2D Additive and Dynamic Shadows

Benjamin Bruneau ben.bruneau99@gmail.com, Matthias Segui Serera matsegui@gmail.com
ESGI School in Paris

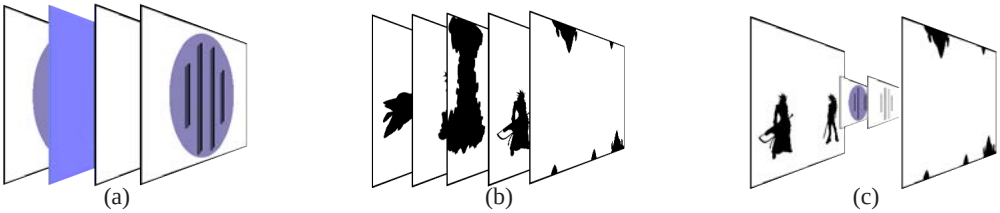

Figure 1: *(a) representation of the Diffuse Map in orange with its Volumetric Map (RGB) and its Self Shadow Map and the Diffuse Map merged with the Self Shadow Map. (b) representation of the Shape Shadow Maps in the Shadow Buffer ordered by their depth. (c) process of merging the Diffuse Map and its Self Shadow Map with the Shape Shadow Maps with a higher depth value.*

1. Introduction

In most 2D video games, shadows are directly integrated (baked) onto character animation sprites. With the success of HD games, we had to find a solution to reduce the space reserved by the multitude of HD sprites in memory. The 2D animated bones (Rayman Legends, Adobe After Effect, League of Legends Launcher) uses a decomposed and deformed sprite to create animations. With this technology, we cannot integrate the shadows directly on the sprites. We have to create a new system that allows a real time shadow baking regarding the sprite volumes. We propose a technology for real-time dynamic shadows on deformable sprites.

Keywords: 2D shadow, Dynamic Shadow, Additive Shadow,2D Self Shadowing, Shadow Map

2 2D Self Shadowing

To create real time shadows on a 2D HD sprite, the sprite need to have volume. Our method is to stock the properties of the volumes of the sprites in a “Volumetric Map” (VM) like we do with the Bump Map and then use this VM to compute the self sha-dowing [GREEN 2007] of the sprite. This Volumetric Map can be created with software like SpriteLamp or Nvidia Photoshop Texture Plugin or manually with Graphic Software like Photoshop. With the help of the Shadow Mapping technology [FERNANDO et al. 2001] used on the Volumetric Map, we get the self shadows of the 2D HD sprite, then we apply it on the sprite. We can apply the defor-mation of the animated 2D HD sprite created with the 2D Bones technology on the Volumetric map, in order to follow the anima-tion and create animated shadows in real time.

3 2D Projected Shadows

For the projection of a shadow of a 2D HD sprite on an 2D HD other sprite, our project computes the shape of the sprite in real time and creates a Shape Map (SM). The Shape Map represents the shape of the sprite in black and white plus alpha.

SIGGRAPH 2014, August 10 – 14, 2014, Vancouver, British Columbia, Canada.

ACM 978-1-4503-2958-3/14/08

The scale and position of the Shape map will be changed by the position of the light, linear and radial deformations, to simulate the shadow behaviour of the sprite. The deformed Shape map is called the Shadow Shape Map (SSM). When the Shadow Shape Map is computed, it’s stocked in the “Shadow Buffer” (SB) with all the other computed Shadow Shape Map of the other 2D HD sprites. In the Shadow Buffer, all the Shadow Shape Maps with the same Y Axis (2D Depth Property) are blended into one map called the “Shadow Depth Map” (SDM) which will be applied to the sprites in the shadow of this Shadow Depth Map. The program computes the application of the projected shadow in the Z axis order of the Shadow Depth Map, each Shadow Depth Map is blended with its upper Shadow Depth Map to simulate the additive shadows.

4 Merged 2D Self Shadow and Projected Shadows

The compute process of the projected shadow is done after the compute process of the self shadow. When in the process of the self shadow a shadow is out of the shape, we add it into a temporary Shape Map and the program blends it with the Shape Map of the Projected Shadow process. When the projected shadows and the self shadows are computed, the program applies the projected shadows to the 2D HD sprite, then the self shadows in order to bring out the lights which illuminate directly the sprite.

5 Future Works

One of our futur works is to reduce the computed time in real time of the dynamic and additive 2D shadows.

An evolution of our project is to apply the deformation of an animated 2D mesh to the Volumetric maps to modify the volume of the sprite in real time. In order to enhance the rendering of the shadows, an idea is to stock into the volumetric maps the properties of the material of the sprite, like in the Physically Based Shading technology to simulate the different shades of shadows.

References

FERNANDO, R., FERNANDER, S., BALA, K. , AND GREENBERG, D. P. 2001. Adaptative shadow maps. In Proceeding of ACM SIGGRAPH 2001, Computer Graphics Proceedings, Annual Conferences Series, 387-390.

GREEN, C. 2007. Efficient self-shadowed radiosity normal mapping. In Proceeding of ACM SIGGRAPH 2007, SIGGRAPH ’07 ACM SIGGRAPH 2007 Courses, 1-8.