Eurographics Italian Chapter Conference (2006)
G. Gallo and S. Battiato and F. Stanco (Editors)

GPU based direct illuminance values computation for interactive lighting CAD

D. Selmo[1], F. Musante[2], M. Rossi[2]

[1] Dip. DICO - Università degli Studi di Milano
[2] Dip. In.D.A.Co - Politecnico di Milano

Abstract

In this work we focus our attention on real-time direct illuminance values computation using dedicated 3D graphics hardware compatible with OpenGL 2 standard. Our method works with real light intensity distribution measured for lighting fixtures by enconding it in a texture for GPU data comunication. Illuminance values are reprensented in real-time by using a perceptual based colour scale. Illuminance computation is limited to direct light. This is not a severe limitation for exterior and road lighting design, and nevertheless it is a useful project parameter for interior lighting design too.

Categories and Subject Descriptors (according to ACM CCS):
I.3.7 [Computer Graphics]: Colour, shading, shadowing, and texture
J.6 [Computer Applications]: Computer-aided design (CAD)

1. Introduction: Towards an interactive Lighting CAD

An historical gap exists between image synthesis research field and illuminating engineering. Some very accurate light sources descriptive models have been introduced for image synthesis. These models may also allow light sources near field description for industrial lighting calculation. Unfortunately today these models are ignored by lighting industries both for lack of information and absence of measurement instruments. For illuminance calculation, the LID (Light Intensity Distribution also known as photometric solid), is used according to measured values. The lighting CAD software available today don't allow an interactive real-time computation of illuminance values with moving light sources. Illuminance computation is usually done in batch mode, and this requires minutes or hours depending on number of light sources and scene complexity. This severe limitation doesn't allow the design of dynamic lighting and limits the CAD facilities for lighting designers. In lighting project of an environment, the designer's main requirement is to get a real-time illuminance values computation representing any modification to light sources positions and orientations.

2. Previous works

The first attempt to consider lighting fixtures LID for batch image synthesis was introduced by Verbeck [VG84], who used a LID for diffuse component calculation in Phong illumination model. A complete analysis of lighting fixtures measurement with image-based methods has been introduced by Ashdown [Ash95]. These methods are based on near-field photometry assumption (unfortunately this assumption nowadays is not considered by lighting fixture manufactures). Deville and Heidrich [DP95] introduced accurate methods for lighting fixtures definition in lighting engineering, methods based on radiosity global lighting. However this method is too complex to be applied in real-time computation. Other works by Siegel [SS96] and Radiant Imaging researches [RW97] [JM00] focus their attention on lamps light field measurement and description with regard to reflectors and projectors lens design. Their methods are based on CCD image-based measurement. Afterwards similar approaches, based on *light field* (or Lumigraph), have been introduced [LH96] [GGSC96]. These methods describe light field surrounding objects and light sources. They describe the luminance of all the rays intersecting couples of virtual planes, an internal one, next to the object, and another one more external. The measurement of the luminance for real or virtual objects is made using couples of images, really or virtually captured, from various angles. However these methods are not used for an accurate description of the light sources nor for lighting design, but for a quick representation of objects in the real-time rendering.

Languenou [LT92] and Albin [AP03] studied the rendering aliasing problems arising when interpolated light intensity values are used for LID. Also a recent research by Goesle [GGHS03] deals with a very accurate light sources measurement, using near field paradigm, for image synthesis. His methods works with light signal spatial low-pass filtering, before acquisition, in order to eliminate aliasing problems in rendering phase. This method has been integrated in photon-map global illumination model and it is not applicable to real-time illuminance computation.

Zotti, Neuman, Purgathofer [ZNP05] proposed a simple method, based luminaire data file format (IESNA IES-LM-63-95), able to create an approximate model of lighting intensity distribution, using at most two OpenGL predefined light source. A light source is modelled as three lights, two spot lights pointing into the lower and upper hemispheres, and a point light of assigned brightness. An optimization algorithm is used to chose the best set of parameters for the three elements used to model the real light source. Hast and Berrera [HBB04] present a model for intensity function evaluation for soft and hard edged spotlights; the main application of this work is lighting scene for computer animated movie.

The main purpose of our method is to light a real scene (or a simplified one) with real light sources described by a given intensity table; our model starts from standard intensity data representation (IESNA IES-LM-63-95 or eulumdat), so it isn't necessary to pre-process the information taken from manufacture's catalogue. As in illuminating engineering illuminance levels representation we're interested in a technical analysis of how much luminous flux arrives on the surfaces in order to achieve a quantitative verification of the lighting project, in compliance with the reference norms [EN102] [CEN03]. The luminance representation considers the light that reaches the eyes of the observer as a function of illuminance levels and materials reflection properties. Unfortunately the luminance is not enough to describe the real perception of light, because we have to face the problem of representing high dynamics luminance luminance, typical of the real word, by a low dynamic tool, such as display or printer: such problem is known as tone-mapping [MRR04] [DPRW04]. The idea that perceivable differences between a colourful area and its contour are mainly due to colour difference rather than to luminance difference was illustrated by Ware [WB88]. Other properties related to colour target detection are linear separation [BJ96] and colour categorization [KUU95] . These effects are useful for rapid and accurate colour targets identification that allow other important visualization techniques like detection of data boundaries, real-time tracking of data regions, and enumeration tasks like counting and estimation [HBE96].

3. Direct illuminance computation using LID

The measured light intensity flowing from lighting fixtures depends on direction D. In order to describe the intensity variation as a function of D, light distribution can be expressed using a LID (see Figure 1), a tabular function $I_v(\theta,\phi)$ where θ,ϕ are two angles in polar coordinates, defined in accordance with the light emission reference frame. Two main standards for LID definition exist: C-γ method for generic and road lighting fixtures, and *V-H* method for light projectors. Lighting industries use the *gonio-photometer* for LID

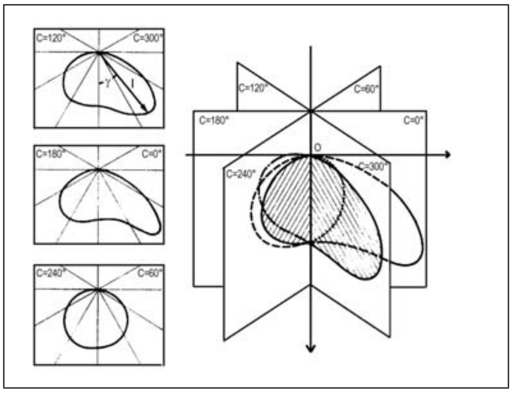

Figure 1: *Lighting fixture LID in C-γ format.*

measurement according to the classical method known as *far field photometry*. This instrument measures the luminous intensity for all directions around the lighting fixture. In the $C-\gamma$ method the lighting fixtures is held to a support, preferably in a position analogous to the real usage, that allows only horizontal rotations $C \in [0,2\pi]$. A mobile arm, that holds a photometer, is moved to describe vertical rotations $\gamma \in [0,\pi]$, thus obtaining the LID $I_v(C,\gamma)$. For measurements, the photometer is placed to a distance of approximately ten meters or more. The reference norms [CIE77] [CIE79] [CIE89] [CIE96] [CIE00]and the industrial practice establish that the C-γ type measurement for generic lighting fixtures should be done with steps of 15° for C angles and 5° for γ angles. For road lighting fixtures the investigation is done with 51 steps for C angles and 36 steps for γ angles, steps width varies depending on luminous flux distribution. The measurement is carried out in a darkned room with the luminaire on the goniophotometer; it is sometimes useful to place the detector at the end of a tunnel which can pass through adjoining rooms. Particular attention and care have to be taken in order to limit stray light during the measurement (a several number of baffles can be used to reach this objective). The equipment can be computer controlled so the reading are taken in definite values for angles C and γ and the goniophotometer are turned by electric motors controlled by

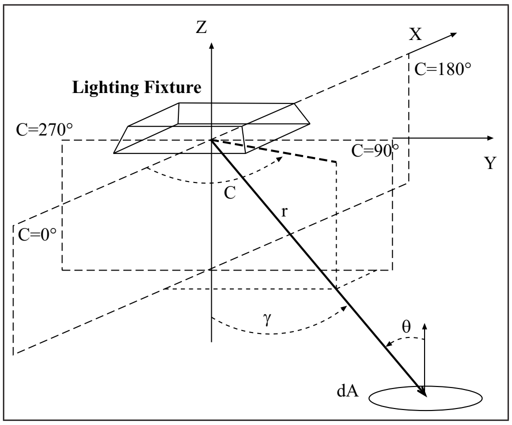

Figure 2: *LID measurement geometry for illuminance computation due to a surface source.*

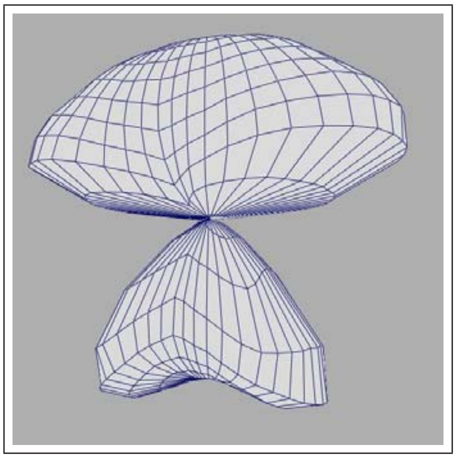

Figure 3: *LID 3D representation*

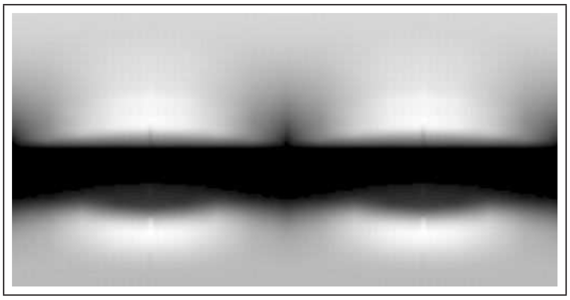

Figure 4: *LID texture encoding example.*

a computer. With reference to Figure 2, in illuminating engineering if the source is punctual (light is originating from a point but with not isotropic angular distribution) the direct illuminance E_v on a surface can be determined as a function of light intensity I_v falling on the surface element with an angle θ to surface normal and with a distance r from light source, according to the simple equation:

$$E_v = \frac{I_{v_{1000lm}}(C,\gamma)\Phi}{r^2}\cos(\theta) \quad (1)$$

where :

- E_v: illuminace on surface infinitesimal element dA, measured in lux
- $I_{v_{1000lm}}(C,\gamma)$: light intensity in cd/1000 lm
- Φ : luminous flux in Klm (kilo-lumen)
- r : distance, expressed in meter, between light source and infinitesimal element dA
- θ: angles, in degree, between normal direction on the lighted surface and direction from light source to infinitesimal element dA

In this equation $I_{v_{1000lm}}(C,\gamma)$ represents the measured LID in candle [cd] / 1000 lumen. The dimensional units of light intensity are cd and the dimensional units of illuminance are lux [lx].

4. LID encoding

Our method is based on a pixel shader written in GLSL language [KBR04], able to run entirely in GPU. In order to make available intensity values, encoded in LID, during pixel shader execution, we need to store this information in a suitable way. The most common paradigm for GPU-based communication consist in storing heterogeneous data in a texture file. For LID representation case, texture encoding works fine because of the intensity values parametric definition. Indeed intensity values for a given LID can be expressed as a function $I = I(C,\gamma)$, where C and γ are angles defined in the LID local reference system as introduced in §3. So we can easily create a map M from intensity values $I(c,\gamma)$ to texels $T(s,t)$ of a gray-level texture as shown in Figure 4.

$$M : I(C,\gamma) \mapsto T(s,t) \quad (2)$$
$$\text{where } C \in [0,2\pi], \gamma \in [0,\pi] \text{ and } s,t \in [0,1]$$

Usually LID are sampled using some dozen of points for angle. Texture resolution can then be chosen according to LID sampling frequency. Otherwise an higher resolution texture can be generated using a standard interpolation technique such as bilinear, bicubic or elliptical interpolation. It must be noticed that interpolation computational expense can be neglected since texture generation can be done in a preprocess step. In our application, we generate textures using bilinear interpolation (as cited in CEN norm [CEN03]) since, for real-time purpose this technique gives us an adequate accuracy .

During shader execution, we need to get an intensity value, from the LID, in order to compute illuminance for each point of the scene geometry. Given a point P, we can calculate the relative normalized direction D (from P to LID center) accordingly to LID reference frame. Texture lookup can be easily implemented using the inverse mapping M^{-1} applied to angles derived from direction D. In particular u, v coordinates for texture lookup can be calculated as:

$$\begin{aligned} u &= \arctan(D_z/D_x)/2\pi \\ v &= \arccos(D_y)/\pi \end{aligned} \quad (3)$$

5. Implementation details

Using programmable GPU we can achieve real-time illuminance values calculation for interactive lighting design applications. A GLSL pixel shader has been developed implementing illuminating engineering equations shown above. For each image pixel, graphics hardware calculates normal and position values, for every scene surface. Using this information along with lighting fixture data, encoded as a texture, illuminance values are evaluated using equation (1). Calculated illuminance values must then be mapped to a colour representation for interactive 3D visualization. In order to obtain a perceptual control over colour scale (also known as colour-map, or false-colour representation) it would be useful to adopt a perceptual colour model like CIE LUV, CIE Lab, or Munsell [WS82].

Index	Colour	r	g	b
0	black	0	0	0
1	red	0.50	0	0
2	magenta	0.50	0	0.50
3	blue	0.27	0.27	1
4	cyan	0	0.63	0.63
5	green	0	0.92	0
6	yellow	1	1	0
7	white	1	1	1

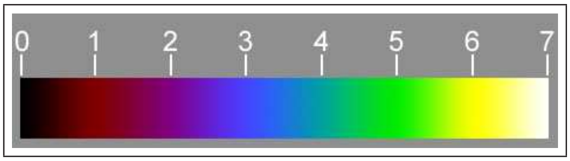

Figure 5: *8-elements colour scale for illuminance value mapping.*

The main requests for interactive lighting design are to allow rapid and accurate perceptive identification of individual data elements and define how many colours we need to display at once. The question that arise is how to choose a colour mapping scale that provide good differentiation between increasing values regions during visualization task. A perceptual based colour mapping, based on gradual increasing colour luminance, has been used [RA05]. Colour map is defined as linear interpolating function between 8 defined colours in RGB-cube. Figure 5 shows colour map elements and resulting image gradient. Colour map has been implemented encoding scale-colours in a simple monodimensional image texture. So, once illuminance values per pixel E_i have been computed during pixel shader execution, we do a global rescale using a user-defined threshold E_{max} obtaining a linear normalized value E_n. E_n is then used for map-texture lookup in order to obtain final colour for each pixel. OpenGL texture lookup settings assure that linear interpolation is used for intermediate values between consecutive colours in colour-map.

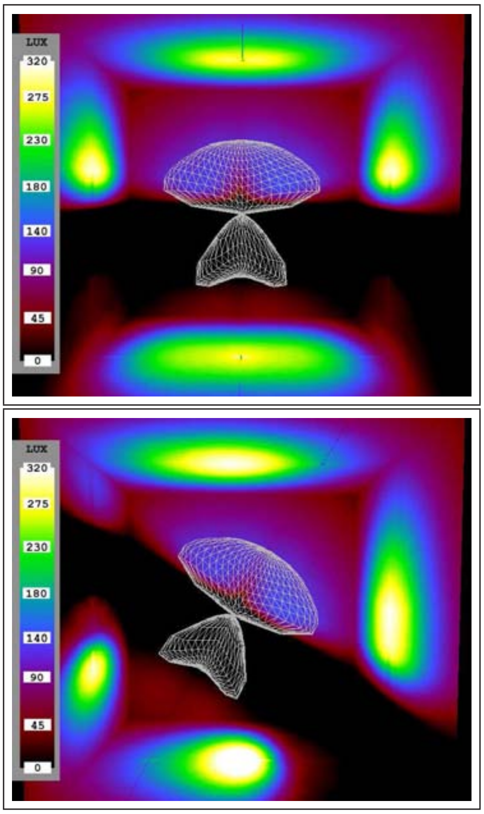

Figure 6: *Result obtained with interactive lighing fixture positioning.*

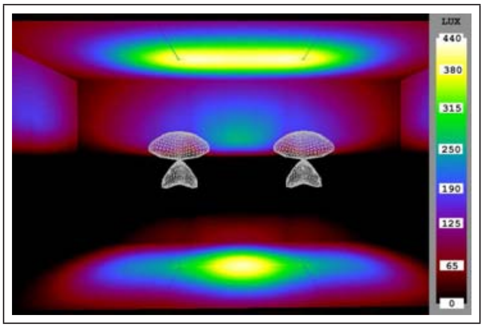

Figure 7: *Result using two lighting fixtures.*

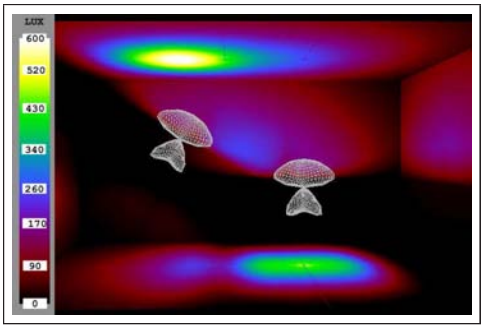

Figure 8: *Result using two lighting fixtures in arbitrary position and orientation.*

6. Results

Our method has been implemented on a Windows workstation using an Intel Centrino 1.7 Ghz processor, 1 GB memory and ATI 9600 graphics card. For testing purposes, a simple square room (4 meter height, 4 meter width, 4 meter length) has been used and luminairies has been placed in different position in order to obtain heterogeneous illuminance levels on room's surface. Two different kind of enviromnent has been tested: the fist one uses only one luminaire, placed in different positions, the second one makes use of two luminaires which can be moved by the user to obtain the wished illuminace level on different surface. In each image we can see the LID representation (this is an important information for the lighting designer, in order to chose the proper luminaire orientation) and the colour scale used to represent the illuminace levels on the scene. The obtained results are collected in Figures (6, 7, 8)

7. Conclusions and future works

A GPU based, real-time algorithm for direct illuminance surface values calculation in simple real-scenes has been presented. Some results and proposal for lighting engineering application has been presented and discussed. Our work is focused in lighting engineering field. Our aim is to create a tool for interactive lighting design using real-measured lighting fixture data. Our method could be used in conjunction with traditional light design tools: users can plan lighting system main characteristics using our interactive calculation. Then results are improved and better defined using batch-mode based calculation methods. In this work we have considered only simple geometry representing rooms with rectangular plan, but our method can be easily extend to arbitrary geometry. Our researches are aimed to improve the presented method including indirect illumination component calculation for rectangular empty rooms, with diffuse lambertian surfaces. Further developments concern taking in account full global illumination computation (suitable methods should be simplified radiosity-based for interactive global illuminance values representation). Direct illumination calculation over arbitrary geometry may also be taken into account.

References

[AP03] ALBIN S., PEROCHE B.: Directionally dependent light sources. *International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision'2003 (WSCG'2003)* (February 2003).

[Ash95] ASHDOWN I.: Near-filed photometry - measuring and modeling complex 3-d light sources. *ACM SIGGRAPH '95 Course Notes- Realistic Input for Realistic Images* (1995).

[BJ96] BAUER B., JOLICOEUR P.: Visual search for colour targets that are or are not linearly-separable from distracters. *Vision Research* (1996).

[CEN03] CEN13201: Road lighting (part 2: Performance requirements, part 3: Calculation of performance, part 4: Methods of measuring lighting performance). *CEN Pubblications* (2003).

[CIE77] CIE34: Road lighting lantern and installation data: photometrics, classification and performance. *CIE Pubblications* (1977).

[CIE79] CIE43: Photometry of floodlights. *CIE Pubblications* (1979).

[CIE89] CIE84: Measurement of luminous flux. *CIE Pubblications* (1989).

[CIE96] CIE121: The photometry and goniophotometry of luminaries. *CIE Pubblications* (1996).

[CIE00] CIE140: Road lighting calculations. *CIE Pubblications* (2000).

[DP95] DEVILLE P., PAUL J.: Modeling the spatial energy distribution of complex light sources for lighting engineering. *Rendering Techniques 95 (Proc. of Eurographics Rendering Workshop)* (1995), 147–159.

[DPRW04] DEBEVEC P., PATTANAIK S., REINHARD E., WARD G.: High-dynamic-range imaging. *ACM SIGGRAPH 2004 Course n. 13* (August 2004).

[EN102] EN12464-1: Lighting of work places - part 1: Indoor work places. *Light and Lighting* (2002).

[GGHS03] GOESLE M., GRANIER X., HEIDRICH W., SEIDEL H.: Accurate light source acquisition and rendering. *ACM SIGGRAPH 2003* (July 2003), 621–630.

[GGSC96] GORTLER S. J., GRZESZCZUK R., SZELINSKI R., COHEN M.: The lumigraph. *Computer Graphics (SIGGRAPH '96 Proceedings)* (August 1996), 43–54.

[HBB04] HAST A., BARRERA T., BENGTSSON E.: Fast intensity distribution functions for soft and hard edged spotlights. *International Conference in Central Europe on Computer Graphics, Visulazation and Computer Vision WSCG' 04 Short Paper* (2004), 95–99.

[HBE96] HEALEY C., BOOTH K., ENNS J.: High-speed visual estimation using preattentive processing. *ACM Transactions on Computer-Human Interaction 3*, 2 (1996), 107–135.

[JM00] JENKINS D., MONCH H.: Source imaging goniometer method of light source characterization for accurate projection system design. *proc. SID Conference* (2000), 862–865.

[KBR04] KESSENICH J., BALDWIN D., R.ROST: The opengl shading language. *Language Version 1.10* (April 2004).

[KUU95] KAWAI M., UCHIKAWA K., UJIKE H.: Influence of color category on visual search. *Annual Meeting of the Association for Research in Vision and Ophthalmology* (1995), 2991.

[LH96] LEVOY M., HANRAHAN P.: Light field rendering. *Computer Graphics (SIGGRAPH '96 Proceedings)* (August 1996), 31–45.

[LT92] LANGUENOU E., TELLIER P.: Including physical light sources and daylight in global illumination. *In Proceedings of the 3rd EGWR* (1992), 217–225.

[MRR04] MARINI D., RIZZI A., ROSSI M.: Post-filtering for color appearance in synthetic images visualization. *Journal of Electronic Imaging, Ed. SPIE, Bellingham WA, USA 13*, 1 (January 2004), 111–11.

[RA05] ROSSI M., ALGERI T.: Real-time 3D color scale representation of computed illuminance levels. *SIOF-Color Conference* (October 2005), 1–8.

[RW97] RYKOWSKY R., WOOLEY C.: Source modeling for illumination design. *Lens Design, Illumination, and Optomechanical Modeling - SPIE 3130* (1997), 204–208.

[SS96] SIEGEL M., STOCK R.: A general near-zone light source model and its application to computer automated reflector design. *SPIE Optical Engineering 35*, 9 (1996), 2661–2679.

[VG84] VERBECK C., GREENBERG D.: A comprehensive light-source description for computer graphics. *IEEE Computer Graphics and Applications 4*, 7 (July 1984), 66–75.

[WB88] WARE C., BEATTY J.: Using colour dimensions to display data dimensions. *Human Factors 30*, 2 (1988), 127–142.

[WS82] WYSZECKI G., STILES W.: Color science: Concepts and methods, quantitative data and formulae. *John Wiley & Sons, 2nd Edition.* (1982).

[ZNP05] ZOTTI G., NEUMANN A., PURGATHOFER W.: Approximating real-world luminaires with OpenGL lights. *WSCG 2005 Short Paper Proceedings* (February 2005), 49–52.