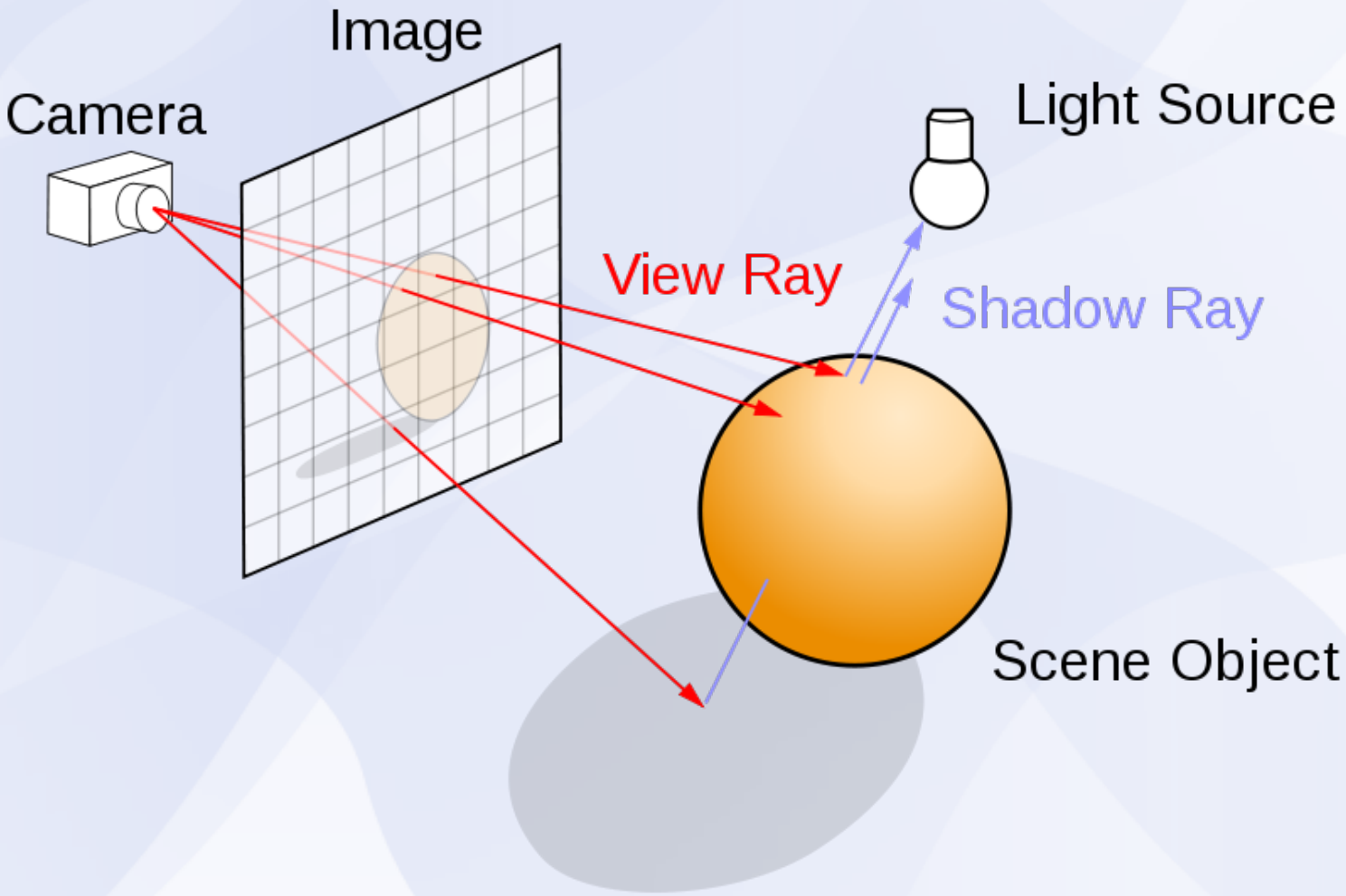
CS420 Assignment 3 Hints

Ray Tracing

Step 1: send rays



- Send out rays from camera position (0,0,0) pointing to −z
- Image size 640x480
  - For debugging, use smaller size

- Send out rays from camera position (0,0,0) pointing to −z
- Image size 640x480
  - For debugging, use smaller size

Step 2: Intersect with scene

- Sphere & triangle
- Analytical solution

Debugging

- Do step by step
  - Intersect with sphere, test code
  - Intersect with triangle, test code
  - Compute sphere color, test code
  - Compute triangle color, test code

Tips

- Ensure B != 0 when dividing A / B
- Before calling sqrt(x), make sure x >= 0
- Remember to normalize the direction vector
- Remember to check len(dir) != 0 before dividing by the length

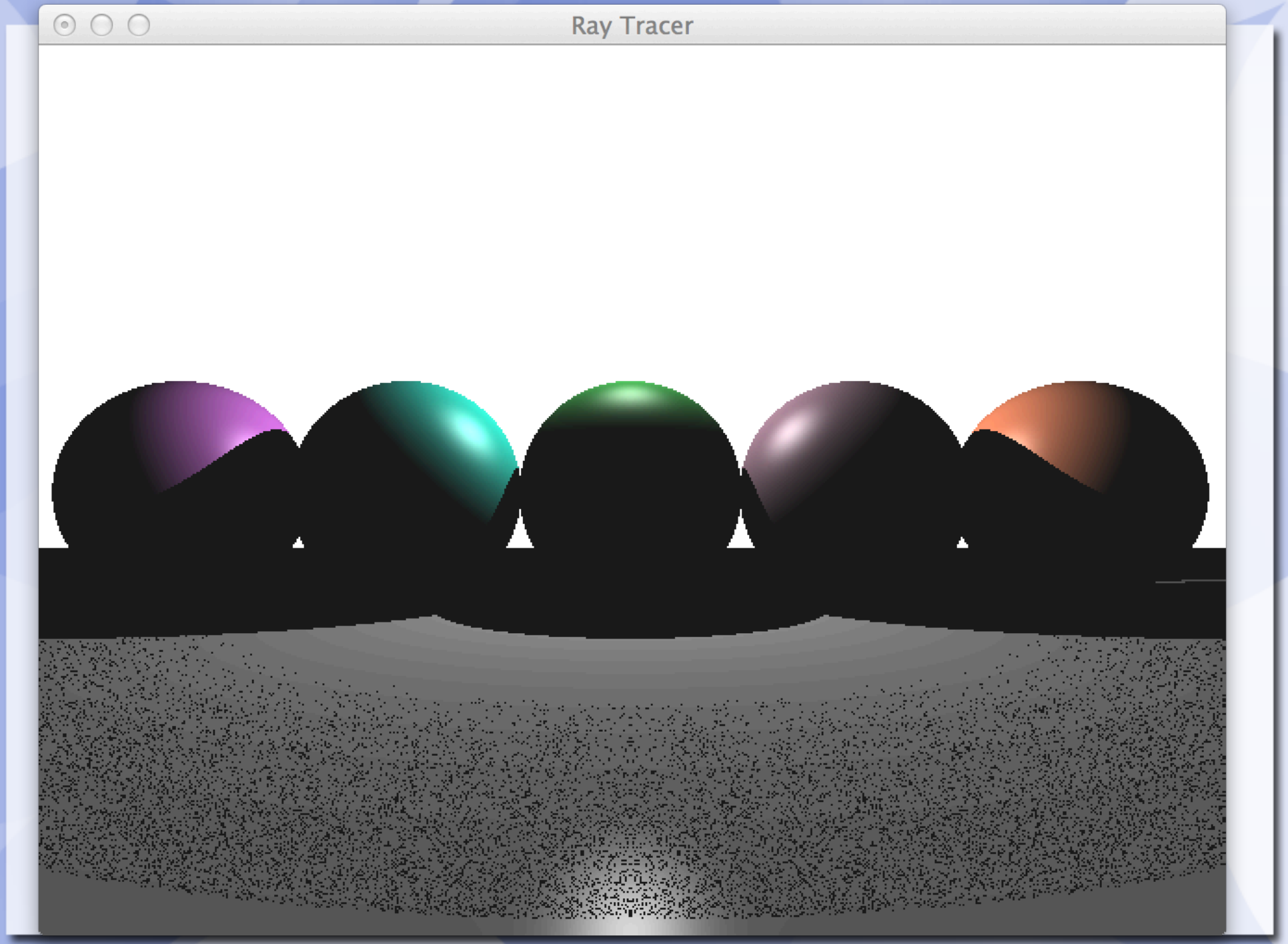
Tips

- Distinguish between normals:
  - normal of a triangle
  - vertex normal
  - normal interpolated from vertex normals

Tips

- Floating-point operations not accurate:
  - When computing shadow rays, use:

    distanceFromLightToFirstObject < distanceFromLightToTargetSurface – smallValue
  - Otherwise, artifacts appear… (see next image)


Ray Tracer

Extra Credits

- Super-sampling
  - anti-aliasing
  - can do adaptively: if some region is smooth, no need to super sampling
- Ray tracing

  (1-ks)*localPhongColor + ks*colorOfReflectedRay
- You can also add refraction

Extra Credit (Cont'd)

- Animation
- Soft shadows
- Parallel computing for faster rendering
  - OpenMP: utilize multi-core CPUs
  - Cuda: use GPU to do parallel computing