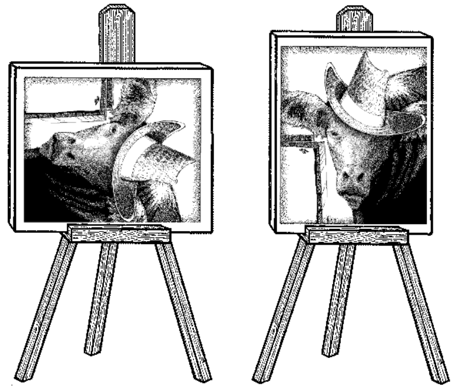
2D Image Transformations

- Geometric Transformations
-

Hearns & Baker

pp. 184-212 (Study)

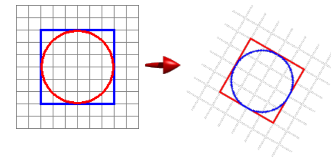


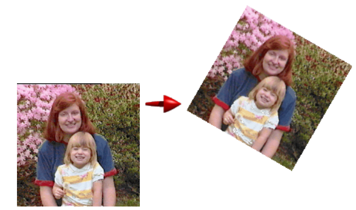
Two-Dimensional Geometric Transforms

Geometric transforms are functions that map points from one place to another

Geometric transforms can be applyed to

- drawing primitives (lines, conics, triangles)
- pixel coordinates of an image
-
-
- (or sprites)





We'll begin with simple transforms and generalize them.

Translations

Translations are a simple family of two-dimensional transforms. Translations were at the heart of our Sprite implementations in Project #1.

Translations have the following form

$$x' = x + t_x$$
$$y' = y + t_y$$

For every translation *there exists an inverse function* which undoes the translation. In our case the inverse looks like:

$$x = x' - t_x$$
$$y = y' - t_y$$

There also exists a special translation, called the *identity*, that leaves every point unchanged.

$$x' = x + 0$$
$$y' = y + 0$$

Groups and Composition

For Translations:

1. There exists an inverse mapping for each function
2. There exists an identity mapping

These properties might seem trivial at first glance, but they are actually very important, because when these conditions are shown for any class of functions it can be proven that such a class is closed under composition (i.e. any series of translations can be composed to a single translation). In mathematical parlance this the same as saying that translations form an **algebraic group**.

$$x' = \underbrace{T_1 T_2 T_3 \cdots T_n}_{T'} x$$

Rotations

Another group of 2-transforms are the rotations about the origin.

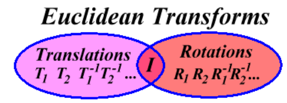
$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = R \begin{bmatrix} x \\ y \end{bmatrix}$$

$$R^{-1} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix}$$

$$R_{\theta=0} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Euclidean Transforms

The union of translations and rotation functions defines the Euclidean Set



Properties of Euclidean Transformations:

- They preserve distances
- They preserve angles

How do you represent these functions?

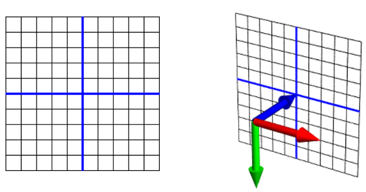
$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix}$$

Problems with this Form

- Must consider Translation and Rotation separately
- Computing the inverse transform involves multiple steps
- Order matters between the R and T parts

$$R(T(\bar{x})) \neq T(R(\bar{x}))$$

These problem can be remedied by considering our 2 dimensional image plane as a 2D subspace within 3D.



Choose a Subspace

We can use any planar subspace as long as it does not contain the origin

WLOG assume the our 2D space of points lies on the 3D plane $z = 1$

Now we can express all Euclidean Transforms in matrix form:

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & t_x \\ \sin\theta & \cos\theta & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

This gives us a three parameter group of Transformations.

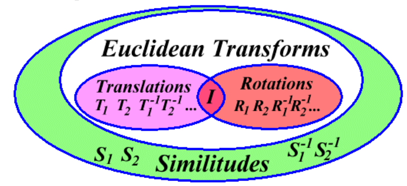
Playing with Euclidean Transforms

- In what order are the translation and rotation performed?
- Will this family of transforms always generate points on our choosen 3-D plane?
  Why?

Previous Lecture 10 Slide 9 6.837 Fall '98 Next

Similitude Transforms

We can define a 4-parameter superset of Euclidean Transforms with additional capabilities



Properties of Similtudes:

- Distance between any 2 points are changed by a fixed ratio
- Angles are preserved
- Maintains "similar" shape (similar triangles, circles map to circles, etc)

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} \sigma\cos\theta & -\sigma\sin\theta & t_x \\ \pm\sigma\sin\theta & \pm\sigma\cos\theta & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Previous Lecture 10 Slide 10 6.837 Fall '98 Next

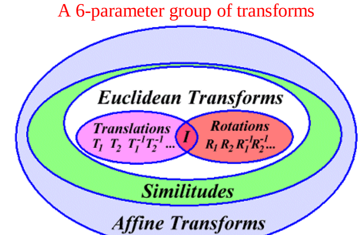
Playing with Similitude Transforms

- Adds reflections
- Scales in x and y must be the same. Why?
- Order?
- Will this family of transforms always generate points on our choosen 3-D plane?
  Why?

Previous Lecture 10 Slide 11 6.837 Fall '98 Next

Affine Transformations

A 6-parameter group of transforms

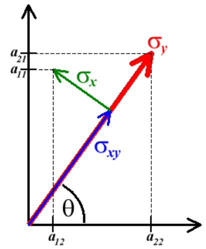


$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} \sigma_{xy}\sin\theta + \sigma_x\cos\theta & \sigma_{xy}\cos\theta - \sigma_x\sin\theta & t_x \\ \sigma_y\sin\theta & \sigma_y\cos\theta & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Previous Lecture 10 Slide 12 6.837 Fall '98 Next

Affine Properties

To the right is a simple illustration of how we can map our parameters into a four arbitrary numbers



Properties of Affine Transforms:

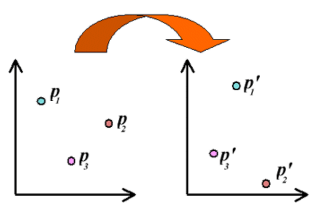
- They perserve our selected plane (sometimes called the *Affine plane*)
- They preserve parallel lines

Playing with Affine Transforms

- σ_x scales the x-dimension
- σ_y scales the y-dimension
- σ_{xy} is often called the *skew* parameter

Determining Affine Transforms

The coordinates of three corresponding points uniquely determine and Affine Transform



If we know where we would like at least three points to map to, we can solve for an Affine transform that will give this mapping.

Solution Method

We've used this technique several times now. We set up 6 linear equations in terms of our 6 unknown values. In this case, we know the coordinates before and after the mapping, and we wish to solve for the entries in our Affine transform matrix.

$$\underbrace{\begin{bmatrix} x_1' \\ y_1' \\ x_2' \\ y_2' \\ x_3' \\ y_3' \end{bmatrix}}_{\mathbf{x}'} = \underbrace{\begin{bmatrix} x_1 & y_1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x_1 & y_1 & 1 \\ x_2 & y_2 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x_2 & y_2 & 1 \\ x_3 & y_3 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x_3 & y_3 & 1 \end{bmatrix}}_{\mathbf{X}} \underbrace{\begin{bmatrix} a_{11} \\ a_{12} \\ a_{13} \\ a_{21} \\ a_{22} \\ a_{23} \end{bmatrix}}_{\mathbf{a}}$$

This gives the following solution:

$$\mathbf{X}^{-1}\mathbf{x}' = \mathbf{a}$$

>

Exercise #3

Part #1

Compute a 3 by 3 transfomation matrix that composes a translation by t_x, t_y and a rotation by θ such that the translation is first and the rotation second. Is the resulting transformation in the Euclidean Group?

Part #2

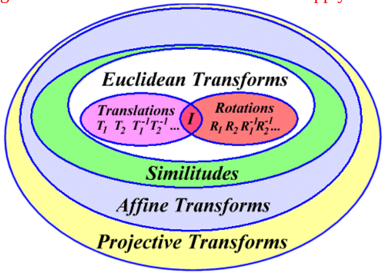
Compute the affine transformation that maps the vertices of a triangle v_1, v_2, and v_3 to the canonical triangle with vertices (1,0), (0,1) and (0,0) respectively. Show how this transform can be used to map any triangle to any other triangle.

Limit your work to a single sheet of 8.5" by 11" paper.

Previous Lecture 10 Slide 17 6.837 Fall '98 Next

Projective Transformations

The most general linear transformation that we can apply to 2-D points



There is something different about this group of transformations. The result will not necessarily lie on our selected plane. Since our world (to this point) is 2D we need some way to deal with this.

$$\begin{bmatrix} wx' \\ wy' \\ w \end{bmatrix} = \begin{bmatrix} p_{11} & p_{12} & p_{13} \\ p_{21} & p_{22} & p_{23} \\ p_{31} & p_{32} & p_{33} \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Previous Lecture 10 Slide 18 6.837 Fall '98 Next

Projection

The mapping of points from an N-D space to a M-D subspace ($M < N$)

We need a rule for mapping points resulting of this transform back onto our plane z = 1.

We will identify points by lines through the origin of the 3-D space that we have embedded our plane within.

Thus, $x' \equiv \alpha x'$

Since the origin lies on all of these lines (and thus cannot be uniquely specified) we will disallow it. This is no big loss since it wasn't on our selected plane anyway (This is the real reason that we chose a plane not containing the origin).

If we want to find the coordinate of any point in our selected plane we need only scale it such that it's third coordinate, w, is 1.

Previous Lecture 10 Slide 19 6.837 Fall '98 Next

Projective Transforms

Since all of the resulting points are defined to within a non-zero scale factor. We can also scale the transformation by an arbitrary and it will still give the same result.

$$\alpha \begin{bmatrix} p_{11} & p_{12} & p_{13} \\ p_{21} & p_{22} & p_{23} \\ p_{31} & p_{32} & p_{33} \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \equiv \begin{bmatrix} p_{11} & p_{12} & p_{13} \\ p_{21} & p_{22} & p_{23} \\ p_{31} & p_{32} & p_{33} \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

We might as well choose α so that one of the parameters of our matrix is 1 (i.e. $p_{33} = 1$).

Previous Lecture 10 Slide 20 6.837 Fall '98 Next

Degrees of Freedom

A projective transform has 8 free-parameters

$$\begin{bmatrix} wx' \\ wy' \\ w \end{bmatrix} = \begin{bmatrix} p_{11} & p_{12} & p_{13} \\ p_{21} & p_{22} & p_{23} \\ p_{31} & p_{32} & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

which can be expressed as the following rational linear equation:

$$x' = \frac{p_{11}x + p_{12}y + p_{13}}{p_{31}x + p_{32}y + 1} \qquad y' = \frac{p_{21}x + p_{22}y + p_{23}}{p_{31}x + p_{32}y + 1}$$

rearranging terms gives a linear expression in the coefficients:

$$x' = p_{11}x + p_{12}y + p_{13} - p_{31}xx' - p_{32}yx'$$

$$y' = p_{21}x + p_{22}y + p_{23} - p_{31}xy' - p_{32}yy'$$

Specifying a projective transform

A projective transform can be uniquely defined by how it maps 4 points

$$\begin{bmatrix} x'_1 \\ y'_1 \\ x'_2 \\ y'_2 \\ x'_3 \\ y'_3 \\ x'_4 \\ y'_4 \end{bmatrix} = \begin{bmatrix} x_1 & y_1 & 1 & 0 & 0 & 0 & -x'_1x_1 & -x'_1y_1 \\ 0 & 0 & 0 & x_1 & y_1 & 1 & -y'_1x_1 & -y'_1y_1 \\ x_2 & y_2 & 1 & 0 & 0 & 0 & -x'_2x_2 & -x'_2y_2 \\ 0 & 0 & 0 & x_2 & y_2 & 1 & -y'_2x_2 & -y'_2y_2 \\ x_3 & y_3 & 1 & 0 & 0 & 0 & -x'_3x_3 & -x'_3y_3 \\ 0 & 0 & 0 & x_3 & y_3 & 1 & -y'_3x_3 & -y'_3y_3 \\ x_4 & y_4 & 1 & 0 & 0 & 0 & -x'_4x_4 & -x'_4y_4 \\ 0 & 0 & 0 & x_4 & y_4 & 1 & -y'_4x_4 & -y'_4y_4 \end{bmatrix} \begin{bmatrix} p_{11} \\ p_{12} \\ p_{13} \\ p_{21} \\ p_{22} \\ p_{23} \\ p_{31} \\ p_{32} \end{bmatrix}$$

Projective Example

With the applet on the right you can select any corner and drag it to where you would like.

Things are starting to look 3-D... We'll get there soon.

Next Time