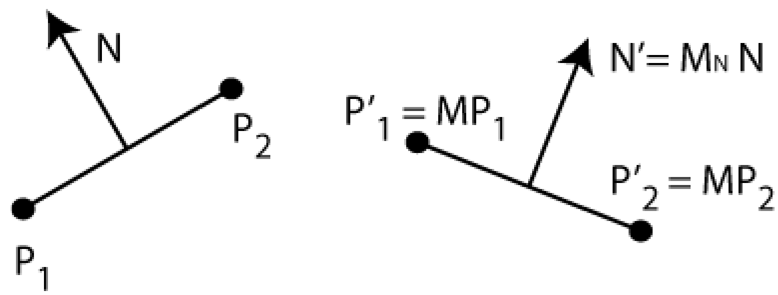
Transforming Normals

Let P_1 and P_2 be points on a surface.
Let M be a matrix which transforms P_1 and P_2 in some way.
Let N be the normal to the surface.

Question: How does N transform? That is, what is M_N?



We know that since the normal must always be perpendicular to the surface that:

$$(P_2 - P_1) \bullet N = (P'_2 - P'_1) \bullet N' = 0$$

We can write this in matrix (rather than vector) form:

$$\begin{aligned}(P_2 - P_1)^T\ N &= (P'_2 - P'_1)^T\ N' \\ &= (M\,P_2 - M\,P_1)^T\ M_N\,N \\ &= (\,M\,(\,P_2 - P_1)\,)^T\ M_N\,N \\ &= (\,P_2 - P_1\,)^T\ M^T\ M_N\,N\end{aligned}$$

Comparing the two sides, we see that we must have

$$M^T\ M_N = \text{Identity}$$

Or

$$M_N = (M^T)^{-1}$$

If M is a rotation, we know that $M^{-1} = M^T$ so that
$M_N = (M^T)^{-1} = (M^{-1})^{-1} = M$

Thus, for rotations, Normals transform the same as points.

Suppose M is a uniform scale, e.g. M = Scale(5) = M^T. Then M^{-1} = Scale (1/5) and we have
$M_N = (M^T)^{-1}$ = Inverse(Scale(5)) = Scale (1/5).
However, we always normalize Normals so that their length is 1. If we take a normalized vector, scale it *uniformly*, then normalize it again, we end up with the same vector back.
Thus, if M is a uniform scale, we are ok setting M_N = M.

However, if M is not a uniform scale, then $M_N = M^{-1} \neq M$. In such a case, we should maintain a variable (equivalent to mv) which transforms the normal.

A Few Exceptions:

- Suppose we have a cube where the normals are parallel to the coordinate axes (e.g. (1,0,0)), then we also are fine using a non-uniform scale as long as we apply it to the cube while it is aligned with the axes (if we first rotate it at some odd angle, then we will have problems).
- Similarly, we can get away with non-uniformly scaling a cylinder under certain situations. Assume the cylinder is aligned with the y axis. Then as long as we scale the x and z axes the same (before rotating), we are also fine.

Do you understand why these exceptions hold?