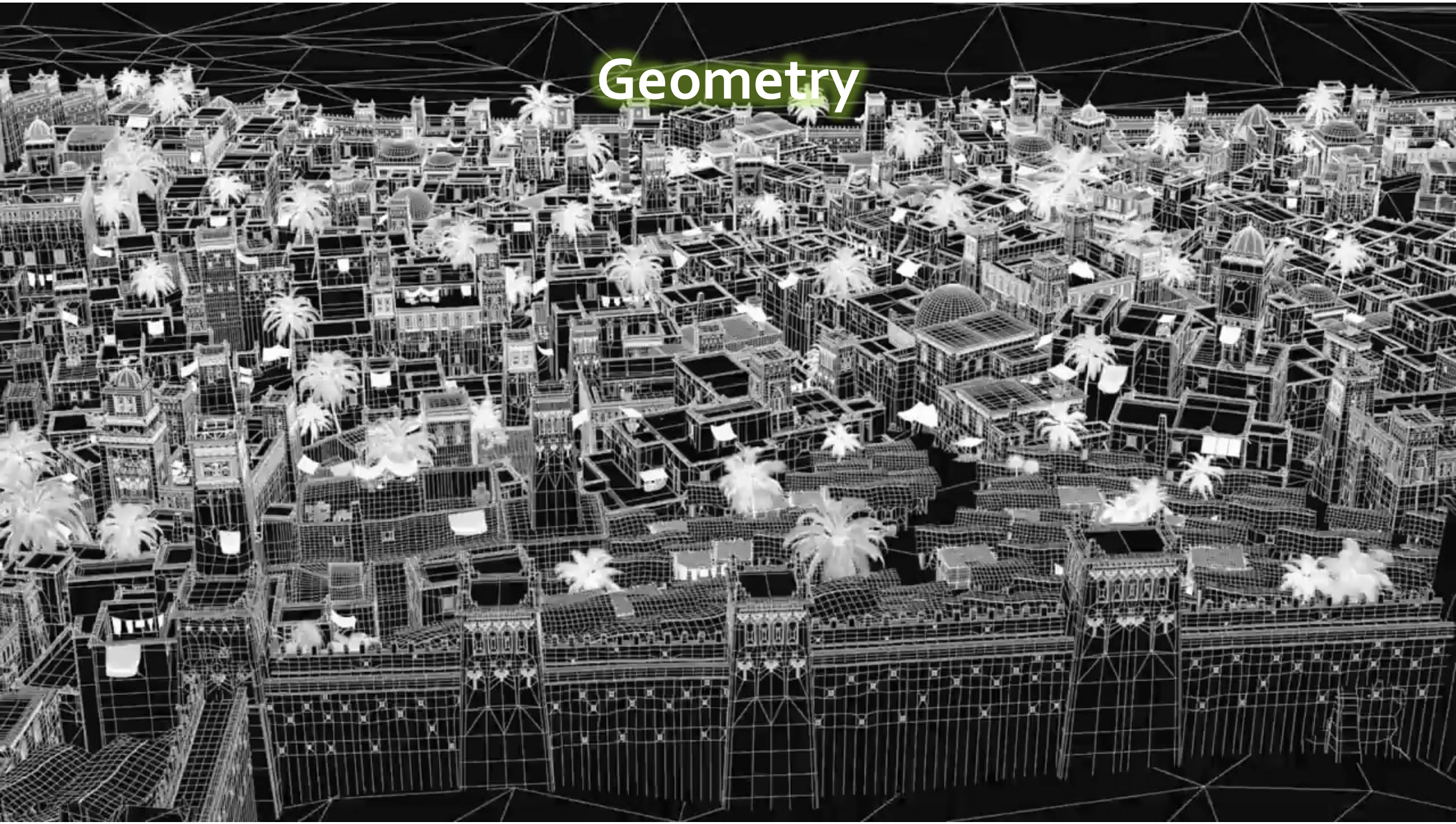
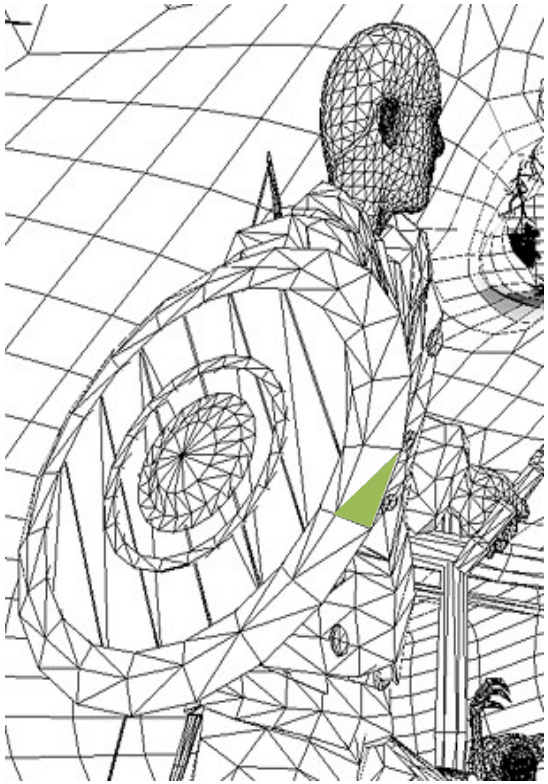
What is Rendering?


Final Image


Geometry

Rendering = turn Geometry into Image



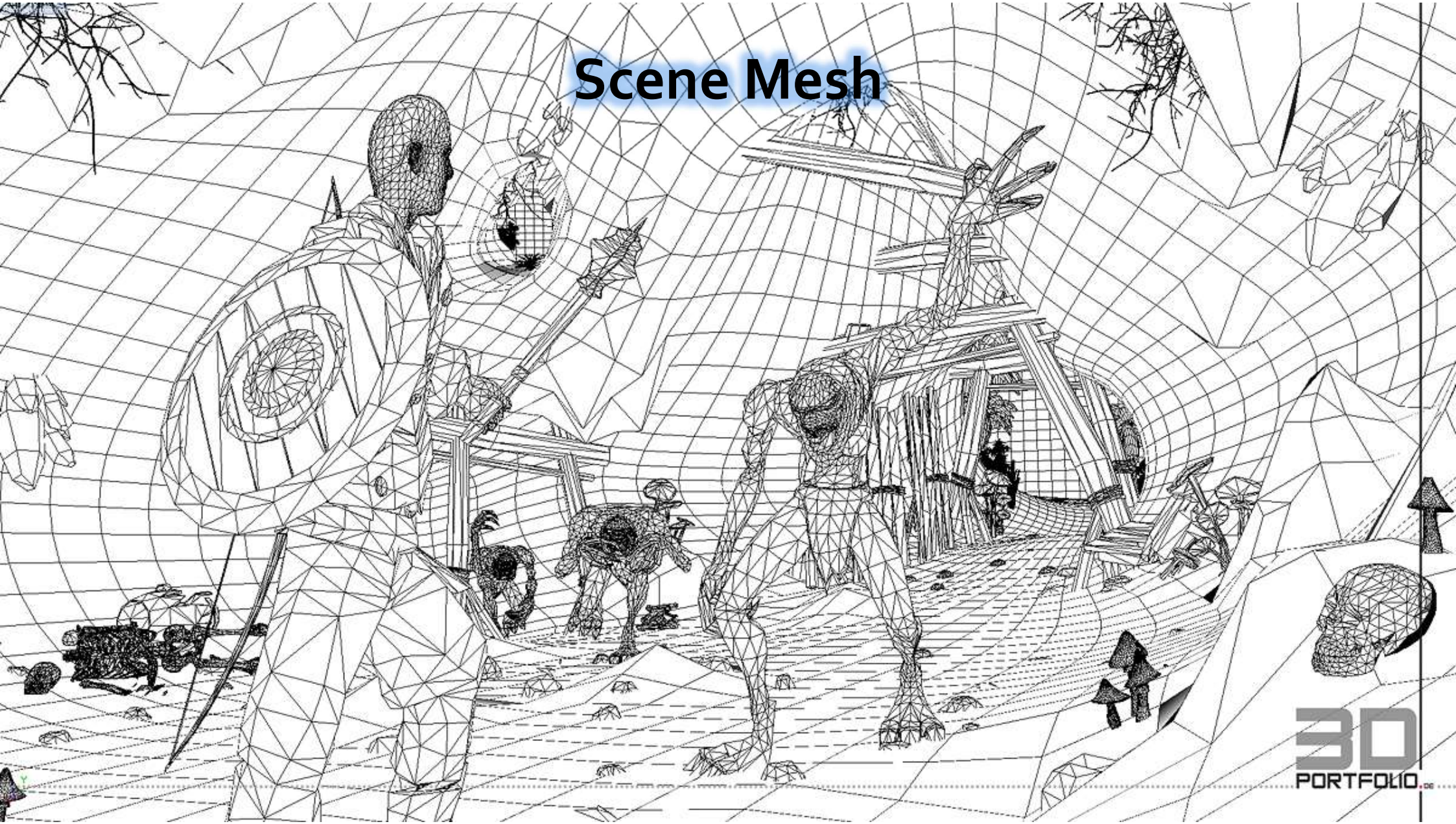
rendering

Rendering Pipeline

What For?

- Insights into **how** things work
  - Understanding algorithms
- Insights into how **fast** things work
  - Performance
- The pipeline is the "engine" that creates images from 3D scenes
- Usually in hardware (on graphics card)
- CPU side only tells hardware what triangles to render (game play, AI, Collision detection, …), GPU does the rest
- What is the rest?

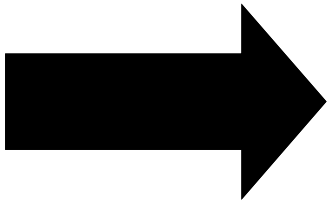
Scene Mesh



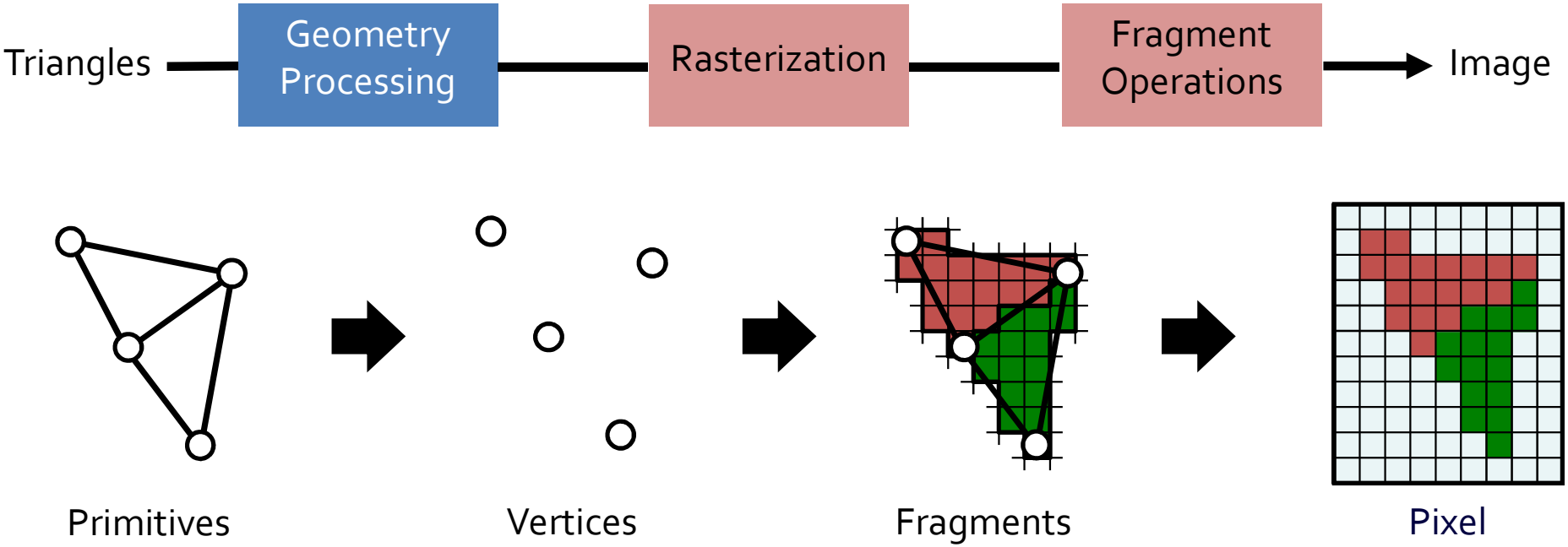

Untextured, shaded
3D
PORTFOLIO


RISEN
Final

Rendering Pipeline

Triangles  Image

Rendering Pipeline

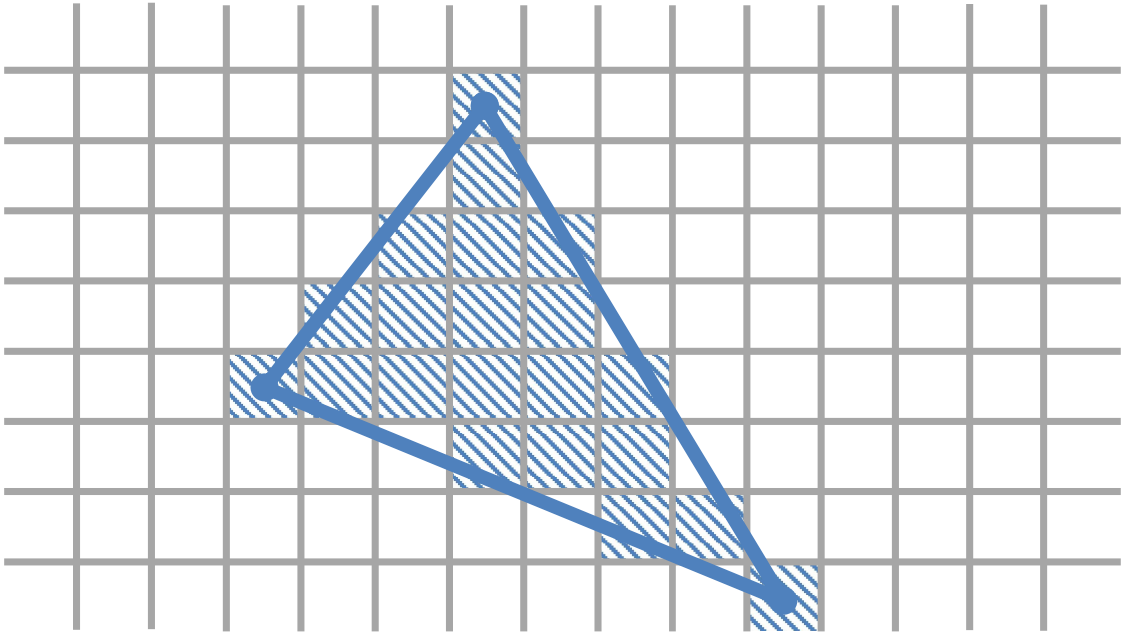


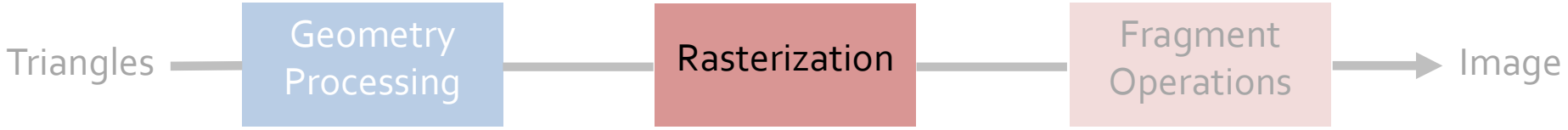
Geometry Processing

- Task: "geometrical" (2D, 3D) operations on the input data (triangles)
  - Animate objects
  - Move objects
  - Move the camera
  - Project onto screen (3D to 2D)
  - Clipping (avoid triangle(s) (parts) outside screen)
  - Execute geometrical shaders (tessellation, geometry, vertex, …)

Rasterization

- Task: turn output from geometry stage into fragments (potential pixels)





Fragment Operations

- Task: mainly combining and/or filtering of fragments to get pixels
- Filtering by
  - **Zbuffer test** (resolve visibility)
  - Alpha test (filter transparent fragments via threshold)
  - Stencil test, scissor test, write mask
- Combining by
  - Blending operations (for semi transparency)
  - Logical operations (logical combinations of fragments and framebuffer)