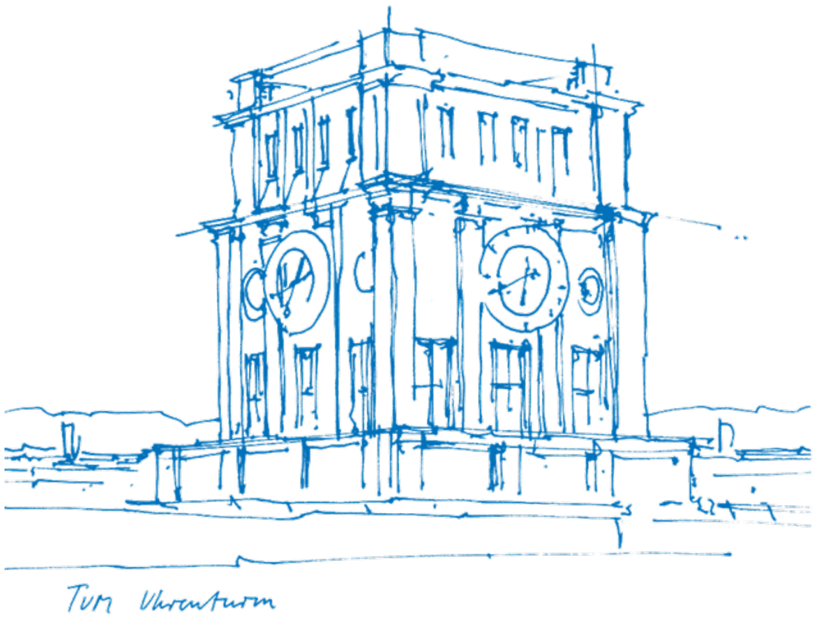
0 : Introduction and Organization

Tao Wu, Yuesong Shen, Zhenzhang Ye

Computer Vision & Artificial Intelligence
Technical University of Munich



General Information

Prerequisites

- ○ (Discrete) probability theory.
- ○ (Basic) graph theory.
- ○ Programming experience in Python (or Matlab).
- + Discrete/continuous optimization.
- + Machine learning.
- + Related courses:
  - Computer Vision I & II.
  - Machine Learning for CV.
  - Convex Optimization for CV & ML.

Outline of the Course

- **Representation**
  - Bayesian network (directed model);
  - Markov network (undirected model);
  - Factor graph, Exponential family.
- **Inference**
  - Exact inference: variable elimination, message passing;
  - Variational inference: mean field, loopy belief propagation;
  - Sampling methods: rejection/importance sampling, Gibbs sampling;
  - MAP inference: Graph cut, Linear programming relaxation.
- **Learning**
  - Maximum likelihood estimation (MLE);
  - Partial observation and expectation-maximization (EM) algorithm;
  - Structured learning: structured support vector machine (SSVM).
- **Further topics** (if time permits)
  - Hidden Markov model and Kalman filter;
  - Boltzmann machines and contrastive divergence, etc.

Contact Information

- Tao's office: 02.09.061
- Yuesong's office: 02.09.039
- Zhenzhang's office: 02.09.060
- Office hours: Please write an email.
- Lecture: Starts at quarter past; Short break in between.
- Course webpage (where you check out announcements):
  `https://vision.in.tum.de/teaching/ss2019/pgm2019`
- Homework: assigned on Monday; hand in on Monday one week after.
- Bonus policy: see the course webpage.
- Submit your programming exercises per email to:
  `pgm-ss19@vision.in.tum.de`
- Passcode for accessing course materials:
  `bayesian`

What and Why about PGM?

Probabilistic Graphical Model

- **Probabilistic graphical model (PGM)**, or **graphical model** for short, is a probabilistic model which uses a graph to represent dependencies among its random variables.

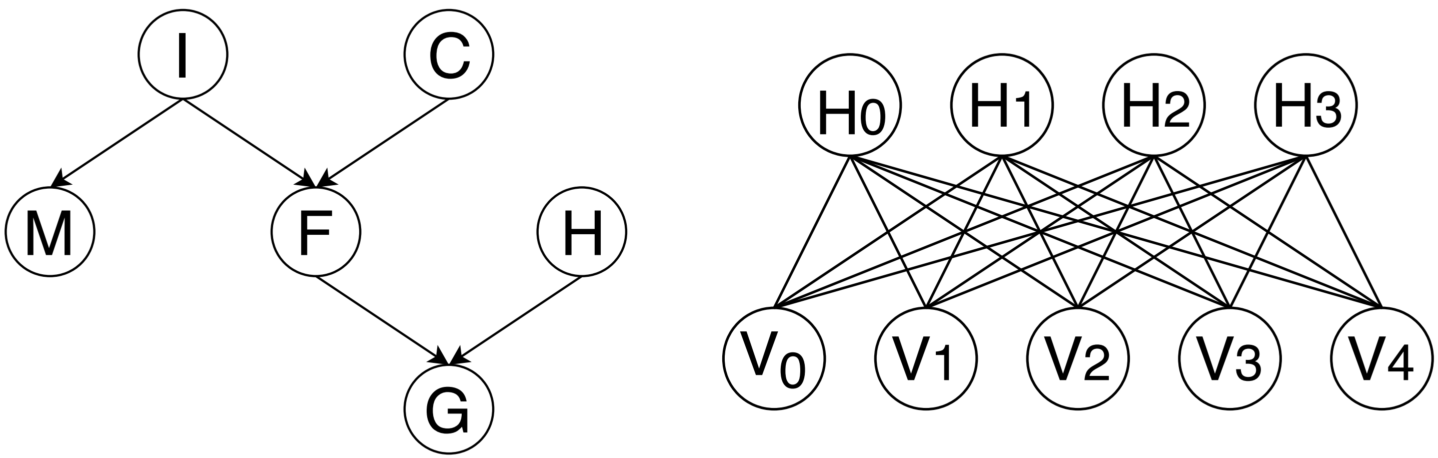


Figure: Examples of graphical models: Bayesian network (left) and Markov network (right).

Graphical Representation

- **Nodes**: random variables;
- **Edges**: interactions;
- **Overall graph**: joint distribution.

$\rightsquigarrow$ Declarative and intuitive graph representation of the probability distribution.

Random variables:

- **I**: interesting subject?
- **C**: cool professor?
- **M**: master thesis?
- **F**: follow course?
- **H**: hard work?
- **G**: good grade?

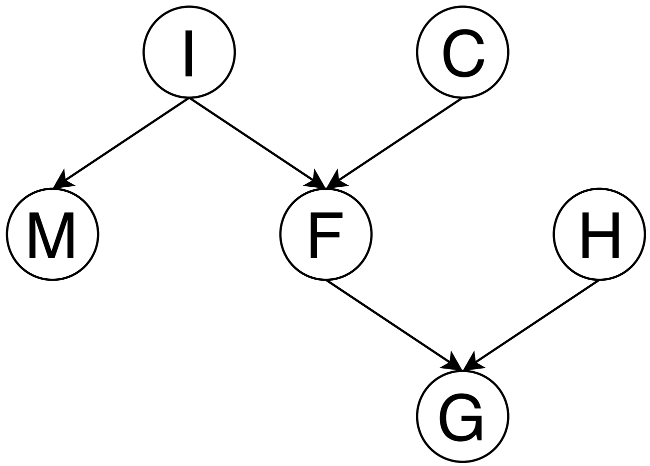


Figure: Corresponding Bayesian network.

Structured Interaction

- Graph structure indicates **independence assumptions**.
- Example: A binary 28×28 MNIST image $\rightsquigarrow |\mathcal{V}| = 784$ binary RVs:
  - In general: $2^{|\mathcal{V}|} - 1 \approx \mathbf{10^{236}}$ free parameters for joint distribution!
  - Full independence: $|\mathcal{V}| = \mathbf{784}$ free parameters;
  - Grid-structured dependence: $|\mathcal{V}| + |\mathcal{E}| = \mathbf{2296}$ free parameters.

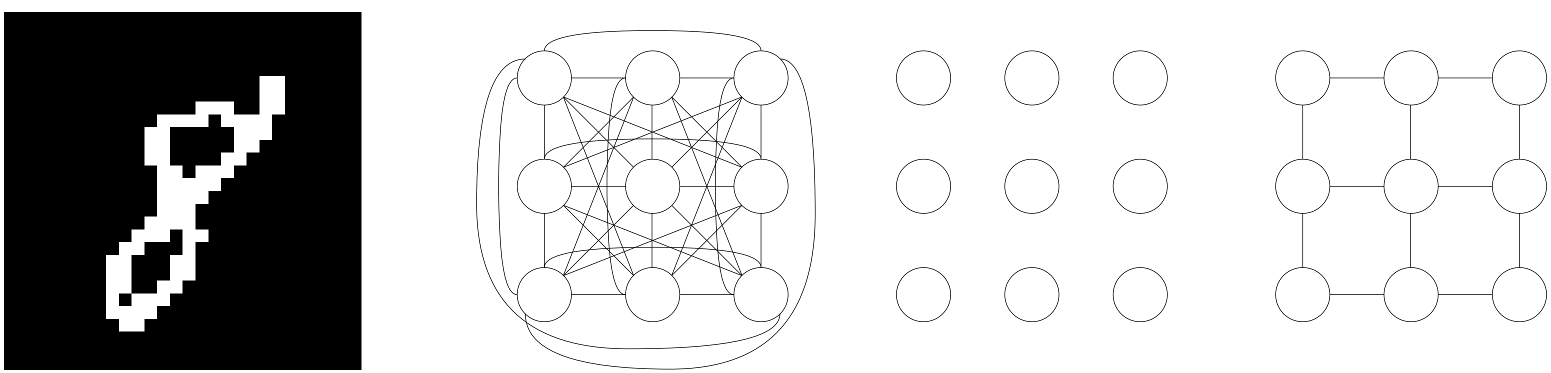


Figure: Binary MNIST image and Markov network with different independence assumptions.

Independence assumptions $\leftrightarrow$ Factorization $\leftrightarrow$ Tractable modeling

Inference: Reasoning with Uncertainty

- Getting info from graphical models $\rightsquigarrow$ reasoning with uncertainty!
- **Inference** process can answer queries like:
  - How likely will I get a good grade: if I **F**ollow the course? if I find the subject **I**nteresting but don't want to work **H**ard?
  - My friend is **F**ollowing this course, how likely is the subject **I**nteresting?
  - What are the most probable values for the missing pixels?

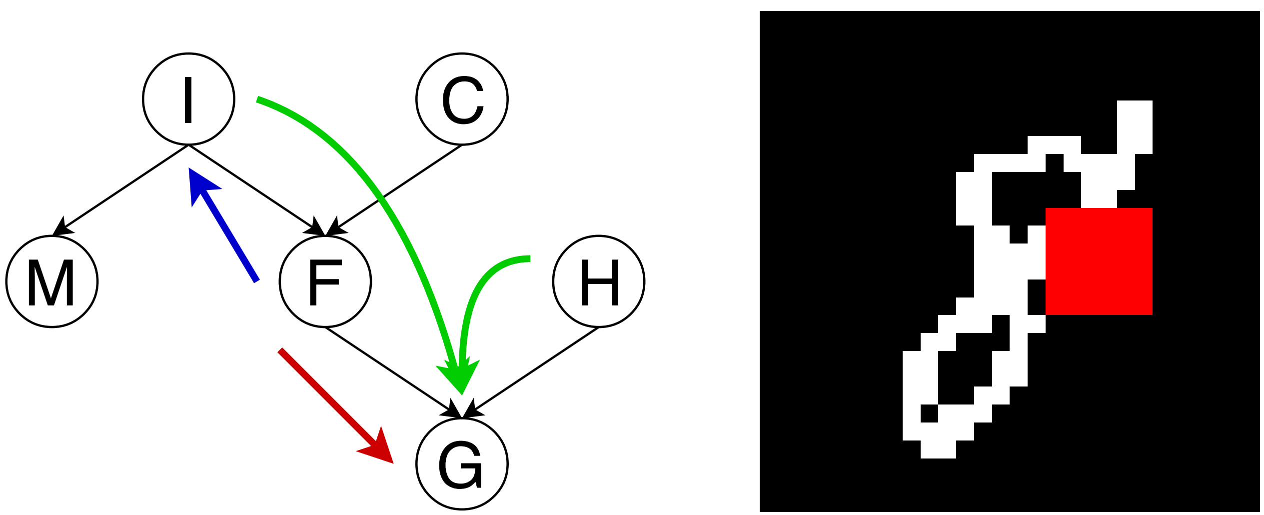


Figure: Illustration of some inference task examples

Computer Vision & Artificial Intelligence
Department of Informatics
Technical University of Munich

Learning: Data-driven Model Design

- Parameters and structure of a graphical model can be set ...
  - manually by human expertise and prior ⇝ **knowledge engineering**
  - automatically trained from observed data ⇝ **machine learning**

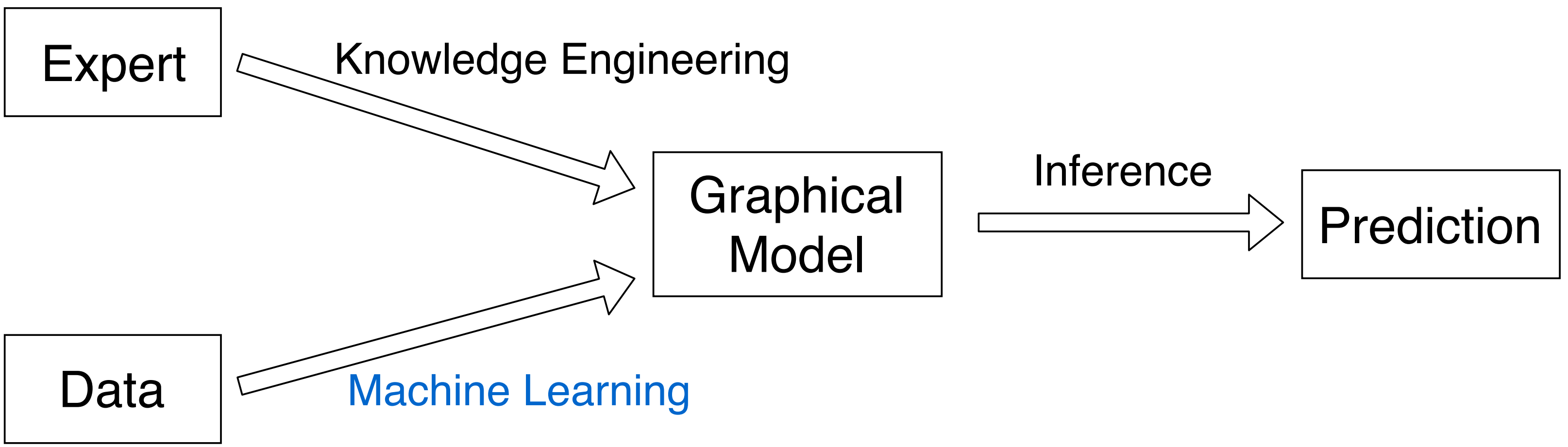


Figure: Design and usage of graphical model.

Applications

Application: Expert System for Medical Diagnosis

- Knowledge engineering with **Bayesian network**.

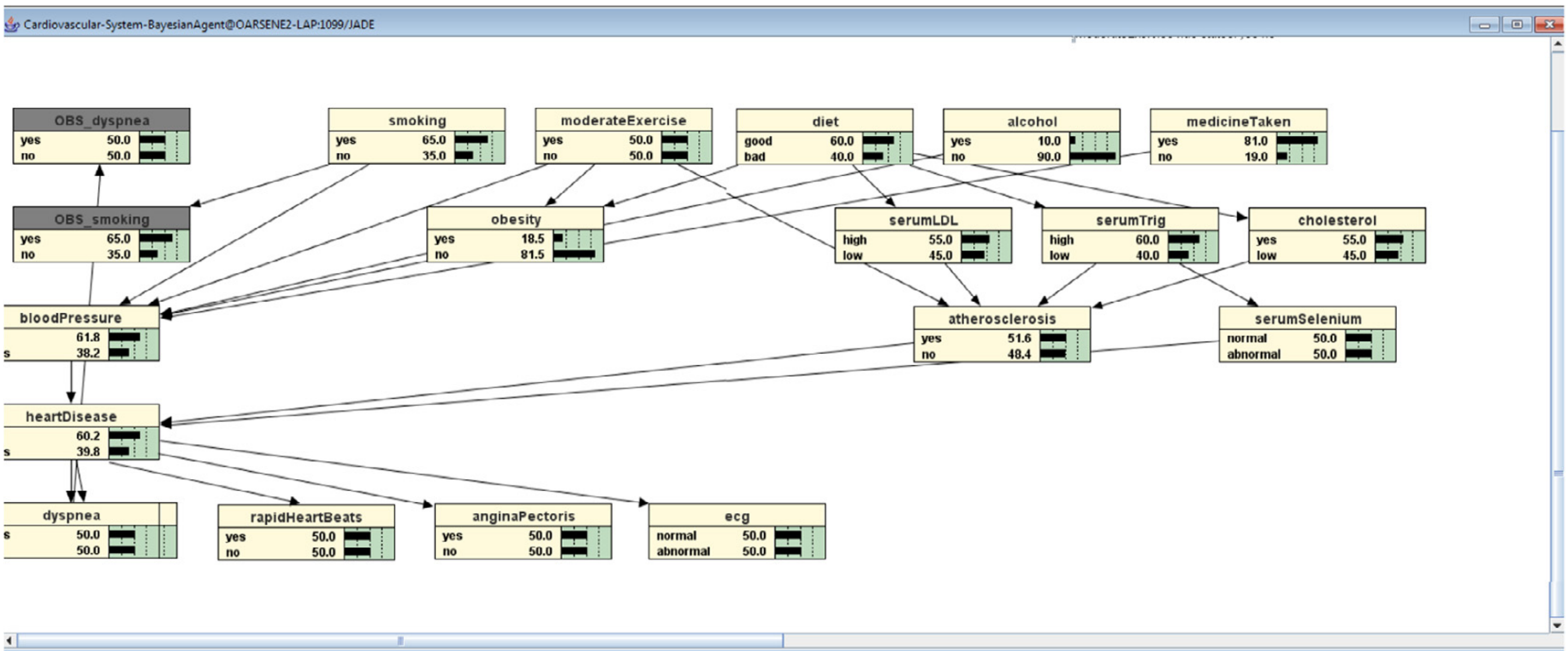


Figure: Illustration of an expert system for medical diagnosis of cardiovascular system[1].

[1] Arsene et al., "Expert system for medicine diagnosis using software agents".

Application: Natural Language Processing (NLP)

- Modeling sequential data with **hidden Markov model**.

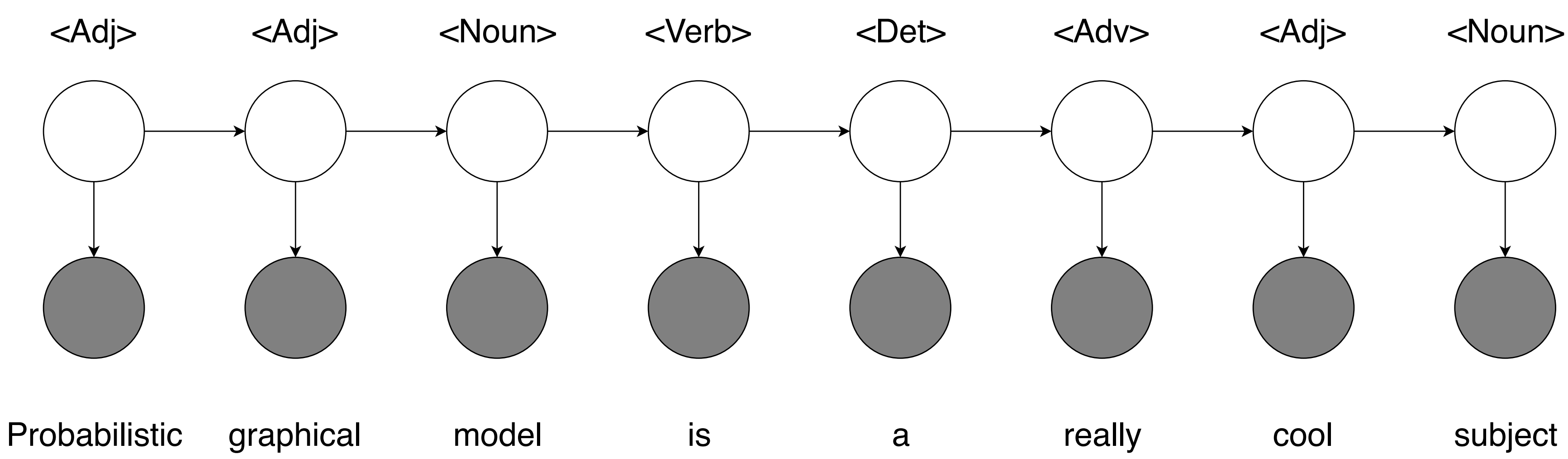


Figure: An example of part-of-speech tagging with hidden Markov model.

Application: Information Theory and Communication

- Probabilistic modeling of noisy communication channel;
- Turbo code, low-density parity check, etc. can be modeled as **factor graphs**;
- **Belief propagation** $\rightsquigarrow$ near Shannon-limit performance[2];
- Widely used for communication protocols such as 3G/4G/5G or Wi-Fi.

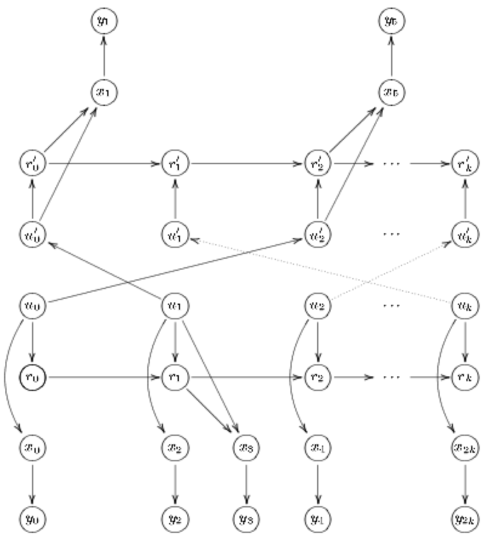


Figure: Telecommunication protocols and illustration of turbo code[3,4].

[2]MacKay, *Information theory, inference and learning algorithms*.

[3]`https://en.wikipedia.org/wiki/Wi-Fi` and `https://en.wikipedia.org/wiki/5G`, accessed on Feb. 20th, 2019.

[4]Lauritzen, “Some modern applications of graphical models”.

Application: Statistical Physics

- Modeling with **Markov random field**.
- Source of inspiration for various inference techniques:
  mean field, **simulated annealing**, **generalized belief propagation**, etc.

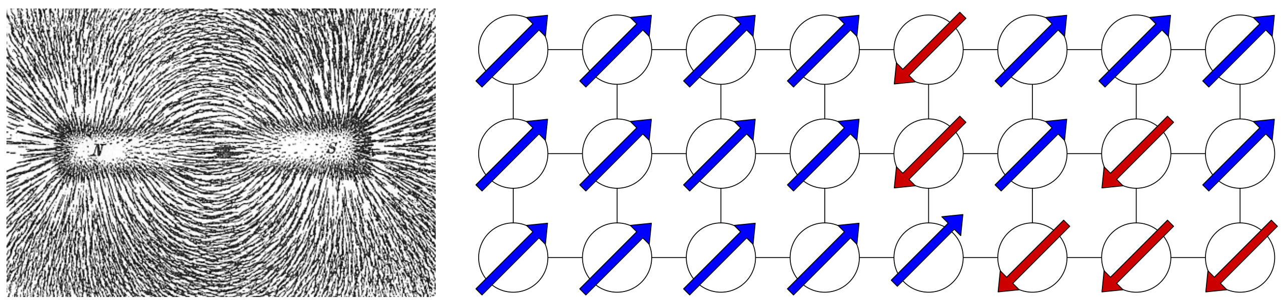


Figure: Illustration of Ising model for ferromagnet, left image from Wikipedia[5].

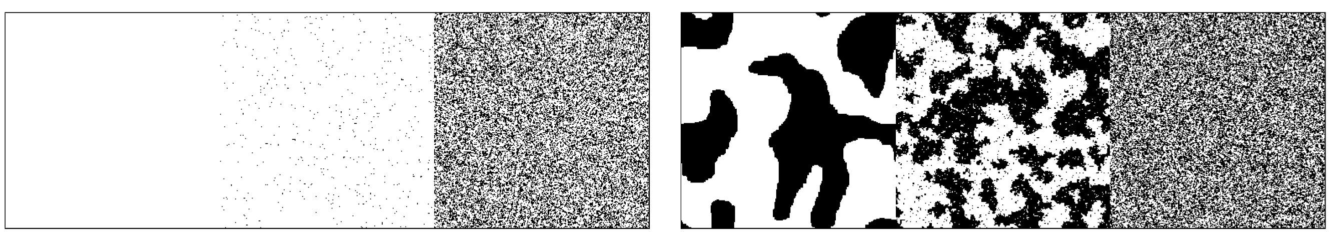


Figure: 2D Ising grid at 3 temperatures with (left) or without (right) external magnetic field[6].

[5] `https://en.wikipedia.org/wiki/Ising_model`, accessed on Feb. 20th, 2019.

[6] Generated from `https://mattbierbaum.github.io/ising.js/`, accessed on Feb. 20th, 2019.

Application: Traffic Modeling and Estimation

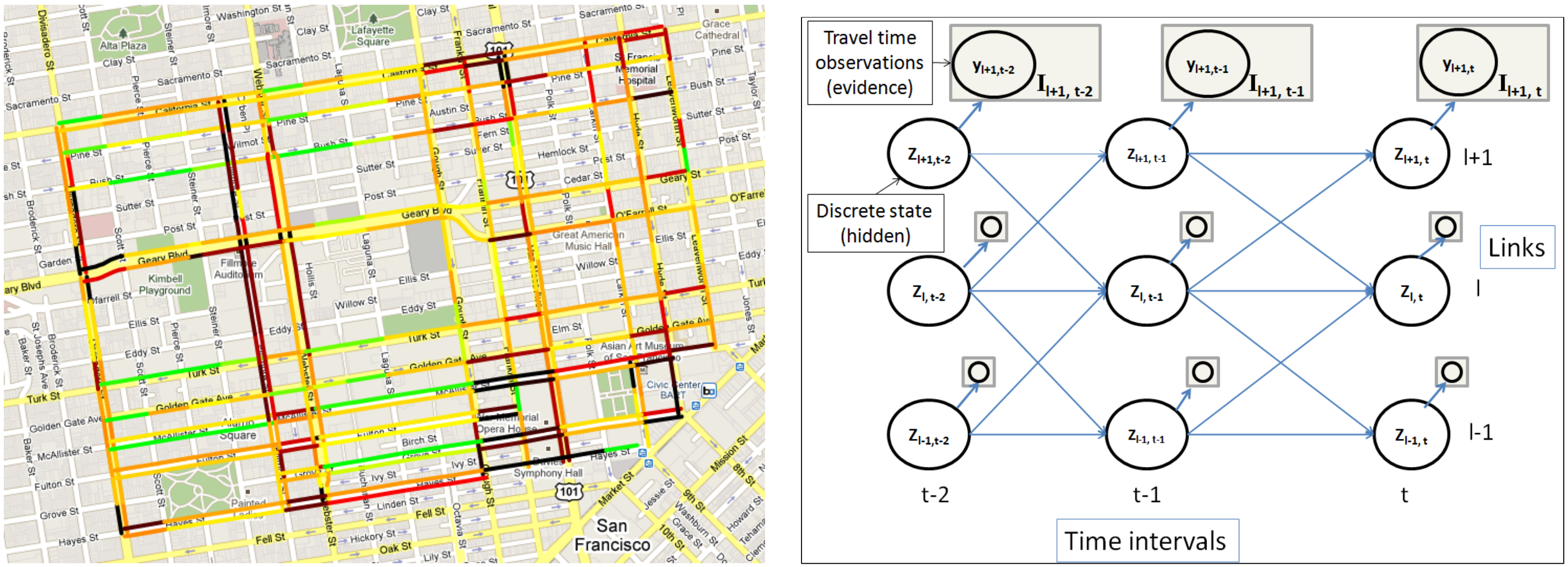


Figure: Traffic modeling and estimation with the help of coupled hidden Markov model[7].

[7] Herring, "Real-time traffic modeling and estimation with streaming probe data using machine learning".

Applications in Computer Vision

- Image data can be represented by **Markov random field**.
- Graphical model has been applied to a variety of vision tasks:

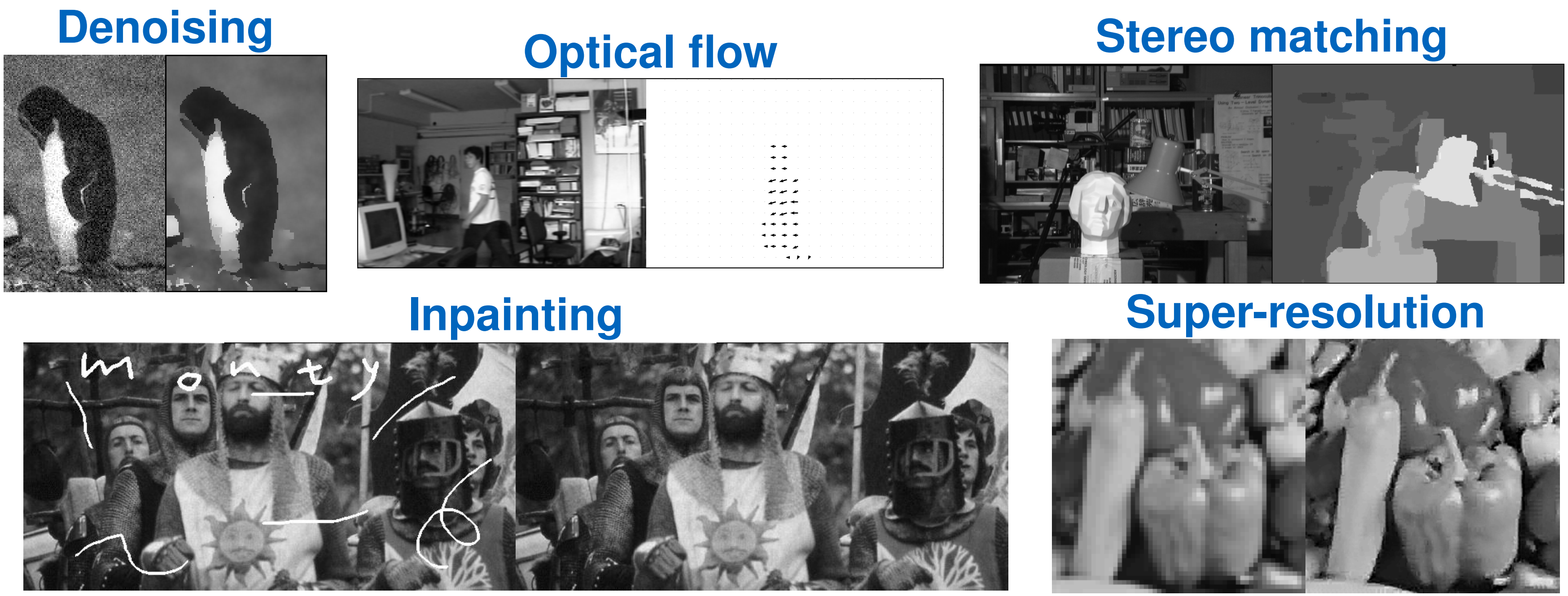


Figure: Various examples of computer vision tasks handled by graphical model[8].

[8] Felzenszwalb and Huttenlocher, “Efficient belief propagation for early vision”; Levin et al., “Learning how to inpaint from global image statistics”; Tappen et al., “Efficient graphical models for processing images”.

Application in CV: Odometry and SLAM

- A classic algorithm for odometry and navigation: **(extended) Kalman filter**.

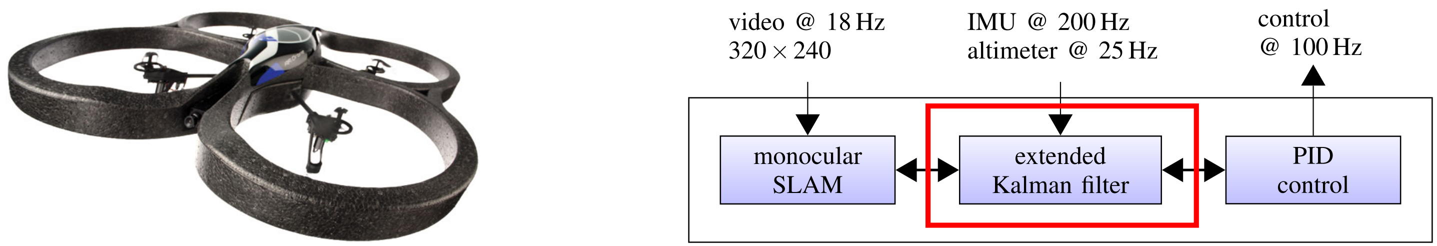


Figure: Extended Kalman filter for navigation of quadrocopter[9].

- Useful for modeling sequential data in general (e.g. sensor fusion).

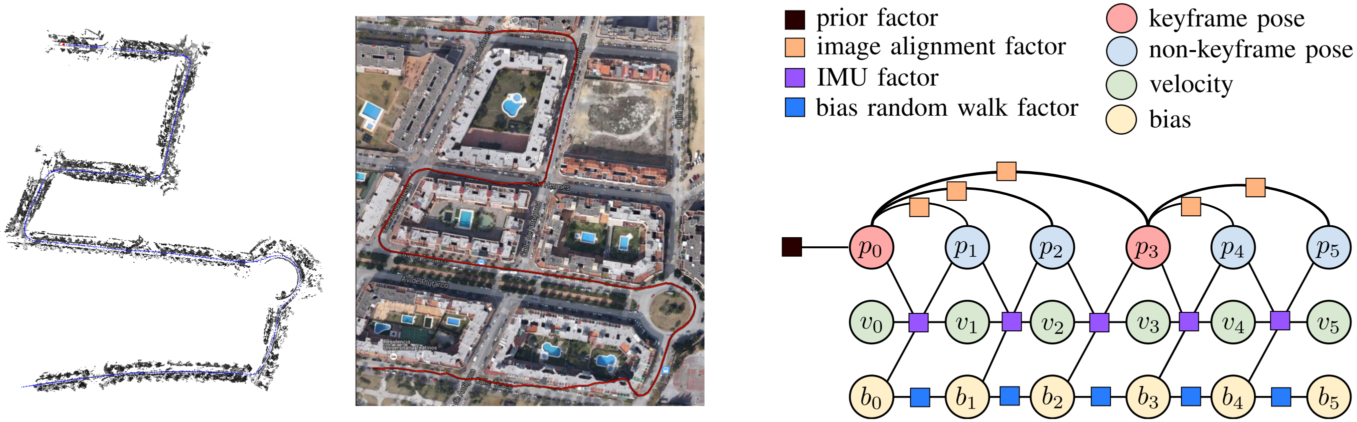
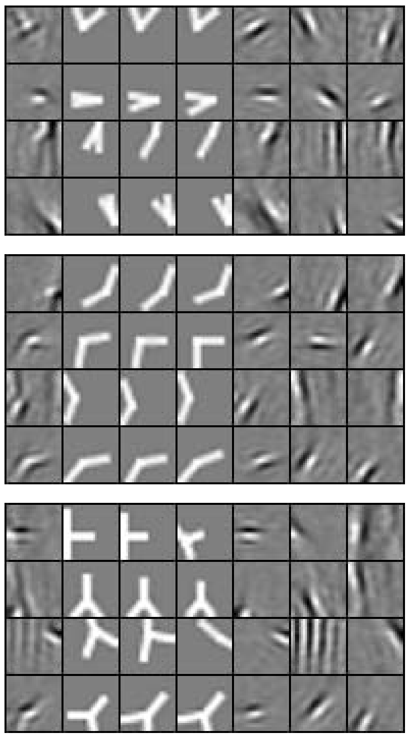


Figure: Factor graph representing the visual-inertial odometry optimization problem[10].

[9] Engel et al., “Camera-Based Navigation of a Low-Cost Quadrocopter”.
[10] Usenko et al., “Direct Visual-Inertial Odometry with Stereo Cameras”.

More applications in CV

(i) Generative modeling; (ii) Structured prediction.



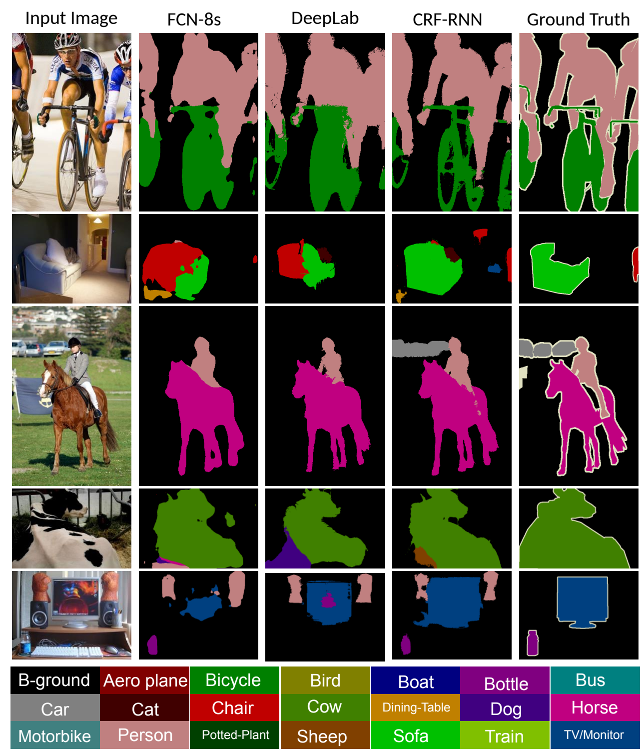


Figure: Graphical model for unsupervised learning[11] (left) and semantic segmentation[12] (right).

[11] Lee et al., "Sparse Deep Belief Net Model for Visual Area V2".

[12] Zheng et al., "Conditional Random Fields As Recurrent Neural Networks".