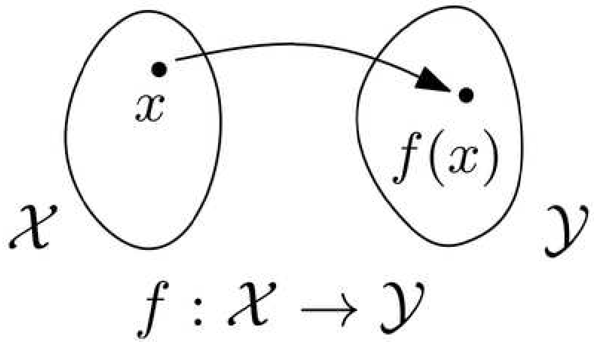
Combinatorial Optimization in Computer Vision (IN2245)

Frank R. Schmidt
Csaba Domokos

Winter Semester 2015/2016

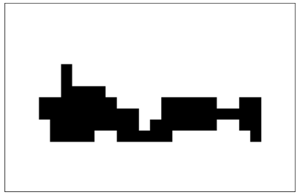
Introduction

We often need to build a model of the real world that relates *observed measurements* $x \in \mathcal{X}$ to *quantities of interest* $y \in \mathcal{Y}$.



Running example:
Recognizing man-made structures in images (i.e. binary image segmentation)

Original image

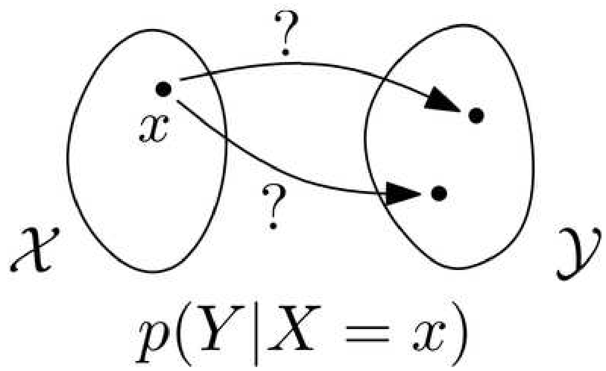


Ground truth (24×16 blocks)

We have one binary variable per 16-by-16 block of pixels.

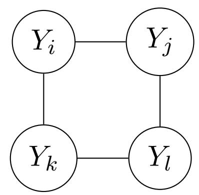
Graphical models

Probabilistic graphical models encode a joint $p(x, y)$ or conditional $p(y \mid x)$ probability distribution such that given some observations we are provided with a full probability distribution over all feasible solutions.



The graphical models allow us to encode relationships between a set of random variables using a concise language, by means of a graph.

Suppose a graph such that for each node a random variable is assigned. The random variables satisfy **conditional independence assumptions** encoded in the graph.

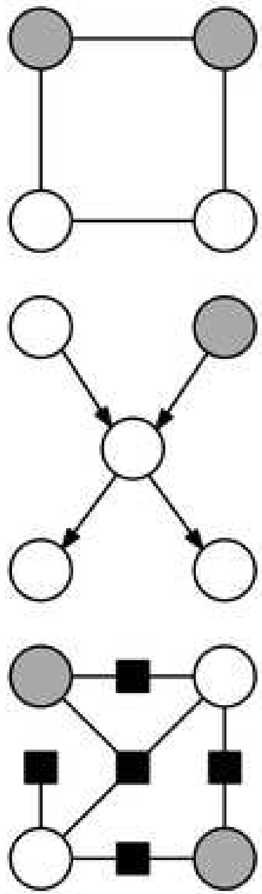


For example: The variables Y_i and Y_l are conditionally independent given Y_j, Y_k:

$$Y_i \perp\!\!\!\perp Y_l \mid Y_j, Y_k \;\Rightarrow\; p(Y_i, Y_l \mid Y_j, Y_k) = p(Y_i \mid Y_j, Y_k) p(Y_l \mid Y_j, Y_k) \; .$$

Popular classes of graphical models

- Undirected graphical models (e.g., Markov random fields)
- Directed graphical models (e.g., Bayesian networks)
- Factor graphs



We will use the following notations

- V denotes a **set of output variables** (e.g., for pixels) and the corresponding random variables are denoted by Y_i, $i \in V$
- The **output domain** $\mathcal{Y}$ is given by the product of individual variable domains $\mathcal{Y}_i$ (e.g., a single label set $\mathcal{L}$), so that $\mathcal{Y} = \bigtimes_{i \in V} \mathcal{Y}_i$
- The **input domain** $\mathcal{X}$ is application dependent (e.g., $\mathcal{X}$ is a set of images)
- The **realization** $Y = y$ means that $Y_i = y_i$ for all $i \in V$
- $G = (V, \mathcal{E})$ is an (un)directed graph, where $\mathcal{E}$ encodes the conditional independence assumption

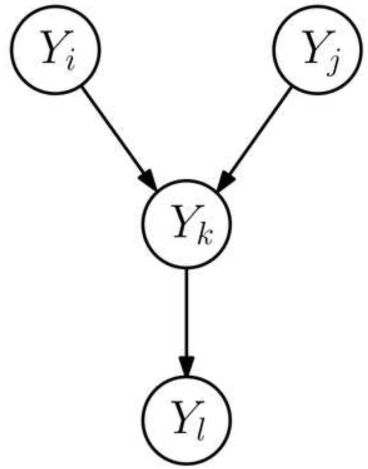
Bayesian networks

Assume a **directed, acyclic** graphical model $G = (V, \mathcal{E})$, where $\mathcal{E} \subset V \times V$.
The *conditional independence assumption* is encoded by G that is a variable is conditionally independent of its non-descendants given its parents.

The factorization is given as

$$p(Y = y) = \prod_{i \in V} p(y_i \mid y_{\mathsf{pa}_G(i)}) ,$$

where $p(y_i \mid y_{\mathsf{pa}_G(i)})$ is a conditional probability distribution on the parents of node $i \in V$

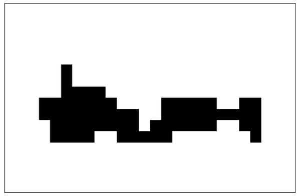


For example:

$$\begin{aligned} p(Y) =& p(y_i, y_j, y_k, y_l) = p(y_l \mid y_i, y_j, y_k)\ p(y_i, y_j, y_k) \\ =& p(y_l \mid y_k)\ p(y_i, y_j, y_k) = p(y_l \mid y_k)\ p(y_k \mid y_i, y_j)\ p(y_i, y_j) \\ =& p(y_l \mid y_k)\ p(y_k \mid y_i, y_j)\ p(y_i)\ p(y_j) . \end{aligned}$$

Example: Man-made structure detection


Original image


Ground truth (24×16 blocks)

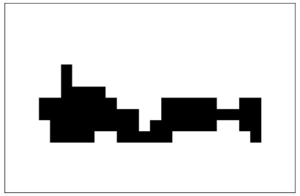
For each block we assign a random variable Y_i. Therefore, V consists of binary output variables corresponding to Y_i, for all $i = 1, \ldots, 384$.

For each random variable Y_i its output domain is $\mathcal{Y}_i = \{0, 1\}$, therefore the output domain in this example is $\mathcal{Y} = \{0, 1\}^{384}$

$\mathcal{X}$ is a set of images, and an input $x \in \mathcal{X}$ is an image.

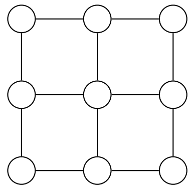
Example: Man-made structure detection


Original image


Ground truth (24×16 blocks)

We consider a simple assumption:
man-made structures are clustered locally together.

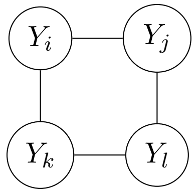
$\mathcal{E}$ consists of edges between 4-connected blocks, which means that we model the relation between neighboring blocks only.


$G(V, \mathcal{E})$

Markov random fields

An undirected graphical model $G = (V, \mathcal{E})$ is called **Markov Random Field** (MRF) if two nodes are conditionally independent whenever they are not connected. In other words, for any node Y_i in the graph, the **local Markov property** holds:

$$p(Y_i \mid Y_{V \setminus \{i\}}) = p(Y_i \mid Y_N(i)) \ ,$$



where $N(i)$ are the neighbors of node i in the graph.
Alternatively, one can use the following equivalent notation:

$$Y_i \perp\!\!\!\perp Y_{V \setminus \mathsf{cl}(i)} \mid Y_{N(i)} \ ,$$

where $\mathsf{cl}(i) = \{i\} \cup N(i)$ is the *closed neighborhood* of i.

For example:

$$Y_i \perp\!\!\!\perp Y_l \mid Y_j, Y_k \Rightarrow p(Y_i, Y_l \mid Y_j, Y_k) = p(Y_i \mid Y_j, Y_k) \ p(Y_l \mid Y_j, Y_k) \ .$$

Gibbs distribution

A probability distribution $p(Y)$ on an undirected graphical model $G = (V, \mathcal{E})$ is called **Gibbs distribution** if it can be factorized into potential functions $\psi_C(y_C) > 0$ defined on cliques (i.e. fully connected subgraph) that cover all nodes and edges of G. That is,

$$p(Y) = \frac{1}{Z} \prod_{C \in \mathcal{C}(G)} \psi_C(y_C) \ ,$$

where $\mathcal{C}(G)$ denotes the set of all (maximal) cliques and

$$Z = \sum_{y \in \mathcal{Y}} \prod_{C \in \mathcal{C}(G)} \psi_C(y_C) \ .$$

is the normalization constant. Z is also known as **partition function**.

Hammersley-Clifford theorem

Let $G = (V, \mathcal{E})$ be an undirected graphical model. The Hammersley-Clifford theorem tells us that the following are equivalent:

- G is an MRF model
- The joint probability distribution $P(Y)$ on G has Gibbs-distribution.

An MRF defines a family of **joint probability distributions** by means of an undirected graph $G = (V, \mathcal{E})$, $\mathcal{E} \subset V \times V$ (there are no self-edges), where the graph encodes conditional independence assumptions between the random variables corresponding to V.

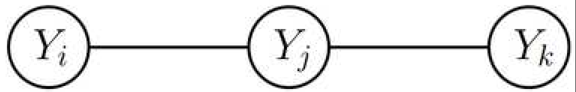
Since, the potential functions $\psi_C(y_c) > 0$

$$\psi_C(y_C) = \exp(-E_C(y_C)) \quad \Leftrightarrow \quad E_C(y_C) = -\log((\psi_C(y_C))) \ .$$

Examples

Cliques $\mathcal{C}(G_1)$: set of nodes $V' \subseteq V$ such that $\mathcal{E} \cap (V' \times V') = V' \times V'$
Here $\mathcal{C}(G_1) = \{\{i\}, \{j\}, \{k\}, \{i,j\}, \{j,k\}\}$, hence

$$p(y) = \frac{1}{Z} \psi_i(y_i) \psi_j(y_j) \psi_k(y_k) \psi_{ij}(y_i, y_j) \psi_{jk}(y_j, y_k)$$

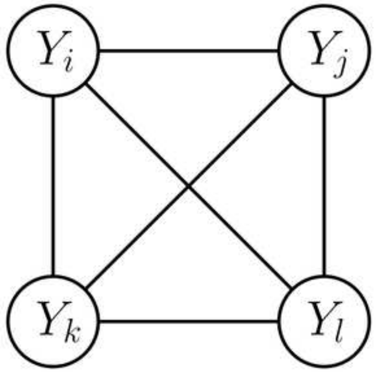


G_1

Here $\mathcal{C}(G_2) = 2^{\{i,j,k,l\}}$ (all subsets of V)

$$p(y) = \frac{1}{Z} \prod_{A \in 2^{\{i,j,k,l\}}} \psi_A(y_A)$$

$$\begin{aligned} 2^{\{i,j,k,l\}} =& \{\{i\}, \{j\}, \{k\}, \{l\}, \{i,j\}, \{i,k\}, \{i,l\}, \{j,k\}, \{j,l\}, \\ & \{i,j,k\}, \{i,j,l\}, \{i,k,l\}, \{j,k,l\}, \{i,j,k,l\}\} \end{aligned}$$



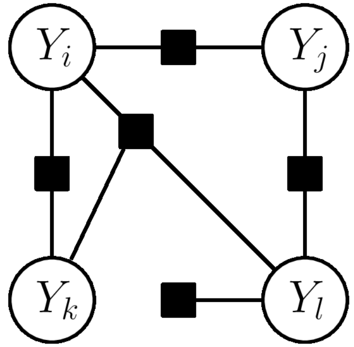
G_2

Factor graphs

Factor graphs are *undirected graphical models* that **make explicit the factorization** of the probability function.

A factor graph $G = (V, \mathcal{F}, \mathcal{E})$ consists of

- variable nodes V (◯) and factor nodes $\mathcal{F}$ (■),
- edges $\mathcal{E} \subseteq V \times \mathcal{F}$ between variable and factor nodes
- $N : \mathcal{F} \to 2^V$ is the *scope of a factor*, defined as the set of neighboring variables, i.e. $N(F) = \{i \in V : (i, F) \in \mathcal{E}\}$.



Factor graph

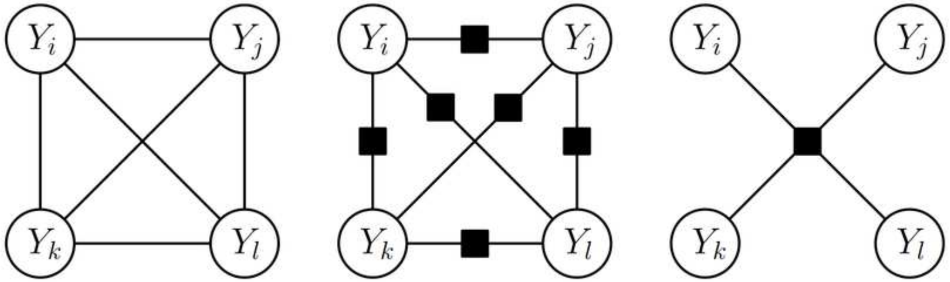
A family of distribution is defined that factorizes according to

$$p(y) = \frac{1}{Z} \prod_{F \in \mathcal{F}} \psi_F(y_{N(F)}) \quad \text{with} \quad Z = \sum_{y \in \mathcal{Y}} \prod_{F \in \mathcal{F}} \psi_F(y_{N(F)}) \; .$$

Each factor $F \in \mathcal{F}$ connects a subset of nodes, hence we write $F = \{v_1, \ldots, v_{|F|}\}$ and $y_F = y_{N(F)} = (y_{v_1}, \ldots, y_{v_{|F|}})$.

Examples

Factor graphs are universal, explicit about the factorization, hence it is easier to work with them.



Examples:

$$
\begin{aligned}
p_1(y) =& \frac{1}{Z_1}\psi_{ij}(y_i, y_j)\psi_{ik}(y_i, y_k)\psi_{il}(y_i, y_l)\psi_{jk}(y_j, y_k)\psi_{jl}(y_j, y_l)\psi_{kl}(y_k, y_l) \\
p_2(y) =& \frac{1}{Z_2}\psi_{ijkl}(y_i, y_j, y_k, y_l)
\end{aligned}
$$

Conditional random fields

We have discussed the joint distribution

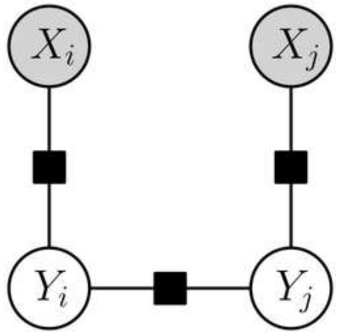
$$p(y) = \frac{1}{Z} \prod_{F \in \mathcal{F}} \psi_F(y_{N(F)}) ,$$

but we often have access to measurements $X = x$, hence the **conditional distribution** $p(Y = y \mid X = x)$ can be directly modeled, too. This can be expressed compactly using **conditional random fields** (CRF) with the factorization

$$p(y \mid x) = \frac{1}{Z(x)} \prod_{F \in \mathcal{F}} \psi_F(y_F; x_F)$$

with the partition function depending on x_F

$$Z(x) = \sum_{y \in \mathcal{Y}} \prod_{F \in \mathcal{F}} \psi_F(y_F; x_F) .$$



Shaded variables: The observations $X = x$.

Potentials and energy functions

We typically would like to infer marginal probabilities $p(Y_F = y_F \mid x)$ for some factors $F \in \mathcal{F}$.

Assuming $\psi_F : \mathcal{Y}_F \to \mathbb{R}_+$, where $\mathcal{Y}_F = \times_{i \in N(F)} \mathcal{Y}_i$ is the product domain of the variables adjacent to F, instead of *potentials*, we can also work with *energies*.

We define an energy function $E_F : \mathcal{Y}_{N(F)} \to \mathbb{R}$ for each factor $F \in \mathcal{F}$.

$$E_F(y_F; x_F) = -\log(\psi_F(y_F; x_F)) \quad \Leftrightarrow \quad \psi_F(y_F; x_F) = \exp(-E_F(y_F; x_F)) .$$

Potentials and energy functions (cont.)

$$
\begin{aligned}
p(y \mid x) =& \frac{1}{Z(x)} \prod_{F \in \mathcal{F}} \psi_F(y_F; x_F) \\
=& \frac{1}{Z(x)} \exp(-\sum_{F \in \mathcal{F}} E_F(y_F; x_F)) = \frac{1}{Z(x)} \exp(-E(y; x))
\end{aligned}
$$

for $E(y;x) = \sum_{F \in \mathcal{F}} E_F(y_F; x_F)$. Hence, $p(y \mid x)$ is completely determined by $E(y;x)$. This provides a natural way to quantify prediction uncertainty by means of marginal distributions $p(y_F \mid x_F)$.

Note that the potentials become also functions of (part of) x, i.e. $\psi_F(y_F; x_F)$ instead of just $\psi_F(y_F)$. Nevertheless, x is **not** part of the probability model, i.e. it is not treated as random variable.

Energy Minimization

Assuming a finite $\mathcal{X}$, the goal is to predict $f : \mathcal{X} \to \mathcal{Y}$ by solving $y^* = \operatorname{argmax}_{y \in \mathcal{Y}} p(y|x)$

$$
\begin{aligned}
\operatorname*{argmax}_{y \in \mathcal{Y}} \; p(y|x) &= \operatorname*{argmax}_{y \in \mathcal{Y}} \; \frac{1}{Z(x)} \exp(-E(y;x)) \\
&= \operatorname*{argmax}_{y \in \mathcal{Y}} \; \exp(-E(y;x)) \\
&= \operatorname*{argmax}_{y \in \mathcal{Y}} \; -E(y;x) \\
&= \operatorname*{argmin}_{y \in \mathcal{Y}} \; E(y;x) \; .
\end{aligned}
$$

Energy minimization can be interpreted as solving for the most likely state of factor graph.

In practice, one typically models the energy function directly.

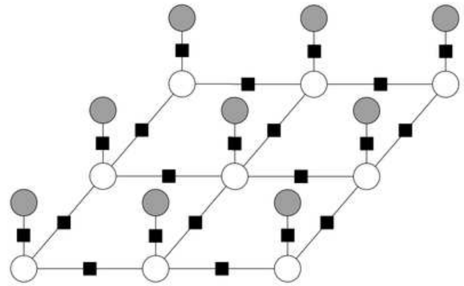
Example: Man-made structure detection

Conditional independences are specified by the factor graph, i.e. all blocks only depend on the neighboring ones.
The conditional distribution factorizes (up to pairwise factors) as

$$p(y \mid x) = \frac{1}{Z(x)} \prod_{i \in V} \psi_i(y_i; x_i) \prod_{i \in V, j \in N(i)} \psi_{ij}(y_i, y_j)$$

with

$$Z(x) = \sum_{y \in \{0,1\}^{384}} \prod_{i \in V} \psi_i(y_i; x_i) \prod_{i \in V, j \in N(i)} \psi_{ij}(y_i, y_j)$$



The corresponding energy function:

$$E(y; x) = \sum_{i \in V} E_i(y_i; x_i) + \sum_{i \in \mathcal{V}, j \in N_i} E_{ij}(y_i, y_j) \ .$$

Example: Man-made structure detection

In order to define energy functions for unary factors, one can consider a set of functions $\phi_i : \mathcal{Y}_i \times \mathcal{X}_i \to [0;1]$:

$$E_i(y_i; x_i) = -\log \phi_i(y_i; x_i) \quad \text{for all } i \in V \ .$$

For pairwise factor energies here we use the **Potts model**, that is

$$E_{ij}(y_i, y_j) = [\![y_i \neq y_j]\!] = \begin{cases} 0, & \text{if } y_i = y_j \\ 1, & \text{otherwise.} \end{cases}$$

The resulting energy function given as

$$\begin{aligned} E(y; x) &= \sum_{i \in V} E_i(y_i; x_i) + \sum_{i \in V, j \in N(i)} E_{ij}(y_i, y_j) \\ &= \sum_{i \in \mathcal{V}} -\log \phi_i(y_i; x_i) + \sum_{i \in V, j \in N(i)} [\![y_i \neq y_j]\!] \ . \end{aligned}$$

Inference

The goal is to make predictions $y \in \mathcal{Y}$, *as good as possible*, about unobserved properties for a given data instance $x \in \mathcal{X}$.

Suppose we are given a graphical model (e.g., a factor graph). **Inference** means the procedure to estimate the probability distribution, encoded by the graphical model, for a given data (or observation).

Maximum A Posteriori (MAP) inference: Given a factor graph and the observation x, find the *state* $y^* \in \mathcal{Y}$ *of maximum probability*,

$$y^* = \operatorname*{argmax}_{y \in \mathcal{Y}} p(Y = y \mid x) = \operatorname*{argmin}_{y \in \mathcal{Y}} E(y; x) \ .$$

Inference (cont.)

Probabilistic inference: Given a factor graph and the observation x, find the value of the *log partition function* and the *marginal distributions* for each factor,

$$\log Z(x) = \log \sum_{y \in \mathcal{Y}} \exp(-E(y; x)) \ ,$$

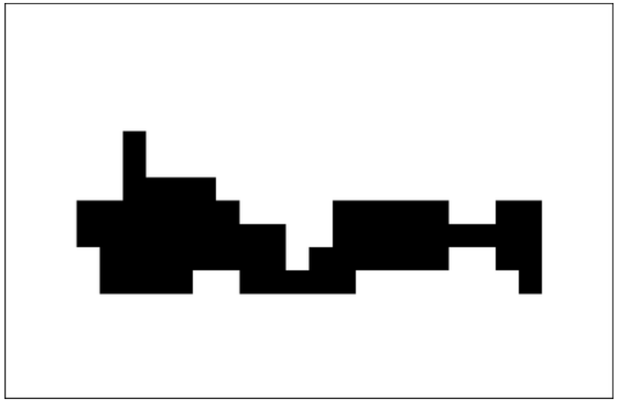
$$\mu_F(y_F) = p(Y_F = y_F \mid x) \quad \forall F \in \mathcal{F}, \forall y_F \in \mathcal{Y}_F \ .$$

This typically includes variable marginals, i.e. $\mu_i = p(y_i \mid x)$, to make a single joint prediction y for all variables.

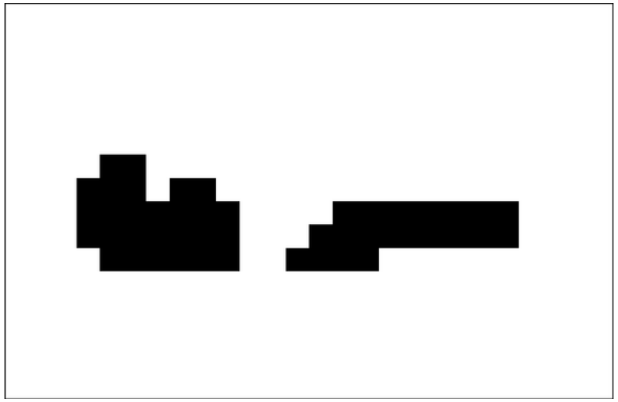
Both inference problems are known to be NP-hard for general graphs and factors, but can be tractable if suitably restricted (see for example pseudo boolean optimization).

Example: Man-made structure detection

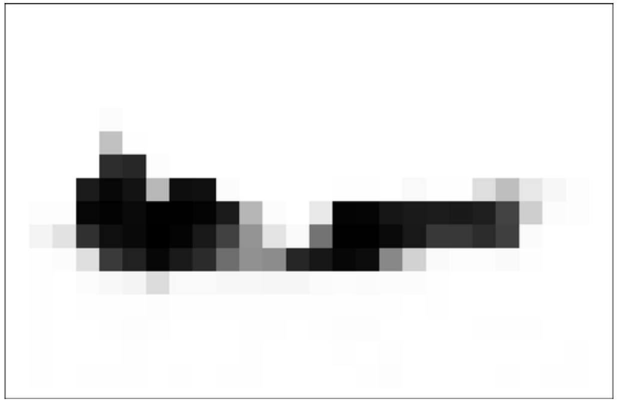
Input image



Ground truth



MAP inference



Probabilistic inference

Literature

Sebastian Nowozin and Christoph H. Lampert.
Structured Prediction and Learning in Computer Vision.
In *Foundations and Trends in Computer Graphics and Vision*, Volume 6, Number 3-4. Note: Chapter 2.