Notes on Ellipsoids/ NSXTOOL

Laurent Chapon, Eric Pellegrini

Institut Laue-Langevin, France

Oct 2013

Homogeneous coordinates

For simplicity, we should use homogeneous coordinates to describe points and vectors. A point X in $\mathbf{R}^3$ is described by the row vector X^t:

$$X^t = (x, y, z, 1) \tag{1}$$

A vector is described by the row vector:

$$V^t = (x, y, z, 0) \tag{2}$$

Affine transformations

Rotation, scaling and translations are given by the matrices:

$$R = \begin{pmatrix} r_{00} & r_{01} & r_{02} & 0 \\ r_{10} & r_{11} & r_{12} & 0 \\ r_{20} & r_{21} & r_{22} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, R^{-1} = \begin{pmatrix} r_{00} & r_{10} & r_{20} & 0 \\ r_{01} & r_{11} & r_{21} & 0 \\ r_{02} & r_{12} & r_{22} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{3}$$

$$T = \begin{pmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{pmatrix}, T^{-1} = \begin{pmatrix} 1 & 0 & 0 & -t_x \\ 0 & 1 & 0 & -t_y \\ 0 & 0 & 1 & -t_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{4}$$

$$S = \begin{pmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, S^{-1} = \begin{pmatrix} \frac{1}{s_x} & 0 & 0 & 0 \\ 0 & \frac{1}{s_y} & 0 & 0 \\ 0 & 0 & \frac{1}{s_z} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{5}$$

Ellipsoid

An ellipsoid can be built from the canonical matrix form of the unit-sphere described by the matrix U:

$$U = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix}, X^t.U.X = 0 \tag{6}$$

A point X on the unit-sphere is transformed to a point on a general ellipsoid by first scaling, then rotating the axes and finally translating:

$$X \longrightarrow T.R.S.X \tag{7}$$

Ellipsoid

Since X contra-vary, the equation of the unit sphere is transformed to the equation of a general ellipse in the following way:

$$U \longrightarrow ((T.R.S)^{-1})^t.U.(T.R.S)^{-1} \quad (8)$$

$$U = ((T.R)^{-1})^t(S^{-1}.U.S^{-1}).(T.R)^{-1} \quad (9)$$

We decompose this equation into the canonical form of the ellipsoid equation E=$S^{-1}.U.S^{-1}$, and the affine transform M_E=T.R. The general equation for an ellipsoid becomes:

$$X^t.(M_E^{-1})^t.E.M_E^{-1}.X = 0, E = \begin{pmatrix} \frac{1}{a^2} & 0 & 0 & 0 \\ 0 & \frac{1}{b^2} & 0 & 0 \\ 0 & 0 & \frac{1}{c^2} & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix} \quad (10)$$

Ellipsoid matrix representation from ellipsoid center, semi-axes and eigenvectors

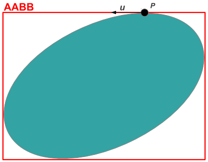
Given the ellipsoid center c=(x_c,y_c,z_c) and unit-vectors
v_0=(v_{0x},v_{0y},v_{0z}),
v_1=(v_{1x},v_{1y},v_{1z}),
v_2=(v_{2x},v_{2y},v_{2z})
associated with the eigenvalues a,b,c, one construct the M_E^{-1} matrix as follows:

$$M_E^{-1} = R^{-1}.T^{-1} = \begin{pmatrix} v_{0x} & v_{0y} & v_{0z} & -x_c \\ v_{1x} & v_{1y} & v_{1z} & -y_c \\ v_{2x} & v_{2y} & v_{2z} & -z_c \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{11}$$

Point inside or outside ellipsoid

To find if a point $P^t = (x, y, z, 1)$ is inside the ellipsoid E, defined by the matrix representation $E' = (M_E^{-1})^t.E.M_E^{-1}$, one just need to check that $P^t.E'P \leq 0$. Since E' is a symmetric matrix, and only multiplications and additions are involved, this is computed very fast. Using $boost :: ublas :: symmetric\_matrix$ in release mode, one can do $2.5.10^7$ such tests/seconds on a 2.7 Ghz intel core i7. It is easy to multi-thread for many Ellipsoids.

Computing AABB of Ellipsoids



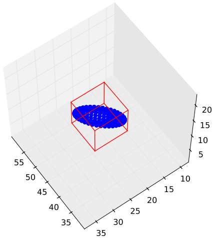
To compute the Axis-Aligned Bounding Box of an ellipsoid, we search the coordinates of the extreme points (such as P), at which the tangents vectors (u) are aligned with the box. The idea is to transform such vectors back to the unit-sphere ($u \longrightarrow u'$), find the corresponding point (P') which is then transformed back to the ellipsoid ($P' \longrightarrow P$)

Computing AABB of Ellipsoids(2)

The half-width of the boxes can be directly computed from the matrix of homogeneous transformation $H = T.R.S$ by calculating the norm of each row of H_{3x3}:

$$\begin{aligned} hwidth &= \sqrt{H_{00}^2 + H_{01}^2 + H_{02}^2} \\ hheight &= \sqrt{H_{10}^2 + H_{11}^2 + H_{12}^2} \\ hdepth &= \sqrt{H_{20}^2 + H_{21}^2 + H_{22}^2} \end{aligned} \quad (12)$$

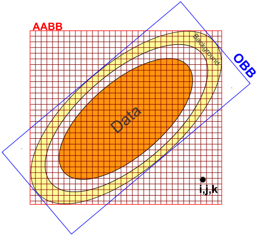
Test of equations in python:



Interest of AABB

- Collision detection with other ellipsoids is extremely fast.
- Since the box is aligned with the axes of the world, it is easy to extract integration on a line, to be used for weak peaks and routines checks.

Integration using AABB, OBB



Integration should run as follows:

1. For each point $P^t = (i, j, k, 1)$ of our AABB, check if P is inside OBB. This is done by transforming P back to the unit-sphere (calculate $P' = (TRS)^{-1}.P$ and check $\forall i, -1 \leq P'_i \leq 1$). This is 4-time faster than testing if P inside Ellipsoid
2. If P inside OBB, check if inside Ellipsoid.
3. Since Ellipsoids are convex, max. two intersections with a line, so this process can stop as soon as

Integrating data in Ellipsoid shells

I think we can play clever tricks here with successive shells. A point $P^t = (x, y, z, 1)$ is inside an Ellipsoid E if $P^t.E'.P \leq 0$. For a second ellipsoid with same center and unit-vectors, but with semi-axes scaled by a factor α ,
$P^t.E''.P = \frac{P^t.E'.P+1}{\alpha^2} - 1$..... need to check this!