Computer image generation overview

Example of computer generated image

<IMAGE: synthetic image which motivates this lecture. We will break image into parts which make pillars of computer image generation>

Goal of computer graphics

- As in computer science, computer graphics problems are solved using **simulation and approximation.**
  - The real world which is simulated is continuous.
- Computer graphics is solving a **problem of generating (synthesizing) images**.
- Computer graphics simulations and its results (generated image) are **discrete**.
  - Computer generated image and display device are 2D array of **pixels** – **raster** graphics and display (grid of x and y coordinates on a display device).

Pillars of computer graphics

- To synthesize an image:
  - We need a **3D scene** which simulates objects and phenomena
  - A way to transform a 3D scene into 2D array of pixels – **rendering**.
  - A way of storing 2D array of pixels for raster display – **image**.

- Computer graphics R&D is this concerned with:
  - Modeling of 3D scene
  - Development of rendering algorithms for creating image out of 3D scene
  - Storing and displaying image.

- Image generation software can be thus split into:
  - **Modeling** – process in which 3D scene is defined
  - **Rendering** – process in which image of 3D scene is generated

- Tasks of computer graphics are intertwined:
  - Rendering process must "understand" how 3D scene is represented
  - Modeling of 3D scene relies on representations which are understood by rendering algorithm

3D scene modeling – a start of image generation journey*

- In order to generate an image (to render it) we need something to render → a 3D scene.
- 3D scene is simulated 3D world from which images can be taken - similarly as photographing a real world. Thus, we also need to simulate a device which is needed for taking images – a **camera** – which also defines from where image is taken.
- The real world is made of wide range of phenomena and objects. For modeling of a 3D scene we would like to simplify those. Therefore we describe **objects** by **shape and appearance**.
- The reason why we see objects around us is due to light reflecting in our eyes. Therefore, for purposes of simulation, we can separate objects in those which generate light (**light sources**) and those which only reflect light.

* As discussed, 3D scene modeling and rendering in order to obtain an image is highly intertwined process. For learning purposes and as well as in research in development this process is separated. Depending on application and teaching the description of whole process can start from rendering as well.

Objects in 3D scene

- Real world objects shapes and appearances are varying. Take for example cloud, chair and river stream.
  - In graphics, we simplify, and categorize different objects being **solid (volume)** or **surface**. Volume objects are used to represent interior since we can see through it (e.g., water). For surface objects, we only care about exterior (e.g., tree)*

    <IMAGE: DIFFERENT OBJECTS AND REPRESENTATIONS>

* Representing objects is not only important for visualizing purposes, it is also important for modeling them. Therefore, even some objects for which we only care about the surface appearance (e.g., car rim) we would like to represent them as solids in certain modeling tools which would enable simulation of cutting and modling the surface). For example: https://www.autodesk.com/products/fusion-360/features#3d-modeling or https://pixologic.com/.

Concept of 3D space

- Before we go to further, we need to understand that all objects in 3D scene are “living” in a 3D space.
- To define any object we need to introduce a concept of 3D space.
- 3D space is represented with 3 coordinate axis. This main coordinate system is called **world.**
  - Coordinate system representation which we are using is **Cartesian coordinate system*.**
- All objects have a position in this world coordinate system.

<IMAGE: coordinate system as concept of space>

* Other representations for 3D space are also possible, one very commonly used is spherical coordinate system.

Surface shape representation 1

- To define object’s shape in a 3D scene, we need to define a concept of points in 3D space.
- **Point** is defined as three floating point numbers for each of Cartesian coordinate system **(x,y,z)**.
- To define a surface, simplest way is to connect points to form a **polygon**. In computer graphics, for computation tractability, we often use co-planar – all points lying on the same plane - polygons and especially **triangles**.
  - Triangle is almost widely used surface shape representation **primitive.** This is because it is very simple and holds great properties for easy calculation thus very much researched and used for efficient rendering purposes. Different shape representations are also introduced for easier modeling, but it is very often that all representations are turned to triangles before/during rendering stage - using very elaborated method called **triangulation**.
- Triangle is basic building block for creating more complex shapes. And modeling is all about creating complex shapes using basic building blocks.

<IMAGE: POINTS, TRIANGLES, COMPLEX SHAPES>

Surface shape representation 2

- Some shapes do not have flat surfaces! Polygonal shape representation will always have flat surfaces*.
- If we would like to represent curved surface, we would need smaller triangles which would fit the curved surface better. In this process we are placing more points on to surface (sampling). Generally, converting smooth surface to triangulated polygon representation – a **triangulated mesh** – is called **tessellation**.
- Main point to take away is that we can represent any object using triangle polygons. Those objects will never be perfect representations of a real world, but triangles of which they consists of can be small enough to display objects in high quality taking in account restrictions of raster display.
  - Amount of details increase realistic representation but also complexity of rendering. Computer graphics often deals with trade-off between visual quality and speed**.

* There are methods in rendering which make surface looks smooth although is represented using polygons. This method is called smooth shading and we will discuss it later.

Surface shape representation 3

- Although polygonal meshes are widely used and popular (e.g., games and film) there are other representations that are more suitable for modeling purposes
- One of these are **curved surfaces** and **subdivision surfaces** used to design manufactured objects and often used in CAD software.
- Foundation of those representations are still **points**, but those points define a **control mesh** from which perfect curved surface can be generated using **analytical description**.
  - Note that this kind of representation is much more compact than polygonal mesh.
- Modeling using control points is very beneficial for curved objects. When it comes to rendering, this representation can not be rendered directly. As discussed, process called **tessellation** must be performed prior rendering.

<IMAGE: CURVED AND SUBDIVISIONS SURFACES>

Note: How are objects created?

- Lot of 3D objects are created by hand
  - Lot of mesh and curve modifiers as well as techniques and methods are developer for faster and easier design
- Some objects are digitized from real world
  - 3D scanning can be used to “import” real object into digital world.
- Some objects are simulated
  - Some objects are very hard to simulate and capture from a real world. For example water stream. For these purposes physical simulations are also employed to generate shape.

<IMAGES: manual modeling, capturing from real world, simulating>

Surface shape representation 4

- For generating shape, simulation can be used, e.g., smoke simulation. For this purposes, it is required to represent 3D space in which simulation is performed as grid of cells which are called **voxels**.
- Each voxel (a cell) is a small volume of space on which computation is performed. Simulation is performed in series of steps. Each step is recorded and transformed to triangulated mesh for rendering.
  - This way, **animated** mesh can be generated.
- Voxels are also widely used for any kind of modeling purposes where it is required to describe object’s interior – e.g., digital sculpting.

Shape representation for modeling

- Constructing complex shapes can be also done using basic primitives (box, sphere, etc.) and series of operations (add, substract, etc.). This kind of modeling is called Constructive Solid Geometry.
- An representation for such basic primitives is called implicit since they are completely analytically defined.
- This enables fast and flexible modeling system, but when it comes to rendering, they need to be converted to triangulated mesh – similarly as for curved surfaces. Algorithm in this case is called marching cubes.

<CSG AND IMPLICIT SHAPES MODELING>

Modeling and representing 3D objects: keypoints

- From previous discussion it is important to note that different representations for 3D models exists and that different representation are better for rendering (triangulated mesh) and other are better for modeling (e.g., implicit surfaces).
- In professional software, different representations for modeling will be available, but also, those are converted to triangulated mesh for efficient rendering.
  - Having one representation for rendering makes rendering process highly efficient since all effort is put into efficient methods for rendering one representation.
  - Converting all objects to same representation (triangulated mesh) is also convenient for applying various rendering effects. For example, techniques for adding more details during rendering time can be only developed to work with triangulated mesh.
  - Lot of research and hardware development was focused onto efficient rendering of triangles as we will see*.

* It is easy to imagine that different rendering primitive is used. But due to historical events this turned out to be a triangle. What is important to note is that mapping of various representations to triangle mesh is possible and thus it is not important which is the rendering primitive as long as it supports various representations.

Other important information on 3D objects

- Until now we discussed shape representation of 3D object.
- Shape is important for determining how object will appear, but also its appearance depends on material and surface details.

Light in 3D scene

Camera in 3D scene

Rendering of 3D scene