CZ2003
Computer Graphics and Visualization

Revision of
Modules 1, 2, 3

AY 2020/2021, Sem. 1

Module 1: Introduction and Foundation Mathematics

- Definition of computer graphics
- Visualization steps (objects, materials, viewpoints, light sources)
- Various coordinate systems → coordinate mapping between systems (Cartesian 2D and 3D (right/left handed), 2D polar, 3D spherical, 3D cylindrical). Right/Left-handed as well as order of coordinates in other systems is important.
- Definition and types of mathematical functions: explicit, implicit parametric functions. How to convert between them.
- Pythagoras' theorem and consequences: angles and distances
- Matrices, vectors and operations on them
- Definition of coordinates by vector operations (sum of scaled vectors)
- Trigonometric functions: any harmonic oscillations (geometry, colors, motions)
- Dot product: angle between vectors
- Cross product: normal to the plane surface

Module 1: Introduction and Foundation Mathematics

- Conversion from polar, cylindrical and spherical coordinates to parametric functions in Cartesian coordinates is based on the right triangle rules:

  $x = r \cos\alpha \qquad y = r \sin\alpha$
- Inverse conversion from parametric functions in Cartesian coordinates to polar coordinates, etc. is still based on the Pythagoras triangle that

  $r^2 = x^2 + y^2 \quad \tan\alpha = y/x$
- Note that polar coordinate α is computed about the origin of the coordinate system.

Module 2: Programming Computer Graphics and Visualization

- Classification of the software tools into imperative and declarative, pixel-based and polygon-based.

Module 3: Geometric Shapes 1/8

- Should be able
  - to define curves parametrically
  - to define surfaces and solids parametrically and implicitly
- Define straight line, ray, and segment by:
  - coordinates of any 2 points on it
  - coordinates of intercept points on the coordinate axes
  - one point and a vector along the line
- Define plane by:
  - coordinates of 3 points on it
  - coordinates of intercept points on the coordinate axes
  - one point and a normal vector to the plane
  - two vectors/lines (intersecting or parallel)
- Remember that coefficients A,B,C of the plane equation A*x*+B*y*+C*z*+D=0 define coordinates of the normal vector to the plane

Module 3: Geometric Shapes 2/8

- In parametric representation:
  - curves – 1 parameter,
  - surfaces – 2 parameters,
  - 3D solids – 3 parameters
- Plane is defined parametrically based on sum of two vectors
- Bilinear surfaces are used for defining 4-sided patches of surfaces (not the whole plane).
- Triangle can be defined as a bilinear patch by putting together two of the vertices in the 4-sided bilinear patch.
- The concept of sweeping is used for defining curves as moving points, surfaces as moving curves and solids as moving surfaces.
- When defining rotational sweeping of plane objects:
  - Do not change the definition of the coordinate corresponding to the axis of rotation,
  - The other coordinate definition is used as a radius of a circle defined in the right-hand sense: 1st axis: *r**cos(), 2nd axis: *r**sin()

Module 3: Geometric Shapes 3/8

- Fast forward:

General rotational and translational sweeping can be defined using matrix transformations for rotation and translation (module 5):

[x,y,z formulas]=[Transformation matrix] [x,y,z formulas].

$$\begin{aligned} x' &= ax + by + cz + l \\ y' &= dx + ey + fz + m \\ z' &= gx + hy + kz + n \end{aligned}$$

$$\mathbf{P}' = \begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} ax + by + cz + l \\ dx + ey + fz + m \\ gx + hy + kz + n \\ 1 \end{bmatrix} = \begin{bmatrix} a & b & c & l \\ d & e & f & m \\ g & h & k & n \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Module 3: Geometric Shapes 4/8

- Example: Scaling by S_x, S_y and S_z with reference to the point (l, m, n)

$$\begin{bmatrix} x'(u) \\ y'(u) \\ z'(u) \\ 1 \end{bmatrix} = \begin{bmatrix} s_x & 0 & 0 & l(1-s_x) \\ 0 & s_y & 0 & m(1-s_y) \\ 0 & 0 & s_z & n(1-s_z) \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x(u) \\ y(u) \\ z(u) \\ 1 \end{bmatrix}$$

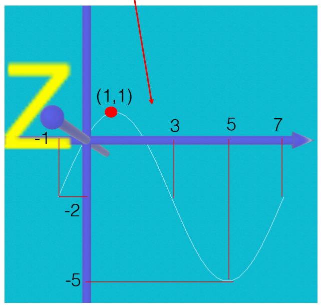
Scaling by 2, 3 and 1 with reference to the point (1, 1, 0)

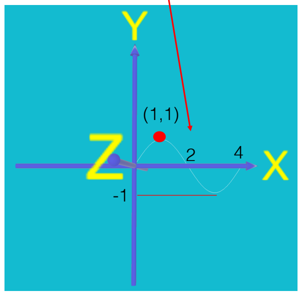
$$\begin{bmatrix} x'(u) \\ y'(u) \\ z'(u) \\ 1 \end{bmatrix} = \begin{bmatrix} 2 & 0 & 0 & 1(1-2) \\ 0 & 3 & 0 & 1(1-3) \\ 0 & 0 & 1 & 0(1-1) \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x(u) \\ y(u) \\ z(u) \\ 1 \end{bmatrix}$$

Module 3: Geometric Shapes 5/8

- Scaling by 2, 3 and 1 with reference to point (1, 1, 0)

$$\begin{bmatrix} 8u-1 \\ 3\sin(2\pi u)-2 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 2 & 0 & 0 & -1 \\ 0 & 3 & 0 & -2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 4u \\ \sin(2\pi u) \\ 0 \\ 1 \end{bmatrix}$$





Module 3: Geometric Shapes 4/8

- Example: Scaling by S_x, S_y and S_z with reference to the point (l, m, n)

$$\begin{bmatrix} x'(u) \\ y'(u) \\ z'(u) \\ 1 \end{bmatrix} = \begin{bmatrix} s_x & 0 & 0 & l(1-s_x) \\ 0 & s_y & 0 & m(1-s_y) \\ 0 & 0 & s_z & n(1-s_z) \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x(u) \\ y(u) \\ z(u) \\ 1 \end{bmatrix}$$

Animating the scaling by 2, 3 and 1 with reference to the point (1,1,0)

$$\begin{bmatrix} x'(u) \\ y'(u) \\ z'(u) \\ 1 \end{bmatrix} = \begin{bmatrix} 1+1t & 0 & 0 & 1(1-(1+1t)) \\ 0 & 1+2t & 0 & 1(1-(1+2t)) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x(u) \\ y(u) \\ z(u) \\ 1 \end{bmatrix}$$

Module 3: Geometric Shapes 5/8

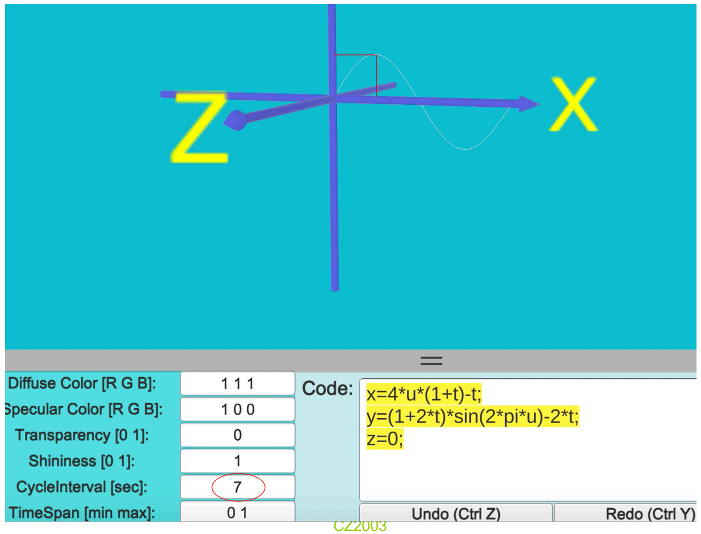
- Example: Scaling by S_x, S_y and S_z with reference to the point (l, m, n)

$$\begin{bmatrix} 4u(1+t)-t \\ (1+2t)\sin(2\pi u)-2t \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 1+t & 0 & 0 & -t \\ 0 & 1+2t & 0 & -2t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 4u \\ \sin(2\pi u) \\ 0 \\ 1 \end{bmatrix}$$

Animating the scaling by 2, 3 and 1 with reference to point (1,1,0)

$$\begin{bmatrix} x'(u) \\ y'(u) \\ z'(u) \\ 1 \end{bmatrix} = \begin{bmatrix} 1+1t & 0 & 0 & 1(1-(1+1t)) \\ 0 & 1+2t & 0 & 1(1-(1+2t)) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x(u) \\ y(u) \\ z(u) \\ 1 \end{bmatrix}$$

Module 3: Geometric Shapes 6/8



Module 3: Geometric Shapes 7/8

- Boolean (Set-Theoretic) operations for solid objects can be defined by inequalities $f(x, y, z) \geq 0$.
  Only in this case,
  $\min(f_1, f_2) \geq 0$ defines intersection,
  $\max(f_1, f_2) \geq 0$ - union,
  $-f \geq 0$ - complement
  $\min(f_1, -f_2) \geq 0$ - difference operations.
- Be able to find an intersection point of a straight line (ray) with a surface (Modules 7-8):
  - Define the straight line parametrically, while the surface – implicitly
  - Substitute parametric definitions $x(u)$ $y(u)$ $z(u)$ into implicit formula for the surface $f(x, y, z) = 0$ and solve it in terms of parameter u.
  - Among several u that can be produced, select the lowest value.
  - Obtain x, y, z by substituting the derived u.
- Blobby shapes add new opportunities and illustrate how implicit definitions cab be further enriched

Module 3: Geometric Shapes 8/8

- How to draw curves and surfaces: formulas have to be sampled to calculate coordinates of the polyline and polygon vertices on curves and surfaces. Sampling is controlled by bounding boxes (domains of the coordinates) and resolutions for implicit functions and by parametric domains and resolutions for parametric functions.
  ★ Formula,
  ★ Domain (bounding box or parameters range),
  ★ Sampling (resolution vales)

End of Part 1

CZ2003
Computer Graphics & Visualization
Revision of Part 2

AY 2020/2021 Semester 1

Final Assignment

- Date/time: 19 Nov 2020 (Thu, week 14) / 16:00 – 17:00
- Detail:
  - see Announcement in course site
    - The Final Assignment can be done on your own computers in the same way as you worked on your labs. SW Lab 3 will be available.
    - Please note that if you miss the Final Assignment you will not be able to retake it.
  - See course site ➔Assignments ➔ Final Assignment
    - get familiar with detailed instruction in advance

Suggestions

- Review each module and pay attention to
  - important concepts
  - basic methods
  - important formulae
- Review & practice on
  - examples given in both TEL & review lectures
  - Tutorial questions
  - Lab assignments (1~5)

The following slides list some fundamental concepts and methods for the 2nd part of the course.

1. 2D transformations

- Homogeneous coordinates

  (x, y) ⇔ (x, y, 1) = (k x, k y, k)

- 2D affine transformations

  x' = a x + b y + m
  y' = c x + d y + n   or

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} a & b & m \\ c & d & n \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

Basic set: translation, scaling, rotation

$$Tr(m,n) = \begin{pmatrix} 1 & 0 & m \\ 0 & 1 & n \\ 0 & 0 & 1 \end{pmatrix} \quad S(sx,sy) = \begin{pmatrix} sx & 0 & 0 \\ 0 & sy & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad Rot(\theta) = \begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

2D transformations

Others: *reflection, …*

$$\mathrm{Ref}_o = \begin{pmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$\mathrm{Ref}_{ox} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad \mathrm{Ref}_{oy} = \begin{pmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

2. 3D transformations

Affine transformation

$$x' = ax + by + cz + l$$
$$y' = dx + ey + fz + m$$
$$z' = gx + hy + iz + n$$

or

$$\begin{pmatrix} x' \\ y' \\ z' \\ 1 \end{pmatrix} = \begin{pmatrix} a & b & c & l \\ d & e & f & m \\ g & h & i & n \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$$

Translation

$$\begin{pmatrix} 1 & 0 & 0 & m \\ 0 & 1 & 0 & n \\ 0 & 0 & 1 & l \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Scaling

$$\begin{pmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Rotation

$$Rot_x = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta & 0 \\ 0 & \sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad Rot_y = \begin{pmatrix} \cos\theta & 0 & \sin\theta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\theta & 0 & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad Rot_z = \begin{pmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

3D transformations

Reflection

$$\mathrm{Ref}_o = \begin{pmatrix} -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$\mathrm{Ref}_{ox} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad \mathrm{Ref}_{oxy} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$\mathrm{Ref}_{oy} = \begin{pmatrix} -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad \mathrm{Ref}_{oyz} = \begin{pmatrix} -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$\mathrm{Ref}_{oz} = \begin{pmatrix} -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad \mathrm{Ref}_{ozx} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Affine transformations in VRML

- Transform {
    - translation dx dy dz
    - rotation ax ay az theta
    - scale sx sy sz
    - children […]

  }

  where
  - the rotation axis is from the origin (0,0,0) to point (ax,ay,az), and theta (in radian) is the rotation angle value;
  - sx, sy, sz are the 3 scaling factors along x, y, z axes;
  - dx, dy, dz are the translation amount along x, y, z axes.

  The order is first scale, then rotation, and finally translation.

2D and 3D transformation problems

- Given two shapes A and B where B is obtained from A by an affine transformation, find the transformation matrix that defines the affine transformation.
- Find an overall transformation that is a composite of several relatively simple affine transformations (such as scaling, rotation, reflection, …).
  - Key: how to convert a "non-standard" transformation to a "standard" transformation

2D and 3D transformation problems (cont)

- Matrix multiplication

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} \begin{bmatrix} b_{11} & b_{12} & b_{13} & b_{14} \\ b_{21} & b_{22} & b_{23} & b_{24} \\ b_{31} & b_{32} & b_{33} & b_{34} \\ b_{41} & b_{42} & b_{43} & b_{44} \end{bmatrix} = ?$$

- Matrix-vector multiplication

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = ?$$

2D and 3D transformation problems (cont)

- Applications in rotational/translational sweeping
  - Use matrix/matrices for rotation
  - Use matrix for translation

3. Motions and morphing

- Simulating three types of speed

Uniform; $\tau = \frac{t - t_1}{t_2 - t_1},\ t \in [t_1, t_2]$

Acceleration; $\tau = 1 - \cos\left(\frac{\pi}{2}\frac{t - t_1}{t_2 - t_1}\right),\ t \in [t_1, t_2]$

Deceleration $\tau = \sin\left(\frac{\pi}{2}\frac{t - t_1}{t_2 - t_1}\right),\ t \in [t_1, t_2]$

- Another way to describe animation
  - Use frame index k
- Methods for motions and morphing
  - Motion by path (three steps)
  - Linear interpolation of two items for morphing

    $v(\tau) = (1-\tau) A + \tau B, \quad 0 \le \tau \le 1$
  - Introducing time into rotational/translational sweeping

Typical problems

- Simulate uniform speed, acceleration, deceleration

- Express an animation using time parameter
- Express an animation using frame index

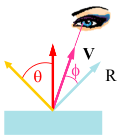
- Design an animation using “motion by path”
- Design an animation using linear interpolation
- Design an animation by introducing time into transformations
- Analyze an animation model
- …

NANYANG TECHNOLOGICAL UNIVERSITY

4. Visual appearance (I)

- Color representation: (R,G,B)
- 3 light sources
- Phong illumination model
  - Ambient reflection
  - Diffuse reflection
  - Specular reflection

$$I = k_a I_a + \sum_{\substack{\text{for each}\\ \text{light } i}} k_d I_i \cos\theta_i + \sum_{\substack{\text{for each}\\ \text{light } s}} k_s I_i (\cos\phi_i)^n$$



- Computation
  - How to compute **unit** vectors L, N, V
  - Vector R = 2 (N •L) N – L

NANYANG TECHNOLOGICAL UNIVERSITY

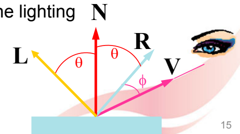
Visual appearance (I)

- Lighting vector L: a vector from a point on the surface towards a light source
- Viewing vector V: a vector from a point on the surface towards the viewer
- Normal vector N: a vector that is perpendicular to the tangent plane of the surface

  For example, for an implicit surface f(x,y,z) = 0, its normal vector is

$$N(x,y,z) = \pm\left[\frac{\partial f}{\partial x} \quad \frac{\partial f}{\partial y} \quad \frac{\partial f}{\partial z}\right]$$

- Reflected vector R: the image of the lighting vector L reflected off the surface



NANYANG TECHNOLOGICAL UNIVERSITY

Typical problems

- Use Phong illumination model to perform some analysis

- How to compute the reflected vector R
- How to compute the normal of a surface

- How to compute the diffuse reflection
- How to compute the specular reflection
- How to compute the overall illumination

- Design functions for r, g, b to define diffuseColor in FVRML

NANYANG TECHNOLOGICAL UNIVERSITY

5. Visual appearance (II)

- Texture mapping
  - Texture (image) is used to change the color.
- Bump mapping
  - Bump map is used to change the normal of the surface.
- Displacement mapping
  - Geometric texture is used to change the surface.

NANYANG TECHNOLOGICAL UNIVERSITY

Visual appearance (II)

Texture domain (u,v) ⟺ Surface parameter domain (s,t) ⟺ Surface point (x,y,z)

Algorithm (parametric texture mapping)

Step 1: Parameterize texture with (u,v) coordinates

Step 2: Parameterize the surface with (s,t) coordinates

$x=x(s,t),\ y=y(s,t),\ z=z(s,t),\quad s \in [s_0,s_1],\ t \in [t_0,t_1]$

Step 3: Define a mapping between (u,v) and (s,t)

$$\frac{u-u_0}{u_1-u_0}=\frac{s-s_0}{s_1-s_0},\quad \frac{v-v_0}{v_1-v_0}=\frac{t-t_0}{t_1-t_0}$$

Step 4: (x,y,z) ➔(u,v) or (u,v) ➔(x,y,z)

Typical problems

- How to perform forward mapping
- How to perform inverse mapping

- The concepts of the three surface mapping methods
- The properties of the three surface mapping methods
- Analyze the basic shape and the geometric texture

- Conduct texture mapping
- Conduct displacement mapping & its function-based extension

NANYANG TECHNOLOGICAL UNIVERSITY