Shade Trees

Robert L. Cook
Computer Division, Lucasfilm Ltd.
SIGGRAPH 1984

Presented by Patrick Smith
Motivation

• It is important for images to look realistic because:
  – Some applications require a high degree of realism
  – Realism is a measure of our techniques and understanding

• Proper shading makes images look more realistic, but “current” shading models only allow the same, fixed model to be applied to all surfaces
Proposed System

• A general approach to shading providing a language to describe surfaces and allow different shading techniques to be combined
• Modular approach assumes no shading technique is suitable for all surfaces
• Handles different complexities without extraneous computation
Traditional Approach

• An appearance parameter is any value that is used in the shading calculation
  – For example, surface normal, color of light, shininess, bump maps, etc.

• Traditional approach to shading uses two steps:
  1. Determine values of appearance parameters
  2. Use those values to determine fixed shading model to be applied to all surfaces
Shade Trees Approach

• Instead of one general equation, create an equation from a set of basic operations, organized as a tree
  – Each node in the tree is an operation that takes in some appearance parameter(s) and outputs some appearance parameter(s)

• Traverse the tree in post-order and the output of the root is the final color
More Trees

• Light Trees: different sources of light require different calculations/parameters, so the light is also modeled as a tree
  – Multiple shade trees can use the same light tree
• Atmosphere Trees: final output of shade tree is the color/intensity of light leaving a surface, and is called the exitance. But the exitance can be affected by some atmospheric effect (ie, haze) before it reaches your eye. An atmosphere tree takes the exitance as an input and outputs the color you really see.
Language Implementation

- Variables represent appearance parameters; instructions describe how to connect the nodes of the tree.

```plaintext
float a=.5, s=.5;
float roughness=.1;
float intensity;
color metal_color=(1,1,1);
intensity = a*ambient() +
           s*specular(normal,viewer,roughness);
final_color = intensity * metal_color;
```

Figure 1a. Shade tree for copper.
Uses of Shade Trees

- New uses for textures. Use texture to store surface normal instead of color for a particular orientation.
Uses of Shade Trees (cont’d)

Figure 3a. Grass Normal Texture Map.

Figure 3b. Grass Rendered with Textured Normals.
Uses of Shade Trees (cont’d)

- Often, we don’t care about the position of the light, just the position of its highlight on a surface
- A shade tree can be used to determine the position of the light given the desired position of the highlight

![Diagram](image)

Figure 1d. “Highlight at” branch of a light tree.
Uses of Shade Trees (cont’d)

• One of the most useful is a mix node, which uses one input to interpolate between the other two

• Allows a pattern of one material inside another or a single material that is not homogenous (for example, wood)

Figure 1b. The mix node in a shade tree for wood.
Results

Figure 6c. Bee Box
Results

Figure 4. Road to Point Reyes.
Results
Conclusion

• Shade trees allow quick and easy modification of appearance parameters
• They are flexible because they are not based on a general shading formula
• Shade trees are efficient because they only calculate what is necessary for each surface