The Scaling Behavior of Viewing Transformations

Salim S. Abi-Ezzi and Leon A. Shirman

3D Technology Group
Sun Microsystems, Inc.
Mountain View, CA 94043, U.S.A.

We give a formal treatment of the scaling behavior of parallel and perspective viewing transformations. We first define the minimum, maximum, and average scale of a transformation at an arbitrary point in space, and then derive closed form expressions for quick evaluation of these values.

Although these results have a wide applicability in computer graphics, we illustrate their usefulness in the context of the dynamic tessellation of curved surfaces. We determine the maximum scale of a viewing transformation in a finite region that bounds a curved surface. Obtaining this scale allows to tessellate a curved surface in a pre-viewing coordinate system, where lighting needs to take place, and still honor a post-viewing approximation threshold.

1. Introduction

Today's graphics accelerators are characterized by having very fast special VLSI for Gouraud shaded Z-buffered triangles, complemented with microprogramable floating-point processors for floating-point intensive tasks. While it is cost effective to produce special VLSI for the triangle primitive, it is not so for more complex surface primitives. This is one of the reasons why the techniques that render surfaces directly ([Lane80], [Lien87]) didn't make it into commercial VLSI. Therefore, it becomes important to tessellate arbitrary classes of curved surfaces into triangles that are consumed by the triangle processing VLSI. In a sense this is the RISC approach in graphics.

The granularity of tessellation of surfaces into triangles in effect determines the density of sampling of both the geometry and the lighting equation, before Gouraud shading takes place. Thus, tessellation is needed for flat surfaces as well as for curved ones, where the former is needed to control the sampling of the lighting equation. Clearly, this sampling should be view dependent in the sense that objects closer to the eye point need to be sampled more finely than those that are farther away. Thus, our goal is the speedy tessellation of objects into appropriately sized triangles for display purposes.

In interactive graphics and virtual reality types of applications it is common that the viewer is continually moving and the relation between the eye point and the objects being rendered is changing frequently. This calls for the dynamic (per frame) reduction of objects into properly sized triangles based on their relation to the eye point. Thus, there is a need

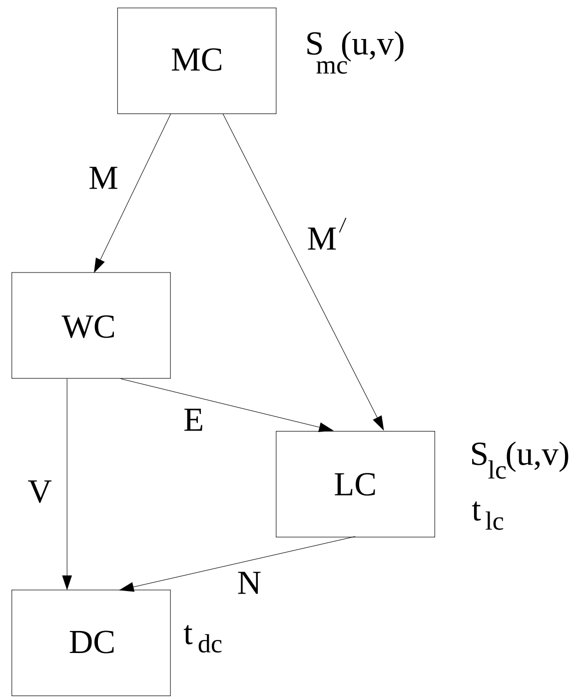

Figure 1: The relation of lighting coordinates to the pipeline. The approximation threshold is specified in DC, and tessellation takes place in LC.

for a formal and fast method that evaluates this relationship for the purpose of determining how finely to tessellate objects. We present such a method based on studying the scaling behavior of viewing transformations.

While it is common to apply heuristic techniques to approximate the scaling effects of a view, in this paper we present formal techniques for the exact scaling behavior of parallel and perspective views. We derive and present closed form expressions based on easily obtainable parameters for the fast computation of these scales at arbitrary points in space.

As an application of our result we discuss the dynamic tessellation of curved surfaces to meet specified approximation thresholds. For viewing purposes it is most useful to specify the threshold in display coordinates. This is the main reason why dynamic tessellation is so important in situations where the transformation is highly varying: as the transformation varies, the transformed surface changes accordingly (in display coordinates), and therefore it needs to be tessellated repeatedly to keep satisfying the given threshold without an excessive amount of triangles.

2. Lighting Coordinates

Lighting Coordinates (LC) exhibit a set of useful properties for accurate lighting, fast transformations, and clipping ([Abi-Ezzi90]). In this paper we point out and utilize yet another useful property of lighting coordinates, namely that the scaling behavior of a perspective becomes tractable due to the special relationship between LC and Display Coordinates (DC).

Lighting Coordinates can be thought of as an intermediate coordinate system that has the geometric properties of World Coordinates (WC), where lighting calculations take place, and that is close to Display Coordinates, where rendering takes place. The idea behind LC is to factor a viewing transformation V into a product of a rigid factor E and a sparse non-rigid factor N. E relates World Coordinates (WC) to LC, while N relates LC to DC (Figure 1). Since WC and LC are related by a rigid transformation, and because LC and DC are related by a sparse matrix, it is convenient to perform tessellation of curved primitives and lighting calculations in LC. However, a tessellation approximation threshold is usually specified in terms of screen size (pixels) in DC. Therefore, we need to scale the approximation threshold from DC to LC, since it is needed in the coordinate system where tessellation takes place.

Since E is rigid, N embodies the complete scaling behavior of V. However, V is usually a full 4×4 matrix while N is sparse and simple, which is precisely why it is more tractable to study the scaling behavior between LC and DC, rather than between WC and DC.

3. The Scaling Behavior

In this section we investigate the scaling behavior of a viewing transformation, and pursue expressions for the scale that a view is capable of at a given point in space. A view is defined as a matrix V that relates WC to DC. In case V is affine the view is parallel, and if V is non-affine (projective) the view is typically perspective.

Let A and B be two points in 3D space; $[AB]$ be the segment from A to B; and $|AB|$ the length of $[AB]$. We define the scaling function of V as follows:

$$s_V(A,B) = \frac{|(VA)(VB)|}{|AB|}.$$

Let σ be the surface of a sphere centered at A and of radius r. Let B be an arbitrary point on σ, and consider the image of σ under transformation V. We define the following characteristics of the transformation of σ by V:

$$s_V^{\max}(\sigma) = \max s_V(A,B), \qquad \forall B \in \sigma,$$

$$s_V^{\min}(\sigma) = \min s_V(A,B), \qquad \forall B \in \sigma,$$

$$(s_V^{\text{ave}})^2(\sigma) = \frac{1}{S_\sigma}\int_\sigma s_V^2(A,B)dB,$$

where S_σ is the surface area of σ. Note that the average scale is defined in terms of averaging the *square* of the scaling function.

The maximum, minimum, and average scales of a viewing transformation at a point A are defined as follows:

$$s_V^{\max}(A) = \lim_{r\to 0} s_V^{\max}(\sigma),$$

$$s_V^{\min}(A) = \lim_{r\to 0} s_V^{\min}(\sigma),$$

$$s_V^{\text{ave}}(A) = \lim_{r\to 0} s_V^{\text{ave}}(\sigma).$$

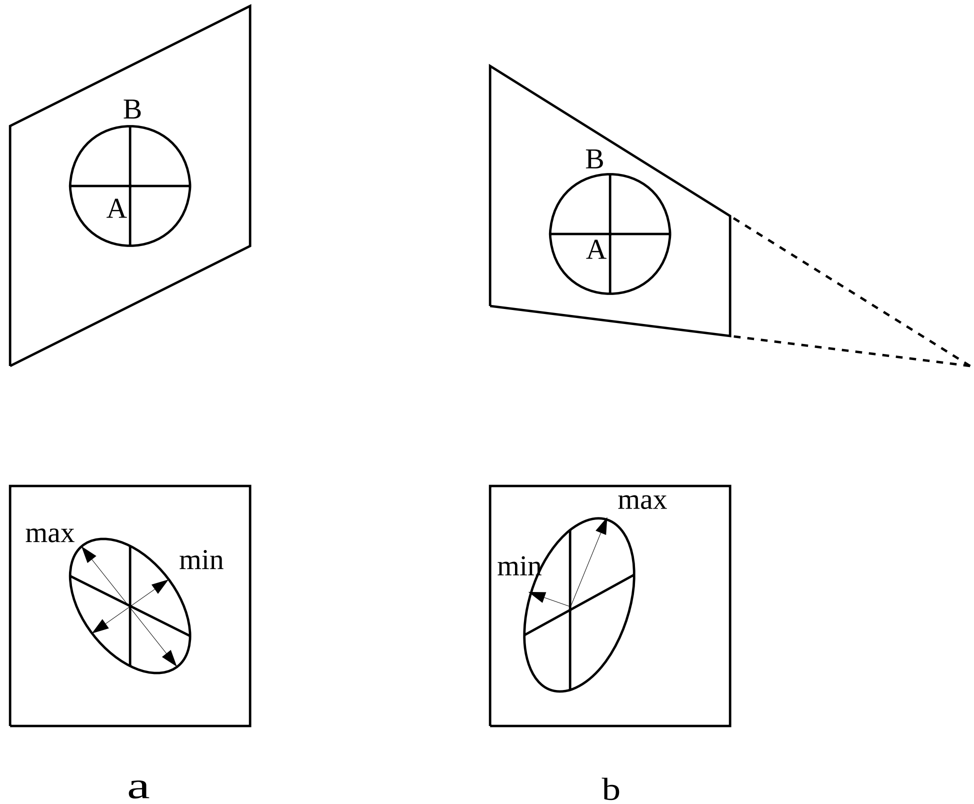

Figure 2: Images of a circle under parallel (**a**) and perspective (**b**) transformations.

Figure 2 shows the influence of a parallel and perspective transformations on a circle.

In this paper we study the function s_V and deduce closed form expressions for $s_V^{\max}(A), s_V^{\min}(A)$, and $s_V^{\text{ave}}(A)$. These closed form expressions allow the efficient calculation of the three characteristic scales of a transformation at a given point A.

The scaling behavior of a parallel view depends only on the orientation of segment $[AB]$. In contrast, the scaling behavior of a perspective is considerably more complicated, since it is dependent on the position, length, and orientation of segment $[AB]$. Also a perspective can map certain finite segments into infinite ones; in Section 3.2 we will show how we deal with this possibility.

Figure 2 illustrates what happens to a circle under parallel and perspective transformations, and shows that in both cases the image of a circle is an ellipse. The important difference is that for a parallel transformation, the center of curvature of the ellipse is the image of the center of the circle, which is not the case for a perspective transformation. Also, the directions of maximum and minimum scale for the perspective transformation are dependent on the position of the circle, which is not the case for parallel transformations.

3.1. Parallel Transformations

Based on [Abi-Ezzi90] we can factor a parallel viewing transformation $V = LE$ as follows:

$$V = \begin{bmatrix} s_x & 0 & s_x h_x & 0 \\ 0 & s_y & s_y h_y & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_{00} & a_{01} & a_{02} & t_x \\ a_{10} & a_{11} & a_{12} & t_y \\ a_{20} & a_{21} & a_{22} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

where E is the rigid component that maps WC to LC, and L is a linear transformation that maps LC to DC. We use the letter L instead of N to emphasize that the transformation is linear and not perspective. Basically, L embodies the non-uniform scale and the shear factors of V; s_x, s_y, and s_z are the scale factors, and h_x and h_y are the shear factors in the corresponding directions.

Since linear transfomations preserve aspect ratios, the value of $s_L(A, B)$ is only dependent on the orientation of $[AB]$ and is independent of its position and length; two segments have the same orientation if they're parallel. Therefore, for two points A' and B' such that $[A'B'] \quad || \quad [AB]$ we have $s_L(A', B') = s_L(A, B)$. Furthermore, the lack of dependence of s_L on position implies that $s_L^{\max}(A), s_L^{\min}(A)$, and $s_L^{\mathrm{ave}}(A)$ are independent of A, and are constant across all points in space; essentially they are characteristics of L.

From linear algebra, the maximum and minimum scales that L is capable of over all segments (orientations) in space is related to the eigenvalues of the matrix $L^* = L^T L$:

$$s_L^{\max} \;=\; \max_{\vec{v} \in \Re^3 - [0,0,0]} \frac{||L\vec{v}||}{||\vec{v}||} \;=\; \sqrt{\lambda_{\max}(L^*)}, \tag{1}$$

$$s_L^{\min} \;=\; \min_{\vec{v} \in \Re^3 - [0,0,0]} \frac{||L\vec{v}||}{||\vec{v}||} \;=\; \sqrt{\lambda_{\min}(L^*)}, \tag{2}$$

where $\lambda_{\max}$ and $\lambda_{\min}$ are the maximum and minimum eigenvalues of L^*, respectively. The eigenvalues of L^* are the roots of the characteristic polynomial

$$\det\left(L^T L - \lambda I\right) = 0,$$

which in the 2D and 3D cases must have two and three real roots, respectively, since L^* is symmetric.

3.1.1. The Analysis in 2D

For the 2D case we ignore the Y dimension and σ becomes a circle in the XZ plane. The resulting characteristic polynomial is a quadratic equation:

$$\lambda^2 - \left(s_x^2\left(1 + h_x^2\right) + s_z^2\right)\lambda + s_x^2 s_z^2 = 0,$$

which has closed form solutions for the two real roots. Note that in the special case where $h_x = 0$, i.e. no shear component, we have:

$$s_L^{\max} = \max\left(s_x, s_z\right), \quad s_L^{\min} = \min\left(s_x, s_z\right).$$

For the average scaling factor, we assume without loss of generality that A is at the origin and we integrate around the circumference of the unit circle σ. Since L^* is symmetric, its normalized eigenvectors form an orthonormal basis; let u_1 and u_2 be the coordinates of B in this basis; let $\vec{v} = \vec{AB}$. We have:

$$s_L^2(A, B) = \|L\vec{v}\|^2 = (L\vec{v}, L\vec{v}) = \left(L^T L\vec{v}, \vec{v}\right) = \lambda_1 u_1^2 + \lambda_2 u_2^2.$$

Therefore,

$$\left(s_L^{\text{ave}}\right)^2 = \frac{1}{2\pi}\int\limits_{\sigma} s_L^2(A,B)\,dB = \frac{1}{2\pi}\int\limits_{\sigma}\left(\lambda_1 u_1^2 + \lambda_2 u_2^2\right)dB.$$

The normalizing factor corresponds to the circumference of the unit circle. Making a change of variables to polar coordinates, we obtain

$$(s_L^{\text{ave}})^2 = \frac{1}{2\pi}\int\limits_0^{2\pi}\left(\lambda_1\cos^2\phi + \lambda_2\sin^2\phi\right)d\phi = \frac{1}{2}(\lambda_1+\lambda_2). \tag{3}$$

Thus, the square of the average scale of a parallel transformation is just the average of the eigenvalues of L^*.

3.1.2. The Analysis in 3D

The 3D case is a straightforward generalization of the 2D case; the only difference is that the characteristic polynomial is cubic:

$$\lambda^3 - \left(s_x^2\left(1+h_x^2\right) + s_y^2\left(1+h_y^2\right) + s_z^2\right)\lambda^2 + \left(s_x^2 s_y^2\left(1+h_x^2+h_y^2\right) + s_z^2\left(s_x^2+s_y^2\right)\right)\lambda - s_x^2 s_y^2 s_z^2 = 0.$$

There are closed form expressions for the roots of a cubic [Press88]. Moreover, since L^* is symmetric, all three roots are real. The largest/smallest root of this equation is the square of the maximum/minimum scale factor that L is capable of.

There are again special cases that simplify the solution:

1. L has no shear components, i.e. $h_x = h_y = 0$. In this case, $s_L^{\max} = \max(s_x, s_y, s_z)$, $s_L^{\min} = \min(s_x, s_y, s_z)$.
2. The transformation is x, y symmetric, which is a fairly common case; i.e. $s_x = s_y = s$ and $h_x = h_y = h$. In this case, the cubic characteristic polynomial can be factored into a product of a linear and a quadratic parts:

$$\left(\lambda - s^2\right)\left(\lambda^2 - \left(s^2\left(1+2h^2\right)+s_z^2\right)\lambda + s^2 s_z^2\right) = 0.$$

For the average scale, we assume without loss of generality that A is at the origin and we integrate around the surface of the unit sphere σ. Since L^* is symmetric, its normalized eigenvectors form an orthonormal basis. Let u_1, u_2, and u_3 be the coordinates of B in this basis. Similarly to the 2D case,

$$\left(s_L^{\text{ave}}\right)^2 = \frac{1}{4\pi}\int\limits_{\sigma} s_L^2(A,B)\,dB = \frac{1}{4\pi}\int\limits_{\sigma}\left(\lambda_1 u_1^2 + \lambda_2 u_2^2 + \lambda_3 u_3^2\right)dB.$$

The normalizing factor in this case corresponds to the area of the unit sphere. Making the change of variables to spherical coordinates, we obtain

$$\begin{aligned}(s_L^{\text{ave}})^2 &= \frac{1}{4\pi}\int\limits_0^{2\pi}\int\limits_0^{\pi}\left(\lambda_1\cos^2\phi\sin^2\psi + \lambda_2\sin^2\phi\sin^2\psi + \lambda_3\cos^2\psi\right)\sin\psi\,d\phi d\psi \\ &= \frac{1}{3}(\lambda_1+\lambda_2+\lambda_3).\end{aligned} \tag{4}$$

The factor $\sin\psi$ is the Jacobian for the spherical coordinates (for the unit radius). Again, the square of the average scaling factor is just the average of the eigenvalues of L^*.

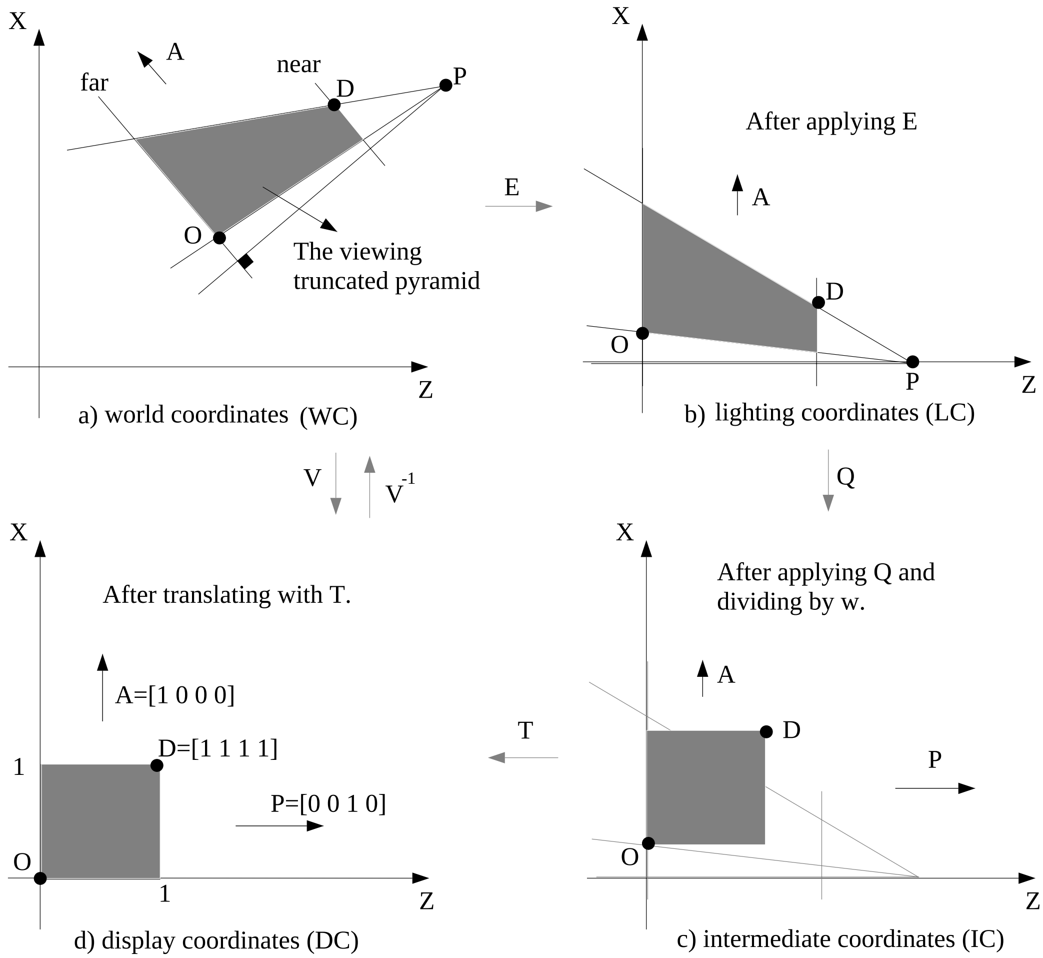

Figure 3: The factoring of a perspective transformation.

3.2. Perspective Transformations

Based on [Abi-Ezzi90] we can factor a perspective viewing transformation $V = NE = TQE$ as follows:

$$V = \begin{bmatrix} 1 & 0 & 0 & t'_x \\ 0 & 1 & 0 & t'_y \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & -1/P_z & 1 \end{bmatrix} \begin{bmatrix} a_{00} & a_{01} & a_{02} & t_x \\ a_{10} & a_{11} & a_{12} & t_y \\ a_{20} & a_{21} & a_{22} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

where E is a rigid transformation that maps WC to LC, and Q is a perspective (non-linear) transformation that maps LC to an intermediate coordinate (IC) system, which is a translation T away from DC (Figure 3). Thus, N is factored further into T and Q, and clearly Q is the only component of V that has any scaling effects. In LC the eye point P lies on the positive Z axis at distance P_z from the origin. Furthermore, the far plane coincides with the XY plane, and the near plane is between P and the far plane.

As indicated earlier, the perspective transform can map finite line segments into infinite ones by wraping them around infinity. Therefore, it only makes sense to study the maximum scale of Q in a subspace that doesn't wrap around infinity. A valid choice for this subspace

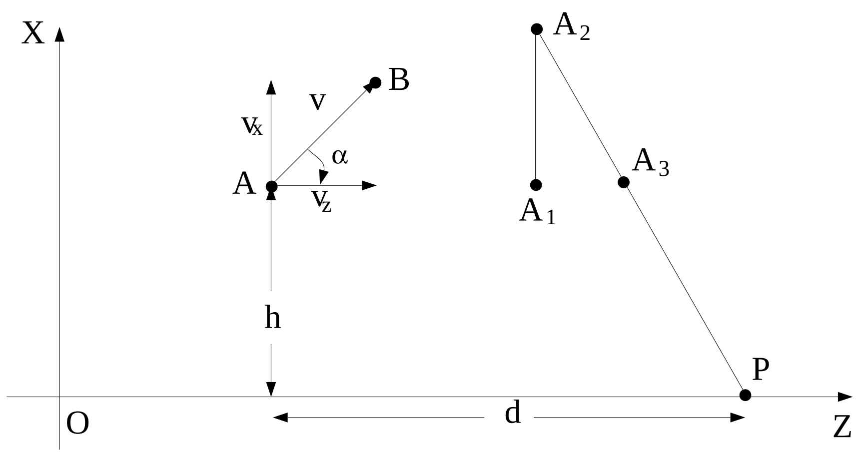

Figure 4: The needed parameters for approaching the 2D case.

could be a view volume, which is indicated by a shaded region in Figure 3. If we desire to study the scaling of a certain primitive, we can look at the intersection of the view volume with the convex hull that contains this primitive (Section 4).

3.2.1. The analysis in 2D

In the 2D case we have:

$$Q = \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_z & 0 \\ 0 & -1/P_z & 1 \end{bmatrix}.$$

We study the scaling influence of Q on segment $[AB]$ where A and B are chosen behind the near plane in relation to the eye point. Without loss of generality assume that A and B are on the positive side of the Z axis. We make the following definitions which are illustrated in Figure 4:

$\vec{v} = (B - A),\ \alpha = \angle(\vec{Z}, \vec{v}),\ \tau = \tan(\alpha),$
$v_x = \tau v_z,\ d = P_z - A_z,\ h = A_x.$

The square of the scale is:

$$s_Q^2(A, B) = \frac{|(QA)(QB)|^2}{|AB|^2}.$$

Substituting for $Q, A,$ and B we get:

$$s_Q^2(A, B) = \frac{P_z^2 \left(P_z^2 s_z^2 + s_x^2 (h + d\tau)^2 \right)}{d^2 (1 + \tau^2)(d - v_z)^2}.$$

Since $v_z = |\vec{v}| \cos(\alpha)$ then:

$$\lim_{|AB| \to 0} s_Q^2(A, B) = \xi_Q^2(A, \alpha) = \frac{P_z^2 \left(P_z^2 s_z^2 + s_x^2 (h + d\tau)^2 \right)}{d^4 (1 + \tau^2)}, \tag{5}$$

which can be thought of as the scale of Q at a point A and at a certain orientation α.

The three parameters that we are concerned with are d, h, and τ. From a simple inspection of Equation 5 we deduce the following:

1. Since h and d are positive then $\xi_Q(A,\alpha)$ attains a maximum for $\tau > 0$, i.e. α must lie in the 1st or in the 3rd quadrant.
2. $\xi_Q(A,\alpha)$ increases with h, so (see Figure 4)

$$\forall\alpha, \quad \xi_Q(A_1,\alpha) < \xi_Q(A_2,\alpha).$$

3. $\xi_Q(A,\alpha)$ increases as d decreases. This can be observed in Figure 4, as

$$\forall\alpha, \quad \xi_Q(A_2,\alpha) < \xi_Q(A_3,\alpha).$$

We show this by substituting $h \leftarrow kd$ for both A_2 and A_3 in Equation 5.

Now we fix h and d and we search for the specific α at which ξ_Q attains a maximum. We find this maximum by finding the zeros of the derivative of ξ_Q^2 with respect to τ.

The numerator of the derivative of ξ_Q^2 is the following quadratic equation:

$$\left(dhs_x^2\right)\tau^2 + \left(h^2 s_x^2 + P_z^2 s_z^2 - d^2 s_x^2\right)\tau - dhs_x^2 = 0.$$

By analyzing the determinant of the above equation we deduce that there are always two real roots, one positive and one negative. After substituting the expression for the positive root in Equation 5, we obtain the following expression:

$$\left(s_Q^{\max}\right)^2(A) = \xi_Q^2(A,\alpha_{\max}) = \frac{P_z^2\left(d^2 s_x^2 + h^2 s_x^2 + P_z^2 s_z^2 + \sqrt{\Delta}\right)}{2d^4}, \tag{6}$$

where

$$\Delta = \left(h^2 s_x^2 + P_z^2 s_z^2 - d^2 s_x^2\right)^2 + 4d^2 h^2 s_x^4.$$

So, given a point $A, \quad A_z \neq P_z$ the above expression gives the maximum scale of Q at A. The minimum occurs at the negative root:

$$\left(s_Q^{\min}\right)^2(A) = \xi_Q^2(A,\alpha_{\min}) = \frac{P_z^2\left(d^2 s_x^2 + h^2 s_x^2 + P_z^2 s_z^2 - \sqrt{\Delta}\right)}{2d^4}. \tag{7}$$

Again, we use integrals to evaluate the average scaling factor. From Equation 5 we have:

$$\left(s_Q^{\text{ave}}\right)^2(A) = \frac{1}{\pi}\int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} \frac{P_z^2\left(P_z^2 s_z^2 + s_x^2(h + d\tau)^2\right)}{d^4(1+\tau^2)} d\alpha.$$

Substituting $\tau = \tan(\alpha)$ and evaluating the integral yields:

$$\left(s_Q^{\text{ave}}\right)^2(A) = \frac{P_z^2}{2d^4}\left(P_z^2 s_z^2 + s_x^2 h^2 + s_x^2 d^2\right). \tag{8}$$

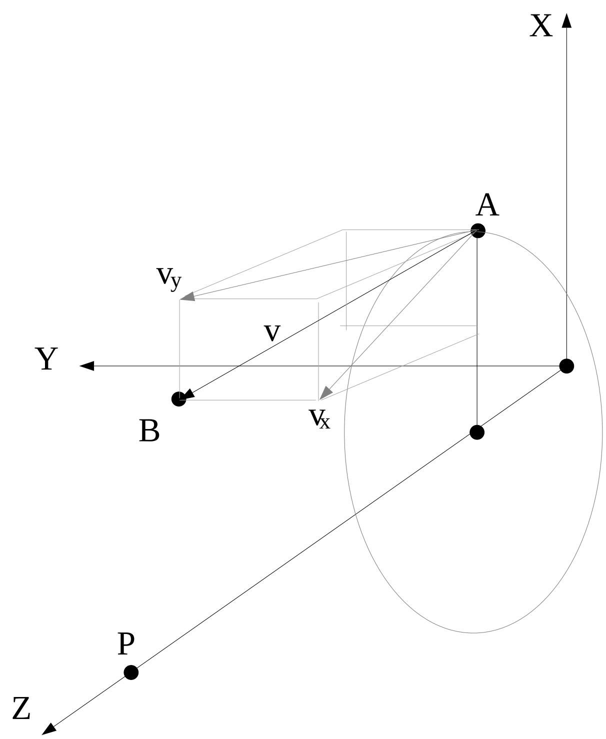

Figure 5: The needed parameters for approaching the 3D case.

3.2.2. The analysis in 3D

Since in practice it is common to have XY symmetric viewing windows and in order to keep the analysis tractable we use $s = \max(s_x, s_y)$ in the 3D perspective matrix:

$$Q = \begin{bmatrix} s & 0 & 0 & 0 \\ 0 & s & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & -1/P_z & 1 \end{bmatrix}.$$

In case $s_x \neq s_y$, it is easy to verify that this simplification will result with an upper bound of the maximum scale that we're after. It is possible to pursue the analysis with different s_x and s_y; however, the resulting expression is too complicated to be practical.

For all points on a circle that lies on a plane parallel to the XY plane, and whose center lies on the Z axis (see Figure 5), Q attains an identical maximum scale, due to symmetry around the Z axis. So, without loss of generality, it is sufficient to consider a point A that lies on the positive half of the XZ plane, while B is an arbitrary point in 3D. Both points must lie behind the near plane. Similarly to the 2D case we define the following (Figure 5):

$\vec{v} = (B - A)$.

α_x and α_y are the angles between the projections of $\vec{v}$ on the XZ and YZ planes respectively and the Z axis.

$\tau_x = \tan(\alpha_x)$ and $\tau_y = \tan(\alpha_y)$, $v_x = \tau_x v_z$ and $v_y = \tau_y v_z$, $d = P_z - A_z$, $h = A_x$.

The square of the scaling function at the limit is:

$$\lim_{|AB| \to 0} s_Q^2(A, B) = \xi_Q^2(A, \alpha_x, \alpha_y) = \frac{P_z^2 \left(s_z^2 P_z^2 + s^2 (h + d\tau_x)^2 + s^2 d^2 \tau_y^2 \right)}{d^4 \left(1 + \tau_x^2 + \tau_y^2\right)}. \qquad (9)$$

Taking partial derivatives with respect to τ_x and τ_y and equating the resulting numerators to 0 yields the following system of equations:

$$\begin{gathered}(dhs^2)\tau_x^2 + (h^2s^2 + P_z^2 s_z^2 - d^2 s^2)\tau_x - dhs^2(\tau_y^2 + 1) = 0,\\ \tau_y(-2dhs^2\tau_x + (d^2s^2 - h^2s^2 - P_z^2 s_z^2)) = 0.\end{gathered}$$

This system of non-linear equations has only two pairs of simultaneous roots, both of which have $\tau_y = 0$. This is a very interesting and useful result which implies that ξ_Q^2 has only one maximum and one minimum that correspond to the 2D case discussed earlier since $\tau_y = 0$. Therefore, closed form expressions for the maximum and the minimum are as in Equations 6 and 7, with s substituted for s_x.

For the average we need to take the double integral of Equation 9 with respect to both α_x and α_y:

$$\left(s_Q^{\text{ave}}\right)^2(A) = \frac{1}{2\pi}\int_{-\frac{\pi}{2}}^{\frac{\pi}{2}}\int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} \frac{P_z^2\left(P_z^2 s_z^2 + s^2(h + d\tan\alpha_x)^2 + s^2 d^2\tan^2\alpha_y\right)}{d^4\left(1 + \tan^2\alpha_x + \tan^2\alpha_y\right)}\cos\alpha_y\, d\alpha_x d\alpha_y.$$

The normalizing factor 2π corresponds to the area of a unit semisphere. The factor $\cos\alpha_y = \sin\left(\psi + \frac{\pi}{2}\right)$ is the Jacobian for the spherical coordinates (cf. Equation 4). This yields the following:

$$\left(s_Q^{\text{ave}}\right)^2(A) = \frac{P_z^2}{2d^4}\left[\left(\frac{3\pi}{2} - 4\right)\left(P_z^2 s_z^2 + s^2h^2\right) + \left(6 - \frac{3\pi}{2}\right)s^2d^2\right]. \tag{10}$$

4. Dynamic Tessellation of Curved Surfaces

An important application of the results above is the dynamic tessellation of curved surfaces to meet a post viewing approximation threshold. In [Abi-Ezzi91] we showed how derivative bounds could be used for step size determination in order to guarantee meeting a certain approximation criterion. While the surface S_{mc} is specified in Modeling Coordinates (MC), the threshold t_{dc} is typically specified in DC (see Figure 1). For our purpose here, we assume that the surface is specified in LC. By using the expressions for the maximum scale of a viewing transformation we map t_{dc} into t_{lc} so that if we tessellate in LC and honor the threshold t_{lc}, then the threshold t_{dc} is guaranteed to be met in DC. This allows us to use view independent derivative bounds that are precomputed in MC to honor post view DC thresholds ([Abi-Ezzi91]). This is important because it is costly to compute derivative bounds and views tend to change frequently.

Suppose we have a Bézier patch $S(u,v)$ with control points B_{ij} in LC, and a perspective transformation Q with a viewing volume (truncated pyramid) determined by a near and a far plane, and a window on a viewing plane. The near plane is at distance d_n from the eye point P, and the window is determined by its four corner points W_k.

We need a bound to the maximum scale that Q is capable of on the non-clipped portion of $S(u,v)$, since there is no need to meet the threshold within the clipped away portion. For point $A, \quad A_z \neq P_z$ we define the following function:

$$\tau(A) = \tan^2\left(\angle(A,P,O)\right) = \frac{A_x^2 + A_y^2}{\left(P_z - A_z\right)^2}.$$

We get the maximum bound by substituting the following values for d and h in Equation 6:

$$\begin{aligned} t &= \min\left(\max_{i,j}\left(\tau(B_{ij})\right), \max_{k}\left(\tau(W_k)\right)\right), \\ d &= \max\left(\min_{i,j}\left(P_z - B_{ij,z}\right), d_n\right), \\ h^2 &= d^2 t. \end{aligned} \tag{11}$$

Notice that this approach automatically deals with the case when a patch penetrates the eye plane in LC and therefore is wraped around infinity by Q. The fact that we obtain the bound for a region that is clamped by the front plane and hence is strictly in front of the eye point guarantees a finite bound.

5. Conclusion

We presented a formal analysis of the behavior of parallel and perspective viewing transformations. We obtained closed form expressions for the immediate evaluation of the maximum, minimum, and average scales of a transformation at a given point in space. While these characteristics are constant for parallel transformations, they vary from point to point for perspective transformations.

Obtaining these scaling characteristics of a transformation at a given point is useful for several purposes in computer graphics. The minimum and maximum scales can be used to obtain strict lower and upper bounds on the scaling of the transformation in a given bounded region. We also gave expressions for the average scale of a transformation that are useful for heuristic types of computations. These expressions for the average scale indicate how transformations perform in general without giving guarantees.

An important application of these results is the dynamic tessellation of curved surfaces. We showed how the maximum scale of a transformation across a bounded region can be used to guarantee meeting post-viewing approximation thresholds which are specified in DC, based on derivative bounds which are precomputed in MC.

6. References

1. [Abi-Ezzi89] Salim Abi-Ezzi: “The graphical Processing of B-splines in a Highly Dynamic Environment,” RPI Ph.D. dissertation, RDRC-TR-89001, Troy, New York (May 1989).

2. [Abi-Ezzi90] Salim Abi-Ezzi and Michael Wozny: “Factoring a Homogeneous Transformation for a more Efficient Graphics Pipeline,” *Proc. EUROGRAPHICS ’90*, (September 1990).

3. [Abi-Ezzi91] Salim Abi-Ezzi and Leon Shirman: "Tessellation of Curved Surfaces under Highly Varying Transformations," *Proc. EUROGRAPHICS '91*, (September 1991).

4. [Lane80] Jeffrey Lane, Loren Carpenter, Turner Whitted, and James Blinn: "Scan Line Methods for Displaying Parametrically Defined Surfaces," *Communications of the ACM*, 23(1), January 1980.

5. [Lien87] Sheue-Ling Lien, Michael Shantz, and Vaughan Pratt: "Adaptive Forward Differencing for Rendering Curves and Surfaces," *Computer Graphics*, 21(4), pp 111–117, July 1987.

6. [Press88] W. Press, B. Flannery, S. Teukolsky, and W. Vetterling: *Numerical Recipes in C*, Cambridge University Press, New York, 1988.