Incremental Learning of Latent Structural SVM for Weakly Supervised Image Classification

Thibaut Durand [1], Nicolas Thome [1], Matthieu Cord [1], David Picard [2]

(1) Sorbonne Universités, UPMC Univ Paris 06, UMR 7606, LIP6
(2) ETIS/ENSEA, University of Cergy-Pontoise, CNRS, UMR 8051

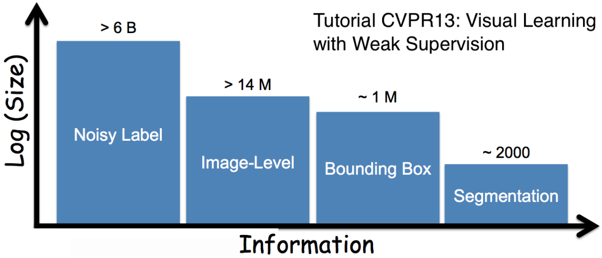
ICIP 2014

Outline

Outline

Context

- Address the problem of weakly supervised object classification
- The goal is to predict the label of the image using object position as latent variable
- Training data only provides image-level annotation (presence/absence of each category)



Context

- Model the (unknown) object location using latent variables
- Desired output during test time: predicted image label

Context

- Learning a **joint model** for both localization and classification
- Widely-used approach:
  - Latent SVM (LSVM) [PAMI10]
  - Latent Structural SVM (LSSVM - extension to structured output) [ICML09]
- Excellent performances for detection tasks
- Performances for categorization are less impressive
- 2 limitations:
  - Computation is very demanding
  - Optimization problem is hard (non-convex)

[PAMI10: Felzenszwalb, Girshick, McAllester, Ramanan. *Object detection with discriminatively trained part based models*]

[ICML09: Yu, Joachims. *Learning structural svms with latent variables*]

Context

- Popular approach for modeling object positions: sliding window

Context

- Popular approach for modeling object positions: sliding window

Context

- Popular approach for modeling object positions: sliding window

Context

- Popular approach for modeling object positions: sliding window

Context

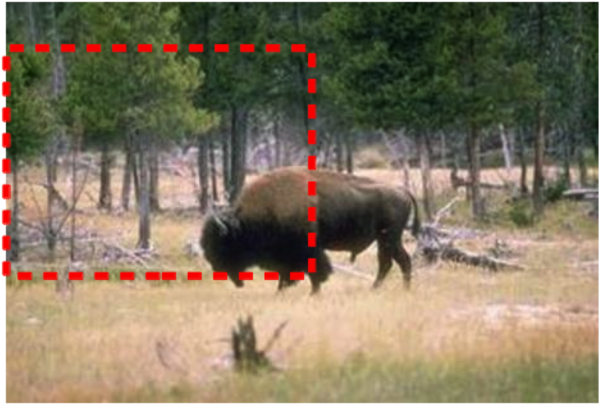
- Popular approach for modeling object positions: sliding window

Context

- Popular approach for modeling object positions: sliding window

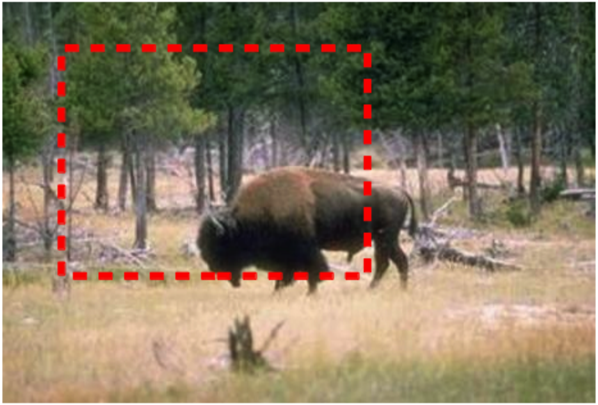
Context

- Popular approach for modeling object positions: sliding window



Context

- Popular approach for modeling object positions: sliding window



Context

- Popular approach for modeling object positions: sliding window

Context

- Popular approach for modeling object positions: sliding window

Context

- Popular approach for modeling object positions: sliding window

Contributions

- Propose an original evolution of the latent parameter space based on **cropping**
- Explore only some boxes at each iteration
- Speed-up training and inference
- Incremental Latent Structural SVM (ILSSVM)

t

t+1

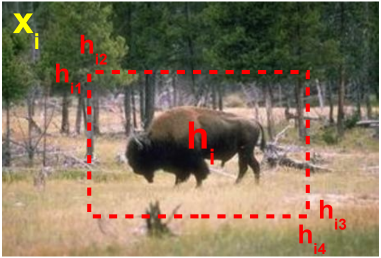
t+2

Outline

1. Context
2. Model
3. Experiments

ILSSVM model

- Multi-class problem
- LSSVM formalism
- Training data: n labeled images $\{(x_1, y_1), \ldots, (x_n, y_n)\}$
- x_i is an image
- $y_i \in \mathcal{Y} = \{1, 2, \ldots, K\}$ is a label
- Latent variable $h_i = (h_{i1}, h_{i2}, h_{i3}, h_{i4})$ represents the bounding box of the predictive object location

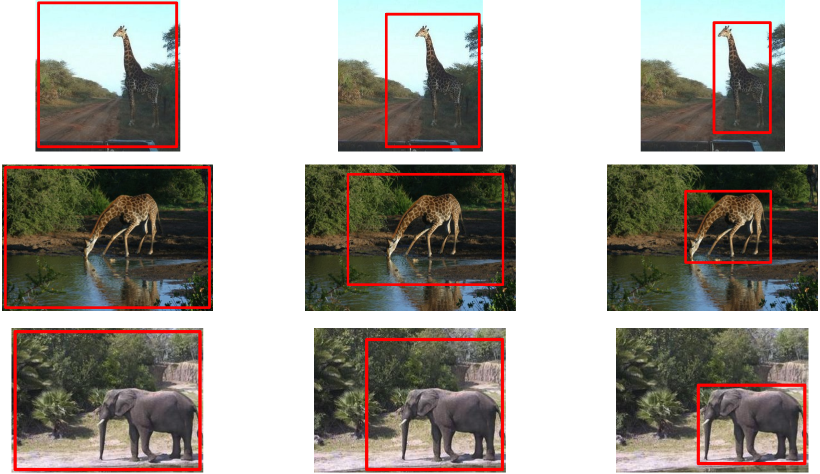


ILSSVM model

- Evolution of the latent parameter space based on cropping
- Explore only some boxes at each iteration
- **Coarse to fine approach**
- Gradually remove the background
- Evolution w.r.t. the previous latent value

Initialization ... t ... t+4

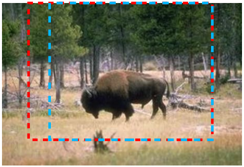
ILSSVM model



Initialization ... t ... t+n

ILSSVM model

(a) no crop

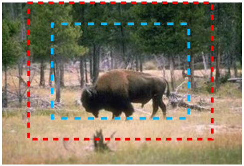


(b) crop left

(c) crop right

(d) crop top

(e) crop down



(f) crop 4 sides

Figure: Examples of possible cropping (blue boxes) for a current bounding box (red)

ILSSVM model

- Learn a LSSVM discriminant function of the form:

$$(y, h) = \arg\max_{y \in \mathcal{Y}, h \in \mathcal{H}} \langle w, \Psi(x_i, y, h) \rangle \tag{1}$$

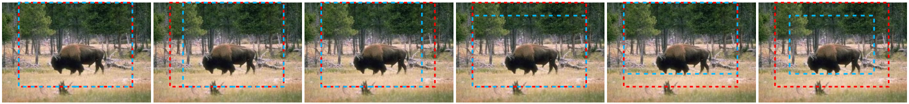
where $\Psi(x_i, y, h)$ is the joint feature.

- Objective function at the iteration t

$$\mathcal{P}_t(w) = \frac{1}{2}\|w\|^2 + \frac{C}{n}\sum_{i=1}^{n}\left(\max_{(y,h)\in\mathcal{Y}\times\mathcal{H}_i^t}[\Delta(y_i, y) + \langle w, \Psi(x_i, y, h)\rangle]\right) - \frac{C}{n}\sum_{i=1}^{n}\max_{h\in\mathcal{H}_i^t}\langle w, \Psi(x_i, y_i, h)\rangle \tag{2}$$

ILSSVM model

1. Fast in inference:
   - 6 windows/image (sliding window $>$ 1000)

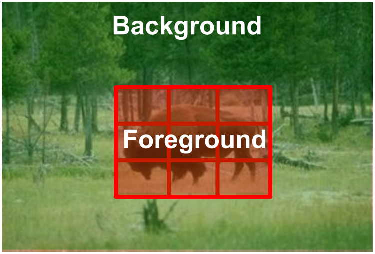


2. Better generalization (curriculum learning)
   - Easy examples = large regions
   - Start with large regions, and gradually cropped these regions

3. No require knowledge on the size and the ratio of objects $\rightarrow$ adapt itself to objects.

Image representation

- Foreground-background feature representation
- Foreground region: spatial pyramid $1 \times 1, 3 \times 3 \rightarrow$ spatial structure of the object
- Background region $\rightarrow$ strong context for classification
- BoW models using SIFT descriptors



[ECCV12: Russakovsky, Lin, Yu, Fei- Fei. Object-centric spatial pooling for image classification]

Image classification

- Use the same coarse-to-fine approach
- Start with a box initialized on the whole image
- Crop it until convergence

Image classification

- Use the same coarse-to-fine approach
- Start with a box initialized on the whole image
- Crop it until convergence

Image classification

- Use the same coarse-to-fine approach
- Start with a box initialized on the whole image
- Crop it until convergence

Image classification

- Use the same coarse-to-fine approach
- Start with a box initialized on the whole image
- Crop it until convergence

Image classification

- Use the same coarse-to-fine approach
- Start with a box initialized on the whole image
- Crop it until convergence

Image classification

- Use the same coarse-to-fine approach
- Start with a box initialized on the whole image
- Crop it until convergence

Image classification

- Use the same coarse-to-fine approach
- Start with a box initialized on the whole image
- Crop it until convergence

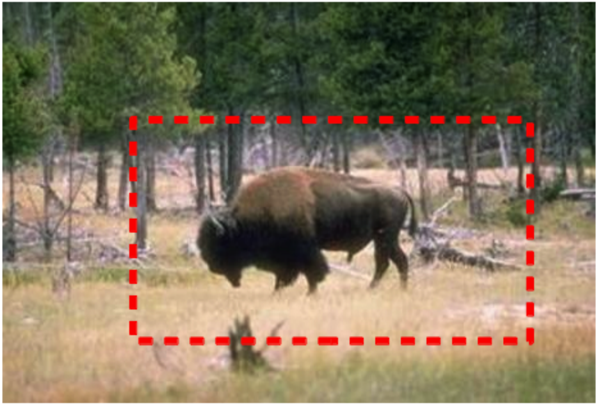


Image classification

- Use the same coarse-to-fine approach
- Start with a box initialized on the whole image
- Crop it until convergence

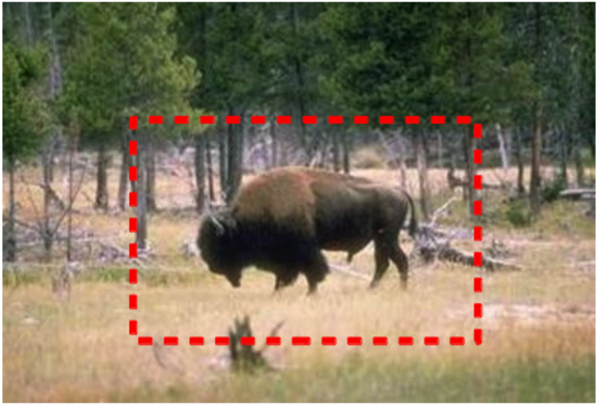


Image classification

- Use the same coarse-to-fine approach
- Start with a box initialized on the whole image
- Crop it until convergence

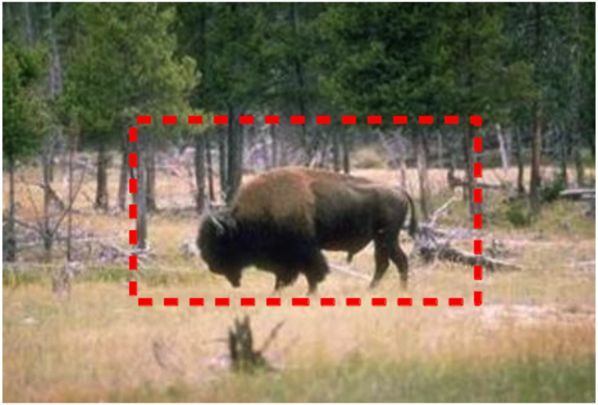


Image classification

- Use the same coarse-to-fine approach
- Start with a box initialized on the whole image
- Crop it until convergence

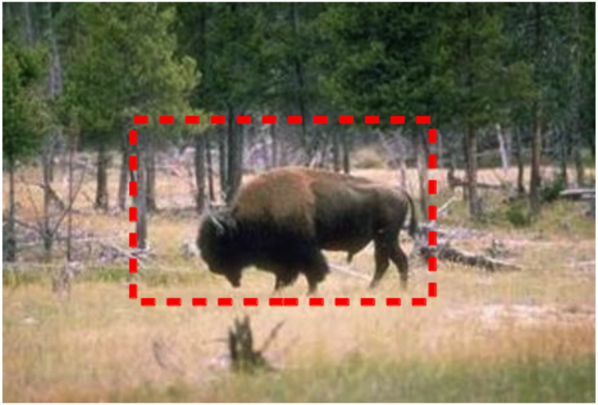


Image classification

- Use the same coarse-to-fine approach
- Start with a box initialized on the whole image
- Crop it until convergence

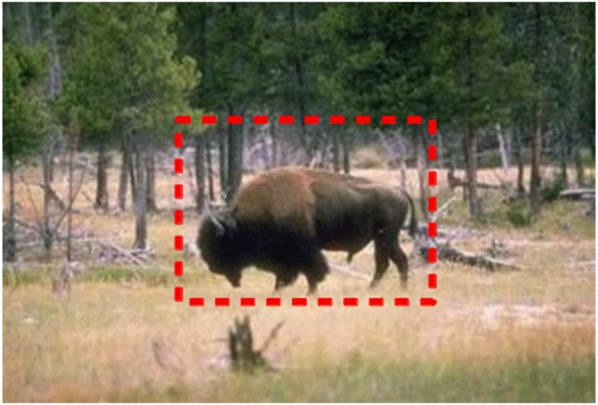
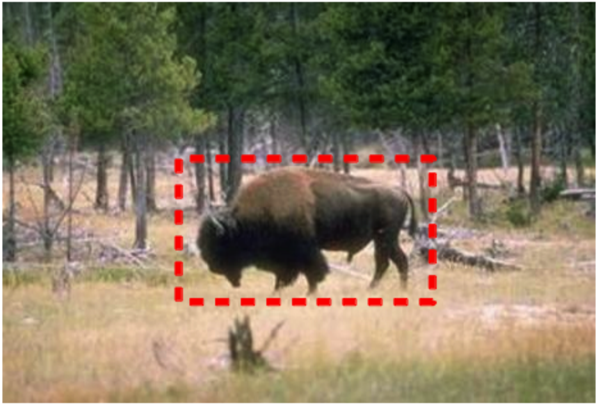


Image classification

- Use the same coarse-to-fine approach
- Start with a box initialized on the whole image
- Crop it until convergence



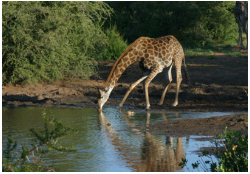
Outline

Mammal dataset

(a) bison

(b) deer

(c) elephant



(d) giraffe

(e) llama

(f) rhino

Figure: Images of the different categories of Mammal dataset

Results

split	SW (6 scales)	ILSSVM
1	22,58	12,90
2	29,03	25,81
3	22,58	22,58
4	16,13	25,81
5	45,16	38,71
6	25,81	22,58
7	35,48	16,13
8	25,81	12,90
9	35,48	22,58
10	35,48	32,26
mean	$29,35 \pm 8,53$	$23,23 \pm 8,16$

Table: Classification error for the 10 splits for multi-scale sliding window (SW) and our method (ILSSVM)

Computation time

method	time
ILSSVM	1 h
one-scale sliding window	3 h
multi-scale sliding window (6 scales, 1 ratio)	30 h
multi-scale sliding window (6 scales, 6 ratios)	250 h

Table: Time Comparisons for the ten splits on 1 CPU

Parameter of evolution of the latent variables

- Influence of the crop step
- Step proportional to the maximum of the width or height

crop step (%)	13	17	20	25	30
classification error (%)	25,81	23,87	23,23	23,55	25,16

Table: Evolution of classification error with respect to the crop step

- Robust to this parameter (small variation <3%)
- More robust than the scale in sliding window (variation > 20%)

Qualitative results of predicted boxes

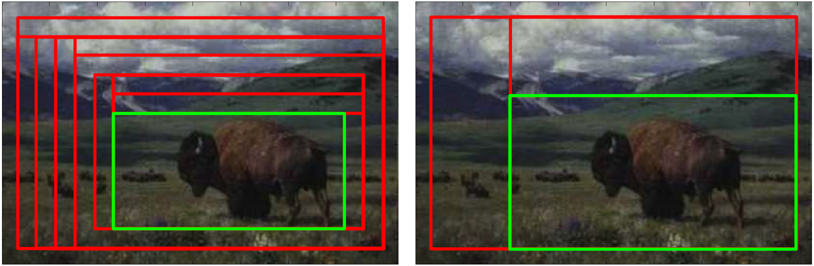


Figure: Examples of predicted boxes for a step of 5% (left) and 20% (right) at different iterations. The green box is the final box

Qualitative results of predicted boxes

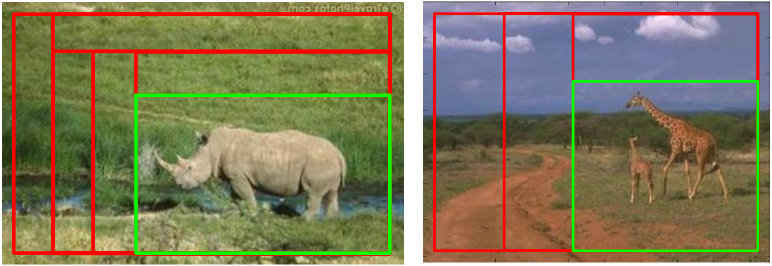


Figure: Examples of predicted boxes

Conclusion

- Original **coarse-to-fine approach** for weakly supervised image classification based on Latent Structural SVM formulation
- **Small and incremental** latent parameter space
- Find better optimum with small computation time

Thank you for your attention!

Questions?

Thibaut Durand[(1)]	thibaut.durand@lip6.fr
Nicolas Thome[(1)]	nicolas.thome@lip6.fr
Matthieu Cord[(1)]	matthieu.cord@lip6.fr
David Picard[(2)]	picard@ensea.fr

(1) Sorbonne Universités, UPMC Univ Paris 06, UMR 7606, LIP6
(2) ETIS/ENSEA, University of Cergy-Pontoise, CNRS, UMR 8051

Java code available on demand