March 21/08

Introductory Techniques for 3-D Computer Vision

Emanuele Trucco
Heriot-Watt University,
Edinburgh, UK

Alessandro Verri
Università di Genova,
Genova, Italy

Prentice Hall
Upper Saddle River, New Jersey 07458

8

Motion

Eppur si muove.[1]
Galileo

This chapter concerns the analysis of the *visual motion* observed in time-varying image sequences.

Chapter Overview

Section 8.1 presents the basic concepts, importance and problems of visual motion.

Section 8.2 introduces the notions of *motion field* and *motion parallax*, and their fundamental equations.

Section 8.3 discusses the *image brightness constancy equation* and the *optical flow*, the approximation of the motion field which can be computed from the changing image brightness pattern.

Section 8.4 presents methods for estimating the motion field, divided in *differential* and *feature-matching/tracking* methods.

Section 8.5 deals with the *reconstruction of 3-D motion and structure*.

Section 8.6 discusses *motion-based segmentation* based on change detection.

What You Need to Know to Understand this Chapter

- Working knowledge of Chapters 2 and 7.
- Eigenvalues and eigenvectors of a matrix.
- Least squares and SVD (Appendix, section A.6).
- The basics of Kalman filtering (Appendix, section A.8).

[1] And yet it is moving.

8.1 Introduction

Until now, we have studied visual computations on single images, or two images acquired simultaneously. In this chapter, we broaden our perspective and focus on the processing of images *over time*. More precisely, we are interested in the visual information that can be extracted from the spatial and temporal changes occurring in an *image sequence*.

Definition: Image Sequence

An image sequence is a series of N images, or *frames*, acquired at discrete time instants $t_k = t_0 + k\Delta t$, where Δt is a fixed time interval, and $k = 0, 1, \dots, N-1$.

☞ In order to acquire an image sequence, you need a frame grabber capable of storing frames at a fast rate. Typical rates are the so called *frame rate* and *field rate*, corresponding to a time interval Δt of 1/24sec and 1/30sec respectively. If you are allowed to choose a different time interval, or simply want to subsample an image sequence, make sure that Δt is small enough to guarantee that the discrete sequence is a representative sampling of the continuous image evolving over time; as a rule of thumb, this means that the apparent displacements over the image plane between frames should be at most a few pixels.

Assuming the illumination conditions do not vary, image changes are caused by a *relative motion between camera and scene*: the viewing camera could move in front of a static scene, or parts of the scene could move in front of a stationary camera, or, in general, both camera and objects could be moving with different motions.

8.1.1 The Importance of Visual Motion

The temporal dimension in visual processing is important primarily for two reasons. First, the apparent motion of objects onto the image plane is a strong visual cue for understanding structure and 3-D motion. Second, biological visual systems use visual motion to infer properties of the 3-D world with little *a priori* knowledge of it. Two simple examples may be useful to illustrate these points.

Example 1: Random Dot Sequences. Consider an image of *random dots*, generated by assigning to each pixel a random grey level. Consider a second image obtained by shifting a squared, central region of the first image by a few pixels, say, to the right, and filling the gap thus created with more random dots. Two such images are shown in Figure 8.1. If you display the two images in sequence on a computer screen, in the same window and one after the other at a sufficiently fast rate, you will unmistakably *see* a square moving sideways back and forth against a steady background. Notice that the

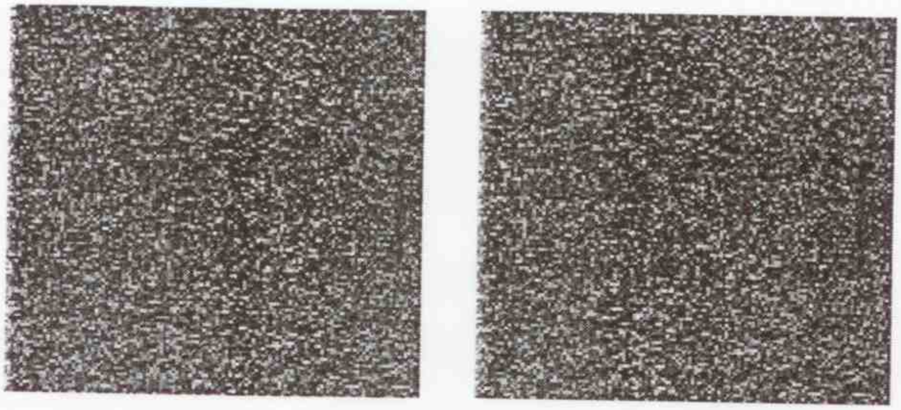

Figure 8.1 A sequence of two random dot images: a square has been displaced between the two frames.

visual system bases its judgement on the only information available in the sequence; that is, the displacement of the square in the two images.[2]

Example 2: Computing Time-to-Impact. Visual motion allows us to compute useful properties of the observed 3-D world with very little knowledge about it. Consider a planar version of the usual pinhole camera model, and a vertical bar perpendicular to the optical axis, travelling towards the camera with constant velocity as shown in Figure 8.2. We want to prove a simple but very important fact: *It is possible to compute the time*, τ, *taken by the bar to reach the camera only from image information;* that is, without knowing either the real size of the bar or its velocity in 3-D space.[3]

As shown in Figure 8.2, we denote with L the real size of the bar, with V its constant velocity, and with f the focal length of the camera. The origin of the reference frame is the projection center. If the position of the bar on the optical axis is $D(0) = D_0$ at time $t = 0$, its position at a later time t will be $D = D_0 - Vt$. Note that L, V, f, D_0, and the choice of the time origin are all unknown, but that τ can be written as

$$\tau = \frac{V}{D}. \tag{8.1}$$

From Figure 8.2, we see that $l(t)$, the *apparent* size of the bar at time t on the image plane, is given by

$$l(t) = f\frac{L}{D}.$$

[2] Incidentally, you can look at the two images of Figure 8.1 as a *random-dot stereogram* to perceive a square floating in the background. Stand a diskette (or a sheet of paper of the same size) between the two images and touch your nose against the diskette, so that each eye can see only one image. Focus your eyes *behind* the page. After a while, the two images should fuse and produce the impression of a square floating against the background.

[3] In the biologically-oriented community of computer vision, τ is called, rather pessimistically, *time-to-collision* or even *time-to-crash*!

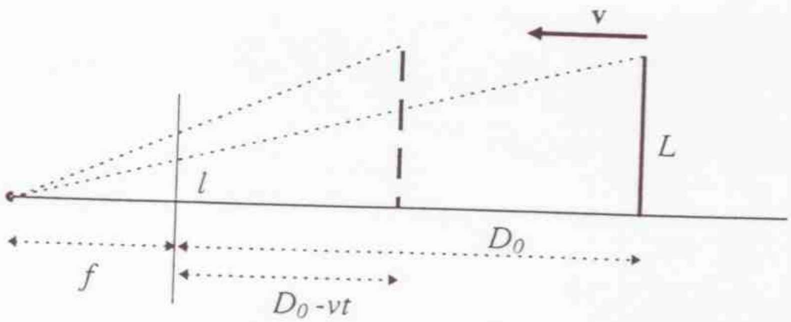

Figure 8.2 How long before the bar reaches the camera?

If we now compute the time derivative of $l(t)$,

$$l'(t) = \frac{dl(t)}{dt} = -f\frac{L}{D^2}\frac{dD}{dt} = f\frac{LV}{D^2},$$

take the ratio between $l(t)$ and $l'(t)$, and use (8.1), we obtain

$$\frac{l(t)}{l'(t)} = \tau. \tag{8.2}$$

This is the equation we were after: since both the apparent size of the bar, $l(t)$, and its time derivative, $l'(t)$, are measured from the images, (8.2) allows us to compute τ in the absence of *any* 3-D information, like the size of the bar and its velocity.

8.1.2 The Problems of Motion Analysis

It is now time to state the main problems of motion analysis. The analogies with stereo suggest to begin by dividing the motion problem into two subproblems.

Two Subproblems of Motion

Correspondence: Which elements of a frame correspond to which elements of the next frame of the sequence?

Reconstruction: Given a number of corresponding elements, and possibly knowledge of the camera's intrinsic parameters, what can we say about the 3-D motion and structure of the observed world?

Main Differences between Motion and Stereo

Correspondence: As image sequences are sampled temporally at usually high rates, the spatial differences (disparities) between consecutive frames are, on average, much smaller than those of typical stereo pairs.

Reconstruction: Unlike stereo, in motion the relative 3-D displacement between the viewing camera and the scene is not necessarily caused by a single 3-D rigid transformation.

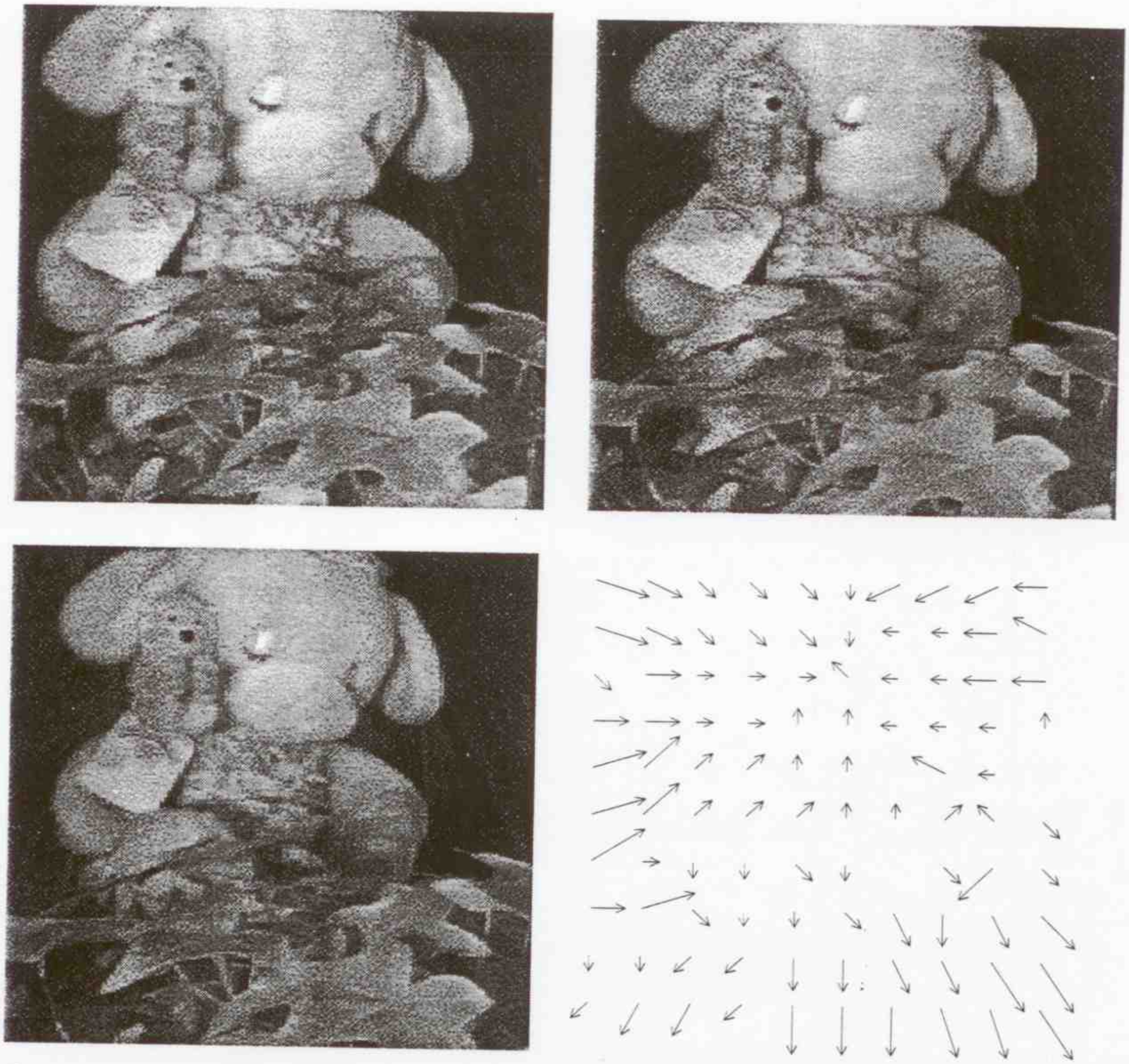

Figure 8.3 Three frames from a long image sequence (left to right and top to bottom) and the optical flow computed from the sequence, showing that the plant in the foreground is moving towards the camera, and the soft toys away from it.

Regarding correspondence, the fact that motion sequences make many, closely sampled frames available for analysis is an advantage over the stereo case for at least two reasons. First, feature-based approaches can be made more effective by *tracking* techniques, which exploit the past history of the features' motion to predict disparities in the next frame. Second, due to the generally small spatial and temporal differences between consecutive frames, the correspondence problem can also be cast as the problem of *estimating the apparent motion of the image brightness pattern*, usually called *optical flow* (see Figure 8.3).

We shall use two strategies for solving the correspondence problem.

Differential methods (section 8.4.1) lead to *dense* measures; that is, computed at each image pixel. They use estimates of time derivatives, and require therefore image sequences sampled closely.

Matching methods (section 8.4.2) lead to *sparse* measures; that is, computed only at a subset of image points. We shall place emphasis on *Kalman filtering* as a technique for matching and tracking efficiently sparse image features over time.

Unlike correspondence, and perhaps not surprisingly, reconstruction is more difficult in motion than in stereo. Even in the presence of only one 3-D motion between the viewing camera and the scene, frame-by-frame recovery of motion and structure turns out to be more sensitive to noise. The reason is that the baseline between consecutive frames, regarded as a stereo pair, is very small (see Chapter 7). 3-D motion and structure estimation from both sparse and dense estimates of the image motion is discussed in sections 8.5.1 and 8.5.2, respectively.

This chapter discusses and motivates methods for solving correspondence and reconstruction under the following simplifying assumption.

Assumption

There is only one, rigid, relative motion between the camera and the observed scene, and the illumination conditions do not change.

This assumption of single, rigid motion implies that *the 3-D objects observed cannot move of different motions*. This assumption is violated, for example, by sequences of football matches, motorway traffic or busy streets, but satisfied by, say, the sequence of a building viewed by a moving observer. The assumption also rules out flexible (nonrigid) objects: deformable objects like clothes or moving human bodies are excluded.

If the camera is looking at more than one moving object, or you simply cannot assume a moving camera in a static environment, a third subproblem must be added.

The Third Subproblem of Motion

The segmentation problem: What are the regions of the image plane which correspond to different moving objects?

The main difficulty here is a chicken and egg problem: should we first solve the matching problem and then determine the regions corresponding to the different moving objects, or find the regions first, and then look for correspondences? This question is addressed in section 8.6 in the hypothesis that the viewing camera is not moving. Pointers to solutions to this difficult problem in more general cases are given in the Further Readings.

We now begin by establishing some basic facts.

8.2 The Motion Field of Rigid Objects

Definition: Motion Field

The motion field is the 2-D vector field of velocities of the image points, induced by the relative motion between the viewing camera and the observed scene.

The motion field can be thought of as *the projection of the 3-D velocity field on the image plane* (to visualize this vector field, imagine to project the 3-D velocity vectors on the image). The purpose of this section is to get acquainted with the theory and geometrical properties of the motion field. *We shall work in the camera reference frame*, ignoring the image reference frame and the pixelization.[4] The issue of camera calibration will be raised in due time.

This section presents some essential facts of motion fields, compares disparity representations in motion and stereo, analyzes two special cases of rigid motion leading to generally useful facts, and introduces the concept of motion parallax.

8.2.1 Basics

Notation. We let $\mathbf{P} = [X, Y, Z]^\top$ be a 3-D point in the usual camera reference frame: The projection center is in the origin, the optical axis is the Z axis, and f denotes the focal length. The image of a scene point, $\mathbf{P}$, is the point $\mathbf{p}$ given by

$$\mathbf{p} = f\frac{\mathbf{P}}{Z}. \tag{8.3}$$

As usual (see Chapter 2), since the third coordinate of $\mathbf{p}$ is always equal to f, we write $\mathbf{p} = [x, y]^\top$ instead of $\mathbf{p} = [x, y, f]^\top$. The relative motion between $\mathbf{P}$ and the camera can be described as

$$\mathbf{V} = -\mathbf{T} - \omega \times \mathbf{P}, \tag{8.4}$$

where $\mathbf{T}$ is the translational component of the motion,[5] and $\boldsymbol{\omega}$ the angular velocity. As the motion is rigid, $\mathbf{T}$ and $\boldsymbol{\omega}$ are the same for any $\mathbf{P}$. In components, (8.4) reads

$$\begin{aligned} V_x &= -T_x - \omega_y Z + \omega_z Y \\ V_y &= -T_y - \omega_z X + \omega_x Z \\ V_z &= -T_z - \omega_x Y + \omega_y X. \end{aligned} \tag{8.5}$$

[4] Remember, this means that we consider the intrinsic parameters known.

[5] Note that **T** denotes a velocity vector only in this chapter, not a displacement vector as in the rest of the book.

The Basic Equations of the Motion Field. To obtain the relation between the velocity of **P** in space and the corresponding velocity of **p** on the image plane, we take the time derivative of both sides of (8.3), which gives an important set of equations.

The Basic Equations of the Motion Field

The motion field, **v**, is given by

$$\mathbf{v} = f\,\frac{Z\mathbf{V} - V_z\mathbf{P}}{Z^2}. \tag{8.6}$$

In components, and using (8.5), (8.6) read

$$\begin{aligned} v_x &= \frac{T_z x - T_x f}{Z} - \omega_y f + \omega_z y + \frac{\omega_x x y}{f} - \frac{\omega_y x^2}{f} \\ v_y &= \frac{T_z y - T_y f}{Z} + \omega_x f - \omega_z x - \frac{\omega_y x y}{f} + \frac{\omega_x y^2}{f}. \end{aligned} \tag{8.7}$$

Notice that *the motion field is the sum of two components, one of which depends on translation only, the other on rotation only*. In particular, the translational components of the motion field are

$$\begin{aligned} v_x^T &= \frac{T_z x - T_x f}{Z} \\ v_y^T &= \frac{T_z y - T_y f}{Z}, \end{aligned}$$

and the rotational components are

$$\begin{aligned} v_x^\omega &= -\omega_y f + \omega_z y + \frac{\omega_x x y}{f} - \frac{\omega_y x^2}{f} \\ v_y^\omega &= \omega_x f - \omega_z x - \frac{\omega_y x y}{f} + \frac{\omega_x y^2}{f}. \end{aligned}$$

Since the component of the motion field along the optical axis is always equal to 0, we shall write $\mathbf{v} = [v_x, v_y]^\top$ instead of $\mathbf{v} = [v_x, v_y, 0]^\top$. Notice that, in the last two pairs of equations, the terms depending on the angular velocity, $\boldsymbol{\omega}$, and depth, Z, are decoupled. This discloses an important property of the motion field: *the part of the motion field that depends on angular velocity does not carry information on depth*.

Comparing Disparity Representations in Stereo and Motion. As we said before, stereo and motion pose similar computation problems, and one of these is correspondence. Point displacements are represented by disparity maps in stereo, and by motion fields in motion. An obvious question is, how similar are disparity maps and motion fields? The key difference is that *the motion field is a differential concept, stereo disparity is not*. The motion field is based on velocity, and therefore on time derivatives:

Consecutive frames must be as close as possible to guarantee good discrete approximations of the continuous time derivatives. In stereo, there is no such constraint on the two images, and the disparities can take, in principle, any value.

Stereo Disparity Map and Motion Field

The spatial displacements of corresponding points between the images of a stereo pair (forming the stereo disparity map) are *finite*, and, in principle, unconstrained.

The spatial displacements of corresponding points between consecutive frames of a motion sequence (forming the motion field) are discrete approximations of time-varying derivatives, and must therefore be suitably small.

The motion field coincides with the stereo disparity map *only if* spatial and temporal differences between frames are sufficiently small.

8.2.2 Special Case 1: Pure Translation

We now analyze the case in which the relative motion between the viewing camera and the scene has no rotational component. The resulting motion field has a peculiar spatial structure, and its analysis leads to concepts very useful in general.

Since $\boldsymbol{\omega} = 0$, (8.7) read

$$v_x = \frac{T_z x - T_x f}{Z}$$
$$v_y = \frac{T_z y - T_y f}{Z} \tag{8.8}$$

We first consider the general case in which $T_z \neq 0$. Introducing a point $\mathbf{p}_0 = [x_0, y_0]^\top$ such that

$$x_0 = f T_x / T_z$$
$$y_0 = f T_y / T_z, \tag{8.9}$$

(8.8) become

$$v_x = (x - x_0)\frac{T_z}{Z}$$
$$v_y = (y - y_0)\frac{T_z}{Z}. \tag{8.10}$$

Equation (8.10) say that *the motion field of a pure translation is radial*: It consists of vectors radiating from a common origin, the point $\mathbf{p}_0$, which is therefore the *vanishing point* of the translation direction. In particular, if $T_z < 0$, the vectors point away from $\mathbf{p}_0$, and $\mathbf{p}_0$ is called the *focus of expansion* (Figure 8.4 (a)); if $T_z > 0$, the motion field vectors point towards $\mathbf{p}_0$, and $\mathbf{p}_0$ is called the *focus of contraction* (Figure 8.4 (b)). In addition, the length of $\mathbf{v} = \mathbf{v}(\mathbf{p})$ is proportional to the distance between $\mathbf{p}$ and $\mathbf{p}_0$, and inversely proportional to the depth of the 3-D point $\mathbf{P}$.

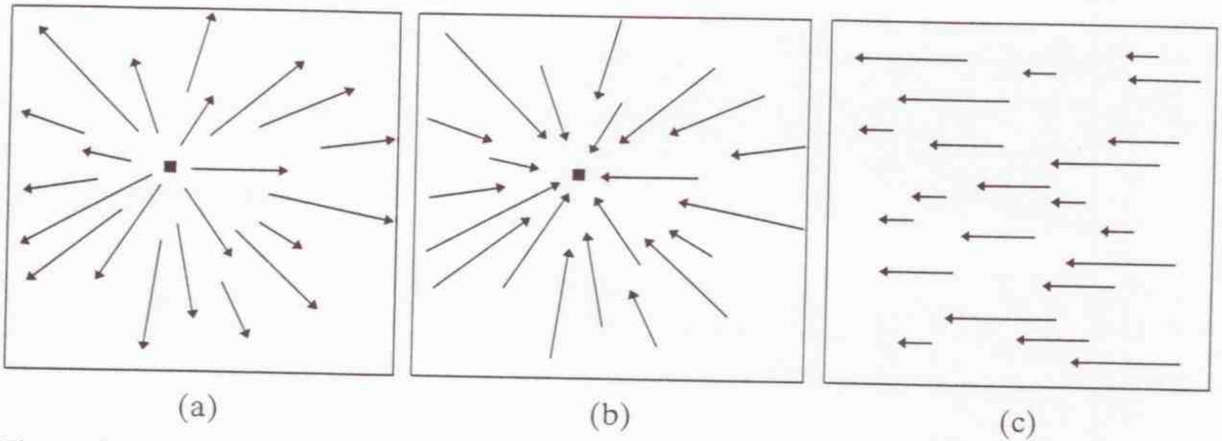

Figure 8.4 The three types of motion field generated by translational motion. The filled square marks the instantaneous epipole.

☞ Notice that the point $\mathbf{p}_0$ retains its significance and many of its properties even in the presence of a rotational component of 3-D motion (section 8.5.2).

If T_z vanishes (a rather special case), (8.8) become

$$v_x = -f\frac{T_x}{Z}$$

$$v_y = -f\frac{T_y}{Z}.$$

Therefore, *if $T_z = 0$, all the motion field vectors are parallel* (see Figure 8.4 (c)) and their lengths are inversely proportional to the depth of the corresponding 3-D points.

☞ In homogeneous coordinates, there would be no need to distinguish between the two cases $T_z \neq 0$ and $T_z = 0$: For *all* possible values of T_z, including $T_z = 0$, $\mathbf{p}_0$ is the vanishing point of the direction in 3-D space of the translation vector $\mathbf{T}$, and the 3-D line through the center of projection and $\mathbf{p}_0$ is parallel to $\mathbf{T}$.

Following is a summary of the main properties of the motion field of a purely translational motion.

Pure Translation: Properties of the Motion Field

1. If $T_z \neq 0$, the motion field is *radial* (see (8.10)), and all vectors point towards (or away from) a single point, $\mathbf{p}_0$, given by (8.8). If $T_z = 0$, the motion field is *parallel*.
2. The length of motion field vectors is inversely proportional to the depth Z; if $T_z \neq 0$, it is also inversely proportional to the distance from $\mathbf{p}$ to $\mathbf{p}_0$.
3. $\mathbf{p}_0$ is the vanishing point of the direction of translation (see (8.10)).
4. $\mathbf{p}_0$ is the intersection of the ray parallel to the translation vector with the image plane.

8.2.3 Special Case 2: Moving Plane

Planes are common surfaces in man-made objects and environments, so it is useful to investigate the properties of the motion field of a moving plane. Assume that the camera is observing a planar surface, π, of equation

$$\mathbf{n}^\top \mathbf{P} = d \tag{8.11}$$

where $\mathbf{n} = [n_x, n_y, n_z]^\top$ is the unit vector normal to π, and d the distance between π and the origin (the center of projection). Let π be moving in space with translational velocity $\mathbf{T}$ and angular velocity $\boldsymbol{\omega}$, so that both $\mathbf{n}$ and d in (8.11) are functions of time. By means of (8.3), (8.11) can be rewritten as

$$\frac{n_x x + n_y y + n_z f}{f} Z = d. \tag{8.12}$$

Solving for Z in (8.12), and plugging the resulting expression into (8.7), we have

$$\begin{aligned} v_x &= \frac{1}{fd}(a_1 x^2 + a_2 xy + a_3 fx + a_4 fy + a_5 f^2) \\ v_y &= \frac{1}{fd}(a_1 xy + a_2 y^2 + a_6 fy + a_7 fx + a_8 f^2) \end{aligned} \tag{8.13}$$

where

$$\begin{aligned} a_1 &= -d\omega_y + T_z n_x, & a_2 &= d\omega_x + T_z n_y, \\ a_3 &= T_z n_z - T_x n_x, & a_4 &= d\omega_z - T_x n_y, \\ a_5 &= -d\omega_y - T_x n_z, & a_6 &= T_z n_z - T_y n_y, \\ a_7 &= -d\omega_z - T_y n_x, & a_8 &= d\omega_x - T_y n_z. \end{aligned}$$

The (8.13) states, interestingly, that *the motion field of a moving planar surface, at any instant t, is a quadratic polynomial in the coordinates (x, y, f) of the image points.*

The remarkable symmetry of the time-dependent coefficients $a_1 \ldots a_8$ is not coincidental. You can easily verify that the a_i remain unchanged if d, $\mathbf{n}$, $\mathbf{T}$, and $\boldsymbol{\omega}$ are replaced by

$$\begin{aligned} d' &= d \\ \mathbf{n}' &= \mathbf{T}/\|\mathbf{T}\| \\ \mathbf{T}' &= \|\mathbf{T}\|\mathbf{n} \\ \boldsymbol{\omega}' &= \boldsymbol{\omega} + \mathbf{n} \times \mathbf{T}/d. \end{aligned}$$

This means that, apart from the special case in which **n** and **T** are parallel, *the same motion field can be produced by two different planes undergoing two different 3-D motions*.[6]

☞ The practical consequence is that it is usually impossible to recover uniquely the 3-D structure parameters, **n** and d, and motion parameters, **T** and ω, of a planar set of points from the motion field *alone*.

You might be tempted to regard this discussion on the motion field of a planar surface as a mere mathematical curiosity. On the contrary, we can draw at least two important and general conclusions from it.

1. Since the motion field of a planar surface is described *exactly and globally* by a polynomial of second degree (see (8.13)), *the motion field of any smooth surface is likely to be approximated well by a low-order polynomial even over relatively large regions of the image plane* (Exercise 8.1). The useful consequence is that very simple parametric models enable a quite accurate estimation of the motion field in rather general circumstances (section 8.4.1).
2. As algorithms recoverying 3-D motion and structure cannot be based on motion estimates produced by coplanar points, measurements must be made at many different locations of the image plane in order to minimize the probability of looking at points that lie on planar or nearly planar surfaces.[7] We will return to this point in sections 8.5.1 and 8.5.2.

We conclude this section with a summary of the main properties of the motion field of a planar surface.

Moving Plane: Properties of Motion Field

1. The motion field of a planar surface is, at any time instant, a quadratic polynomial in the image coordinates.
2. Due to the special symmetry of the polynomial coefficients, the same motion field can be produced by two different planar surfaces undergoing different 3-D motions.

8.2.4 Motion Parallax

The decoupling of rotational parameters and depth in the (8.7) is responsible for what is called *motion parallax*. Informally, motion parallax refers to the fact that *the relative motion field of two instantaneously coincident points does not depend on the rotational*

[6] This result should not surprise you. Planar surfaces lack generality: The eight-point algorithm (Chapter 7), for example, fails to yield a unique solution if the points are all coplanar in 3-D space.

[7] A "nearly planar" surface is a surface that can be approximated by a plane within a given tolerance, which is typically proportional to the distance of the surface from the image plane.

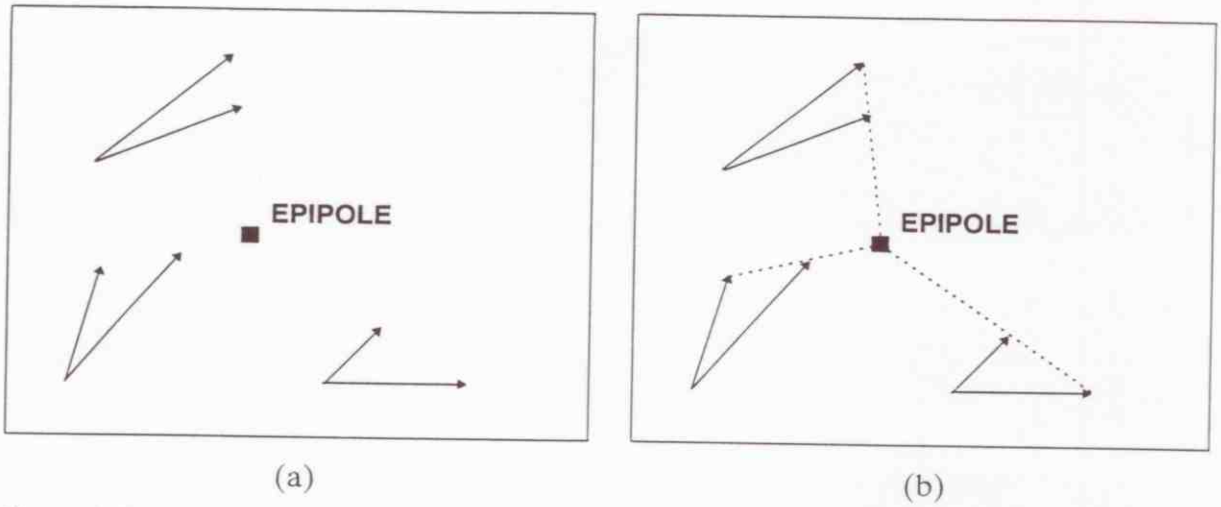

Figure 8.5 Three couples of instantaneously coincident image points and their flow vectors (a); the difference vectors point towards the instantaneous epipole (b).

component of motion in 3-D space; this section makes this statement more precise. Motion parallax will be used in section 8.5.2 to compute structure and motion from optical flow.

Let two points $\mathbf{P} = [X, Y, Z]^\top$ and $\bar{\mathbf{P}} = [\bar{X}, \bar{Y}, \bar{Z}]^\top$ be projected into the image points $\mathbf{p}$ and $\bar{\mathbf{p}}$, respectively. We know that the corresponding motion field vectors can be written as

$$v_x = v_x^T + v_x^\omega$$
$$v_y = v_y^T + v_y^\omega$$

and

$$\bar{v}_x = \bar{v}_x^T + \bar{v}_x^\omega$$
$$\bar{v}_y = \bar{v}_y^T + \bar{v}_y^\omega.$$

If, at some instant t, the points $\mathbf{p}$ and $\bar{\mathbf{p}}$ happen to be coincident (Figure 8.5(a)), we have

$$\mathbf{p} = \bar{\mathbf{p}} = [x, y]^\top,$$

and the rotational components of the observed motion, (v_x^ω, v_y^ω) and $(\bar{v}_x^\omega, \bar{v}_y^\omega)$, become

$$\begin{aligned} v_x^\omega = \bar{v}_x^\omega &= -\omega_y f + \omega_z y + \frac{\omega_x x y}{f} - \frac{\omega_y x^2}{f} \\ v_y^\omega = \bar{v}_y^\omega &= \omega_x f - \omega_z x - \frac{\omega_y x y}{f} + \frac{\omega_x y^2}{f}. \end{aligned} \tag{8.14}$$

Therefore, by taking the difference between $\mathbf{v}$ and $\bar{\mathbf{v}}$, the rotational components cancel out, and we obtain

$$\Delta v_x = v_x^T - \bar{v}_x^T = (T_z x - T_x f)(\frac{1}{Z} - \frac{1}{\bar{Z}})$$

$$\Delta v_y = v_y^T - \bar{v}_y^T = (T_z y - T_y f)(\frac{1}{Z} - \frac{1}{\bar{Z}}).$$

The vector $(\Delta v_x, \Delta v_y)$ can be thought of as the *relative motion field*. Other factors being equal, Δv_x and Δv_y increase with the separation in depth between $\mathbf{P}$ and $\bar{\mathbf{P}}$.

Notice that the ratio between Δv_y and Δv_x can be written as

$$\frac{\Delta v_y}{\Delta v_x} = \frac{y - y_0}{x - x_0}$$

with $[x_0, y_0]^\top$ image coordinates of $\mathbf{p}_0$, the vanishing point of the translation direction (Figure 8.5(b)).[8] Hence, *for all possible rotational motions, the vector* $(\Delta v_x^T, \Delta v_y^T)$ *points in the direction of* $\mathbf{p}_0$. Consequently, the dot product between the motion field, $\mathbf{v}$, and the vector $[y - y_0, -(x - x_0)]^\top$, which is perpendicular to $\mathbf{p} - \mathbf{p}_0$, depends neither on the 3-D structure of the scene nor on the translational component of motion, and can be written as

$$v_\perp = (y - y_0)v_x^\omega - (x - x_0)v_y^\omega.$$

We will make use of this result in section 8.5.2, where we will learn how to compute motion and structure from dense estimates of the motion field.

☞ Be aware that the vanishing point of translation, $\mathbf{p}_0$, and the point at which $\mathbf{v}$ vanishes, call it $\mathbf{q}$, are in general *different*; they coincide only if the motion is purely translational. Any rotational component about an axis not perpendicular to the image plane shifts the position of $\mathbf{q}$, whereas the position of $\mathbf{p}_0$ remains unchanged, as it is determined by the translational component only. Somewhat deceptively, the flow field in the neighborhood of $\mathbf{q}$ might still look very much like a focus of expansion or contraction (see Figure 8.3).

And here is the customary summary of the main ideas.

Motion Parallax

The relative motion field of two instantaneously coincident points:

1. does not depend on the rotational component of motion
2. points towards (away from) the point $\mathbf{p}_0$, the vanishing point of the translation direction

[8] Section 8.2.5 makes it clear that this point can be regarded as an *instantaneous epipole*.

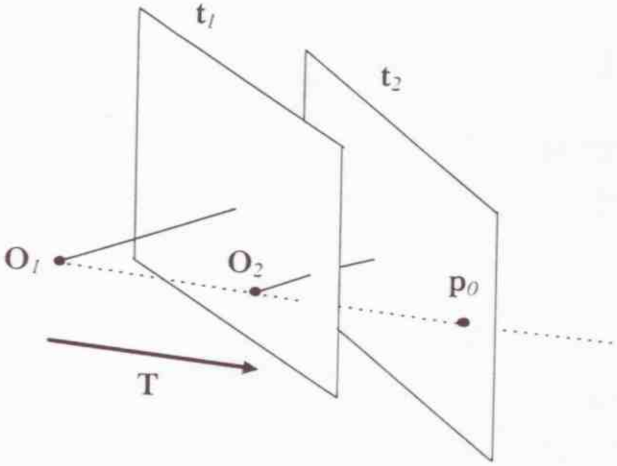

Figure 8.6 The point $\mathbf{p}_0$ as instantaneous epipole.

8.2.5 The Instantaneous Epipole

We close this introductory section with an important remark. The point $\mathbf{p}_0$, being the intersection of the image plane with the direction of translation of the center of projection, can be regarded as the *instantaneous epipole* between pairs of consecutive frames in the sequence (Figure 8.6). The main consequence of this property is that *it is possible to locate* $\mathbf{p}_0$ *without prior knowledge of the camera intrinsic parameters* (section 8.5.2).

☞ Notice that, as in the case of stereo, knowing the epipole's location in image coordinates is *not* equivalent to knowing the direction of translation (the baseline vector for stereo). The relation between epipole location and translation direction is specified by (8.9), which is written in the camera (not image) frame, and contains the focal length f. Therefore, *the epipole's location gives the direction of translation only if the intrinsic parameters of the viewing camera are known.*

8.3 The Notion of Optical Flow

We now move to the problem of *estimating the motion field from image sequences*, that is, from the spatial and temporal variations of the image brightness. To do this, we must model the link between brightness variations and motion field, and arrive at a fundamental equation of motion analysis, the *image brightness constancy equation*. We want also to analyze the power and validity of this equation, that is, understand how much and how well it can help us to estimate the motion field. For simplicity, we will assume that *the image brightness is continuous and differentiable as many times as needed in both the spatial and temporal domain.*

8.3.1 The Image Brightness Constancy Equation

It is common experience that, under most circumstances, the apparent brightness of moving objects remains constant. We have seen in Chapter 2 that the image irradiance is proportional to the scene radiance in the direction of the optical axis of the camera; if we assume that the proportionality factor is the same across the entire image plane, the constancy of the apparent brightness of the observed scene can be written as the stationarity of the image brightness E over time:

$$\frac{dE}{dt} = 0. \tag{8.15}$$

☞ In (8.15), the image brightness, E, should be regarded as a function of both the spatial coordinates of the image plane, x and y, and of time, that is, $E = E(x, y, t)$. Since x and y are in turn functions of t, the *total* derivative in (8.15) should not be confused with the *partial* derivative $\partial E/\partial t$.

Via the chain rule of differentiation, the total temporal derivative reads

$$\frac{dE(x(t), y(t), t)}{dt} = \frac{\partial E}{\partial x}\frac{dx}{dt} + \frac{\partial E}{\partial y}\frac{dy}{dt} + \frac{\partial E}{\partial t} = 0. \tag{8.16}$$

The partial spatial derivatives of the image brightness are simply the components of the spatial image gradient, ∇E, and the temporal derivatives, dx/dt and dy/dt, the components of the motion field, $\mathbf{v}$. Using these facts, we can rewrite (8.16) as the image brightness constancy equation.

The Image Brightness Constancy Equation

Given the image brightness, $E = E(x, y, t)$, and the motion field, $\mathbf{v}$,

$$(\nabla E)^\top \mathbf{v} + E_t = 0. \tag{8.17}$$

The subscript t denotes partial differentiation with respect to time.

We shall now discuss the relevance and applicability of this equation for the estimation of the motion field.

8.3.2 The Aperture Problem

How much of the motion field can be determined through (8.17)? *Only its component in the direction of the spatial image gradient,*[9] v_n. We can see this analytically by isolating the measurable quantities in (8.17):

$$-\frac{E_t}{\|\nabla E\|} = \frac{(\nabla E)^\top \mathbf{v}}{\|\nabla E\|} = v_n \tag{8.18}$$

[9] This component is sometimes called the *normal component*, because the spatial image gradient is normal to the spatial direction along which image intensity remains constant.

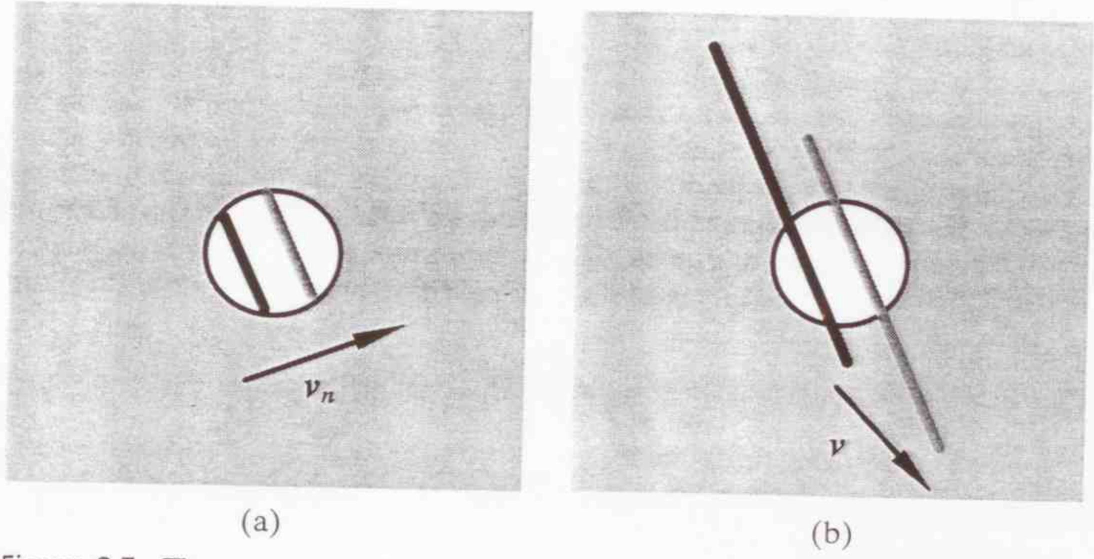

Figure 8.7 The aperture problem: the black and grey lines show two positions of the same image line in two consecutive frames. The image velocity perceived in (a) through the small aperture, $\mathbf{v}_n$, is only the component parallel to the image gradient of the true image velocity, $\mathbf{v}$, revealed in (b).

The Aperture Problem

The component of the motion field in the direction *orthogonal* to the spatial image gradient is not constrained by the image brightness constancy equation.

The aperture problem can be visualized as follows. Imagine to observe a thin, black rectangle moving against a white background through a small aperture. "Small" means that the corners of the rectangle are not visible through the aperture (Figure 8.7(a)); the small aperture simulates the narrow support of a differential method. Clearly, there are many, actually infinite, motions of the rectangle compatible with what you see through the aperture (Figure 8.7(b)); the visual information available is only sufficient to determine the velocity in the direction *orthogonal* to the visible side of the rectangle; the velocity in the *parallel* direction cannot be estimated.

☞ Notice that the parallel between (8.17) and Figure 8.7 is not perfect. Equation (8.17) relates the image gradient and the motion field at the *same* image point, thereby establishing a constraint on an *infinitely small* spatial support; instead, Figure 8.7 describes a state of affairs over a *small but finite* spatial region. This immediately suggests that a possible strategy for solving the aperture problem is to look at the spatial and temporal variations of the image brightness over a neighborhood of each point.[10]

[10] Incidentally, this strategy appears to be adopted by the visual system of primates.

8.3.3 The Validity of the Constancy Equation: Optical Flow

How well does (8.17) estimate the normal component of the motion field? To answer this question, we can look at the difference, Δv, between the true value and the one estimated by the equation. To do this, we must introduce a model of image formation, accounting for the reflectance of the surfaces and the illumination of the scene.

For the purposes of this discussion, we restrict ourselves to a Lambertian surface, S, illuminated by a pointwise light source infinitely far away from the camera (Chapter 2). Therefore, ignoring photometric distorsion, we can write the image brightness, E, as

$$E = \rho \mathbf{I}^\top \mathbf{n}, \tag{8.19}$$

where ρ is the surface albedo, $\mathbf{I}$ identifies the direction and intensity of illumination, and $\mathbf{n}$ is the unit normal to S at $\mathbf{P}$.

Let us now compute the total temporal derivative of both sides of (8.19). The only quantity that depends on time on the right hand side is the normal to the surface. If the surface is moving relative to the camera with translational velocity $\mathbf{T}$ and angular velocity ω, the orientation of the normal vector $\mathbf{n}$ will change according to

$$\frac{d\mathbf{n}}{dt} = \omega \times \mathbf{n}, \tag{8.20}$$

where × indicates vector product. Therefore, taking the total temporal derivative of both sides of (8.19), and using (8.17) and (8.20), we have

$$\nabla E^\top \mathbf{v} + E_t = \rho \mathbf{I}^\top (\omega \times \mathbf{n}). \tag{8.21}$$

We can obtain the desired expression for Δv from (8.18) and (8.21):

$$|\Delta v| = \rho \frac{|\mathbf{I}^\top \omega \times \mathbf{n}|}{\|\nabla E\|}.$$

We conclude that, even under the simplifying assumption of Lambertian reflectance, the image brightness constancy equation yields the true normal component of the motion field (that is, $|\Delta v|$ is identically 0 for every possible surface) only for (a) purely translational motion, or (b) for any rigid motion such that the illumination direction is parallel to the angular velocity.

Other factors being equal, the difference Δv decreases as the magnitude of the spatial gradient increases; this suggests that *points with high spatial image gradient are the locations at which the motion field can be best estimated by the image brightness constancy equation.*

In general, $|\Delta v|$ is unlikely to be identically zero, and *the apparent motion of the image brightness is almost always different from the motion field*. For this reason, to avoid confusion, we call the apparent motion an *optical flow*, and refer to techniques estimating the motion field from the image brightness constancy equation as *optical flow techniques*. Here is a summary of similarities and differences between motion field and optical flow.

Definition: Optical Flow

The *optical flow* is a vector field subject to the constraint (8.17). and loosely defined as the *apparent motion* of the image brightness pattern.

Optical Flow and Motion Field

The optical flow is the *approximation of the motion field* which can be computed from time-varying image sequences. Under the simplifying assumptions of

- Lambertian surfaces
- pointwise light source at infinity
- no photometric distortion

the *error* of this approximation is

- *small* at points with high spatial gradient
- *exactly zero* only for translational motion or for any rigid motion such that the illumination direction is parallel to the angular velocity

We are now ready to learn algorithms estimating the motion field.

8.4 Estimating the Motion Field

The estimation of the motion field is a useful starting point for the solution of many motion problems. The many techniques devised by the computer vision community can be roughly divided into two major classes: *differential techniques* and *matching techniques*. Differential techniques are based on the spatial and temporal variations of the image brightness at all pixels, and can be regarded as methods for computing optical flow. Matching techniques, instead, estimate the disparity of special image points (features) between frames. We examine differential techniques in section 8.4.1; matching is the theme of section 8.4.2.

8.4.1 Differential Techniques

In recent (and not so recent) years a large number of differential techniques for computing optical flow have been proposed. Some of them require the solution of a system of partial differential equations, others the computation of second and higher-order derivatives of the image brightness, others again least-squares estimates of the parameters characterizing the optical flow. Methods in the latter class have at least two advantages over those in the first two:

- They are not iterative; therefore, they are genuinely local, and less biased than iterative methods by possible discontinuities of the motion field.
- They do not involve derivatives of order higher than the first; therefore, they are less sensitive to noise than methods requiring higher-order derivatives.

We describe a differential technique that gives good results. The basic assumption is that the motion field is well approximated by a *constant* vector field, $\mathbf{v}$, within any small region of the image plane.[11]

Assumptions

1. The image brightness constancy equation yields a good approximation of the normal component of the motion field.
2. The motion field is well approximated by a *constant* vector field within any small patch of the image plane.

An Optical Flow Algorithm. Given Assumption 1, for each point $\mathbf{p}_i$ within a small, $N \times N$ patch, Q, we can write

$$(\nabla E)^\top \mathbf{v} + E_t = 0$$

where the spatial and temporal derivatives of the image brightness are computed at $\mathbf{p}_1, \mathbf{p}_2 \dots \mathbf{p}_{N^2}$.

☞ A typical size of the "small patch" is 5 × 5.

Therefore, the optical flow can be estimated within Q as the constant vector, $\bar{\mathbf{v}}$, that minimizes the functional

$$\Psi[\mathbf{v}] = \sum_{\mathbf{p}_i \in Q} \left[(\nabla E)^\top \mathbf{v} + E_t \right]^2 .$$

The solution to this least squares problem can be found by solving the linear system

$$A^\top A \mathbf{v} = A^\top \mathbf{b}. \tag{8.22}$$

The i-th row of the $N^2 \times 2$ matrix A is the spatial image gradient evaluated at point $\mathbf{p}_i$:

$$A = \begin{bmatrix} \nabla E(\mathbf{p}_1) \\ \nabla E(\mathbf{p}_2) \\ \cdot \\ \cdot \\ \cdot \\ \nabla E(\mathbf{p}_{N \times N}) \end{bmatrix}, \tag{8.23}$$

and $\mathbf{b}$ is the N^2-dimensional vector of the partial temporal derivatives of the image brightness, evaluated at $\mathbf{p}_1, \dots \mathbf{p}_{N^2}$, after a sign change:

$$\mathbf{b} = -\left[E_t(\mathbf{p}_1), \dots, E_t(\mathbf{p}_{N \times N})\right]^\top . \tag{8.24}$$

[11] Notice that this is in agreement with the first conclusion of section 8.2.3 (motion field of moving planes) regarding the approximation of smooth motion fields.

The least squares solution of the overconstrained system (8.22) can be obtained as[12]

$$\bar{\mathbf{v}} = (A^\top A)^{-1} A^\top \mathbf{b}. \tag{8.25}$$

$\bar{\mathbf{v}}$ is the optical flow (the estimate of the motion field) at the center of patch Q; repeating this procedure for all image points, we obtain a dense optical flow. We summarize the algorithm as follows:

Algorithm CONSTANT_FLOW

The input is a time-varying sequence of n images, $E_1, E_2, \dots E_n$. Let Q be a square region of $N \times N$ pixels (typically, $N = 5$).

1. Filter each image of the sequence with a Gaussian filter of standard deviation equal to σ_s (typically $\sigma_s = 1.5$ pixels) along each spatial dimension.
2. Filter each image of the sequence along the temporal dimension with a Gaussian filter of standard deviation σ_t (typically $\sigma_t = 1.5$ frames). If $2k + 1$ is the size of the temporal filter, leave out the first and last k images.
3. For each pixel of each image of the sequence:
   (a) compute the matrix A and the vector $\mathbf{b}$ using (8.23) and (8.24)
   (b) compute the optical flow using (8.25)

The output is the optical flow computed in the last step.

☞ The purpose of spatial filtering is to attenuate noise in the estimation of the spatial image gradient; temporal filtering prevents aliasing in the time domain. For the implementation of the temporal filtering, imagine to stack the images one on top of the other, and filter sequences of pixels having the same coordinates. Note that the size of the temporal filter is linked to the maximum speed that can be "measured" by the algorithm.

An Improved Optical Flow Algorithm. We can improve CONSTANT_FLOW by observing that the error made by approximating the motion field at $\mathbf{p}$ with its estimate at the center of a patch increases with the distance of $\mathbf{p}$ from the center itself. This suggests a *weighted* least-square algorithm, in which the points close to the center of the patch are given more weight than those at the periphery. If W is the weight matrix, the solution, $\bar{\mathbf{v}}_w$, is given by

$$\bar{\mathbf{v}}_w = (A^\top W^2 A)^{-1} A^\top W^2 \mathbf{b}.$$

Concluding Remarks on Optical Flow Methods. It is instructive to examine the image locations at which CONSTANT_FLOW fails. As we have seen in Chapter 4, the 2×2 matrix

$$A^\top A = \begin{pmatrix} \sum E_x^2 & \sum E_x E_y \\ \sum E_x E_y & \sum E_y^2 \end{pmatrix}, \tag{8.26}$$

[12] See Appendix, section A.6 for alternative ways of solving overconstrained linear systems.

computed over an image region Q, is singular if and only if all the spatial gradients in Q are null or parallel. In this case the aperture problem cannot be solved, and the only possibility is to pick the solution of minimum norm, that is, the normal flow. The fact that we have already met the matrix $A^\top A$ in Chapter 4 is not a coincidence; the next section tells you why.

Notice that CONSTANT_FLOW gives good results because the spatial structure of the motion field of a rigid motion is well described by a low-degree polynomial in the image coordinates (as shown in section 8.2.3). For this reason, the assumption of local constancy of the motion field over small image patches is quite effective.

8.4.2 Feature-based Techniques

The second class of methods for estimating the motion field is formed by so-called *matching techniques*, which estimate the motion field at feature points only. The result is a sparse motion field. We start with a two-frame analysis (finding feature disparities between consecutive frames), then illustrate how *tracking* the motion of a feature across a long image sequence can improve the robustness of frame-to-frame matching.

Two-Frame Methods: Feature Matching. If motion analysis is restricted to two consecutive frames, the same matching methods can be used for stereo and motion.[13] This is true for both correlation-based and feature-based methods (Chapter 7). Here we concentrate on *matching feature points*. You can easily adapt this method for the stereo case too.

The point-matching method we describe is reminiscent of the CONSTANT_FLOW algorithm, and based on the features we met in Chapter 4. There, we looked at the matrix $A^\top A$ of (8.26), computed over small, square image regions: the features were the centers of those regions for which the smallest eigenvalue of $A^\top A$ was larger than a threshold. The idea of our matching method is simple: compute the displacement of such feature points by iterating algorithm CONSTANT_FLOW.

The procedure consists of three steps. First, the uniform displacement of the square region Q is estimated through CONSTANT_FLOW, and added to the current displacement estimate (initially set to 0). Second, the patch Q is *warped* according to the estimated flow. This means that Q is displaced according to the estimated flow, and the resulting patch, Q', is resampled in the pixel grid of frame I_2. If the estimated flow equals (v_x, v_y), the gray value at pixel (i, j) of Q' can be obtained from the gray values of the pixels of Q close to $(i - v_y, j - v_x)$. For our purpose, bilinear interpolation[14] is sufficient. Third, the first and second steps are iterated until a stopping criterion is met. Here is the usual algorithm box, containing an example of stopping criterion.

[13] But keep in mind the discussion of section 8.2.1 on the differences between stereo and motion disparities.

[14] Bilinear interpolation means that the interpolation is linear in each of the four pixels closest to $(i - v_y, j - v_x)$.