

Texture Mapping

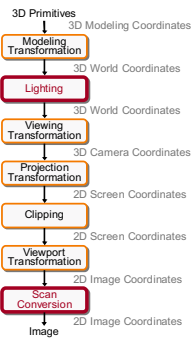
Adam Finkelstein
Princeton University
COS 426, Fall 2001

Textures

- Describe color variation in interior of 3D polygon
 - When scan converting a polygon, vary pixel colors according to values fetched from a texture

Angel Figure 9.3

3D Rendering Pipeline (for direct illumination)



Texture mapping

Surface Textures

- Add visual detail to surfaces of 3D objects



Polygonal model

With surface texture

Surface Textures

- Add visual detail to surfaces of 3D objects



[Daren Horley]

Overview

- Texture mapping methods
 - Parameterization
 - Mapping
 - Filtering
- Texture mapping applications
 - Modulation textures
 - Illumination mapping
 - Bump mapping
 - Environment mapping
 - Image-based rendering
 - Non-photorealistic rendering

Parameterization

geometry + image = texture map

- Q: How do we decide *where* on the geometry each color from the image should go?

Option: Varieties of projections

[Paul Bourke]

Option: unfold the surface

[Piponi2000]

Option: make an atlas

charts atlas surface

[Sander2001]

Overview

- Texture mapping methods
 - Parameterization
 - Mapping
 - Filtering
- Texture mapping applications
 - Modulation textures
 - Illumination mapping
 - Bump mapping
 - Environment mapping
 - Image-based rendering
 - Non-photorealistic rendering

Texture Mapping

- Steps:
 - Define texture
 - Specify mapping from texture to surface
 - Lookup texture values during scan conversion

Texture Coordinate System (s, t) → Modeling Coordinate System (u, v) → Image Coordinate System (x, y)

13

Texture Mapping

- When scan convert, map from ...
 - image coordinate system (x,y) to
 - modeling coordinate system (u,v) to
 - texture image (t,s)

The diagram illustrates the texture mapping process in three stages:

- Texture Coordinate System:** A 2D grid with axes s and t . A red quadrilateral is highlighted within the grid.
- Modeling Coordinate System:** A 3D cube with axes u , v , and w . The red quadrilateral is mapped onto the $u-v$ face of the cube. Vertices are labeled $(0,0)$, $(1,0)$, $(0,1)$, and $(1,1)$.
- Image Coordinate System:** A 2D grid with axes x and y . The red quadrilateral is mapped onto the grid.

14

Texture Mapping

- Texture mapping is a 2D projective transformation
 - texture coordinate system: (t,s) to
 - image coordinate system (x,y)

15

Texture Mapping

- Scan conversion
 - Interpolate texture coordinates down/across scan lines
 - Distortion due to bilinear interpolation approximation
 - » Cut polygons into smaller ones, or
 - » Perspective divide at each pixel

The diagram shows a triangle on a grid. The vertices are labeled with texture coordinates: (u_1, v_1) , (u_2, v_2) , and (u_3, v_3) . A scan line is drawn across the triangle, and the intersection points are labeled α and β .

16

Texture Mapping

Two checkerboard patterns are shown to compare interpolation methods:

- Linear interpolation of texture coordinates:** The checkerboard pattern is distorted, appearing skewed and stretched.
- Correct interpolation with perspective divide:** The checkerboard pattern is correctly aligned and undistorted.

Hill Figure 8.42

17

Overview

- Texture mapping methods
 - Parameterization
 - Mapping
 - Filtering
- Texture mapping applications
 - Modulation textures
 - Illumination mapping
 - Bump mapping
 - Environment mapping
 - Image-based rendering
 - Non-photorealistic rendering

18

Texture Filtering

- Must sample texture to determine color at each pixel in image

19

Texture Filtering

- Aliasing is a problem

Point sampling
Area filtering

20

Texture Filtering

- Ideally, use elliptically shaped convolution filters

In practice, use rectangles

21

Texture Filtering

- Size of filter depends on projective warp
 - Can prefiltering images
 - Mip maps
 - Summed area tables

Magnification
Minification

22

Mip Maps

- Keep textures prefiltered at multiple resolutions
 - For each pixel, linearly interpolate between two closest levels (e.g., trilinear filtering)
 - Fast, easy for hardware

23

Summed-area tables

- Keep at each texel the sum of all values down & right
 - To compute sum of all values within a rectangle, simply subtract two entries
 - Better ability to capture very oblique projections
 - But, cannot store values in a single byte

24

Overview

- Texture mapping methods
 - Parameterization
 - Mapping
 - Filtering
- Texture mapping applications
 - Modulation textures
 - Illumination mapping
 - Bump mapping
 - Environment mapping
 - Image-based rendering
 - Non-photorealistic rendering

25

Modulation textures

Map texture values to scale factor

Wood texture

Texture value

26

Illumination Mapping

Map texture values to surface material parameter

- K_A
- K_D
- K_S
- K_T
- n

$K_T = T(s,t)$

27

Bump Mapping

Map texture values to perturbations of surface normals

28

Bump Mapping

H&B Figure 14.100

29

Environment Mapping

Map texture values to perturbations of surface normals

H&B Figure 14.93

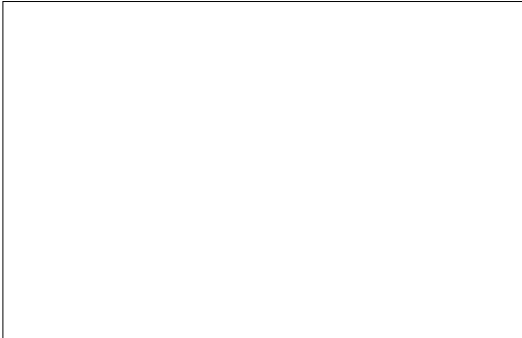
30

Image-Based Rendering

Map photographic textures to provide details for coarsely detailed polygonal model

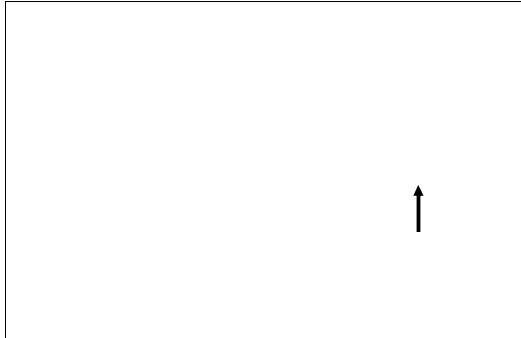
21

Nonphotorealistic Rendering



22

Art-Maps



23

Summary

- Texture mapping methods
 - Parameterization
 - Mapping
 - Filtering
- Texture mapping applications
 - Modulation textures
 - Illumination mapping
 - Bump mapping
 - Environment mapping
 - Image-based rendering
 - Non-photorealistic rendering