Weakly Supervised Learning for Visual Recognition

Thibaut Durand

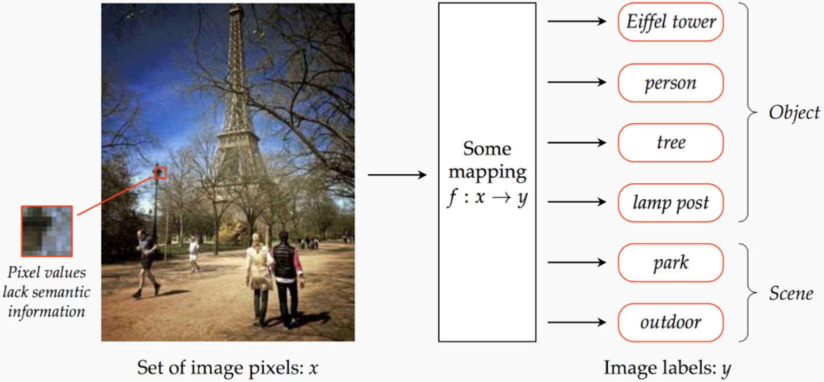
September 20, 2017

Thesis committee

Francis BACH
Patrick PÉREZ
Cordelia SCHMID
Nicolas THOME
Matthieu CORD
Alain RAKOTOMAMONJY
Véronique SERFATY

Introduction

Image classification

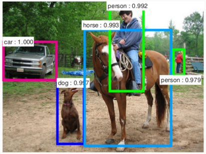


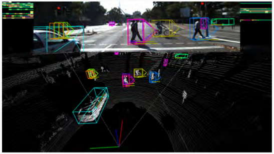
- Central problem to computer vision
- Learning parameters of f with supervised learning methods
    - Labeled training data
    - Computational resources

[Credit Hanlin Goh]

Why is image classification important?

- Immense and increasing collection of visual data
  - **2.4 billion** images are uploaded every day
  - $\mathbf{10^{12}}$ photos taken in 2016
  - Methods to exploit that collection of visual data





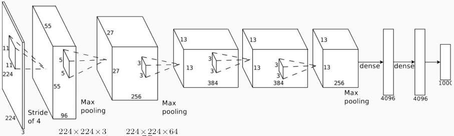
- Complementary with other visual recognition tasks

Deep ConvNets for image classification

ImageNet

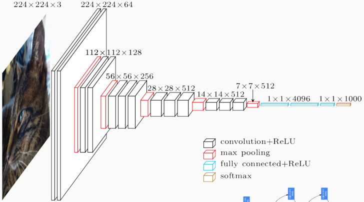
- **AlexNet**

  [Krizhevsky, NIPS12]



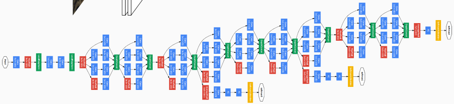
- **VGG16 / Very Deep**

  [Simonyan, ICLR15]



- **Inception**

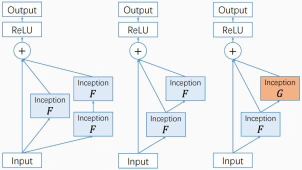
  [Szegedy, CVPR15]

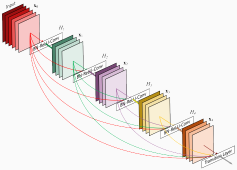


- **ResNet** [He, CVPR16]

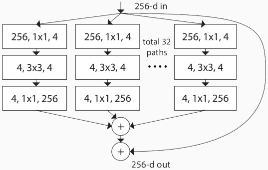
Deep ConvNets for image classification

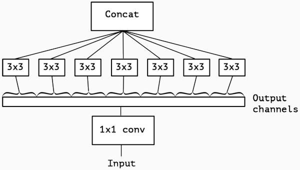
ImageNet





PolyNet [Zhang, CVPR17] **DenseNet** [Huang, CVPR17]

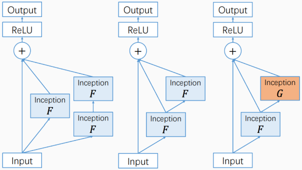




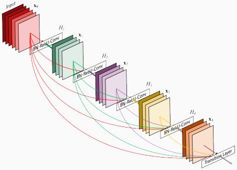
ResNeXt [Xie, CVPR17] **Xception** [Chollet, CVPR17]

Deep ConvNets for image classification

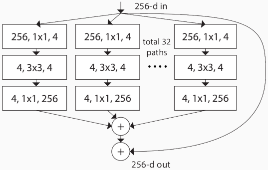
MS COCO



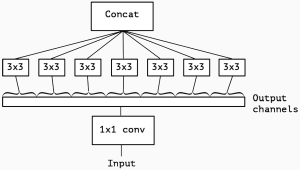
PolyNet [Zhang, CVPR17]



DenseNet [Huang, CVPR17]



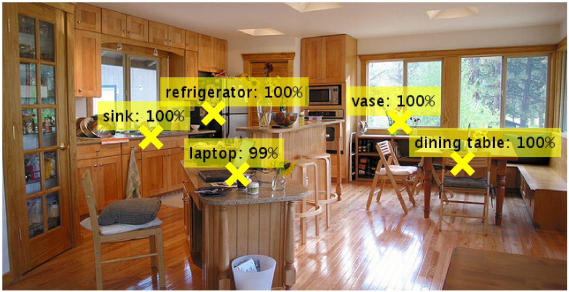
ResNeXt [Xie, CVPR17]



Xception [Chollet, CVPR17]

Contributions

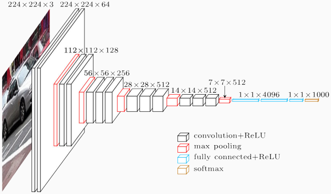
- How to use deep architecture on complex scenes?
  - Learn localized representation
- Weakly supervised learning
  - Reduce the cost of annotation: use only image-level labels
  - Make learning and recognition more challenging
    - Efficient model for structured output prediction
  - Adapt deep architecture
    - Transfer, pooling



Outline

Model: Transfer & Pooling in Deep Architecture

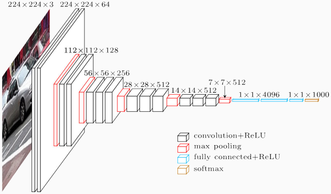
ImageNet

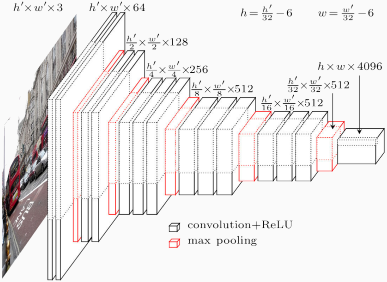

224 × 224 × 3
224 × 224 × 64
112 × 112 × 128
56 × 56 × 256
28 × 28 × 512
14 × 14 × 512
7 × 7 × 512
1 × 1 × 4096
1 × 1 × 1000
convolution+ReLU
max pooling
fully connected+ReLU
softmax

?

From ImageNet to complex images: FCN

ImageNet

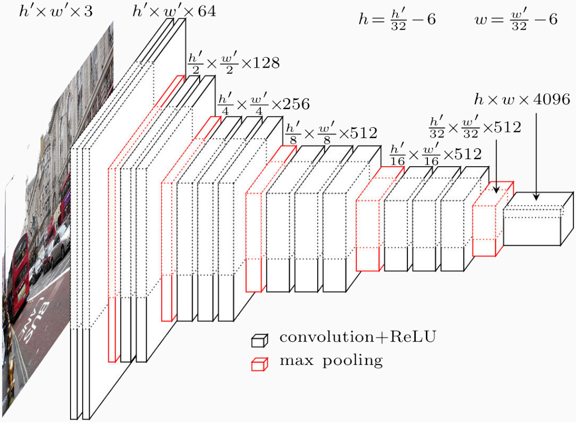


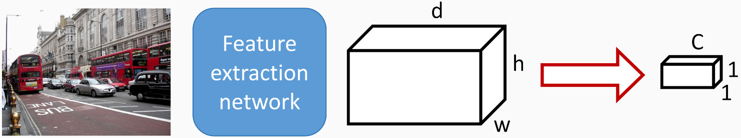


fully convolutional network

feature sharing, efficient computation, arbitrary-sized input images

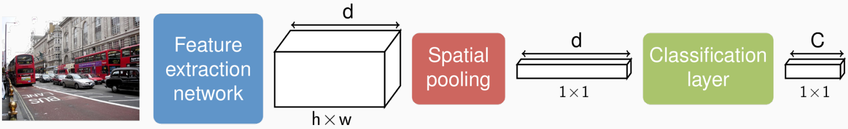
Fully convolutional network (FCN)



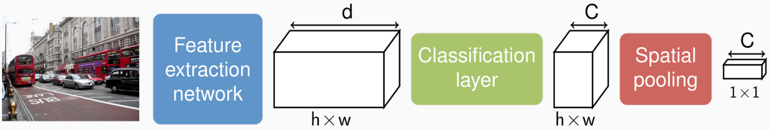


Feature vs class score pooling

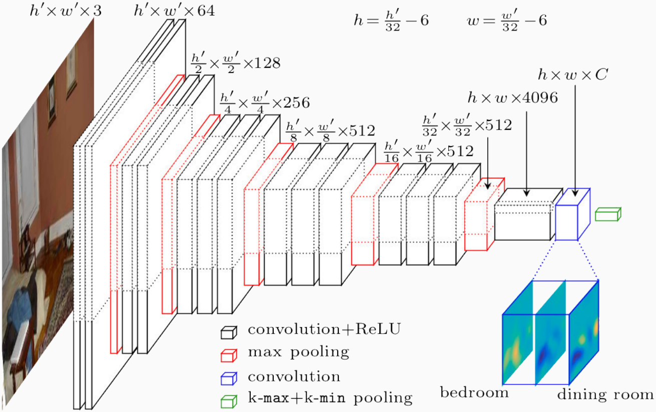
- Classical strategy: **feature pooling**
  - GAP, ResNet, Inception, VGG16, ...
  - No spatial class information



- Our strategy: **class score pooling**
  - Spatial class information
  - Better performances

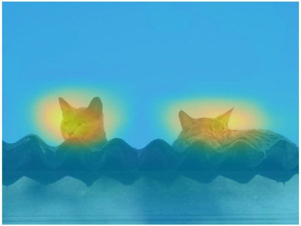


- **Class Activation Maps (CAM)** for WELDON



- Invariant to object location
- Exploit CAM: localization, segmentation

Class activation maps

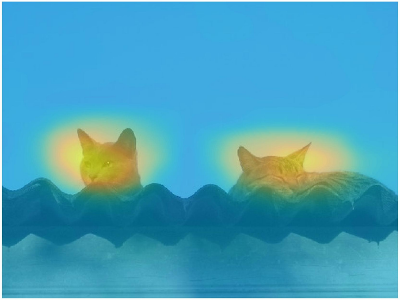

bus | cat | horse



bird | bottle | bicycle

Outline

How to pool?



map z^c

spatial pooling $\longrightarrow$ • score y^c

Max [Oquab, CVPR15]

$$y^c = \max_{i,j} z^c_{ij}$$

Use 1 region

Average (GAP) [Zhou, CVPR16]

$$y^c = \frac{1}{N} \sum_{i,j} z^c_{ij}$$

Use all regions

Average pooling limitation

- Classifying with all regions
- Not efficient for small objects: lots of "noisy" regions

Max pooling

$$y^c = \max_{i,j} z_{ij}^c \quad (1)$$

- Classifying only with the max scoring region

- Loss of contextual information

Max pooling

$$y^c = \max_{i,j} z_{ij}^c \tag{1}$$

- Classifying only with the max scoring region

- Loss of contextual information

- **Pooling function** $$y^c = \max_{i,j} z_{ij}^c + \min_{i,j} z_{ij}^c \qquad (2)$$
- $\mathbf{h}^+$: presence of the class $\rightarrow$ high $\mathbf{h}^+$
- $\mathbf{h}^-$: localized evidence of the absence of class: **negative evidence**

true class

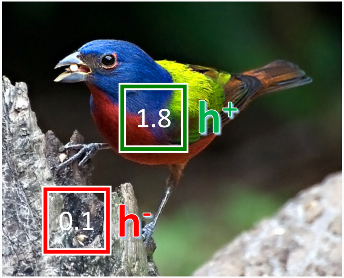
painted bunting

wrong class

indigo bunting

- **Pooling function** $$y^c = \max_{i,j} z_{ij}^c + \min_{i,j} z_{ij}^c \quad (2)$$
- $\mathbf{h}^+$: presence of the class $\rightarrow$ high $\mathbf{h}^+$
- $\mathbf{h}^-$: localized evidence of the absence of class: **negative evidence**

true class



painted bunting

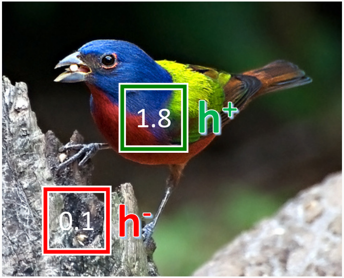
wrong class

indigo bunting

`max+min` **pooling**

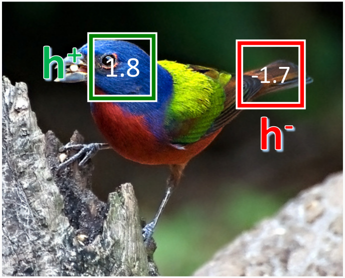
- **Pooling function** $$y^c = \max_{i,j} z_{ij}^c + \min_{i,j} z_{ij}^c \qquad (2)$$
- $\mathbf{h}^+$: presence of the class $\rightarrow$ high $\mathbf{h}^+$
- $\mathbf{h}^-$: localized evidence of the absence of class: **negative evidence**

true class



painted bunting

wrong class

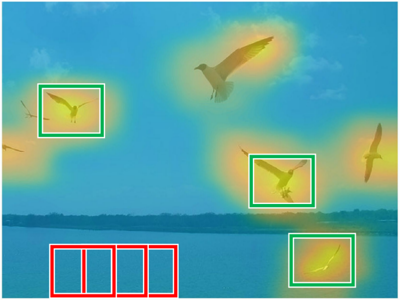


indigo bunting

WELDON pooling

- Extension of `max+min` pooling
- Using several regions, more robust region selection

k=1



k=3

WELDON pooling

- Extension of `max+min` pooling
- Using several regions, more robust region selection

$$y^c = s^{top}_{k^+}(z^c) + s^{low}_{k^-}(z^c) \tag{3}$$

$$s^{top}_{k^+}(z^c) = \frac{1}{k^+} \sum_{i=1}^{k^+} i\text{-th-max}(z^c) \tag{4}$$

$$s^{low}_{k^-}(z^c) = \frac{1}{k^-} \sum_{i=1}^{k^-} i\text{-th-min}(z^c) \tag{5}$$

WILDCAT pooling

- `max+min` pooling:
  - Both types of region are important
  - Complementary information
  - Not the same importance
- Pooling function

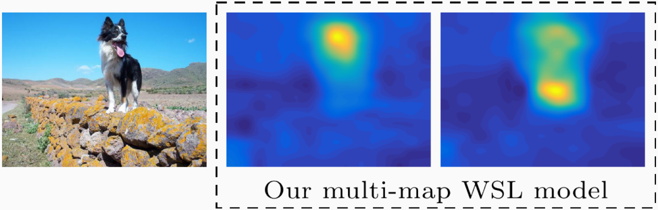
$$y^c = s_{k^+}^{top}(z^c) + \alpha \cdot s_{k^-}^{low}(z^c) \tag{6}$$

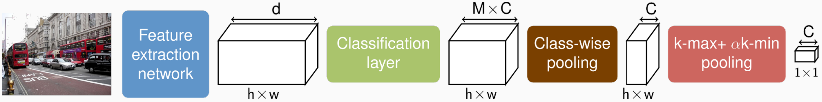
  - $\alpha \in [0, 1]$: trade off parameter

Pooling	k^+	k^-	α
`max`	1	0	0
GAP	n	0	0
`max+min`	1	1	1
WELDON	k	k	1

- WELDON: 1 model per class
    - Generalization to M models per class
    - Catch multiple class-related modalities

$$z_{ij}^{c} = \sum_{m=1}^{M} z_{ij}^{cm} \tag{7}$$



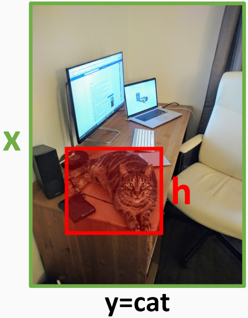


Learning & Optimization

Notations

Variable	Notation	Train	Test
Input	$\mathbf{x}$	observed	observed
Output	$\mathbf{y}$	observed	unobserved
Latent	$\mathbf{h}$	unobserved	unobserved

- $\mathbf{y}^\star$: ground-truth label
- $\mathbf{w}$: model parameters
- $\Psi(\mathbf{x},\mathbf{y},\mathbf{h}) = \psi(\mathbf{y},\Phi(\mathbf{x},\mathbf{h}))$ joint feature map
  - $\Phi(\mathbf{x},\mathbf{h})$: feature map (deep)
- $\mathbf{h}_\mathbf{y}^+ = \arg\max_{\mathbf{h}\in\mathcal{H}} \ \langle \mathbf{w}, \Psi(\mathbf{x},\mathbf{y},\mathbf{h})\rangle$
- $\mathbf{h}_\mathbf{y}^- = \arg\min_{\mathbf{h}\in\mathcal{H}} \ \langle \mathbf{w}, \Psi(\mathbf{x},\mathbf{y},\mathbf{h})\rangle$
- Optimization problem:

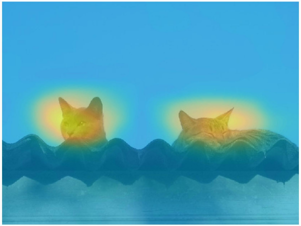


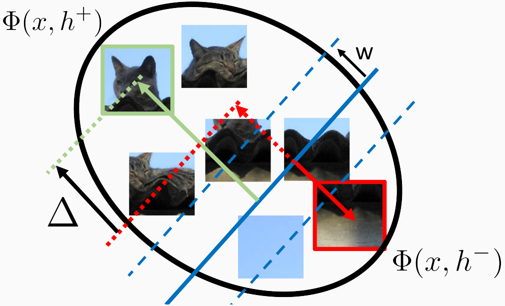
$$\min_{\mathbf{w}} \ \Omega(\mathbf{w}) + C\mathcal{L}(\mathbf{w},\mathcal{D}) \qquad \mathcal{D}: \text{dataset}$$

Feature map: $\Psi(\mathbf{x}, \mathbf{y}, \mathbf{h}) = \frac{\mathbf{y}}{2}\Phi(\mathbf{x}, \mathbf{h})$ $\qquad$ $\mathbf{y} \in \{-1, 1\}$

Prediction $\quad s_{\mathbf{w}}(\mathbf{x}) = \langle \mathbf{w}, \Phi(\mathbf{x}, h^+) \rangle + \langle \mathbf{w}, \Phi(\mathbf{x}, h^-) \rangle \qquad (8)$

- $s_{\mathbf{w}}(\mathbf{x}) > 0$: positive class
- $s_{\mathbf{w}}(\mathbf{x}) < 0$: negative class





Constraint: $\forall i \in \mathcal{D} \quad y_i^\star \left[\langle \mathbf{w}, \Phi(\mathbf{x}_i, h_i^+) + \Phi(\mathbf{x}_i, h_i^-) \rangle \right] \geq 1 \qquad (9)$

Objective function

$$\mathcal{P}(\mathbf{w}) = \frac{1}{2}\|\mathbf{w}\|^2 + C\mathcal{L}(\mathbf{w},\mathcal{D}) \tag{10}$$

$$\mathcal{L}(\mathbf{w},\mathcal{D}) = \frac{1}{N}\sum_{i\in\mathcal{D}}\left[1 - y_i^\star\left(\max_{h\in\mathcal{H}}\langle\mathbf{w},\Phi(\mathbf{x}_i,h)\rangle + \min_{h\in\mathcal{H}}\langle\mathbf{w},\Phi(\mathbf{x}_i,h)\rangle\right)\right]_+$$

$$[z]_+ = \max(0,z)$$

Objective function

$$\mathcal{P}(\mathbf{w}) = \frac{1}{2}\|\mathbf{w}\|^2 + C\mathcal{L}(\mathbf{w}, \mathcal{D}) \tag{10}$$

$$\mathcal{L}(\mathbf{w}, \mathcal{D}) = \frac{1}{N}\sum_{i \in \mathcal{D}} \left[1 - y_i^\star \left(\max_{h \in \mathcal{H}} \langle \mathbf{w}, \Phi(\mathbf{x}_i, h)\rangle + \min_{h \in \mathcal{H}} \langle \mathbf{w}, \Phi(\mathbf{x}_i, h)\rangle\right)\right]_+$$

$$[z]_+ = \max(0, z)$$

- $\min_{\mathbf{w}} \mathcal{P}(\mathbf{w})$: non-convex optimization problem
- Re-write the objective as a **difference of convex functions**

$$\mathcal{P}(\mathbf{w}) = u(\mathbf{w}) - v(\mathbf{w}) \tag{11}$$

  - u and v are convex on $\mathbf{w}$

Algorithm 1 for training with CCCP

Input: training set $\{(\mathbf{x}_i, y_i)\}_{i=1,\ldots,N}$

1: Initialize model
2: Linearize the concave part $-v$
3: **repeat**
4: Solve convexified problem
5: Linearize the concave part $-v$ at the current solution
6: **until** stopping criterion reached

Solver

- Primal: stochastic gradient descent
- Dual: cutting plane algorithm

Outline

Pair of latent variables

$$\mathbf{h}^{+}_{i,\mathbf{y}} = \arg\max_{\mathbf{h}\in\mathcal{H}} \langle \mathbf{w}, \Psi(\mathbf{x}_i, \mathbf{y}, \mathbf{h}) \rangle \tag{12}$$

$$\mathbf{h}^{-}_{i,\mathbf{y}} = \arg\min_{\mathbf{h}\in\mathcal{H}} \langle \mathbf{w}, \Psi(\mathbf{x}_i, \mathbf{y}, \mathbf{h}) \rangle \tag{13}$$

Scoring function

$$s_{\mathbf{w}}(\mathbf{x}_i, \mathbf{y}) = \langle \mathbf{w}, \Psi(\mathbf{x}_i, \mathbf{y}, \mathbf{h}^{+}_{i,\mathbf{y}}) \rangle + \langle \mathbf{w}, \Psi(\mathbf{x}_i, \mathbf{y}, \mathbf{h}^{-}_{i,\mathbf{y}}) \rangle \tag{14}$$

Prediction function

$$\hat{\mathbf{y}}_i = f_{\mathbf{w}}(\mathbf{x}_i) = \arg\max_{\mathbf{y}\in\mathcal{Y}} s_{\mathbf{w}}(\mathbf{x}_i, \mathbf{y}) \tag{15}$$

Learning formulation

- Enforce the constraint

$$\forall \mathbf{y} \neq \mathbf{y}_i^\star, \quad s_\mathbf{w}(\mathbf{x}_i, \mathbf{y}_i^\star) \geq \Delta(\mathbf{y}_i^\star, \mathbf{y}) + s_\mathbf{w}(\mathbf{x}_i, \mathbf{y}) \tag{16}$$

- $\Delta(\mathbf{y}_i^\star, \mathbf{y}) \geq 0$: user-specified loss (domain knowledge)

Objective function

$$\mathcal{P}(\mathbf{w}) = \frac{1}{2}\|\mathbf{w}\|^2 + \frac{C}{N}\sum_{i=1}^{N} \mathcal{L}_\mathbf{w}(\mathbf{x}_i, \mathbf{y}_i^\star) \tag{17}$$

$$\mathcal{L}_\mathbf{w}(\mathbf{x}_i, \mathbf{y}_i^\star) = \max_{\mathbf{y}\in\mathcal{Y}} [\Delta(\mathbf{y}_i^\star, \mathbf{y}) + s_\mathbf{w}(\mathbf{x}_i, \mathbf{y}) - s_\mathbf{w}(\mathbf{x}_i, \mathbf{y}_i^\star)] \tag{18}$$

Optimization $\quad \min_\mathbf{w} \mathcal{P}(\mathbf{w})$

- Non-convex cutting plane algorithm [Do, JMLR12]

Instantiation

Definition

- Joint feature map Ψ
- Loss function Δ

Solver

- Inference problem

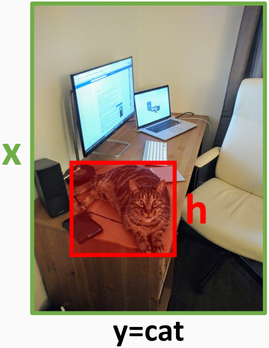
$$\hat{\mathbf{y}} = \arg\max_{\mathbf{y} \in \mathcal{Y}} s_{\mathbf{w}}(\mathbf{x}_i, \mathbf{y}) \tag{19}$$

- Loss-augmented inference (LAI) problem

$$\bar{\mathbf{y}} = \arg\max_{\mathbf{y} \in \mathcal{Y}} \Delta(\mathbf{y}_i^\star, \mathbf{y}) + s_{\mathbf{w}}(\mathbf{x}_i, \mathbf{y}) \tag{20}$$

Multi-class instantiation

- **Input x**: image
- **Output y**: multi-class label
  $\mathbf{y} \in \mathcal{Y} = \{1, \dots, K\}$
- **Latent h**: region
- **Loss function** Δ: 0/1 loss
- **Joint feature map** Ψ



$$\Psi(\mathbf{x}, \mathbf{y}, \mathbf{h}) = [I(\mathbf{y} = 1)\Phi(\mathbf{x}, \mathbf{h}), \dots, I(\mathbf{y} = K)\Phi(\mathbf{x}, \mathbf{h})] \in \mathbb{R}^{Kd} \quad (21)$$

- $\Phi(\mathbf{x}, \mathbf{h}) \in \mathbb{R}^d$ vectorial representation of image **x** at location **h**

- Inference and LAI: exhaustive search

AP ranking instantiation

- 2 classes: *positive* ($\mathcal{P}$) vs *negative* ($\mathcal{N}$)
- **Input**: all the examples $\mathbf{x} = \{\mathbf{x}_i, i = 1, \dots, N\}$.
- **Output**: ranking matrix $\mathbf{y}$ of size $N \times N$ providing an ordering of the training examples
  - $y_{ij} = 1$ if $\mathbf{x}_i \prec_{\mathbf{y}} \mathbf{x}_j$ *i.e.* $\mathbf{x}_i$ is ranked ahead of $\mathbf{x}_j$;
  - $y_{ij} = -1$ if $\mathbf{x}_j \prec_{\mathbf{y}} \mathbf{x}_i$ *i.e.* $\mathbf{x}_j$ is ranked ahead of $\mathbf{x}_i$;
  - $y_{ij} = 0$ if $\mathbf{x}_i$ and $\mathbf{x}_j$ are assigned the same rank.
- **Loss function** $\Delta(\mathbf{y}^\star, \mathbf{y}) = 1 - AP(\mathbf{y}^\star, \mathbf{y})$

- Optimizing AP with latent variable: very complex problem
- No efficient solution for max pooling model: LSSVM [Yu, ICML09]
- Approximate solution: LAPSVM [Behl, TPAMI15]

Aseem Behl and Pritish Mohapatra and C. V. Jawahar and M. Pawan Kumar
Optimizing Average Precision Using Weakly Supervised Data.
In *IEEE Trans. on Pattern Analysis and Machine Intelligence (TPAMI)*, 2015.

AP ranking instantiation

- 2 classes: *positive* ($\mathcal{P}$) vs *negative* ($\mathcal{N}$)
- **Input**: all the examples $\mathbf{x} = \{\mathbf{x}_i, i = 1, \dots, N\}$.
- **Output**: ranking matrix $\mathbf{y}$ of size $N \times N$ providing an ordering of the training examples
  - $y_{ij} = 1$ if $\mathbf{x}_i \prec_{\mathbf{y}} \mathbf{x}_j$ *i.e.* $\mathbf{x}_i$ is ranked ahead of $\mathbf{x}_j$;
  - $y_{ij} = -1$ if $\mathbf{x}_j \prec_{\mathbf{y}} \mathbf{x}_i$ *i.e.* $\mathbf{x}_j$ is ranked ahead of $\mathbf{x}_i$;
  - $y_{ij} = 0$ if $\mathbf{x}_i$ and $\mathbf{x}_j$ are assigned the same rank.
- **Loss function** $\Delta(\mathbf{y}^\star, \mathbf{y}) = 1 - AP(\mathbf{y}^\star, \mathbf{y})$
- **Joint feature map**

$$\Psi(\mathbf{x}, \mathbf{y}, \mathbf{h}) = \frac{1}{|\mathcal{P}||\mathcal{N}|} \sum_{p \in \mathcal{P}} \sum_{n \in \mathcal{N}} y_{pn} (\Phi(\mathbf{x}_p, \mathbf{h}_{p,n}) - \Phi(\mathbf{x}_n, \mathbf{h}_{n,p})) \quad (22)$$

  - $\Phi(\mathbf{x}, \mathbf{h}) \in \mathbb{R}^d$ vectorial representation of image $\mathbf{x}$ at location $\mathbf{h}$

Proposition 1.

$\forall(\mathbf{x}, \mathbf{y})$, *$s_{\mathbf{w}}(\mathbf{x}, \mathbf{y})$ for the ranking instantiation rewrites as* $\Theta(\mathbf{x}, \mathbf{y})$*:*

$$\Theta(\mathbf{x}, \mathbf{y}) = \frac{1}{|\mathcal{P}||\mathcal{N}|} \sum_{p \in \mathcal{P}} \sum_{n \in \mathcal{N}} y_{pn} \left(\langle \mathbf{w}, \Phi_{-}^{+}(\mathbf{x}_p) \rangle - \langle \mathbf{w}, \Phi_{-}^{+}(\mathbf{x}_n) \rangle \right) \quad (23)$$

where $\langle \mathbf{w}, \Phi_{-}^{+}(\mathbf{x}_i) \rangle = \max_{\mathbf{h} \in \mathcal{H}_i} \langle \mathbf{w}, \Phi(\mathbf{x}_i, \mathbf{h}) \rangle + \min_{\mathbf{h} \in \mathcal{H}_i} \langle \mathbf{w}, \Phi(\mathbf{x}_i, \mathbf{h}) \rangle$

AP ranking instantiation

Proposition 1.

$\forall(\mathbf{x},\mathbf{y})$, $s_{\mathbf{w}}(\mathbf{x},\mathbf{y})$ *for the ranking instantiation rewrites as* $\Theta(\mathbf{x},\mathbf{y})$*:*

$$\Theta(\mathbf{x},\mathbf{y}) = \frac{1}{|\mathcal{P}||\mathcal{N}|}\sum_{p\in\mathcal{P}}\sum_{n\in\mathcal{N}} y_{pn}\left(\langle \mathbf{w}, \Phi_-^+(\mathbf{x}_p)\rangle - \langle \mathbf{w}, \Phi_-^+(\mathbf{x}_n)\rangle\right) \quad (23)$$

where $\langle \mathbf{w}, \Phi_-^+(\mathbf{x}_i)\rangle = \max_{\mathbf{h}\in\mathcal{H}_i} \langle \mathbf{w}, \Phi(\mathbf{x}_i,\mathbf{h})\rangle + \min_{\mathbf{h}\in\mathcal{H}_i} \langle \mathbf{w}, \Phi(\mathbf{x}_i,\mathbf{h})\rangle$

Proposition 2.

Inference for the ranking instantiation is solved exactly by sorting the examples in descending order of score $\langle \mathbf{w}, \Phi_-^+(\mathbf{x}_i)\rangle$

Proposition 1.

$\forall(\mathbf{x},\mathbf{y})$, $s_{\mathbf{w}}(\mathbf{x},\mathbf{y})$ for the ranking instantiation rewrites as $\Theta(\mathbf{x},\mathbf{y})$:

$$\Theta(\mathbf{x},\mathbf{y}) = \frac{1}{|\mathcal{P}||\mathcal{N}|}\sum_{p\in\mathcal{P}}\sum_{n\in\mathcal{N}} y_{pn}\left(\langle \mathbf{w}, \Phi_{-}^{+}(\mathbf{x}_p)\rangle - \langle \mathbf{w}, \Phi_{-}^{+}(\mathbf{x}_n)\rangle\right) \quad (23)$$

where $\langle \mathbf{w}, \Phi_{-}^{+}(\mathbf{x}_i)\rangle = \max_{\mathbf{h}\in\mathcal{H}_i} \langle \mathbf{w}, \Phi(\mathbf{x}_i,\mathbf{h})\rangle + \min_{\mathbf{h}\in\mathcal{H}_i} \langle \mathbf{w}, \Phi(\mathbf{x}_i,\mathbf{h})\rangle$

Proposition 2.

Inference for the ranking instantiation is solved exactly by sorting the examples in descending order of score $\langle \mathbf{w}, \Phi_{-}^{+}(\mathbf{x}_i)\rangle$

Proposition 3.

Efficient solution for the loss-augmented inference (LAI) problem if there exists a solver for the fully-supervised LAI problem

Experiments

Outline

1 Model: Transfer & Pooling in Deep Architecture

2 Learning & Optimization

3 Experiments

- Classification
- Weakly supervised localization
- Weakly supervised segmentation

4 Conclusion

ImageNet

VOC

MS COCO

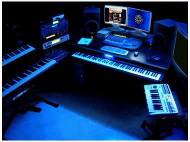
Datasets

CUB-200




MIT67



VOC Action




Datasets

Dataset	#Train	#Test	#Classes	Evaluation
VOC 07	5,011	4,952	20	MAP
VOC 12	11,540	10,991	20	MAP
VOC 12 Action	2,296	2,292	10	MAP
MS COCO	82,783	40,504	80	MAP
MIT67	5,360	1,340	67	accuracy
CUB-200	5,994	5,794	200	accuracy
ILSVRC 2012	1,281,167	50,000	1000	accuracy

- Feature extraction network: ResNet-101 pretrained on ImageNet

Method	VOC 2007	VOC 2012	MS COCO
ResNet-101	89.8	89.2	72.5
Deep MIL	-	86.3	62.8
ProNet	-	89.3	70.9
SPLeaP	88.0	-	-
WILDCAT	**95.0**	**93.4**	**80.7**

ImageNet	Top-5 error
ResNet-101 (1 crop)	6.21
ResNet-200 (10 crops)	4.93
ResNeXt-101 (1 crop)	4.4
Inception-ResNet-v2 (12 crops)	**4.1**
WILDCAT (**$M = 1$**)	4.23

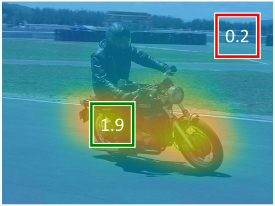
Visual results

- Negative evidence regions can be parts of other objects classes

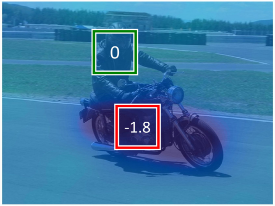
train

bus

- Multi-label: learn correlation between classes



motorbike



bottle

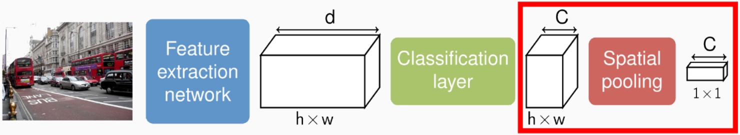
AP ranking experiments

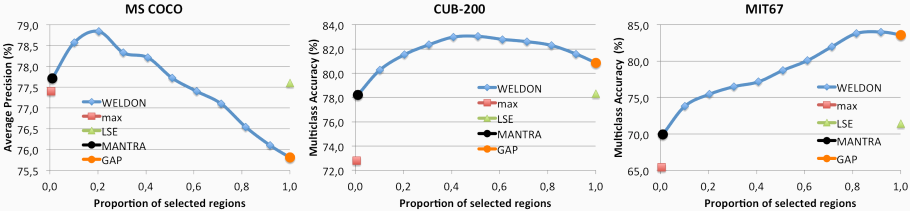
Dataset	VOC07	VOCAct	MS COCO
`max` + classif. loss	86.8	71.8	77.4
`max` + AP loss (LAPSVM)	87.9	73.3	77.9
`max+min` + classif. loss	89.9	78.5	77.7
`max+min` + AP loss	**91.2**	**80.7**	**78.7**

- Optimizing the evaluation metric during training is important

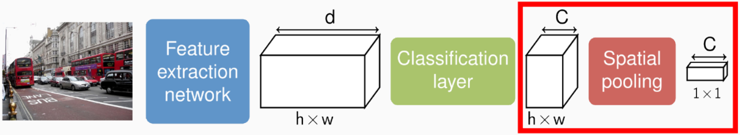
Aseem Behl and Pritish Mohapatra and C. V. Jawahar and M. Pawan Kumar
Optimizing Average Precision Using Weakly Supervised Data.
In *IEEE Trans. on Pattern Analysis and Machine Intelligence (TPAMI)*, 2015.

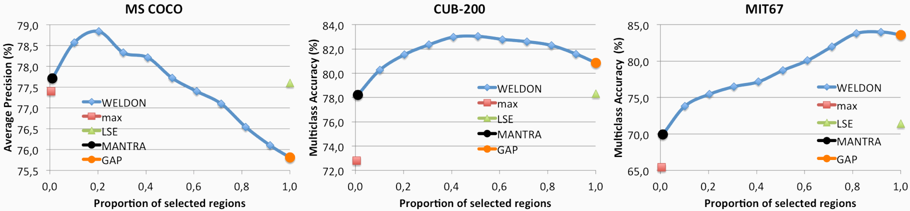
Pooling analysis





- `max` / LSSVM
- `max+min` / MANTRA
- `k-max+k-min` / WELDON
- average / GAP
- soft-max / LSE / HCRF

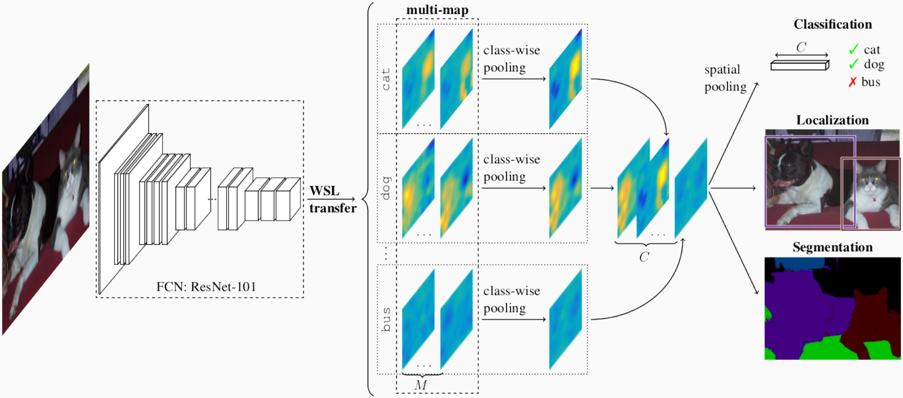




Unified pooling function

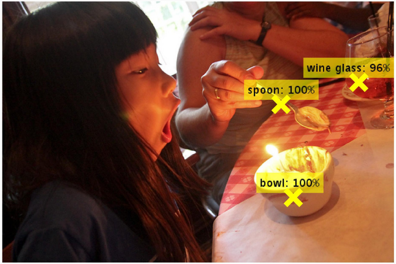
$$s_{\mathbf{w}}^{(\alpha,\beta_h^+,\beta_h^-)}(\mathbf{x},\mathbf{y}) = \frac{1}{2\beta_h^+}\log\left(\frac{1}{|\mathcal{H}|}\sum_{\mathbf{h}\in\mathcal{H}}\exp[\beta_h^+\langle\mathbf{w},\Psi(\mathbf{x}_i,\mathbf{y},\mathbf{h})\rangle]\right)$$

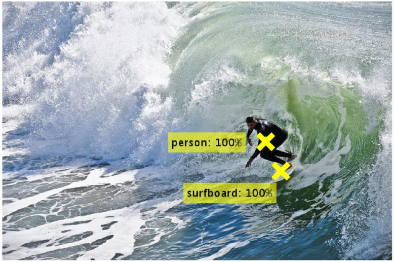
$$+\alpha\frac{1}{2\beta_h^-}\log\left(\frac{1}{|\mathcal{H}|}\sum_{\mathbf{h}\in\mathcal{H}}\exp[\beta_h^-\langle\mathbf{w},\Psi(\mathbf{x}_i,\mathbf{y},\mathbf{h})\rangle]\right)$$



- Weakly supervised localization
- Weakly supervised segmentation

Weakly supervised localization

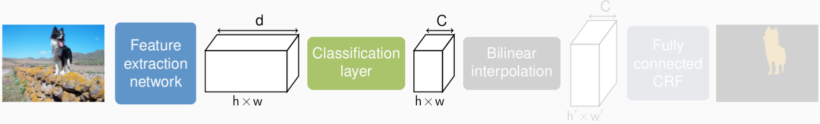




METHOD	VOC 2012	MS COCO
Deep MIL [Oquab, CVPR15]	74.5	41.2
ProNet [Sun, CVPR16]	77.7	46.4
WSLocalization [Bency, ECCV16]	79.7	49.2
WILDCAT	**82.9**	**53.4**

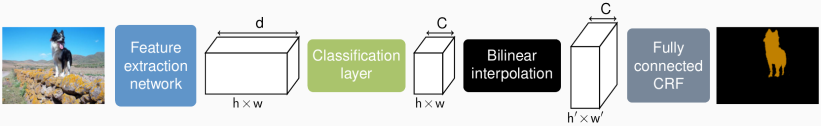
- Pointwise metric [Oquab, CVPR15]

- Test architecture



Method	Mean IoU
MIL-FCN	24.9
MIL-Base+ILP+SP-sppxl	36.6
EM-Adapt + FC-CRF	33.8
CCNN + FC-CRF	35.3
WILDCAT + FC-CRF	**43.7**

- Test architecture



Method	Mean IoU
MIL-FCN	24.9
MIL-Base+ILP+SP-sppxl	36.6
EM-Adapt + FC-CRF	33.8
CCNN + FC-CRF	35.3
WILDCAT + FC-CRF	**43.7**

Weakly supervised segmentation

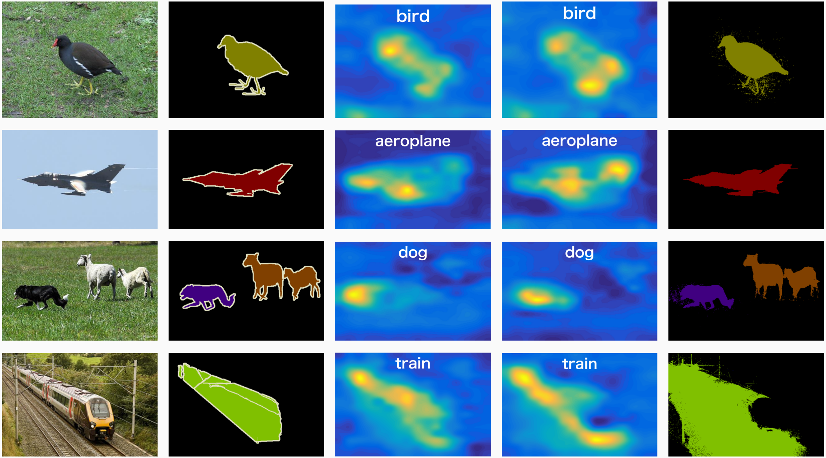


image ground truth heatmap1 heatmap2 prediction

Conclusion

Contributions

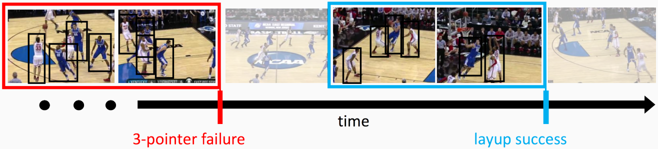
- **Pooling:** negative evidence model
  - Deep architecture
    - Can easily be integrated into any architecture
  - Latent Structured SVM framework
- **Transfer**
  - Multi-map transfer layer
- **Structured output prediction:** AP ranking
- Application on different type of data: image, text, molecule
- Publications: 1 ICCV, 2 CVPR, 2 journals under review

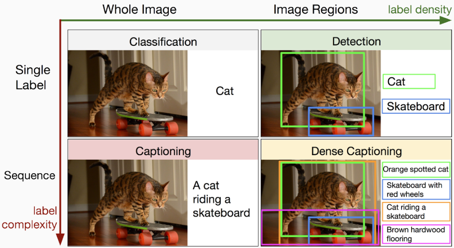
durandtibo/wildcat.pytorch

Future work

- **Pooling**
  - Learning the number of regions k^+ and k^- for each class
  - Learning the number of maps per class
- **What is the optimal architecture?**
  - Deep structure analysis / understanding
  - Learning deep architecture: convolutional neural fabrics [Saxena, NIPS16], Genetic CNN [Xie, ICCV17]

- **Deep learning for complex images**
  - Spatial resolution of detection maps: FPN [Lin, CVPR17]
  - Deep Structured ConvNets: [Chen, ICML15]
  - Applications to WSL tasks: pose estimation, segmentation, sport analytics (video)...





Publications

Thibaut Durand, Nicolas Thome, and Matthieu Cord
MANTRA: Minimum Maximum Latent Structural SVM for Image Classification and Ranking.
In *IEEE International Conference on Computer Vision (ICCV)*, 2015.

Thibaut Durand, Nicolas Thome, and Matthieu Cord
WELDON: Weakly Supervised Learning of Deep ConvNets.
In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

Thibaut Durand*, Taylor Mordan*, Nicolas Thome, and Matthieu Cord
WILDCAT: Weakly Supervised Learning of Deep ConvNets for Image Classification, Pointwise Localization and Segmentation.
In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

Under review

Thibaut Durand, Nicolas Thome, and Matthieu Cord
SyMIL: MinMax Latent SVM for Weakly Labeled Data.
In *IEEE Transactions on Neural Networks and Learning Systems*.

Thibaut Durand, Nicolas Thome, and Matthieu Cord
Exploiting Negative Evidence for WSL of Deep Structured Models.
In *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Publications

Thibaut Durand, Nicolas Thome, Matthieu Cord, and Sandra Avila
Image Classification using Object Detectors.
In *IEEE International Conference on Image Processing (ICIP)*, 2013.

Thibaut Durand, David Picard, Nicolas Thome, and Matthieu Cord
Semantic Pooling for Image Categorization using Multiple Kernel Learning.
In *IEEE International Conference on Image Processing (ICIP)*, 2014.

Thibaut Durand, Nicolas Thome, Matthieu Cord, and David Picard
Incremental Learning of Latent Structural SVM for Weakly Supervised Image Classification.
In *IEEE International Conference on Image Processing (ICIP)*, 2014.

Yue Zhu, Thibaut Durand, Eric Chenin, Marc Pignal, Patrick Gallinari, Régine Vignes-Lebbe
Using a Deep Convolutional Neural Network for Extracting Morphological Traits from Herbarium Images.
In *Proceedings of TDWG*, 2017.

Bibliography

References i

[1] Stuart Andrews, Ioannis Tsochantaridis, and Thomas Hofmann.
Support Vector Machines for Multiple-Instance Learning.
In *Advances in Neural Information Processing Systems (NIPS)*, 2003.

[2] Archith J. Bency, Heesung Kwon, Hyungtae Lee, S. Karthikeyan, and B. S. Manjunath.
Weakly Supervised Localization using Deep Feature Maps.
In *European Conference on Computer Vision (ECCV)*, 2016.

[3] Liang-Chieh Chen, Alexander Schwing, Alan Yuille, and Raquel Urtasun.
Learning deep structured models.
In *International Conference on Machine Learning (ICML)*, 2015.

[4] Francois Chollet.
Xception: Deep Learning with Depthwise Separable Convolutions.
In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[5] Thomas Deselaers and Vittorio Ferrari.
A Conditional Random Field for Multiple-Instance Learning.
In *International Conference on Machine Learning (ICML)*, 2010.

[6] Trinh-Minh-Tri Do and Thierry Artières.
Regularized bundle methods for convex and non-convex risks.
Journal of Machine Learning Research (JMLR), 2012.

[7] Peter Gehler and Olivier Chapelle.
Deterministic Annealing for Multiple-Instance Learning.
In *International Conference on Artificial Intelligence and Statistics (AISTAT)*, 2007.

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun.
Deep Residual Learning for Image Recognition.
In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[9] Justin Johnson, Andrej Karpathy, and Li Fei-Fei.
DenseCap: Fully Convolutional Localization Networks for Dense Captioning.
In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[10] Armand Joulin and Francis Bach.
A convex relaxation for weakly supervised classifiers.
In *International Conference on Machine Learning (ICML)*, 2012.

[11] M. Kim and Fernando De la Torre.
Multiple Instance Learning via Gaussian Processes.
Data Mining and Knowledge Discovery (DMKD), 2013.

[12] Alex Krizhevsky, Ilya Sutskever, and Geoffrey Hinton.
ImageNet Classification with Deep Convolutional Neural Networks.
In *Advances in Neural Information Processing Systems (NIPS)*, 2012.

[13] Gabriel Krummenacher, Cheng S. Ong, and Joachim Buhmann.
Ellipsoidal Multiple Instance Learning.
In *International Conference on Machine Learning (ICML)*, 2013.

[14] Li-Jia Li, Hao Su, Yongwhan Lim, and Li Fei-Fei.
Object Bank: An Object-Level Image Representation for High-Level Visual Recognition.
In *Int. J. Comput. Vision*, 2014.

[15] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie.
Feature Pyramid Networks for Object Detection.
In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[16] O.L. Mangasarian and E.W. Wild.
Multiple Instance Classification via Successive Linear Programming.
Journal of Optimization Theory and Applications, 2008.

[17] Maxime Oquab, Leon Bottou, Ivan Laptev, and Josef Sivic.
Is Object Localization for Free? - Weakly-Supervised Learning With Convolutional Neural Networks.
In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[18] Shreyas Saxena and Jakob Verbeek.
Convolutional Neural Fabrics.
In *Advances in Neural Information Processing Systems (NIPS)*, 2016.

[19] Karen Simonyan and Andrew Zisserman.
Very Deep Convolutional Networks for Large-Scale Image Recognition.
In *International Conference on Learning Representations (ICLR)*, 2015.

[20] Chen Sun, Manohar Paluri, Ronan Collobert, Ram Nevatia, and Lubomir Bourdev.
ProNet: Learning to Propose Object-specific Boxes for Cascaded Neural Networks.
In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[21] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich.
Going Deeper with Convolutions.
In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[22] Lingxi Xie and Alan Yuille.
Genetic CNN.
In *IEEE International Conference on Computer Vision (ICCV)*, 2017.

[23] Saining Xie, Ross Girshick, Piotr Dollar, Zhuowen Tu, and Kaiming He.
Aggregated residual transformations for deep neural networks.
In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[24] Chun-Nam Yu and Thorsten Joachims.
Learning structural svms with latent variables.
In *International Conference on Machine Learning (ICML)*, 2009.

[25] Dan Zhang, Jingrui He, Luo Si, and Richard D. Lawrence.
MILEAGE: Multiple Instance LEArning with Global Embedding.
In *International Conference on Machine Learning (ICML)*, 2013.

[26] Xingcheng Zhang, Zhizhong Li, Chen Change Loy, and Dahua Lin.
PolyNet: A Pursuit of Structural Diversity in Very Deep Networks.
In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

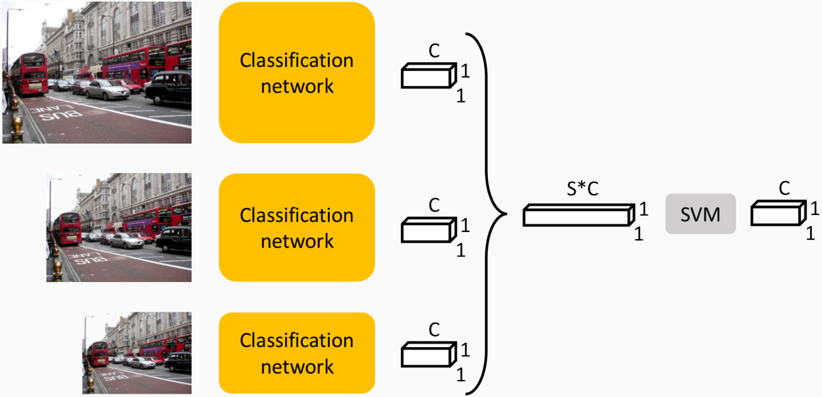
[27] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba.
Learning Deep Features for Discriminative Localization.
In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[28] Zhi-Hua Zhou, Yu-Yin Sun, and Yu-Feng Li.
Multi-instance Learning by Treating Instances As non-I.I.D. Samples.
In *International Conference on Machine Learning (ICML)*, 2009.

Appendices

Multi-scale architecture

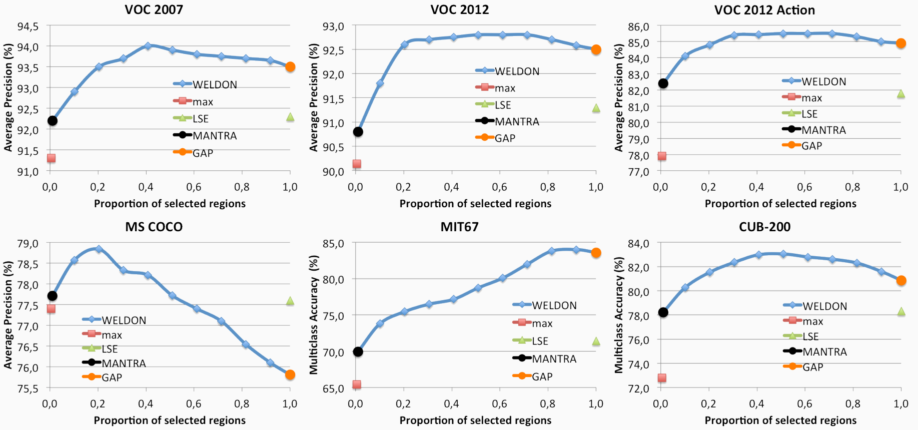
- Object Bank strategy [Li, IJCV14]
- Learn automatically the weight of each scale



State-of-the-art results

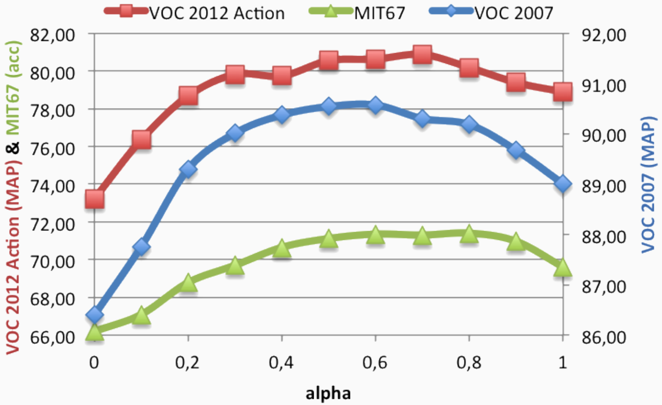
Method	CUB-200	MIT67	VOC Action
CaffeNet Places	-	68.2	-
MOP CNN	-	68.9	-
Compact Bilinear Pooling	84.0	76.2	-
ResNet-101	72.5	78.0	77.9
Spatial Transformer	84.1	-	-
Negative parts	-	77.1	-
GoogLeNet-GAP	63.0	66.6	-
SPLeaP	-	73.5	-
WILDCAT	**85.6**	**84.0**	**86.4**

Pooling analysis

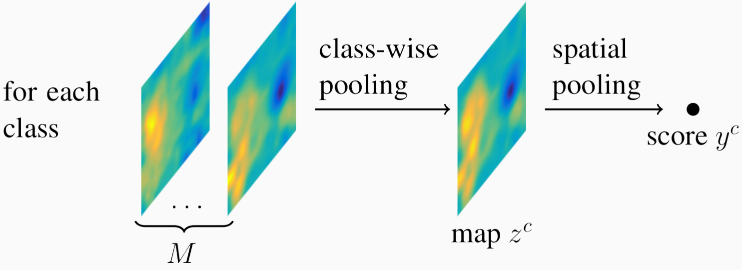


Pooling analysis

$$y^c = s_{k^+}^{top}(z^c) + \alpha \cdot s_{k^-}^{low}(z^c) \tag{24}$$



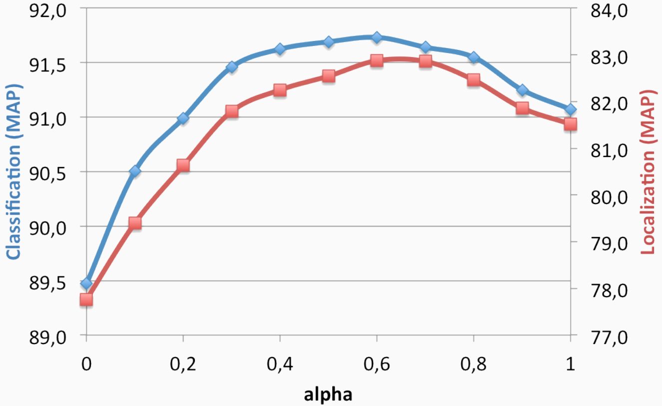
Pooling analysis



M	1	2	4	8	12	16
VOC 2007	89.0	91.0	91.6	**92.5**	92.3	92.0
VOC 2012 Action	78.9	81.5	82.1	**83.2**	83.0	82.7
MIT67	69.6	71.8	72.0	72.8	**73.1**	72.9

Weakly supervised localization

- Analysis of trade off parameter α on Pascal VOC 2012

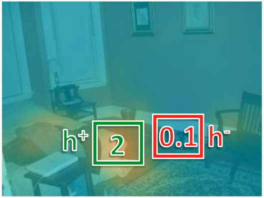


- Correlation between classification and localization

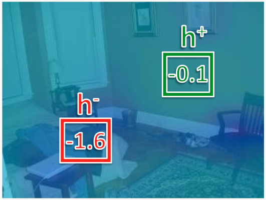
MANTRA: `max+min` pooling

- $\mathbf{h}^+$: presence of the class $\rightarrow$ high $\mathbf{h}^+$
- $\mathbf{h}^-$: localized evidence of the absence of class: **negative evidence**

original image



bedroom (2.1)

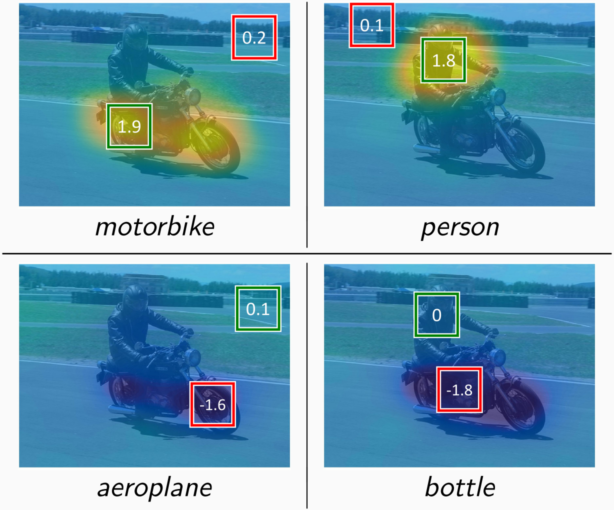


airport inside (-1.7)

dining room (0.4)

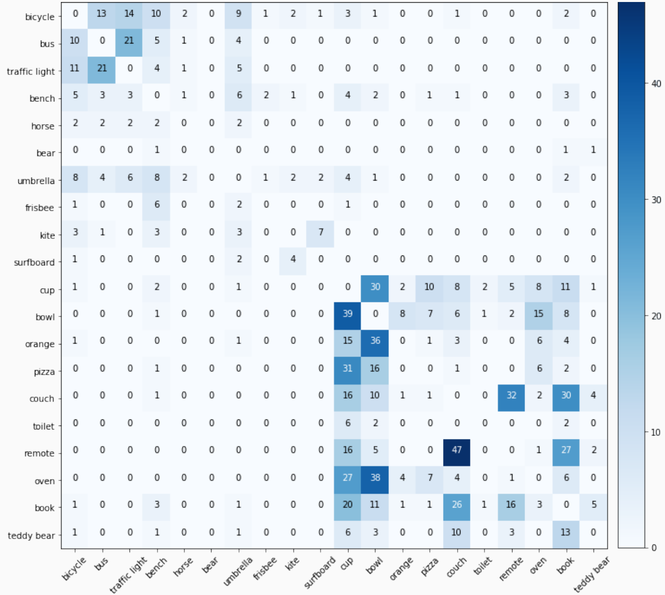
MANTRA: `max+min` pooling

- Multi-label: learn correlation between classes

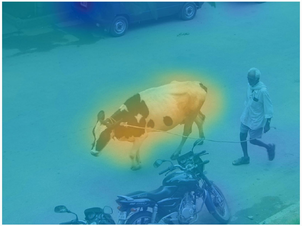


Pascal VOC 2007: co-occurence matrix

MS COCO: co-occurence matrix



Class activation maps



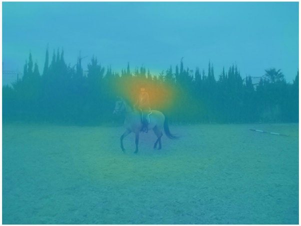
cow

motorbike

horse

person

car



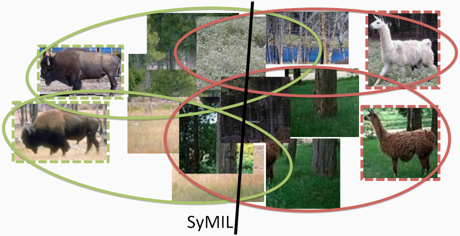
person

SyMIL

- Binary classification (*e.g. bison* vs *llama*)
- Pooling function

$$y = \begin{cases} \max_{i,j} z_{ij} & \text{if } \max_{i,j} z_{ij} \geq -\min_{i,j} z_{ij} \\ \min_{i,j} z_{ij} & \text{otherwise} \end{cases} \tag{25}$$

  - $y > 0$: *bison* class
  - $y < 0$: *llama* class



SyMIL

Re-write the objective as a **difference of convex functions**:

$$\mathcal{P}(\mathbf{w}) = u(\mathbf{w}) - v(\mathbf{w}) \tag{26}$$

- u and v are convex on $\mathbf{w}$

Property:

$$\max(0, a - b) = \max(a, b) - b \tag{27}$$

Example: first term of the loss

$$\max(0, \underbrace{1 - \max_{h\in\mathcal{H}}\langle \mathbf{w}, \Phi(\mathbf{x}_i, h)\rangle}_{\textit{concave}}) = \underbrace{\max(0, \max_{h\in\mathcal{H}}\langle \mathbf{w}, \Phi(\mathbf{x}_i, h)\rangle - 1)}_{\textit{convex}} - \underbrace{(\max_{h\in\mathcal{H}}\langle \mathbf{w}, \Phi(\mathbf{x}_i, h)\rangle - 1)}_{\textit{convex}} \tag{28}$$

- $a = 0$
- $b = -(1 - \max_{h\in\mathcal{H}}\langle \mathbf{w}, \Phi(\mathbf{x}_i, h)\rangle)$

SyMIL: difference of convex functions

$$
\begin{aligned}
\mathcal{P}(\mathbf{w}) =& u(\mathbf{w}) - v(\mathbf{w}) \\
u(\mathbf{w}) =& \frac{1}{2}\|\mathbf{w}\|^2 + \frac{C}{N}\Bigg(\sum_{i \in \mathcal{P}} \Big[\frac{N}{N^+} \max\Big(0, \max_{h \in \mathcal{H}} \langle \mathbf{w}, \Phi(\mathbf{x}_i, h) \rangle - 1\Big) \\
& + \lambda \max\Big(1 - \min_{h \in \mathcal{H}} \langle \mathbf{w}, \Phi(\mathbf{x}_i, h) \rangle, \max_{h \in \mathcal{H}} \langle \mathbf{w}, \Phi(\mathbf{x}_i, h) \rangle\Big)\Big] \\
& + \sum_{i \in \mathcal{N}} \Big[\frac{N}{N^-} \max\Big(0, -\min_{h \in \mathcal{H}} \langle \mathbf{w}, \Phi(\mathbf{x}_i, h) \rangle - 1\Big) \\
& + \lambda \max\Big(1 + \max_{h \in \mathcal{H}} \langle \mathbf{w}, \Phi(\mathbf{x}_i, h) \rangle, -\min_{h \in \mathcal{H}} \langle \mathbf{w}, \Phi(\mathbf{x}_i, h) \rangle\Big)\Big]\Bigg) \\
v(\mathbf{w}) =& \frac{C}{N}\Bigg(\sum_{i \in \mathcal{P}} \Big[\Big(\frac{N}{N^+} + \lambda\Big) \max_{h \in \mathcal{H}} \langle \mathbf{w}, \Phi(\mathbf{x}_i, h) \rangle - \frac{N}{N^+} \Big] \\
& + \sum_{i \in \mathcal{N}} \Big[-\Big(\frac{N}{N^-} + \lambda\Big) \min_{h \in \mathcal{H}} \langle \mathbf{w}, \Phi(\mathbf{x}_i, h) \rangle + \frac{N}{N^-} \Big] \Bigg)
\end{aligned}
$$

SyMIL: primal

- Linearization of the concave part $-v(\mathbf{w})$

$$\nabla_{\mathbf{w}} v(\mathbf{w}_t) = \left(\sum_{i \in \mathcal{P}} (\frac{N}{N^+} + \lambda) \Phi(\mathbf{x}_i, h^+_{i,t}) - \sum_{i \in \mathcal{N}} (\frac{N}{N^-} + \lambda) \Phi(\mathbf{x}_i, h^-_{i,t}) \right)$$

- Upper bound $-v(\mathbf{w}) \leq -\langle \mathbf{w}, \nabla_{\mathbf{w}} v(\mathbf{w}_t) \rangle$
- Convexified optimization problem

$$\mathcal{P}_t^{CCCP}(\mathbf{w}) = u(\mathbf{w}) - \langle \mathbf{w}, \nabla_{\mathbf{w}} v(\mathbf{w}_t) \rangle \tag{29}$$

SyMIL: primal (gradient)

$$\nabla_w \mathcal{P}_t^{CCCP}(\mathbf{w}) = \begin{cases} \mathbf{w} + \frac{C}{N}(D + E - (\frac{N}{N^+} + \lambda)\Phi(\mathbf{x}_i, h_{i,t}^+)) & \text{if } y_i^\star = +1 \\ \mathbf{w} + \frac{C}{N}(F + G + (\frac{N}{N^-} + \lambda)\Phi(\mathbf{x}_i, h_{i,t}^-)) & \text{otherwise} \end{cases}$$

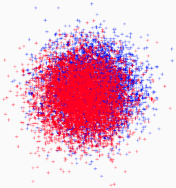
$$D = \begin{cases} \frac{N}{N^+}\Phi(\mathbf{x}_i, h_i^+) & \text{if } \langle \mathbf{w}, \Phi(\mathbf{x}_i, h_i^+)\rangle - 1 > 0 \\ 0 & \text{otherwise} \end{cases}$$

$$E = \begin{cases} -\lambda\Phi(\mathbf{x}_i, h_i^-) & \text{if } 1 - \langle \mathbf{w}, \Phi(\mathbf{x}_i, h_i^-)\rangle > \langle w, \Phi(\mathbf{x}_i, h_i^+)\rangle \\ \lambda\Phi(\mathbf{x}_i, h_i^+) & \text{otherwise} \end{cases}$$

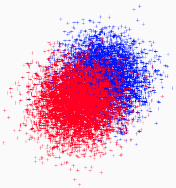
$$F = \begin{cases} -\frac{N}{N^-}\Phi(\mathbf{x}_i, h_i^-) & \text{if } -\langle \mathbf{w}, \Phi(\mathbf{x}_i, h_i^-)\rangle - 1 > 0 \\ 0 & \text{otherwise} \end{cases}$$

$$G = \begin{cases} \lambda\Phi(\mathbf{x}_i, h_i^+) & \text{if } 1 + \langle \mathbf{w}, \Phi(\mathbf{x}_i, h_i^+)\rangle > -\langle \mathbf{w}, \Phi(\mathbf{x}_i, h_i^-)\rangle \\ -\lambda\Phi(\mathbf{x}_i, h_i^-) & \text{otherwise} \end{cases}$$

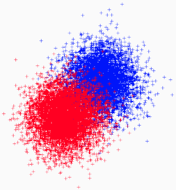
SyMIL: toy experiments



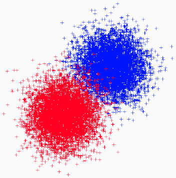
$\alpha = 0.1$



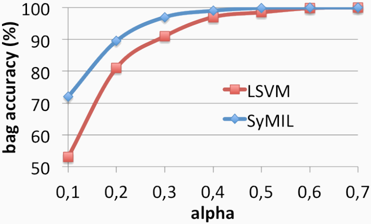
$\alpha = 0.3$



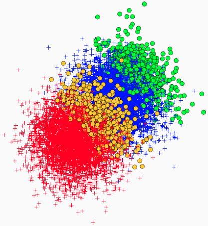
$\alpha = 0.5$



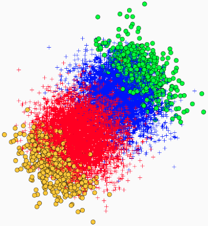
$\alpha = 0.7$



a) Test accuracy w.r.t. α



b) LSVM

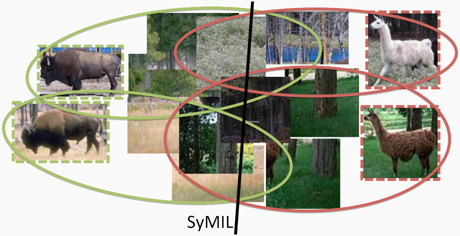


c) SyMIL

SyMIL: toy experiments on image data

- Classification performances (accuracy) on Mammal dataset

Method	bison vs llama	llama vs bison
LSVM	90.3	87.7
SyMIL	95.7	95.7



SyMIL: toy experiments on text data

- Text dataset from Reuters21578
  - positive class: *money*
  - negative class: *ship, crude*

	LSVM	SyMIL
a) Predictive accuracy		
	96.3%	97.6%
b) Similarity between instances and category		
	Bag $\oplus = 74\%$	Bag $\oplus = 73\%$
	Bag $\ominus = 67\%$	Bag $\ominus = 78\%$
c) Examples		
Bag $\oplus$	bank, currency, money, exchange, treasury	bank, exchange, rate, currency, monetary
Bag $\ominus$	west, finance, bank, british, money	oil, opec, shipping, port, union

SyMIL: standard MIL dataset results i

Dataset	Image	Musk1	Musk 2	Text
pos/neg bags	100/100	47/45	39/63	200/200
instances/bag	~ 6.5	5.17	64.69	~ 8
feature dimension	230	166	166	$\sim 66\,500$

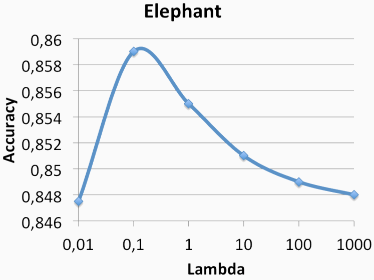
Method		Image	Musk	Text
mi-SVM		73.4	84.5	81.6
MI-SVM		75.5	81.7	80.3
LSVM		74.4	82.7	80.0
SyMIL	linear	79.1	88.2	**84.8**
	RBF	**80.2**	**89.2**	-
Without constraints 1 & 2	linear	78.1	86.9	83.7
	RBF	78.7	87.5	-

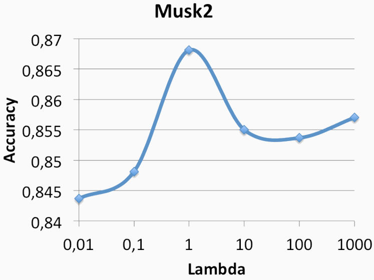
SyMIL: standard MIL dataset results ii

Method	Image	Musk	Text	Avg.
SyMIL	**80.2**	89.2	**84.8**	**84.7**
mi-SVM [1]	72.9	85.5	81.6	80.0
MI-SVM [1]	74.4	81.1	81.4	79.0
ALP-SVM [7]	77.9	86.3	-	-
MICA [16]	73.9	87.5	82.3	80.1
MIGraph [28]	76.1	**90.0**	-	-
MiGraph [28]	78.1	89.6	-	-
MI-CRF [5]	78.5	86.7	-	-
Convex relaxation [10]	75.8	-	-	-
GP-WDA [11]	79.0	88.4	83.2	83.5
eMIL [13]	77.0	85.3	82.7	81.7
MILEAGE [25]	77.7	-	-	-

SyMIL: hyper-parameter analysis

- Accuracy performance with respect to hyper-parameter λ (logarithmic scale)

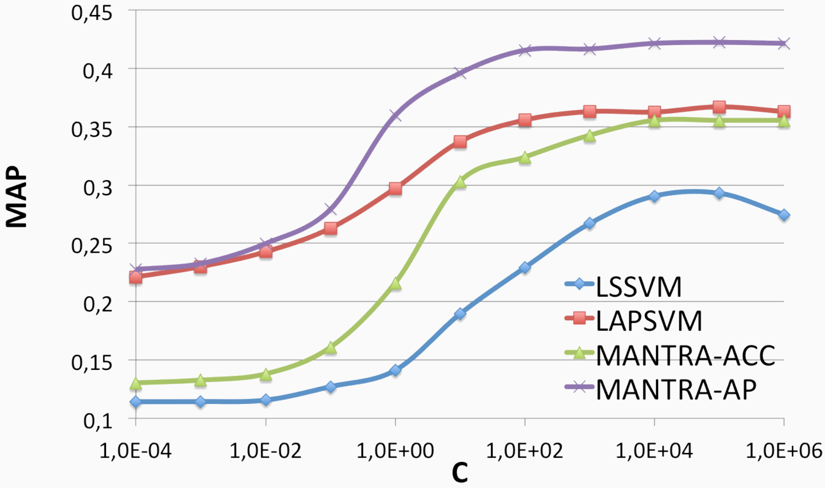




MANTRA: comparison to LSSVM

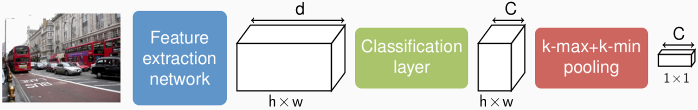
Method	UIUC	15 Scene	PPMI	MIT67
Multi-class accuracy (%)				
LSSVM	73.3 ± 0.3	65 ± 1.5	13.3	26.6
MANTRA	**93.2 ± 1.0**	**80.7 ± 0.7**	**51.0**	**56.4**
LSSVM-N	71.6 ± 1.3	64.3 ± 0.9	13.6	25.2
MANTRA-C	**93.2 ± 0.9**	80.4 ± 0.6	50.9	**56.5**
Average training time (seconds)				
LSSVM	1863	14179	21327	156360
MANTRA	**61**	**843**	**2593**	**41805**

MANTRA: impact of hyper-parameter C



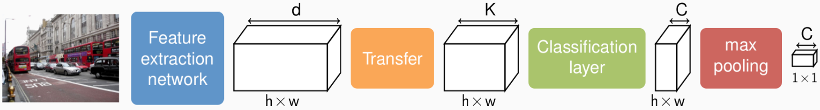
Region-based strategy

- WELDON (ProNet [Sun, CVPR16])



[1] Thibaut Durand, Nicolas Thome, and Matthieu Cord
WELDON: Weakly Supervised Learning of Deep ConvNets.
In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

- Deep MIL



[1] Maxime Oquab, Léon Bottou, Ivan Laptev and Josef Sivic
Is object localization for free? – Weakly-supervised learning with CNNs.
In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.