计算机视觉与模式识别

Computer Vision and Pattern Recognition

Motion & Optical Flow

3D motion field, 2D image motion, Apparent motion, Optical flow computation (Nonparametric methods), KLT feature tracking

人工智能与机器人研究所

Institute of Artificial Intelligence and Robotics

袁泽剑

Email: yuan.ze.jian@xjtu.edu.cn

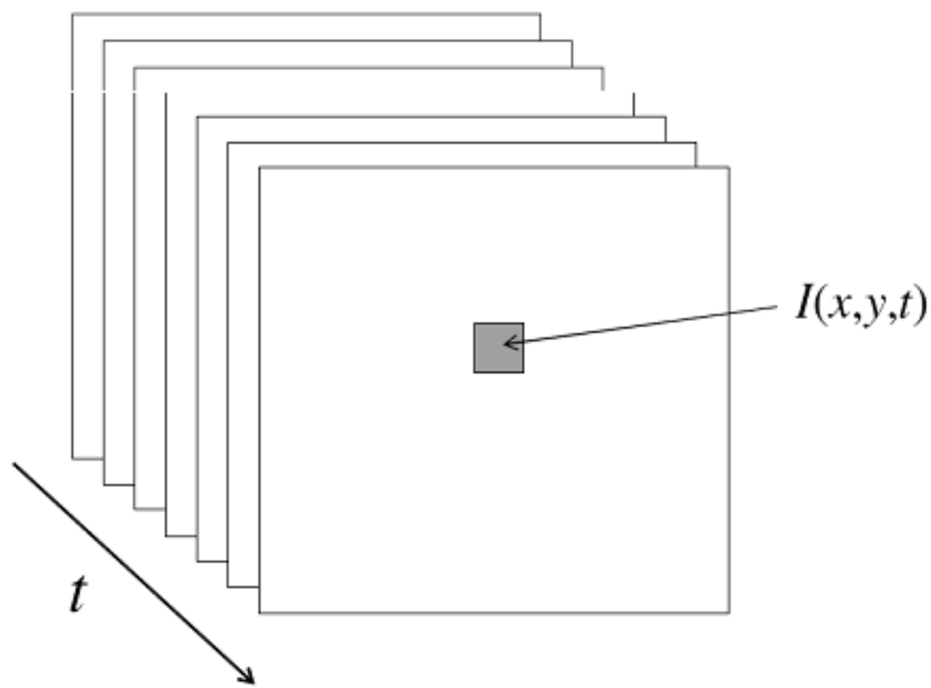
科学馆102室

Outline

- Motion
  - Applications
  - e.g. Time to Collision (TTC)
- **Motion Field (3D&2D motion field)**
  - Derivation
  - Visual navigation & structure for motion
- Apparent motion &Optical Flow (Non-parametric)
  - Brightness constancy constraint
  - Aperture problem
  - Lucas-Kanade flow
  - Iterative refinement
  - Global parametric motion
  - Coarse-to-fine estimation
- KLT Feature Tracking

Video

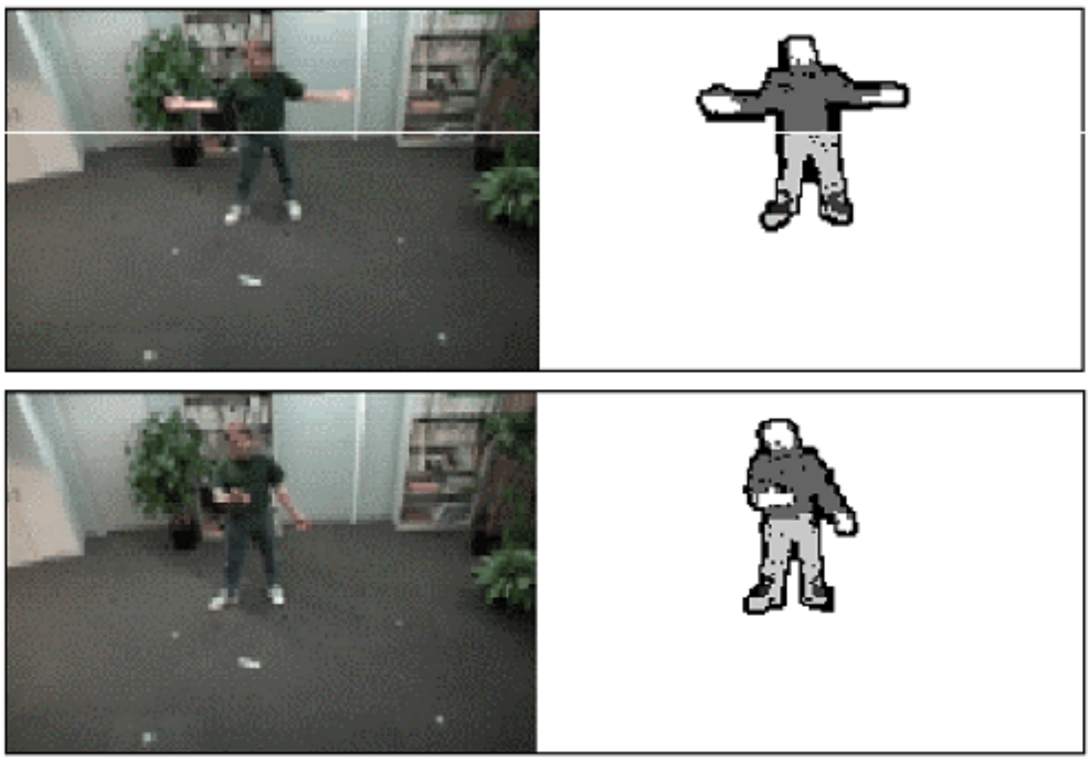
- Video
  - A video is a sequence of frames captured over time
  - The image data is a function of space (x, y) and time (t)



Slide credit: Svetlana Lazebnik

Video Segmentation

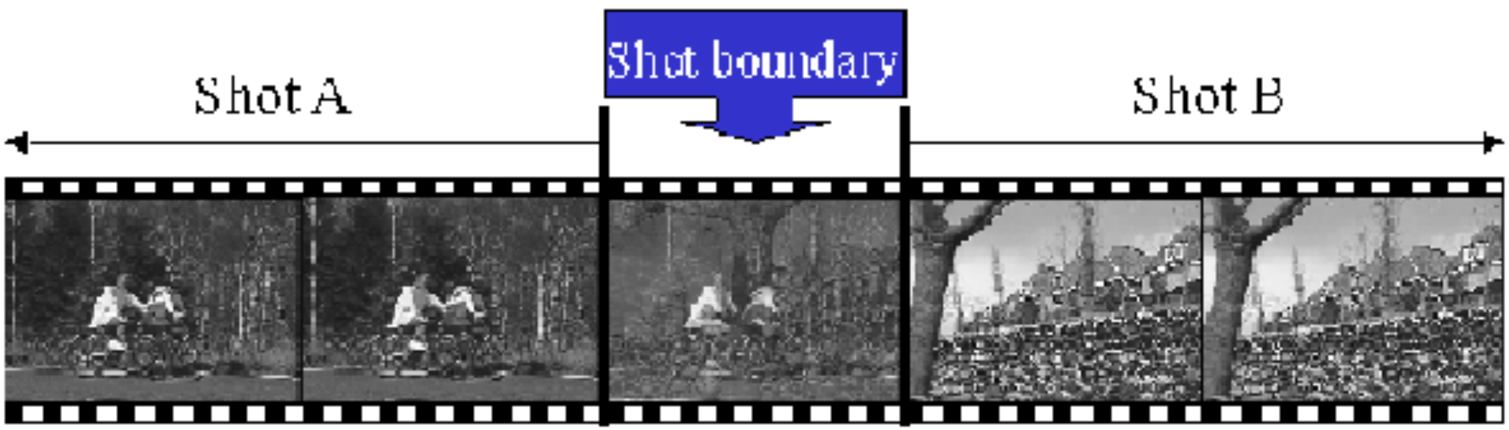
- **Background subtraction**
  - A static camera is observing a scene.
  - Goal: separate the static background from the moving foreground.



Slide credit: S. Lazebnik, K. Grauman

Video Segmentation

- **Shot boundary detection**
  - Commercial video is usually composed of shots or sequences showing the same objects or scene.
  - Goal: segment video into shots for summarization and browsing (each shot can be represented by a single key-frame in a user interface).
  - Difference from background subtraction: the camera is not necessarily stationary.



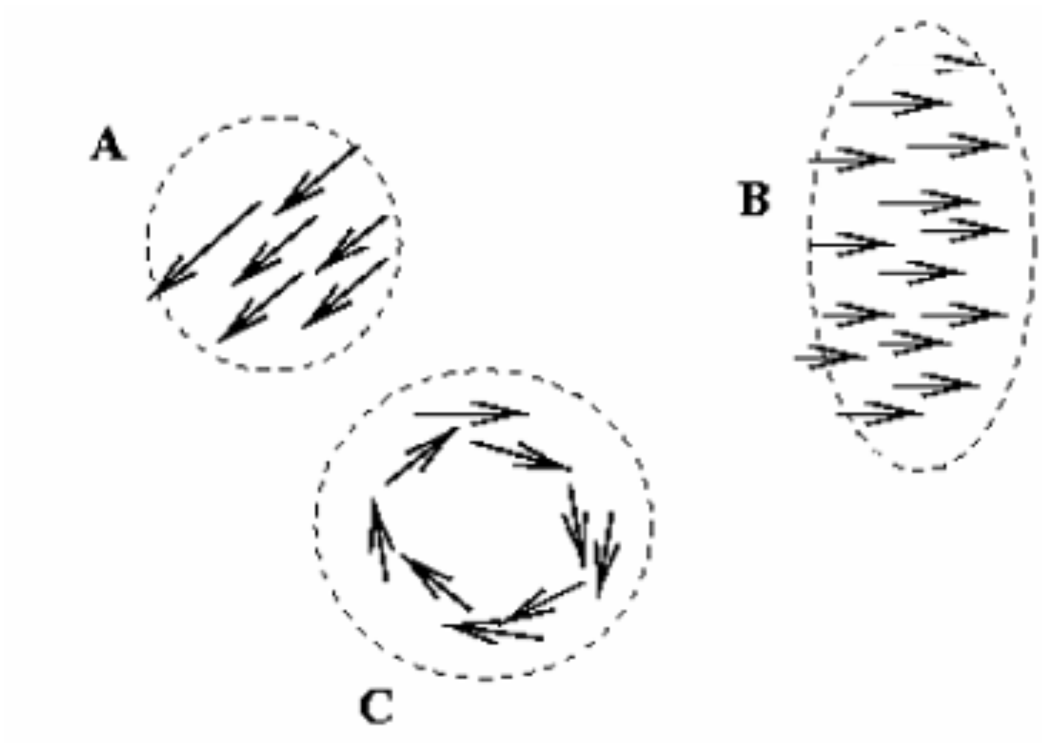
Slide credit: Svetlana Lazebnik

Video Segmentation

- Shot boundary detection
  - For each frame, compute the distance between the current frame and the previous one: (measure between frames)
    - Pixel-by-pixel differences
    - Differences of color histograms
    - Block comparison
  - If the distance is greater than some threshold, classify the frame as a shot boundary.

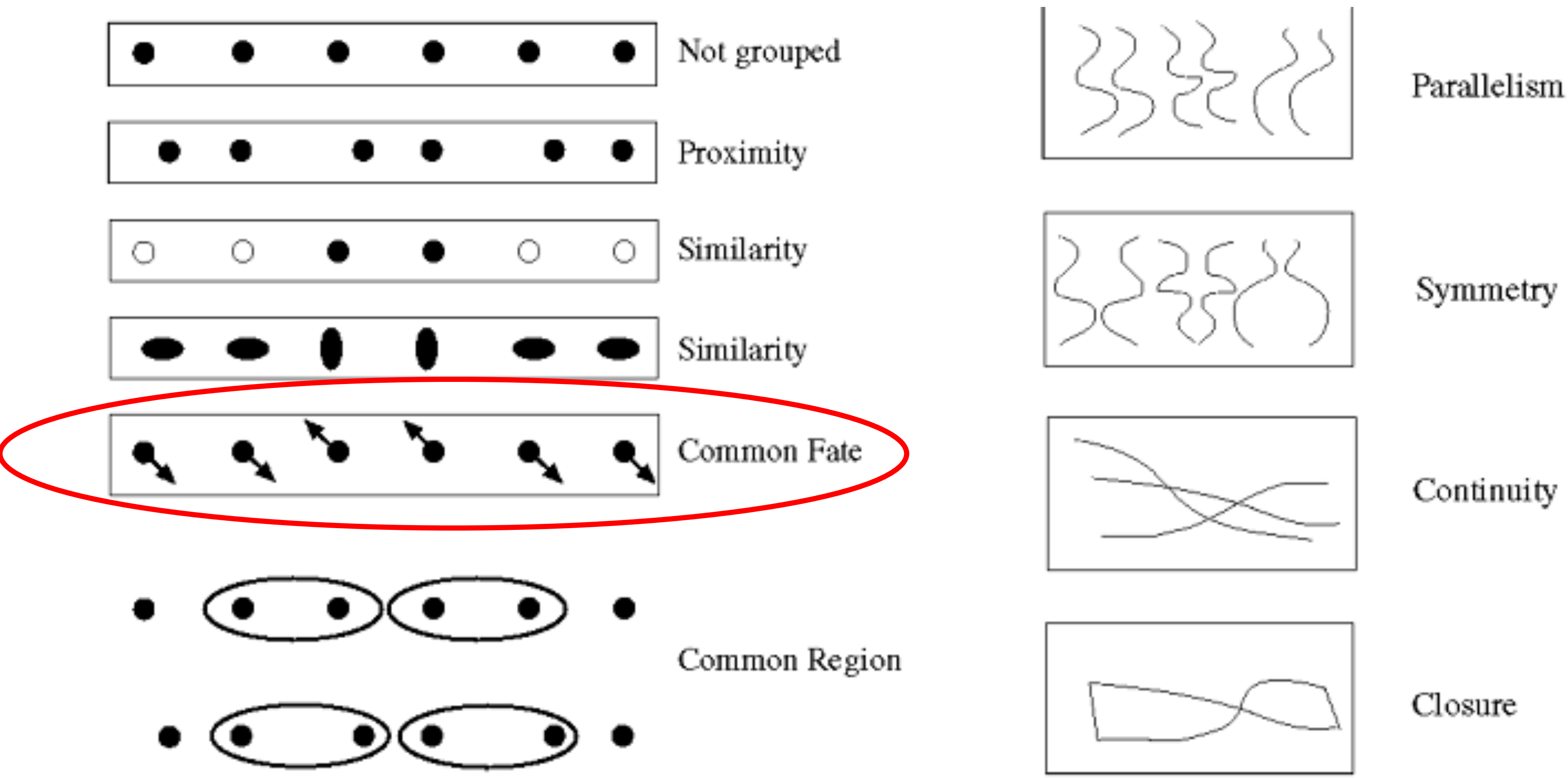
Video Segmentation

- Motion segmentation
  - Segment the video into multiple coherently moving objects



Slide credit: Svetlana Lazebnik

Motion and Perceptual Organization



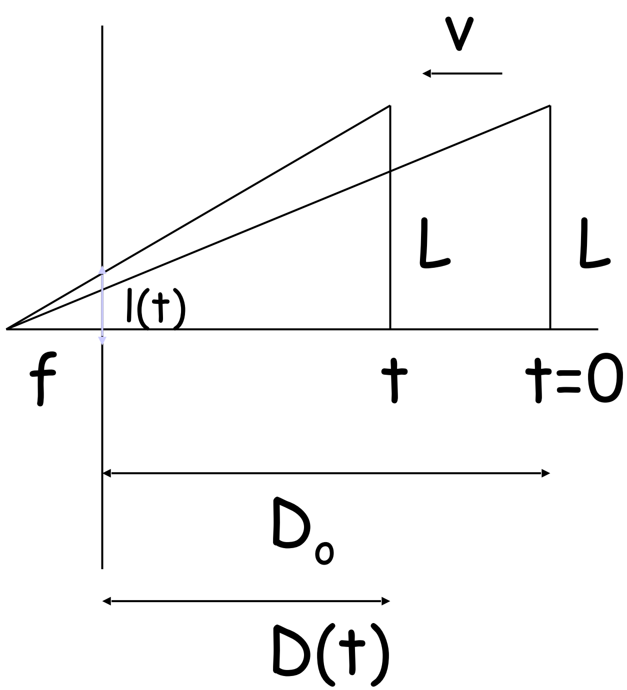
Sometimes, motion is the only cue

Slide credit: Svetlana Lazebnik

Uses of Motion

- Estimating 3D structure
  - Directly from optic flow
  - Indirectly to create correspondences for SfM
- Segmenting objects based on motion cues
- Learning dynamical models
- Recognizing events and activities
- Improving video quality (motion / image stabilization)
- **Time to Collision**

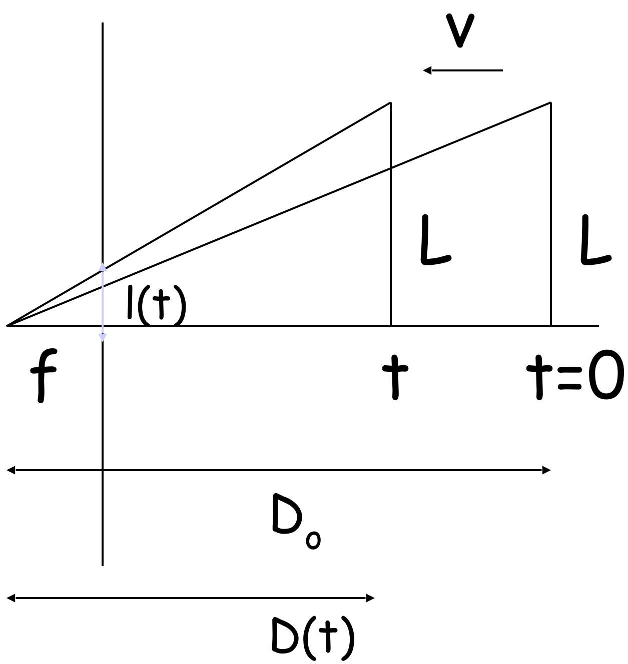
Time to Collision



An object of height L moves with constant velocity v:

- At time t=0 the object is at:
  - $D(0) = D_o$
- At time t it is at
  - $D(t) = D_o - vt$
- It will crash with the camera at time:
  - $D(\tau) = D_o - v\tau = 0$
  - $\tau = D_o/v$

Time to Collision



The image of the object has size l(t):

$$l(t) = \frac{fL}{D(t)}$$

Taking derivative wrt time:

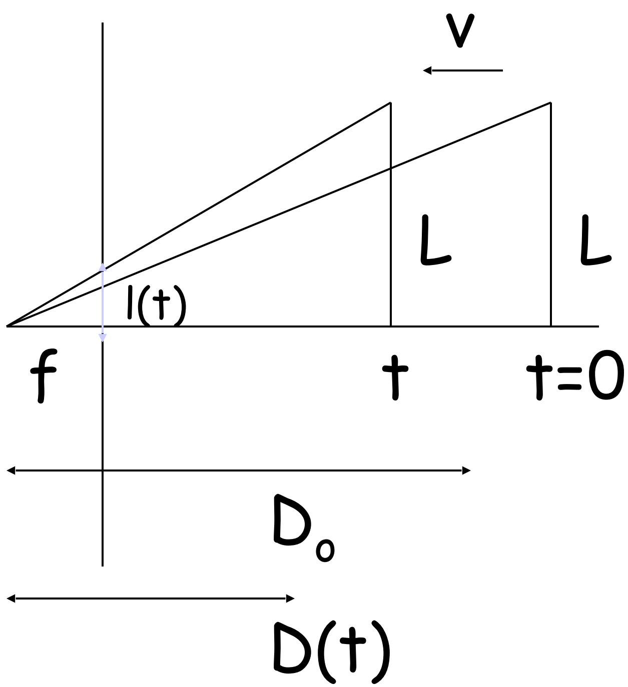
$$l'(t) = \frac{dl(t)}{dt} = fL\frac{d(1/D(t))}{dt}$$

$$l'(t) = fL\frac{-1}{D^2(t)}\frac{d(D(t))}{dt}$$

$$\frac{d(D(t))}{dt} = -v$$

$$l'(t) = fL\frac{v}{D^2(t)}$$

Time to Collision



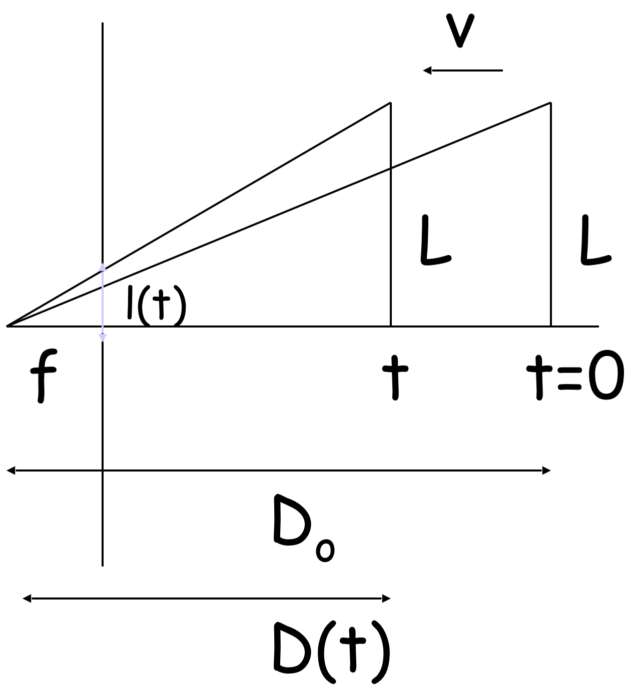
$$l'(t) = fL\frac{v}{D^2(t)}$$

$$l(t) = \frac{fL}{D(t)}$$

And their ratio is:

$$\frac{l(t)}{l'(t)} = \frac{fL}{D(t)}\frac{D^2(t)}{fLv} = \frac{D(t)}{v} = \tau$$

Time to Collision



$$l'(t) = fL\frac{v}{D^2(t)}$$

$$l(t) = \frac{fL}{D(t)}$$

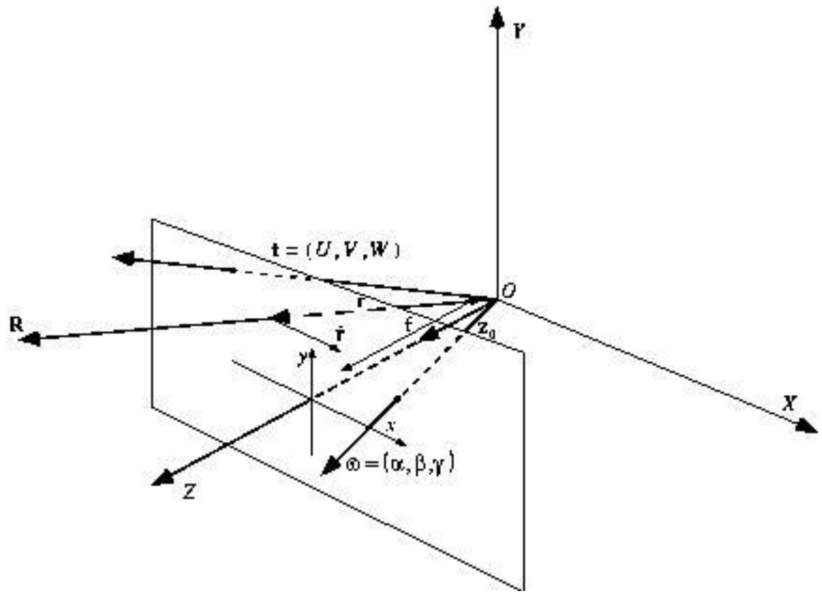
Can be directly measured from image

And time to collision:

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

Can be found, without knowing L or D_o or v !!

Passive Navigation and Structure from Motion



The system moves with a rigid motion with translational velocity

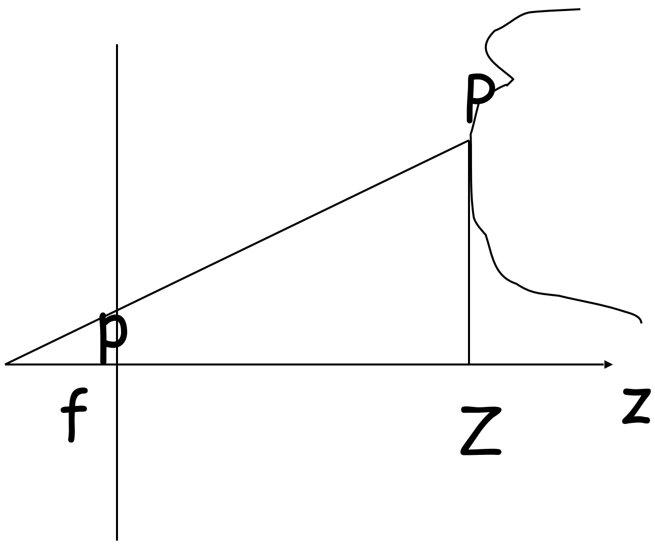
$\mathbf{t} = (t_x, t_y, t_z)^T$ and rotational velocity $\omega = (\omega_x, \omega_y, \omega_z)^T$.

Scene points $\mathbf{R} = (X, Y, Z)^T$ project onto image points $\mathbf{r} = (x, y, f)$

and the 3D velocity $\dot{\mathbf{R}} = (V_x, V_y, V_z)$ of a scene point is observed in the image

as velocity $\dot{\mathbf{r}} = (u, v.0)$

Consider a 3D point P and its image:



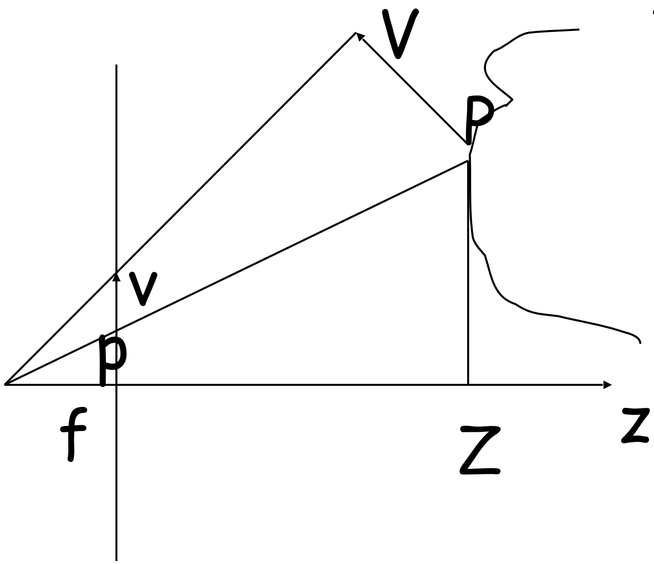
$$P = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

$$p = \begin{bmatrix} x \\ y \\ f \end{bmatrix}$$

Using pinhole camera equation:

$$p = \frac{fP}{Z}$$

Relative motion



The <u>relative velocity</u> of P wrt camera:

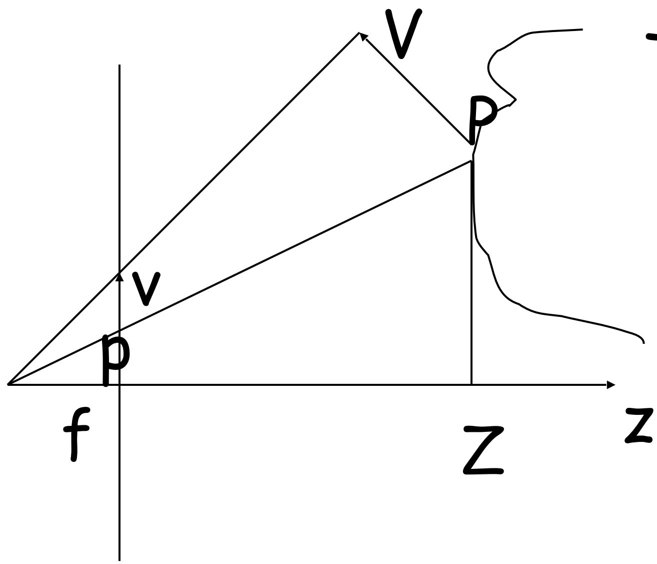
$$V = -t - \omega \times P$$

Translation velocity

Rotation angular velocity

$$t = \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix} \qquad \omega = \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}$$

3D Relative Velocity:



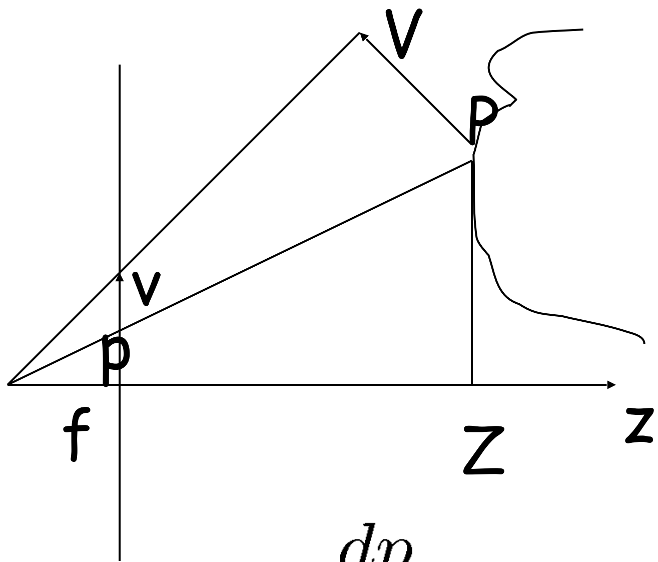
The relative velocity of P wrt camera:

$$V = -t - \omega \times P$$

$$t = \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix} \quad \omega = \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} \quad P = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

$$\begin{aligned} V_x &= -t_x - \omega_y Z + \omega_z Y \\ V_y &= -t_y - \omega_z X + \omega_x Z \\ V_z &= -t_z - \omega_x Y + \omega_y X \end{aligned}$$

Motion Field:



$$p = \frac{fP}{Z}$$

Taking derivative wrt time:

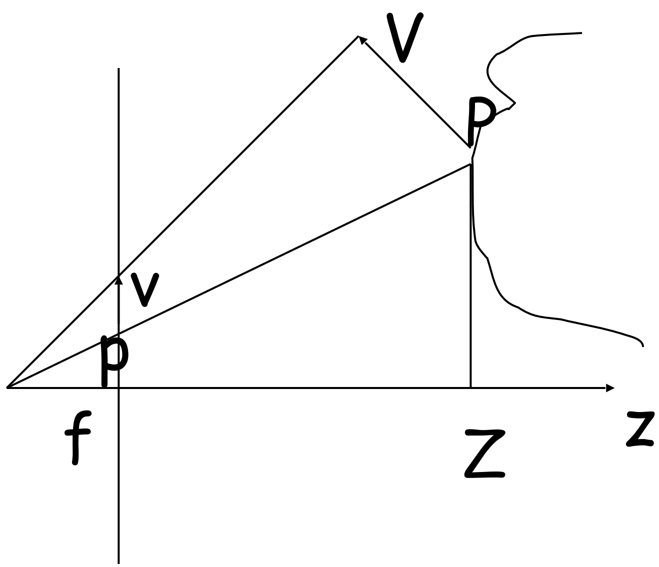
$$\frac{dp}{dt} = \mathbf{v} = \frac{d\frac{fP}{Z}}{dt} \qquad = \frac{f}{Z^2}\left[\frac{dP}{dt}.Z - P.\frac{dZ}{dt}\right]$$

$$= \frac{f}{Z^2}[V.Z - P.V_z]$$

the velocity of p

$$\boxed{\mathbf{v} = f\frac{V}{Z} - p\frac{V_z}{Z}}$$

Motion Field:



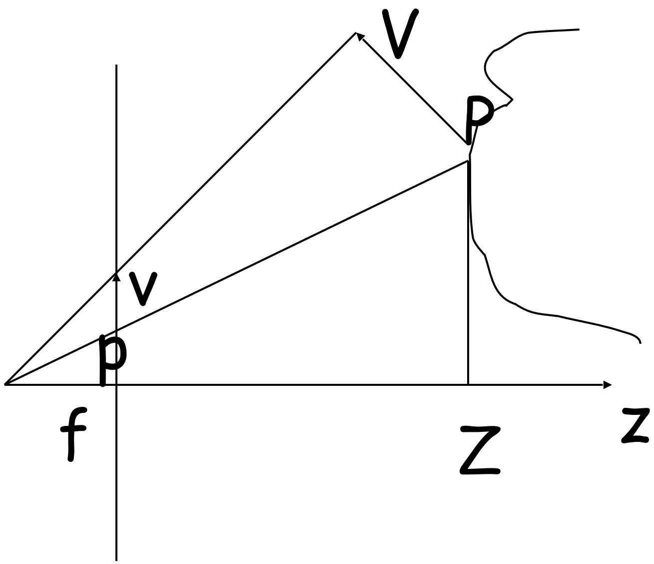
$$\boxed{v = f\frac{V}{Z} - p\frac{V_z}{Z}}$$

$$v_x = u \quad = \quad f\frac{V_x}{Z} - x\frac{V_z}{Z}$$

$$v_y = v \quad = \quad f\frac{V_y}{Z} - y\frac{V_z}{Z}$$

$$v_z \quad = \quad f\frac{V_z}{Z} - f\frac{V_z}{Z} = 0$$

Motion Field:

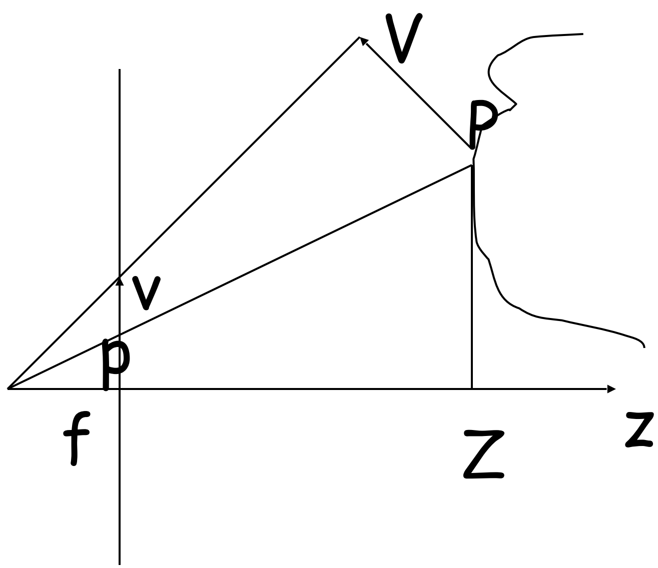


$$
\begin{aligned}
u &= f\frac{V_x}{Z} - x\frac{V_z}{Z} \\
v &= f\frac{V_y}{Z} - y\frac{V_z}{Z}
\end{aligned}
$$

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

$$
\begin{aligned}
u &= \frac{t_z x - t_x f}{Z} + \frac{\omega_x x y}{f} - \omega_y\left(f + \frac{x^2}{f}\right) + \omega_z y \\
v &= \frac{t_z y - t_y f}{Z} + \omega_x\left(f + \frac{y^2}{f}\right) - \frac{\omega_y x y}{f} - \omega_z x
\end{aligned}
$$

Motion Field:



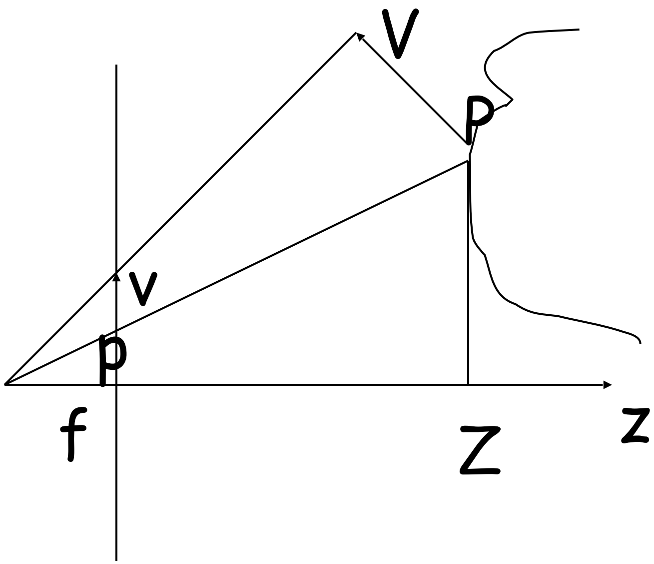
Translational component

Scaling ambiguity
(t and Z can only
be derived up to a scale
Factor)

$$u = \frac{t_z x - t_x f}{Z} + \frac{\omega_x xy}{f} - \omega_y\left(f + \frac{x^2}{f}\right) + \omega_z y$$

$$v = \frac{t_z y - t_y f}{Z} + \omega_x\left(f + \frac{y^2}{f}\right) - \frac{\omega_y xy}{f} - \omega_z x$$

Motion Field:



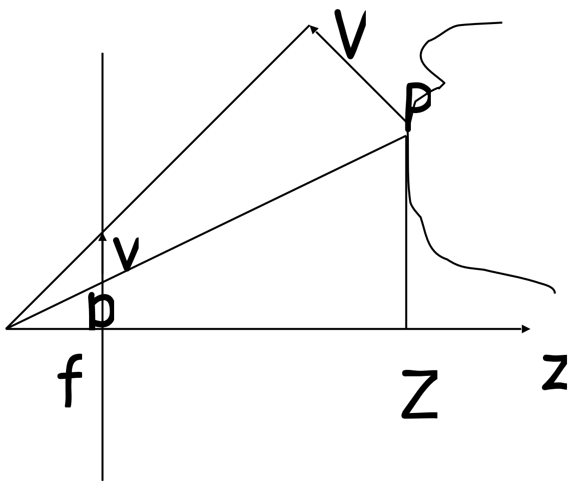
Rotational component

$$u = \frac{t_z x - t_x f}{Z} + \frac{\omega_x x y}{f} - \omega_y\left(f + \frac{x^2}{f}\right) + \omega_z y$$

$$v = \frac{t_z y - t_y f}{Z} + \omega_x\left(f + \frac{y^2}{f}\right) - \frac{\omega_y x y}{f} - \omega_z x$$

NOTE: The rotational component is independent of depth Z !

Motion Field:



$$\mathbf{U}(x,y) = \frac{1}{Z(x,y)}\mathbf{A}(x,y)\mathbf{V} + \mathbf{B}(x,y)\omega$$

$$\mathbf{A}(x,y) = \begin{bmatrix} -f & 0 & x \\ 0 & -f & y \end{bmatrix} \qquad \mathbf{B}(x,y) = \begin{bmatrix} \frac{xy}{f} & -f - \frac{x^2}{f} & y \\ f + \frac{y^2}{f} & -\frac{xy}{f} & x \end{bmatrix}$$

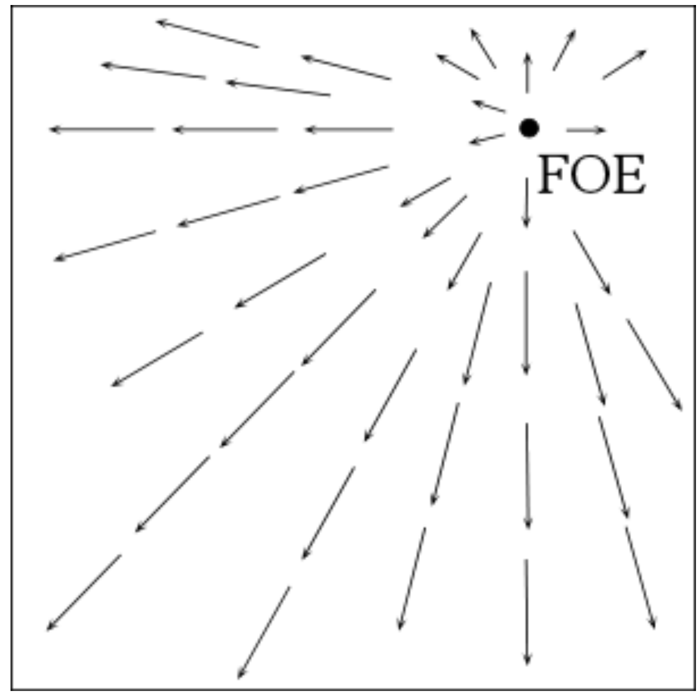
$$V = \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix} \qquad \omega = \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}$$

Translational flow field

$$t_z \neq 0$$

$$u_{tr} = (x - x_o)\frac{t_z}{Z}$$

$$v_{tr} = (y - y_o)\frac{t_z}{Z}$$



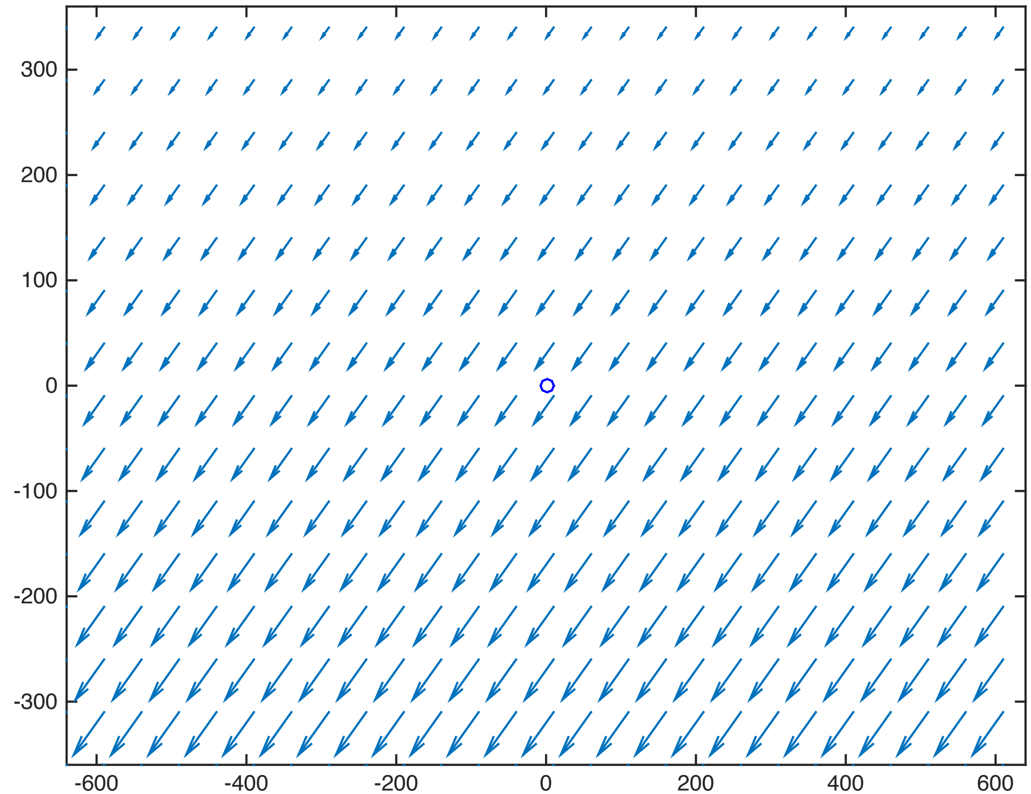
where $p_o = (x_0, y_0) = \left(\frac{t_x}{t_z} \cdot f, \frac{t_y}{t_z} \cdot f\right)$ is the focus of expansion (FOE) or focus of contraction (FOC).

Translational flow field

$$t_z = 0$$

$$u_{tr} = -\frac{t_x \cdot f}{Z}$$

$$v_{tr} = -\frac{t_y \cdot f}{Z}$$



where $p_o = (x_0, y_0) = (\infty, \infty)$ is the focus of expansion (FOE)

or focus of contraction (FOC).

All motion field vectors are parallel to each other and inversely proportional to depth !

Pure Translation: Properties of the MF

- If $t_z \neq 0$ the MF is RADIAL with all vectors pointing towards (or away from) a single point p_o. If $t_z = 0$ the MF is PARALLEL.
- The length of the MF vectors is inversely proportional to depth Z. If $t_z \neq 0$ it is also directly proportional to the distance between p and p_o.

$$
\begin{aligned}
u_{tr} &= (x - x_o)\frac{t_z}{Z} \\
v_{tr} &= (y - y_o)\frac{t_z}{Z}
\end{aligned}
$$

Pure Translation: Properties of the MF

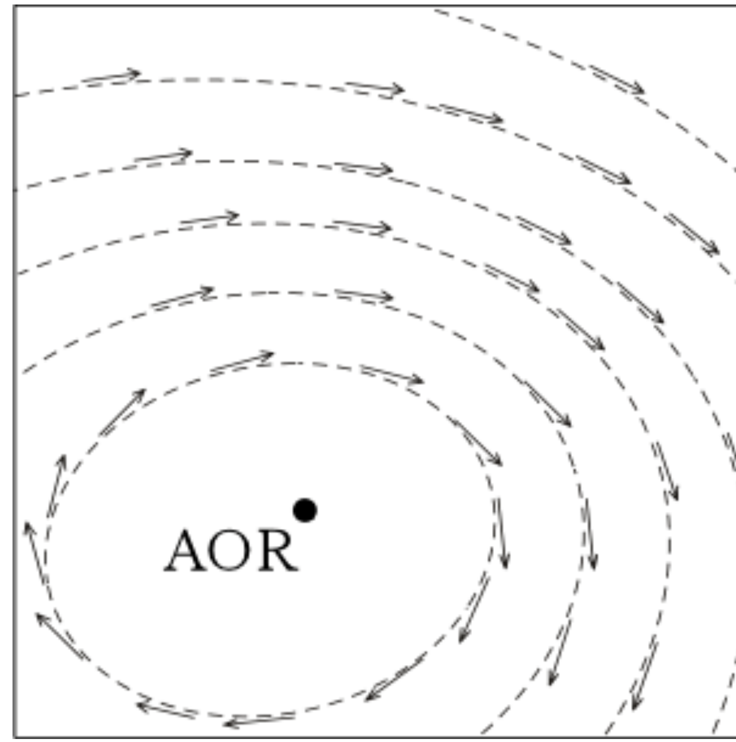
- p_o is the vanishing point of the direction of translation.
- p_o is the intersection of the ray parallel to the translation vector and the image plane.

Rotational flow field

$$u_r = \frac{\omega_x xy}{f} - \omega_y \left(f + \frac{x^2}{f} \right) + \omega_z y$$

$$v_r = -\frac{\omega_y xy}{f} + \omega_x \left(f + \frac{x^2}{f} \right) - \omega_z x$$

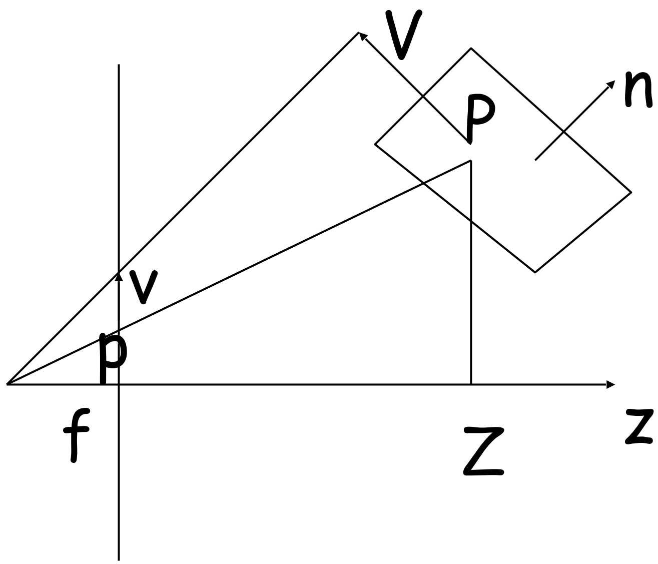
$$\begin{cases} u_r = 0 \\ v_r = 0 \end{cases} \Rightarrow \begin{cases} rx_0 = \dfrac{\omega_x}{\omega_z} f \\ ry_0 = \dfrac{\omega_y}{\omega_z} f \end{cases}$$



$\left(\dfrac{\omega_x}{\omega_z} f, \dfrac{\omega_y}{\omega_z} f \right)$ is the point where the rotation axis pierces the image plane (AOR).

NOTE: The rotational component is independent of depth Z !

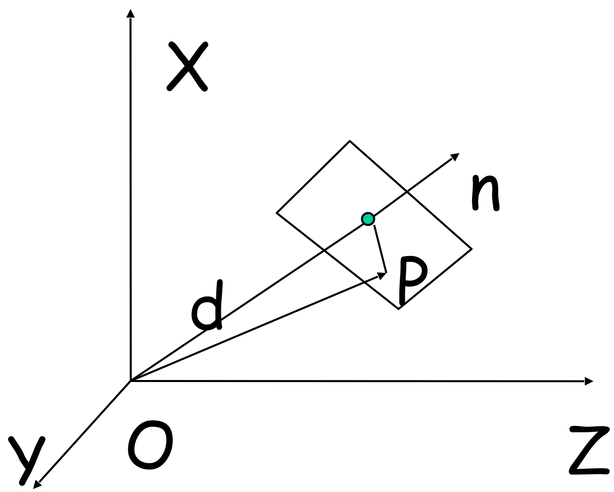
Special Case: Moving Plane



Planar surfaces are common in man-made environments

Question: How does the MF of a moving plane look like?

Special Case: Moving Plane



Points on the plane must satisfy the equation describing the plane.

Let

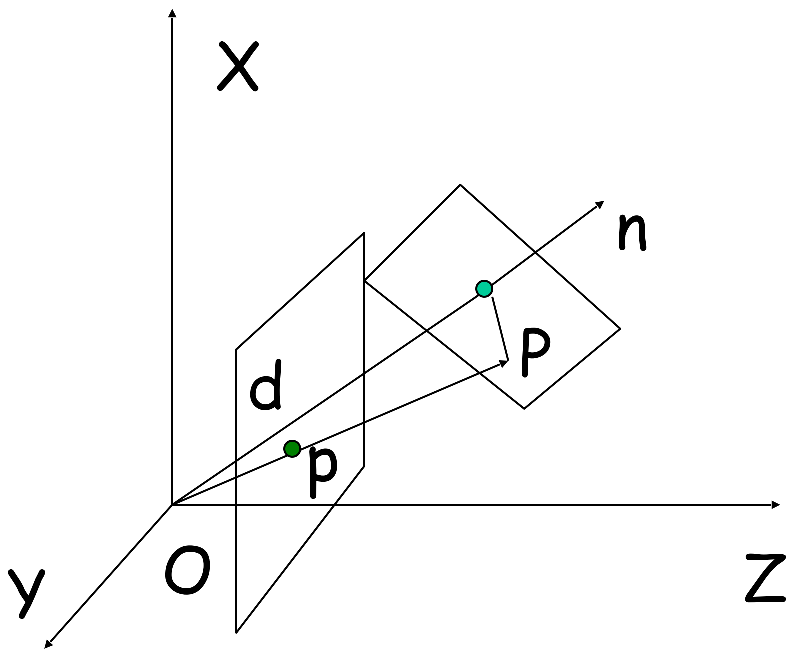
- n be the unit vector normal to the plane.
- d be the distance from the plane to the origin.
- NOTE: If the plane is moving wrt camera, n and d are functions of time.

Then:

$$\boxed{\mathbf{n}^T.\mathbf{P} = d}$$

where $n = \begin{bmatrix} n_x \\ n_y \\ n_z \end{bmatrix}$ $P = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$

Special Case: Moving Plane



Let $p = \begin{bmatrix} x \\ y \\ f \end{bmatrix}$ be the image of P

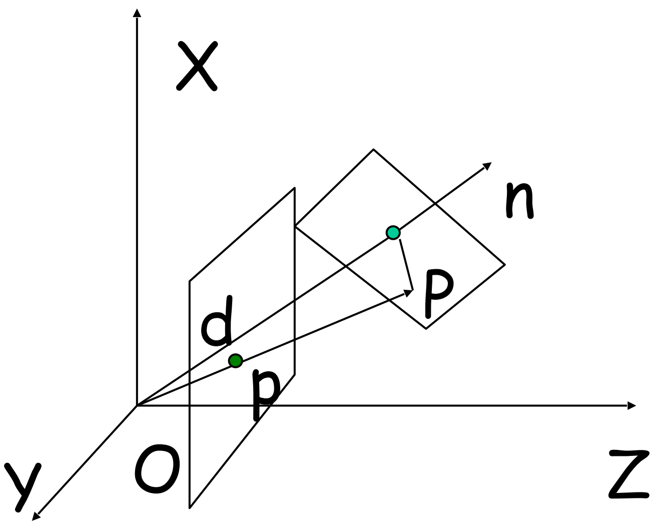
Using the pinhole projection equation:

$$p = \frac{fP}{Z} \Rightarrow P = \frac{pZ}{f}$$

Using the plane equation: $\mathbf{n}^T.\mathbf{P} = d \Rightarrow \mathbf{n}^T.\mathbf{p}\frac{Z}{f} = d$

Solving for Z: $Z = \dfrac{fd}{n_x x + n_y y + n_z f}$

Special Case: Moving Plane



Now consider the MF equations:

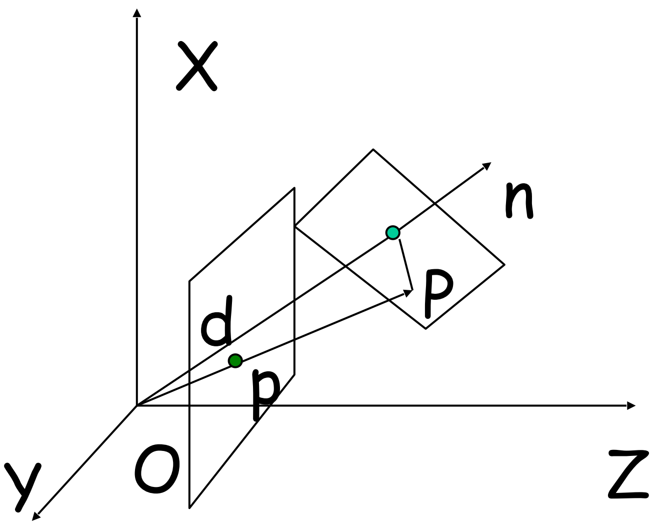
$$u = \frac{t_z x - t_x f}{Z} - \omega_y f + \omega_z y + \frac{\omega_x xy}{f} - \frac{\omega_y x^2}{f}$$

$$v = \frac{t_z y - t_y f}{Z} + \omega_x f - \omega_z x - \frac{\omega_y xy}{f} + \frac{\omega_x y^2}{f}$$

And Plug in:

$$Z = \frac{fd}{n_x x + n_y y + n_z f}$$

Special Case: Moving Plane



The MF equations become:

$$u = \frac{1}{fd}(a_1 x^2 + a_2 xy + a_3 fx + a_4 fy + a_5 f^2)$$

$$v = \frac{1}{fd}(a_1 xy + a_2 y^2 + a_6 fy + a_7 fx + a_8 f^2)$$

where

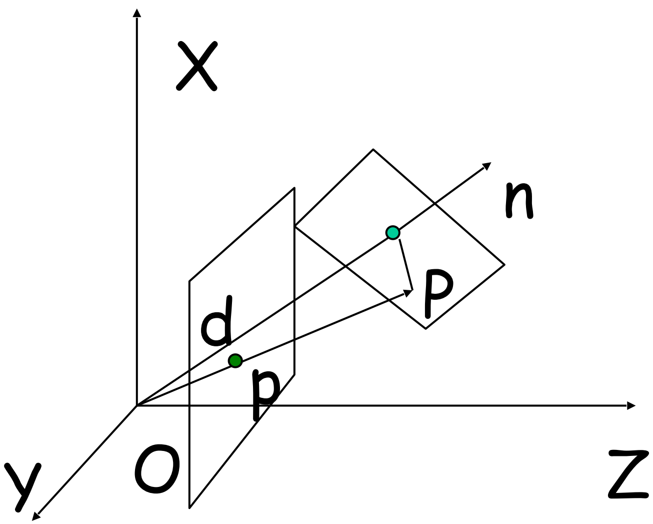
$$a_1 = -d\omega_y + t_z n_x \qquad a_2 = d\omega_x + t_z n_y$$

$$a_3 = t_z n_z - t_x n_x \qquad a_4 = d\omega_z - t_x n_y$$

$$a_5 = -d\omega_y - t_x n_z \qquad a_6 = t_z n_z - t_y n_y$$

$$a_7 = -d\omega_z - t_y n_x \qquad a_8 = d\omega_x - t_y n_z$$

Special Case: Moving Plane



MF equations:

$$u = \frac{1}{fd}(a_1 x^2 + a_2 xy + a_3 fx + a_4 fy + a_5 f^2)$$
$$v = \frac{1}{fd}(a_1 xy + a_2 y^2 + a_6 fy + a_7 fx + a_8 f^2)$$

Q: What is the significance of this?

A: The MF vectors are given by low order (second) polynomials.

- Their coeffs. a_1 to a_8 (only 8 !) are functions of n, d, t and ω.
- The same coeffs. (or MF) can be obtained with a different plane and relative velocity.

8 equations, 9 dof

Moving Plane: Properties of the MF

- The MF of a planar surface is at any time a quadratic function in the image coordinates.

- A plane $n^T P=d$ moving with velocity $V=-t-\omega \times P$ has the same MF than a plane with normal $n'=t/|t|$, distance d and moving with velocity

$$V=|t|n-(\omega+n \times t/d) \times P$$

$$n'=t/|t| \longrightarrow n \times t=0$$

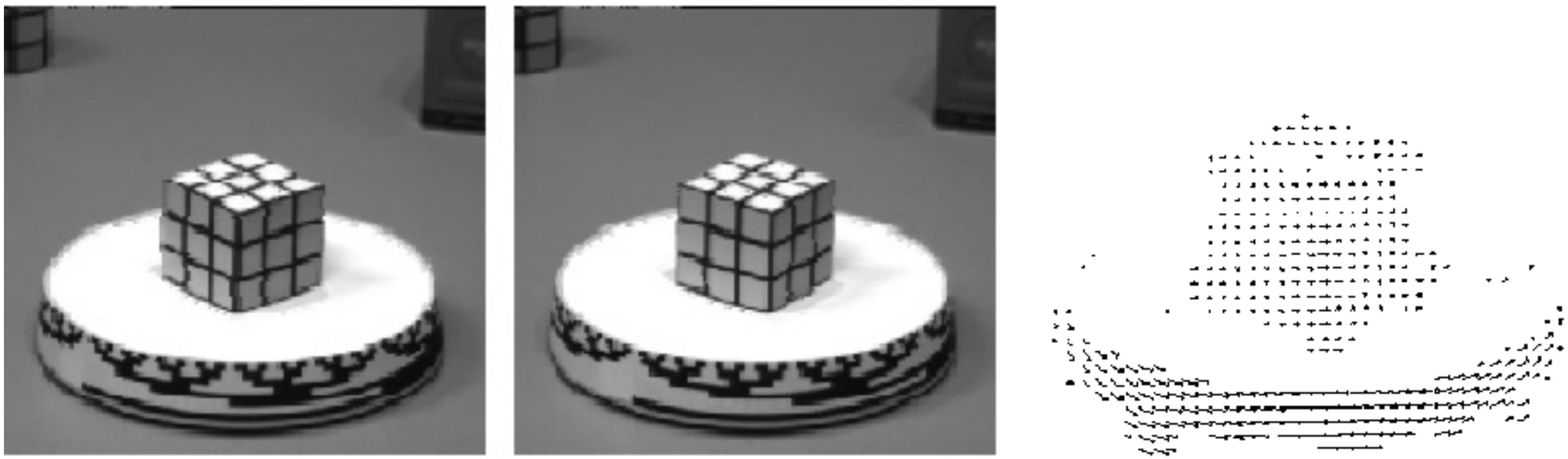
Visual navigation & structure for motion

$$\mathbf{U}(x,y)=\frac{1}{Z(x,y)}\mathbf{A}(x,y)\mathbf{V}+\mathbf{B}(x,y)\omega$$

$$\mathbf{U}(x,y) \Rightarrow Z(x,y);\ \ V;\ \ \omega$$

Motion Field

- The motion field is the projection of the 3D scene motion into the image



Slide credit: Svetlana Lazebnik

Motion Estimation Techniques

- **Direct methods**
  - Directly recover image motion at each pixel from spatio-temporal image brightness variations
  - **Dense motion fields**, but sensitive to appearance variations
  - Suitable for video and when image motion is small
- **Feature-based methods**
  - Extract visual features (corners, textured areas) and track them over multiple frames
  - **Sparse motion fields**, but more robust tracking
  - Suitable when image motion is large

Slide credit: Svetlana Lazebnik