Autonomous Vision Group
Prof. Dr.-Ing. Andreas Geiger

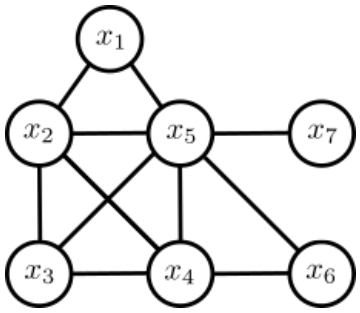
Eberhard Karls
Universität
Tübingen

Computer Vision

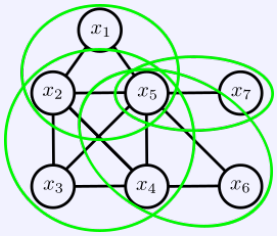
Exercise 3 – Graphical Models

1 Pen and Paper

1.1 Markov Random Fields



a) For the undirected graph $\mathcal{G}$ above, draw a circle around each maximal clique. What is the Markov Random Field $p(x_1, \ldots, x_7)$ of the graph?



$$p(x_1, \ldots, x_7) = \frac{1}{Z}\phi_1(x_1, x_2, x_5)\phi_2(x_2, x_3, x_4, x_5)\phi_3(x_4, x_5, x_6)\phi_4(x_5, x_7)$$

b) Gobal Markov Property: Given $\mathcal{S} = \{x_2, x_5, x_6\}$, define disjoint sets $\mathcal{A}, \mathcal{B}$ such that $\mathcal{A} \perp\!\!\!\perp \mathcal{B} \mid \mathcal{S}$. How many of such pairs of sets can you find?
Further, what is the Markov Blanket of x_4?

Instructions on how to count pairs of $\mathcal{A}$ and $\mathcal{B}$:
$\mathcal{A}$ and $\mathcal{B}$ cannot be empty.
Swapping $\mathcal{A}$ and $\mathcal{B}$ does not count as a new pair.
$\mathcal{A} \cup \mathcal{B} \cup \mathcal{S}$ must contain all variables in the graph.

$\mathcal{A} = \{x_1, x_7\}, \mathcal{B} = \{x_3, x_4\}$
$\mathcal{A} = \{x_1, x_3, x_4\}, \mathcal{B} = \{x_7\}$
$\mathcal{A} = \{x_1\}, \mathcal{B} = \{x_3, x_4, x_7\}$
The Markov blanket is given by $ne(x_4) = \{x_2, x_3, x_5, x_6\}$.

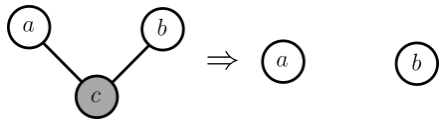
c) Local Markov Property: Which of the following statements are correct, which are wrong and why?

- $p(x_4 \mid x_1, x_2, x_3, x_5, x_6, x_7) = p(x_4 \mid x_2, x_3, x_5, x_6)$

- $x_1 \perp\!\!\!\perp x_3 \mid x_5$
- $x_2 \perp\!\!\!\perp x_6 \mid \{x_4, x_5\}$
- Marginalizing over x_5 makes x_1 and x_7 dependent.

True
False
True
True

d) Prove the conditional independence property $a \perp\!\!\!\perp b \mid c$



by showing that $p(a, b \mid c) = p(a \mid c)p(b \mid c)$.

We want to prove $p(a, b \mid c) = p(a \mid c)p(b \mid c)$.
Start with the left side of the equation:

$$p(a, b \mid c) = \frac{p(a, b, c)}{p(c)} = \frac{p(a, b, c)}{\sum_{a,b} p(a, b, c)} = \frac{\frac{1}{Z}\phi_1(a, c)\phi_2(b, c)}{\sum_{a,b} \frac{1}{Z}\phi_1(a, c)\phi_2(b, c)} = \frac{\phi_1(a, c)\phi_2(b, c)}{\sum_{a,b} \phi_1(a, c)\phi_2(b, c)} \quad (1)$$

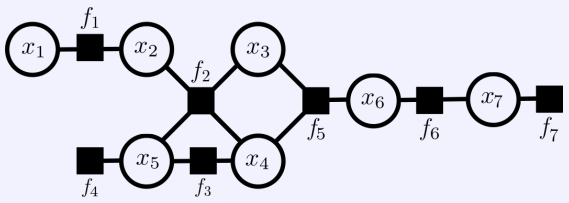
Now look at the right side of the equation:

$$\begin{aligned}
p(a \mid c)p(b \mid c) &= \frac{p(a, c)}{p(c)} \frac{p(b, c)}{p(c)} \\
&= \frac{\sum_b p(a, b, c)}{\sum_{a,b} p(a, b, c)} \cdot \frac{\sum_a p(a, b, c)}{\sum_{a,b} p(a, b, c)} \\
&= \frac{\left(\sum_b \frac{1}{Z}\phi_1(a, c)\phi_2(b, c)\right) \cdot \left(\sum_a \frac{1}{Z}\phi_1(a, c)\phi_2(b, c)\right)}{\left(\sum_{a,b} \frac{1}{Z}\phi_1(a, c)\phi_2(b, c)\right)^2} \\
&= \frac{\frac{1}{Z}\phi_1(a, c)\left(\sum_b \phi_2(b, c)\right) \cdot \frac{1}{Z}\phi_2(b, c)\left(\sum_a \phi_1(a, c)\right)}{\frac{1}{Z^2}\left(\sum_{a,b} \phi_1(a, c)\phi_2(b, c)\right)^2} \\
&= \frac{\phi_1(a, c)\phi_2(b, c)\left(\sum_a \sum_b \phi_1(a, c)\phi_2(b, c)\right)}{\left(\sum_{a,b} \phi_1(a, c)\phi_2(b, c)\right)^2} \\
&= \frac{\phi_1(a, c)\phi_2(b, c)}{\sum_{a,b} \phi_1(a, c)\phi_2(b, c)} \\
&\overset{Eq.\ (??)}{=} p(a, b \mid c)
\end{aligned}$$

1.2 Factor Graphs

a) Draw the factor graph for function

$$f(x_1, x_2, x_3, x_4, x_5, x_6) = f_1(x_1, x_2)f_2(x_2, x_3, x_4, x_5)f_3(x_4, x_5)f_4(x_5)f_5(x_3, x_4, x_6)f_6(x_6, x_7)f_7(x_7).$$



b) Given $\mathcal{X} = \{x_1, \ldots, x_6\}$ with $x_i \in \{0, 1\}\ \forall i$, a Markov Random Field is given by the distribution

$$p(x_1, \ldots, x_6) = \frac{1}{Z}\phi_1(x_1)\phi_2(x_1, x_2, x_3, x_4)\phi_3(x_3, x_5, x_6).$$

Define the potentials as

$$\begin{aligned} \phi_1(x_1) &= [x_1 = 1] \\ \phi_2(x_1, x_2, x_3, x_4) &= [x_1 = x_2] \cdot [x_3 = x_4] \\ \phi_3(x_3, x_5, x_6) &= 2x_3 + x_5 + x_6 \end{aligned}$$

Calculate the value of the partition function Z. Is $p(x_1, \ldots, x_6)$ a Gibbs distribution? Explain your answer.

The partition function or normalization coefficient is given by

$$\begin{aligned} Z &= \sum_{\mathcal{X}} \prod_{k=1}^{K} \phi_k(\mathcal{X}_k) \\ &= \sum_{x_1, \ldots, x_6} \phi_1(x_1)\phi_2(x_1, x_2, x_3, x_4)\phi_3(x_3, x_5, x_6). \end{aligned}$$

The only non-zero terms are the ones where $x_1 = x_2 = 1$ and $x_3 = x_4$. That reduces the sum to 8 terms:

$x_1x_2x_3x_4x_5x_6$	$\phi_1(x_1)\phi_2(x_1, x_2, x_3, x_4)\phi_3(x_3, x_5, x_6)$
111100	2
111110	3
111101	3
111111	4
110000	0
110010	1
110001	1
110011	2

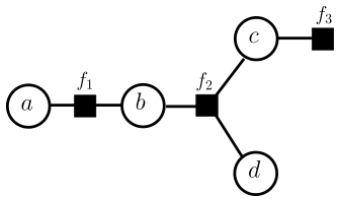
Summing that up leads to $Z = 16$.
For $p(x_1, \ldots, x_6)$ to be a Gibbs distribution, all potentials must be strictly positive (not only non-negative). Therefore, $p(x_1, \ldots, x_6)$ is not a Gibbs distribution.

1.3 Belief Propagation

a) In one or two sentences, what is the nugget or key observation behind Belief Propagation?

Belief Propagation assumes a singly-connected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, which means it has $|\mathcal{V}| - 1 = \mathcal{O}(|\mathcal{V}|)$ many edges (in contrast to $|\mathcal{V}|(|\mathcal{V}| - 1)/2 = \mathcal{O}(|\mathcal{V}|^2)$ of a fully connected graph). That simplifies the computation of any marginal distribution significantly, since most potentials of the MRF only depend on a few variables.

b) Marginal Inference. Consider the Markov Random Field defined on binary variables $a, b, c, d \in \{0, 1\}$ with the following factor graph:



The potentials are given by $f_1(a,b) = [a = b]$, $f_2(b,c,d) = 0.5b + 0.3c + 0.2d$ and $f_3(c) = [c = 1]$. Compute all marginal distributions using the *Sum-Product Algorithm*. Here, please do not use the log representation for the messages μ.

Initialization: $\mu_{a\to f_1}(a) = 1$, $\mu_{f_3\to c}(c) = f_3(c)$ and $\mu_{d\to f_2}(d) = 1$.
Compute all messages towards f_2:

$$\mu_{f_1\to b}(b) = \mu_{b\to f_2}(b) = \sum_a f_1(a,b)\cdot 1 = 1$$
$$\mu_{c\to f_2}(c) = f_3(c)$$

Then get all outgoing messages from f_2:

$$\mu_{f_2\to b}(b) = \sum_{c,d} f_2(b,c,d)\cdot f_3(c)\cdot 1 = b + 0.8$$
$$\mu_{f_2\to c}(c) = \sum_{b,d} f_2(b,c,d)\cdot 1\cdot 1 = 1.2c + 1.4$$
$$\mu_{f_2\to d}(d) = \sum_{b,d} f_2(b,c,d)\cdot 1\cdot f_3(c) = 0.4d + 1.1$$

There are only 3 messages left to compute:

$$\mu_{b\to f_1}(b) = b + 0.8$$
$$\mu_{c\to f_3}(c) = 1.2c + 1.4$$
$$\mu_{f_1\to a}(a) = \sum_b f_1(a,b)\cdot \mu_{b\to f_1}(b) = a + 0.8$$

From these messages we infer the marginal distributions using $p(x) \propto \prod_{f\in ne(x)} \mu_{f\to x}(x)$ and normalization:

$$p(a) = \begin{cases} 0.308 & a = 0 \\ 0.692 & a = 1 \end{cases}$$
$$p(b) = \begin{cases} 0.308 & b = 0 \\ 0.692 & b = 1 \end{cases}$$
$$p(c) = [c = 1]$$
$$p(d) = \begin{cases} 0.423 & d = 0 \\ 0.577 & d = 1 \end{cases}$$

c) Maximum-A-Posteriori. Consider a distribution defined over binary variables $a, b, c \in \{0, 1\}$:

$$p(a,b,c) = \frac{1}{Z}\, f_1(a,b) f_2(b,c) f_3(c)$$

The factors are given as

$$f_1(a,b) = \begin{pmatrix} 0.5 & 0.1 \\ 0.1 & 0.3 \end{pmatrix}, \qquad f_2(b,c) = \begin{pmatrix} 0.2 & 0.4 \\ 0.1 & 0.3 \end{pmatrix}, \qquad f_3(c) = \begin{pmatrix} 0.5 \\ 0.5 \end{pmatrix}$$

so e.g. $f_2(b = 0, c = 1) = 0.4$. What is the most likely joint configuration $a^*, b^*, c^* = \text{argmax}_{a,b,c}\, p(a, b, c)$? Again, please use *Max-Product Algorithm* and do not use the log representation.

Initialization: $\mu_{a\to f_1}(a) = 1$ and $\mu_{f_3\to c}(c) = f_3(c)$.
From left to right compute:

$$\mu_{f_1\to b}(b) = \mu_{b\to f_2}(b) = \max_a f_1(a, b) \cdot 1 = \begin{cases} 0.5 & b = 0 \\ 0.3 & b = 1 \end{cases}$$

$$\mu_{f_2\to c}(c) = \mu_{c\to f_3}(c) = \max_b f_2(b, c) \cdot \mu_{b\to f_2}(b) = \begin{cases} 0.1 & c = 0 \\ 0.2 & c = 1 \end{cases}$$

And from right to left:

$$\mu_{c\to f_2}(c) = f_3(c)$$

$$\mu_{f_2\to b}(b) = \mu_{b\to f_1}(b) = \max_c f_2(b, c) \cdot f_3(c) = \begin{cases} 0.2 & b = 0 \\ 0.15 & b = 1 \end{cases}$$

$$\mu_{f_1\to a}(a) = \max_b f_1(a, b) \cdot \mu_{b\to f_1}(b) = \begin{cases} 0.1 & a = 0 \\ 0.045 & a = 1 \end{cases}$$

Now infer the most likely joint configuration a^*, b^*, c^*:

$$a^* = \operatorname*{argmax}_a \ \mu_{f_1\to a}(a) = 0$$
$$b^* = \operatorname*{argmax}_b \ \mu_{f_1\to b}(b) \cdot \mu_{f_2\to b}(b) = 0$$
$$c^* = \operatorname*{argmax}_c \ \mu_{f_2\to c}(c) \cdot \mu_{f_3\to c}(c) = 1$$

Alternatively, you can reach the same solution if you follow the algorithm shown in the lecture slides 52 and 53 of lecture 5, which is specialized for chain structured graphs (like the one given here).

d) Consider a pairwise Markov Random Field defined on a ring (a chain that is connected at both ends) with 100 binary variables:

$$p(x) = \phi_0(x_1, x_{100}) \prod_{i=1}^{99} \phi_i(x_i, x_{i+1})$$

Is it possible to compute $\text{argmax}_{x_1,\ldots,x_{100}}\, p(x)$ efficiently?

Hint: Think about what happens if you condition on one of the nodes.

Assume we know the most likely state of one of the nodes, say x_1, the remaining graph is singly connected and we can compute the message passing efficiently. As x_1 is a binary variable, there is only two possible state of it, $x_1 = 0$ or $x_1 = 1$.

For $x_1 = 0$, we can find the most likely state of $x_2, x_3, \cdots, x_{100}$ by max product algorithm:

$$x_2^*, x_3^*, \cdots, x_{100}^* | x_1 = 0 = \operatorname*{argmax}_{x_2,x_3,\cdots,x_{100}} \underbrace{p(x_2, \ldots, x_{100} | x_1 = 0)}_{\text{singly connected}}$$

Similarly, we can find the mosth likely state $x_2, x_3, \cdots, x_{100}$ for $x_1 = 1$.

$$x_2^*, x_3^*, \cdots, x_{100}^* | x_1 = 1 = \operatorname*{argmax}_{x_2,x_3,\cdots,x_{100}} \underbrace{p(x_2, \ldots, x_{100} | x_1 = 1)}_{\text{singly connected}}$$

Then we can plug these states to $p(x)$ and pick the state with higher potential.

1.4 Graphical Models for Multi-View Reconstruction

a) What are the advantages of using Probabilistic Graphical Models for Computer Vision tasks like Multi-View Reconstruction? And what are the challenges or drawbacks of those models?

+ Include prior knowledge into the estimation (increase robustness e.g. to textureless regions in MVS), constraints only need to be non-negative.
+ Probabilistic formulation gives a certainty estimate.
+ Non linear constraints can be incorporated.
- Accurate inference / guaranteed convergence is only given for non-loopy graphs.
- Inference in multiply connected graphs or high order potentials can get more complex and intractable.

b) In the lecture, we discussed a probabilistic dense multi-view reconstruction method using an MRF on a voxel grid. Which assumption(s) is this model based on? And in which cases are those violated?

Color constancy between pixels showing the same 3D point. Since the model works on aerial images this is violated for a couple of cases:

i) Most prominently for non-Lambertian surfaces.

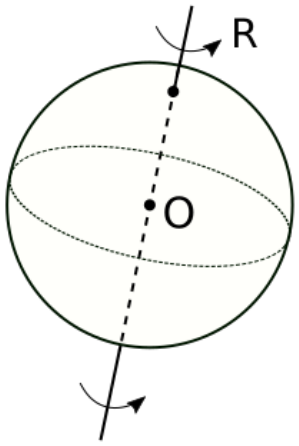
ii) But basically for all surfaces (especially if it is sunny) since the appearance depends on the view angle (wrt. to the surface normal) which is not included in the model.

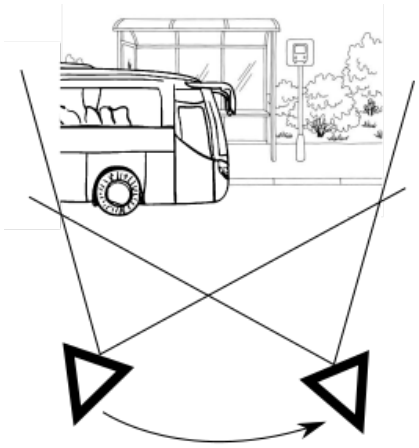
iii) Illumination changes (daytime, season).

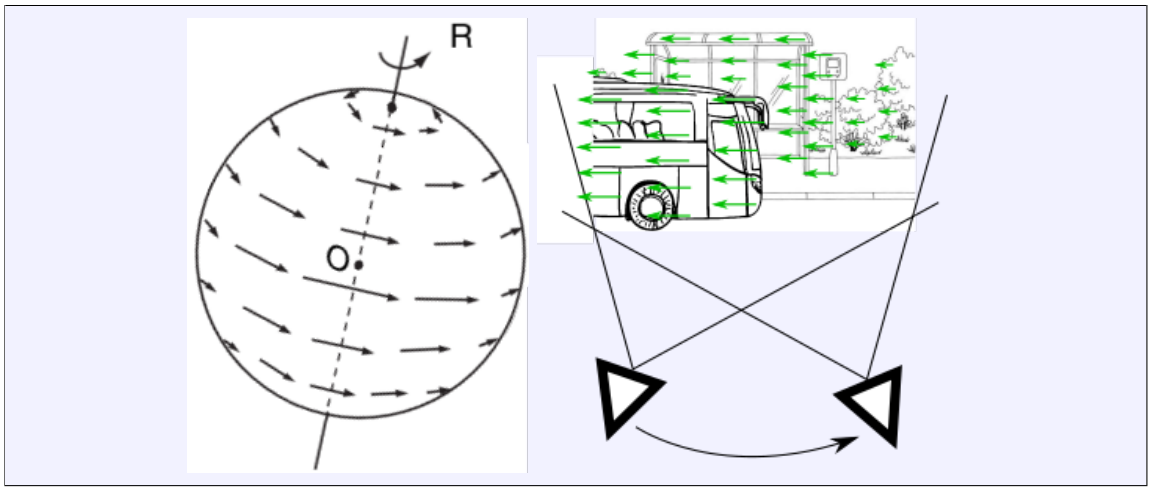
iv) Camera noise is assumed to be negligible.

1.5 Graphical Models for Optical Flow

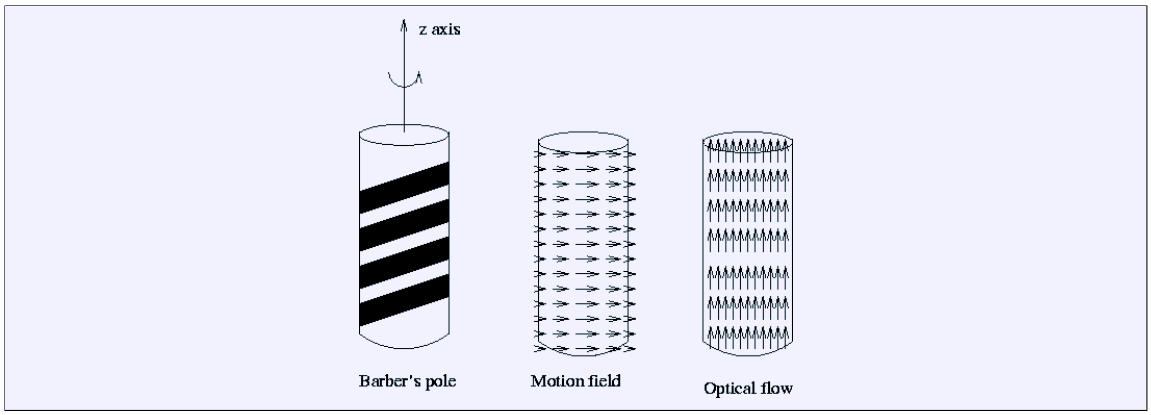
a) **Flow fields.** Draw a sketch of the optical flow fields for the following two scenarios: The first is a sphere rotating around the shown axis. Please assume the sphere to be textured. The second shows a static bus stop scene with the camera moving from left to right. Indicate with arrows the magnitude of the optical flow for different points in the scene.







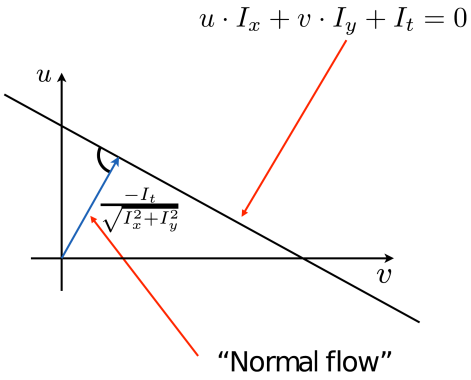
b) In the lecture the Barber Pole was discussed as an example for the aperture problem. Answer the question from the slides concerning the Barber Pole: What is the motion field? What is the optical flow field?



c) **Aperture problem.** The aperture problem arises from the brightness consistency contraint / optical flow constraint equation

$$u_{x,y} \cdot I_x(x,y) + v_{x,y} \cdot I_y(x,y) + I_t(x,y) = 0$$

This constraint can be graphically represented as follows:



Understand the relationship and explain the aperture problem using the figure above. Additionally, derive the term for the normal flow.

We determine Optical Flow based on the assumption of constant brightness

$$I(x + u(x, y), y + v(x, y), t + 1) = I(x, y, t).$$

As shown in the lecture, this constraint can be linearized using a first-order multi variable Taylor series yielding

$$u \cdot I_x + v \cdot I_y + I_t = 0 \tag{2}$$

for a single pixel (x, y). This equation is known as the brightness consistency contraint or optical flow constraint equation.

Note that this is an equation with 2 unknowns, u and v, so we can only measure the normal velocity or normal flow but no optical flow perpendicular to this direction. This ambiguity is known as the **aperture problem**. To get an estimate of the full velocity vector, we would need an additional constraint.

Graphically, Eq. (**??**) describes a line in the space of u and v which is defined by the intensity gradient of the image, as shown in the figure. To derive the component of optical flow that is in the direction of the intensity gradient, or normal velocity, we rewrite Eq. (**??**) as

$$u \cdot I_x + v \cdot I_y = -I_t \tag{3}$$

and determine the distance d of the line to the origin: First, we get the normal as

$$\mathbf{n} = \begin{pmatrix} I_x \\ I_y \end{pmatrix} \Big/ \sqrt{I_x^2 + I_y^2}.$$

Then the distance of the line to the orign is given by the length of the orthogonal projection of any point (u, v) on the line on $\mathbf{n}$:

$$d = (u, v)^T \cdot \mathbf{n} = \frac{u \cdot I_x + v \cdot I_y}{\sqrt{I_x^2 + I_y^2}} \overset{Eq.\ (??)}{=} \frac{-I_t}{\sqrt{I_x^2 + I_y^2}}.$$

2 Coding Exercises

This exercise is split into two parts: `localization.ipynb` and `denoising.ipynb`. In the first part you will implement max-product belief propagation for chain structured Markov Random Field to solve a pre-specified vehicle localization problem. In the second part the belief propagation algorithm for general Markov random fields is already implemented. You will apply this algorithm to an image denoising problem and experiment with the strength of the prior to obtain the best possible denoising result.