Graphical Models in Computer Vision

Andreas Geiger

Max Planck Institute for Intelligent Systems
Perceiving Systems

April 11, 2016
Organization
Team

Javier Romero  Andreas Geiger  Gerard Pons-Moll  Joel Janai  Naureen Mahmood
Organization

- Lecture: 2 hours/week
  - Mon: 12:15 – 14:00, Room A301
- Exercises: 2 hours/week
  - Mon: 14:00 – 16:00, Room A301
- Exception
  - 25.4.: Kleiner Hörsaal, Sand 6/7
- Course web page: http://cv.is.tue.mpg.de/
  - Slides
  - Pointers to Books and Papers
  - Homework assignments
- Mailing list http://groups.google.com/d/forum/cv-is
  - Please register!
Exercises & Exam

- Credits: 4 LP (2+2)
- Exercises:
  - Goal: Understand theory and transfer into computer experiments
  - Work in teams of up to two
  - Pen and paper exercises
  - Computing exercises
    - Will use Linux & Python
    - We provide a VirtualBox (webpage)
    - Would a brief Python tutorial be useful?
- Exam
  - Oral exam
  - English or german
  - 50 % of exercise points required!
Questions on organizational part?
Topics & Materials
Graphical Models...

- Models
- Inference
- Learning

... in Computer Vision

- Image Denoising
- Human Pose Estimation
- Human Body Models
- Stereo
- Optical Flow
- Image Segmentation
- Object Detection
### Syllabus

<table>
<thead>
<tr>
<th>Date</th>
<th>Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>11.04.2016</td>
<td>Introduction</td>
</tr>
<tr>
<td>18.04.2016</td>
<td>Graphical Models 1</td>
</tr>
<tr>
<td>25.04.2016</td>
<td>Graphical Models 2 (Sand 6/7)</td>
</tr>
<tr>
<td>02.05.2016</td>
<td>Graphical Models 3</td>
</tr>
<tr>
<td>09.05.2016</td>
<td>Graphical Models 4</td>
</tr>
<tr>
<td>23.05.2016</td>
<td>Body Models 1</td>
</tr>
<tr>
<td>30.05.2016</td>
<td>Body Models 2</td>
</tr>
<tr>
<td>06.06.2016</td>
<td>Body Models 3</td>
</tr>
<tr>
<td>13.06.2016</td>
<td>Body Models 4</td>
</tr>
<tr>
<td>20.06.2016</td>
<td>Stereo</td>
</tr>
<tr>
<td>27.06.2016</td>
<td>Optical Flow</td>
</tr>
<tr>
<td>04.07.2016</td>
<td>Segmentation</td>
</tr>
<tr>
<td>11.07.2016</td>
<td>Object Detection 1</td>
</tr>
<tr>
<td>18.07.2016</td>
<td>Object Detection 2</td>
</tr>
</tbody>
</table>
Main Book for Graphical Model Part

- Available online **for free**
- Comes with graphical model toolbox (for Matlab)
For the curious ones ...


Links are available on the course website.
Computer Vision References

- Szeliski, **Computer Vision: Algorithms and Applications**
- Hartley & Zisserman, **Multiple View Geometry in Computer Vision**
- Bernd Jähne, **Digital Image Processing and Image Formation**

Links are available on the course website.
Introduction
Why is Visual Perception hard?
Why is Visual Perception hard?

What we see

What the computer sees
Why is Computer Vision hard?

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
PROJECT MAC

Artificial Intelligence Group
Vision Memo. No. 100.

July 7, 1966

THE SUMMER VISION PROJECT
Seymour Papert

The summer vision project is an attempt to use our summer workers effectively in the construction of a significant part of a visual system. The particular task was chosen partly because it can be segmented into sub-problems which will allow individuals to work independently and yet participate in the construction of a system complex enough to be a real landmark in the development of "pattern recognition".
Why is Visual Perception hard?

Slide credits: Antonio Torralba
Why is Visual Perception hard?

Slide credits: Antonio Torralba
Why is Visual Perception hard?

Slide credits: Antonio Torralba
Why is Visual Perception hard?
Why is Visual Perception hard?
Intelligent Systems require Robust Vision

- Feature invariance
- Good prior
- Tractable representations
- Efficient learning and inference
- Model uncertainty
Some Examples from our Lab

https://ps.is.tuebingen.mpg.de/research
Probability Theory Review
Brief Review

- A random variable $X$ can take values from some discrete set of outcomes $\mathcal{X}$ (think six-sided dice)
- We usually use the short-hand notation
  
  $$p(x) \text{ for } p(X = x) \in [0, 1]$$

  for the probability that $X$ takes value $x$
- With
  
  $$p(X)$$

  we denote the probability distribution over $X$
- $p(x)$ must satisfy the following conditions:
  
  $$p(x) \geq 0$$

  $$\sum_{x \in \mathcal{X}} p(x) = 1$$
Brief Review

- Joint probability (of $X$ and $Y$)

$$p(x, y) \text{ instead } p(X = x, Y = y)$$

- Conditional probability

$$p(x|y) \text{ instead } p(X = x|Y = y)$$

- Two RVs are called independent if

$$p(X = x, Y = y) = p(X = x)p(Y = y)$$
Vocabulary

- **Joint Probability**
  \[ p(x_i, y_j) = \frac{n_{ij}}{N} \]

- **Marginal Probability**
  \[ p(x_i) = \frac{c_i}{N} \]

- **Conditional Probability**
  \[ p(y_j | x_i) = \frac{n_{ij}}{c_i} \]

![Joint Probability Table]

\[ c_i = \sum_{j} n_{ij} \]
\[ N = \sum_{ij} n_{ij} \]
The Rules of Probability

- **Sum rule**
  \[ p(X) = \sum_{y \in Y} p(X, Y = y) \]

  we “marginalize out \( y \)”. \( p(X = x) \) is also called a marginal probability

- **Product Rule**
  \[ p(X, Y) = p(Y|X)p(X) \]

- **And as a consequence: Bayes Theorem**
  \[ p(Y|X) = \frac{p(X|Y)p(Y)}{p(X)} \]
Probability Densities

- Now $X$ is a **continuous** random variable, e.g. taking values in $\mathbb{R}$
- Probability that $X$ takes a value in the interval $(a, b)$ is

$$ p(X \in (a, b)) = \int_a^b p(x) \, dx $$

and we call $p(x)$ the **probability density over** $x$
Probability Densities

- $p(x)$ must satisfy the following conditions

\[
p(x) \geq 0 \quad \text{and} \quad \int_{-\infty}^{\infty} p(x) \, dx = 1
\]

- The probability that $x$ lies in $(-\infty, z)$ is given by the \textbf{cumulative distribution function}

\[
P(z) = \int_{-\infty}^{z} p(x) \, dx
\]
Probability Densities

Probability density of a continuous variable

$p(x)$

$P(x)$

$\delta x$
Illustration

Joint, marginal, conditional probability
Expectation and Variances

- **Expectation**

\[
\mathbb{E}[f] = \sum_{x \in \mathcal{X}} p(x)f(x)
\]

\[
\mathbb{E}[f] = \int_{x \in \mathcal{X}} p(x)f(x) \, dx
\]

- Sometimes we denote the distribution that we take the expectation over as a subscript, eg

\[
\mathbb{E}_p[f] = \sum_{x \in \mathcal{X}} p(x)f(x)
\]

- **Variance**

\[
\text{var}[f] = \mathbb{E} \left[ (f(x) - \mathbb{E}[f(x)])^2 \right]
\]
Structured Prediction
Standard Regression:

\[ f : \mathcal{X} \rightarrow \mathbb{R} \]

- inputs \( \mathcal{X} \) can be any kind of objects
  - images, text, audio, sequence of amino acids, ...
- output \( y \) is a real number
  - classification, regression, density estimation, ...

Structured Output Learning:

\[ f : \mathcal{X} \rightarrow \mathcal{Y} \]

- inputs \( \mathcal{X} \) can be any kind of objects
- outputs \( y \in \mathcal{Y} \) are complex (structured) objects
  - images, parse trees, folds of a protein, ...
What is structured output prediction?

**Ad hoc definition:** predicting *structured* outputs from input data
(in contrast to predicting just a single number, like in classification or regression)

- Natural Language Processing:
  - Automatic Translation (output: sentences)
  - Sentence Parsing (output: parse trees)

- Bioinformatics:
  - Secondary Structure Prediction (output: bipartite graphs)
  - Enzyme Function Prediction (output: path in a tree)

- Speech Processing:
  - Automatic Transcription (output: sentences)
  - Text-to-Speech (output: audio signal)

- Robotics:
  - Planning (output: sequence of actions)
Graphical Models...

- Models
- Inference
- Learning

... in Computer Vision

- Object Detection
- Human Pose Estimation
- Optical Flow
- Stereo
- Image Denoising
- Segmentation
- Semantic Segmentation
- Image Stitching
- Tracking

This is the language ...

... for these problems.
Example: Human Pose Estimation

Given an image, where is a person and how is it articulated?

\[ f : \mathcal{X} \rightarrow \mathcal{Y} \]

Image \( x \), but what is human pose \( y \in \mathcal{Y} \) precisely?
Human Pose $\mathcal{Y}$

- **Body Part:** $y_{\text{head}} = (u, v, \theta)$ where $(u, v)$ center, $\theta$ rotation
  - $(u, v) \in \{1, \ldots, M\} \times \{1, \ldots, N\}$, $\theta \in \{0, 45^\circ, 90^\circ, \ldots\}$

- **Entire Body:** $y = (y_{\text{head}}, y_{\text{torso}}, y_{\text{left-lower-arm}}, \ldots) \in \mathcal{Y}$
Human Pose $\mathcal{Y}$

Image $x \in \mathcal{X}$

Example $y_{\text{head}}$

Head detector

- **Idea:** Have a head classifier (SVM, Random Forest, NN, ...)

$$\psi(y_{\text{head}}, x) \in \mathbb{R}^+$$

- Evaluate everywhere and record score
- Repeat for all body parts
Human Pose Estimation

Image $x \in \mathcal{X}$

Prediction $y^* \in \mathcal{Y}$

- Compute

$$y^* = (y^*_{head}, y^*_{torso}, \cdots) = \arg\max_{y_{head}, y_{torso}, \cdots} \psi(y_{head}, x) \psi(y_{torso}, x) \cdots$$

$$= (\arg\max_{y_{head}} \psi(y_{head}, x), \arg\max_{y_{torso}} \psi(y_{torso}, x), \cdots)$$

- Great! Problem solved!?
Human Pose Estimation

Image $x \in \mathcal{X}$

Prediction $y^* \in \mathcal{Y}$

▶ Compute

$$y^* = (y^*_\text{head}, y^*_\text{torso}, \cdots) = \arg\max_{y^*_\text{head}, y^*_\text{torso}, \cdots} \psi(y^*_\text{head}, x) \psi(y^*_\text{torso}, x) \cdots$$

$$= (\arg\max_{y^*_\text{head}} \psi(y^*_\text{head}, x), \arg\max_{y^*_\text{torso}} \psi(y^*_\text{torso}, x), \cdots)$$

▶ Great! Problem solved!?
Idea: Connect Body Parts

- Ensure *head* is on top of *torso*

\[ \psi(y_{\text{head}}, y_{\text{torso}}) \in \mathbb{R}_+ \]

- Compute

\[ y^* = \arg\max_{y_{\text{head}}, y_{\text{torso}}, \ldots} \psi(y_{\text{head}}, x)\psi(y_{\text{torso}}, x)\psi(y_{\text{head}}, y_{\text{torso}}) \ldots \]

Problem? Does not decompose anymore!
The General Recipe

Structured output function: $\mathcal{X} = \text{anything} \rightarrow \mathcal{Y} = \text{anything}$

1) Define auxiliary function $g : \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}$:

\[
e.g. \quad g(x, y) = \prod_i \psi_i(y_i, x) \prod_{i \sim j} \psi_{ij}(y_i, y_j, x)
\]

2) Obtain $f : \mathcal{X} \rightarrow \mathcal{Y}$ by maximization:

\[
f(x) = \text{argmax}_{y \in \mathcal{Y}} g(x, y)
\]
A Probabilistic View

Computer Vision problems usually deal with *uncertain* information

- Incomplete information (observe static images, projections, etc)
- Annotation is ”noisy” (wrong or ambiguous cases)

*Uncertainty* is captured by (conditional) probability distributions: \( p(y|x) \)

- for input \( x \in \mathcal{X} \), how *likely* is \( y \in \mathcal{Y} \) the correct output?

We can also phrase this as

- what’s the probability of observing \( y \) given \( x \)?
- how strong is our *belief* in \( y \) if we know \( x \)?
A Probabilistic View on $f : \mathcal{X} \rightarrow \mathcal{Y}$

Structured output function $\mathcal{X} = \text{anything} \rightarrow \mathcal{Y} = \text{anything}$

We need to define an auxiliary function, $g : \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}$.

$$g(x, y) := p(y|x).$$

e.g.

Then maximization

$$f(x) = \arg\max_{y \in \mathcal{Y}} g(x, y) = \arg\max_{y \in \mathcal{Y}} p(y|x)$$

becomes maximum a posteriori (MAP) prediction.

**Interpretation:** The MAP estimate $y \in \mathcal{Y}$, is the most probable value (there can be multiple).
Probability Distributions

\[ \forall y \in \mathcal{Y} \quad p(y) \geq 0 \quad \text{(positivity)} \]
\[ \sum_{y \in \mathcal{Y}} p(y) = 1 \quad \text{(normalization)} \]

Example: binary (”Bernoulli”) variable \( y \in \mathcal{Y} = \{0, 1\} \)
- 2 values,
- 1 degree of freedom
Conditional Probability Distributions

∀x ∈ X ∀y ∈ Y  p(y|x) ≥ 0  (positivity)

∀x ∈ X  ∑_{y ∈ Y} p(y|x) = 1  (normalization w.r.t. y)

For example: binary prediction X = \{images\}, y ∈ Y = \{0, 1\}

- each x: 2 values, 1 d.o.f.
  → two functions
Multi-class prediction, \( y \in \mathcal{Y} = \{1, \ldots, K\} \)

- each \( x \): \( K \) values, \( K - 1 \) d.o.f.
  \( \rightarrow K - 1 \) functions
- or 1 vector-valued function with \( K - 1 \) outputs

Typically: \( K \) functions, plus explicit normalization

Example: predicting the center point of an object

\[ y \in \mathcal{Y} = \{(1, 1), \ldots, (width, height)\} \]
- for each \( x \): \( |\mathcal{Y}| = W \cdot H \) values,

\[ y = (y_1, y_2) \in \mathcal{Y}_1 \times \mathcal{Y}_2 \] with
\[ \mathcal{Y}_1 = \{1, \ldots, width\} \) and
\[ \mathcal{Y}_2 = \{1, \ldots, height\}. \]
- each \( x \): \( |\mathcal{Y}_1| \cdot |\mathcal{Y}_2| = W \cdot H \) values,
Structured objects: predicting $M$ variables jointly

$\mathcal{Y} = \{1, K\} \times \{1, K\} \cdots \times \{1, K\}$

For each $x$:

- $K^M$ values, $K^M - 1$ d.o.f.
- $K^M$ functions

Example: Object detection with variable size bounding box

$\mathcal{Y} \subset \{1, \ldots, W\} \times \{1, \ldots, H\}$

$\times \{1, \ldots, W\} \times \{1, \ldots, H\}$

$y = (\text{left}, \text{top}, \text{right}, \text{bottom})$

For each $x$:

- $\frac{1}{4} W(W-1)H(H-1)$ values
  (millions to billions...)

\[ p(y|x) \]

\[ 19 \quad \cdots \quad 36 \quad \cdots \quad 22 \]
Example: image denoising

\[ \mathcal{Y} = \{640 \times 480 \text{ RGB images}\} \]

For each \( x \):
- \( 16777216^{307200} \) values in \( p(y|x) \)
- \( \geq 10^{2000000} \) functions
- How many atoms in universe? too much!

We cannot consider all possible distributions, we must impose structure.
Decision Theory
Digit classification

- Classify digits “a” versus “b”

The digits “a” and “b”

- **Goal:** classify new digits such that probability of error is minimized
Digit classification - Priors

Prior Distribution?

- How often do the letters “a” and “b” occur?
- Let us assume

$$C_1 = a \quad p(C_1) = 0.75$$
$$C_2 = b \quad p(C_2) = 0.25$$

- The *prior* has to be a distribution, in particular

$$\sum_{k=1,2} p(C_k) = 1$$
Digit classification - Class conditionals

- We describe every digit using some **feature vector** $x$
  - the number of black pixels in each box
  - relation between width and height
- **Likelihood:** How likely has $x$ been generated from $p(x \mid a)$ or $p(x \mid b)$?
Digit classification

- Which class should we assign $x$ to?
- Class a
Digit classification

- Which class should we assign \( x \) to?
- Class b
Digit classification

Which class should we assign \( x \) to?

The answer?
Bayes Theorem

- How do we formalize this?
- We already mentioned Bayes Theorem

\[ p(Y|X) = \frac{p(X|Y)p(Y)}{p(X)} \]

- Now we apply it:

\[ p(C_k|x) = \frac{p(x|C_k)p(C_k)}{p(x)} = \frac{p(x|C_k)p(C_k)}{\sum_j p(x|C_j)p(C_j)} \]
Bayes Theorem

- Some terminology! Repeated from last slide:

\[
p(C_k|x) = \frac{p(x|C_k)p(C_k)}{p(x)} = \frac{p(x|C_k)p(C_k)}{\sum_j p(x|C_j)p(C_j)}
\]

- We use the following names

\[
\text{Posterior} = \frac{\text{Likelihood } \times \text{ Prior}}{\text{Normalization Factor}}
\]

- Normalization Factor is also called the **Partition Function** or **Evidence** (commonly denoted with the symbol ‘Z’).
Bayes Theorem

\[ \text{Posterior} = \frac{\text{Likelihood} \times \text{Prior}}{\text{Normalization Factor}} \]
How to decide?

- Two class problem $C_1, C_2$, plotting Likelihood $\times$ Prior

- What is the probability of making an error?
Minimizing the Error

\[ p(\text{error}) = p(x \in R_2, C_1) + p(x \in R_1, C_2) \]
\[ = p(x \in R_2 | C_1)p(C_1) + p(x \in R_1 | C_2)p(C_2) \]
\[ = \int_{R_2} p(x | C_1)p(C_1)dx + \int_{R_1} p(x | C_2)p(C_2)dx \]
General Loss Functions

- So far we considered misclassification error only
- This is also referred to as 0/1 loss
- Now suppose we are given a more general loss function

\[ \Delta : \mathcal{Y} \times \mathcal{Y} \rightarrow \mathbb{R}_+ \]

\[ (y, \hat{y}) \mapsto \Delta(y, \hat{y}) \]

- How do we read this?
- \( \Delta(y, \hat{y}) \) is the cost we have to pay if \( y \) is the true class, but we predict \( \hat{y} \) instead
Example: Predicting Cancer

- General loss function:

\[ \Delta : \mathcal{Y} \times \mathcal{Y} \rightarrow \mathbb{R}_+ \]

\( (y, \hat{y}) \mapsto \Delta(y, \hat{y}) \)

- Given: X-Ray image
  - Question: Cancer yes or no?
  - Should we have a medical doctor check the patient?

- For discrete sets \( \mathcal{Y} \) this is a loss matrix. How does it look?

- Loss function:

<table>
<thead>
<tr>
<th></th>
<th>cancer</th>
<th>normal</th>
</tr>
</thead>
<tbody>
<tr>
<td>cancer</td>
<td>0</td>
<td>1000</td>
</tr>
<tr>
<td>normal</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>
Digit Classification

- Which class should we assign \( x \) to? \( (p(a) = p(b) = 0.5) \)
- The answer
- It depends on the loss
Minimizing Expected Error

- But we do not know the correct class $y$
- The expected error for $x$ (averaged over all decisions):

$$\mathbb{E}[\Delta] = \sum_{k=1, \ldots, K} \sum_{j=1, \ldots, K} \int_{R_j} \Delta(C_k, C_j) p(x, C_k) dx$$
Minimizing Expected Error

- But we do not know the correct class $y$
- The expected error for $x$ (averaged over all decisions):

$$\mathbb{E}[\Delta] = \sum_{k=1,\ldots,K} \sum_{j=1,\ldots,K} \int_{R_j} \Delta(C_k, C_j)p(x, C_k)dx$$

- And how do we predict, given an $x$? Decide on one $y$!

$$y^* = \arg\min_{y \in \mathcal{Y}} \sum_{k=1,\ldots,K} \Delta(C_k, y)p(C_k|x)$$

$$= \arg\min_{y \in \mathcal{Y}} \mathbb{E}_{p(\cdot|x)}[\Delta(\cdot, y)]$$
Inference and Decision

- We broke down the process into two steps
  - Inference: obtaining the probabilities $p(C_k|x)$
  - Decision: Obtain optimal class assignment
- The probabilities $p(·|x)$ represent our belief of the world
- The loss $\Delta$ tells us what to do with it!
- 0/1 loss implies deciding for max probability (exercise)
Three approaches to solve decision problems

1. **Generative models**: infer the class conditionals

   \[ p(x|C_k), \quad k = 1, \ldots, K \]

   then combine using Bayes Theorem

2. **Discriminative models**: infer posterior probabilities directly

   \[ p(C_k|x) \]

3. Find discriminative function minimizing expected loss \( \Delta \)

   \[ f : \mathcal{X} \rightarrow \{1, \ldots, K\} \]

Let’s discuss these options …
Generative Models

Pros:
- The name *generative* is because we explain the *generative* process of the data
- Intuitive, “understand” your process
- We can generate samples $x$ from $p(x)$

Cons:
- With high dimensionality of $x \in \mathcal{X}$ we need large training set to determine the class-conditionals
- We may just not be interested in all quantities
Discriminative Models

Pros:
- No need to model $p(x|C_k)$
  $\Rightarrow$ easier

Cons:
- No access to model $p(x|C_k)$
Discriminative Functions

Pros:

- One integrated system
- Directly estimate the quantity of interest $f(x)$
- *When solving a problem of interest, do not solve a harder / more general problem as an intermediate step.* [Vladimir Vapnik]

Cons:

- Need $\Delta$ during training time
- Revision of $\Delta$ requires re-learning
- No probabilities, no uncertainty, no reject?
- Prominent example: SVMs
Next Time ...