

Rasterizzazione

Algoritmo di rendering, anche detto *transform & lighting*:

- proiezione della scena tridimensionale sul piano dello schermo;
- separatamente, calcolo dell'illuminazione.

La scena è composta da primitive che sappiamo proiettare e rasterizzare (convertire in pixel), cioè punti, segmenti e triangoli.

Costo:

$$kv + \sum_{p \in S} R(p),$$

dove k è il costo di trasformare un vertice, v è il numero di vertici, S è l'insieme di primitive e R è il costo di rasterizzare p .

A differenza del ray tracing, la rasterizzazione ha una complessità più prevedibile, è più efficiente sulle GPU correnti e si adatta meglio a scene dinamiche. Tuttavia rende difficile tenere in considerazione gli effetti globali della luce e può far uso di un insieme molto ristretto di primitive.

La rasterizzazione è adatta ad una pipeline: si scorrono tutte le primitive processandole una sola volta. Nel ray tracing, per calcolare i rimbalzi è necessario tenere sempre in memoria l'intera scena.

Rasterizzazione di primitive

punti si colora il pixel in cui si trovano (*splat*);

segmenti ci si sposta a passi di un pixel lungo la linea colorando i pixel che si incontrano;

triangoli si colorano i pixel che hanno il centro dentro il triangolo.