Particle Filtering aka Condensation, Sequential Monte Carlo (SMC), ...

Tomáš Svoboda, svoboda@cmp.felk.cvut.cz
Czech Technical University in Prague, Center for Machine Perception
http://cmp.felk.cvut.cz

Last update: April 12, 2010

- density propagation
- importance sampling
- efficient 3D head tracking by particle filter
- 2D tracking

What is tracking?

- At a certain time we need decide about one state (position) of the target object.
- Inner state representation can be arbitrary.
- Let represent the state of the object by probability density.
- Representing of the probability density by particles is one of the effective choices.

What is tracking?

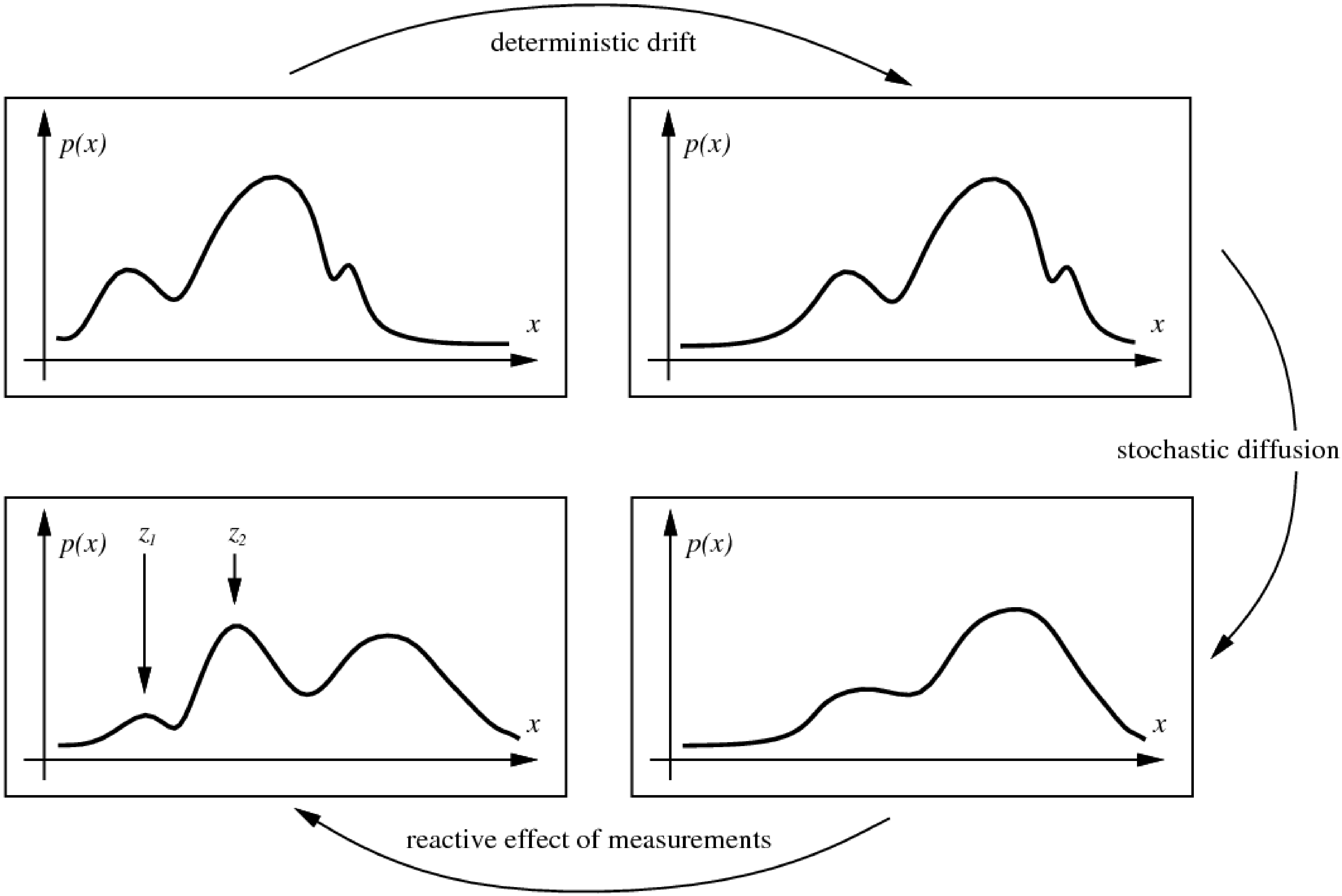
- At a certain time we need decide about one state (position) of the target object.
- Inner state representation can be arbitrary.
- Let represent the state of the object by probability density.
- Representing of the probability density by particles is one of the effective choices.

Particle filter: Particles a the input, measurements, update, . . . , particles at the output.

Particle filter in computer vision

- technique known outside computer vision for long
- popularized under the acronym CONDENSATION in 1996 [4]
- CONDENSATION stands for CONditional DENSity propagATION
- simple, easy to implement, robust . . .
- frequently used in many algorithms
- comprehensive overview [2]

Density propagation



[1]

[1]Figure from [1]

Particle filtering

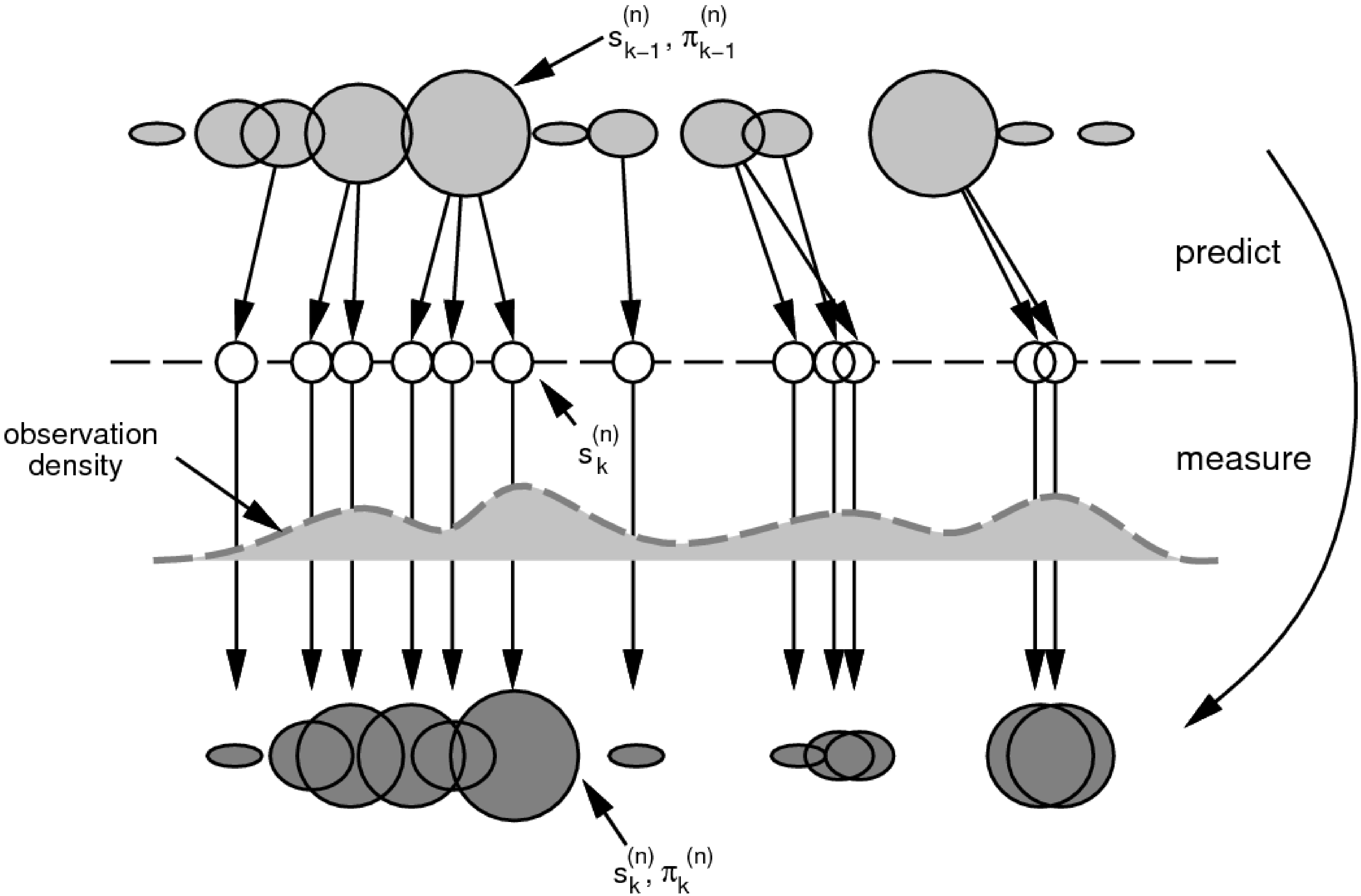
Input: $S_{t-1} = \left\{ (\mathbf{s}_{(t-1)i}, \pi_{(t-1)i}) \right\} , \quad i = 1, 2, \ldots, N \, .$

Output: S_t and object state (position) if required

Workflow for time t

1. Resample data S_{t-1} by using importance sampling.
2. Predict $\tilde{\mathbf{s}}_{(t)i}$, think about position and velocity model.
3. Uncertainty in the state change $\longrightarrow$ noisify the predicted states.
4. Measure how well the predicted states fit the observation, and update weights $\boldsymbol{\pi}_t$.
5. If needed compute the mean state (where is the target, actually?).
6. Update the prediction model if used.

One condensation step



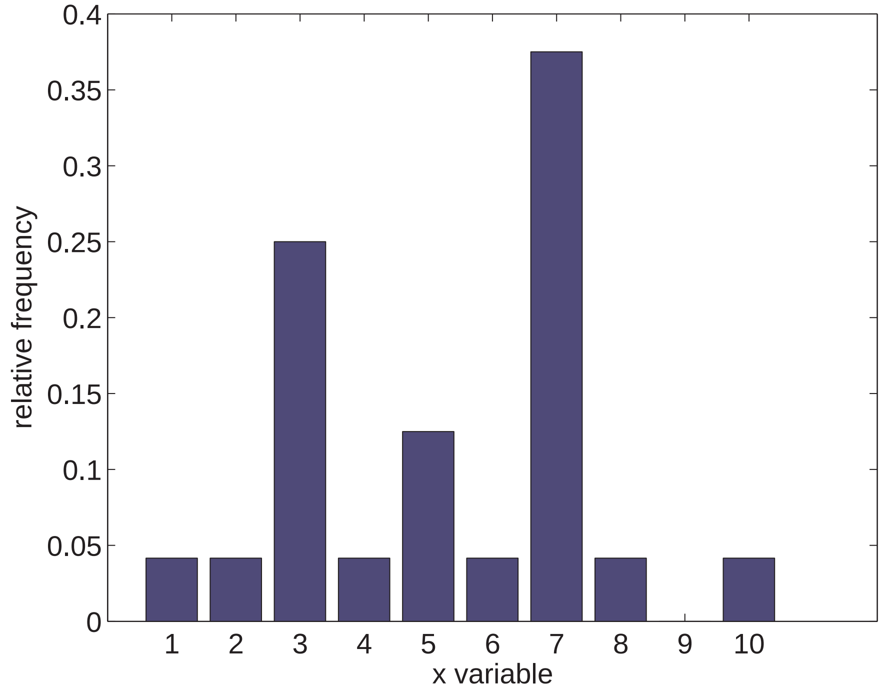
[2]

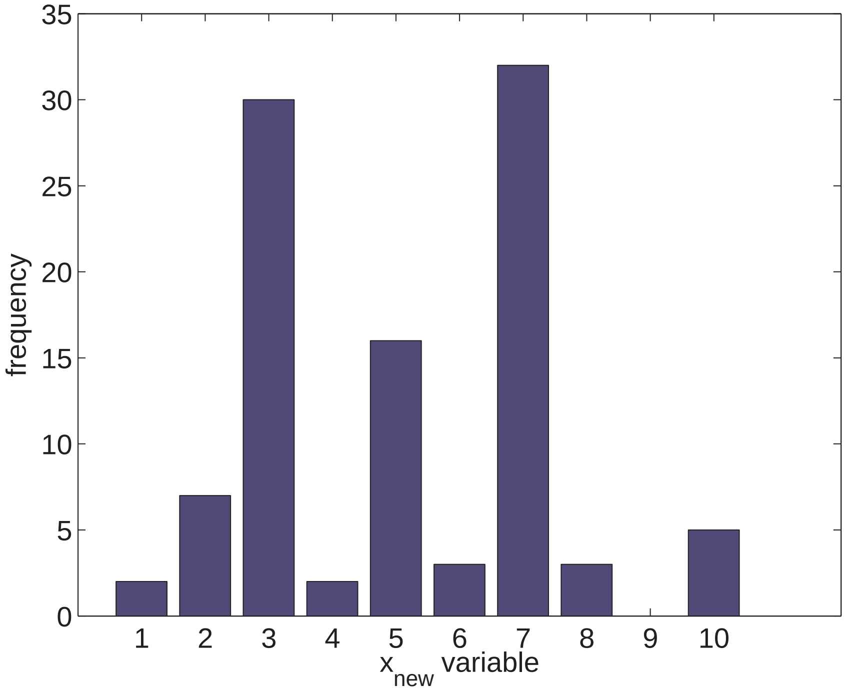
[2]Figure from [1]

Importance sampling

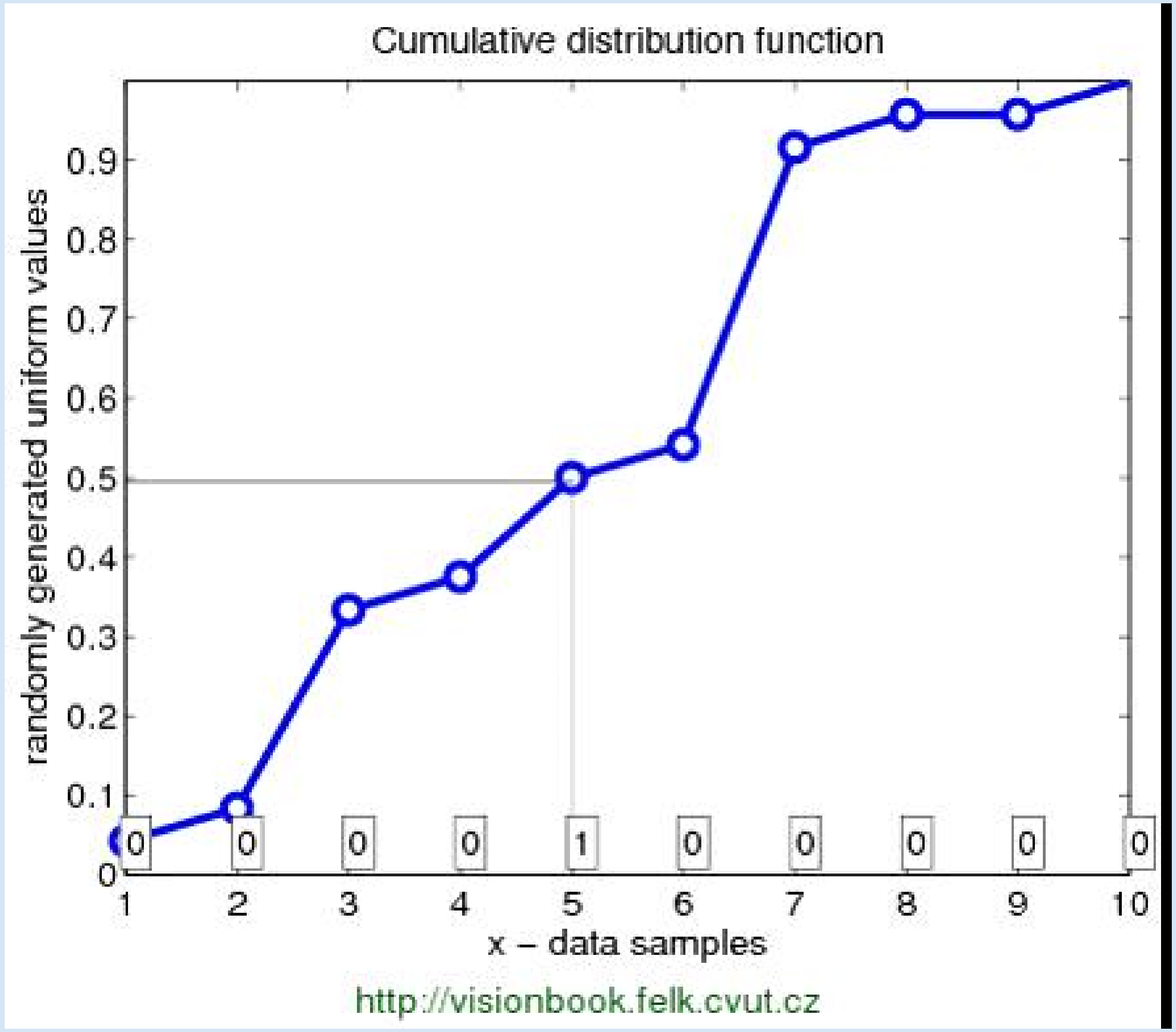
Input: set of samples with associated probabilities

Ouput: new set of samples where the frequency depends proportionally on their probabilities



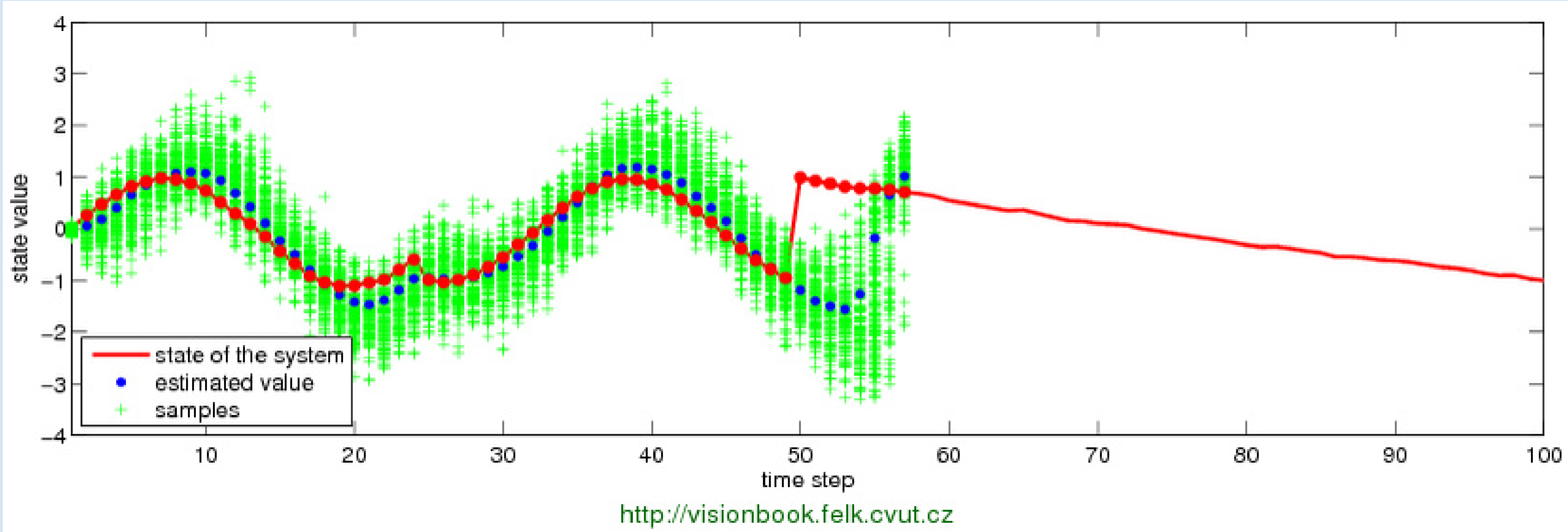


Importance sampling



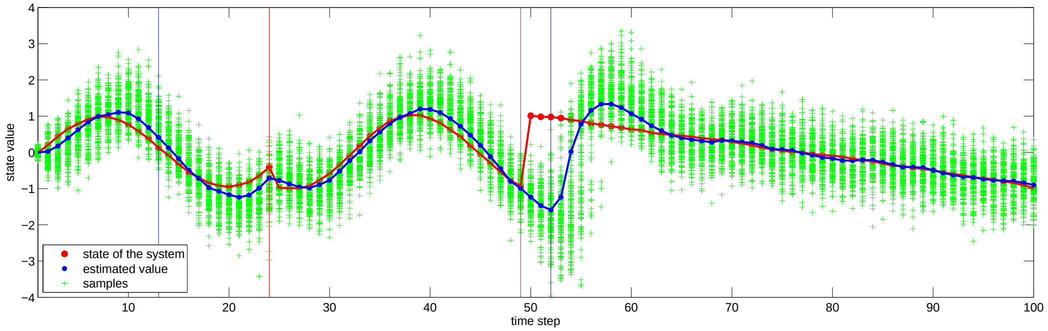
video: importance sampling

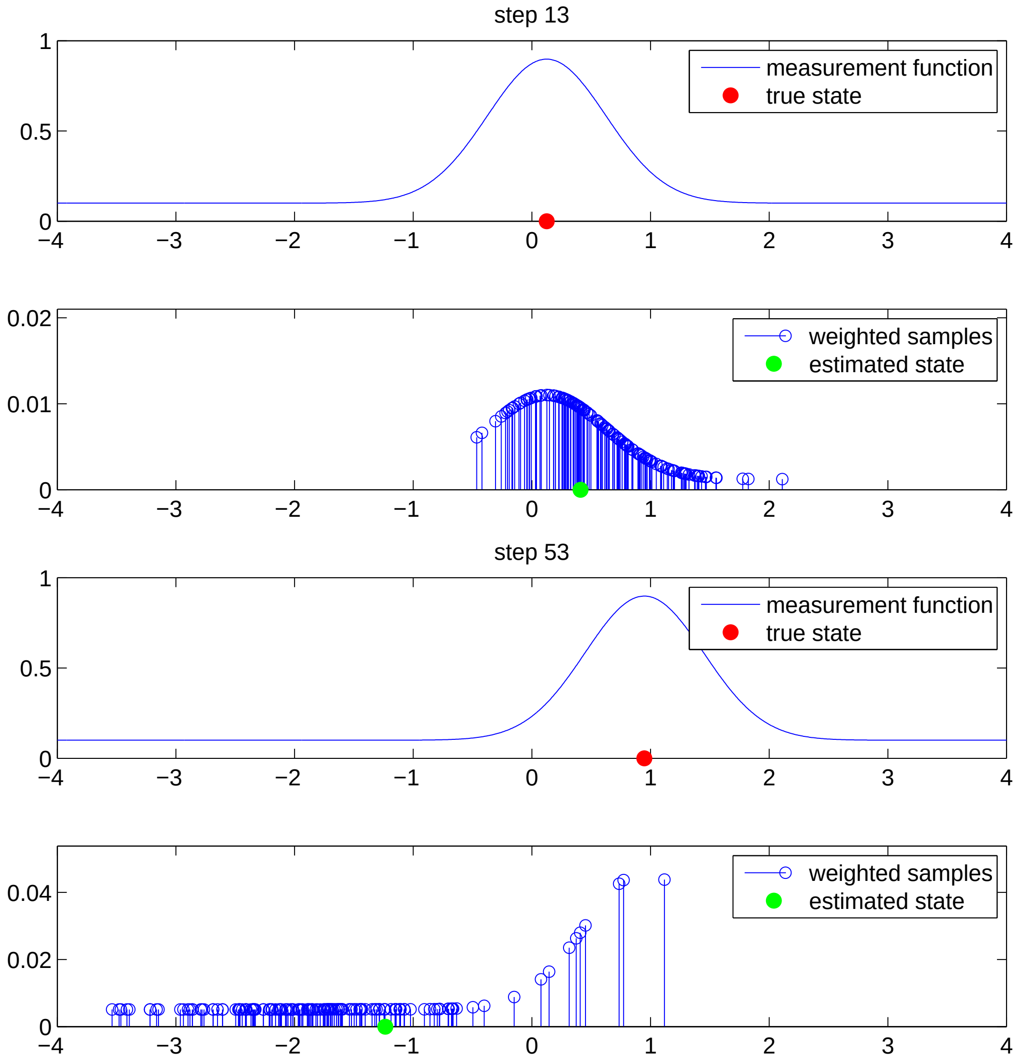
Example: 1–D tracking



video: 1–D tracking

Example: 1–D tracking, closer look



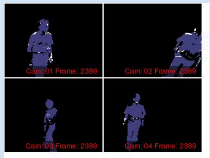


Application: 3D head tracking in multicamera system

video

3D head tracking in multicamera system—essentials

Assume calibrated system, P^j, and motion segmentated projections



video

- Head modeled as ellipsoid
- State comprises position, orientation, velocity vector . . .
- Ellipsoid project as ellipses into cameras
- We measure how far are the ellipses from contours

We will go step by step . . .

Ellipsoid and its 2D projection

Quadric surface $\mathtt{Q}$

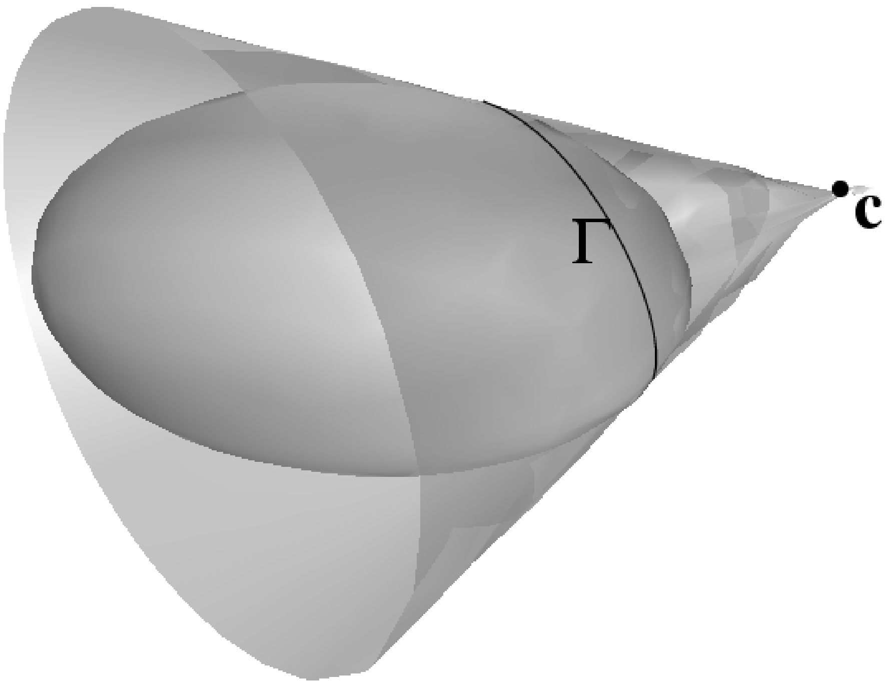
$$\mathbf{X}^\top \mathtt{Q} \mathbf{X} = 0$$

project to a (line) conic

$$\mathtt{C}^* = \mathtt{P}\mathtt{Q}^*\mathtt{P}^\top$$

point conic $\mathtt{C}$ which is dual to $\mathtt{C}^*$

$$\mathbf{u}^\top \mathtt{C} \mathbf{u} = 0$$

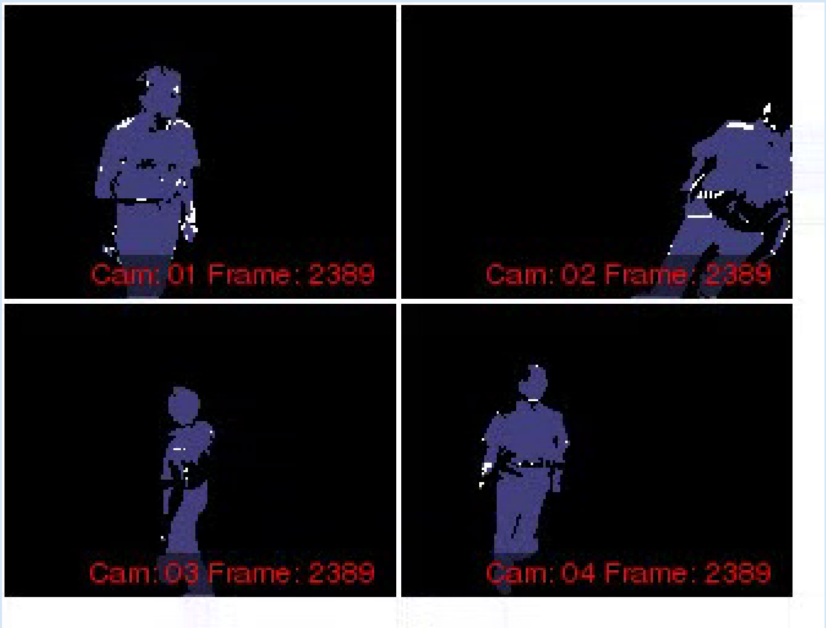


[3]

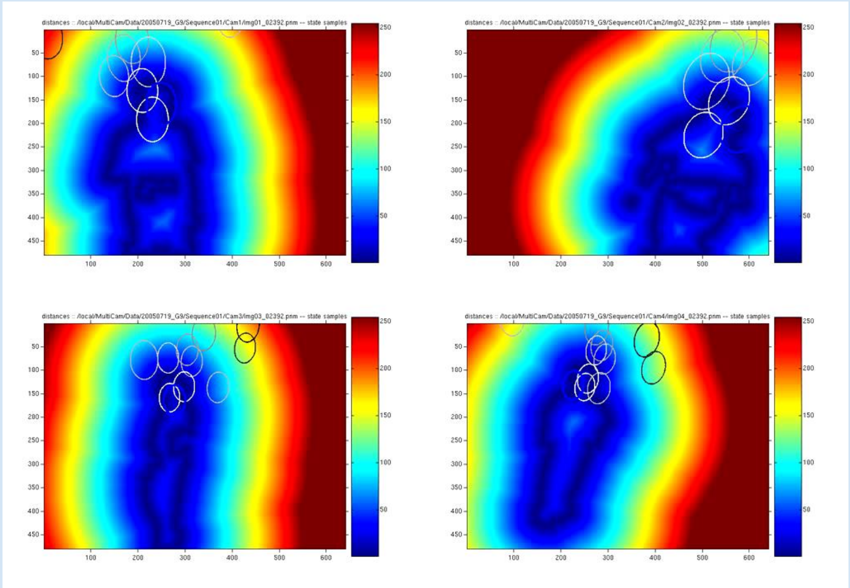
[3] Image from [3]

Measurement in (multiple) images

Remeber, we can efficiently project outline of the ellipsoid to images.



video: segmented data



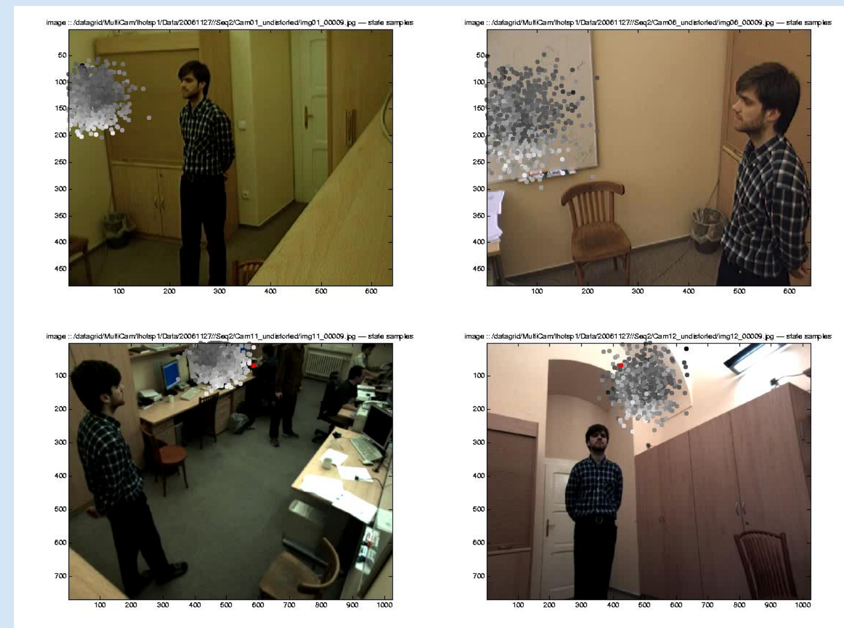
video: distance map to edges

Chamfer distance

- distance map computed just once per image
- measuring samples is just reading out values from a table

Head 3D tracking — results

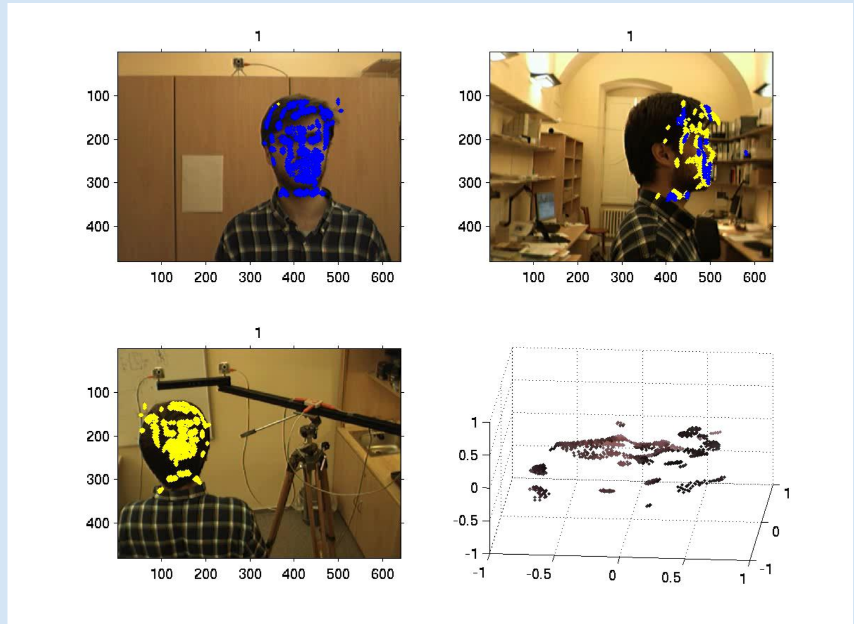
video: 3D localization results



video: example of particles convergence

Problem: 3D position only, no orientation . . .

Learning appearance

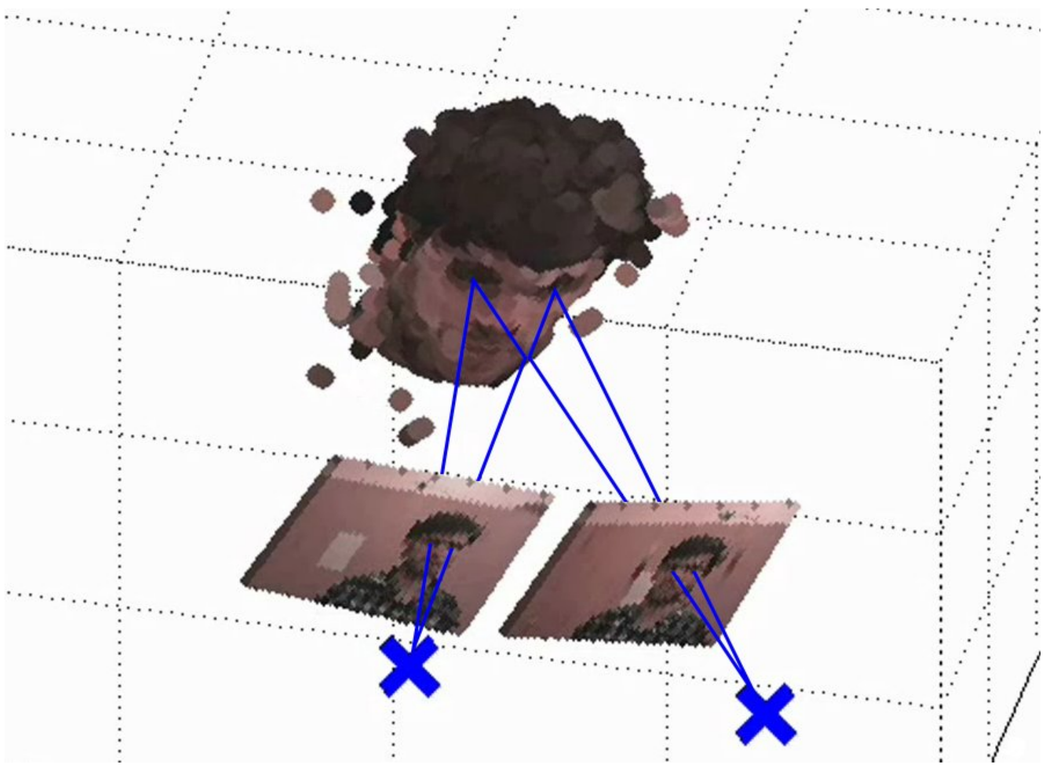


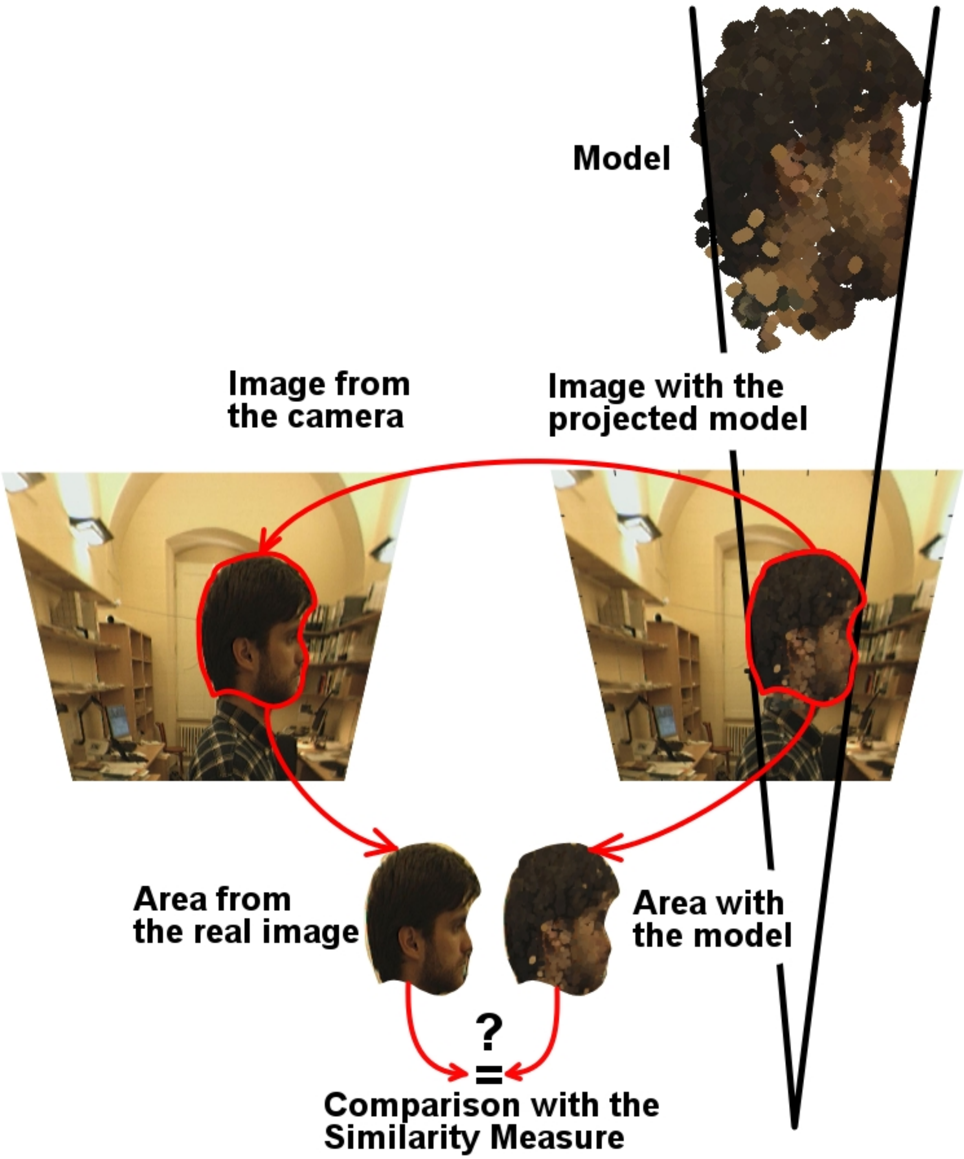
video: Learning head appearance

- Combines stereo and gradient based localization.
- Explanation of the principle [PDF; www[4]]. More in [6].

[4] http://cmp.felk.cvut.cz/projects/multicam/Demos/3Dtracking.html

3D tracking — including appearance





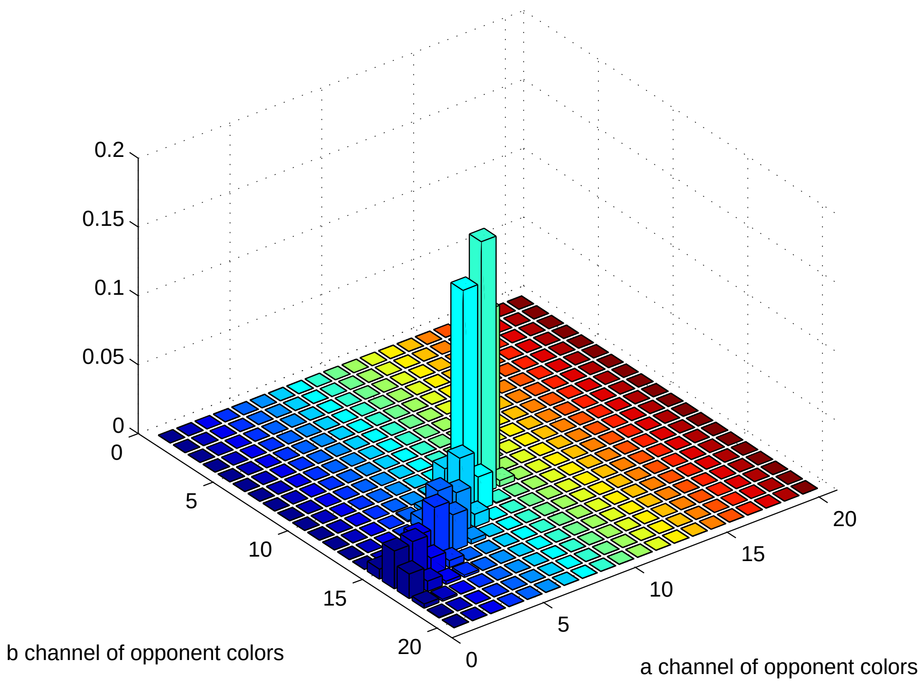
See [5] for details.

3D tracking — similarity measure

Oponent colors

$$a = \frac{1}{2}(R - G)\,, \quad b = \frac{1}{4}(2B - R - G)\,, \quad a, b \in \langle -128, 127 \rangle.$$

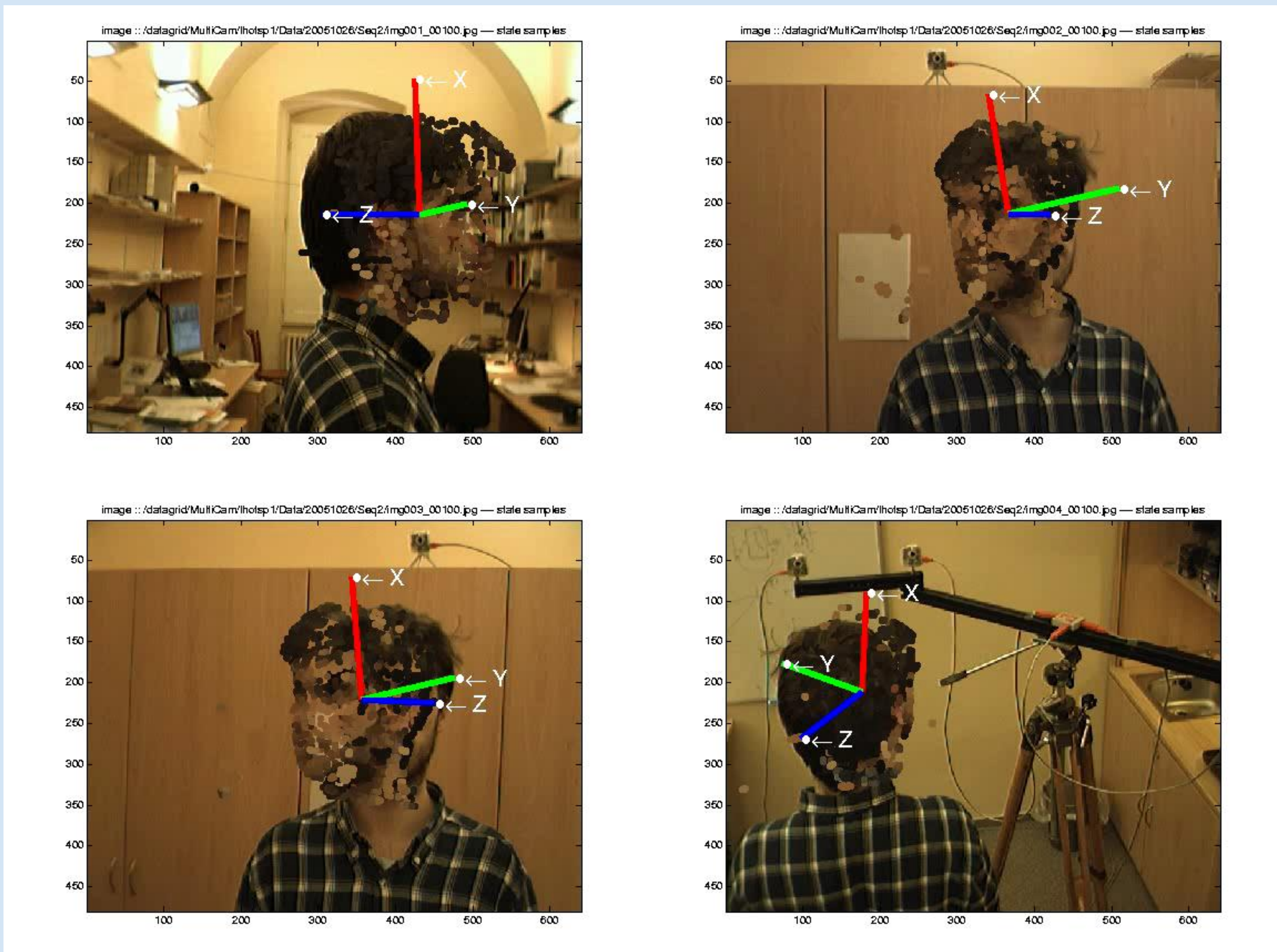
Histogram of oponent colors



Bhattacharya distance

$$\mathrm{bhatta}(\mathbf{I}, \mathbf{M}) = \sum_{k,l} \sqrt{\mathtt{I}_{k,l} \cdot \mathtt{M}_{k,l}}\,.$$

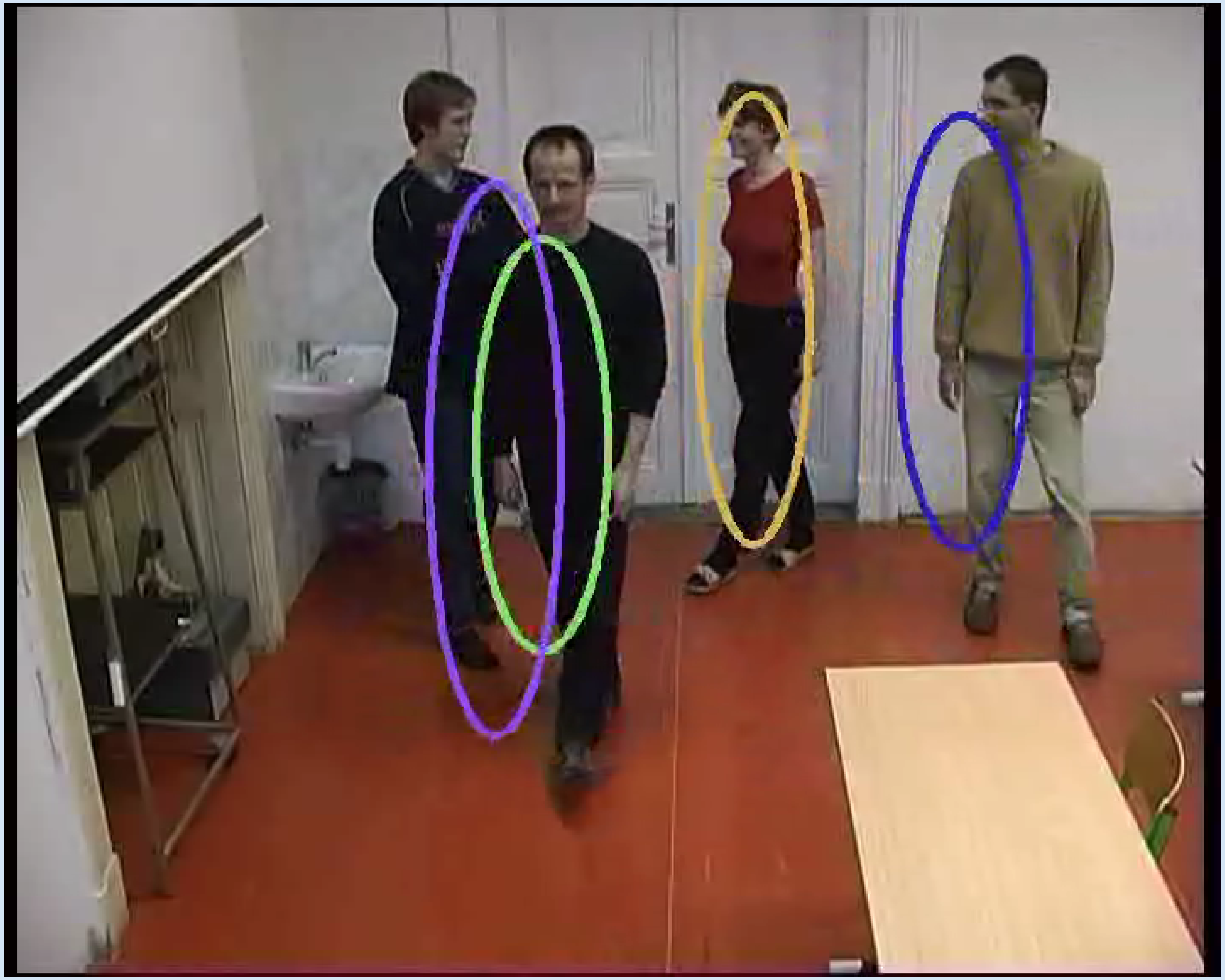
3D tracking — Results



video: 3D tracking including orientation

No post-processing, no smoothing applied.

2D tracking — object modeled by color histogram



video

References

[1] Andrew Blake and Michael Isard. Active Contours : The Application of Techniques form Graphics, Vision, Control Theory and Statistics to Visual Tracking of Shapes in Motion. Springer, London, Great Britain, 1998. On-line available at http://www.robots.ox.ac.uk/~contours/.

[2] Arnaud Doucet, Nando De Freitas, and Neil Gordon. Sequential Monte Carlo Methods in Practice. Springer, 2001.

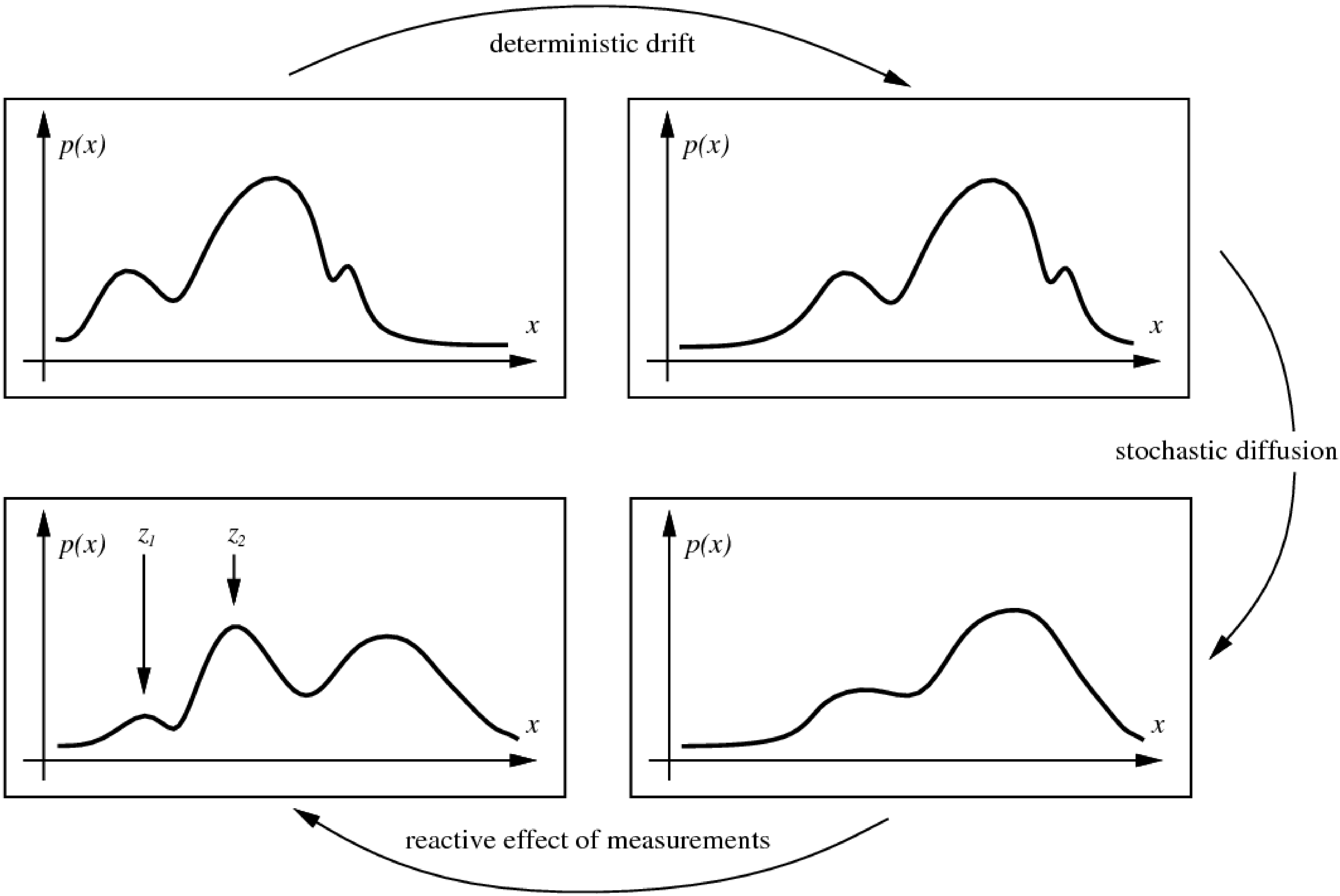
[3] R. Hartley and A. Zisserman. Multiple View Geometry in Computer Vision. Cambridge University Press, Cambridge, UK, 2000. On-line resources at: `http://www.robots.ox.ac.uk/~vgg/hzbook/hzbook1.html`.

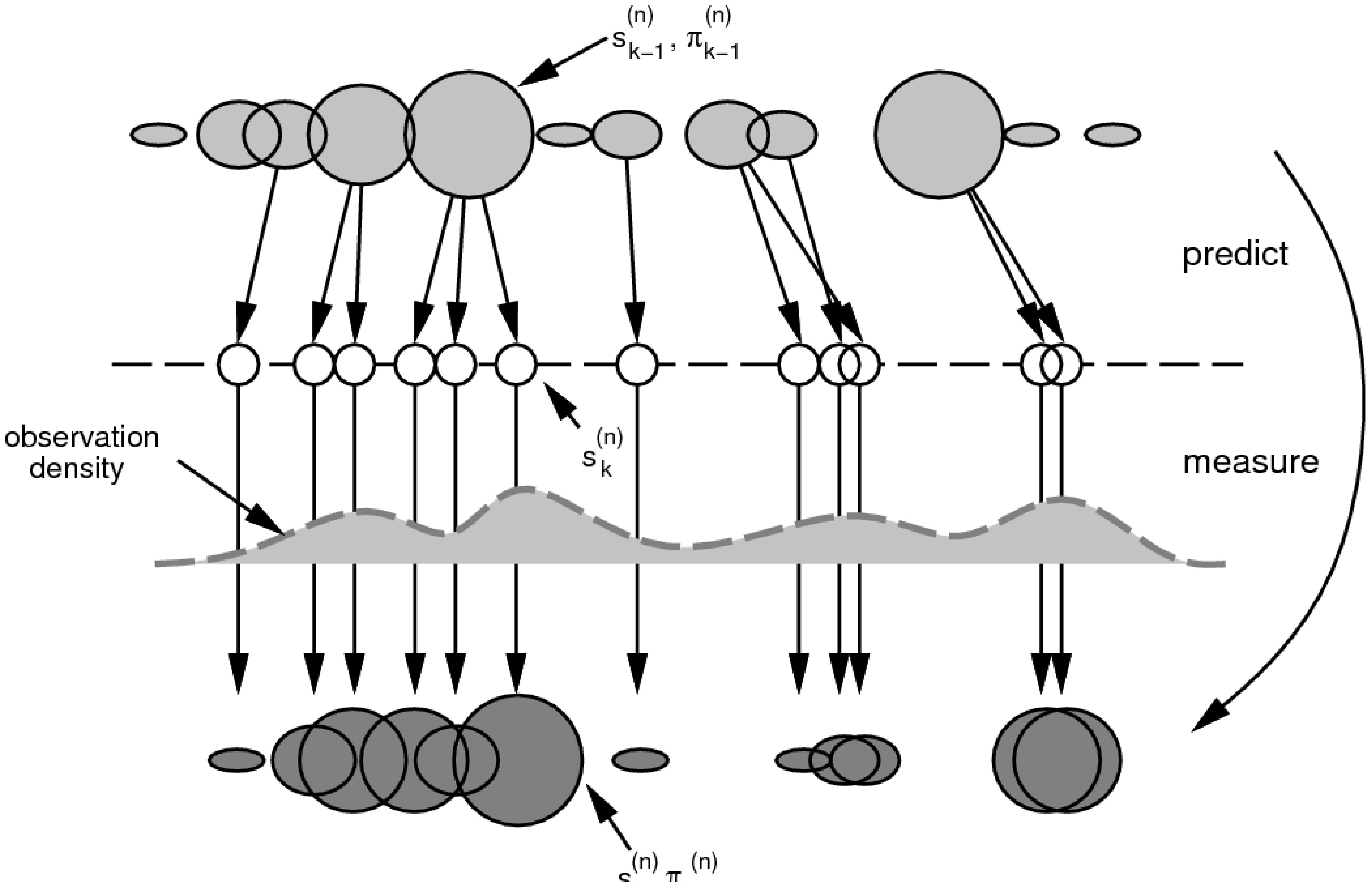
[4] Michael Isard and Blake Andrew. Contour tracking by stochastic propagation of conditional density. In Proceedings of European Conference on Computer Vision, pages 343–356, 1996. Demos, code, and more detatailed info available at `http://www.robots.ox.ac.uk/~misard/condensation.html`.

[5] Petr Lhotský. Detection and tracking objects using sequential monte carlo method. MSc Thesis K333–24/07, CTU–CMP–2007–01, Department of Cybernetics, Faculty of Electrical Engineering Czech Technical University, Prague, Czech Republic, January 2007.

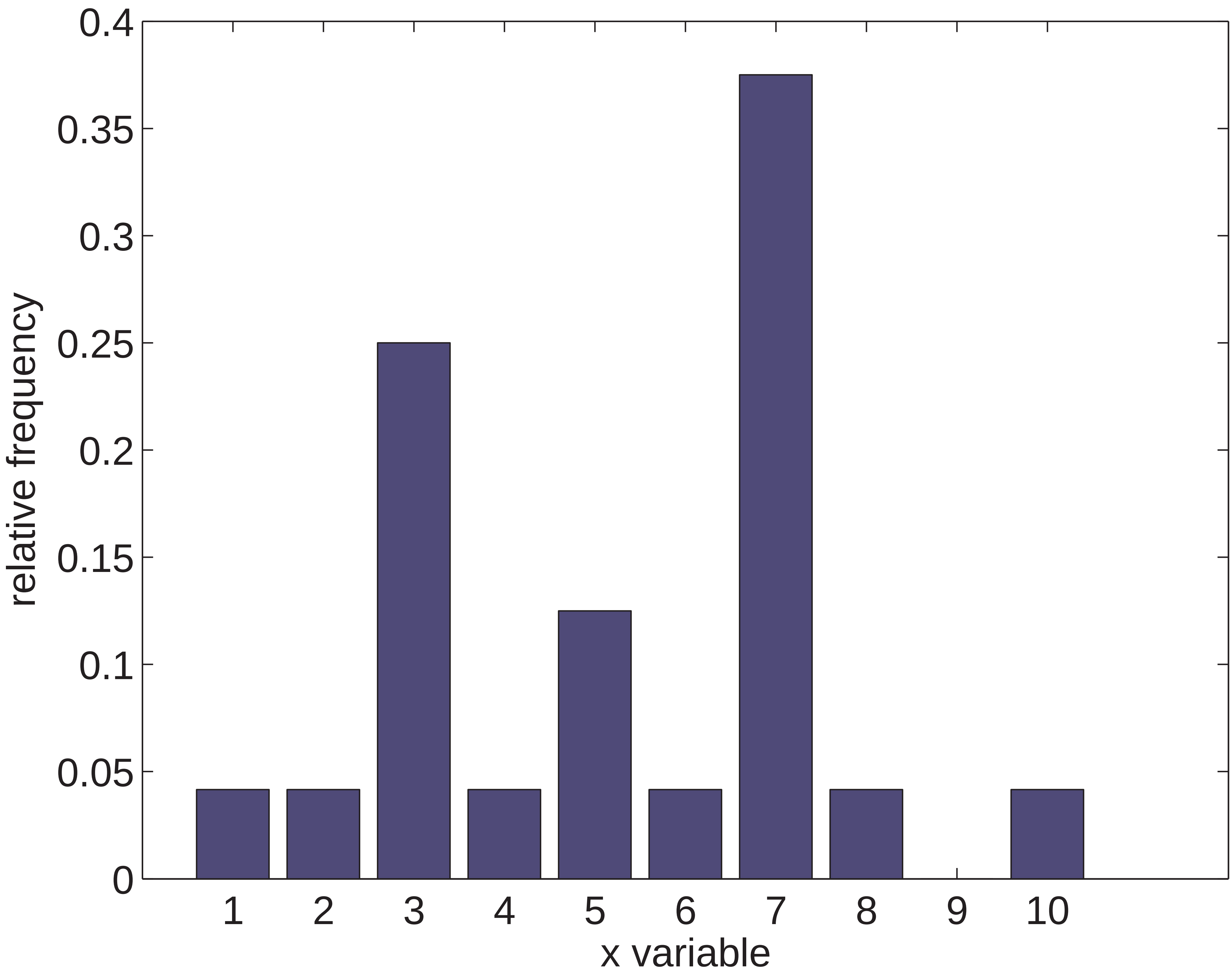
[6] Karel Zimmermann, Tomáš Svoboda, and Jiří Matas. Multiview 3D tracking with an incrementally constructed 3D model. In Third International Symposium on 3D Data Processing, Visualization and Transmission, Chapel Hill, USA, June 2006. University of North Carolina.

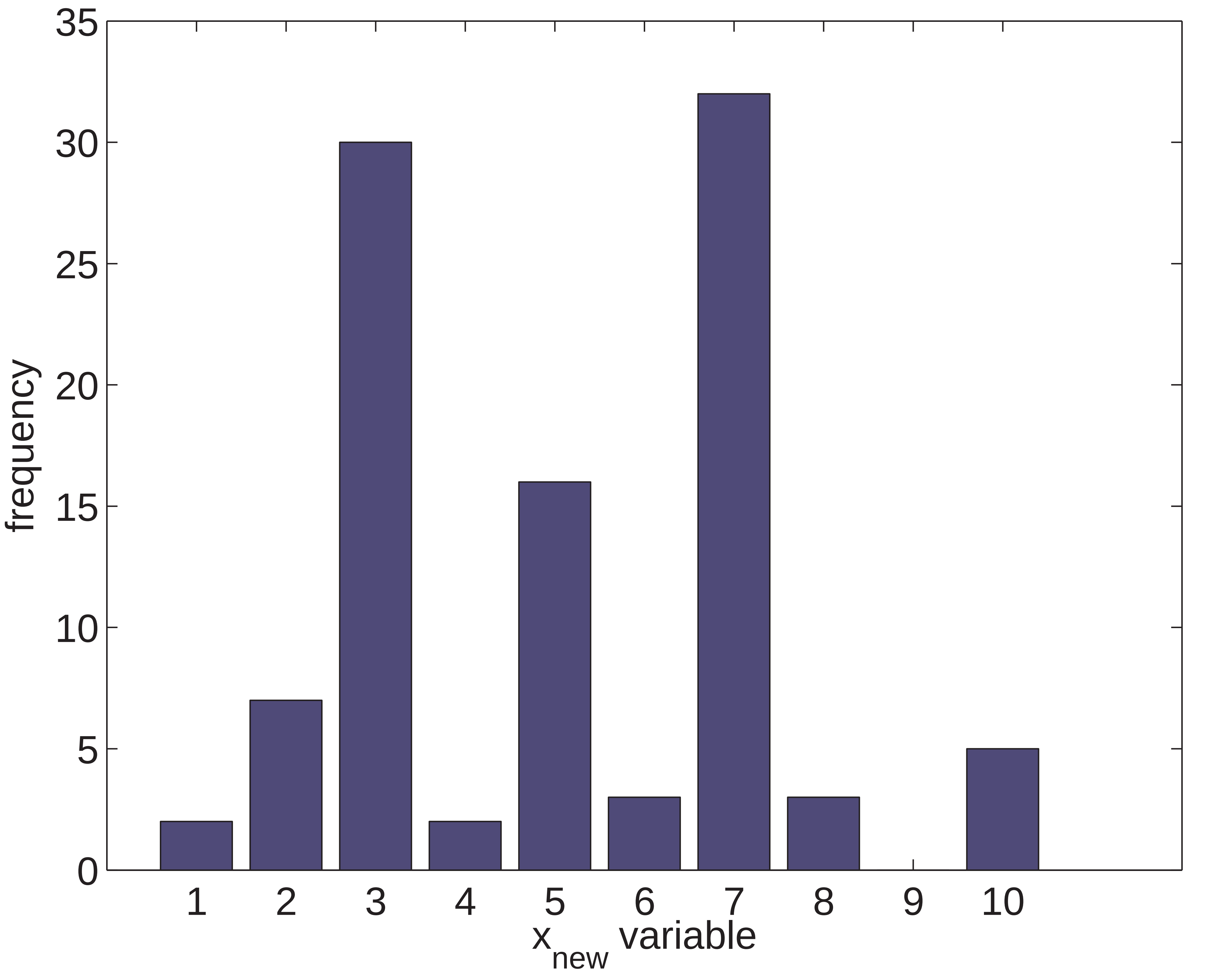
End


deterministic drift
p(x)
x
p(x)
x
stochastic diffusion
p(x)
z1
z2
x
p(x)
x
reactive effect of measurements

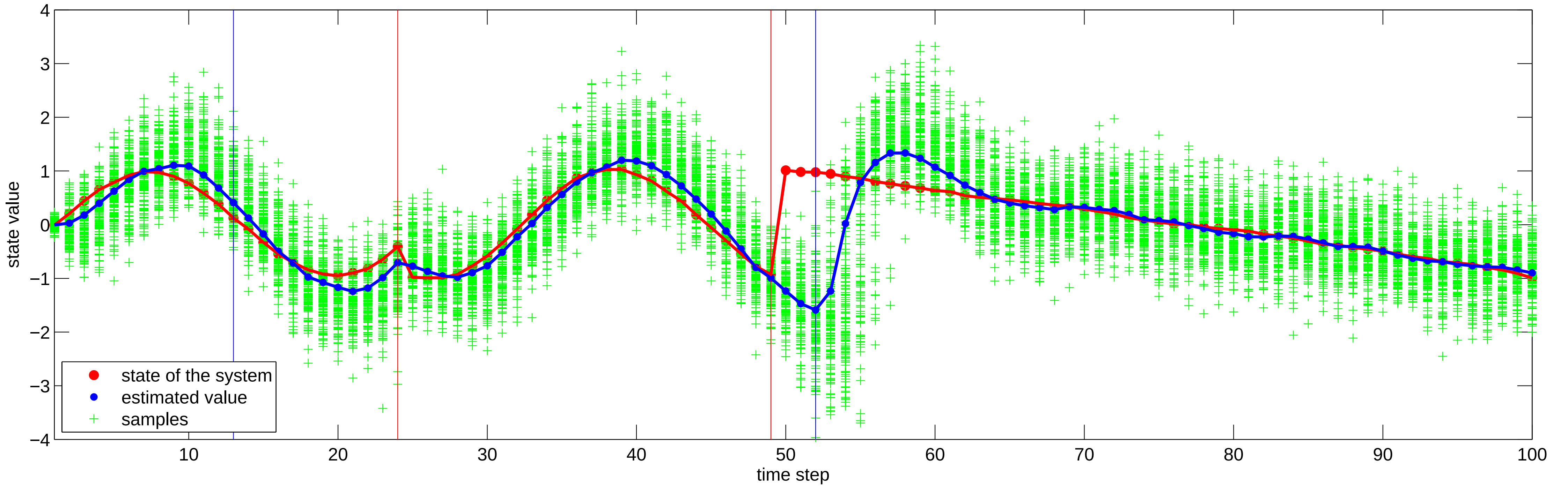


$s_{k-1}^{(n)}, \pi_{k-1}^{(n)}$
predict
$s_k^{(n)}$
observation density
measure
$s_k^{(n)}, \pi_k^{(n)}$

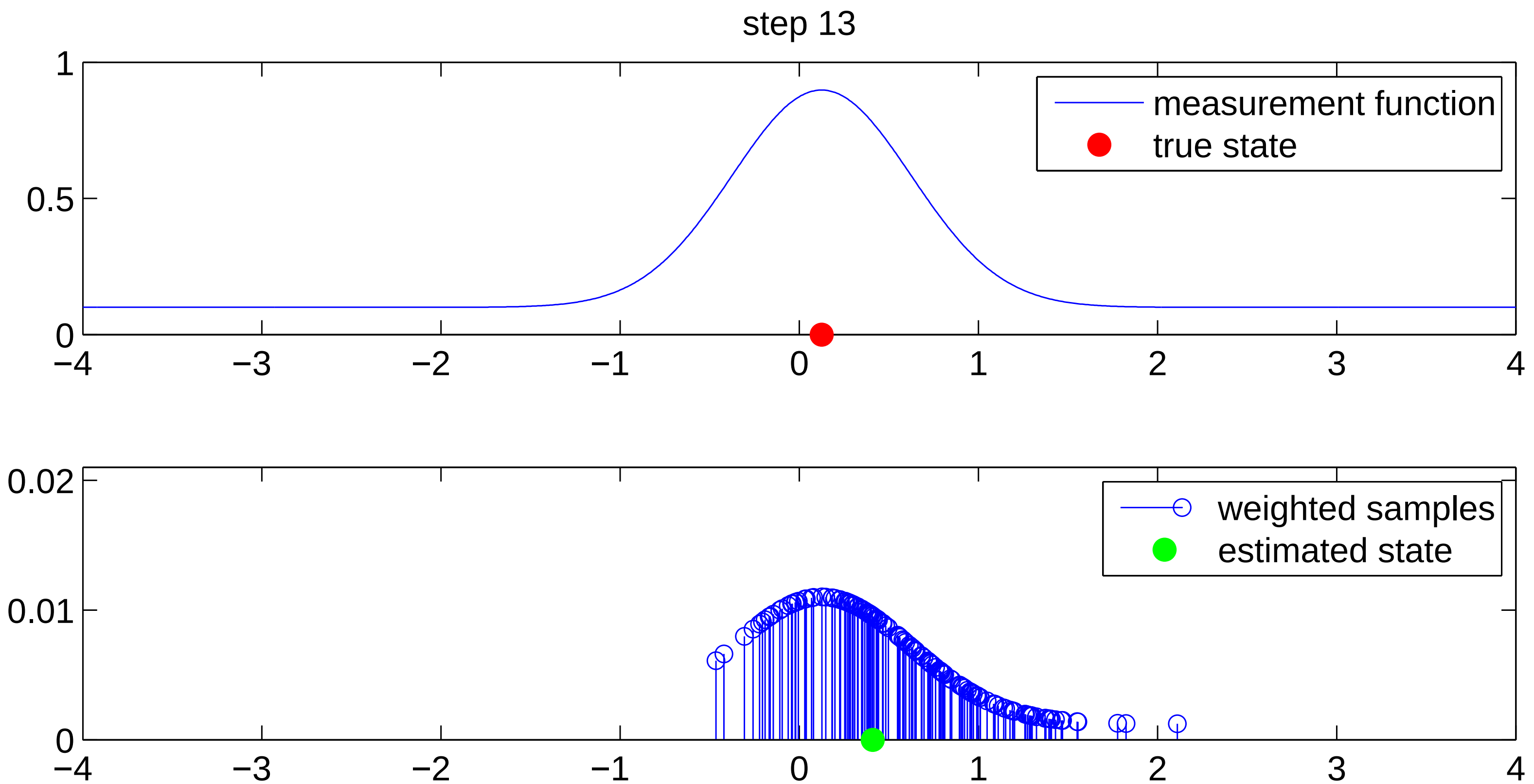

0.4
0.35
0.3
0.25
0.2
0.15
0.1
0.05
0
relative frequency
1
2
3
4
5
6
7
8
9
10
x variable

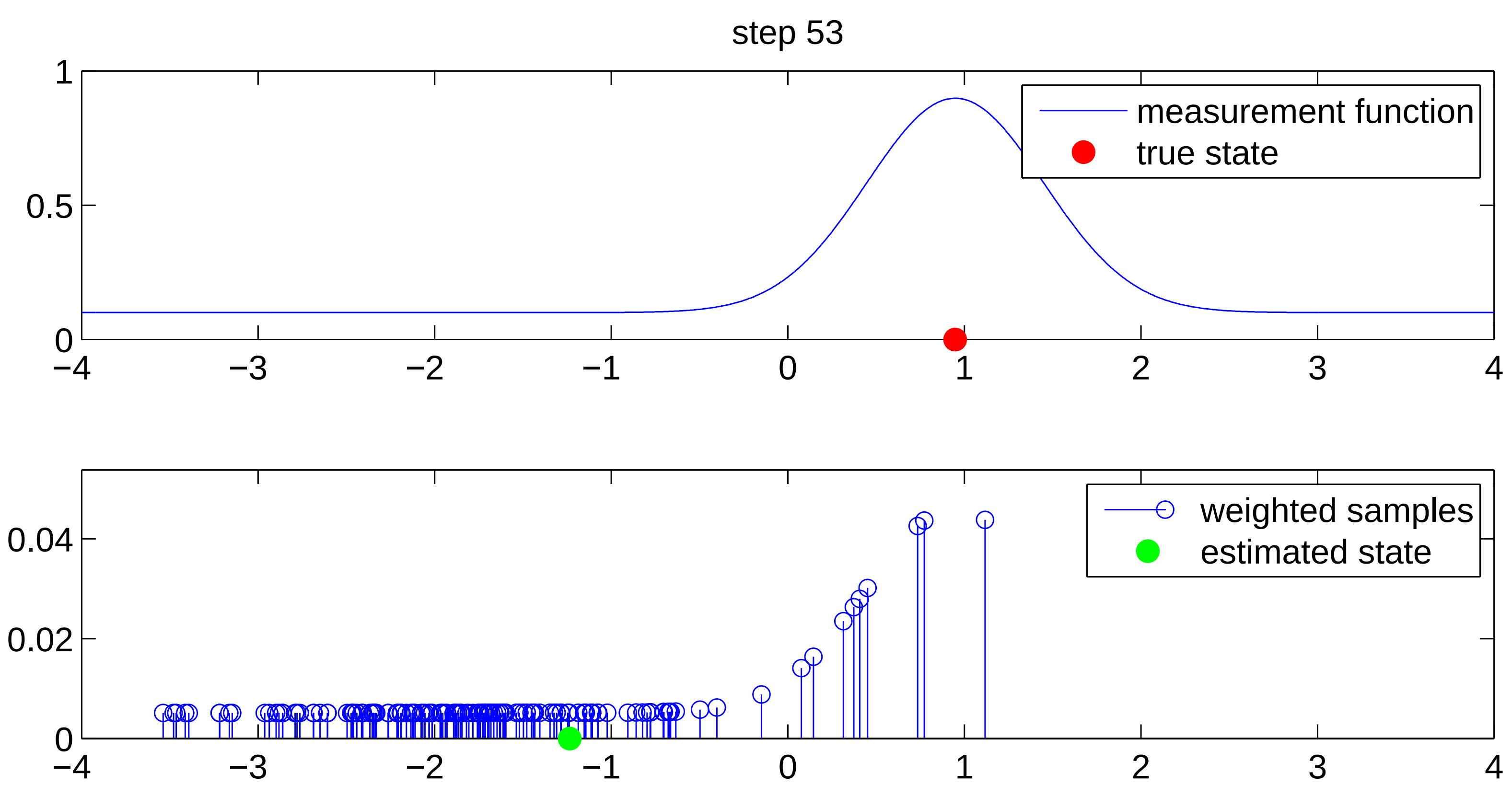


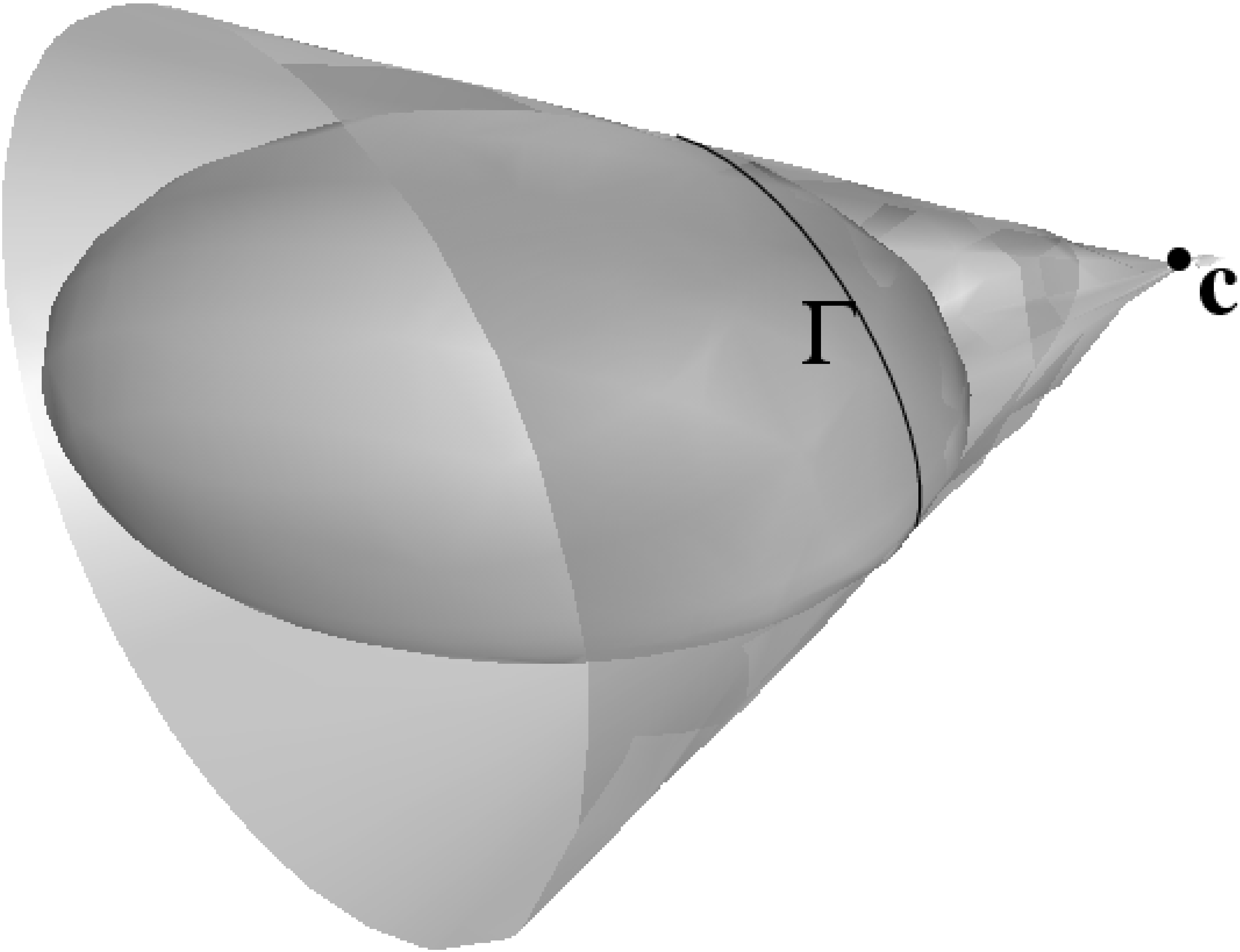
frequency
35
30
25
20
15
10
5
0
1
2
3
4
5
6
7
8
9
10
x_{new} variable

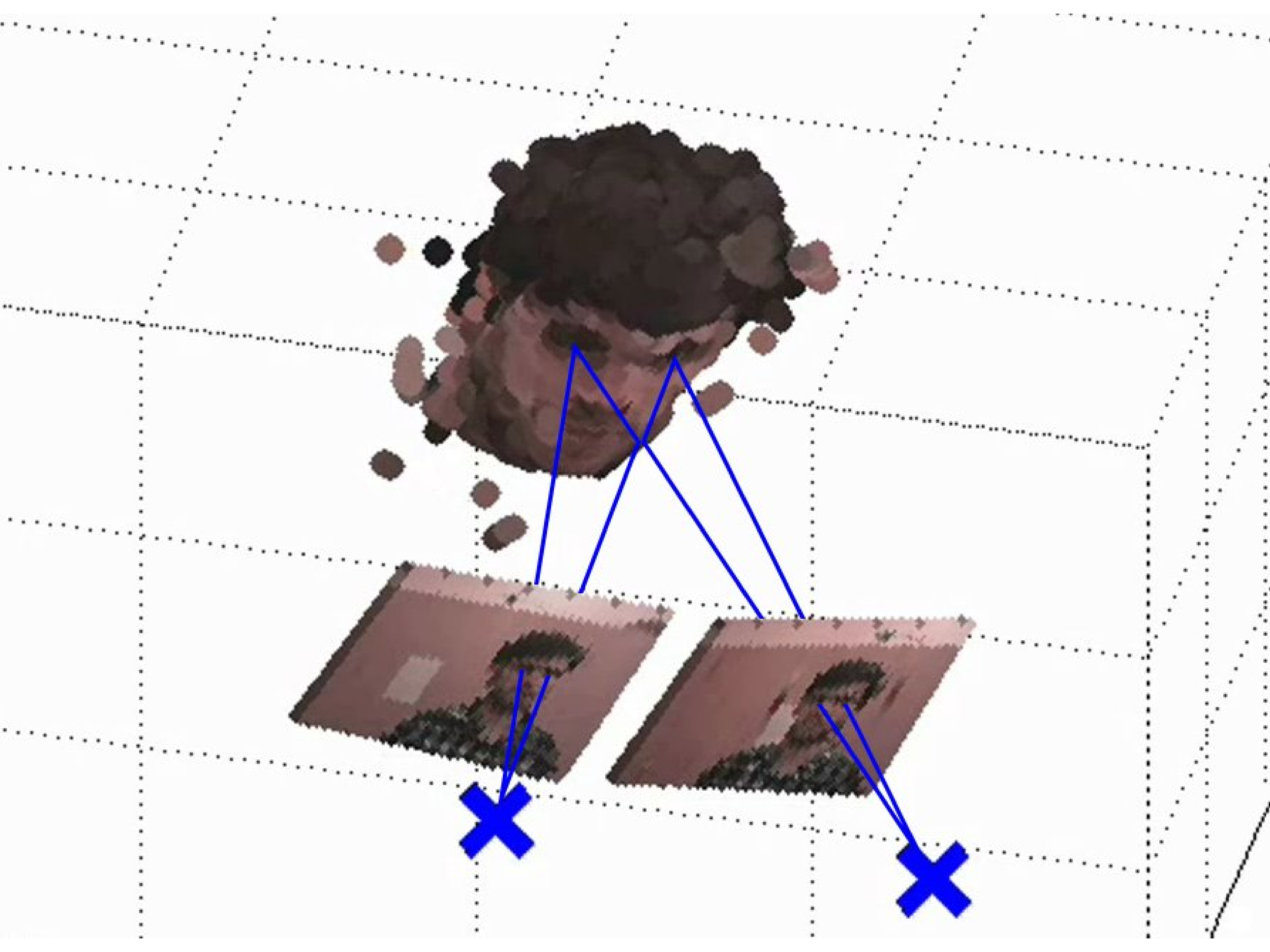


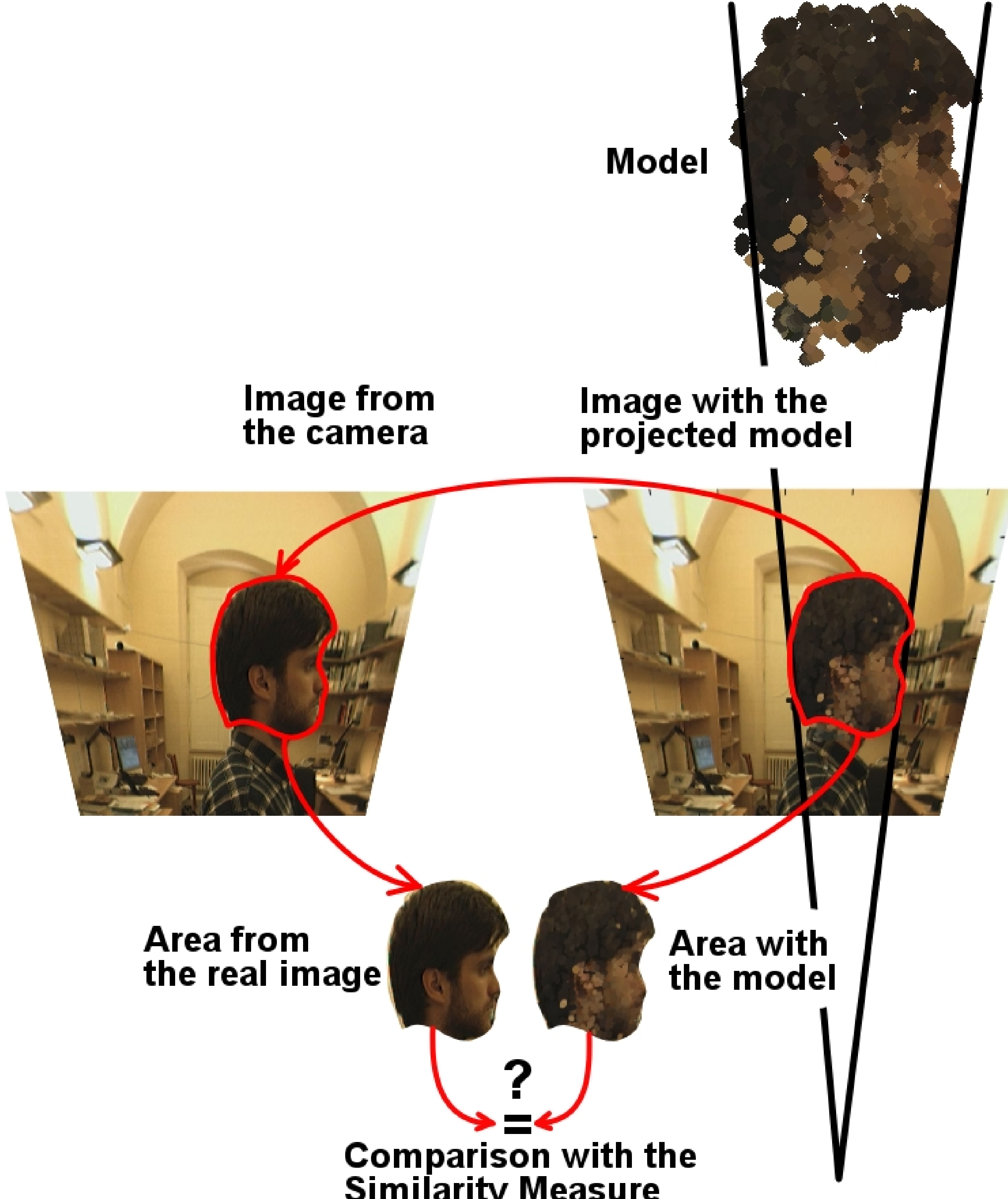
4
3
2
1
0
−1
−2
−3
−4
state value
10
20
30
40
50
60
70
80
90
100
time step
state of the system
estimated value
samples

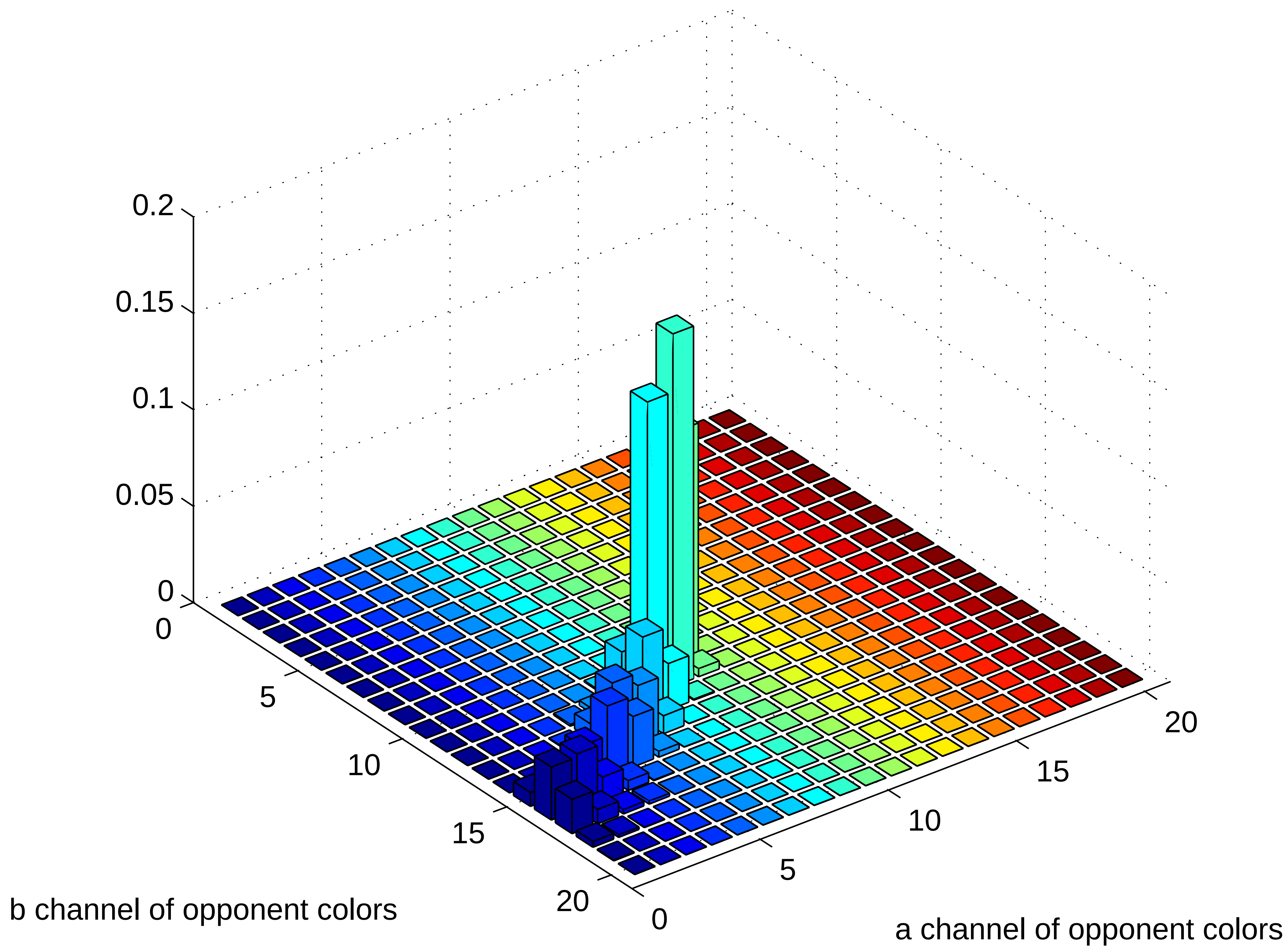

step 13
measurement function
true state
weighted samples
estimated state
0
0.5
1
0
0.01
0.02
−4
−3
−2
−1
0
1
2
3
4


step 53
measurement function
true state
weighted samples
estimated state
0
0.5
1
0
0.02
0.04
−4
−3
−2
−1
0
1
2
3
4


Γ
c




Model
Image from
the camera
Image with the
projected model
Area from
the real image
Area with
the model
?
=
Comparison with the
Similarity Measure


0.2
0.15
0.1
0.05
0
0
5
10
15
20
b channel of opponent colors
0
5
10
15
20
a channel of opponent colors