Announcements

- First tutorial is going to be today **11am-12pm** in **BV462**
  - C++ and OpenGL as preparation for programming portion of assignment 1 (continuation next week)

- Blackboard/Portal should now be available

- Lecture Notes and Slides for last week are on the course webpage:

  http://www.cs.toronto.edu/~ls/teaching.html

- Assignment 1 will go out on Wednesday

Last week's review

- What is computer graphics
- Definition of raster displays
- Line drawing
  - Simple line drawing
  - Efficient line drawing using Bresenham's algorithm
- Polygon filling
  - with active edge lists
- Clipping
  - Very briefly, we'll rev-visit it later

Parametric Curves and Polygons

Computer Graphics, CSCD18

Fall 2008

Instructor: Leonid Sigal

2D Geometric Objects

- 2D objects are simpler, so we'll start with these before jumping to 3D
  - Most of the concepts we will be talking about in 2D can (and will) be extended to 3D
- We will consider 2 key object classes
  - Parametric curves (we already started looking at lines which is a special case)
  - Polygons

Parametric Curves in 2D

- **Goal:** model geometry using mathematic representation

- There are many ways of representing curves in 2D
  - **Explicit**

    $$y = f(x)$$ **given x, find y**

  - **Implicit**

    $$f(x, y) = 0$$ **(we already saw this for Bresenham's alg)**

  - **Parametric**

    $$\overline{\mathbf{p}}(\lambda) = \left(\mathbf{x}(\lambda), \mathbf{y}(\lambda)\right)$$ **Function from R -> R^2**

Explicit Equation of a Line

- Explicit equation form: $\mathbf{y} = \mathbf{f}(\mathbf{x})$
- Explicit equation for the line: $\mathbf{y} = \mathbf{mx} + \mathbf{b}$

Problem: this representation doesn't work for vertical lines. Why?

Implicit Equation of a Line

- Implicit equation form: $\mathbf{f}(\mathbf{x},\mathbf{y}) = 0$

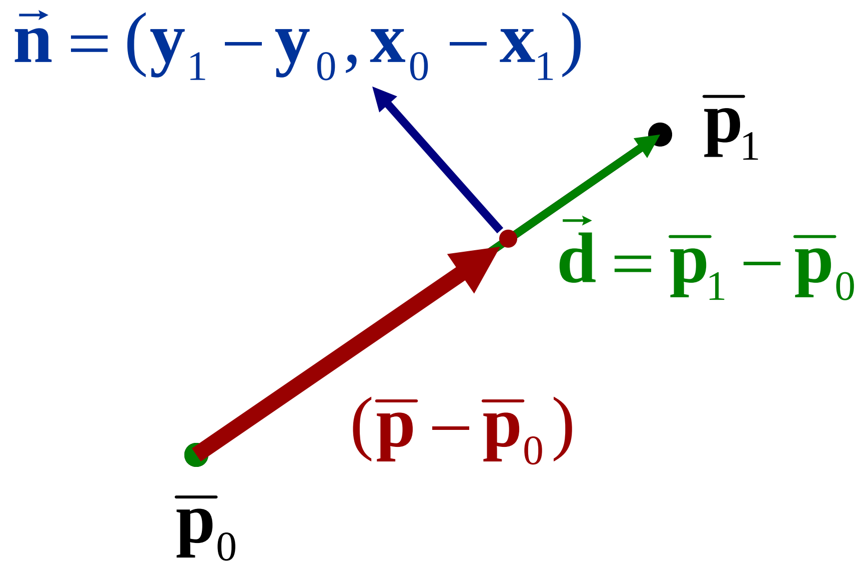
  $\mathbf{f}(\bar{\mathbf{x}}) = 0$ **vector form**

- Implicit equation for the line from $\bar{p}_0=(x_0,y_0)$ to $\bar{p}_1=(x_1,y_1)$:

$$(\mathbf{x}-\mathbf{x}_0)(\mathbf{y}_1-\mathbf{y}_0)-(\mathbf{y}-\mathbf{y}_0)(\mathbf{x}_1-\mathbf{x}_0)=0$$

- **Intuition**

Implicit Equation of a Line

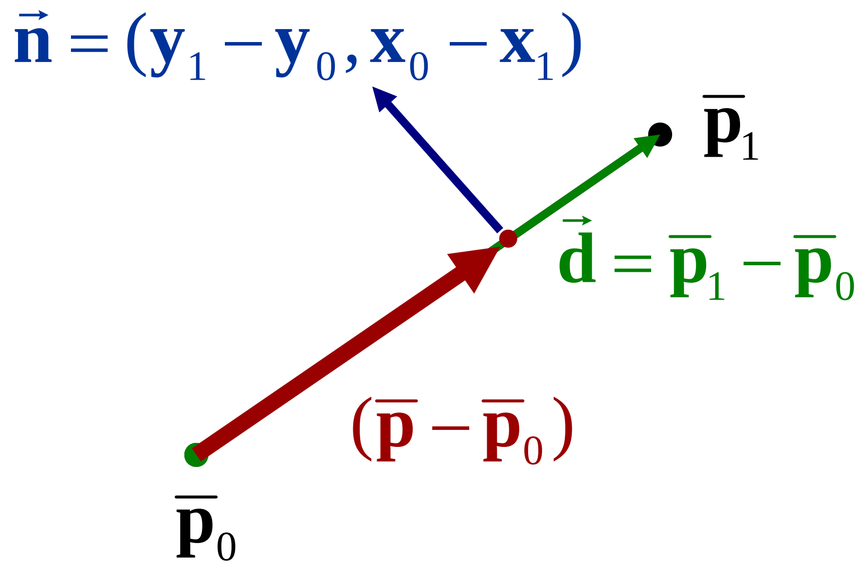


- **Intuition**
  - Direction of the line is vector: $\vec{\mathbf{d}} = \overline{\mathbf{p}}_1 - \overline{\mathbf{p}}_0$
  - So any vector for from the starting point to any point on the line $(\overline{\mathbf{p}} - \overline{\mathbf{p}}_0)$ must be parallel to

$$(\mathbf{x} - \mathbf{x}_0)(\mathbf{y}_1 - \mathbf{y}_0) - (\mathbf{y} - \mathbf{y}_0)(\mathbf{x}_1 - \mathbf{x}_0) = 0$$

  - Alternatively, $(\overline{\mathbf{p}} - \overline{\mathbf{p}}_0)$ must be perpendicular to the normal $\vec{\mathbf{n}} = (\mathbf{y}_1 - \mathbf{y}_0, \mathbf{x}_0 - \mathbf{x}_1)$
  - We can check this by taking a dot product of the normal with the $(\overline{\mathbf{p}} - \overline{\mathbf{p}}_0)$, thereby deriving the implicit equation for the line

Implicit Equation of a Line



$$(\mathbf{x} - \mathbf{x}_0)(\mathbf{y}_1 - \mathbf{y}_0) - (\mathbf{y} - \mathbf{y}_0)(\mathbf{x}_1 - \mathbf{x}_0) = 0$$

Parametric Equation of a Line

- Parametric equation form: $\overline{\mathbf{p}}(\lambda) = \left(\mathbf{x}(\lambda), \mathbf{y}(\lambda)\right)$

- Parametric equation for the line through $\overline{\mathrm{p}}_0$ and $\overline{\mathrm{p}}_1$:

$$\overline{\mathbf{p}}(\lambda) = \overline{\mathbf{p}}_0 + \lambda \vec{\mathbf{d}}$$

  where $\vec{\mathbf{d}} = \overline{\mathbf{p}}_1 - \overline{\mathbf{p}}_0$

- For parametric equation bounds for λ must be specified
  - Line segment from $\overline{\mathrm{p}}_0$ to $\overline{\mathrm{p}}_1$: $0 \le \lambda \le 1$
  - Ray from $\overline{\mathrm{p}}_0$ in the direction of $\overline{\mathrm{p}}_1$: $0 \le \lambda < \infty$
  - Line passing though $\overline{\mathrm{p}}_0$ and $\overline{\mathrm{p}}_1$: $-\infty < \lambda < \infty$

Explicit Equation of a Circle

- Explicit equation form: $\mathbf{y} = \mathbf{f}(\mathbf{x})$
- Explicit equation for the line:

Does no exist !!!

y is multi-function of x (for every x there are 2 y's)

Implicit Equation of a Circle

- Implicit equation form: $\mathbf{f}(\mathbf{x},\mathbf{y}) = 0$

  $\mathbf{f}(\bar{\mathbf{x}}) = 0$ **vector form**

- Implicit equation for the circle:

$$(\mathbf{x}-\mathbf{x}_c)^2 + (\mathbf{y}-\mathbf{y}_c)^2 - \mathbf{r}^2 = 0$$

or in vector form

$$\left\|\bar{\mathbf{p}}-\bar{\mathbf{p}}_c\right\|^2 - \mathbf{r}^2 = 0$$

Parametric Equation of a Circle

- Parametric equation form: $\overline{\mathbf{p}}(\lambda) = \left(\mathbf{x}(\lambda), \mathbf{y}(\lambda)\right)$

- Parametric equation for circle:

$$\overline{\mathbf{p}}(\lambda) = \left(\mathbf{r}\cos(2\pi\lambda), \mathbf{r}\sin(2\pi\lambda)\right)$$

Note: this is the polar coordinate representation of the circle.

Note: There are an infinite number of parametric representations for most curves

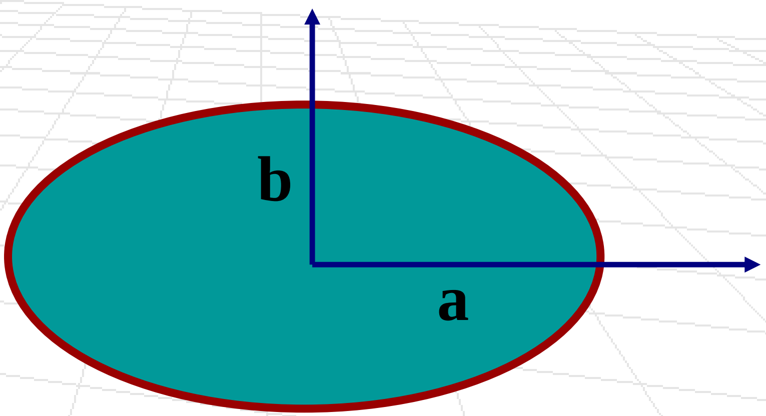
Ellipses

- Special case where ellipse is centered at origin (0,0) and major axis is aligned with y-axis
- Implicit form:

$$\frac{\mathbf{x}^2}{\mathbf{a}^2} + \frac{\mathbf{y}^2}{\mathbf{b}^2} - 1 = 0$$

- Parametric form:

$$\mathbf{x}(\lambda) = \mathbf{a}\cos(2\pi\lambda)$$
$$\mathbf{y}(\lambda) = \mathbf{b}\sin(2\pi\lambda)$$



Superellipses

- Derived from ellipse to allow a "squarness" parameter η

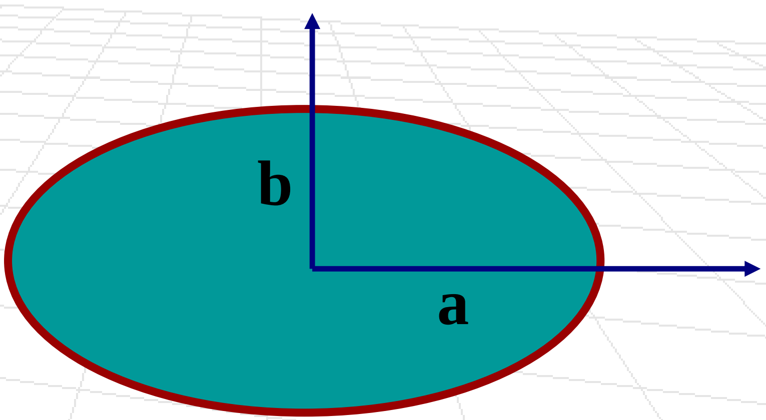
- Parametric form:

$$\mathbf{x}(\lambda) = \mathbf{a}\ \mathbf{sign}\big(\cos(2\pi\lambda)\big)\left\|\cos(2\pi\lambda)\right\|^{2/\eta}$$

$$\mathbf{y}(\lambda) = \mathbf{b}\ \mathbf{sign}\big(\sin(2\pi\lambda)\big)\left\|\sin(2\pi\lambda)\right\|^{2/\eta}$$

where

$$\mathbf{sign}(\mathbf{x}) = \begin{cases} 1 & \mathbf{if}\ \mathbf{x} > 0 \\ -1 & \mathbf{if}\ \mathbf{x} < 0 \\ 0 & \mathbf{if}\ \mathbf{x} = 0 \end{cases}$$

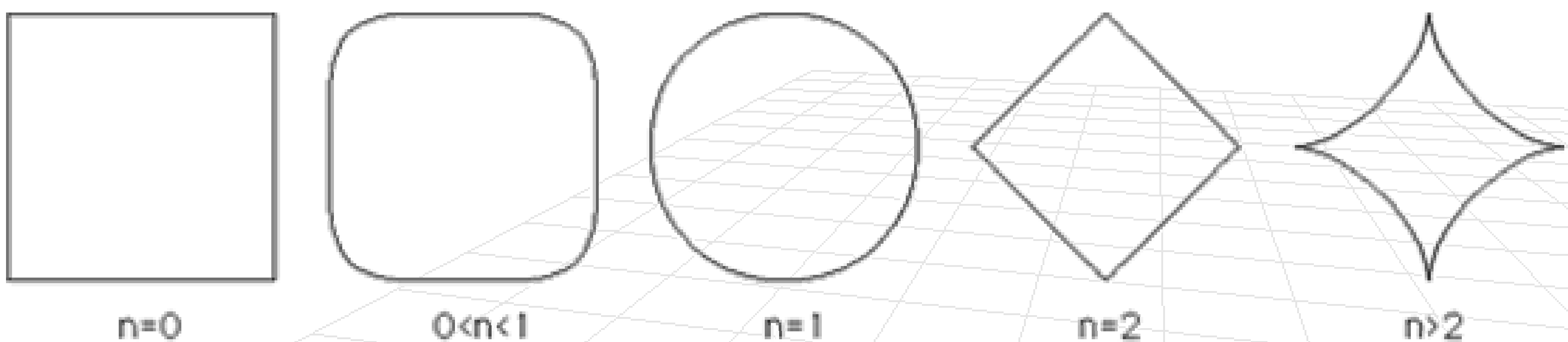


Superellipses

- Derived from ellipse to allow a “squarness” parameter η
- Parametric form:

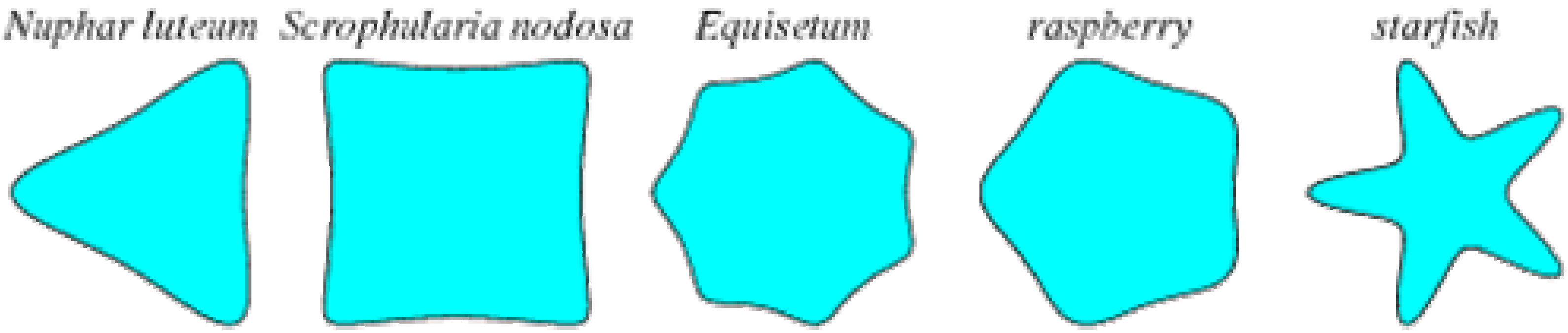
$$\mathbf{x}(\lambda) = \mathbf{a}\ \mathbf{sign}\left(\cos(2\pi\lambda)\right)\left\|\cos(2\pi\lambda)\right\|^{2/\eta}$$

$$\mathbf{y}(\lambda) = \mathbf{b}\ \mathbf{sign}\left(\sin(2\pi\lambda)\right)\left\|\sin(2\pi\lambda)\right\|^{2/\eta}$$



Images from Paul Bourke on-line reference
[http://local.wasp.uwa.edu.au/~pbourke/surfaces_curves/superellipse/]

Further extensions …



From Wolfram MathWorld

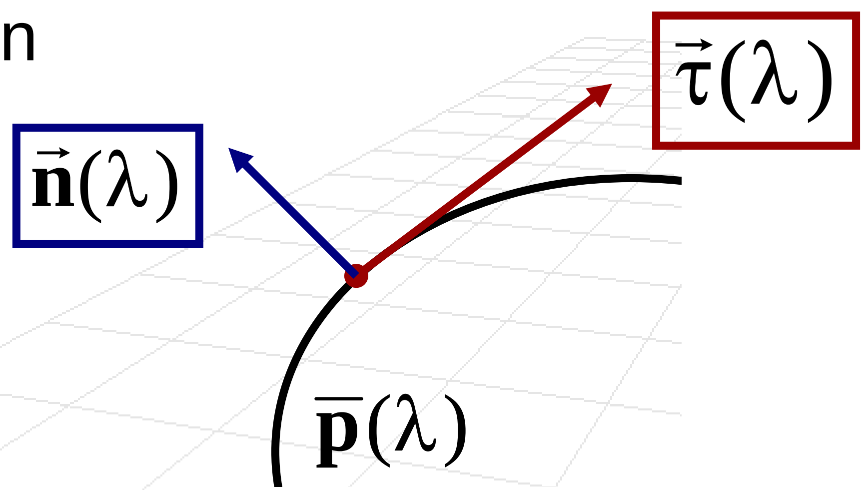
[http://mathworld.wolfram.com/Superellipse.html]

Tangents and Normals

- **Tangent** to a curve at a point is the instantaneous direction of that curve
  - Line containing the tangent intersects curve at the point
  - Is given by the

$$\vec{\tau}(\lambda) = \frac{\mathbf{d}\overline{\mathbf{p}}(\lambda)}{\mathbf{d}\lambda} = \left(\frac{\mathbf{dx}(\lambda)}{\mathbf{d}\lambda}, \frac{\mathbf{dy}(\lambda)}{\mathbf{d}\lambda} \right)$$

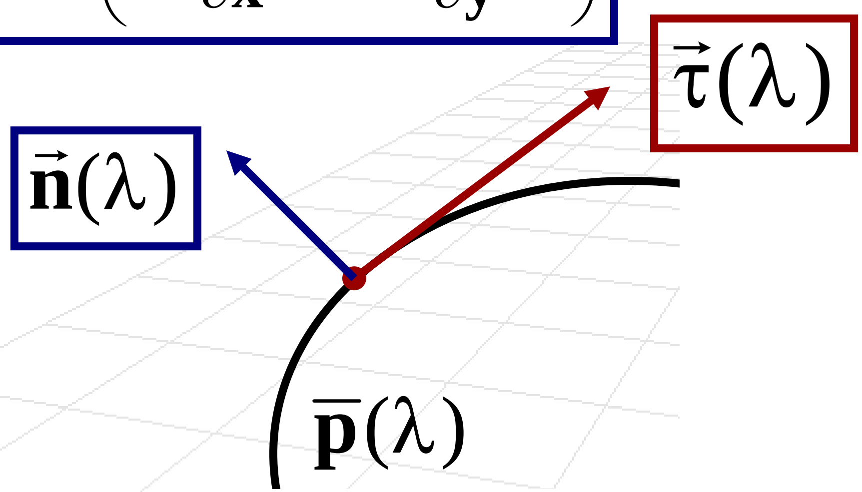
- **Normal** to a curve is a perpendicular to the tangent direction



Tangents and Normals

- **Tangent** to a curve at a point is the instantaneous direction of that curve
- **Normal** to a curve is a perpendicular to the tangent direction
  - Can be derived from the implicit form for point $\overline{p}$ on curve $f(x,y)$

$$\vec{\mathbf{n}}(\lambda) = \nabla \mathbf{f}(\overline{\mathbf{p}})\,|_{\overline{\mathbf{p}}} = \left(\frac{\partial \mathbf{f}(\mathbf{x},\mathbf{y})}{\partial \mathbf{x}}, \frac{\partial \mathbf{f}(\mathbf{x},\mathbf{y})}{\partial \mathbf{y}}\right)$$



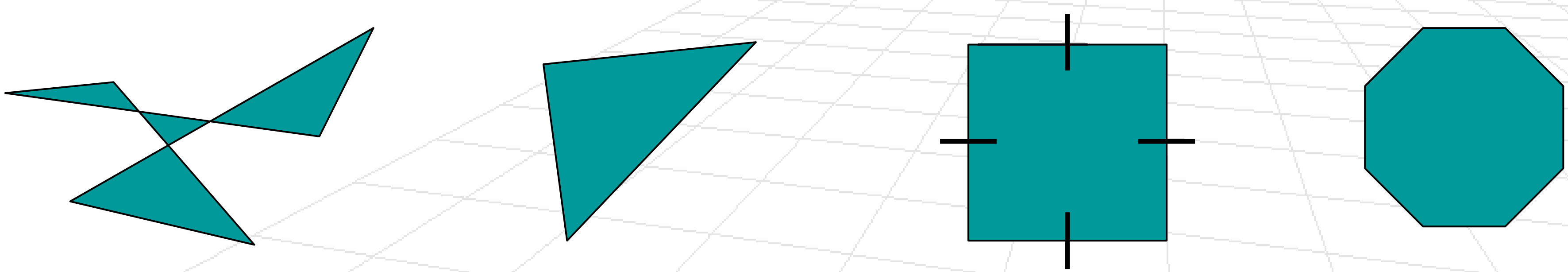
Proof

On board and in Lecture Notes

Polygons

- **Polygon** is a continuous, piecewise linear, closed planar curve
  - **Simple polygon** is non self-intersecting
  - **Regular polygon** is simple, equilateral and equiangular
  - **N-gon** regular polygon with N sides

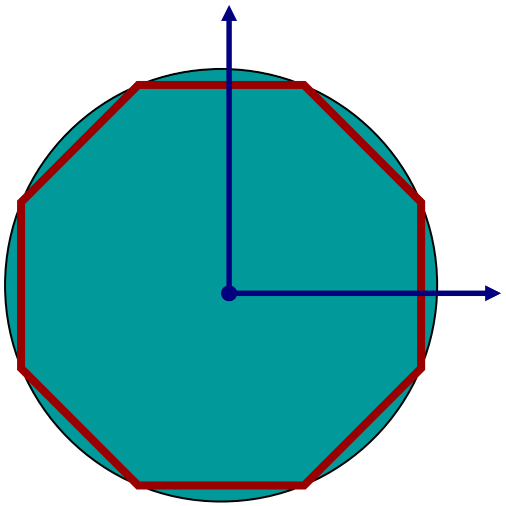
- Where do we get them
  - Approximations to simple parametric curves
  - Make them up



Polygon **Simple Polygon** **Regular Polygon** **N(8)-gon**

Example: Building N-gon from a Circle

- **Idea:** to find vertices of an N-gram, find N equidistant points on a circle



- In polar coordinates

$$(\mathbf{x_i}, \mathbf{y_i}) = \mathbf{r}(\cos\theta_\mathbf{i}, \sin\theta_\mathbf{i})$$

where

$$\theta_\mathbf{i} = \mathbf{i}\frac{2\pi}{\mathbf{N}} \qquad 0 \le \mathbf{i} \le \mathbf{N} - 1$$

- What if we don't want N-gon with the center at **(0,0)**
  - To translate – add $(\mathbf{x_c}, \mathbf{y_c})$ to each point
  - To scale – change r
  - To rotate – add (delta) to each θ_i

2D Transformations

Computer Graphics, CSCD18

Fall 2008

Instructor: Leonid Sigal

Use of 2D Transformations

- **Goal:** given point cloud, polygon, or sampled parametric curve
  - Change coordinate frames
  - Compose objects made of simple parts
    - By defining position/scale/orientation of parts with respect to other parts (hierarchical models)
  - Deform the shape to create new more interesting shapes
  - Useful for animation

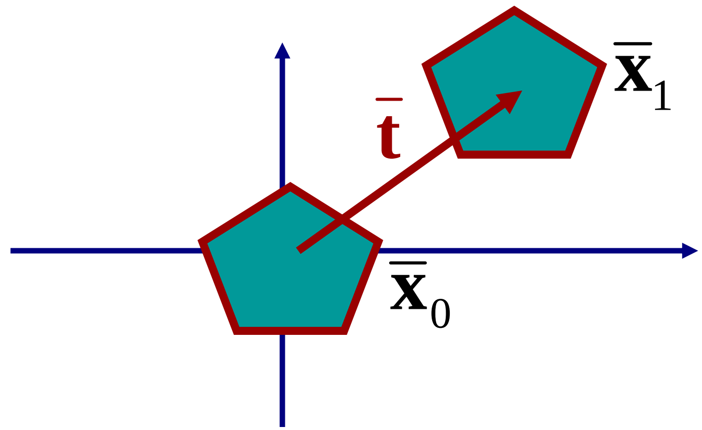
Transformation Types

- **Rigid transformations**
  - **Examples:** Translations, Rotations
  - **Properties:** preserve distance and angles
- **Conformal transformations**
  - **Examples:** translations, rotations, uniform scale
  - **Properties:** preserves angles (not distance)
- **Affine transformations**
  - **Examples:** translations, rotations, general scaling, reflections
  - **Properties:** preserves parallelism, preserves linearity (lines remain lines

Elementary Transformations

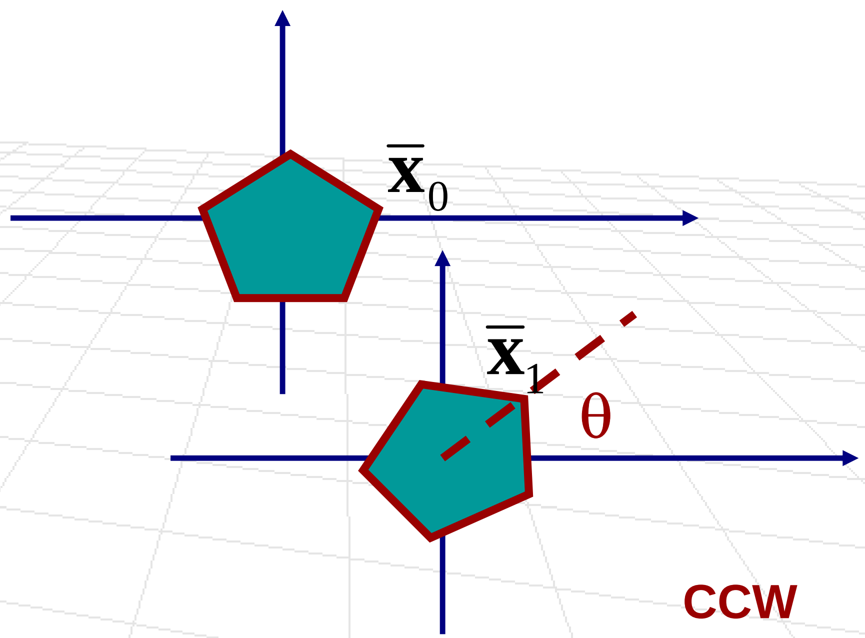
- Translation

$$\bar{\mathbf{x}}_1 = \bar{\mathbf{x}}_0 + \bar{\mathbf{t}}$$



- Rotation

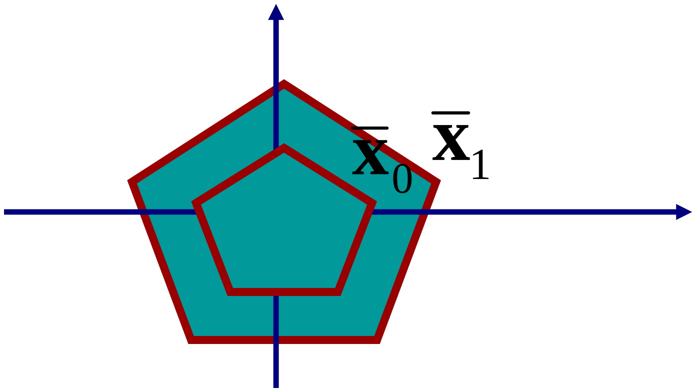
$$\bar{\mathbf{x}}_1 = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \bar{\mathbf{x}}_0$$



Elementary Transformations

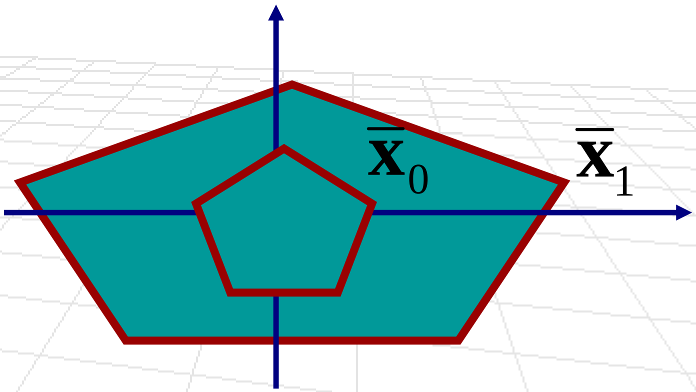
- Uniform Scaling

$$\bar{\mathbf{x}}_1 = \begin{pmatrix} \mathbf{a} & 0 \\ 0 & \mathbf{a} \end{pmatrix} \bar{\mathbf{x}}_0$$



- Non-uniform Scaling

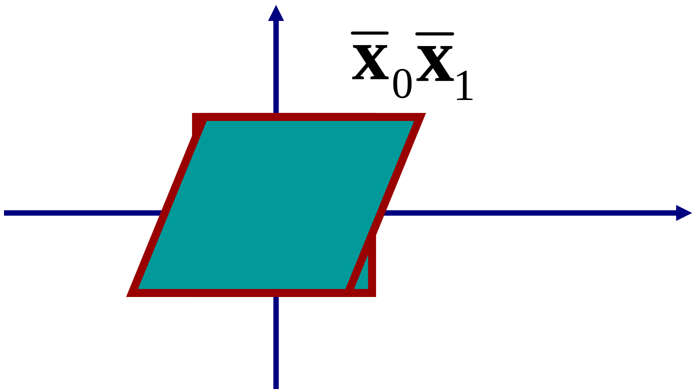
$$\bar{\mathbf{x}}_1 = \begin{pmatrix} \mathbf{a} & 0 \\ 0 & \mathbf{b} \end{pmatrix} \bar{\mathbf{x}}_0$$



Elementary Transformations

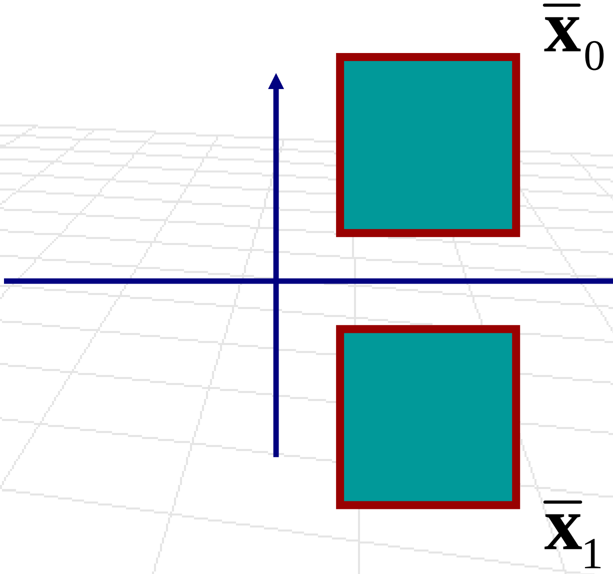
- Shear

$$\bar{\mathbf{x}}_1 = \begin{pmatrix} 1 & \mathbf{h} \\ 0 & 1 \end{pmatrix} \bar{\mathbf{x}}_0$$



- Reflection

$$\bar{\mathbf{x}}_1 = \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix} \bar{\mathbf{x}}_0$$



Affine Transformation

$$\bar{\mathbf{x}}_1 = \mathbf{A}\,\bar{\mathbf{x}}_0 + \bar{\mathbf{t}}$$

- Any linear transformation $\mathbf{A}$ (can be rotation, scaling, reflection, etc.) followed by a translation $\mathbf{t}$
- Thereby translation, rotation, scaling, sheer are all special cases of affine transformation

- **Properties**
  - inverse of affine transformation is also affine
  - lines are preserved
  - given closed region (polygon) area under the affine transformation is scaled by $\det(\mathbf{A})$
  - compositions of affine transformations is still affine transformation

Announcements

- **Assignment 1 is out**

- Writing portion
  - 4 question
  - When we ask for “prove” something, we mean proof in a mathematical sense
  - Electronic submissions are preferred

- Programming
  - Start early
  - Starter code available for Linux and VC++

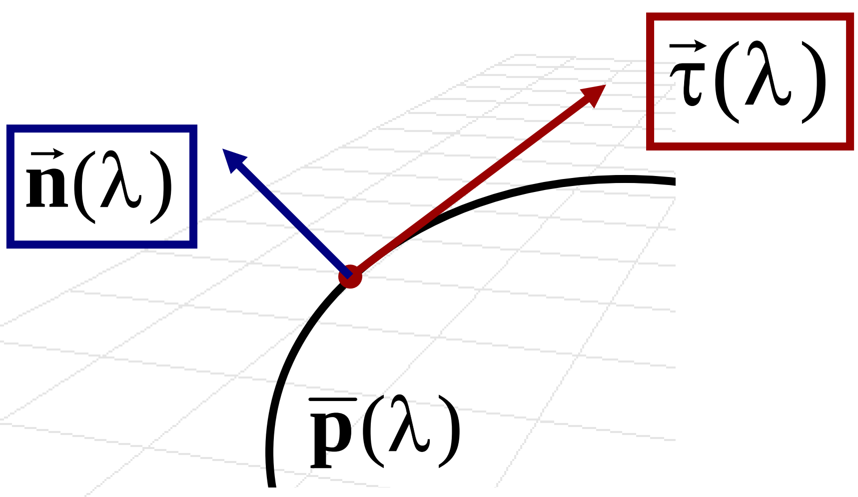
Last class review

	Line	Circle	Ellipse
Explicit:	$\mathbf{y} = \mathbf{mx} + \mathbf{b}$	**N/A**	**N/A**
Implicit:	$(\overline{\mathbf{x}} - \overline{\mathbf{x}}_0)\vec{\mathbf{n}} = 0$	$\lVert \overline{\mathbf{p}} - \overline{\mathbf{p}}_\mathbf{c} \rVert^2 - \mathbf{r}^2 = 0$	$\frac{\mathbf{x}^2}{\mathbf{a}^2} + \frac{\mathbf{y}^2}{\mathbf{b}^2} - 1 = 0$
Parametric:	$\overline{\mathbf{p}}(\lambda) = \overline{\mathbf{p}}_0 + \lambda\vec{\mathbf{d}}$	$\overline{\mathbf{p}}(\lambda) = \begin{bmatrix} \mathbf{r}\cos(2\pi\lambda) \\ \mathbf{r}\sin(2\pi\lambda) \end{bmatrix}$	$\overline{\mathbf{p}}(\lambda) = \begin{bmatrix} \mathbf{a}\cos(2\pi\lambda) \\ \mathbf{b}\sin(2\pi\lambda) \end{bmatrix}$

Tangents and Normals

- **Tangent from parametric form:** $$\vec{\tau}(\lambda) = \left(\frac{dx(\lambda)}{d\lambda}, \frac{dy(\lambda)}{d\lambda} \right)$$
  derivative

- **Normal from implicit form:** $$\vec{n}(\lambda) = \left(\frac{\partial f(x,y)}{\partial x}, \frac{\partial f(x,y)}{\partial y} \right)$$
  gradient



Transformations

- **Rigid transformations**
  - **Examples:** Translations, Rotations
  - **Properties:** preserve distance and angles
- **Conformal transformations**
  - **Examples:** translations, rotations, uniform scale
  - **Properties:** preserves angles (not distance)
- **Affine transformations**
  - **Examples:** translations, rotations, general scaling, reflections
  - **Properties:** preserves parallelism, preserves linearity (lines remain lines

Affine Transformation

$$\overline{\mathbf{q}} = \mathbf{A}\,\overline{\mathbf{p}} + \overline{\mathbf{t}}$$

- Any linear transformation $\mathbf{A}$ (can be rotation, scaling, reflection, etc.) followed by a translation $\mathbf{t}$
- Thereby translation, rotation, scaling, sheer are all special cases of affine transformation

- **Properties**
  - inverse of affine transformation is also affine
  - lines are preserved
  - given closed region (polygon) area under the affine transformation is scaled by $\det(\mathbf{A})$
  - compositions of affine transformations is still affine transformation

Proof: Inverse of Affine Transformation is also an Affine Transformation

$$\overline{\mathbf{q}} = \mathbf{A}\overline{\mathbf{p}} + \overline{\mathbf{t}}$$

$$\overline{\mathbf{q}} - \overline{\mathbf{t}} = \mathbf{A}\overline{\mathbf{p}}$$

$$\mathbf{A}^{-1}(\overline{\mathbf{q}} - \overline{\mathbf{t}}) = \overline{\mathbf{p}}$$

assime $\mathbf{A}^{-1}$ exists

$$\mathbf{A}^{-1}\overline{\mathbf{q}} - \mathbf{A}^{-1}\overline{\mathbf{t}} = \overline{\mathbf{p}}$$

$$\overline{\mathbf{p}} = \mathbf{B}\overline{\mathbf{q}} + \overline{\mathbf{v}}$$

where

$$\mathbf{B} = \mathbf{A}^{-1} \qquad \overline{\mathbf{v}} = \mathbf{A}^{-1}\overline{\mathbf{t}}$$

Proof: compositions of affine transformations is still affine transformation

$$\mathbf{F}_1(\overline{\mathbf{p}}) = \mathbf{A}_1\overline{\mathbf{p}} + \overline{\mathbf{t}}_1$$

$$\mathbf{F}_2(\overline{\mathbf{p}}) = \mathbf{A}_2\overline{\mathbf{p}} + \overline{\mathbf{t}}_2$$

$$\mathbf{F}_2\left(\mathbf{F}_1(\overline{\mathbf{p}})\right) = \mathbf{A}_2\left(\mathbf{A}_1\overline{\mathbf{p}} + \overline{\mathbf{t}}_1\right) + \overline{\mathbf{t}}_2$$

$$= \mathbf{A}_2\mathbf{A}_1\overline{\mathbf{p}} + \mathbf{A}_2\overline{\mathbf{t}}_1 + \overline{\mathbf{t}}_2$$

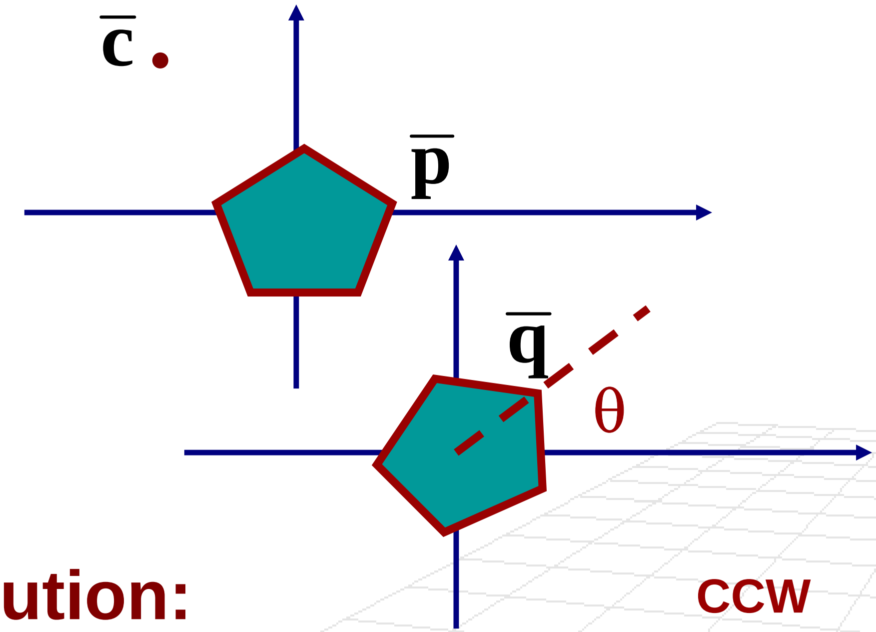
$$= \mathbf{A}\overline{\mathbf{p}} + \overline{\mathbf{t}}$$

where

$$\mathbf{A} = \mathbf{A}_2\mathbf{A}_1 \qquad \overline{\mathbf{t}} = \mathbf{A}_2\overline{\mathbf{t}}_1 + \overline{\mathbf{t}}_2$$

Why composing transformations useful?

- Rotations as we have seen It in the last class rotate the object about the origin in CCW, what if we want to rotate about some other point $\mathbf{c}$?



$$F_1: \quad \mathbf{A}_1 = \mathbf{I} \quad \bar{\mathbf{t}}_1 = -\bar{\mathbf{c}}$$
$$F_2: \quad \mathbf{A}_2 = \mathbf{R}(\theta) \quad \bar{\mathbf{t}}_2 = 0$$
$$F_3: \quad \mathbf{A}_3 = \mathbf{I} \quad \bar{\mathbf{t}}_3 = \bar{\mathbf{c}}$$

$$\bar{\mathbf{q}} = \mathbf{F}_3\left(\mathbf{F}_2\left(\mathbf{F}_1(\bar{\mathbf{p}})\right)\right)$$

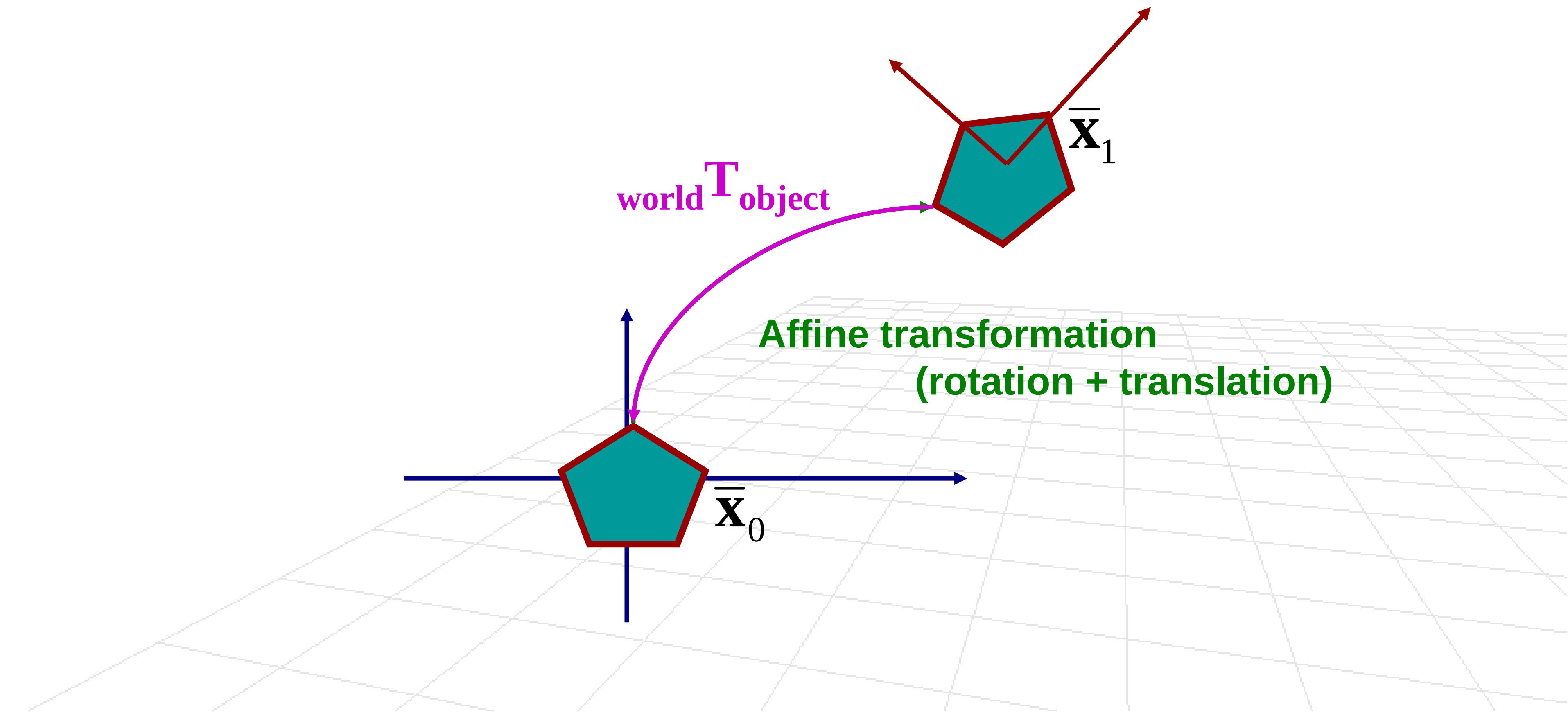
- **Solution:**
  - Translate by $-\bar{\mathbf{c}}$ (so that $\bar{\mathbf{c}}$ is the new origin)
  - Rotate
  - Translate back by $\bar{\mathbf{c}}$

Additional Affine Transformation Properties Proofs

In the Lecture Notes

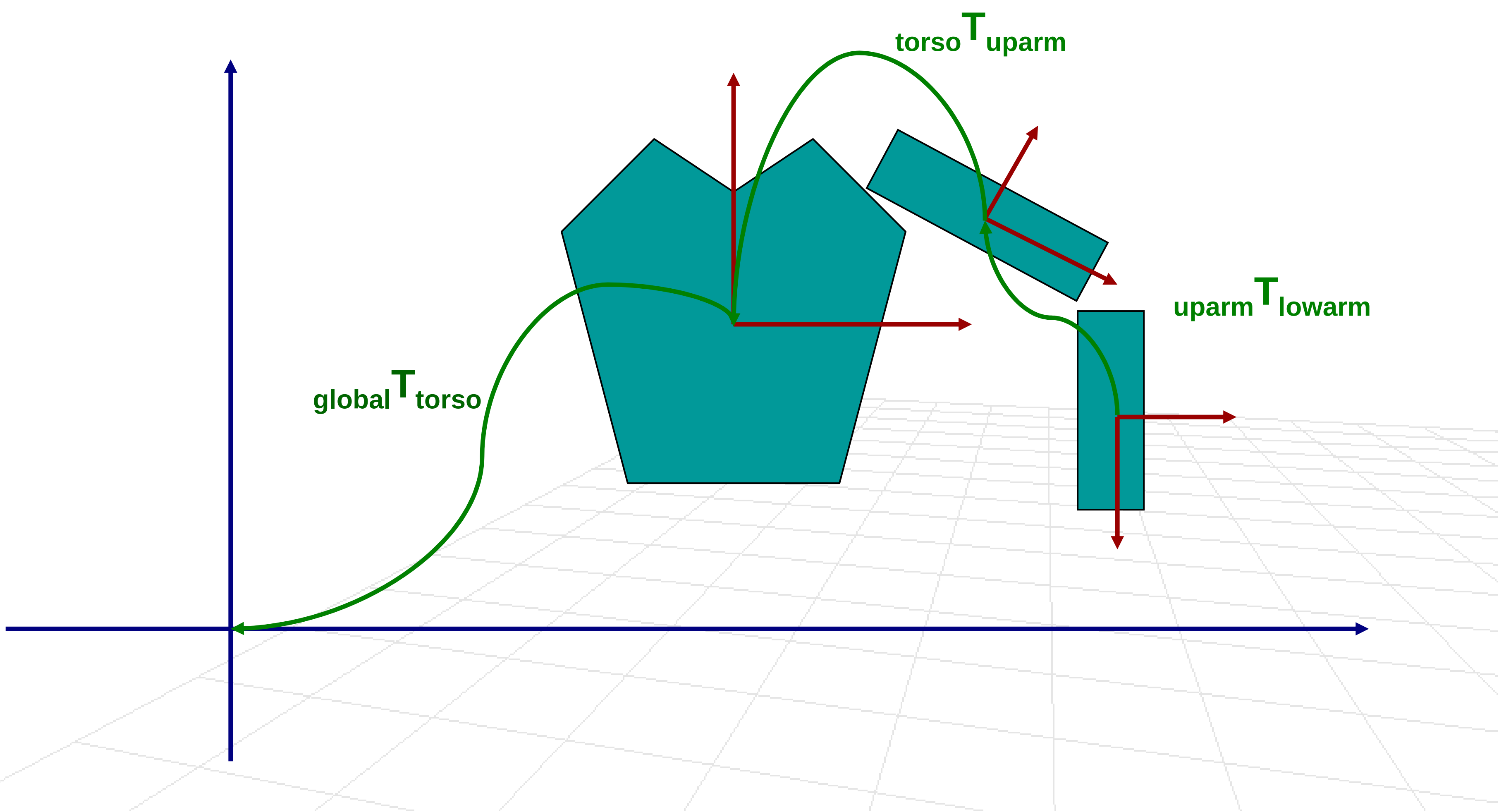
Changing Coordinate Frames

Can be interpreted as the transformation from object coordinate frame (red) to world coordinate frame (blue)



Hierarchical Models

$$\bar{p}_{global} = {}_{global}T_{torso} \times {}_{torso}T_{uparm} \times {}_{uparm}T_{lowarm} \times \bar{p}_{lowarm}$$



Homogeneous Coordinates

Computer Graphics, CSCD18

Fall 2008

Instructor: Leonid Sigal

Homogeneous Coordinates

- **Problem:** affine transformations often become complex and unwieldy to keep track of
- **Homogeneous coordinates** allow all the transformations to be specified by a single matrix multiply (OpenGL)
- How do we express a Cartesian point in homogeneous coordinates?

$$\begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} \longrightarrow \alpha \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha\, \mathbf{x} \\ \alpha\, \mathbf{y} \\ \alpha \end{bmatrix} \qquad \alpha \neq 0$$

Cartesian point **Homogeneous point**

Homogeneous Coordinates

Example:

$$\begin{bmatrix} 1 \\ 2 \end{bmatrix} \rightarrow \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix}, \quad \begin{bmatrix} 1 \\ 2 \end{bmatrix} \rightarrow \begin{bmatrix} 5 \\ 10 \\ 5 \end{bmatrix}, \quad \begin{bmatrix} 1 \\ 2 \end{bmatrix} \rightarrow \begin{bmatrix} 2 \\ 4 \\ 2 \end{bmatrix}$$

$$\begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} \longrightarrow \alpha \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha\,\mathbf{x} \\ \alpha\,\mathbf{y} \\ \alpha \end{bmatrix} \qquad \alpha \neq 0$$

Cartesian point → **Homogeneous point**

Converting from Homogeneous Coordinates

$$\begin{bmatrix} \mathbf{x} \\ \mathbf{y} \\ \alpha \end{bmatrix} = \begin{bmatrix} \mathbf{x}/\alpha \\ \mathbf{y}/\alpha \\ 1 \end{bmatrix} \qquad \alpha \neq 0 \qquad \longrightarrow \qquad \begin{bmatrix} \mathbf{x}/\alpha \\ \mathbf{y}/\alpha \end{bmatrix}$$

Homogeneous point **Cartesian point**

- **Note:** two homogeneous points are not equal if they are not scalar multiples of one another

Homogeneous Transformations

- Turns out that many transformations become linear in homogeneous coordinates (mainly affine)

Affine in Cartesian Coordinates

$$\bar{\mathbf{q}} = \mathbf{A}\bar{\mathbf{p}} + \bar{\mathbf{t}}$$

$$\begin{bmatrix} \mathbf{q}_x \\ \mathbf{q}_y \end{bmatrix} = \begin{bmatrix} \mathbf{a} & \mathbf{b} \\ \mathbf{c} & \mathbf{d} \end{bmatrix} \begin{bmatrix} \mathbf{p}_x \\ \mathbf{p}_y \end{bmatrix} + \begin{bmatrix} \mathbf{t}_x \\ \mathbf{t}_y \end{bmatrix}$$

Affine in Homogeneous Coordinates

$$\bar{\mathbf{q}} = \begin{bmatrix} \mathbf{A} & \bar{\mathbf{t}} \end{bmatrix} \hat{\mathbf{p}}$$

$$\begin{bmatrix} \mathbf{q}_x \\ \mathbf{q}_y \end{bmatrix} = \begin{bmatrix} \mathbf{a} & \mathbf{b} & \mathbf{t}_x \\ \mathbf{c} & \mathbf{d} & \mathbf{t}_y \end{bmatrix} \begin{bmatrix} \mathbf{p}_x \\ \mathbf{p}_y \\ 1 \end{bmatrix}$$

- But it's easier to always keep track of homogeneous representation, so

$$\hat{\mathbf{q}} = \begin{bmatrix} \mathbf{A} & \bar{\mathbf{t}} \\ \begin{bmatrix} 0 & 0 \end{bmatrix} & 1 \end{bmatrix} \hat{\mathbf{p}}$$

This is linear and easy to keep track of

Properties of Affine Transformation (cont.)

- With homogeneous representation for affine transformation, several additional properties of affine transformations become apparent
  - affine transformations are **associative**

$$(\mathbf{F}_3\ \mathbf{F}_2)\mathbf{F}_1 = \mathbf{F}_3\ (\mathbf{F}_2\ \mathbf{F}_1)$$

  - Affine transformations are **not** in general **commutative**

(proof of this is a homework question)

$$\mathbf{F}_2\ \mathbf{F}_1 \neq \mathbf{F}_1\ \mathbf{F}_2$$

Vectors in Homogeneous Coordinates

$$\hat{\vec{\mathbf{v}}} = \begin{bmatrix} \mathbf{v}_x \\ \mathbf{v}_y \\ 0 \end{bmatrix}$$

Homogeneous vector
(third component 0!)

Example:

$$\begin{bmatrix} \mathbf{v}_x \\ \mathbf{v}_y \\ 0 \end{bmatrix} + \begin{bmatrix} \mathbf{p}_x \\ \mathbf{p}_y \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{p}_x + \mathbf{v}_x \\ \mathbf{p}_y + \mathbf{v}_y \\ 1 \end{bmatrix}$$

Homogeneous vector **Homogeneous point** **Homogeneous point**

What else can we do with Homogeneous Coordinates?

- The equation of the line

$$\mathbf{y} = \mathbf{mx} + \mathbf{d}$$

$$0 = \mathbf{ax} + \mathbf{by} + \mathbf{c}$$

$$\mathbf{a} = -\mathbf{bm}$$
$$\mathbf{c} = -\mathbf{bd}$$

- In homogeneous coordinates

$$\begin{bmatrix} \mathbf{a} & \mathbf{b} & \mathbf{c} \end{bmatrix} \begin{bmatrix} \mathbf{p_x} \\ \mathbf{p_y} \\ 1 \end{bmatrix} = \bar{\mathbf{l}}^{\mathrm{T}} \hat{\mathbf{p}} = 0$$

Vector holding line parameters

Vector holding homogeneous coordinate of a point

Finding Line Passing Through 2 Points

- Equation of the line in homogeneous coordinates:

$$\begin{bmatrix}\mathbf{a} & \mathbf{b} & \mathbf{c}\end{bmatrix}\begin{bmatrix}\mathbf{p_x} \\ \mathbf{p_y} \\ 1\end{bmatrix} = \bar{\mathbf{l}}^{\mathbf{T}}\,\hat{\mathbf{p}} = 0$$

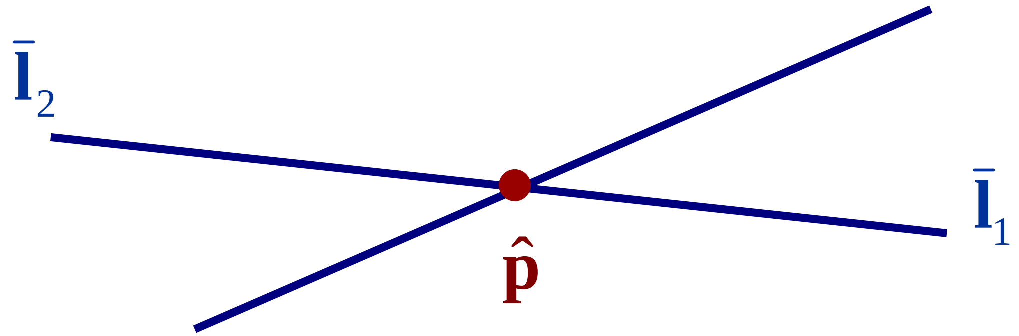
- If two homogeneous points $\hat{\mathbf{p}}_1$ and $\hat{\mathbf{p}}_2$ are on the line then

$$\bar{\mathbf{l}}^{\mathbf{T}}\,\hat{\mathbf{p}}_1 = 0 \qquad \bar{\mathbf{l}}^{\mathbf{T}}\,\hat{\mathbf{p}}_2 = 0$$

(vector $\bar{\mathbf{l}}$ must perpendicular to two 3D vectors)

$$\boxed{\bar{\mathbf{l}} = \hat{\mathbf{p}}_1 \times \hat{\mathbf{p}}_2}$$

Finding Intersection of Two Lines



- If two homogeneous points $\mathbf{p_0}$ and $\mathbf{p_1}$ are on the line then

$$\bar{\mathbf{l}}_1^{\mathbf{T}}\,\hat{\mathbf{p}} = 0 \qquad\qquad \bar{\mathbf{l}}_2^{\mathbf{T}}\,\hat{\mathbf{p}} = 0$$

(point $\hat{\mathbf{p}}$ must perpendicular to two 3D vectors holding the line parameters)

$$\boxed{\hat{\mathbf{p}} = \bar{\mathbf{l}}_1 \times \bar{\mathbf{l}}_2}$$