



Two Stage Fuzzy Inference Systems for Autonomous Lighting in 3D Animated Movie Scene

Andreas Andreas^{1,2,3}Mochamad Hariadi^{1,2*}Mauridhi H. Purnomo^{1,2}Kunio Kondo⁴¹*Department of Electrical Engineering, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia*²*Department of Computer Engineering, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia*³*Department of Information System, Universitas Pelita Harapan, Surabaya, Indonesia*⁴*Department of Media Science, Tokyo University of Technology, Tokyo, Japan*

* Corresponding author's Email: mochar@ee.its.ac.id

Abstract: Lighting is one of the very important things in cinematography. Lighting in cinematography is used not only to illuminate the objects, but it has many important roles such as to make the object looks sharper, has tone, depth and separation, and so on. There are some parameters that affect the lighting result, such as light source type that is used, its colour, intensity, position, and direction. Unfortunately, lighting has so many permutations and variations and there is more than one right way to light the scene. In this research, a designer designs some scenes that will be used as inputs. In this study, the authors found that lighting set is strongly influenced by five important things, which are: the situation of the scene, Point-of-View, the intensity of the attached light sources, camera position, and emotion of the characters, that we called as the SPICE system that is the abbreviation from situation, PoV, intensity, camera, and emotion. Some additional light sources will be added into the scene automatically according to the scenario and director's behaviour using the SPICE system that we propose. The SPICE framework has two fuzzy inference systems in calculating the intensity of light that hits an object and some additional light sources that are added to the scene automatically to make it look much better. This research will make a calculation to gain the best scenes as an output. There are 20 scenes used in this research. As the final result, designers assess 80% outputs have been as expected.

Keywords: Machinima, Autonomous lighting, Virtual director, Fuzzy logic.

1. Introduction

There are three components that are very important in cinematography: the scenario, camera, and lighting [1]. A good scenario can produce a good movie. Camera's role is to visualize the scenario. A dynamic camera movement is needed to make the visualizations appear more alive. Lighting makes the visualization to life. There are several jobs that lighting does. It creates an image that has a full range of tones and gradations of tone, colour control and balance, shape and dimension in the individual subjects, separation so subjects can stand out against the background, depth and dimension in the frame, texture, mood and tone, and exposure. That is why lighting is a very important thing in cinematography. Kol [2] said that light is a natural phenomenon that is

increasingly featured in movies and games, as it is visually pleasing and lends realism to a scene. Even in the game, it is said that game-cinematography could improve the game-play experience of the users [3].

There are some parameters that affect the lighting result, such as light source type that is used, the color, intensity, position, and direction [1, 4]. Lighting has so many permutations and variations. That is why there is more than one right way to light the scene. Light is very important in cinematography. A slight change in lighting conditions can affect the appearance of the objects from one image to another [5]. Therefore, it is very difficult for us to make a simple list of proper lighting techniques. What designers can do is to try to identify how they want the lighting does for them [1].

In order to gain a good lighting, it often requires the additional light source other than the light source that has been installed [1,6-7]. The addition of this additional light source is depending on the demand of the scenario also highly dependent to the director's behavior.

Lighting design is a very important part in computer animation production phases where it is going to establish the mood of the scene and enhance storytelling. It can be considered as an inverse problem of global illumination. According to Lin [8] lighting design plays a crucial role in indoor lighting design, computer cinematograph and many other applications.

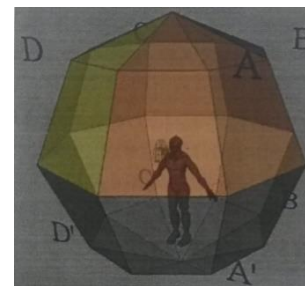
In the global illumination theory [9-10], it is said that there is a very complicated relationship between light parameters and the lighting effect result. It is counter-intuitive even for a very well-trained designer. To meet the desired lighting effect specified by a movie director, a lot of additional light set are often needed. In order to achieve the desired effect, a lighting designer needs to put extra effort to carefully and repeatedly adjust the light parameters of each additional light source. It is a very labor-intensive and time-consuming process. Handling the lighting design is also felt like a counter-intuitive process. On the other hand, repeatable rendering processes due to multiple experiments on the possibility of adding additional light sources will be very high in cost and time [11].

Montes in [5] said that radiance is influenced by the characteristics of materials with which electromagnetic radiation interacts by scattering, reflection, and refraction. Reflection is the process by which electromagnetic flux, incident on a stationary surface or medium, leaves that surface or medium from the incident side without change in frequency. In other hands, the high cost of simulating scattering events is very challenging in rendering participating media [9].

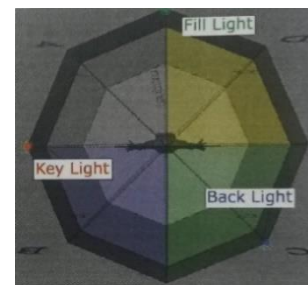
Based on the research of [12] the object needs to be illuminated by two light sources, which are at 45° between camera and the object as front lighting, and at the opposite of front light as back lighting. Meanwhile, according to [4] using lighting scrapbook, the designers only need three points of additional light position, which are 45° in the front of the object as the main light, at the side of the object as the key light, and at 45° behind the object as the back light. The position of these three light sources can differ depending on the situation of the scene. The parameters that affect the placement position of the light source will be discussed in another section of this article.

The three-point lighting setup is also mentioned by [1]. He said that the name is a misnomer as there are actually more than three lights involved, even in the simplest setups. Fig. 1 shows the diagram of a typical three-point lighting setup. Kicker and background light are not always needed in every movie scene.

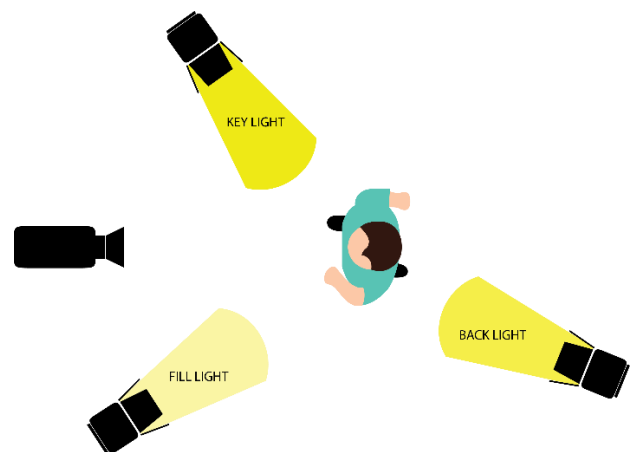
Although there are several articles that discuss lighting in cinematography, there is very little research in the field of lighting, especially autonomous lighting. For this reason, we conducted interviews with three film directors to discuss this problem, then we put their knowledge into the structure of fuzzy logic.



(a)



(b)



(c)

Figure. 1 Three points of light source position: (a) in 3D view, (b) top view, and (c) diagram of a typical three-point lighting setup

This research tries to add some necessary light sources autonomously based on the scenario and the director's behavior. We build two Fuzzy Inference Systems (FIS) to be involved in this research. The first FIS will calculate the amount of light intensity that hits an object so that the real intensity can be calculated correctly. The intensity of an object not only comes from the direct light sources but also comes from the lights that reflected by other objects around it [13]. This reflected light depends on the level of the glitter from the surface of the object, the level of smoothness, and its color. Then the output data from the first FIS will be used as input for the second one. The second FIS will accept five input parameters, which are the scene situations according to the scenario, point of view of the active camera, the intensity of the lights in the scene, the position of the active camera, and the emotions of the main character. Those both FISes are built in a system that we called the SPICE system.

By using the system that we propose, a designer only needs to create a design according to the scenario, then the needed additional light sources will be added automatically by this SPICE system. Thus, a designer does not need to understand the theory of lighting in the animated 3D films to create scenes that are in accordance with the director's wishes.

2. Global scheme

This research starts with the scenario of the short 3D animation movie. First, the scenario will be divided into several scenes. This is part of the director's job. In this article, the needed parameters are retrieved by exploring each scene manually and saved in an ASCII file using a simple web-based application. These parameters are needed in order to synchronize between the application's rules and the director's rules (behaviour). List of parameters needed in this research is explained on the other part of this article.

The designer builds every scene in Unity 2018 based on the storyboard from the scenario. Designer design the scenes with initial light sources as describe in the storyboard or scenario. These initial light sources are the main light source with directional light type, and some point lights attached in any light models that exist in the scenes. The position and direction (rotation) of these point lights are as shown in the map (movie set).

This research embeds some scripts written in C# into each Unity files to do the calculations. There are some calculations to do such as get the light intensity on the surface of each object in the scene, determine the positions (coordinates) and direction (rotation) of

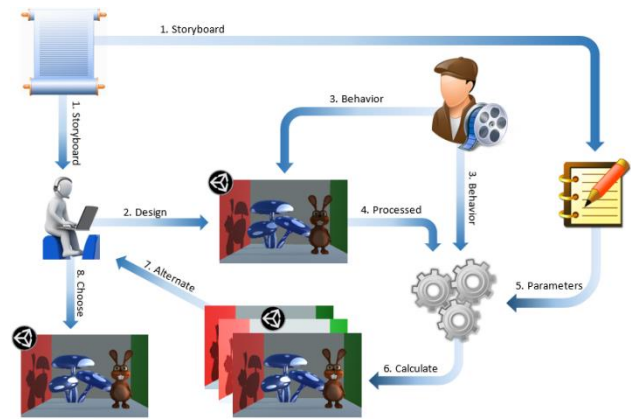


Figure. 2 Diagram of the global scheme of our research

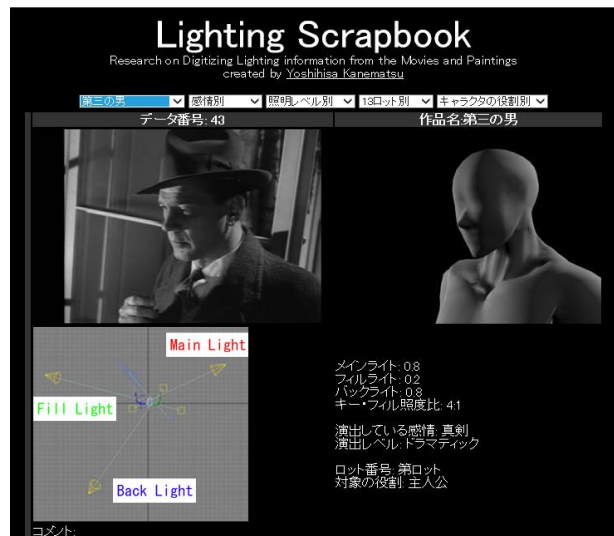


Figure. 3 An example of initial lighting set using lighting scrapbook application

the required additional light sources and each its intensity. This script will process the scene created by designer with director's behaviour and a list of parameters involved.

After some calculations, this script will give some alternate additional light sources as outputs. As a final phase, designers will manually assess the output (modified light set) to determine whether the resulting output is in line with expectations or not. Fig. 2 describes the global scheme of this research that we proposed.

To gain a better initial lighting set, the script uses the lighting scrapbook. Lighting scrapbook is a web-based application that gives a lighting set based on a database [6]. The database itself is based on many experiments that involved the director's behaviour. Fig. 3 shows the example of the initial lighting set as an output of lighting scrapbook.

This application only to ease the script's work in determining the point of initial light sources placement. It still needs to adjust the position and direction (rotation) of those light sources.

The main purpose of this research is to make it easier for designers to make 3D animated films that are in line with the scenario and the director's behaviour.

3. The SPICE system

This article uses two short animation movies created using Unity 2018 while some models are created using Blender. The movie will be divided into several scenes. The changing of the scene marked with active camera switching. There are some scripts that embedded into Unity files of each scenes in the start () and update () functions. These functions are automatically called every time we run the scene and can be activated or deactivated by a toggle key on the keyboard.

3.1 Get all objects

As the first step, this script gets all objects in the scene. The object could be models, cameras, and lights. Information about position and scale of all models are needed to map their existence against the camera. There could be more than one camera in a scene, but this research only needs one camera which is the active one. For light objects, the needed information is its type, position, rotation (direction), and intensity. Fig. 4 illustrates the objects that been get from the scene.

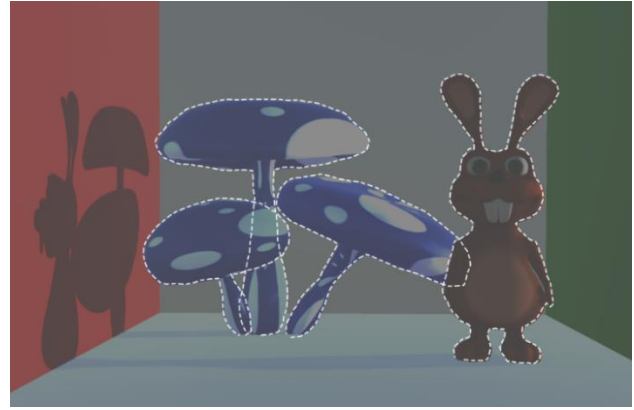


Figure. 4 Tracing the scene to get all objects inside it

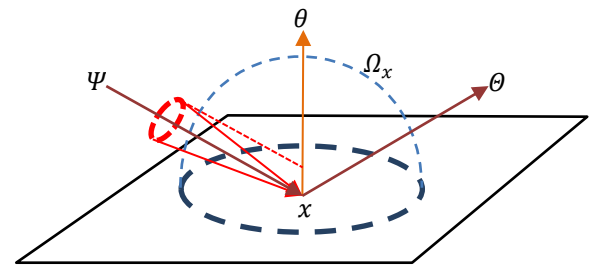


Figure. 5 Light comes from Ψ to x that reflected to Θ

3.2 Calculate the light's intensity

After all the objects contained in the scene has been retrieved, the script will calculate the light intensity at the surface of all objects that facing the active camera using Eq. (1).

$$I_{(x,y,z)} = \frac{I}{4\pi((x-x_0)^2+(y-y_0)^2+(z-z_0)^2)} \quad (1)$$

where I is the intensity of the direct light source, $I_{(x,y,z)}$ is the intensity of the object's surface at (x, y, z) coordinate, while (x_0, y_0, z_0) is the coordinate of the light source.

This intensity will be added with the additional intensity from the bounced (reflected) lights. Fig. 5 shows how a light comes from Ψ to point x will be reflected to Θ . There is a normal line θ that perpendicular to the plane that representing the surface of the object inside the scene.

The reflected light intensity is calculated using the modified BRDF formula as below:

$$dL_{reflected} = f_r(x, \Psi \rightarrow \Theta)dE(x \leftarrow \Psi) \quad (2)$$

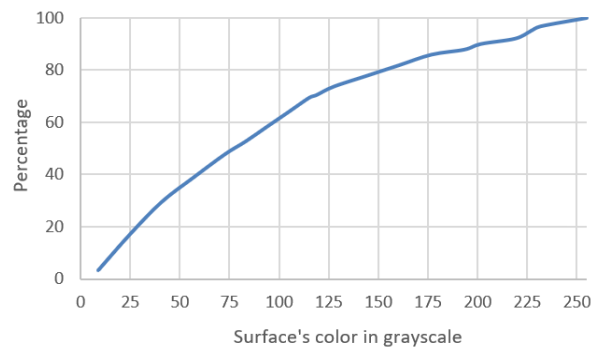


Figure. 6 Relation between surface's color and percentage of the reflected light's intensity

Lights that point to x is not only came from Ψ but also came from surround it, so the equation is modified to:

$$dL_{reflected} = f_r(x, \Psi \rightarrow \Theta)L(x \leftarrow \Psi)d\omega_\Psi \cos \Theta \quad (3)$$

Since the intensity of the light that came from x to Θ will be affected by the lights in the Ω_x sphere, so the equation is modified as below:

$$L(x \rightarrow \Theta) = \int_{\Omega_x} f_r(x, \Psi \rightarrow \Theta)L(x \leftarrow \Psi) \cos \Theta d\omega_\Psi \quad (4)$$

There are some parameters that affect the percentage of intensity from reflected light. Those

Table 1. Example of rules used to the first fuzzy inference

No	Rule
1	IF c is Very Dark AND g is Very Dull AND s is Very Rough THEN Bouncing Percentage is Very Low
2	IF c is Very Dark AND g is Very Dull AND s is Rather Rough THEN Bouncing Percentage is Very Low
3	IF c is Dark AND g is Dull AND s is Rough THEN Bouncing Percentage is Low
4	IF c is Light AND g is Normal AND s is Normal THEN Bouncing Percentage is Quite High
5	IF c is Bright AND g is Normal AND s is Quite Smooth THEN Bouncing Percentage is High
6	IF c is Very Bright AND g is Quite Glossy AND s is Very Smooth THEN Bouncing Percentage is Very High

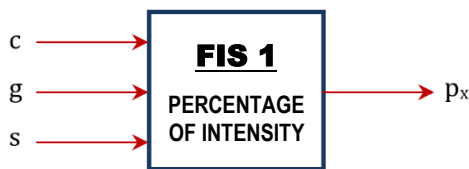


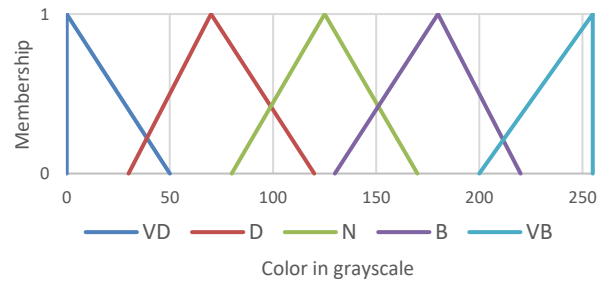
Figure. 7 Fuzzy logic diagram to determine the percentage of intensity of the reflected light that influenced by the glossy level, smoothness level, and color of the surface

parameters are the glossy level, the smoothness level, and the color of the surface where the light was bounced. A fuzzy logic is implemented in this article to gain the proper percentage of reflected light's intensity.

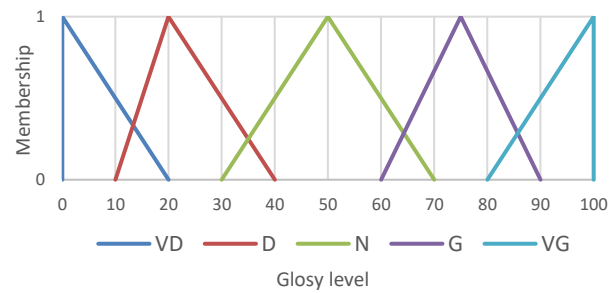
According to the result of the experiment using photometer, there is a significant relation between light's colour and reflected intensity. Fig. 6 shows the relation between the colour of the surface and the percentage of the reflected light's intensity. X-axis represents the colour of the object's surface where the light is bounced, and the y-axis represents the percentage of the reflected light's intensity. Scale 0 in the x-axis means the colour is black (#00000000) and scale 255 means white (#FFFFFFF).

Three conditions of the surface where incoming light is bounced are combined as inputs for the fuzzy logic diagram to gain the proper percentage of the reflected light. Fig. 7 shows the fuzzy logic diagram, where *c* is the color of the surface, *g* is the glossy level of the surface, *s* is the smoothness level of the surface, and p_x is the percentage of the reflected light on point *x*.

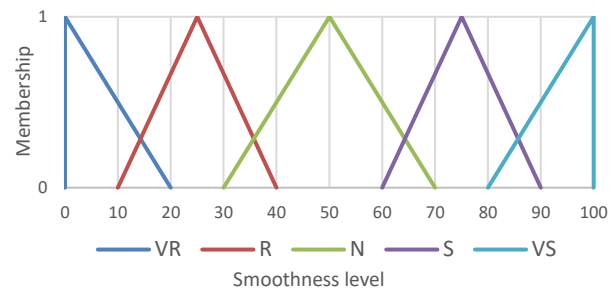
There are 125 rules used in this fuzzy inference. The formation of these rules can be done by an expert



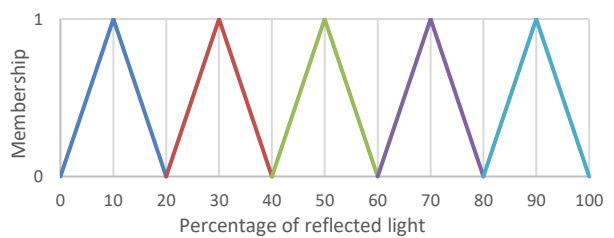
(a)



(b)



(c)



(d)

Figure. 8 Fuzzy membership function: (a) colour of the surface in grayscale, (b) glossy level of the surface, (c) smoothness level of the surface, and (d) percentage of the intensity of the reflected light

using a photometer by considering the weight of each predetermined criterion. Examples of decision rules used in this study are given in Table 1.

To facilitate calculations, two of three parameters that become inputs, which are *g* (glossy level) and *s* (smoothness level) of the surface, and p_x (percentage of reflected light) as an output will be described on the x-axis in the fuzzy membership function with a

range between 0 to 100. The third input, which is the colour of the surface (in grayscale), will be described with a range between 0 to 255. The label given for c is {very dark (VD), dark (D), neutral (N), bright (B), very bright (VB)}, g is {very dull (VD), dull (D), neutral (N), gloss (G), very gloss (VG)}, and s is {very rough (VR), rough (R), neutral (N), smooth (S), very smooth (VS)}.

In the fuzzy logic structure, the c , g , and s parameters are denoted as x_1 , x_2 , and x_3 , whereas p_x as the output is denoted by u . The rules of the relation are shown in Eq. 5.

$$R_{11} = (x_{11} \cdot u_{11}) \cup (x_{12} \cdot u_{12}) \cup \dots \cup (x_{1n} \cdot u_{1n}) \quad (5)$$

$$\begin{aligned} R_{12} &= (x_{11} \cdot u_{21}) \cup (x_{12} \cdot u_{22}) \cup \dots \cup (x_{1n} \cdot u_{2n}) \\ &: \\ R_{32} &= (x_{31} \cdot u_{31}) \cup (x_{32} \cdot u_{32}) \cup \dots \cup (x_{3n} \cdot u_{3n}) \end{aligned}$$

Then the equation of the output from the fuzzy logic structure will be as shown in Eq. 6.

$$u^* = (x_1 \circ R_{11}) \cap (x_2 \circ R_{21}) \cap \dots \cap (x_3 \circ R_{31}) \quad (6)$$

After the percentage p_x value is obtained, further will be combined into Eq. 4 to get the intensity of the reflected light at point x . Thus, only p_x percent of the result of Eq. 4 will be used in the next process.

$$\begin{aligned} L(x \rightarrow \Theta) \\ = p_x \int_{\Omega_x} f_r(x, \Psi \rightarrow \Theta) L(x \leftarrow \Psi) \cos \Theta \, d\omega_\Psi \end{aligned} \quad (7)$$

Beside those three things, position of the light source is also affecting the intensity of the reflected light. This research uses the Eq. 8 to calculate the reflected radiance on the surface at position x_p .

$$s(x_p, v, l) = \int_{\Omega^+} \rho(x_p, v, v_i) (n \cdot v_i) s_i(x_p, v_i) \, dv_i \quad (8)$$

where v is the direction to view the radiance, l is the position (coordinate) of the light, ρ is denote the BRDF, n is normal surface at position x_p , while $s_i(x_p, v_i)$ is the incoming radiance from direction v_i to x_p .

A set of reflectance parameter $a(x_p)$ is used in a function ρ_c to describe the ρ parameter in BRDF, thus:

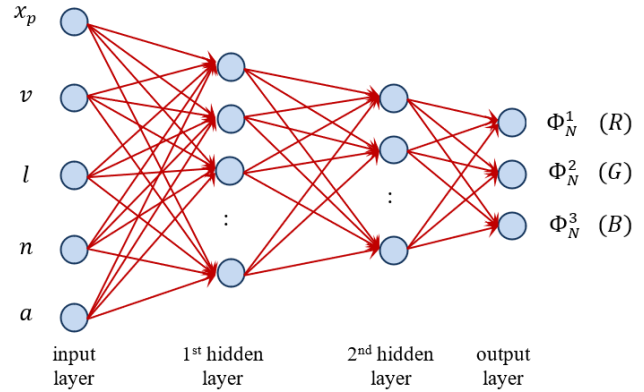


Figure. 9 Feed-forward neural network structure in RRF model which defines a mapping from the input vector (x_p, v, l, n, a) to the output RGB components

$$\rho(x_p, v, v_i) = \rho_c(v, v_i, a(x_p)) \quad (9)$$

Substituting Eq. 9 into Eq. 8, indirect illumination component can be express as below:

$$\begin{aligned} s^+(x_p, v, l) \\ = \int_{\Omega^+} \rho_c(v, v_i, a(x_p)) (n(x_p) \cdot v_i) s_i^+(x_p, v_i) \, dv_i \end{aligned} \quad (10)$$

Reference [9] used consists of a set of N input-output pairs that sample the indirect illumination $s^+(x_p, v, l)$ as the training data set. Each pair comprises the pair (x^i, y^i) , where $x^i = [x_p^i, v^i, l^i, n^i, a^i]^T$, $y^i = s^+(x_p^i, v^i, l^i)$, and $i = 1, \dots, N$. x^i is the input vector and y^i is the output vector. There are four components in the input vector that will affect the output vector: v (incoming light direction), l (incoming light position), n (normal surface), and a (vector represented by a set of texture map).

The least square error (Eq. 11) in the radiance regression function Φ should be minimized.

$$E = \sum_i \|y^i - \Phi(x_p^i, v^i, l^i, n^i, a^i)\|^2 \quad (11)$$

The dimension of the vector (x_p, v, l, n, a) correspond the nodes in the input layer, while there are three nodes in the output layer that correspond to the RGB color. There is a new vector w in every hidden layer which is weight of each node. The weight from node k in previous layer to node j in current layer will be written as w_{jk}^i . So, the output of this neural network RRF is $\Phi_N^i(x, w)$ where $i = 1, 2, 3$:

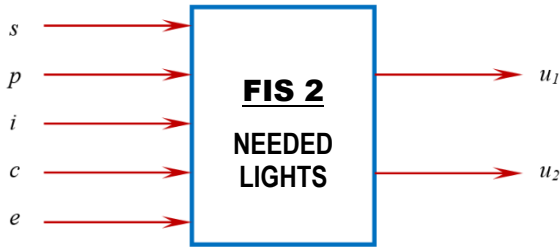


Figure. 10 Fuzzy structure to determine part of the model that needs to be more illuminated

Table 2. Example of rules used to the second fuzzy inference

No	Rule
1	IF <i>s</i> is Very Quiet AND <i>p</i> is High Angle AND <i>i</i> is Very Dark AND <i>c</i> is Medium Long Shot AND <i>e</i> is Very Sad THEN Needed Intensity is Very Dark AND Spot Angle is Quite Narrow
2	IF <i>s</i> is Neutral AND <i>p</i> is Eye Level AND <i>i</i> is Neutral AND <i>c</i> is Close Up AND <i>e</i> is Sad THEN Needed Intensity is Dark AND Spot Angle is Neutral
3	IF <i>s</i> is Crowded AND <i>p</i> is Eye Level AND <i>i</i> is Bright AND <i>c</i> is Medium Shot AND <i>e</i> is Neutral THEN Needed Intensity is Bright AND Spot Angle is Very Wide
4	IF <i>s</i> is Quiet AND <i>p</i> is Bird View AND <i>i</i> is Very Bright AND <i>c</i> is Extreme Long Shot AND <i>e</i> is Neutral THEN Needed Intensity is Bright AND Spot Angle is Very Wide
5	IF <i>s</i> is Very Quiet AND <i>p</i> is Eye Level AND <i>i</i> is Neutral AND <i>c</i> is Big Close Up AND <i>e</i> is Very Angry THEN Needed Intensity is Bright AND Spot Angle is Very Narrow

$$\Phi_N^i(x, w) = w_N^3 + \sum_{j>0} w_{ij}^3 \sigma \left(w_{j0}^2 + \sum_{k>0} w_{jk}^2 \sigma \left(w_{k0}^1 + \sum_{l=1}^9 w_{kl}^1 x_l \right) \right) \tag{12}$$

3.3 Decide the area that needs more illuminated

There are some issues that underlies decision making in adding and setting the additional light sources. This research got help from two movie directors and producers. From the interview, there are more than eight items that can influence the decision making. This research only involves five items in common as parameters: situation of the scene (scenario) (*s*), point of view of the active camera (*p*), intensity of attached light sources (*i*), camera position/distance (*c*), and emotion of the characters

(*e*). Those five items will become the parameter list that is taken from the storyboard. That list was scene by scene noted manually in this research. Fig. 10 shows the structure of the fuzzy system to determine the part that needs more illuminated (*u₁*) and the needed intensity (*u₂*).

The x-axis range of the membership function of every element (*s*, *c*, *i*, and *e*) except point of view (*p*) is 0 to 100 and is divided into five categories. The point of view is divided into seven categories. Element *s* expressed as *x₁*, *p* as *x₂*, *i* as *x₃*, *c* as *x₄*, *e* as *x₅*, additional light source’s position as *u₁*.

There are 4,375 rules used in this fuzzy inference. Examples of decision rules used for this fuzzy inference are given in Table 2.

To facilitate calculations, all parameters that become inputs will be described on the x-axis in the fuzzy membership function with a range between 0 to 100. The first output, which is the needed intensity, will be described with a range between 0 to 20. The second output, which is the spot angle, will be described with a range between 0 to 180. The range of intensity and spot angle is adjusted to the conditions at Unity. The label given for input *s* is {very quiet (VQ), quiet (Q), neutral (N), crowded (C), very crowded (VC)}, *p* is {bird view (BV), high angle (HA), eye level (EY), low angle (LA), worm view (WV)}, *i* is {very dark (VD), dark (D), neutral (N), bright (B), very bright (VB)}, *c* is {extreme long shot (EL), long shot (L), medium long shot (ML), medium shot (M), close up (C), big close up (BC), extreme close up (EC)}, and *e* is {very sad/angry (VS), sad/angry (S), neutral (N), happy/calm (H), very happy/calm (VH)}.

In the fuzzy logic structure, the *s*, *p*, *i*, *c*, and *e* parameters are denoted as *x₁*, *x₂*, *x₃*, *x₄*, and *x₅*, whereas the needed intensity and the light spot angle as the outputs are denoted by *u₁* and *u₂*. The rules of the relation are shown in Eq. 13.

$$R_{11} = (x_{11}x_{u_{11}}) \cup (x_{12}x_{u_{12}}) \cup \dots \cup (x_{1n}x_{u_{1n}}) \tag{13}$$

$$R_{12} = (x_{11}x_{u_{21}}) \cup (x_{12}x_{u_{22}}) \cup \dots \cup (x_{1n}x_{u_{2n}})$$

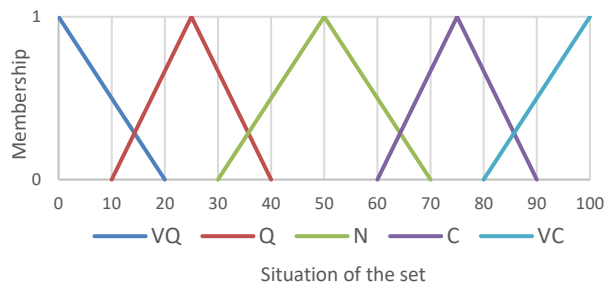
$$\vdots$$

$$R_{52} = (x_{51}x_{u_{21}}) \cup (x_{52}x_{u_{22}}) \cup \dots \cup (x_{5n}x_{u_{2n}})$$

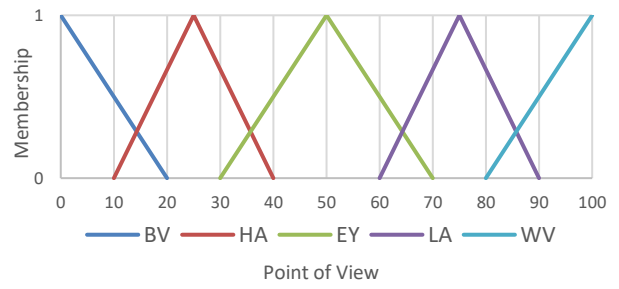
Then the fuzzy output:

$$u_1^* = (x_1 \circ R_{11}) \cap (x_2 \circ R_{21}) \cap \dots \cap (x_5 \circ R_{51}) \tag{14}$$

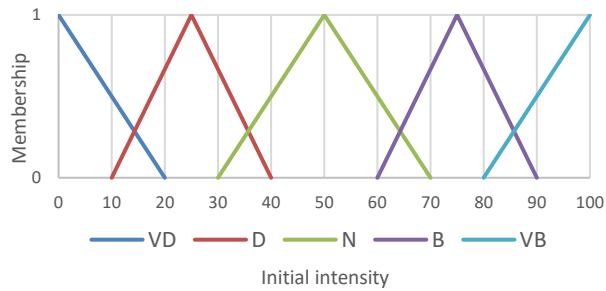
$$u_2^* = (x_1 \circ R_{12}) \cap (x_2 \circ R_{22}) \cap \dots \cap (x_5 \circ R_{52})$$



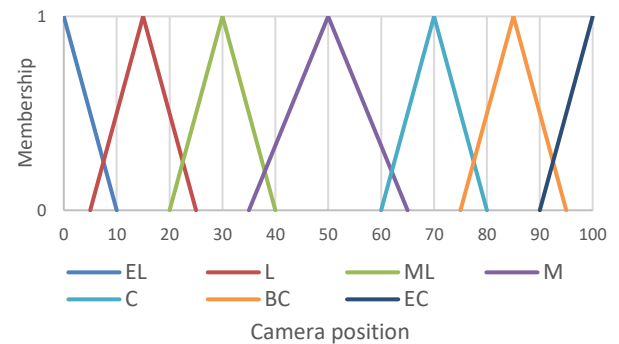
(a)



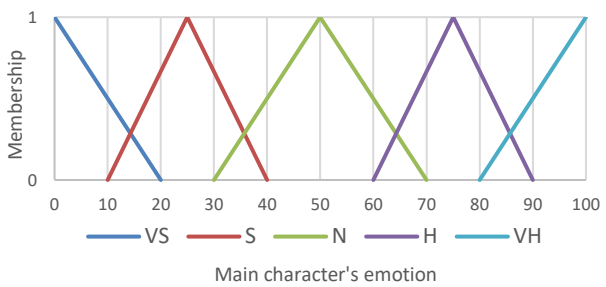
(b)



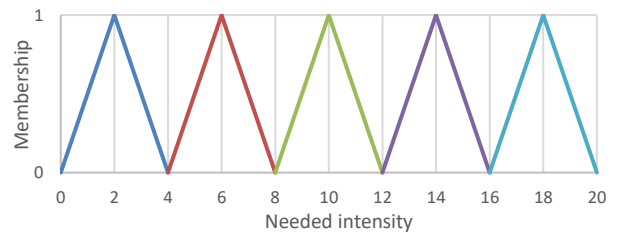
(c)



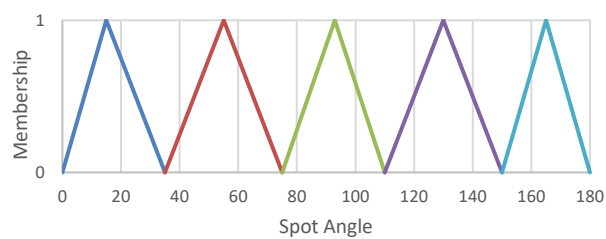
(d)



(e)



(f)



(g)

Figure. 11 Fuzzy membership function: (a) for the situation of the set, (b) point of view, (c) the intensity of the attached light sources, (d) emotion of the main character, (e) for the camera position, (f) the needed intensity, and (g) spot angle of the additional light sources.

Fig. 12 illustrates an example of the result of the decision to add more light in certain parts of the scene. The center of the illuminated area will be the coordinate of the destination point (x, y, z) and the radius of the circle will be the spot angle of the light source, while the distance between the destination point and the position of the additional

light source will be the light's range. the light source, while the distance between the destination point and the position of the additional light source will be the light's range.



Figure. 12 The example of the result of the decision to add more light in certain parts of the objects.

3.4 Calculate all the possible points

To place additional light sources, the room in the scene is divided into several points. The distance between points is within 0.5 on x, y, and z-axis. Fig. 13 illustrates the spread of the points to place all the possible position of the additional light sources.

After the script gets all possible point coordinates to place the additional light source, it will calculate the intensity of light which come from each point to the destination point. The intensity of the light source is depended on the needed intensity and spot angle of the area. To speed up the calculation process, all points that are behind the rearmost object will be eliminated first. This calculation process will leave one to three points which are adjacent on the calculation result. Fig. 14 shows the example result.

If there is only one remaining point, then this point will be directly used as the coordinate of the additional light position. If there are two remaining points, then these two points will be replaced by one point which is the nearest with the middle coordinate. If there are three remaining points, then those points will be replaced with a point which is the nearest with the coordinate of the center of gravity using Eq. 15.

$$\min_i \left(\sqrt{(x_i - x_0)^2 + (y_i - y_0)^2 + (z_i - z_0)^2} \right) \quad (15)$$

where (x_i, y_i, z_i) is the coordinate (position) of the additional lights and (x_0, y_0, z_0) is the coordinate of the center of gravity. If the remainder points are more than three, then those points will be sorted by distance, and only three top points will be kept as a result.

3.5 Add all additional light sources

The last step of our research is to bring the output which is all possible combinations of lighting set to the designer and let him/her chooses the best one. Fig. 15 shows the example of the output.

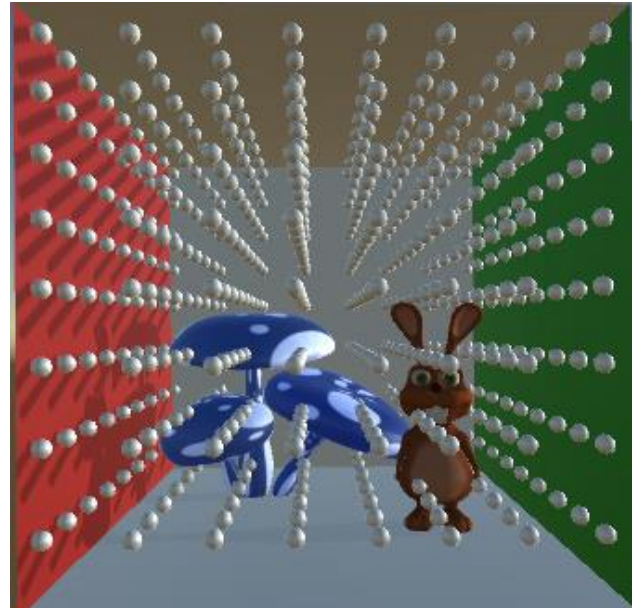


Figure. 13. Room/space of the scene is divided into several possible light points

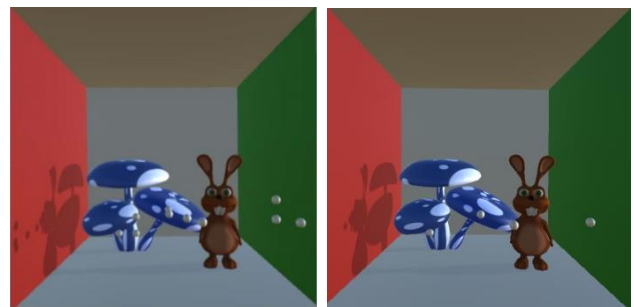


Figure. 14 Example of the calculation result: (a) the remaining light spots and (b) after the substitution

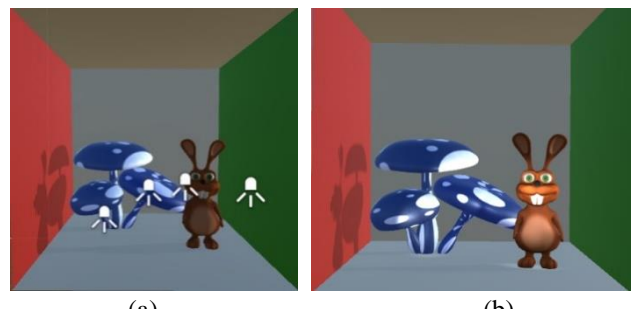


Figure. 15 The output of the script: (a) is which looks at the design view and (b) which appear in the play view

4. Result

This article uses 20 scenes from two short animation movies as the object of the research. We use 8 scenes created using Cornell Box and 12 scenes using a free design. From the twenty scenes, we created several conditions so that this research could be carried out. The conditions to be considered are the number of objects in one scene, the position of the

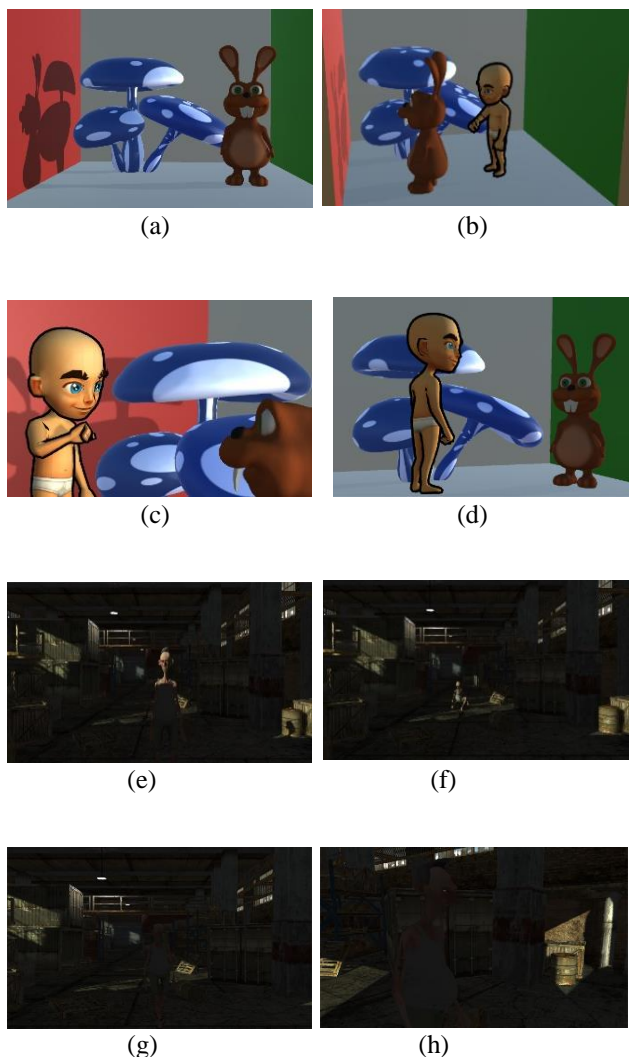


Figure. 16 The example scenes that used in this research: (a) Mr. Rabbit stands alone in front of mushrooms, (b) Mr. Rabbit meets a boy, (c) a boy talks to Mr. Rabbit in close-up camera, (d) a boy talks behind Mr. Rabbit with a camera with a bird view position, (e) a man stands alone in the storage, (f) long shot of camera, (g) medium shot of camera, and (h) close-up camera view.

active camera, the desired intensity of light, the Point of View of the active camera, and the emotion of the main characters. Fig. 16 shows the example scenes that been used in this article. Scene (a) to (d) are taken from animation movie that uses the Cornell box while scene (e) to (h) are taken from another movie that uses a free design.

Here should be 20 lists of parameters in this article since there are 20 scenes used in this research, but only one list shown in this article as an example. Table 3 shows the example of the parameters that affect the calculation to make the decision in adding the additional light sources into scene (b).

The autonomous lighting process in the SPICE system is divided into three stages, which are scene

Table 3. Rules for objects' name and tag

Object	Prefix	Tag's Name
Characters	char_	Character
Supporting objects	supp_	Support
Installed cameras	cam_	Camera
Installed light sources	lgt_	Light
Planes	pln_	Plane
Interface	ui_	UI

Table 4. Required attributes of each type of objects

Object Type	Attribute	Sub Attribute
Character	Position	X, Y, and Z
	Rotation	X, Y, and Z
	Dimension	X, Y, and Z
Support	Position	X, Y, and Z
	Rotation	X, Y, and Z
	Dimension	X, Y, and Z
Camera	Position	X, Y, and Z
	Rotation	X, Y, and Z
	Point of View	-
Directional light	Position	X, Y, and Z
	Rotation	X, Y, and Z
	Intensity	-
	Colour	-
Point light	Position	X, Y, and Z
	Rotation	X, Y, and Z
	Intensity	-
	Colour	-
	Range	-
Spotlight	Position	X, Y, and Z
	Rotation	X, Y, and Z
	Intensity	-
	Colour	-
	Range	-
	Spot Angle	-

exploration, calculation of light intensity, and addition of light sources.

4.1. Scene exploration

In this study we used six types of objects, which are characters (emphasized on main characters), supporting objects (including other characters that are not as main characters), installed cameras (both active and not), mounted light sources, planes (for map formation), and interface. To facilitate the exploration process, an agreement was made in giving prefixes of the object's names in the scene as shown in Table 3.

As a first step, the system will run a script that is placed in the start () section. This script will explore the scene to get the needed parameters, which are in the XML file which is the recording from the Parameters Extractor application and from the scene itself. After all of the objects in the scene are obtained, the next step is to get all the attributes owned by each object. Table 4 presents the required attributes of each type of object.

4.2. Lights’ intensity calculation

The next stage is to calculate the intensity of the light on the surface of the character type objects which facing the active camera. The intensity of the direct light can be calculated using Eq. (1). There are two main characters in Fig. 17 which are the boy and Mr. Rabbit, three mushrooms as the supported objects, and one attached light source in scene 2. The centre of the coordinate of the boy was described as (x_1, y_1, z_1) and Mr. Rabbit as (x_2, y_2, z_2) while the attached light source as (x_0, y_0, z_0) . Table 5 shows the objects in scene 2 and Table 6 shows the parameters from Parameter Extractor.

The membership function calculation process is performed using the triangle function $T(x, a, b, c)$ using the definition as in Eq. 16. We use the triangle membership function because the changes in the parameters are not too significant, so using the linear function can meet the expectations. For colour in grayscale, the value in scene 2 is 82.4730, so the membership function can be illustrated in Figure 18. The membership degree values of c_2 and c_3 at $x =$

Table 5. The value of the parameters in scene 2

Object’s Name	Attribute	Value
lgt_top	Position – X	-2.76000
	Position – Y	5.28000
	Position – Z	-0.33000
	Intensity	8.00000
char_boy	Position – X	0.05300
	Position – Y	0.64900
	Position – Z	0.27300
char_rabbit	Position – X	0.00000
	Position – Y	0.00000
	Position – Z	1.78519

Table 6. Attribute from parameter extractor

Parameter	Value
Colour	82.4730
Glossy level	65.0000
Smoothness level	60.0000

82.4730 can be calculated using Eq. 16 and obtained value of $u_{T_{c2}} = 0.75054$ and $u_{T_{c3}} = 0.05497$.

$$T_{(x,a,b,c)} = \begin{cases} 0 & ; x < a \\ \frac{x-a}{b-a} & ; a \leq x \leq b \\ \frac{c-x}{c-b} & ; b \leq x \leq c \\ 0 & ; x > c \end{cases} \quad (16)$$

In the same way, the μT value of each parameter can be obtained as shown in Table 7.

The fuzzy set as input for the first FIS for scene 2 can be seen in Table 8.

After all μT values are obtained, the next step is to calculate each α value using Eq. 17. The examples of the significant value of α are presented in Table 9.

$$\alpha_n = \min(\mu_{T_c}, \mu_{T_g}, \mu_{T_s}) \quad (17)$$

By using Eq. 18 which is the opposite of Eq. 16 we get the x value as shown in Table 10.

$$x = \begin{cases} a + \mu_T(b - a) & ; a \leq \mu_T \leq b \\ c - \mu_T(c - b) & ; b \leq \mu_T \leq c \end{cases} \quad (18)$$

Then Eq. 19 is used in the defuzzification stage to calculate the centre of gravity as a basis for calculations using the Center of Area (COA) or centroid method, as shown in Fig. 19.

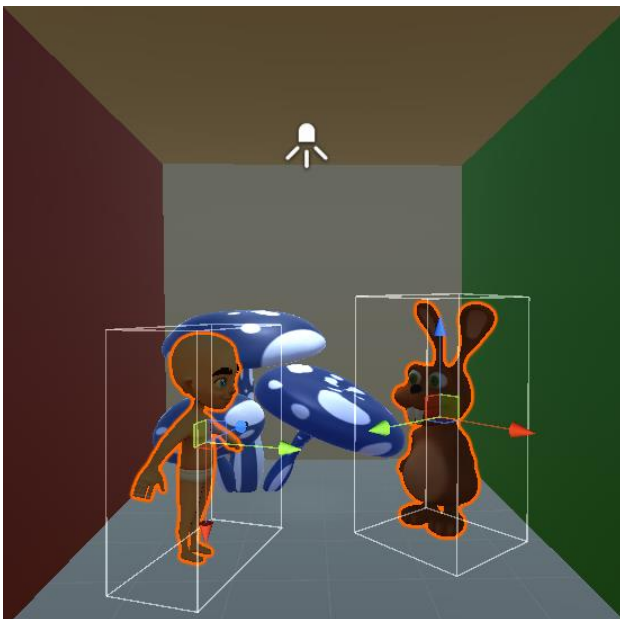


Figure. 17 Situation in scene 2 with two main characters, three supported objects, and one attached light source

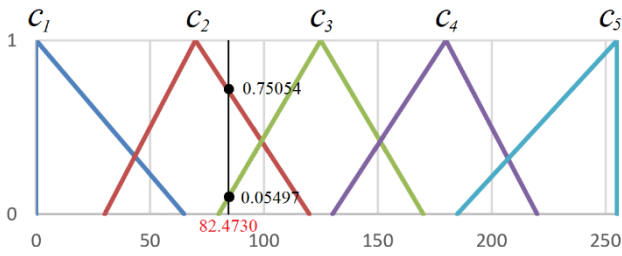


Figure. 18 The curve of the membership degree of the parameter c in the triangle function

Table 7. The membership degree value of scene 2

Parameter	x	Value of μ_T	
c	82.4730	$\mu_{T_{c2}} = 0.75054$	$\mu_{T_{c2}} = 0.05497$
g	65.0000	$\mu_{T_{g3}} = 0.20000$	$\mu_{T_{g4}} = 0.50000$
s	60.0000	$\mu_{T_{s3}} = 0.28571$	$\mu_{T_{s4}} = 0.66667$

Table 8. The first fuzzy set of scene 2

Parameter	x	Value of μ_T
c	82.4730	{0, 0.75054, 0.05497, 0, 0}
g	65.0000	{0, 0, 0.20000, 0.50000, 0}
s	60.0000	{0, 0, 0.28571, 0.66667, 0}

Table 9. The examples of the significant value of α

α	Value of α
α_1	$\min\{0.05497, 0.20000, 0.28571\} = 0.05497$
α_2	$\min\{0.17564, 0.75054, 0.50000\} = 0.17564$
α_3	$\min\{0.28571, 0.66667, 0.75054\} = 0.28571$
α_4	$\min\{0.19647, 0.20000, 0.40036\} = 0.19647$
α_5	$\min\{0.50428, 0.20000, 0.66667\} = 0.20000$
α_6	$\min\{0.75054, 0.50000, 0.66667\} = 0.66667$

Table 10. The examples of the value of x for each rule

Rule	Value of x_1	Value of x_2
1	82.4730	167.5270
2	34.5197	65.4803
3	53.7684	86.2316
4	57.1605	77.8395
5	62.7403	82.2597

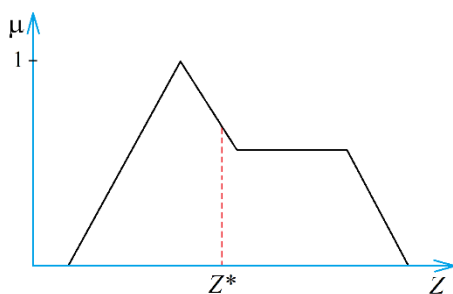


Figure. 19 Visualization of Eq. (21)

$$Z^* = \frac{\int \mu_x(z) \cdot z dz}{\int \mu_x(z) dz} \quad (19)$$

where $\int \mu_x(z) \cdot z dz$ is a moment and $\int \mu_x(z) \cdot dz$ is the area of the curve. From this example, the area of the curve is 19.8012 and the moment is 310.8254, so that the value of z^* for the percentage of reflected light is 15.6973.

After calculating the intensity of light that hit the surface of the character type of object, the next step is to identify how much light intensity is needed in each of these objects. For this reason, calculations are performed using the second FIS which is almost the same as the first FIS but uses different parameters. To get these parameters we use the web-based Parameter Extractor application. Table 11 shows the parameters acquired using that application.

The membership function calculation process is performed using the definition as in Eq. 16. The map situation value in scene 2 is 50, so the membership function can be illustrated in Figure 20. In this situation, we got only one value for s_3 at $x = 50$ which is 1 since this coordinate is at the top of the triangle.

In the same way, the μ_T value of each parameter can be obtained as shown in Table 12.

The fuzzy set as input for the second FIS for scene 2 can be seen in Table 13.

Like at the first FIS, after all μ_T values are obtained then each α is calculated using Eq. 20. The examples of the significant value of α are presented in Table 14.

$$\alpha_n = \min(\mu_{T_s}, \mu_{T_p}, \mu_{T_i}, \mu_{T_c}, \mu_{T_e}) \quad (20)$$

Then we use Eq. 18 to get Table 15.

In the defuzzification stage we use Eq. 19 to calculate the center of gravity. From this example, the area of the curve is 17.5612 and the moment is 89.913344, so that the value of z^* for the needed intensity on the surface of char_rabbit in scene 2 is 5.12. Because this value is the intensity of the light that is on the surface of the object, while the location of the light source is at some distance from the object, it is necessary to calculate the amount of light intensity coming out of the additional light source.

In this study, the SPICE system uses a value of 3 in determining the range of all additional light sources. From the SPICE system script, it was found that the coordinate of char_rabbit is (-4.5, 0.02999997, 0.51). Since the intensity of the additional light source on the surface of the object is 5.12, it will be multiplied by 3 to gain the intensity of the additional light source itself, which is 15.36. This

Table 11. List of the needed parameters from scene 2

Parameter	Value
Situation	50
Point of View	40
Current intensity	30
Camera distance	45
Character emotion	{55, 75}

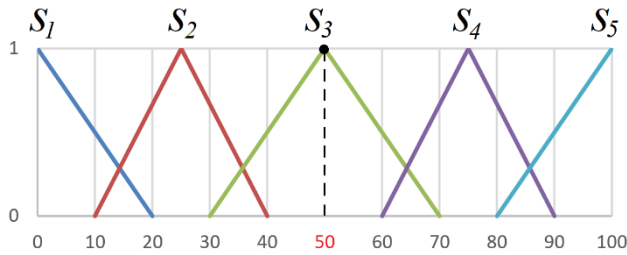


Figure. 20 The curve of the membership degree of the parameter s in the triangle function

Table 12. The membership degree value of scene 2 for char_rabbit object

Parameter	x	Value of μ_T	
s	50	$\mu_{T_{s3}} = 1.00000$	$\mu_{T_{s3}} = 1.00000$
p	40	$\mu_{T_{p2}} = 0.00000$	$\mu_{T_{p3}} = 0.50000$
i	30	$\mu_{T_{i2}} = 0.66667$	$\mu_{T_{i3}} = 0.00000$
c	45	$\mu_{T_{c3}} = 0.00000$	$\mu_{T_{c4}} = 0.66667$
e	75	$\mu_{T_{e4}} = 1.00000$	$\mu_{T_{e4}} = 1.00000$

Table 13. The second fuzzy set of scene 2

Parameter	x	Value of μ_T
s	50.0000	{0, 0, 1.0000, 0, 0}
p	40.0000	{0, 0, 0.50000, 0, 0}
i	30.0000	{0, 0.6667, 0, 0, 0}
c	45.0000	{0, 0, 0, 0.6667, 0}
e	75.0000	{0, 0, 0, 1.0000, 0}

Table 14. The examples of the significant value of α in the second FIS

α	Value of α
α_1	$\min(1.0000, 0.5000, 0.6667, 0.6667, 1.0000) = 0.5000$
α_2	$\min(0.3333, 0.6667, 0.5000, 0.2500, 0.5000) = 0.2500$
α_3	$\min(0.5000, 0.2000, 0.3500, 0.3333, 0.5000) = 0.2000$
α_4	$\min(0.6667, 0.2000, 0.2500, 0.5000, 0.5000) = 0.2000$
α_5	$\min\{1.0000, 1.0000, 0.3333, 0.5000, 1.0000\} = 0.30000$

number is used as the value of the intensity of the added light source.

Table 15. The examples of the value of x for each rule

Rule	Value of x_1	Value of x_2
1	79.6108	132.0267
2	46.7316	58.6347
3	61.9633	87.6418
4	53.6241	71.687
5	84.9617	94.6872

4.3. The addition of the light sources

To add the additional light sources that fit this need, then there are some additional calculations that must be done, which are the determination of the position of additional light sources and its rotation. We use ordinary vector formulas to determine the position and rotation of additional light sources as shown in Figure 21 and Eq. 21. The value of y_1 is equal to y_0 because the height of the additional light source is equated with the height of the illuminated object.

$$|R| = \frac{|A| \sin \alpha}{\sin \beta} \tag{21}$$

where $|A|$ is the distance from origin point O to object coordinate (x_0, y_0, z_0) , $|B|$ is the x distance from O to additional light source coordinate which is the

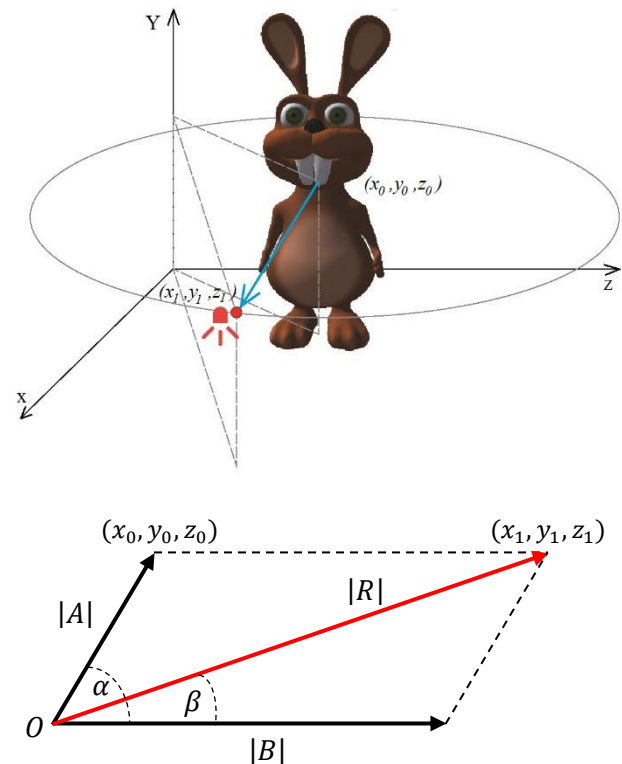


Figure. 21 The position and rotation of the object and the additional light



Figure. 22 The example of the comparison: (a) original design and (b) after additional light sources addition

addition of x_0 and 3 (light range), and $|R|$ is the distance to the additional light source on coordinate (x_1, y_1, z_1) .

The additional light sources are added to the scenes based on the output of this SPICE system. The result of the comparison between before and after being given additional light sources into scene 2 can be seen in Figure 22.

After adding some light sources into the scene, the next step is to provide an assessment of the scene that has been updated lighting system through the SPICE system. This assessment process is carried out by showing a new scene design to several designers to be given an assessment manually whether it is in accordance with what is expected or not.

5. Conclusion

The output of this system is the scene that has been updated by adding additional light sources automatically and will then be assessed manually by several designers. Of the 20 scenes used in this study it was found that there were 16 scenes or 80% could fulfil the desires of these designers. For more details, the results of this study can be seen in Table 16. Scene numbers that marked with * are scenes that do not meet the desires of the designer or in other words the results of this application do not match what the designer is doing manually.

If further observed, the four scenes that have not fulfilled the desire of the designers have the long shot and extreme-long shot types of point-of-view. This can provide a bias in the calculation process because of the considerable distance between the position of the camera and the object to be given additional light.

However, our research has been able to provide good results in the lighting process in 3D cinematography. The SPICE system that we propose can provide additional light sources to the scene and adjust it automatically. Thus, the designers can save time in the process of designing animated films because it does not need to be bothered with the setting of additional light sources.

Table 16. Output values from SPICE system

Scene	Position	Spot Angle	Intensity
1	(-4.045, 0.03, 1.959)	48.44971	13.04
2	(-4.5, 0.03, 0.51)	63.20537	15.36
	(-1.14, 0.05, 1.54)	65.30439	14.81
3	(-2.05, 0.05, 2.21)	51.68712	16.77
4	(-4.31, 1.23, -0.2)	48.89775	10.04
5	(-5.96, 1.23, 0.42)	59.91657	18.26
6*	(-2.753, 1.387, 2.985)	44.30624	16.70
	(-6.05, 2.81, -1.68)	61.68726	15.26
7	(-0.058, 0.698, 2.127)	52.87634	12.70
8	(1.02, 1.387, 4.019)	58.73184	16.37
	(-7.41, 1.23, 1.119)	61.79428	15.62
9	(-2.711, 1.58, 2.911)	42.37942	11.84
10	(1.168, 1.387, 6.027)	57.40078	13.19
11	(3.881, 2.834, -2.204)	52.85089	15.28
12	(2.911, 1.387, -2.921)	45.16779	16.21
	(3.881, 2.834, -2.204)	62.7871	13.78
13	(3.831, 2.834, -4.072)	48.79345	6.54
14	(4.852, 3.529, -1.604)	58.14476	8.34
15*	(5.618, 1.387, -6.812)	63.3278	20.49
	(3.646, 2.834, -3.772)	64.87796	18.77
16	(3.102, 1.387, -5.119)	50.76471	11.01
	(3.565, 2.834, -4.502)	57.88713	13.76
17	(8.697, 6.387, -1.509)	40.76471	15.26
18*	(5.739, 0.761, 1.228)	62.57128	18.91
	(8.866, 1.676, -5.562)	58.2267	17.76
19	(3.823, 1.235, -4.649)	46.14622	12.50
20*	(3.326, 5.964, -5.356)	52.83081	16.85

Conflicts of Interest

The authors declare no conflict of interest.

Author Contributions

Conceptualization, Mochamad Hariadi and Mauridhi H. Purnomo; methodology, Andreas and Kunio Kondo; software, Andreas; validation, Andreas; formal analysis, Andreas, Mochamad Hariadi and Maudidhi H. Purnomo; writing—original draft preparation, Andreas; writing—review and editing, Andreas and Mauridhi H. Purnomo; visualization, Andreas and Mochamad Hariadi; supervision, Mochamad Hariadi, Maudidhi H. Purnomo and Kunio Kondo; project administration, Andreas.

References

- [1] B. Brown, *Cinematography Theory, and Practice: Image Making for Cinematographers and Directors*, Second Edition, Elsevier, 2012.
- [2] T. R. Kol, O. Klehm, H-P. Seidel, and E. Eisemann, "Expressive Single Scattering for Light Shaft Stylization", *IEEE Transactions on*

- Visualization and Computer Graphics*, Vol. 23, No. 7, pp. 1753-1766, 2017.
- [3] I. S. L. Moh and S. F. M. Zaidi, "Game Cinematography – Towards Understanding Relationship between Spatial Distortion and Game-play", In: *Proc. of the 2019 3rd International Conf. on Virtual and Augmented Reality Simulations*, pp. 26-32, 2019.
- [4] Y. Kanematsu, R. Motegi, K. Mikami, K. Kondo, "Direction Aided Light Set for 3DCG Production", *Journal of Graphic Science of Japan*, Vol. 50, No. 3, pp. 3-9, 2017.
- [5] R. Montes and C. Urena, "An Overview of BRDF Models", Technical Report University of Granada, 2012.
- [6] J. Jendersie, D. Kuri and T. Grosch, "Precomputed Illuminance Composition for Real-Time Global Illumination", In: *Proc. of 20th ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, pp. 129-137, 2016.
- [7] M. McGuire, M. Mara, D. Nowrouzezahrai, and D. Luebke, "Real-Time Global Illumination Using Precomputed Light Field Probes", In: *Proc. of 21st ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, Art. No.2, pp. 1-11, 2017.
- [8] W.C. Lin, T.S. Huang, T.C. Ho, Y.T. Chen, and J.H. Chuang, "Interactive Lighting Design with Hierarchical Light Representation", In: *Proc. of EGSR'13 Proc. of the Eurographics Symposium on Rendering*, Vol. 32, No. 4, pp. 133-142, 2013.
- [9] F.J. Luan, *Real-Time Global Illumination with Radiance Regression Functions*, Final Project Cornell University, 2014.
- [10] L. Belcour, K. Bala, and C. Soler, "A Local Frequency Analysis of Light Scattering and Absorption", *ACM Transactions on Graphics Journal*, Vol.33, Issue 5, Art. No. 163, pp. 1-17, 2014.
- [11] P. Ren, J. Wang, M. Gong, S. Lin, X. Tong, and B. Guo, "Global Illumination with Radiance Regression Functions", *ACM Transactions on Graphics Journal*, Vol.23, Issue 4, Art. No. 130, pp. 1-12, 2013.
- [12] I. Gkioulekas, S. Zhao, K. Bala, T. Zickler, and A. Levin, "Inverse Volume Rendering with Material Dictionaries: Supplementary Material", *ACM Transactions on Graphics Journal*, Vol.32, Issue 6, Art. No. 162, pp. 1-13, 2013.
- [13] L. Kallberg and T. Larsson, "Optimized Phong and Blinn-Phong Glossy Highlights", *Journal of Computer Graphics Techniques (JCGT)*, Vol. 3, No. 3 pp. 1-6, 2014.