Wild Magic: The Camera Model

David Eberly
Magic Software, Inc.
`http://www.magic-software.com`
January 16, 2002

1 Camera Parameters

The camera has a coordinate system associated with it. The *eye point* $\vec{E}$ is the origin of the coordinate system. The *direction vector* is $\vec{D}$, the *up vector* is $\vec{U}$, and the *left vector* is $\vec{L}$. These vectors are unit length, mutually orthogonal, and form a right–handed system. If the vectors are written as columns in a matrix $R = [\vec{L}\,\vec{U}\,\vec{D}]$, the matrix is a rotation matrix. A consequence of being a right–handed system is that $\vec{L} \times \vec{U} = \vec{D}$, $\vec{U} \times \vec{D} = \vec{L}$, and $\vec{D} \times \vec{L} = \vec{U}$. Any point $\vec{X}$ can be written in this coordinate system in the form $\vec{X} = \vec{E} + R\vec{Y}$, so the *camera coordinates of* $\vec{X}$ are $\vec{Y} = R^{\mathrm{T}}(\vec{X} - \vec{E})$.

It is important to understand the relationships of the coordinate axis directions. At the eye point, if you look in the camera direction $\vec{D}$ and are standing so that your up direction is $\vec{U}$, the vector $\vec{L}$ really does point to your left. Figure 1 illustrates this. The reason for understanding this is that various discussions

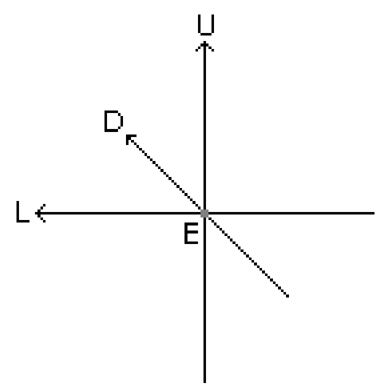

Figure 1: The camera coordinate system.

about a camera model use a left–handed coordinate system where you use a *right vector* $\vec{R}$ rather than a left vector.

The eye point may only see objects that are within a bounded region of space called the *view frustum*. The frustum is limited by six planes. Along the direction of view, the two planes are called the *near plane* and the *far plane*. These planes are selected to be perpendicular to $\vec{D}$. The near plane is $n > 0$ units of distance from the eye point, called the *near distance*. The plane has unit length normal $\vec{D}$. The point on the plane closest to the eye point is $\vec{E} + n\vec{D}$, so the equation for the plane is $\vec{D} \cdot (\vec{X} - (\vec{E} + n\vec{D})) = 0$, or

$$\vec{D} \cdot \vec{X} = \vec{D} \cdot \vec{E} + n. \tag{1}$$

The far plane is f units of distance from the eye point, called the *far distance*. The plane has unit length normal $-\vec{D}$, chosen to point to the interior of the frustum. The point on the plane closest to the eye point is $\vec{E} + f\vec{D}$, so the equation for the plane is $-\vec{D} \cdot (\vec{X} - (\vec{E} + f\vec{D}) = 0$, or

$$-\vec{D} \cdot \vec{X} = -(\vec{D} \cdot \vec{E} + f). \tag{2}$$

The near plane will be used for perspective projection of objects in 3D. If $\vec{X}$ is a point in 3D, its projection is obtained by computing the intersection of the ray $\vec{P}(s) = \vec{E} + s(\vec{X} - \vec{E})$ and the near plane. Replacing $\vec{P}(s)$ in equation (1) and solving leads to $s = n/\vec{D} \cdot (\vec{X} - \vec{E})$. To be on the ray, it is necessary that $\vec{D} \cdot (\vec{X} - \vec{E}) > 0$. Geometrically this says that $\vec{X}$ is in front of the eye point. The limitations by the near and far plane allow us to project only those points between the two, so the component of $\vec{X} - \vec{E}$ in the $\vec{D}$ direction must be in the interval $[n, f]$. That component is $\delta = \vec{D} \cdot (\vec{X} - \vec{E})$.

The other four limiting planes for the frustum are defined by selecting a *viewport* in the near plane, a rectangular region whose sides are parallel to the left vector $\vec{L}$ and the up vector $\vec{U}$. In the $\vec{L}$ direction, the limiting values are ℓ (left) and r (right) with $\ell < r$. In the $\vec{U}$ direction, the limiting values are b (bottom) and t (top) with $b < t$. The four corners of the viewport are

$$\vec{E} + n\vec{D} + \ell\vec{L} + b\vec{U}, \;\; \vec{E} + n\vec{D} + \ell\vec{L} + t\vec{U}, \;\; \vec{E} + n\vec{D} + r\vec{L} + b\vec{U}, \;\; \vec{E} + n\vec{D} + r\vec{L} + t\vec{U}. \tag{3}$$

The view frustum is said to be an *orthogonal frustum* if $r = -\ell$ and $t = -b$. The eye point $\vec{E}$ and the left edge of the viewport are contained by a plane called the *left plane*. Similarly there is a *right plane*, a *bottom plane*, and a *top plane*. These four side planes intersect the far plane to form another rectangle. The corners of that rectangle are

$$\vec{E} + \frac{f}{n}(n\vec{D} + \ell\vec{L} + b\vec{U}), \;\; \vec{E} + \frac{f}{n}(n\vec{D} + \ell\vec{L} + t\vec{U}, \;\; \vec{E} + \frac{f}{n}(n\vec{D} + r\vec{L} + b\vec{U}), \;\; \vec{E} + \frac{f}{n}(n\vec{D} + r\vec{L} + t\vec{U}). \tag{4}$$

An equation of the left plane may be constructed. It is important to remember that ℓ and r are measurements with respect to $\vec{L}$, **not** with respect to a right vector. Figure 2 shows a view of the frustum with $\vec{L}$ pointing to the left. The left plane shows up on your right and the right plane shows up on your left. Confusing, no doubt, if you are used to the standard presentation of the left–handed camera system where the left plane is on the left and the right plane is on the right. The problem is that the concepts of "left" and "right" are relative to a coordinate system, just as "up" and "down" are. A way to remember *where* the left and right planes are is to think in terms of the scalar values that specify them. Regardless of the camera model, given a specified vector $\vec{V}$ with two scalars $v_{\min}$ and $v_{\max}$ with $v_{\min} < v_{\max}$, the left plane goes with the minimum $v_{\min}$ and the right plane goes with the maximum $v_{\max}$. The same idea applies even in the up direction $\vec{U}$. The top plane goes with the maximum scalar, t, and the bottom plane goes with the minimum scalar, b.

The normals for the side planes should point to the inside of the frustum. The left plane contains two nonparallel rays, $\vec{E} + s\vec{V}_{\ell b}$ and $\vec{E} + s\vec{V}_{\ell t}$ where $\vec{V}_{\ell b} = n\vec{D} + \ell\vec{L} + b\vec{U}$ and $\vec{V}_{\ell t} = n\vec{D} + \ell\vec{L} + t\vec{U}$. The vectors $\vec{V}_{\ell b}$ and $\vec{V}_{\ell t}$ lie in the left plane, so their cross product is perpendicular to the plane. The order in the cross

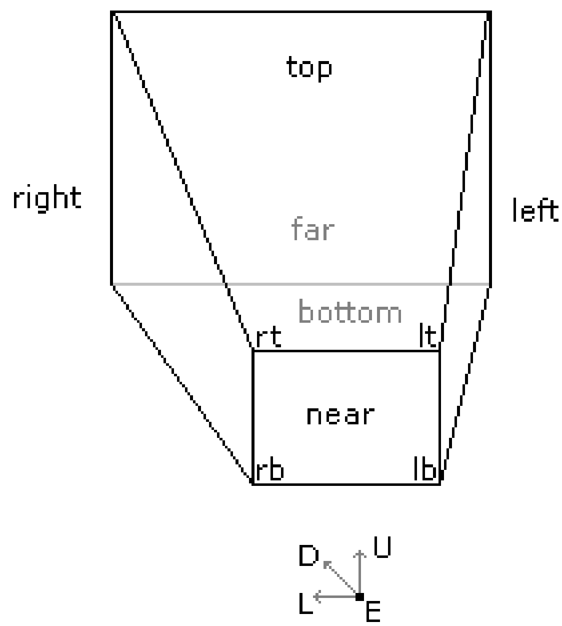

Figure 2: The view frustum.

product is chosen so that the resulting vector points inside the frustum:

$$\begin{aligned}
\vec{V}_{\ell t} \times \vec{V}_{\ell b} &= ((n\vec{D} + \ell\vec{L}) + t\vec{U}) \times ((n\vec{D} + \ell\vec{L}) + b\vec{U}) \\
&= (n\vec{D} + \ell\vec{L}) \times b\vec{U} + t\vec{U} \times (n\vec{D} + \ell\vec{L}) \\
&= (t - b)\vec{U} \times (n\vec{D} + \ell\vec{L}) \\
&= (t - b)(n\vec{U} \times \vec{D} + \ell\vec{U} \times \vec{L}) \\
&= (t - b)(n\vec{L} - \ell\vec{D}).
\end{aligned}$$

An inner–pointing, unit–length normal for the left plane is

$$\vec{N}_\ell = \frac{\vec{V}_{\ell t} \times \vec{V}_{\ell b}}{|\vec{V}_{\ell t} \times \vec{V}_{\ell b}|} = \frac{n\vec{L} - \ell\vec{D}}{\sqrt{n^2 + \ell^2}}.$$

The point $\vec{E}$ is on the left plane, so the equation for the left plane is $\vec{N}_\ell \cdot (\vec{X} - \vec{E}) = 0$, or

$$\vec{N}_\ell \cdot \vec{X} = \vec{N}_\ell \cdot \vec{E}. \tag{5}$$

Similar constructions apply for the other side planes. Define $\vec{V}_{rb} = n\vec{D} + r\vec{L} + b\vec{U}$ and $\vec{V}_{rt} = n\vec{D} + r\vec{L} + t\vec{U}$.

An inner–pointing, unit–length normal for the right plane is

$$\vec{N}_r = \frac{\vec{V}_{rb} \times \vec{V}_{rt}}{|\vec{V}_{rb} \times \vec{V}_{rt}|} = \frac{-n\vec{L} + r\vec{D}}{\sqrt{n^2 + r^2}}.$$

The equation for the right plane is $\vec{N}_r \cdot (\vec{X} - \vec{E}) = 0$, or

$$\vec{N}_r \cdot \vec{X} = \vec{N}_r \cdot \vec{E}. \tag{6}$$

An inner–pointing, unit–length normal for the bottom plane is

$$\vec{N}_b = \frac{\vec{V}_{\ell b} \times \vec{V}_{rb}}{|\vec{V}_{\ell b} \times \vec{V}_{rb}|} = \frac{n\vec{U} - b\vec{D}}{\sqrt{n^2 + b^2}}.$$

The equation for the bottom plane is $\vec{N}_r \cdot (\vec{X} - \vec{E}) = 0$, or

$$\vec{N}_r \cdot \vec{X} = \vec{N}_r \cdot \vec{E}. \tag{7}$$

An inner–pointing, unit–length normal for the top plane is

$$\vec{N}_t = \frac{\vec{V}_{rt} \times \vec{V}_{\ell t}}{|\vec{V}_{rt} \times \vec{V}_{\ell t}|} = \frac{-n\vec{U} + t\vec{D}}{\sqrt{n^2 + t^2}}.$$

The equation for the top plane is $\vec{N}_t \cdot (\vec{X} - \vec{E}) = 0$, or

$$\vec{N}_t \cdot \vec{X} = \vec{N}_t \cdot \vec{E}. \tag{8}$$

In the `Camera` class, `m_afCoeffL[]` stores the coefficients of $\vec{L}$ and $\vec{D}$ in $\vec{N}_\ell$, `m_afCoeffR[]` stores the coefficients of $\vec{L}$ and $\vec{D}$ in $\vec{N}_r$, `m_afCoeffB[]` stores the coefficients of $\vec{U}$ and $\vec{D}$ in $\vec{N}_b$, and `m_afCoeffT[]` stores the coefficients of $\vec{U}$ and $\vec{D}$ in $\vec{N}_t$. These values are set in the callback `OnFrustumChange` since they only need to be updated when the frustum changes, not when the camera coordinate axes change. The world plane equations (1), (2), (5), (6), (7), and (8) are computed in the callback `OnFrameChange`. This callback is executed whenever $\vec{L}$, $\vec{U}$, and $\vec{D}$ are changed (*i.e.* rotated).