

1.

百科上对计算机图形学的定义为是一种使用数学算法将二维或三维图形转化为计算机显示器上的科学。按我的理解简单来说就是利用一些数学算法来将图形在显示屏上显示出来，并且能够其它方面的处理，使之来显示，逼近一些现实上的东西，或者是将一些本来不可视的东西，像一些计算，统计结果给可视化。

而根据链接中所说，计算机图形学主要包含四大部分的内容来完成上述的工作：建模，渲染，动画和人机交互。建模是用数学上的函数或者是曲面上的采样点及其连接关系所表达的三角网格来表达。渲染就是在建模之后通过不同的渲染方法将模型画出来，实现复杂，逼真的视觉特效。而动画就是采用连续播放静止图像的方法产生物体运动的效果。最后人机交互就是人和机器之间通过一定的交互方式进行信息交换，让计算机完成人所需的任务。

根据我的理解，以上四个部分就是一个渐进的过程，先是通过建模来表达出我们想要显示的事物模型，然后通过渲染方法来将其逼真的显示在显示屏上，以上都是一些静态的画面，所以进而就是通过这些静态的画面来连续播放制作出动画，最后就是利用上述所得的静态，动态画面或者说利用人的动作进行建模识别进行人机交互，获得更好的用户体验。

2.

OpenGL 是一个由 Khronos 组织制定并维护的规范，规定了每个函数应该怎么执行以及他们的输出。而 OpenGL 库的具体实现是由开发者自己决定的，只要功能和输出结果和规范符合就行。

OpenGL ES 是 OpenGL 三维图形 API 的一部分，针对手机、PDA 和游戏主机等嵌入式设备而设计，去除了 glBegin/glEnd，四边形、多边形等复杂图元等非必要的特性。

WebGL 是一种 3D 绘图协议，允许把 JavaScript 和 OpenGL ES 2.0 结合在一起，通过增加 OpenGL ES 2.0 的一个 JavaScript 绑定，WebGL 可以为 HTML5 的 Canvas 提供硬件 3D 加速渲染，在浏览器里更流畅地展示 3D 场景和模型和增加数据可视化功能。

Vulkan 是一个跨平台的 2D 和 3D 绘图应用程序接口，针对实时 3D 程序设计，特性有独特的跨 OS 支持，能够支持深入硬件底层的控制，带来更快的性能和更高的影像质量，还提供超高的 OS 兼容性、渲染特性和硬件效率。还有自动兼容 GCN 架构，利用基于 GCN 架构的 Radeon™ 显卡拥有强大的“异步计算”功能。

DirectX，是由微软公司创建的多媒体编程接口。由 C++ 编程语言实现，遵循 COM。被广泛使用于 Microsoft Windows、Microsoft XBOX、Microsoft XBOX 360 和 Microsoft XBOX ONE 电子游戏开发，并且只能支持这些平台。加强 3D 图形和声音效果，并提供设计人员一个共同的硬件驱动标准，让游戏开发者不必为每一品牌的硬件来写不同的驱动程序，也降低了用户安装及设置硬件的复杂度。

3.

gl 库是核心库，包含了最基本的 3D 函数。

glu 是实用库，为了减轻繁重的编程工作，封装了 OpenGL 函数，Glu 函数通过调用核心库 gl 的函数，为开发者提供相对简单的用法，实现一些较为复杂的操作。

glew 是一个跨平台的 C++ 扩展库，基于 OpenGL 图形接口。GLEW 能自动识别你的平台所支持的全部 OpenGL 高级扩展函数。也就是说，只要包含一个 glew.h 头文件，你就能使用 gl,glu,glExt,wgl,glx 的全部函数，支持目前流行的各种操作系统。

4.

使用 GLFW 是因为我们在创建 OpenGL 上下文和显示窗口的时候，操作在每个不同系统上是不一样的，而 OpenGL 有目的地从这些操作抽象(Abstract)出去。这意味着我们不得不自己处理创建窗口，定义 OpenGL 上下文以及处理用户输入。使用 GLFW 可以节省了我們书写操作系统相关代码的时间，提供给我们一个窗口和上下文用来渲染。

Freeglut 和 GLFW 的作用是差不多的。

5.

我比较喜欢的专题是 VR/AR 专题，在 2017 年的 SIGGRAPH 上，NVIDIA 提出了一个机器人培养系统，使机器人可以在与虚拟世界里面进行学习和探索，相当于一个机器学习的过程，降低了用机器学习培养机器人的成本，也降低了发生不可控危险的可能性，将机器人的训练通过 VR 来实现可视化。而在 AR 方面的话，提出了变焦虚拟的思路，即近视显示的新型光学设计，可以让不在眼睛当前可视距离的物体变模糊，使用了 ChromaBlur 模糊技术进行计算，考虑了人眼的高阶和彩色像差。