# Introduction to Advanced Topics in Data Modelling (ATDM)

## A joint course on MSc in Computer Science and Bioinformatics

Kim Steenstrup Pedersen

## Plan for today

- Introduction to the course (formalities)
- Teaser-slides on the course content

# Teachers

- Kim Steenstrup Pedersen, kimstp@di.ku.dk, DIKU Image section (course responsible)
- Christian Igel, igel@di.ku.dk, DIKU Image section
- Thomas Hamelryck, thamelry@binf.ku.dk, Bioinformatics



## Course goals

- Introduce students to advanced methods for modeling and analysis of noisy data.

- The focus is mainly on stochastic modeling and machine learning approaches.

- Allow students to get experience with the theory and, to some extend, the practicalities of advanced data modeling.

## Course learning objectives

At course completion, the student should be able to:

1. Recognize and describe possible applications of selected stochastic and deterministic data models and analysis methods.
2. Explain, contrast and apply selected data representations.
3. Explain and contrast static and dynamic data models and their applications.
4. Apply static and dynamic data models within appropriate applications.
5. Implement selected methods and models.

# Introduction to Advanced Topics in Data Modelling (ATDM)

## Probabilistic Graphical Models

**A joint course on MSc in Computer Science and Bioinformatics**

Kim Steenstrup Pedersen

## Format

- This course consists of:
  - Lectures on foundational theoretical modeling topics
  - Written assignments (throughout the course)

- This course requires your active participation:

  - We encourage you to try to implement and experiment with some methods on your own.

  - As part of the assignments you will also work with selected methods.

## We assume that …
### (Students Prerequisites)

- You have passed the course "Statistical Methods for Machine Learning" or similar.
- You are expected to have a mature and operational mathematical knowledge. Knowledge of linear algebra, geometry, basic mathematical analysis, and basic statistics are relevant.
- You are able to program in a language suitable for scientific modeling (e.g. Python, Matlab, R, and C++).

## Schedule: When and where?

Lectures:

- Mondays 13:15 – 15:00 in room UP1 4-0-17

- Wednesdays 10:15 – 12:00 in room UP1 4-0-17

# Tentative Schedule

| Date | Lecture topic |
|---|---|
| Wed. 22/4 | Bayesian networks |
| Mon. 27/4 | Directed graphical models and inference |
| Wed. 29/4 | Sequential models (HMM, DBN) and inference |
| Mon. 4/5 | Undirected models (Markov random fields) and inference (Gibbs sampling) |
| Wed. 6/5 | Boltzmann machines |
| Mon. 11/5 | Sampling and Markov Chain Monte Carlo methods I |
| Wed. 13/5 | Sampling and Markov Chain Monte Carlo methods II |
| Mon. 18/5 | Sequential inference – Kalman filtering |
| Wed. 20/5 | Sequential inference – particle filtering |
| Wed. 27/5 | Generalized Expectation-Maximization algorithm |
| 8/6 – 10/6 | No lectures |

## Exam

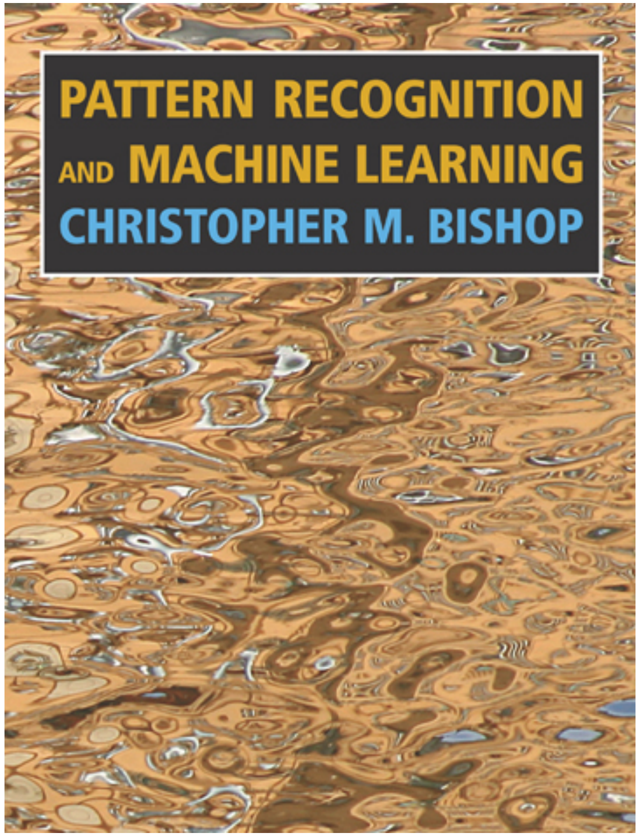
- Continuous assessment
- Three written assignments – solved individually.
- Grading using the 7-point scale by internal grading (intern censur). Grade given based on all assignments.

- Deadlines:
  - Assignment 1: May 4
  - Assignment 2: May 18
  - Assignment 3: June 8

- Assignment 1 hand-out today.

## Course Material

### Main reference




### Additional material

- Selected scientific papers and other relevant references. Available in Absalon under the “Course material” menu item.

- You are also expected to search for relevant literature on your own.

# How to get help

- Discussion board in Absalon

- Talk with the teachers at class or per e-mail

Enough about the formalities!

What modeling topics will be covered on this course?

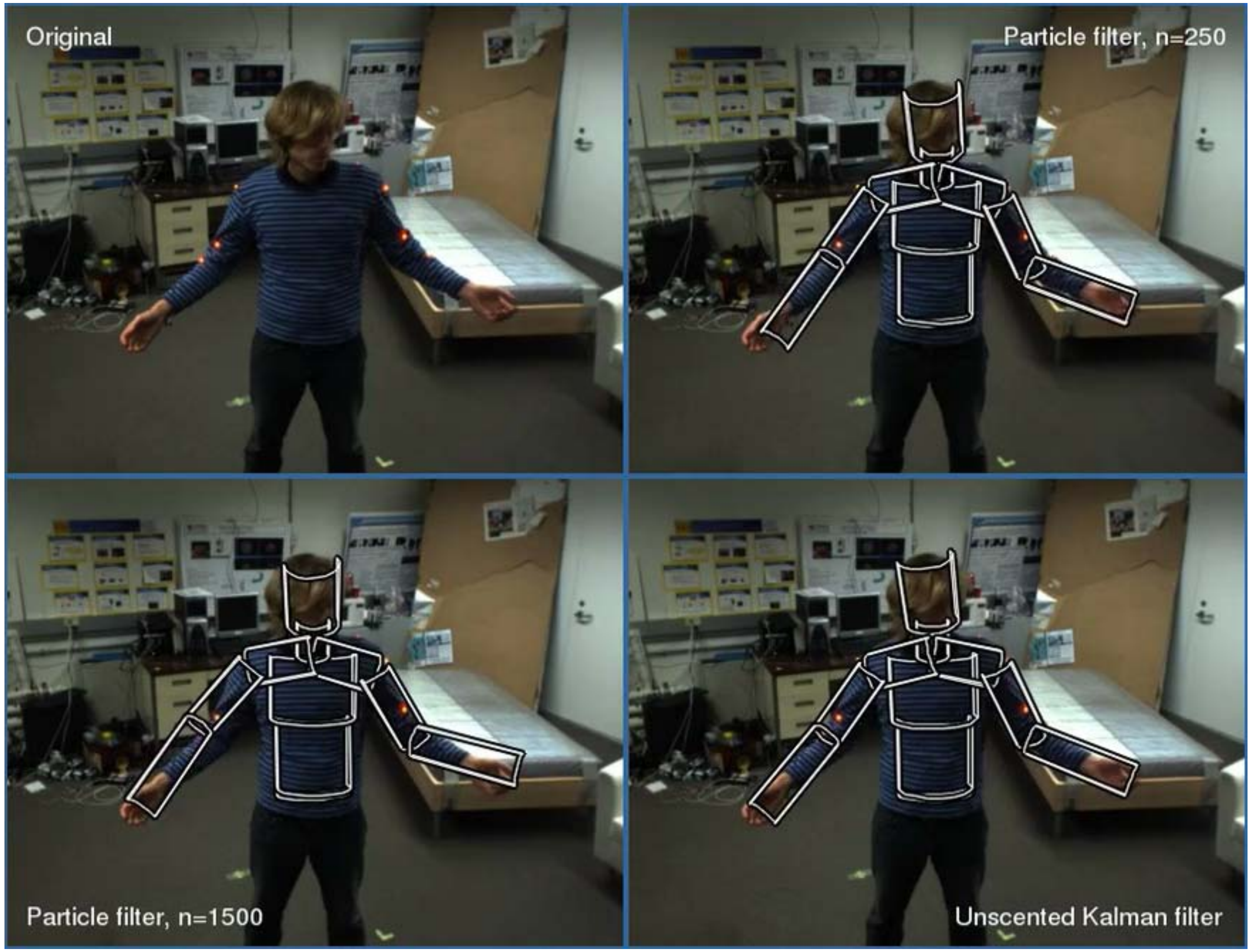
## Foundational modeling topics

- Probabilistic graphical models
- Hidden Markov Models (HMM)
- Markov random field models
- Boltzmann machines
- Sampling methods
- Sequential inference - Kalman and Particle filtering
- Generalized EM algorithm

You will work with parts of these topics in the assignments.

Let me give an example of an application that covers the four of these topics

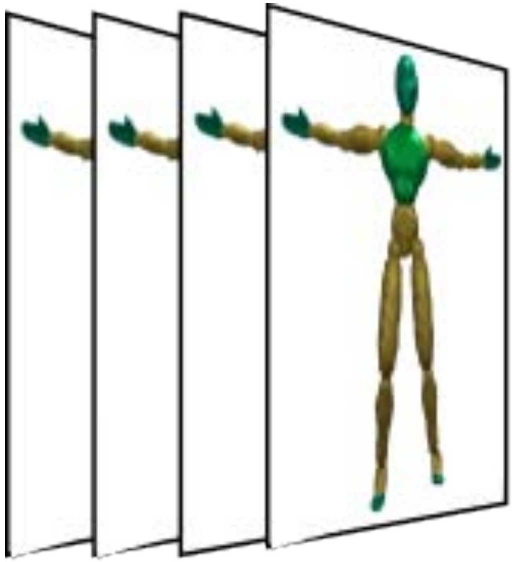
# Visual 3D tracking of human motion



# This is what we want to do
## (Tracking = seq. estimation of model from observations)

Video sequence $I_{1:t}$

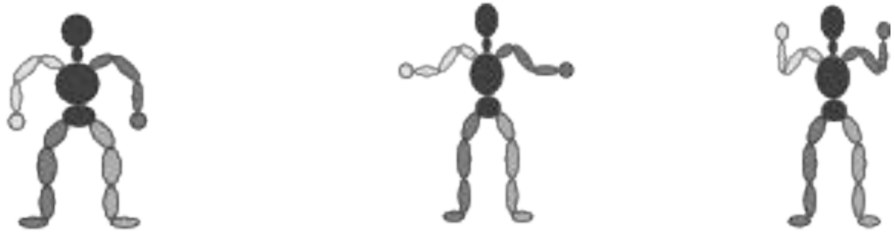
…



$I_{t-3}$ $I_{t-2}$ $I_{t-1}$ $I_t$

Sequence of estimated states $\Theta_{0:t}$

…



$\Theta_{t-1}$ $\Theta_t$ $\Theta_{t+1}$

Short-hand notation:

$$I_{1:t} = \{I_1, \ldots, I_t\}$$

$$\Theta_{0:t} = \{\Theta_0, \ldots, \Theta_t\}$$

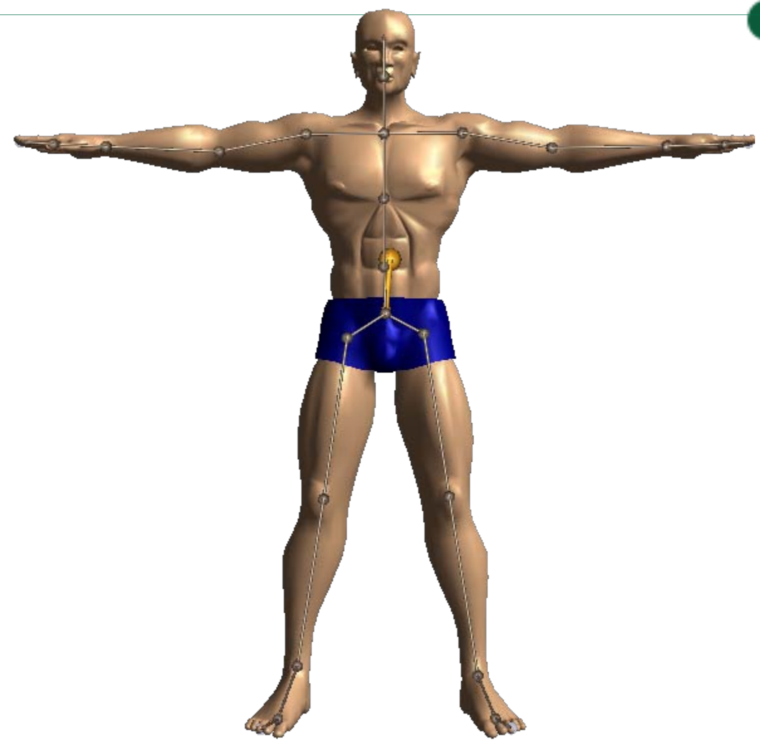
Prediction of next state

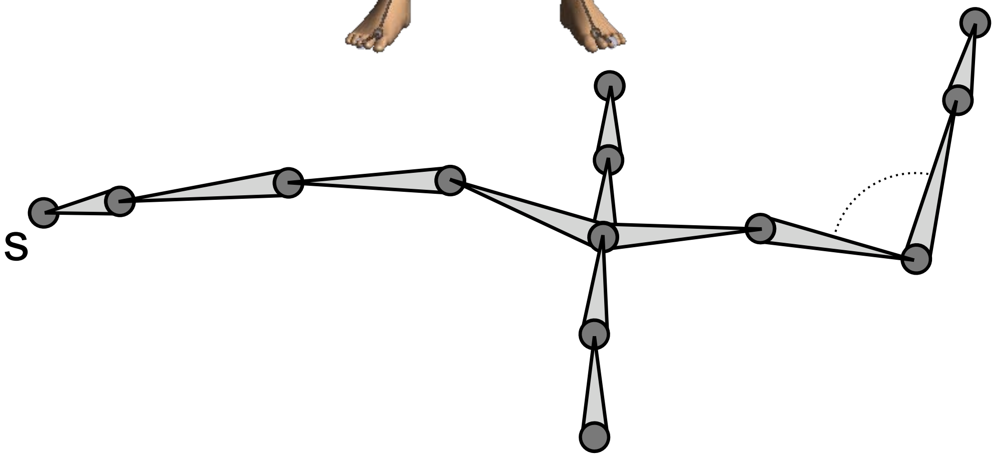
# Visual human motion modeling and tracking
**(Tracking = seq. estimation of model from observations)**

- **Model:** Stick figure representation of human body (rigid sticks connected by joints, no mass).
  - Model parameters a.k.a. the state:
    Vector of joint angles or 3D positions of joints

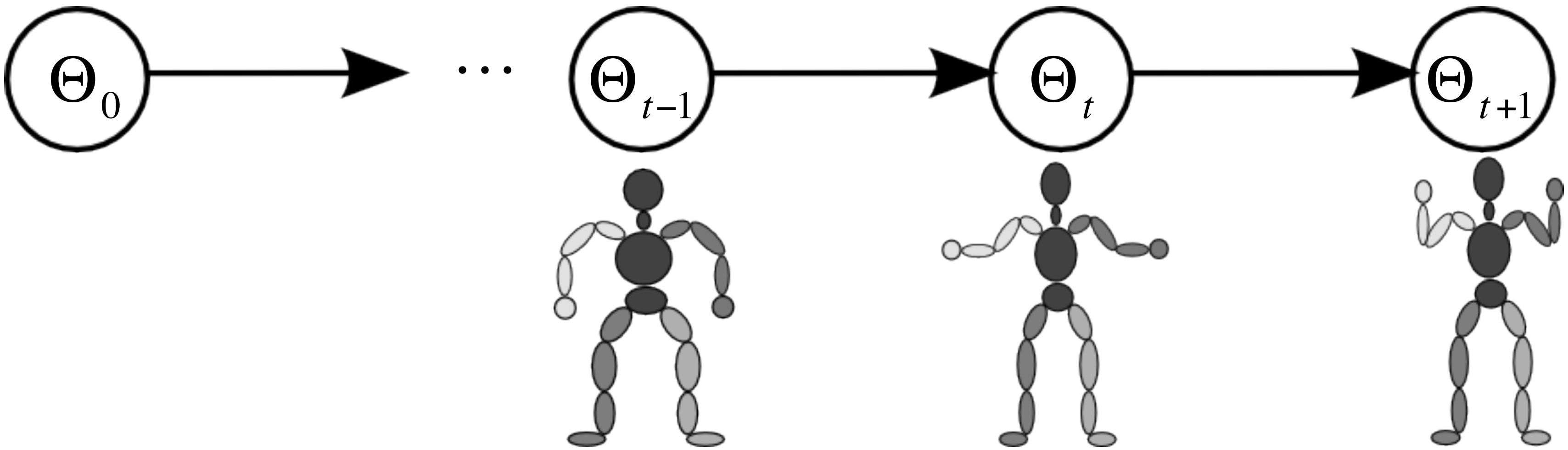
$$\Theta = \left[\theta_1, \ldots, \theta_D\right]^T$$

  - (And other representations exists)



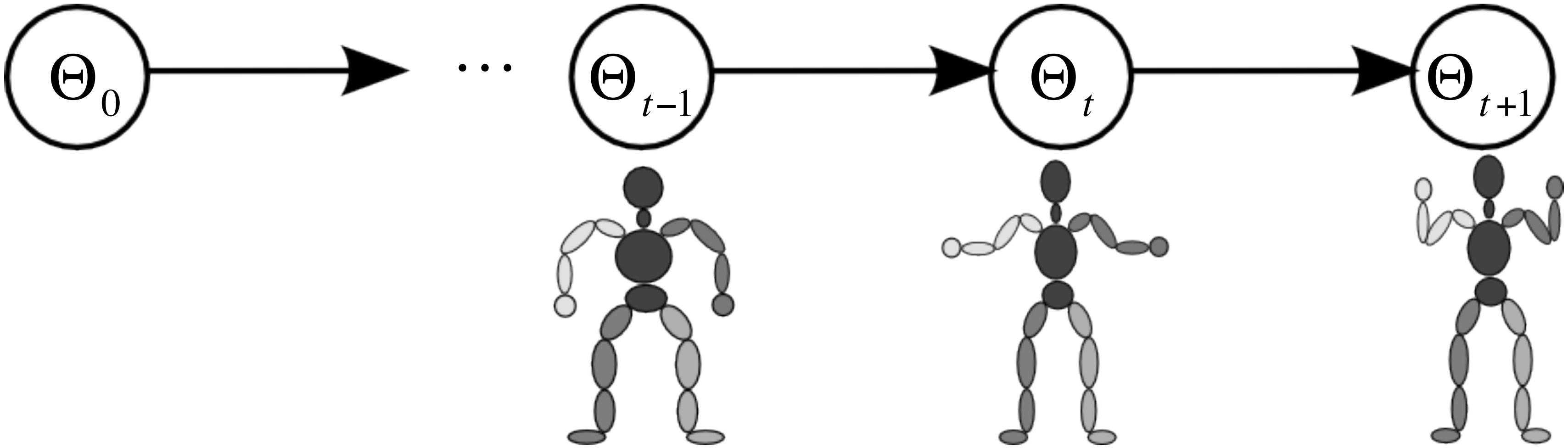


## How do we include dynamics into our model?



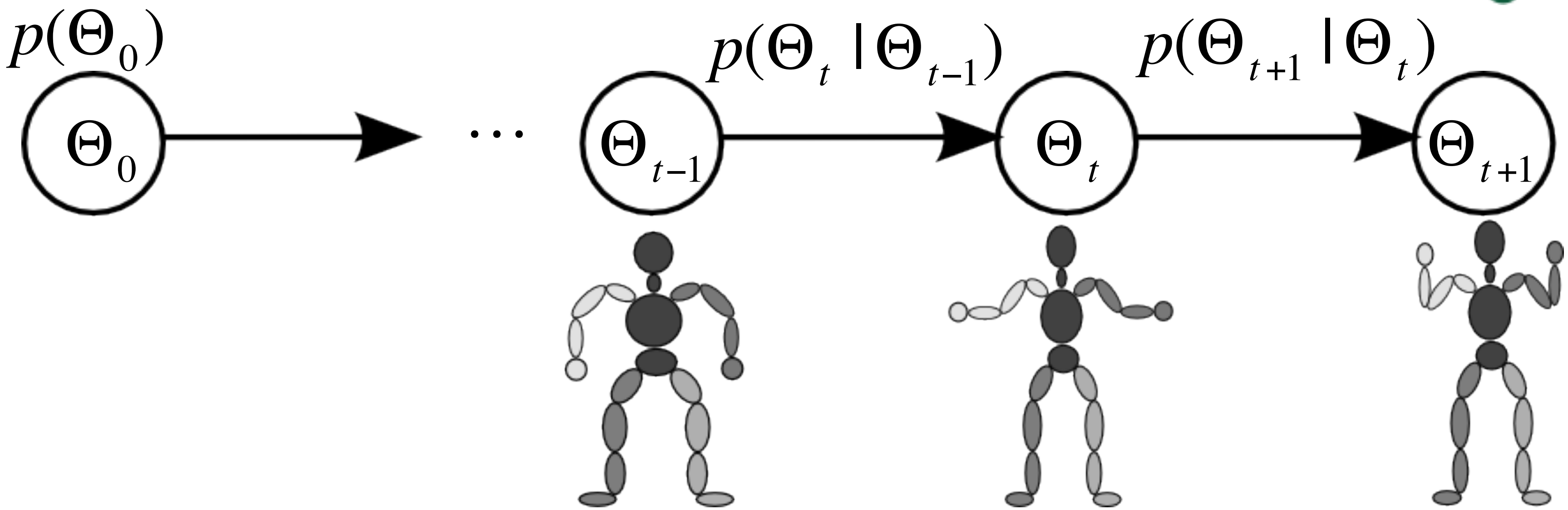

- Lets assume that the current state only depends on the immediate past state. We assume that somehow we can compute the new state given the old state!
- However, this update of states is stochastic (uncertain) – we are going to estimate it from noisy observations.

## Enter probabilistic graphical models
### (More on this in Thomas Hamelryck's lectures)




- This is an example of a simple graphical model:
  - First order Markov chain
  - A directed acyclic graph (DAG) or tree, if you will.
- **Def. Graphical model:** Graph based model where *nodes* represent random quantities / variables and *edges* represents dependencies among variables.

## Adding a probabilistic dynamical model



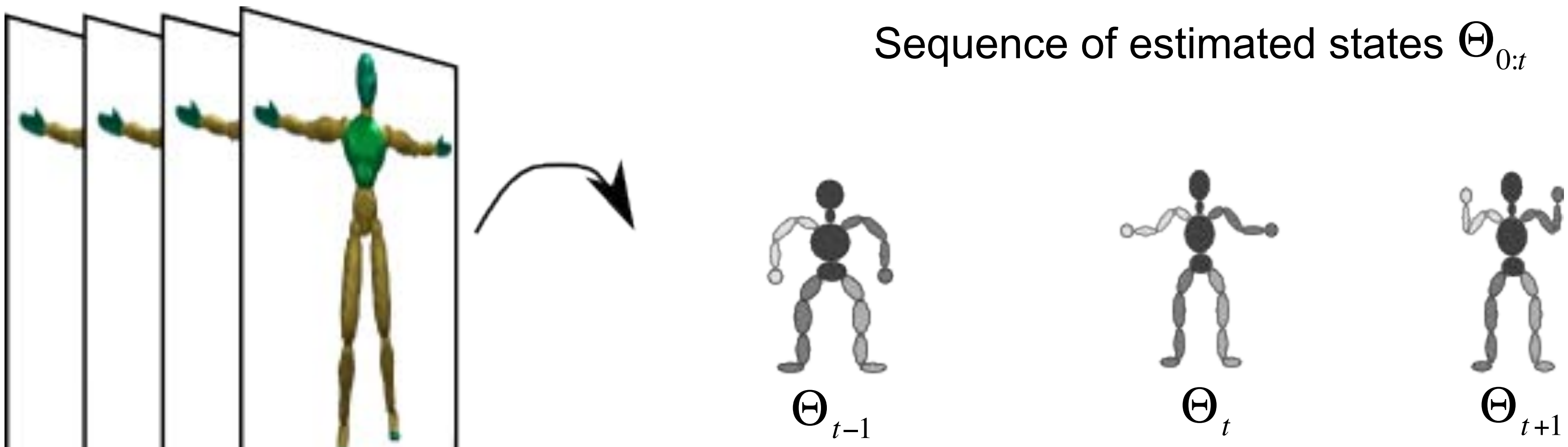

- If we know the transition conditional probability distributions and the prior distribution on the initial state, we can compute the probability distribution of state sequences (of particular sequences of stick figures):

$$p(\Theta_0,\ldots,\Theta_t) = p(\Theta_{0:t}) = p(\Theta_0)\prod_{i=1}^{t} p(\Theta_i \mid \Theta_{i-1})$$

# How to relate the model state with observations?
## (Tracking = estimation of model from observations)

Video sequence $I_{1:t}$

Sequence of estimated states $\Theta_{0:t}$



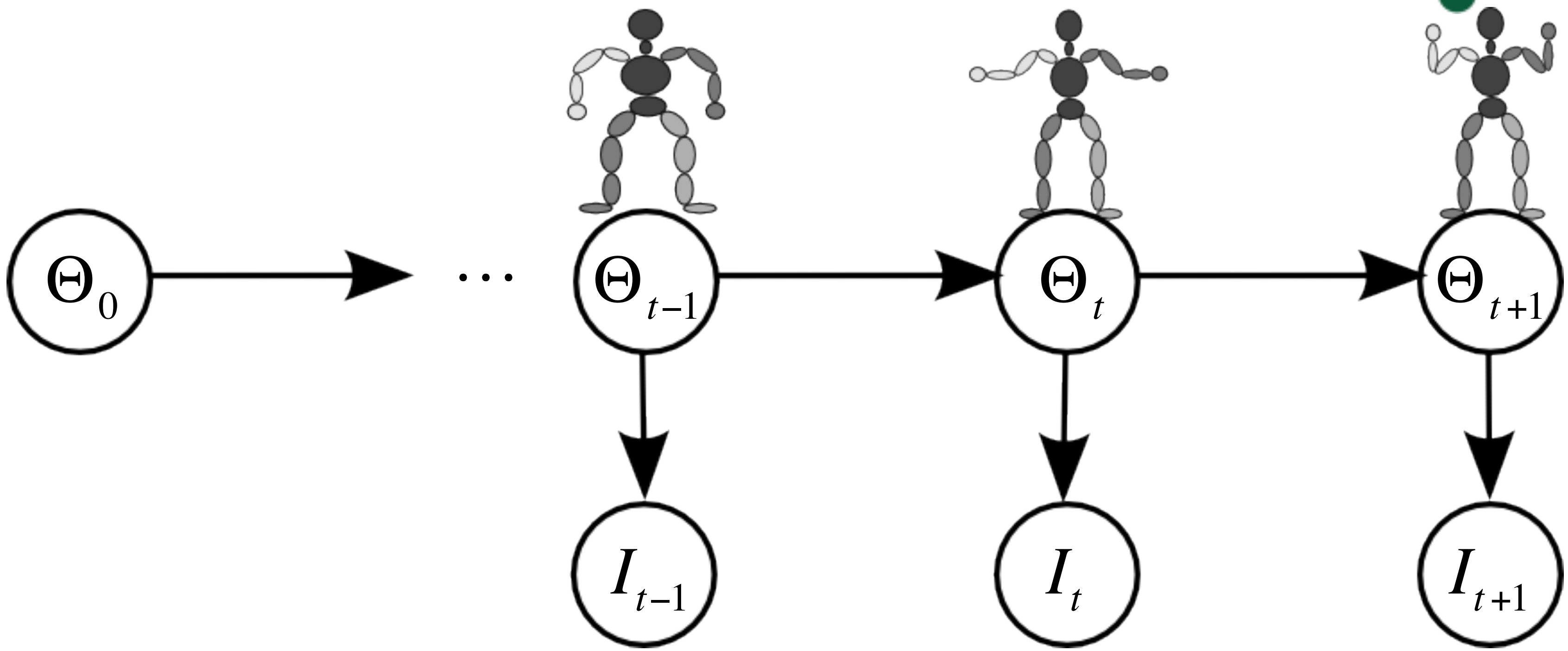

What do the image of a particular stick figure look like? Hard problem!

Lets introduce a potentially non-linear function for "*drawing the stick figure*" in image space and compare with the observed image. Something like this,

$$\left\| I_t - F(\Theta_t) \right\|^2$$

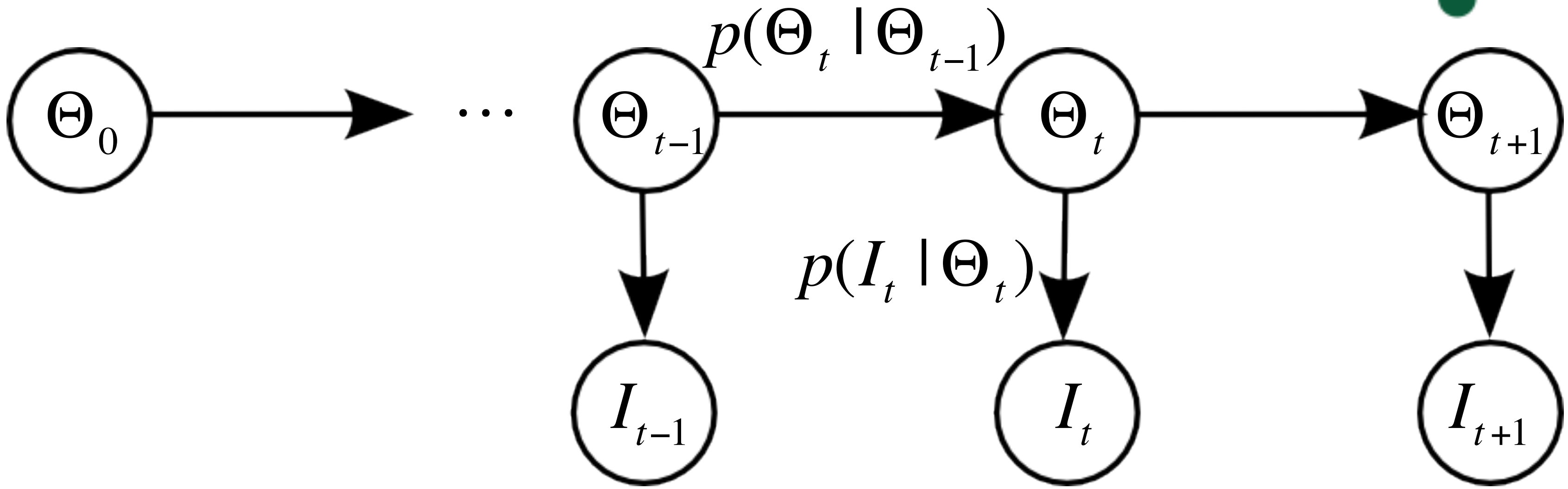
However our observations are noisy so we want a probabilistic model for this!

# Enter hidden Markov models (HMM)
**(More on this in Thomas Hamelrycks lectures)**




- This model states that we only observe the images directly and the states indirectly – they are hidden (latent variables). If states are discrete we have a HMM.
- First order Markov chain in the states.

# We need a probabilistic observation model




How about a Gaussian observation model?

$$p(I_t \mid \Theta_t) = \frac{1}{Z}\exp\left(-\frac{\left\|I_t - F(\Theta_t)\right\|^2}{2\sigma^2}\right)$$

Or perhaps more useful $\quad p(I_t \mid \Theta_t) = \dfrac{1}{Z}\exp\left(-H(I_t, \Theta_t)\right)$

# Observations in practice

For now lets consider $H(I_t, \Theta_t)$ as a black-box that compares image $I_t$ (here a point cloud from a stereo depth map) and state $\Theta_t$

## How to do tracking?

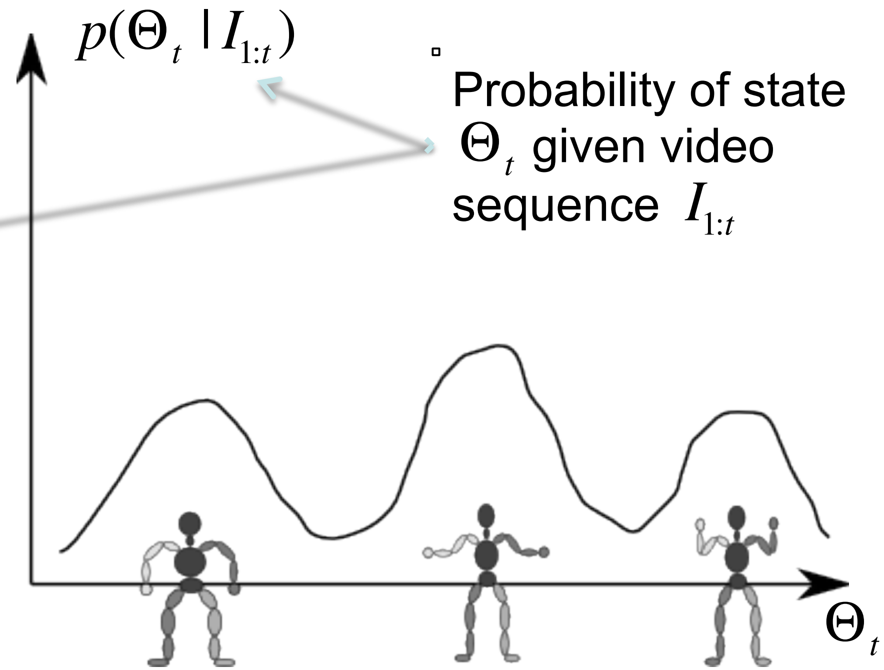
- The model gives us the joint distribution

$$p(I_{1:t},\Theta_{0:t}) = p(\Theta_0)\prod_{i=1}^{t} p(I_i \mid \Theta_i)\, p(\Theta_i \mid \Theta_{i-1})$$

- If we want to do real-time tracking we need

$$p(\Theta_t \mid I_{1:t})$$

- And then compute e.g. averages to estimate the current state.



## Remember that …

- by applying the sum and product rules we have

$$p(\Theta_t \mid I_{1:t}) = \frac{p(I_{1:t}, \Theta_t)}{p(I_{1:t})}$$

and

$$p(I_{1:t}, \Theta_t) = \int p(I_{1:t}, \Theta_{0:t}) d\Theta_{0:t-1}$$

$$p(I_{1:t}) = \int p(I_{1:t}, \Theta_{0:t}) d\Theta_{0:t}$$

Hence, what we need, can be derived from the joint distribution.

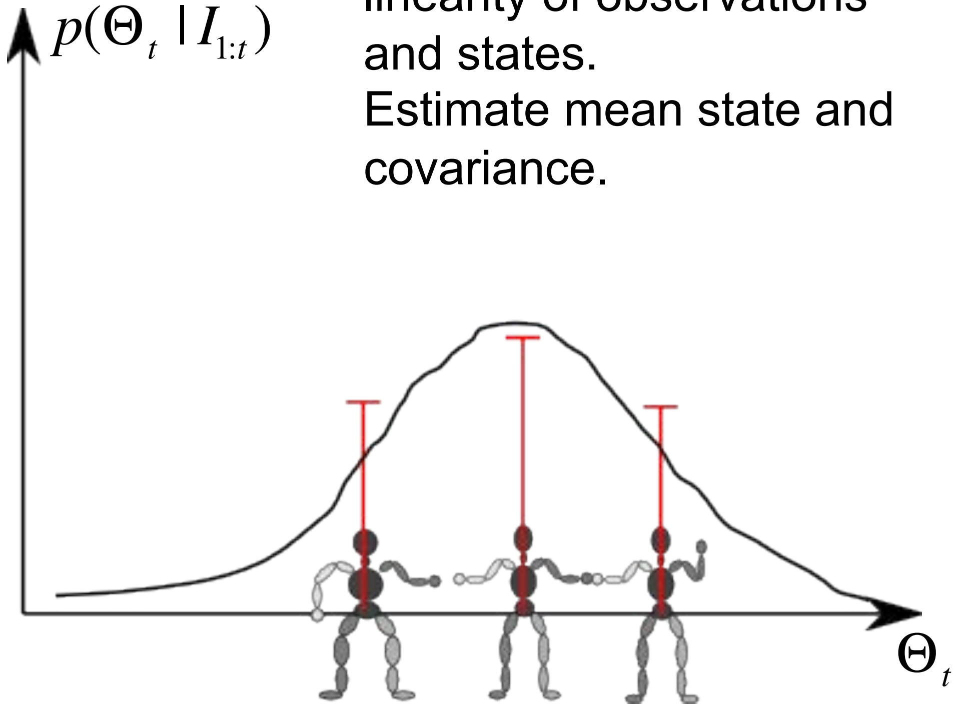
(At least in theory!)

## So we need to sequentially estimate $p(\Theta_t \mid I_{1:t})$

**(I will give lectures on this)**

### Kalman filtering

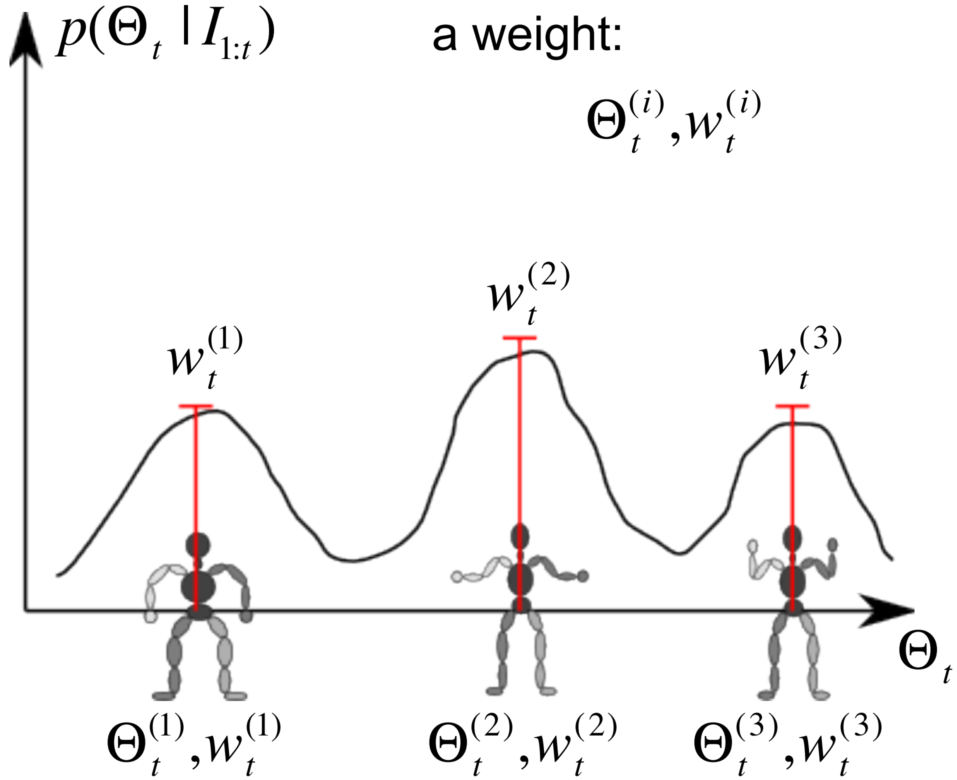
- Assume Gaussianity and linearity of observations and states.
Estimate mean state and covariance.




### Particle filtering

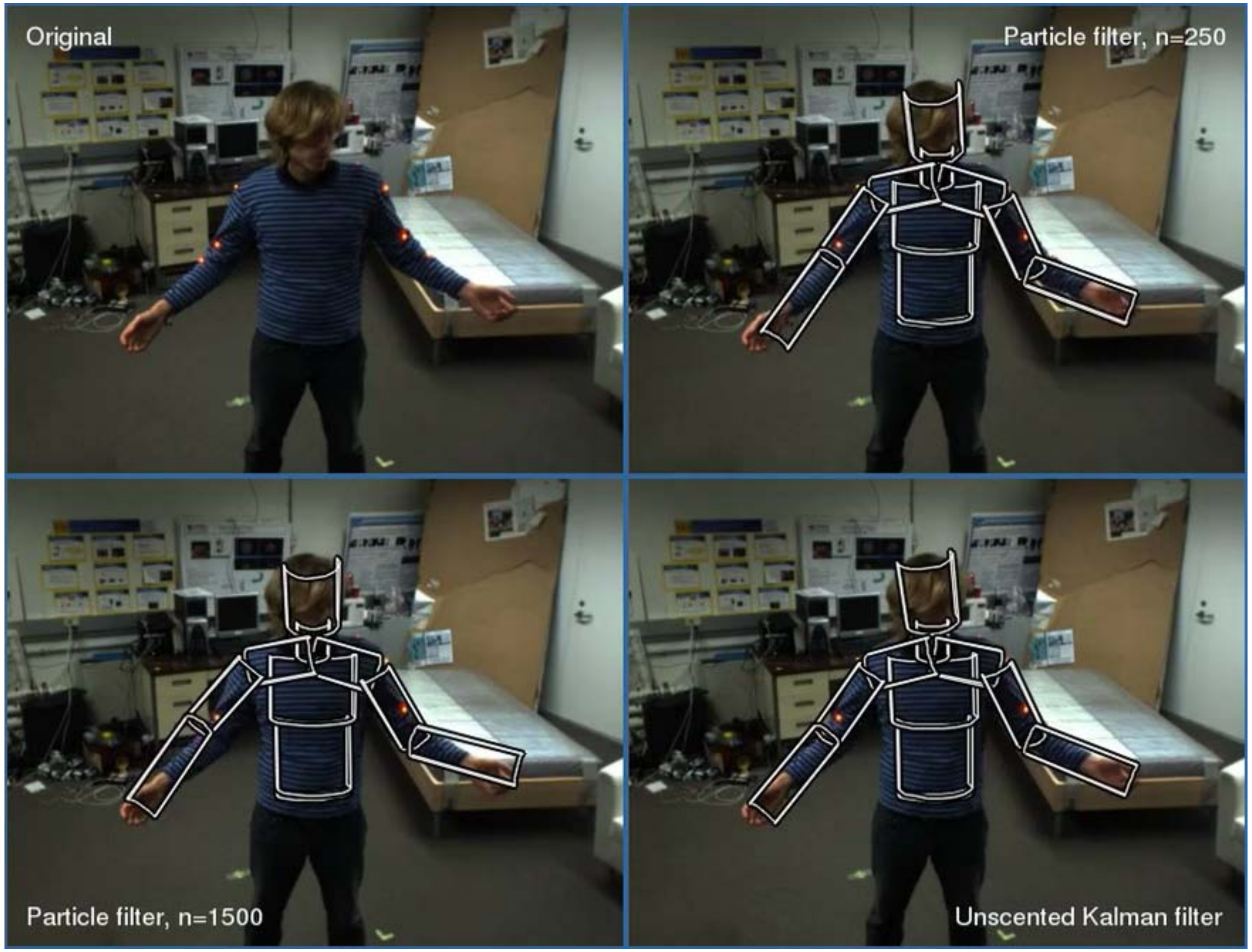
- Particles: A state and a weight:

$$\Theta_t^{(i)}, w_t^{(i)}$$



# Voila! We now have a visual tracker

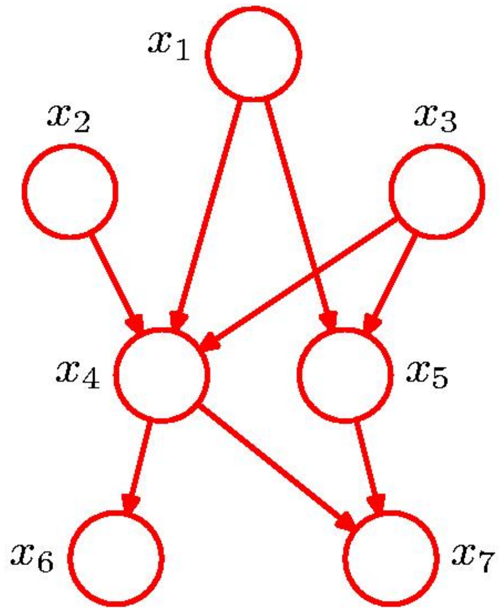
# And here is another sequence showing different inference methods

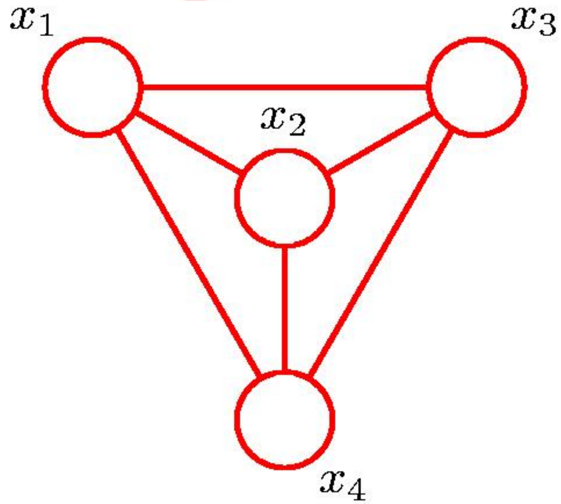


End of motivating tracking example…

## Graphical models in general
**(More on this in Thomas Hamelryck's first lecture)**

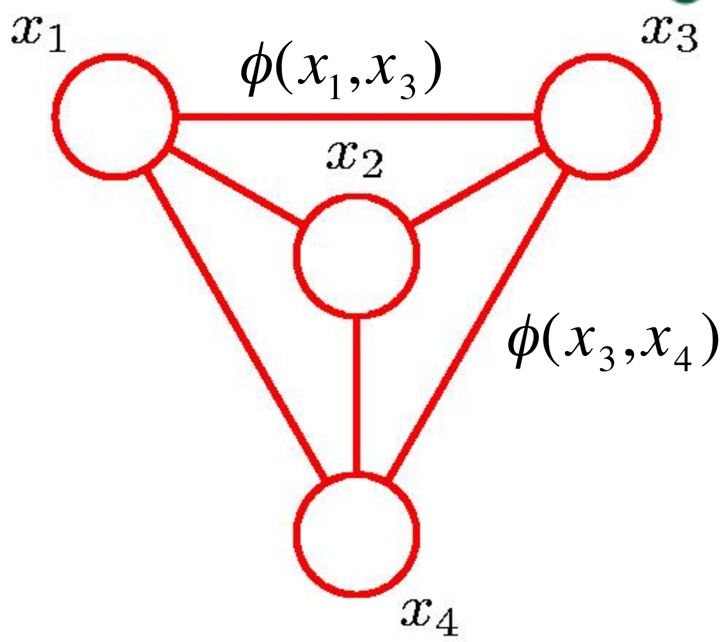
- Includes both
  - Directed graphs (e.g. Bayesian networks, Markov chains)
  - Undirected graph (e.g. Markov random fields)
- General algorithms exist for performing inference on graphical models – computing joint and marginal probabilities and expectations.



## Markov random field (MRF) models
**(More on this in Christian Igels lectures)**

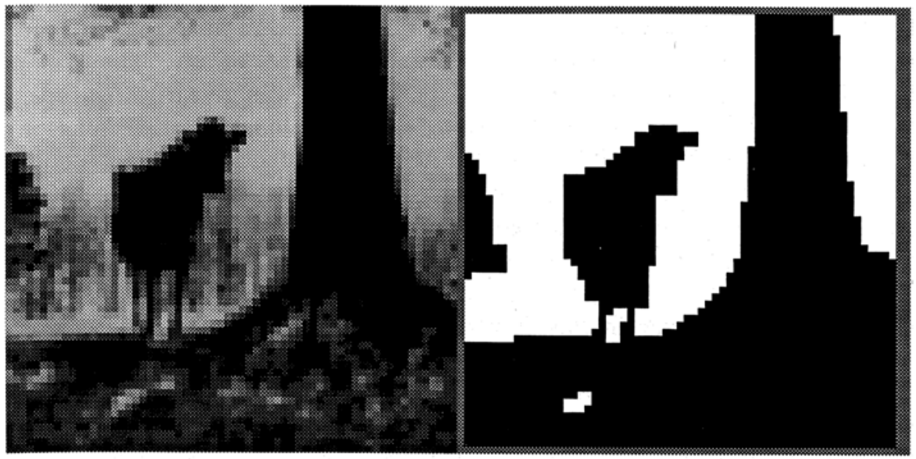
- Originates from statistical physics (atomic / molecular lattice models).
- Provide models of ordered data (lattices, arrays, images).
- Defined via interaction potentials $\phi(x_i, x_j)$, e.g.




$$p(x_1, x_2, x_3, x_4) = \frac{1}{Z}\exp\left[-\sum_{i,j}\phi(x_i, x_j)\right] \quad \text{(Gibbs distribution)}$$

# What can we use MRFs for?

**Mumford’s cow segmentation (Ising model – binary pixel-wise segmentation):**



**Observational model for tracking (or texture model):**

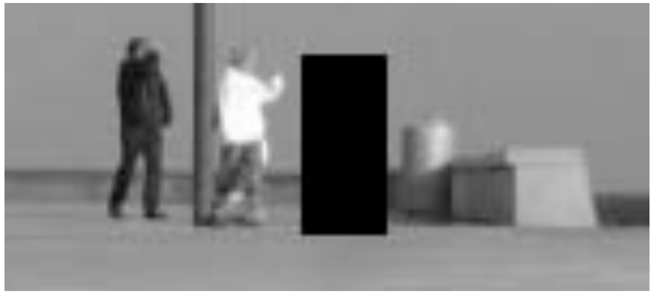

$$H(I_t, \Theta_t) = \sum_{i,j} \phi(I_{t,i}, I_{t,j}, \Theta_t)$$

## Particle filtering and MRF for image inpainting: Fill holes of missing pixels

- Synthesize content to fill holes in images.
- Exemplar-based: find similar image patches and paste (jigsaw-puzzle).
- Our approach: Keep several hypotheses in play. E.g. allow for several solution and choose the one that is globally optimal.

Original

Hole



Exemplar approach

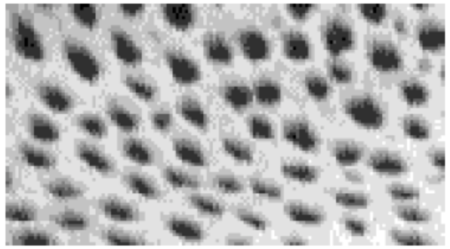
Our approach

Cuzol et al: Field of Particle Filters for Image Inpainting. In Journal of Mathematical Imaging and Vision, 31(2-3): 147-156, 2008.

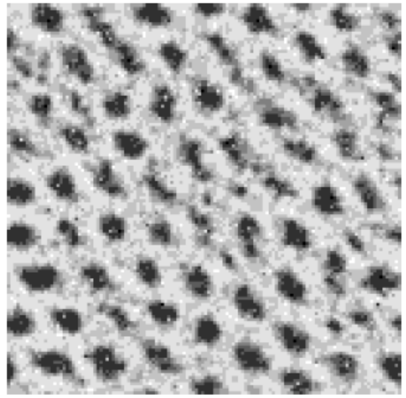
## MRFs for texture modelling: The FRAME model

- Given a set of input images, learn a MRF model based on filter responses of the image.
- Since it is generative we may synthesize new images from the model.
- Zhu et al: Minimax Entropy Principle and Its Application to Texture Modelling. In Neural Computation, 9(8): 1627-1660, 1997.

Original texture:

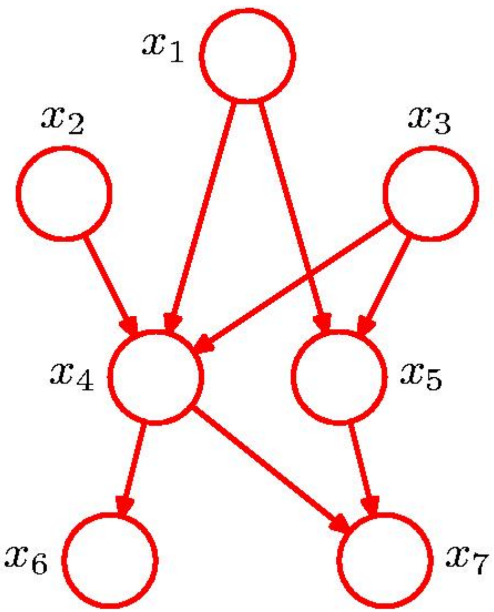


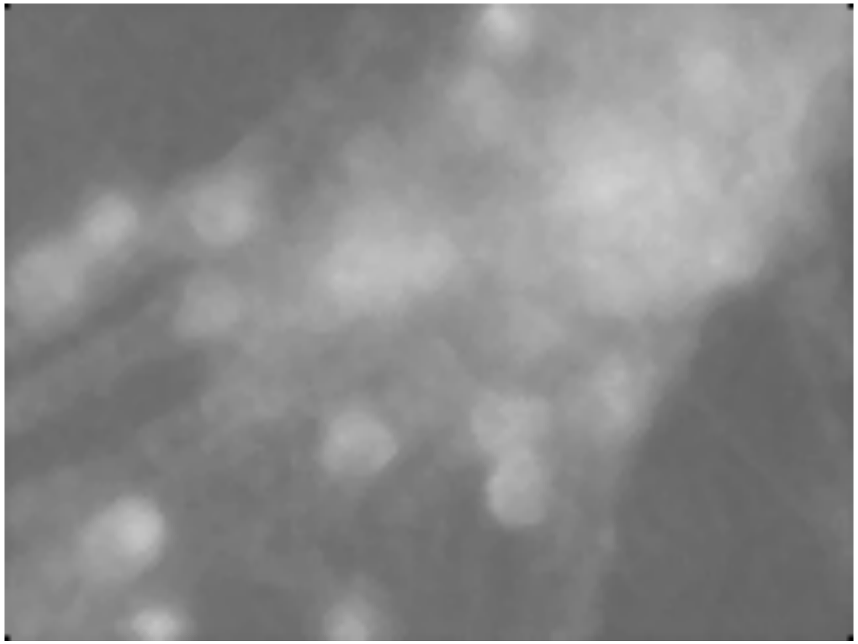
Texture synthesis:



## Dynamical Bayesian networks for neuron cell activation patterns

- Given microscopy video sequences of a neuron cell network (with calcium dye), we want to model the activation patterns in the cell network using dynamical Bayesian networks.



The theoretical details and more examples will be given throughout this course

Next lecture on Wednesday 10:15 – 12:00

Bayesian networks

by

Thomas Hamelryck