

# Superpixel

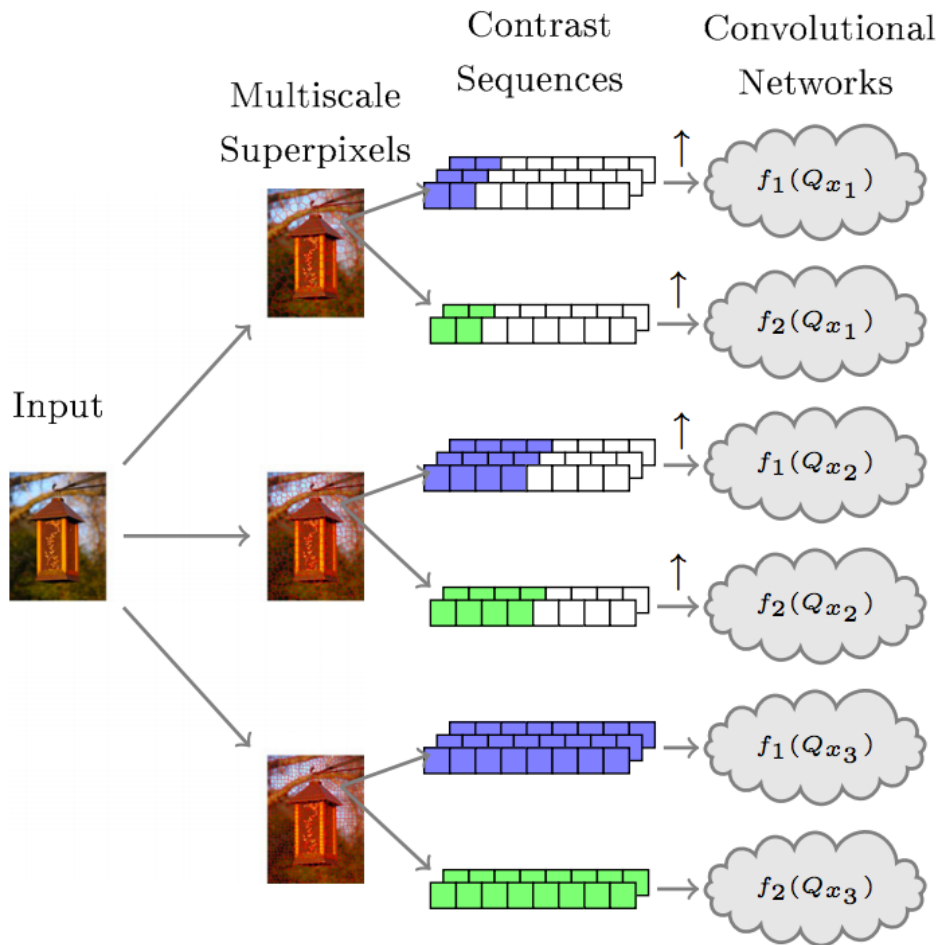
16. November 2016

## 1 Einleitung

- Superpixel: Menge von  $n$  Pixeln  $S_i = \{t_1, \dots, t_n\}$ , wobei  $t_i \in \{1, \dots, N\}$  jeweils einen Pixel beschreibt und die Menge von  $S_i$  räumlich verbunden ist
- Menge von Superpixeln:  $S = \{S_1, \dots, S_m\}$ , sodass  $S_i \cap S_j = \emptyset$  für alle  $i, j$  und  $\cup_i S_i = \cup_j t_j$
- Nachbarschaft:  $(S_i, S_j) \in \mathcal{N}$ , wenn  $S_i$  und  $S_j$  räumlich verbunden sind
- Vorteile:
  - Superpixel bieten eine Möglichkeit, die Größe des Problems zu minimieren
  - CNNs auf Bildern sind rauschend
  - große Netze auf Bildern mit Megapixeln rechnen langsam
- Nachteil: Superpixel haben einen bestimmten Fehlergrad
- $\Rightarrow$  finde den besten Ausgleich zwischen Größe und Fehlergrad

## 2 Lernen von Superpixeln

- **SuperCNN**: anstatt eines Bilders wird eine Sequenz von Superpixeln in das CNN gefüttert
- Problem: kontextbezogene Informationen gehen verloren (Methoden wie Superpixel Lattices adressieren dieses Problem, opfern aber Genauigkeit)
- $\Rightarrow$  zwei Kernel sollen Information wiederherstellen:
  1. *Spatial Kernel*: beschreibt Einzigartigkeit der Farben
  2. *Range Kernel*: beschreibt Farbverteilung
- zusätzlich: Multiscale Struktur des Netzes mit *Shared Weights*
- SuperCNN berechnet für individuelles Bild in etwa genauso lange wie klassische CNNs auf Bildern ( 0.45s)



- Vorteile:
  - benötigt weniger Trainingsdaten
  - Trainingsdaten werden generalisierter genutzt  $\Rightarrow$  Netz fällt es leichter, für unbekannte Bilder Gemeinsamkeiten zu erkennen
  - gleiche oder bessere Performance

### 3 Umwandlung in eine Graph-Repräsentation

- jeder Superpixel bildet einen Knoten im Graphen
- es existiert eine Kante zwischen den Knoten, wenn die entsprechenden Superpixel benachbart sind oder die Distanz zwischen Superpixeln unter eine Schranke  $\epsilon$  fällt (müsste aber durch die Distanz der Kanten abgedeckt sein im CNN auf Graphen)
- Knotenattribute:
  - Farbe (Mean, Range?, Absolute difference?)
  - Schwerpunkt/Position
  - Größe, d.h. Anzahl Pixel (prozentual?)

- Ausdehnung/Form  $\Rightarrow$  z.B. über vereinfachten Polygonzug
  - minimales gedrehtes Hüllrechteck
- Kantenattribute:
  - Distanz zu den Schwerpunkten der Superpixel