Accelerated Deterministic Simulation of X-ray Attenuation Using Graphics Hardware

F. P. Vidal [a], M. Garnier [b], N. Freud [c], J. M. Létang [c], and N. W. John [d]

[a] University of California, San Diego, CA
[b] BRGM, Orléans, France
[c] INSA - Lyon, France
[d] Bangor University, UK

We propose a deterministic simulation of X-ray transmission imaging on graphics hardware. Only the directly transmitted photons are simulated, using the Beer-Lambert law. Our previous attempt to simulate X-ray attenuation from polygon meshes utilising the GPU showed significant increase of performance, with respect to a validated software implementation, without loss of accuracy. However, the simulations were restricted to monochromatic X-rays and finite point sources. We present here an extension to our method to perform physically more realistic simulations by taking into account polychromatic X-rays and focal spots causing blur.

X-ray Attenuation

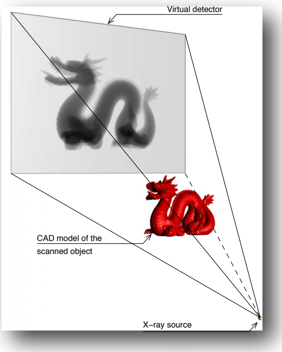
The Beer-Lambert law relates the absorption of light to the properties of the material through which the light is travelling. For a polychromatic incident X-ray beam, it is:

$$N_{out}(E) = N_{in}(E) \exp\left(-\sum_{i=0}^{i<objs} \mu(E,i) L_p(i)\right)$$

with $N_{in}(E)$ the number of incident photons at energy E, $N_{out}(E)$ the number of transmitted photons at energy E. $objs$ is the total number of objects in the 3D scene. μ is the linear attenuation coefficient (in cm^{-1}), which depends on: i) E, the energy of incident photons, and ii) the material properties of the object. $L_p(i)$ is the path length (in cm) of the ray in the i^{th} object.

X-ray Transmission Imaging

Given a virtual detector, an X-ray source and a scanned object, the Beer-Lambert law is computed for every pixel of the detector.

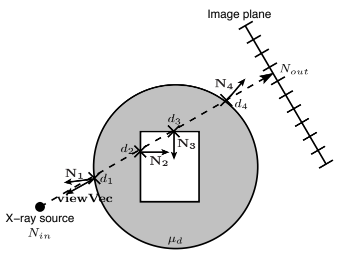


L-Buffer

Polygon meshes are used to represent scanned objects. Normal vectors at the surface of objects are outward. In this example, the ray penetrates into the disk when the dot product between **viewVec** and $\mathbf{N_1}$, the normal at the intersection point, is positive. Conversely, the ray leaves an object if the dot product between **viewVec** and $\mathbf{N_2}$ is negative. The path length of the ray in a given object can be written as follows:

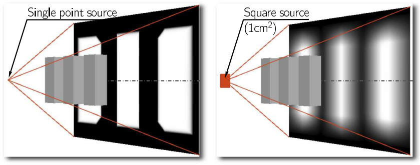
$$L_p = \sum_i -\mathrm{sgn}(\mathbf{viewVec} \cdot \mathbf{N_i}) d_i$$

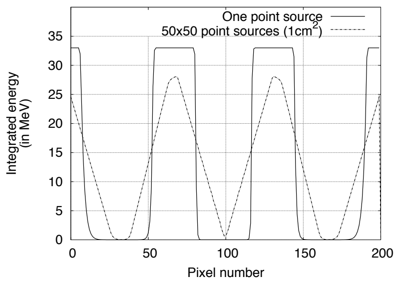
where i refers to the i^{th} intersection found in an arbitrary order, d_i is the distance from the X-ray source to the intersection point of the ray with the triangle, $\mathrm{sgn}(\mathbf{viewVec} \cdot \mathbf{N_i})$ stands for the sign of the dot product between **viewVec** and $\mathbf{N_i}$. This dot product and d_i must be computed for each intersection point. These operations can be efficiently achieved on the GPU using a fragment program.



Geometric Unsharpness

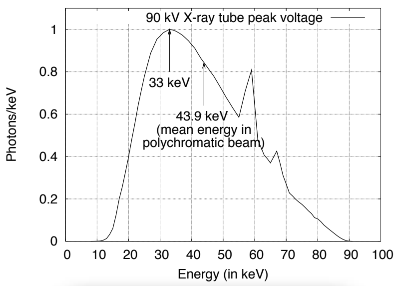
The shape of the source is modelled using a variable number of point sources. Each point is assigned a fraction of the total number of photons in the system.

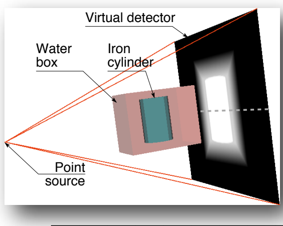


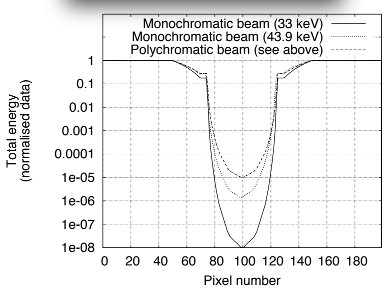


Polychromatism

The incident beam is split into discrete energy channels. To produce the final image, the total amount of energy received by each pixel is computed.

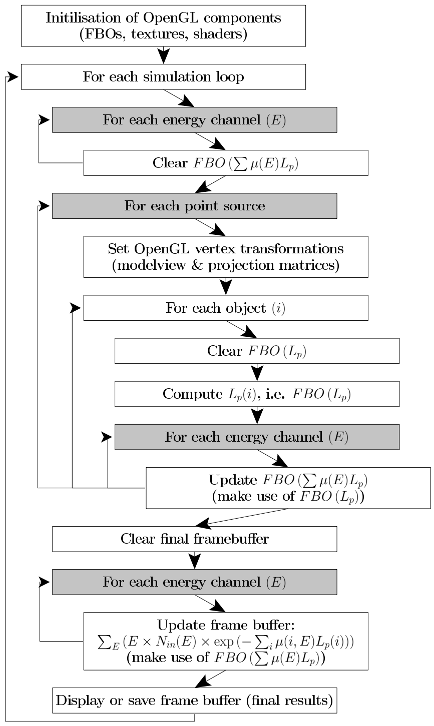






Simulation Pipeline

The algorithm has been implemented using GLSL. The principle of computing direct images is to emit rays at different energies depending on the beam from each of the X-ray point source to every pixel of the detector. For each ray, the total path length through each object is determined using geometrical computations. Finally, the total energy received by a given pixel is updated using the recorded path lengths and X-ray attenuation coefficients at the given energy of the ray.



Results and Conclusion

X-ray transmission images can be fully simulated on the GPU, by using the Beer-Lambert law with polychromatic X-rays and taking into account the shape of the source. Additional loops have been added to the simulation pipeline and the computation cost proportionally increases depending on the number of source points and energy channels. This is a useful development to improve the level of realism in simulations, when both speed and accuracy have to be retained.

References

- FREUD N., DUVAUCHELLE P., LÉTANG J. M., BABOT D.: Fast and robust ray casting algorithms for virtual X-ray imaging. *Nucl Instrum Methods Phys Res B 248*, 1 (2006), 175–180.
- VIDAL F. P., GARNIER M., FREUD N., LÉTANG J. M., JOHN N. W.: Simulation of X-ray attenuation on the GPU. In *Proc Theory Pract Comput Graph* (2009), pp. 25–32.
- VILLARD P. F., VIDAL F. P., HUNT C., BELLO F., JOHN N. W., JOHNSON S., GOULD D. A.: Simulation of per- cutaneous transhepatic cholangiography training simulator with real-time breathing motion. *Int J Comput Assist Radiol Surg 4*, 9 (2009), 571–578.

Contact: franck.p.vidal@gmail.com