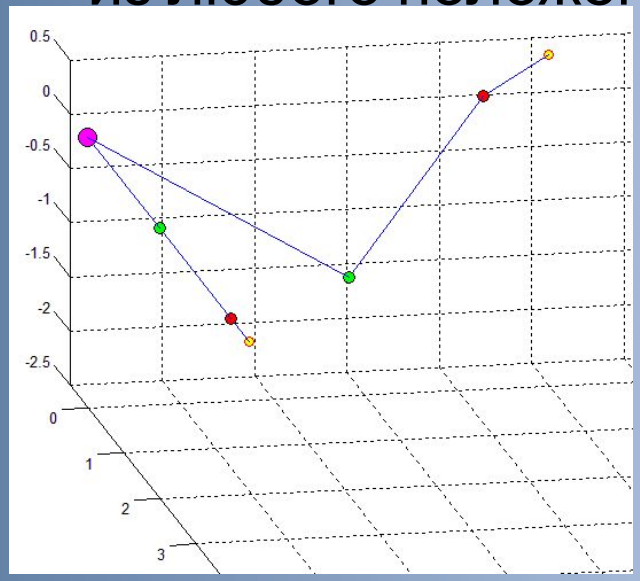
Курсовой проект по предмету
«Информатика»

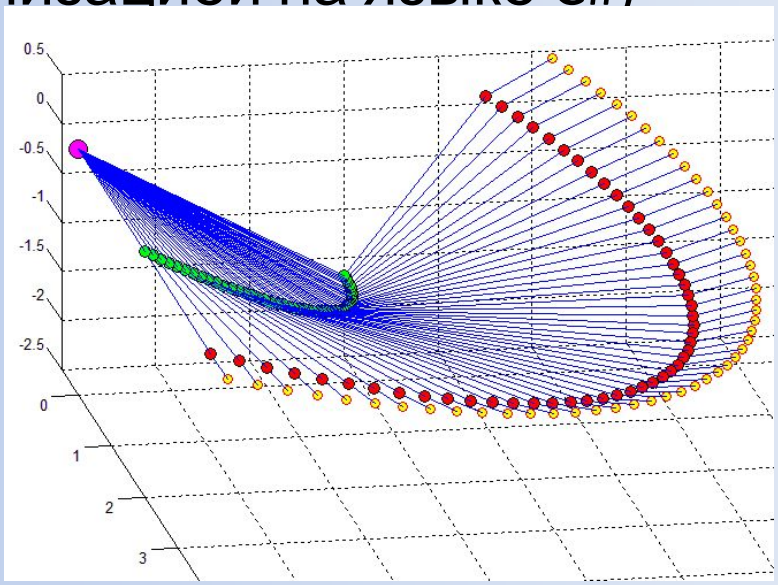
Математическая модель руки фехтовальщика

Выполнил:
Студент гр.20510/1 ФМФ
Дзенушко Дайнис

Цель работы

Создать математическую модель, которая по заданным
параметрам (углам в суставах, координатам плеча) восстанавливает положение руки и приходит в него из любого положения (с визуализацией на языке C#)

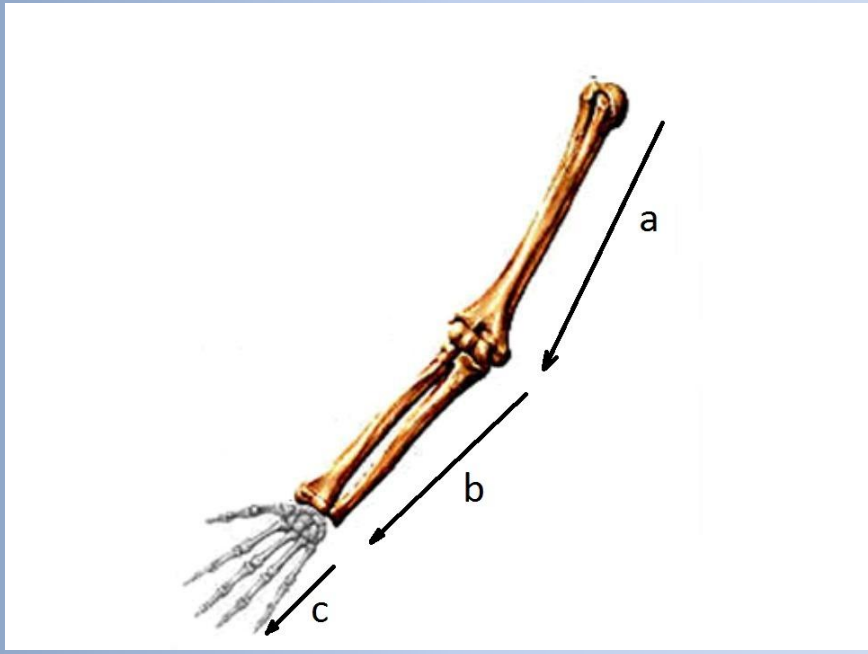




Степени свободы

Наша модель руки имеет 9 степеней свободы:

2 — поступательное движение плеча (вперед/назад; вверх/вниз)
3 — плечевой сустав (сферический шарнир)
1 — локтевой сустав
1 — вращение кистью вокруг своей оси
2 — запястье (2 перпендикулярных цилиндрических шарнира)



Поворот руки

Повороты частей осуществляются посредством поворота соответствующего
вектора вокруг определенной оси. Это достигается посредством скалярного
умножения вектора слева на тензор поворота, который рассчитывается
следующим образом

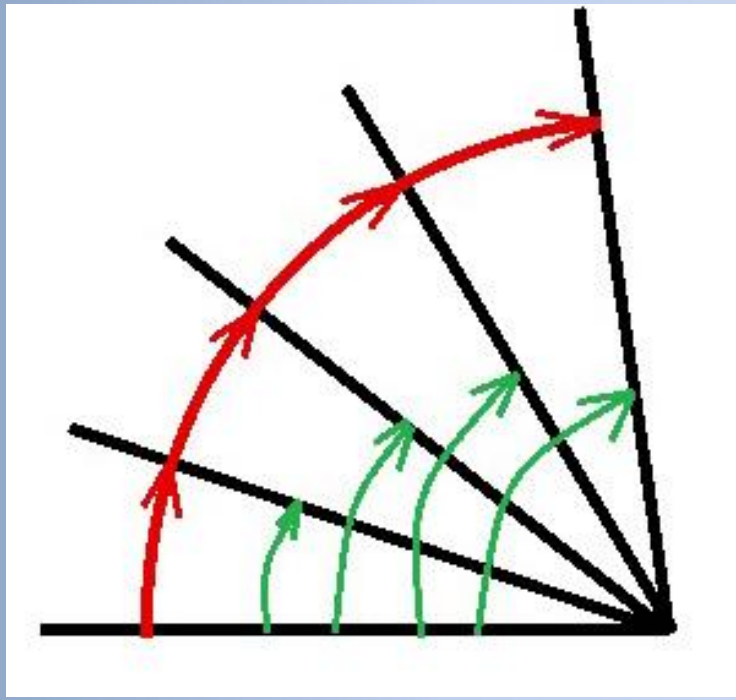
$$\underline{\underline{P}}(\phi, \underline{e}) = \underline{e}\underline{e} + (\underline{\underline{E}} - \underline{e}\underline{e})cos(\phi) + \underline{e} \times \underline{\underline{E}}sin(\phi)$$

$$\tilde{\underline{a}} = \underline{\underline{P}}(\xi, \tilde{\tilde{\underline{k}}}) \cdot \underline{\underline{P}}(\psi, \tilde{\underline{j}}) \cdot \underline{\underline{P}}(\varphi, \underline{i}) \cdot \underline{a}$$

После этого необходимо также повернуть остальные (отстоящие еще дальше от
плеча чем рассматриваемый) векторы и соответствующие им базисы также
умножив их слева на тот же тензор поворота. Таким образом осуществляется
поворот всех 3х векторов на необходимые углы.
Затем вычисляем координаты суставов.

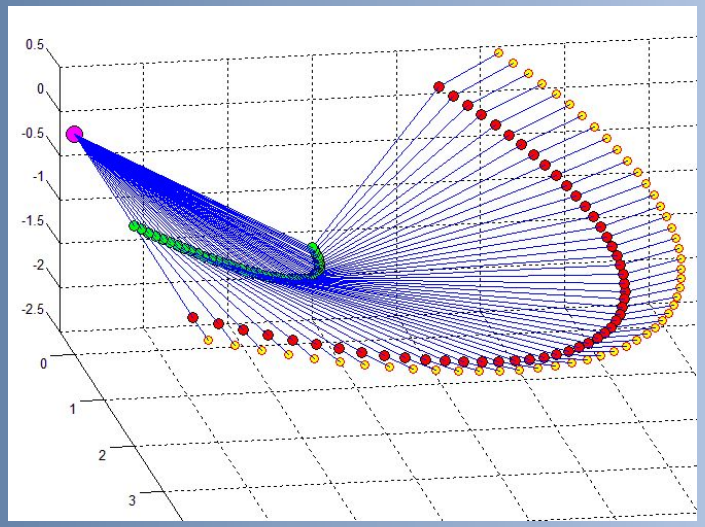
Реализация вращения

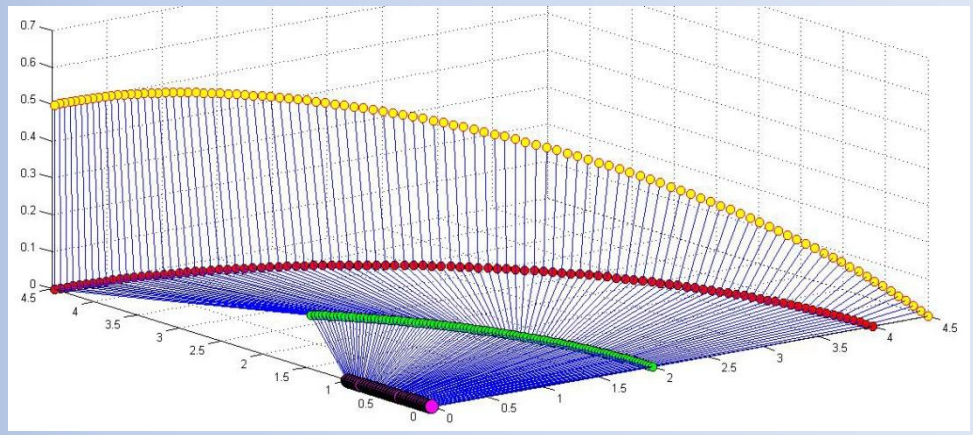
Для визуализации вместо того чтобы поворачивать руку на небольшие углы а потом из нового положения поворачивать ее дальше, было решено поворачивать руку на каждом шаге из начальной точки траектории с каждым шагом на все большие углы.



Реализация в пакете MATLAB

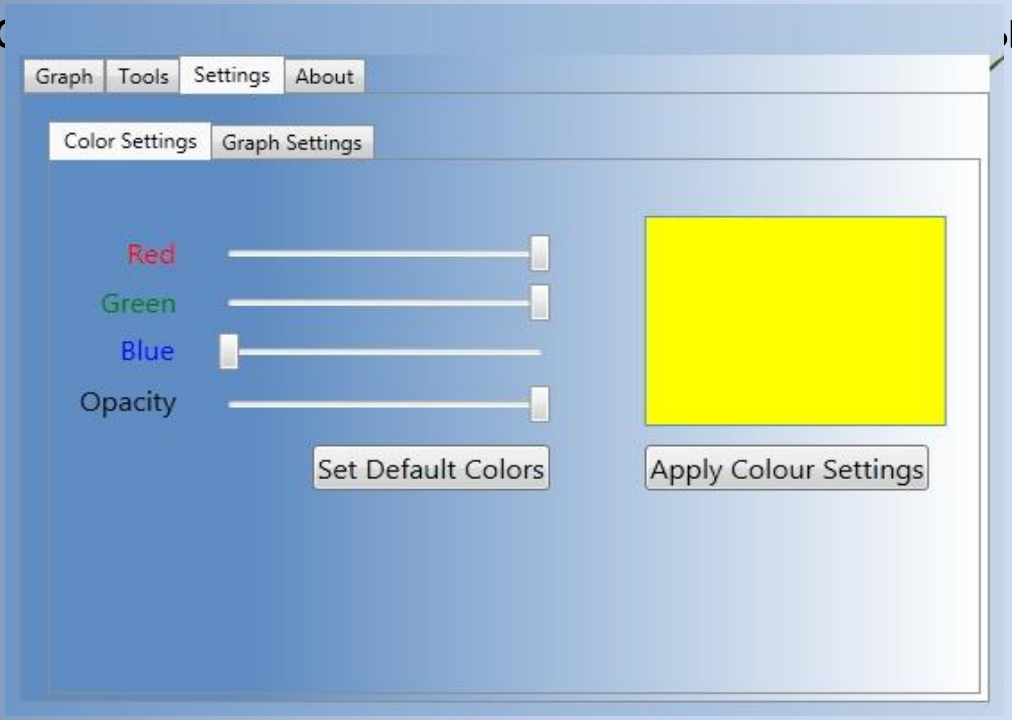
Используя пакет MATLAB, был реализован данный алгоритм
поворота, с целью построить траекторию движения руки и
проверить правильность алгоритма.





Структура интерфейса программы (на языке C#)

Интерфейс реализован на основе вкладок (TabControl). И состоит из двух
уровней: внешнего (Graph, Tools, Settings, About) и внутреннего для вкладки Settings (Color Settings, Graph Settings)
Далее рассмо ые вкладки.



Вкладка Graph

Данная вкладка содержит визуализацию нынешнего положения руки и обладает следующим функционалом: вращение камеры и Zoom.

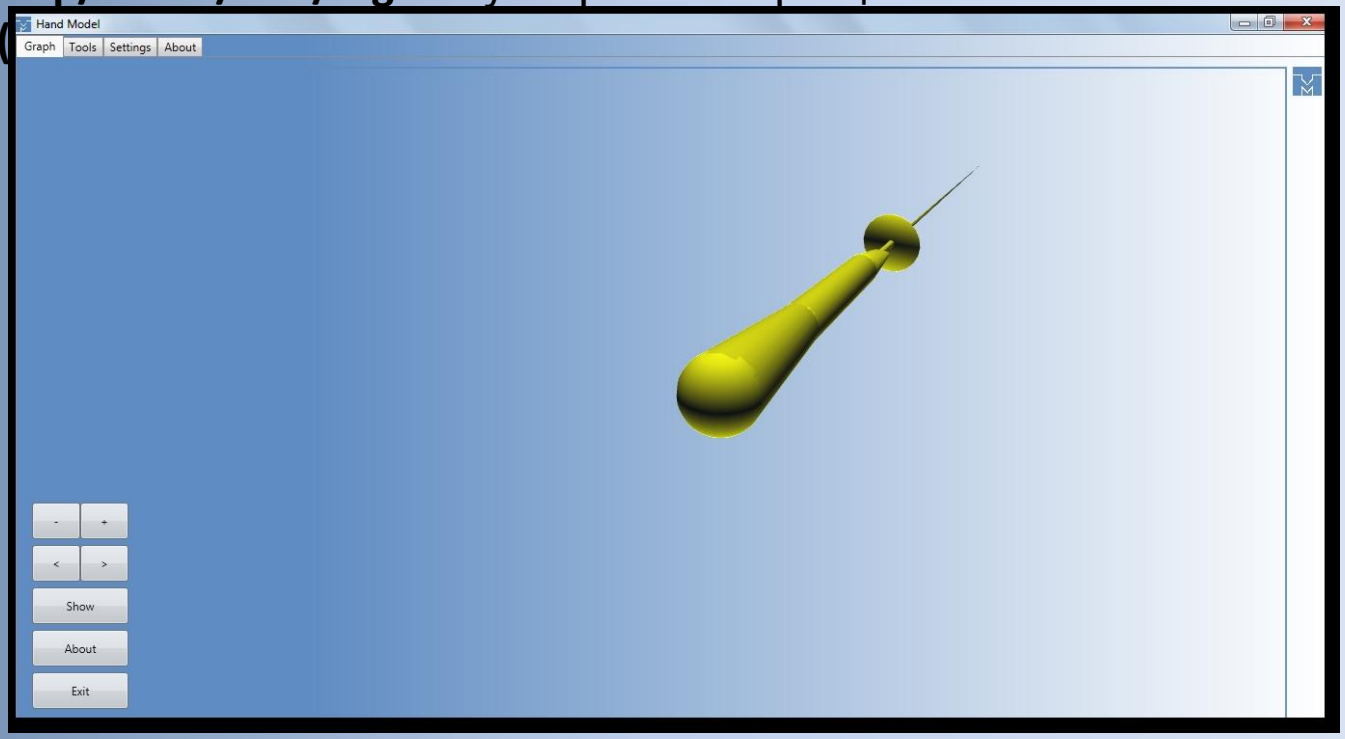
В ней работают комбинации клавиш:

Up/Down – вращение вокруг горизонтальной оси

Left/Right – вращение вокруг вертикальной оси

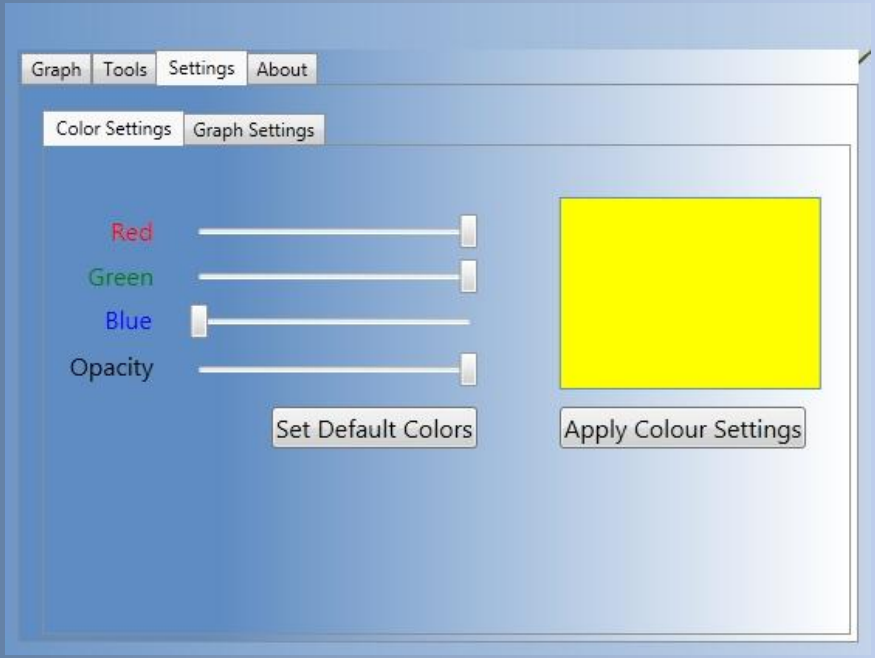
Ctrl+Up/Down/Left/Right – ускоренное вращение

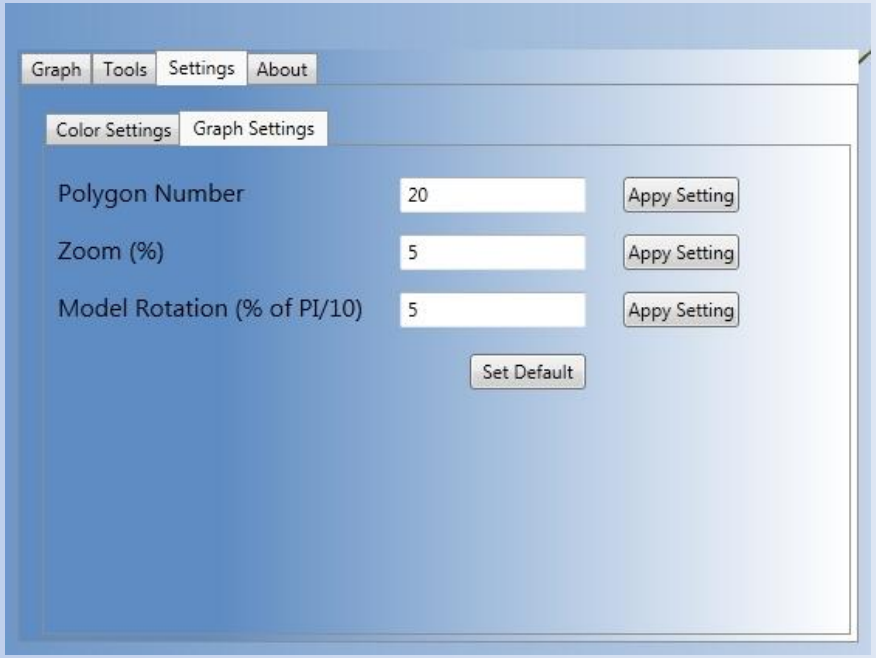
+/- (



Вкладка Settings

Данная вкладка отвечает за настройку параметров отображения модели. Она содержит две подвкладки.
Первая (Color Settings) позволяет настроить цвет и прозрачность модели.
Вторая (Graph Settings) позволяет задавать качество модели, а также коэффициенты поворота и Zoom.



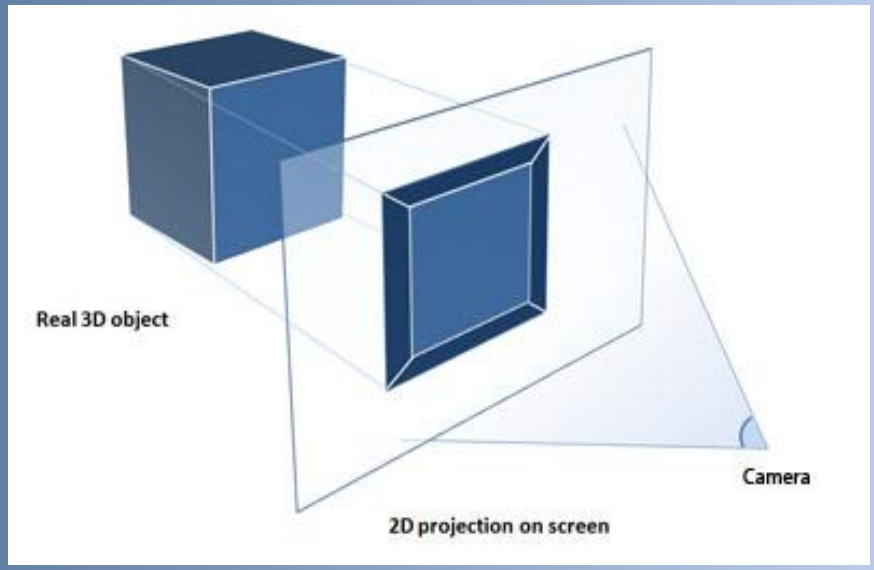


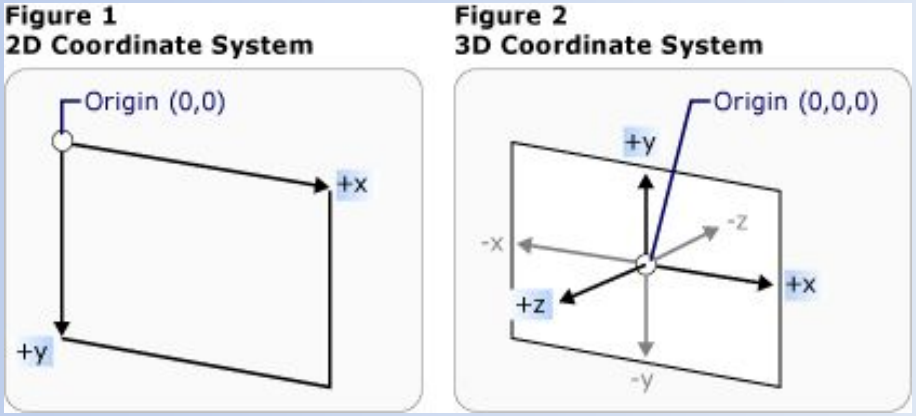
Программная реализация 3D-графики

Интерфейс программы реализован при помощи библиотеки WPF (Windows Presentation Foundation) и встроенного в Microsoft Visual Studio редактора языка XAML.

Элемент Viewport3D

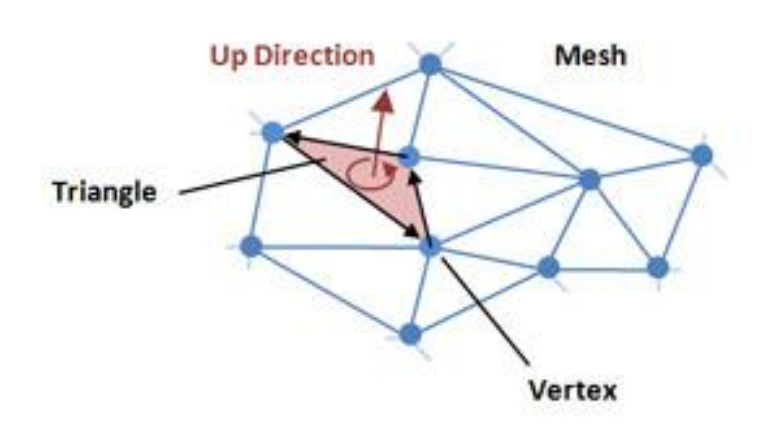
Графическое содержимое 3-D в приложении WPF инкапсулировано в элементе Viewport3D, который может участвовать в структуре двумерного элемента. Графическая система рассматривает Viewport3D как двухмерный визуальный элемент, подобный многим другим в WPF. Viewport3D функционирует как окно просмотра трехмерной сцены. Говоря точнее, это поверхность, на которую проецируется сцена 3-D.

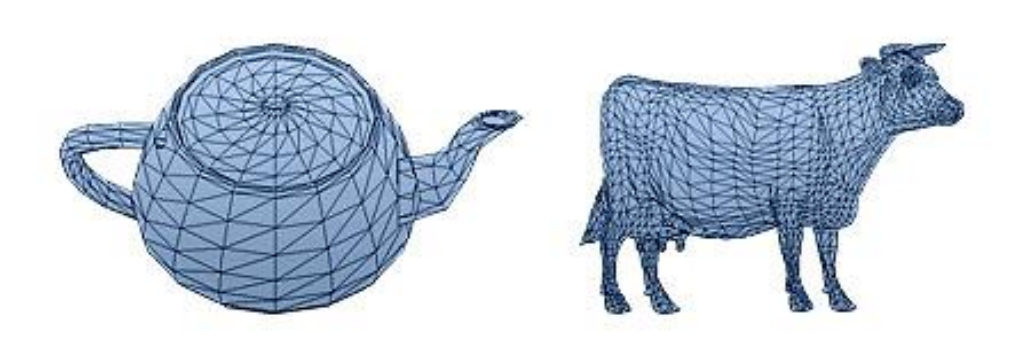




Рисование объектов в 3-D

Рисование объектов осуществляется путем их триангуляции т.е. составления их из треугольников. Заметим что одна сторона треугольника видима а противоположная прозрачна.





Спасибо за внимание