Branch&Rank: Efficient, Non-Linear Object Detection

Alain D. Lehmann, Peter V. Gehler, and Luc Van Gool

Computer Vision Laboratory, ETH Zurich, Switzerland

30 August, 2011
Detection means to **localise and categorise** objects.
Appearance variations make it difficult

- intra-class variations
- different views/poses
- illumination changes
- occlusions, etc.

sophisticated
expensive
models
Localise objects among thousands of hypotheses

Search space size

- >10’000 locations
- >1’000 classes

avoid exhaustive enumeration
Efficient detection by ranking sub-images

Runtime = (classifier cost) × (#calls)

- **reduce cost:** cascades [Viola et al. 04, Vedaldi et al. 09]
  - *exhaustive search → not scalable*

- **reduce calls:** branch&bound [Lampert et al. 08, Lehmann et al. 09]
  - *bounds not tight enough → not effective*

**Ranking: ”learn the bound”**

- branch, but not bound
- often <100 classifier calls → non-linear SVMs
- classification for detection
1. Detection: *best-first search*

2. Training: *ranking hypothesis sets*

3. Multi-tasks aspects

4. Results and conclusion
Efficiency by means of adaptive partitioning

Sets of hypothesis

- exploit correlations
- split promising sets
Sets of hypothesis

- exploit correlations
- split promising sets
Efficiency by means of adaptive partitioning

Sets of hypothesis
- exploit correlations
- split promising sets
Efficiency by means of adaptive partitioning

Sets of hypothesis

- exploit correlations
- split promising sets
Efficiency by means of adaptive partitioning

Sets of hypothesis
- exploit correlations
- split promising sets

Branch&Rank: Efficient, Non-Linear Object Detection
Efficiency by means of adaptive partitioning

Sets of hypothesis
- exploit correlations
- split promising sets
Efficiency by means of adaptive partitioning

Sets of hypothesis

- exploit correlations
- split promising sets
Efficiency by means of adaptive partitioning

Sets of hypothesis

- exploit correlations
- split promising sets
Efficiency by means of adaptive partitioning

Sets of hypothesis

- exploit correlations
- split promising sets
Efficiency by means of adaptive partitioning

Sets of hypothesis
- exploit correlations
- split promising sets
Efficiency by means of adaptive partitioning

Sets of hypothesis
- exploit correlations
- split promising sets
- correspond to subimages

Ranking function $f$ prioritises

- $f\left(\text{contains obj.}\right) > f\left(\text{no object}\right)$
- supersedes upper bounds
Training with sets for increased efficiency

Structured SVM ranking [Tschantaridis et al. 04, Blaschko et al. 08]

\[
\min_{w, \xi_i \geq 0} \|w\|^2 + C \sum_i \xi_i \\
\quad f(\Lambda^+_i) - f(\Lambda^-) \geq \Delta(\Lambda^-) - \xi_i \\
with f(\Lambda) = \langle w, \phi(\Lambda) \rangle
\]

- bag-of-words descriptor \( \phi(\Lambda) \)
- kernelize with RBF-\( \chi^2 \) kernel
- \( \Lambda^+ \): generate with oracle
- \( \Lambda^- \): delayed constraint generation

Alain D. Lehmann, Peter V. Gehler, and Luc Van Gool
Branch&Rank: Efficient, Non-Linear Object Detection
From image classification to object categorisation

Image Classification Task

Object Detection Task

(Multiple) Intermediate Tasks

Large sets
- object somewhere
- image classification

Small sets
- object centred
- object categorisation

Task-adapted ranking $f(\Lambda) = \langle w_{q(\Lambda)}, \phi(\Lambda) \rangle$

- task mapping $q(\Lambda)$
- leverage set information
- exploit context
- improved AP by $\approx 10\%$
Branch&rank detects in often <50 iterations

Dataset: PASCAL VOC 2007 (Horses) [Everingham et al., 2007]

- non-linear RBF-$\chi^2$ SVMs
- no cascade approximations

more efficient
detections

more confident detections

costly classifier feasible

Alain D. Lehmann, Peter V. Gehler, and Luc Van Gool
Branch&Rank: Efficient, Non-Linear Object Detection
More results

<table>
<thead>
<tr>
<th>branch&amp;rank</th>
<th>part-based detector</th>
<th>best in challenge</th>
</tr>
</thead>
<tbody>
<tr>
<td>[Lehmann et al. 2011]</td>
<td>[Felzenszwalb et al. 2008]</td>
<td>[Everingham et al. 2007]</td>
</tr>
<tr>
<td>Horse</td>
<td>36.8%</td>
<td>30.1%</td>
</tr>
<tr>
<td>Cow</td>
<td>10.8%</td>
<td>16.5%</td>
</tr>
<tr>
<td>Cat</td>
<td>17.6%</td>
<td>11.0%</td>
</tr>
</tbody>
</table>

Future work

- combine multiple features
- use task-adapted features
Branch&rank is efficient
- less than 100 classifier calls
- non-linear SVMs feasible

Process hypothesis sets
- during detection and training
- “learn the bound”

Multiple task
- combine classification and detection