Light AI

Magy Seif El-Nasr

Information Science and Technology
Pennsylvania State University
magy@ist.psu.edu

Abstract

Lighting design is an important topic of 3D scene design. There are many functions that lighting assumes in game environments, including directing attention, visibility, evoking emotions, setting atmosphere, and providing depth. Due to the unpredictability of interactive environments, current lighting design models do not adequately adapt the lighting to serve these aesthetic and communicative functions. In this paper, I will discuss the contributions of ELE (Expressive Lighting Engine) as an intelligent lighting control system that automatically adjusts the lighting to achieve aesthetic and communicative functions, including evoking emotions, directing visual focus, and providing visibility and depth in game environments. In particular, I present results from interfacing ELE to the Unreal 2.0 engine (used in Unreal Tournament 2003 and 2004). These preliminary results show (1) enhanced control of lighting in an interactive environment to provide visual focus, set atmosphere, evoke emotions, and establish visibility, and (2) acceleration in the development process due to the introduction of an automated system for lighting that can be overridden by designers at a high-level, and thus eliminating the time consuming process of setting individual light parameters for each level and/or scene.

Introduction

Game designers have identified the importance of visual composition (Carson, 2000; Chen & Brown, 2001; Maattaa, 2002). Visual composition is a term used to define the designer's choices of camera placement and movement, lighting colors and angles, and characters' placements and blocking. At GDC (Game developers Conference), Will Wright identified several functions that visual composition assumes in game environments, including directing player's focus to important elements in a game by balancing saturation, brightness, and hue of the different objects in the environment and the geometry of the level (Wright, 2004). Film and theatre design theories identified these functions and others, including the use of lighting to establish depth, visibility, style, and the use of camera movement to establish pace, urgency, and focus (Alton, 1995; Birn, 2000; Block, 2001; Brown, 1996; Campbell, 1999; Viera, 1993).

Current lighting design models in games rely on static lighting or simple scripted dynamic lights (e.g. lights that follow characters to ensure their visibility). Due to the unpredictable nature of interactive environments, using these techniques does not fully realize the communicative and aesthetic properties of light. I argue that an automatic lighting system (in this paper I use the term light AI to denote such a system) is important and essential to enhance the player's experience.

Some game developers have recognized the importance of modulating the visual design to enhance the player's experience. However, most current work focus on intelligent camera control – Camera AI (Carlisle, 2003; Giors, 2004; Kharkar, 2003; Stone, 2004). For interactive environments, light AI can assume many communicative and aesthetic functions that a camera cannot without disrupting suspension of disbelief. Since the camera is in the control of the participant, controlling the camera is very restrictive and can break suspension of disbelief. Lighting, on the other hand, can be changed more freely; its effects are subtle yet very effective. Many film and theatre designers have used gradual unnoticeable shifts in intensities and colors to shift attention of the audience or to evoke emotions and tension (Alton, 1995; Birn, 2000; Block, 2001; Brown, 1996; Campbell, 1999; Viera, 1993). Thus, lighting provides a subtle mechanism for adjusting player's attention, evoking emotions, providing depth of field, and establishing visibility.

In addition to adapting the design to establish communicative and aesthetic functions documented by traditional media, using an adaptive intelligent lighting system can expedite the development process. Lighting design is a time consuming process, which involves many weeks of tweaking the lighting to the given atmosphere, level design, and predicted blocking. A system that automatically places lights and determines colors and angles depending on a scene graph and dynamic parameters, including locations of characters, characters' relationships, and tension value, will expedite the development process. This is important for rapid prototyping and testing of design ideas.

In this paper, I will present results from interfacing ELE (Expressive Lighting Engine), an intelligent lighting system, to a first person shooter game engine (Unreal Tournament 2003). According to preliminary results collected, ELE has two main contributions: (1) it automatically adjusts the lighting to the given situation, which enhances control of visibility, visual focus, depth, and evoked emotions, and (2) it expedites the development process by providing designers a small set of high-level lighting parameters that allows them to quickly establish the lighting style and atmosphere. It should be noted at this point that the paper will focus on the contribution of ELE in game environments, e.g. first person shooters, with a summary of ELE. A more detailed description of ELE and its utility for interactive narrative and interactive drama is discussed in (Seif El-Nasr, 2003; Seif El-Nasr & Horswill, to appear).

Related Work

Lighting design is a very important element of an interactive 3D production. The game industry has developed many techniques and lighting models, I will discuss these models in this section.

Ambient lighting is a method where the objects are given constant luminance values (Moller & Haines, 1999). It is a fast and simple lighting model in which all objects are equally visible. This type of lighting has been used in games, including Sims and Sim City. Although the technique supplies the desired look and feel for games like Sims and Sim City, it cannot be generalized for use in first-person shooter or action games, where a realistic and atmospheric lighting style is often more appropriate.

Another lighting technique that is employed by many adventure and role playing games is realistic lighting, whereby lights are placed to achieve realistic goals, including realistic effects such as shadows cast on a character's face to show the effects of the direction of a light source (such as the sun or a torch), the reflectance of a character's shadows on scene geometry, and more accurate lighting calculations.

Realism is only one of the many important goals of a lighting design. Other important goals include ensuring visual focus, providing visibility, and paralleling the tension of the interaction (Block, 2001; Cheshire & Knopf, 1979; Kidd, 2001; Lowell, 1992). To achieve these goals, designers often employ several tricks such as halos around objects to direct users to important objects and spotlights that follow characters around to ensure their visibility. These techniques are limited by the underlying narrative architecture. For example, if we allow the importance of objects to change unpredictably during interaction, then a halo that suddenly appears around an object is unrealistic and/or distracting. Theatre and film apply very subtle techniques to achieve these goals. However, such techniques rely on a model of lighting design that coordinates the properties of each light in a scene and integrates their function.

One advantage of using a model based on theatric and cinematic theory is the ability to automatically adapt to the continuous variation in tension and action. By examining games such as Devil May Cry it is clear that tension is broken into discrete missions that are materialized when an appropriate level is loaded. In some cases, the difference between the levels is only in texture or lighting colors; for example, the last level of Devil May Cry is colored in a distinct saturated red color, signifying the climax. There are several problems with this method: first, it is very tedious to redesign and relight each level; second, the design involves breaking the continuous flow in tension and manually adjusting the textures or lighting to accommodate the increase and decrease in tension.

Preliminary Results from Eye Tracker Experiments

We have collected some preliminary results from an eye tracking experiment where we asked 28 students to play unreal tournament while we track their eye movements and actions. Results show that inadequate visual composition of the scene caused frustration because players were not able to decipher the environment fast enough to respond to enemy attacks.

We invited 28 subjects to play unreal tournament 2003. We asked them to wear a baseball cap that has an eye tracker mounted on it. We video taped their interaction. Only 7 of the subjects have previously played Unreal Tournament. However, all subjects have previously played a form of first person shooter.

By examining their eye movements and actions, we concluded that subjects (especially the ones who didn't previously play Unreal) were not able to decipher the environment quickly enough to respond to enemy attacks, and thus were shot and killed instantly which made their experience unpleasant.

ELE – Expressive Lighting Engine

The problem discussed in section 3 is one of the many problems with lighting in game environments today, which motivates the move to an automatic lighting control system. Since ELE has been published elsewhere (Seif El-Nasr & Horswill, to appear), I will only summarize ELE in this section to give the reader a quick overview of ELE. For a more comprehensive treatment of ELE, readers are referred to (Seif El-Nasr, 2003; Seif El-Nasr & Horswill, to appear).

ELE is an automatic intelligent lighting control system developed based on cinematic and theatrical lighting design theories; it is designed to automatically select the number of lights, their positions, colors, and angles. To accomplish this task, ELE uses lighting-design rules represented mathematically in an optimization function. The use of optimization is important to balance conflicting lighting-design goals. While adapting the lighting to the interaction, ELE also maintains visual continuity and style.

I assume that there exists a system that passes several parameters to ELE, including a set of parameters describing style, local light sources, scene graph, characters' dimensions, focus (the area/characters to which attention should be directed) and the dramatic intensity of the situation. Using these parameters, ELE computes the number of lights to be used. For each of these lights, it computes the type of instrument (e.g., spot light or point light), color in RGB color space, attenuation, position as a 3D point, orientation including the facing and up vectors, range, masking parameters, and, depending on the light instrument used, the Penumbra and Umbra angles. These parameters are given to a rendering engine to render the frame.

The lighting system divides the visible area into *n* different areas. It categorizes these areas as: focus, describes the focus of the scene, non-focus, areas surrounding the focus area, and background areas. ELE determines where to direct viewers' attention (or the focus) given the number of characters in the frame and the dramatic importance of their actions (for Unreal Tournament I used a danger function (e.g. number of kills, accuracy of shots) to signify enemies' importance). ELE then assigns lights to each area depending on its category. The angle and color systems assign angles and colors depending on the number of lights assigned to each area and the area's category.

The angle system selects an angle for each key[1] light according to a number of requirements, which include ensuring visual continuity, maintaining the illusion of a local light source, maintaining mood, and ensuring that all the characters are sufficiently lit.

Cinematic rules used to satisfy these requirements often contradict with one another. Angles used to establish mood, for example, don't usually produce good visibility, e.g. rim or silhouette angles. Thus, I softened these rules into cost functions, where the designer controls weights associated with the contradicting requirements, such as mood, visibility, and modeling, as follows:

$$\text{cost}(k,k^-,m)=\lambda_v V(k)+\lambda_-\left|k-k^-\right|+\lambda_m\left|k-m\right|+\lambda_l\min_i\left|k-l_i\right|,$$

where k is the key light angle, k^- is the key light angle from the previous frame, and m is the mood angle suggested by the designer, λ_v is the cost of deviation from an orientation of light, which establishes best visibility, λ_- is the cost of changing the key light angle over time (to enforce visual continuity), λ_l is the cost of deviation from an established light source angle/direction. In addition, for each light source, and l_i is the angle of light from the light source i on the subject or area in question, and λ_m is the cost of deviation from an angle that shows a specific mood. Based on Millerson's work (Millerson, 1991), I derived the following formula to evaluate visibility and modeling:

$$V(k)=\sin(\left|k-c\right|)\cos(k-s),$$

where k and c are the azimuth angles of the key light and camera relative to the subject and s is the azimuth angle toward which the subject is facing. Millerson recommended an elevation angle between $\pi/6$ and $\pi/3$ (Millerson, 1991).

I then used a linear optimization based on hill climbing to select an angle for each key light that minimizes the cost function above.

Fill and backlight azimuth angles are calculated depending on the value of the key light angle and the angle between the camera and the subject. According to the guidelines described by Millerson (Millerson, 1991), fill light azimuth and elevation angles are calculated to be the mirror image of the key light angle. I derived a formula to compute backlight azimuth angle as follows:

$$b=\left(k-c+\pi\right)\bmod 2\pi,$$

where k and c, again, are the are the respective angles of the key light and camera. Backlight elevation angle is set to $\pi/4$ as recommended by Millerson (Millerson, 1991).

The interaction between colors assigned for each area in a scene composes the contrast and feeling of the entire image. I calculate contrast and depth as the difference between colors assigned to each area category.

Using ideal values for visibility, depth, saturation, warmth, and lightness for each area category, and their associated costs, the system uses constrained nonlinear optimization to select a color for each individual light in the scene that will minimize the cost function:

$$\text{cost}(t,c_i,c_{i-1})=P(t,c_i,c_{i-1})+\varepsilon\sum_j^{\rho}\log(-g_i(x)),$$

where g: $R^3 \rightarrow [0,1]$ describes the desired color palette. P is defined as follows:

$$P(t,c_i,c_{i-1})=\begin{array}{l}\lambda_s(S(c_i)-s)^2+\lambda_l(L(c_i)-l)^2+\\\lambda_d(D(c_i)-d)^2+\lambda_w(W(c_i)-w)^2+\\\lambda_c(C_\phi(c,c_i)-c_i)^2+\lambda_{ch}E(c_i,c_{i-1}),\end{array}$$

where $S(c_i)$ is saturation, $L(c_i)$ is lightness for color c_i and are all calculated using formulae defined in (Castleman, 1996). $C_\phi(c, c_i)$ denotes contrast of color c_i given c. The values for the costs λ_s, λ_c, λ_w, λ_l, λ_{ch}, λ_c are given by the designer, and λ_s, λ_c, λ_w, λ_l are computed by the system for every area type (focus, non-focus, and background).

It is important for designers/artists to retain control over the lighting process. ELE is designed to adapt to the situation while maintaining and satisfying artistic constraints concerning style and the overall lighting design goals. It allows artists to control or override its behavior at a high-level or a low-level. Additionally, ELE supplies artists with a language to write rules for specific lighting changes or setup.

The lighting design parameters and model used for ELE was a result of an iterative design process where feedback from an onsite cinematographer was used to further enhance and refine the system. Such a process is important to ensure usability of the system.

Implementation and Results

ELE has been implemented in C#. It is an API library. It sits on top of a rendering engine. It has also been

[1] Key light is the main source of light that establishes the direction and the shadows

configured to use XML to interact with rendering engines. This allowed the architecture to be more extensible. I have implemented two interfaces for ELE on two rendering engines: Unreal 2.0 (used by Unreal Tournament 2003) and Wildtangent (publicly available graphics engine). ELE was not modified to accommodate for the different rendering engines, I needed only to implement an interface to interface between ELE and the rendering engine.

One concern was the speed of first person shooters and the possibility that ELE may not be able to catch up given the optimization algorithms used. There were no performance problems with interfacing ELE to Unreal Tournament 2003 (UT2003). The game interfaced with ELE ran at a 30 frames/second rate using a Pentium 4 with an NVIDIA GeForce 4.0. There were some issues concerning the limitation of the unreal rendering engine to render dynamic spot lights, shadows, and correctly calculate the colors of characters given the light effects of weapons and the surroundings. This was a major issue with Unreal, and imposes many limitations. In spite of these shortcomings, I was able to port ELE and get some good results.

Figure 1. Different Levels

As mentioned above, ELE has several contributions. Using ELE designers can quickly set the lighting design in terms of style and aesthetic functions without worrying about low-level details of placements, angles, or color. This expedites the development process. Figures 1 and 2 show ELE used to light three different levels within UT2003. This did not require any designer interference, except for initial setting of constraints specifying the style, which took a designer around one minute to set. ELE configures the lighting according to the scene graph.

I have also identified ELE's contribution in portraying communicative and aesthetic functions, such as visual focus, depth of field, emotions and tension, while maintaining visual continuity and style. Figures 1 and 2 show some variations of the parameters set for the levels. Figure 1 shows three different renderings where ELE was configured differently in regards to contrast and overall atmosphere. The upper left image shows a scene where ELE was configured for good visibility, while the lower right image shows a scene where ELE was configured for mood lighting, and thus the character in the lower right scene was backlit. The two scenes on the right are set for low-key torch lighting, while the image of the scene in the lower left shows a more atmospheric lighting style with shades of green.

Figure 2 shows several transitions where ELE was configured to adjust for visual focus. As it can be seen in the figure, as the player comes nearer to the enemy, the enemy becomes the focus of the image, and thus ELE adjusts the contrast, which in this case brightness contrast, accordingly.

Figure 2. Visual focus setting

Conclusions and Future Work

In this paper, I described ELE, a lighting system that automatically adapts lighting to the continually changing situation using rules developed based on cinematic and theatric lighting design theory. The results discussed show great promise for the success of the approach. As discussed in the results section, the system was very successful in expediting the development process and accommodating the dynamic situation.

I acknowledge that there will be a transition for designers who use ELE, from adjusting the lights manually in a scene to using the lighting design parameters advocated by ELE; however, it is a worthwhile transition for several reasons. ELE allows designers to manipulate the lighting in a scene at the level of lighting design functions and goals, and thus allows designers to think in terms of artistic and aesthetic goals of the design rather than the design's implementation. In addition, ELE supplies artists with many lighting effects and functions that are only possible through automatic real-time modulation of lighting during interaction.

Many future directions remain to be explored. The current model of ELE does not incorporate shadows (self cast shadows, cast shadows), inter-reflection of light, or outdoor lighting. These are important properties for 3D scenes. For future research, I aim to extend ELE to include shadows. In addition, I also aim to focus on the area of lighting appearance and extending the Unreal engine or developing a game engine that handles one level or two level reflections/refractions and Shaders.

Acknowledgements

I would like to thank and acknowledge Dr. Frank Ritter and his students at the Applied Cognitive Science Lab in the School for Information Science at Pennsylvania State University for setting up and administering the eye tracking experiments.

References

Alton, J. 1995. *Painting with Light*. Berkeley: University of California Press.

Birn, J. (Ed.). 2000. *Digital Lighting & Rendering*. Indianapolis: New Riders.

Block, B. 2001. *The Visual Story: Seeing the Structure of Film, TV, and New Media*. New York: Focal Press.

Brown, B. 1996. *Motion Picture and Video Lighting*. Boston: Focal Press.

Campbell, D. 1999. *Technical Theatre for Non-technical People*: Allworth Press.

Carlisle, P. 2003. An AI Approach to Creating an Intelligent Camera System. In S. Rabin (Ed.), *AI Game Programming Wisdom 2*: Charles River Media.

Carson, D. 2000. Environmental Storytelling: Creating Immersive 3D Worlds Using Lessons Learned from the Theme Park Industry. *Gamasutra*.

Castleman, K. R. 1996. *Digital Image Processing*. Englewood Cliffs: Prentice Hall.

Chen, S., & Brown, D. 2001. *The Architecture of Level Design.* Paper presented at the Game Developers Conference, San Jose.

Cheshire, D., & Knopf, A. 1979. *The Book of Movie Photography*. London: Alfred Knopf, Inc.

Giors, J. 2004. *The Full Spectrum Warrior Camera System.* Paper presented at the Game Developers Conference, San Jose.

Kharkar, S. 2003. A Modular Camera Architecture for Intelligent Control. In S. Rabin (Ed.), *AI Game Programming Wisdom 2*: Charles River Media.

Kidd, W. (Writer), & J. J. Krakora (Director). 2001. Vermeer: Master of Light Video. In J. J. K. a. E. Bryant (Producer). Washington: National Gallery of Art.

Lowell, R. 1992. *Matters of Light and Depth*. New York: Lowel-light Manufacturing, Inc.

Maattaa, A. 2002. GDC 2002:Realistic Level Design for Max Payne. *Gamasutra*.

Millerson, G. 1991. *The Technique of Lighting for Television and Film* (3rd ed.). Oxford: Focus Press.

Moller, T., & Haines, E. 1999. *Real-time Rendering*. MA: A K Peters, Ltd.

Seif El-Nasr, M. 2003. *Automatic Expressive Lighting for Interactive Scenes.* PhD. Dissertation, Northwestern University, Evanston, IL.

Seif El-Nasr, M., & Horswill, I. to appear. Automating Lighting Design for Interactive Entertainment. *ACM Computers in Entertainment*.

Stone, J. 2004. Third-Person Camera Navigation. In A. Kirmse (Ed.), *Game Programming Gems 4*: Charles River Media.

Viera, D. 1993. *Lighting for Film and Electronic Cinematography*. Belmont: Wadsworth Publishing Company.

Wright, W. 2004. *Triangulation: A Schizophrenic Approach to Game Design.* Paper presented at the Game Developers Conference, San Jose.