Tabula Rasa: Model Transfer for Object Category Detection

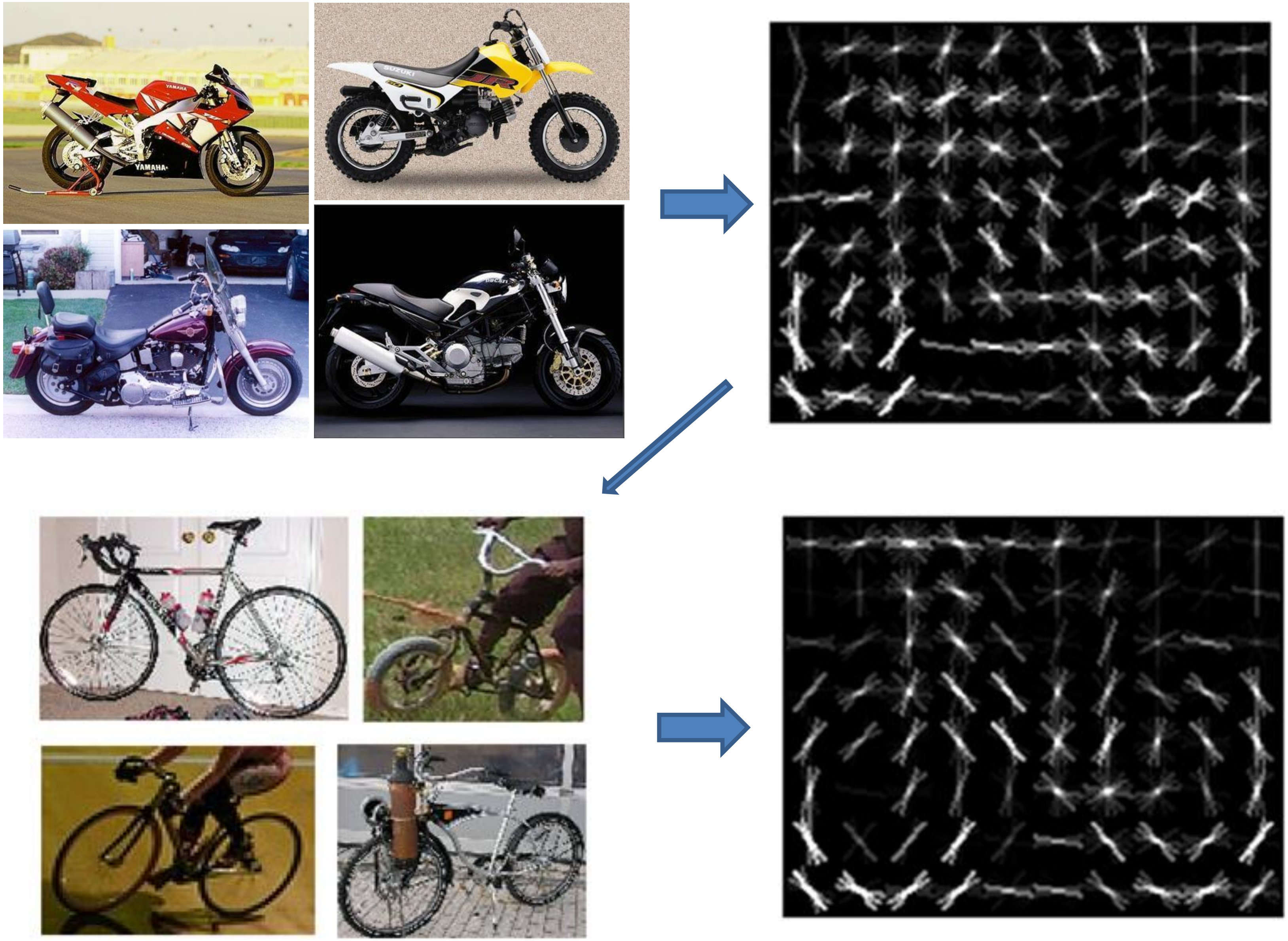
Yusuf Aytar & Andrew Zisserman,
Department of Engineering Science
Oxford

(Presented by Elad Liebman)

General Intuition I

- We have: a discriminatively trained classification model for category A.
- We need: a classifier for a new category B.
- Can we use it to make learning a model for category B easier?
  - Less examples?
  - Better accuracy?

General Intuition II



Tabula Rasa: Model Transfer for Object Category Detection, Aytar & Zisserman
Motorbike images courtesy of the Caltech Vision Group, collated by Svetlana Lazebnik

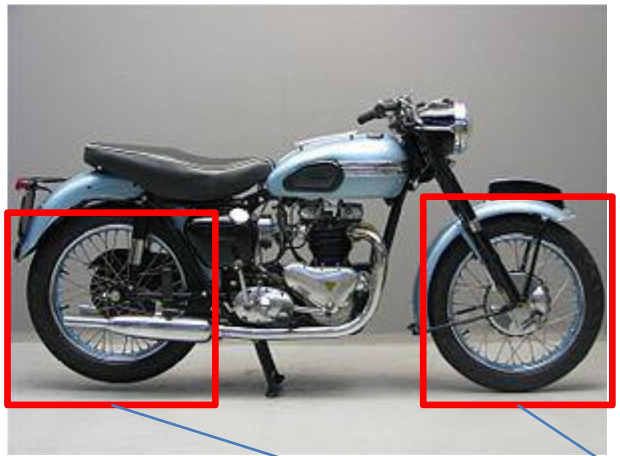
Background I

- Good:
  - There has been considerable progress recently in object category detection.
  - Successful tools are readily available.
- Bad:
  - current methods require training the detector from scratch.
  - Training from scratch is very costly in terms of sample size required.
  - Not scalable in multi-category settings.

Background II

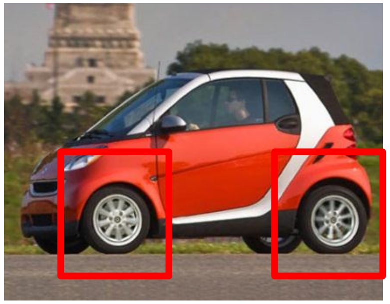
- Possible solution:
  - Represent categories by their attributes, and re-use attributes.
  - Attributes are learned from multiple classes, so training data is abundant.
  - Attributes learned can be used even for categories that didn't "participate" in the learning, as long as they share the attribute.

Background III



Wheel Detector

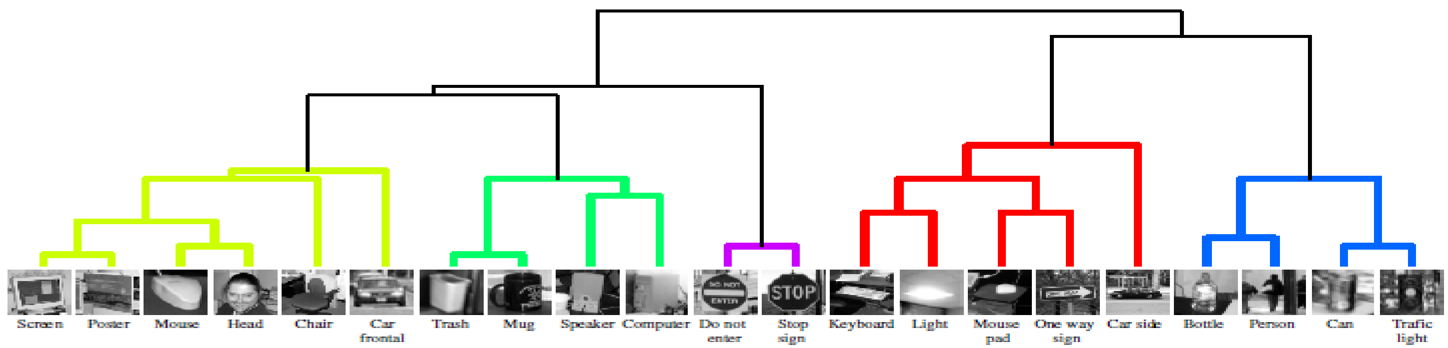
Use for detection of objects with "wheel" attributes

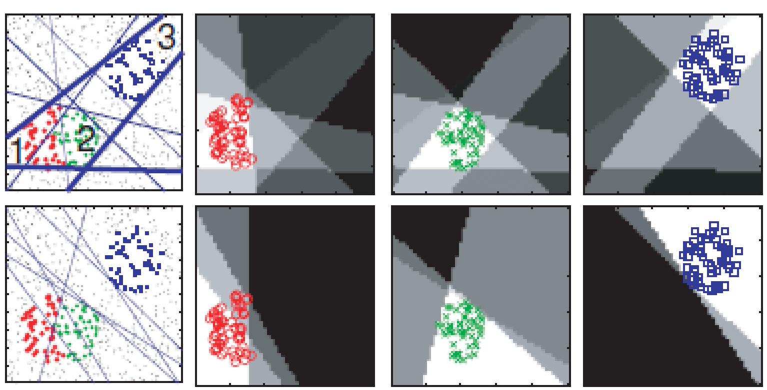


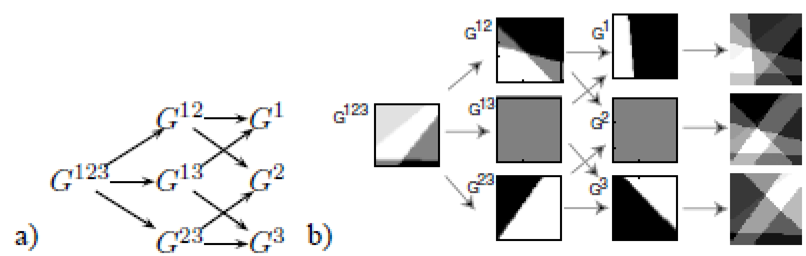
(This idea should sound familiar…)

“**Sharing visual features for multiclass and multiview object detection**”, Torralba et al., 2007

- Training multiple category classifiers at the same time with lower sample and runtime complexity using shared features.
- Uses a variation on boosting and shared regression stumps.

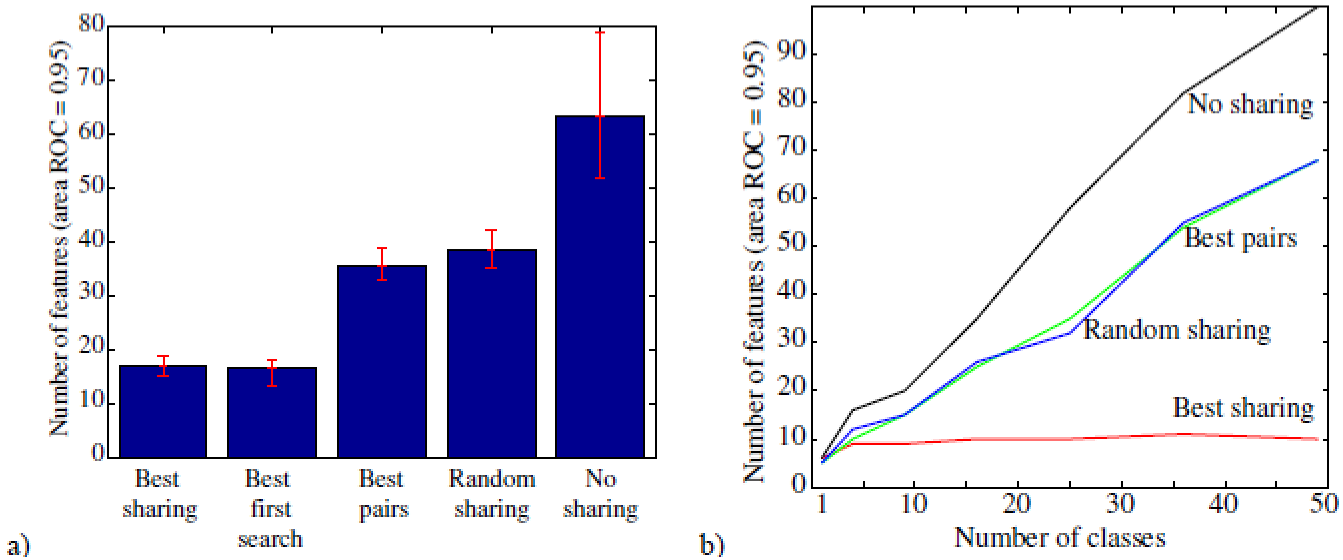




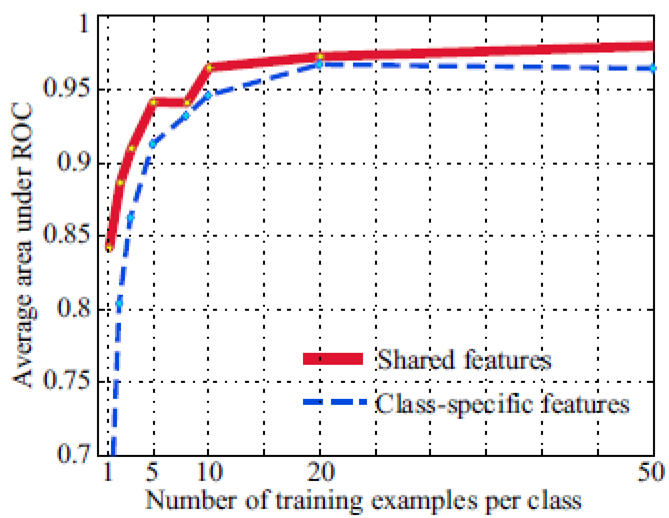


Torralba et al. – cont. I

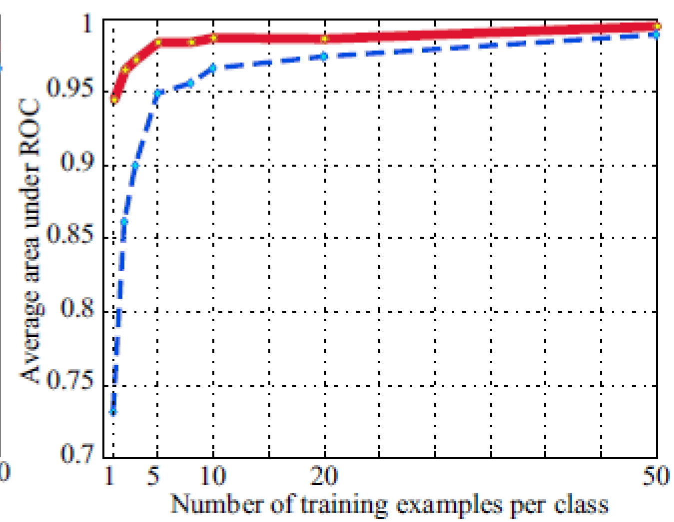
Number of required features



Effect on learning



12 different categories

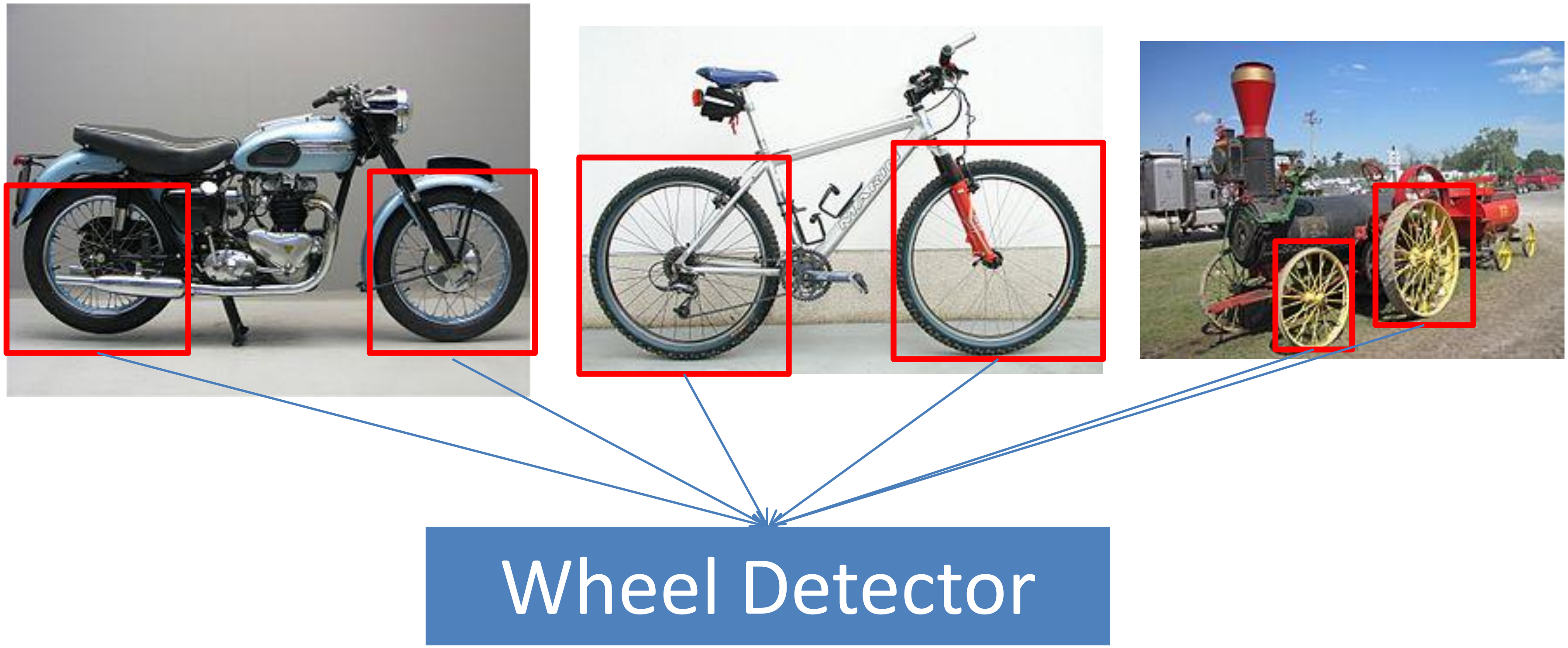


12 views of same category

Torralba et al. – cont. II

- There is a difference in motivations here.
- Torralba et al. are mostly concerned with scalability.
  - Reduce the cost of training multiple detectors.
  - Use shared features when learning full sets of distinctive features per category is infeasible.
- Knowledge transfer is more concerned with sample complexity.
  - Use preexisting related classifiers when new examples are hard to come by.

(*Back to our paper…*)



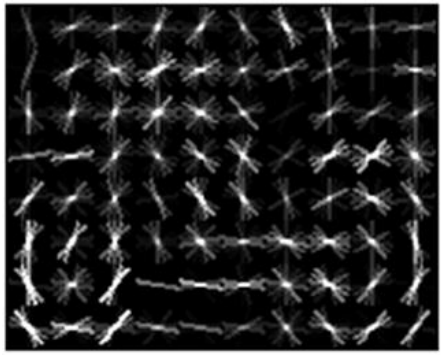
- Unfortunately, this approach proves inferior in practice to discriminative training (true for both detection and classification). (true to when the paper was published…)

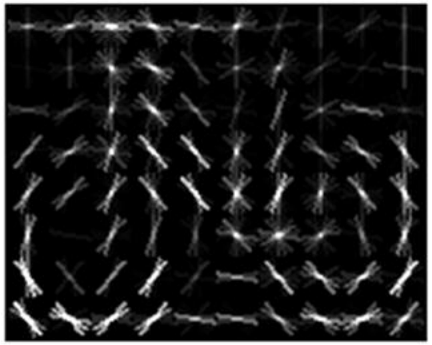
Background IV

- An alternative approach:
  - Benefit from previously-learned category detectors.
  - Previously learned categories should be similar.
- We need a way to transfer information from one classifier to the next.

Aytar & Zisserman I

- Consider the SVM discriminative training framework for HOG template models of Dalal & Triggs & Felzenszwalb et al.
- <u>Observation</u>: learned template records the spatial layout of positive and negative orientations.
- Classes that are geometrically similar will give rise to similar templates.





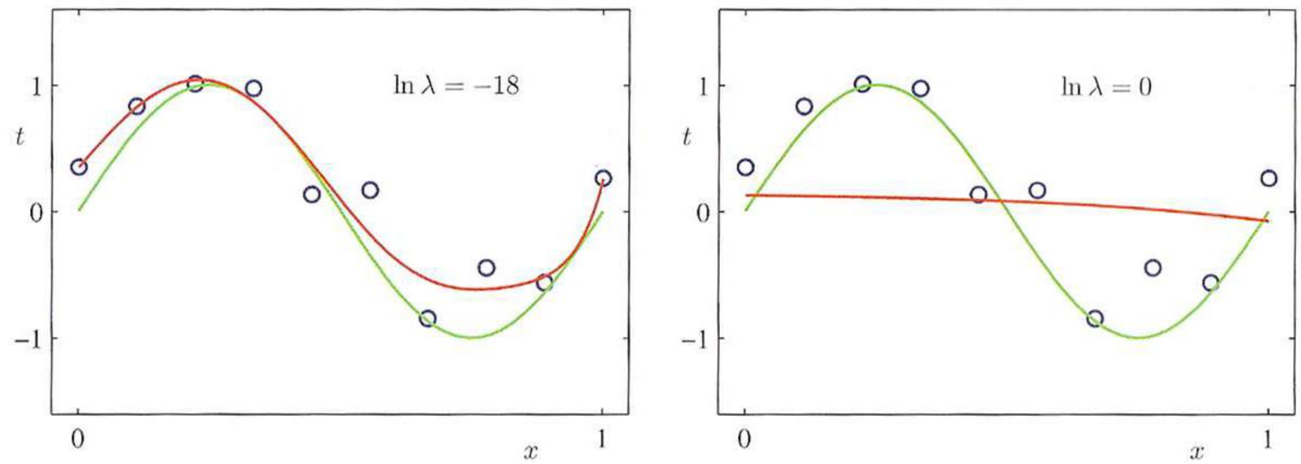
Aytar & Zisserman II

- Apply transfer learning from one detector to another.
- To do this, the previously learned template is used as a regularizer in the cost function of the new classifier.
- This enables learning with a reduced number of examples.

Some (*a few*) Words on Regularization

- From a Bayesian standpoint, it's similar to introducing a prior.
- Often used to prevent overfitting or solve ill posed problems.
- A good example for regularization: ridge regression

$$argmin_{\beta}\{\|Y - X\beta\|^2 + \|\Gamma\beta\|^2\}$$



Images taken from Andrew Rosenberg's slides, ML course, CUNY

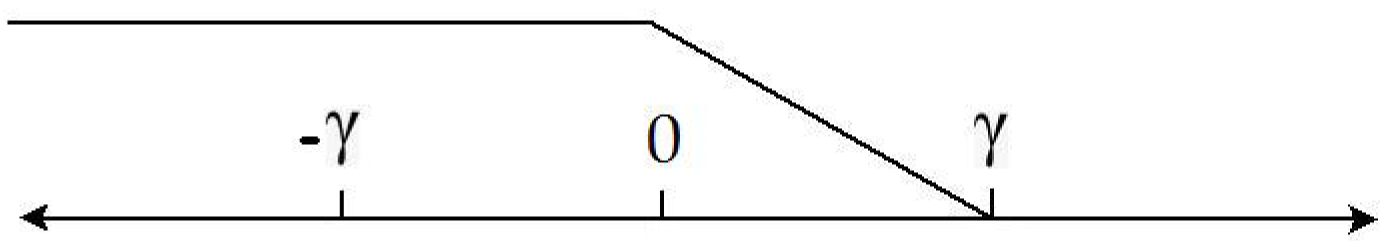
Model Transfer Support Vector Machines

- We wish to detect a target object category.
- We already have a well trained detector for a different source category.
- Three strategies to transfer knowledge from the source detector to the target detector:
  - Adaptive SVMs
  - Projective Model Transfer SVMs
  - Deformable Adaptive SVMs

Adaptive SVMs I

- Learn from the source model w^s by regularizing the distance between the learned model w and w^s.
- x_i are the training examples, $y_i \in \{-1,1\}$ are the labels, and the loss function is the hinge loss:

$$l(x_i, y_i; w, b) = \max(0, 1 - y_i(w^T x_i + b))$$



Adaptive SVMs II

- Reminder: in regular SVMs we want to optimize:

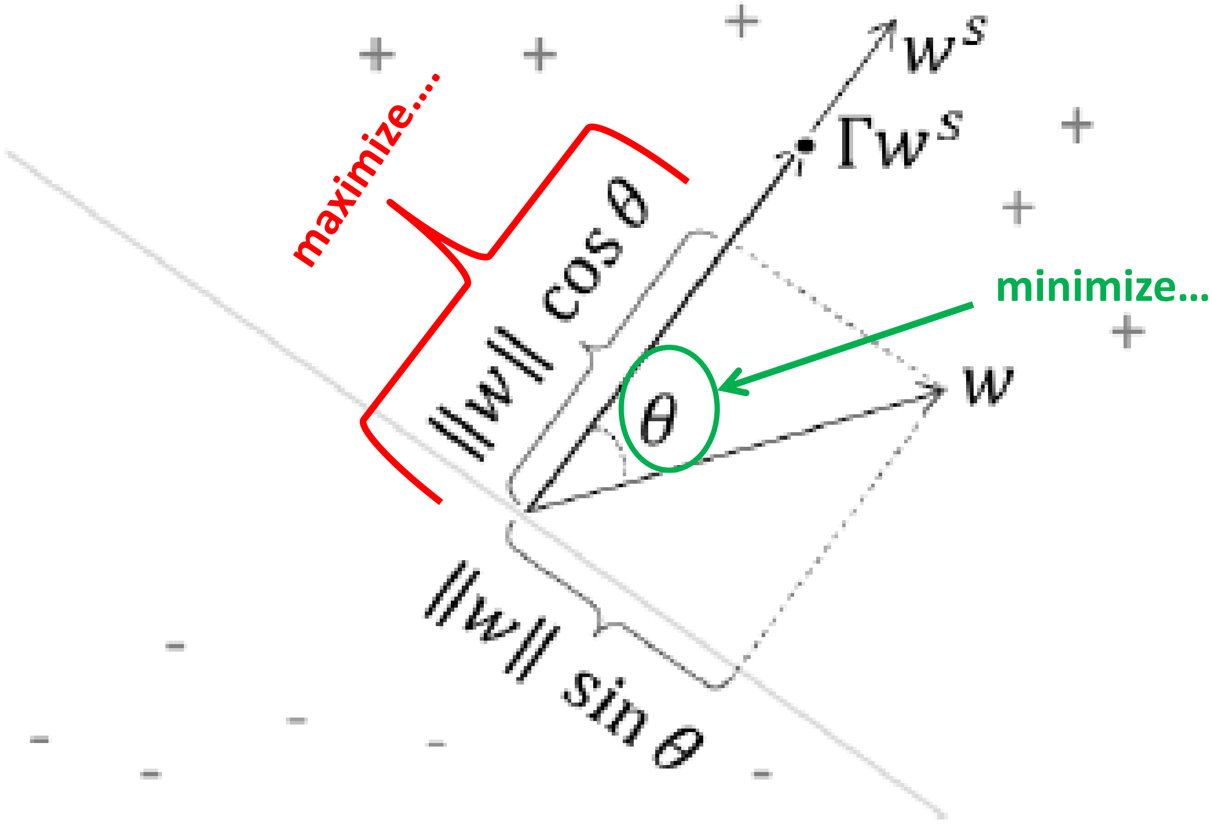
$$L_A = \min_{w,b}\{\,\|w\|^2 + C\sum_{i}^{N} l(x_i, y_i; w, b)\}$$

- But now, our goal is to optimize:

$$L_A = \min_{w,b}\{\,\|w - \Gamma w^s\|^2 + C\sum_i^N l(x_i, y_i; w, b)\}$$

- Γ controls the amount of transfer regularization, C controls the weight of the loss function and N is the number of samples.

An Illustration



Adaptive SVMs III

- We note that if w^s is normalized to 1 then:

$$\|w - \Gamma w^s\|^2 = \|w\|^2 - 2\Gamma\|w\|cos\theta + \Gamma^2$$

  - $\|w\|^2$ - “normal” SVM margin.
  - $(-2\Gamma\|w\|cos\theta)$ - the transfer.

- We wish to minimize θ, the angle between w^s and w.

- However, $-2\Gamma\|w\|cos\theta$ also encourages w to be larger, so Γ controls a tradeoff between margin maximization and knowledge transfer.

Projective Model Transfer SVMs I

- Rather than transfer by maximizing $\|w\| cos\theta$, we can instead minimize the projection of w onto the separating hyperplane orthogonal to w^s.
- This directly translates to optimizing:

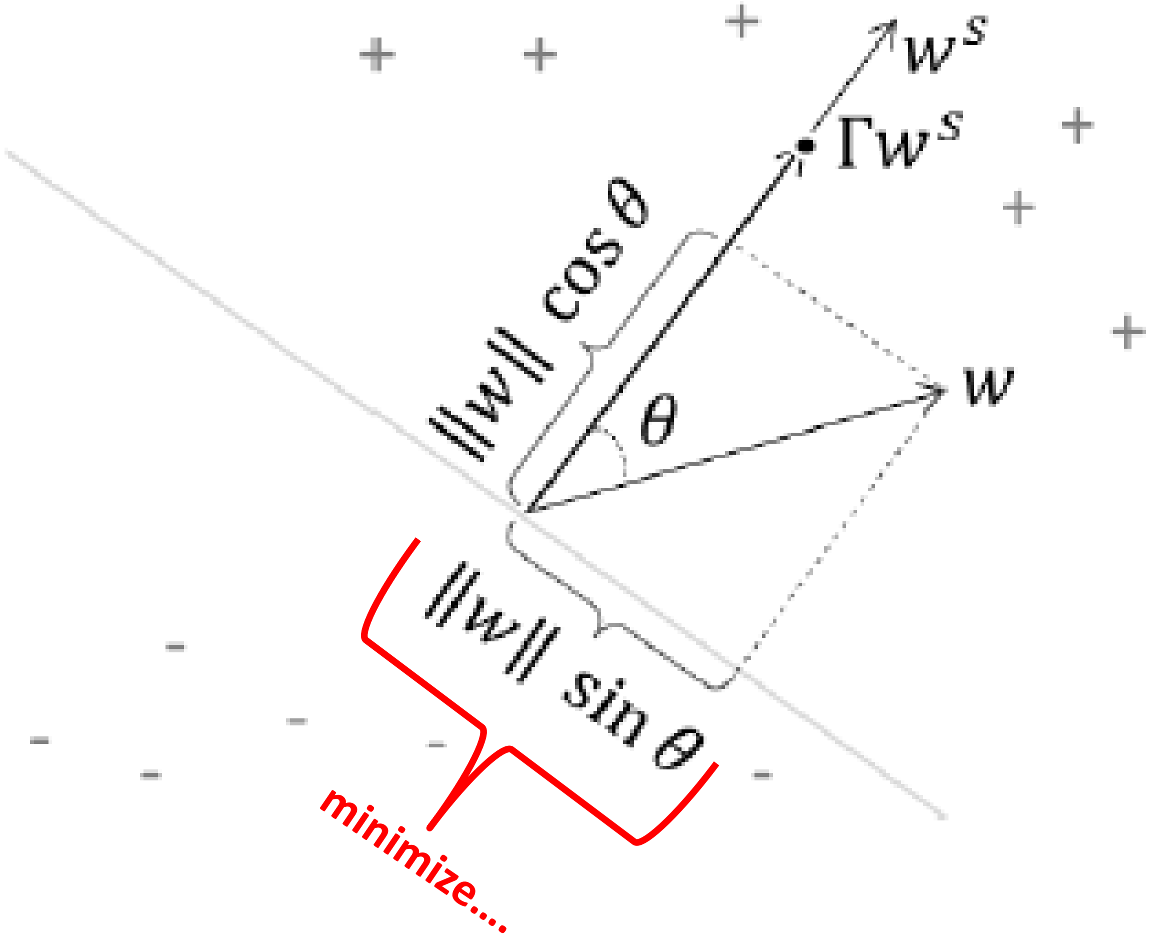
$$L_{PMT} = \min_{w,b} \|w\|^2 + \Gamma \|Pw\|^2 + C \sum_i^N l(\mathbf{x}_i, y_i; w, b)$$

$$st \quad : \quad w^\top w^s \geq 0$$

- Where P is the projection matrix:

$$P = I - \frac{w^s {w^s}^\top}{{w^s}^\top w^s}$$

Yet another illustration

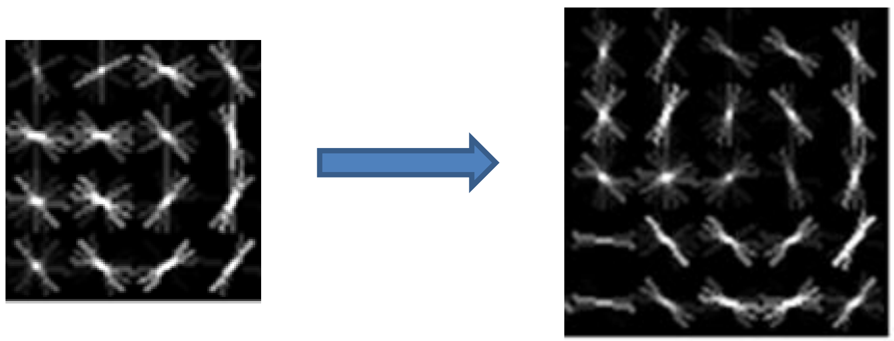


Projective Model Transfer SVMs II

- We note that $\|Pw\|^2$ is the squared norm of the projection of w onto the source hyperplane: $\|Pw\|^2 = \|w\|^2 sin^2\theta$
- $w^T w^s \geq 0$ constraints w to the positive halfspace defined by w^s.
- Here too Γ controls the transfer. As $\Gamma \to 0$, the PMT-SVM reduces to a classic SVM optimization problem.

Deformable Adaptive SVMs I

- Regularization shouldn't be "equally forced".
- Imagine we have a deformable source template – small local deformations are allowed to better fit the source to the target.
- For instance, when transferring from a motorbike wheel to a bicycle wheel:



- We need more flexible regularization…

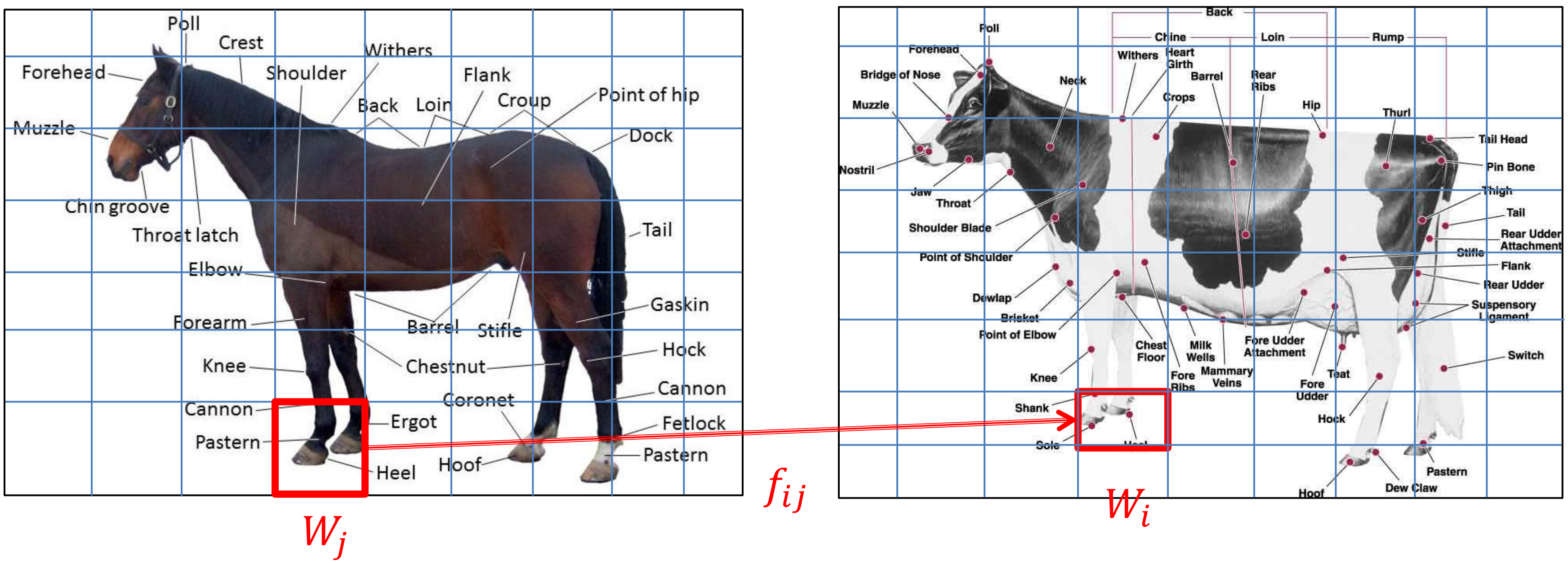
Deformable Adaptive SVMs II

- Local deformations are described as a flow of weight vectors from one cell to another, governed by the following flow definition:

$$\tau(w^s)_i = \sum_j^M f_{ij} w_j^s$$

- τ represents the flow transformation, w_j^s is the j^{th} cell in the source template, and f_{ij} denotes the amount of transfer from the j^{th} cell in the source to the i^{th} cell in the target.

Deformable Adaptive SVMs III



Deformable Adaptive SVMs IV

- Now, the Deformable-Adaptive-SVM is simply a generalization of the adaptive SVM we've seen before, with w^s replaced with its deformable version $\tau(w^s)$:

$$
\begin{aligned}
L_{DA} \quad = \quad & \min_{f,w,b} \|w - \Gamma\tau(w^s)\|^2 + C\sum_{i}^{N} l(\mathbf{x}_i, y_i; w, b) \\
& +\lambda \left(\sum_{i \neq j}^{M,M} f_{ij}^2 d_{ij} + \sum_{i}^{M} (1 - f_{ii})^2 d \right)
\end{aligned}
$$

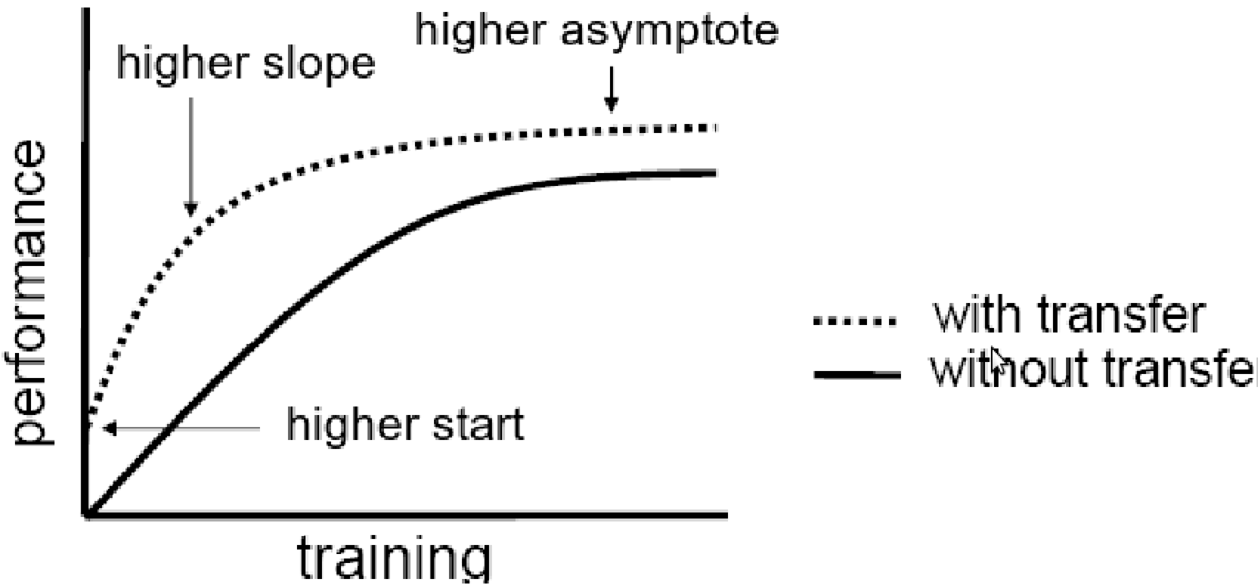
(λ is the weight of the deformation, d_{ij} is the distance between cells i, j and d is the penalty for overflow)

Deformable Adaptive SVMs V

- λ in effect controls the extent of deformability.
- High λ values make the model more rigid (you pay more for the deformations you make), pushing the solution closer to that of the simple adaptive SVM.
- Low λ values allow for a more flexible source template with less regularization.
- (Amazingly enough, the term $\|w \; - \Gamma\tau(w^{s}\|^2$ is still convex.)

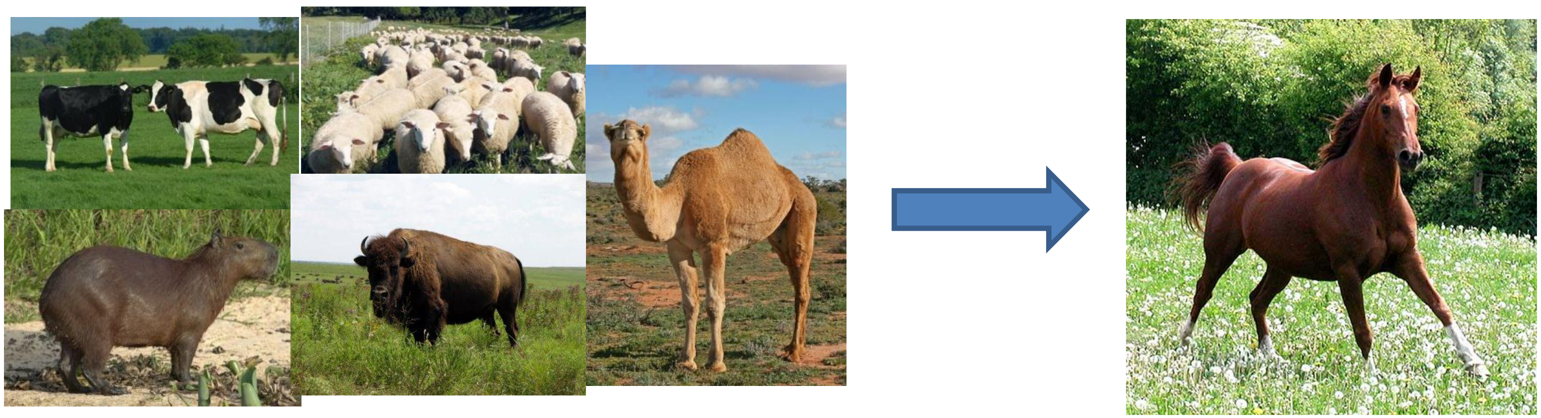
Experiments I.I

- In general, transfer learning can offer three major benefits:
  - Higher starting point
  - Higher slope (*we learn faster*)
  - Higher asymptote (*learning converges into a better classifier*)

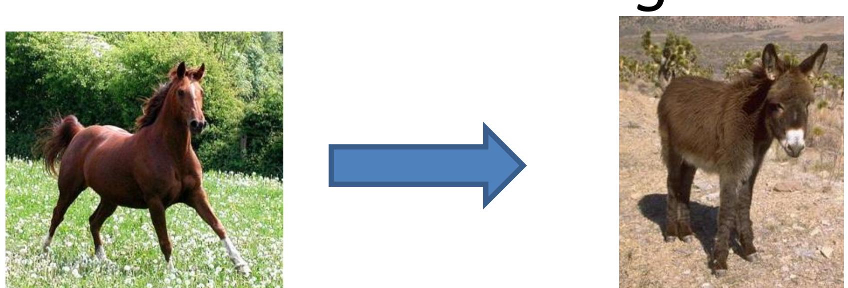


Experiments I.II

- Two types of transfer experiments:
  - Specialization (*we know how to recognize quadrupeds, now we want to recognize horses*)



  - Interclass transfer (*we know how to recognize horses, now we want to recognize donkeys*)



Experiments II – Interclass

- Baseline detectors are the SVM classifiers trained directly without any transfer learning.
- Two scenarios studied:
  - transferring from motorbikes to bicycles
  - transferring from cows to horses
- Two variants discussed:
  - One shot learning – we can only choose one (!) example from the target class, and study our starting point.
  - Multiple shot learning

Experiments III – One Shot Learning

Top 15

(middle)

Low 15

Ranks	Base. SVM	A-SVM	DA-SVM	PMT-SVM
01-15	40.5 ± 07.2	**53.9 ± 04.2**	53.7 ± 04.3	53.5 ± 05.7
16-30	33.0 ± 13.5	52.5 ± 08.3	51.9 ± 08.8	**54.7 ± 05.7**
31-45	26.4 ± 13.3	47.1 ± 07.3	47.1 ± 07.6	**48.5 ± 08.7**
46-60	14.0 ± 09.3	42.4 ± 03.7	**42.5 ± 04.2**	27.8 ± 11.3

(Looks good, but a bit unfair, especially when using lower-grade examples from the target category…)

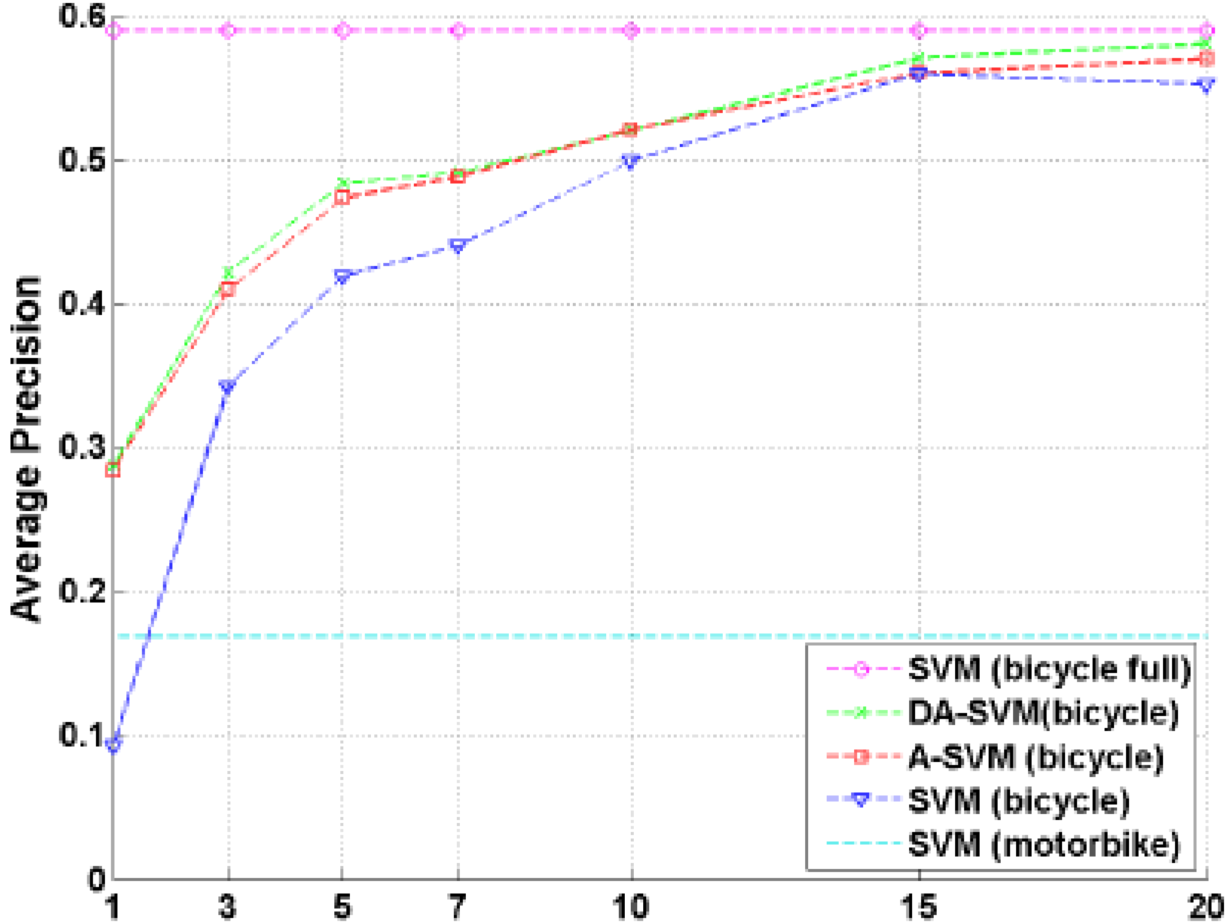
Experiments IV – Multiple Shot

Number of Samples		1	3	5
Test-procedure: pascal-side-only	**Base. SVM**	09.3 ± 08.8	34.2 ± 11.5	41.9 ± 05.9
	A-SVM	28.4 ± 08.1	40.9 ± 06.1	47.3 ± 04.4
	DA-SVM	$\mathbf{28.7 \pm 08.2}$	$\mathbf{42.1 \pm 05.7}$	$\mathbf{48.3 \pm 03.6}$
Test-procedure: pascal-default	**Base. SVM**	07.0 ± 04.4	18.6 ± 05.2	22.7 ± 02.1
	A-SVM	14.9 ± 02.5	20.1 ± 02.7	24.0 ± 01.7
	DA-SVM	$\mathbf{15.3 \pm 02.5}$	$\mathbf{20.6 \pm 02.4}$	$\mathbf{24.5 \pm 01.6}$

Number of Samples		7	10	15	20	30	50
Test-procedure: pascal-side-only	**Base. SVM**	44.0 ± 09.9	49.9 ± 05.4	55.9 ± 06.8	55.2 ± 03.5	57.9 ± 02.0	58.9 ± 01.3
	A-SVM	48.8 ± 08.4	$\mathbf{52.0 \pm 05.9}$	56.0 ± 03.8	57.0 ± 03.3	59.0 ± 01.6	$\mathbf{60.2 \pm 01.5}$
	DA-SVM	$\mathbf{49.1 \pm 07.6}$	$\mathbf{52.0 \pm 05.2}$	$\mathbf{57.0 \pm 04.7}$	$\mathbf{58.0 \pm 01.9}$	$\mathbf{60.3 \pm 02.0}$	59.5 ± 00.9
Test-procedure: pascal-default	**Base. SVM**	24.7 ± 04.5	27.1 ± 02.3	29.6 ± 01.9	30.1 ± 01.2	30.7 ± 01.4	31.6 ± 00.9
	A-SVM	25.2 ± 03.1	27.0 ± 02.0	29.9 ± 01.2	31.0 ± 00.9	31.5 ± 01.3	$\mathbf{32.3 \pm 00.5}$
	DA-SVM	$\mathbf{25.5 \pm 03.4}$	$\mathbf{27.3 \pm 01.7}$	$\mathbf{30.2 \pm 01.0}$	$\mathbf{31.1 \pm 00.8}$	$\mathbf{31.5 \pm 01.3}$	32.2 ± 00.7

(We note that by ~10 examples, basic SVM has caught up with us…)

Experiments V – Multiple Shot



Experiments VI - Specialization

- "Quadruped" detector trained with instances of cows, sheep and horses.
- Then specialization for cows and horses was attempted via transfer.

Number of Samples		1	3	5
Test-procedure: pascal-side-only	**Base. SVM**	03.6 ± 03.8	14.3 ± 07.6	20.0 ± 09.0
	A-SVM	**21.2 ± 05.5**	**29.7 ± 06.0**	30.9 ± 04.3
	DA-SVM	20.9 ± 05.6	29.2 ± 06.0	**31.5 ± 03.9**
Test-procedure: pascal-default	**Base. SVM**	03.6 ± 03.6	10.3 ± 02.6	10.6 ± 01.8
	A-SVM	**11.5 ± 04.0**	**14.5 ± 03.2**	**13.8 ± 03.3**
	DA-SVM	11.3 ± 04.5	14.2 ± 03.4	13.6 ± 03.0

Number of Samples		7	10	15	20	30	50
Test-procedure: pascal-side-only	**Base. SVM**	25.0 ± 07.3	29.9 ± 04.3	35.9 ± 05.7	40.1 ± 02.8	**45.8 ± 02.6**	**47.1 ± 02.3**
	A-SVM	**32.6 ± 04.7**	35.3 ± 03.0	**37.8 ± 05.6**	**40.4 ± 03.3**	43.6 ± 03.5	45.4 ± 01.3
	DA-SVM	32.1 ± 04.4	**36.6 ± 02.8**	37.2 ± 04.7	40.3 ± 02.9	42.9 ± 03.1	44.0 ± 01.0
Test-procedure: pascal-default	**Base. SVM**	12.7 ± 02.0	13.8 ± 03.3	14.6 ± 02.4	16.6 ± 01.1	**19.9 ± 00.9**	**21.1 ± 01.5**
	A-SVM	15.2 ± 03.4	16.0 ± 01.8	16.0 ± 02.8	**17.6 ± 01.0**	**19.9 ± 01.4**	20.6 ± 00.8
	DA-SVM	**15.3 ± 03.0**	**16.2 ± 01.7**	**16.1 ± 02.7**	**17.6 ± 01.0**	19.8 ± 01.9	20.8 ± 00.4

(Once again we note that by ~15-20 examples, basic SVM has caught up with us…)

Discussion

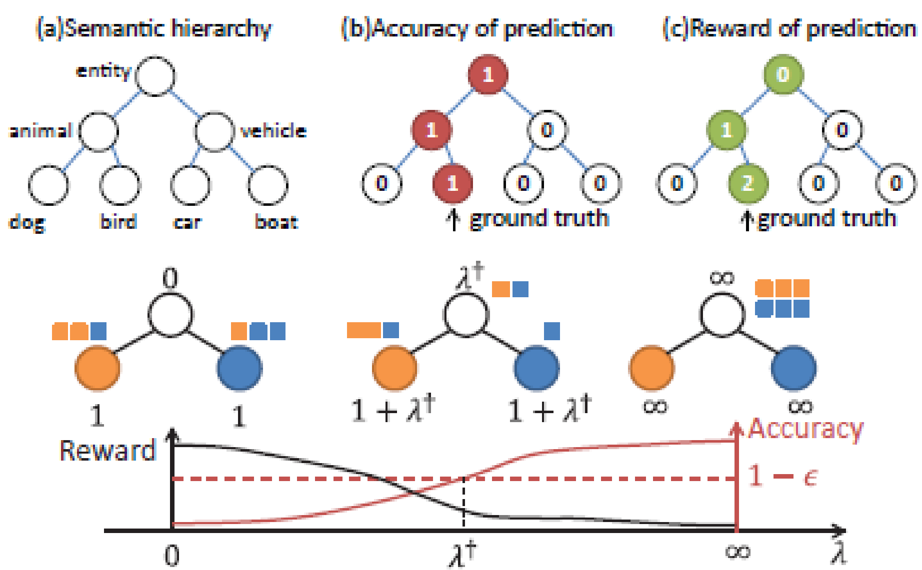
- Pros:
  - An interesting and fairly straightforward expansion of the basic category detection scheme.
  - Provides a far better starting point for classifying new categories.
  - A different perspective on multi-category settings.
- Cons:
  - “Closeness” between classes is very poorly defined.
  - One-shot experiments not particularly convincing.
  - Advantage degrades the more samples you have.
  - PMT-SVM doesn’t scale very well…

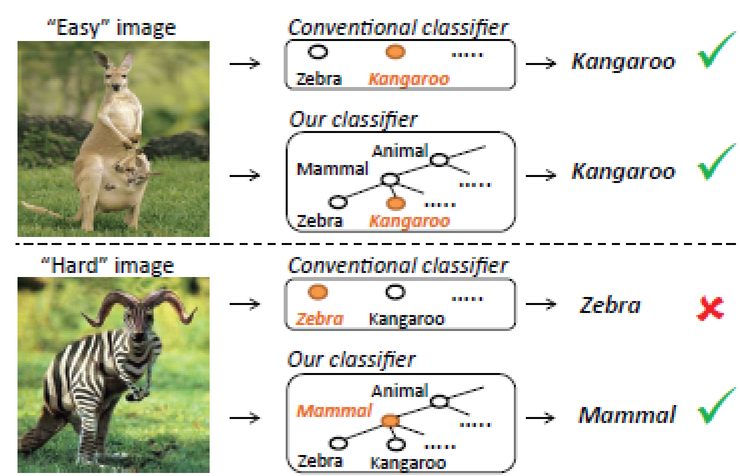
Something Related (But Different)

("If you liked Aytar & Zisserman, you might also enjoy this paper")

"**Hedging Your Bets: Optimizing Accuracy Specificity Trade-Offs in Large Scale Visual Recognition**", Deng et al., 2012

- Object categories form a semantic hierarchy.
- Make more reliable predictions about less specific classification when faced with uncertainty.

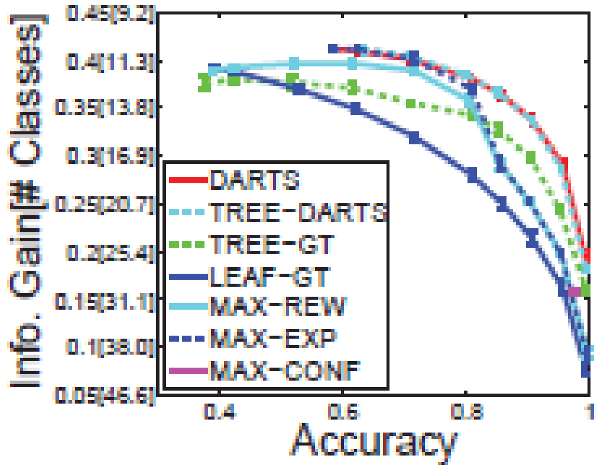


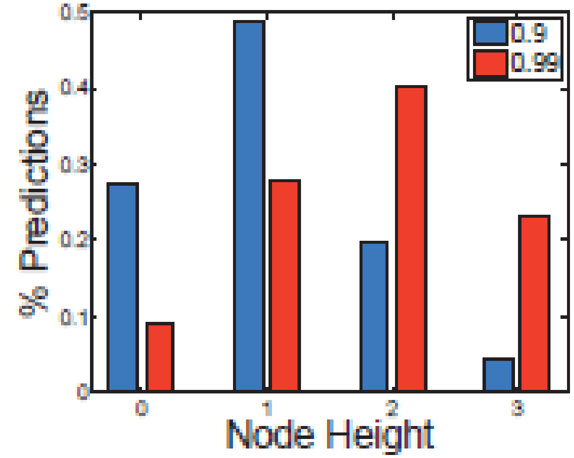


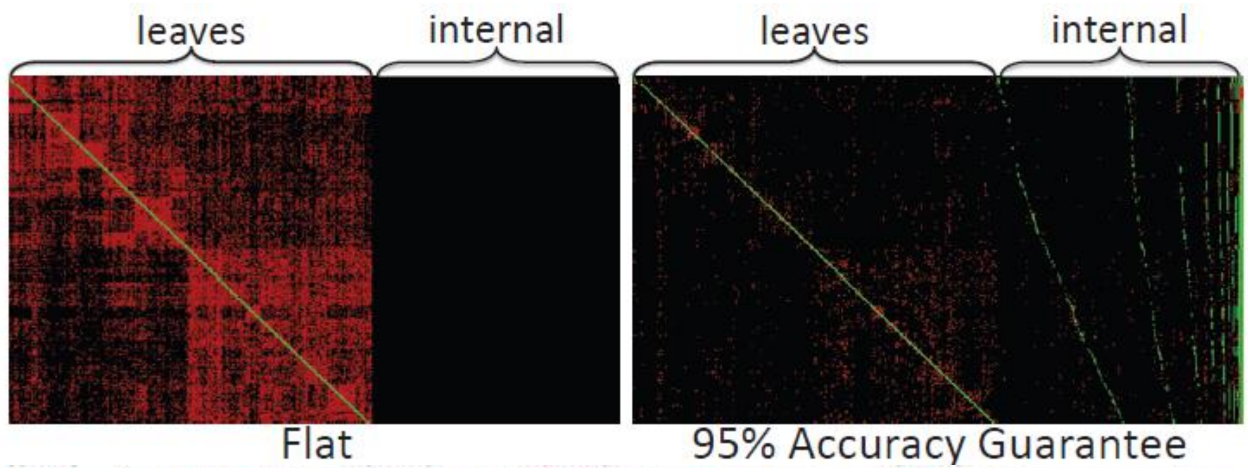
Deng et al. – cont. I

- Given a hierarchy graph, a label is correct either if it’s the right leaf, or any of its ancestors.
- In this setting, maximizing accuracy alone cannot work.
- Instead – maximize information gain while maintaining an error rate $\geq$ a required threshold.
- Done via a generalization of the Lagrange multipliers method, with regular SVM one-vs-all classifiers for posterior probabilities on the leaves.

Deng et al. – cont. II







(*Main References*)

- Tabula Rasa: Model Transfer for Object Category Recognition. Aytar & Zisserman, IEEE International Conference on Computer Vision, 2011.
- Histograms of Oriented Gradients for Human Detection. Dalal & Triggs, International Conference on Computer Vision & Pattern Recognition - June 2005.
- Regularized Adaptation: Theory, Algorithms and Applications. Xiao Li, PhD Dissertation, U. Washington, 2007.