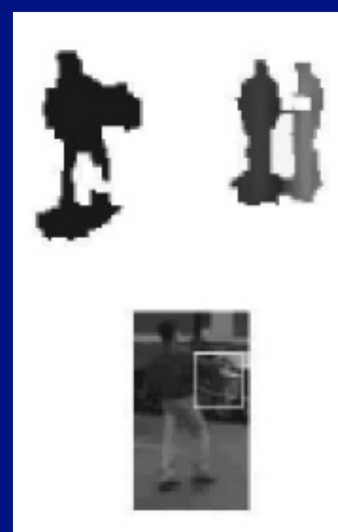
Tracking

- Establish where an object is, other aspects of state, using time sequence
  - Biggest problem -- Data Association
- Key ideas
  - Tracking by detection
  - Tracking through flow

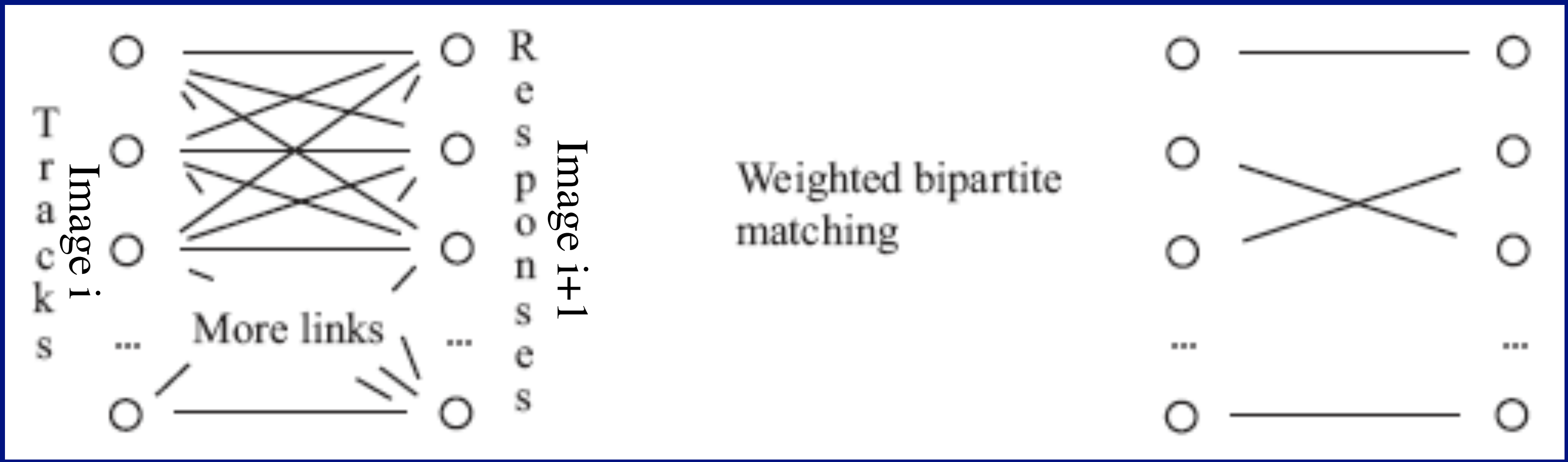
Track by detection (simple form)

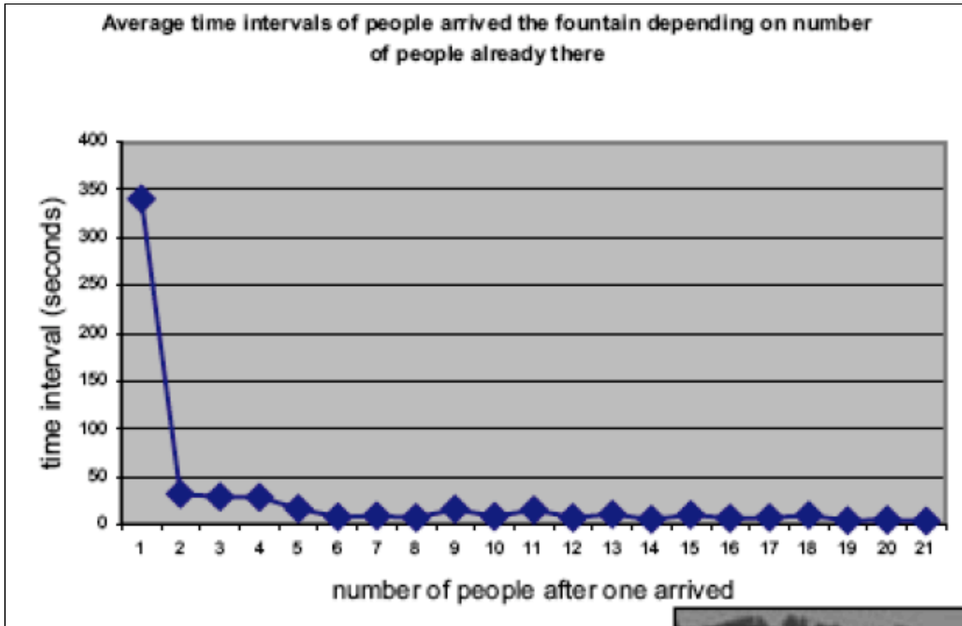
- Assume
  - a very reliable detector (e.g. faces; back of heads)
  - detections that are well spaced in images (or have distinctive properties)
    - e.g. news anchors; heads in public
- Link detects across time
  - only one - easy
  - multiple - weighted bipartite matching



Matching

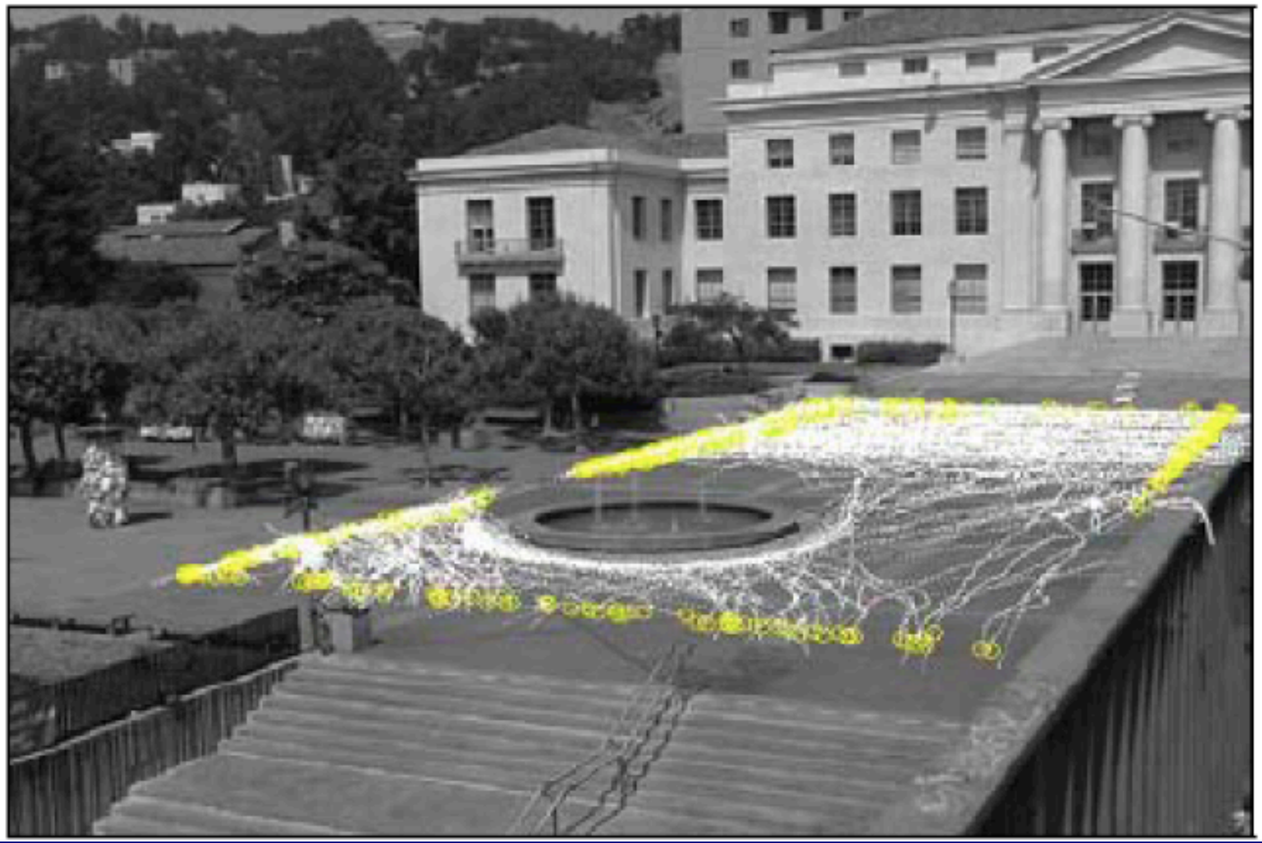
- Established problem
  - Use Hungarian algorithm
  - or nearest neighbours



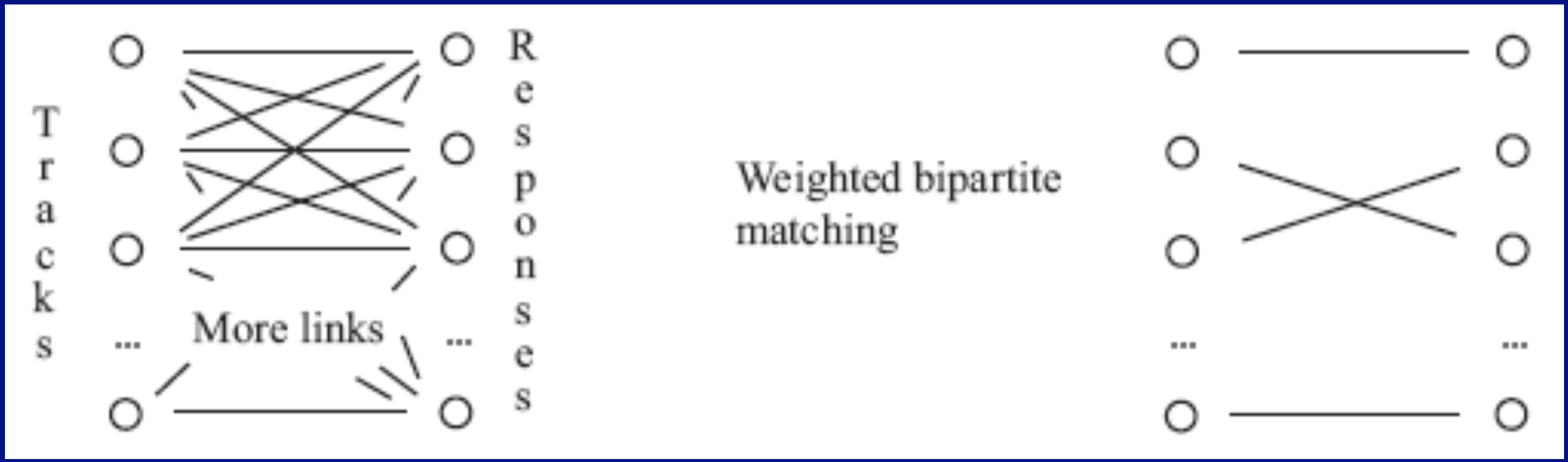


Point tracks reveal curious phenomena in public spaces

Yan+Forsyth, 04



Tracks



- Some detections might fail
- Build “tracks”
  - detect in each frame
  - link detects to tracks using matching algorithm
    - measurements with no track? create new track
    - tracks with no measurement? wait, then reap
  - (perhaps) join tracks over time with global considerations
- What happens if the objects move?

Matching

- Patch is at u, t; moves to u+h, t+1; h is small
- Error is sum of squared differences

$$E(\boldsymbol{h}) = \sum_{\boldsymbol{u} \in \mathcal{P}_t} [I(\boldsymbol{u}, t) - I(\boldsymbol{u} + \boldsymbol{h}, t+1)]^2$$

- This is minimized when

$$\nabla_{\boldsymbol{h}} E(\boldsymbol{h}) = 0.$$

- substitute $I(\boldsymbol{u} + \boldsymbol{h}, t+1) \approx I(\boldsymbol{u}, t) + \boldsymbol{h}^T \nabla I$

- get

$$\left[\sum_{\boldsymbol{u} \in \mathcal{P}_t} (\nabla I)(\nabla I)^T \right] \boldsymbol{h} = \sum_{\boldsymbol{u} \in \mathcal{P}_t} [I(\boldsymbol{u}, t) - I(\boldsymbol{u}, t+1)] \nabla I$$

Matching

- We can tell if the match is good by looking at

$$\left[\sum_{u \in \mathcal{P}_t} (\nabla I)(\nabla I)^T\right]$$

  - which will be poorly conditioned if matching is poor
    - eg featureless region
    - eg flow region

Matching

- Match must work from i to i+1
  - Method is OK so far for this
  - what about 1 to 100?
- Second test; compare with first frame, by minimizing, testing

$$E(\mathcal{M}, \boldsymbol{c}) = \sum_{\boldsymbol{u} \in \mathcal{P}_1} \left[I(\boldsymbol{u}, 1) - I(\mathcal{M}\boldsymbol{u} + \boldsymbol{c}, t) \right]^2 .$$

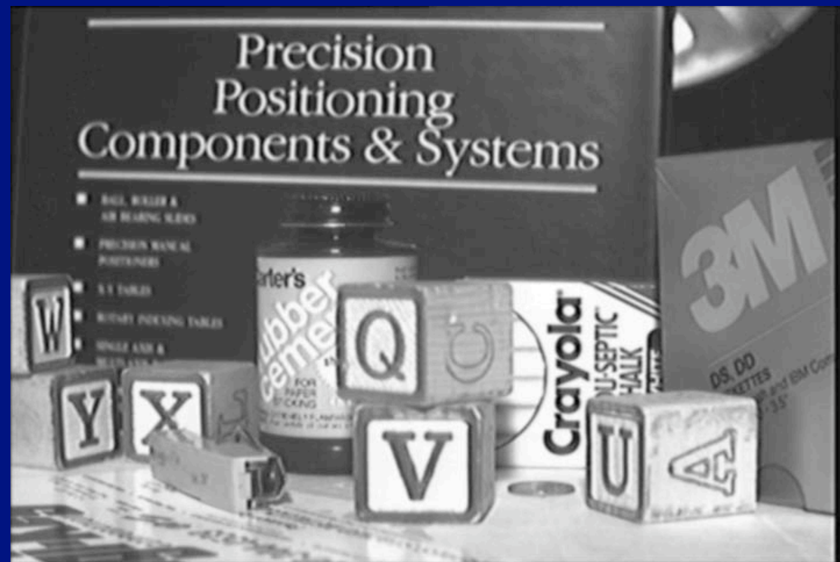
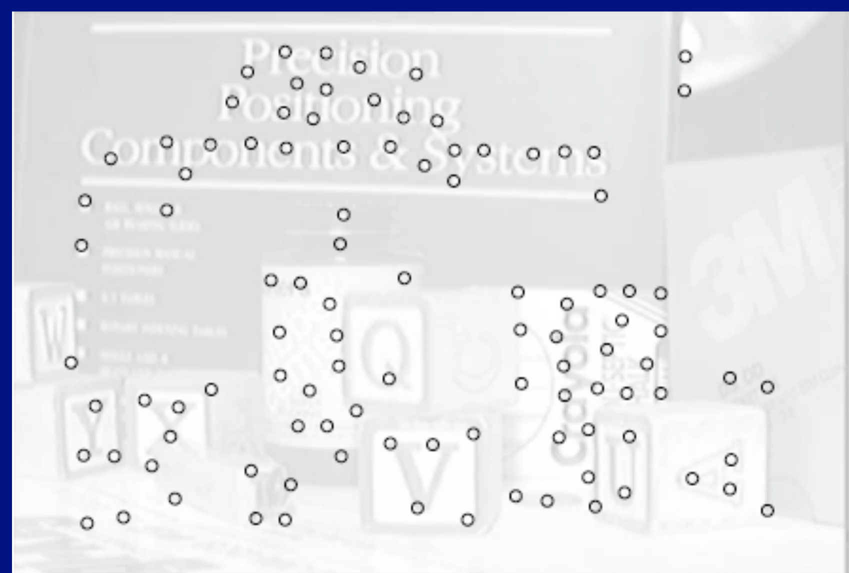
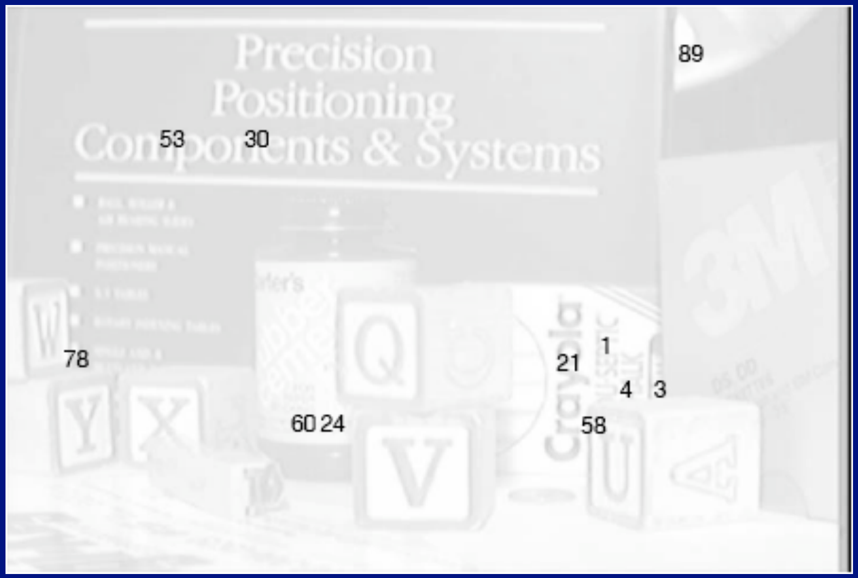


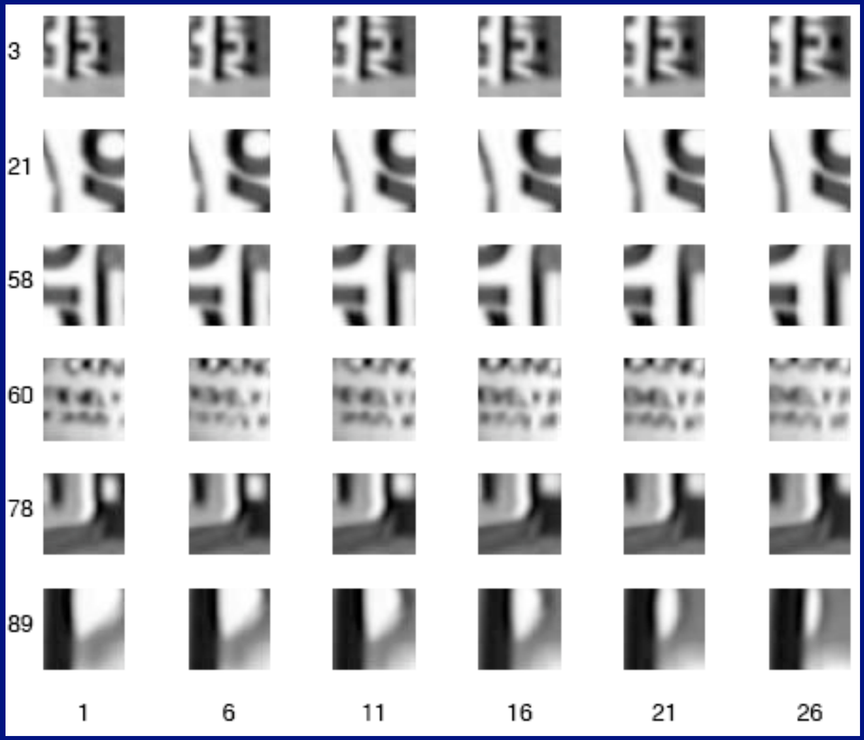
Image frame, from a sequence (Shi Tomasi 94)

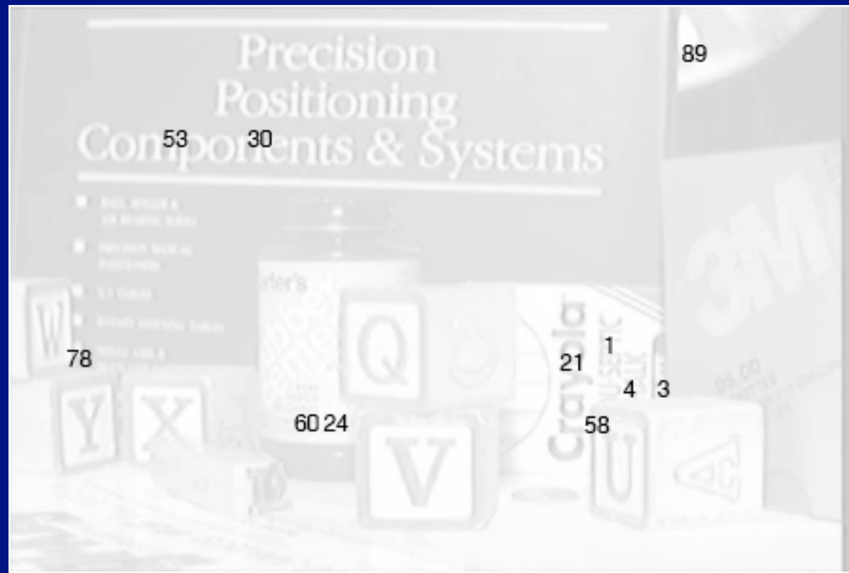


Strongly textured points (Shi Tomasi 94)

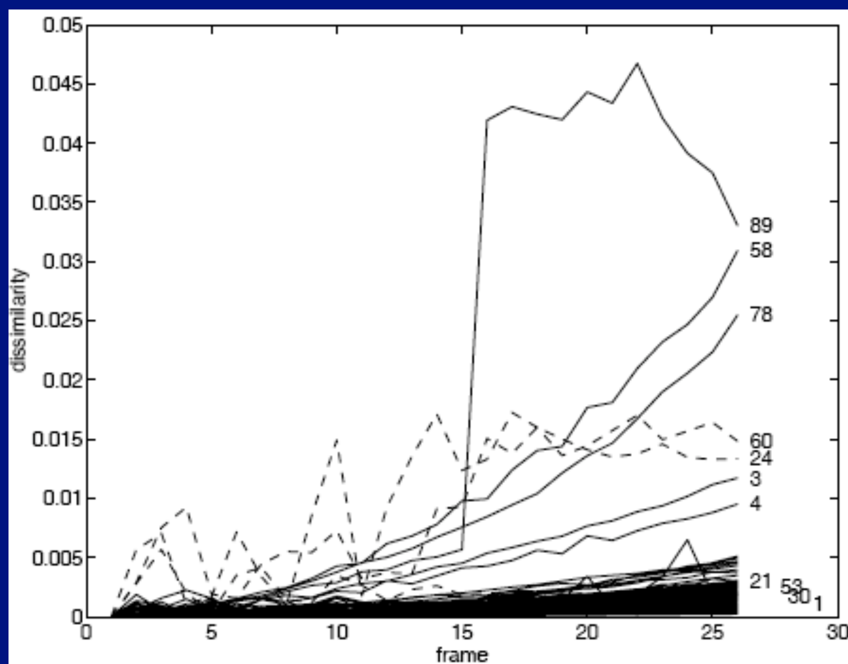


Point patches in tracks (Shi Tomasi 94)





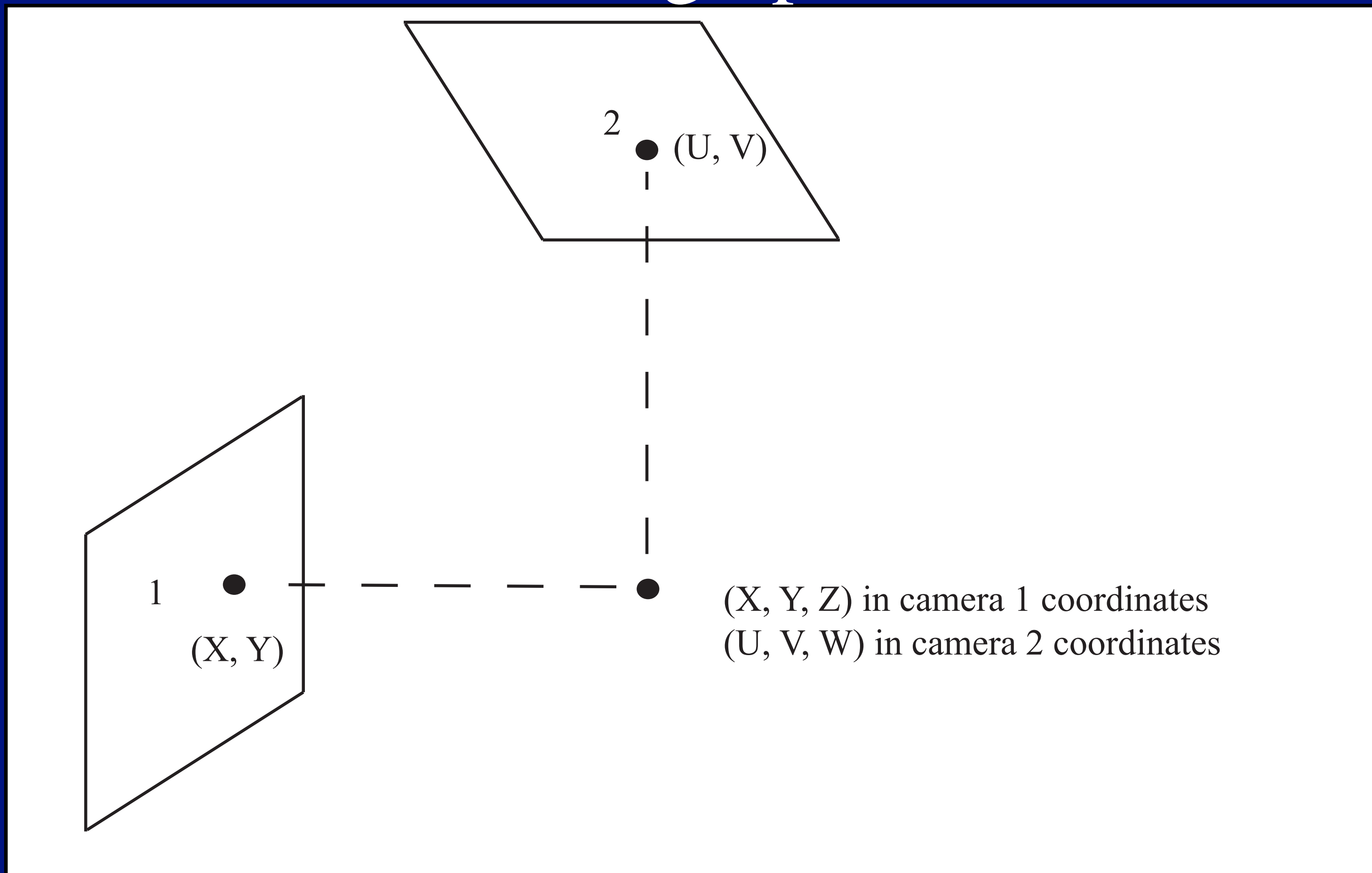
Dissimilarity (Shi Tomasi 94)



Simple reconstruction from multiple views

- Assume
  - fixed set of points, all can be seen in each view
  - orthographic cameras that rotate and translate
  - origin of world coordinates is at center of gravity of points
    - origin in each camera is at center of gravity of points in camera
  - points are tracked, so we know which is which in each view
- All these assumptions can be relaxed, with work

What does an orthographic camera do?

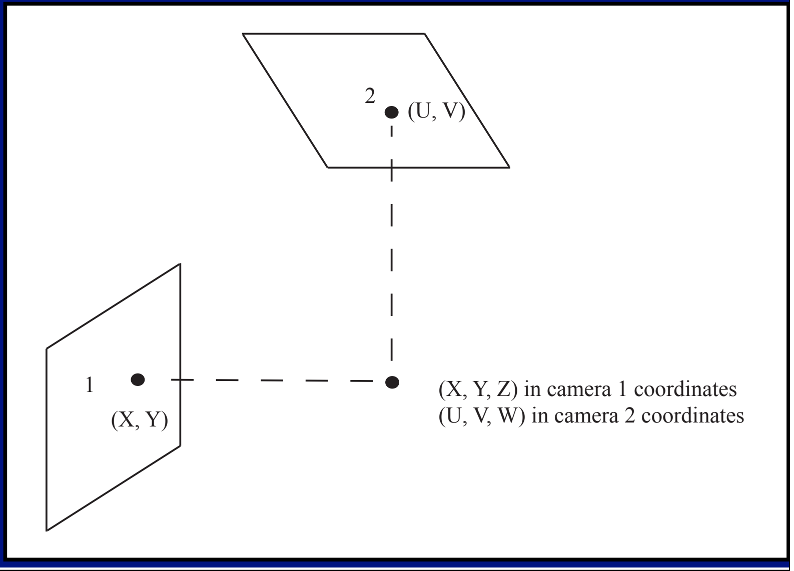


What does an orthographic camera do?

- Camera 2 is rotated and translated
  - with respect to camera 1
  - no translation - center of gravity assumption
- Hence

$$\begin{pmatrix} U \\ V \\ W \end{pmatrix} = \mathcal{R} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix}$$

-



What orthographic cameras do

- Model

$$x_{im} = \mathbf{v}_x^T \mathbf{X}_{3D}$$
$$y_{im} = \mathbf{v}_y^T \mathbf{X}_{3D}$$

- constraints

$$\mathbf{v}_x^T \mathbf{v}_y = 0$$

$$\mathbf{v}_x^T \mathbf{v}_x - \mathbf{v}_y^T \mathbf{v}_y = 0$$

Multiple views

$$x_{i,j} = \mathbf{v}_{x,i}^T \mathbf{X}_j$$

$$y_{i,j} = \mathbf{v}_{y,i}^T \mathbf{X}_j$$

Point index is j

View index is i

Multiple views

$$\begin{pmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,n} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,n} \\ \cdots & & & \\ y_{m,1} & y_{m,2} & \cdots & y_{m,n} \\ y_{1,1} & y_{1,2} & \cdots & y_{1,n} \\ y_{2,1} & y_{2,2} & \cdots & y_{2,n} \\ \cdots & & & \\ y_{m,1} & y_{m,2} & \cdots & y_{m,n} \end{pmatrix} = \begin{pmatrix} \mathbf{v}_{x,1}^T \\ \mathbf{v}_{x,2}^T \\ \cdots \\ \mathbf{v}_{x,m}^T \\ \mathbf{v}_{y,1}^T \\ \mathbf{v}_{y,2}^T \\ \cdots \\ \mathbf{v}_{y,m}^T \end{pmatrix} \begin{pmatrix} \mathbf{X}_1 & \mathbf{X}_2 & \ldots & \mathbf{X}_n \end{pmatrix}$$

$$\mathcal{D} = \mathcal{V}\mathcal{X}$$

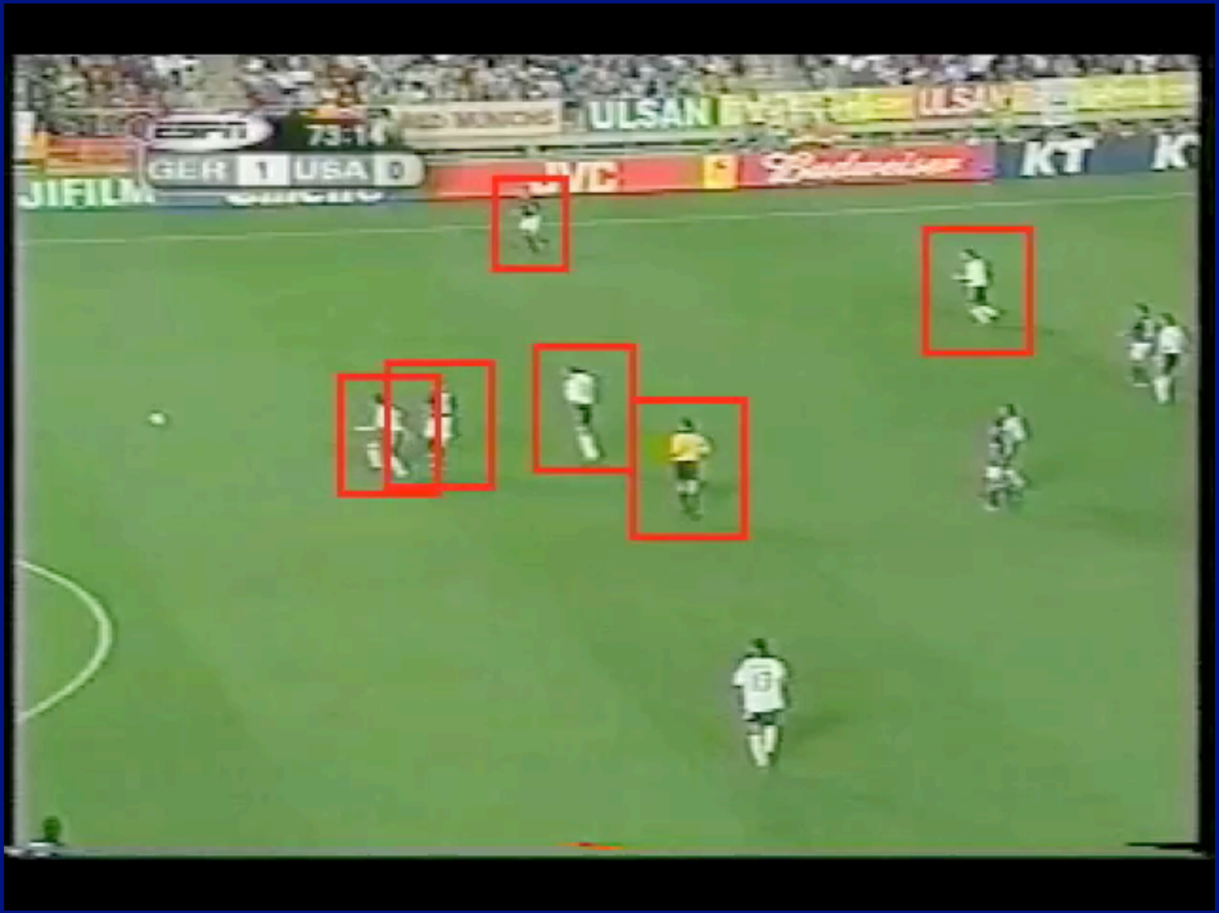
Data - observed!

Multiple views

- The data matrix has rank 3
  - so we can factor it into an mx3 factor and a 3xn factor
- Procedure
  - Build data matrix
  - Factor into point matrix, camera matrix
  - Use constraints to choose correct camera matrix
  - Output:
    - all points in 3D
    - all camera orientations in 3D

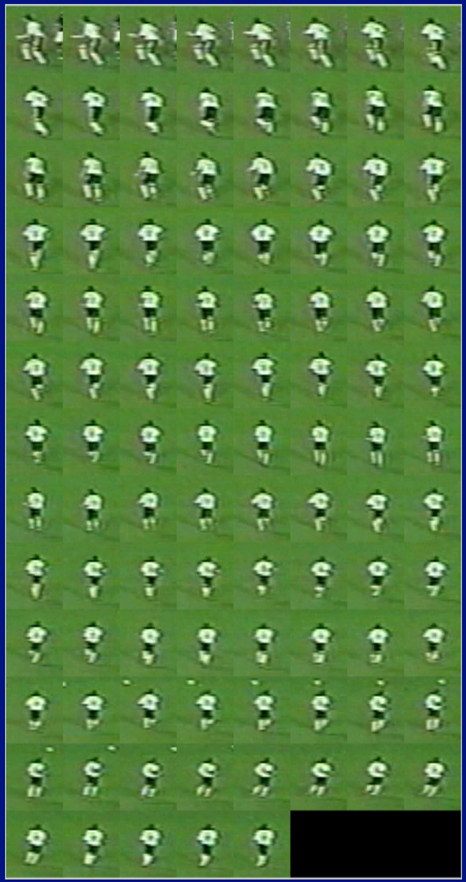
Setting up factorization

- We need to fill in a data matrix
- Strategy
  - find points in one frame
  - link each to corresponding point in next frame; etc.
- Cues for linking
  - patches
    - “look the same”
    - “don’t move much”



Efros et al, 03

Efros et al, 03



Efros et al, 03

What if the pixels get mixed up?

- Describe with histograms
- Match with procedure called “mean shift” (chapter)

When are large motions "easy"?

- When they're "predictable"
  - e.g. ballistic motion
  - e.g. constant velocity
- Need a theory

Tracking - more formal view

- Very general model:
  - We assume there are moving objects, which have an underlying state X
  - There are observations Y, some of which are functions of this state
  - There is a clock
    - at each tick, the state changes
    - at each tick, we get a new observation
- Examples
  - object is ball, state is 3D position+velocity, observations are stereo pairs
  - object is person, state is body configuration, observations are frames, clock is in camera (30 fps)