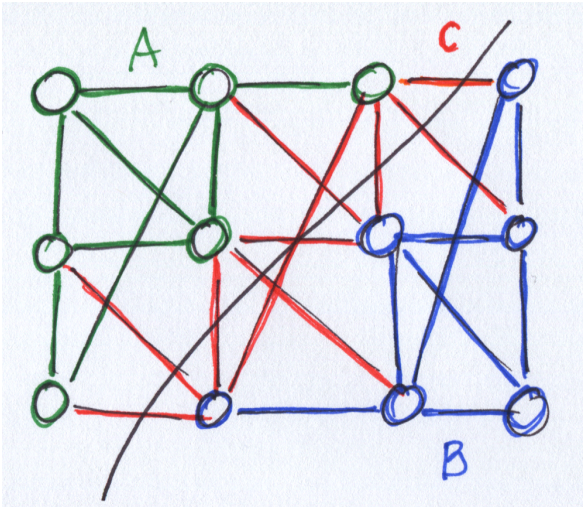
Computer Vision: Segmentierung II

D. Schlesinger – TUD/INF/KI/IS

- Normalized Cut
- Diskrete Energieminimierung
- Random Walk

Normalized Cut

Zunächst Schnitte etwas allgemeiner:



Die Idee: das Bild ist ein Graph $G = (V, E)$, Pixel sind Knoten V,
Kanten E verbinden benachbarte Knoten
Kanten haben **Kosten** w_{uv}, die davon anhängen:

- wie ähnlich die entsprechenden Bildausschnitte sind (Färbungen, Merkmalsvektoren etc.);
- wie weit die entsprechenden Pixel von einander entfernt sind.

Ein Schnitt ist eine Teilmenge der Kanten C so,
dass der Graph nach dem Entfernen dieser Teilmenge nicht mehr zusammenhängend ist.

Alternativ kann man das als eine **Partitionierung** der Knotenmenge V
auf zwei Teilmengen A und B verstehen, d.h. $A \cup B = V$, $A \cap B = \emptyset$

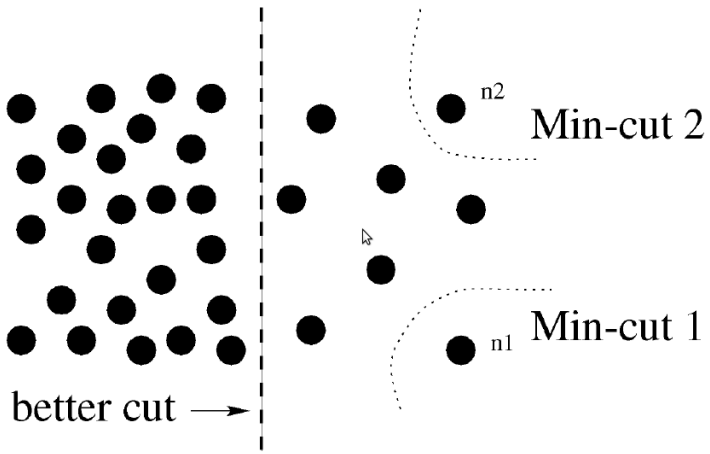
Die Kosten einer Partitionierung ergeben sich als die Summe aller Kantenkosten im Schnitt:

$$cut(A, B) = \sum_{u \in A, v \in B} w_{uv}$$

Die Aufgabe besteht in der Suche nach dem Schnitt optimaler (meist minimaler) Kosten.

Normalized Cut

[Shi, Malik, 2000]:
Schnitte allein berücksichtigen nicht die Kardinalität der Partition (nicht balanciert).



Association:

$$assoc(A, V) = \sum_{u \in A, t \in V} w_{ut}$$

1) Normalized Cut:

$$NCut(A, B) = \frac{cut(A, B)}{assoc(A, V)} + \frac{cut(A, B)}{assoc(A, V)}$$

2) Total normalized association:

$$Nassoc(A, B) = \frac{assoc(A, A)}{assoc(A, V)} + \frac{assoc(B, B)}{assoc(B, V)}$$

Interessant – das ist dasselbe:

$$\begin{aligned} Ncut(A, B) &= \frac{cut(A, B)}{assoc(A, V)} + \frac{cut(A, B)}{assoc(B, V)} = \\ &= \frac{assoc(A, V) - assoc(A, A)}{assoc(A, V)} + \frac{assoc(B, V) - assoc(B, V)}{assoc(B, V)} = \\ &= 2 - Nassoc(A, B) \rightarrow \min_{(A,B)} \end{aligned}$$

Normalized Cut – Lösungsweg

Sei $x_i \in \{+1, -1\}$ eine Indikatorvariable: „$+1$“ heißt die menge A, „-1“ – die Menge B

$$Ncut(A, B) = \frac{\sum_{x_i<0, x_j>0} -w_{ij} x_i x_j}{\sum_{x_i>0} d(i)} + \frac{\sum_{x_i>0, x_j<0} -w_{ij} x_i x_j}{\sum_{x_i<0} d(i)}$$

mit $d_i = \sum_j w_{ij}$.
...

$$\min_x Ncut(x) = \min_y \frac{y^T(D - W)y}{y^T D y}$$

mit $y_i \in \{1, -b\}$, $y^T D 1 = 0$,
D ist die Diagonalmatrix aus d_i, W ist die Matrix aus w_{ij}, b ergibt sich aus w_{ij}

Das ist ein **Rayleigh quotient**. Wenn $y_i \in [1, -b]$ gilt, ergibt sich die Lösung aus dem **generalized eigenvalue system**

$$(D - W)y = \lambda D y$$

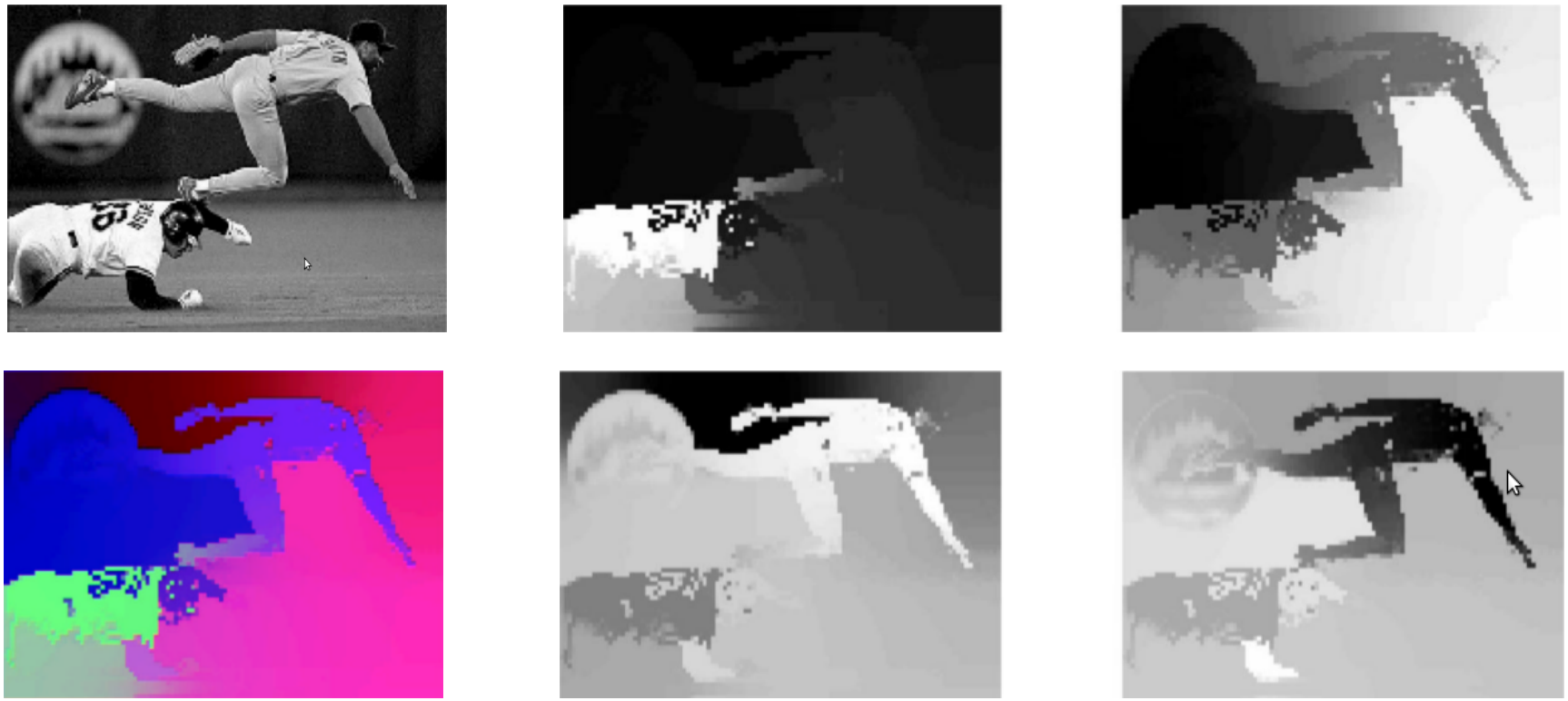
...

... Thus, the second smallest eigenvector of the generalized eigensystem is the real valued solution to our normalized cut problem.

λ-s – **Spektrum des Graphen**.

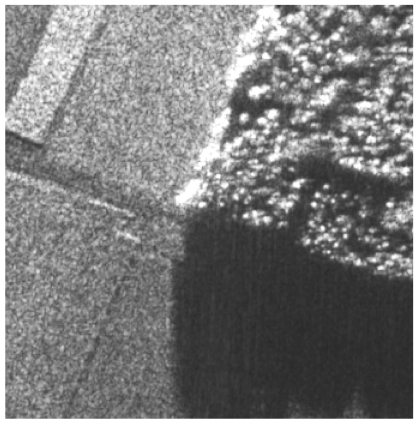
Normalized Cut

... the eigenvector with the third smallest eigenvalue is the real valued solution that optimally subpartitions the first two parts. In fact, this line of argument can be extended to show that one can subdivide the existing graphs, each time using the eigenvector with the next smallest eigenvalue.



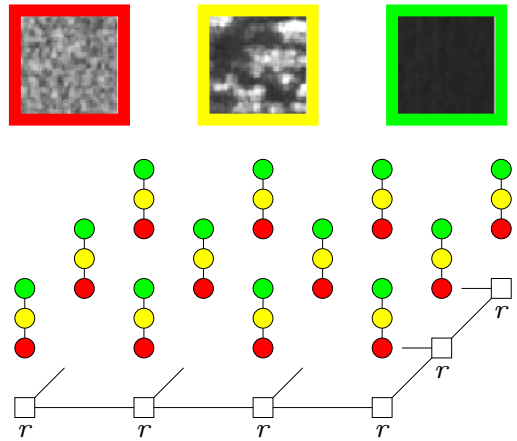
„Soft Segmentation“

Diskrete Energieminimierung

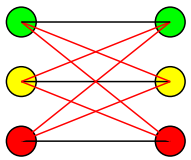


Original

A possible segmentation



Compactness terms



$k = 3$: Shadow

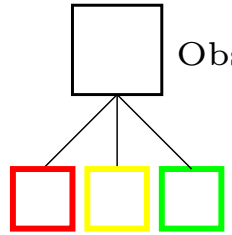
$k = 2$: Forest

$k = 1$: Field

Penalty

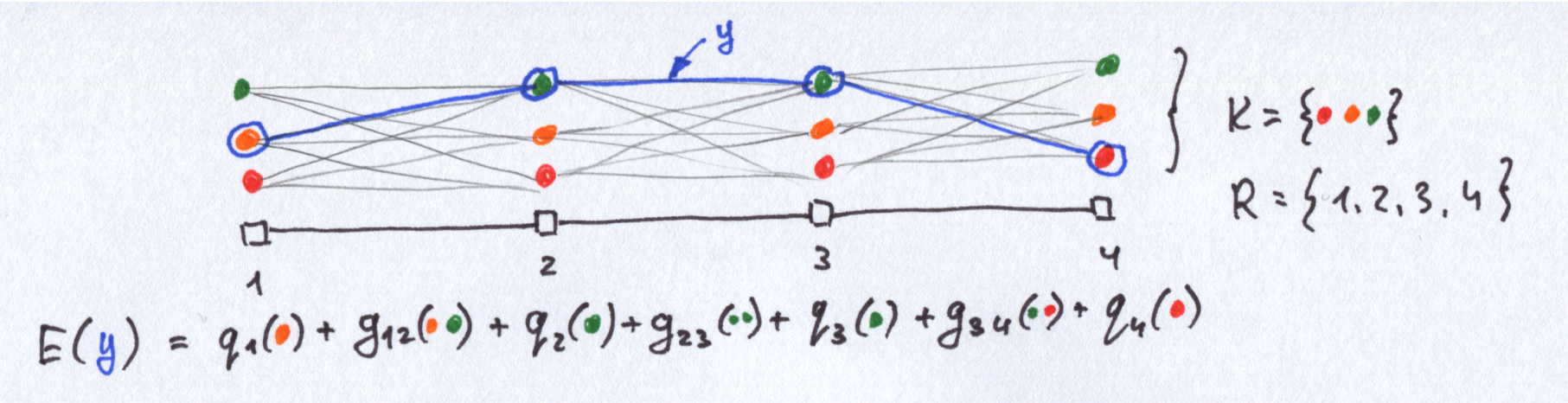
Zero

Data terms



Observed features

Dissimilarity measure



Graph $V = (R, E)$ mit der Knotenmenge R und der Kantenmenge E.

Labelmenge K, Labelling $y : R \to K$

Bewertungsfunktionen $q_r : K \to \mathbb{R}$ und $g_{rr'} : K \times K \to \mathbb{R}$

Gesucht wird nach dem Labelling minimaler summarischen Bewertung:

$$y^* = \arg\min_y E(y) = \arg\min_y \left[\sum_r q_r(y_r) + \sum_{rr'} g_{rr'}(y_r, y_{r'}) \right]$$

Diskrete Energieminimierung

Relationen zu anderen:

- Ncut ist eingeschränkter als DEM, weil nur für binäre Segmentierung
- DEM ist eingeschränkter als Ncut, weil nicht normalisiert
- Gibt es keine $g_{rr'}$, so ist DEM eine Clusterung

Algorithmen:

- Dynamische Programmierung (Ketten, Bäume)
- Suchtechniken: Iterated Conditional Modes, α-expansion, $\alpha - \beta$-swap
- Simulated Annealing, Branch and Bound ...
- LP-relaxations:
  - Boros, Hammer: Quadratic Pseudo-Boolean Optimization
  - Belief Propagation für MinSum – im Grunde Diffusion
  - Kolmogorov: Message Passing, TRWS („gezielte“ äquivalente Transformationen)
  - Komodakis, M. Schlesinger: Subgradient Algorithmus
  - Rother ..., Fusion Moves Algorithmus ?
  - Werner: Cutting Plane Algorithmus
  - Shechovtsov?, ...: Sub-/Supermodular Decomposition
  - Savchynskyy, Kappes, Schmidt, Schnörr: Nesterov's Scheme ... ?

Interaktive Segmentierung

Frage: Wie sind die Funktionen q und g zu wählen (Bildspezifisch/allgemein)?

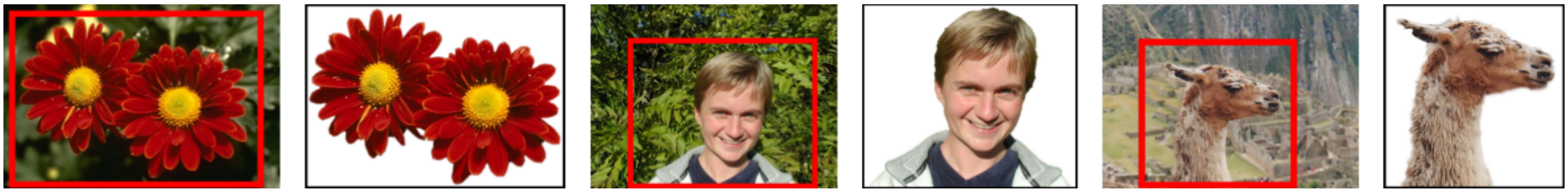
Normalerweise entsprechen g a-priori Annahmen über die Szene, z.B. Kompaktheit, d.h. ziemlich allgemein, ein Beispiel – das Potts Modell $g(k, k') = \lambda \cdot \mathbb{I}(k \neq k')$

q entsprechen dagegen (oft) den Farbverteilungen der Objekte, d.h. $-\ln p(x|k)$
$\Rightarrow$ müssen für jedes Bild neu gelernt werden, Lernen im Voraus nicht sinnvoll.

Voll unüberwacht – nur bedingt möglich, instabil

$\Rightarrow$ **teilüberwachtes** Lernen

Beispiel: **GrabCut** [Rother, Kolmogorov, Blake, 2004].

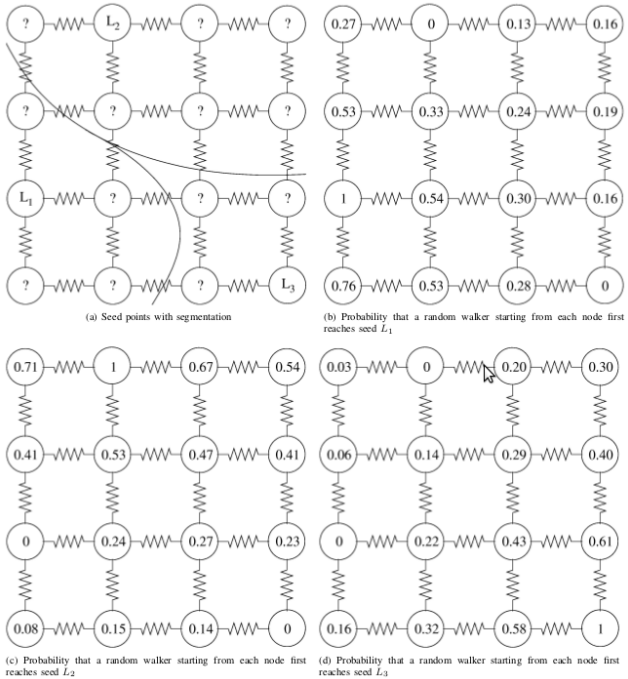


Aus dem initialen Rechteck werden Farbverteilungen gelernt
Iteriere solange sich etwas ändert:

- Finde die neue Segmentierung,
- Lerne daraus die Farbverteilungen erneut.

Random Walk

[Grady, 2006]



(a) Seed points with segmentation

(b) Probability that a random walker starting from each node first reaches seed L_1

(c) Probability that a random walker starting from each node first reaches seed L_2

(d) Probability that a random walker starting from each node first reaches seed L_3

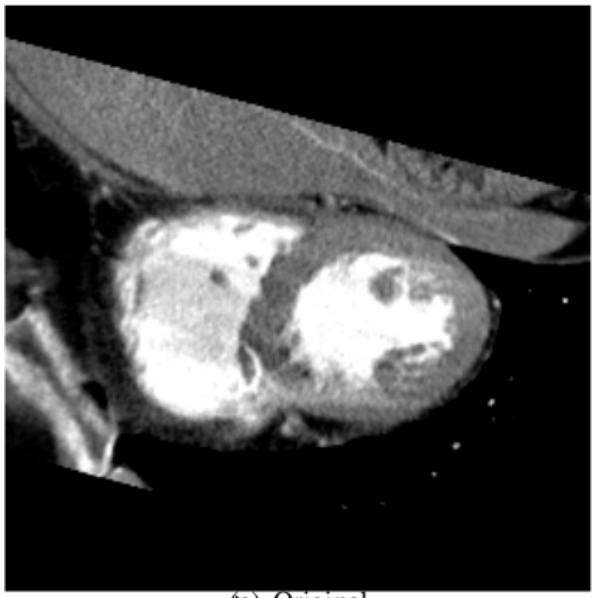
Ein Teilchen bewegt sich (stochastisch) vom Knoten zu Knoten (Random Walk).

Seine Übergänge hängen von den Eigenschaften der Kanten ab.

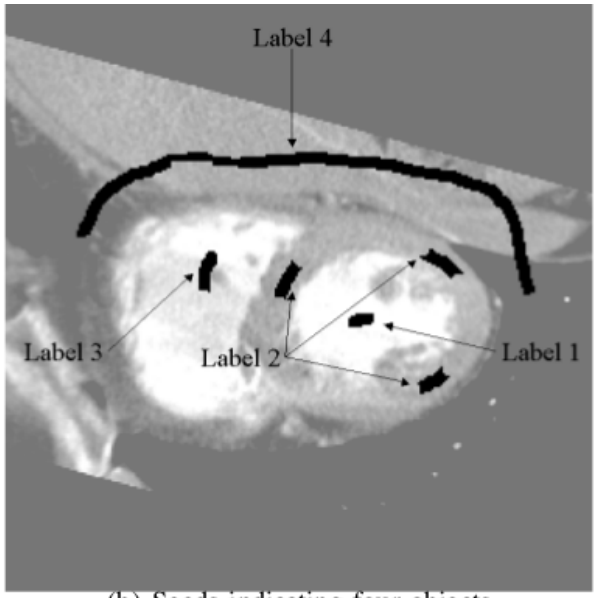
Gegeben seien die „Zielpositionen“, die unterschiedlichen Segmenten entsprechen.

Für jeden Pixel („Startposition“) und jede Zielposition wird die Wahrscheinlichkeit dafür berechnet, dass das Teilchen diese Zielposition als erste erreicht.

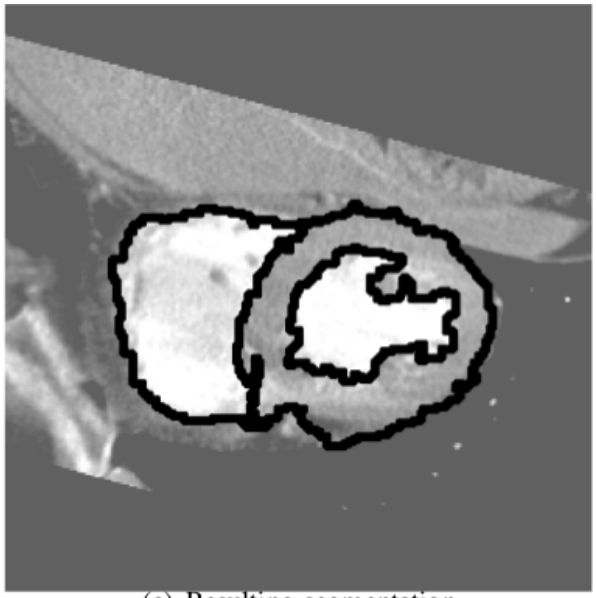
Random Walk



(a) Original



(b) Seeds indicating four objects

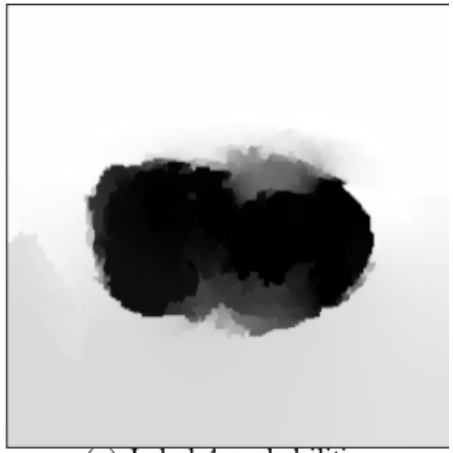


(c) Resulting segmentation

(d) Label 1 probabilities

(e) Label 2 probabilities

(f) Label 3 probabilities



(g) Label 4 probabilities

Weitere Themen

- Statistisch – MRF
- Lernen
- Diskriminatives Lernen – CRF
- Modelle höherer Ordnung (Energieminimierung und statistisch)
- Formmodelle
- Hierarchische Modelle, Schichten
- Zusammengesetzte Modelle: Pixelebene (regulär) $\leftrightarrow$ Pictorial Structures
- ...