Lecture 22:
Camera Motion

Readings: T&V Sec 8.1 and 8.2
From one time to the next, the camera undergoes rotation (roll, pitch, yaw) and translation (tx, ty, tz).
Motion (Displacement) Fields

Time $t$

Displacement field

Time $t+1$
Motion Field vs Optic Flow

Motion Field: projection of 3D relative velocity vectors onto the 2D image plane.

Optic Flow: observed 2D displacements of brightness patterns in the image.

Motion field is what we want to know. Optic flow is what we can estimate.
Motion Field vs Optic Flow

Sometimes optic flow is a good approximation to the unknown motion flow.

We can then infer relative motion between the camera and objects in the world.
Warning: Optic Flow ≠ Motion Field

Consider a moving light source:

\[ MF = 0 \text{ since the points on the scene are not moving} \]
\[ OF \neq 0 \text{ since there is a moving pattern in the images} \]
Motion Field

We are going to derive an equation relating 3D scene structure and velocity to the 2D motion flow field.
Motion Field

What is a **Field** anyways?

Image a vector at each point in space. This is a vector field. In 3D space, we will look at the field of 3D velocity vectors induced by camera motion.

In 2D, we will be looking at the projections of those 3D vectors in the image. There will be a 2D flow vector at each point in the image. This is the 2D motion flow field.
Recall: General Projection Equation

Strategy:
1) Assume internal params known (set to identity)
2) At time t, set R=I, T=0
3) At time t+1, movement generates a relative R and T
   simplifying assumption: small motion --> small rotation
4) Compute 3D velocity vector
5) Compute 2D velocity vector
   a function of X,Y,Z,R,T,f
Displacement of 3D World Point

Time t: 3D position P
Time t+1: 3D position RP+T

3D Displacement = RP+T - P

Now consider short time period (like time between two video frames = 1/30 sec). Can assume a small rotation angle in that amount of time. Make a small angle approximation and rewrite displacement. In the limit (infinitesimal time period), we will get a velocity.
Write Rotation matrix in terms of Euler Angles

2.2.1 Euler Angle Transformation

Three principal rotations:

\[ \lambda = [0, 0, 1]^T \quad \beta = \gamma \]
\[ \lambda = [0, 1, 0]^T \quad \beta = \theta \]
\[ \lambda = [1, 0, 0]^T \quad \beta = \phi \]

\[ R_{x,\phi} = \begin{bmatrix}
1 & 0 & 0 \\
0 & c\phi & -s\phi \\
0 & s\phi & c\phi
\end{bmatrix} \]

(1) Rotation over yaw angle \( \psi \) about \( z \).
Note that \( u_z = w_z \).

(2) Rotation over pitch angle \( \theta \) about \( y \).
Note that \( v_z = v_r \).

(3) Rotation over roll angle \( \phi \) about \( x \).
Note that \( u_r = u_z \).
2.2.1 Euler Angle Transformation

Linear velocity transformation (zyx-convention):

\[ \dot{p}^n = R^n_b(\Theta)v^b_o \]

where

\[
R^n_b(\Theta) := R_{z_{\psi}}R_{y_{\theta}}R_{x_{\phi}} \quad R^n_b(\Theta)^{-1} = R^n_b(\Theta) = R^T_{x_{\phi}}R^T_{y_{\theta}}R^T_{z_{\psi}}
\]

Example:

\[ v^n_o = R^n_b(\Theta)v^b_o \]
Small Angle Approx

for small angle $x$

$\cos x \approx 1$

$\sin x \approx x$

$\sin x \sin y \approx 0$
2.2.1 Euler Angle Transformation

Linear velocity transformation (zyx-convention):

$$\dot{p}^n = R_b^n(\Theta)v_o^b$$

Example:

$$v_o^b = R_b^n(\Theta)v_o^b$$

where

$$R_b^n(\Theta) := R_{z,\psi} R_{y,\phi} R_{x,\theta}$$

$$R_b^n(\Theta)^{-1} = R_b^n(\Theta) = R_{x,\phi}^T R_{y,\phi}^T R_{z,\psi}^T$$

$$R_b^n(\Theta) = \begin{bmatrix}
    c\psi c\theta & -s\psi c\phi + c\psi s\theta s\phi & s\psi s\phi + c\psi c\phi s\theta \\
    s\psi c\theta & c\psi c\phi + s\phi s\theta s\psi & -c\psi s\phi + s\theta s\psi c\phi \\
    -s\theta & c\theta s\phi & c\theta c\phi
\end{bmatrix}$$

Small angle approximation:

$$R \approx \begin{bmatrix}
    1 & -\psi & \theta \\
    \psi & 1 & -\phi \\
    -\theta & \phi & 1
\end{bmatrix} = I + \begin{bmatrix}
    0 & -\psi & \theta \\
    \psi & 0 & -\phi \\
    -\theta & \phi & 0
\end{bmatrix} = I + S$$
3D Velocity

Under small angle approx, displacement = RP+T - P
= (I+S)P+T-P
= SP+T

Note: SP = [θx, θy, θz]^T \times P

In limit, displacement becomes a velocity

V = T + ω \times P

where ω =

Caution: Abuse of notation ==> we are re-using T as a velocity
The relative velocity of P wrt camera:

\[ V = -T - \omega \times P \]

\[ T = \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix}, \quad \omega = \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}, \quad P = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \]

Note: we change signs of the velocity to be consistent with the book. This should not cause concern. It just depends on whether you want to think of the motion as being due to the camera or the scene.
3D Relative Velocity

The relative velocity of $P$ wrt camera:

\[ V = -T - \omega \times P \]

\[
T = \begin{bmatrix}
T_x \\
T_y \\
T_z
\end{bmatrix}
\]

\[
\omega = \begin{bmatrix}
\omega_x \\
\omega_y \\
\omega_z
\end{bmatrix}
\]

\[
P = \begin{bmatrix}
X \\
Y \\
Z
\end{bmatrix}
\]

\[
V_x = -T_x - \omega_y Z + \omega_z Y
\]

\[
V_y = -T_y - \omega_z X + \omega_x Z
\]

\[
V_z = -T_z - \omega_x Y + \omega_y X
\]
3D Relative Velocity

Where are we in this lecture?

• We just derived an equation relating $R, T$ and to the 3D velocity vector at each scene point.

• We can think of that velocity as a little vector in the scene.

• Now ask, what does the projection of that vector look like in the image? It is a 2D vector. It is one of the vectors that make up the Motion Field!
Motion Field: the 2D velocity of $p$

\[ p = \frac{fP}{Z} \]

Perspective projection

Taking derivative wrt time:

\[ \frac{dp}{dt} = v = \frac{d}{dt} \left( \frac{fP}{Z} \right) \]

\[ \frac{dp}{dt} = v = \frac{f}{Z^2} \left[ \frac{dP}{dt} \cdot Z - P \cdot \frac{dZ}{dt} \right] \]

\[ \frac{dp}{dt} = v = \frac{f}{Z^2} \left[ V \cdot Z - P \cdot V_z \right] \]
Motion Field: the velocity of $p$

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

\[
p = \frac{fP}{Z} \quad P = \frac{pZ}{f}
\]

\[
v = f \frac{V}{Z} - p \frac{V_z}{Z}
\]
Motion Field: the velocity of \( p \)

\[
v = f \frac{V}{Z} - p \frac{V_z}{Z}
\]

\[
v_x = f \frac{V_x}{Z} - x \frac{V_z}{Z}
\]

\[
v_y = f \frac{V_y}{Z} - y \frac{V_z}{Z}
\]

\[
v_z = f \frac{V_z}{Z} - f \frac{V_z}{Z} = 0
\]
Motion Field: the velocity of p

\[ v_x = f \frac{V_x}{Z} - x \frac{V_z}{Z} \]
\[ v_y = f \frac{V_y}{Z} - y \frac{V_z}{Z} \]

\[ V_x = -T_x - \omega_y Z + \omega_z Y \]
\[ V_y = -T_y - \omega_z X + \omega_x Z \]
\[ V_z = -T_z - \omega_x Y + \omega_y X \]

\[ v_x = \frac{T_z x - T x f}{Z} - \omega_y f + \omega_zy + \frac{\omega_{xy} y}{f} - \frac{\omega_{yx} x^2}{f} \]
\[ v_y = \frac{T_z y - T y f}{Z} + \omega_x f - \omega_z x - \frac{\omega_{xy} y}{f} + \frac{\omega_{yx} x^2}{f} \]

THIS IS THE EQUATION WE WANT!!!!
Motion Field: the velocity of \( p \)

\[
\begin{align*}
v_x &= \frac{T_z x - T_x f}{Z} - \frac{\omega_y f + \omega_z y + \frac{\omega_{xy} y}{f} - \frac{\omega_y x^2}{f}}{Z} \\
v_y &= \frac{T_z y - T_y f}{Z} + \frac{\omega_x f - \omega_z x - \frac{\omega_{yx} y}{f} + \frac{\omega_y x^2}{f}}{Z}
\end{align*}
\]

Translational component
Motion Field: the velocity of p

\[ v_x = \frac{T_{zx} - T_{xf}}{Z} - \omega_y f + \omega_{zy} + \frac{\omega_{xx} y}{f} - \frac{\omega_y x^2}{f} \]

\[ v_y = \frac{T_{zy} - T_{yf}}{Z} + \omega_x f - \omega_{zx} - \frac{\omega_{yy} x}{f} + \frac{\omega_{xy} y^2}{f} \]

**NOTE:** The rotational component is independent of depth Z!
Special Case I: Pure Translation

Assume $T_z \neq 0$

Define:

$$\rho_0 = \begin{bmatrix} x_0 \\ y_0 \\ f \end{bmatrix} = \begin{bmatrix} \frac{f T_x}{T_z} \\ \frac{T_y}{T_z} \\ f \end{bmatrix}$$

$$v_x = \frac{T_z x - T_z x_0}{Z} = (x - x_0) \frac{T_z}{Z}$$

$$v_y = \frac{T_z y - T_z y_0}{Z} = (y - y_0) \frac{T_z}{Z}$$
Special Case I: Pure Translation

ω = 0

What if $T_z = 0$?

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

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

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

TIE IN WITH SIMPLE STEREO!
Special Case I: Pure Translation

\[ v_x = (x - x_o) \frac{T_z}{Z} \]
\[ v_y = (y - y_o) \frac{T_z}{Z} \]

The motion field in this case is RADIAL:
- It consists of vectors passing through \( p_o = (x_o, y_o) \)
- If:
  - \( T_z > 0 \), (camera moving towards object)
    - the vectors point away from \( p_o \)
    - \( p_o \) is the POINT OF EXPANSION
  - \( T_z < 0 \), (camera moving away from object)
    - the vectors point towards \( p_o \)
    - \( p_o \) is the POINT OF CONTRACTION
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 $p_o$. 
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.
Special Case II: Moving Plane

Planar surfaces are common in man-made environments

Question: How does the MF of a moving plane look like?
Special Case II: Moving Plane

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

Let

- \( \mathbf{n} \) be the unit vector normal to the plane.
- \( d \) be the distance from the plane to the origin.

**NOTE:** If the plane is moving with respect to the camera, \( \mathbf{n} \) and \( d \) are functions of time.

Then:

\[
\mathbf{n}^T \cdot \mathbf{P} = d
\]

where

\[
\mathbf{n} = \begin{bmatrix} n_x \\ n_y \\ n_z \end{bmatrix}, \quad \mathbf{P} = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}
\]
Special Case II: Moving Plane

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

Using the pinhole projection equation:
\[
p = \frac{fP}{Z} \quad \Rightarrow \quad P = \frac{pZ}{f}
\]

Using the plane equation:
\[
n^T \cdot P = d \quad \Rightarrow \quad n^T \cdot p \frac{Z}{f} = d
\]

Solving for \( Z \):
\[
Z = \frac{fd}{n_xx + n_yy + n_zf}
\]
Special Case II: Moving Plane

Now consider the MF equations:

\[
\begin{align*}
v_x &= \frac{T_{zx} - T_{xf}}{Z} - \omega_y f + \omega_z y + \frac{\omega_{xxy}}{f} - \frac{\omega_{yx}^2}{f} \\
v_y &= \frac{T_{zy} - T_{yf}}{Z} + \omega_x f - \omega_z x - \frac{\omega_{xyy}}{f} + \frac{\omega_{xy}^2}{f}
\end{align*}
\]

And Plug in:

\[
Z = \frac{f d}{n_{xx} + n_{yy} + n_z f}
\]

Camps, PSU
Special Case II: Moving Plane

The MF equations become:

\[
\begin{align*}
v_x &= \frac{1}{fd}(a_1x^2 + a_2xy + a_3fx + a_4fy + a_5f^2) \\
v_y &= \frac{1}{fd}(a_1xy + a_2y^2 + a_6fy + a_7fx + a_8f^2)
\end{align*}
\]

where

\[
\begin{align*}
a_1 &= -d\omega_y + T_zn_x \\
a_3 &= T_zn_z - T_xn_x \\
a_5 &= -d\omega_y - T_xn_z \\
a_7 &= -d\omega_z - T_yn_x \\
a_2 &= d\omega_x + T_zn_y \\
a_4 &= d\omega_z - T_xn_y \\
a_6 &= T_zn_z - T_yn_y \\
a_8 &= d\omega_x - T_yn_z
\end{align*}
\]
Special Case II: Moving Plane

MF equations:

\[ v_x = \frac{1}{fd} (a_1 x^2 + a_2 xy + a_3 fx + a_4 fy + a_5 f^2) \]
\[ v_y = \frac{1}{fd} (a_1 xy + a_2 y^2 + a_6 fy + a_7 fx + a_8 f^2) \]

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 \( \omega \).
- That is, can estimate 8 param global flow rather than 2 params per pixel! Huge savings in time and robustness.

Camps, PSU
Application: 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 into 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:

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

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

Camps, PSU
Time to Collision

\[
l'(t) = fL \left( -\frac{1}{D^2(t)} \right) \frac{d(D(t))}{dt}
\]

\[
D(t) = D_o - vt
\]

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

\[
l'(t) = fL \frac{v}{D^2(t)}
\]
Time to Collision

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

\[ l(t) = \frac{fL}{D(t)} \]

And their ratio is:

\[ \frac{l(t)}{l'(t)} = fL \frac{D^2(t)}{D(t) fLv} = \frac{D(t)}{v} = \tau \]
Time to Collision

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

\[ l(t) = \frac{fL}{D(t)} \]

And time to collision:

\[ \tau = \frac{l(t)}{l'(t)} \]

Can be found, without knowing \( L \) or \( D_o \) or \( v \)!!
Application: Decomposition of Vehicle Flow Field into Rotational and Translational Components

Motivation

• Estimate steering angles
• Computation of scene structure simplifies when rotational motion is removed
Review: Flow Due to Self-Motion

Note: I’ve changed sign again!

\[
\begin{align*}
    u' - u &= f \omega_Y - v \omega_Z + \frac{u^2}{f} \omega_Y - \frac{uv}{f} \omega_X + \frac{TX}{Z} - u \frac{TZ}{Z} \\
    v' - v &= -f \omega_X + u \omega_Z - \frac{v^2}{f} \omega_X + \frac{uv}{f} \omega_Y + \frac{TY}{Z} - v \frac{TZ}{Z}
\end{align*}
\]

Flow:

Rotation component

Translation component
Approach

Use prior knowledge of car’s likely motion to simplify the problem. Assume we know approx direction of translation (Tx, Ty, Tz) 
  [ e.g. if driving forward, we choose (0,0,1) ]
Assume roll angle of motion (wz) is 0
Problem reduces to solving for pitch and yaw angles (wx and wy)

\[
\begin{align*}
  u' - u &= f\omega_Y - v\omega_Z + \frac{u^2}{f}\omega_Y - \frac{uv}{f}\omega_X + fT_x/Z - uT_z/Z \\
  v' - v &= -f\omega_X + u\omega_Z - \frac{v^2}{f}\omega_X + \frac{uv}{f}\omega_Y + fT_y/Z - vT_z/Z
\end{align*}
\]
Approach (continued)

For each observed flow vector, form a linear constraint on \( wx \) and \( wy \) by taking the dot product of the flow equation with a new vector that is constructed to “annihilate” the translation component of flow.

This annihilation vector is

\[
\begin{bmatrix}
  f.T_Y - u.T_Z \\
  -f.T_X + u.T_Z 
\end{bmatrix}
\]

Note: these are all known values!

Verify:

\[
\begin{bmatrix}
  f.T_X - u.T_Z \\
  f.T_Y - v.T_Z \\
  f.T_X - v.T_Z \\
  f.T_Y - u.T_Z \\
  f.T_X - u.T_Z \\
  -f.T_X + u.T_Z \\
  \end{bmatrix}
\begin{bmatrix}
  f.T_Y - u.T_Z \\
  -f.T_X + u.T_Z 
\end{bmatrix}
= 0
\]
For each observed flow vector, we annihilate the translation component to form one linear constraint of the form \((a_i \, wx + b_i \, wy + c_i) = 0\).

Given the whole set of \(n\) observed flow vectors, we seek \(wx\) and \(wy\) that simultaneously satisfy the set of \(n\) linear equations.

Geometric intuition:
Use Robust Estimator

Solving using least-squares will be sensitive to outliers (grossly incorrect data). Instead, we use a robust estimator (Random Sample Consensus – RANSAC).