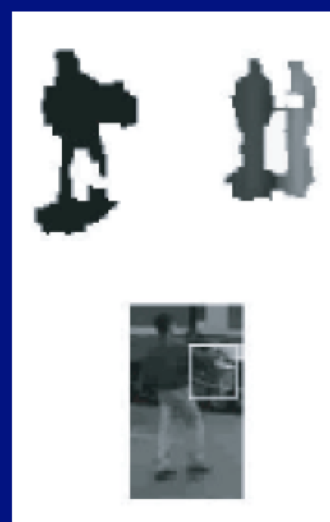
# Tracking

- 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)

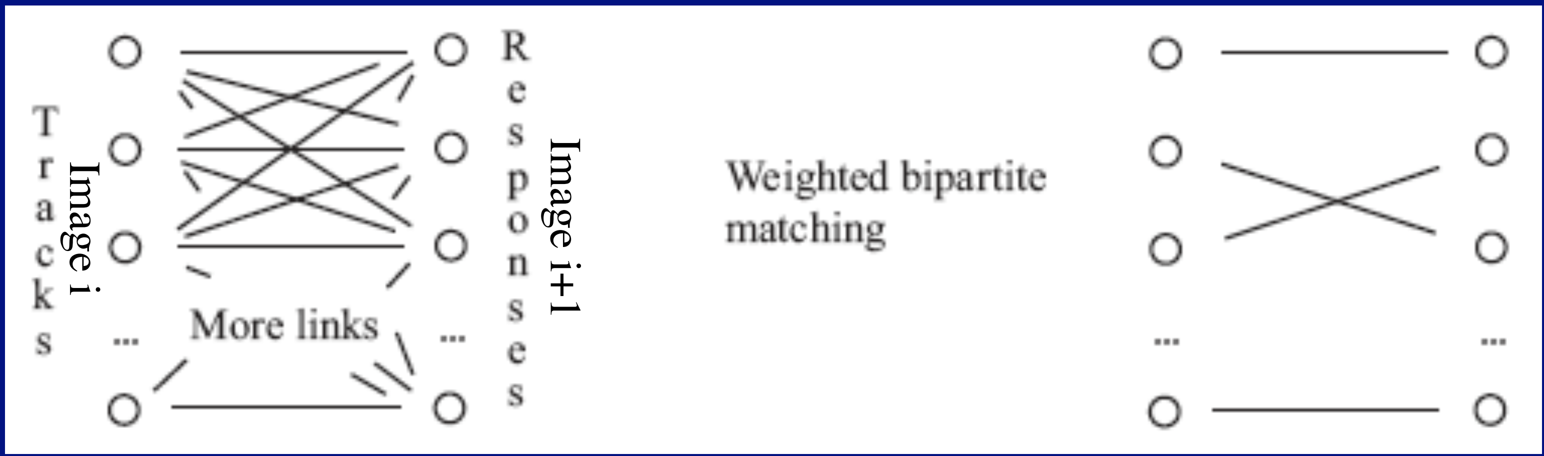
- Assume
  - a very reliable detector (e.g. faces; back of heads)
  - detections that are well spaced in images (or have distinctive properties)
    - e.g. news anchors; heads in public
- Link detects across time
  - only one - easy
  - multiple - weighted bipartite matching

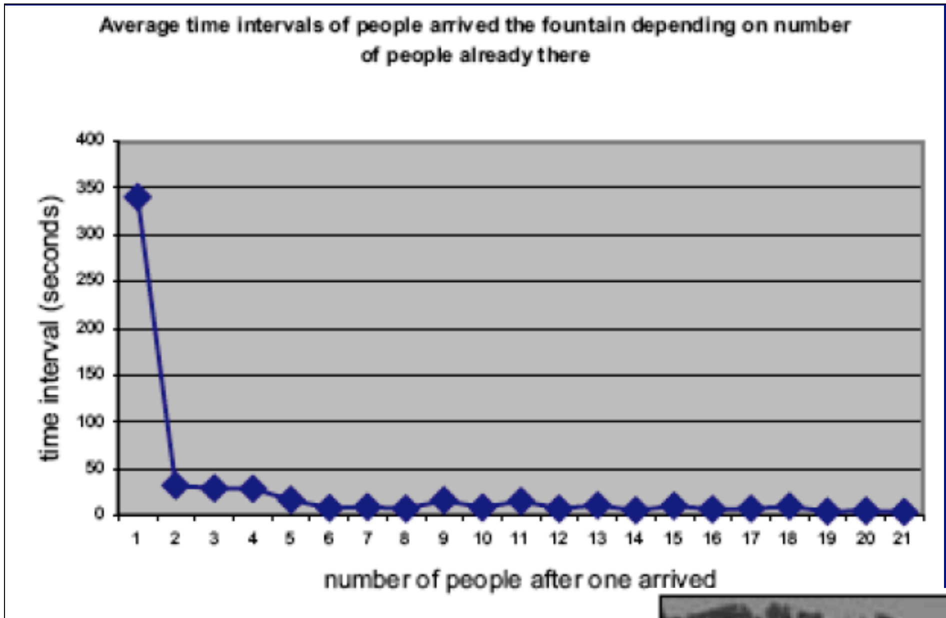




# Matching

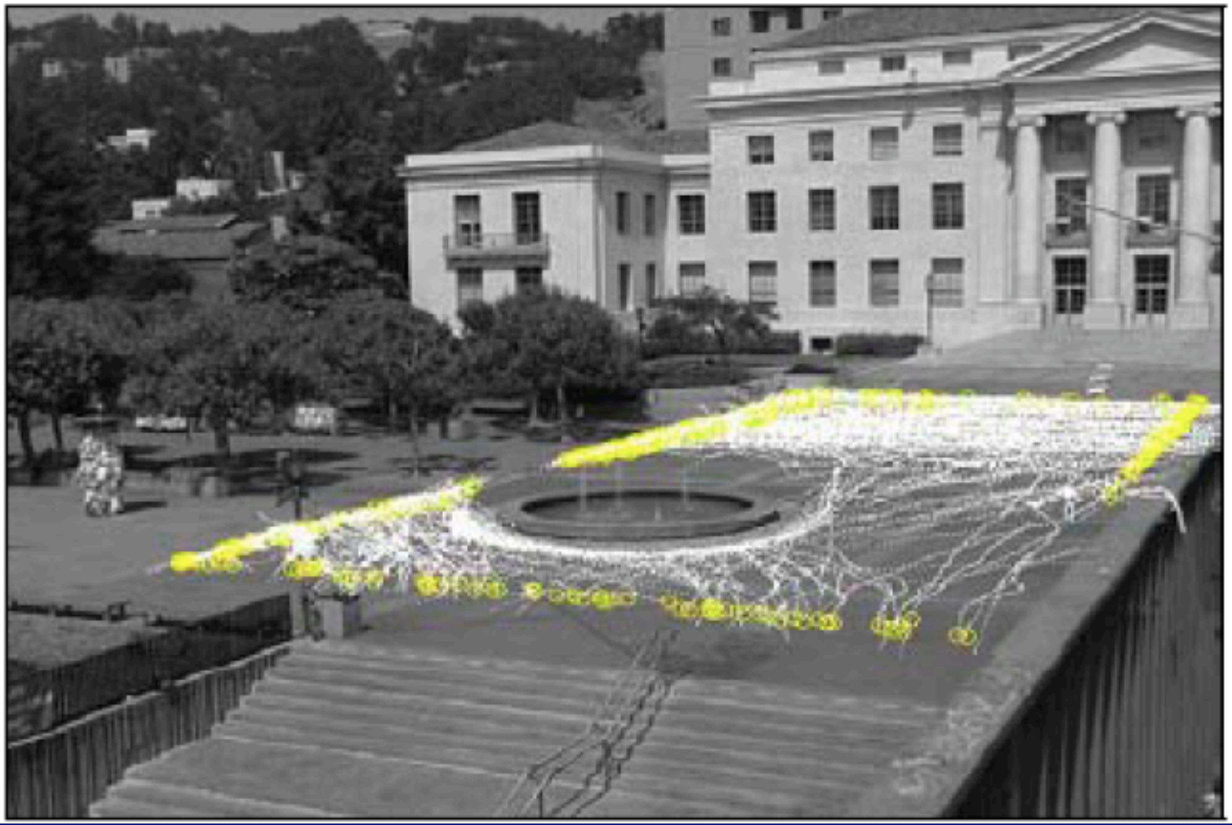
- Established problem
  - Use Hungarian algorithm
  - or nearest neighbours



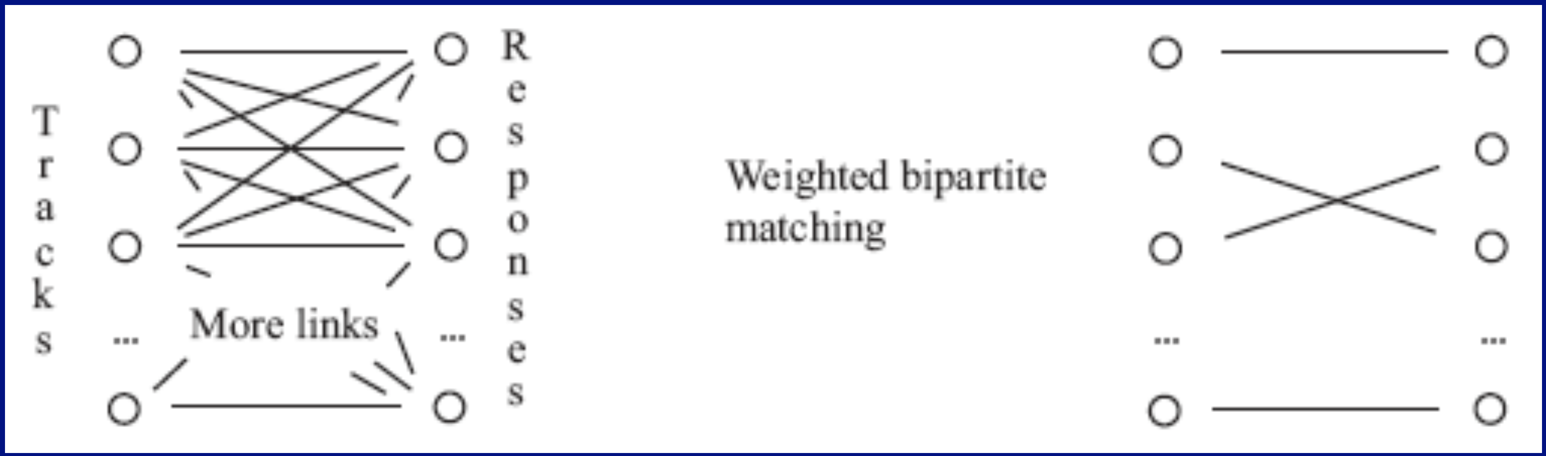



Point tracks reveal curious phenomena in public spaces

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?

# Example:  SFM

- 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"

# Matching

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

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

- This is minimized when

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

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

- get

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

# 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

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

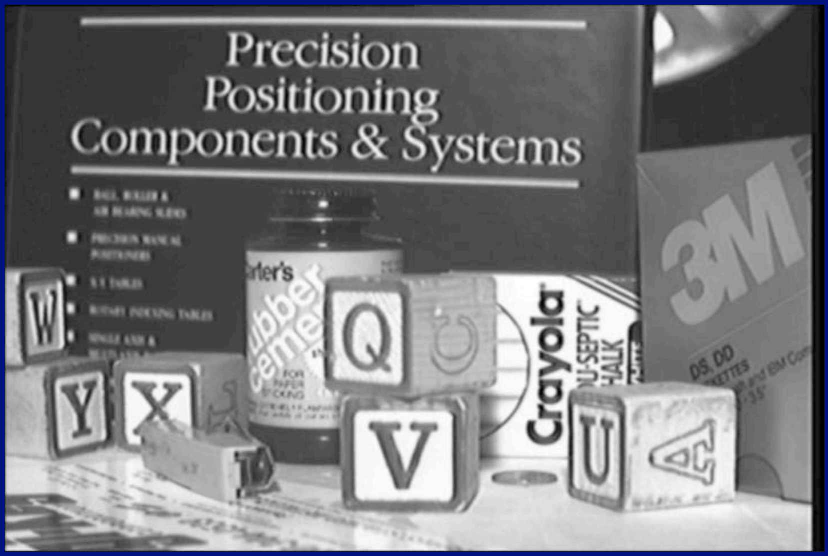
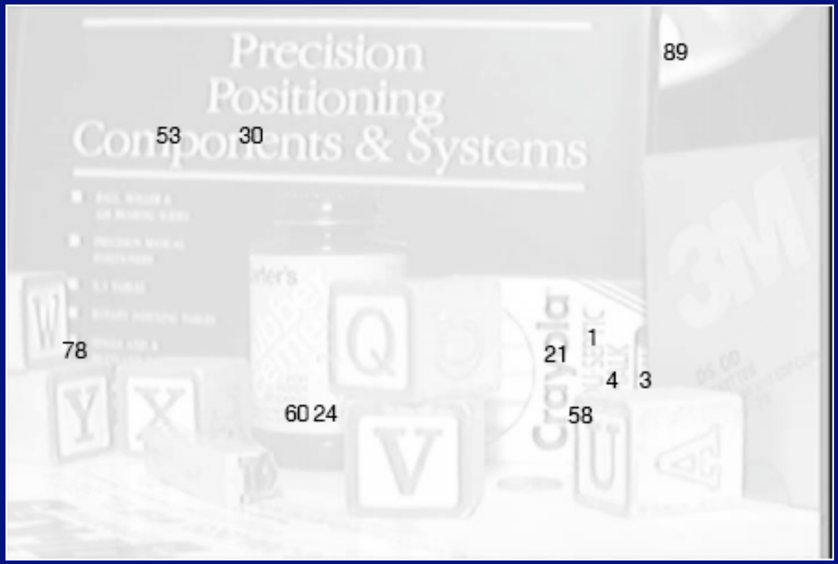
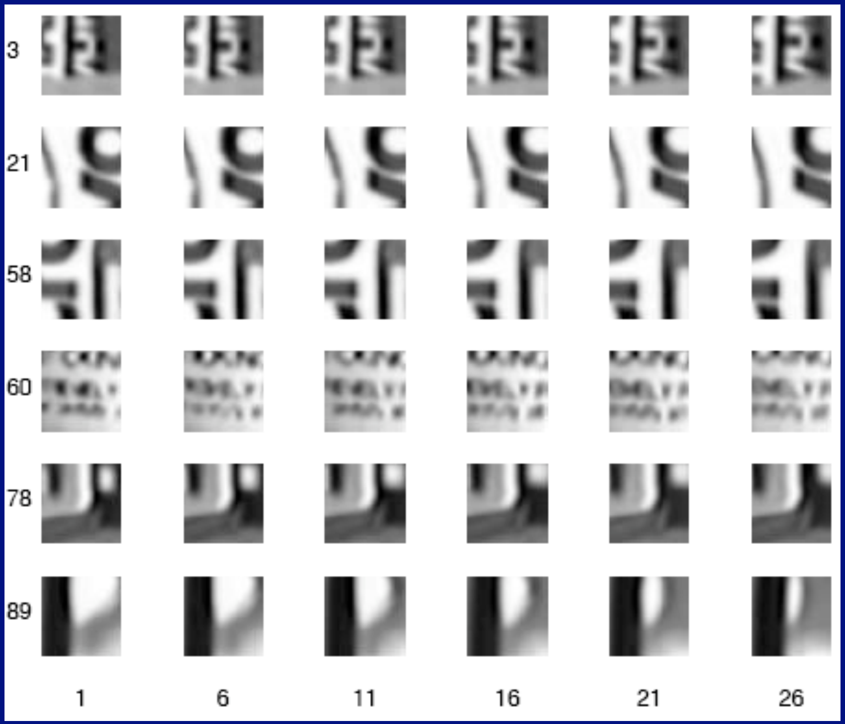


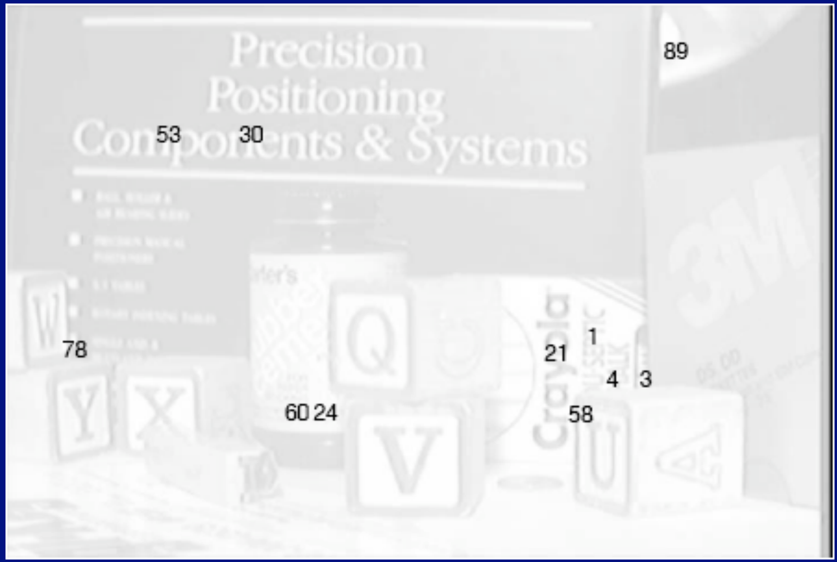

Image frame, from a sequence (Shi Tomasi 94)

Strongly textured points (Shi Tomasi 94)

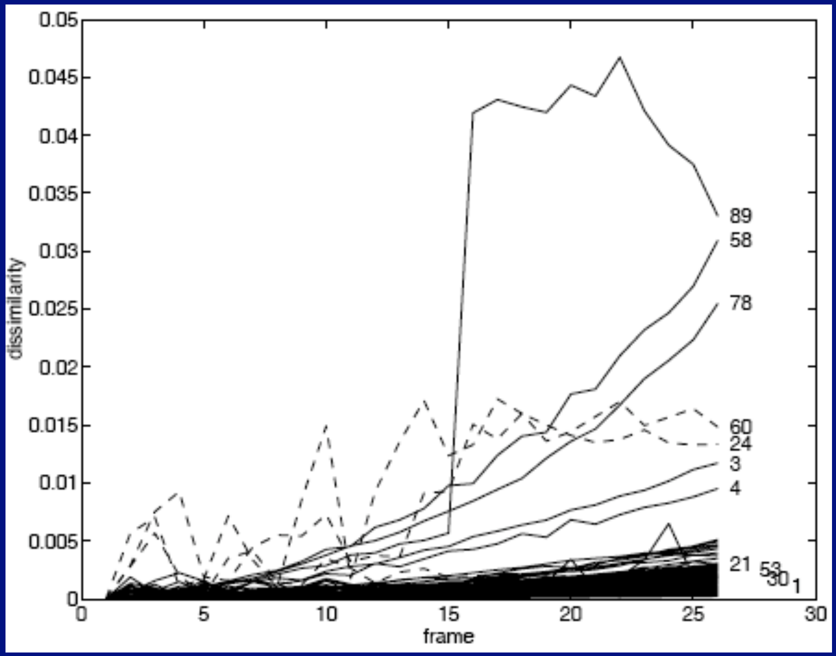



Point patches in tracks (Shi Tomasi 94)



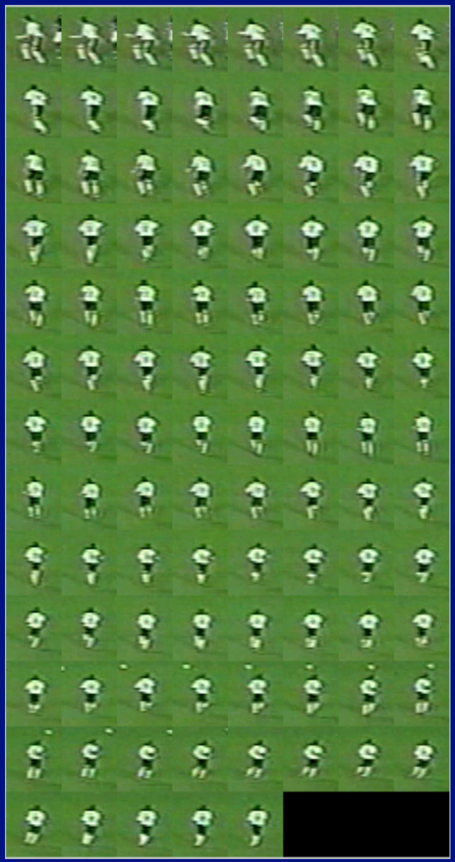



Dissimilarity (Shi Tomasi 94)



Efros et al, 03

[redacted]



Efros et al, 03

# What if the pixels get mixed up?


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

# Track by flow (simple form)

- Assume
  - appearance unknown (but domain, parametric flow model known)
  - optic flow assumptions, as before
- Initialize
  - mark out domain
- Track
  - choose flow model parameters that align domain in pic n with n+1 best
  - push domain through flow model

$$\sum_{\mathbf{x}\in\mathcal{D}_t}\left[\mathcal{I}(\mathbf{x},t)-\mathcal{I}(\mathbf{x}+\mathbf{V}(\mathbf{x},\theta),t)\right]^2$$

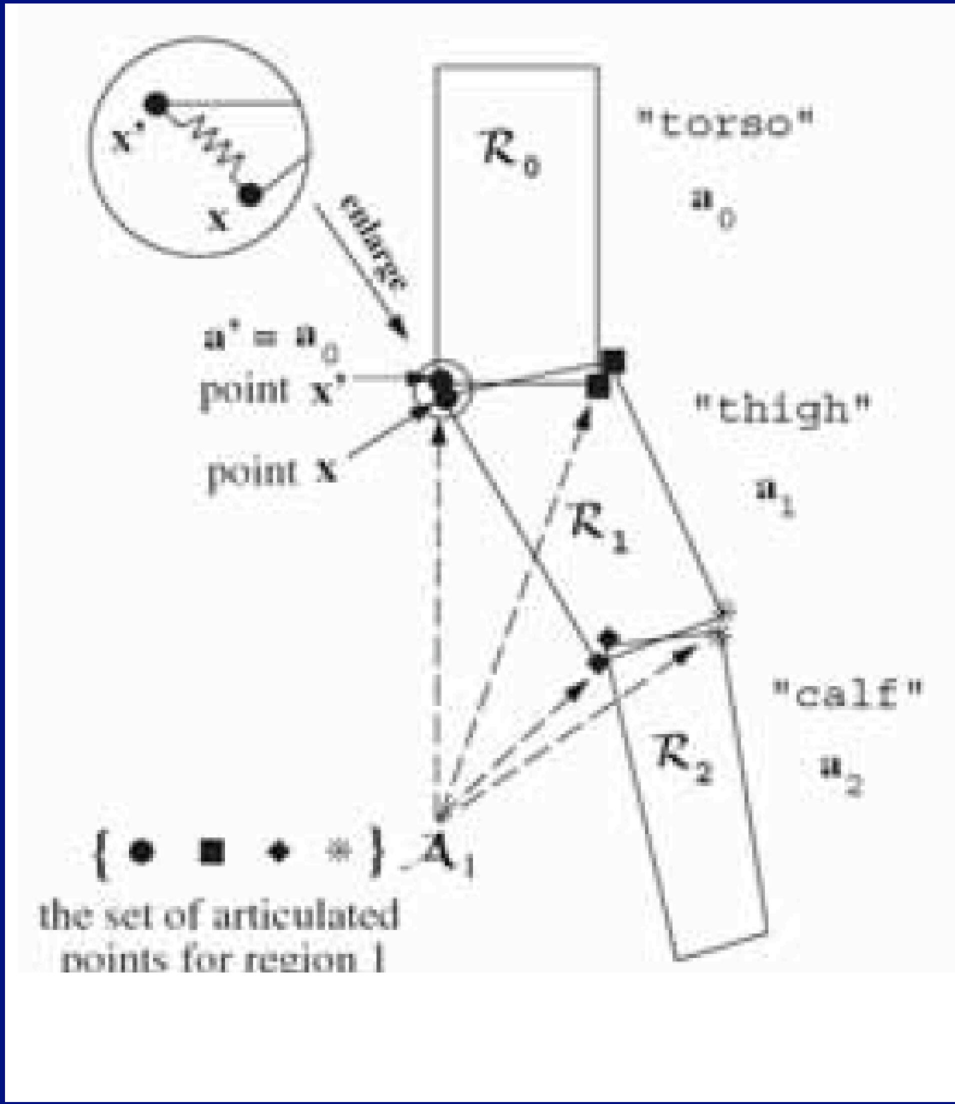
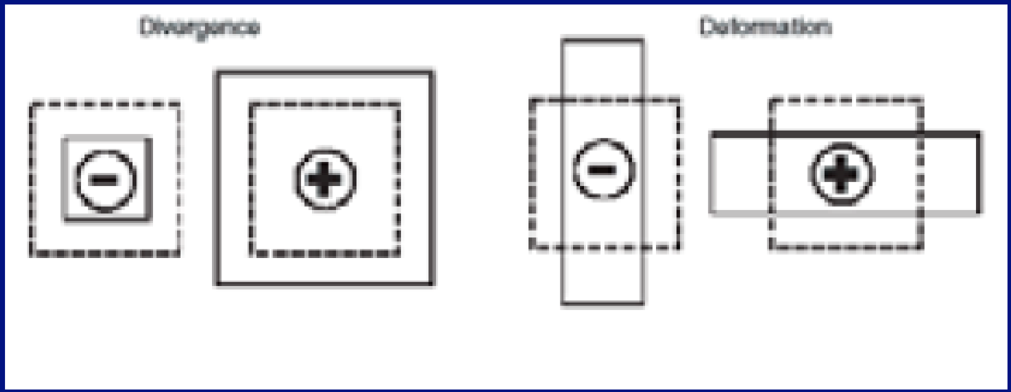



Figure from Ju, Black and Yacoob, “Cardboard people”

Figure from Ju, Black and Yacoob, “Cardboard people”



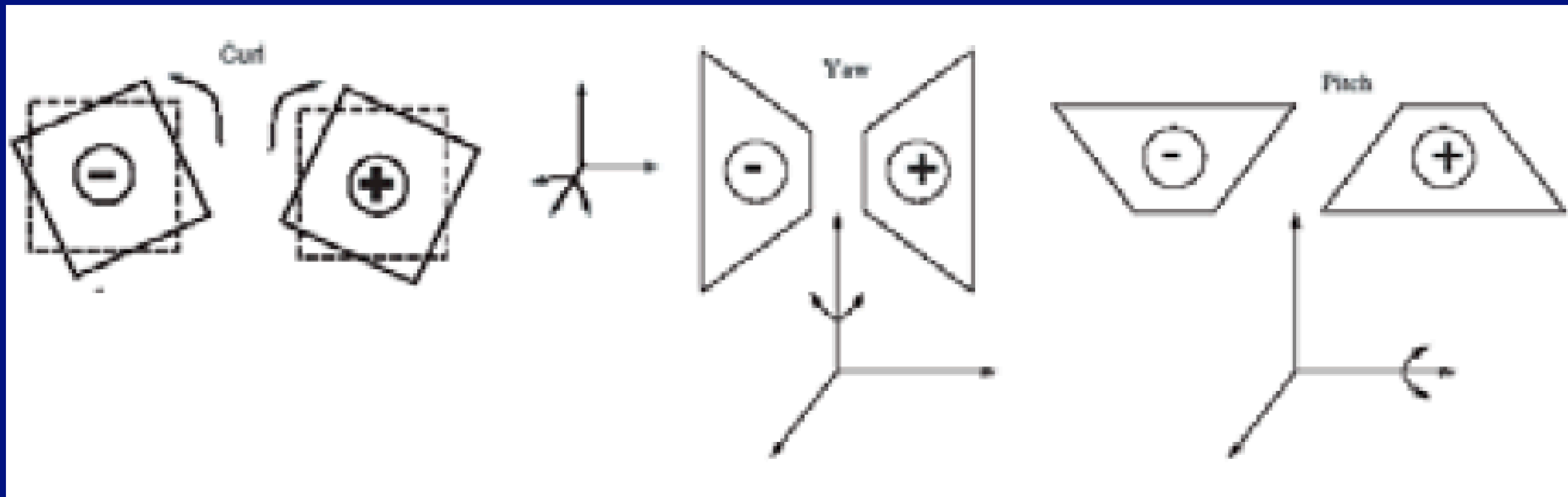

$\mathbf{a} = (0, 1, 0, 0, 0, 1, 0, 0)$

$\mathbf{a} = (0, 1, 0, 0, 0, -1, 0, 0).$

$(u(\mathbf{x}), v(\mathbf{x})^T =$

$(a_0 + a_1 x + a_2 y + a_6 x^2 + a_7 xy, a_3 + a_4 x + a_5 y + a_6 xy + a_7 y^2)$

Figure from Ju, Black and Yacoob, "Cardboard people"




$$(0, 0, -1, 0, 1, 0, 0, 0)$$

$$(0, 0, 0, 0, 0, 1, 0)$$

$$(0, 0, 0, 0, 0, 0, 1)$$

Figure from Ju, Black and Yacoob, “Cardboard people”

# Dangers

- Loss of track
  - small errors accumulate in model of appearance
  - DRIFT
- Appearance often isn't constant

# 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)

# Tracking - Probabilistic formulation

- Given
  - $P(X_{i-1}|Y_0, ..., Y_{i-1})$
    - "Prior"
- We should like to know
  - $P(X_i|Y_0, ..., Y_{i-1})$
    - "Predictive distribution"
  - $P(X_i|Y_0, ..., Y_i)$
    - "Posterior"

## The three main issues in tracking

- **Prediction:** we have seen $\boldsymbol{y}_0, \ldots, \boldsymbol{y}_{i-1}$ — what state does this set of measurements predict for the $i$'th frame? to solve this problem, we need to obtain a representation of $P(\boldsymbol{X}_i | \boldsymbol{Y}_0 = \boldsymbol{y}_0, \ldots, \boldsymbol{Y}_{i-1} = \boldsymbol{y}_{i-1})$.
- **Data association:** Some of the measurements obtained from the $i$-th frame may tell us about the object's state. Typically, we use $P(\boldsymbol{X}_i | \boldsymbol{Y}_0 = \boldsymbol{y}_0, \ldots, \boldsymbol{Y}_{i-1} = \boldsymbol{y}_{i-1})$ to identify these measurements.
- **Correction:** now that we have $\boldsymbol{y}_i$ — the relevant measurements — we need to compute a representation of $P(\boldsymbol{X}_i | \boldsymbol{Y}_0 = \boldsymbol{y}_0, \ldots, \boldsymbol{Y}_i = \boldsymbol{y}_i)$.

Key assumptions:

- **Only the immediate past matters:** formally, we require

$$P(\boldsymbol{X}_i|\boldsymbol{X}_1,\ldots,\boldsymbol{X}_{i-1}) = P(\boldsymbol{X}_i|\boldsymbol{X}_{i-1})$$

  This assumption hugely simplifies the design of algorithms, as we shall see; furthermore, it isn't terribly restrictive if we're clever about interpreting $\boldsymbol{X}_i$ as we shall show in the next section.

- **Measurements depend only on the current state:** we assume that $\boldsymbol{Y}_i$ is conditionally independent of all other measurements given $\boldsymbol{X}_i$. This means that

$$P(\boldsymbol{Y}_i,\boldsymbol{Y}_j,\ldots\boldsymbol{Y}_k|\boldsymbol{X}_i) = P(\boldsymbol{Y}_i|\boldsymbol{X}_i)P(\boldsymbol{Y}_j,\ldots,\boldsymbol{Y}_k|\boldsymbol{X}_i)$$

  Again, this isn't a particularly restrictive or controversial assumption, but it yields important simplifications.

## Tracking as Induction - base case

Firstly, we assume that we have $P(\boldsymbol{X}_0)$

$$
\begin{aligned}
P(\boldsymbol{X}_0|\boldsymbol{Y}_0=\boldsymbol{y}_0) &= \frac{P(\boldsymbol{y}_0|\boldsymbol{X}_0)P(\boldsymbol{X}_0)}{P(\boldsymbol{y}_0)} \\
&= \frac{P(\boldsymbol{y}_0|\boldsymbol{X}_0)P(\boldsymbol{X}_0)}{\int P(\boldsymbol{y}_0|\boldsymbol{X}_0)P(\boldsymbol{X}_0)d\boldsymbol{X}_0} \\
&\propto P(\boldsymbol{y}_0|\boldsymbol{X}_0)P(\boldsymbol{X}_0)
\end{aligned}
$$

## Tracking as induction - induction step

Given

$P(\boldsymbol{X}_{i-1}|\boldsymbol{y}_0, \ldots, \boldsymbol{y}_{i-1})$.

**Prediction**

Prediction involves representing

$$P(\boldsymbol{X}_i|\boldsymbol{y}_0, \ldots, \boldsymbol{y}_{i-1})$$

Our independence assumptions make it possible to write

$$\begin{aligned} P(\boldsymbol{X}_i|\boldsymbol{y}_0, \ldots, \boldsymbol{y}_{i-1}) &= \int P(\boldsymbol{X}_i, \boldsymbol{X}_{i-1}|\boldsymbol{y}_0, \ldots, \boldsymbol{y}_{i-1}) d\boldsymbol{X}_{i-1} \\ &= \int P(\boldsymbol{X}_i|\boldsymbol{X}_{i-1}, \boldsymbol{y}_0, \ldots, \boldsymbol{y}_{i-1}) P(\boldsymbol{X}_{i-1}|\boldsymbol{y}_0, \ldots, \boldsymbol{y}_{i-1}) d\boldsymbol{X}_{i-1} \\ &= \int P(\boldsymbol{X}_i|\boldsymbol{X}_{i-1}) P(\boldsymbol{X}_{i-1}|\boldsymbol{y}_0, \ldots, \boldsymbol{y}_{i-1}) d\boldsymbol{X}_{i-1} \end{aligned}$$

# Tracking as induction - induction step

**Correction**

Correction involves obtaining a representation of

$$P(\boldsymbol{X}_i|\boldsymbol{y}_0, \ldots, \boldsymbol{y}_i)$$

Our independence assumptions make it possible to write

$$\begin{aligned}
P(\boldsymbol{X}_i|\boldsymbol{y}_0, \ldots, \boldsymbol{y}_i) &= \frac{P(\boldsymbol{X}_i, \boldsymbol{y}_0, \ldots, \boldsymbol{y}_i)}{P(\boldsymbol{y}_0, \ldots, \boldsymbol{y}_i)} \\
&= \frac{P(\boldsymbol{y}_i|\boldsymbol{X}_i, \boldsymbol{y}_0, \ldots, \boldsymbol{y}_{i-1})P(\boldsymbol{X}_i|\boldsymbol{y}_0, \ldots, \boldsymbol{y}_{i-1})P(\boldsymbol{y}_0, \ldots, \boldsymbol{y}_{i-1})}{P(\boldsymbol{y}_0, \ldots, \boldsymbol{y}_i)} \\
&= P(\boldsymbol{y}_i|\boldsymbol{X}_i)P(\boldsymbol{X}_i|\boldsymbol{y}_0, \ldots, \boldsymbol{y}_{i-1})\frac{P(\boldsymbol{y}_0, \ldots, \boldsymbol{y}_{i-1})}{P(\boldsymbol{y}_0, \ldots, \boldsymbol{y}_i)} \\
&= \frac{P(\boldsymbol{y}_i|\boldsymbol{X}_i)P(\boldsymbol{X}_i|\boldsymbol{y}_0, \ldots, \boldsymbol{y}_{i-1})}{\int P(\boldsymbol{y}_i|\boldsymbol{X}_i)P(\boldsymbol{X}_i|\boldsymbol{y}_0, \ldots, \boldsymbol{y}_{i-1})d\boldsymbol{X}_i}
\end{aligned}$$

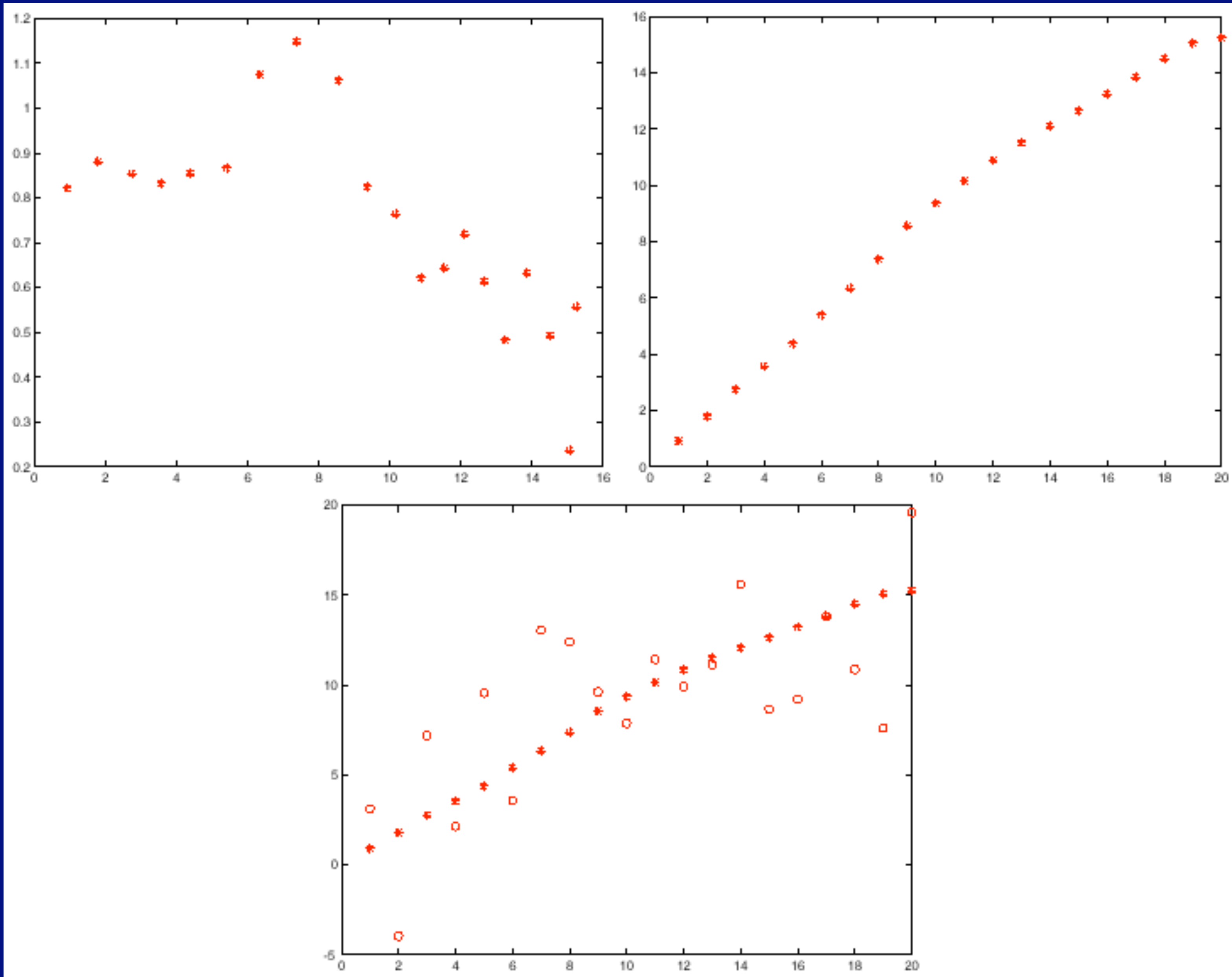
Linear Dynamic Models

$$x_i \sim N(\mathcal{D}_i \boldsymbol{x}_{i-1}; \Sigma_{d_i})$$

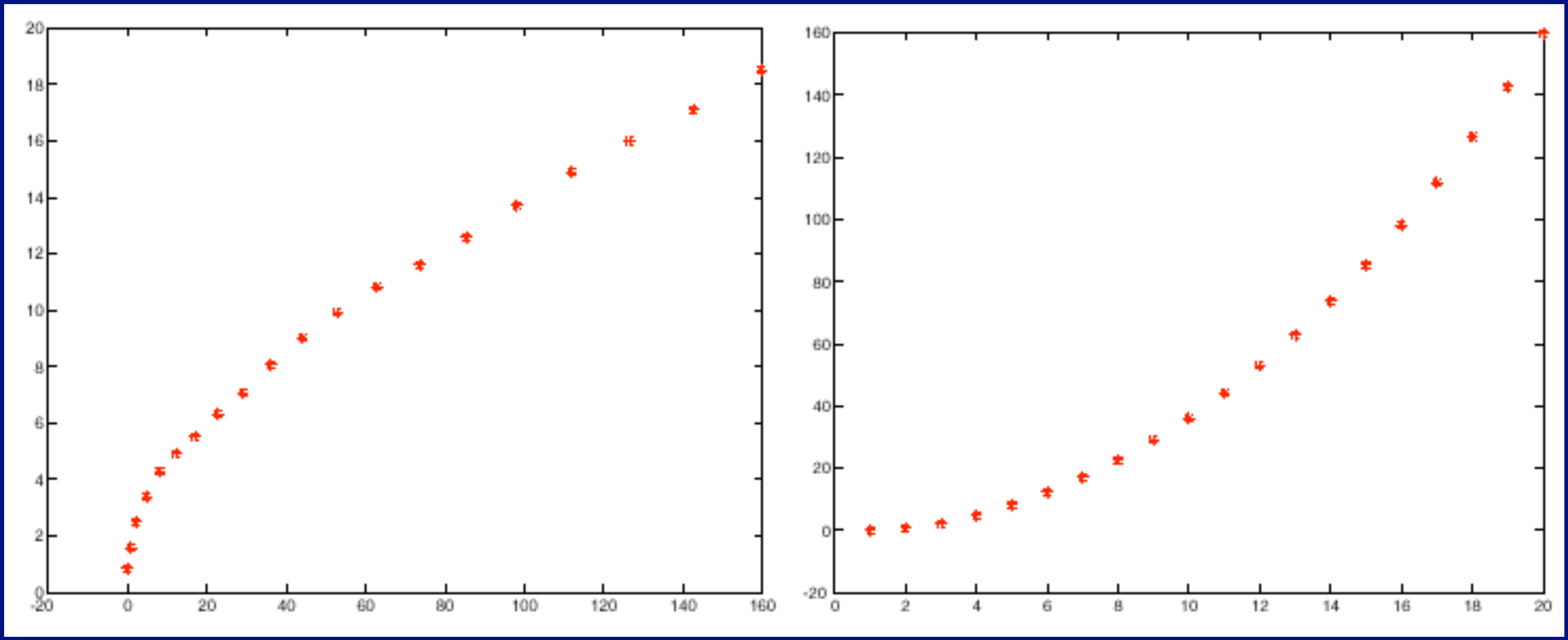
$$y_i \sim N(\mathcal{M}_i \boldsymbol{x}_i; \Sigma_{m_i})$$

# Examples

- Drifting points
  - Observability
- Points moving with constant velocity
- Points moving with constant acceleration
- Periodic motion
- Etc.



1.2
1.1
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0
2
4
6
8
10
12
14
16
0
2
4
6
8
10
12
14
16
0
2
4
6
8
10
12
14
16
18
20
20
15
10
5
0
-5
0
2
4
6
8
10
12
14
16
18
20


20
18
16
14
12
10
8
6
4
2
0
-20
0
20
40
60
80
100
120
140
160
160
140
120
100
80
60
40
20
0
-20
0
2
4
6
8
10
12
14
16
18
20

# The Kalman Filter

- Key ideas:
  - Linear models interact uniquely well with Gaussian noise - make the prior Gaussian, everything else Gaussian and the calculations are easy
  - Gaussians are really easy to represent --- once you know the mean and covariance, you're done

# The Kalman Filter in 1D

- Dynamic Model

$$x_i \sim N(d_i x_{i-1}, \sigma_{d_i}^2)$$

$$y_i \sim N(m_i x_i, \sigma_{m_i}^2)$$

- Notation

mean of $P(X_i|y_0, \ldots, y_{i-1})$ as $\overline{X}_i^-$

mean of $P(X_i|y_0, \ldots, y_i)$ as $\overline{X}_i^+$

the standard deviation of $P(X_i|y_0, \ldots, y_{i-1})$ as $\sigma_i^-$

of $P(X_i|y_0, \ldots, y_i)$ as $\sigma_i^+$

**Dynamic Model:**

$$x_i \sim N(d_i x_{i-1}, \sigma_{d_i})$$

$$y_i \sim N(m_i x_i, \sigma_{m_i})$$

**Start Assumptions:** $\overline{x}_0^-$ and $\sigma_0^-$ are known
**Update Equations:** Prediction

$$\overline{x}_i^- = d_i \overline{x}_{i-1}^+$$

$$\sigma_i^- = \sqrt{\sigma_{d_i}^2 + (d_i \sigma_{i-1}^+)^2}$$

**Update Equations:** Correction

$$x_i^+ = \left( \frac{\overline{x}_i^- \sigma_{m_i}^2 + m_i y_i (\sigma_i^-)^2}{\sigma_{m_i}^2 + m_i^2 (\sigma_i^-)^2} \right)$$

$$\sigma_i^+ = \sqrt{\left( \frac{\sigma_{m_i}^2 (\sigma_i^-)^2}{(\sigma_{m_i}^2 + m_i^2 (\sigma_i^-)^2)} \right)}$$

**Dynamic Model:**

$$\boldsymbol{x}_i \sim N(\mathcal{D}_i \boldsymbol{x}_{i-1}, \Sigma_{d_i})$$

$$\boldsymbol{y}_i \sim N(\mathcal{M}_i \boldsymbol{x}_i, \Sigma_{m_i})$$

**Start Assumptions:** $\overline{\boldsymbol{x}}_0^-$ and $\Sigma_0^-$ are known
**Update Equations:** Prediction

$$\overline{\boldsymbol{x}}_i^- = \mathcal{D}_i \overline{\boldsymbol{x}}_{i-1}^+$$

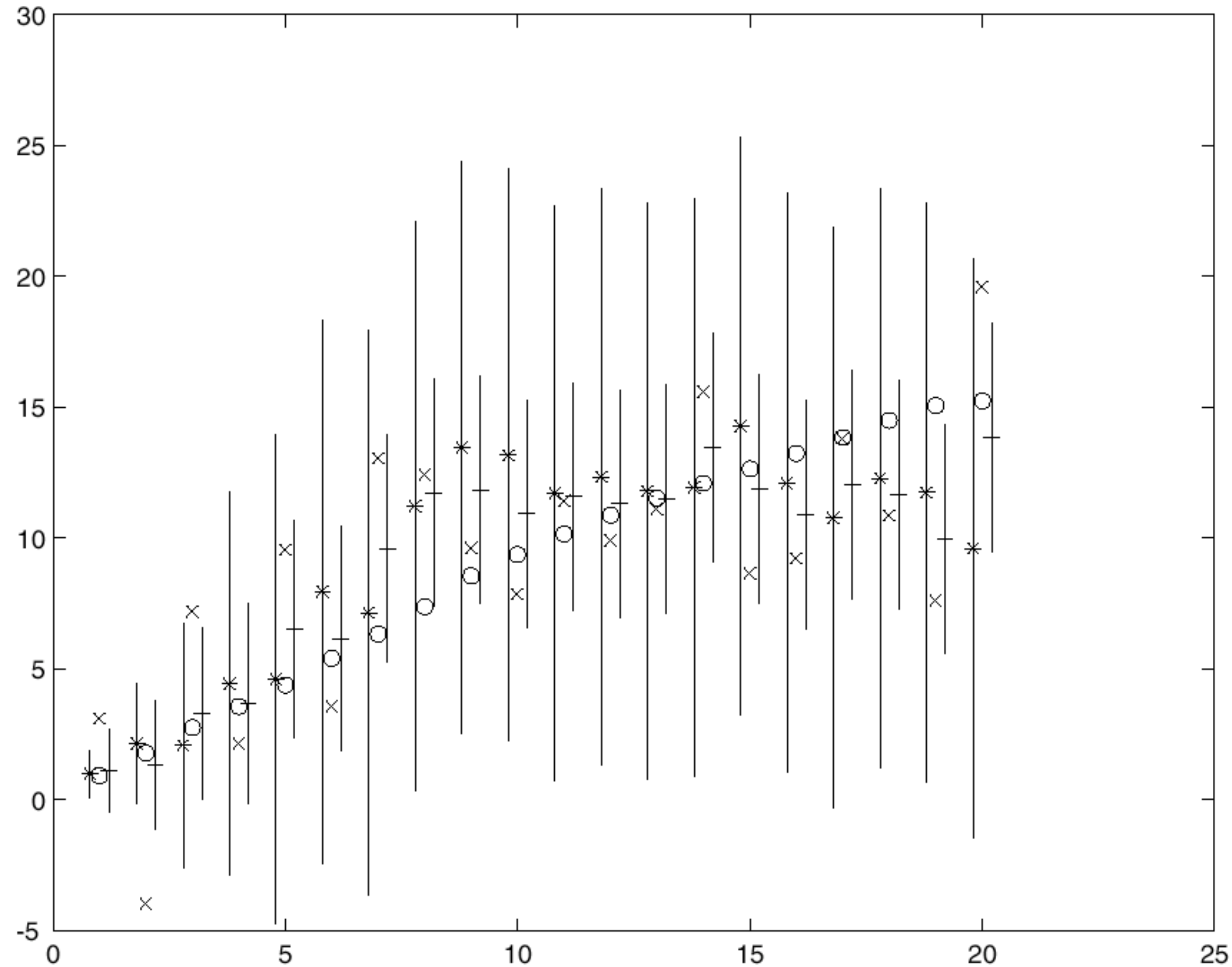
$$\Sigma_i^- = \Sigma_{d_i} + \mathcal{D}_i \sigma_{i-1}^+ \mathcal{D}_i$$

**Update Equations:** Correction

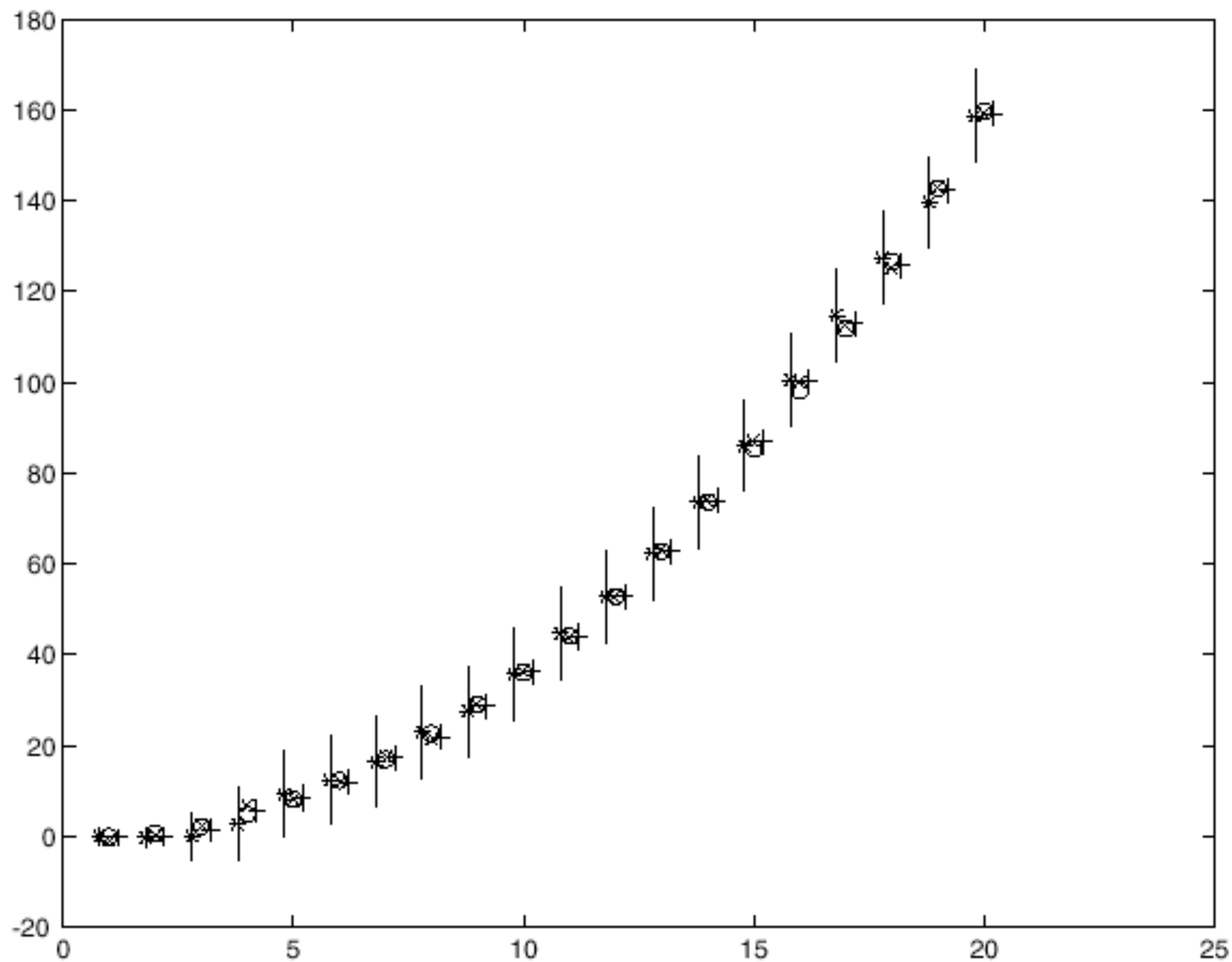
$$\mathcal{K}_i = \Sigma_i^- \mathcal{M}_i^T \left[\mathcal{M}_i \Sigma_i^- \mathcal{M}_i^T + \Sigma_{m_i}\right]^{-1}$$

$$\overline{\boldsymbol{x}}_i^+ = \overline{\boldsymbol{x}}_i^- + \mathcal{K}_i \left[\boldsymbol{y}_i - \mathcal{M}_i \overline{\boldsymbol{x}}_i^-\right]$$

$$\Sigma_i^+ = [Id - \mathcal{K}_i \mathcal{M}_i]\, \Sigma_i^-$$



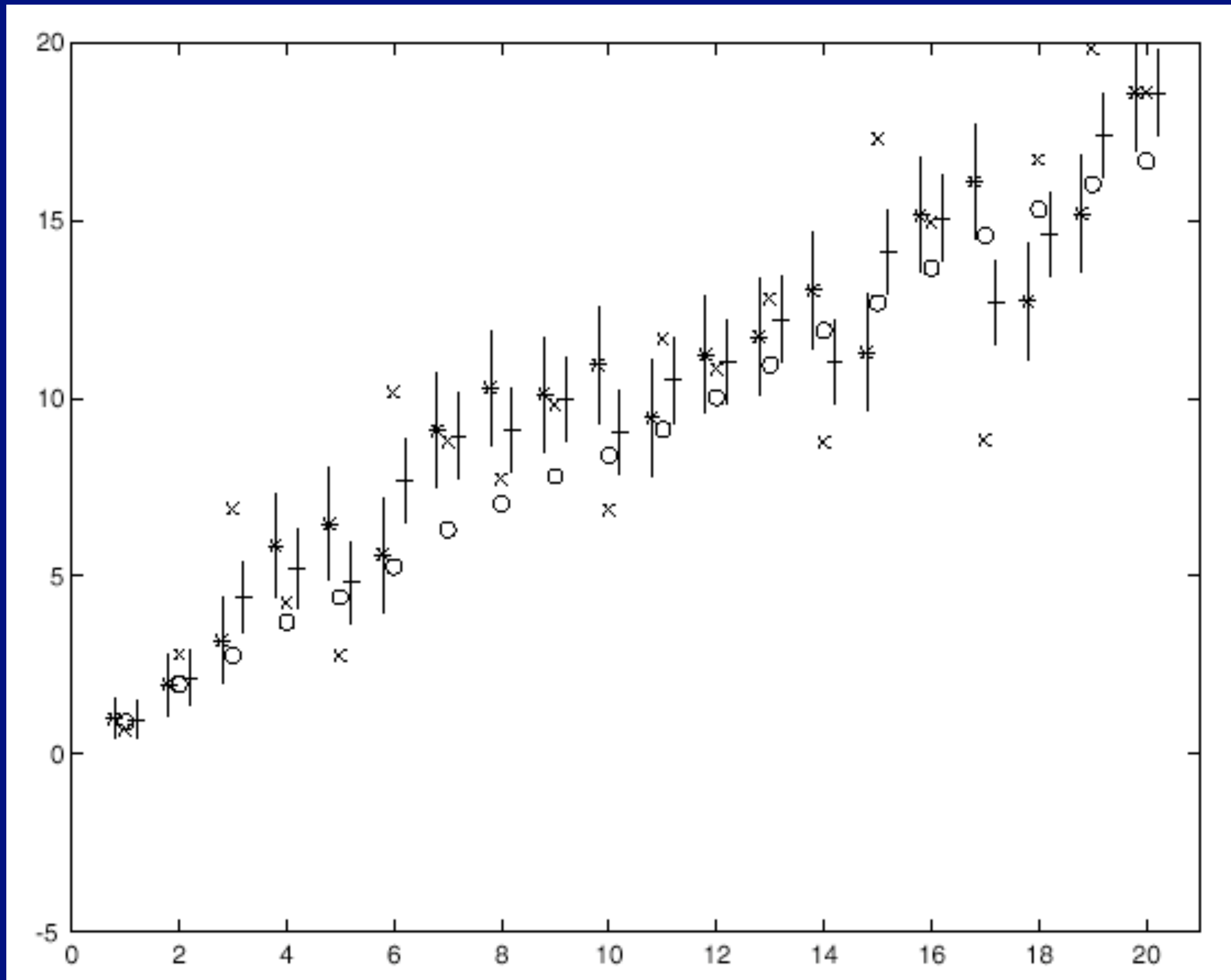
30
25
20
15
10
5
0
-5
0
5
10
15
20
25

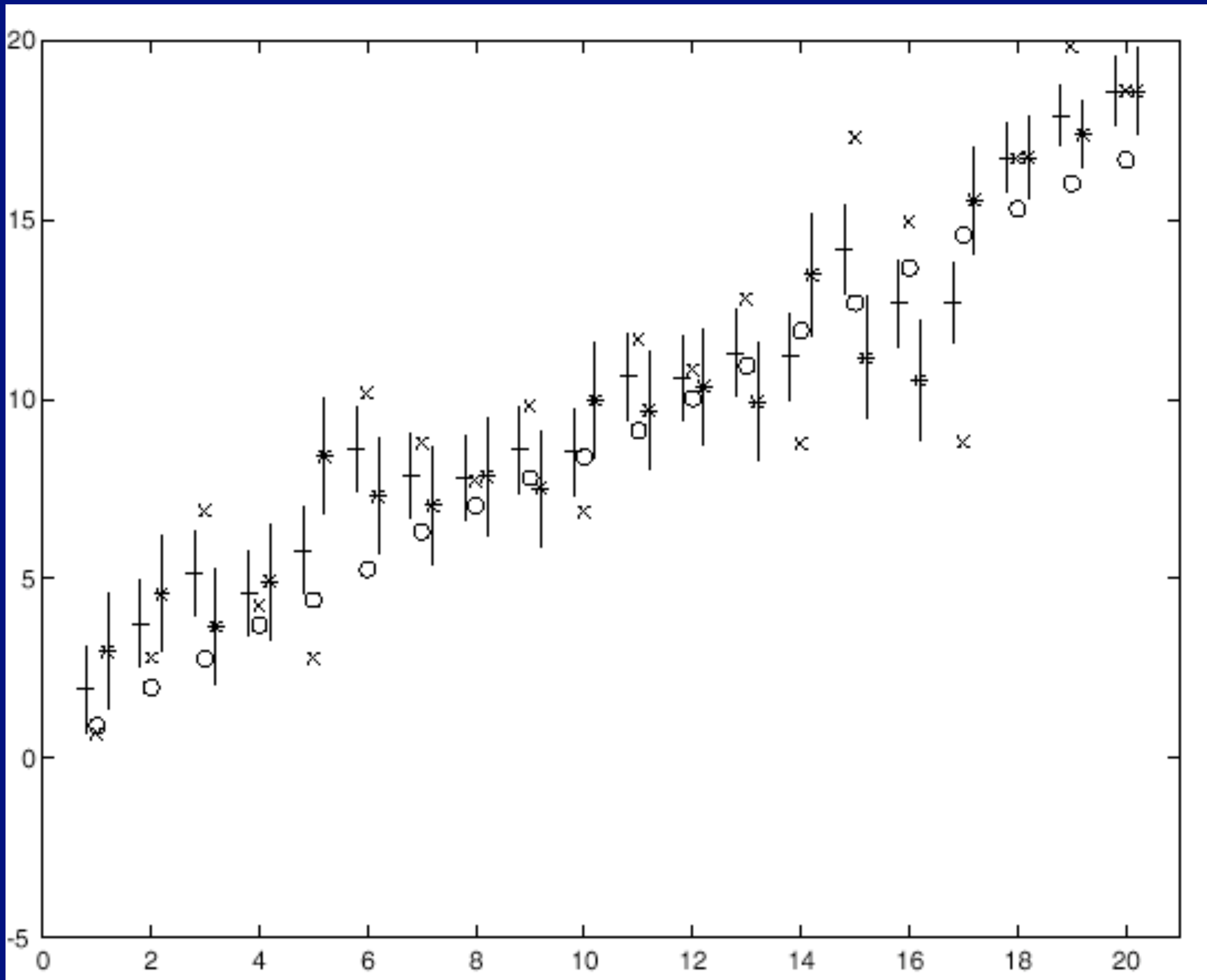


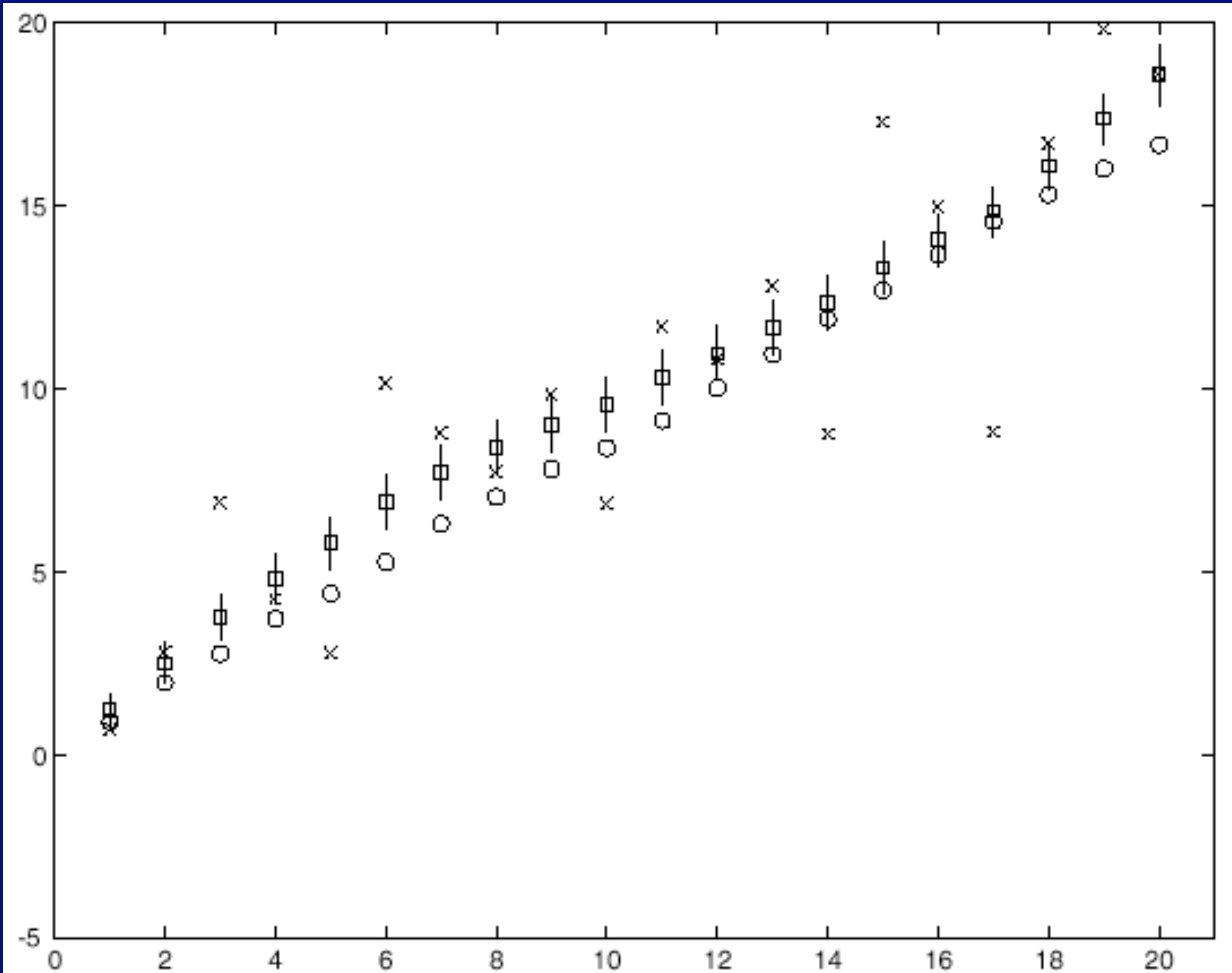
180
160
140
120
100
80
60
40
20
0
-20
0
5
10
15
20
25

# Smoothing

- Idea
  - We don’t have the best estimate of state - what about the future?
  - Run two filters, one moving forward, the other backward
  - Now combine state estimates


20
15
10
5
0
-5
0
2
4
6
8
10
12
14
16
18
20

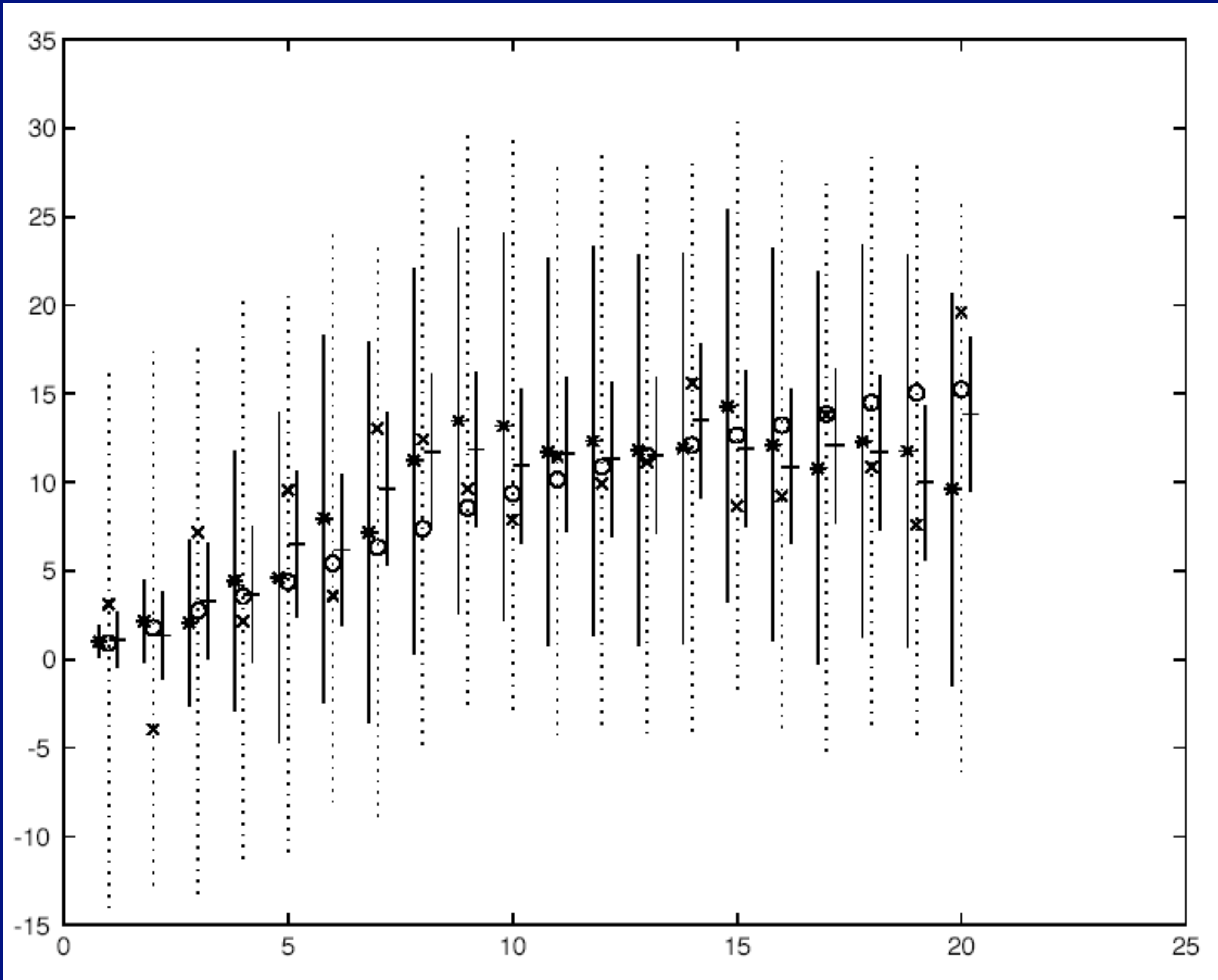

20
15
10
5
0
-5
0
2
4
6
8
10
12
14
16
18
20

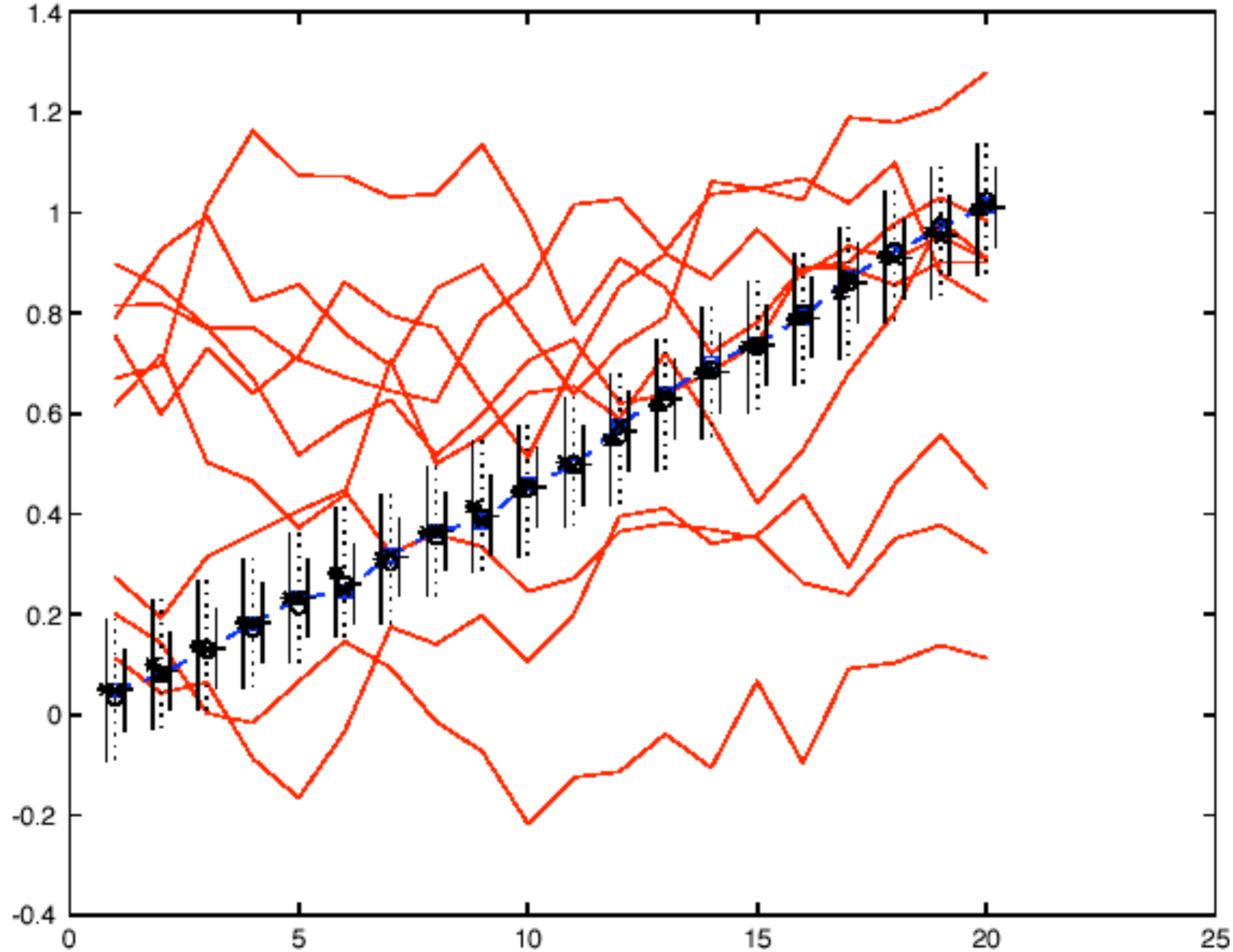


20
15
10
5
0
-5
0
2
4
6
8
10
12
14
16
18
20

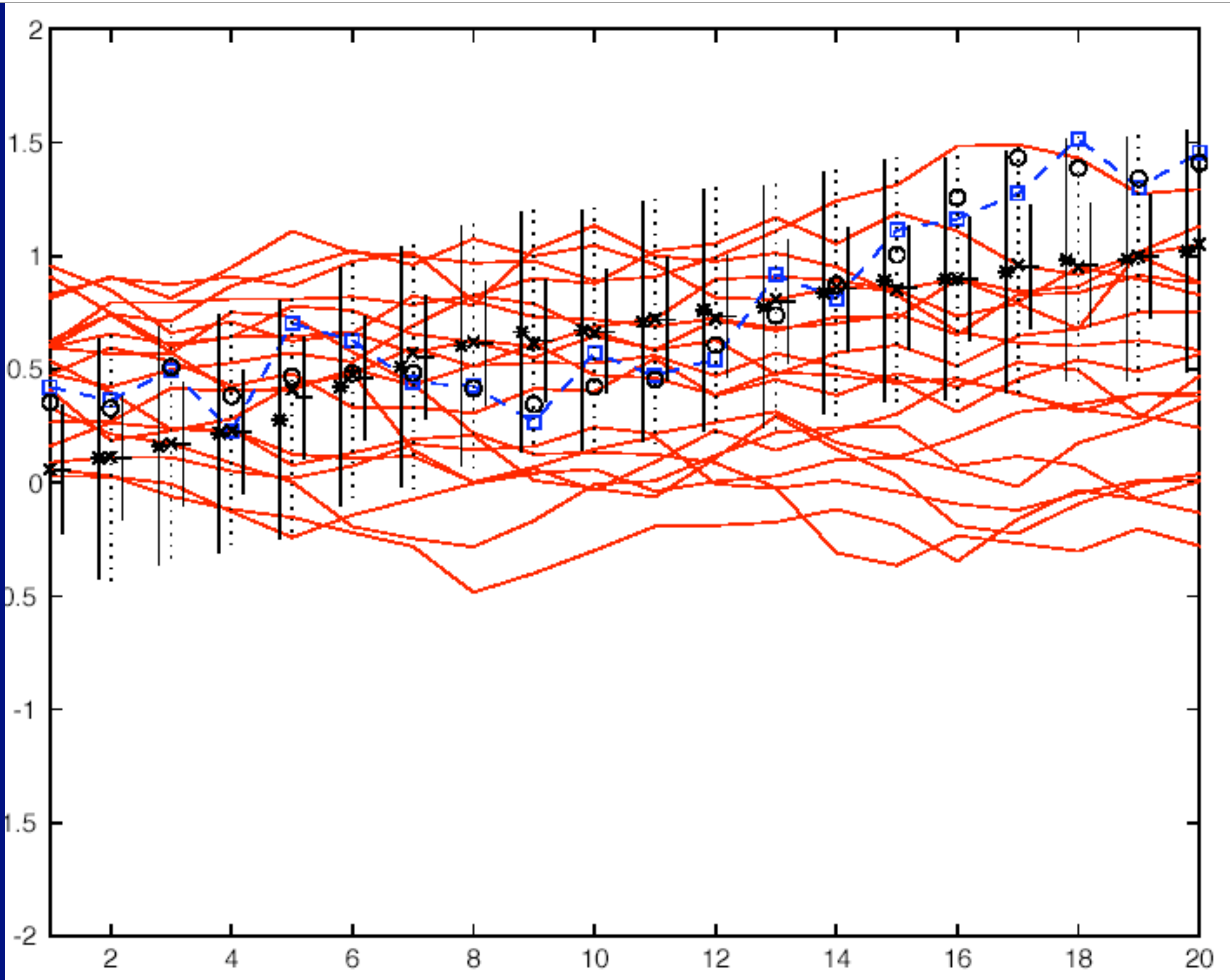
# Data Association

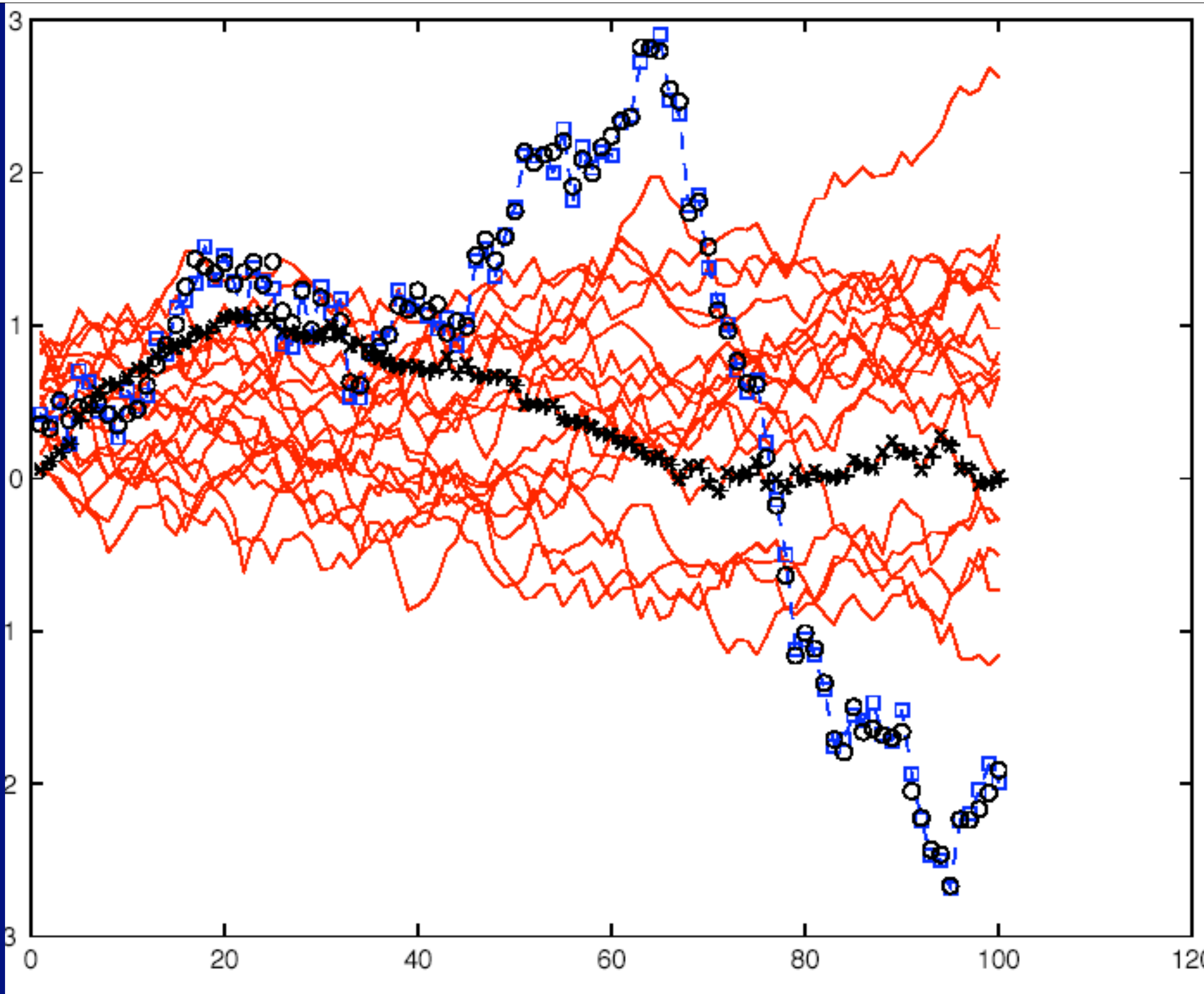
- Nearest neighbours
  - choose the measurement with highest probability given predicted state
  - popular, but can lead to catastrophe
- Probabilistic Data Association
  - combine measurements, weighting by probability given predicted state
  - gate using predicted state


35
30
25
20
15
10
5
0
-5
-10
-15
0
5
10
15
20
25



1.4
1.2
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
0
5
10
15
20
25


2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
2
4
6
8
10
12
14
16
18
20


3
2
1
0
1
2
3
0
20
40
60
80
100
120

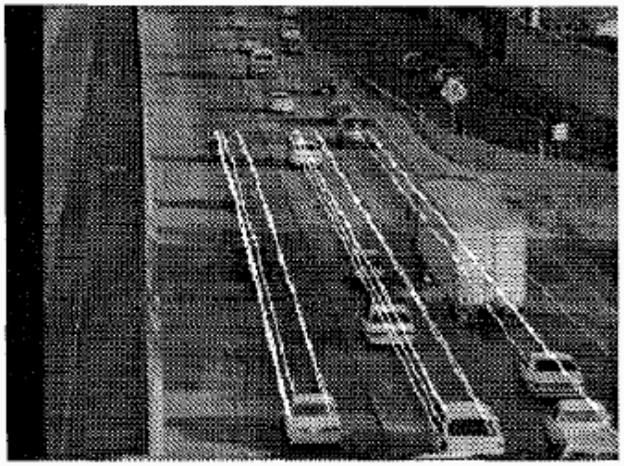


Figure 4: Example tracks of corner features.

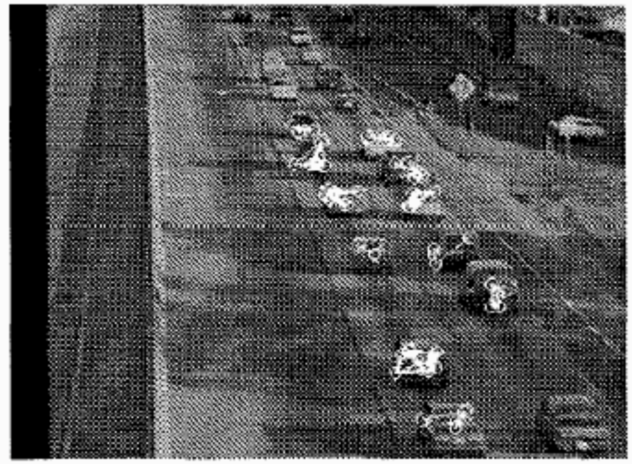
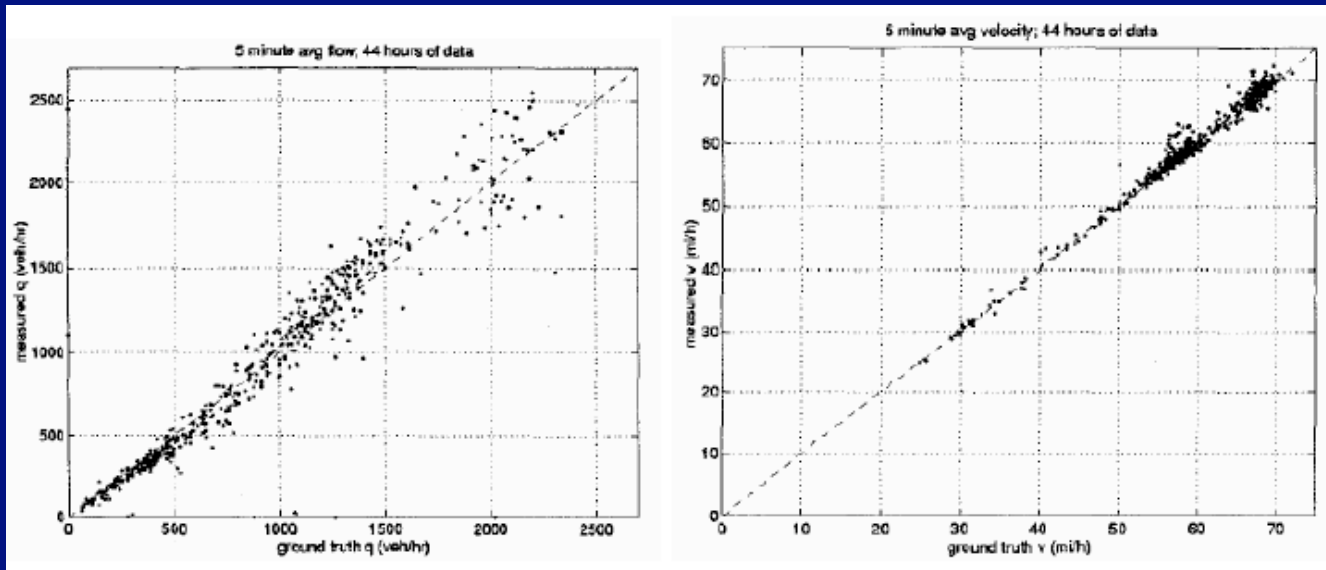


Figure 5: Example groups of corner features.

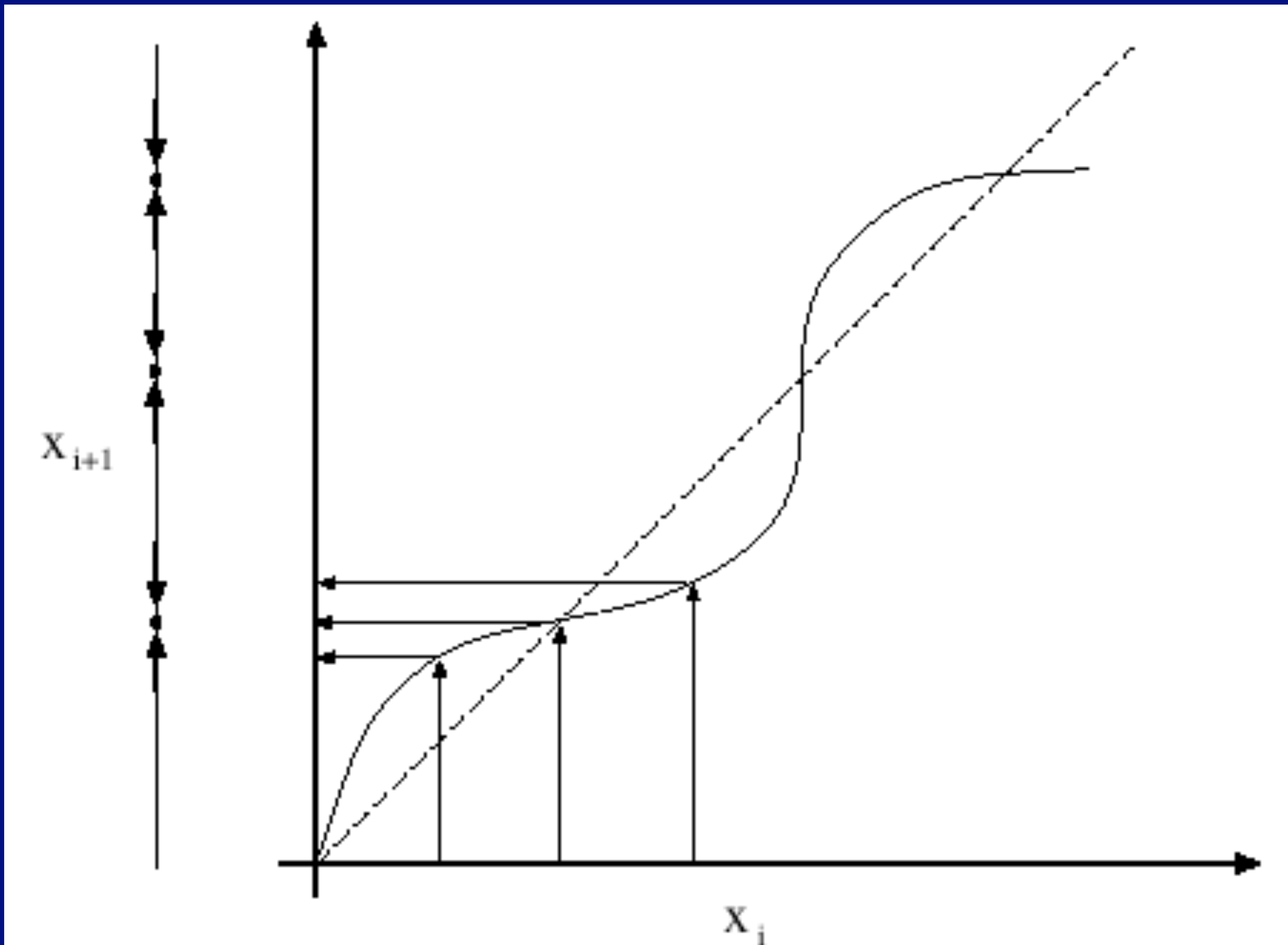
resent vehicles. *figure from A Real-Time Computer Vision System for Measuring Traffic Parameters, Beymer, McClachlan, Coifman and Malik et al. p.498, in the fervent hope of receiving permission*



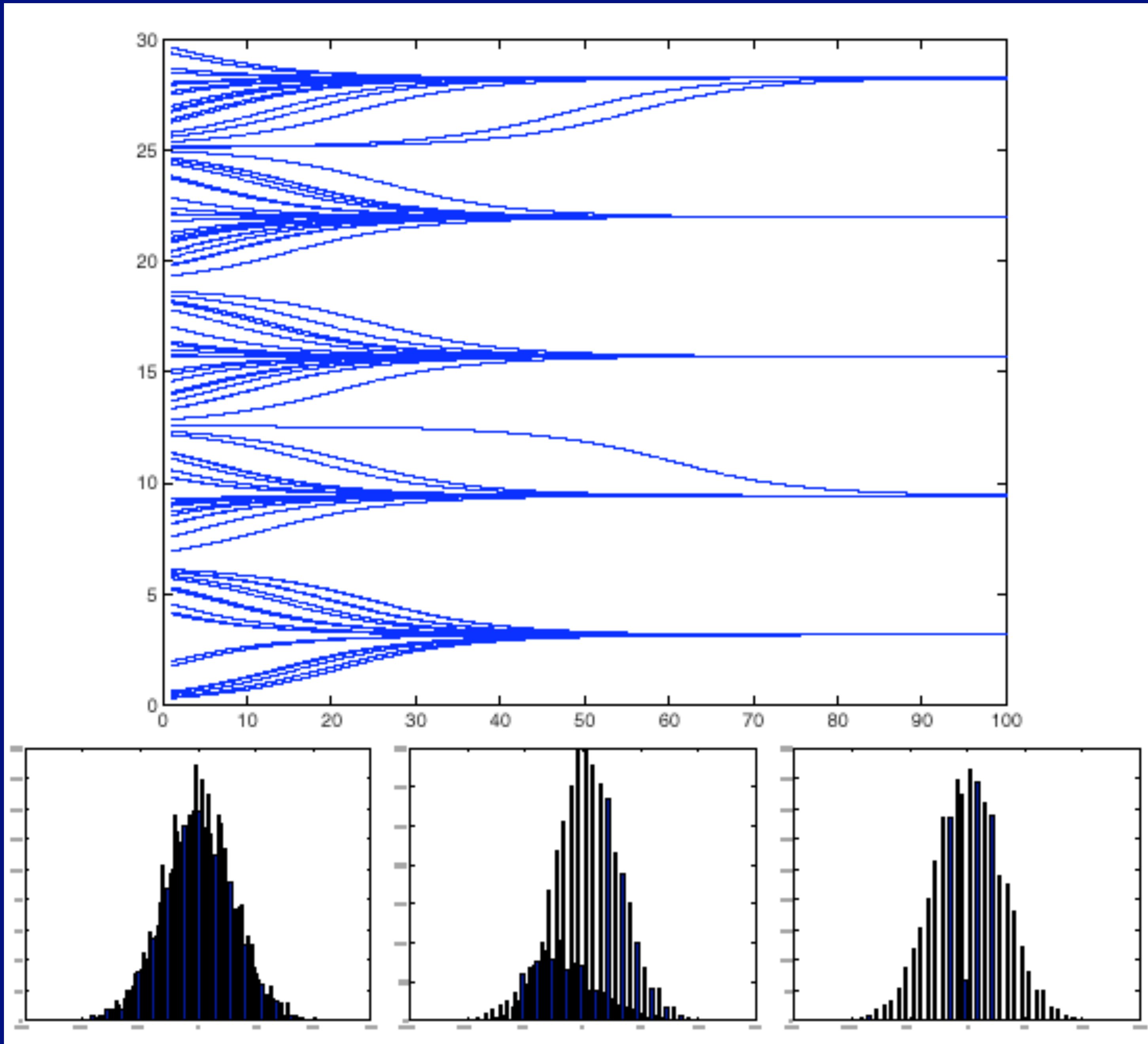

*from A Real-Time Computer Vision System for Measuring Traffic Parameters, Beymer, McClachlan, Coifman and Malik et al. p.500, in the fervent hope of receiving permission*

# Beyond the Kalman Filter

- Various phenomena lead to multiple modes
  - nonlinear dynamics
  - kinematic ambiguities
  - data association problems
- Kalman filters represent these poorly
  - alternatives
    - Mixture models
    - particle filters



$X_{i+1}$
$X_i$


30
25
20
15
10
5
0
0
10
20
30
40
50
60
70
80
90
100

# Multiple Modes from Data Association

- Linear dynamics, Linear measurement, two measurements
  - Both Gaussian, one depends on state and other doesn't
  - Not known which depends on state
- One hidden variable per frame
- Leads to 2^(number of frames) mixture of Gaussians