Transforms and Texture Mapping 3

CS 184: Foundations of Computer Graphics

1 Basic Transforms

1. Isometric transformations preserve the distances between every pair of points on an object. Which transformations are isometric? Which transformations aren't?

2. Using homogeneous coordinates, define a point, $\mathbf{p} = \begin{pmatrix} 2 \\ 5 \\ 1 \end{pmatrix}$, a vector, $\mathbf{v} = \begin{pmatrix} 2 \\ 5 \\ 0 \end{pmatrix}$, and a translation transformation, $\mathbf{T} = \begin{pmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{pmatrix}$. What is the result of applying $\mathbf{T}$ to $\mathbf{p}$?

3. What is the result of applying $\mathbf{T}$ to $\mathbf{v}$? Justify this result.

4. Write the transformation matrix for a 2D object that is reflected across the y-axis, translated up by 1 unit and right by 3 units, then rotated around the origin by $90°$ counterclockwise.

2 Rebecca Takes a Look

In world space coordinates, a teapot sits at $(0, 0, 0)$. Rebecca holds her camera at position $(1, 0, 2)$ — the eye point, **e**, of the camera.

1. Rebecca points her camera at the teapot, intending to take a picture. What is the view direction, **v**, of her camera?

2. In world space coordinates, in what direction does the positive z-axis of Rebecca's "standard" camera space point?

3. The up vector, **u**, points in the direction of the positive y-axis of "standard" camera space. In world space coordinates, $\mathbf{u} = \begin{pmatrix} 0 & 1 & 0 \end{pmatrix}^\intercal$. Calculate the right vector, **r**, which points in the direction of the positive x-axis of "standard" camera space.

4. Write the matrix that transforms coordinates from "standard" camera space to world space. The inverse of this matrix is the *Look-At* matrix. What coordinate system transformation does the *Look-At* matrix perform?

3 Barycentric Coordinates

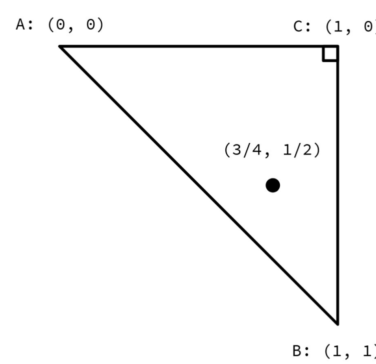
The first step in mapping a texture onto a triangle is to convert the screen pixel to barycentric coordinates. These coordinates (α, β, γ) can be thought of as the weights assigned to each vertex. The weighted average of the vertices is a screen-space coordinate: $(x, y) = \alpha A + \beta B + \gamma C$.

1. What happens if α, β, or γ is less than zero?

2. What is the range of (x, y) if $\alpha + \beta + \gamma = 1$ and $\alpha = 0$?

3. What is the range of (x, y) if $\alpha + \beta + \gamma = 1$ and $\alpha < 0$?

4. What are the barycentric coordinates of the point corresponding to the screen-space coordinates $(x, y) = (\frac{3}{4}, \frac{1}{2})$ in the following diagram? (Hint: Use the definition of barycentric coordinates, not the closed-form solution.)



5. If the RGB values of A, B, C are `(1, 0, 0)`, `(0, 1, 0)`, `(0, 0, 1)`, respectively, then what is the interpolated color at the selected point above?

4 A Texture and Mipmap Adventure

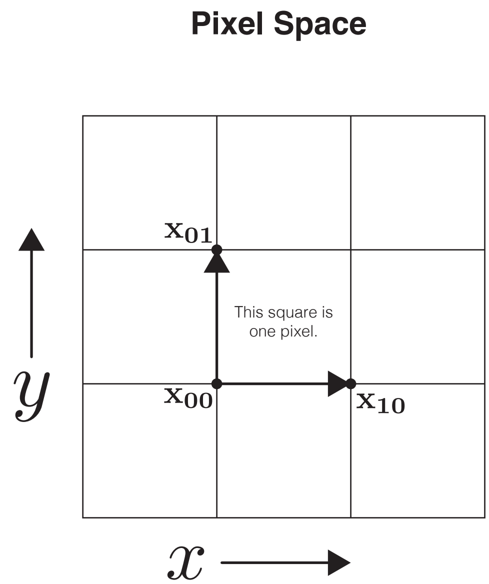
In this problem, we explore texture coordinates and mipmaps. A pixel may cover more or less than one texel, leading to aliasing. To handle this, we use mipmaps—pre-filtered, downscaled versions of the texture.

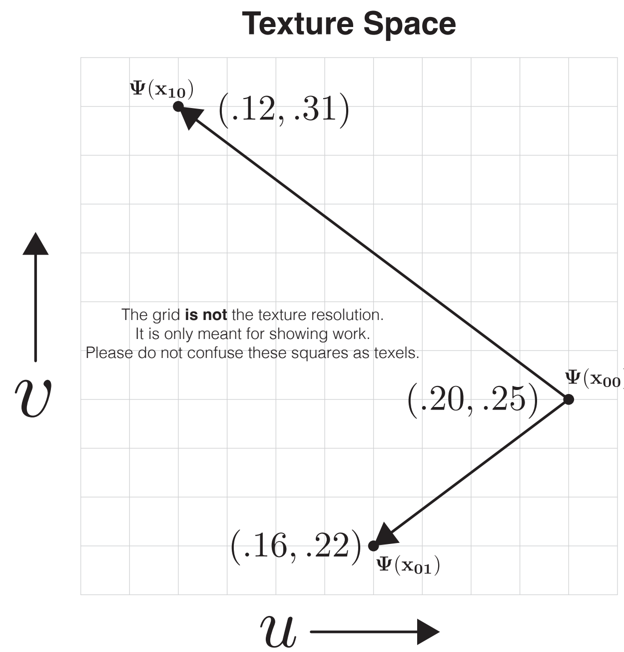
Consider an image mapped to a texture with a resolution of $(128, 128)$. We'll work through sampling a pixel and retrieving its value from the appropriate mipmap level. Note that the texture coordinates (u, v) lie in the range $[0, 1]$, not the texture resolution range.

1. **Conceptual Question:** What is the resolution of the mipmap at level 3?

2. **Conceptual Question:** Derive an expression for the amount of space needed to store all mipmap levels, relative to the memory needed to store the original texture.

3. We are given pixel point $\mathbf{x_{00}} = (x, y)$, as well as $\mathbf{x_{01}}$ and $\mathbf{x_{10}}$, each 1-pixel away from $\mathbf{x_{00}}$. These three points are mapped to their corresponding locations in texture space by the mapping $\mathbf{\Psi}(\mathbf{x})$. *Note that the mapping uses barycentric coordinates!* The texel space is in the range $[0, 1]$. At what mipmap level, L, should we sample to retrieve the texture for point $\mathbf{x_{00}}$?





4. For mipmap levels 0 through 5, the texture value at $\mathbf{\Psi}(x_{00})$ is given by:

$$T_0 = 0.38, \quad T_1 = 0.42, \quad T_2 = 0.36, \quad T_3 = 0.40, \quad T_4 = 0.39, \quad T_5 = 0.37.$$

   Given our answer to the previous question, what texture value should we use?

5. How can we combine values from two neighboring mipmap levels to get an even smoother result?