Rendering Virtual Shadow Puppet Theatres

Catherine Taylor
cct43@bath.ac.uk
Department of Computer Science,
University of Bath

Ken Cameron
K.M.Cameron@bath.ac.uk
Department of Computer Science,
University of Bath

Motivation: Shadow puppet theatres are a traditional tool in storytelling, particularly in countries such as China and Malaysia, where they have been named as pieces of 'intangible heritage' by UNESCO [3]. However, the popularity of this art form has decreased significantly in recent years due to the complexity and skill required to produce a successful performance and an increased disinterest from children. The development of a virtual shadow puppet theatre is a method to modernise this tradition. Previous work to create a virtual shadow puppet theatre has been focused on controlling the movement of the virtual puppet and have used texture mapping to model the appearance of the puppet or have created stylised shadows, rather than rendering photo-realistic images [3]. This paper proposes a novel approach to render photo-realistic shadow puppets in real-time, by modelling the shadow projection within shadow maps and calculating the visibility at each point of the scene using summed area tables. Perceptual testing of our algorithm has concluded that the images produced are perceived to be more likely to be real than photographs of real shadow puppets.

Shadow Puppet Model: After experimentation with a real shadow puppet theatre, a model describing the behaviour of shadow puppets was developed. The key features are as follows. The shadow is a perspective projection of the puppet onto the screen and, due to global illumination, as the distance between the screen and the puppet increases the shadow intensity decreases. The screen is a matte surface and so the lighting on the screen can be represented using the Blinn-Phong lighting model [1], where the components are ambient and diffuse reflection. An incandescent bulb was used as a light source, creating a shadow with both an inner and outer umbra and penumbra. These were named *soft shadow puppets*.

Shadow Puppet Algorithm: The real-time software was created using the graphics API, OpenGL. Using this framework, a large quantity of calculations can be carried out on the GPU. The virtual puppet is modelled using a quad with a scanned image of a real puppet mapped to it. The incandescent bulb can be modelled using two area light sources; the inner with higher intensity and smaller area and the outer with a lower intensity but larger area. For each light source, a shadow map was created by rendering the scene from the point of view of the light source. In contrast to traditional shadow maps, a perspective projection was applied to the puppet to accurately model the shadow projection as observed in the real theatre. Each fragment in the shadow map can store an RGB vector, which is black if the fragment corresponds to a point on the puppet and white otherwise, and the depth of the current fragment.

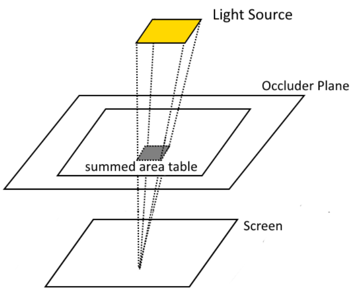

Figure 1: Finding the visibility function using a summed area table.

There exists a bijection between the light incident on the point on the screen and the region of intersection between the occluder plane and light frustrum as shown in Figure 1 [2]. This occluder plane has characteristic equation: $\delta : P = 1$ if P is on occluder, $P = 0$ otherwise, where P is a point on the screen. In contrast to the approach by Eismann and Decoret [2], the occluder plane in our method is calculated from the shadow map rather than from the occluding object itself. The shadow intensity at a point on screen, or *visibility function*, can be found by integrating $1 - \delta$ over the region where the frustrum of that point to the light source intersects with occluder [2]. Summed area tables (SATs) can be used to determine the ratio of shaded to total pixels within the region of intersection, in order to solve this integral. Each entry in a SAT is equal to the accumulated sum of pixel values which fall above and to the left of the current point. A SAT can be used to determine the total sum of pixel values within a rectangular portion of the original image and so a SAT created for a shadow map, where a point on the puppet is worth 1 and any other point is worth 0, can be used to calculate the number of shaded pixels in that region. The visibility information for both light sources can be stored as a texture. These are combined with the depth information, lighting model and screen texture, which can be modelled using texture mapping and a photograph of the real screen, in a final fragment shader to produce the virtual puppet.

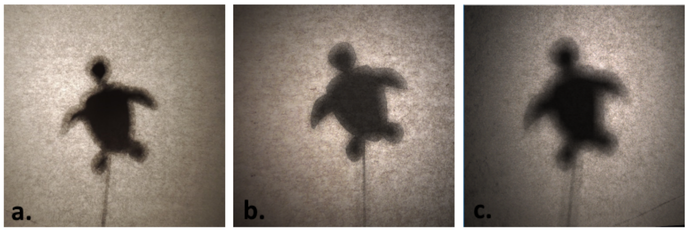

Figure 2: (a) photograph of a real shadow puppet, (b) ground truth image created using Monte-Carlo ray-tracing, (c) shadow puppet algorithm.

Results: The real-time software can render still images in HD at a rate of 259.55fps and animations at 51.87fps. A survey was constructed to evaluate the software output. 32 participants were shown pairs of images, consisting of photographs of real shadow puppets (R), computer generated ground-truth images (MC) and computer generated images using our proposed algorithm (PA), and asked to select the images they considered most likely to be real. A two-tailed t-test was used to analyse the results. The participants were shown two pairs of images, where each pair was made up of one MC and one PA. The expected mean number of PA selected was 1. The observed mean was $\bar{x} = 0.813$ with standard deviation $\sigma = 0.738$ and t-value $t = -1.438$. This is not significant in a 5% confidence interval and so the participants are unable to distinguish between these sets of images. The users were also shown two pairs of still images (one R and PA in each) and the following statistics found: $\bar{x} = 1.5, \sigma = 0.672, t = 4.209$. This t-value is significant and so a higher number of real-time generated images were considered likely to be real than photographs of real shadow puppets. Finally, two videos (MC + PA) were compared. This test had $\bar{x} = 0.212, \sigma = 0.415, t = -3.293$, suggesting the participants were able to tell that the rendered animation was computer generated.

Conclusion: The proposed shadow puppet rendering algorithm can be used to create realistic virtual shadow puppet theatres which can run in real-time. Future experimental studies should focus on the material of the puppet with research carried out on the effect the colour, translucency and transparency of the puppet has on the shadow produced.

[1] James F Blinn. Models of light reflection for computer synthesized pictures. In *ACM SIGGRAPH Computer Graphics*, volume 11, pages 192–198. ACM, 1977.

[2] Elmar Eisemann and Xavier Décoret. Plausible image based soft shadows using occlusion textures. In *SIBGRAPI'06*, pages 155–162. IEEE, 2006.

[3] Khor Kheng Kia and Yuen May Chan. A study on the visual styles of wayang kulit kelantan and its capturing methods. *CGIV2009*, pages 423–428, 2009.