Optical Flow and Motion Analysis

Ying Wu
Electrical Engineering & Computer Science
Northwestern University
Evanston, IL 60208
yingwu@ece.northwestern.edu

Contents

1 Motion Fields and Optical Flow

1.1 Notes

Motion is one of the most important research topics in computer vision. It is the base for many other problems such as visual tracking, structure from motion, 3D reconstruction, video representation, and even video compression. There are several important issue to investigate:

- 2D and 3D motion representations
- calculating 2D motion
- inferring 3D motion
- structure and motion

A major interests of motion analysis is to estimate 3D motion. The motion analysis tasks could be roughly categorized into three different settings: 2D-2D, 2D-3D, and 3D-3D, depending on the correspondences. The 3D-3D problem is to calculate 3D motion based on a set of 3D correspondences. But generally, direct 3D data are difficult to obtain. To ease the problem, we can assume 2D-3D correspondences. The 2D-3D problem is to determine 3D motion based on the correspondence between 3D model and 2D image projections. 3D model-based analysis is one of such examples. Without using any 3D models, 2D-2D analysis only assume the correspondences between 2D image projections but aim at calculating 3D motion from such 2D correspondences.

A critical but difficult problem for motion analysis, obviously, is constructing correspondences. Correspondences could be in totally different forms, e.g., point correspondences, line correspondences, curve correspondences, even region correspondences. Sometimes, we can easily get some geometrical primitives from images, but sometimes not.

According to my understanding, there are two major methodologies: "dense" approach, and "sparse" or "feature-based" approach. The dense approach tries to build correspondences pixel by pixel, while feature-based approach tries to associate different image features. These two ideas result in totally different taste of motion and structure analysis. In this lecture, let's get some feelings about the "dense" approach.

1.2 2D Motion Fields

A velocity vector is associated to each image point, and a collection of such velocity vectors is a 2D motion field. In Figure 1, $\mathbf{p}$ is a 3D point, i.e., $\mathbf{p} = [X, Y, Z]^T$, and $\mathbf{m}$ is its image projection, i.e., $\mathbf{m} = [x, y]^t$. Then we have:

$$\mathbf{P} = Z\hat{\mathbf{m}}$$

where $\hat{\mathbf{m}}$ is the homogeneous coordinate of $\mathbf{m}$. Then we have:

$$\frac{d\mathbf{p}}{dt} = \frac{dZ}{dt}\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} + Z\begin{bmatrix} dx/dt \\ dy/dt \\ 0 \end{bmatrix}$$

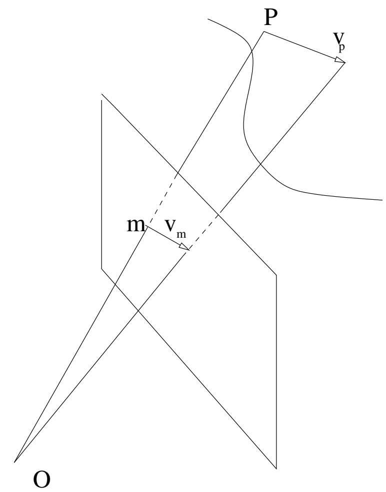

Figure 1: Motion Field

i.e.,

$$\dot{\mathbf{p}} = \dot{Z}\hat{\mathbf{m}} + Z\dot{\hat{\mathbf{m}}}$$

i.e.,

$$\mathbf{V}_p = (\mathbf{V}_p^T\mathbf{k})\hat{\mathbf{m}} + Z\mathbf{v}_m$$

where $\mathbf{k}$ is the unit vector of the depth direction. So,

$$\mathbf{v}_m = \frac{1}{Z}(\mathbf{V}_p - (\mathbf{V}_p^T\mathbf{k})\hat{\mathbf{m}})$$

which means that the 2D motion field $\mathbf{v}_m$ is a function of $\mathbf{V}_p/Z$.

1.3 Optical Flow

Unfortunately, we can not observe such motion field directly, since we have no idea of how the image projection of a 3D point moves. On the other hand, what we can observe are only images, specifically, image points, i.e., under certain assumption, what we can say is that an image point moves from here to there, which indicates optical flow. By definition, optical flow is the apparent motion of the brightness pattern. Obviously,

$$\texttt{motion field} \neq \texttt{optical flow}$$

.

Consider a perfectly uniform sphere. There will be some shadows on the surface. When the sphere rotates, such shading pattern won't move at all. In this case, the optical is zero but motion field is not. On the other hand, we keep the sphere still, but the lighting source moves, which results in the changes in the shading patterns. In this case, the optical flow is not zero but the motion filed is.

But since what we can observe is the optical flow, we expect in many cases that optical flow is not too different from the motion field. This is our underline assumption!

2 Calculating Optical Flow

2.1 Optical Flow Constraint Equation

We denote an image by $I(x, y, t)$, and the velocity of an image pixel $\mathbf{m} = [x, y]^T$ is

$$\mathbf{v}_m = \dot{\mathbf{m}} = [v_x, v_y]^T = \begin{bmatrix} dx/dt \\ dy/dt \end{bmatrix}$$

Assuming the intensity of $\mathbf{m}$ keeps the same during dt, i.e.,

$$I(x + v_x dt, y + v_y dt, t + dt) = I(x, y, t)$$

If the brightness changes smoothly with x,y and t, we expend the left-hand-side by Taylor series:

$$I(x, y, t) + \frac{\partial I}{\partial x} v_x dt + \frac{\partial I}{\partial y} v_y dt + \frac{\partial I}{\partial t} dt + O(dt^2) = I(x, y, t)$$

So, we have

$$\frac{\partial I}{\partial x} v_x + \frac{\partial I}{\partial y} v_y + \frac{\partial I}{\partial t} = 0 \tag{1}$$

i.e.

$$\nabla I \cdot \mathbf{v}_m + \frac{\partial I}{\partial t} = 0$$

where $\nabla I = [\frac{\partial I}{\partial x}, \frac{\partial I}{\partial y}]^T$ is image gradient at pixel $\mathbf{m}$, which can be obtained from images. Also $\frac{\partial I}{\partial t}$ can also be obtained from images easily. We call this equation *optical flow constraint equation*.

Apparently, for each pixel, we have only one constraint equation, but we need to solve two unknowns, i.e., v_x and v_y, which means that we CAN'T determine optical flow uniquely only from such optical flow constraint equation. Figure 2 gives a geometrical explanation of the constraint equation.

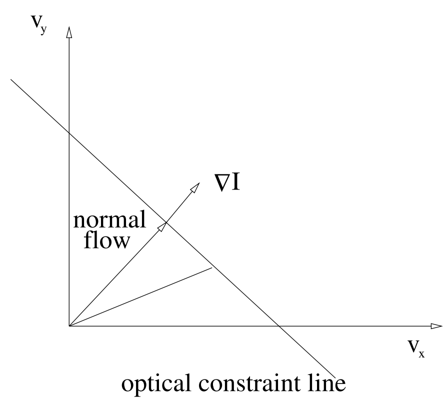

Figure 2: Geometrical explanation of the optical flow constraint equation. The optical flow for a given image pixel can be any point on the constraint line in v_x-v_y plane.

Given this constrain equation, we can only determine the "normal flow", i.e., the flow along the direction of image gradient, but we can not determine those flow on the tangent direction of isointensity contour, i.e., the direction perpendicular to the image gradient. This is so called "aperture problem".

2.2 Aperture Problem

The aperture problem here means that we can not determine the flow perpendicular to the image gradient, i.e., the tagent of isointensity contour. Two examples are given in Figure 3. Suppose the contours are isointensity contours. Many points around p have the

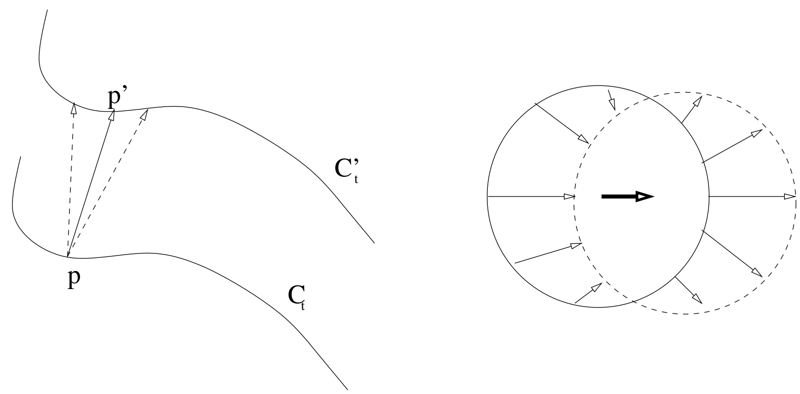

Figure 3: The aperture problem in optical flow.

same intensity. So, it is impossible to determine which point p' on C'_t match point p on C_t. In the other example, we can only determine the normal flow along the direction of ∇I.

As a result, to determine the optical flow uniquely, we need some other constraints. Below, we describe two approaches: the first one introduces a global smoothness constraint, and the second one employs a local constraint for each pixel.

2.3 A Global Method: Horn-Schunck's Method

The assumption made in this method is that optical flow varies smoothly, i.e., the variation of the optical flow field can not be too big. Apparently, this is a global requirement for the whole image. Such smoothness constraint is indicated by the derivatives of optical flow, i.e., ∇v_x and ∇v_y. The measure of departure from smoothness can be written by:

$$\begin{aligned} e_s &= \int\int (||\nabla v_x||^2 + ||\nabla v_y||^2) dxdy \qquad (2) \\ &= \int\int (\frac{\partial v_x}{\partial x})^2 + (\frac{\partial v_x}{\partial y})^2 + (\frac{\partial v_y}{\partial x})^2 + (\frac{\partial v_y}{\partial xy})^2 dxdy \qquad (3) \end{aligned}$$

The error of optical flow is:

$$e_c = \int\int (\nabla I \cdot \mathbf{v}_m + \frac{\partial I}{\partial t})^2 dxdy \qquad (4)$$

So, we want to minimize:

$$\begin{aligned} e &= e_c + \lambda e_s \\ &= \int\int (\nabla I \cdot \mathbf{v}_m + \frac{\partial I}{\partial t})^2 + \lambda(||\nabla v_x||^2 + ||\nabla v_y||^2) dxdy \end{aligned}$$

The 4-neighbor of an image pixel (i,j) is (i-1,j), (i+1,j), (i,j-1), (i,j+1). So, we can write the discrete version of smoothness error for each pixel:

$$\begin{aligned}
s(i,j) &= \frac{1}{4}[(v_x(i,j)-v_x(i-1,j))^2+(v_x(i+1,j)-v_x(i,j))^2 \\
&+ (v_x(i,j+1)-v_x(i,j))^2+(v_x(i,j)-v_x(i,j-1))^2 \\
&+ (v_y(i,j)-v_y(i-1,j))^2+(v_y(i+1,j)-v_y(i,j))^2 \\
&+ (v_y(i,j+1)-v_y(i,j))^2+(v_y(i,j)-v_y(i,j-1))^2]
\end{aligned}$$

And

$$c(i,j)=\left[\frac{\partial I}{\partial x}v_x(i,j)+\frac{\partial I}{\partial y}v_y(i,j)+\frac{\partial I}{\partial t}\right]^2$$

So, we need to

$$\min E=\sum_i\sum_j[c(i,j)+\lambda s(i,j)]$$

Calculating the derivatives, we have:

$$\begin{aligned}
\frac{\partial E}{\partial v_x(i,j)} &= 2\left(\frac{\partial I}{\partial x}v_x(i,j)+\frac{\partial I}{\partial y}v_y(i,j)+\frac{\partial I}{\partial t}\right)\frac{\partial I}{\partial x}+2\lambda(v_x(i,j)-\bar{v}_x(i,j))=0 \\
\frac{\partial E}{\partial v_y(i,j)} &= 2\left(\frac{\partial I}{\partial x}v_x(i,j)+\frac{\partial I}{\partial y}v_y(i,j)+\frac{\partial I}{\partial t}\right)\frac{\partial I}{\partial y}+2\lambda(v_y(i,j)-\bar{v}_y(i,j))=0
\end{aligned}$$

where $\bar{v}_x$ and $\bar{v}_y$ are local average of v_x and v_y. So, we can write:

$$\begin{aligned}
\left[\lambda+\left(\frac{\partial I}{\partial x}\right)^2\right]v_x+\frac{\partial I}{\partial x}\frac{\partial I}{\partial y}v_y &= \lambda\bar{v}_x-\frac{\partial I}{\partial x}\frac{\partial I}{\partial t} \\
\frac{\partial I}{\partial x}\frac{\partial I}{\partial y}v_x+\left[\lambda+\left(\frac{\partial I}{\partial y}\right)^2\right]v_y &= \lambda\bar{v}_y-\frac{\partial I}{\partial y}\frac{\partial I}{\partial t}
\end{aligned}$$

which suggest an iterative scheme:

$$v_x^{k+1} = \bar{v}_x^k-\left[\frac{\left(\frac{\partial I}{\partial x}\right)\bar{v}_x^k+\left(\frac{\partial I}{\partial y}\right)\bar{v}_y^k+\frac{\partial I}{\partial t}}{\lambda+\left(\frac{\partial I}{\partial x}\right)^2+\left(\frac{\partial I}{\partial y}\right)^2}\right]\frac{\partial I}{\partial x} \tag{5}$$

$$v_y^{k+1} = \bar{v}_y^k-\left[\frac{\left(\frac{\partial I}{\partial x}\right)\bar{v}_x^k+\left(\frac{\partial I}{\partial y}\right)\bar{v}_y^k+\frac{\partial I}{\partial t}}{\lambda+\left(\frac{\partial I}{\partial x}\right)^2+\left(\frac{\partial I}{\partial y}\right)^2}\right]\frac{\partial I}{\partial y} \tag{6}$$

Concisely, it is:

$$\mathbf{v}^{k+1}=\bar{\mathbf{v}}^k-\alpha(\nabla I) \tag{7}$$

For each iteration, the new optical flow field is constrained by its local average and the optical flow constraints.

2.4 A Local Method: Locas-Kanade's Method

Horn-Schunck's method introduces a regularization term and it is a global method. Let's look at a local approach. Assume we can use a constant model to describe the optical flow in a small window Ω. We define a window function $W(\mathbf{m}), \mathbf{m} \in \Omega$. We let the weight of the center bigger then others, i.e., the window function favors the center. The optical flow of the center pixel can be calculated by

$$\min_{\mathbf{v}} E = \sum_{\mathbf{m} \in \Omega} W^2(\mathbf{m}) \left(\nabla I \cdot \mathbf{v} + \frac{\partial I}{\partial x} \right)^2$$

Writing out the derivatives, we have:

$$\begin{aligned} \frac{\partial E}{\partial v_x} &= \sum W^2(\mathbf{m}) \left(\frac{\partial I}{\partial x} v_x + \frac{\partial I}{\partial y} v_y + \frac{\partial I}{\partial t} \right) \frac{\partial I}{\partial x} = 0 \\ \frac{\partial E}{\partial v_y} &= \sum W^2(\mathbf{m}) \left(\frac{\partial I}{\partial x} v_x + \frac{\partial E}{\partial y} v_y + \frac{\partial I}{\partial t} \right) \frac{\partial I}{\partial y} = 0 \end{aligned}$$

To solve it, we let:

$$\mathbf{A} = \begin{bmatrix} \frac{\partial I_1}{\partial x_1} & \frac{\partial I_1}{\partial y_1} \\ \vdots & \vdots \\ \frac{\partial I_N}{\partial x_N} & \frac{\partial I_N}{\partial y_N} \end{bmatrix}_{N \times 2}$$

$$\mathbf{W} = diag(W(\mathbf{m}_1), \ldots, W(\mathbf{m}_N))_{N \times N}$$

$$\mathbf{v} = \left[\frac{\partial x}{\partial t}, \frac{\partial y}{\partial t} \right]^T = [v_x, v_y]^t$$

$$\mathbf{b} = - \begin{bmatrix} \frac{\partial I_1}{\partial t} \\ \vdots \\ \frac{\partial I_N}{\partial t} \end{bmatrix}_{N \times 1}$$

Using such notations, we have

$$\mathbf{A}^T \mathbf{W}^2 \mathbf{A} \mathbf{v} = \mathbf{A}^T \mathbf{W}^2 \mathbf{b}$$

So, the flow for the image pixel $\mathbf{m}$ can be solved, i.e.,

$$\mathbf{v} = (\mathbf{A}^T \mathbf{W}^2 \mathbf{A})^{-1} \mathbf{A}^T \mathbf{W}^2 \mathbf{b} \tag{8}$$

We should note here that if $\mathbf{A}^T \mathbf{W}^2 \mathbf{A}$ is not singular, we can uniquely determine the optical flow for image pixel $\mathbf{m}$. We should be aware that the reliability of the estimation of $\mathbf{v}$ is revealed by the eigenvalues of $\mathbf{A}^T \mathbf{W}^2 \mathbf{A}$ (assume $\lambda_1 \leq \lambda_2$). If both eigenvalues are large, then the flow can be uniquely determined; if λ_1 is much larger than λ_2, only the normal flow can be determined; if $\lambda_2 = 0$, the flow can not be determined at all.

Since Lucas-Kanade's method use a local window to determine the flow of a particular image point, this is the reason that it is called a local method. Intuitively, Lucas-Kanade's method calculate the flow of a point $\mathbf{m}$ by identifying an intersection of all the flow constraint lines corresponding to the image pixels within the window of $\mathbf{m}$. Those flow constraint lines will have an intersection, since this method assumes that the flow within the window is constant.

2.5 Notes

Besides Horn-Schunck's method and Lucas-Kanade's method, the Bayesian approach based on Markov random field (MRF) and Gibbs random field (GRF) was also proposed to calculate optical flow. In this probabilistic frame, different techniques have been investigated, such as iterated conditional mode (ICM), Metropolis sampling and Gibbs sampling, highest confident first (HCF), and mean field methods. I am not going to discuss these methods, but if interested please read some references.

3 Flow-based Motion Analysis

3.1 3D-2D Motion Models

3D velocity vector of a rigid object can be written as

$$\begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{bmatrix} = \begin{bmatrix} 0 & -\Omega_3 & \Omega_2 \\ \Omega_3 & 0 & -\Omega_1 \\ -\Omega_2 & \Omega_1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + \begin{bmatrix} T_x \\ T_y \\ T_x \end{bmatrix} \tag{9}$$

i.e.,

$$\dot{\mathbf{X}} = \mathbf{\Omega} \times X + \mathbf{T}$$

where $\mathbf{\Omega} = [\Omega_1, \Omega_2, \Omega_3]^T$ is the angular velocity vector, and $\mathbf{T} = [T_x, T_y, T_z]$ is the translation velocity vector.

3.1.1 Under Orthographic Projection

Under orthographic projection, we can write the optical flow by:

$$\begin{aligned} v_x &= \dot{x} = \dot{X} = T_x + \Omega_2 Z - \Omega_3 Y \\ v_y &= \dot{y} = \dot{Y} = T_y + \Omega_3 X - \Omega_1 Z \end{aligned}$$

3.1.2 Under Perspective Projection

$$\begin{aligned} v_x &= f\frac{\dot{X}}{Z} - x\frac{\dot{Z}}{Z} = f(\frac{T_x}{Z} + \Omega_2) - \frac{T_z}{Z}x - \Omega_3 y - \frac{\Omega_1}{f}xy + \frac{\Omega_2}{f}x^2 \\ v_y &= f\frac{\dot{Y}}{Z} - y\frac{\dot{Z}}{Z} = f(\frac{T_y}{Z} - \Omega_1) - \Omega_3 x - \frac{T_z}{Z}y + \frac{\Omega_2}{f}xy - \frac{\Omega_1}{f}y^2 \end{aligned}$$

when let $f = 1$, we have

$$\begin{aligned} v_x &= \frac{-T_x + xT_z}{Z} + \Omega_1 xy - \Omega_2(1 + x^2) + \Omega_3 y \\ v_y &= \frac{-T_y + yT_z}{Z} + \Omega_1(1 + y^2) - \Omega_2 xy - \Omega_3 x \end{aligned}$$

Eliminating Z, we end up with a nonlinear equation:

$$
\begin{aligned}
-v_y e_1 + v_x e_2 - x(\Omega_1 + \Omega_3 e_1) - y(\Omega_2 + \Omega_3 e2) - xy(\Omega_2 e_1 + \Omega_1 e_2) + \\
(x^2 + y^2)\Omega_3 + (1 + y^2)\Omega_1 e_1 + (1 + x^2)\Omega_2 e_2 = y v_x - x v_y
\end{aligned}
$$

where $e_1 = T_x/T_z$ and $e_2 = T_y/T_z$. In these equations, e_1,e_2,Ω_1,Ω_2,Ω_3 are unknowns. x and y are image coordinates, and v_x and v_y are optical flow. Since this nonlinear equation has 5 unknowns, i.e., $\{e_1, e_2, \Omega_1, \Omega_2, \Omega_3\}$:

- there are at most 10 solutions with 5 optical flow vectors;
- optical flow at 6 or more points almost always determine 3D motion uniquely;
- if the motion is pure rotation, then it is uniquely determined by two optical flow values;
- in the case of 3D planar surface, optical flow at 4 points almost always gives two solutions.

3.2 Optical Flow Models and Fitting

3.2.1 Affine Flow

Affine model is under two assumptions:

- planar surface
- orthographic projection

We can write a 3D plan by $Z = AX + BY + C$. Then we get 6-parameter affine flow model:

$$
\begin{bmatrix} v_x \\ v_y \end{bmatrix} = \begin{bmatrix} a_1 & a_2 \\ a_3 & a_4 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} a_5 \\ a_6 \end{bmatrix} \tag{10}
$$

where

$$
\begin{aligned}
a_1 &= A\Omega_2,\ a_2 = B\Omega_2 - \Omega_3,\ a_5 = T_x + C\Omega_2, \\
a_3 &= \Omega_2 - A\Omega_1,\ a_4 = -B\Omega_1,\ a_6 = T_y - C\Omega_1
\end{aligned}
$$

In this case, flow can be determined by at least 3 points.

3.2.2 Quadratic Flow

Quadratic model is under two assumptions:

- planar surface
- perspective projection

Under perspective projection, a plane can be written as

$$\frac{1}{Z} = \frac{1}{C} - \frac{A}{C}X - \frac{B}{C}Y$$

So, we have

$$\begin{aligned} v_x &= a_1 + a_2 x + a_3 y + a_7 x^2 + a_8 xy \\ v_y &= a_4 + a_5 x + a_6 y + a_7 xy + a_8 y^2 \end{aligned}$$

where

$$\begin{aligned} a_1 &= f(\frac{T_x}{C} + \Omega_2),\ a_2 = -(f\frac{T_x A}{C} + \frac{T_z}{C}),\ a_3 = -(f\frac{T_x B}{C} + \Omega_3) \\ a_4 &= f(\frac{T_y}{C} - \Omega_1),\ a_5 = -(f\frac{T_y A}{C} - \Omega_3),\ a_6 = -(f\frac{T_y B}{C} + \frac{T_z}{C}) \\ a_7 &= (\frac{T_z A}{C} + \frac{\Omega_2}{f}),\ a_8 = (\frac{T_z B}{C} - \frac{\Omega_1}{f}) \end{aligned}$$

In this case, if we know at least 4 points on a planar object, we can also $\{a_1, \ldots, a_8\}$.

3.3 Direct Methods

Direct method means that we can replace the optical flow vectors with their estimates in terms of spatial-temporal image intensity gradient.

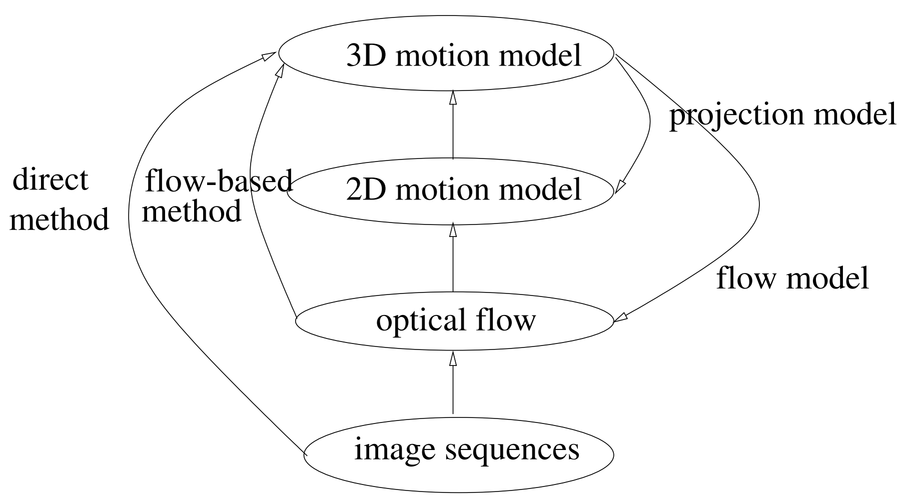

Figure 4: Illustration of the basic idea of direct method for motion analysis.

4 Flow-based Motion Segmentation

4.1 The Problem

Motion segmentation is to segment out different objects according to their motion coherence. The basic problem is illustrated in Figure 5. Since optical flow provides a representation of

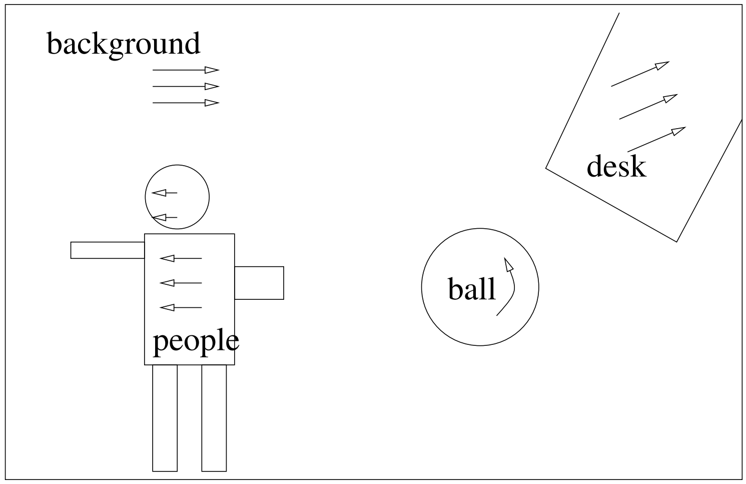

Figure 5: What is motion segmentation?

motion, it is a very good clue for motion segmentation. Optical flow-based motion segmentation has below setup:

- Given: optical flow calculated
- To solve: (a) grouping pixels belong to the same moving object; (b) the number of moving objects

4.2 Ideas and Approaches

Compared to static image segmentation, what kind of ideas will we come up with?

4.2.1 Generalized Hough Transformation

It is straightforward to extend the basic idea of Hough transformation to flow-based motion segmentation. For example, we can use the affine flow model, i.e., optical flow can be parameterized by $\{a_1, \ldots, a_6\}$. When quantizing such a $\mathcal{R}^6$ dimensional parametric space, each image pixel will "vote" for a set of parameters which minimizing

$$e^2(\mathbf{x}) = e_x^2(\mathbf{x}) + e_y^2(\mathbf{x})$$

where $e_x(\mathbf{x}) = v_x - a_1 - a_2x - a_3y$, and $e_y(\mathbf{x}) = v_y - a_4 - a_5x - a_6y$. Obviously, this approach takes time.

4.2.2 Layer and K-Means

The K-means clustering idea can also be adapted to our problem:

1. dividing image into small $m \times m$ blocks;
2. fitting affine flow model to image black pairs B(t) and B(t+1);

3. treating the 6 parameters of the affine model as a training data point;
4. performing K-means clustering on $\mathcal{R}^6$;
5. updating these k affine models;
6. re-labelling (segmentation);
7. fitting affine motion models for each segment;
8. loop between 6 and 7 until it converges.

4.2.3 Bayesian Segmentation

I do not want to say more about Bayesian segmentation, since we have done that before, i.e., EM-based segmentation! But I still want to emphasize the very basic idea:

- Assumption: we know there are k motion model
- Missing data: which motion model does a pixel belong to?
- E-step: estimating the missing data, i.e., classifying each image pixel into different motion models in a probabilistic way;
- M-step: fitting new motion models.