Motion Analysis

Motion analysis of digital images is based on a temporal sequence of image frames of a coherent scene.

"sparse sequence"	**=>**	**few frames, temporally spaced apart, considerable differences between frames**
"dense sequence"	**=>**	**many frames, incremental time steps, incremental differences between frames**
video	**=>**	**50 half frames per sec, interleaving, line-by-line sampling**

Motion detection
Register locations in an image sequence which have change due to motion

Moving object detection and tracking
Detect individual moving objects, determine and predict object trajectories, track objects with a moving camera

Derivation of 3D object properties
Determine 3D object shape from multiple views ("shape from motion")

1

Case Distinctions for Motion Analysis

stationary observer
moving observer

single moving object
multiple moving objects

rigid objects
jointed objects
deformable objects

perspective projection
weakly perspective projection
orthographic projection

rotation only
translation only
unrestricted motion

2 image analysis
multiple image analysis

incremental motion
large-scale motion

B/W images
colour images

xray images
IR images
natural images

noisy data
ideal data

monocular images
stereo images

dense flow
sparse flow
no flow
paralaxis

quatitative motion
qualitative motion

small objects
extended objects

polyeder
smooth objects
arbitrary objects

matte surfaces
specular surfaces
textured surfaces
arbitrary surfaces

without occlusion
with occlusion

uncalibrated camera
calibrated camera

data-driven
expectation-driven

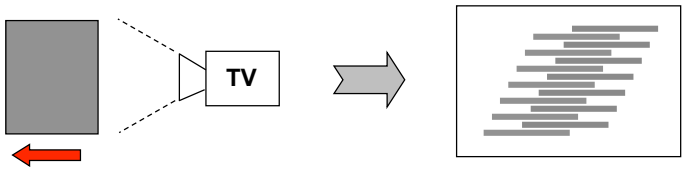
real-time
no real-time

parallel computation
sequential computation

Many motion analysis methods are only applicable in restricted cases!

2

Motion in Video Images



moving object

TV halfframes

TV-rate sampling affects images of moving objects:

- contours show saw-tooth pattern
- deformed angles
- limited resolution

Example:

- 512 pixels per row
- length of dark car is ca. 3.5 m » 130 pixel
- speed is ca. 50 km/h » 14 m/s
- displacement between halfframes is ca. 10 pixels

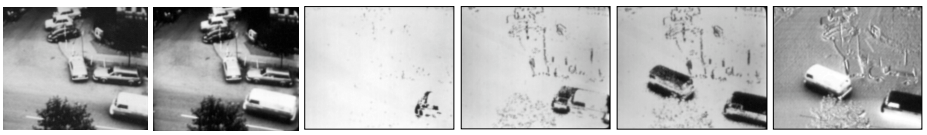
3

Difference Images

An obvious technique for motion detection is based on difference images:

- take the pixelwise difference of images of a sequence
- threshold the magnitude of the differences
- regions above threshold may be due to motion

Examples:



frame1 | frame12 | difference frame2 - frame1 threshold 30 | difference frame12 - frame1 threshold 30 | difference frame34 - frame1 threshold 30 | difference frame34 - frame1 without threshold

Note effects which prohibit reliable motion detection:

- phase jitter between frames (pixels do not correspond exactly)
- spurious motion of branches, pedestrians, dogs, etc.
- motion of uniform brightness regions does not show
- temporal changes of illumination cause non-motion differences

4

Counting Differences

If the goal is to isolate the images of moving objects, it may be useful to

- count how often a pixel differs from its initial value (first-order difference picture FODP)
- count how often a pixel of a FODP region differs from its previous value (second-order difference picture SODP)

(R. Jain 76)

frame1

difference
frame4 - frame1
FODP (yellow)
SODP (red)

difference
frame10 - frame1
FODP (yellow)
SODP (red)

difference
frame30 - frame1
FODP (yellow)
SODP (red)

The problem of uniform brightness regions is not really overcome.

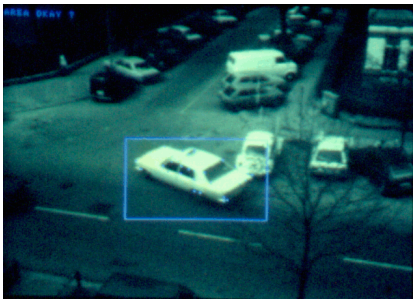
5

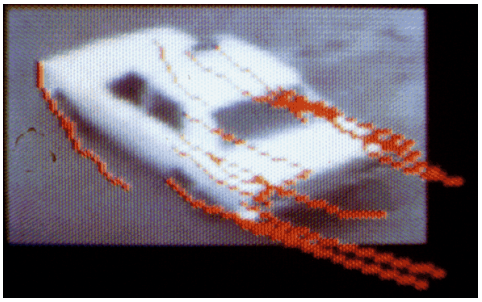
Corresponding Interest Points

Detection of moving objects by

- finding "interest points" in all frames of a sequence
- determining the correspondence of interest points in different frames
- chaining correspondences over time
- grouping interest points into object candidates

Example: Tracking interest points of a taxi turning off Schlüterstraße (Dreschler and Nagel 82)



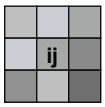


6

Moravec Interest Operator

Interest points (feature points) are image locations where an interest operator computes a high value. Interest operators measure properties of a local pixel neighbourhood.

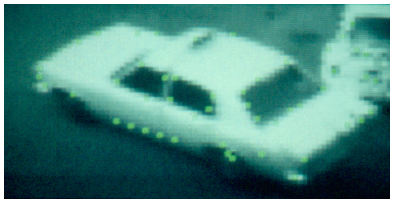
Moravec interest operator: $M(i,j) = \frac{1}{8} \sum_{m=i-1}^{i+1} \sum_{n=j-1}^{j+1} |g(m,n) - g(i,j)|$



This simple operator measures the distinctness of a point w.r.t. its surround.

Refinement of Moravec operator:

Determine locations with strong brightness variations along two orthogonal directions (e.g. based on variances in horizontal, vertical and diagonal direction).



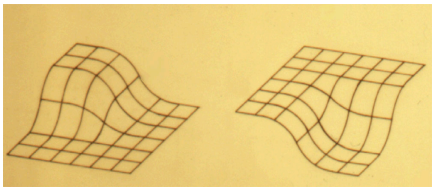
Interest points in different frames may not correspond to identical physical object parts due to their small neighbourhood and noise.

7

Corner Models

Interest points may be based on models of interesting facets of the image function, e.g. corners.

"corner" = location with extremal Gaussian curvatures
(Dreschler and Nagel 81)



Zuniga-Haralick operator:

- fit a cubic polynomial

  $f(i,j) = c_1 + c_2x + c_3y + c_4x^2 + c_5xy + c_6y^2 + c_7x^3 + c_8x^2y + c_9xy^2 + c_{10}y^3$

  For a 5x5 neighbourhood the coefficients of the best-fitting polynomial can be directly determined from the 25 greyvalues

- compute interest value from polynomial coefficients

  $ZH(i,j) = \frac{-2(c_2^2c_6 - c_2c_3c_5 - c_3^2c_4)}{(c_2^2 + c_3^2)^{\frac{3}{2}}}$ measure of "cornerness" of the polynomial

8

Correspondence Problem

The correspondence problem is to determine which interest points in different frames of a sequence mark the same physical part of a scene.

Difficulties:

- **scene may not offer enough structure to uniquely locate points**
- **scene may offer too much structure to uniquely locate points**
- **geometric features may differ strongly between frames**
- **photometric features differ strongly between frames**
- **there may be no corresponding point because of occlusion**

Note that these difficulties apply to single-camera motion analysis as well as multiple-camera 3D analysis (e.g. binocular stereo).

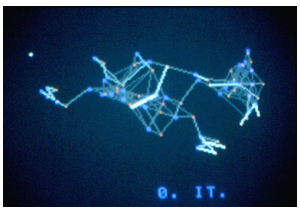
9

Correspondence by Iterative Relaxation

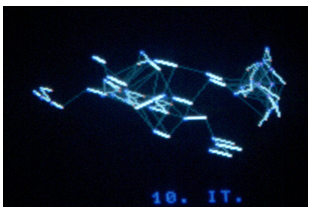
Basic scheme (Thompson and Barnard 81) modified by Dreschler and Nagel:

- **initialize correspondence confidences between all pairs of interest points in 2 frames based on**
  - **similarity of greyvalue neighbourhoods**
  - **plausibility of distance (velocity)**
- **modify confidences iteratively based on**
  - **similarity of displacement vectors in the neighbourhood**
  - **confidence of competing displacement vectors**

interest points of 2 frames (red and blue)



initialized confidences

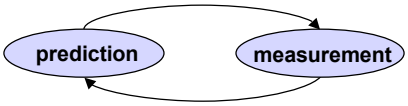


confidences after 10 iterations

10

Kalman Filters (1)

A Kalman filter provides an iterative scheme for (i) predicting an event and (ii) incorporating new measurements.



Assume a linear system with observations depending linearly on the system state, and white Gaussian noise disturbing the system evolution and the observations:

$$\underline{x}_{k+1} = A_k\underline{x}_k + \underline{w}_k$$
$$\underline{z}_k = H_k\underline{x}_k + \underline{v}_k$$

What is the best estimate of $\underline{x}_k$ based on the previous estimate $\underline{x}_{k-1}$ and the observation $\underline{z}_k$?

- $\underline{x}_k$ quantity of interest ("state") at time k
- A_k model for evolution of $\underline{x}_k$
- $\underline{w}_k$ zero mean Gaussian noise with covariance Q_k
- $\underline{z}_k$ observations at time k
- H_k relation of observations to state
- $\underline{v}_k$ zero mean Gaussian noise with covariance R_k

Often, A_k, Q_k, H_k and R_k are constant.

11

Kalman Filters (2)

The best a priori estimate of $\underline{x}_k$ before observing $\underline{z}_k$ is

$$\underline{x}_k' = A_{k-1}\underline{x}_{k-1}$$

After observing $\underline{z}_k$, the a priori estimate is updated by

$$\underline{x}_k'' = \underline{x}_k' + K_k(\underline{z}_k - H_k\underline{x}_k')$$

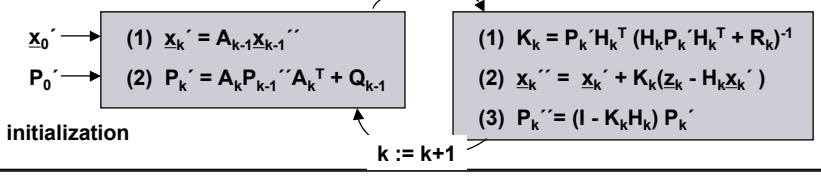
K_k is Kalman gain matrix. K_k is determined to minimize the a posteriori variance P_k'' of the error $\underline{x}_k - \underline{x}_k''$. The minimizing K_k is

$$K_k = P_k'H_k^T (H_kP_k'H_k^T + R_k)^{-1}$$

with $P_k' = A_kP_{k-1}''A_k^T + Q_{k-1}$ and $P_k'' = (I - K_kH_k) P_k'$

P_k' is covariance of error $x_k - \underline{x}_k'$ before observation of $\underline{z}_k$.

Iterative order of computations:



12

Kalman Filter Example

Track positions p_k and velocities v_k of an object moving along a straight line. Assume unknown accelerations a_k with probability density N(0, q^2) and measurements of positions p_k corrupted by white noise b_k with probability density N(0, r^2).

$$\underline{x}_{k+1} = A_k\underline{x}_k + \underline{w}_k \Rightarrow \begin{bmatrix} p_{k+1} \\ v_{k+1} \end{bmatrix} = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} \begin{bmatrix} p_k \\ v_k \end{bmatrix} + \begin{bmatrix} T^2/2 \\ T \end{bmatrix} a_k$$

T is time increment

$$\underline{z}_k = H_k\underline{x}_k + \underline{v}_k \Rightarrow \begin{bmatrix} z_k \\ 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} p_k \\ v_k \end{bmatrix} + \begin{bmatrix} b_k \\ 0 \end{bmatrix}$$

$z_k = p_k + b_k$

$$\underline{x}_0' = \begin{bmatrix} p_0 \\ v_0 \end{bmatrix} \quad P_0' = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$

initialization (here: position and velocity values are known with certainty)

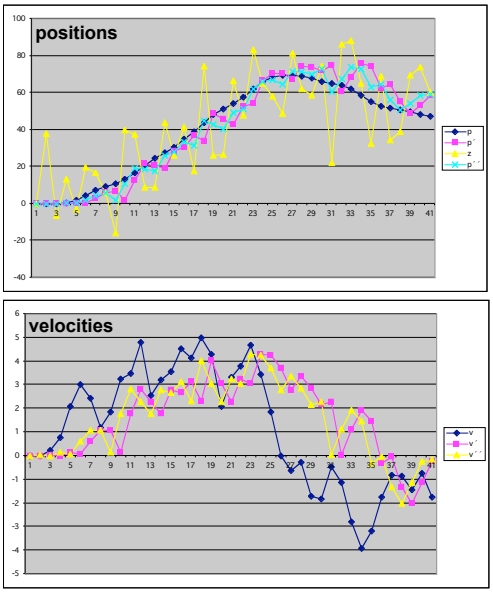
$$K_0 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \quad \underline{x}_0'' = \begin{bmatrix} p_0 \\ v_0 \end{bmatrix} \quad P_0'' = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$

$$\underline{x}_1' = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} \begin{bmatrix} p_0 \\ v_0 \end{bmatrix} = \begin{bmatrix} p_0 + v_0T \\ v_0 \end{bmatrix} \quad P_1' = q^2 \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}$$

$$K_1 = \frac{q^2}{q^2 + r^2} \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \quad \underline{x}_1'' = \begin{bmatrix} p_0 + v_0T \\ v_0 \end{bmatrix} + \frac{q^2}{q^2 + r^2} \begin{bmatrix} z_1 - (p_0 + v_0T) \\ 0 \end{bmatrix} \quad P_1'' = \frac{q^2}{q^2 + 1} \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}$$

13

Diagrams for Kalman Filter Example (1)

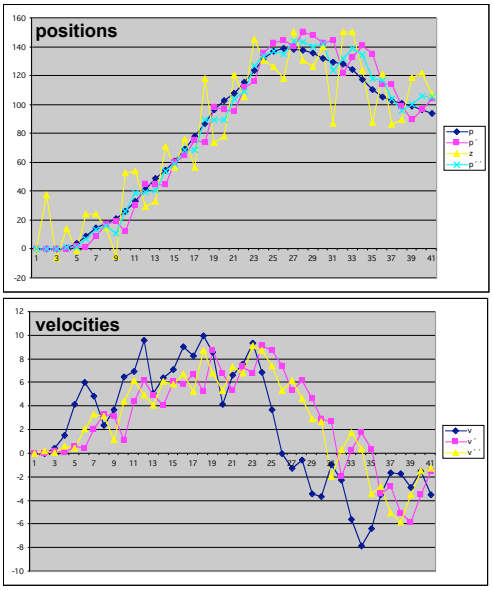


T = 1	time step
q = 1	standard deviation of acceleration bursts
r = 20	standard deviation of position sensor
p_0 = 0	initial position
v_0 = 0	initial velocity

The standard deviation of the estimated position p is around 12 before observing $\underline{z}$ and around 10 after observing $\underline{z}$.

14

Diagrams for Kalman Filter Example (2)



T = 1	time step
q = 2	standard deviation of acceleration bursts
r = 20	standard deviation of position sensor
$p_0 = 0$	initial position
$v_0 = 0$	initial velocity

The standard deviation of the estimated position p is around 15 before observing z and around 12 after observing z.

15

Optical Flow Constraint Equation

Optical flow is the displacement field of surface elements of a scene during an incremental time interval dt ("velocity field").

Assumptions:
- Observed brightness is constant over time (no illumination changes)
- Nearby image points move similarly (velocity smoothness constraint)

For a continuous image g(x, y, t) a linear Taylor series approximation gives

$$g(x+dx, y+dy, t+dt) \approx g(x, y, t) + g_x dx + g_y dy + g_t dt$$

For motion without illumination change we have

$$g(x+dx, y+dy, t+dt) = g(x, y, t)$$

Hence $g_x dx/dt + g_y dy/dt = g_x u + g_y v = -g_t$ u, v velocity components

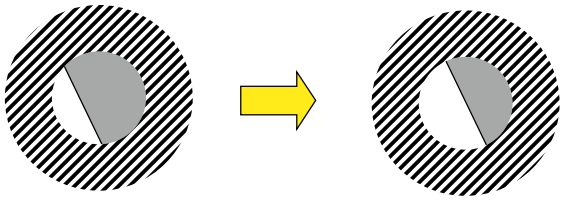
$$g_x u + g_y v = -g_t$$ optical flow constraint equation

$g_x \approx \Delta g/\Delta x$, $g_y \approx \Delta g/\Delta y$, $g_t \approx \Delta g/\Delta t$ may be estimated from the spatial and temporal surround of a location (x, y), hence the optical flow constraint equation provides one equation for the two unknowns u and v.

16

Aperture Effect

The optical flow constraint allows for ambiguous motion interpretations. This can be illustrated by the aperture effect.



In which direction has the edge moved?

Compare with the barber pole effect:

Due to the linear approximation of the image function, the velocity vector cannot be determined uniquely from a local neighbourhood.

17

Optical Flow Smoothness Constraint

For dynamic scenes one can often assume that the velocity field changes smoothly in a spatial neighbourhood:

- large objects
- translational motion
- observer motion, distant objects

Hence, as an additional constraint, one can minimize a smoothness error:

$$e_s = \int\int ((u_x^2 + u_y^2) + (v_x^2 + v_y^2))\ dx\ dy$$

One also wants to minimize the error in the optical flow constraint equation:

$$e_c = \int\int (g_x u + g_y v + g_t)^2\ dx\ dy$$

Using a Lagrange multiplier λ, both constraints can be combined into an error functional, to be minimized by the calculus of variations:

$$e = \int\int (g_x u + g_y v + g_t)^2 + \lambda(u_x^2 + u_y^2 + v_x^2 + v_y^2)\ dx\ dy$$

18

Optical Flow Algorithm

The solution for optical flow with smoothness constraint is given in terms of a pair of partial differential equations:

$$u = \bar{u} - g_x \frac{g_x \bar{u} + g_y \bar{v}}{\lambda^2 + g_x^2 + g_y^2} \qquad v = \bar{v} - g_y \frac{g_x \bar{u} + g_y \bar{v}}{\lambda^2 + g_x^2 + g_y^2}$$

$\bar{u}$ and $\bar{v}$ denote mean velocity values based on the local neighbourhood

The equations can be solved by a Gauss-Seidel iteration based on pairs of consecutive images (Horn & Schunck 81).

Basic optical flow algorithm (Sonka et al. 98, pp. 687):

1. Initialize velocity vectors $\underline{c}(i, j)$ for all (i, j) where $\underline{c}^T = [u\ v]$
2. Estimate g_x, g_y, g_t for all (i, j) from the pair of consecutive images
3. For the k-th iteration, compute

$$u^k(i,j) = \bar{u}^{k-1}(i,j) - g_x(i,j)Q^{k-1}(i,j)$$
$$v^k(i,j) = \bar{v}^{k-1}(i,j) - g_y(i,j)Q^{k-1}(i,j)$$

with $$Q^{k-1}(i,j) = \frac{g_x(i,j)\bar{u}^{k-1}(i,j) + g_y(i,j)\bar{v}^{k-1}(i,j)}{\lambda^2 + g_x^2(i,j) + g_y^2(i,j)}$$

4. Repeat step 3 until the error e is below a threshold

$$e^k = \sum_i \sum_i [g_x(i,j)u^k(i,j) + g_y(i,j)v^k(i,j) + g_t(i,j)]^2 + \lambda[u_x^{k2}(i,j) + u_y^{k2}(i,j) + v_x^{k2}(i,j) + v_y^{k2}(i,j)\ <\ \varepsilon$$

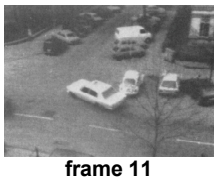
λ is a fixed value chosen to balance the constraints

Optical Flow Improvements

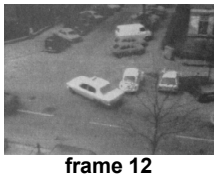
(from Nagel and Enkelmann 86)

Several improvements of the Horn & Schunck optical flow computation have been suggested. For example, Nagel (1983) introduced the "oriented smoothness constraint" which does not enforce smoothness across edges.

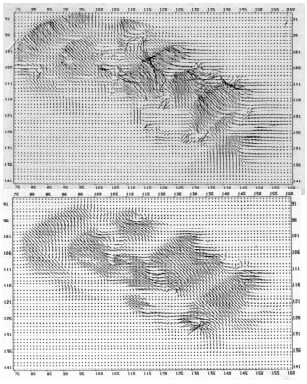
2 frames of the taxi sequence



frame 11



frame 12



needle diagram of optical flow for taxi motion with isotropic smoothness constraint after 30 iterations

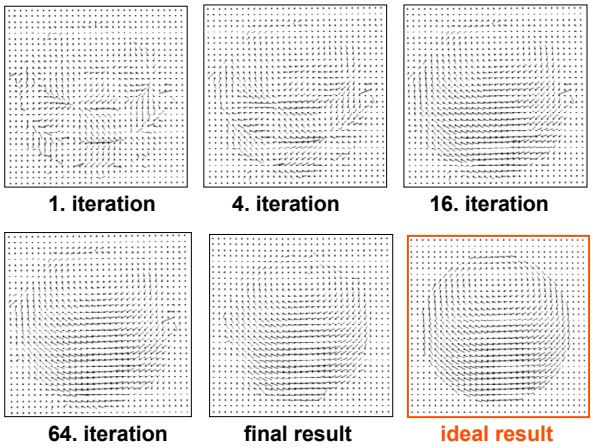
the same with oriented smoothness constraint

Optical Flow and Segmentation

The optical flow smoothness constraint is not valid at occluding boundaries ("silhouettes"). In order to inhibit the constraint, one may try to segment the image based on optical flow discontinuities while performing the iterations.

Checkered sphere rotating before randomly textured background

(From B.K.P. Horn, Robot Vision, 1986)

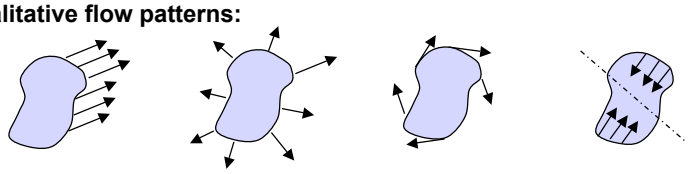


21

Optical Flow Patterns

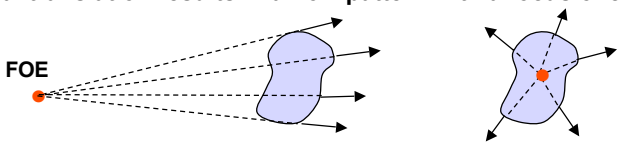
Complex optical flow fields may be segmented into components which show a consistent qualitative pattern.

Qualitative flow patterns:



translation at constant distance | translation in depth | rotation at constant distance | rotation about axis parallel to image plane

General translation results in a flow pattern with a focus of expansion (FOE):



As the direction of motion changes, the FOE changes its location.

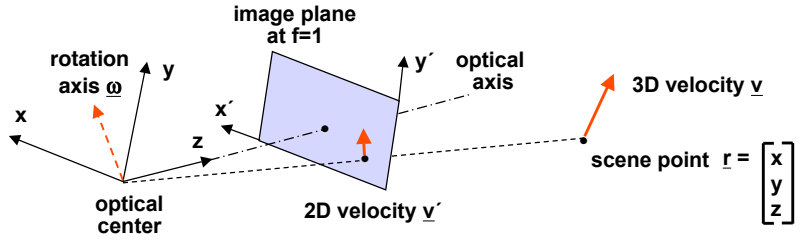
22

Optical Flow and 3D Motion (1)

In general, optical flow may be caused by an unknown 3D motion of an unkown surface.

How do the flow components u´, v´ depend on the 3D motion parameters?

Assume camera motion in a static scene, optical axis = z-axis, rotation about the origin.



3D velocity $\underline{v}$ of a point $\underline{r}$ is determined by rotational velocity $\underline{\omega}$ and translational velocity $\underline{t}$:

$$\underline{v} = -\underline{t} - \underline{\omega} \times \underline{r}$$

23

Optical Flow and 3D Motion (2)

By taking the component form of $\underline{v} = -\underline{t} - \underline{\omega} \times \underline{r}$ with $\underline{t}^T = [t_x, t_y, t_z]$, $\underline{\omega}^T = [a, b, c]$ and $\underline{r}^T = [x\ y\ z]$ and computing the perspective projection we get

$$u' = \frac{\dot{x}}{z} - \frac{x\dot{z}}{z^2} = \left(-\frac{t_x}{z} - b + cy'\right) - x'\left(-\frac{t_z}{z} - ay' + bx'\right)$$

$$v' = \frac{\dot{y}}{z} - \frac{y\dot{z}}{z^2} = \left(-\frac{t_y}{z} - cx' + a\right) - y'\left(-\frac{t_z}{z} - ay' + bx'\right)$$

Observation of u´and v´at location (x´, y´) gives 2 equations for 7 unknowns. Note that motion of a point at distance kz with translation k$\underline{t}$ and the same rotation ω will give the same optical flow, k any scale factor.

The translational and rotational parts may be separated:

$$u'_{\text{translation}} = -\frac{t_x + x't_z}{z} \qquad u'_{\text{rotation}} = ax'y' - b(x'^2 + 1) + cy'$$

$$v'_{\text{translation}} = -\frac{t_y + y't_z}{z} \qquad v'_{\text{rotation}} = a(y'^2 + 1) - bx'y' + cx'$$

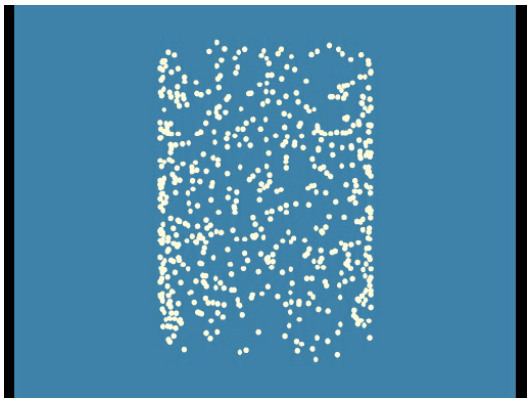
For pure translation we have 2 equations for 3 unknows (z fixed arbitrarily).

24

3D Motion Analysis Based on 2D Point Displacements

2D displacements of points observed on an unknown moving rigid body may provide information about

- the 3D structure of the points
- the 3D motion parameters



Rotating cylinder experiment by S. Ullman (1981)

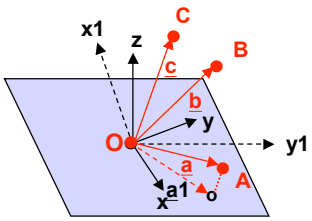
Cases of interest:

- stationary camera, moving object(s)
- moving camera, stationary object(s)
- moving camera, moving object(s)

} camera motion parameters may be known

Structure from Motion (1)

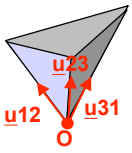
Ullman showed 1979 that the spatial structure of 4 rigidly connected non-coplanar points may be recovered from 3 orthographic projections.



projection plane P_1

O, A, B, C	4 rigid points
$\underline{a}$, $\underline{b}$, $\underline{c}$	vectors to A, B, C
Π_1, Π_2, Π_3	projection planes
xi, yi	coordinate axes of P_i
$\underline{ai}$, $\underline{bi}$, $\underline{ci}$	coordinate pairs of points A, B, C in projection plane Π_i

The problem is to determine the spatial orientations of Π_1, Π_2, Π_3 from the 9 projection coordinate pairs $\underline{ai}$, $\underline{bi}$, $\underline{ci}$, i = 1, 2, 3.



The 3 projection planes intersect and form a tetrahedron. $\underline{u}12$, $\underline{u}23$, $\underline{u}31$ are unit vectors along the intersections. The idea is to determine the $\underline{u}ij$ from the observed coordinates $\underline{ai}$, $\underline{bi}$, $\underline{ci}$.

Structure from Motion (2)

The projection coordinates are

$a1_x = \underline{a}^T\underline{x}1$ $\quad a1_y = \underline{a}^T\underline{y}1$

$b1_x = \underline{b}^T\underline{x}1$ $\quad b1_y = \underline{b}^T\underline{y}1$

$c1_x = \underline{c}^T\underline{x}1$ $\quad c1_y = \underline{c}^T\underline{y}1$

Since each $\underline{u}ij$ lies in both planes Π_i and Π_j, it can be written as

$\underline{u}ij = \alpha_{ij}\underline{x}i + \beta_{ij}\underline{y}i$

$\underline{u}ij = \gamma_{ij}\underline{x}j + \delta_{ij}\underline{y}j$

$\Rightarrow \quad \alpha_{ij}\underline{x}i + \beta_{ij}\underline{y}i = \gamma_{ij}\underline{x}j + \delta_{ij}\underline{y}j$

Multiplying with $\underline{a}^T$, $\underline{b}^T$ and $\underline{c}^T$ we get

$\alpha_{ij}ai_x + \beta_{ij}ai_y = \gamma_{ij}aj_x + \delta_{ij}aj_y$

$\alpha_{ij}bi_x + \beta_{ij}bi_y = \gamma_{ij}bj_x + \delta_{ij}bj_y$

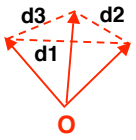
$\alpha_{ij}ci_x + \beta_{ij}ci_y = \gamma_{ij}cj_x + \delta_{ij}cj_y$

Exploiting the constraints $\alpha_{ij}^2 + \beta_{ij}^2 = 1$ and $\gamma_{ij}^2 + \delta_{ij}^2 = 1$, we can solve for α_{ij}, β_{ij}, γ_{ij}, δ_{ij}.

27

Structure from Motion (3)

From the coefficients α_{ij}, β_{ij}, γ_{ij}, δ_{ij} one can compute the distances between the 3 unit vectors $\underline{u}12$, $\underline{u}23$, $\underline{u}31$:



$d1 = \| \underline{u}23 - \underline{u}12 \| = \| (\alpha_{23} - \alpha_{12})\underline{x}i + (\beta_{23} - \beta_{12})\underline{y}i \| = (\alpha_{23} - \alpha_{12})^2 + (\beta_{23} - \beta_{12})^2$

$d2 = (\alpha_{31} - \alpha_{23})^2 + (\beta_{31} - \beta_{23})^2$

$d3 = (\alpha_{12} - \alpha_{31})^2 + (\beta_{12} - \beta_{31})^2$

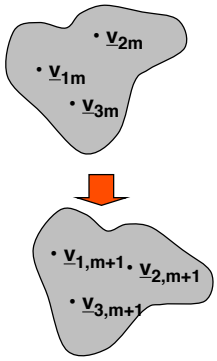
Hence the relative angles of the projection planes are determined.

The spatial positions of A, B, C relative to the projection planes (and to the origin O) can be determined by intersecting the projection rays perpendicular on the projected points $\underline{a}i$, $\underline{b}i$, $\underline{c}i$.

28

Perspective 3D Analysis of Point Displacements

- relative motion of one rigid object and one camera
- observation of P points in M views



For each point $\underline{v}_p$ in 2 consecutive images we have:

$\underline{v}_{p,m+1} = R_m \underline{v}_{pm} + \underline{t}_m$ motion equation

$\underline{v}_{pm} = \lambda_{pm} \underline{v}_{pm}'$ projection equation

For P points in M images we have

- 3MP unknown 3D point coordinates $\underline{v}_{pm}$
- 6(M-1) unkown motion parameters R_m and $\underline{t}_m$
- MP unknown projection parameters λ_{pm}
- 3(M-1)P motion equations
- 3MP projection equations
- 1 arbitrary scaling parameter

equations ≥ # unknowns => $P \geq 3 + \frac{2}{2M-3}$ =>

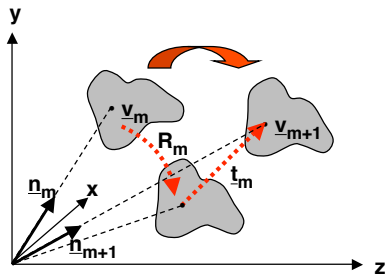
M	P
2	5
3	4
4	4
5	4

29

Essential Matrix

Geometrical constraints derived from 2 views of a point in motion

- motion between image m and m+1 may be decomposed into
  1) rotation R_m about origin of coordinate system (= optical center)
  2) translation $\underline{t}_m$
- observations are given by direction vectors $\underline{n}_m$ and $\underline{n}_{m+1}$ along projection rays



$R_m\underline{n}_m$, $\underline{t}_m$ and $\underline{n}_{m+1}$ are coplanar: $[\underline{t}_m \times R_m\underline{n}_m]^T \underline{n}_{m+1} = 0$

After some manipulation: $\underline{n}_m^T E_m \underline{n}_{m+1} = 0$ E = essential matrix

with $E_m = \begin{bmatrix} | & | & | \\ \underline{t}_m \times \underline{r}_1 & \underline{t}_m \times \underline{r}_2 & \underline{t}_m \times \underline{r}_3 \\ | & | & | \end{bmatrix}$ and $R_m = \begin{bmatrix} | & | & | \\ \underline{r}_1 & \underline{r}_2 & \underline{r}_3 \\ | & | & | \end{bmatrix}$

30

Solving for the Essential Matrix

$\underline{n}_m^T E_m \underline{n}_{m+1} = 0$ formally one equation for 9 unknowns e_{ij}

But: only 6 degrees of freedom
(3 rotation angles, 3 translation components)

e_{ij} can only be determined up to a scale factor

Basic solution approach:

- observe P points, alltogether in 2 views, P >> 8
- fix e_{11} arbitrarily
- solve an overconstrained system of equations for the other 8 unknown coefficients e_{ij}

E may be decomposed into S and R by Singular Value Decomposition (SVD).

E may be written as $E = S R^{-1}$ with R = rotation matrix and $S = \begin{bmatrix} 0 & -t_z & t_y \\ t_z & 0 & -t_x \\ -t_y & t_x & 0 \end{bmatrix}$

Note: S (and therefore E) has rank 2

31

Singular Value Decomposition of E

Any m x n matrix A, m ≥ n, may be decomposed as $A = U D V^T$ where

U	has orthonormal columns	m x n
D	is non-negative diagonal	n x n
V^T	has orthonormal rows	n x n

This can be applied to E to give $E = U D V^T$ with

$R = U G V^T$ or $R = U G^T V^T$

$S = V Z V^T$

where $G = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ and $Z = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$

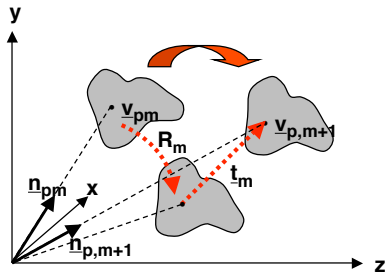
32

Nagel-Neumann Constraint

Consider 2 views of 3 points $\underline{v}_{pm}$, p = 1 ... 3, m = 1, 2

The planes through $R_m\underline{n}_{pm}$ and $\underline{n}_{p,m+1}$ all intersect in $\underline{t}_m$

=> the normals of the planes are coplanar



Coplanarity condition for 3 vectors $\underline{a}$, $\underline{b}$, $\underline{c}$: $(\underline{a} \times \underline{b})^T \underline{c} = 0$

$$([R_m\underline{n}_{1m} \times \underline{n}_{1,m+1}] \times [R_m\underline{n}_{2m} \times \underline{n}_{2,m+1}])^T [R_m\underline{n}_{3m} \times \underline{n}_{3,m+1}] = 0$$

Nonlinear equation with 3 unknown rotation parameters.
=> Observation of at least 5 points required to solve for the unknowns.

33

Reminder: Homogeneous Coordinates

- (N+1)-dimensional notation for points in N-dimensional Euclidean space
- allows to express projection and translation as linear operations

Normal coordinates: $\underline{v}^T = [x\ y\ z]$

Homogeneous coordinates: $\underline{v}^T = [wx\ wy\ wz\ w]$
$w \neq 0$ is arbitrary constant

Rotation and translation in homogeneous coordinates:

$\underline{v}' = A\underline{v}$ with $A = \begin{bmatrix} R & \underline{t} \\ \underline{0} & 1 \end{bmatrix}$

Projection in homogeneous coordinates:

$\underline{v}' = B\underline{v}$ with $B = \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix}$

Divide the first N components by the (N+1)rst component to recover normal coordinates

34

From Homogeneous World Coordinates to Homogeneous Image Coordinates

x, y, z = $\underline{v}$ = scene coordinates

x_p'', y_p'' = $\underline{v}_P$ = image coordinates

$$\begin{bmatrix} wx_p'' \\ wy_p'' \\ w \end{bmatrix} = \begin{bmatrix} K\,R & K\,\underline{t} \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = M \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \quad \Rightarrow \quad \underline{v}_p = M\,\underline{v}$$

$$K = \begin{bmatrix} fa & fb & x_{p0} \\ 0 & fc & y_{p0} \\ 0 & 0 & 1 \end{bmatrix}$$ intrinsic camera parameters ("camera calibration matrix K")

R, $\underline{t}$ extrinsic camera parameters

M = 3 x 4 projective matrix

fa = scaling in x_P-axis
fc = scaling in y_P-axis
fb = slant of axes
x_{P0}, y_{P0} = "principal point" (optical center in image plane)

35

Camera Calibration

Determine intrinsic and/or extrinsic camera parameters for a specific camera-scene configuration. Prior calibration may be needed

- to measure unknown objects
- to navigate as a moving observer
- to perform stereo analysis
- to compensate for camera distortions

Important cases:

1. Known scene

   Each image point corresponding with a known scene point provides an equation $\underline{v}_p = M\,\underline{v}$

2. Unknown scene

   Several views are needed, differing by rotation and/or translation

   a. Known camera motion

   b. Unknown camera motion ("camera self-calibration")

36

Calibration of One Camera from a Known Scene

- "known scene" = scene with prominent points, whose scene coordinates are known
- prominent points must be non-coplanar to avoid degeneracy

Projection equation $\underline{v}_p = M\,\underline{v}$ provides 2 linear equations for unknown coefficients of M:

$x_p\,(m_{31}x + m_{32}y + m_{33}z + m_{34}) = m_{11}x + m_{12}y + m_{13}z + m_{14}$

$y_p\,(m_{31}x + m_{32}y + m_{33}z + m_{34}) = m_{21}x + m_{22}y + m_{23}z + m_{24}$

Taking N points, N > 6, M can be estimated with a least-square method from an overdetermined system of 2N linear equations.

From $M = [\,KR \;\; K\underline{t}\,] = [\,A\;\underline{b}\,]$, one gets K and R by Principle Component Analysis (PCA) of A and $\underline{t}$ from $\underline{t} = K^{-1}\underline{b}$.

37

Fundamental Matrix

The fundamental matrix F generalizes the essential matrix E by incorporating the intrinsic camera parameters of two (possibly different) cameras.

Essential matrix constraint for 2 views of a point:

$$\underline{n}^T E\,\underline{n}' = 0$$

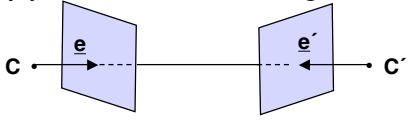
From $\underline{v}_p = K\alpha\,\underline{n}$ and $\underline{v}_p' = K'\beta\,\underline{n}'$ we get:

$$\underline{v}_p\,(K^{-1})^T E\,(K')^{-1}\,\underline{v}_p' = \underline{v}_p\,F\,\underline{v}_p' = 0$$

$$K = \begin{bmatrix} fa & fb & x_{p0} \\ 0 & fc & y_{p0} \\ 0 & 0 & 1 \end{bmatrix}$$

Note that E and hence F have rank 2.

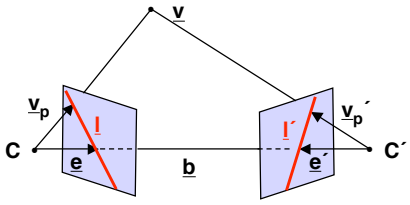
For each epipole of a 2-camera configuration we have $\underline{e}^T F = 0$ and $F\underline{e}' = 0$.



38

Epipolar Plane

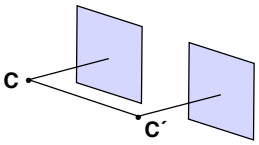
The epipolar plane is spanned by the projection rays of a point $\underline{v}$ and the baseline CC´of a stereo camera configuration.



The epipoles $\underline{e}$ and $\underline{e}´$ are the intersection points of the baseline with the image planes. The epipolar lines $\underline{l}$ and $\underline{l}´$ mark the intersections of the epipolar plane in the left and right image, respectively.

Search for corresponding points in stereo images may be restricted to the epipolar lines.

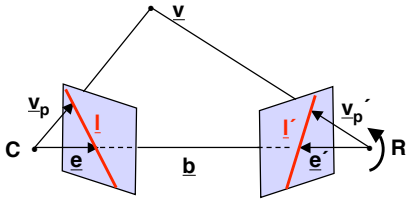
In a canonical stereo configuration (optical axes parallel and perpendicular to baseline) all epipolar lines are parallel:



39

Algebra of Epipolar Geometry

Observation $\underline{v}_p´$ can be modelled as a second observation after translation $\underline{b}$ and rotation R of the optical system.



Coplanarity of $\underline{v}_p$, $\underline{b}$ and $\underline{v}_p´$ (rotated back into coo-system at C) can be expressed as

$$\underline{v}_p \, [\, \underline{b} \times R\underline{v}_p´ \,] = 0 = \underline{v}_p \, [\, b \,] \, R\underline{v}_p´ = \underline{v}_p \, E \, \underline{v}_p´$$

essential matrix

A vector product $\underline{c} \times \underline{d}$ can be written in matrix form:

$$\underline{c} \times \underline{d} = \begin{bmatrix} c_y d_z - c_z d_y \\ c_z d_x - c_x d_z \\ c_x d_y - c_y d_x \end{bmatrix} = \begin{bmatrix} 0 & -c_z & c_y \\ c_z & 0 & -c_x \\ -c_y & c_x & 0 \end{bmatrix} \begin{bmatrix} d_x \\ d_y \\ d_z \end{bmatrix} = [\, c \,] \, \underline{d}$$

40

Correspondence Problem Revisited

For multiple-view 3D analysis it is essential to find corresponding images of a scene point - the correspondence problem.

Difficulties:

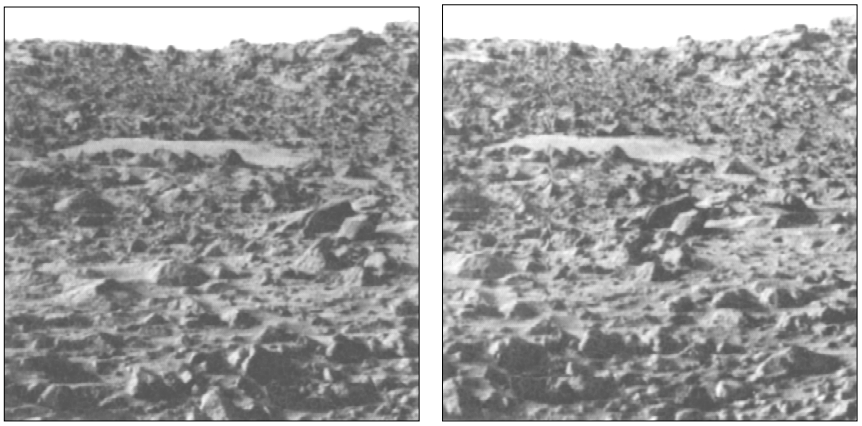
- scene may not offer enough structure to uniquely locate points
- scene may offer too much structure to uniquely locate points
- geometric features may differ strongly between views
- there may be no corresponding point because of occlusion
- photometric features differ strongly between views

Note that difficulties apply to multiple-camera 3D analysis (e.g. binocular stereo) as well as single-camera motion analysis.

41

Correspondence Between Two Mars Images

Two images taken from two cameras of the Viking Lander I (1978). Disparities change rapidly moving from the horizon to nearby structures. (From B.K.P. Horn, Robot Vision, 1986)



42

Constraining Search for Correspondence

The ambiguity of correspondence search may be reduced by several (partly heuristic) constraints.

- **Epipolar constraint**
  reduces search space from 2D to 1D
- **Uniqueness constraint**
  a pixel in one image can correspond to only one pixel in another image
- **Photometric similarity constraint**
  intensities of a point in different images may differ only a little
- **Geometric similarity constraint**
  geometric features of a point in different images may differ only a little
- **Disparity smoothness constraint**
  disparity varies only slowly almost everywhere in the image
- **Physical origin constraint**
  points may correspond only if they mark the same physical location
- **Disparity limit constraint**
  in humans disparity must be smaller than a limit to fuse images
- **Ordering constraint**
  corresponding points lie in the same order on the epipolar line
- **Mutual correspondence constraint**
  correspondence search must succeed irrespective of order of images

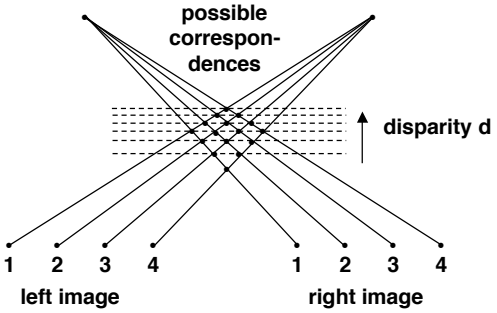
43

Neural Stereo Computation

Neural-network inspired approach to stereo computation devised by Marr and Poggio (1981)

Exploitation of 2 constraints:

- each point in the left image corresponds only to one point in the right image
- depth varies smoothly



Relaxation procedure:
Modify correspondence values c(x, y, d) interatively until values converge.

$$c_{n+1}(x,y,d) = w_1 \sum_{S_1} c_n(x',y',d') - w_2 \sum_{S_2} c_n(x',y',d') + w_0 c_0(x,y,d)$$

S_1 = { neighbours of (x, y) with d´= d }
S_2 = { neighbours of (x, y) with |d´- d| = 1 and (x, y) = (x´, y´) }

44