Chapter 1

Geometry of Image Formation

There are two fundamental questions related to image formation:

- Where is a point in the world imaged?
- How bright is the resulting image point?

We start with the first question, for which it is adequate to use the **pinhole camera** model. Historically, this originated with the camera obscura. That the image was inverted confused people for quite some time and delayed the application of this model to image formation in the retina. Finally, Kepler in 1604, Descartes in 1620s experimentally showed that the image really is inverted, there was no system of mirrors or lenses in the eye that made the image the right side up.

An understanding of the basic mathematics of perspective precedes Kepler and Descartes. It goes back to Euclid, Alhazen, and of course the painters of the Italian Renaissance. While credit for the first artistic creations is given to Brunelleschi and Masaccio, the first formal statement of the principles is usually attributed to Alberti (1435).

1.1 Perspective Projection

A pinhole camera consists of a pinhole opening, O, at the front of a box, and an image plane at the back of the box , see Figure 1.1. We will use a three-dimensional coordinate system with the origin at O and will consider a point P in the scene, with coordinates (X, Y, Z). P gets projected to the point P' in the image plane with coordinates (x, y, z). If f is the distance from the pinhole to the image plane, then by similar triangles, we can derive the following equations:

$$\frac{-x}{f} = \frac{X}{Z}, \; \frac{-y}{f} = \frac{Y}{Z} \quad \Rightarrow \quad x = \frac{-fX}{Z}, \; y = \frac{-fY}{Z} \; .$$

These equations define an image formation process known as perspective projection. Note that the Z in the denominator means that the farther away an object is, the smaller its image will be. Also, note that the minus signs mean that the image is *inverted*, both left–right and up–down, compared with the scene.

Equivalently, we can model the perspective projection process with the projection plane being at a distance f in *front* of the pinhole. This device of imagining a projection surface in front was first recommended to painters in the Italian Renaissance by Alberti in 1435 as a technique for constructing geometrically accurate depictions of a three-dimensional scene. For our purposes, the main advantage of this model is that it avoids lateral inversion and thereby eliminates the negative signs in the perspective projection equations. Note that it isn't essential that the projection surface be a plane, it could equally well be a sphere centered at the pinhole. The key aspect here is the 1-1 mapping from rays through the pinhole to points on a projection surface.

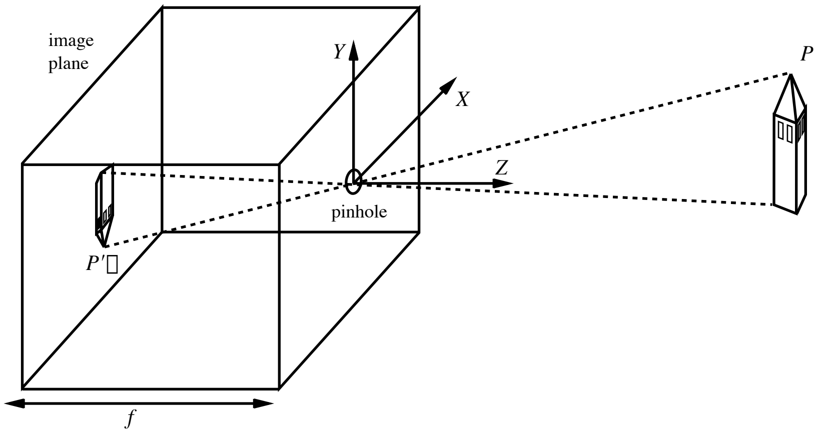

Figure 1.1: Pinhole Camera.

Canonically, in spherical perspective, the projection surface is a sphere of unit radius ("viewing sphere") centered at the center of projection. A point (ρ, θ, ϕ) gets mapped to $(1, \theta, \phi)$. The ray from the point in the scene through the center of projection is perpendicular to the imaging surface. Spherical perspective avoids an artifact of plane perspective known as the "position effect".

Let's summarize the preceding discussion in vector notation. A world point $\mathbf{X}$ projects to a point on a plane, $\mathbf{p}$, under planar projection, and equivalently a point on a sphere, $\mathbf{q}$ under spherical projection:

$$\begin{aligned}\mathbf{p} &= f\frac{\mathbf{X}}{Z} \\ \mathbf{q} &= \frac{\mathbf{X}}{\|\mathbf{X}\|}\end{aligned}$$

where $\mathbf{p} = (x, y, f)$, and $\mathbf{q}$ is a unit vector. Note, precisely the same information is represented in each case - namely the ray *direction* which is the most that can be recovered from an image point. It is straightforward to transform between $\mathbf{p}$ and $\mathbf{q}$: $\mathbf{q} = \mathbf{p}/||\mathbf{p}||$, $\mathbf{p} = f\mathbf{q}/q_z$. The connection between the vector notation and a planar image is that if $\mathbf{p} = (x, y, f)$, then $\mathbf{x} = (x, y)$.

1.2 Projection of lines

A line of points in 3D can be represented as $\mathbf{X} = \mathbf{A} + \lambda\mathbf{D}$, where $\mathbf{A}$ is a fixed point, $\mathbf{D}$ a unit vector parallel to the line, and λ a measure of distance along the line. As λ increases points are increasingly further away and in the limit:

$$\lim_{\lambda\to\infty} \mathbf{p} = f\frac{\mathbf{A} + \lambda\mathbf{D}}{A_Z + \lambda D_Z} = f\frac{\mathbf{D}}{D_Z}$$

i.e. the image of the line terminates in a *vanishing point* with coordinates $(fD_X/D_Z, fD_Y/D_Z)$, unless the line is parallel to the image plane ($D_Z = 0$). Note, the vanishing point is unaffected (invariant to) line position, $\mathbf{A}$, it only depends on line orientation, $\mathbf{D}$. Consequently, the family of lines parallel to $\mathbf{D}$ have the same vanishing point.

Under spherical perspective, a line in the scene projects to half of a great circle. This circle is defined by the intersection of the viewing sphere with the plane containing the line and center of projection. There are two vanishing points here corresponding to the endpoints of the half great circle. You should convince yourself that these are the same for a family of parallel straight lines.

1.3 Projection of planes

A plane of points in 3D can be represented as $\mathbf{X}.\mathbf{N} = d$ where $\mathbf{N}$ is the unit plane normal, and d the perpendicular distance of the plane from the origin. A point $\mathbf{X}$ on the plane is imaged at $\mathbf{p} = f\mathbf{X}/Z$. Taking the scalar product of both sides with $\mathbf{N}$ gives $\mathbf{p}.\mathbf{N} = f\mathbf{X}.\mathbf{N}/Z = fd/Z$. In the limit of points very distant:

$$\lim_{Z\to\infty} \mathbf{p}.\mathbf{N} = 0$$

which is the equation of a plane through the origin parallel to the world plane (i.e. which has the same normal $\mathbf{N}$). The plane $\mathbf{p}.\mathbf{N} = 0$ intersects the image plane in a *vanishing line*

$$xN_x + yN_y + fN_z = 0$$

Note, the vanishing line is unaffected (invariant to) plane position, d, it only depends on plane orientation, $\mathbf{N}$. All planes with the same orientation have the same vanishing line, also called the horizon.

Consider a line on the plane. It can be shown (exercise) that the vanishing points of all lines on the plane lie on the vanishing line of the plane. Thus, two vanishing points determine the vanishing line of the plane.

Under spherical perspective, the horizon of a plane is a great circle, found by translating the plane parallel to itself until it passes through the center of projection, and then intersecting it with the viewing sphere.

1.4 Terrestrial Perspective

Consider an observer standing on a ground plane looking straight ahead of her. Since the ground plane has surface normal $\mathbf{N} = (0, 1, 0)$, the equation of the horizon is $y = 0$. In this canonical case, the horizon lies in the middle of the field of view, with the ground plane in the lower half and the sky in the upper half.

Let us work out how objects of different heights and at different locations on the ground plane project. We will suppose that the eye, or camera, is a height h_c above the ground plane. Consider an object of height δY resting on the ground plane, whose bottom is at $(X, -h_c, Z)$ and top is at $(X, \delta Y - h_c, Z)$. The bottom projects to $(fX/Z, -fh_c/Z)$ and the top to $(fX/Z, f(\delta Y - h_c)/Z)$.

We note the following:

1. The bottoms of nearer objects (small Z) project to points lower in the image plane, farther objects have bottoms closer to the horizon.

2. If the object has the same height as the camera ($\delta Y = h_c$), the projection of its top lies on the horizon.

3. The ratio of the height of the object to the height of the camera, $\delta Y/h_c$ is the ratio of the apparent vertical height of the object in the image to the vertical distance of its bottom from the horizon (Verify).

1.5 Orthographic Projection

If the object is relatively shallow compared with its distance from the camera, we can approximate perspective projection by scaled orthographic projection. The idea is as follows: If the depth Z of points on the object varies within some range $Z_0 \pm \Delta Z$, with $\Delta Z \ll Z_0$, then the perspective scaling factor f/Z

can be approximated by a constant $s = f/Z_0$. The equations for projection from the scene coordinates (X, Y, Z) to the image plane become $x = sX$ and $y = sY$. Note that scaled orthographic projection is an approximation that is valid only for those parts of the scene with not much internal depth variation; it should not be used to study properties "in the large." For instance, under orthographic projection, parallel lines stay parallel instead of converging to a vanishing point!

1.6 Summary

- Plane perspective
$$(X, Y, Z) \mapsto (\frac{fX}{Z}, \frac{fY}{Z}, f) \tag{1.1}$$
- Spherical perspective
$$(X, Y, Z) \mapsto (\frac{X, Y, Z}{\sqrt{X^2 + Y^2 + Z^2}}) \tag{1.2}$$
- Lines $\mapsto$ vanishing points
$$A + \lambda D \mapsto (\frac{fD_x}{D_z}, \frac{fD_y}{D_z}) \tag{1.3}$$
- Planes $\mapsto$ vanishing lines (horizons)
$$X \cdot N = d \mapsto xN_x + yN_y + fN_z = 0 \tag{1.4}$$

1.7 Exercises

1. Show that the vanishing points of lines on a plane lie on the vanishing line of the plane.

2. Show that, under typical conditions, the silhouette of a sphere of radius r with center $(X$,0, $Z)$ under planar perspective projection is an ellipse of eccentricity $X/\sqrt{(}X^2 + Z^2 - r^2)$. Are there circumstances under which the projection could be a parabola or hyperbola? What is the silhouette for spherical perspective?

3. An observer is standing on a ground plane looking straight ahead. We want to calculate the accuracy with which she will be able to estimate the depth Z of points on the ground plane, assuming that she can visually discriminate angles to within $1'$. Derive a formula relating depth error δZ to Z. For simplicity, just consider points straight ahead of the observer($x = 0$). Given a Z value (say 10 m), your formula should be able to predict the δZ.

Chapter 2

Pose, Shape and Geometric Transformations

Points on an object can be characterized by their 3D coordinates with respect to the camera coordinate system. But what happens, when we move the object? In a certain sense when a chair is moved in 3D space, it remains the "same" even though the coordinates of points on it with respect to the camera (or any fixed) coordinate system do change. This distinction is captured by the use of the terms **pose** and **shape**.

- *Pose:* The position and orientation of the object with respect to the camera. This is specified by 6 numbers (3 for its translation, 3 for rotation). For example, we might consider the coordinates of the centroid of the object relative to the center of projection, and the rotation of a coordinate frame on the object with respect to that of the camera.
- *Shape:* The coordinates of the points of an object relative to a coordinate frame on the object. These remain invariant when the object undergoes rotations and translations.

To make these notions more precise, we need to develop the basic theory of **Euclidean Transformations**. The set of transformations defines a notion of "congruence" or having the same shape. In high school geometry we learned that two planar triangles are congruent if one of them can be rotated and translation so as to lie exactly on top of another. Rotation and translation are examples of Euclidean transformations, also known as **isometries** or **rigid body motions**, defined as transformation that preserve distances between any pair of points. When I move a chair, this holds true between any pair of points on the chair, but obviously not for points on a balloon that is being inflated.

In this chapter we will review the basic concepts relevant to Euclidean transformations. Then we will study a more general class of transformations, called **affine transformations**, which include Euclidean transformations as a subset. The set of **projective transformations** is even more general, and is a superset of affine transformations. All three classes of transformations find utility in a study of vision.

2.1 Euclidean Transformations

$\mathbf{A}$	Matrix
$\mathbf{a}$	Vector
$\mathbf{I}$	The identity matrix
$\psi : \mathbb{R}^n \mapsto \mathbb{R}^n$	Transformation
$\mathbf{x} \cdot \mathbf{y}$	Dot product (scalar product)
$\mathbf{x} \wedge \mathbf{y}$	Cross product (vector product)
$\lvert\lvert\mathbf{x}\rvert\rvert = \sqrt{\mathbf{x} \cdot \mathbf{x}}$	Norm

Definition 1 *Euclidean transformations (also known as isometries) are transformations that preserve distances between pairs of points.*

$$||\psi(\mathbf{a}) - \psi(\mathbf{b})|| = ||\mathbf{a} - \mathbf{b}|| \tag{2.1}$$

Translations, $\psi(\mathbf{a}) = \mathbf{a} + \mathbf{t}$, are isometries, since

$$||\psi(\mathbf{a}) - \psi(\mathbf{b})|| = ||\mathbf{t} + \mathbf{a} - (\mathbf{t} + \mathbf{b})|| = ||\mathbf{a} - \mathbf{b}|| \tag{2.2}$$

We now define orthogonal transformations; these constitute another major class of isometries.

Definition 2 *A linear transformation:* $\psi(\mathbf{a}) = \mathbf{A}\mathbf{a}$*, for some matrix* $\mathbf{A}$*.*

Definition 3 *Orthogonal transformations are linear transformations which preserve inner products.*

$$\mathbf{a} \cdot \mathbf{b} = \psi(\mathbf{a}) \cdot \psi(\mathbf{b}) \tag{2.3}$$

Property 1 *Orthogonal transformations preserve norms.*

$$\mathbf{a} \cdot \mathbf{a} = \psi(\mathbf{a}) \cdot \psi(\mathbf{a}) \implies ||\mathbf{a}|| = ||\psi(\mathbf{a})|| \tag{2.4}$$

Property 2 *Orthogonal transformations are isometries.*

$$\begin{align}
(\psi(\mathbf{a}) - \psi(\mathbf{b})) \cdot (\psi(\mathbf{a}) - \psi(\mathbf{b})) &\stackrel{?}{=} (\mathbf{a} - \mathbf{b}) \cdot (\mathbf{a} - \mathbf{b}) \tag{2.5}\\
||\psi(\mathbf{a})||^2 + ||\psi(\mathbf{b})||^2 - 2(\psi(\mathbf{a}) \cdot \psi(\mathbf{b})) &\stackrel{?}{=} ||\mathbf{a}||^2 + ||\mathbf{b}||^2 - 2(\mathbf{a} \cdot \mathbf{b}) \tag{2.6}
\end{align}$$

By property 1,

$$\begin{align}
||\psi(\mathbf{a})||^2 &= ||\mathbf{a}||^2 \tag{2.7}\\
||\psi(\mathbf{b})||^2 &= ||\mathbf{b}||^2. \tag{2.8}
\end{align}$$

By definition 3,

$$\psi(\mathbf{a}) \cdot \psi(\mathbf{b}) = \mathbf{a} \cdot \mathbf{b}. \tag{2.9}$$

Thus, equality holds.

Note that translations do not preserve norms (the distance with respect to the origin changes) and are not even linear transformations, except for the trivial case of translation by $\mathbf{0}$.

2.1.1 Properties of orthogonal matrices

Let ψ be an orthogonal transformation whose action we can represent by matrix multiplication, $\psi(\mathbf{a}) = \mathbf{A}\mathbf{a}$. Then, because it preserves inner products:

$$\psi(\mathbf{a}) \cdot \psi(\mathbf{b}) = \mathbf{a}^T\mathbf{b}\,. \tag{2.10}$$

By substitution,

$$\begin{aligned}\psi(\mathbf{a}) \cdot \psi(\mathbf{b}) &= (\mathbf{A}\mathbf{a})^T(\mathbf{A}\mathbf{b}) &(2.11)\\ &= \mathbf{a}^T\mathbf{A}^T\mathbf{A}\mathbf{b}\,. &(2.12)\end{aligned}$$

Thus,

$$\mathbf{a}^T\mathbf{b} = \mathbf{a}^T\mathbf{A}^T\mathbf{A}\mathbf{b} \implies \mathbf{A}^T\mathbf{A} = \mathbf{I} \implies \mathbf{A}^T = \mathbf{A}^{-1}\,. \tag{2.13}$$

Note that $\det(\mathbf{A})^2 = 1$ which implies that $\det(\mathbf{A}) = +1$ or -1. Each column of $\mathbf{A}$ has norm 1, and is orthogonal to the other column.

In 2D, these constraints put together force $\mathbf{A}$ to be one of two types of matrices.

$$\underbrace{\begin{bmatrix}\cos\theta & -\sin\theta\\ \sin\theta & \cos\theta\end{bmatrix}}_{\text{rotation, det=+1}} \text{or} \underbrace{\begin{bmatrix}\cos\theta & \sin\theta\\ \sin\theta & -\cos\theta\end{bmatrix}}_{\text{reflection, det=}-1}$$

Under a rotation by angle θ,

$$\begin{bmatrix}1\\0\end{bmatrix} \mapsto \begin{bmatrix}\cos\theta\\ \sin\theta\end{bmatrix} \text{and} \begin{bmatrix}0\\1\end{bmatrix} \mapsto \begin{bmatrix}-\sin\theta\\ \cos\theta\end{bmatrix}$$

The reflection matrix above corresponds to reflection around the line with angle $\frac{\theta}{2}$ (verify). Note that two rotations one after the other give another rotation, while two reflections give us a rotation.

Let us now construct some examples in 3D. Just as in 2D, rotations are characterized by orthogonal matrices with det $= +1$. For orthogonal matrices, each column vector has length 1, and the dot product of any two different columns is 0. This gives rise to six constraints (3 pairwise dot product constraints, and 3 length constraints), so for a 3 dimensional rotation matrix

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & a_{13}\\ a_{21} & a_{22} & a_{23}\\ a_{31} & a_{32} & a_{33}\end{bmatrix} \tag{2.14}$$

with 9 total parameters, there are really only three free parameters. There are several methods by which these parameters can be specified, as we will study later. Here are a few example rotation matrices.

- Rotation about z-axis by θ:

$$\mathbf{R} = \begin{bmatrix} \cos\theta & -\sin\theta & 0\\ \sin\theta & \cos\theta & 0\\ 0 & 0 & 1\end{bmatrix} \tag{2.15}$$

- Rotation about x-axis by θ:

$$\mathbf{R} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{bmatrix} \tag{2.16}$$

2.1.2 Group structure of isometries

Theorem 2.1 *Any isometry can be expressed as the combination of an orthogonal transformation followed by a translation as follows:*

$$\psi(\mathbf{a}) = \mathbf{A}\mathbf{a} + \mathbf{t} \tag{2.17}$$

where $\mathbf{A}$ *represents the orthogonal matrix and* $\mathbf{t}$ *is the translation vector.*

The set of rigid body motions constitutes a *group*[1]. In our notation, $\psi_1 \circ \psi_2$, ψ_1 composed with ψ_2, denotes that we apply ψ_2 first and then ψ_1.

We will show first that isometries are closed under composition. Consider two rigid body motions, ψ_1 and ψ_2:

$$\psi_1(\mathbf{a}) = \mathbf{A}_1\mathbf{a} + \mathbf{t}_1 \qquad \psi_2(\mathbf{a}) = \mathbf{A}_2\mathbf{a} + \mathbf{t}_2. \tag{2.18}$$

Then we have

$$\begin{aligned} \psi_1 \circ \psi_2(\mathbf{a}) &= \mathbf{A}_1(\mathbf{A}_2\mathbf{a} + \mathbf{t}_2) + \mathbf{t}_1 && (2.19) \\ &= \mathbf{A}_1\mathbf{A}_2\mathbf{a} + \mathbf{A}_1\mathbf{t}_2 + \mathbf{t}_1 && (2.20) \\ &= (\mathbf{A}_1\mathbf{A}_2)\mathbf{a} + (\mathbf{A}_1\mathbf{t}_2 + \mathbf{t}_1) && (2.21) \\ &= \mathbf{A}_3\mathbf{a} + \mathbf{t}_3 && (2.22) \end{aligned}$$

where $\mathbf{A}_3 = \mathbf{A}_1\mathbf{A}_2$ and $\mathbf{t}_3 = \mathbf{A}_1\mathbf{t}_2 + \mathbf{t}_3$. Thus, $\psi_1 \circ \psi_2 = \psi_3$ is also a rigid body motion, under the assumption that the product of two orthogonal matrices is orthogonal (Verify!)

Note that translations and rotations are closed under composition, but reflections are not.

We can verify the remaining axioms for showing that isometries constitute a group

- Identity: $\mathbf{A} = \mathbf{I}$, $\mathbf{d} = 0$.
- Inverse: We need $\mathbf{A}_1\mathbf{A}_2 = \mathbf{I}$ and $\mathbf{t}_3 = \mathbf{A}_1\mathbf{t}_2 + \mathbf{t}_1 = 0$. This means that for ψ_1 to be the inverse of ψ_2, $\mathbf{A}_1 = \mathbf{A}_2^T$ and $\mathbf{d}_2 = -\mathbf{A}_1^{-1}\mathbf{t}_1$
- Associativity: left as an exercise for the reader.

[1] A group $(G, \circ)$ is a set G with a binary operation $\circ$ that satisfies the following four axioms: Closure: For all a, b in G, the result of $a \circ b$ is also in G. Associativity: For all a, b and c in G, $(a \circ b) \circ c = a \circ (b \circ c)$. Identity element: There exists an element e in G such that for all a in G, $e \circ a = a \circ e = a$. Inverse element: For each a in G, there exists an element b in G such that $a \circ b = b \circ a = e$, where e is an identity element.

2.2 Parametrizing Rotations in 3D

Recall that rotation matrices have the property that each column vector has length 1 and the dot product of any 2 different columns is 0. These 6 constraints leave only 3 degrees of freedom. Here are some alternative notations used to represent orthogonal matrices in 3-D:

- Euler angles which specify rotations about 3 axes
- Axis plus amount of rotation
- Quaternions which generalize complex numbers from 2-D to 3-D. (Note, a complex number can represent a rotation in 2-D)

We will use the axis and rotation as the preferred representation of an orthogonal matrix: $\mathbf{s}$, θ, where $\mathbf{s}$ is the unit vector of the axis of rotation and θ is the amount of rotation.

Definition 4 *A matrix* $\mathbf{S}$ *is skew-symmetric if* $\mathbf{S} = -\mathbf{S}^T$.

Skew symmetric matrices can be used to represent "cross" products or vector products. Recall:

$$\begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} \wedge \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} = \begin{bmatrix} a_2b_3 - a_3b_2 \\ a_3b_1 - a_1b_3 \\ a_1b_2 - a_2b_1 \end{bmatrix}$$

We define $\widehat{\mathbf{a}}$ as:

$$\widehat{\mathbf{a}} \overset{def}{=} \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix}$$

Thus, multiplying $\widehat{\mathbf{a}}$ by any vector gives:

$$\begin{aligned} \widehat{\mathbf{a}} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} &= \begin{bmatrix} -a_3b_2 + a_2b_3 \\ a_3b_1 - a_1b_3 \\ -a_2b_1 + a_1b_2 \end{bmatrix} \\ &= \mathbf{a} \wedge \mathbf{b} \end{aligned}$$

Consider now, the equation of motion of a point q on a rotating body:

$$\dot{\mathbf{q}}(t) = \omega \wedge \mathbf{q}(t)$$

where the direction of ω specifies the axis of rotation and $\|\omega\|$ specifies the angular speed. Rewriting with $\widehat{\omega}$

$$\dot{\mathbf{q}}(t) = \widehat{\omega}\mathbf{q}(t)$$

The solution of this differential equation involves the exponential of a matrix. (In matlab, this is the operator `expm`.)

$$\mathbf{q}(t) = e^{\widehat{\omega}t}\mathbf{q}(0)$$

Where,

$$e^{\widehat{\omega}t} = \mathbf{I} + \widehat{\omega}t + \frac{(\widehat{\omega}t)^2}{2!} + \frac{(\widehat{\omega}t)^3}{3!} + ...$$

Collecting the odd and even terms in the above equation, we get to **Roderigues Formula** for a rotation matrix $\mathbf{R}$.

$$\begin{aligned}\mathbf{R} &= e^{\phi\widehat{s}} \\ &= \mathbf{I} + \sin\phi\ \widehat{\mathbf{s}} + (1 - \cos\phi)\widehat{\mathbf{s}}^2\end{aligned}$$

Here $\mathbf{s}$ is a unit vector along ω and $\phi = \|\omega\|t$ is the total amount of rotation. Given an axis of rotation, $\mathbf{s}$, and amount of rotation ϕ we can construct $\widehat{\mathbf{s}}$ and plug it in.

2.3 Affine transformations

Thus far we have focused on Euclidean transformations, $\psi(\mathbf{a}) = \mathbf{A}\mathbf{a} + \mathbf{t}$, where $\mathbf{A}$ is an orthogonal matrix. If we allow $\mathbf{A}$ to be any non-singular matrix (i.e., $\det\mathbf{A} \neq 0$), then we get the set of affine transformations. Note that the Euclidean transformations are a subset of the affine transformations.

2.3.1 Degrees of freedom

Let us count the degrees of freedom in the parameters that specify a transformation. For $\psi : \mathbb{R}^2 \mapsto \mathbb{R}^2$, Euclidean transformations have 3 free parameters (1 rotation, 2 translation), whereas Affine transformations have 6 (4 in $\mathbf{A}$ and 2 in $\mathbf{t}$). For $\psi : \mathbb{R}^3 \mapsto \mathbb{R}^3$, Euclidean transformations have 6 free parameters (3 rotation, 3 translation), whereas Affine transformations have 12 (9 in $\mathbf{A}$ and 3 in $\mathbf{t}$).

2.4 Exercises

1. Show that in $\mathcal{R}^2$ reflection about the $\theta = \alpha$ line followed by reflection about the $\theta = \beta$ is equivalent to a rotation of $2(\beta - \alpha)$.

2. Verify Roderigues formula by considering the powers of the skew-symmetric matrix associated with the cross product with a vector.

3. Write a Matlab function for computing the orthogonal matrix $\mathbf{R}$ corresponding to rotation ϕ about the axis vector $\mathbf{s}$. Find the eigenvalues and eigenvectors of the orthogonal matrices and study any relationship to the axis vector. Verify the formula $\cos\phi = \frac{1}{2}\{\text{trace}(\mathbf{R}) - 1\}$. Show some points before and after the rotation has been applied.

4. Write a Matlab function for the converse of that in the previous problem i.e. given an orthogonal matrix $\mathbf{R}$, compute the axis of rotation $\mathbf{s}$ and ϕ). Hint: Show that $\mathbf{R} - \mathbf{R}^T = (2\sin\phi)\widehat{\mathbf{s}}$

Chapter 3

Dynamic Perspective

3.1 Optical Flow

Motion in the 3D world, either of objects or of the camera, projects to motion in the image. We call this **optical flow**. At every point (x, y) in the image we get a 2D vector, corresponding to the motion of the feature located at that point. Thus optical flow is a 2D vector field. As first pointed out by Gibson, the optical flow field of a moving observer contains information to infer the 3D structure of the scene, as well as the movement of the observer, so-called **egomotion**. An example flow field is shown in Figure 3.1.

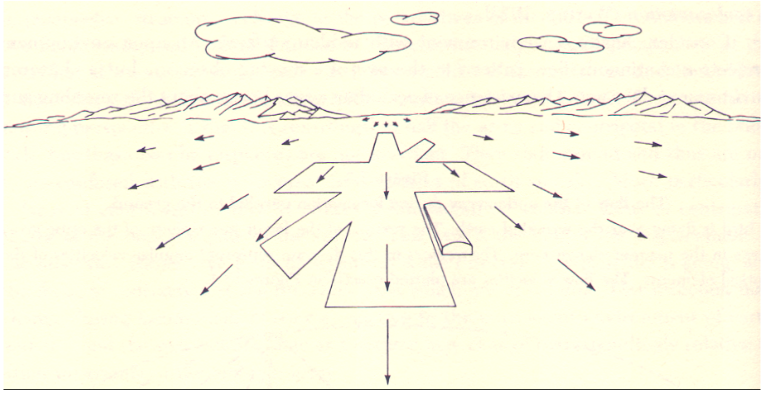

Figure 3.1: The optical flow field of a pilot just before takeoff

3.2 From 3-D Motion to 2-D Optical Flow

- $\mathbf{X} = (X, Y, Z)$: 3-D coordinates in the world
- (x, y): 2-D coordinates in the image
- $\mathbf{t} = (t_x, t_y, t_z)$: translational component of motion
- $\omega = (\omega_x, \omega_y, \omega_z)$: rotational component of motion

- $(u, v) = (\dot{x}, \dot{y})$: optical flow field

Let us start by deriving the equations relating motion in the 3-D world to the resulting optical flow field on the 2-D image plane. For simplicity we will focus on a single point in the scene $\mathbf{X} = (X, Y, Z)$.

Assume that the camera moves with translational velocity $\mathbf{t} = (t_x, t_x, t_z)$ and angular velocity $\omega = (\omega_x, \omega_y, \omega_z)$. Eq.(3.1) is used to characterize the movement of $\mathbf{X}$,

$$\dot{\mathbf{X}} = -\mathbf{t} - \omega \wedge \mathbf{X}, \tag{3.1}$$

which can be written out in coordinates as Eq.(3.2):

$$\begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{bmatrix} = - \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix} - \begin{bmatrix} \omega_y z - \omega_z y \\ \omega_z x - \omega_x z \\ \omega_x y - \omega_y x \end{bmatrix}. \tag{3.2}$$

Assume the image plane lies at $f = 1$, then $x = \frac{X}{Z}$ and $y = \frac{Y}{Z}$. Taking the derivative, we have

$$\dot{x} = \frac{\dot{X}Z - \dot{Z}X}{Z^2}, \dot{y} = \frac{\dot{Y}Z - \dot{Z}Y}{Z^2}. \tag{3.3}$$

Substitute $\dot{X}, \dot{Y}, \dot{Z}$ in Eq.(3.3) using Eq.(3.2), plug in $x = \frac{X}{Z}, y = \frac{Y}{Z}$, and simplify it, we get

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = \frac{1}{Z} \begin{bmatrix} -1 & 0 & x \\ 0 & -1 & y \end{bmatrix} \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix} + \begin{bmatrix} xy & -(1+x^2) & y \\ 1+y^2 & -xy & -x \end{bmatrix} \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} \tag{3.4}$$

We can use these equations to solve the forward (graphics) problem of determining the movement in the image given the movement in the world. If we assume that the parameters $\mathbf{t}$, ω are the same for all the points, that is equivalent to a rigidity assumption. It is obviously true if only the camera moves. Else if we have independently moving objects, then we have to consider each object separately.

Can all the unknowns be recovered, given enough points at which the optical flow is known? There is a scaling ambiguity about which we can do nothing. Consider a surface S_2 that is a dilation of the surface S_1 by a a factor of k, i.e. suppose that the corresponding point of surface S_2 is at depth $kZ(x, y)$. Furthermore suppose that the translational motion is k times faster. It is clear that the optical flow would be exactly the same for the two surfaces. Intuitively, farther objects moving faster generate the same optical flow as nearby objects moving slower. This is very convenient for generating special effects in Hollywood movies!

3.3 Pure translation

If the motion of the camera is purely translational, the terms due to rotation in Eq. (3.4) can be dropped and the flow field becomes

$$u(x,y) = \frac{-t_x + xt_z}{Z(x,y)}, v(x,y) = \frac{-t_y + yt_z}{Z(x,y)}. \tag{3.5}$$

We can gain intuition by considering the even more special case of translation along the optical axis, i.e. $t_z \neq 0, t_x = 0, t_y = 0$, the flow field in Eq.(3.5) becomes

$$u(x,y) = \frac{xt_z}{Z(x,y)}, v(x,y) = \frac{yt_z}{Z(x,y)}; \tag{3.6}$$

or equivalently

$$[u,v]^T(x,y) = \frac{t_z}{Z}[x,y]^T \tag{3.7}$$

This flow field has a very simple structure, as shown in Figure 3.2. It is zero at the origin, and at any other point, the optical flow vector points radially outward from the origin. We say that the origin is the **Focus of Expansion** of the flow field. The proportionality factor $\frac{t_z}{Z}$ is significant because it is the reciprocal of the **time to collision** $\frac{Z}{t_z}$ There is considerable evidence that this variable is used by flies, birds, humans etc as a cue for controlling locomotion. Note that while we are unable to estimate either the true speed (t_z) or the distance to the obstacle (Z), we are able to estimate what truly matters for controlling locomotion. Sometimes nature is kind!

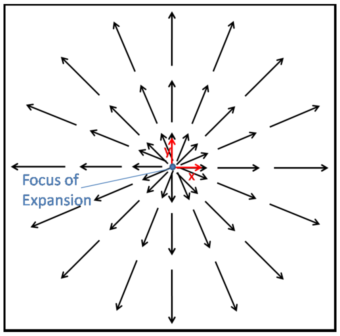

Figure 3.2: Optical flow field of an observer moving along the z-axis towards a frontoparallel wall

The case of general translation is essentially the same. We define the **Focus of Expansion** (FOE) of the optical flow field to be the point, where the optical flow is zero. Set $(u,v) = (0,0)$ in Eq.(3.5), we can solve for the coordinates of the FOE,

$$(x_{FOE}, y_{FOE}) = (\frac{t_x}{t_z}, \frac{t_y}{t_z}). \tag{3.8}$$

Note that the coordinates of the FOE tell us the direction of motion (we can't hope to know the speed, anyway!). It is also worth remarking that the FOE is just the vanishing point of the direction of translation.

Suppose we change the origin to the FOE by applying the following coordinate change to Eq.(3.5),

$$x' = x - \frac{t_x}{t_z}, y' = y - \frac{t_y}{t_z}, \tag{3.9}$$

then the optical flow field becomes

$$[u, v]^T(x', y') = \frac{t_z}{Z}[x', y']^T. \tag{3.10}$$

which should look very familiar. Thus the general case too corresponds to optical flow vectors pointing outwards from the FOE, justifying the choice of the term. Figure 3.3 shows such an optical vector field.

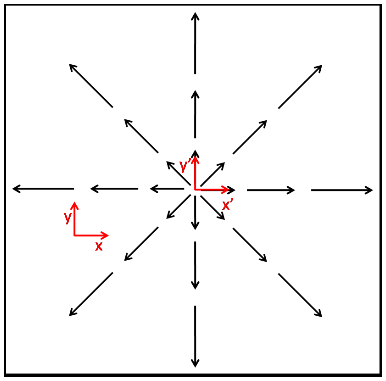

Figure 3.3: Optical flow vector field for general translational motion

We can also detect depth discontinuities from the optical flow field. If there is a sharp change in the lengths of flow vectors of two neighboring points, that indicates a discontinuity in depth. The ratio of their lengths tells us the ratio of their depths ($\frac{Z_1}{Z_2}$) ; however, we can't deduce the absolute depths (Z_1, Z_2), which is illustrated in Figure 3.4.

Thus optical flow is one of the most important cues to image segmentation (video segmentation, actually!). Even camouflaged animals (and snipers) must learn to stay very still to avoid detection.

3.4 General Motion

We begin by studying pure rotation. The most important thing to note is that the rotational component, obtaining by setting **t** to zero, has no dependence on Z. Therefor it conveys no information about the scene depth, only about the rotation of the observer. For moving animals in a stationary scene, this

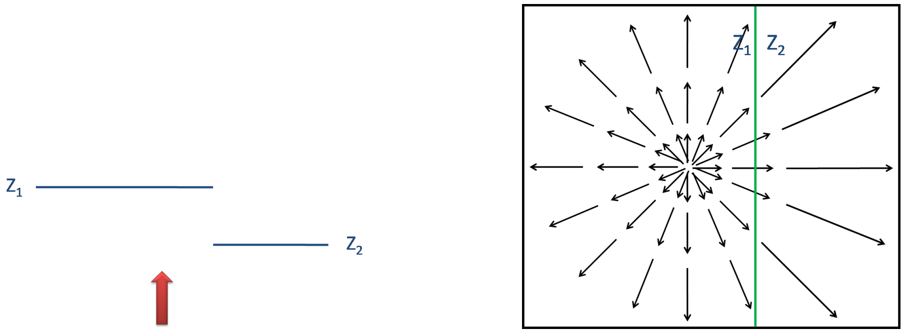

Figure 3.4: Depth discontinuity in optical flow field

commonly arises due to eye movements, which correspond to a rotation about the center of projection.

Thus the optical flow field corresponding to a general motion can be thought of as having a translational component very useful for inferring time to collision, depth boundaries in the scene, etc., and a rotational component which carries no information about the external 3D world. In the context of a moving animal where the rotational component is due to eye movements, some part of the animal brain has access to the rotational signal, since the eye movement was commanded by the brain itself. Hence the so-called **efference copy** carries information that can be used to subtract the rotational component. The residual is a purely translational flow field which can be be analyzed more straightforwardly. Amazingly, this is actually the case in humans (and probably in other animals with eye movements).

3.5 Summary

- Optical flow is the motion of the 3-D world projected on to the 2-D image. It can be used to derive cues about the structure of the 3-D scene as well as egomotion.

- The optical flow field for pure translation enables us to infer

  - The direction of movement, but not the absolute speed
  - The time to collision
  - Locations of depth discontinuities

3.6 Exercises

1. Implement the equations which relate the point wise optical flow to the six parameters of rigid body translation and rotation, and depth. Construct displays for some interesting cases.

2. As a test for the code that you have written in the previous exercise, suppose that I am driving my car along a straight stretch of freeway at a speed of 25 m/s. My eye height above the surface of the road is 1.25 m. What is the flow vector (in degrees/s)

   (a) At a point on the ground 25 m straight ahead.

   (b) At a point on the ground to my left at a distance of 25 m.

   (c) At points on the rear end of a 2 m wide car at a height of 1.25 m above the ground. This car has a headway of 25 m in front of me and is travelling at a speed of 20 m/s.