Lighting Objects on Photographs

Gergely Vass

Dept. of Control Engineering and Information Technology, Technical University of Budapest, Hungary

---

Abstract

In this paper a simple image-based method is introduced which makes possible the re-lighting of objects on real photographs. To do this the normal vector of each surface element is obtained from several still images. These pictures should be taken in special lighting conditions, which are described in the paper.

---

1. Introduction

It is a common need among 3D graphic artists to modify or tweak the rendered image without re-generating it. Being able to do this the process of the lighting setup is very straightforward. This need can be easily fulfilled using any current 3D package since almost all of they offer some sort of “real time rendering” solution. This is realized by writing out some extra information - needed to do the rendering - to a file. These information are usually the depth channel, the motion vectors of the image, or the normal vector distribution. Using these data the re-rendering is very simple.

Today computer graphics and digital image processing is a key element in the postproduction. It is a great help for the directors and cinematographers to be able to modify the shot material in post since the re-shooting is always expensive and time consuming. Unfortunately the pictures taken by a conventional camera have no extra information, which could help the software to modify the images. In this paper a simple method is introduced, which makes possible the re-lighting of real objects on still images.

2. Illumination of Surfaces

The main goal of photorealistic image synthesis is to generate images similar to real photographs. Today many production proven algorithms are known, which simulate the behavior of light very well. These methods are used to make “photographs” of virtual 3D objects. The two main tasks of these algorithms are:

- The lighting conditions of the virtual environment should be calculated.
- Knowing the incoming light the color of the surface elements should be evaluated.

Since we want to illuminate — or re-color — the objects in post we have to deal with the two problems above. We have to define the new lighting conditions and the new color of the pixels under this new environment.

2.1. New lighting Condition

The calculation of any realistic lighting condition is very hard when all intersurface light reflection is considered. This kind of calculation is called global illumination. Since this paper concentrates on the illumination of single objects the accuracy of global illumination not needed. In order to achieve a fast and simple solution no intersurface light reflection is considered and the light sources are supposed to be directional lights. This kind of simplification — called local illumination — does not affect the quality of the solution much.

2.2. Illumination

Knowing the lighting conditions the color of the current surface element can be evaluated. Depending on the available data there are two possible ways to do this step.

One possible image-based method is proposed by Paul Debevec[3]. Using a special mechanism — equipped with a directional light source — a human sized object can be lit from arbitrary direction. A high-speed camera takes pictures from this still object lit from every possible angle. During this process a large database is built, which is later used as a lookup table to shade the object. This method requires a high precision hardware to be built thus it is not a simple, nor a cheap solution.

If there is no measured data present the pixel’s color should be evaluated using mathematical and physical relations. This approach is the way the traditional rendering programs work. To calculate the incoming radiance to tha camera from the surface — which is the physical equivalent of what we call "color" — the rendering equation[1] should be evaluated. Assuming that only one directional light is used to illuminate the surfaces the rendering equation becomes very simple:

$$L(\vec{x},\omega) = L_{light} \cdot f_r(\omega',\vec{x},\omega) \cdot \cos\theta' \tag{1}$$

where $L()$ sands for the incoming radiance to the camera, L_{light} is the intensity of the light, f_r is the bi-directional radiance distribution function describing the light reflection properties of any material, and θ' is the angle between the light direction and the normal vector of the surface (figure 1).

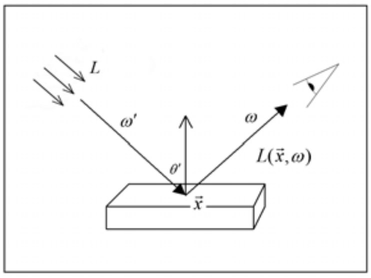

Figure 1: *The notation of the rendering equation.*

The BRDF tells what proportion of the incoming light is reflected towards the outgoing direction. In the field of computer graphics several methods are known to define an arbitrary BRDF. There are some techniques, which make possible the measuring of the BRDF of the objects visible on photographs[4]. However, this method requires the exact virtual model of the scene thus it does not fit our needs. The most widespread way to define a BRDF is to use simple mathematical formulas, which describe a behavior similar to real materials. These formulas are called empirical illumination models[5]. To illuminate the objects in the project two simple and well known models were used: the Lambert and the Phong models.

2.2.1. Lambert model

The Lambert model describes diffuse behavior based on the Lambert rule. In this case the reflected radiance is independent from the viewing direction.

$$f_r(\omega',\vec{x},\omega) = k \tag{2}$$

To evaluate the Lambert illumination model only the user given constants and the angle between the light's vector and the normal vector should be calculated.

2.2.2. Phong model

The Lambert model does not handle specular reflection thus a different model, the Phong model should be used to deal with shiny surfaces. This illumination model can be evaluated and implemented in a simple way:

$$f_r(\omega',\vec{x},\omega) = k_s \cdot \frac{(\vec{R}\cdot\vec{\omega})^n}{\cos\theta'} \tag{3}$$

where k_s and n control the shininess, $\vec{V}$ is the viewing vector, $\vec{R}$ is the reflection of the light vector on the normal vector. The equation shows that the evaluation of the Phong model requires the user given constants, the normal vector, the lighting vector and the viewing vector.

3. Required Data

The use of the Phong and the Lambert model requires data of the object to be lit that is not present in the source photographs. The user should define these data or it should be extracted from the input images using some image-based method.

3.1. lighting direction

Since the local illumination model is used the only light present is user defined and no reflections are considered. This light is supposed to be only one directional light, which is given by a directional vector.

3.2. Viewing direction

It is almost impossible to extract any information from still images regarding to the viewing angle and lens distortion of the camera. Because of that the viewing direction at each pixel is hard to determine. The goal is to specify a direction from some virtual camera to each pixel, which matches the original viewing direction.

For the sake of easy implementation and fast calculation the camera is supposed to do non-distorting perspective projection. The chosen position of the virtual camera - which can be changed freely - is seen on figure 2.

Using this assumption the viewing vector for each pixel can be calculated in the following form:

$$\omega = (x,y,0) - (0.5,0.5,1) \tag{4}$$

It is obvious that no real camera works this way, however, the tests have proved that this model can be used for the calculations.

3.3. The normal vector of the surface

The hardest task is to determine the normal vector for each pixel. If the user had to figure out this kind of data, he or she

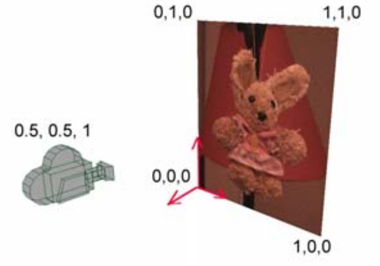

Figure 2: *The camera position and the coordinate system used for the calculations.*

had to create the exact geometric model of the object on the photograph. The 3D graphics packages can than export the normal vector distribution using the modeled stand-in object. This method cannot be used always, however, since it requires the model to match the real object perfectly. The other way would be to use an image-based method, since it extracts the information regarding the normal vector distribution from images. We know that the normal vectors and the geometry of the object have great correlation, thus any of them can be used to calculate the illumination in post for the objects.

- If the *geometry* of the surface is acquired the normal vector can be calculated easily from the partial derivatives. This way of figuring out the normals has many advantages, however, the actual geometrical information cannot be extracted from still images. All the image-based modeling methods need image sequences, stereo pictures or several still photographs taken from different directions to work effectively. Unfortunately a still image — or more images taken from the emphsame direction — do not have enough information to build the exact “virtual replica” of the object. In order to propose a very simple method — which would work with simple photos — this approach was not taken into consideration.
- To figure out the *normal vector distribution* directly from the image it’s information content should be analyzed. The main factors affecting each pixel’s intensities ($I_{x,y}$) are the following:

  $$I_{x,y} = f(\vec{N}_{x,y}, \omega', \omega_{x,y}, f_{r_{x,y}}, L) \tag{5}$$

  Considering the unknown relationship above — in which only one, directional light is assumed — the color information in the picture depends on — at least — five factors: the normal vector , the light vector, the viewing vector, the BRDF and the light intensity. The goal is to make pictures under such conditions that the pixel’s color depends on *only* the unknown normal vector, all other parameters are known:

  $$I_{x,y} = f(\vec{N}_{x,y}) \tag{6}$$

  so that the inverse function exists

  $$f^{-1}(I_{x,y}) = \vec{N}_{x,y} \tag{7}$$

  If there is such function the normal vector distribution can be evaluated from the image. Using this information the illumination of the object can be calculated in post using the Phong or Lambert model.

4. Acquiring the Normal Vectors

In order to find out a relation like 6 — where the color of the pixel depends on the normal vector only — all the other parameters in 5 (except the normal) should be fixed. This means that special conditions are required for the photographing thus for each pixel the viewing vector, the light’s vector, the BRDF and the light intensity are known.

4.1. Special Requirements

It is very hard to figure out or measure the exact BRDF of any object. Using photograps the model of the scene[4] or at least the normal vector distribution is needed to acquire this function. Since this function might be very complex - making the calculations slow, if not impossible - it is supposed that the objects we deal with are diffuse surfaces. Based on the Lambert model it is known that the color of the diffuse surfaces is independent from the viewing direction. Using this strict assumption not only the BRDF is known but the viewing direction is also eliminated from the relation as well.

The lighting direction and intensity is also known if the light source placed at some fixed position. It is assumed that only one directional light is used to light the object. This can be realized by one spot or point light placed far enough from the object. In this case the lighting intensity is almost perfectly constant all over the object. With all these restrictions only the normal vector distribution is unknown in the pictures.

The exact relation between the color of the pixels and the normal vector needs to be investigated. To be able to reconstruct the normal vector directions, this relation must be invertable. However, if no further restrictions are forced some surface elements with different normal vectors might have the same color. To have a one-to-one mapping between the color values and the normal vector some special lighting conditions should be applied.

4.2. lighting Environment

The definition of any vector in the 3D space is equivalent to the definition of its three components. It is an efficient way of decomposition to evaluate the components separately. In the coordinate system of the calculations the x, y and z axes are the horizontal, vertical and depth directions.

Let us consider the horizontal component first. According to the Lambert model the brightest point on the surface

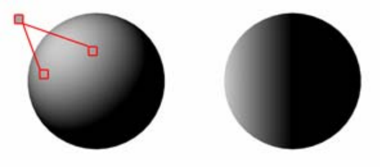

Figure 3: *If an object is lit from arbitrary angle, pixels with different $\vec{N}_x$ component may have the same color. If the light comes from the side, different color intensity means different $\vec{N}_x$.*

is where the normal vector points toward the light source. Knowing this we can avoid to have pixels with the same color but different horizontal normal vector component: if the light shines from the "left" or the "right" side the negative or the positive horizontal component of the normal vector can be evaluated (figure 3). The formal relation is derived from 1 and 2.

$$L(\vec{x},\omega) = L_{light} \cdot k \cdot \cos\theta' \tag{8}$$

Let us suppose that the light comes from the right — which is the positive x direction. In this case the angle between the normal vector and the light's vector — both of them with unit length - is equal to the angle between the normal vector and the x axle. The cosine of this angle is — surprisingly — the x component of the normal vector. Using this relation:

$$L(\vec{x},\omega) = L_{light} \cdot k \cdot N_x \tag{9}$$

This *invertable* relation shows that the intensity of any object lit from the right side is in linear relation with the horizontal component of the normal vector. The vertical component can be evaluated the same way using lighting direction from up and down. The surfaces with negative z normal vector component are certainly not visible, while the positive z component is proportional with the reflected light intensity when lit from front. With the decompositions above the equation 7 is obtained using five different and more simple relation. In practice this means that not one but five photographs will be input for the algorithm.

5. Applying the Method in Practice

Five still pictures should be made with different lighting: one from left, right, up, down and front respectively (picture 4). These five photographs can be used to get the normal vector distribution of the object. With this data the lighting of real objects in post can be realized with any empirical illumination model like Phong or Lambert. To use the proposed method no special hardware or custom software is needed. The photographing may be done with a traditional camera, a standard light bulb is fine to be the light source and the calculations can be done in any scriptable image processing or compositing software.

5.1. Taking Photographs

If it is possible to avoid any unwanted reflection — like in a professional studio — the only light source should be the one we set up. Taking five pictures with the necessary lights — and some other with other lighting conditions at will — the lighting in post can be done using the five pictures. If the environment of the shooting is not that perfect, not five but six photographs are needed. From the very same camera position one picture should be made with the natural lights and five other with the extra directional lights. Taking the difference of the first and the other pictures we get the same result if only the artificial lights were present.

If we do not want to make that many pictures colored lights may be used. Using the three color channels of digital images the effect of the three lights with different color can be separated. Using this technique the five pictures are reduced to two, since two RGB images contain six channels of information.

When taking the pictures some areas of the object will be in shadow. This makes it impossible to gain the correct information about the normal vector in those areas. Because of this the proposed method works well for objects without great grooves. However, the tests have proved that using objects with great self-shadowing the results are still acceptable.

5.2. Extracting Information

The pictures taken should be prepared before processing. The maximum intensity should be at value 1 and the darkest color intensity should be value 0. At this point via some sort of touch up the unwanted light reflections can be eliminated as well.

It is efficient to store the normal vector distribution in one image file where the color channels contain the component values. Because the image processing programs usually do not handle negative numbers the storage of the normal vector could be this:

$$(x,y,z) = \left(\frac{R+1-L}{2}, \frac{U+1-B}{2}, \frac{F+1}{2}\right) \tag{10}$$

where the x,y,z are the three color channels to be stored, L,R,U,D and F are the intensities of the pictures taken with left, right, up, down and front lighting respectively. This way the direction of the normal vector can be stored in one RGB image while all values are between 0 and 1.

5.3. Evaluating illumination

Knowing the normal vector the evaluation of the Phong or Lambert model is very easy. Considering equations 1, 2 and

Figure 4: *The special lighting environment.*

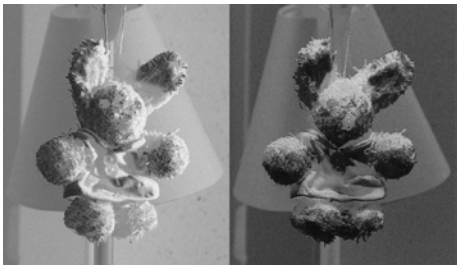

Figure 5: *The x and y components of the normal vector calculated with relation 10.*

Figure 7: *The calculated light is added to the original image.*

3 the relation between the pixel's color and the variables $(\vec{N}, \vec{L}, \vec{\omega})$ can be expressed in a simple, easy to implement equation. The expression of the Lambert model:

$$I = k \cdot \cos\theta' = k \cdot (\vec{N} \cdot \vec{L}) \tag{11}$$

where k is a user given constant and the cosine is evaluated using the dot product of the normal and the light unit vectors. Care should be taken to have the unit vectors normalized. The expression of the Phong model is a little bit more complex:

$$I = k \cdot \frac{(\vec{R} \cdot \vec{\omega})^r}{\cos\theta'} = q \cdot (\vec{R} \cdot \vec{\omega})^r \tag{12}$$

where q is a user given value, $\vec{R}$ is the reflection of the light vector on the normal vector. To evaluate this reflected vector the following relation should be used:

$$\vec{R} = \vec{L} - 2 \cdot \vec{N} \cdot (\vec{N} \cdot \vec{L}) \tag{13}$$

Substituting relations 4 and 13 into 12 we get the expression of the Phong model, which can be implemented easily:

$$I = q \cdot ((\vec{L} - 2 \cdot \vec{N} \cdot (\vec{N} \cdot \vec{L})) \cdot (x, y, 0) - (0.5, 0.5, 1))^r \tag{14}$$

6. Conclusions

A simple image based method was proposed, which makes possible the lighting of objects on still photographs in post. This is done using only image processing tools. The process includes the photographing of the object in special lighting conditions in order to extract the normal vector distribution from the images. Any image processing programs or compositing softwares are suitable to implement the expressions derived in the paper.

There are two major drawbacks of the proposed method. The application to image sequences is not solved yet. A possible procedure may be to shoot the object with twice frame rate, and change the lighting conditions for every second frame. This could be done using synchronized and colored stroboscopes. The second disadvantage is that the method fails at areas where great self-shadowing occurs. However, the testing has proved that it is not a much recognizable artifact.

Many applications are possible in production environment, since the lighting is a key element in emphasizing or weakening the “message” of any picture. The proposed method may help for example the work of still photographers and art directors making advertisements or commercials offering a tool to touch up the lighting of the pictures without reshooting.

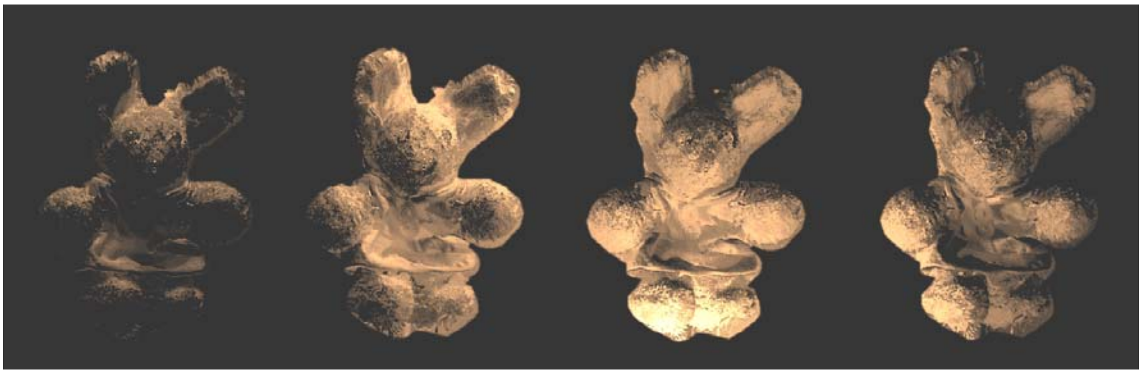

Figure 6: *The artificial lighting of the object produced by the proposed method.*

Figure 8: *The three components of the normal vector extracted from photographs.*

References

1. Kajiya, J.T:, The rendering equation, *in Proceedings of SIGGRAPH 86, p. 143-150*
2. Yu, Y.: Image Based Surface Details *Course 16 in Course Notes of SIGGRAPH 2000.*
3. Debevec, P., Hawkins, T., Tchou, C., Duiker, H.P., Sarokin, W., Sagar, M.: Acquiring the Reflectance Field of a Human Face *in Proceedings of SIGGRAPH 2000, p. 145-156*
4. Yu, Y., Debevec, P., Malik, J., Hawkins, T.: Inverse Global Illumination: Recovering Reflectance Models of Real Scenes From Photographs *in Proceedings of SIGGRAPH 99, p. 215-224*
5. Szirmay-Kalos, L.: Számítógépes grafika *ComputerBooks, 1999.*

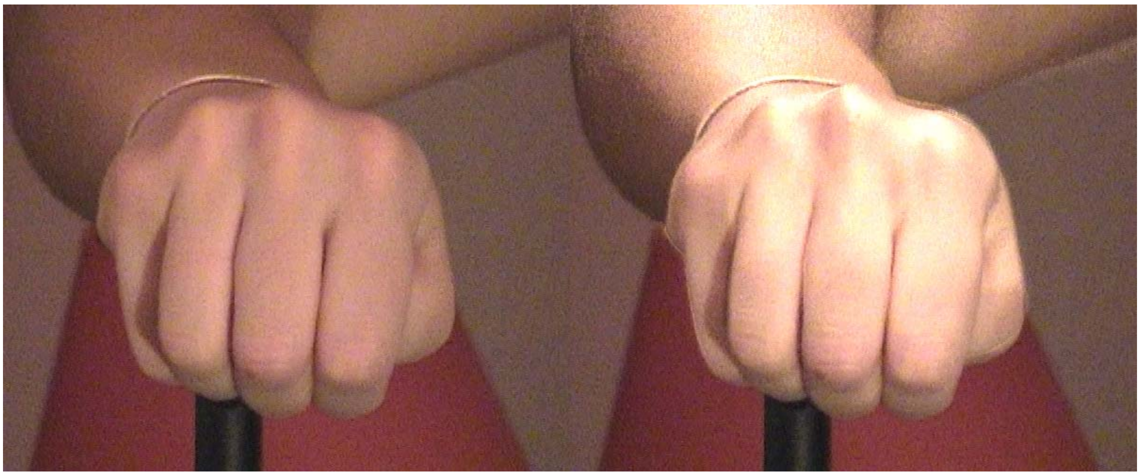

Figure 9: *The fist with and without the light generated by the proposed method. The specular component of the Phong model is recognizable.*