Cutting-Plane Training of Non-associative Markov Network for 3D Point Cloud Segmentation

Roman Shapovalov, Alexander Velizhev
Lomonosov Moscow State University

Hangzhou, May 18, 2011
Semantic segmentation of point clouds

- LIDAR point cloud without color information

- Class label for each point
System workflow

Segmentation

Feature computation

Graph construction

CRF inference
Non-associative CRF

\[
\sum_{i \in N} \phi(x_i, y_i) + \sum_{(i,j) \in E} \phi(x_{ij}, y_i, y_j) \rightarrow \max_y
\]

- **Node features**
- **Edge features**
- **Point labels**

- **Associative CRF:** \( \phi(x_{ij}, y_i, y_j) \leq \phi(x_{ij}, k, k) \)
- **Our model:** no such constraints!

[Shapovalov et al., 2010]
CRF training

- parametric model
- parameters need to be learned!
Structured learning
[Anguelov et al., 2005; and a lot more]

- Linear model: \( \phi(x_i, y_i) = \mathbf{w}_{n,i}^T \mathbf{x}_i \mathbf{y} \)
\( \phi(x_i, y_i, y_j) = \mathbf{w}_{e,ij}^T \mathbf{x}_{ij} y_{i,k} y_{j,l} \)

- CRF negative energy: \( \mathbf{w}^T \Psi(\mathbf{x}, \mathbf{y}) \rightarrow \max_y \)

- Find \( \mathbf{w} \) such that
\( \mathbf{w}^T \Psi(\text{a}, \text{b}) > \mathbf{w}^T \Psi(\text{c}, \text{d}) \)
\( \mathbf{w}^T \Psi(\text{e}, \text{f}) > \mathbf{w}^T \Psi(\text{g}, \text{h}) \)
\( \mathbf{w}^T \Psi(\text{i}, \text{j}) > \mathbf{w}^T \Psi(\text{k}, \text{l}) \)
\( \cdots \)
\( \mathbf{w}^T \Psi(\text{m}, \text{n}) > \mathbf{w}^T \Psi(\text{o}, \text{p}) \)
Structured loss

- Define $\mathbf{x} = \text{features}(\cdots)$
- Define structured loss, for example:

$$\Delta(y, \bar{y}) = \sum_{i \in N} [y_i \neq \bar{y}_i]$$

- Find $\mathbf{w}$ such that

$$w^T \Psi(x, \cdots) > w^T \Psi(x, \cdots) + \Delta(\cdots, \cdots)$$

$$w^T \Psi(x, \cdots) > w^T \Psi(x, \cdots) + \Delta(\cdots, \cdots)$$

$$w^T \Psi(x, \cdots) > w^T \Psi(x, \cdots) + \Delta(\cdots, \cdots)$$

$$w^T \Psi(x, \cdots) > w^T \Psi(x, \cdots) + \Delta(\cdots, \cdots)$$

$$\cdots$$

$$w^T \Psi(x, \cdots) > w^T \Psi(x, \cdots) + \Delta(\cdots, \cdots)$$
Cutting-plane training

- A lot of constraints ($K^n$)
  \[ w^T \Psi(x, y) > w^T \Psi(x, \bar{y}) + \Delta(y, \bar{y}), \forall \bar{y} \]
- Maintain a working set
- Add iteratively the most violated one:
  \[ \bar{y} = \arg \max_{\bar{y}} \left\{ v^T \Psi(x, \bar{y}) + \Delta(y, \bar{y}) \right\} \]
- Polynomial complexity
- SVM$^{\text{struct}}$ implementation [Joachims, 2009]
Results

[Munoz et al., 2009]  Our method
Results: balanced loss better than the Hamming one

![Bar chart showing recall values for different categories and models.](chart.png)
Results: RBF better than linear
Results: fails at very small classes

Ground f-score
Vehicle f-score
Tree f-score
Pole f-score

[SVM, 2009]
SVM-LIN
SVM-RBF

0.2% of trainset
Analysis

• Advantages:
  – more flexible model
  – accounts for class imbalance
  – allows kernelization

• Disadvantages:
  – really slow (esp. with kernels)
  – learns small/underrepresented classes badly