Designing a pipeline for product designers using image based illumination and high dynamic range imagery to more effectively add computer generated product visualizations into live action photographs.

A Non-Thesis Project
Presented in Partial Fulfillment of the Requirements for
The Degree of Master of Fine Arts in the
Graduate School of The Ohio State University

By
Brock J. Stearn
The Ohio State University
2003

Master's Examination Committee:

Dr. Wayne Carlson, *Advisor*

Professor Maria Palazzi

Dr. Matthew Lewis

Acknowledgements

Family – my second star to the right in the night sky. I can never express the amount of appreciation I have for my family in their unwavering support of my work, interests and fascinations over the years. Even more important has been their simple lesson to follow my dreams, while they have done everything possible to support me in those quests. This incredible strength from them is the greatest asset I have.

Dr. Wayne Carlson – my advisor, who I greatly thank for his direction and advice in my work throughout my time at Ohio State. It was in a class of his, that I began my exploration into Debevec's work and through his guidance I've continued that exploration. Wayne continuously kept my focus and was always sure I was meeting all the goals of the project. His ongoing patience, understanding and willingness to allow me to pursue my interests unhindered have been a continued foundation throughout my production and writing.

Professor Maria Palazzi – my constant source of direction, ideas and constructive feedback over the years from Ringling to Ohio State and hopefully beyond. First I'd like to thank Maria for bringing me into ACCAD which has become a very important step in the ladder of my education and career. I'd like to thank her for her wisdom in her support and encouragement of exploring a new area in my thesis work. Her welcoming of my work and contributions to the Jane Project has allowed for my original thesis idea to develop and grow as well as to educate others for continuing my work throughout the duration of that project.

Dr. Matthew Lewis – my greatest resource in exploring the world of rendering and lighting. Without Matt's feedback, patience and subtle nudging in the right direction, my work in this thesis would never have come to a conclusion. His continuing interest and ideas in the various fields of exploration I've stumbled across in truly amazing and greatly appreciated.

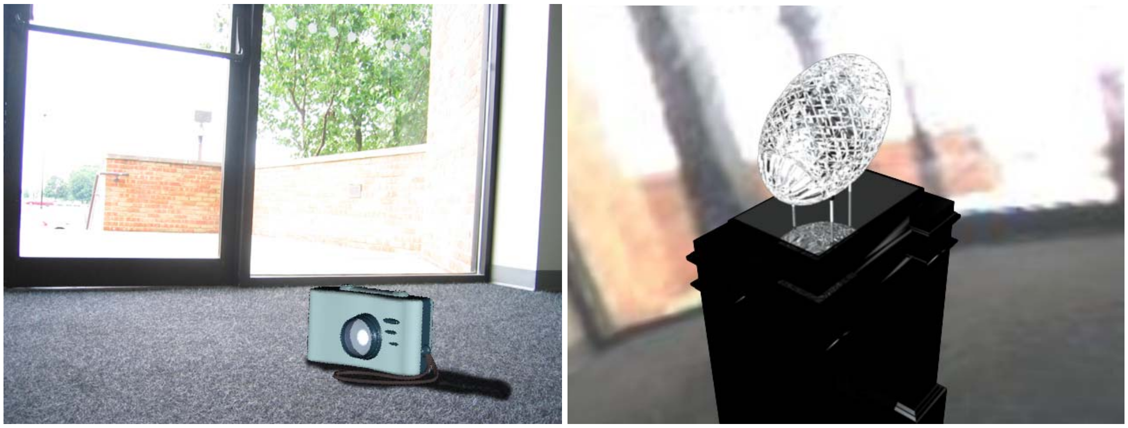
Dr. Paul Debevec – my inspiration for this project. Paul's work has been greatly circulated around the world and I'm thankful that I've been able to work with it. I truly hope that the work I've completed in this body of work can only add to the collection of knowledge that Debevec has started for all of us.

Table of Contents

Abstract

Product designer's computer-generated design visualizations are often composited into live action or "real images" in order to promote, display, or sell their product. Traditionally, this has been a long and painstaking process of matching the lighting and shadows from the live action image with the computer generated elements and then compositing both together for a final result. For a single designer or a group of designers, this must be completed within a tight budget and schedule. This presents a problem in developing necessary workflows to create believable renderings of a 3D product in a live action setting. In large design studios the product designer is usually not responsible for the promotional images for their work. For small studios and especially students, a product designer may be responsible for more duties then just designing the mock-up. These other duties would include compositing their product mock-up into a photograph from real life in order to promote that product. These small studios and students require a practical and very efficient solution to this problem of compositing their mock-ups into live action images (Figure 1).

Figure 1



Examples of the existing approach of compositing a cg mock-up into a live action background

In film, a similar process of combining computer generated elements with live action has become very successful through the use of Image Based Illumination and High Dynamic Range Imagery that has produced believable results, beginning with X-Men (Figure 2) and continuing up to current films, such as Matrix Reloaded (Figure 4). Already, there has been some exploration in using Image Based Illumination in types of architecture visualization (Figure 4) and web-based interior visualization (Figure 5). However, there has yet to be a direct connection using this technique in product design to composite computer generated mock-ups into live action photographs for designers.

Figure 2

Senator Kelly's death scene from X-men

Figure 3

Neo's fight scene (bottom) with 300 Agent Smiths from The Matrix Reloaded

Figure 4

Exterior architecture visualization using IBI and HDRI

Figure 5

Web-based visualization using HDRI

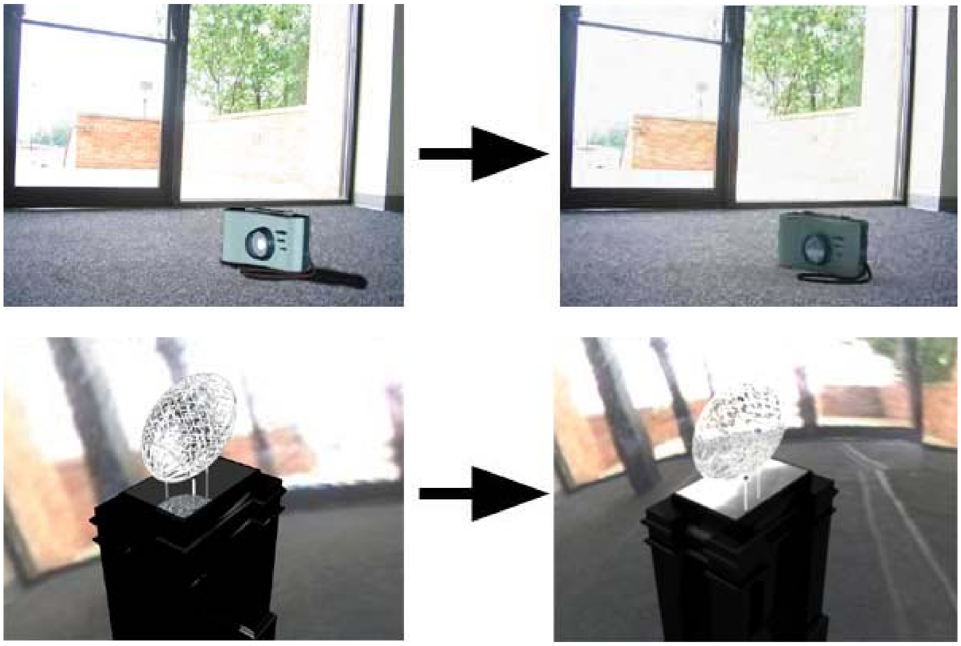
This paper presents a solution for the product designer's needs in achieving realistic lighting effects of 3D models by using information from real life imagery, in order to better facilitate the process of inserting the 3D model into a live action image. The intention is to address a specific pipeline for a product designer's need in combining CG product design mock-ups into photographs, using commercially or freely available software. The techniques presented here are based on several general concepts that could be modified for other similar 3D software and rendering applications.

Introduction

For over one-hundred fifty years people have been taking photographs of real objects or scenery. For even longer, artists and designers have been trying to create convincing imagery that looks real. Today, the mediums of computer-generated and live images are being merged together by combining real and non-real objects together seamlessly in photography and/or film.

A product designer's role has often been to create a "mock up" of a product design for a client. In a small studio or as a student, the product designer's role is often expanded to create imagery to promote, display or sell that product. Today, these mock ups are created digitally in a 3D modeling or animation software package. For promotional needs these mock ups are composited into real life imagery. The process of matching the lighting qualities of the computer generated product with the real-life imagery has often been long and painstaking. These conditions must be matched so that when the product is composited with the live action imagery for the final result, it looks natural or real (Figure 6).

Figure 6



A traditional approach (left) of compositing a separately lit product in with a photograph of a background compared to the result (right) of using this proposed pipeline

Recently a technique has been developed known as Image Based Illumination (IBI) or Image Based Lighting (IBL), which allows the use of real world lighting to illuminate a computer generated object that can then be composited more effectively back into the live action film[1]. An extension of this technique uses High Dynamic Range Imagery in combination with Image Based Illumination to provide even more accurate illumination of the computer generated object than standard Image Based Illumination would[2]. Both of these techniques make a direct connection to the lighting of the real-life environment into which the computer generated object will be added. This could be used to help product designers in their similar process.

[1] Kaufman, Debra. "X-Men Undergoes a CG Transformation." Computer Graphics World August (2000): 30.
[2] Debevec, Paul. *Image-Based Lighting.* SIGGRAPH 2001. 2001, http://www.debevec.org/IBL2001/

This paper will explore a connection in using Image Based Illumination in combination with High Dynamic Range Imagery for the promotional visualization of product designs. Due to the difficult nature of writing a custom renderer this proposed pipeline can fit into commercial renderers, such as Mental Ray™ or RenderMan™, both of which will be explored.

Problem Statement

Artists and designers are trained to use light and color to effectively render their work. Despite such training, it is still very difficult to match real world lighting well enough to properly blend a computer generated object with a live action image or sequence of images. We have seen through feature film that production studios are able to combine computer generated imagery into live action very effectively. Most of this work is done by a large team of people with custom software. However, the theories are applicable to other avenues of exploration. There simply needs to be a tangible pipeline for designers to use these available techniques.

The design goal for this project will focus on:

- How to capture real lighting information
- Using that lighting information to create images that can be mapped
- Rendering with IBI and HDRI

Hypothesis

Product designers, at any level, can use Image Based Illumination and High Dynamic Range Imagery to more effectively combine their product with live action imagery in order to promote, display, or sell their product.

Previous Work and Background

The technique of Image Based Illumination (IBI) is actually based on an extension of Reflection Mapping, a technique for simulating the appearance of a reflective object by using an image file for the reflection of the surface in a shader. This can be used to produce a shiny object reflecting a real environment corresponding to the sourced image (Figure 7). IBI thus far has mainly been used in film. IBI has been largely promoted with the use of High Dynamic Range Imagery (HDRI) by Paul Debevec of University of California, Berkley. Since 1997, Debevec has brought new advancements to this pipeline to each SIGGRAPH and related conferences. In 2000, X-Men became the first major theatrical film to use IBI and HDRI in production for Senator Kelly's death scene[3] (Figure 2). While there has been consistent development in these areas, most of this work has been done on proprietary systems or custom renderers. Recently, commercially available renderers have started to support IBI and HDRI[4]. These renderers, in combination with commercially available 3D software, allows for the possibility for any end user to use these techniques.

[3] Bunker, Simon. *RenderMania*. 2000, http://www.RenderMania.com/HDRI/
[4] PIXAR. *RenderMan – Artist Tools – News*. 2001, https://RenderMan.pixar.com/products/news/rat50.html

Figure 7

An early example by Gene Miller of Reflection Mapping

In the design field there has been minor work in using IBI in interior design (Figure 8), and architectural visualization[5] (Figure 4, Figure 9). This work was inspired by Paul Debevec's SIGGRAPH 98 paper *Rendering Synthetic Objects Into Real Scenes: Bridging Traditional and Image-Based Graphics With Global Illumination and High Dynamic Range Photography*. Most notably, work in the area of architecture visualization has been continued by Jaemin Lee and Ergun Akleman of Texas A&M University. Their work and pipeline has also been featured in the industry publication, *Architecture Week*[6]. Other work regarding visualizing computer generated architectural exteriors in real world environments using IBI has been done by Mehlika Inanici. These architecture visualizations are usually lit just using IBI as the visualization isn't dependent upon shadows and other lighting elements. Despite these explorations there has been no available documented research in using IBI for product designers.

[5] Lee, Jaemin, *Deriving Lights from Pixels*, Architecture Week. 2002, http://www.architectureweek.com/2003/0528/tools_1-1.html
[6] Lee, Jaemin, *Deriving Lights from Pixels*, Architecture Week. 2002, http://www.architectureweek.com/2003/0528/tools_1-1.html

Figure 8

Interior visualization using IBI techniques

Figure 9

Architecture visualization using IBI techniques

Given the lack of research in this area for product designers but the already existing information for using IBI and HDRI in film, it is the intent to append to the film pipeline with steps, procedures, and information that is relevant to product designers using commercially or freely available software for this proposed pipeline.

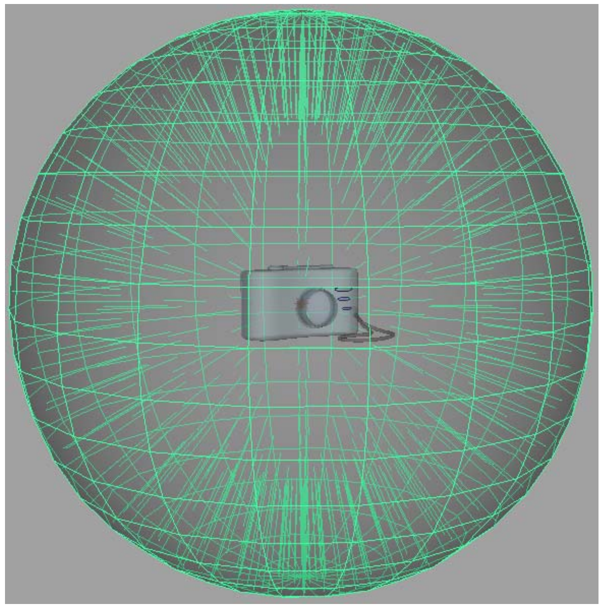
Research

Image Based Illumination

Image Based Illumination or Lighting is a relatively new process that has received a lot of attention lately due to the impressive work in this area by Paul Debevec of the University of California, Berkeley. Debevec refers to this technique as Image Based Lighting (IBL) while many others use Image Based Illumination (IBI). In this pipeline it will be referred to as Image Based Illumination or IBI. IBI allows for artists and designers to emulate through computer generated imagery objects that can be lit to be visually distinguishable from the real world images they are composited into.

IBI is the process of illuminating a computer generated object through the use of a single image file. This single image file is used in a renderer to illuminate surface points of a computer generated object. Different ways in which this may be used include surrounding an object with another piece of geometry (Figure 10) with the image texture mapped onto it, or determining where this surrounding object would be virtually through calculated IBI functions.

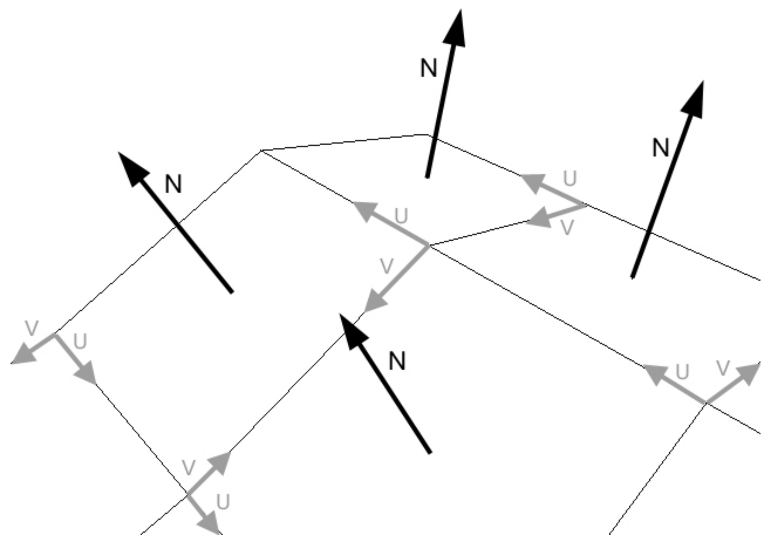
Figure 10



IBI sphere surrounding a cg design mock-up of a camera

The theory behind Image Based Illumination is that the normal vector of a surface point is extended until it intersects a piece of surrounding, whether actually present or virtually calculated, geometry. This surrounding geometry has an image mapped to it. A sampling of the color information of the image is done at this intersection. The color information from this pixel is then used to illuminate the object at the surface point. This process is done for every surface point with an IBI function attached to it. A normal vector is perpendicular to both the U and V vectors of the geometry (Figure 11) in a flat shader. The normal vector's direction can be changed depending upon additional shader functions, such as a bump or displacement.

Figure 11



Surface Point's Normal (N) calculated based on the U and V vectors of the geometry

In a renderer, IBI can be used with the technique of Global Illumination (GI). GI uses an algorithm that simulates the way light would react in real life. In this type of environment light travels in a straight line from a light until it hits a surface. From there, depending upon the surface, it can be bounced away, absorbed or a combination of both. At that point, as before, the light rays continue to travel in a straight line away, if bounced, until they intersect the next surface. This process is continuous, with each bounce causing the light ray to become weaker until eventually it is invisible. When these light rays are absorbed, some of the absorbed object's color is bounced away from the object (Figure 12). This technique of GI is ideal for creating photorealistic renders as it calculates lighting information from all the light sources. GI is a very computational technique and the renders are often very slow. Instead of calculating color information from all of the light sources, IBI uses the surface point's normal vector to determine the appropriate color from the surrounding geometry (Figure 13). This sampling from the IBI image on the IBI surface is far less computational then GI, and does not require light sources as GI does.

Figure 12

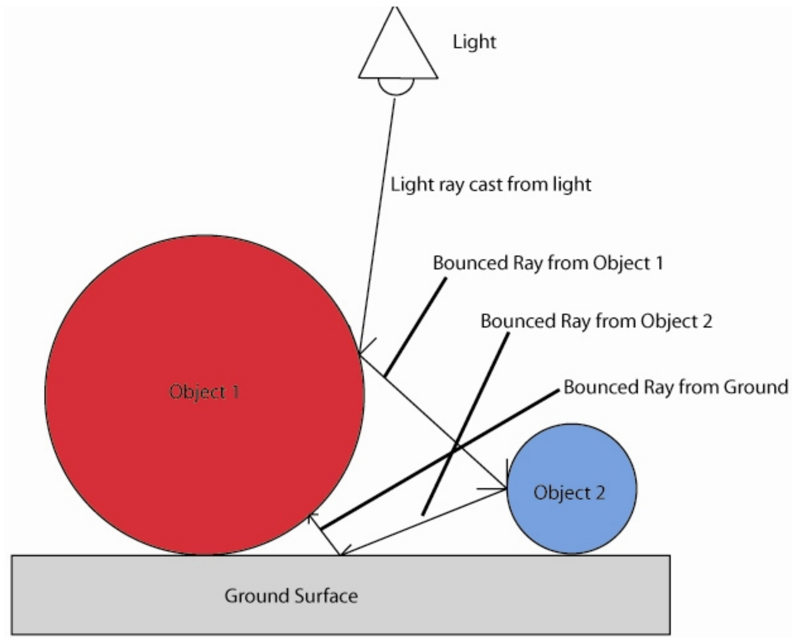
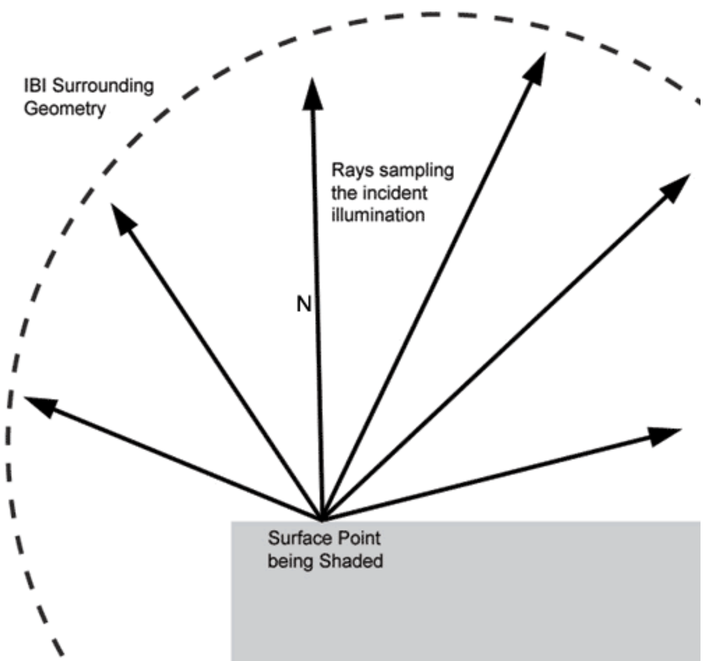


Diagram of how Global Illumination works

Figure 13



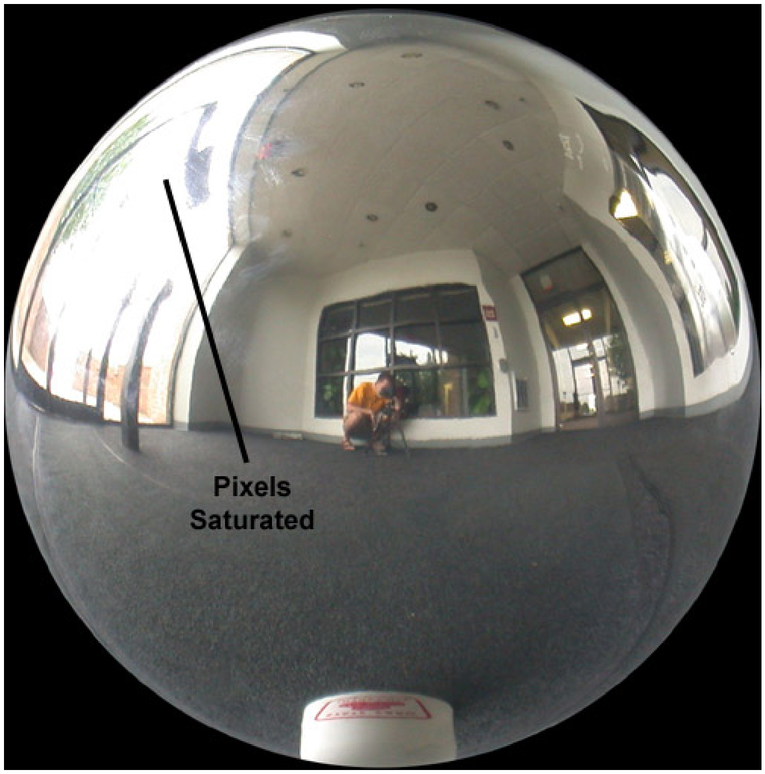
A diagram of Image Based Illumination being calculated at each surface point using scattered rays.

High Dynamic Range Imagery

High Dynamic Range Imagery (HDRI) is an image format created and developed by Paul Debevec and Greg Ward[7]. In order to understand High Dynamic Range Imagery, Low Dynamic Range Imagery (LDRI) must first be explained. LDRI is what most people are familiar with, whether they are aware of it or not. Photography and film are both examples of LDRI. In a sense, LDRI is an image that captures one set of lighting information. When a camera's trigger is pressed, the shutter opens for a specified period of time and captures that amount of lighting information. The resulting image is a LDR image. In a digital LDR image, when a scene is too bright the pixels of the image will saturate to their maximum values. Due to the non-linear encoding in digital images, the intensity and color is lost (Figure 14). As humans, our visual system adapts automatically to any changes in light so that we still only see one range at a time so the loss of color may not be apparent to our visual system.

[7] Debevec, Paul. *Paul Debevec Homepage*. 2003, http://www.debevec.org

Figure 14

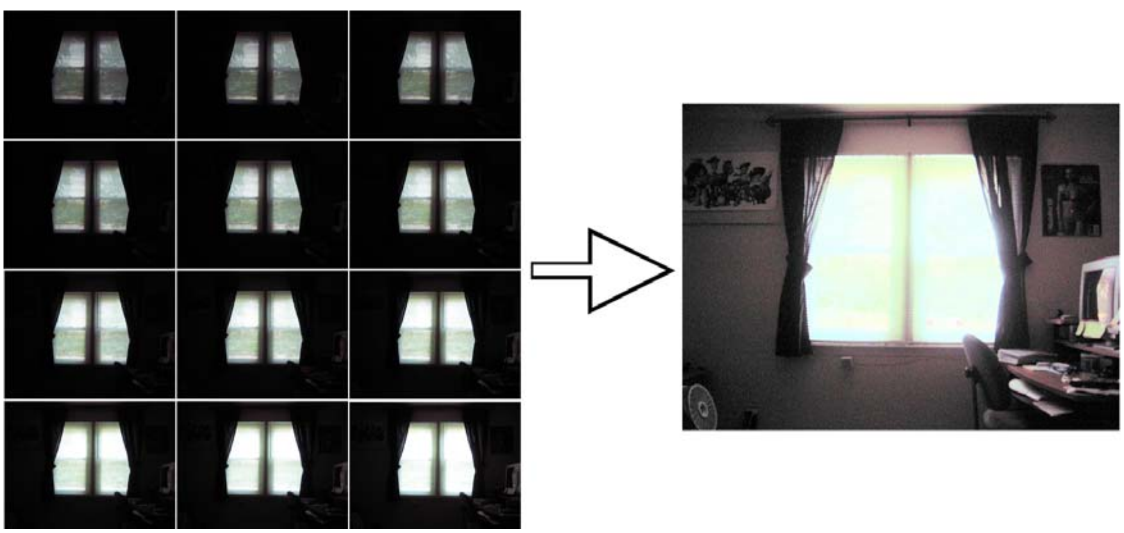


Demonstration of when a scene is too bright the pixels become saturated in the resulting captured image and intensity and color details are lost in a LDR image.

In contrast, HDRI has an important property in which the pixel values are proportional to the amount of light at any given point. Every amount of light is stored at every pixel, instead of the single range of on-screen colors that LDRI stores. Also, HDRI is defined by floating point numbers such as 0.597184 or 381121.0121 with a very high upper limit, instead of the more common 0 to 255 range in LDRI. This allows HDRI to display a very large range, or the ratio between dark and bright regions, of lighting information in one single image.

Since there is no way to directly capture an HDR image, the image must be complied together from other sources. The principle is to take a full range of exposure information through a series of LDR images. A few of these images are used to generate a camera response curve since each camera is different. Along with the camera response curve and the specified exposure difference, an HDR image can be assembled from the range of LDR images taken at different exposures (Figure 15). By using different exposures it is possible to capture image information that would normally be lost in a single exposure. In the figure example in a low exposure setting it is possible to see outside the window, but the inside of the room is dark. On the higher exposure settings the window over exposes are it is not possible to recognize objects outside the window, however the room becomes more visible. This is then written to a single file with all the HDR information. There are a few file formats that can store the High Dynamic Range data completely. Some of these include a floating point version of the TIFF format, an HDR format, the Portable Format variant of Jeff Postsanzer's Portable Pixmap format[8], the recently open-sourced EXR format from Industrial Light and Magic (ILM) [9]and in the half datatype in the NVIDIA based Cg language[10].

Figure 15



An exposure series of LDR images combined to an HDR image

[8] Debevec, Paul. *HDRShop Introduction*. 2001, http://www.debevec.org/HDRShop/main-pages/intro.html
[9] Industrial Light and Magic. *OpenEXR*. 2003, http://www.openexr.net
[10] Debevec, Paul. *Image Based Lighting – SIGGRAPH 2003*. 2003, http://www.debevec.org/IBL2003/

Capturing Images

As mentioned, there is currently no way to capture an HDR image in one shot. This must be done through compiling an exposure range of LDR images together. This is usually done using a digital camera, as there is more control and less error than a traditional camera. Exposure ranges can be done in three common ways. The first is to use a series of Neutral Density (ND) filters (Figure 16), stacking an additional one on top of the other for each additional exposure[11]. The second way is to adjust the F-Stops to cover the complete exposure range[12]. The third is to use a range of shutter speed settings to cover the exposure range[13]. There are various pros and cons to each of these options. Not all digital cameras allow filters to be attached to the lens. F-Stops are commonly known in traditional photography and are emulated in digital cameras. However, every digital camera has a different range of F-Stops so it is very difficult to set a consistent pipeline if every camera is different. The last method, using different shutter speeds, is the easiest and cheapest way to shoot a range of LDR images (Figure 17), as even low cost digital cameras allow control of the shutter speed.. A typical range should be from 9 to 13 images. Odd numbers work out best as there can be a starting exposure and then a step an equal amount each direction in the speed setting..

[11] Debevec, Paul. *Image-Based Lighting. SIGGRAPH 2001*. 2001, http://www.debevec.org/IBL2001/
[12] Debevec. *Image-Based Lighting. SIGGRAPH 2001*
[13] Debevec. *Image-Based Lighting. SIGGRAPH 2001*

Figure 16

Neutral Density Filters

Figure 17

Shutter speeds example

Ideally, we want an HDR image to capture as much as possible from a real life environment. A common approach to this is to shoot a sequence of images in different directions from one spot then "stitch" them together to create one whole image[14]. However, since we need to shoot between nine and thirteen images for each single shot, this can be a very time consuming process. Also for outdoor environments, because light cannot be controlled in the time it would take to shoot a whole sequence of LDR images to assemble one HDR image the range would be inconsistent as the light would continuously be changing over this time. Other processes include using fisheye lenses or special panoramic cameras, such as those from Panoscan or Sphereon that use a rotating camera head to scan the full 360°.

Today, the most common approach for capturing these images is a very efficient and cost effective process which uses a mirrored or reflective sphere[15]. The reflective sphere is placed in the desired environment and the camera is used to photograph that sphere.

[14] Downing, Greg. *Stitched HDRI*. 2001, http://www.gregdowning.com/HDRI/stitched/
[15] Debevec. *Image-Based Lighting. SIGGRAPH 2001*

Given how the light rays refract around the reflective sphere, in combination with the refractive and tangent angles, it is possible to capture more than the assumed 180° of information (Figure 18).

Figure 18

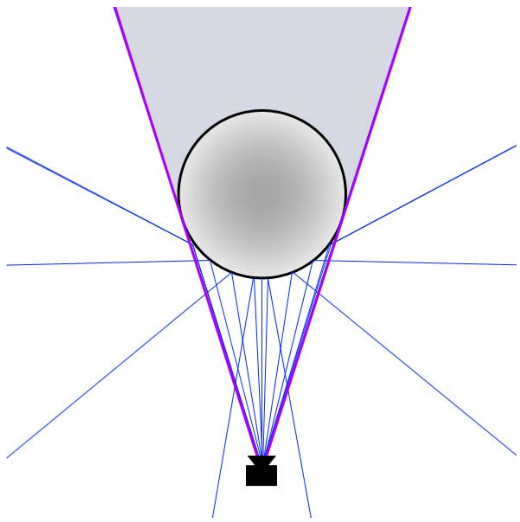


Image showing how from a camera it is possible to capture more then 180° of information.

Depending on the distance from the reflective sphere, it is possible to capture close to 310° of information. In this scenario, the camera can be mounted on a tripod in order to maintain stability. The reflective sphere should be mounted similarly. Using a reflective sphere, the image characteristics of that sphere are omnidirectional, or "for every direction in the world there is a pixel in the image that corresponds to that direction" (Debevec) and "their pixel values are linearly proportional to the amount of light in the real world" (Debevec)[16]. Once these images are gathered and compiled into an HDR image, the result is what is referred to by Paul Debevec as a lightprobe. A lightprobe is an angular map of the image. If a reflective sphere is used to gather the images, there will be a direct correlation to the lightprobe since both have a spherical base. Even though the

[16] Debevec. *Image-Based Lighting. SIGGRAPH 2001*

image looks distorted in the lightprobe, all important environmental information is still there (Figure 19).

Figure 19

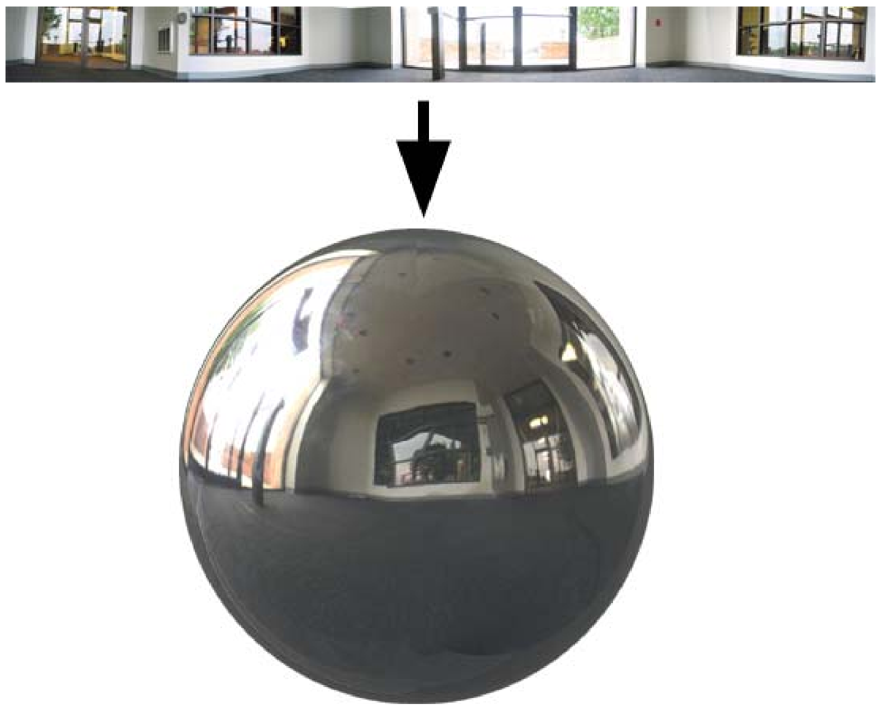
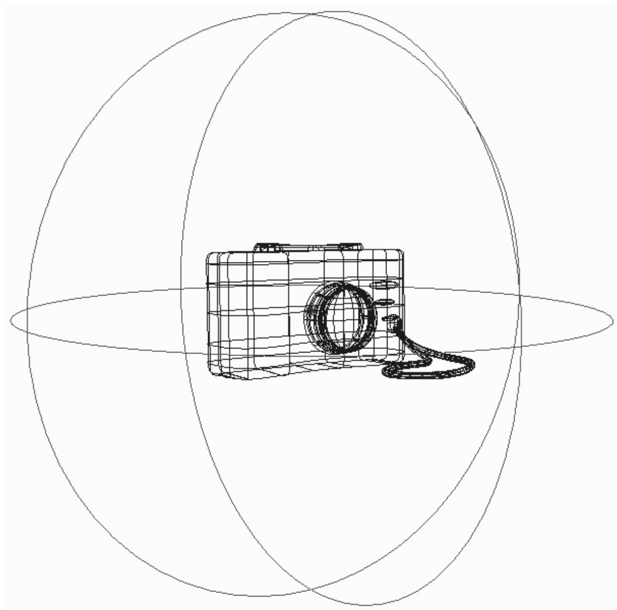


Image of the ACCAD lobby and the resulting lightprobe

Rendering with IBI and HDRI

The basic concept of using IBI is to wrap the computer generated object being illuminated with a surface that has the desired illumination image texture mapped onto it. Since each renderer is different and may not have its own integrated IBI function, there are some basic workflows for emulating IBI. The first is to create a light, such as a solar light or a pseudo area light (Figure 20), that is directional independent and using the illumination image to determine the color. Usually these lights illuminate in a spherical fashion, so it is important to make sure the illumination image is in the identical mapping format. This technique of IBI is easy to implement but since any given light shader will only produce a minimal amount of light, it is difficult to get a large range of detail through this technique.

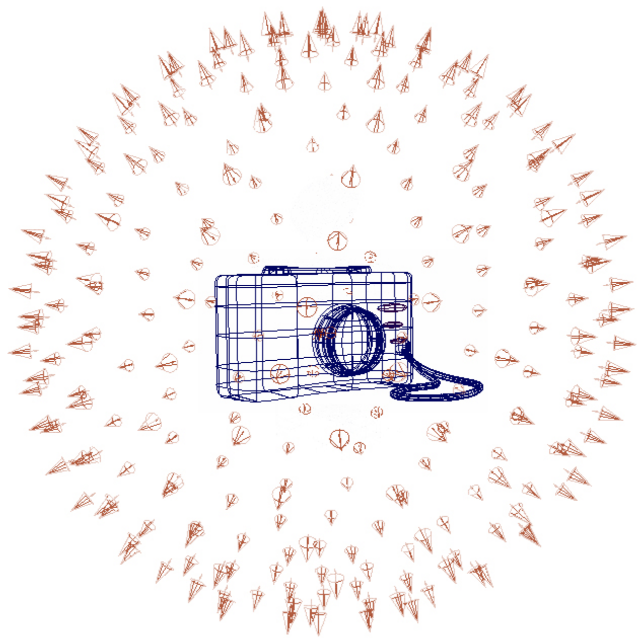
Figure 20



Pseudo area light in Maya

A second technique of emulating IBI is to create a sphere of lights, with each light being fed a specific color from the illumination image (Figure 21). This method follows the same idea as wrapping a sphere around the computer generated object and feeding that sphere the spherical illumination map for the color. However, instead of using a sphere for lighting, since not all renderers may support an IBI function, a function is called that creates a specified amount of lights evenly placed apart in place of the sphere. The function then breaks the illumination image up accordingly and passes each chunk of that image to the appropriate light. This method also gives the effect of IBI, but in a large scene the designer must remember to make sure the lights are far enough away to cover the entire scene, which can be difficult in a series of animated frames. Another disadvantage to this technique is that each additional light increases the render time. The more lights present the higher the quality of the image, so the renders can take a very long time for a high quality image.

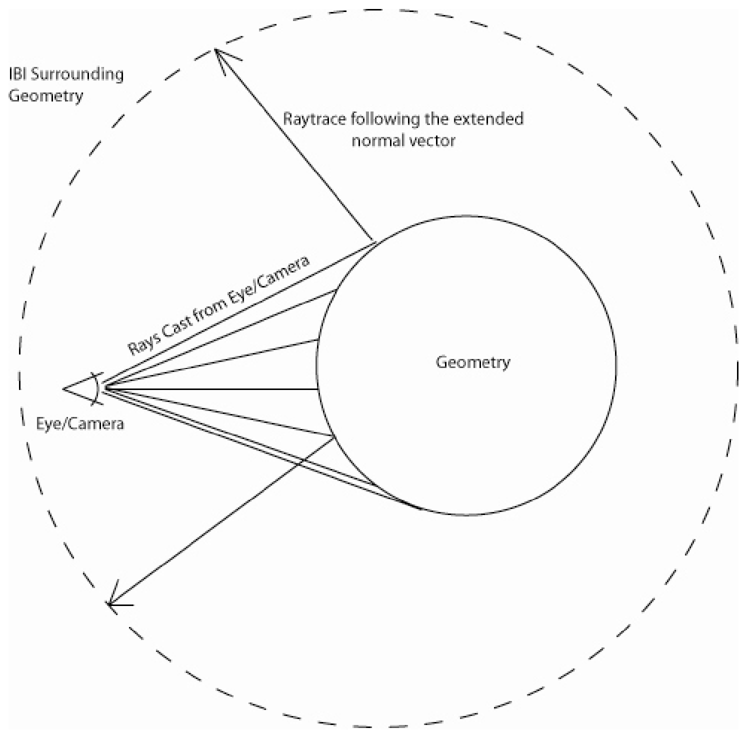
Figure 21



A sphere of lights with each light being fed a part of the IBI

If the renderer supports raytracing, it is possible to create a custom IBI function using raytrace functions. Raytracing is designed to follow the ray of light from the camera or eye to an object, and then to another object, therefore it makes an ideal method of implementing IBI (Figure 22). With the corresponding illumination map, it is possible to use a raytrace call to follow the extended vector of the normal of a point on the object to the surrounding object and then to bounce back with the inherited color of that point on the surrounding object. This technique works well but if there are other objects in the way that can interfere with the trace function in just capturing IBI information. Also, raytracing itself can often take longer then a typical radiosity renderer, since once a radiosity render has been completed that information can be stored.

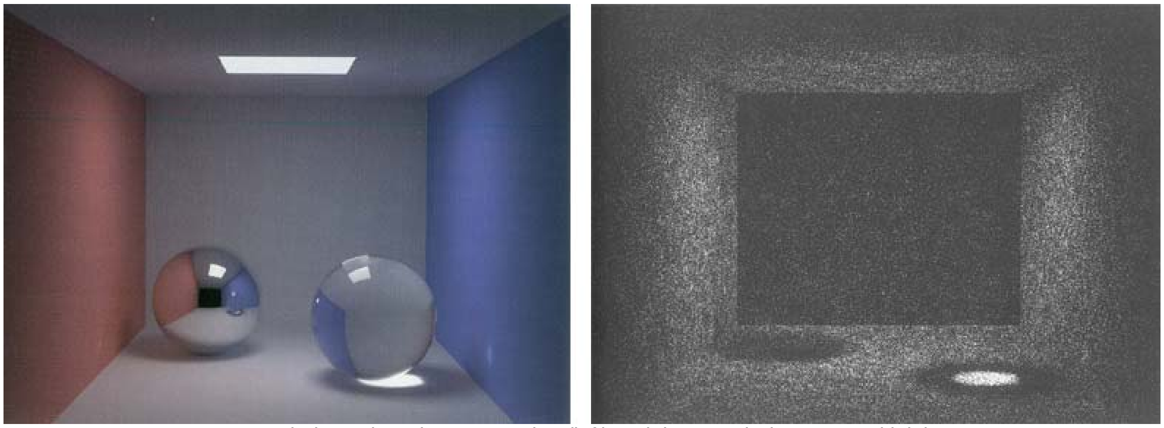
Figure 22



An illustrated example of IBI Raytracing

Each renderer may also have its own integrated IBI function or technique. In Mental Ray for Maya, IBI is achieved by creating a physical piece of surrounding geometry and enabling final gathering. With final gathering, the rays are first cast from the surrounding geometry. When the computer generated object is struck by these rays, a series of rays are diverted at random angles to calculate the light energy from the surrounding geometry. All of this information is stored into what is called a photon map (Figure 23). The photon map is then used to add the effect of the bounced light in the raytracing process. When using RenderMan, the application SLIM already contains an existing IBI template which can be applied to geometry. In RenderMan this information is not saved to a file and cannot be cached as it can in Mental Ray for Maya. Both of these methods allow for the addition of lighting information to tweak the render as well.

Figure 23

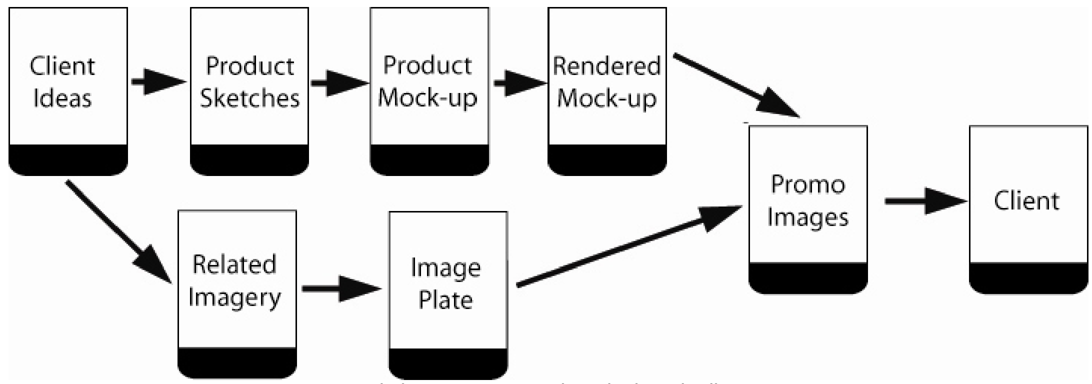


A rendering using Photon Mapping (left) and the actual Photon Map (right)

Development

The Existing Pipeline

Figure 24

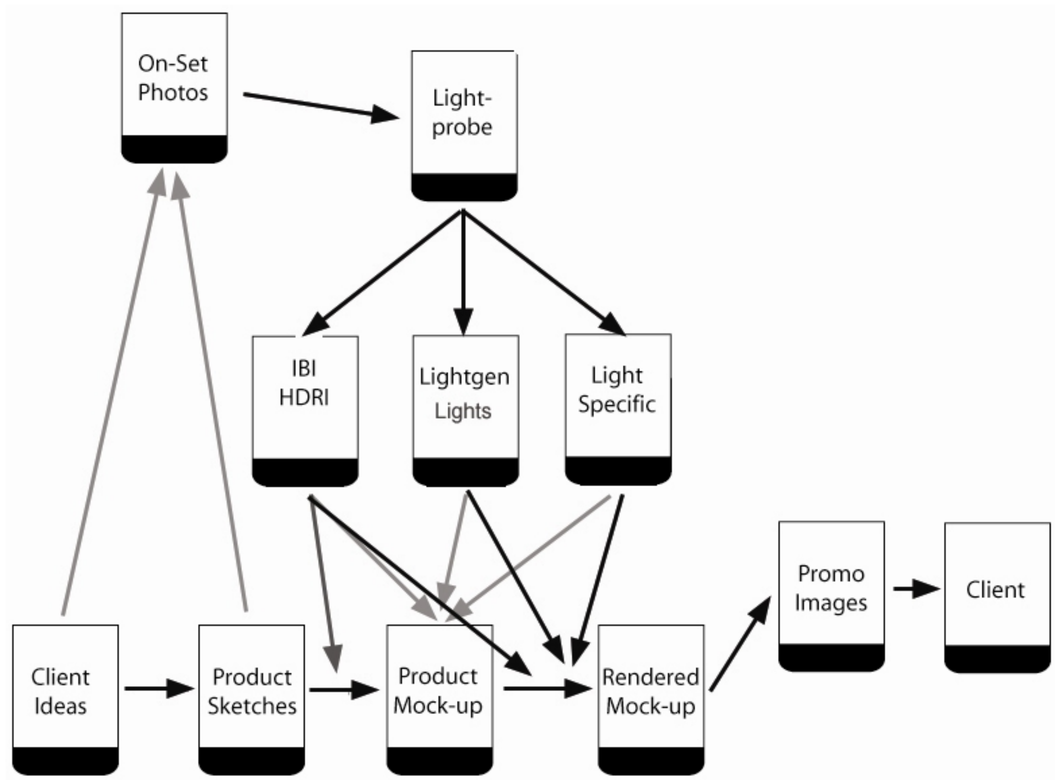


Existing common product design pipeline

The pipeline used by small design studios and design students (Figure 24) is very effective for the role of designing the product but ineffective for the latter part of compositing the product into the live action imagery for promotion due to lack of training in lighting and/or not knowing the actual environment the design will be composited into earlier in the process. Again, in a large studio there are many people working on many different aspects of a product's design and promotion. Larger studios usually have skilled people whose only responsibility is compositing and lighting. However, for those attempting to achieve an accurate environmental lighting result without having dedicated compositers and lighters, there is difficulty in creating convincing images of that product composited into live action.

Proposed Pipeline

Figure 25



Proposed pipeline for product designers using IBI and HDRI

This proposed pipeline (Figure 25) is ideal for those small firm designers or students who must complete several aspects of the design pipeline themselves. At the beginning of the proposed pipeline there is more interaction between the client and the designer in that they must together find a suitable environment to later use to composite the product mock-up into that environment. This ideally would occur between brainstorming and initial design sketches as at this point the final details of the product are usually being solidified. Also taken into consideration are any specific utilitarian uses of the product that would require certain background environments. Next there is a fork in the pipeline where the sketches continue to go through the process of becoming a 3D mock-up while the environment image goes through several stages to produce lighting information.

Throughout these stages certain lighting elements from the environment image are passed back to the product design pipeline. This allows for the lighting to gradually develop which can allow the designer to see initial lighting of their object very quickly. Once all of the areas of lighting information are combined with the model it is then rendered and can be composited into the live action image. Depending on the camera angle of the image, shadows and other specific elements from the mock-up should be rendered separately and then composited into the live action image separately.

In this proposed pipeline there are again five basic areas of importance. These areas are:

- “On set” image capturing
- Creating and assembling a useable HDR image
- Gathering any other lighting information from the HDR image
- Rendering with the IBI and HDRI information
- Compositing everything together.

These steps are based on the already discussed research but have been modified or appended specifically for the pipeline a designer may use. Of course each step also has its own stages which are applicable for this proposed pipeline.

Figure 26

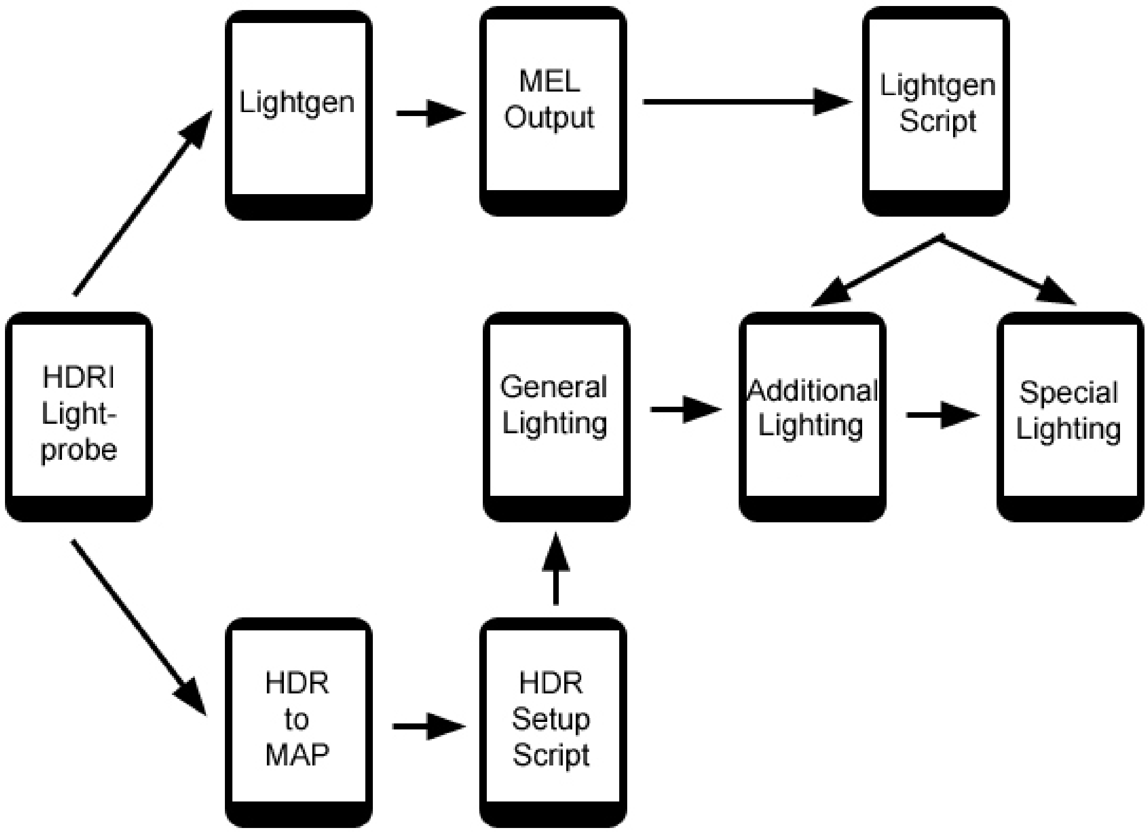


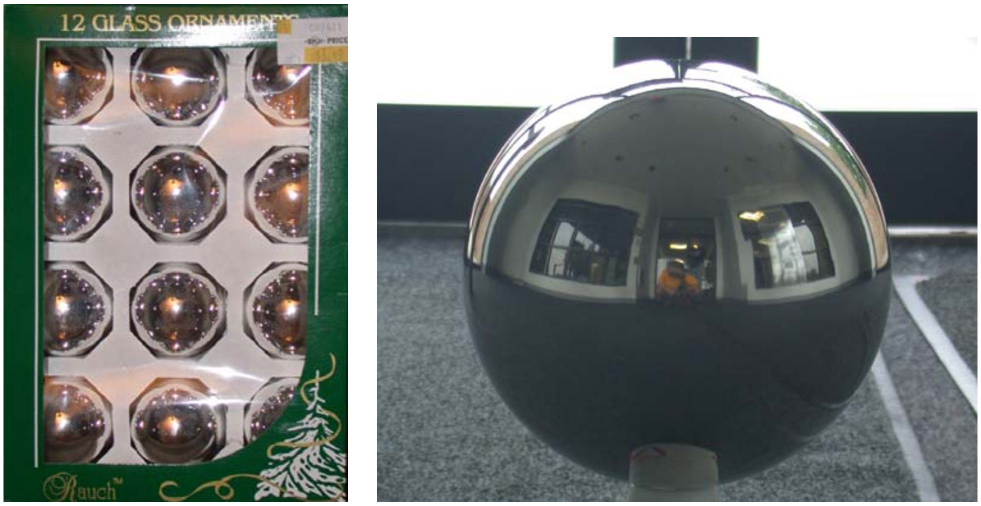
Diagram of the stages involved in the setup for rendering using IBI and HDRI

These additional stages are using IBI and HDRI to generate general lighting, using the HDRI to derive actual Maya lights and then scene specific lights (). There are two scripts designed for this pipeline that allow for these specific stages. These two scripts are *Brock's HDRSetup Control* and *Brock's Lightgen Control* and along with the accompanied tutorial a designer can follow this pipeline.

“On Set” Image capturing setup

In this pipeline, as with any other IBI pipeline the very first step is gathering environmental information into photographs. These images will be used as reference for creating the illumination in the 3D software and renderer. The most efficient way to capture an exposure range is by controlling the shutter speed. However, the efficiency in capturing the entire environment has not yet been discussed in detail. The main method found in IBI research is photographing a mirrored or reflective sphere to create the lightprobe. Traditionally, the reflective sphere used is a gazing ball (Figure 27). True gazing balls can be very expensive, but there are other options which small studios or students can easily afford. One option is to use silver Christmas ornaments which are available for a few dollars and provide good image quality (Figure 27). An extra advantage is that Christmas ornaments come in packs so that there are extra ornaments in case any break. Another option is that there are also generic gazing balls which are not blown, as true gazing balls are, but are molded together. Since in rendering IBI is blurred just a bit before illuminating, minor defects in the reflective sphere such as molding are not a problem. The main reason for using a gazing ball versus a Christmas ornament is the size of the actual environment. A larger sphere can capture more information on camera then a smaller one for large environments. However, for smaller indoor environments such as indoor rooms a Christmas ornament will work just fine.

Figure 27



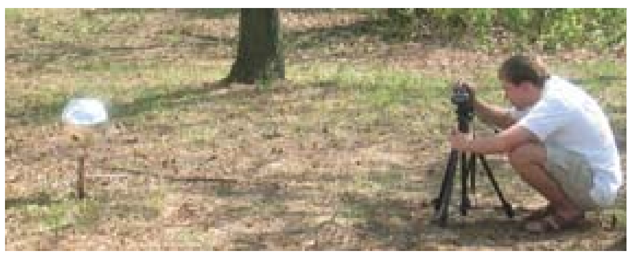
Christmas ornaments (left) and a gazing ball (right)

For each size of reflective sphere the shooting setup must be adjusted accordingly. The formula for how far away the camera should be is dependent upon the strength of the zoom lens on the camera. A rule of thumb established here in this pipeline is that the tripod should be four times the distance back of the diameter of the reflective sphere. Thus with 2 inch sphere, such as a Christmas ornament, the tripod should be approximately 8 inches away (Figure 28). With a 10 inch sphere, such as a gazing ball, the tripod should be 40 inches away or 3-4 feet (Figure 29). Ideally, the reflective sphere should be placed where the CG elements will be composited into (Figure 30).

Figure 28

Setup using a Christmas ornament in a small room

Figure 29



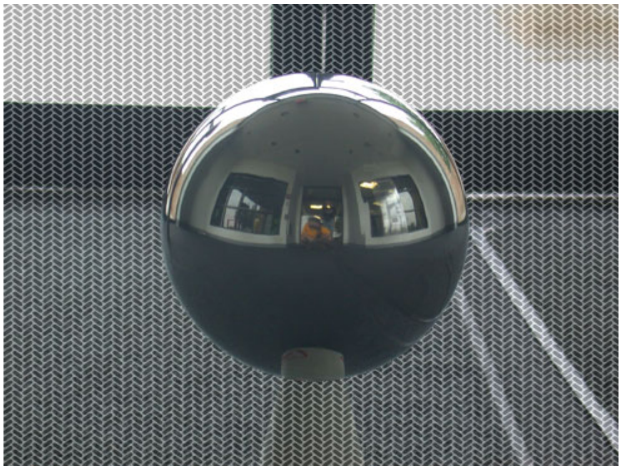
Shooting with a gazing ball

Figure 30

Images taken of the environment where the reflective sphere will be placed

After the tripod is setup, the camera should zoom in so that the reflective sphere takes up at least 30% of the frame (Figure 31). If the sphere is taking up more then 50% of the frame most likely it is also capturing too much of the photographer and camera which is undesired. Also make sure the line from the camera lens to the reflective sphere is parallel to the ground plane.

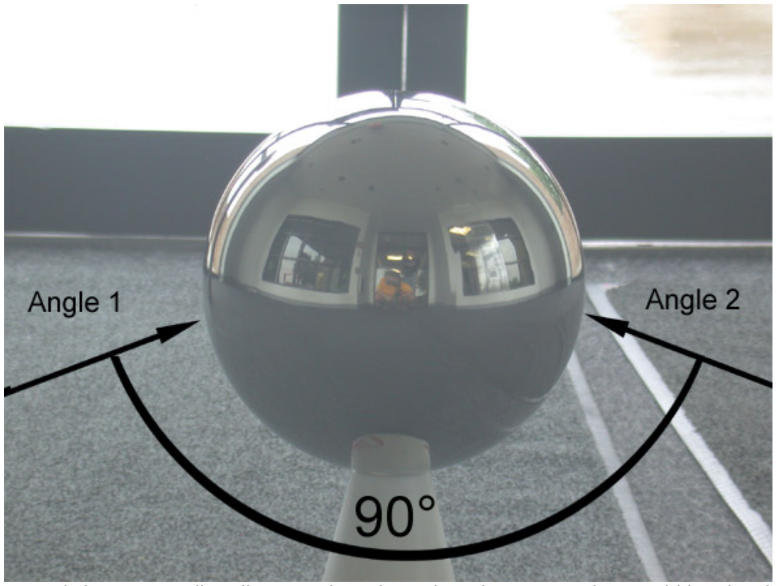
Figure 31



A photograph taken of a reflective sphere in an indoors environment. The sphere should take up at minimum 30% of the image.

Through this method of shooting the reflective sphere, we are able obtain about 280° to 310° of environmental information. The area remaining will just be black which is not ideal for lighting an object from all 360° in the renderer A solution, is to shoot from two different angles and combine the images from those two angles together to create one full lightprobe with 360° of information. Also with this technique it is possible to remove the photographer and camera from the image so that the 360° lightprobe only contains environmental information. The two different angles need to be around 90-100° apart from each other in order to fully capture the entire environment. Since usually a specific visual angle will be found, which the designer will want to composite their work into, this can be established as the “center line” or “line of action” as used in film (Figure 32). Once this center line is established move the camera 45-50° each direction and take the range of photos.

Figure 32



Photograph from center line diagramming where the other two angles would be shot from.

If the camera setup is not identical at each angle the images gathered will be significantly off and are not ideal for compositing. A simple solution for a designer is to simply tie a string around the base of the reflective sphere's stand and use the string to move an even distance back for either angle (Figure 33). After all the images are taken and uploaded they should be labeled immediately regarding which shot, angle and exposure setting to save time later when compiling an HDR image. It is also possible to use an image editing program, such as Photoshop, to paint in the missing areas as well if two angles were not shot. The tutorial that accompanies this paper contains more step by step instructions in for this step in the section "Camera Setup."

Figure 33

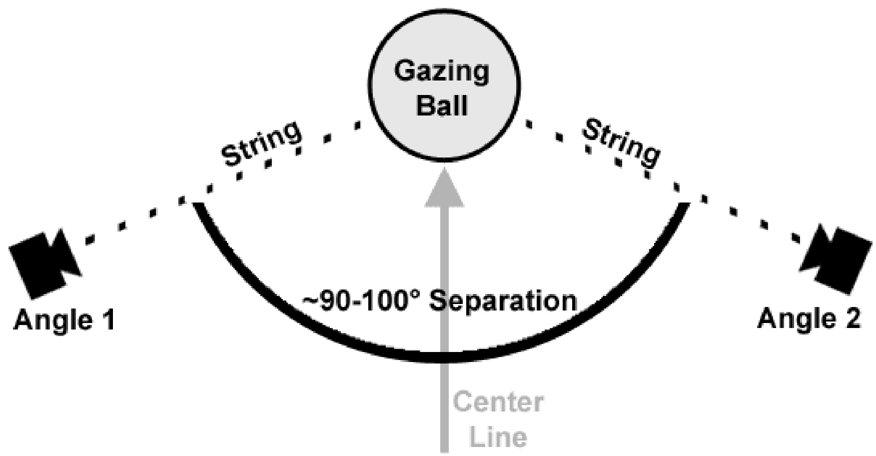


Illustration of a two angle setup using a string tied around the base of the reflective sphere, a gazing ball.

Assembling HDR images

Figure 34

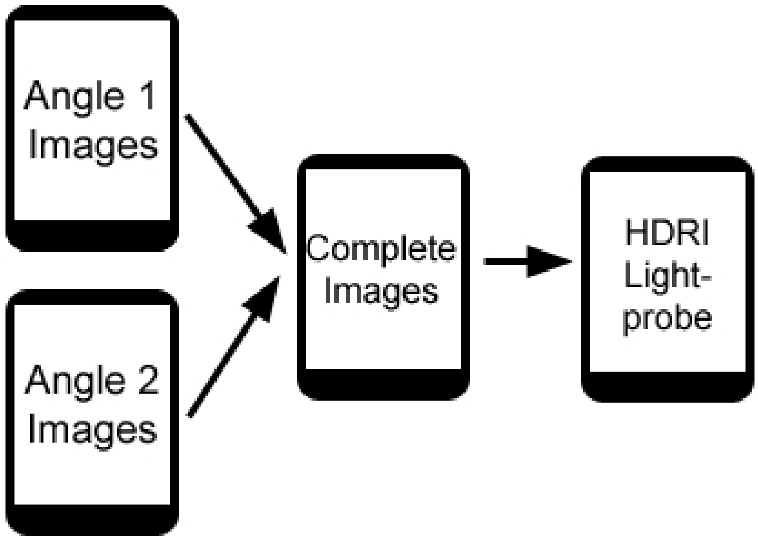
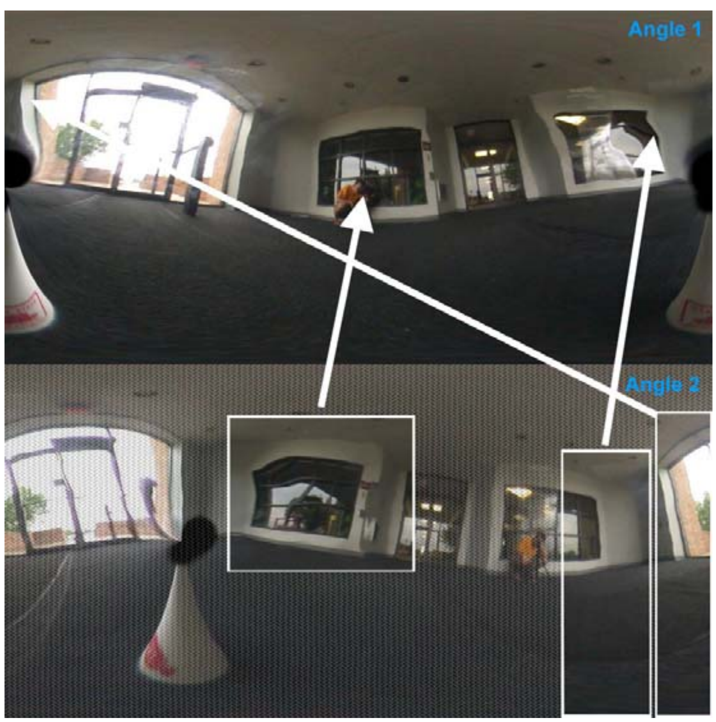


Diagram of this step of the pipeline

HDR is still a relatively new format and there are only a handful of publicly available programs that can load or create an HDR image from a range of LDR images. Currently these programs are mkhdr, a shell based utility, Photogenics HDR, which is commercially available, HDRIE, an open source program from the University of Illinois, Photosphere, a shareware program from Greg Ward only available on OSX and HDRShop which is freely available from Paul Debevec’s homepage. Each program has its own settings for creating an HDR image. This pipeline uses HDRShop to make the initial HDR image and lightgen to export additional lighting information. Also, Photoshop is used for the single LDR image corrections. The tutorial that accompanies this proposed pipeline specifically uses these programs as well as custom created or modified scripts. The section on “Cropping and masking the images” in the tutorial details several steps that must be performed before assembling an HDRI.

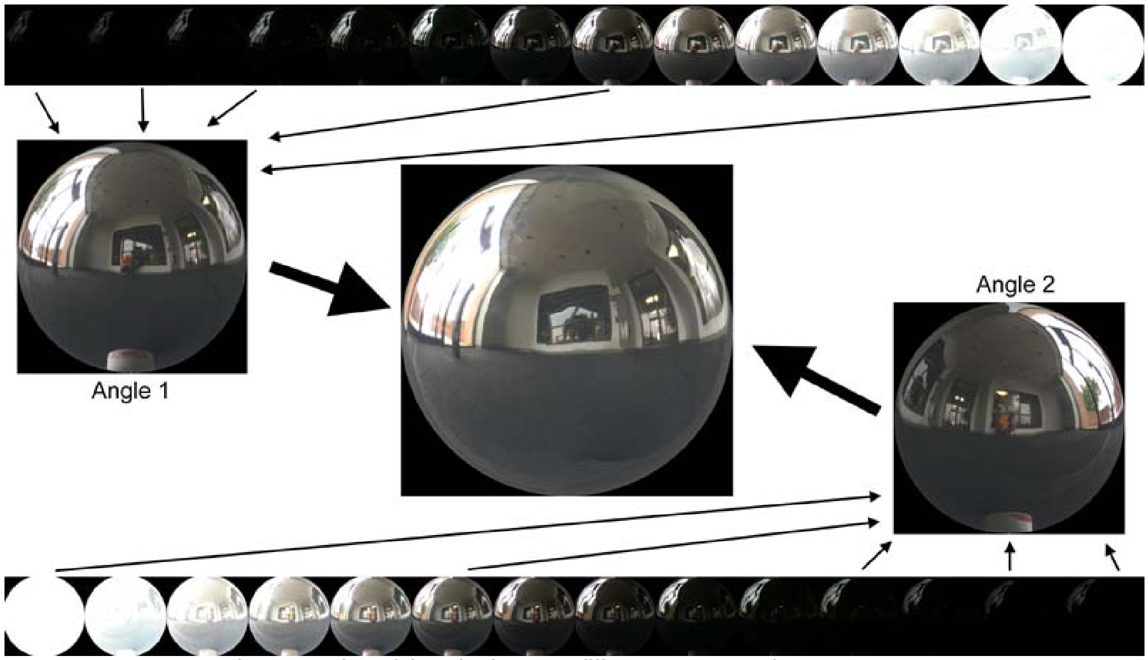
When using two different angles of images, the images must first be transformed so that they can be composited together. Each spherical map must be converted to a latitude longitude or planar map in HDRShop. One angle's set of images should be rotated 90° clockwise in Y so that the information from one angle can be used to fix the other angle (Tutorial – "Converting from Angular to Latitude-Longitude"). Once the latitude longitude images are created, they can then be brought into Photoshop and by using the good information from the rotated angle, it is possible to fix and remove objects from the other angle (Figure 35) (Tutorial – "Compositing the two sets of images together"). When this is completed there should be a new set of adjusted LDR images which can be combined to make one whole HDRI lightprobe (Figure 36).

Figure 35



Breakdown of using two different latitude longitude angles to create one complete angle.

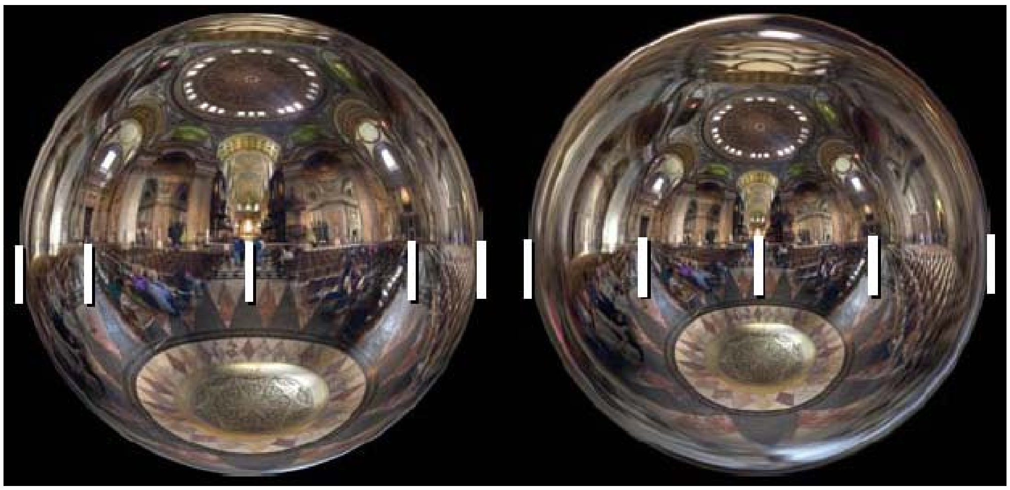
Figure 36



Creating a complete Lightprobe from two different camera angles' exposure range

In HDRShop, an HDR image can be created by loading in the set of LDR images and setting the calibration curve for the camera and the specified change of exposure between the images. Once an HDR image is created, it can be converted to various mapping formats and saved out in an HDR capable format or converted to a LDR format. In Image Based Illumination, there are basically three types of mapping; angular, cubic, and latitude longitude. Angular mapping is a spherical image that is represented as a lightprobe. The difference between a simple cropped photograph of the reflective sphere and an angular mapped image is that an angular image corrects for the distortion that occurs as the image bends around the sphere (Figure 37). Cubic mapping is a "cross looking" image with each section representing the texture that will go on each plane in a Cube (box) that will surround the CG model (Figure 38). Latitude longitude mapping gives is a planar texture (Figure 39), that can be used in several different types of surrounding surfaces, but is usually used on a sphere. In this pipeline a latitude longitude image will be used as it is used to map on spheres which are the surfaces that will be used in IBI (Tutorial – "Creating a High Dynamic Range Image (HDRI)").

Figure 37

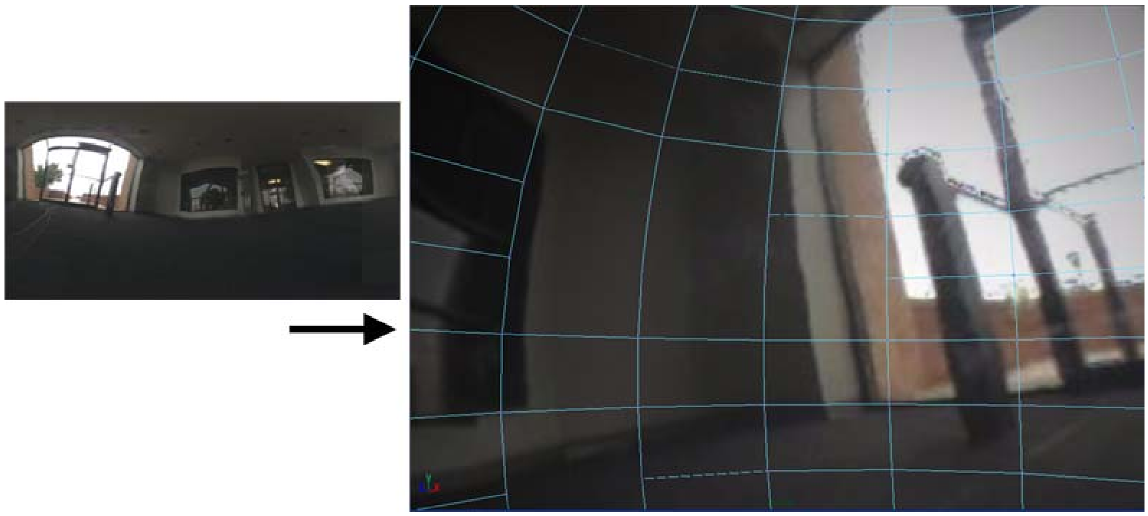


A photograph of the reflective sphere on the right compared to an angular map of the same image on the left. The hash marks indicate how an angular map accounts for distortion that normally occurs as the image bends around the sphere.

Figure 38

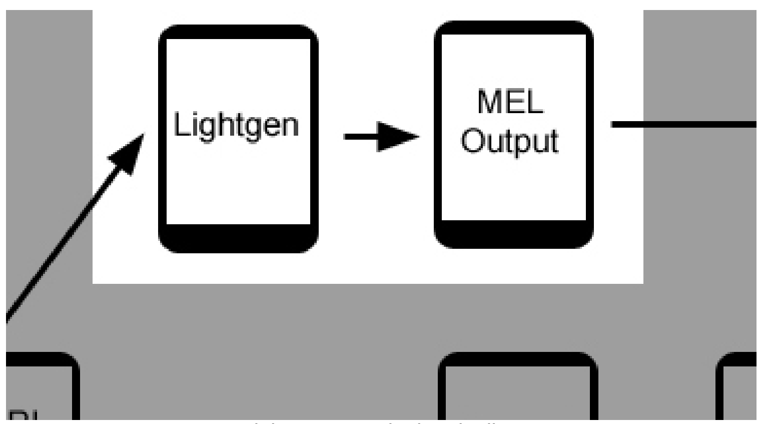
A cubic "cross" map used on a cube in 3D software

Figure 39



A planar "latitude longitude" map used on a sphere in 3D software

Figure 40



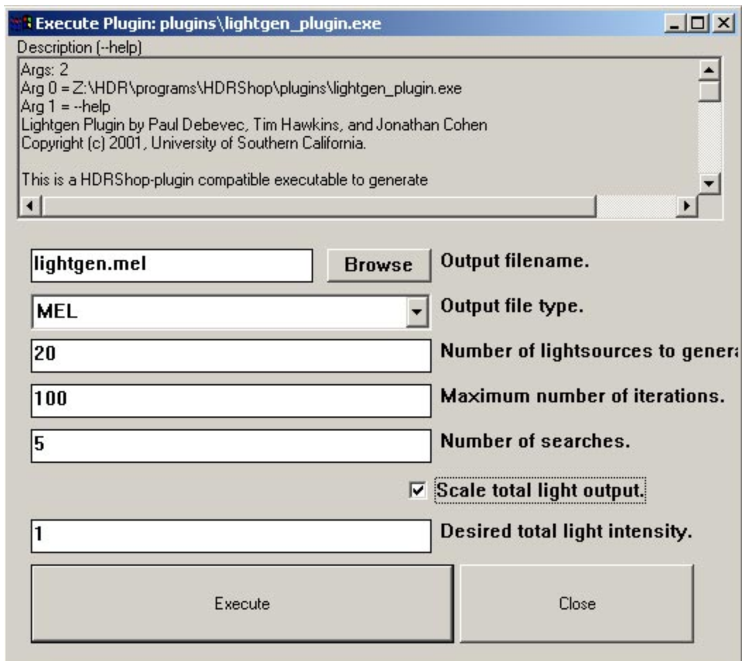
Lightgen stage in the pipeline

At this point a complete HDR image has been created to be used in IBI. Since IBI is already giving us lighting information, just not the depth, it would make sense to use it for an additional starting point for creating 3D lights. A freely available algorithm titled lightgen from Jonathan Cohen helps with this. Lightgen accepts an HDR image as an input and writes out a file with a specified amount of lights from the input file[17] (Figure 41). This file can be written out in several formats, including Maya Embedded Language

[17] Cohen, Jonathan. LightGen. 2001, http://www.ict.usc.edu/~jcohen/lightgen/lightgen.html

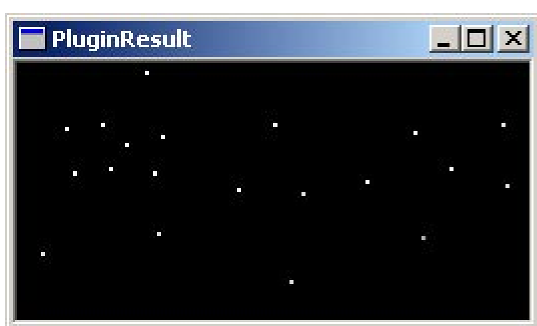
(MEL) format. The light information is derived from “hot spots” of the image. The result will be look like a starfield (Figure 44). The lights created from lightgen are not usually used in combination with IBI in existing pipelines which use one or the other, and therefore is a considerable difference in this proposed pipeline. It is helpful to blur the HDR image before running the lightgen algorithm in order to obtain better color information at each pixel (Tutorial – “Setting up for IBI”).

Figure 41



Lightgen plug-in in HDRShop

Figure 42



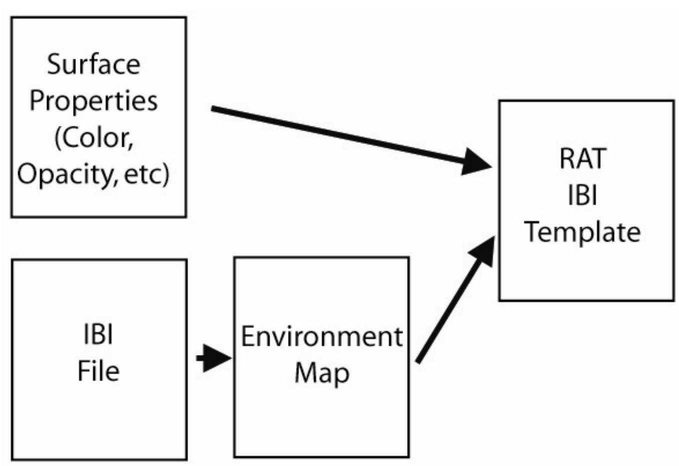
Lightgen result

3D Software Setup to use IBI and HDRI

In this pipeline the most direct approach to using IBI is to use Mental Ray. However, it is also possible to use RenderMan which is also a very suitable and popular renderer. Since RenderMan is a popular renderer it is worth noting how IBI and HDRI can be used in RenderMan. The most straightforward approach to use RenderMan in Maya is to use SLIM, which is part of the RenderMan Artist Toolkit (RAT) and has a built in IBI surface shader. This shader when mapped with a latitude longitude image allows the designer to use a GUI to attach that shader to certain objects.

To use this IBI template in Maya, SLIM, or Shading Language In Maya, allows the import of compiled SL and SLIM shaders as well as use their templates to create custom shaders. In SLIM, there is already an IBI surface shader, which we will first be using for our IBI tests in RenderMan. The IBI template has nodes for surface properties that the designer can tailor to each object (Figure 43).

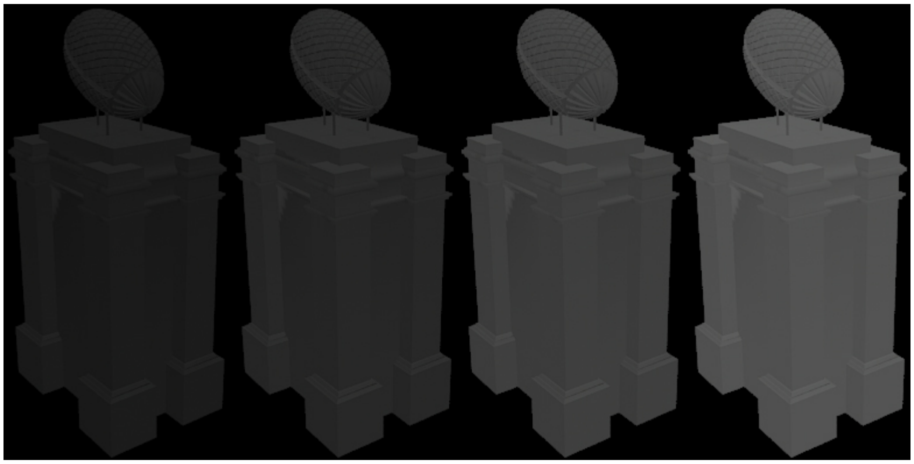
Figure 43



RAT IBI diagram

By default, SLIM's IBI shader doesn't offer any exposure or gamma compensation so that has to be controlled in the render globals. The problem with this is that these settings globally control the rendered image. Therefore, if that exposure is increased, it will increase the exposure for all elements in the scene. The easiest way around this is to just render different passes of the different objects and then composite the objects back together. In the render globals, there are two exposure compensation areas, gain and gamma. For our work we are concerned with gamma. For dark images 2-3 works out well, since digital cameras operate usually at a gamma setting of 2.2, and for lighter images the default 1 works fine (Figure 44). If the digital camera gamma settings are already known, use that gamma number in the calculation area. Another limitation in RAT is that the IBI shader template is parented to the object. With this parenting system when any changes occur with the attached object the template changes to match. For example as the object rotates so does the template therefore the lighting information is the same on all surface points of the object. This is effective if just the camera is moving around an object. However, if the object is animated and moving, the illumination information will never change due to the IBI template mirroring every change the object makes. This makes an ineffective use of IBI for serious animation but can be corrected through custom templates or perhaps the next version of the template.

Figure 44



Gamma changes in RenderMan (2, 3, 4, and 5)

With the release of Maya 4.5, Alias, formerly Alias|Wavefront, began offering a free public beta of Mental Ray for Maya. With the later release of Maya 5.0, Mental Ray was fully integrated into Maya. Mental Ray, being a native raytracer, works very well with Image Based Illumination and as mentioned is explored in more detail in this pipeline.

Mental Ray's native image file format is MAP, which fortunately HDR images can be converted to (Tutorial – "Making a MAP image"). In Mental Ray, the actual surrounding geometry must be created and then attached with a shader that has the color, specular, and incandescence channels mapped using the converted HDR file. Mental Ray uses final gathering, in combination with photon mapping, for IBI. With final gathering activated the image can be rendered and illumination will be determined through IBI from the surrounding geometry. By adding in linear ramp shaders it is also possible to control settings, such as faked exposure increase or decrease, by sliding the ramp between 0 and 1.

A discovery in working in this pipeline that should be noted is that Maya has a range clamping with certain HDR image formats that currently is not fixed as of version 5.0. With HDR images, depending on the exposure range, Maya could clamp the image so that the exposure is limited to one side of the exposure range. However, when using floating point TIFF and MAP file formats this limitation does not occur.

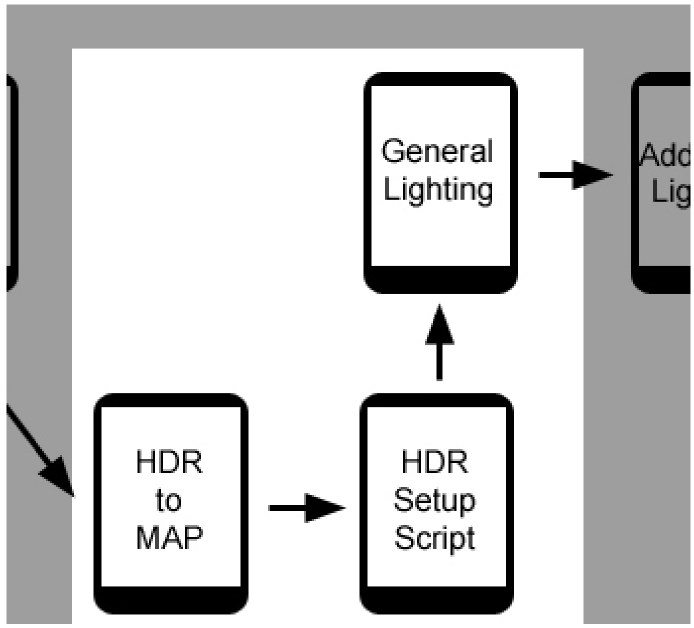
Rendering

IBI and HDRI are not the answer to all lighting situations but is a valuable tool in using images to generate realistic lighting. With the default use of IBI, it will completely illuminate the object; however, the resulting lighting still may not completely match the live action images. This is due to intensity levels of the light and any additional light specific properties, such as the casting of caustics and shadows. However, there are ways to use IBI and HDRI to create the rest of this lighting setup in this stage of the pipeline.

There are four basic steps in this stage of the proposed pipeline when working with IBI and HDRI. Each step must be tweaked for every individual scene, but once the pipeline is familiar it becomes more time-efficient.

- Use IBI to provide general lighting
- Derive actual lights from the HDRI
- Scene specific lighting (i.e. caustics, fog, etc)
- Dirtying the image and compositing

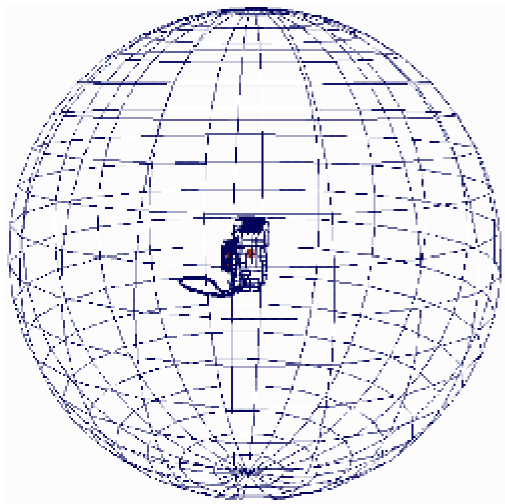
Figure 45



General lighting stage through IBI and HDRI

Since each renderer is different, the first step is usually different. However, as this pipeline is focused on using Mental Ray it will be explained accordingly. In Maya create a polygonal sphere enclosing the entire scene. Map a latitude longitude HDR converted to MAP image to that sphere. Rotate the sphere accordingly so that it matches the live action scene. Enable final gathering and calculate the minimum and maximum photon settings based on the size of the Maya scene (Figure 46). It is now possible to perform test renders and adjust the intensity of the illumination properties. Keep in mind that this illumination should be similar to ambient, low diffuse and specular lighting (Figure 47). We will add in the main elements through traditional 3D lighting. This process has been automated in a script for this pipeline titled *Brock's HDRSetup Control*, which is modified for use in Mental Ray (Figure 48) (Tutorial – "Using HDR for general lighting'"). This script will step through the same process as above, including creating a spherical IBI surface, but also allows the designer to control exposure, gamma and blur settings for the MAP image.

Figure 46

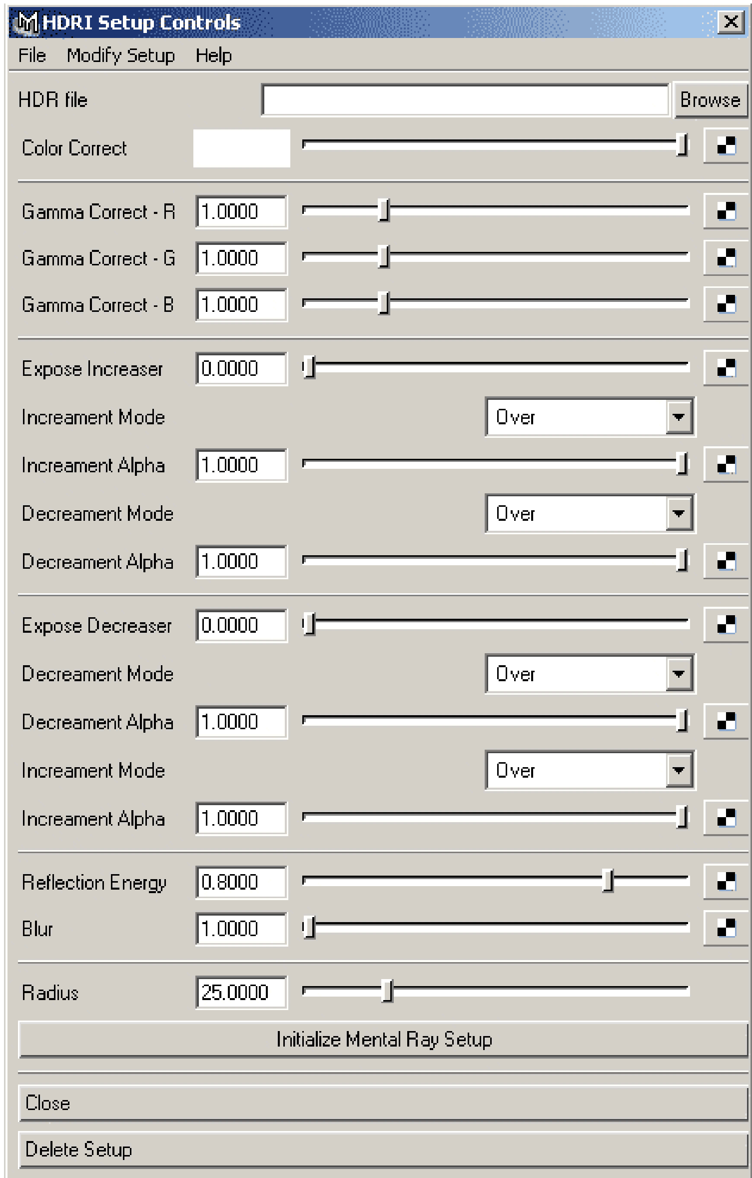


Step 1: Using IBI and HDRI to generate the general lighting

Figure 47

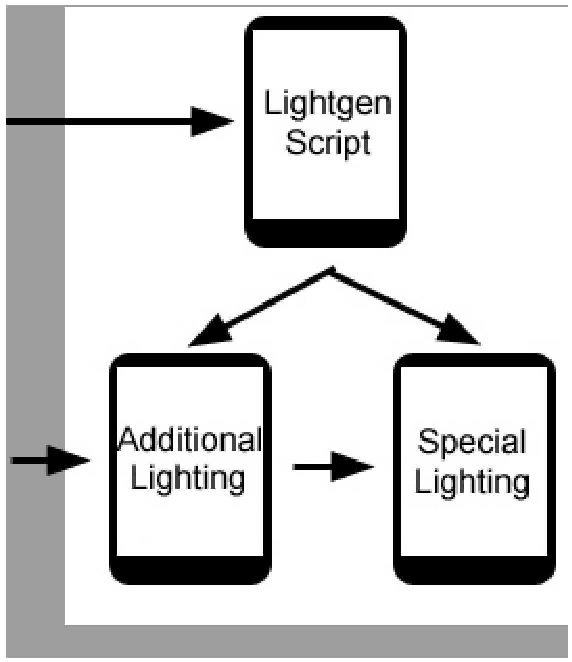
The cg camera mock-up with only IBI general lighting composited over the live action background

Figure 48



Brock's HDRSetup Control

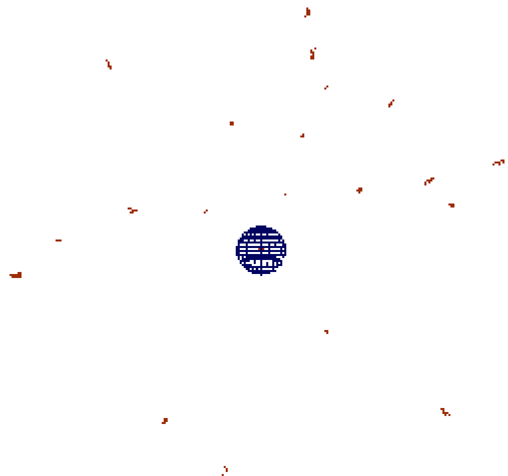
Figure 49



Additional and special lighting stages

The lightgen information created before is then passed through the 3D software package and directional lights are created (Figure 50). Now, the designer must alter these lights to be more accurate, which would include changing the distance and intensity of selected lights, as well as enabling shadows (Figure 51). Another script is available with this pipeline, titled *Brock's Lightgen Control* (Figure 52) that allows the designer to import the lightgen MEL file and control the intensity and shadows of several lights at once (Tutorial – "Using the HDR image to generate lights to create lights'").

Figure 50

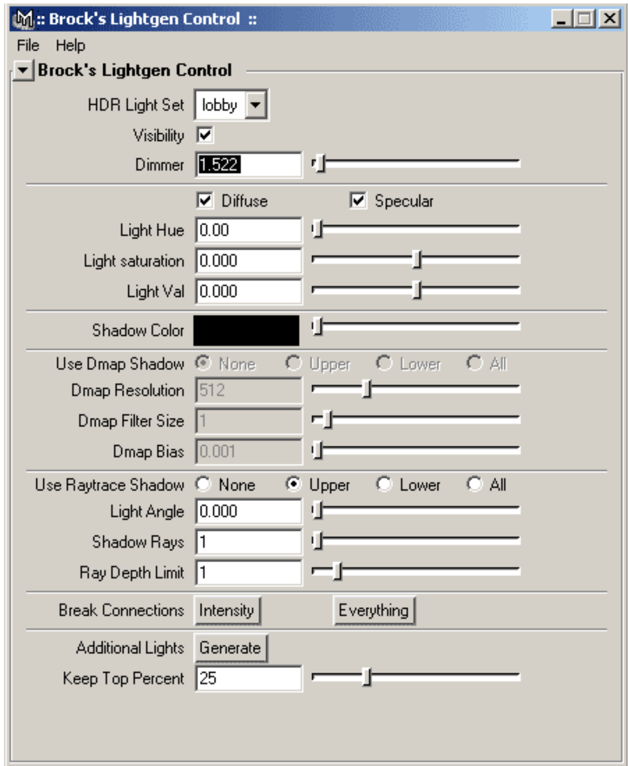


Step 2: Using the exported lightgen file to create Maya lights

Figure 51

The cg camera mock-up with both IBI general and lightgen derived lighting composited over the live action background

Figure 52

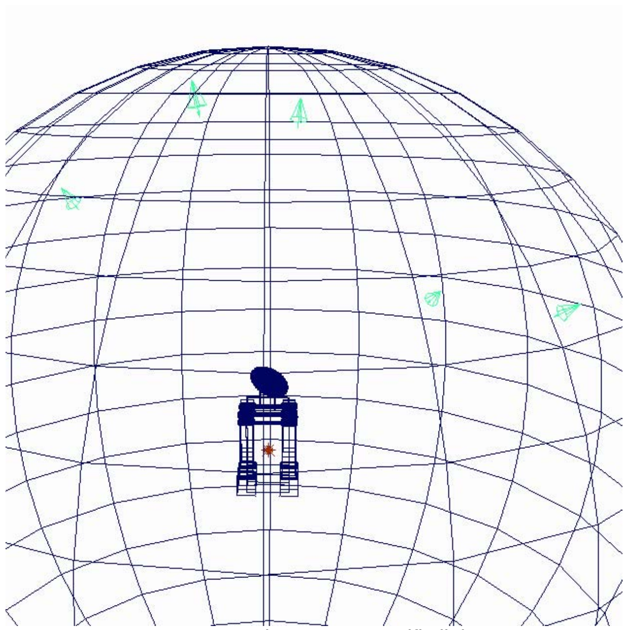


Brock's Lightgen Control script

This step in the process seems to fit the title Image Based Lighting since actual lights are being created from the image file. It is because of this step that in this pipeline the term Image Based Illumination (IBI) is used instead of Image Based Lighting (IBL) which other researchers may use. The designer should focus on the creation of shadows from where the lights in the 3D software are that match where the bright spots of the image are. Usually light is largely coming from the top or bottom so the script allows for shadows to be quickly enabled for those groups in depth map or raytraced shadows. Using *Brock's Lightgen Control* allows the designer to create actual lights in Maya quickly and accurately. As mentioned this stage in the pipeline is different then other existing pipelines in using IBI and HDRI.

The following step is specific to the product designer's scene and again is different then any existing pipelines in using IBI. For the example scene used here of the Circuit City Trophy, there is a glass object which when light passes through it would generate caustics. That is the specific effect that must be recreated to make the rendered image match what one would expect from those lighting conditions in the real environment. Again, we are going to use the lighting information from lightgen for this step. The script, *Brock's Lightgen Control*, also allows the designer to import additional directional lights and to selectively choose what percentage of lights should remain (Tutorial – "Creating scene specific elements"). In real life, the deepest shadows are usually created by the most intense lights. For caustics the same principle is true and since caustics will often overlap we can use a few lights only to create the caustics (Figure 53). Again, the most intense lights are an ideal choice and the photon settings for the caustics should be set accordingly. Since these lights are only generating photons for the caustics, the actual intensity can be turned off. These lights can be positioned at any distance from the object, as long as they are done so evenly to generate the caustic effects (Figure 54).

Figure 53



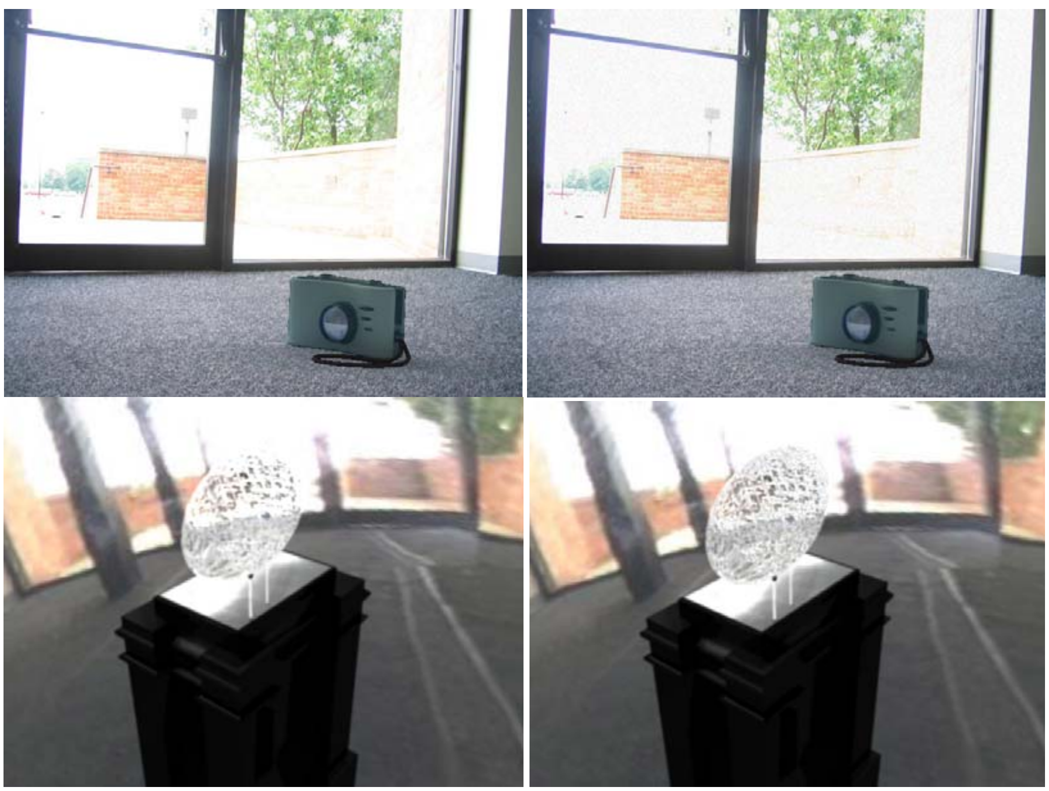
Step 3: Creating scene specific lights

Figure 54

Above is a product design that requires scene specific special lighting. The cg mock-up of the Circuit City trophy is lit just with special lighting, in this case caustics. On the right the image is composited over top a live action background.

The last step can be referred to as final blending and is very scene specific. Computer generated imagery often has the look of being too perfect. This is because the imagery that we see through photographs or film has always contained noise in the print. Also, the human eye can only focus on one area at a time so everything else will fall out of focus. With a computer generated camera the focus is usually evenly spread and must be later altered using the depth of field of the camera. These settings must all be considered when blending the computer generated imagery with the live action imagery. Another important note is that in real life very few hard edges exist, but in computer generated models most of them have many hard edges by default. Simple beveling of those edges can create a softer look which will also make the computer generated image seem more natural to the real world setting. For the camera example noise and blur must be done on the image to help blend it into the live action imagery (Figure 55). It should be clear at this point that this pipeline can produce a better result then a normally lit example (Tutorial – "Dirtying up the image").

Figure 55

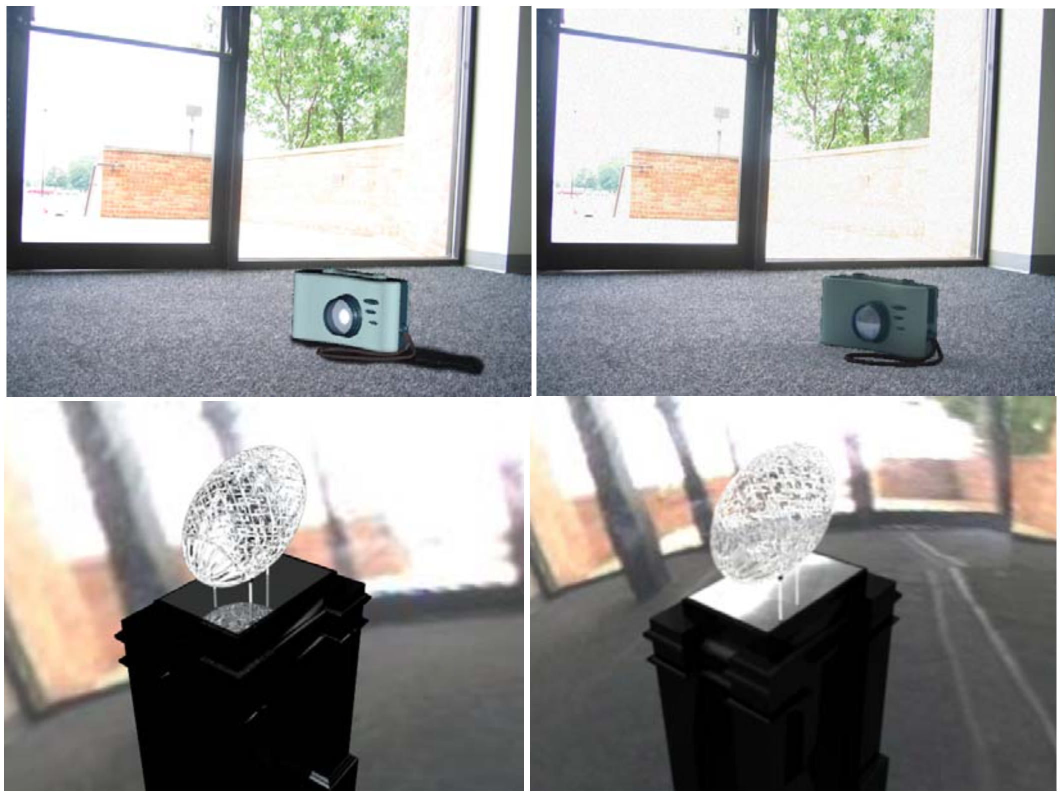


The cg camera and trophy mock-up are shown on the left with the first four stages of lighting. On the right the same image but with added defocus, blur, and noise to blend the cg mock-up into the live action background.

Significance of Work

Considerations and Limitations

Figure 56



Comparison of a traditional lit example (left) to the same product using this proposed pipeline (right)

It is clear that IBI and HDR are powerful tools and when used in combination with existing tools and techniques can provide a more efficient way for a designer of various disciplines to combine their computer generated product with live action imagery (Figure 56). The main difficulty with IBI and HDRI is that it is not a one-stop solution for a problem which a designer may face. Also, some recent films have begun to use IBI and HDRI more effectively. The first major theatrical film to use this technique was X-Men, with effects done by Rhythm and Hues. This technique was used again in Matrix

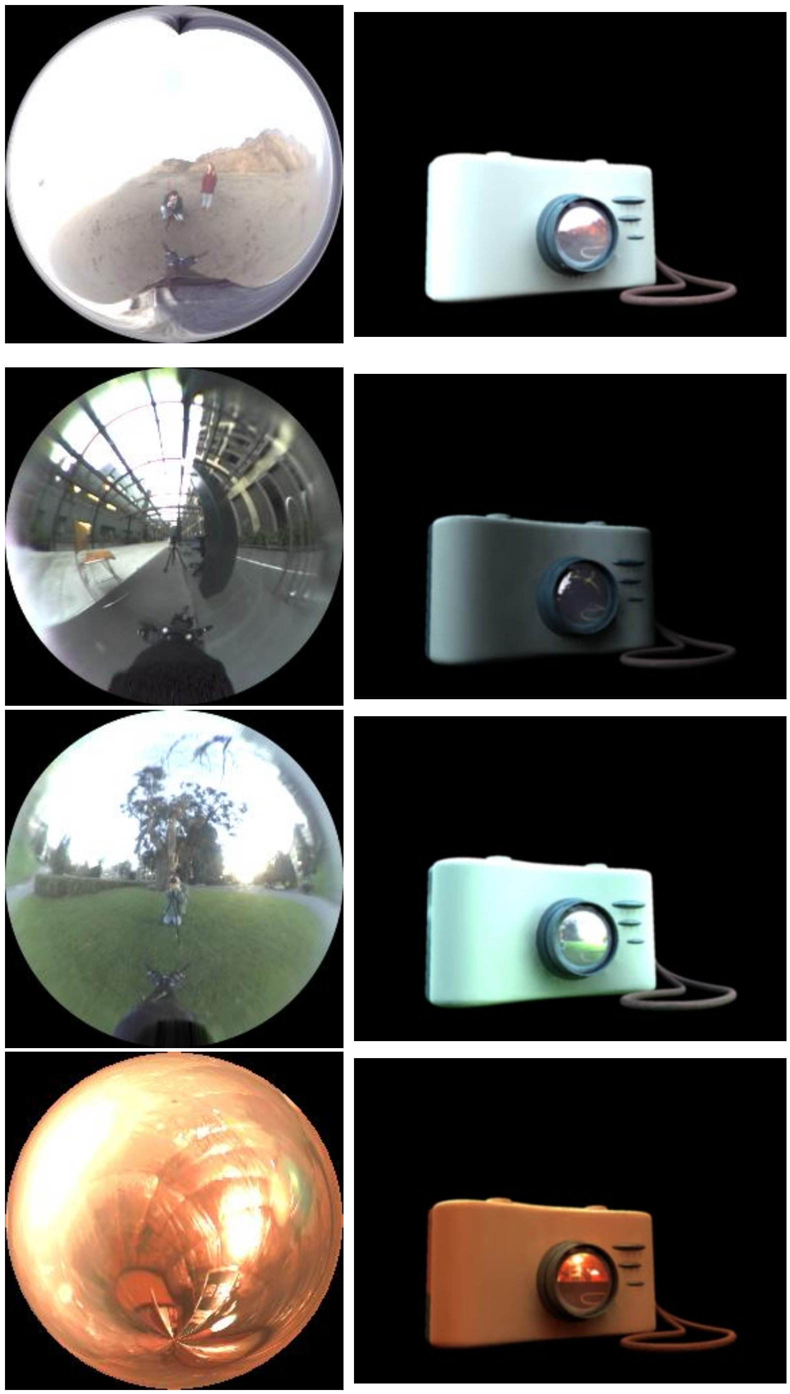
Reloaded[18], X-Men 2[19], Men in Black II[20], Gangs of New York[21], Daredevil[22], Terminator 3: Rise of the Machines[23], Pirates of the Caribbean: Curse of the Black Pearl[24], The Hulk[25] and the upcoming film Cat in the Hat[26]. This has prompted more studios to adopt IBI and HDR capabilities into their proprietary renderers, in order to take advantage of these techniques. Given this, it can be expected for more commercial 3D software packages and renderers to also take better advantage of IBI and HDRI.

This proposed pipeline has combined some of the steps and techniques from exiting film pipelines to a product design pipeline with some significant differences. These differences include the combination of IBI as well as creating actual 3D lights from the HDRI in an easy to use pipeline. Even with these differences, there are some limitations that exist for this proposed pipeline. The first of which is that if the designer cannot shoot, or have someone else shoot a specific lightprobe for their environment, then there is not a specific use for this pipeline. However, with more people creating lightprobes and making them publicly available[27], a designer may find a suitable replacement probe to generate the ambient lighting and then continue to traditionally light it from that point on. Also, with the availability of many different lightprobes, it is possible a designer may quickly move their product from environment to environment using different probes.

[18] Silberman, Steve. Matrix2. May 2003, http://www.wired.com/wired/archive/11.05/matrix2_pr.html
[19] Robertson, Barbara. "x-cellent" Computer Graphics World July (2003): 34.
[20] Industrial Light and Magic. *OpenEXR*. 2003, http://www.openexr.com/
[21] Industrial Light and Magic. *OpenEXR*
[22] McEachern, Martin. "new wrinkles" Computer Graphics World July (2003): pp. 38-39.
[23] Bogart, Robert, Industrial Light and Magic, SIGGRAPH 2003
[24] Bogart, Robert
[25] Bogart, Robert
[26] Adamson, Judith
[27] Dosch Design. Dosch HDRI. 2002, http://www.newmagic.com.au/NM_pages/products/dosch/DD_HDRI.html

These probes are available in HDR format and therefore can be used in this proposed pipeline (Figure 57).

Figure 57



Examples of using different lightprobes to provide different lighting scenarios for the same cg mock-up camera

Another limitation is the difference between raytracing and radiosity renderers can cause different results with IBI. Raytracing usually produces a better result but can be much more time consuming which may be a factor in adopting this pipeline. Also, another limitation is that this pipeline has been developed for the use in Maya. Maya, while a very effective and popular 3D animation package, is not a primary package for designers. It is the hope that this pipeline could be further extended to more product design based software, such as Alias Studio Tools, using similar scripts and similar renderers.

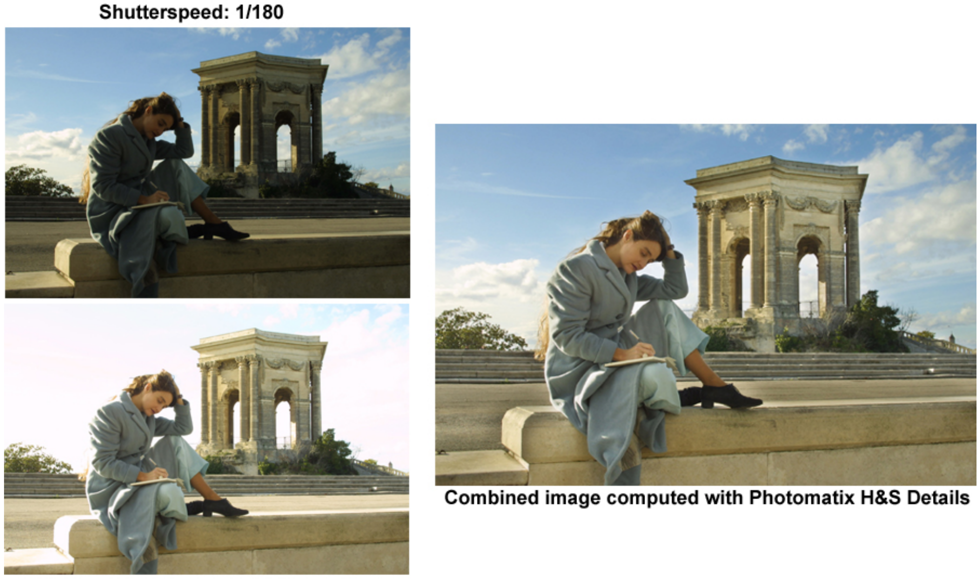
As noted most large design firms have specific roles established for lighters and compositers and some products are not lit accurately for specific reasons. For these cases this pipeline may not be directly effective. However, for certain cases this pipeline could still be adopted but would require specific studio direction in expanding the communication from the on-set personnel, to the modeler, and to the lighter/compositer since the IBI work is critical to all of those stages.

Another aspect of using HDRI that can be useful for all designers is that an HDR image can be used through certain pipelines to create and display more accurate and visually pleasing LDRI. In order to create HDRI, many LDRI at different exposure levels are compiled with different lighting information. In HDRI we can see all of the dark and light areas clearly through these levels. It could be conceived that if we were able to take the details in both the light and dark areas and flatten it together to a single image that we would have a very sharp image. This is possible through techniques using bilateral filtering and gradient tone reproduction more commonly referred to as Tonemapping

(Fattal)[28]. With this technique it is possible for designers to use more accurate photographic images in their work. Several companies have also started selling this technique in their own programs by accepting the input of several different exposed photographs and then combining them into one main image[29] (Figure 58). Even further expansion has been done to create virtual turnarounds in this accurate 3D space of a photograph, especially for interior visualization through the World Wide Web (Figure 5). For designers who have already created a physical mock-up of their model this could be a useful and cost-effective solution to creating a very high quality image of their physical mock-up with a common digital camera and tripod.

Figure 58

Shutterspeed: 1/180



Combined image computed with Photomatix H&S Details

Shutterspeed: 1/45

Photomatrix's approach to using HDRI to produce a more pleasing and accurate LDRI

The viability of the proposed pipeline was tested by submitting a problem design to a designer only peripherally familiar with the approach. This person was trained in the use

[28] Reinhard, Erik, Stark, Mike, Shirley, Peter and Ferwerda, Jim. *Photographic Tone Reproduction for Digital Images*. 2001, http://www.cs.ucf.edu/%7Ereinhard/cdrom/tonemap.pdf
[29] MultimediaPhoto. *Photomatrix*, 2003. http://www.multimediaphoto.com/photomatix/

of Maya, but had not used the IBI pipeline proposed. He followed the approach of the pipeline with a geometric model of a typical product, the camera, and successfully accomplished a more realistic lighting and compositing of that product into a live action photograph. The tutorial was then modified to simplify the process of loading scripts, using default values accordingly and elaborating what actually is being done in each step. Though using this pipeline and then continuing to visually place the lights the person was able to produce a realistic result (Figure 59).

Figure 59

Above is a result from the test student using the proposed pipeline as well as visually placing the lights following the tutorial.

Future Development

As already noted, there are some various limitations to IBI and HDRI. Some of these can be corrected or better implemented through future development. A beginning step is to improve the capturing setup. Even with a reflective sphere this process can still take a long time. A recent answer is the SpheroCam HDR which captures High Dynamic Range images with a dynamic range of 26 f-stops[30] (Figure 60). This saves the need to manually adjust the exposure setting between each photo. This camera can take them all within a short time period without user interference. This camera is currently being manufactured by one company in Germany and is very expensive. For production companies, this could be a major time saver but for designers on a budget this is a very expensive purchase.

Figure 60

The SpheroCam HDR camera

The next area of consideration is in rendering which has always been a time consuming process. When rendering with IBI and HDRI, render times continue to increase so it is

---

[30] SpheronVR. *Spherocam HDR*, 2002. http://www.spheron.com/products/SpheroCamHDR/spherocam_hdr.html

important to streamline this process as much as possible. Since IBI is directly related to lighting, it is important to have as much "in real time" viewing of the lighting as possible. However, this "in real time" IBI isn't possible in any 3D software application. A possible solution to this is that graphics cards are now advanced enough to carry a large load of the processing power for the rendering of complex shading and lighting information. Given this capability it could be possible for the graphics processor to perform view independent radiosity calculations to result in a real time "faked" IBI result. The result may not be as nice as raytracing but since raytracing is view dependent and is currently beyond the capabilities of current video cards, the result will still provide a fairly accurate representation of IBI. Using programming languages native to graphics cards such as Microsoft's HLSL for DirectX or Cg for NVIDIA cards; it is possible to implement this. Recent versions of the major 3D animation software applications support Cg, so it is conceivable that an IBI shader could be created in Cg and real time viewing of the lighting could be seen, interacted with, or controlled. At render time, the appropriate software renderer would replace the Cg IBI shader. Some recent work in this has been done for demonstration purposes for ATI and NVIDIA cards in only displaying existing HDR rendered scenes in real time. There has yet to be implementation for real time creation and interaction of IBI lighting environments in any 3D animation software package specifically for HDRI. Ideally, this next step is right around the corner as the half datatype in the Cg language is able to store HDRI information natively and is fully implemented in all FX based NVIDIA cards.

Surprisingly enough, there has also been development in using HDRI for gaming engines. The Terragen program, used to create levels for the Quake engine, allows for the support of HDRI through a custom created script. This is useful if the environment the characters are participating in is a real life or close to real life environment as the lighting can be solved quickly and easily through the use of HDRI as lighting is not as critical in games as it is in feature films.

Other possible expansions can include better caching and baking of IBI once a final IBI setting has been determined. Already the possibility of baking traditional lighting into the textures of a scene has sped up the rendering process. It is conceivable that once a determined IBI setup has been completed that this information could be baked similarly. In Mental Ray it is possible to cache the final gather photon map which speeds up the rendering of several frames but there is clearly more that can still be done.

Another issue has been displaying HDRI as no common displays can display that large range. At SIGGRAPH 2003, Sunnybrook Technologies demonstrated their HDR Display. These displays replaces the usual LCD backlight of LCD displays with a custom controlled array of ultra-high brightness white or tri-color light emitting diodes which can be individually controlled. The result is a high resolution HDRI that produces a 60,000:1 ratio of light to dark compared to the high end LCD 400:1 ratio and the common CRT 300:1 ratio. The result is still not a “true” HDR image but it is the closest visual display currently possible.

All of these areas require further expansion, but each one could be combined with another to speed up rendering as well as making it more accessible for the designer. These areas of exploration, in combination with the further advancement of renderers and their support of IBI make this pipeline extremely applicable and important for a designer to use. It is clear that studios have made use of IBI and HDRI in film very well. Now there is the chance for the product designer to use the same theories and principles in displaying their work. This approach may be most applicable to those product designers that do the promotion themselves, since this requires a back and forth method of using both the model and the lighting information. For larger design studios that wish to assimilate these techniques they must establish their own pipeline structure for the transformation of information and images between the different members of the team.

It is also apparent that this approach would be extremely valuable for students who are constantly seeking professionally looking promotional displays but rarely contain all the skills or resources to create that quality look themselves. This pipeline has been based on the principles of IBI and HDRI. Given this, it is also possible to continue the expansion and alteration for each product designer's specific needs. The development of IBI and HDRI in available 3D packages and renderers is a very valuable tool that now is within grasp of everyone. With this pipeline these now available methods are now structured so that product designers can create realistic blends between computer-generated mock-ups and real life environments.

Bibliography

Apodaca, Anthony and Gritz, Larry. *Advanced RenderMan*. 1999, Morgan Kaufmann

ATI Developer. *Rendering with Natural Light in Real Time*. 2002, http://www.ati.com/developer/demos/R9700.html

Bauer, Christian. *CGTechniques.com*. 2001, http://www.cgtechniques.com

Bunker, Simon. *RenderMania*. 2000, http://www.RenderMania.com/HDRI/

Cohen, Jonathan. *LightGen*. 2001, http://www.ict.usc.edu/~jcohen/lightgen/lightgen.html

Debevec, Paul. *Paul Debevec Homepage*. 2003, http://www.debevec.org

Debevec, Paul. *HDRShop*. 2002, http://www.debevec.org/HDRShop/

Debevec, Paul. *High Dynamic Range Imagery and Image-Based Lighting*. 2002, http://www.gdconf.com/archives/2002/paul_debevec.ppt

Debevec, Paul. *High Dynamic Range Imagery and Image-Based Lighting*. 2003, http://www.gdconf.com/archives/2003/Debevec_Paul.ppt

Debevec, Paul. *High Dynamic Range Imagery and Image-Based Lighting*. 2003, SIGGRAPH 2003

Debevec, Paul. *Image-Based Lighting*. SIGGRAPH 2001. 2001, http://www.debevec.org/IBL2001/

Debevec, Paul, *Recovering High Dynamic Range Radiance Maps from Photographs*. SIGGRAPH 1997. 1997, http://athens.ict.usc.edu/Research/HDR/

Debevec, Paul, *Rendering Synthetic Objects into Real Scenes*. SIGGRAPH 1998. 1998, http://athens.ict.usc.edu/Research/IBL/

Debevec, Paul. *The Story of Reflection Mapping*. 1997, http://athens.ict.usc.edu/ReflectionMapping/

Downing, Greg. *Stitched HDRI*. 2001, http://www.gregdowning.com/HDRI/stitched/

Durand, Fredo and Doresy, Julie. *Fast Bilateral Filtering for the Display of High-Dynamic-Range Images*. SIGGRAPH 2002. 2002, http://graphics.lcs.mit.edu/~fredo/PUBLI/Siggraph2002/DurandBilateral.pdf

Fattal, Raanan, Lischinski, Dani and Werman, Michael. *Gradient Domain High Dynamic Range Compression*. SIGGRAPH 2002. 2002, http://www.cs.huji.ac.il/%7Edanix/hdr/hdrc.pdf

Haeberli, Paul and Segal, Mark. *Texture Mapping as a Fundamental Drawing Primitive*. 1993, http://www.sgi.com/grafica/texmap/

Inanici, Mehlika, *Research Objectives*. 2002, http://www-personal.umich.edu/~inanici/resobjectives.pdf

Industrial Light and Magic. *OpenEXR*. 2003, http://www.openexr.com/

Jensen, Henrik Wann. *Realistic Image Synthesis Using Photon Mapping*. 2001, A K Peters, Ltd

Kawase, Masaki. *Rthdribl – Real Time High Dynamic Range Image-Based Lighting*. 2003, http://www.daionet.gr.jp/~masa/rthdribl/index.html

Lee, Jaemin, *Practical Image Based Lighting and Compositing*. 2002, http://www-viz.tamu.edu/students/softviz/thesis/thesis_screen.pdf

Lee, Jaemin, *Deriving Lights from Pixels*, Architecture Week. 2002, http://www.architectureweek.com/2003/0528/tools_1-1.html

MultimediaPhoto. *Photomatrix*, 2003. http://www.multimediaphoto.com/photomatix/

Reinhard, Erik, Stark, Mike, Shirley, Peter and Ferwerda, Jim. *Photographic Tone Reproduction for Digital Images*. 2001, http://www.cs.ucf.edu/%7Ereinhard/cdrom/tonemap.pdf

Silberman, Steve. Matrix2. May 2003, http://www.wired.com/wired/archive/11.05/matrix2_pr.html

University of Illinois. *ACM SIGGRAPH at the University of Illinois*. 2002, http://www.acm.uiuc.edu/siggraph/HDRIE/

Figure Reference

Figure 1: Existing pipeline results – Brock Stearn

Figure 2: Senator Kelly's death scene, X-Men – Sony Pictures

Figure 3: Neo's fight scene, The Matrix Reloaded – Warner Bros. Pictures

Figure 4: Outdoor Light Probes – Paul Debevec

Figure 5: ADR panorama with SPi-V – fieldofview inc.

Figure 6: Pipeline results comparion – Brock Stearn
Pipeline testing lighting of cg camera – Jeff Ostergaard

Figure 7: Early Reflection mapping – Gene Miller

Figure 8: Classroom interior – Mehlika Inanici

Figure 9: Exterior sculpture visualization – Jaemin Lee

Figure 10: IBI lighting setup using the CG camera mock-up – Brock Stearn

Figure 11: Normals Diagram – Brock Stearn

Figure 12: Global Illumination Diagram – Brock Stearn

Figure 13: Image Based Illumination Diagram – Brock Stearn

Figure 14: Pixels Saturated – Brock Stearn, Original Photo – Jeff Ostergaard

Figure 15: Low Dynamic Range Imagery to High Dynamic Range Imagery illustration – Brock Stearn

Figure 16: Neutral Density filters – Paul Debevec

Figure 17: Shutter speed example – Brock Stearn

Figure 18: Single camera diagram – Brock Stearn

Figure 19: Lobby environment comparison – Brock Stearn

Figure 20: IBI: Pseudo Area light – Brock Stearn

Figure 21: IBI: Sphere of lights – Brock Stearn

Figure 22: Image Based Illumination raytracing diagram – Brock Stearn

Figure 23: Photon Mapping Example - Henrik Jensen

Figure 24: Existing product design pipeline – Brock Stearn

Figure 25: Proposed pipeline – Brock Stearn

Glossary

Caustics – Light patterns created when light from a light source illuminates a surface by one or more specular reflections.

Global Illumination – Lighting effects calculated between lighting that occurs between all of the surfaces of a scene that may cause light to be reflected or transmitted differently.

HDRIE – An open source program available from the University of Illinois that can read and modify high dynamic range imagery. http://www.acm.uiuc.edu/siggraph/eoh_projects/eoh2002.html

HDRShop – A program developed by Paul Debevec's team and the University of California, Berkeley that can read, create and manipulate HDR images. http://www.ict.usc.edu/graphics/HDRShop/download/

High Dynamic Range Image – An image that has a greater dynamic range than what can be shown on a standard display device. HDR images have the important property in that their pixel values are proportional to the amount of light in the world corresponding to that pixel.

Image Based Illumination/Lighting (IBI/L) – The process of illuminating an object through the use of an image file.

Lightgen – An algorithm developed by Jonathan Cohen that can create a file with lighting information of the "hot spots" of a high dynamic range image. http://www.ict.usc.edu/%7ejcohen/lightgen/download.html

Lightprobe – an omnidirectional, high dynamic range image that records the incident illumination conditions at a particular point in space

Local Illumination – Lighting derived only from the light sources in a scene

Low Dynamic Range Image – An image that can be shown on a standard display device and can be captured with a standard camera in one exposure.

MkHDR – A shell based program that can compile several LDR sources into a HDR. http://dcine.dyndns.org/mkhdr/

Photogenics HDR – Commercially available software for the reading, creation, and modifying of high dynamic range images.

Photon Mapping – The technique in which photons are emitted from the light sources that can bounce and stick to the objects in the scene. This information is gathered into a Photon Map which can be used to accelerate specific lighting methods such as Global Illumination and Final Gathering.

Radiosity ("Raycasting") – A ray from each pixel on the screen is cast into the volume along the view vector. While following the ray the voxels that are pierced are samples and the resulting colors and opacities are calculated in the ray's final intensity.

Raytracing – Rays are cast from the camera or eye that continue until they hit an object and at that point are bounced or absorbed depending upon the object. The rays continue to gather information as they move around the scene. This information is used to create a more photo-real look.
Reflection Mapping - A technique for simulating the appearance of a reflective object by using an image file for the reflection of the surface in a shader

Texture Mapping – A method for attaching images onto objects using a lookup per pixel of the image to the model's coordinates.

Tonemapping – The process of using values from several exposures to generate a more accurate low dynamic ranged image.

Voxel – The 3D counterpart of the 2D pixel. Each voxel is a quantum unit of volume and has a numeric value (or values) associated with it that represents properties of the object (i.e. color, opacity).

Appendices

A Designer’s Tutorial in using Image Based Illumination and High Dynamic Range Imagery

August, 2003

http://www.accad.ohio-state.edu/~bstearn/mfa_thesis/tutorial/

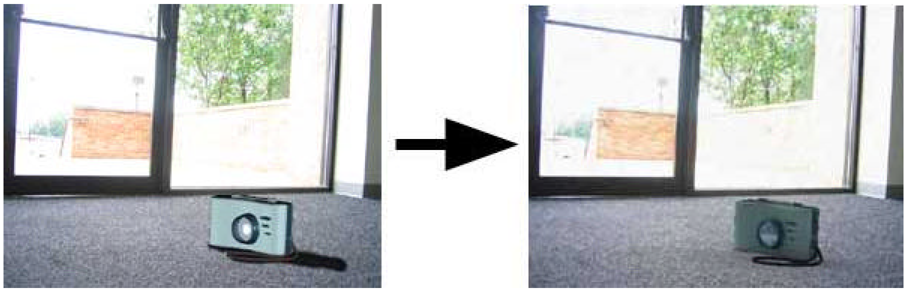
A Designer's Tutorial in using Image Based Illumination and High Dynamic Range Imagery

Brock J. Stearn
Department of Industrial, Interior and Visual Communication Design
The Ohio State University

Introduction

This tutorial follows a pipeline using Image Based Illumination and High Dynamic Range Imagery to more effectively composite a CG mock-up into a live action photograph. For example the mock-up camera seen here is composited into a lobby. On the left is the typical way a designer will composite a mock-up into a photograph by lighting the model with basic lighting and then using Photoshop to place it into the scene. On the right is the same camera composited into the same photograph using this pipeline.

Figure 1



Comparison of a result from the exisiting pipeline (left) to the result of this proposed pipeline (right)

This pipeline is specific to Alias Maya 5.0 or higher, but could be adapted with some effort into other software packages. Also, there will be references to my MFA thesis paper on this pipeline and related work throughout this tutorial. The entire paper is available here in PDF format. The page references listed in this tutorial are to that document.

Software/Downloads

For this tutorial there are certain available pieces of software, plug-ins, and scripts that I will be referencing. Below are links to download them and directions as to where to install them.

- **HDRShop** - Install to any directory you like.
- **Lightgen** - Install to your HDRShop/plugins directory
- **hdr2map** - Extract to a directory in your PATH
- **Brock's HDRSetup Control** - Extract to your current version of Maya's script directory
- **Brock's Lightgen Control** - Extract to your current version of Maya's script directory

Note: This tutorial includes instructions for **Adobe Photoshop 7.01** and **Alias Maya 5.0 (which includes Mental Ray)** which are not available for free.

"On Set" Capturing

The first step in this process is to find a location to take photographs. This location should ideally be where you could see your CG mock-up existing realistically. Once this location has been found you will need to take some photographic reference of this area. The best way to do this is with a digital camera and a reflective sphere, such as a gazing ball.

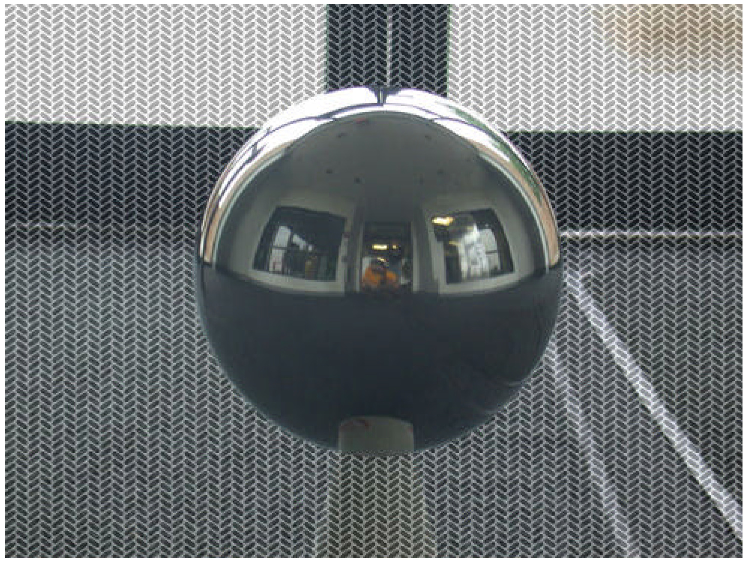
Camera Setup

Place the reflective sphere where the computer generated mock-up will later be composited in. Find the desired angle of the camera and mark this spot (pp 36). Now rotate from that spot in 45-50° each direction and these two spots will be where you will take the photographs of the reflective sphere (pp 37-39). The reason we are shooting two different angles is so that they can be composited together to make one whole image without any loss of environmental information, or any objects in the images, such as the camera and camera person (pp 42).

Framing the Sphere

It is important to get as much information from the reflective sphere as possible without allowing the camera, and you, to take up a large part of the sphere. The best way to do this is to fully zoom in on the reflective sphere and then manually walk the camera towards the sphere so that the sphere takes up 30-50% of the image (pp 37-38).

Figure 2



Photograph showing Sphere taking up between 30-50% of the image

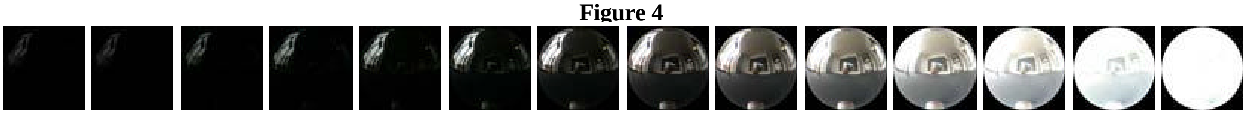
Figure 3

The environment in which the reflective sphere is placed

Range of Images

In order to use High Dynamic Range Imagery (HDRI) we want to take a full exposure range of images of the sphere. So adjust the camera's shutter speed setting and take one photo at each setting from the fastest shutter speed, such as 1/2000, to the slowest, such as 4. All of these exposure images will be later compiled into one image that more accurately captures all of the lighting information of the environment (pp 20). Below is a cropped example of photographs of the reflective sphere taken through a range of exposures.

Figure 4



1/2000 1/1000 1/500 1/250 1/125 1/60 1/30 1/15 1/8 1/4 1/2 1 2 4

The full exposure range of photographs of the reflective sphere in the lobby

Matching the Angles

As both angles' range of images are going to be compiled together it is very important that each angle's images were taken from the same distance from the sphere with the same settings. A very simple way to do this it to tie a string to the base of the sphere. Pull this string back to where your first angle is and when it reaches the base of the tripod either mark or cut the string there. When you are ready to shoot your second angle use the string to pivot around the base and set the tripod at the same point on the string at the other angle. Make sure you take a few photos at the center line which you can use later for compositing (pp 36).

Figure 5

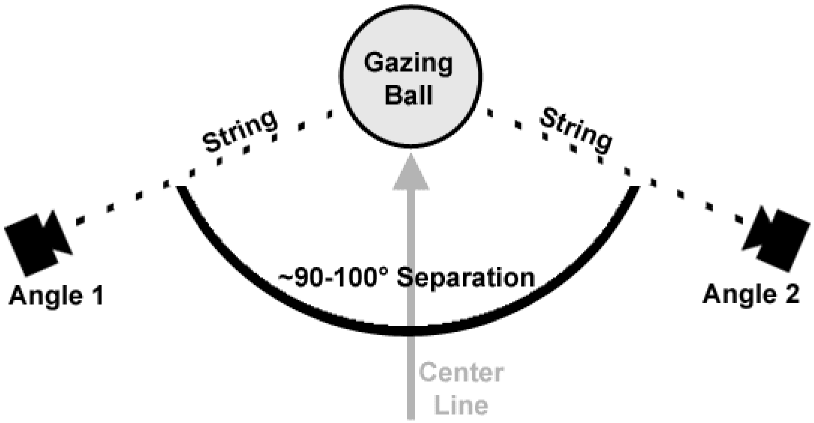


Illustration of Shooting two angles while pivoting from a center line with a string tied to the base of the Gazing Ball

Assembling the images

Now that all the photographs are taken we need to assemble them so that we can use them in our 3D and Rendering software. To do this we need two pieces of software; Adobe Photoshop (or something similar such as Gimp) and HDRShop available free for noncommercial use.

Cropping and Masking the images

We only need the sphere for our information so we need to crop the rest out and mask everything but the sphere. To do this drag a circle marquee around the sphere in the photograph and select the inverse and fill that in with total black. All of the photographs should have the sphere in an identical position in the image. Given this, it is possible to simply copy the black filled area to each image file. Once the sphere is cropped out in every image, save the images in a name you will recognize, i.e. angle1_1_2000.jpg, in JPEG format.

Figure 6

Image after sphere isolated from background. Blue lines indicate area to be cropped

See a Video of this Step

Large Medium [MPEG Medium]

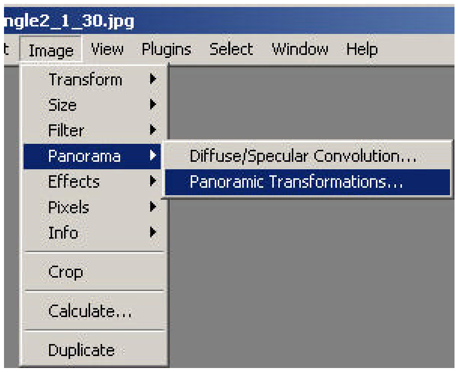
Converting from Angular to Latitude-Longitude

A perfect sphere's image can be transformed into several other formats since we know the equations of a perfect sphere. The most convenient way to do this is in HDRShop. We are going to convert angle one's images to Latitude-Longitude, or planar, format directly. The other set we are going to do the same process for angle 2 but offset by 90° so that we can composite the two together (pp 41).

For *Angle 2*:

- Import your *Angle 2* images in HDRShop
- For <u>each</u> *Angle 2* image click **Image - Panorama - Panoramic Transformations**

Figure 7

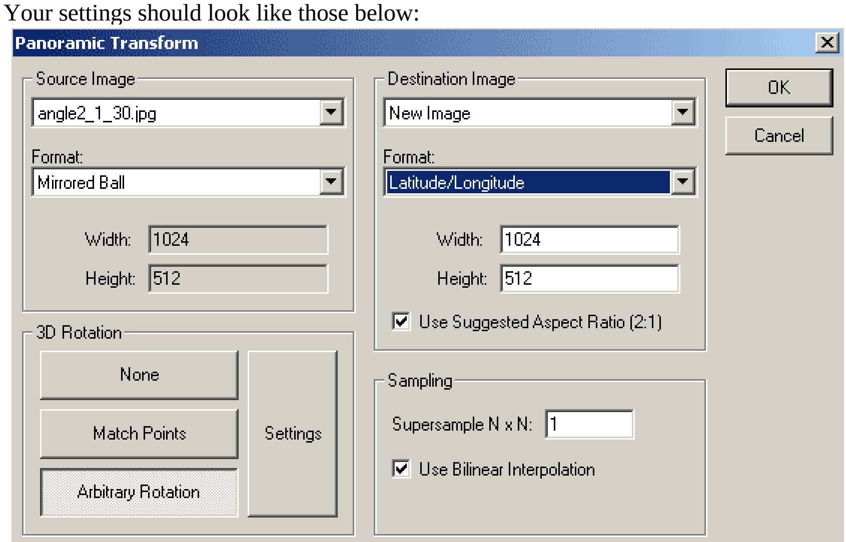


Panoramic Transformation in HDRShop

- Leave format as mirrored ball, change destination format to latitude/longitude
- In the 3d rotation section, click **Arbitrary Rotation** and then click **Settings**
- Set **Y axis** to **90**, leave the others at **0**
- Press **OK** on both windows

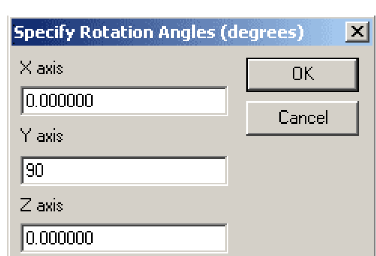
Figure 8

Your settings should look like those below:



Panoramic Transformation Settings for Angle 2

Figure 9



Arbitrary Rotation Settings for Angle 2

For *Angle 1*:

- Repeat the same steps as above but set your Arbitrary Rotation to 0 for all axises

See a Video of this Step

Large Medium MPEG Medium

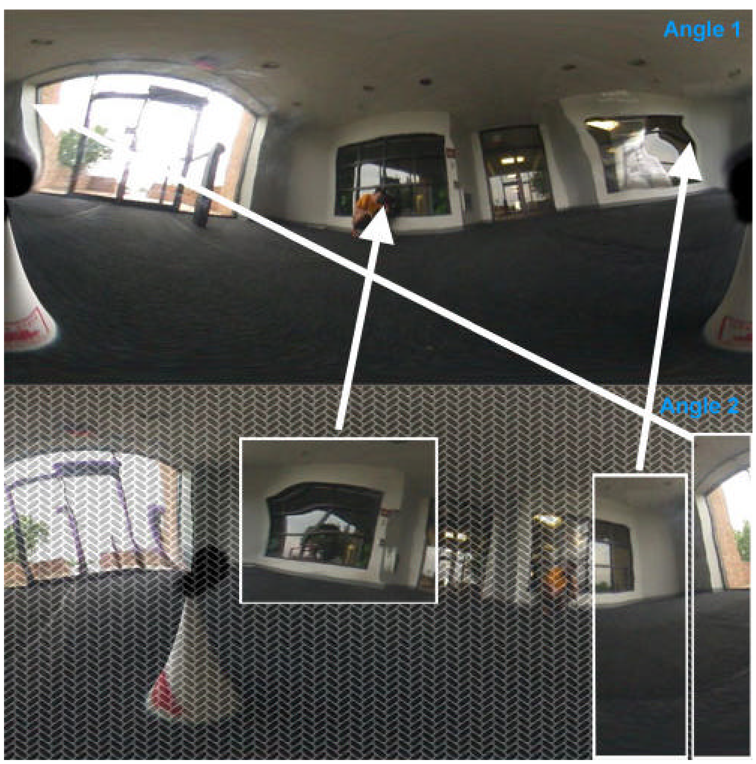
Compositing the two sets of images together.

This next step is perhaps the longest process but it is necessary to create a very nice lightprobe. I used Adobe Photoshop for this process but any similar program that supports multiple layers and layer locking should work.

Bring each exposure of angle 1 into Photoshop and copy each image into a new Photoshop file, labeling the layers accordingly. Repeat the same process for Angle 2 but places those images in a seperate Photoshop file. Bring in your complete image range for *Angle 1* into a Photoshop file with a layer for each step in the range. Repeat the same for *Angle 2*.

Now look at the areas of *Angle 1* that need to be removed. This is usually the camera, cameraman, whatever is holding up the reflective sphere, and any black areas (which is where you didn't capture any information in the photographs) You need to find those areas in *Angle 2* and copy them to *Angle 1* and transform them into the right location and rotation (Figure 10). The reason for doing this in Photoshop with multiple layers is that you can lock your layers together after you paste in each part. This allows everything you do to one layer to affect all the other linked layers. Below are the angles shot from the steps above and the final image composited together. Since this image will be just used for illumination purposes the images do not have to look precisely accurate as long as the local area of colors is still intact. For these images below you will notice the indoor window's edge in the middle of the image doesn't match properly but it will be enough for a IBI file. Once all the angles have been combined to create a better image, save out each exposure into its own JPEG file and once again name it accordingly (pp 40).

Figure 10



Angle 1 (top) and Angle 2 (bottom) after being converted to Lat/Long. Boxes indicate areas used from Angle 2 to fix and complete Angle 1.

Figure 11

Lat/Long image after compositing. There are still some minor differences in the corrected areas but since this will get blurred at rendering it isn't critical.

See a Video of this Step

Large Medium MPEG Medium

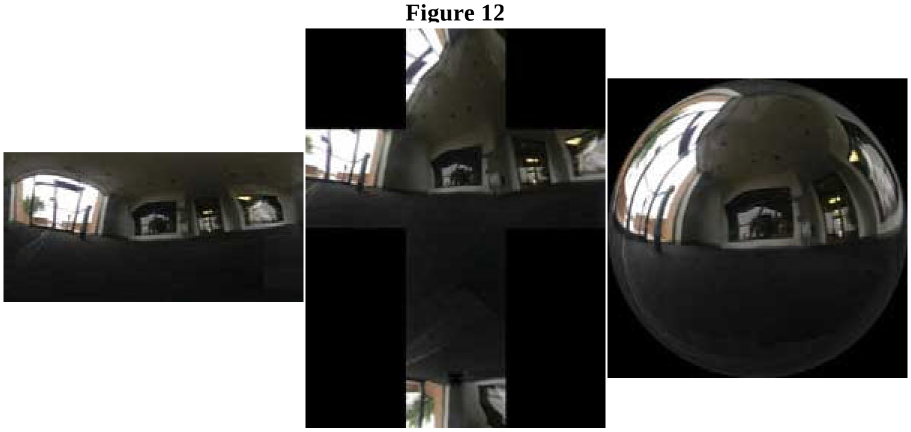
Creating a High Dynamic Range Image (HDRI)

The best application I have found for creating HDRI is HDRShop, but there are other utilities available such as MkHDR a command line utility, and HDRIE and Photogenics HDR which are both commercial applications with some minor image editing capabilities. For our purposes we will be continuing to use HDRShop (pp 41-42).

Creating an HDRI

- Launch HDRShop
- From the **Create** menu choose **Assemble HDR from Image Sequence**
- Click **Load Images** and load in your newly created range of fixed JPEG (.jpg) images.
- Your images should automatically be sorted correctly, if not then manually adjust their order so that the exposures are in the correct order.
- Click the **Calculate** button so it automatically adjusts the exposure difference
- Once that is complete click **Generate Image** and you will have a HDRI
- If your new HDR image seems dark then use the + or - to find what should be a new middle ground for your exposure range. Once this is found go to **Image** - **Pixels** - **Scale to current Exposure** and the HDR image will be adjusted.
- If needed you can now convert your image to the type of map you need by going to **Image - Panorama - Panoramic Transformations** and select the appropriate pull down menus. It is possible to create a Latitude/Longitude, Cubic, and Angular maps (Figure 12). For this tutorial we will be using a Latitude/Longitude map so create one of those.
- Save a Latitude/Longitude HDRI in HDR format and a compressed HDR Iin JPEG format.

Figure 12



Latitude/Longitude Map of Composited image

Cubic Map of Composited image

Angular Map of Composited image

How Image Based Illumination works

Principles

The basic principle behind Image Based Illumination is that a computer generated model is surrounded by another piece of geometry (i.e. sphere, cube), with an image mapped onto that geometry. To calculate the lighting of a particular point of the computer genereated model, IBI takes extends the rays, usually based on the vector of the normal (N), until it intersects with the surrounding surface. It then samples the color at that particular point on that surrounding surface's mapped image and uses that for the lighting color of the point where the vector started. Below is an example of this occuring with several rays.

Figure 13

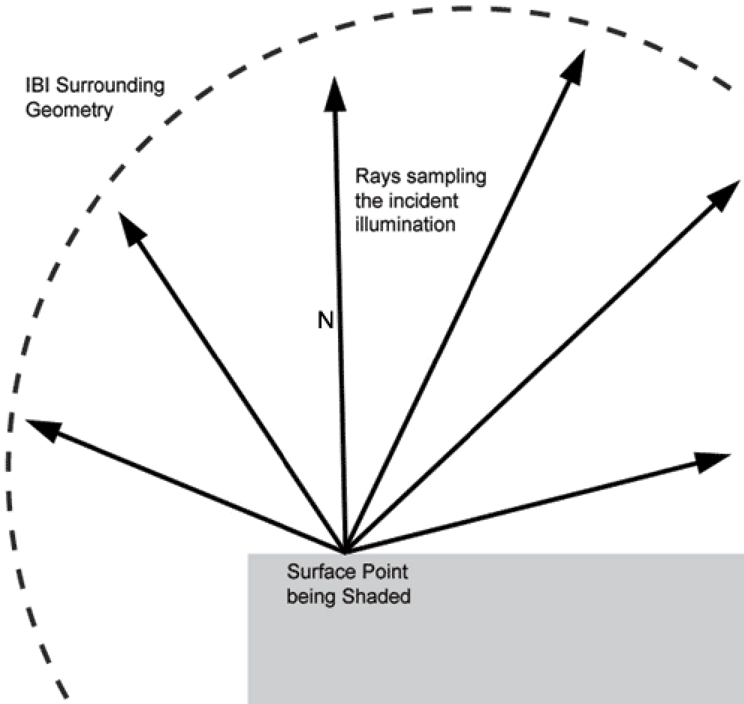
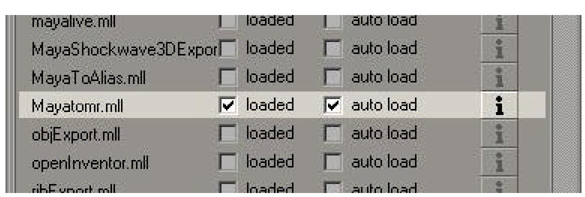


Illustration of the principle behind Image Based Illumination

In Maya and Mental Ray

For Mental Ray, which comes bundled in with Maya 5.0, IBI is implemented through creating some surrounding geometry and using the Mental Ray Final Gathering technique. This is the most straight forward method and what will be used in this tutorial. There are some scripts I have written or modified for this tutorial which have been written for Maya 5.0 but should be useable in any later versions as long as Mental Ray is installed and the plug-in is loaded as shown below. The plug-in can be loaded by going through **Windows** -**Settings** - **Plug-in Manager** and checking load box for the **Mayatomr.mll** plugin.

Figure 14



Mental Ray Plug-in

Setting Up for IBI

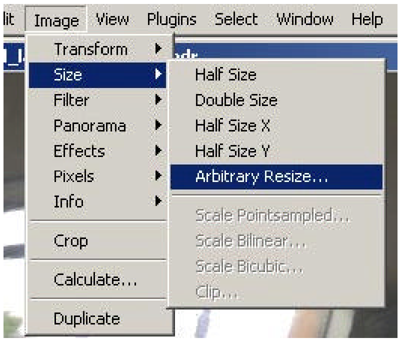
More work in HDRShop

There are two more steps in HDRShop that we must complete before moving onto Maya. The first is to open your compiled HDR image and set the exposure to 0 by pressing **0**. If the image appears dark this is probably due to a Gamma difference from the camera to the application. We will solve this through a script in Maya but it is important to note whether the image is bright or dark at 0 exposure. The second step is to use the Lightgen plug-in for HDRShop on our HDR image. Make sure the plug-in is in the HDRShop plug-in directory (pp 44-45).

Lightgen

- Launch HDRShop
- Open your composited HDR image
- Blur the image by going to **Image** - **Filter** - **Guassian Blur**
- The image will need to be shrinked for Lightgen so from the **Image** menu choose **Arbitrary Resize**

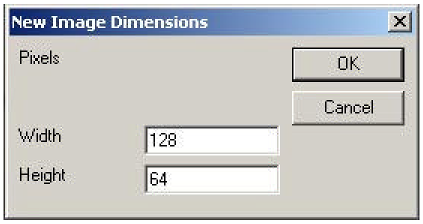
Figure 15



Arbitrary Resize in HDRShop

- Enter **128** and **64** for the new size

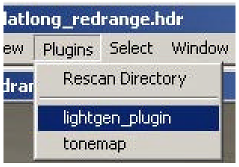
Figure 16



New Image Resize numbers for Lightgen

- Once the image is resized go to the **Plugins** menu choose **Lightgen**

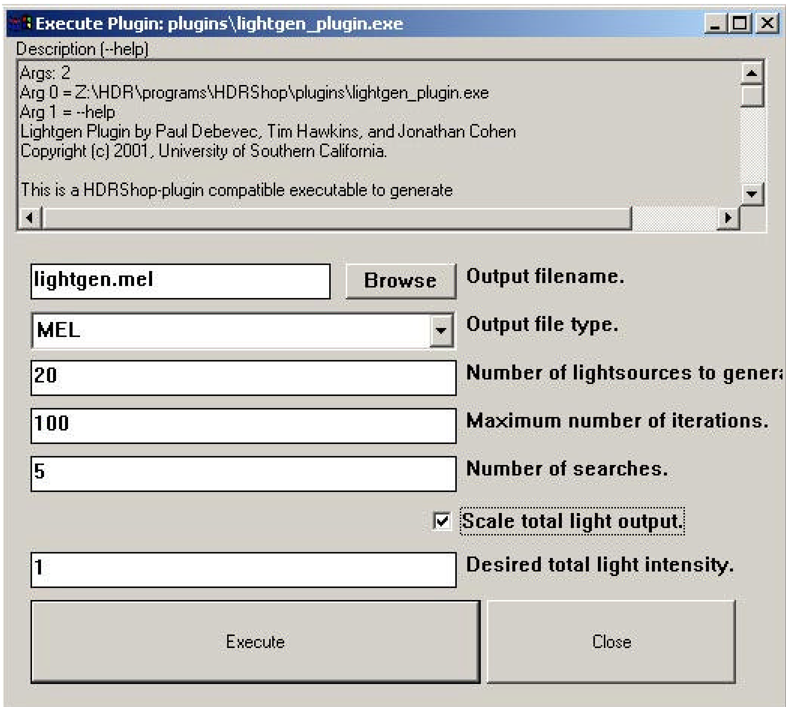
Figure 17



Launching the Lightgen plugin

- Enter an **Output filename** (Browse to your desired location) and set the rest of the settings to the image below

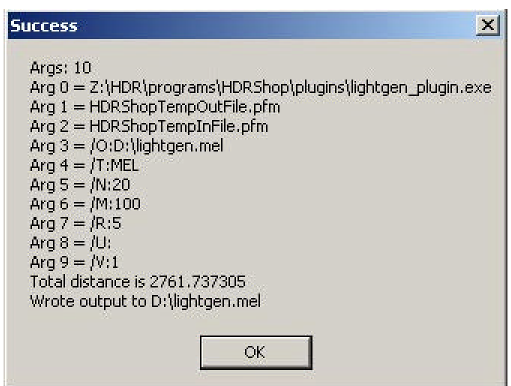
Figure 18



Lightgen GUI plugin in HDRShop

- Click **Execute** and Lightgen will think for a short bit as it is creating the output
- If the plug-in was successful you should receive the following message

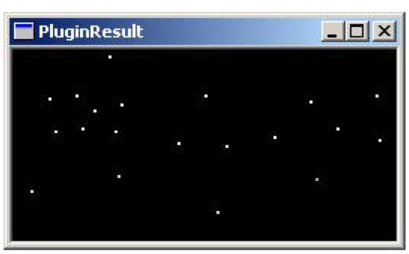
Figure 19



Successful output in Lightgen

- The final result should look like a bunch of colored dots in a black field

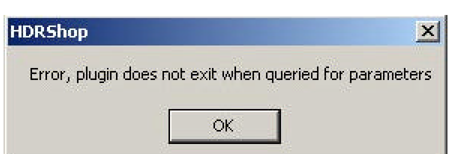
Figure 20



Visual output from Lightgen

- If you receive the following error it is most likely you forgot to resize your image before launching Lightgen

Figure 21



Lightgen plugin error in HDRshop

- Lightgen is also useable through the command line. Type **Lightgen.exe** to list the different flags

Making a MAP image

Mental Ray's native image file format is MAP. We are going to convert our HDR image to MAP to speed up render times in Mental Ray and to eliminate the Range Clamping that Maya does with HDR images.

hdr2map

- Open a command shell
- Navigate to where your HDR file is located
- Use my hdr2map batch file to convert your map
- Useage: **hdr2map filename.hdr filename.map**
- A MAP file will be created in that directory for you to use

Working in Maya

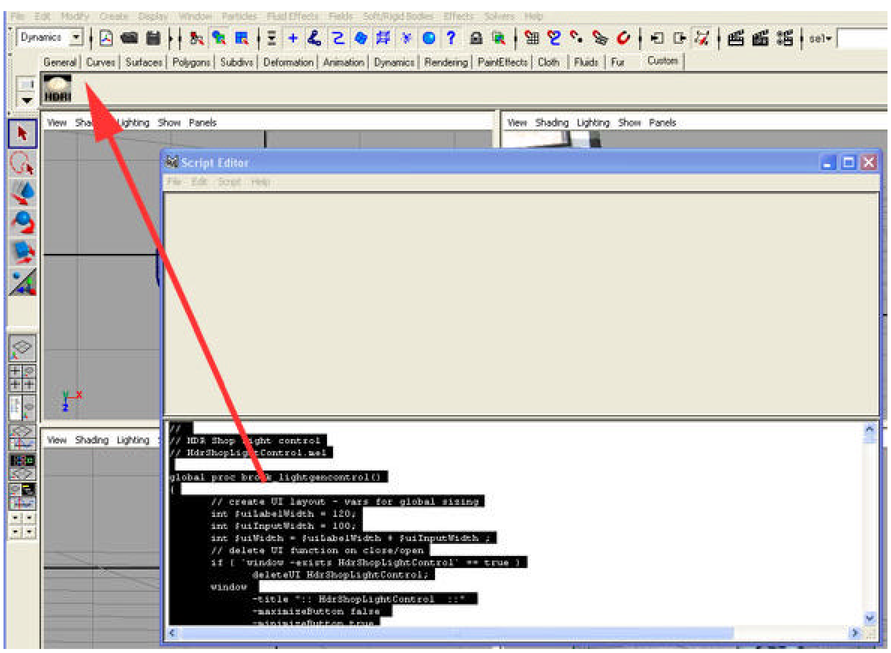
Basic Workflow

There are four basic steps in this pipeline for using Image Based Illumination and High Dynamic Range Imagery. The first is using the HDR image to provide the basic low level lighting, such as ambience and incandescence. If the object is reflective it can also be mapped into the specular channels. The second step is to use the HDR image to derive actual lights in Maya and to set them up so that they are convincing. The third step is specific to any special effects or elements the model may have. This may include Caustics, Fog, Sub-Surface Scattering and so on. The final step is to "dirty up" the image so it doesn't look so real and can blend back in with the live action image. The two scripts available in this tutorial greatly speed up this process. It should be noted that throughout this process it will be neccessary to move back and forth between the different stages to produce a great result.

Installing Scripts

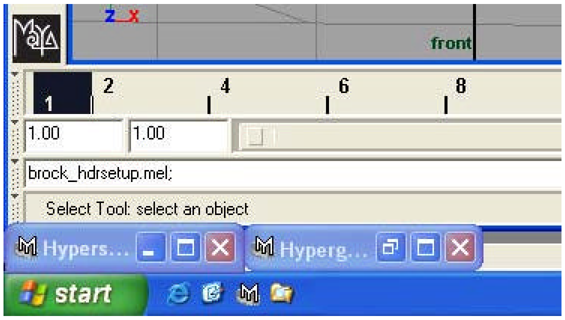
Both scripts should be downloaded to your Maya's script directory. I recommend making shelf buttons for both scripts so that they can be easily accessed. In Maya open the Script editor and inside the script editor Open the brock_lightgen.mel script. The contents of the script will be displayed in the Script Editor window. Highlight the all of the text and drag, using the middle mouse button, to the Shelf and release. A new icon will be created that will launch Brock's Lightgen Control script. For Brock's HDRSetup Control script type brock_hdrsetup.mel into the command line and drag that text to the Shelf to create a new icon.

Figure 22



Placing brock_lightgenconrol.mel into the Shelf

Figure 23

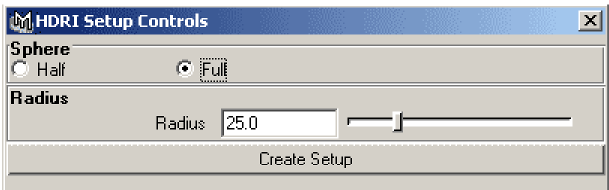


Command Line entering of brock_hdrsetup.mel

Step 1: Using just HDR through IBI for general lighting

- Launch Maya
- Open a blank scene with just your model (or import your model into a new scene) and place the model as close to 0,0,0 as possible
- Load **Brock's HDRSetup** script
- A Create HDRSetup window will appear. Size the Sphere so that it will completely surround your object but not so that it is so far away. The closer it is the less the render time can be with Final Gathering. We also want a **Full Sphere** for our setup.

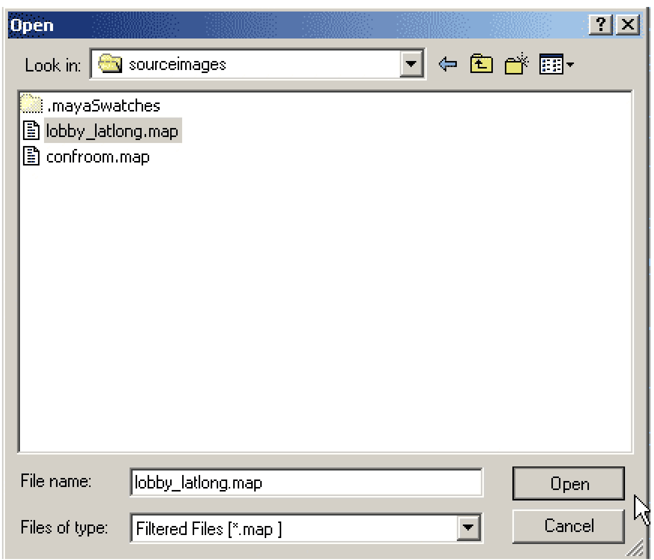
Figure 24



IBI Sphere setup

- After the Sphere is created you will be prompted with a new HDRI Setup menu
- Click the **Browse** button and locate your MAP file. I recommend placing this MAP file in your sourceimages directory of the active project in case you are using a distributed renderer.

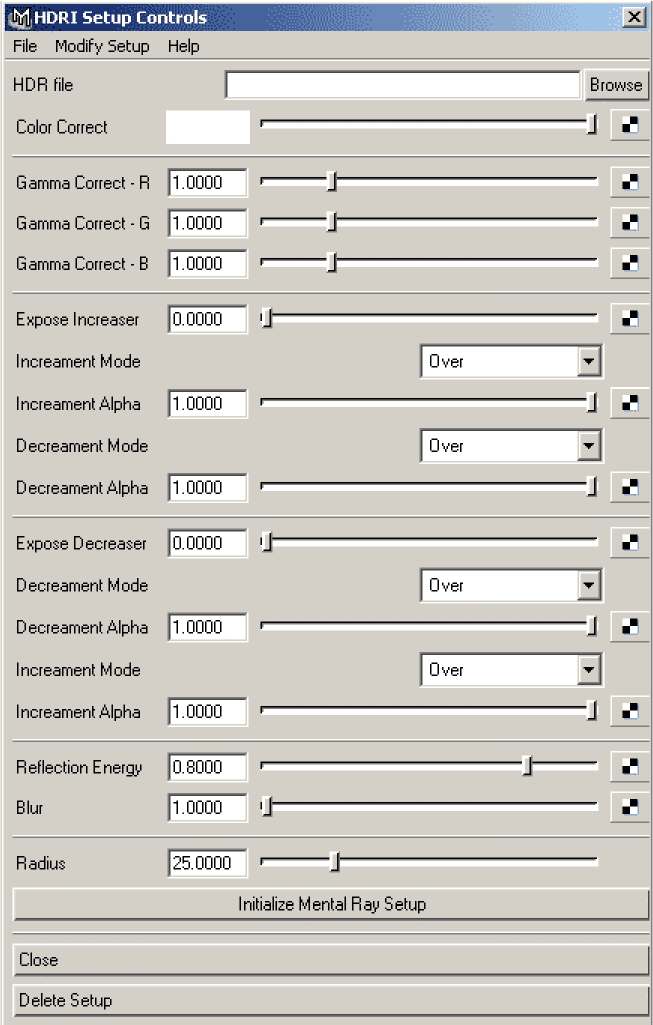
Figure 25



Loading a MAP image

- After the file has been loaded you can now start affecting the MAP image through the HDRI Setup Controls panel

Figure 26



Brock's HDRI Setup Script

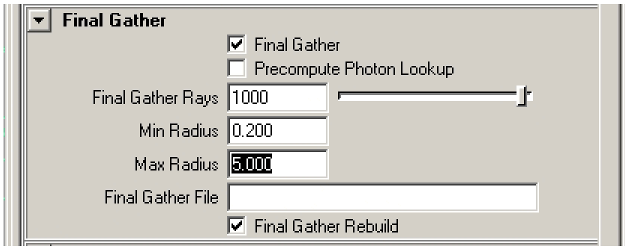
- Remember earlier when I mentioned to jot down what the gamma is of your image/camera. Enter that number in the **Gamma Correct** boxes. If needed you can skew the gamma to one channel or another since the correction is broken down to Red, Green, and Blue.
- The **Expose Increaser** and **Expose Decreaser** sections allow you to fake the exposure difference in the HDR image. This normally doesn't need to be touched unless you are trying to give a brighter or darker look then the original image.
- **Reflection Energy** controls how much the surrounding IBI geometry will continue to reflect back the image, the default value is usually fine.
- Before rendering the image should be blurred slightly since we are just using this image to generate the fill light for the object. In the **Blur** slider, a setting of1 results in no effect.
- If you want to resize the IBI sphere, use the **Radius** slider
- If you render now, you should receive a blank scene. This is because we need to setup Mental Ray accordingly.

See a Video of this Step

Large Medium MPEG Medium

- Click the **Initialize Mental Ray Setup** button
- Open up the **Render Globals** and click on the **Mental Ray** tab
- The **Final Gather** tab should already be opened

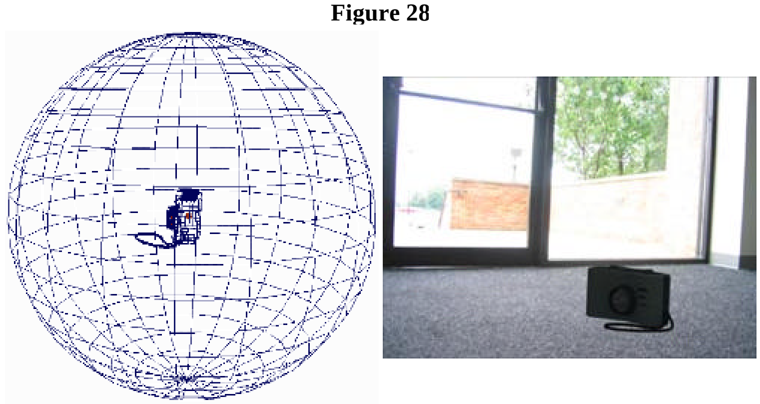
Figure 27



Final Gather settings in Mental Ray

- These values have just been initialized through the HDR Setup script. If you render now you will actually see your model being lit. However, the **Max Radius** needs to be tweaked to be just a bit larger then the radius of your surrounding sphere. The **Final Gather Rays** can also be increased to produce a better result, but that will also increase the render time.
- Render the image and continue to adjust these various settings until you have a good amount of fill lighting such as the image below
- In order to provide a visual example of where the placement of the image on the surrounding image is the MAP file must be replaced with a JPEG file. Launch the HDRSetup Control script again and open the JPEG file. Before rendering, the MAP file should be loaded back in.

Figure 28

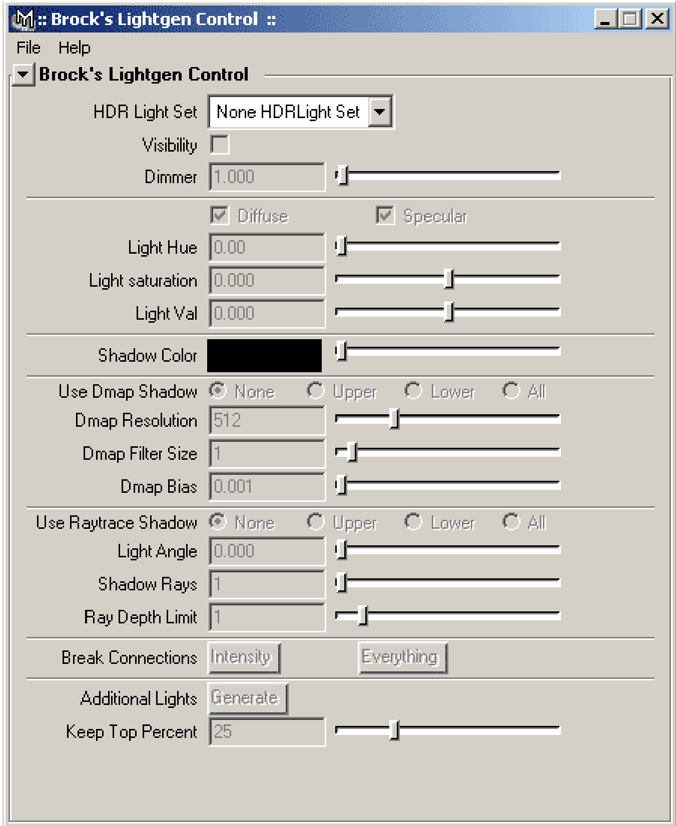

First step in Maya (left) and the resulting rendered image (right)

Step 2: Using the HDR image to generate lights to create lights

At this stage we will be deriving actual Maya lights from the HDRI image through the use of the exported Lightgen file that was created earlier (pp 56).

- Load **Brock's Lightgen Control** script
- This will bring up a panel which allows you to import your lightgen outputted MEL file

Figure 29



Brock's Lightgen Control at load up

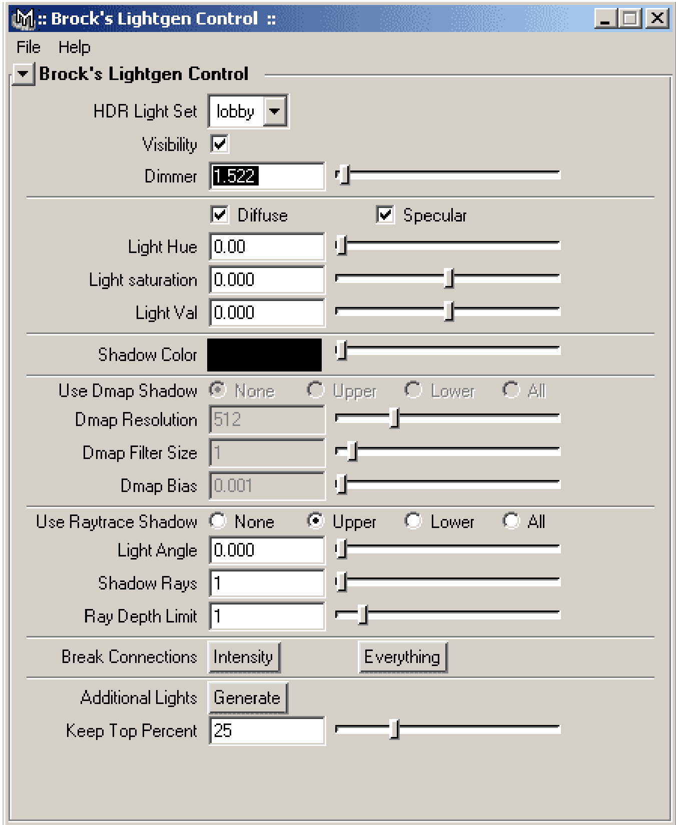
- From the **File** Menu choose **Lightgen Script Open** and open the MEL file you created in Lightgen.

Figure 30

Open the lightgen script

- After the script is opened you will now be able to control the directional lights created from the script.

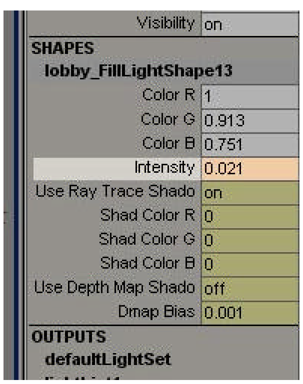
Figure 31



Brock's Lightgen Control after MEL script loaded in

- It is also possible to adjust the Light Hue, Saturation, and Value for all the lights.
- Shadows may be enabled from the Upper, Lower, or all of the Lights in Dmap or Raytrace
- Once the lights are looking good it is important to break the Intensity so that each light may be tweaked individualy. Without breaking the intensity the values are locked to the dimmer

Figure 32



Locked intensities

- It is also possible to break all the other connections to the lights so that the shadows may be tweaked individually too.

Figure 33

Breaking connections

- Continue to modify the lights so that they provide good lighting for the model as shown below.

Figure 34

Second step in Maya (left) and the resulting rendered image (right)

Step 3: Creating scene specific elements

For some CG mock-ups there may be "special" lighting situations that need to be created. Examples include caustics and fog. If this step is not required for your CG mock-up, you may skip this step (pp 57).

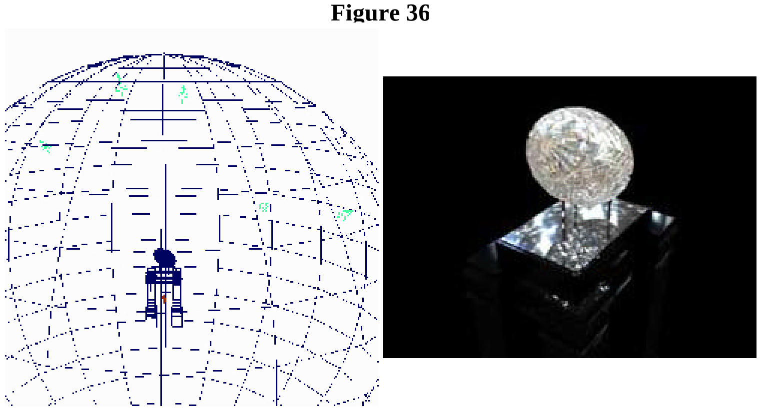
- The Brock's Lightgen Control script also allows the input of an additional Lightgen MEL file

Figure 35

Generating additional lights

- Set the percentage first of what amount of lights to keep then click Generate and browse to the appropriate file.
- Once the top percentage of the lights have been selected then enable light specific items for those lights
- Below is an example image of an object that would require special lighting. On the left is the Maya setup and on the right is the image rendered in Maya with just a caustic pass.

Figure 36



Third step in Maya (left) and the resulting rendered image (right)

See a Video of this Step

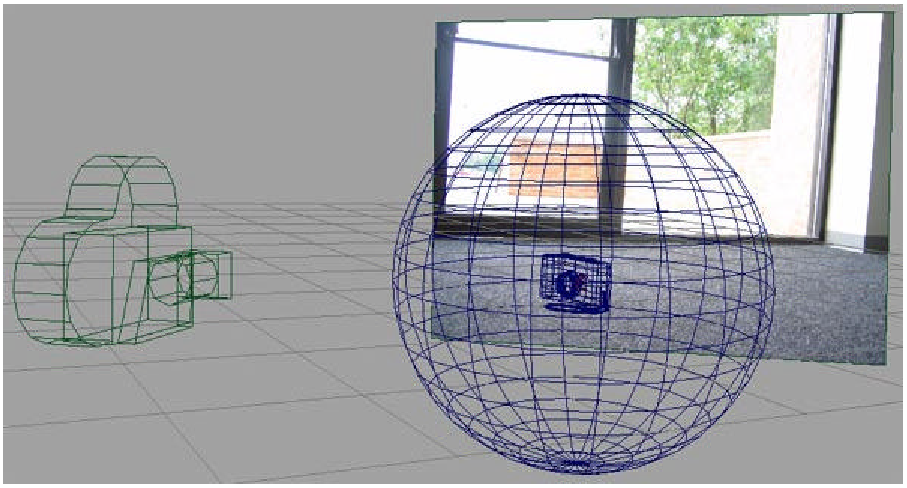
Large Medium [MPEG](#) Medium

Advanced Use: Matching a specific angle and creating shadows

In order to make the final result look as convincing as possible there are two additional advanced used stages that can help blend a CG mock-up into the live action photograph.

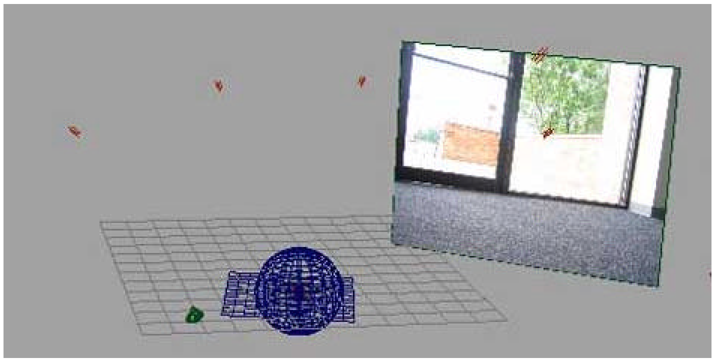
- Any camera in Maya can have an image plane attached to it. By mapping in the desired background image into this image plane it is possible to interactively place the camera an image. This is valuable for making sure the camera's angle matches that of the real camera.

Figure 37


A scene in Maya with an the background image plane attached to the camera

- For shadows you need to create a ground plane that matches the one in the photograph. By renderning in seperate layers we can use the difference of the ground plane with the ground plane with shadows to extract just the shadow information. This can be done easily using the Difference layer in Photoshop.

Figure 38


A scene in Maya with an the background image plane attached to the camera and a ground plane to reveive shadows

By using these advanced steps you are able to create an even more realistic looking composite. Keep in mind at this point the lights have not been manipulated in position or placement at all from the default.

Figure 39

Rendered and composited image with all the stages of this pipeline plus shadows

Step 4: Dirtying up the image

- At this point you should be able to produce a pretty convincing high quality render. However, the image will still need to be composited into the background and "dirtied up" to make it look convincing. Above is how the image looks with all of the elements combined before dirtying up the image.
- Launch **Photoshop** and open your image(s)
- Blur and Noise work well for dirtying up the image
- Render out as much as possible in layers so that it can be composited separately
- Below is an example of this pipeline with the cg mock-up camera composited on top of the background image.

Figure 40

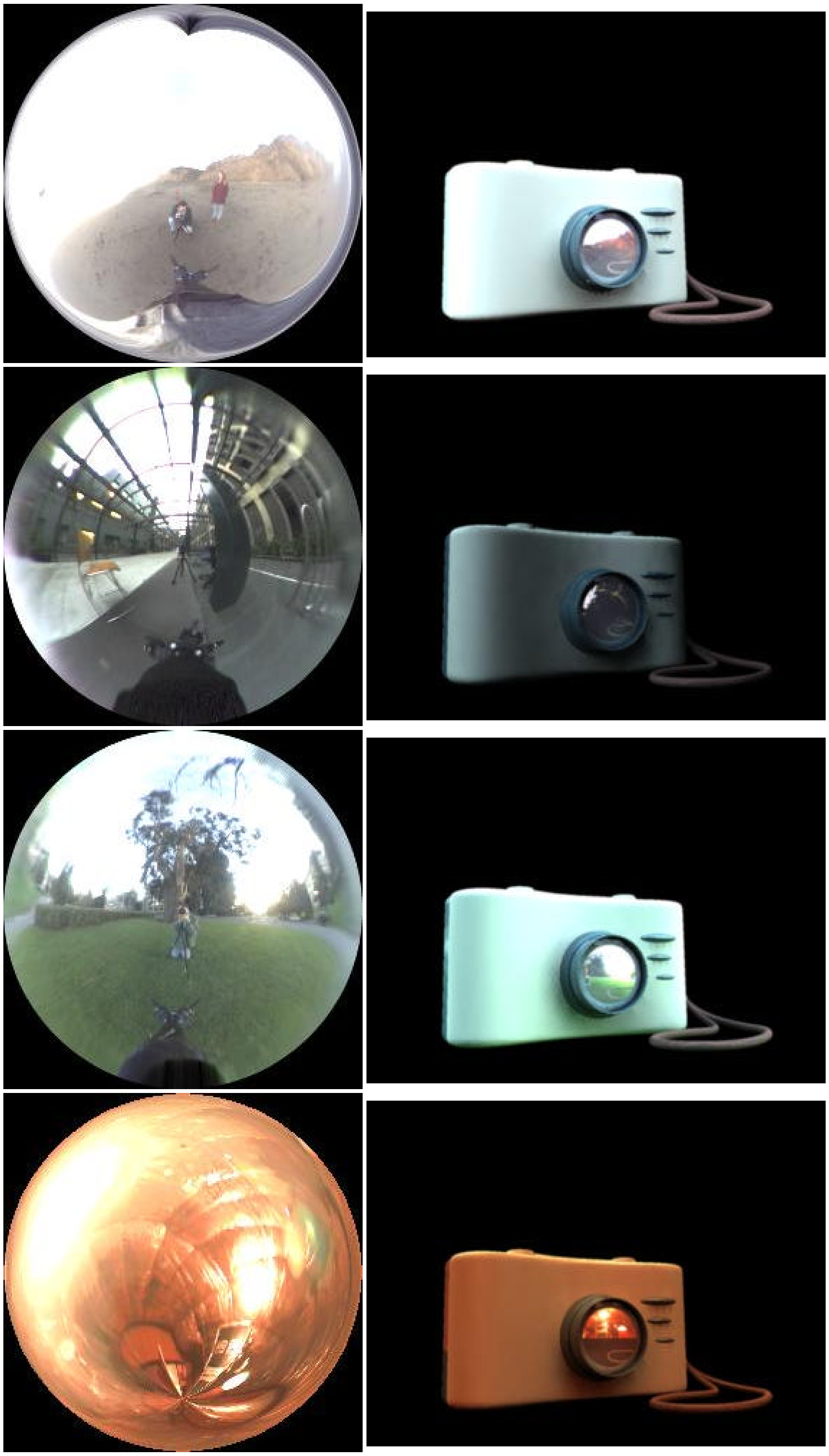
Rendered and composited image with all the stages of this pipeline plus shadows plus noise and blur

As with any lighting setup there needs to be constant tweaking in order to produce the best results. It should be clear that using just

IBI alone is not a very effective solution, but by stepping though this pipeline it is possible to generate an accurate rendered image from using the HDRI file as the base for all of the different lighting stages.

Figure 41

Below are some examples of the camera being lit through different lightprobes (left) that are HDR images in angular format. The lighting is just using the first stage of this pipeline to visualize the different lighting aspects possible



My Related Links:

[MFA Thesis Writing]() - the down and dirty parts of this pipeline

Thesis Production - a HTML page that describes parts of my thesis work over the past year in more detail

My Created Lightprobes

Accumulated Library of Lightprobes (Mine and Downloaded)

Other Links:

Brent Watkins - the person responsible for helping me with my scripting difficulties

Paul Debevec's Homepage - The creator and probably the largest promoter of HDRI

HDRShop - Debevec's free HDR creation program

Debevec GDC 2002 - powerpoint presentation of Debevec's 2002 Game Developers Conference course

Debevec GDC 2003 - powerpoint presentation of Debevec's 2003 Game Developers Conference course

Debevec SIGGRAPH 2001 - Debevec's Image Based Lighting 2001 SIGGRAPH course

Debevec SIGGRAPH 1997 - Debevec's Recovering Range Radiance Maps from Photographs 1997 SIGGRAPH course

Debevec SIGGRAPH 1998 - Debevec's Reendering Synthetic Objects into Real Scenes 1998 SIGGRAPH course

Jonathan Cohen - Developer of Lightgen

Simon Bunker - a rendering guru with some excellent work and descriptions of HDRI

CGTechniques.com - A site with several tutorials regarding HDRI

OpenEXR - An open source format from ILM that supports HDR information

HDRIE - A free HDR image reader and editor from the University of Illinois

ATI - Rendering with Natural Light in Real Time - a real time example of Debevec's *Rendering with Natural Light*

rthdribl DX9 - Real Time High Dynamic Range Image-Based Lighting in Direct X 9