The Mathematics of Separating-Plane Perspective Shadow Mapping

Morten Mikkelsen
mm@ioi.dk

August 27, 2007

This document is a supplement to the paper "Separating-Plane Perspective Shadow Mapping" in the *journal of graphics tools*. In the following, the fundamental inequality on which the paper is based will be derived. For an introduction to the problem solved, please refer to the paper.

1 Theory

Throughout this document, a few industry terms will be used. One is *post-projective space*, which is the space in which the view frustum exists as a centered, axis-aligned $2 \times 2 \times 2$ cube ($2 \times 2 \times 1$ for D3D). Additionally, texture coordinates will be referred to as either normalized or unnormalized, which is not to be confused with vector normalization. Texture coordinates are normalized when a single square unit covers the entire texture and unnormalized when a single square unit only covers a texel. This is similar to the relationship between post-projective space and *screen space*, though for a given depth, the surface of the screen in post-projective space is four square units and not one.

It will be implied in the following that all the applied matrices are nonsingular. Let M_1 and M_2 be two such 4×4 transformations from world space to separate texture spaces of unnormalized texture coordinates. In Section 1.1, a simple inequality will be derived for which of the two provide better area-sampling density at a given point.

1.1 A General Comparison Test

Let C be the camera-to-world matrix, C_{proj} the camera-projection matrix, and (s, t, q) a point in screen space given some arbitrary range for the depth buffer $[z_{\min}; z_{\max}]$ (see Figure 1):

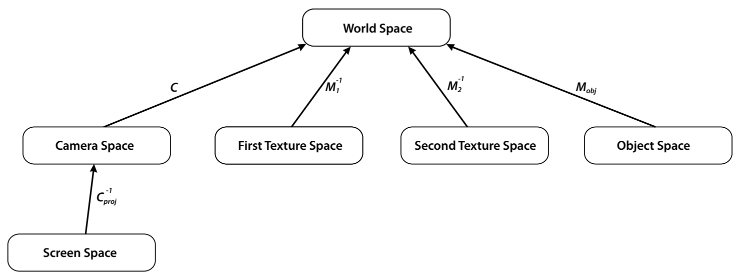


Figure 1: The transformation hierarchy.

$$M_i' = M_i \cdot C \cdot C_{\text{proj}}^{-1}, \qquad i \in \{1,2\},$$

$$\begin{bmatrix} x_i(s,t,q) \\ y_i(s,t,q) \\ z_i(s,t,q) \\ w_i(s,t,q) \end{bmatrix} = M_i' \cdot \begin{bmatrix} s \\ t \\ q \\ 1 \end{bmatrix}. \tag{1}$$

The final texture coordinates are computed using the following equations:

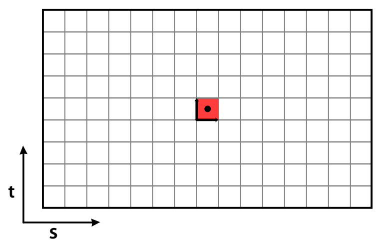
$$\begin{aligned} f_i(s,t,q) &= x_i(s,t,q)/w_i(s,t,q), \\ g_i(s,t,q) &= y_i(s,t,q)/w_i(s,t,q). \end{aligned}$$

The signed area covered by a screen space pixel in the texture maps can be evaluated by the determinant of the Jacobian matrix of (f_i, g_i) with respect to s and t (see Figure 2):

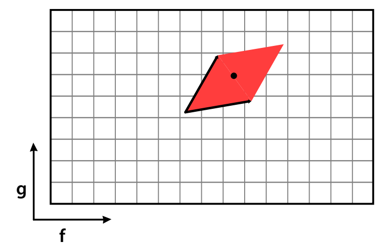
$$\begin{aligned} J(f_i, g_i) &= \begin{bmatrix} \frac{df_i}{ds} & \frac{df_i}{dt} \\ \frac{dg_i}{ds} & \frac{dg_i}{dt} \end{bmatrix} \\ &= \begin{bmatrix} \frac{\frac{dx_i}{ds}\cdot w_i - x_i \cdot \frac{w_i}{ds}}{w_i^2} & \frac{\frac{dx_i}{dt}\cdot w_i - x_i \cdot \frac{w_i}{dt}}{w_i^2} \\ \frac{\frac{dy_i}{ds}\cdot w_i - y_i \cdot \frac{w_i}{ds}}{w_i^2} & \frac{\frac{dy_i}{dt}\cdot w_i - y_i \cdot \frac{w_i}{dt}}{w_i^2} \end{bmatrix}. \end{aligned}$$

And we can now evaluate the determinant of $J(f_i, g_i)$:

$$\begin{aligned} \det[J(f_i, g_i)] &= \frac{df_i}{ds} \cdot \frac{dg_i}{dt} - \frac{dg_i}{ds} \cdot \frac{df_i}{dt} \\ &= \frac{\left(\frac{dy_i}{ds}\frac{dw_i}{dt} - \frac{dw_i}{ds}\frac{dy_i}{dt}\right)\cdot x_i + \left(\frac{dw_i}{ds}\frac{dx_i}{dt} - \frac{dx_i}{ds}\frac{dw_i}{dt}\right)\cdot y_i + \left(\frac{dx_i}{ds}\frac{dy_i}{dt} - \frac{dy_i}{ds}\frac{dx_i}{dt}\right)\cdot w_i}{w_i^3}. \end{aligned}$$



(a) screen space.



(b) texture space.

Figure 2: (a) The area of a single pixel in screen space. (b) The pixel after a transformation and projection into texture space. This transformation also depends on the depth of the pixel in screen space.

We can simplify this equation a little by introducing the vector

$$\overrightarrow{n_i} = \begin{pmatrix} \frac{dx_i}{ds} \\ \frac{dy_i}{ds} \\ \frac{dw_i}{ds} \end{pmatrix} \times \begin{pmatrix} \frac{dx_i}{dt} \\ \frac{dy_i}{dt} \\ \frac{dw_i}{dt} \end{pmatrix}. \tag{2}$$

Let the symbol $\bullet$ denote the dot product between two vectors. Now the signed area can be expressed as

$$\det[J(f_i, g_i)] = \frac{\overrightarrow{n_i} \bullet (x_i, y_i, w_i)}{w_i^3}.$$

Note that from (1), it follows that $\frac{d}{ds}(x_i, y_i, z_i, w_i)$ and $\frac{d}{dt}(x_i, y_i, z_i, w_i)$ are equal to the first and second column of M_i'.

It now follows that the test for largest sampling density can be done as

$$\begin{aligned} |\det[J(f_1, g_1)]| \quad &> \quad |\det[J(f_2, g_2)]| \Leftrightarrow \\ |\overrightarrow{n_1} \bullet (x_1, y_1, w_1)| \cdot |w_2^3| \quad &> \quad |\overrightarrow{n_2} \bullet (x_2, y_2, w_2)| \cdot |w_1^3|. \end{aligned} \tag{3}$$

An important subtlety to acknowledge is that (3) takes points of the form $(s, t, q, 1)^T$ as input. However, the inequality will work for a point even when the last component is not 1 since the effect of using it on such a point $(s', t', q', r')^T$ corresponds to applying the inequality for $(s'/r', t'/r', q'/r', 1)^T$ and scaling by r'^4 on both sides, which is redundant.

This means that given a transformation M_{obj} from some arbitrary object space to world space, we can transform and test points from this object space without having to make a stop in screen space to perform the divide. We do this by using the transformation sequence

$$M_i' \cdot C_{\text{proj}} \cdot C^{-1} \cdot M_{\text{obj}} = M_i \cdot M_{\text{obj}}$$

The vector $\overrightarrow{n_i}$ is still derived from M_i' and is constant. Now let us re-examine $M' = M_1'$ (the i will be omitted for now). We have $M' = M \cdot C \cdot C_{\text{proj}}^{-1}$, and let

$$A = M \cdot C = \begin{bmatrix} a & b & c & d \\ e & f & g & h \\ i & j & k & l \\ m & n & o & p \end{bmatrix},$$

$$C_{\text{proj}}^{-1} = \begin{bmatrix} s_x' & 0 & 0 & c_x' \\ 0 & s_y' & 0 & c_y' \\ 0 & 0 & 0 & -1 \\ 0 & 0 & k_1' & k_2' \end{bmatrix},$$

$$M' = A \cdot C_{\text{proj}}^{-1} = \begin{bmatrix} s_x'a & s_y'b & k_1'd & c_x'a + c_y'b - c + k_2'd \\ s_x'e & s_y'f & k_1'h & c_x'e + c_y'f - g + k_2'h \\ s_x'i & s_y'j & k_1'l & c_x'i + c_y'j - k + k_2'l \\ s_x'm & s_y'n & k_1'p & c_x'm + c_y'n - o + k_2'p \end{bmatrix}.$$

Given (2), we now have

$$\overrightarrow{n} = \overrightarrow{n_1} = s_x' \cdot s_y' \cdot \begin{pmatrix} a \\ e \\ m \end{pmatrix} \times \begin{pmatrix} b \\ f \\ n \end{pmatrix},$$

and after evaluation of the first term in (3), we get

$$\begin{aligned} |\overrightarrow{n} \bullet (x, y, w)| &= \left| \begin{bmatrix} \overrightarrow{n}_x & \overrightarrow{n}_y & 0 & \overrightarrow{n}_z \end{bmatrix} \cdot M' \cdot \begin{bmatrix} s \\ t \\ q \\ 1 \end{bmatrix} \right| \\ &= |s_x's_y' \cdot (\det(A_{33}) \cdot (k_1'q + k_2') - \det(A_{34}))|. \qquad (4) \end{aligned}$$

The syntax A_{ij} represents the 3×3 submatrix we obtain by removing the ith row and the jth column. Thus it is clear the term only depends on q.
In Section 1.2, the analysis will be focused on shadow mapping.

1.2 Shadow Mapping

In this section, *shadow map space* will be analogous to texture map space (or texture space) from Section 1.1.

For a given constellation of a camera and a light, we can evaluate a pair of projection matrices M_i and M_j from world space to the shadow map ($i, j \in \{1, 2, 3, 4, 5\}$ and $i \neq j$).

1. ssm—standard full light-frustum shadow mapping.
2. fsm—also standard but focused on the intersection between bounds.
3. tsm—trapezoidal shadow mapping.
4. lispsm—light-space perspective shadow mapping.
5. psm—perspective shadow mapping.

Let I be the intersection volume between the world bound, camera frustum, and light frustum. Generally for methods 2–5, the side planes and back plane of the light frustum are placed tightly around I. Subsequently the near plane is pushed forward until it reaches the intersection volume between this new light frustum and the world bound. This completes the new light frustum, let V denote its volume (see Figure 3). This process is what distinguishes fsm from ssm.

Theorem

For any such pair M_i and M_j, (3) is reduced to the point set on one side of a plane equation. Furthermore, the resulting plane contains the eye point of the light.

Proof

This will be shown through examination of $\det(A_{33})$ and $\det(A_{34})$ for each of the five methods. The proof will be divided into two parts. The first part will prove the theorem for the first four methods, and the second part will cover the last of the five methods (psm).

Part 1:

The first four methods have roughly the same transformation sequence:

$$A = N_t \cdot L_{\text{proj}} \cdot B.$$

The matrix B is the camera-to-light matrix. Note that our only assumption about B is that it has an inverse and that the bottom row is $\begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix}$:

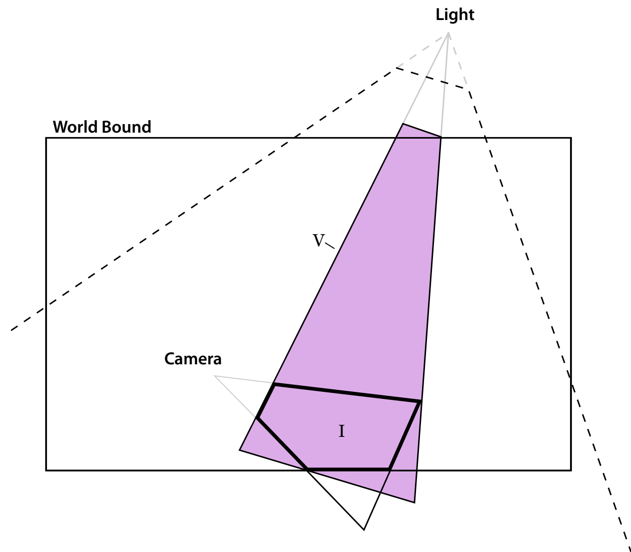


Figure 3: The process of focusing a light on the intersection volume I between bounds. The new light volume V is shown here in purple.

$$B = \begin{bmatrix} \overrightarrow{c1}_x & \overrightarrow{c2}_x & \overrightarrow{c3}_x & t_x \\ \overrightarrow{c1}_y & \overrightarrow{c2}_y & \overrightarrow{c3}_y & t_y \\ \overrightarrow{c1}_z & \overrightarrow{c2}_z & \overrightarrow{c3}_z & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

The matrix L_{proj} is the projection matrix into the light's post-projective space:

$$L_{proj} = \begin{bmatrix} r_x & 0 & l_x & 0 \\ 0 & r_y & l_y & 0 \\ 0 & 0 & o_1 & o_2 \\ 0 & 0 & -1 & 0 \end{bmatrix}.$$

The matrix N_t is a sequence of transformations that are designed to transform a trapezoidal bound around I (possibly a sheared one depending on the implementation) into a nice quadratic square as seen from the light. This principle is shared between tsm and lispsm. For ssm and fsm, we will simply consider the matrix N_t to be the identity matrix.

The steps involved in creating the transformation N_t are outlined in the **tsm** paper and will not be explained here in detail.

It will now be shown that the matrix N_t is of the form

$$N_t = \begin{bmatrix} k_{11} & k_{12} & 0 & k_{14} \\ k_{21} & k_{22} & 0 & k_{24} \\ k_{31} & k_{32} & k_{33} & k_{34} \\ k_{41} & k_{42} & 0 & k_{44} \end{bmatrix}.$$

Notice that the product of two such matrices results in a matrix of the same form. It just so happens that every matrix in the sequence of N_t is of this form:

$$N_t = T' \cdot S \cdot N_{\text{proj}} \cdot T \cdot R.$$

The matrices T' and T are translations, and S is a nonuniform scale. The matrices on the left side, T' and S, are not part of the original definition of N_t but have been added here to complete the transformation into unnormalized coordinates. The matrix R is a rotation (2D) around the z-axis, and N_{proj} is a projection along the y-axis:

$$N_{\text{proj}} = \begin{bmatrix} q_{11} & q_{12} & 0 & 0 \\ 0 & q_{22} & 0 & q_{24} \\ 0 & q_{32} & q_{33} & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}.$$

So they all comply, and thus N_t is compliant with the form. In the case of fsm or ssm, N_t is the identity, which is also compliant. We can now take advantage of the three zeros in the third column of N_t, the fourth column of L_{proj}, and the fourth row of B:

$$\begin{aligned} A_{33} &= N_{t33} \cdot L_{\text{proj}34} \cdot B_{43} \Rightarrow \det(A_{33}) = \det(N_{t33}) \cdot \det(L_{\text{proj}34}) \cdot \det(B_{43}), \\ A_{34} &= N_{t33} \cdot L_{\text{proj}34} \cdot B_{44} \Rightarrow \det(A_{34}) = \det(N_{t33}) \cdot \det(L_{\text{proj}34}) \cdot \det(B_{44}). \end{aligned}$$

Note that since B is the camera-to-light matrix, we can obtain the position of the light in camera space using $B^{-1} \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix}^T$, which after division by $w = 1$ results in $\ell = B_{44}^{-1} \begin{bmatrix} -t_x & -t_y & -t_z \end{bmatrix}^T$. It now follows that

$$\ell_z = \left(B_{44}^{-1} \begin{bmatrix} -t_x \\ -t_y \\ -t_z \end{bmatrix} \right)_z = \frac{-(\overrightarrow{c_1} \times \overrightarrow{c_2})^T}{\det(B_{44})} \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix} = \frac{-\det(B_{43})}{\det(B_{44})}.$$

And now we have $\det(A_{33}) = -\ell_z \cdot \det(A_{34})$, which by substitution will simplify (4) further:

$$\vec{n} \bullet (x, y, w) = -s'_x s'_y \det(A_{34}) \cdot (\ell_z (k'_1 q + k'_2) + 1)\,. \tag{5}$$

Now let M_1 and M_2 be two projection matrices evaluated by any of the first four methods, and let

$$\begin{aligned} A &= M_1 \cdot C, \\ A' &= M_2 \cdot C. \end{aligned}$$

We can now substitute (5) for A and A' into (3) and obtain the following:

$$|\det(A_{34})||w_2^3| > |\det(A'_{34})||w_1^3|.$$

In addition, we can take advantage of the fact that given any of the five listed shadow-mapping methods, we have $\forall p \in V : w(p) > 0$. Finally, inequality (3) is reduced to

$$|\sqrt[3]{\det(A_{34})}|w_2 - |\sqrt[3]{\det(A'_{34})}|w_1 > 0, \tag{6}$$

which is the point set entirely on one side of a plane equation. The fact that the plane contains the eye point of the light is a result of the following:

$$L_{\text{proj}} \cdot \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ o_2 \\ 0 \end{bmatrix}.$$

So $w = 0$ for ssm and fsm. Given the known form of N_t this property also holds for tsm and lispsm:

$$N_t \cdot \begin{bmatrix} 0 \\ 0 \\ o_2 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ k_{33}o_2 \\ 0 \end{bmatrix}.$$

Part 2:
In the second part, it will be shown that the proof still holds for the last method. The exact details of the psm transformation sequence will not be explained here, but the concept of the method is to set up the light projection in post-projective space of the camera, thereby increasing the number of shadow-map pixels available near the camera eye point.

To allow psm to support the pull-back value, instead of using the actual projection C_{proj}, an initial translation T', followed by a variant C'_{proj} ,is used. This allows the frustum to be enlarged, which gives you the freedom to tweak the xy-distribution as seen from the light.

For psm, the transformation sequence M from world space to shadow-map space is

$$M = L_{\text{proj}} \cdot Q \cdot C'_{\text{proj}} \cdot T' \cdot C^{-1},$$

and once again we evaluate A, that is, the transformation from camera space to shadow map space:

$$A = M \cdot C = L_{\text{proj}} \cdot Q \cdot C'_{\text{proj}} \cdot T'.$$

The matrix L_{proj} is simply a chosen projection matrix into shadow-map space, and the remaining matrix Q has the following form:

$$Q = \begin{bmatrix} \overrightarrow{r_1}_x & \overrightarrow{r_1}_y & \overrightarrow{r_1}_z & t'_x \\ \overrightarrow{r_2}_x & \overrightarrow{r_2}_y & \overrightarrow{r_2}_z & t'_y \\ \overrightarrow{r_3}_x & \overrightarrow{r_3}_y & \overrightarrow{r_3}_z & t'_z \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Compared to the first four methods, the transformation sequence is quite different, however, as it turns out; the same strategy as that of the first part can be used to complete the proof.

As previously mentioned, ℓ is the eye of the light in camera space. The position $\begin{bmatrix} t_x & t_y & t_z \end{bmatrix}^T$ is the same point transformed into the modified post-projective space of the camera, that is, $(C'_{\text{proj}} \cdot T') \begin{bmatrix} \ell_x & \ell_y & \ell_z & 1 \end{bmatrix}^T$ (after the divide). And additionally, $\begin{bmatrix} t'_x & t'_y & t'_z \end{bmatrix}^T = Q_{44} \begin{bmatrix} -t_x & -t_y & -t_z \end{bmatrix}^T$, so we can split up Q accordingly:

$$Q = \begin{bmatrix} 1 & 0 & 0 & t'_x \\ 0 & 1 & 0 & t'_y \\ 0 & 0 & 1 & t'_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \overrightarrow{r_1}_x & \overrightarrow{r_1}_y & \overrightarrow{r_1}_z & 0 \\ \overrightarrow{r_2}_x & \overrightarrow{r_2}_y & \overrightarrow{r_2}_z & 0 \\ \overrightarrow{r_3}_x & \overrightarrow{r_3}_y & \overrightarrow{r_3}_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \overrightarrow{r_1}_x & \overrightarrow{r_1}_y & \overrightarrow{r_1}_z & 0 \\ \overrightarrow{r_2}_x & \overrightarrow{r_2}_y & \overrightarrow{r_2}_z & 0 \\ \overrightarrow{r_3}_x & \overrightarrow{r_3}_y & \overrightarrow{r_3}_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & -t_x \\ 0 & 1 & 0 & -t_y \\ 0 & 0 & 1 & -t_z \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

The rows of Q_{44} are typically a reconstructed version of the light's orientation. It is, however, not a necessary limitation for this to work. We only demand that Q_{44} has an inverse (and that $\begin{bmatrix} t_x & t_y & t_z \end{bmatrix}^T$ is as previously described). The matrix L_{proj} is, as mentioned, a chosen projection matrix for the light with three appropriate zeros in the last column:

$$C'_{\text{proj}} = \begin{bmatrix} s_x & 0 & c_x & 0 \\ 0 & s_y & c_y & 0 \\ 0 & 0 & k_1 & k_2 \\ 0 & 0 & -1 & 0 \end{bmatrix}, \qquad T' = \begin{bmatrix} 1 & 0 & 0 & p_x \\ 0 & 1 & 0 & p_y \\ 0 & 0 & 1 & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$N = \begin{bmatrix} 1 & 0 & 0 & -t_x \\ 0 & 1 & 0 & -t_y \\ 0 & 0 & 1 & -t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot C'_{\text{proj}} \cdot T' = \begin{bmatrix} s_x & 0 & c_x + t_x & s_x p_x + p_z(c_x + t_x) \\ 0 & s_y & c_y + t_y & s_y p_y + p_z(c_y + t_y) \\ 0 & 0 & k_1 + t_z & k_2 + p_z(k_1 + t_z) \\ 0 & 0 & -1 & -p_z \end{bmatrix}.$$

We can now rewrite transformation A as

$$A = L_{\text{proj}} \cdot \begin{bmatrix} \overrightarrow{r_1}_x & \overrightarrow{r_1}_y & \overrightarrow{r_1}_z & 0 \\ \overrightarrow{r_2}_x & \overrightarrow{r_2}_y & \overrightarrow{r_2}_z & 0 \\ \overrightarrow{r_3}_x & \overrightarrow{r_3}_y & \overrightarrow{r_3}_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot N.$$

And similar to part 1, the distribution of zeros leads to

$$\begin{aligned}
A_{33} &= L_{\text{proj34}} \cdot Q_{44} \cdot N_{43} \Rightarrow \det(A_{33}) = \det(L_{\text{proj34}}) \cdot \det(Q_{44}) \cdot \det(N_{43}), \\
A_{34} &= L_{\text{proj34}} \cdot Q_{44} \cdot N_{44} \Rightarrow \det(A_{34}) = \det(L_{\text{proj34}}) \cdot \det(Q_{44}) \cdot \det(N_{44}).
\end{aligned}$$

Now when evaluating $(C'_{\text{proj}} \cdot T') \begin{bmatrix} \ell_x & \ell_y & \ell_z & 1 \end{bmatrix}^T$, we get $t_z = (-k_1) + \frac{-k_2}{p_z + \ell_z}$.

This is used for evaluation of $\det(N_{43})$ and $\det(N_{44})$:

$$\begin{aligned}
\det(N_{44}) &= s_x s_y \left(k_1 + t_z\right) = s_x s_y \left(\frac{-k_2}{p_z + \ell_z}\right), \\
\det(N_{43}) &= s_x s_y \left(k_2 + p_z(k_1 + t_z)\right) = s_x s_y \left(k_2 + p_z \left(\frac{-k_2}{p_z + \ell_z}\right)\right) \\
&= s_x s_y \left(\frac{k_2 \ell_z}{p_z + \ell_z}\right) = -\ell_z \cdot \det(N_{44}).
\end{aligned}$$

And again, it follows that $\det(A_{33}) = -\ell_z \cdot \det(A_{34})$, which once again leads to (5) and subsequently (6).

That $w = 0$ for the eye point of the light is clear since $\left(L_{\text{proj}} \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix}^T\right)_w = 0$, and subsequently the proof is complete.