Bildverarbeitung: Energieminimierung II

D. Schlesinger – TUD/INF/KI/IS

Entrauschung

Beide Definitionsbereich und Wertebereich sind diskret.

$R \in \mathbb{Z}^2$ – die Pixelmenge, $E \subset R^2$ – die Nachbarschaftstruktur (z.B. 4-Nachbarschaft)
$x : R \to \mathbb{Z}$ – das Ausgangsbild, $y : R \to K$ – die gesuchte Abbildung (das restaurierte Bild). $k \in K$ repräsentiert den „wahren“ Grauwert (Label).

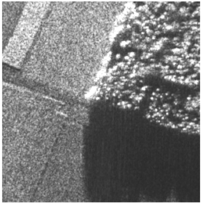
Die Energieminimierung:

$$y^* = \arg\min_y \Big[E_d(y) + \alpha E_m(y) \Big]$$

$$\text{z.B.} \quad \arg\min_y \Big[\sum_{r \in R} (x_r - y_r)^2 + \alpha \sum_{rr' \in E} (y_r - y_{r'})^2 \Big]$$

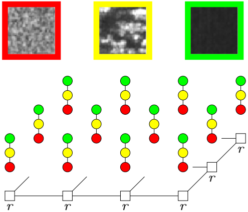
An sich vielleicht nicht so sehr sinnvoll, aber ...

Die Menge der Pixel ist auf Teilmengen partitioniert – auf wieviele?

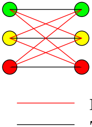


Original

A possible segmentation



Compactness terms



$k = 3$: Shadow

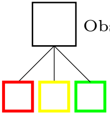
$k = 2$: Forest

$k = 1$: Field

Penalty

Zero

Data terms



Observed features

Dissimilarity measure

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

Iterated Conditional Modes

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

Die Idee – wähle immer wieder das energetisch günstigste Label bei fixiertem Rest.

Wiederhole oft für alle r:

$$y_r = \arg\min_k \left[q_r(k) + \sum_{r':rr' \in E} g_{rr'}(k, y_{r'}) \right]$$

+ Extrem einfach, parallelisierbar.
− „Koordinatenweise" Optimierung

  → konvergiert nicht zum globalen Optimum selbst bei einfachen Modellen.

  Die Anzahl der Iterationen ist schwierig abzuschätzen.

Erweiterung: fixiere nicht alle Variablen bis auf eine, sondern nur eine Teilmenge so, das der Rest einfach optimierbar ist (zum Beispiel eine Kette).

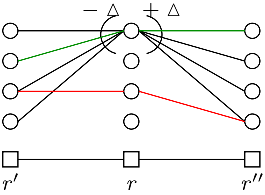
Für Bilder – Zeilenweise/Spaltenweise Optimierung.

Äquivalente Transformationen (Reparametrisierung)

Zwei Aufgaben $A = (q, g)$ und $A' = (q', g')$ sind zu einander äquivalent, wenn

$$\left[\sum_r q_r(y_r) + \sum_{rr'} g_{rr'}(y_r, y_{r'})\right] = \left[\sum_r q'_r(y_r) + \sum_{rr'} g'_{rr'}(y_r, y_{r'})\right]$$

für alle Labellings y. $\mathcal{A}(A)$ – Äquivalenzklasse (alle zu A äquivalenten Aufgaben).



Äquivalente Transformation
$\Phi = \big(\varphi_r(k)\ \forall r, k,\quad \varphi_{rr'}(k),\ \forall rr', k\big)$ so dass

$$\varphi_r(k) + \sum_{r':rr' \in E} \varphi_{rr'}(k) = 0 \quad \forall r, k$$

Sei $A = (q, g)$ eine Aufgabe, $A' = (q', g') = \Phi(A)$ ist die Aufgabe nach der Anwendung der Äquivalenten Transformation Φ, d.h.

$$q'_r(k) = q_r(k) + \varphi_r(k)$$
$$g'_{rr'}(k, k') = g_{rr'}(k, k') + \varphi_{rr'}(k, k') + \varphi_{r'r}(k', k)$$

$\Rightarrow$ A und A' sind zu einander äquivalent.

Äquivalente Transformationen

Eigenschaften:

$\Phi\Big(\Phi'(A)\Big) = \Phi'\Big(\Phi(A)\Big) = (\Phi \oplus \Phi')(A)$ – Superposition.

$\Phi^{-1}\Big(\Phi(A)\Big) = A$, d.h. $\Phi \oplus \Phi^{-1} = \Phi^0$ – Inverse Transformationen, Leere Transformation.

Die Menge aller Φ bildet eine Gruppe.

Beispiel für Verwendung ÄT – Quadratic Pseudo-Bolean Optimization (QPBO):
Für binäre Probleme (d.h. $K = \{0, 1\}$) lässt sich die Energie wie folgt schreiben:

$$E(y) = \sum_r y_r\, q_r + \sum_{rr'} y_r\, y_{r'}\, g_{rr'}$$

q_r und $g_{rr'}$ sind Knoten- bzw. Kantenspezifische Zahlen.
Die Energie ist ein Polynom zweiter Ordnung.
Sind alle alle $g_{rr'} \leq 0$, so ist die Energie „konkav".

Sind zwei Aufgaben A und A' äquivalent,
so existiert eine Äquivalente Transformation Φ so, dass $A' = \Phi(A)$.

Scheinbare Qualität

$$E(A) = \min_y \Big[\sum_r q_r(y_r) + \sum_{rr'} g_{rr'}(y_r, y_{r'}) \Big]$$

$$\geq$$

$$SQ(A) = \sum_r \min_k q_r(k) + \sum_{rr'} \min_{kk'} g_{rr'}(k, k')$$

Die Äquivalenten Transformationen ändern $E(A)$ nicht, $SQ(A)$ aber schon.

Die Idee – suche die Aufgabe größter Scheinbarer Qualität in der Äquivalenzklasse $\mathcal{A}(A)$ – maximiere die untere Schranke der Energie:

$$\sum_r \min_k \Big(q_r(k) + \varphi_r(k) \Big) + \sum_{rr'} \min_{kk'} \Big(g_{rr'}(k, k') + \varphi_{rr'}(k) + \varphi_{r'r}(k') \Big) \to \max_\Phi$$

$$\text{s.t. } \varphi_r(k) + \sum_{r':rr' \in E} \varphi_{rr'}(k) = 0 \quad \forall r, k$$

Eine Aufgabe heißt trivial, wenn $E(A) = SQ(A)$.

- Wie ist $SQ(A)$ (effizient) zu maximieren?
- Trivialität zu prüfen ist NP im Allgemeinen.
- Für welche A gibt es einen trivialen Äquivalent?

Diffusion Algorithmus

Wiederhole oft für alle r, k

1) Sammeln – gießen so viel wie möglich in $q_r(k)$:

$$\triangle_{rr'}(k) = \min_{k'} g_{rr'}(k,k')$$
$$q_r(k) = q_r(k) + \sum_{r':rr' \in E} \triangle_{rr'}(k,k')$$
$$g_{rr'}(k,k') = g_{rr'}(k,k') - \triangle_{rr'}(k,k')$$

2) Verteile gleichmäßig auf inzidente Kanten $g_{rr'}(k,k')$:

$$\triangle_r(k) = q_r(k)/4 \quad \text{(bei 4-Nachbarschaft)}$$
$$g_{rr'}(k,k') = g_{rr'}(k,k') + \triangle_r(k)$$
$$q_r(k) = 0$$

Es ist nicht ganz klar, welche Aufgabe der Algorithmus eigentlich löst.
Im Allgemeinen wird SQ damit nicht global optimiert.

Praktisch funktioniert oft befriedigend.

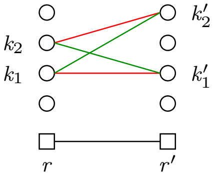
Erweiterung: Message Passing Algorithmen – „gezielte“ (nicht gleichmäßige) Verteilung.

Im Allgemeinen sind diskrete Energieminimierung Probleme NP-vollständig.

Bekannte polynomiell lösbare Fälle:

– Der Graph der Aufgabe ist einfach, zum Beispiel eine Kette.
– Die Funktionen g haben bestimmte Eigenschaften.

Submodulare Aufgaben:



Sei die Menge der Label K vollständig geordnet, d.h. $K = \{1, 2, \ldots |K|\}$, sei $k_1 \leq k_2$ und $k'_1 \leq k'_2$ in dieser Ordnung. Die Funktion $g_{rr'}$ heißt submodular, wenn

$$g(k_1, k'_1) + g(k_2, k'_2) \leq g(k_1, k'_2) + g(k_2, k'_1)$$

für alle derartige viertuppel k_1, k_2, k'_1, k'_2. Die Aufgabe heißt submodular, wenn alle Funktionen submodular sind.

Beispiele:
Entrauschung mit $(y_r - y_{r'})^2$ oder $|y_r - y_{r'}|$, manche binäre Segmentierungen usw.

Es gibt auch gemischte polynomiell lösbare Fälle.