LookAt Function

The function

\[
\text{LookAt}(\text{eye}, \text{look}, \text{up})
\]

in mat.h calculates the matrix \( V \) which transforms the vertices from the World Coordinate System to the Camera (Eye) Coordinate System, where

- \( \text{eye} = \) location of the camera
- \( \text{look} = \) the point the camera is looking at
- \( \text{up} = \) the VUP vector

Move to origin

\[
T(t_x, t_y, t_z) = \begin{pmatrix}
1 & 0 & 0 & t_x \\
0 & 1 & 0 & t_y \\
0 & 0 & 1 & t_z \\
0 & 0 & 0 & 1
\end{pmatrix}
\]

Orient axes:

\[
\hat{n} = \frac{\text{look} - \text{eye}}{\text{look} - \text{eye}}
\]

\[
\hat{u} = \frac{\hat{n} \times \text{up}}{\hat{n} \times \text{up}} 
\Rightarrow \quad R^{-1} = \begin{pmatrix}
-u_x & u_y & u_z & 0 \\
v_x & v_y & v_z & 0 \\
-n_x & -n_y & -n_z & 0 \\
0 & 0 & 0 & 1
\end{pmatrix}
\]

The \( V \) matrix can be calculated as \( R^{-1} T^{-1} \):

\[
\text{mat4 LookAt( const vec4& eye, const vec4& at, const vec4& up ) }
\]

\{
    \text{vec4 n = normalize(eye - at);}
    \text{vec4 u = vec4(normalize(cross(up,n)),0);}
    \text{vec4 v = vec4(normalize(cross(n,u)),0);}
    \text{vec4 t = vec4(0.0, 0.0, 0.0, 1.0);}
    \text{mat4 c = mat4(u, v, n, t);}
    \text{return c * Translate( -eye );}
\}