Appendix A

Mathematics of the Shear-Warp Factorization

This appendix contains complete derivations of the shear-warp factorization for rendering a volume with either a parallel or a perspective projection. The derivations assume that we are given only an arbitrary four-by-four viewing transformation matrix and the dimensions of the volume to be rendered. The results of the derivations are the shear and warp matrix factors.

A.1 Coordinate Systems and Definitions

We will use the four coordinate systems illustrated in Figure A.1: object coordinates, standard object coordinates, sheared object coordinates (also called intermediate image coordinates), and image coordinates. All are right-handed coordinate systems.

The object coordinate system is the natural coordinate system of the volume data. The origin is located at one corner of the volume. The unit distance along each axis equals the length of one voxel along that axis. The axes are labeled x_o, y_o and z_o.

We form the standard object coordinate system by permuting the axes of the object coordinate system so that the principal viewing axis becomes the third coordinate axis. The principal viewing axis is the object-space axis that is most parallel to the viewing direction. The standard object coordinate system axes are labeled i, j and k, where k is the principal

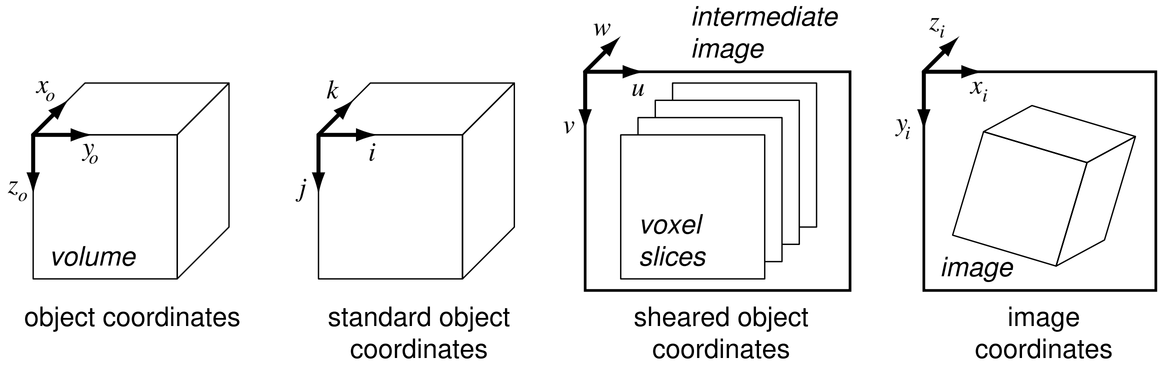


Figure A.1: Coordinate systems used in the derivation of the shear-warp factorization.

viewing axis.

We form the sheared object coordinate system by shearing the standard object coordinate system with the shear matrix from the shear-warp factorization. The sheared object coordinate system is also the coordinate system of the intermediate image. The origin is located at the upper-left corner of the intermediate image. The axes are labeled u, v and w.

The image coordinate system is the coordinate system of the final image. The warp matrix of the shear-warp factorization transforms sheared-object coordinates into image coordinates, and the original viewing transformation matrix transforms object coordinates into image coordinates. The origin of the image coordinate system is located at the upper-left corner of the image. The axes are labeled x_i, y_i and z_i.

In the following derivations $\vec{v}$ represents a vector and v_x represents a component of $\vec{v}$. The viewing transformation matrix is a four-by-four matrix M_{view} that transforms homogeneous points from object space to image space:

$$\begin{bmatrix} x_i \\ y_i \\ z_i \\ w_i \end{bmatrix} = M_{\text{view}} \begin{bmatrix} x_o \\ y_o \\ z_o \\ w_o \end{bmatrix}$$

All vectors are column vectors.

A.2 The Affine Factorization

In this section we derive a factorization for affine viewing transformations. Affine transformations include uniform and non-uniform scales, translations and shears used in parallel projections. The goal of the derivation is to factor an arbitrary affine viewing transformation matrix M_{view} as follows:

$$M_{\text{view}} = M_{\text{warp}} \cdot M_{\text{shear}}$$

M_{shear} is a shear matrix and M_{warp} is an affine transformation matrix. The matrix factors must be chosen such that we can project a volume from 3D to 2D between the shear and warp transformations and still produce a correct image.

A.2.1 Finding the Principle Viewing Axis

We define the principal viewing axis as the object-space axis that forms the smallest angle with the viewing direction vector.[1] In image space the viewing direction vector is:

$$\vec{v}_i = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

Let $\vec{v}_o$ be the viewing direction vector transformed to object space. It obeys the linear system of equations:

$$\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{bmatrix} \begin{bmatrix} v_{o,x} \\ v_{o,y} \\ v_{o,z} \end{bmatrix}$$

where the m_{ij} are elements of the viewing transformation matrix M_{view}. Only the upper-left 3x3 submatrix of M_{view} is necessary since $\vec{v}_i$ and $\vec{v}_o$ are vectors. In the remainder of this section let $M_{\text{view,3x3}}$ be the 3x3 submatrix.

[1] This definition is not equivalent to choosing the normal vector for the set of voxel slices most perpendicular to the viewing direction. If the viewing transformation includes non-uniform scaling then for some viewing directions this alternative definition results in larger (sub-optimal) shear coefficients.

By Cramer's Rule the solution to the linear system is:

$$v_{o,x} = \frac{\begin{vmatrix} 0 & m_{12} & m_{13} \\ 0 & m_{22} & m_{23} \\ 1 & m_{32} & m_{33} \end{vmatrix}}{|M_{\text{view,3x3}}|} = \frac{m_{12}m_{23} - m_{22}m_{13}}{|M_{\text{view,3x3}}|}$$

$$v_{o,y} = \frac{\begin{vmatrix} m_{11} & 0 & m_{13} \\ m_{21} & 0 & m_{23} \\ m_{31} & 1 & m_{33} \end{vmatrix}}{|M_{\text{view,3x3}}|} = \frac{m_{21}m_{13} - m_{11}m_{23}}{|M_{\text{view,3x3}}|}$$

$$v_{o,z} = \frac{\begin{vmatrix} m_{11} & m_{12} & 0 \\ m_{21} & m_{22} & 0 \\ m_{31} & m_{32} & 1 \end{vmatrix}}{|M_{\text{view,3x3}}|} = \frac{m_{11}m_{22} - m_{21}m_{12}}{|M_{\text{view,3x3}}|}$$

Since the denominator is the same for all three components of $\vec{v}_o$ it can be eliminated, yielding:

$$\vec{v}_o = \begin{bmatrix} m_{12}m_{23} - m_{22}m_{13} \\ m_{21}m_{13} - m_{11}m_{23} \\ m_{11}m_{22} - m_{21}m_{12} \end{bmatrix} \tag{A.1}$$

The cosine of the angle between the viewing direction and each object-space axis is proportional to the dot product of $\vec{v}_o$ with each of the object-space unit vectors. The largest dot product corresponds to the smallest angle. Thus we find the principal viewing axis by computing:

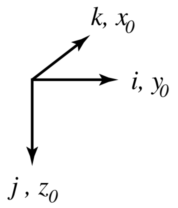
$$c = \max(|v_{o,x}|, |v_{o,y}|, |v_{o,z}|) \tag{A.2}$$

If $c = |v_{o,x}|$ then the principal viewing axis is the x_o axis. If $c = |v_{o,y}|$ then the principal viewing axis is the y_o axis. Otherwise, the principal viewing axis is the z_o axis.

A.2.2 Transformation to Standard Object Coordinates

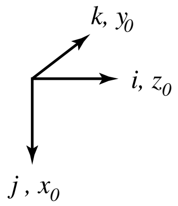
The shear-warp rendering algorithm operates by resampling and compositing the set of voxel slices that is perpendicular to the principal viewing axis. To eliminate special cases for each of the three axes in the remainder of the derivation we first transform the volume into standard object coordinates. If the principal viewing axis is the x_o axis then the permutation matrix is:

$$P = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



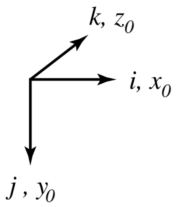
The diagram shows the correspondence between axes in standard object coordinates (i, j, k) and object coordinates (x_o, y_o, z_o). If the principal viewing axis is the y_o axis then the permutation matrix is:

$$P = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



If the principal viewing axis is the z_o axis then the permutation matrix is the identity matrix:

$$P = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



The transformation from object coordinates to standard object coordinates is:

$$\begin{bmatrix} i \\ j \\ k \\ 0 \end{bmatrix} = P \cdot \begin{bmatrix} x_o \\ y_o \\ z_o \\ 0 \end{bmatrix}$$

Now let M'_{view} be a permuted viewing transformation matrix:

$$M'_{\text{view}} = M_{\text{view}} P^{-1} \tag{A.3}$$

This matrix transforms points from standard object space into image space. In standard object space the k axis is always the principal viewing axis.

A.2.3 The Shear and Warp Factors

The next step in the derivation is to factor the permuted viewing transformation M'_{view} into a shear in the i and j directions (M'_{shear}) followed by an affine warp (M'_{warp}). The factorization must satisfy the following condition: after the shear transformation the viewing direction must be perpendicular to the (i, j) plane.

In standard object space the viewing direction vector is:

$$\vec{v}_{so} = P \cdot \vec{v}_o = \begin{bmatrix} m'_{12}m'_{23} - m'_{22}m'_{13} \\ m'_{21}m'_{13} - m'_{11}m'_{23} \\ m'_{11}m'_{22} - m'_{21}m'_{12} \end{bmatrix}$$

where the m'_{ij} are elements of the permuted viewing transformation matrix M'_{view}. The projection of the direction vector onto the (i, k) plane has a slope of $v_{so,i}/v_{so,k}$ (Figure A.2). The shear necessary in the i direction to make the viewing direction perpendicular to the constant-k planes is the negative of this slope. A similar argument holds for the shear in the j direction. Thus the shear coefficients are:

$$s_i = -\frac{v_{so,i}}{v_{so,k}} = \frac{m'_{22}m'_{13} - m'_{12}m'_{23}}{m'_{11}m'_{22} - m'_{21}m'_{12}} \tag{A.4}$$

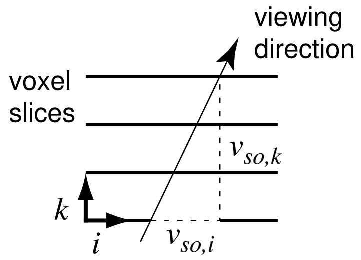


Figure A.2: Determining the shear coefficients: This diagram shows a cross-section of the volume and the viewing direction in standard object space. To transform the volume to sheared object space we must shear the volume in the i direction by $-v_{so,i}/v_{so,k}$. This shear causes the viewing direction to become perpendicular to the slices of the volume.

$$s_j = -\frac{v_{so,j}}{v_{so,k}} = \frac{m'_{11}m'_{23} - m'_{21}m'_{13}}{m'_{11}m'_{22} - m'_{21}m'_{12}} \tag{A.5}$$

We can now write the shear-warp factorization of the permuted viewing transformation matrix:

$$\begin{aligned} M'_{\text{view}} &= M'_{\text{view}} \begin{bmatrix} 1 & 0 & -s_i & 0 \\ 0 & 1 & -s_j & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & s_i & 0 \\ 0 & 1 & s_j & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} m'_{11} & m'_{12} & (m'_{13} - s_i m'_{11} - s_j m'_{12}) & m'_{14} \\ m'_{21} & m'_{22} & (m'_{23} - s_i m'_{21} - s_j m'_{22}) & m'_{24} \\ m'_{31} & m'_{32} & (m'_{33} - s_i m'_{31} - s_j m'_{32}) & m'_{34} \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & s_i & 0 \\ 0 & 1 & s_j & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned}$$

The left factor is an affine warp matrix and the right factor is a shear matrix. We have constructed the two matrices such that after we apply the shear transformation to a volume we may compute a 2D projection by compositing voxel slices with constant k into the slice located at $k = 0$. The composited slices form the intermediate image. We can then apply the warp transformation in 2D.

A.2.4 Projection to the Intermediate Image

Finally, we must define a coordinate system for the intermediate image. If we apply the shear transformation that we just derived to the standard object coordinate system the result is a new "sheared" coordinate system. However this coordinate system is not convenient because the origin is not located at a corner of the intermediate image. We therefore translate the sheared coordinate system to reposition the origin at the top-left corner of the intermediate image. The shear transformation followed by the translation produces the intermediate image coordinate system.

Figure A.3 illustrates four possible cases and the corresponding formulae for computing the required translation. Each case is identified by the signs of the two shear coefficients.

The rendering algorithm requires one additional piece of information to compute the intermediate image: the stacking order of the voxel slices in the intermediate image. The voxel slices are ordered by their k coordinate in standard object space. A front-to-back traversal of the voxel slices can correspond to looping through the slices in increasing order or in decreasing order depending on the viewing direction. We find the stacking order by examining the component of the viewing direction vector corresponding to the principal viewing axis, $v_{so,k}$. If $v_{so,k}$ is positive then the voxel slice in the $k = 0$ plane is the front slice; otherwise the slice in the $k = k_{\max}$ plane is the front slice.

Once we have chosen the translation we can rewrite the shear and warp matrix factors. The shear factor, which transforms standard object coordinates to intermediate image coordinates, equals:

$$M_{\text{shear}} = \begin{bmatrix} 1 & 0 & 0 & t_i \\ 0 & 1 & 0 & t_j \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & s_i & 0 \\ 0 & 1 & s_j & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & s_i & t_i \\ 0 & 1 & s_j & t_j \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{A.6}$$

The warp factor, which transforms intermediate image coordinates to image coordinates,

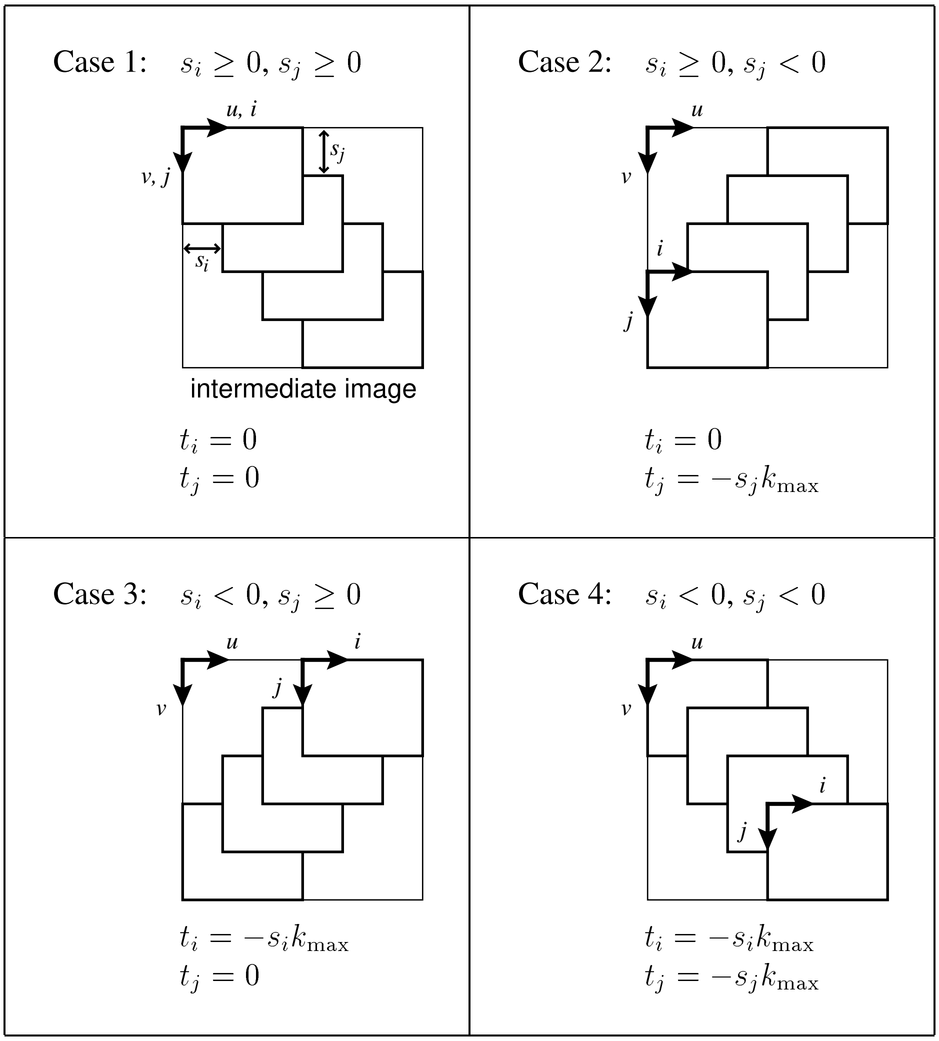


Figure A.3: Definition of the intermediate image coordinate system. The signs of the shear coefficients (s_i, s_j) distinguish each case. The translation (t_i, t_j) specifies the displacement from the origin of the standard object coordinate system $(i = 0, j = 0)$ to the origin of the intermediate image coordinate system $(u = 0, v = 0)$. $k_{\max}$ is the maximum voxel slice index in the volume.

equals:

$$M_{\text{warp}} = \begin{bmatrix} m'_{11} & m'_{12} & (m'_{13} - s_i m'_{11} - s_j m'_{12}) & m'_{14} \\ m'_{21} & m'_{22} & (m'_{23} - s_i m'_{21} - s_j m'_{22}) & m'_{24} \\ m'_{31} & m'_{32} & (m'_{33} - s_i m'_{31} - s_j m'_{32}) & m'_{34} \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & -t_i \\ 0 & 1 & 0 & -t_j \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Since we perform the warp in 2D we can delete the third row and third column of M_{warp}:

$$M_{\text{warp2D}} = \begin{bmatrix} m'_{11} & m'_{12} & (m'_{14} - t_i m'_{11} - t_j m'_{12}) \\ m'_{21} & m'_{22} & (m'_{24} - t_i m'_{21} - t_j m'_{22}) \\ 0 & 0 & 1 \end{bmatrix} \tag{A.7}$$

This is a 3x3 warp matrix that transforms the 2D intermediate image to the 2D final image:

$$\begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} = M_{\text{warp2D}} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}$$

A.2.5 The Complete Affine Factorization

In summary, the shear-warp factorization of an arbitrary affine viewing transformation matrix includes a permutation, a 3D shear, and a 2D warp:

$$M_{\text{view}} = M_{\text{warp}} M_{\text{shear}} P$$

To compute the factorization we perform the following steps:

1. Find the principal viewing axis (Equations A.1 and A.2) and choose the corresponding permutation matrix P.

2. Compute the permuted view matrix from M_{view} and P (Equation A.3).

3. Compute the shear coefficients (Equations A.4 and A.5) from the permuted view matrix.

4. Compute the translation between the origins of standard object coordinates and intermediate image coordinates (Figure A.3).

5. Compute the shear matrix M_{shear} (Equation A.6) and the warp matrix M_{warp2D} (Equation A.7).

The rendering algorithm uses the principal viewing axis to choose the set of voxel slices to resample and composite. The sign of $v_{so,k}$ determines which ordering of the slices results in a front-to-back traversal of the volume. The algorithm computes the location of each voxel slice in the intermediate image using M_{shear} and warps the intermediate image using M_{warp2D}.

A.3 The Perspective Factorization

In this section we derive a factorization for perspective viewing transformations. As in the previous derivation, the goal is to factor an arbitrary viewing transformation matrix M_{view} as follows:

$$M_{\text{view}} = M_{\text{warp}} \cdot M_{\text{shear}}$$

In this case M_{shear} specifies both shearing and perspective scaling transformations and M_{warp} is a perspective transformation matrix. The perspective shear-warp factorization reduces to the affine shear-warp factorization when the view transformation matrix is affine.

A.3.1 Finding the Principle Viewing Axis

Choosing a principal viewing axis is not as straightforward in the case of a perspective projection because the viewing rays are not parallel, so it may not be possible to choose a single consistent principal axis.

The first step is to find the location of the eye (the center of projection). A perspective viewing transformation must transform the object-space location of the eye ($\vec{e}_o$) to the image-space point with homogeneous coordinates $(0, 0, -1, 0)$. Thus the location of the eye can be found by solving the linear system:

$$\begin{bmatrix} 0 \\ 0 \\ -1 \\ 0 \end{bmatrix} = M_{\text{view}} \cdot \begin{bmatrix} e_{o,x} \\ e_{o,y} \\ e_{o,z} \\ e_{o,w} \end{bmatrix}$$

M_{view} must be a perspective transformation matrix, not a perspective projection matrix, so that the linear system is non-singular.

Using Cramer's Rule, the solution is:

$$e_{o,x} = \frac{\begin{vmatrix} 0 & m_{12} & m_{13} & m_{14} \\ 0 & m_{22} & m_{23} & m_{24} \\ -1 & m_{32} & m_{33} & m_{34} \\ 0 & m_{42} & m_{43} & m_{44} \end{vmatrix}}{|M_{\text{view}}|} = \frac{-\begin{vmatrix} m_{12} & m_{13} & m_{14} \\ m_{22} & m_{23} & m_{24} \\ m_{42} & m_{43} & m_{44} \end{vmatrix}}{|M_{\text{view}}|} \tag{A.8}$$

$$e_{o,y} = \frac{\begin{vmatrix} m_{11} & 0 & m_{13} & m_{14} \\ m_{21} & 0 & m_{23} & m_{24} \\ m_{31} & -1 & m_{33} & m_{34} \\ m_{41} & 0 & m_{43} & m_{44} \end{vmatrix}}{|M_{\text{view}}|} = \frac{\begin{vmatrix} m_{11} & m_{13} & m_{14} \\ m_{21} & m_{23} & m_{24} \\ m_{41} & m_{43} & m_{44} \end{vmatrix}}{|M_{\text{view}}|} \tag{A.9}$$

$$e_{o,z} = \frac{\begin{vmatrix} m_{11} & m_{12} & 0 & m_{14} \\ m_{21} & m_{22} & 0 & m_{24} \\ m_{31} & m_{32} & -1 & m_{34} \\ m_{41} & m_{42} & 0 & m_{44} \end{vmatrix}}{|M_{\text{view}}|} = \frac{-\begin{vmatrix} m_{11} & m_{12} & m_{14} \\ m_{21} & m_{22} & m_{24} \\ m_{41} & m_{42} & m_{44} \end{vmatrix}}{|M_{\text{view}}|} \tag{A.10}$$

$$e_{o,w} = \frac{\begin{vmatrix} m_{11} & m_{12} & m_{13} & 0 \\ m_{21} & m_{22} & m_{23} & 0 \\ m_{31} & m_{32} & m_{33} & -1 \\ m_{41} & m_{42} & m_{43} & 0 \end{vmatrix}}{|M_{\text{view}}|} = \frac{\begin{vmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{41} & m_{42} & m_{43} \end{vmatrix}}{|M_{\text{view}}|} \tag{A.11}$$

Since $\vec{e}_o$ represents a point in homogeneous coordinates the four components may be multiplied by the constant $|M_{\text{view}}|$, thereby eliminating the denominator.

The next step is to form vectors from the eyepoint to each corner of the volume and to find the principal viewing direction for each of the eight directions. These directions form corners of the bounding box for all viewing rays that hit the volume. For each vector we use the same procedure as for an affine viewing direction to find the corresponding principal viewing axis.

Let the location of a corner of the volume in object space be $\vec{p}_o$. The vector from the eye to the corner is $\vec{p}_o - \vec{e}_o$. Substituting this for $\vec{v}_o$ in Equation A.2 (from the affine derivation), we get:

$$c = \max(|p_{o,x} - e_{o,x}|, |p_{o,y} - e_{o,y}|, |p_{o,z} - e_{o,z}|) \tag{A.12}$$

The axis corresponding to the largest component is the principal viewing axis for the particular viewing ray. We repeat the procedure for each corner of the volume. If the eye point coincides with any corner then that corner should be omitted.

If the principal viewing axis is not the same for all eight corners then the volume must be subdivided as described in Section 4.2.5. We then proceed with the factorization for each of the required principal viewing directions.

A.3.2 Transformation to Standard Object Coordinates

The transformation from object coordinates to standard object coordinates is the same as in the affine derivation (Section A.2.2): we choose one of three permutation matrices based on the principal viewing axis. If the location of the eye lies in the $k = 0$ plane of the volume then an additional transformation T_{shift} that translates the origin of the volume must also be applied (to avoid division by zero later in the derivation). The location of the origin is arbitrary, so it could be shifted along the k axis to the opposite side of the volume for instance.

Once again we define a permuted viewing transformation matrix M'_{view} that transforms points from standard object space to image space:

$$M'_{\text{view}} = M_{\text{view}} P^{-1} T^{-1}_{\text{shift}} \tag{A.13}$$

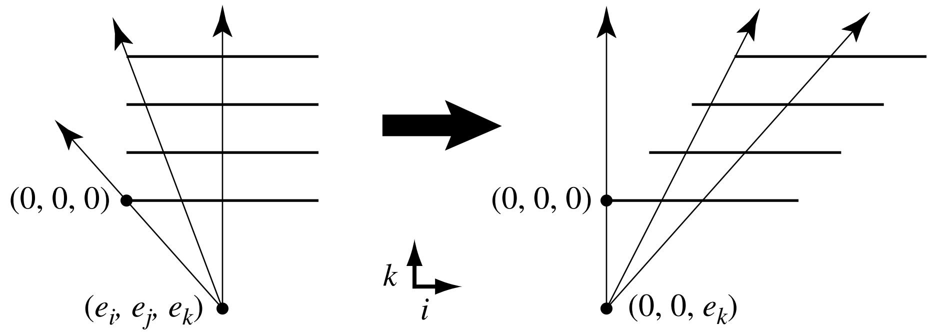


Figure A.4: The shear transformation in the perspective shear-warp factorization.

A.3.3 The Shear and Warp Factors

Next we factor M'_{view}. The "shear" factor (M'_{shear}) contains both a shear transformation and a transformation that scales each slice. The warp factor (M'_{warp}) is a perspective warp. The factorization must satisfy the following condition: after the shear and scale transformations the viewing direction must be perpendicular to the (i, j) plane.

The shear component of M'_{shear} shears the volume so that the ray from the eye through the origin of standard object space becomes perpendicular to the slices of the volume (Figure A.4). The location of the eye in standard object space is:

$$\vec{e}_s = T_{\text{shift}} P \cdot \vec{e}_o$$

For convenience, define the homogenized eye coordinates as follows:

$$\begin{aligned} e_i &= e_{s,i}/e_{s,w} \\ e_j &= e_{s,j}/e_{s,w} \\ e_k &= e_{s,k}/e_{s,w} \end{aligned}$$

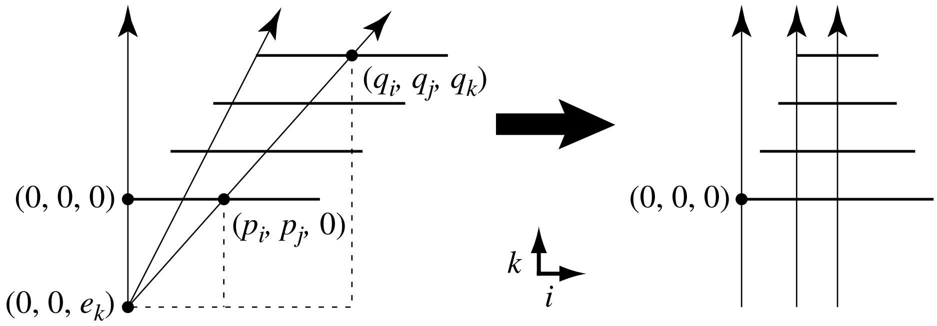


Figure A.5: The scale transformation in the perspective shear-warp factorization.

By inspection the required shear transformation is:

$$M_{\text{sh}} = \begin{bmatrix} 1 & 0 & -\frac{e_i}{e_k} & 0 \\ 0 & 1 & -\frac{e_j}{e_k} & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The translation specified by T_{shift} guarantees that e_k is nonzero.

The scale component of M_{shear} scales the slices of the volume so that the sheared viewing rays become parallel. Thus the required transformation is a pure perspective transformation (Figure A.5).

Without loss of generality we arbitrarily choose to scale the voxel slice in the $k = 0$ plane by a factor of one. Now consider a particular viewing ray passing through point $\vec{p}$ in the $k = 0$ plane and point $\vec{q}$ in some other plane (Figure A.5, left side). The perspective transformation must scale the voxel slice containing $\vec{q}$ by p_i/q_i in the i direction and p_j/q_j in the j direction. Using the similar triangles,

$$\frac{p_i}{q_i} = \frac{-e_k}{q_k - e_k} = \frac{1}{1 - q_k/e_k}$$

The value for p_j/q_j equals the same expression. Thus the required perspective transformation is:

$$M_{\mathrm{s}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -\frac{1}{e_k} & 1 \end{bmatrix}$$

The complete perspective shear factor is the product of the pure shear and the perspective scaling transformations:

$$M'_{\mathrm{shear}} = M_{\mathrm{s}} \cdot M_{\mathrm{sh}} = \begin{bmatrix} 1 & 0 & -\frac{e_i}{e_k} & 0 \\ 0 & 1 & -\frac{e_j}{e_k} & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -\frac{1}{e_k} & 1 \end{bmatrix}$$

Substituting the homogeneous coordinates of the eye location we obtain:

$$M'_{\mathrm{shear}} = M_{\mathrm{s}} \cdot M_{\mathrm{sh}} = \begin{bmatrix} 1 & 0 & -\frac{e_{o,i}}{e_{o,k}} & 0 \\ 0 & 1 & -\frac{e_{o,j}}{e_{o,k}} & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -\frac{e_{o,w}}{e_{o,k}} & 1 \end{bmatrix} \tag{A.14}$$

The warp transformation equals the permuted view transformation multiplied by the inverse of the shear transformation:

$$M'_{\mathrm{warp}} = M'_{\mathrm{view}} \cdot M'^{-1}_{\mathrm{shear}} = M'_{\mathrm{view}} \cdot \begin{bmatrix} 1 & 0 & \frac{e_{o,i}}{e_{o,k}} & 0 \\ 0 & 1 & \frac{e_{o,j}}{e_{o,k}} & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & \frac{e_{o,w}}{e_{o,k}} & 1 \end{bmatrix}$$

A.3.4 Projection to the Intermediate Image

The shear matrix we just derived transforms standard object space into a deformed space. The final step in the derivation is to define a transformation from this deformed coordinate system to the intermediate image coordinate system. First we choose a uniform scale so that the front-most voxel slice is scaled by a factor of one. Then, as in the affine derivation, we choose a translation to position the origin of the intermediate image coordinate system at the top-left corner of the intermediate image.

The shear matrix M'_{shear} we derived in the previous section guarantees that the voxel slice located in the $k = 0$ plane has a scale factor of one. However that slice is not necessarily the front voxel slice, depending on the stacking order of the voxel slices in the intermediate image. The sign of the largest component of $\vec{p}_o - \vec{e}_o$ determines the stacking order: if the sign is positive then the voxel slice in the $k = 0$ plane is the front slice; otherwise the slice in the $k = k_{\max}$ plane is the front slice.

If the front slice is in the $k = 0$ plane then the scale factor for the front slice after applying M'_{shear} is already one, so no additional scaling is necessary. However, if the front slice is in the $k = k_{\max}$ plane then M'_{shear} results in a scale factor of:

$$f = \frac{1}{1 - \frac{e_{o,w}}{e_{o,z}} k_{\max}}$$

This scaling must be inverted by the transformation to intermediate image coordinates. The required scaling matrix is:

$$S = \begin{bmatrix} f' & 0 & 0 & 0 \\ 0 & f' & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{A.15}$$

where

$$f' = \begin{cases} 1 & \text{if slice } k = 0 \text{ is in front} \\ 1 - \frac{e_{o,w}}{e_{o,z}} k_{\max} & \text{otherwise} \end{cases}$$

To choose the translation, we transform the corners of the front and rear voxel slices by $S \cdot M_{\text{shear}}$. From the transformed coordinates we find the bounding box of the volume after transformation and we choose the translation $T(t_i, t_j, 0)$ that positions the upper-left corner of the bounding box at the origin.

Combining all of the matrix factors, we arrive at the final shear-warp factorization for perspective projections. The shear factor equals:

$$M_{\text{shear}} = T \cdot S \cdot \begin{bmatrix} 1 & 0 & \frac{e_{o,x}}{e_{o,z}} & 0 \\ 0 & 1 & \frac{e_{o,y}}{e_{o,z}} & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & \frac{e_{o,w}}{e_{o,z}} & 1 \end{bmatrix} \tag{A.16}$$

The warp factor equals:

$$M_{\text{warp}} = M'_{\text{view}} \cdot M_{\text{shear}}^{-1} \cdot S^{-1} \cdot T^{-1} \tag{A.17}$$

As in the affine derivation we can delete the third row and third column of M_{warp} to produce a 2D warp since the intermediate image is located in the $k = 0$ plane and the z_i coordinate is not used.

A.3.5 The Complete Perspective Factorization

In summary, the shear-warp factorization of an arbitrary perspective viewing transformation matrix includes a permutation, a shift of the origin (if the eye point lies in the $k = 0$ slice of the volume), a 3D shear/perspective scale, and a 2D warp:

$$M_{\text{view}} = M_{\text{warp}} M_{\text{shear}} T_{\text{shift}} P$$

To compute the factorization we perform the following steps:

1. Find the location of the eye in object space (Equations A.8–A.11).

2. Find the principal viewing axis (Equation A.12) and choose the corresponding permutation matrix P. If necessary, split the volume into sections with different principal viewing axes.

3. Determine T_{shift} and compute the permuted view matrix from M_{view} and P (Equation A.13).

4. Compute the shear and perspective scale coefficients from the eye coordinates (Equation A.14).

5. Compute the scale and translation that complete the transformation to intermediate image coordinates (Equation A.15).

6. Compute the shear matrix M_{shear} (Equation A.16) and the warp matrix M_{warp} (Equation A.17).

The rendering algorithm uses the principal viewing axis to choose the set of voxel slices to resample and composite. The sign of the largest component computed in Equation A.12 determines which ordering of the slices results in a front-to-back traversal of the volume. The algorithm computes the location of each voxel slice in the intermediate image and the slice scale factor using M_{shear} and warps the intermediate image using M_{warp}.