Object Detection with Incomplete Supervision

Jakob Verbeek

LEAR team, INRIA, Grenoble, France

Joint work with: Gokberk Cinbis and Cordelia Schmid
Why learning from incomplete supervision?

- Fully supervised training requires costly bounding box annotations
- Weakly supervised learning only uses image-wide labels
Overview of this presentation

• Preliminaries on object localization
  ▶ Challenges
  ▶ Representations
  ▶ Search and learning

• Learning with incomplete supervision
  ▶ Multiple instance learning approach
  ▶ Multi-fold training to improve performance
  ▶ Object instance hypothesis refinement

• Experimental evaluation and analysis
Challenging factors in object detection

- Intra-class appearance variation
  - Deformable objects: e.g. animals
  - Transparency: e.g. bottles
  - Sub-categories: e.g. ferry vs yacht

- Scene composition
  - Heavy occlusions: e.g. tables and chairs
  - Clutter: coincidental image content present in bounding box

- Imaging conditions
  - viewpoint, scale, lighting conditions
Representations

- Need for strong appearance features to separate classes despite strong intra-class variability and subtle inter-class variations
  - Consider deformability of cats and dogs
  - Similarity between furry cats and dogs in the similar poses

- Fischer vector representation
  
  [Sanchez et al., IJCV, 2013]
  - Local SIFT descriptors, PCA to 64 dim.
  - 64 component GMM for soft quantization
  - Record first and second order moments of features assigned to each Gaussian
  - 4x4 SPM grid, power and L2 normalization
  - 140K dimensional descriptor
  - PQ compression to reduce storage cost
Representations

- Need for strong appearance features to separate classes despite strong intra-class variability and subtle inter-class variations
  - Consider deformability of cats and dogs
  - Similarity between furry cats and dogs in the similar poses

- Global Convolutional Neural Network feature
  [Jia et al., caffe.berkeleyvision.org]
  - Trained on 1000 ImageNet 2012 categories
  - Caffe framework
  - Use last shared layer for representation
  - Resize detection windows to 224x224 pixels
  - L2 normalization
  - 4K dimensional descriptor
A typical object detection system

- Training a binary classifier that will score object windows
  - Positives given by manual annotation (hundreds to thousands)
  - Potential pool of negatives outside positive boxes (zillions)
    - Repetitive access to find useful/hardest negative samples
    - Store or re-extract feature vectors of these examples

- At test image, classify windows of different shapes and sizes
  - Detection speed proportional to number of considered windows
Issues with classic scanning windows

- Number of detection windows in an image is huge
  - Quadratic in image size
- Features are expensive to evaluate
- Features are expensive to store
- Alternatives to dense exhaustive search are needed
Alternatives to exhaustive sliding window search

- **Sliding window**
  (Viola and Jones 2002; Felzenszwalb et al. 2008, ...)

- **Branch & bound**
  (Lampert et al. 2008; Lehmann et al. 2013)

- **Selective Search**
  (Alexe et al. 2010; Sande et al. 2011)
Alternatives to exhaustive sliding window search

- Branch-and-bound techniques
  - Imposes requirements on type of classifiers / features
    [Lampert, Blaschko, Hofmann, PAMI 2009]

- Feature cascades
  - Requires set of fast features in early stages
    [Viola & Jones, IJCV 2004]

- Coarse-to-fine search
  - Requires compositionality of classifier score
    [Felzenszwalb, Girshick, McAllester, CVPR 2010]

- Data driven generic object hypotheses
  - Consider boxes aligned with low-level image contours
  - Does not impose constraints on classifiers / features
    [Alexe, Deselaers, Ferrari, CVPR 2010]
Search: restricted scanning of bounding box space

- Selective search method [Uijlings et al., IJCV, 2013]
  - 1000 - 2000 windows per image
  - Covers over 95% of true objects with sufficient accuracy
  - Unsupervised multi-resolution hierarchical segmentation
  - Candidate detections generated as bounding box of segments
- Candidate windows used for hard negative mining and testing
- Feature compression using PQ codes and lossless compression
Overview of this presentation

• Preliminaries on object localization
  ▶ Challenges
  ▶ Representations
  ▶ Search and learning

• Learning with incomplete supervision
  ▶ Multiple instance learning approach
  ▶ Multi-fold training to improve performance
  ▶ Object instance hypothesis refinement

• Experimental evaluation and analysis
Why learning from incomplete supervision?

- Fully supervised training requires costly bounding box annotations
- Weakly supervised learning only uses image-wide labels
Learning from incomplete supervision

- Joint identification problem: recognition model and training instances
- Alternating optimization: fix one, optimize the other
State-of-the-art weakly-supervised detector training

- Vast majority of work relies on multiple-instance learning

- Approaches vary in terms of
  - Initialization strategy
  - Object descriptors and detector
  - Utilization of pair-wise window similarities

- Some alternative recent approaches are based on topic models
  Shi, Hospedales, Xiang, ICCV 2013.
  Wang, Ren, Huang, Tan, ECCV 2014.
Multiple instance learning

- Examples come in labeled “bags”
  - Dietterich et al., *Artif. Intell.*, 1997
    - Selective search gives ~1500 windows per image = bag
    - Positive images contain at least one positive window
    - Negative images only have negative windows in the bag

- Multiple Instance SVM
  - Andrews et al., NIPS 2002
    - Initialize initial selection of samples from positive bags
    - Train SVM with selection
    - Select top scoring sample in each positive bag
    - Repeat until convergence
Problems in standard multiple instance learning

- MIL gets stuck at poor local optima
  - Non-convex optimization problem

- Windows used in training get higher score than other windows
  - Biased towards re-localizing on the training windows
Problems in standard multiple instance learning

- Linear SVM classifier score is weighted sum of dot products:
  \[ w^T x = \sum_i \alpha_i (x_i^T x) \]

- Fisher Vector descriptors are near-orthogonal = near zero dot product
  - But recall that descriptors are unit normalized

- Linear SVM scores much higher for windows used in training
  - This causes the degenerate re-localization behaviour
Problems in standard multiple instance learning

- MIL gets stuck at poor local optima
  - Non-convex optimization problem
- Windows used in training get higher score than other windows
  - Biased towards re-localizing on the training windows
Solution: Multi-fold training for multiple instance learning

- Separate sets of positive images for training and re-localization
  - Negative images do not need to be split, since no relocalization there
- Repeat two steps
  - Divide positive training images randomly into K folds
  - For fold \( k = 1,\ldots,K \)
    - Train detector from all training images, except those in fold \( k \)
    - Select top-scoring window in each positive image in fold \( k \)

- Avoids the re-localization bias since windows used for training and evaluation are always different
Solution: Multi-fold training for multiple instance learning

**Algorithm 1** — Multi-fold weakly supervised training

1) Initialization: positive and negative examples are set to entire images up to a 4% border.
2) For iteration $t = 1$ to $T$
   a) Divide positive images randomly into $K$ folds.
   b) For $k = 1$ to $K$
      i) Train using positive examples in all folds but $k$, and all negative examples.
      ii) Re-localize positives by selecting the top scoring window in each image of fold $k$ using this detector.
   c) Train detector using re-localized positives and all negative examples.
   d) Add new negative windows by hard-negative mining.
3) Return final detector and object windows in train data.
A quick look at standard and multi-fold training

<table>
<thead>
<tr>
<th>standard</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>multi-fold</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
The trouble with cats and dogs ...

- Weakly supervised learning can only be expected to learn the most repetitive and discriminative patterns.
- These patterns may not correspond to the full objects, but to parts
... and our solution to cats and dogs

- Refinement of the output of the multi-fold training procedure
- Final detector trained using these refined hypotheses
- Exploit low-level (non-category) contour detection to promote windows aligning with contours
Object hypothesis refinement

- Edge-driven method to generate object hypotheses
  “Edge Boxes”, Zitnick & Dollar, ECCV'14
- Promotes windows that
  - align with long contours,
  - few contours straddle the window boundary
- Here used to re-assess windows using average of detection and objectness score, only considering top-10 detection windows
Overview of this presentation

- Preliminaries on object localization
  - Challenges
  - Representations
  - Search and learning

- Learning with incomplete supervision
  - Multiple instance learning approach
  - Multi-fold training to improve performance
  - Object instance hypothesis refinement

- Experimental evaluation and analysis
Evaluations based on PASCAL VOC'07 benchmark
Evaluation of multi-fold training

- Standard detection AP on test set
- Localization performance on positive training images
  - Fraction of images with correct localization (CorLoc)
    Deselaers et al., PAMI 2012
- Both averaged over all 20 classes

<table>
<thead>
<tr>
<th></th>
<th>Standard</th>
<th>Multi-fold</th>
</tr>
</thead>
<tbody>
<tr>
<td>FV</td>
<td>29.7</td>
<td>38.8 (+9.1)</td>
</tr>
<tr>
<td>CNN</td>
<td>41.2</td>
<td>45.0 (+3.8)</td>
</tr>
<tr>
<td></td>
<td>CorLoc</td>
<td>Detection AP</td>
</tr>
<tr>
<td>FV</td>
<td>15.5</td>
<td>22.4 (+6.9)</td>
</tr>
<tr>
<td>CNN</td>
<td>24.3</td>
<td>25.9 (+1.6)</td>
</tr>
</tbody>
</table>
Evaluation of multi-fold training

- CorLoc over the re-training / re-localization iterations
- Iteration n: n-th iteration after initialization from full image
- For both features: averaged over all 20 classes

Multi-fold training improves both learning from both features
- 10 folds suffice
- 5 to 10 iterations suffice
Window refinement and combining features

- Refinement helps improve performance
- Combining features boosts performance

<table>
<thead>
<tr>
<th>Refinement</th>
<th>No</th>
<th>Yes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CorLoc</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FV</td>
<td>38.8</td>
<td>46.1 (+7.3)</td>
</tr>
<tr>
<td>CNN</td>
<td>45.0</td>
<td>54.2 (+9.2)</td>
</tr>
<tr>
<td>FV+CNN</td>
<td>47.3</td>
<td>52.0 (+4.7)</td>
</tr>
<tr>
<td>Detection AP</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FV</td>
<td>22.4</td>
<td>23.3 (+0.9)</td>
</tr>
<tr>
<td>CNN</td>
<td>25.9</td>
<td>28.6 (+2.7)</td>
</tr>
<tr>
<td>FV+CNN</td>
<td>27.4</td>
<td>30.2 (+2.8)</td>
</tr>
</tbody>
</table>
Analysis: The relation between CorLoc and detection AP

- Relation between localization during training and final test performance
  - Very highly correlated, similar coefficient for both features
Analysis: The relation between CorLoc and detection AP

- Relative performance of weakly supervised learning with respect to performance with full supervision
  - Ratio of AP with weak vs full supervision
  - Stable performance when CorLoc is > 40%, around 80% relative
  - Smaller CorLoc results in rapid deterioration
Analysis: What type of errors are made?

- More correct localization with multi-fold training
- Less overshoot of true object for multi-fold training, more undershoot
- Refinement fixes “undershoot” cases
- Complete failure (<10%) relatively rare: explains robustness
Analysis: what makes weakly supervised learning hard?

- Performance for the shades of grey between fully and weakly supervised learning scenario

<table>
<thead>
<tr>
<th>Supervision</th>
<th>Neg on Pos</th>
<th>Positive Set</th>
<th>mAP(FV)</th>
<th>mAP(CNN)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Image labels only</td>
<td>No</td>
<td>Non-diff/trunc</td>
<td>22.4</td>
<td>25.9</td>
</tr>
<tr>
<td>Cand box for one obj</td>
<td>No</td>
<td>Non-diff/trunc</td>
<td>30.8</td>
<td>36.5</td>
</tr>
<tr>
<td>Cand box for all obj</td>
<td>No</td>
<td>Non-diff/trunc</td>
<td>30.7</td>
<td>35.7</td>
</tr>
<tr>
<td>Cand box for all obj</td>
<td>Yes</td>
<td>Non-diff/trunc</td>
<td>32.0</td>
<td>41.2</td>
</tr>
<tr>
<td>Exact box for all obj</td>
<td>Yes</td>
<td>Non-diff/trunc</td>
<td>32.8</td>
<td>40.5</td>
</tr>
<tr>
<td>Exact box for all obj</td>
<td>Yes</td>
<td>All</td>
<td>35.4</td>
<td>42.8</td>
</tr>
</tbody>
</table>

- The two most critical factors for performance
  - Getting one example right per positive image
  - Hard-negative mining on positive images
Comparison the recent state of the art

- Separation between methods based on whether they leverage external training data to learn CNN features

<table>
<thead>
<tr>
<th></th>
<th>aero</th>
<th>b icy</th>
<th>b ird</th>
<th>boa</th>
<th>bot</th>
<th>bus</th>
<th>car</th>
<th>cat</th>
<th>cha</th>
<th>cow</th>
<th>dtab</th>
<th>dog</th>
<th>hors</th>
<th>mbik</th>
<th>pers</th>
<th>plnt</th>
<th>she</th>
<th>sofa</th>
<th>trai</th>
<th>tv</th>
<th>Av.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pandey and Lazebnik’11 [32]</td>
<td>11.5</td>
<td>—</td>
<td>—</td>
<td>3.0</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>20.3</td>
<td>9.1</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>13.2</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td></td>
</tr>
<tr>
<td>Siva and Xiang’11 [42]</td>
<td>13.4</td>
<td>44.0</td>
<td>3.1</td>
<td>3.1</td>
<td>0.0</td>
<td>31.2</td>
<td>43.9</td>
<td>7.1</td>
<td>0.1</td>
<td>9.3</td>
<td>9.9</td>
<td>1.5</td>
<td>29.4</td>
<td>38.3</td>
<td>4.6</td>
<td>0.1</td>
<td>0.4</td>
<td>3.8</td>
<td>34.2</td>
<td>0.0</td>
<td>13.9</td>
</tr>
<tr>
<td>Russakovsky et al.’12 [35]</td>
<td>30.8</td>
<td>25.0</td>
<td>—</td>
<td>3.6</td>
<td>—</td>
<td>26.0</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>21.3</td>
<td>29.9</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>15.0</td>
</tr>
<tr>
<td>Ours (FV-only)</td>
<td>36.9</td>
<td>38.3</td>
<td>11.5</td>
<td>11.1</td>
<td>1.0</td>
<td>39.8</td>
<td>45.7</td>
<td>16.5</td>
<td>1.2</td>
<td>26.4</td>
<td>4.3</td>
<td>17.7</td>
<td>31.8</td>
<td>44.0</td>
<td>13.1</td>
<td>11.0</td>
<td>31.4</td>
<td>9.7</td>
<td>38.5</td>
<td>36.9</td>
<td>23.3</td>
</tr>
</tbody>
</table>

methods using additional training data

<table>
<thead>
<tr>
<th></th>
<th>aero</th>
<th>b icy</th>
<th>b ird</th>
<th>boa</th>
<th>bot</th>
<th>bus</th>
<th>car</th>
<th>cat</th>
<th>cha</th>
<th>cow</th>
<th>dtab</th>
<th>dog</th>
<th>hors</th>
<th>mbik</th>
<th>pers</th>
<th>plnt</th>
<th>she</th>
<th>sofa</th>
<th>trai</th>
<th>tv</th>
<th>Av.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Song et al.’14 [43]</td>
<td>27.6</td>
<td>41.9</td>
<td>19.7</td>
<td>9.1</td>
<td>10.4</td>
<td>35.8</td>
<td>39.1</td>
<td>33.6</td>
<td>0.6</td>
<td>20.9</td>
<td>10.0</td>
<td>27.7</td>
<td>29.4</td>
<td>39.2</td>
<td>9.1</td>
<td>19.3</td>
<td>20.5</td>
<td>17.1</td>
<td>35.6</td>
<td>7.1</td>
<td>22.7</td>
</tr>
<tr>
<td>Song et al.’14 [44]</td>
<td>36.3</td>
<td>47.6</td>
<td>23.3</td>
<td>12.3</td>
<td>11.1</td>
<td>36.0</td>
<td>46.6</td>
<td>25.4</td>
<td>0.7</td>
<td>23.5</td>
<td>12.5</td>
<td>23.5</td>
<td>27.9</td>
<td>40.9</td>
<td>14.8</td>
<td>19.2</td>
<td>24.2</td>
<td>17.1</td>
<td>37.7</td>
<td>11.6</td>
<td>24.6</td>
</tr>
<tr>
<td>Bilen et al.’14 [6]</td>
<td>42.2</td>
<td>43.9</td>
<td>23.1</td>
<td>9.2</td>
<td>12.5</td>
<td>44.9</td>
<td>45.1</td>
<td>24.9</td>
<td>8.3</td>
<td>24.0</td>
<td>13.9</td>
<td>18.6</td>
<td>31.6</td>
<td>43.6</td>
<td>7.6</td>
<td>20.9</td>
<td>26.6</td>
<td>20.6</td>
<td>35.9</td>
<td>29.6</td>
<td>26.4</td>
</tr>
<tr>
<td>Wang et al.’14 [50]</td>
<td>48.8</td>
<td>41.0</td>
<td>23.6</td>
<td>12.1</td>
<td>11.1</td>
<td>42.7</td>
<td>40.9</td>
<td>35.5</td>
<td>11.1</td>
<td>36.6</td>
<td>18.4</td>
<td>35.3</td>
<td>34.8</td>
<td>51.3</td>
<td>17.2</td>
<td>17.4</td>
<td>26.8</td>
<td>32.8</td>
<td>35.1</td>
<td>45.6</td>
<td>30.9</td>
</tr>
<tr>
<td>Wang et al.’14 [50] +context</td>
<td>48.9</td>
<td>42.3</td>
<td>26.1</td>
<td>11.3</td>
<td>11.9</td>
<td>41.3</td>
<td>40.9</td>
<td>34.7</td>
<td>10.8</td>
<td>34.7</td>
<td>18.8</td>
<td>34.4</td>
<td>35.4</td>
<td>52.7</td>
<td>19.1</td>
<td>17.4</td>
<td>35.9</td>
<td>33.3</td>
<td>34.8</td>
<td>46.5</td>
<td>31.6</td>
</tr>
<tr>
<td>Ours</td>
<td>39.3</td>
<td>43.0</td>
<td>28.8</td>
<td>20.4</td>
<td>8.0</td>
<td>45.5</td>
<td>47.9</td>
<td>22.1</td>
<td>8.4</td>
<td>33.5</td>
<td>23.6</td>
<td>29.2</td>
<td>38.5</td>
<td>47.9</td>
<td>20.3</td>
<td>20.0</td>
<td>35.8</td>
<td>30.8</td>
<td>41.0</td>
<td>20.1</td>
<td>30.2</td>
</tr>
</tbody>
</table>

- Improvements over the state of the art without external training data
- With external training data: comparable to best methods [Wang et al.,'14]
Summary and outlook

- State-of-the-art weakly supervised object detection performance
  - Strong appearance cues for recognition: FV and CNN descriptor
  - Re-localization bias suppression: Multi-fold MIL training
  - Recognition and localization decoupling: hypothesis refinement

- From here on forward:
  - Dealing with noise on the image labels (e.g., google-image download)
  - Concurrent training of categories: leverage explaining away
  - Richer interactions between recognition and segmentation

- Relevant publications
  - “Multi-fold MIL training for weakly supervised object localization”, CVPR'14
  - Journal paper under review: CNN features and refinement
Object Detection with Incomplete Supervision

Jakob Verbeek

LEAR team, INRIA, Grenoble, France

Joint work with: Gokberk Cinbis and Cordelia Schmid