Lecture 12: Coordinate Free Formulas for Affine and Projective Transformations

be ye transformed by the renewing of your mind Romans 12:2

1. Transformations for 3-Dimensional Computer Graphics

Computer Graphics in 2-dimensions is grounded on simple transformations of the plane. Turtle Graphics and LOGO are based on conformal transformations; Affine Graphics and CODO on affine transformations. The turtle uses translation, rotation, and uniform scaling to navigate on the plane, and traces of the turtle's path generate a wide variety of shapes, ranging from simple polygons to complex fractals. Affine Graphics uses affine transformations to position, orient, and scale objects as well as to model shape. The vertices of simple figures like polygons and stars are typically positioned using translation and rotation; ellipses are generated by scaling circles non-uniformly along preferred directions; fractals are built using iterated function systems.

Computer Graphics in 3-dimensions is grounded in transformations of 3-space. But in 3-dimensions Computer Graphics employs not only nonsingular conformal and affine transformations, but also affine projections and projective transformations. Projections play a key role in 3-dimensional Computer Graphics because we need to display 3-dimensional geometry on a 2-dimensional screen. Orthogonal projections of 3-dimensional models from several different directions are traditional in many engineering applications. Perspective projection plays a central role in Computer Graphics because, as in classical art, we often want to display 3-dimensional shapes in perspective just as they appear to the human eye. Visual realism requires projective transformations.

Here we are going to study 3-dimensional affine and projective transformation, using the coordinate free techniques of vector geometry.

2. Affine and Projective Transformations

An affine transformation is a transformation that maps points to points, maps vectors to vectors, and preserves linear and affine combinations. In particular, A is an affine transformation if

$$\begin{aligned} &A(u+v) = A(u) + A(v) \\ &A(cv) = c\,A(v) \\ &A(P+v) = A(P) + A(v) \end{aligned} \tag{1}$$

As a consequence of Equation (1), non-degenerate affine transformations in 3-dimensions map lines to lines, triangles to triangles, and parallelograms to parallelograms just as in 2-dimensions. The most important affine transformations in 3-dimensional Computer Graphics are the rigid motions (translation, rotation, and mirror image), scaling (both uniform and non-uniform), and orthogonal projection.

Perspective projection is not an affine transformation. Indeed, as we shall see shortly, perspective projection is not defined at every point nor is perspective well-defined for any vector. Nevertheless, in Section 5.2 we will derive a simple, straightforward formula for computing perspective projection on any point where perspective is well-defined.

In this lecture we are going to derive explicit formulas for each of the most important affine and projective transformations in 3-dimensional Computer Graphics -- translation, rotation, mirror image, uniform and non-uniform scaling, and orthogonal and perspective projection -- using coordinate free vector algebra. In the next lecture, we will apply these formulas to construct matrices to represent each of these transformations.

Before we proceed, we need to recall the following results. In most of our derivations we shall be required to decompose an arbitrary vector v into components $v_{||}$ and $v_{\perp}$ parallel and perpendicular to a fixed unit vector u (see Figure 1). Recall from Lecture 9 that

$$
\begin{aligned}
v &= v_{||} + v_{\perp} \\
v_{||} &= (v \bullet u)u \qquad (2) \\
v_{\perp} &= v - (v \bullet u)u
\end{aligned}
$$

Be sure that you are comfortable with these formulas before you continue because we shall use these equations again and again throughout this lecture.

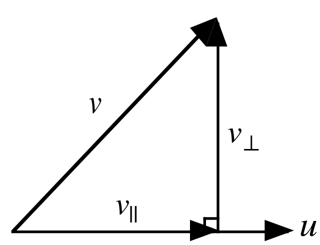

Figure 1: Decomposing a vector v into components parallel and perpendicular to a fixed vector u.

3. Rigid Motions

Rigid motions are transformations that preserve lengths and angles. The rigid motions in 3-dimensions are composites of three basic transformations: translation, rotation, and mirror image.

3.1 Translation. Translation is defined by specifying a distance and a direction. Since a distance and direction designate a vector, a translation is defined by specifying a translation vector. The formulas for translation in 3-dimensions are similar to the formulas for translation in 2-dimensions: points are affected by translation, vector are unaffected by translation (see Figure 2). Let w be a translation vector. If P is a point and v is a vector, then

$$\begin{aligned} v^{new} &= v \\ P^{new} &= P + w \end{aligned} \tag{3}$$

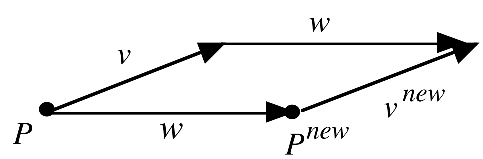

Figure 2: Translation. Points are affected by translation; vectors are not affected by translation.

3.2 Rotation. Rotation in 3-dimensions is defined by specifying an angle and an axis of rotation. The axis line L is typically described by specifying a point Q on L and a unit direction vector u parallel to L. We shall denote the angle of rotation by θ (see Figure 3).

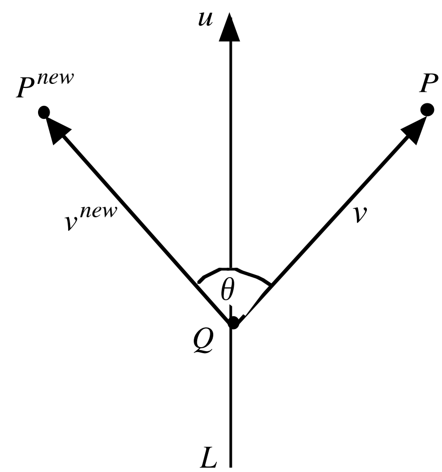

Figure 3: Rotation around the axis line L through the angle θ.

The *Formula of Rodrigues* computes rotation around the axis L through the angle θ by setting

$$\begin{aligned} v^{new} &= (\cos\theta)v + (1-\cos\theta)(v \bullet u)u + (\sin\theta)u \times v \\ P^{new} &= Q + (\cos\theta)(P-Q) + (1-\cos\theta)\big((P-Q)\bullet u\big)u + (\sin\theta)u \times (P-Q) \end{aligned} \tag{4}$$

Notice that since

$$P = Q + (P-Q)$$

it follows that

$$P^{new} = Q^{new} + (P-Q)^{new} = Q + (P-Q)^{new} .$$

so the rotation formula for points follows immediately from the rotation formula for vectors. Therefore it is enough to prove Equation (4) for vectors.

To derive the *Formula of Rodrigues* for vectors, split v into two components: the component $v_{||}$ parallel to u and the component $v_{\perp}$ perpendicular to u (see Figure 4, left). Then

$$v = v_{||} + v_{\perp}$$

so by linearity

$$v^{new} = v_{||}^{new} + v_{\perp}^{new} .$$

Since $v_{||}$ is parallel to the axis of rotation,

$$v_{||}^{new} = v_{||} .$$

Hence we need only compute the effect of rotation on $v_{\perp}$. Now $v_{\perp}$ lies in the plane perpendicular to u; moreover, $u \times v_{\perp}$ is perpendicular to $v_{\perp}$ and lies in the plane perpendicular u (see Figure 4, right). Therefore using our standard approach to rotation in the plane, we find that

$$v_{\perp}^{new} = (\cos\theta)v_{\perp} + (\sin\theta)u \times v_{\perp} .$$

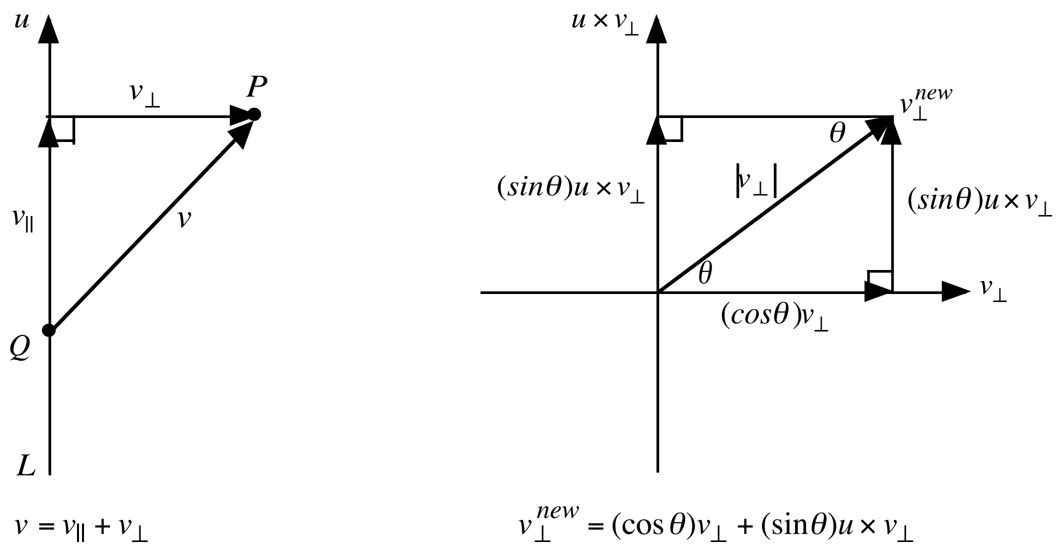

Figure 4: Decomposing v into components parallel and perpendicular to the axis of rotation (left), and rotation in the plane perpendicular to the axis of rotation (right).

Now we have

$$v^{new} = v_{||}^{new} + v_{\perp}^{new}$$

$$v_{||}^{new} = v_{||}$$

$$v_{\perp}^{new} = (\cos\theta)v_{\perp} + (\sin\theta)u \times v_{\perp} .$$

Therefore,

$$v^{new} = v_{||} + (\cos\theta)v_{\perp} + (\sin\theta)u \times v_{\perp} . \qquad (*)$$

But since u is a unit vector,

$$v_{||} = (v \bullet u)u$$

$$v_{\perp} = v - v_{||} = v - (v \bullet u)u \ .$$

Substituting these two identities into (*) yields

$$v^{new} = (v \bullet u)u + (\cos\theta)\big(v - (v \bullet u)u\big) + (\sin\theta)u \times \big(v - (v \bullet u)u\big)$$

Thus, since cross product distributes through addition and $u \times u = 0$,

$$v^{new} = (\cos\theta)v + (1 - \cos\theta)(v \bullet u)u + (\sin\theta)u \times v ,$$

which is the *Formula of Rodrigues* for vectors.

3.3 Mirror Image. A mirror in 3-dimensions is a plane S, typically specified by a point Q on S and a unit vector n normal to S. (see Figure 5, left).

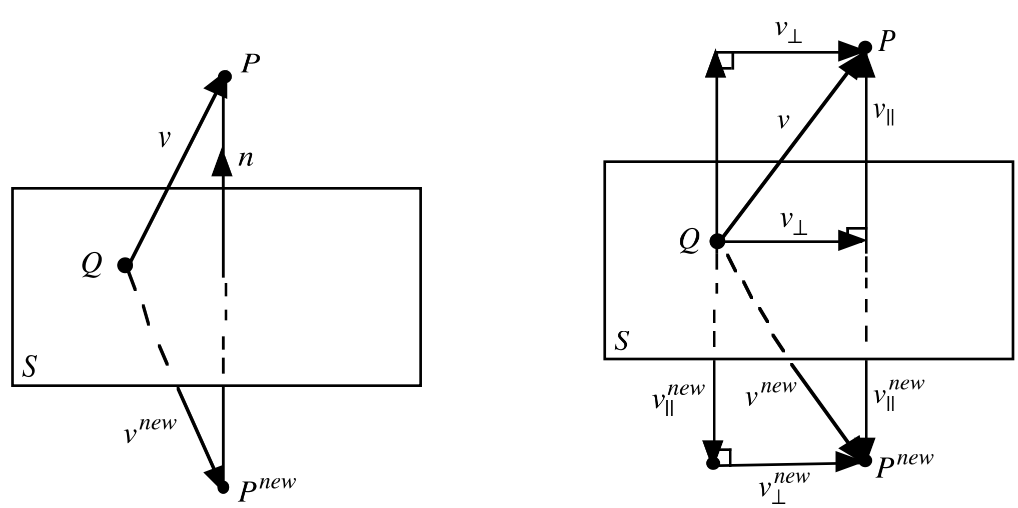

Figure 5: A mirror plane S specified by a point Q and a normal vector n *(left)*, and the decomposition of a vector into components parallel and perpendicular to the normal vector (right).

If P is a point and v is a vector, then the formulas for mirror image are

$$\begin{aligned} v^{new} &= v - 2(v \bullet n)n \\ P^{new} &= P + 2\big((P - Q) \bullet n\big)n \end{aligned} \qquad (5)$$

Notice that since

$$P = Q + (P - Q)$$

it follows that

$$P^{new} = Q^{new} + (P - Q)^{new} = Q + (P - Q)^{new} \ .$$

so the formula for mirror image for points follows immediately from the formula for mirror image for vectors. Therefore once again it is enough to prove the mirror image formula for vectors.

To derive the mirror image formula for vectors, we again split v into two components: the component $v_{||}$ parallel to n and the component $v_{\perp}$ perpendicular to n (see Figure 5, right). Then

$$v = v_{\perp} + v_{||}$$

so by linearity

$$v^{new} = v_{\perp}^{new} + v_{||}^{new}$$

Since $v_{\perp}$ lies in the plane S,

$$v_{\perp}^{new} = v_{\perp} \ .$$

Moreover, since $v_{||}$ is parallel to the normal vector n,

$$v_{||}^{new} = -v_{||}.$$

Therefore,

$$v^{new} = v_{\perp}^{new} + v_{||}^{new} = v_{\perp} - v_{||}. \qquad (**)$$

But

$$v_{\perp} = v - v_{||}$$

$$v_{||} = (v \bullet w)w \ .$$

Substituting these identities into (**) yields

$$v^{new} = v - 2(v \bullet n)n \,,$$

which is the mirror image formula for vectors.

4. Scaling

Both uniform and non-uniform scaling are important in 3-dimensional Computer Graphics. Below we treat each of these transformations in turn.

4.1 Uniform Scaling. Uniform scaling is defined by specifying a point Q from which to scale distance and a scale factor s. The formulas for uniform scaling in 3-dimensions are identical to the formulas for uniform scaling in 2-dimensions (see Figure 6): If P is a point and v is a vector, then

$$\begin{aligned} v^{new} &= sv \\ P^{new} &= Q + s(P - Q) = sP + (1-s)Q. \end{aligned} \qquad (6)$$

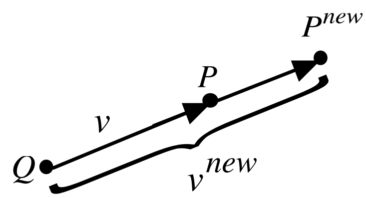

Figure 6: Uniform scaling for points and vectors.

4.2 Non-Uniform Scaling. For non-uniform scaling, we need to designate a unit vector w specifying the scaling direction, as well as a point Q and a scale factor s (see Figure 7, left).

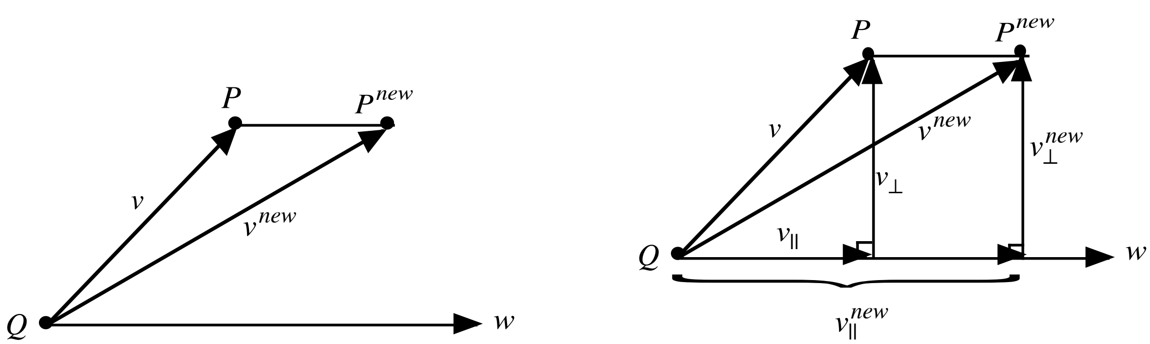

Figure 7: Non-uniform scaling is specified around a point Q in a direction w *(left)*. The decomposition of a vector into components parallel and perpendicular to the direction vector (right).

If P is a point and v is a vector, then the formulas for non-uniform scaling are

$$\begin{aligned} v^{new} &= v + (s-1)(v \bullet w)w \\ P^{new} &= P + (s-1)\big((P-Q) \bullet w\big)w. \end{aligned} \tag{7}$$

As usual non-uniform scaling for points follows easily from non-uniform scaling for vectors, since

$$P = Q + (P - Q)$$

so

$$P^{new} = Q^{new} + (P-Q)^{new} = Q + (P-Q)^{new}.$$

Therefore once again it is enough to prove the non-uniform scaling formula for vectors.

To derive the non-uniform scaling formula for vectors, we again split v into two components: the component $v_{||}$ parallel to w and the component $v_\perp$ perpendicular to w (see Figure 7, right). Then, as usual,

$$v = v_\perp + v_{||} ,$$

so by linearity

$$v^{new} = v_\perp^{new} + v_{||}^{new} .$$

Since $v_\perp$ is perpendicular to the scaling direction w,

$$v_\perp^{new} = v_\perp ,$$

and since $v_{||}$ is parallel to the scaling direction w,

$$v_{||}^{new} = s v_{||}.$$

Therefore,

$$v^{new} = v_\perp^{new} + v_{||}^{new} = v_\perp + s v_{||}. \qquad (***)$$

But once again

$$v_\perp = v - v_{||}$$

$$v_{||} = (v \bullet w) w .$$

Substituting these identities into (***) yields

$$v^{new} = v + (s-1)(v \bullet w) w ,$$

which is the non-uniform scaling formula for vectors. Notice, by the way, that mirror image is just non-uniform scaling along the normal direction to the mirror plane, with scale factor $s = -1$.

5. Projections

To display 3-dimensional geometry on a 2-dimensional screen, we need to project from 3-dimensions to 2-dimensions. The two most important types of projections are orthogonal projection and perspective projection.

5.1 Orthogonal Projection. The simplest projection is orthogonal projection (see Figure 8).

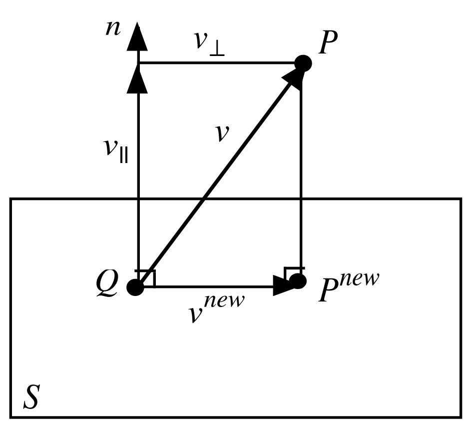

Figure 8: Orthogonal projection.

The screen is represented by a plane S defined by a point Q and a unit normal vector n. If P is a point and v is a vector, then the formulas for orthogonal projection are

$$v^{new} = v - (v \bullet n)n \\ P^{new} = P - \big((P-Q) \bullet n\big)n. \tag{8}$$

These formulas are easy to derive. The formula for vectors is immediate from the observation that v^{new} is just the perpendicular component of v relative to the normal vector n, so

$$v^{new} = v_{\perp} = v - v_{||} = v - (v \bullet n)n.$$

Moreover, for points, we simply observe, as usual, that orthogonal projection for points follows easily from orthogonal projection for vectors. Indeed, since

$$P = Q + (P - Q) \ .$$

it follows that

$$P^{new} = Q^{new} + (P-Q)^{new} = Q + (P-Q)^{new} = P - \big((P-Q) \bullet n\big)n.$$

5.2 Perspective. Perspective projection is required in Computer Graphics to simulate visual realism. Perspective projection is defined by specifying an eye point E and a plane S into which to project the image. The plane S is specified, as usual, by a point Q on S and a unit vector n normal to S. The image of a point P under perspective projection from the eye point E to the plane S is the intersection of the line EP with the plane S (see Figure 9, left). Notice that unlike all of the other transformations that we have developed in this lecture, perspective projection is not an affine transformation because perspective projection is not a well defined transformation at every point nor is perspective well defined for any vector (see Figure 9, right).

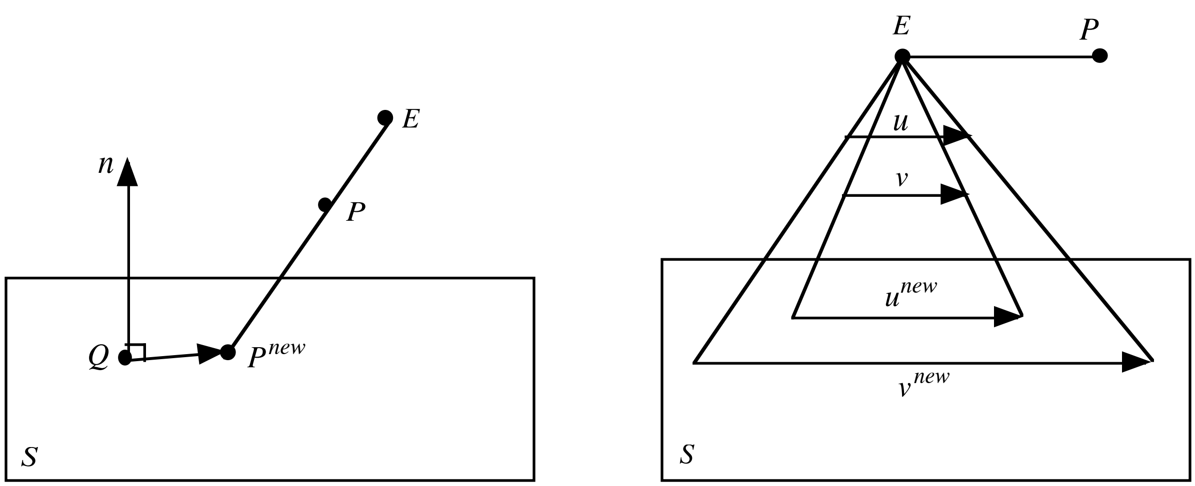

Figure 9: Perspective projection from the eye point E to the point P is defined by the intersection of the line EP with the viewing plane S (left). Notice that perspective projection is not well defined on points P where the line EP is parallel to the plane S nor is perspective defined on any vector, since $u = v$ but $u^{new} \neq v^{new}$ (right).

The image of a point P under perspective projection from the eye point E to the plane S is given by

$$P^{new} = \frac{\big((E-Q)\bullet n\big)P + \big((Q-P)\bullet n\big)E}{(E-P)\bullet n} \,. \tag{9}$$

To derive this formula, we simply compute the intersection of the line EP with the plane S. Since the point P^{new} lies on the line EP, there is a constant c such that

$$P^{new} = P + c(E-P) \,.$$

Moreover, since the point P^{new} also lies on the plane S, the vector $P^{new} - Q$ must be perpendicular to the normal vector n; therefore

$$(P^{new} - Q)\bullet n = 0 \,.$$

Substituting the first equation for P^{new} into the second equation for P^{new} gives

$$(P-Q)\bullet n + c(E-P)\bullet n = 0 \,.$$

Solving for c, we find that

$$c = \frac{(Q-P)\bullet n}{(E-P)\bullet n} \,.$$

Therefore

$$P^{new} = P + c(E-P) = P + \frac{(Q-P)\bullet n}{(E-P)\bullet n}(E-P) = \frac{\big((E-Q)\bullet n\big)P + \big((Q-P)\bullet n\big)E}{(E-P)\bullet n} \,.$$

Notice that if the line EP is parallel to the plane S, then the denominator $(E-P)\bullet n = 0$. In this case the expression for P^{new} is not defined, so perspective projection is not well-defined for any point P for which the line EP is parallel to the plane S. This problem arises because unlike any of our previous transformation, the expression for P^{new} has P in the denominator, so perspective projection is a rational function. We shall have more to say about rational transformations in the next lecture.

6. Summary

Below is a summary of the highlights of this lecture. Also listed below for your convenience are the coordinate free formulas from vector algebra for all the transformations that we have studied in this lecture.

6.1 Affine and Projective Transformations without Matrices. One of the main themes of this lecture is that we do not need matrices to compute the effects of affine and projective transformations on points and vectors. Instead we can use vector algebra to derive coordinate free expressions for all of the important affine and projective transformations in 3-dimensional Computer Graphics. Moreover, except for rotation, these formulas from vector algebra are actually more efficient than the corresponding matrix formulations. Nevertheless, formulas from vector algebra are not convenient for composing transformations, whereas matrix representations are easily composed by matrix multiplication. Therefore, in the next lecture we shall use these formulas from vector algebra to derive matrix representations for each of the corresponding affine and projective transformations.

Our main strategy, used over and over again, for deriving explicit formulas for transformations on vectors is to decompose an arbitrary vector into components parallel and perpendicular to a particular direction vector in which is it especially easy to compute the transformation. We then analyze the transformation on each of these components and sum the transformed components to find the effect of the transformation on the original vector.

This strategy works because most of the transformations that we have investigated are specified in terms of either a fixed line or a fixed plane: rotations revolve around an axis line; projections collapse into a projection plane. Both lines and planes can be specified by a point Q and a unit vector u. For lines L, the vector u is parallel to L; for planes S, the vector u is normal to S. Our strategy for computing transformations on a vector v is to decompose v into components $v_{||}$ and $v_{\perp}$ parallel and perpendicular to the unit vector u. Typically one of these components is unaffected by the transformation and the effect of the transformation on the other component is easy to compute. We then reassemble the transformation on these components to derive the effect of the transformation on the original vector.

Formulas for transforming points follow easily from formulas for transforming vectors, since a point P can typically be expressed as the sum of a vector $P-Q$ and a fixed point Q of the transformation. This fixed point is typically the point Q that defines the line or plane that specifies the transformation. Thus once we know the effect of the transformation on arbitrary vectors, we can easily find the effect of the transformation on arbitrary points

The general approach outlined above is important and powerful; you should incorporate this strategy into your analytic toolkit. We shall have occasion to use this strategy again in other applications later in this course.

6.2 Formulas for Affine and Projective Transformations.

Translation -- by the vector w

$$v^{new} = v$$

$$P^{new} = P + w$$

Rotation -- by the angle θ around the line L determined by the point Q and the unit direction vector w (Formula of Rodrigues)

$$v^{new} = (\cos\theta)v + (1-\cos\theta)(v \bullet w)w + (\sin\theta)w \times v$$

$$P^{new} = Q + (\cos\theta)(P-Q) + (1-\cos\theta)\big((P-Q)\bullet w\big)w + (\sin\theta)w \times (P-Q)$$

Mirror Image -- in the plane S determined by the point Q and the unit normal n

$$v^{new} = v - 2(v \bullet n)n$$

$$P^{new} = P + 2\big((P-Q)\bullet n\big)n$$

Uniform Scaling -- around the point Q by the scale factor s

$$v^{new} = sv$$

$$P^{new} = Q + s(P-Q) = sP + (1-s)Q$$

Non-Uniform Scaling -- around the point Q by the scale factor s in the direction of the unit vector w

$$v^{new} = v + (s-1)(v \bullet w)w$$

$$P^{new} = P + (s-1)\big((P-Q)\bullet w\big)w$$

Orthogonal Projection -- into the plane S determined by the point Q and the unit normal n

$$v^{new} = v - (v \bullet n)n$$

$$P^{new} = P - \big((P-Q)\bullet n\big)n$$

Perspective Projection -- from the eye point E into the plane S determined by the point Q and the unit normal vector n

$$P^{new} = \frac{\big((E-Q)\bullet n\big)P + \big((Q-P)\bullet n\big)E}{(E-P)\bullet n}$$

Exercises:

1. Let A be an affine transformation. Show that

$$A(\sum_k \lambda_k P_k) = \sum_k \lambda_k A(P_k) \quad whenever \quad \sum_k \lambda_k \equiv 0,1 .$$

2. A *shear transformation* is defined in terms of a shearing plane S, a unit vector u in the plane S, and an angle ϕ in the following fashion. Given any point P, project P orthogonally onto a point P' in the shearing plane S. Now slide P parallel to u to a point P^{new} so that $\angle P^{new}P'P = \phi$. The point P^{new} is the result of applying the shearing transformation to the point P (see Figure 10).

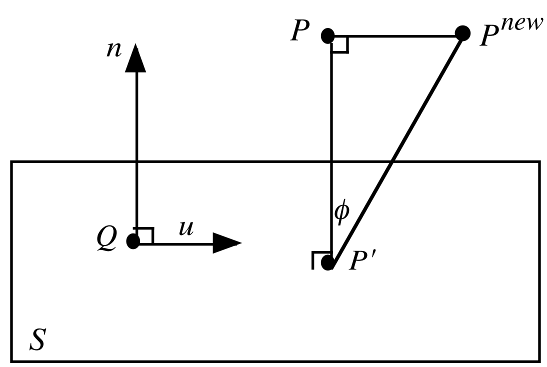

Figure 10: Shear

Let

S = Shearing plane

n = Unit vector perpendicular to S

Q = Point on S

u = Unit vector in S (i.e. unit vector perpendicular to n)

ϕ = Shear angle

a. Show that for any point P

$$P^{new} = P + \tan(\phi)\big((P - Q) \bullet n\big)u$$

b. For each vector $v = P - R$, define $v^{new} = P^{new} - R^{new}$. Show that

$$v^{new} = v + \tan(\phi)(v \bullet n)u$$

c. Conclude that

i, v^{new} is independent of the choices of the points P and R that represent the vector v.

ii. $v^{new} = v \iff v \perp n$

d. Interpret the result of part b geometrically.

3. Parallel projection is projection onto a plane S along the direction of a unit vector u. As usual the plane S is defined by a point Q and a unit normal vector n (see Figure 11). Show that:

a. $v^{new} = v - \dfrac{v \bullet n}{u \bullet n} u$

b. $P^{new} = P - \dfrac{(P-Q) \bullet n}{u \bullet n} u$

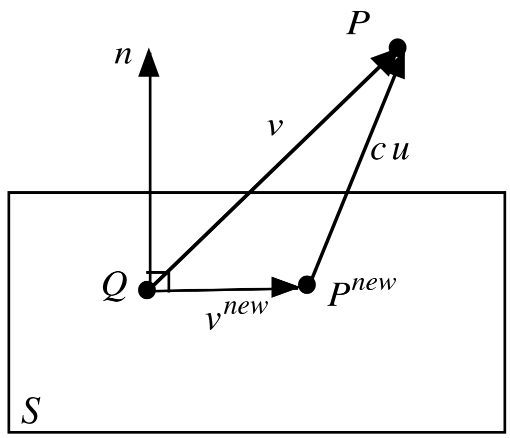

Figure 11: Parallel projection along a direction u into a plane S.

4. Let v_1, v_2 be unit vectors, and let $u = v_1 \times v_2$.
   a. Show that the transformation

   $$R(v) = (v_1 \bullet v_2)v + \frac{v \bullet u}{1 + v_1 \bullet v_2} u + u \times v,$$

   rotates the vector v_1 into the vector v_2.
   b. Interpret this result geometrically.

5. Recall from Lecture 4 that the image of one point and two linearly independent vectors determines a unique affine transformation in the plane.
   a. Show that the image of one point and three linearly independent vectors determines a unique affine transformation in 3-space.
   b. Fix 2 sets of 3 vectors v_1, v_2, v_3 and w_1, w_2, w_3, and define

   $$A(u) = \frac{Det(u \;\; v_2 \;\; v_3)w_1 + Det(v_1 \;\; u \;\; v_3)w_2 + Det(v_1 \;\; v_2 \;\; u)w_3}{Det(v_1 \;\; v_2 \;\; v_3)}.$$

   Show that

   $$A(v_i) = w_i \qquad i = 1,2,3.$$

6. Recall from Lecture 4 that the image of three non-collinear points determines a unique affine transformation in the plane.

a. Show that the image of four non-coplanar points determines a unique affine transformation in 3-space.

b Fix 2 sets of 4 points P_1, P_2, P_3, P_4 and Q_1, Q_2, Q_3, Q_4, and define

$$A(R) = \frac{Det(P_2 - R \;\; P_3 - R \;\; P_4 - R)Q_1 - Det(P_1 - R \;\; P_3 - R \;\; P_4 - R)Q_2}{Det(P_2 - P_1 \;\; P_3 - P_1 \;\; P_4 - P_1)}$$
$$+ \frac{Det(P_1 - R \;\; P_2 - R \;\; P_4 - R)Q_3 - Det(P_1 - R \;\; P_2 - R \;\; P_3 - R)Q_4}{Det(P_2 - P_1 \;\; P_3 - P_1 \;\; P_4 - P_1)}.$$

Show that

$$A(P_i) = Q_i \qquad i = 1,2,3,4.$$

7. Recall from Lecture 4, Exercise 19 that every nonsingular affine transformation in the plane is the composition of 1 translation, 1 rotation, 1 shear, and 2 non-uniform scalings. Show that every nonsingular affine transformation in 3-dimensions is the composition of 1 translation, 2 rotations, 2 shears, and 3 non-uniform scalings.

8. Let u, w be unit vectors and define the map $w \otimes u$ by setting

$$(w \otimes u)(v) = (v \bullet w)u .$$

Show that

a. $w \otimes u$ is a linear transformation

b. $(u \otimes u)(v) = v_{||}$, where $v_{||}$ is the parallel projection of v on u.

9. Let u be a unit vector and define the map $u \times \_$ by setting

$$(u \times \_)(v) = u \times v .$$

Show that $u \times \_$ is a linear transformation.

10. Let $R(v)$ denote the vector generated by rotating the vector v by the angle θ around the vector w. Using the Formula of Rodrigues, show that

a. $R(u) \bullet R(v) = u \bullet v$

b. $R(u \times v) = R(u) \times R(v)$

11. Let $M(v)$ denote the mirror image of the vector v in a plane normal to the vector n. Show that

a. $M(u) \bullet M(v) = u \bullet v$

b. $M(u \times v) = -M(u) \times M(v)$

12. Let $S(v)$ denote the transformation that scales each vector v by the scale factor s. Show that

a. $S(u) \bullet S(v) = s^2(u \bullet v)$

b. $S(u \times v) = \big(S(u) \times S(v)\big) / s$

13. Derive the formula for spherical linear interpolation (SLERP, see Lecture 11) in 3-dimensions

$$v(t) = \frac{\sin((1-t)\phi)}{\sin(\phi)}v_0 + \frac{\sin(t\phi)}{\sin(\phi)}v_1, \textit{ where } \phi \textit{ is the angle between the vectors } v_0 \textit{ and } v_1$$

by rotating the vector v_0 by the angle $t\phi$ around the vector $w = \dfrac{v_0 \times v_1}{|v_0 \times v_1|}$.

14. Let T_1, T_2 be transformations from $\mathbf{R}^3$ to $\mathbf{R}^3$, and let $T(\lambda) = (1-\lambda)T_1 + \lambda T_2$. Show that:

    a. If T_1, T_2 are affine transformations, then $T(\lambda)$ is also an affine transformation.

    b. If T_1, T_2 are rotations, then $T(\lambda)$ is not necessarily a rotation.