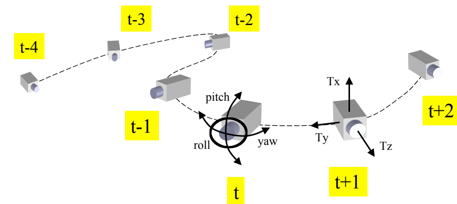
Lecture 22:
Camera Motion

Readings: T&V Sec 8.1 and 8.2

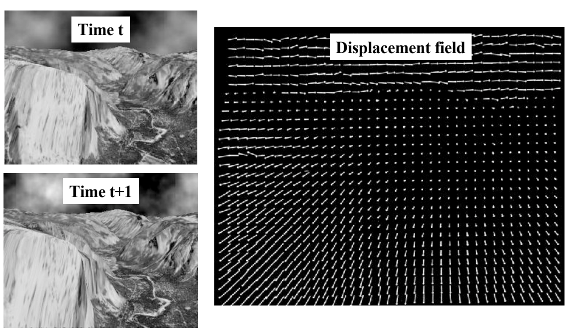
Moving Camera



Camera takes a sequence of images (frames) indexed by time t

From one time to the next, the camera undergoes rotation (roll, pitch, yaw) and translation (tx,ty,tz)

Motion (Displacement) Fields



Motion Field vs Optic Flow

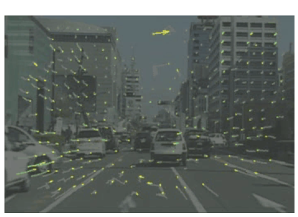
Motion Field: projection of 3D relative velocity vectors onto the 2D image plane.

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.

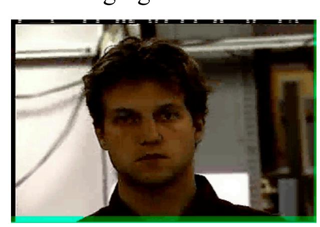


optic flow field

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 since the points on the scene are not moving

OF ≠ 0 since there is a moving pattern in the images

Robert Collins
CSE486, Penn State

Motion Field

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

Robert Collins
CSE486, Penn State

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.

Robert Collins
CSE486, Penn State

Recall : General Projection Equation

$$\begin{array}{ccccccc} & & \text{Internal params} & & \text{Projection} & & \text{Relative R,T} & \\ \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} & \sim & \cancel{\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ 0 & 0 & 1 \end{bmatrix}} & & \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} & & \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} & \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \end{array}$$

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

Robert Collins
CSE486, Penn State

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.

NTNU

Write Rotation matrix in terms of Euler Angles

2.2.1 Euler Angle Transformation

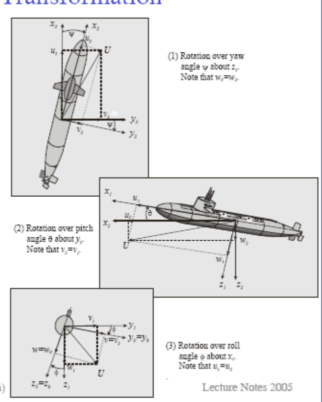
Three principal rotations:

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

$$\mathbf{R}_{z,\psi} = \begin{bmatrix} c\psi & -s\psi & 0 \\ s\psi & c\psi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{R}_{y,\theta} = \begin{bmatrix} c\theta & 0 & s\theta \\ 0 & 1 & 0 \\ -s\theta & 0 & c\theta \end{bmatrix}$$

$$\mathbf{R}_{x,\phi} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\phi & -s\phi \\ 0 & s\phi & c\phi \end{bmatrix}$$



(1) Rotation over yaw angle ψ about z_1. Note that $w_1 = w_2$.

(2) Rotation over pitch angle θ about y_2. Note that $v_2 = v_r$.

(3) Rotation over roll angle ϕ about x_r. Note that $u_r = u_2$

NTNU

2.2.1 Euler Angle Transformation

Linear velocity transformation (*zyx*-convention):

$$\dot{\mathbf{p}}^n = \mathbf{R}_b^n(\Theta)\mathbf{v}_o^b$$

Example:

$$\mathbf{v}_o^n = \mathbf{R}_b^n(\Theta)\mathbf{v}_o^b$$

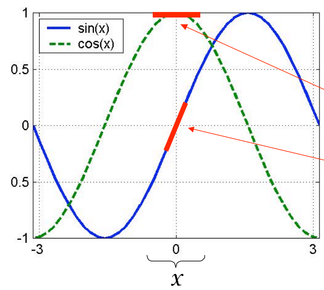
where

$$\mathbf{R}_b^n(\Theta) := \mathbf{R}_{z,\psi}\mathbf{R}_{y,\theta}\mathbf{R}_{x,\phi} \qquad \mathbf{R}_b^n(\Theta)^{-1} = \mathbf{R}_n^b(\Theta) = \mathbf{R}_{x,\phi}^T\mathbf{R}_{y,\theta}^T\mathbf{R}_{z,\psi}^T$$

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

Robert Collins
CSE486, Penn State

Small Angle Approx



for small angle x

$$\cos x \approx 1$$

$$\sin x \approx x$$

$$\sin x \sin y \approx 0$$

NTNU

2.2.1 Euler Angle Transformation

Linear velocity transformation (*zyx*-convention):

$$\dot{\mathbf{p}}^n = \mathbf{R}_b^n(\Theta)\mathbf{v}_o^b$$

Example:

$$\mathbf{v}_o^n = \mathbf{R}_b^n(\Theta)\mathbf{v}_o^b$$

where

$$\mathbf{R}_b^n(\Theta) := \mathbf{R}_{z,\psi}\mathbf{R}_{y,\theta}\mathbf{R}_{x,\phi} \qquad \mathbf{R}_b^n(\Theta)^{-1} = \mathbf{R}_n^b(\Theta) = \mathbf{R}_{x,\phi}^T\mathbf{R}_{y,\theta}^T\mathbf{R}_{z,\psi}^T$$

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

Small angle approximation:

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

10 TTK 4190 Guidance and Control (T. I. Fossen) Lecture Notes 2005

Robert Collins
CSE486, Penn State

3D Velocity

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

$$S = \begin{bmatrix} 0 & -\theta_z & \theta_y \\ \theta_z & 0 & -\theta_x \\ -\theta_y & \theta_x & 0 \end{bmatrix}$$

Note: SP = $[\theta x, \theta y, \theta z]^T$ X P

In limit, displacement becomes a velocity

V = T + ω X P where $\omega = \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}$

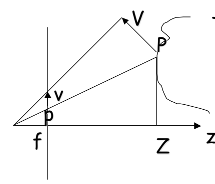
3D velocity
linear velocity
angular velocity

Caution: Abuse of notation ==> we are re-using T as a velocity

---

Robert Collins
CSE486, Penn State

3D Relative Velocity



The relative velocity of P wrt camera:

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

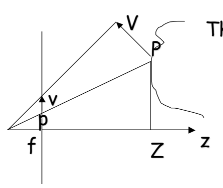
$$T = \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix} \quad \omega = \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} \quad P = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

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.

Camps, PSU

Robert Collins
CSE486, Penn State

3D Relative Velocity



The relative velocity of P wrt camera:

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

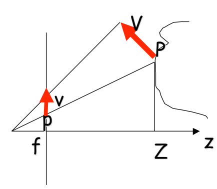
$$T = \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix} \quad \omega = \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} \quad P = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

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

Camps, PSU

Robert Collins
CSE486, Penn State

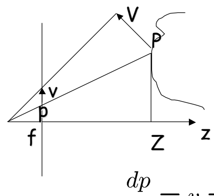
3D Relative Velocity



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**

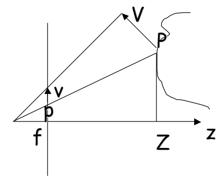
Taking derivative wrt time:

$$\frac{dp}{dt} = v = \frac{d\frac{fP}{Z}}{dt}$$

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

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

Motion Field: the velocity of p

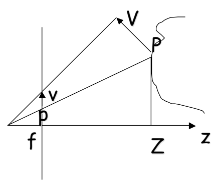


$$\frac{dp}{dt} = v = \frac{f}{Z^2}\left[V.Z - P.V_z\right]$$

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

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

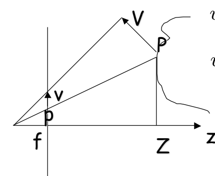
Motion Field: the velocity of p



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

$$\begin{aligned} v_x &= f\frac{V_x}{Z} - x\frac{V_z}{Z} \\ v_y &= f\frac{V_y}{Z} - y\frac{V_z}{Z} \\ v_z &= f\frac{V_z}{Z} - f\frac{V_z}{Z} = 0 \end{aligned}$$

Motion Field: the velocity of p



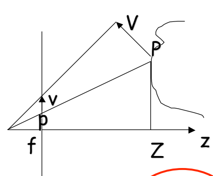
$$\begin{aligned} v_x &= f\frac{V_x}{Z} - x\frac{V_z}{Z} \\ v_y &= f\frac{V_y}{Z} - y\frac{V_z}{Z} \end{aligned}$$

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

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

THIS IS THE EQUATION WE WANT!!!!

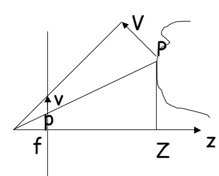
Motion Field: the velocity of p



Translational component

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

Motion Field: the velocity of p



Rotational component

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

NOTE: The rotational component is independent of depth Z !

Special Case I: Pure Translation

$$\omega = 0 \Rightarrow \quad v_x = \frac{T_z x - T_x f}{Z} \qquad v_y = \frac{T_z y - T_y f}{Z}$$

Assume $T_z \neq 0$

Define: $$p_o = \begin{bmatrix} x_o \\ y_o \\ f \end{bmatrix} = \begin{bmatrix} \frac{fT_x}{T_z} \\ \frac{fT_y}{T_z} \\ f \end{bmatrix}$$

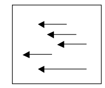
$$\Rightarrow \quad v_x = \frac{T_z x - T_z x_o}{Z} = (x - x_o)\frac{T_z}{Z} \qquad v_y = \frac{T_z y - T_z y_o}{Z} = (y - y_o)\frac{T_z}{Z}$$

Special Case I: Pure Translation

$$\omega = 0 \Rightarrow \quad v_x = \frac{T_z x - T_x f}{Z} \qquad v_y = \frac{T_z y - T_y f}{Z}$$

What if $T_z = 0$?

$$\Rightarrow \quad v_x = -f\frac{T_x}{Z} \qquad v_y = -f\frac{T_y}{Z}$$

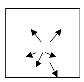


All motion field vectors are parallel to each other and inversely proportional to depth !

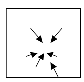
$\boxed{\text{TIE IN WITH SIMPLE STEREO!}}$

Special Case I: Pure Translation

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



$T_z > 0$



$T_z < 0$

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

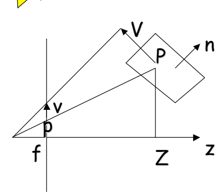
- If $T_z \neq 0$ the MF is RADIAL with all vectors pointing towards (or away from) a single point p_o. If $T_z = 0$ the MF is PARALLEL.
- The length of the MF vectors is inversely proportional to depth Z. If $T_z \neq 0$ it is also directly proportional to the distance between p and p_o.

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.

Self-study

Special Case II: Moving Plane

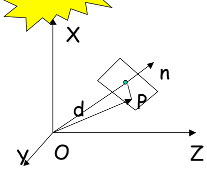


Planar surfaces are common in man-made environments

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

Special Case II: Moving Plane

Self-study



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

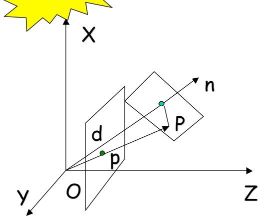
Let
- n be the unit vector normal to the plane.
- d be the distance from the plane to the origin.
- NOTE: If the plane is moving wrt camera, n and d are functions of time.

Then: $$\boxed{\mathrm{n}^T.\mathrm{P} = d}$$

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

Special Case II: Moving Plane

Self-study



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

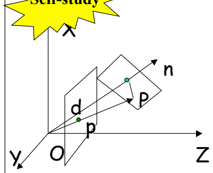
Using the pinhole projection equation:
$$p = \frac{fP}{Z} \Rightarrow P = \frac{pZ}{f}$$

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

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

Special Case II: Moving Plane

Self-study

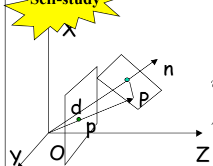


Now consider the MF equations:
$$v_x = \frac{T_z x - T_x f}{Z} - \omega_y f + \omega_z y + \frac{\omega_x xy}{f} - \frac{\omega_y x^2}{f}$$
$$v_y = \frac{T_z y - T_y f}{Z} + \omega_x f - \omega_z x - \frac{\omega_y xy}{f} + \frac{\omega_x y^2}{f}$$

And Plug in:
$$Z = \frac{fd}{n_x x + n_y y + n_z f}$$

Special Case II: Moving Plane

Self-study

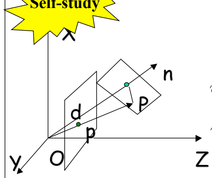


The MF equations become:
$$v_x = \frac{1}{fd}(a_1 x^2 + a_2 xy + a_3 fx + a_4 fy + a_5 f^2)$$
$$v_y = \frac{1}{fd}(a_1 xy + a_2 y^2 + a_6 fy + a_7 fx + a_8 f^2)$$

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

Special Case II: Moving Plane

Self-study



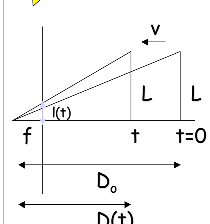
MF equations:
$$v_x = \frac{1}{fd}(a_1 x^2 + a_2 xy + a_3 fx + a_4 fy + a_5 f^2)$$
$$v_y = \frac{1}{fd}(a_1 xy + a_2 y^2 + a_6 fy + a_7 fx + a_8 f^2)$$

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

Application: Time to Collision

Self-study



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:
  - $D(\tau) = D_o - v\tau = 0$
  - $\tau = D_o/v$

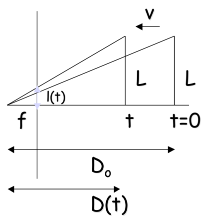
Robert Collins
CSE486, Penn State

Self-study

Time to Collision

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

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



Taking derivative wrt time:

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

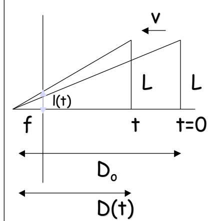
$$l'(t) = fL\frac{-1}{D^2(t)}\frac{d(D(t))}{dt}$$

Camps, PSU

Robert Collins
CSE486, Penn State

Self-study

Time to Collision



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

$$D(t) = D_o - vt$$

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

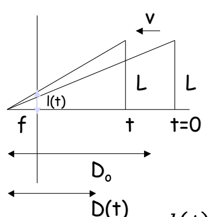
$$l'(t) = fL\frac{v}{D^2(t)}$$

Camps, PSU

Robert Collins
CSE486, Penn State

Self-study

Time to Collision



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

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

And their ratio is:

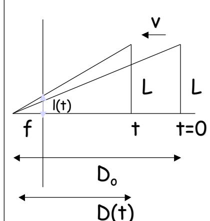
$$\frac{l(t)}{l'(t)} = \frac{fL}{D(t)}\frac{D^2(t)}{fLv} = \frac{D(t)}{v} = \tau$$

Camps, PSU

Robert Collins
CSE486, Penn State

Self-study

Time to Collision



$$\left.\begin{aligned} l'(t) &= fL\frac{v}{D^2(t)} \\ l(t) &= \frac{fL}{D(t)} \end{aligned}\right\}$$ 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 !!

Camps, PSU

Robert Collins
CSE486, Penn State

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

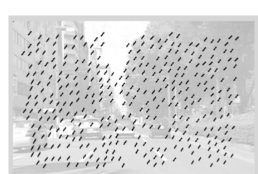
Robert Collins
CSE486, Penn State

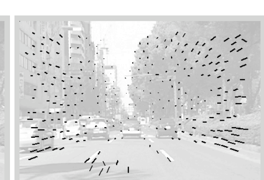
Review: Flow Due to Self-Motion

Note: I've changed sign again!

Flow:

$$\begin{aligned} u' - u &= \underbrace{\boxed{f\omega_Y - v\omega_Z + \frac{u^2}{f}\omega_Y - \frac{uv}{f}\omega_X}}_{\text{Rotation}} + \underbrace{\boxed{f\frac{T_X}{Z} - u\frac{T_Z}{Z}}}_{\text{Translation}} \\ v' - v &= -f\omega_X + u\omega_Z - \frac{v^2}{f}\omega_X + \frac{uv}{f}\omega_Y + f\frac{T_Y}{Z} - v\frac{T_Z}{Z} \end{aligned}$$


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)

known

$$u' - u = f\omega_Y - \underbrace{\cancel{v\omega_Z}}_{0} + \frac{u^2}{f}\omega_Y - \frac{uv}{f}\omega_X + f\frac{\boxed{T_X}}{Z} - u\frac{\boxed{T_Z}}{Z}$$

$$v' - v = -f\omega_X + \cancel{u\omega_Z} - \frac{v^2}{f}\omega_X + \frac{uv}{f}\omega_Y + f\frac{\boxed{T_Y}}{Z} - v\frac{\boxed{T_Z}}{Z}$$

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 $\begin{pmatrix} f\,T_Y - v\,T_Z \\ -f\,T_X + u\,T_Z \end{pmatrix}$ Note: these are all known values!

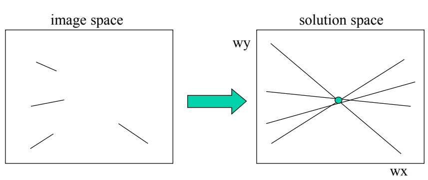
Verify: $\begin{pmatrix} f\frac{T_X}{Z} - u\frac{T_Z}{Z} \\ f\frac{T_Y}{Z} - v\frac{T_Z}{Z} \end{pmatrix} \bullet \begin{pmatrix} f\,T_Y - v\,T_Z \\ -f\,T_X + u\,T_Z \end{pmatrix} = 0$

Approach (continued)

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.

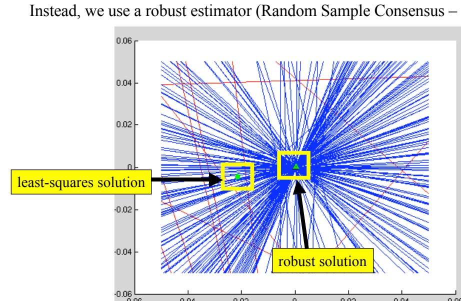
Given the whole set of n observed flow vectors, we seek wx and wy that simultaneously satisfy the set of n linear equations

Geometric intuition:

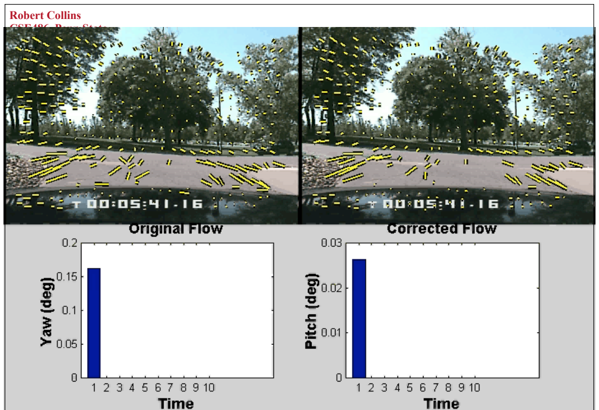


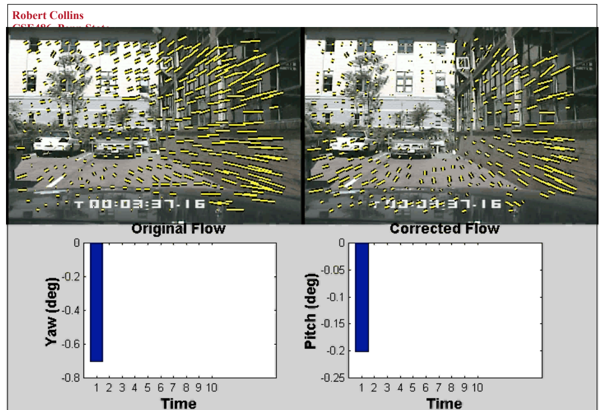
Use Robust Estimator

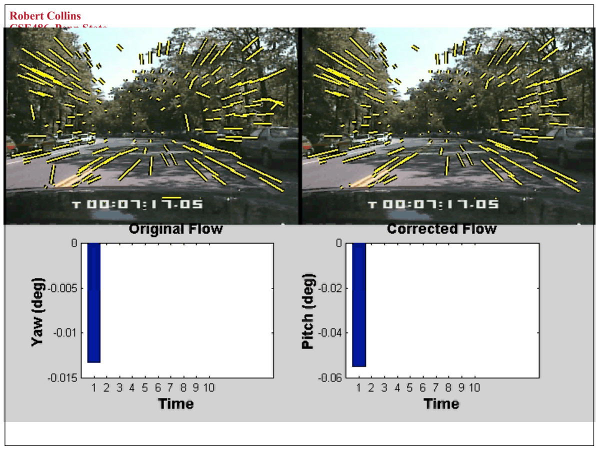
Solving using least-squares will be sensitive to outliers (grossly incorrect data).
Instead, we use a robust estimator (Random Sample Consensus – RANSAC).

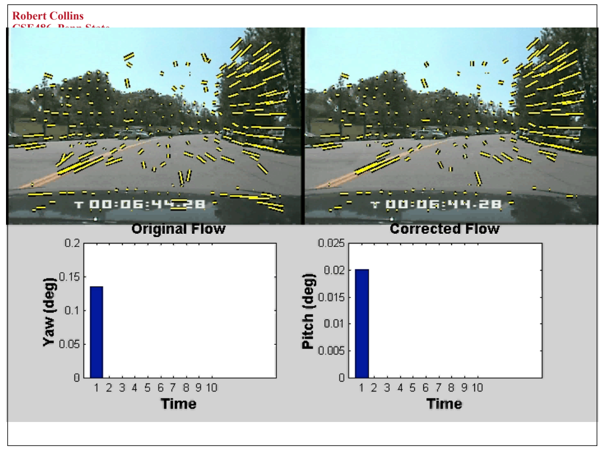


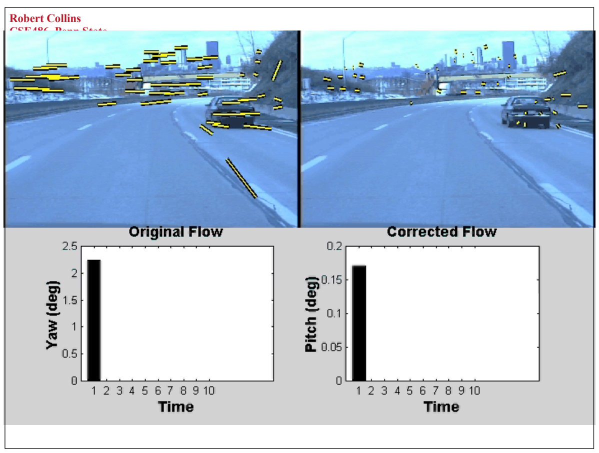
red: outliers
blue: inliers

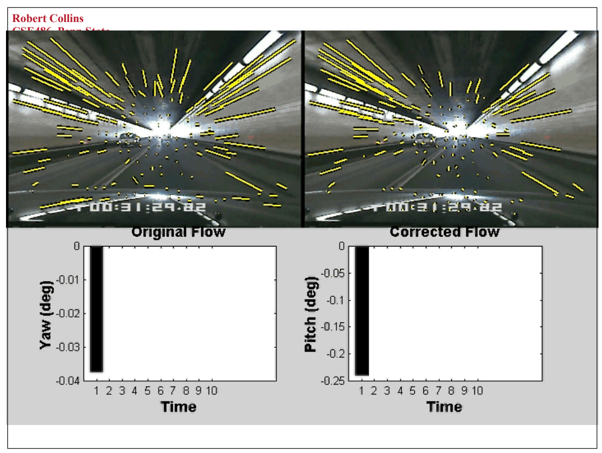


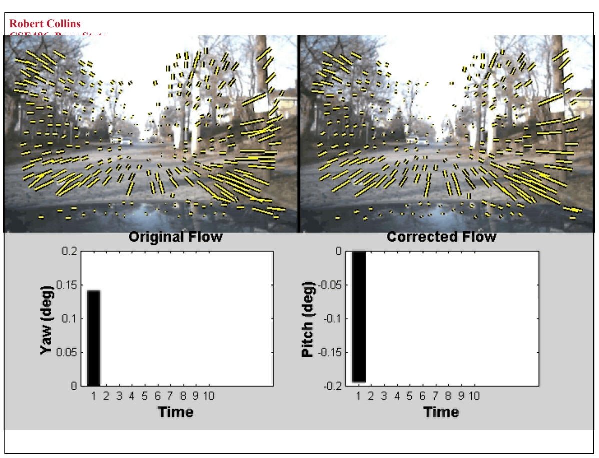


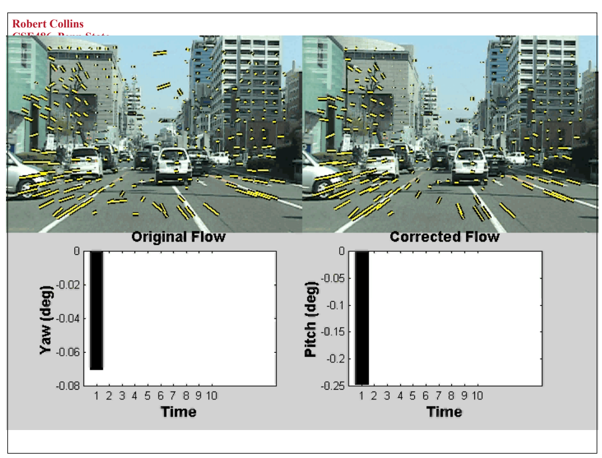

Robert Collins
Original Flow
Corrected Flow
Yaw (deg)
0
-0.005
-0.01
-0.015
Pitch (deg)
0
-0.02
-0.04
-0.06
1 2 3 4 5 6 7 8 9 10
Time


Robert Collins
Original Flow
Corrected Flow
Yaw (deg)
0.2
0.15
0.1
0.05
0
Pitch (deg)
0.025
0.02
0.015
0.01
0.005
0
1 2 3 4 5 6 7 8 9 10
Time


Robert Collins
Original Flow
Corrected Flow
Yaw (deg)
2.5
2
1.5
1
0.5
0
Pitch (deg)
0.2
0.15
0.1
0.05
0
1 2 3 4 5 6 7 8 9 10
Time


Robert Collins
Original Flow
Corrected Flow
Yaw (deg)
0
-0.01
-0.02
-0.03
-0.04
Pitch (deg)
0
-0.05
-0.1
-0.15
-0.2
-0.25
1 2 3 4 5 6 7 8 9 10
Time


Robert Collins
Original Flow
Corrected Flow
Yaw (deg)
0.2
0.15
0.1
0.05
0
Pitch (deg)
0
-0.05
-0.1
-0.15
-0.2
1 2 3 4 5 6 7 8 9 10
Time


Robert Collins
Original Flow
Corrected Flow
Yaw (deg)
0
-0.02
-0.04
-0.06
-0.08
Pitch (deg)
0
-0.05
-0.1
-0.15
-0.2
-0.25
1 2 3 4 5 6 7 8 9 10
Time