameters of the BRDF model as it is for example done in [45]. The set of BRDF samples is then replaced by a few parameters resulting in a more compact representation.

To allow for variations of the reflectance properties over the object’s surface Marschner et al. [36] extracted the purely diffuse part (albedo map) of the object’s texture for each visible point using a similar technique. The resulting texture includes only view-independent color information and no specular reflection. Albedo maps plus one reflection model per surface patch have been acquired for indoor scenes by Yu et al. [53] which assumed that material properties only change from patch to patch.

An approach to acquire distinct reflection properties for every surface point has been published by Debevec et al. [11]. A set of images of an object, e.g. a person’s face, is taken from one viewpoint while the position of a point light source is changed. Hereby, the set of incident light directions is densely sampled. The collected data allows for realistic relighting of the object illuminated by arbitrary virtual environments. Unfortunately, a very large amount of data is needed both during the acquisition and for display.

4.3 Measuring Spatially Varying BRDFs

Based on Marschner's approach, Lensch et al. [29] developed a technique that is able to reconstruct spatially varying reflection properties by just a very few images (around 25). The key idea here is that most objects typically consist of a small number of materials only, i.e. many points on the object's surface have approximately the same reflection properties. By clustering points with different normals but consisting of the same materials, a large number of BRDF samples of that material can be collected by just a few images. After measuring the BRDF for clusters of points, separate reflection properties for each single point are determined to account for subtle details and small changes. The BRDF for each point is determined as a weighted sum of the clusters' BRDFs.

Thus, a high quality and very compact representation of the original object can be obtained with moderate acquisition effort.

4.3.1 Data Acquisition

The entire procedure is as follows: The geometry of the object is obtained by use of a 3D scanner, e.g. a structured light or computer tomography scanner, yielding a triangle mesh. In order to capture the reflection properties a small number of high dynamic range (HDR) images of the object are taken showing the object lit by a single point light source. In a next step the camera position (see Section 6) as well as the light source position relative to the geometric model are recovered for all images.

For every point on the object's surface all available data (geometric and photometric) is collected from the different views in a data structure called *lumitexel*. It contains the position of the surface point and its normal derived from the triangular mesh. Additionally, a lumitexel stores a list of radiance samples together with the corresponding viewing and lighting directions, one radiance sample for every HDR image where the point is visible and lit. The radiance sample is obtained by resampling the color value at the position of the surface point projected into the image.

4.3.2 Clustering of Materials

Because only a limited number of different views and lighting directions is acquired a single lumitexel does not carry enough information to reliably fit a BRDF model to the radiance samples. To provide more data from which the parameters can be derived, the lumitexels are grouped into clusters of similar materials. Starting with a single cluster containing all lumitexels, the parameters of an average

BRDF are fitted using the Levenberg-Marquardt algorithm to perform a non-linear least square optimization.

In order to separate the distinct materials the initial cluster has to be split. Given the average BRDF, two new sets of parameters are generated by varying the fitted parameters along the direction of maximum variance, yielding two slightly distinct BRDFs.

The lumitexels of the original cluster are then assigned to the nearest of these BRDFs, forming two new clusters. A stable separation of the materials in the clusters is obtained by repeatedly fitting BRDFs to the two clusters and redistributing the original lumitexels. Further splitting isolates the different materials until the number of clusters matches the number of materials of the object as illustrated in Figure 4.3.

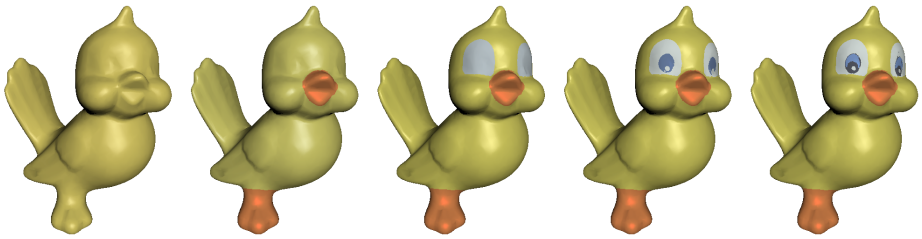

Figure 4.3: *The clustering process at work. In every image a new cluster was created.The object was reshaded using only the single BRDFs fitted to each cluster before the projection into a basis of multiple BRDFs.*

4.3.3 Spatially Varying Behavior

After the clustering the same reflection behavior is assigned to all lumitexels/points in one cluster. However, small features on the surface and smooth transition between adjacent materials can only be represented if every lumitexel is assigned its own BRDF.

In the algorithm, this BRDF is a weighted sum of the BRDFs recovered by the clustering procedure. The spatially varying reflection properties can be represented by a set of basis BRDFs for the entire model plus a set of weighting coefficients for each lumitexel.

The weighting coefficients are found by projecting the lumitexel's data into the basis of per cluster BRDFs. An optimal set of weighting coefficients minimizes the error between the measured radiance and the weighted sum of radiance values obtained by evaluating the basis BRDFs for the viewing and lighting direction of the measured sample. To recover the coefficients the least square solution of the

Figure 4.4: *Left: Last result of the clustering step. Right: Bird with the spatially varying BRDF determined by projecting each lumitexel into a basis of BRDFs. Note the subtle changes of the materials making the object look realistic.*

corresponding system of equations is computed using singular value decomposition (see [29] for more details).

In Figure 4.4 the result of projecting the collected data for every point into a basis of BRDF is shown. The method allows for accurately shaded, photorealistic rendering of complex solid objects from new viewpoints under arbitrary lighting conditions with relatively small acquisition effort. The reconstructed BRDFs can further be used to classify the objects based on their materials.

4.4 Normal Maps

The resolution of the acquired geometry of an object is typically limited by the used 3D scanning device (see Section 5). Additional processing of the 3D data like combining multiple scans, smoothing the surface to remove noise, and mesh simplification to reduce the complexity of the model further erases fine scale geometric detail.

When reconstructing the object using a coarse geometric model, smaller features in the surface's structure like bumps, cracks or wrinkles can be simulated by the use of normal maps or bump maps [5] (see Figure 7.1). These textures store a perturbation of the surface normal for each surface point. After applying the perturbation, the modified normals are used for the lighting calculations. This results in a change of the angle between the viewing direction and the surface at that point as well as between the light direction and the surface. This step approximates the correct lighting of a fine scale geometry model.

Normal maps recording small imperfections of the surface can be acquired for real world objects: Rushmeier et al. calculated normal directions from a set of images showing the same view of the object illuminated by a point light source placed at different but known positions for each image [44]. The surface is assumed to be perfectly diffuse (Lambertian), reflecting incident light equally in all directions, and thus its color can again be represented by an albedo map [43].

The restriction of a purely diffuse surfaces can be removed if techniques like [29] (see Section 4.3) are used to first measure the approximate reflection properties at each surface point and then use this data to measure the normal directions.

Since the BRDF at one point is defined for viewing and lighting directions with respect to the local tangent frame at that point, all directions have to be transformed based on the point's surface normal. To measure the exact normal at a point, an initial normal is obtained from the triangular mesh. Given the viewing and lighting directions for the radiance samples in world coordinates, the current estimate of the normal is used to transform them into the local coordinate frame. Then, the error between the measured radiance values and the reconstructed radiance values is computed where the reconstructed radiance values are obtained by evaluating the measured BRDF using the transformed directions. If enough radiance samples are provided for each point the actual normal direction at the point can be found by minimizing this error using a non-linear least square optimization technique. Figure 4.5 shows the quality of the reconstructed normals compared to the normals of the original mesh.

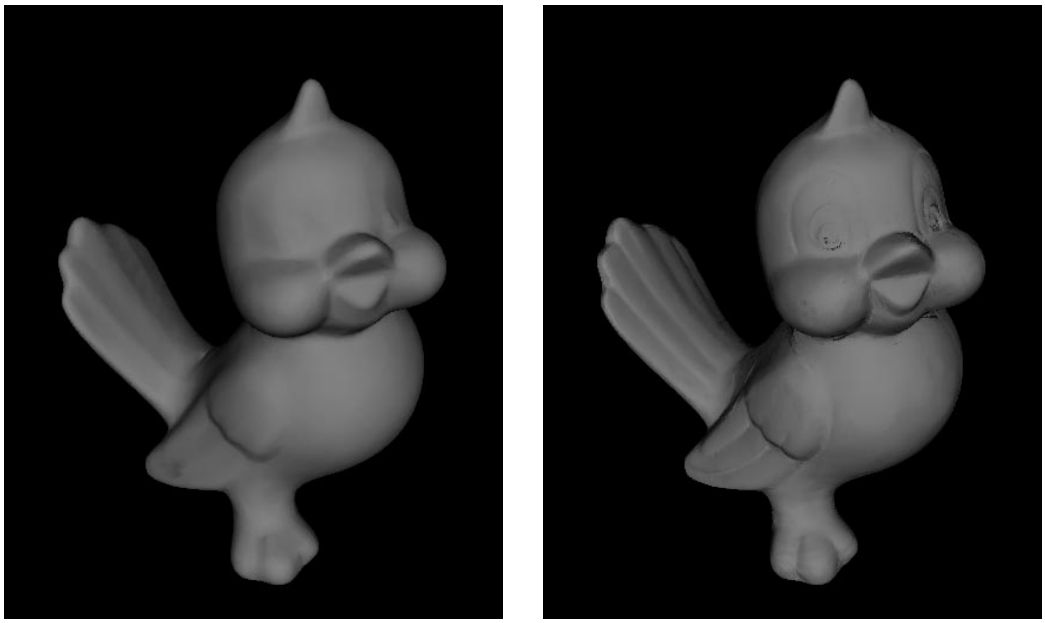

Figure 4.5: *Left: Normals of the original mesh. Right: Normals optimized using spatially varying BRDFs*

5 Acquisition of 3D Geometry

In most cases there exists no high quality 3D geometry model of real world objects like pieces of art. But even if it would exist (e.g. because the object was manufactured using computer based manufacturing methods) it is often only available to a very limited number of persons. Therefore, it is most often necessary to acquire the geometry of objects using a 3D scanner.

Several research groups including [32, 3] have built their own 3D scanner – some of them tailored to specific requirements. Furthermore, there is a broad range of commercial products made by companies like Cyberware, Minolta, or Steinbichler.

There are several different approaches to acquire the 3D geometry of an object (for an overview see [9]) but most of the systems for small or medium sized objects are based on an active stereo structured light approach. One or several patterns are projected onto the object with a computer controlled projection system (e.g. a video projector, a color coded flash stripe projector, or a laser beam). The projected light patterns on the object are observed by a digital camera which is rigidly connected to the projection system. The 3D location of a point on the surface of an object is then defined by the intersection of a ray from the projected pattern with the viewing ray that corresponds to the pixel in the digital image that observed this ray (see Figure 5.1).

The position of these rays in space is determined in a separate calibration step: The patterns are projected onto a calibration target – typically a flat board or a three-dimensional structure with a regular pattern whose geometric properties are exactly known. The acquired images are analyzed to recover the intrinsic parameters (e.g. focal length, lens distortion) and extrinsic parameters (the relative position and orientation) of the projection system and the camera using standard camera calibration techniques (e.g. [49, 54, 18]).

Using the active stereo approach most objects cannot be acquired with a single scan either because front and back part of the object cannot be scanned with a single scan or because for a given configuration not all parts of the object are visible from both the position of the projection system and the digital camera. Therefore

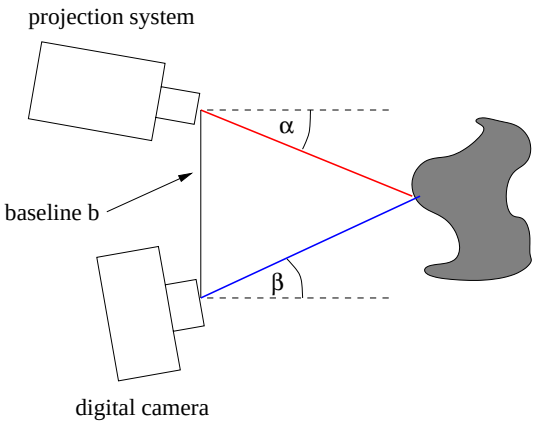

Figure 5.1: *Schematic drawing of an active stereo 3D scanner. Given the intrinsic parameters of the projection system and the camera, the baseline b and the angles α and β, the position of a surface point can be recovered using triangulation.*

several scans have to be registered against each other in order to combine them into a single set of surface points. This is commonly done using a variant of the iterative closest point method (ICP) [4, 40]. The resulting point cloud is triangulated leading to a single triangular mesh using one of a large variety of methods (for an overview see [9]). Further processing steps include smoothing to reduce noise (e.g. using [47, 25]) and editing of the resulting mesh for which a huge selection of tools is available including [26].

Kobbelt et al. [27] give a detailed description of the techniques used for the acquisition and processing of 3D geometry data.

6 Registration of Geometry and Texture Data

Since texture and geometry are typically acquired by two different processes the collected data has to be merged afterwards. This requires the alignment of the geometry data and the captured images. Only for scanning devices that capture geometry and texture data with the same sensor, the alignment or registration is already given. But in such a case the user is limited to the texture data provided by the scanner and the lighting setup cannot be changed to perform appearance measurements. Because of this, we further consider the case of two different sensors, a 3D scanner and a digital camera.

6.1 Manual Registration

In order to align or register the 3D model to the texture data one has to recover the parameters of the camera transformation that maps points in 3-space (the 3D geometry) onto the 2D image. These parameters describe the camera position, its orientation and the focal length (see Section 3.3.1). Further parameters are the aspect ratio, the principle point and the lens distortion, which are in the following assumed to be already known.

A simple approach to recover the camera position and orientation is to manually select corresponding points on the geometric model and in the picture [42]. If enough correspondences are established the transformation can be directly determined using one of various kinds of camera calibration methods (e.g [49, 54, 18]). But selecting corresponding points for a set of images is a time-consuming and tedious task. Additionally, the precision is limited by the user, although accuracy could be improved by selecting more points.

6.2 Automatic Registration

In order simplify the registration process some semi-automatic approaches have been published [37, 38]. The user is asked to roughly align the 3D model to the image. The algorithm then tries to optimize for the camera parameters by minimizing the distance between the outline of the 3D model rendered with the current set of camera parameters and the outline of the object found in the image. For each tested set of camera parameters the distance between the outlines has to be computed. This is a time-consuming step since the 3D model has to be rendered, its outline must be traced and for some points on it the minimum distance to the other outline must be computed.

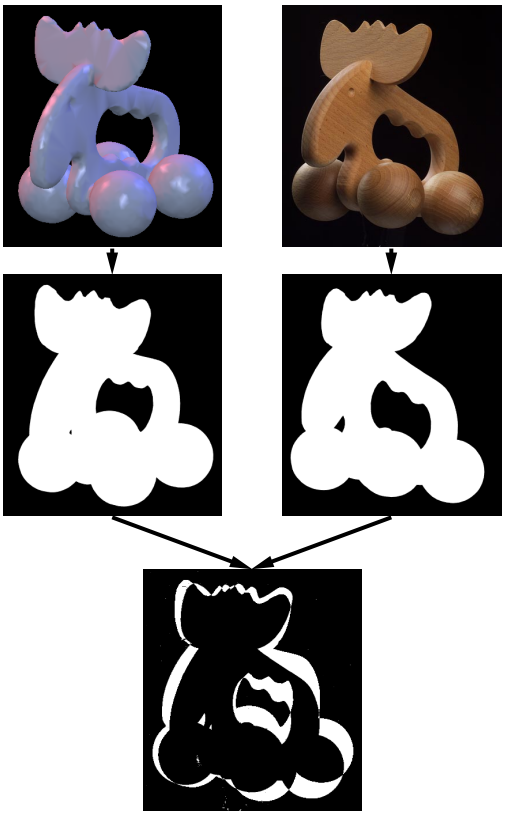

Figure 6.1: *Measuring the difference between photo (right) and one view of the model (left) by the area occupied by the XOR-ed foreground pixels.*

In [30], Lensch et al. proposed a method to compute the distance between a view of the 3D model and the 2D image in a different way. Here, silhouettes are compared directly instead of using their outlines. At first the silhouette of the object in the images is extracted by classification of the image in foreground

and background pixels, which can be done by any segmentation algorithm. Then, the geometry is rendered in front of a black background using a monochrome color. It is combined with the segmented image using the XOR-operation as is visualized in Figure 6.1. The resulting image will be black except for those pixels which are covered by just one silhouette but not by the other, that is to say exactly those pixels where the silhouettes differ. The number of remaining pixels is a measure for the distance between the silhouettes. These pixels can be counted by evaluating the histogram. The optimal set of camera parameters can be found by minimizing the number of remaining pixels.

Note that all three steps, rendering, combining, and histogram evaluation can be performed using graphics hardware and thus can be computed very fast, speeding up the optimization.

Additionally, it is also possible to automatically find a rough initial guess for the camera parameters. The effective focal length is first approximated by the focal length of the applied lens system. Depending on the focal length and the size of the object, the distance to the object can be approximated. It is assumed that the object is centered in the image. What remains to be estimated is the orientation of the camera. The optimization is simply started for a number of equally distributed sample orientation allowing just a few optimization steps per sample. The best result is then taken as a starting point for further optimization.

6.3 Texture Preparation

Knowing all camera parameters or the entire camera transformation for one image, it can be stitched onto the surface of the 3D model. The image is projected onto the the 3D model using projective texture mapping. Given a triangular mesh the stitching is done by computing texture coordinates for each vertex of the model that is visible in the image. Texture coordinates are calculated by projecting the 3D coordinates of the vertices into the image plane using the recovered camera transformation. All visible triangles can then be textured by the image as shown in Figure 6.2.

Further, the exact transformation for projecting surface points into the images is known. This information is required when collecting all radiance samples for one point on the objects surface into a lumitexel (compare Section 4.3.1).

A task that is still left is to determine the set of surface points for which a lumitexel should be generated. In order to obtain the highest quality with respect to the input images, the sampling density of the surface points must match that of the images. To achieve this, every triangle of the 3D model is projected into each image using the previously determined camera parameters. The area of the projected triangle is measured in pixels and the triangle is assigned to the image in

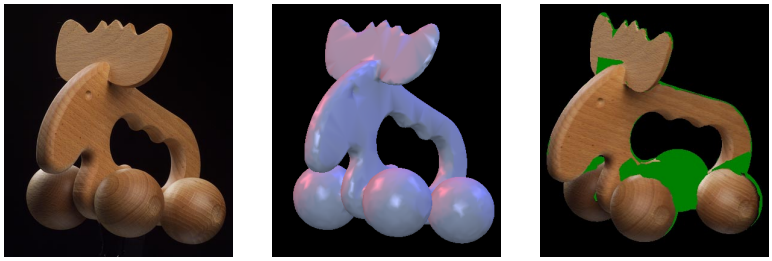

Figure 6.2: *The 3D model is aligned to a captured picture which then can be mapped as a texture onto the geometry.*

which its projected area is largest. For every pixel within the projected triangle a lumitexel is generated. The position of the surface point for the lumitexel is given by the intersection of the ray from the camera through the pixel with the mesh (see Figure 6.3).

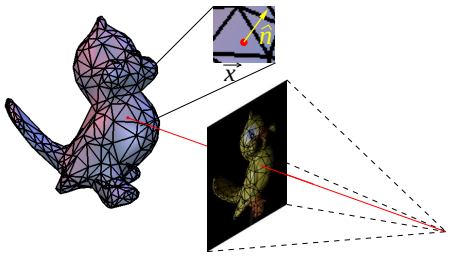

Figure 6.3: *The correspondence between pixel position and point position on the object is computed by tracing a ray through the image onto the object.*

Since every lumitexel is assigned to a triangular region within one of the HDR images it is possible to construct a 2D texture of lumitexels. This texture will unfortunately consist of a large number of separate triangles. Larger patches can be obtained by grouping adjacent triangles of the same input image. However, a significant number of isolated regions will remain. Instead of treating these regions as independent textures, it is more convenient to pack the regions into a single image, e.g. using the technique proposed by Rocchini et al. [42]. A result of this packing is shown in Figure 6.4 where the original color values of the input images are used to show the regions for which lumitexels are constructed.

During texture generation all parts of the original images where only the background is visible are discarded. Combined with dense packing of the remaining parts into one image, this reduces the size of the texture compared to the overall volume of the original images. A single image has the further advantage that it can be compressed and transformed into a streamable representation with less effort.

Figure 6.4: *Packing of the constructed texture regions for the elk model. Only three pictures were considered in this case to better visualize the layout .*

7 Interactive Display[1]

After measuring the reflection properties of the object and transforming the images into a single texture, we explain in this section how the combined data can be displayed interactively.

7.1 Rendering with Arbitrary BRDFs

At first we will investigate the case of one homogeneous material, i.e. one BRDF per object. Standard OpenGL only supports the empirical and physically implausible Phong model, which makes surfaces always look "plastic"-like.

In order to render surfaces with other BRDFs two similar approaches [17, 23] can be used. Both approaches decompose the four-dimensional BRDF $f_r(\hat{\omega}_o, \hat{\omega}_i)$ into a product of two two-dimensional functions $g(\hat{\omega}_o)$ and $h(\hat{\omega}_i)$. These two functions are stored in two texture maps and re-multiplied using blending. The approach by Heidrich and Seidel [17] decomposes the analytical Cook-Torrance model [8]. The approach by Kautz and McCool [23] numerically decomposes (almost) any BRDF by choosing a better parameterization for the BRDF.

Rendering is very simple. For every vertex of every polygon you have to compute $\hat{\omega}_o$ and $\hat{\omega}_i$ and use it as texture coordinates. Then the polygon has to be texture mapped with the textures containing $g(\hat{\omega}_o)$ and $h(\hat{\omega}_i)$ and the computed texture coordinates. Blending has to be set to modulate, so that $g(\hat{\omega}_o)$ and $h(\hat{\omega}_i)$ are multiplied together.

For an example of this technique, see Figure 4.1.

7.2 Rendering with Normal Maps

Blinn [5] has shown how wrinkled surfaces can be simulated by only perturbing the normal vector, without changing the underlying surface itself. The perturbed

[1] This section has been kindly provided by Jan Kautz.

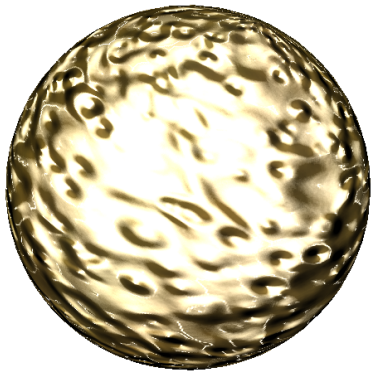

Figure 7.1: *A normal map applied to a sphere*

normal is then used for the lighting calculations instead of the original surface normal. This technique is generally called bump mapping.

A new algorithm has been proposed to render bump maps [24] (as shown in Figure 7.1) at interactive rates using texture maps containing per-pixel normals, which are used to perform the lighting calculations instead of per-vertex normals.

This algorithm relies on features now supported by many graphics cards. These features include per-pixel dot-products, multiplication, addition, subtraction, so lighting models/BRDFs using only these operations can be used to do bump mapping.

Usually the Blinn-Phong model [6] is used to perform bump mapping, because this model mainly uses dot-products. For more details, please see [24].

Heidrich et al. [16] also computed consistent illumination on bump maps in fractions of a second exploiting regular graphics hardware.

7.3 Spatially Varying BRDFs

One reflection model per surface can be evaluated very fast using the approach presented in [17, 23]. If the reflection properties vary across the surface spatially varying BRDFs must be considered which have been interactively rendered by Kautz et al. [22], see Figure 7.2.

Some of these algorithms take advantage of new features of current graphics hardware, e.g. multi texturing and texture combiners [39]. Although they are currently not available on all client machines they will become more and more widespread. In the future there should be a standardized way of transmitting and rendering more complex appearance models including color, BRDFs and bump maps.

Figure 7.2: *A spatially varying BRDF applied to a sphere*

8 Examples

In this section we describe some examples for high quality 3D object acquisition. Geometry and reflection data have been acquired for a bronze bust of Max Planck, a clay bird, and a painted models of two angles. Some statistics about the meshes and the number of acquired views are listed in Table 8.1.

The model of the angels was generated by extracting an isosurface of a computer tomography scan. The 3D geometry model of the bust and the bird were acquired using a Steinbichler Tricolite structured light 3D scanner. More than 20 scans per object were necessary to cover most of the surface. After a manual approximate alignment the scans were pairwise registered against each other. Finally, an optimization procedure reduced the global error. The resulting point clouds were triangulated to form triangle meshes.

Because a structured light scanner can only acquire surface points that are visible from the camera and projector position at the same time the bust mesh contained several holes – mainly around the ears. They were filled manually. Afterwards, a filtering step was applied to improve the smoothness of the meshes. In order to accelerate further processing the triangle count of the initial models was reduced by simplifying the meshes.

The images for the textures and reflection properties were taken with a Kodak DCS 560 professional digital camera, which outputs images consisting of 6 mil-

model	triangles	views	lumitexels	rad. samples	clusters	basis BRDFs
angels	47000	27	1606223	7.6	9	6
bird	14000	25	1917043	6.3	5	4
bust	50000	16	3627404	4.2	3	4

Table 8.1: *This table lists the number of triangles of each model, the number of views we used to reconstruct the spatially varying BRDFs, the number of acquired lumitexels and the average number of radiance samples per lumitexel, the number of partitioned material clusters, and the number of basis BRDFs per cluster.*

Figure 8.1: *A bronze bust rendered with a spatially varying BRDF, which was acquired with the presented reconstruction method.*

lion pixels. To acquire data for the entire surface several views with varying light source positions were captured per model (see Table 8.1). For each view around 15 photographs were necessary: two for recovering the light source position, one to extract the silhouette of the object for the 2D–3D registration, and the rest to provide the necessary high dynamic range.

The acquisition takes about 2.5h. The high dynamic range conversion, registration with the 3D model, and the resampling into lumitexels takes about 5h but is a completely automated task. The clustering and the final projection to recover the BRDFs takes about 1.5h[1].

Figure 4.3 shows how five successive split operations partition the lumitexels (the surface points) for the bird into its five basic materials. Only the per-cluster BRDFs determined by the clustering process are used for shading. Because of this the object looks rather flat. After performing the projection step every lumitexel is represented as a linear combination in a basis of four BRDFs, now resulting in a much more detailed and realistic appearance, see Figure 4.4.

The bust in Figure 8.1 shows another reconstructed object with very different reflection properties. The bronze look is very well captured.

A comparison between an object rendered with an acquired BRDF (using the presented method) and a photograph of the object is shown in Figure 8.2. They are very similar, but differences can be seen in highlights and in places where not enough radiance samples were captured. Capturing more samples will increase

[1] All timings were measured on a single processor SGI Octane 300 MHz.

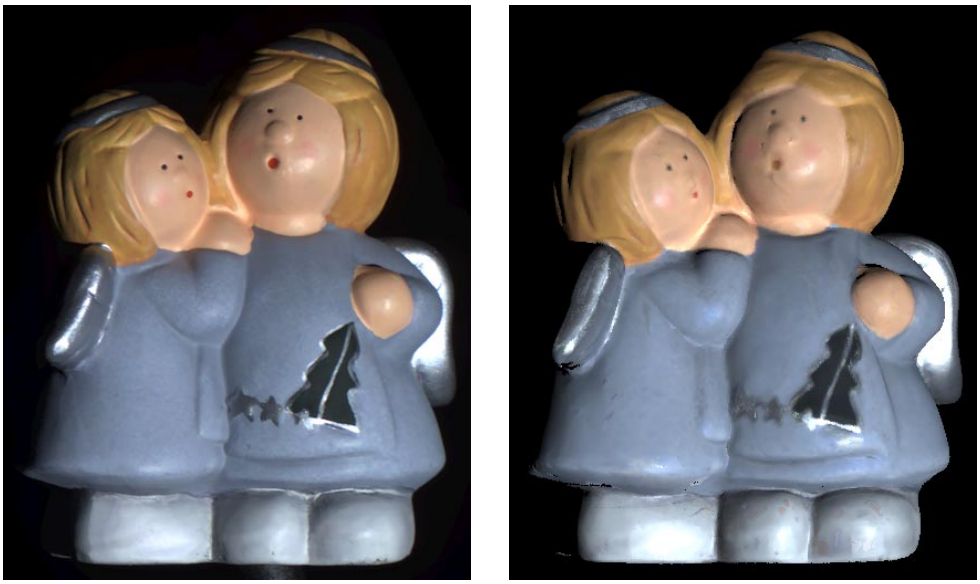

Figure 8.2: *Left side: Photograph of model. Right side: Model with acquired BRDF rendered from the same view with similar lighting direction. The difference in the hair region is due to missing detail in the triangle mesh.*

the quality. The difference in the hair region is due to missing detail in the triangle mesh. Those would be resolved by recovering the normal map for the object as described in Section 4.4.

Generally it can be said that for all the models only a few clusters were needed to accurately represent all the materials since the projection takes care of material changes. In our experiments even Lafortune BRDFs [28] consisting of a single lobe were sufficient to form good basis for the clustering and projection.

Further examples and movies of the acquired objects can be found at *http://www.mpi-sb.mpg.de/∼lensch/proj/BRDFMeasurement/BRDFMeasurement.html*.

9 Conclusion

We presented a framework for acquiring high quality 3D models of real world objects. The resulting models include both geometry and appearance information such as textures, normal maps or spatially varying BRDFs. Each of these is captured with a different setup. Afterwards all data is merged into a single model which is a fairly complete representation of the geometry and surface properties of a large class of real world objects. In order to achieve the highest possible quality, state-of-the-art computer vision and computer graphics techniques need to be combined in the acquisition and model generation stage of the framework.

Given such a detailed model, many computer vision algorithms such as the reconstruction of surface normals [44] or the detection of different materials can be improved or extended to other types of objects. Common assumptions about the characteristics of the object (e.g., pure diffuse reflection) are no longer necessary.

The demand for high quality 3D models will further increase in applications such as computer games, digital libraries and encyclopedias, or e-commerce applications. In order to satisfy these demands the presented methods need to be further improved with respect to acquisition speed, automation and quality. Currently, the class of materials that can be acquired and displayed are limited to isotropic materials. Future algorithms should also take effects like anisotropy and subsurface scattering into account.

10 Acknowledgments

We would like to thank Thomas Neumann, Kolja Kähler, Christian Rössl, Mario Botsch and the IMP Erlangen for their help in acquiring some of the presented data sets. Thanks also to Jan Kautz for providing Section 7, and to Philippe Bekaert for proofreading this paper.

Bibliography

[1] T. M. C. Abbott. In situ CCD testing. Available at http://www.cfht.hawaii.edu/˜tmca/cookbook/top.html, 1995.

[2] D. Banks. Illumination in Diverse Codimensions. In *Proceedings of SIGGRAPH 1994*, pages 327–334, July 1994.

[3] F. Bernardini, J. Mittleman, and H. Rushmeier. Case study: Scanning michelangelo's florentine pietà. In *Course Notes for SIGGRAPH 1999*, August 1999.

[4] P.J. Besl and N.D. McKay. A method for the registration of 3-d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 14(2):239–258, 1992.

[5] J. Blinn. Simulation of Wrinkled Surfaces. In *Proceedings of SIGGRAPH 1978*, pages 286–292, August 1978.

[6] J. Blinn. Models of Light Reflection For Computer Synthesized Pictures. In *Proceedings SIGGRAPH*, pages 192–198, July 1977.

[7] Colorimetry. Publication CIE No. 15.2, 1986.

[8] Robert L. Cook and Kenneth E. Torrance. A reflectance model for computer graphics. In *Computer Graphics (Proceedings of SIGGRAPH 81)*, pages 307–316, August 1981.

[9] Brian Curless and Steven Seitz. 3D Photography. In *Course Notes for SIGGRAPH 2000*, July 2000.

[10] P. Debevec and J. Malik. Recovering High Dynamic Range Radiance Maps from Photographs. In *Proceedings of SIGGRAPH 97*, pages 369–378, August 1997.

[11] Paul Debevec, Tim Hawkins, Chris Tchou, Haarm-Pieter Duiker, Westley Sarokin, and Mark Sagar. Acquiring the reflectance field of a human face. *Proceedings of SIGGRAPH 2000*, pages 145–156, July 2000.

[12] Michael Goesele, Wolfgang Heidrich, Hendrik P.A. Lensch, and Hans-Peter Seidel. Building a Photo Studio for Measurement Purposes. In *Proceedings of the 5th Conference on Vision, Modeling, and Visualization (VMV-00)*, November 2000.

[13] Michael Goesele, Wolfgang Heidrich, and Hans-Peter Seidel. Color calibrated high dynamic range imaging with ICC profiles. In *Proceedings of the 9th Color Imaging Conference*, Scottsdale, USA, 2001.

[14] Michael Goesele, Wolfgang Heidrich, and Hans-Peter Seidel. Entropy-based dark frame subtraction. In *Proceedings of PICS 2001: Image Processing, Image Quality, Image Capture, Systems Conference*, Montreal, Canada, April 2001. The Society for Imaging Science and Technology (IS&T).

[15] P. Haeberli and M. Segal. Texture Mapping As A Fundamental Drawing Primitive. In *Fourth Eurographics Workshop on Rendering*, pages 259–266, June 1993.

[16] W. Heidrich, K. Daubert, J. Kautz, and H.-P. Seidel. Illuminating micro geometry based on precomputed visibility. In *Proc. of SIGGRAPH 2000*, pages 455–464, July 2000.

[17] Wolfgang Heidrich and Hans-Peter Seidel. Realistic, hardware-accelerated shading and lighting. In *Proceedings of SIGGRAPH 99*, Computer Graphics Proceedings, Annual Conference Series, pages 171–178, August 1999.

[18] J. Heikkila and O. Silven. A Four-Step Camera Calibration Procedure With Implicit Image Correction. In *CVPR97*, 1997.

[19] Robert W. G. Hunt. *The reproduction of colour*. Fountain Press, 5. ed. edition, 1995.

[20] Richard S. Hunter and Richard W. Harold. *The measurement of appearance*. Wiley, 2. ed., 5. print. edition, 1987.

[21] Specification ICC.1:1998-09, File Format for Color Profiles. available from http://www.color.org, 1998.

[22] J.Kautz and H.-P. Seidel. Towards interactive bump mapping with anisotropic shift-variant brdfs. In *Proceedings of the Eurographics/SIGGRAPH Workshop on Graphics Hardware 2000*, pages 51–58, August 2000.

[23] J. Kautz and M. McCool. Interactive Rendering with Arbitrary BRDFs using Separable Approximations. In *10th Eurographics Rendering Workshop 1999*, pages 281–292, June 1999.

[24] M. Kilgard. *A Practical and Robust Bump-mapping Technique for Today's GPUs*. NVIDIA Corporation, April 2000. Available from http://www.nvidia.com.

[25] Leif Kobbelt. Discrete fairing. In *Proceedings of the Seventh IMA Conference on the Mathematics of Surfaces*, pages 101–131, 1996.

[26] Leif Kobbelt, Swen Campagna, Jens Vorsatz, and Hans-Peter Seidel. Interactive multi-resolution modeling on arbitrary meshes. *Proceedings of SIGGRAPH 98*, pages 105–114, July 1998.

[27] Leif P. Kobbelt, Stephan Bischoff, Mario Botsch, Kolja Kähler, Christian Rössl, Robert Schneider, and Jens Vorsatz. Geometric modeling based on polygonal meshes. Technical Report MPI-I-2000-4-002, Max-Planck-Institut für Informatik, July 2000.

[28] E. Lafortune, S.-C. Foo, K. Torrance, and D. Greenberg. Non-Linear Approximation of Reflectance Functions. In *Proceedings of SIGGRAPH 1997*, pages 117–126, August 1997.

[29] Hendrik Lensch, Jan Kautz, Michael Goesele, Wolfgang Heidrich, and Hans-Peter Seidel. Image-based reconstruction of spatially varying materials. In Steven Gortler and Karol Myszkowski, editors, *Proceedings of the 12th Eurographics Workshop on Rendering*, pages 104–115, London, Great Britain, 2001. Springer.

[30] Hendrik P. A. Lensch, Wolfgang Heidrich, and Hans-Peter Seidel. Automated texture registration and stitching for real world models. In *Pacific Graphics '00*, pages 317–326, October 2000.

[31] Marc Levoy and Pat Hanrahan. Light field rendering. In *Computer Graphics (SIGGRAPH '96 Proceedings)*, pages 31–42, August 1996.

[32] Marc Levoy, Kari Pulli, Brian Curless, Szymon Rusinkiewicz, David Koller, Lucas Pereira, Matt Ginzton, Sean Anderson, James Davis, Jeremy, Ginsberg, Jonathan Shade, and Duane Fulk. The digital michelangelo project: 3D scanning of large statues. In *Proceedings of SIGGRAPH 2000*, pages 131–144, July 2000.

[33] R. Luther. Aus dem Gebiet der Farbreizmetrik. *Zeitschrift fr technische Physik*, 8(12):540–558, 1927.

[34] Brian C. Madden. Extended Intensity Range Imaging. Technical report, University of Pennsylvania, GRASP Laboratory, 1993.

[35] S. R. Marschner. *Inverse rendering for computer graphics*. PhD thesis, Cornell University, 1998.

[36] S. R. Marschner, B. Guenter, and S. Raghupathy. Modeling and rendering for realistic facial animation. In *Eurographics Rendering Workshop 2000*, pages 231–242, June 2000.

[37] Kenji Matsushita and Toyohisa Kaneko. Efficient and handy texture mapping on 3d surfaces. *Computer Graphics Forum*, 18(3):349–358, September 1999.

[38] Peter J. Neugebauer and Konrad Klein. Texturing 3d models of real world objects from multiple unregistered photographic views. *Computer Graphics Forum*, 18(3):245–256, September 1999.

[39] NVIDIA Corporation. *NVIDIA OpenGL Extension Specifications*, October 1999. Available from http://www.nvidia.com.

[40] K. Pulli, T. Duchamp, H. Hoppe, J. McDonald, L. Shapiro, and W. Stuetzle. Robust meshes from multiple range maps. In *Proceedings of IEEE International Conference on Recent Advances in 3-D Digital Imaging and Modeling*, 1997.

[41] Mark A. Robertson, Sean Borman, and Robert L. Stevenson. Dynamic Range Improvement Through Multiple Exposures. In *Proceedings of the 1999 International Conference on Image Processing (ICIP-99)*, pages 159–163. IEEE, oct. 1999.

[42] C. Rocchini, P. Cignoni, and C. Montani. Multiple textures stitching and blending on 3D objects. In *Eurographics Rendering Workshop 1999*. Eurographics, June 1999.

[43] Holly Rushmeier, Fausto Bernardini, Joshua Mittleman, and Gabriel Taubin. Acquiring input for rendering at appropriate levels of detail: Digitizing a pietà. *Eurographics Rendering Workshop 1998*, pages 81–92, June 1998.

[44] Holly Rushmeier, Gabriel Taubin, and André Guéziec. Applying shape from lighting variation to bump map capture. *Eurographics Rendering Workshop 1997*, pages 35–44, June 1997.

[45] Hartmut Schirmacher, Wolfgang Heidrich, Martin Rubick, Detlef Schiron, and Hans-Peter Seidel. Image-based BRDF reconstruction. In Bernd Girod, Heinrich Niemann, and Hans-Peter Seidel, editors, *Proceedings of the 4th Conference on Vision, Modeling, and Visualization (VMV-99)*, pages 285–292, nov 1999.

[46] Wolfgang Stuerzlinger. Imaging all visible surfaces. In *Graphics Interface '99*, pages 115–122, June 1999.

[47] G. Taubin. A signal processing approach to fair surface design. *Proceedings of SIGGRAPH 1995*, pages 351–358, 1995.

[48] K. E. Torrance and E. M. Sparrow. Theory for off-specular reflection from roughened surfaces. *Journal of the Optical Society of America*, 57(9):1105–1114, September 1967.

[49] R. Tsai. A versatile camera calibration technique for high accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses. *IEEE Journal of Robotics and Automation*, 3(4), August 1987.

[50] Dawn Wallner. Building ICC profiles – the mechanics and engineering. available at http://www.color.org/iccprofiles.html.

[51] G. Ward. Measuring and modeling anisotropic reflection. In *Proceedings of SIGGRAPH 1992*, pages 265–272, July 1992.

[52] Don Williams and Peter D. Burns. Diagnostics for digital capture using MTF. In *Proceedings of PICS 2001: Image Processing, Image Quality, Image Capture, Systems Conference*, pages 227–232, Montreal, Canada, April 2001. The Society for Imaging Science and Technology (IS&T).

[53] Yizhou Yu, Paul Debevec, Jitendra Mali, and Tim Hawkins. Inverse global illumination: Recovering reflectance models of real scenes from photographs. *Proceedings of SIGGRAPH 99*, pages 215–224, August 1999.

[54] Zhengyou Zhang. A Flexible New Technique for Camera Calibration. Technical Report MSR-TR-98-71, Microsoft Research, 1999. Updated version of March 25, 1999.