Cognitive Vision for Cognitive Systems

Barbara Caputo, Marco Fornoni
Idiap Research Institute
http://www.idiap.ch/~bcaputo
http://www.idiap.ch/~mfornoni
bcaputo@idiap.ch
mfornoni@idiap.ch
Exam dates:

- Feb 01 2013 at 10:00, room
- **Hard Deadline** for report submission: Jan 28 2013, 23:59 CET
Life-Long Learning
Learning of a Concept Incrementally

ImageNet is an image database organized according to the WordNet hierarchy (currently only the nouns) in which each node of the hierarchy is depicted by hundreds and thousands of images. Currently we have an average of over five hundred images per node. We hope ImageNet will become a useful resource for researchers, educators, students and all of you who share our passion for pictures.

Click here to learn more about ImageNet, Click here to join the ImageNet mailing list.
Learning of a Concept Incrementally

Online Learning

Online algorithms observe examples in sequence of rounds, and construct the classification function incrementally.

It's Valeria!
Learning of a Concept Incrementally

Online Learning

Given an input, the classifier predicts an output.

I don’t know this baby...
Learning of a Concept Incrementally

Online Learning

...then the teacher reveals the true label

She is Valeria

I don't know this baby...
Learning of a Concept Incrementally

Online Learning

The classifier compares its prediction with the true label and update its knowledge.

She is Valeria

Ok! I will update my knowledge.
Continuous Learning of a Concept
Continuous Learning of a Concept

- illumination
- furniture
- objects
- people
- decoration
Learning of a New Concept

Firetrack is a vehicle, but not a known one!
Learning of a New Concept

Is it easier to learn from scratch a Firetruck or an Airplane?
Learning of a New Concept
Transfer Learning: The Vision Perspective
Example - Learning

**Task**
“I want to learn Italian”

**Training Experience**

**A performance measure**

“Bionji”… “Buonyo”

“Buongiorno”

An agent **learns** if its performance at a task improves with experience (Mitchell, 1996).
Example – Learning to Learn

Tasks “I want to learn Italian and French”

Training Experience

Performance measures

It: “Buongiorno”
Fr: “Bonjour”

An agent learns to learn if its performance at each tasks improves with experience and with the number of tasks (Thrun, 1996). 

2
What is this?

Okapi

**Goal:** correct classification when we see another instance.
Does it look like some other animal?

Okapi

**Goal:** correct classification when we see another instance.
Does it look like some other animal?

**Analogy making** is a cognitive process in which two conceptualizations are analyzed for common structural patterns (Gentner, 1989).
Knowledge Transfer

Storing knowledge gained while solving one problem and applying it to a different but related problem (West et al., 2007).

Source/Sources

many samples

Target: Okapi

few samples

Transfer across multiple tasks with different label sets.
Domain adaptation is needed when the data distribution in the test domain is different from that in the training domain (Daumé and Marcu, 2006).

**Source/Sources**

many samples

**Target**

few samples

Transfer across tasks with the same label set.
Learning to Learn

Sharing Information
  • Knowledge Transfer
  • Domain Adaptation
  • Multi-Task Learning

Dynamic Process
  • Online Learning: continuous update of the current knowledge.
Knowledge Transfer: Advantages

Effect: decrease the amount of new annotation effort required to achieve good performance on a target task.

(Torrey and Shavlik, 2009)
Knowledge Transfer: Challenges

What to Transfer? Specify the form of the knowledge to transfer: instances, features, models.

How to Transfer? Define a learning algorithm able to exploit prior knowledge.

When to Transfer? Evaluate the task relatedness, keep useful knowledge and reject bad information (avoid negative transfer).
L- Fei Fei, R. Fergus, P. Perona. One-Shot Learning of Object Categories. PAMI 2006

- First paper to attack the problem of one shot learning in the computer vision community
- Knowledge transferred in the form of prior probability density function in the space of model parameters
• **Key intuition:** once the model has learned $N$ object categories, it is possible to extract some general knowledge about their ‘objectness’, and use this knowledge to boost the learning process even from few, only one training image.

$$R = \frac{p(O_{fg}|I,I_t)}{p(O_{bg}|I,I_t)} = \frac{p(I|I_t,O_{fg})}{p(I|I_t,O_{bg})} \frac{p(O_{fg})}{p(O_{bg})}.$$ 

with $R$ ratio of the class posteriors, $I$ query image, $O_{fg}$ foreground category, $O_{bg}$ background category and $I_t$ set of training images.
Introducing a parametric model:

\[
R \propto \frac{\int p(\mathcal{I}|\theta, \mathcal{O}_{fg})p(\theta|\mathcal{I}_t, \mathcal{O}_{fg}) \, d\theta}{\int p(\mathcal{I}|\theta_{bg}, \mathcal{O}_{bg})p(\theta_{bg}|\mathcal{I}_t, \mathcal{O}_{bg}) \, d\theta_{bg}}
\]

the prior over the parameters is computed on the previously learned models, therefore encoding in the model of the new class some sort of prior for the search of the new parameters which is related to the ‘objectness’ of the known categories.
Concretely: constellation model
Results on Caltech four database (faces)

Category detection problem, 50 foreground/background training images, test 1-6 images
C. Lampert, H. Nickish, S. Harmeling. *Learning to detect unseen object classes by between-class attribute transfer.*

Proc CVPR09

Main contribution: transfer learning across different modalities
The Dilemma of Learning From Examples

visual object categories for which we have training images (hundreds)

dog bus car

elbow alpaca
vanguard grig
ramp axolotl

all visual object categories (tens of thousands)
Object Classification

Which of these images shows an alpaca?
Object Classification

Training images:

blobfish  alpaca  pink fairy armadillo  alpaca

Which of these images shows an alpaca?

We can classify objects if we have training examples
Object Classification

Training images:

- cat
- mole
- dog
- whale

Which of these images shows an alpaca?

We can classify objects if we have training examples of the same classes.
Transfer Learning

classes with example images
- cat
- mole
- dog
- whale

classes without example images
- elbow
- ramp
- grig
Attribute Based Classification

Which of these images shows an axolotl?
Attribute Based Classification

Description: **Axolotls**
- live in **water**,
- are **white**,
- have no **long fur**.

Which of these images shows an **axolotl**?

We can classify objects based on a **description**.
Attribute Based Classification

Description: Axolotls

- have property X,
- have property Y,
- don’t have property Z.

Which of these images shows an axolotl?

To classify objects based on a description, if we need to understand the description.
Attribute Based Classification

- **Doves** have property Y, but not X or Z.
- **Persian cats** have property Y and Z, but not X.
- **Polar bears** have property X, Y and Z.
- **Walruses** have property X, but not Y or Z.
Attribute Based Classification

- dove
- persian cat
- polar bear
- walrus

Descriptions:
- **Doves** have property Y, but not X or Z.
- **Persian cats** have property Y and Z, but not X.
- **Polar bears** have property X, Y and Z.
- **Walruses** have property X, but not Y or Z.

We can learn the properties from classes with examples. We can then detect the properties in images of any class.
Object Classification

Description: **Axolotls**
- properties $X$, $Y$, $\neg Z$

Which of these images shows an *axolotl*?
Object Classification

Description: Axolotls

- properties X, Y, ¬Z

Which of these images shows an axolotl?
Object Classification

Description: **Axolotl**
- properties $X, Y, \neg Z$

Which of these images shows an **axolotl**?

Using the descriptions, we were able to transfer **informating** between object categories.
Flat Multiclass Classification

- images $x$,
- class labels $y_1, \ldots, y_K \in \mathcal{Y}$ (at training time)
- class labels $z_1, \ldots, z_L \in \mathcal{Z}$ (at test time)
Classification with Attribute Layer

- images $x$,
- class labels $y_1, \ldots, y_K \in \mathcal{Y}$ (at training time)
- class labels $z_1, \ldots, z_L \in \mathcal{Z}$ (at test time)
- attributes $a_1, \ldots, a_M \in \{0, 1\}^M$ (encode description)
Can this actually work?

Example: *Animals with Attributes* dataset
- training: 40 animal categories with training images
- evaluation: predict 10 disjoint animals categories
- each class has a description by 85 binary attributes
Results: *Animals with Attributes* dataset

- training: 40 animal categories with training images
- evaluation: predict 10 disjoint animal categories
- each class has a description by 85 binary attributes
15 min break!
Main contribution: knowledge transferred in terms of hyperplanes parameters of an SVM (BMVC09, CVPR10) or confidence values (ICCV11), model learns from where and how much to transfer automatically.
When to transfer: contribution
When to transfer: contribution
When to transfer: contribution

Sources: many information

Target: few samples
How to transfer: contribution

**Previous work**
- Generative models (Fei-Fei et al., 2006, Raina et al., 2006).
- Binary classification (Fei-Fei et al., 2006, Dai et al., 2007).
- Disjoint source and target label sets (Rohrbach et al., 2010).

**Main contribution**
- Discriminative approaches.
- Multiclass classification.
- Overlapping source and target label sets.
What: Different solutions depending on how much supervision the target learner has on the sources.
What to transfer: choices

**What:** Different solutions depending on how much supervision the target learner has on the sources.

![Diagram](image)

Source Learner = Target Learner

**Full control:** the target learner knows all the details about how the source knowledge was formulated.
What to transfer: choices

**What:** Different solutions depending on how much supervision the target learner has on the sources.

No control: the target learner has no access neither to the source data nor to the source learning process.
Full Control
Target Problem

I want to learn ... vs

- Given a set of data \( D = \{x_i, y_i\}_{i=1}^N \)
- Find a function \( f : \mathcal{X} \rightarrow \mathcal{Y} \)

Minimize the structural risk

\[
\Omega(f) + C \sum_{i=1}^N \ell(f(x_i), y_i)
\]

- Linear models \( f(x) = \mathbf{w} \cdot \phi(x) \)
- Feature mapping \( \phi(x) \) with \( K(x_j, x_i) = \phi(x_j) \cdot \phi(x_i) \)

Optimization problem

\[
\min_{\mathbf{w}} \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^N \ell(\mathbf{w} \cdot \phi(x_i), y_i)
\]
Source Problem

I already know ... vs

• A source a set of data $\hat{D} = \{\hat{x}_i, \hat{y}_i\}_{i=1}^{\hat{N}}$
• With $\hat{N} \gg N$
• Pre-learned model on the source.
  $\hat{\mathbf{w}}$ : solution of the learning problem on the source.
What to Transfer

- $\hat{\mathbf{w}}_j$: solution of the learning problem on the $j$-th source expressed as a weighted sum of kernel functions.
- Use $\hat{\mathbf{w}}_j$ as a reference knowledge when learning $\mathbf{w}$.

**What to transfer?** Discriminative models.
How and When to Transfer

**How:** adaptive regularization.
**When, how much:** reweighted source knowledge.

\[
\min_w \frac{1}{2} \| w - \sum_{j=1}^{J} \beta_j \hat{w}_j \|^2 + C \sum_{i=1}^{N} \ell(w \cdot \phi(x_i), y_i)
\]

We name **KT** the obtained Knowledge Transfer approach.

(Tommasi and Caputo, **BMVC** 2009)
(Tommasi et al., **CVPR** 2010)
Solve the target learning problem

Use the square loss \( \ell^S(f(x), y) = (f(x) - y)^2 \)

Solve
\[
\min_w \frac{1}{2} \left\| w - \sum_{j=1}^{J} \beta_j \hat{w}_j \right\|^2 + \frac{\xi}{2} \sum_{i=1}^{N} \xi_i^2
\]
subject to \( y_i = w \cdot \phi(x_i) + \xi_i \) for \( i = 1, \ldots, N \).

**Adaptive Least-Square Support Vector Machines**

LS-SVM (Suykens et al., 2002)
- square loss: evaluates square error on each sample;
- not sparse: all the training samples contribute to the solution;
- solution: set of linear equations.
Solving Procedure

In matricial form

\[
\begin{bmatrix} \mathbf{K} + \frac{1}{2} \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{a} \end{bmatrix} = \begin{bmatrix} \mathbf{y} - \sum_{j=1}^{J} \beta_j \hat{y}_j \end{bmatrix},
\]

where \( \mathbf{y} = [y_1, \ldots, y_N]^T \), \( \hat{y}_j = [\hat{w}_j \cdot \phi(x_1), \ldots, \hat{w}_j \cdot \phi(x_N)]^T \).

The model parameters can be calculated by matrix inversion

\[
\begin{bmatrix} \mathbf{a} \end{bmatrix} = \mathbf{P} \begin{bmatrix} \mathbf{y} - \sum_{j=1}^{J} \beta_j \hat{y}_j \end{bmatrix}.
\]

Solution: \( \mathbf{w} = \sum_{j=1}^{J} \beta_j \hat{w}_j + \sum_{i=1}^{N} a_i \phi(x_i) \)

Classifier \( f(\mathbf{z}) = \mathbf{w} \cdot \phi(\mathbf{z}) \)
Leave-One-Out Prediction

We can train the learning method on N samples and obtain as a byproduct the prediction for each training sample as if it was left out from the training set.

**Proposition**

Let $a' = Py$ and $a'^j = P\hat{y}_j$ with $a = a' - \sum_{j=1}^{J} \beta_j a'^j$. The prediction $\hat{y}_i$, obtained on sample $i$ when it is removed from the training set, is

$$\hat{y}_i = y_i - \frac{a'_i}{P_{ii}} + \sum_{j=1}^{J} \beta_j \frac{a'^j}{P_{ii}}.$$

The Leave-One-Out error is an almost unbiased estimator of the generalization error (Lunz and Brailovsky, 1969).
Evaluate the relevance of each source

\[ y_i \tilde{y}_i = 1 - y_i \left( \frac{a'_i}{P_{ii}} - \sum_{j=1}^{J} \beta_j \frac{a''_{ij}}{P_{ii}} \right), \]

The best values for beta are those producing positive values for \( y_i \tilde{y}_i \) for each \( i \). To have a convex formulation consider

\[ \ell(\tilde{y}_i, y_i) = \max\{0, 1 - y_i \tilde{y}_i\} \]

and solve

\[ \min_{\beta} \sum_{i=1}^{N} \ell(y_i, \tilde{y}_i) \quad \text{subject to} \quad \|\beta\| \leq 1, \ \beta_j \geq 0. \]

\( \beta \) chooses from where and how much to transfer.
Experimental Evaluation

Visual Object Category Detection
Binary problems: Object vs Non-Object

Data sets:
• Caltech 256

Image Features:
• SIFT (BOW)
• Linear Kernel
Experimental Evaluation

**Visual Object Category Detection**
Binary problems: Object vs Non-Object

**Data sets:**
- Caltech 256

**Image Features:**
- SIFT (BOW)
- Linear Kernel
Experimental Evaluation

Visual Object Category Detection
Binary problems: Object vs Non-Object

Data sets:
• Caltech 256

Image Features:
• SIFT (BOW)
• Linear Kernel
Visual Object Category Detection
Binary problems: Object vs Non-Object

Data sets:
- Caltech 256

Image Features:
- SIFT (BOW)
- Linear Kernel
Experimental Evaluation

Visual Object Category Detection
Binary problems: Object vs Non-Object

Data sets:
• Caltech 256

Image Features:
• SIFT (BOW)
• Linear Kernel
Experimental Evaluation

10 mixed classes, 1 target and 9 sources.
Experimental Evaluation

10 mixed classes, 1 target and 9 sources.

Target

In turn...
Results

Training set:
1-10 positive, 10 negative

Test set:
50 positive, 50 negative

no transfer: $\beta_i = 0$

A-SVM: average $\beta_i = 1/J$
(Aytar and Zisserman, 2011)

Single-KT: best $\beta_i$
(Tommasi and Caputo, BMVC 2009)
Experiments – 6 Unrelated Classes

- Visual Object Classification
- Caltech-256
- Binary problems: object vs non-object
- Features: PHOG, SIFT, Region Covariance, LBP

6 unrelated classes, one target and five sources.
Results – 6 Unrelated Classes

![Graph showing recognition rate vs. number of positive training samples for 6 classes unrelated. The graph compares 'no transfer' and 'KT' methods.](image-url)
Experiments – 2 Unrelated Classes

- Visual Object Classification
- Caltech-256
- Binary problems: object vs non-object
- Features: SIFT

2 unrelated classes, one target and one source.
Results – 2 Unrelated Classes

cereal box – mountain bike

Recognition Rate

# of positive training samples
Transfer Weights and Semantic Similarity

- Use the vectors $\beta$ to define a matrix of class dissimilarities.
- Represent the relation among the classes as distance.
- Apply multidimensional scaling (two dimensions).
Transfer Weights and Semantic Similarity

- Use the vectors $\beta$ to define a matrix of class dissimilarities.
- Represent the relation among the classes as distance.
  ➞ Apply multidimensional scaling (two dimensions).
Moving Forward

Domain Adaptation
Multiclass source and target tasks with exactly the same set of labels.

Personalization of a pre-existent model.

Task: Hand posture classification.

Target: new patient who starts using a mechanical hand prosthesis.

Goal: augment the control abilities, reduce training time.
• 10 sEMG electrodes @100 Hz
  – 8 electrodes in elastic band in a constant position with respect to the radio-humeral joint
  – 2 spare electrodes at flexor and extensors of finger muscles
  – active double-differential

• 22 sensor Cyberglove II @25 Hz

• Inclinometer

Atzori et al. *Building the NINAPRO database: a resource for the biorobotics community*. BIOROB 2012
• 3 exercises for a total of 52 postures
  – 12 basic finger movements
  – 17 static hand postures & basic wrist movements
  – 23 grasps
• 10 repetitions for each posture
• Duration is approximately 2 hours
• 28 subjects
  – 1 amputee
  – 21 male vs. 7 female
  – 26 right-handed vs. 2 left-handed
  – 28.1 ± 3.4 years old
(a) 12 basic flexions and extensions of the fingers

(b) 8 isometric and isotonic hand configurations

(c) 9 basic wrist movements

(d) 23 grasp and functional movements
15 min break!
No Control
No Control
What to transfer

- Source knowledge in the form of black-box classifiers.
- Use them as experts that predict on the target samples.
- Consider the output of the prediction as additional feature elements.

What to transfer? Feature descriptors.
What to transfer

- Source knowledge in the form of black-box classifiers.
- Use them as experts that predict on the target samples.
- Consider the output of the prediction as additional feature elements.

**What to transfer?** Feature descriptors.
How and When to transfer

- Feature combination problem.
- Cast the problem in the multi-kernel learning framework.

**Multiple Kernel Transfer Learning (MKTL).**

\[
\begin{align*}
S_1 & \quad S_2 & \quad S_3 & \quad S_4 & \quad S_5 & \quad S_6 & \quad \text{image} \\
W^1 & \quad W^2 & \quad W^3 & \quad W^4 & \quad W^5 & \quad W^6 & \quad W^0 = \overline{W}
\end{align*}
\]

(Jie*, Tommasi*, Caputo, *ICCV 2011*)
How and When to transfer

- Feature combination problem.
- Cast the problem in the multi-kernel learning framework.

**Multiple Kernel Transfer Learning (MKTL).**

$$S_1 S_2 S_3 S_4 S_5 S_6$$

$$W^{(0,1)} W^{(0,2)} W^{(0,3)} W^{(0,4)} W^{(0,5)} W^{(0,6)} W^0$$

F

F’

Couples: how relevant is each F for each F’.

(Jie*, Tommasi*, Caputo, **ICCV 2011**)
Solve the target learning problem

Optimization problem

$$\min_{\vec{w}} \Omega(\vec{w}) + C \sum_{i=1}^{N} \ell(\vec{w}, x_i, y_i) ,$$

Regularizer

$$\Omega(\vec{w}) = \frac{1}{2} \| \vec{w} \|_{2,p}^2$$

$$= \frac{1}{2} \left\| \left[ \| w^{(0)} \|_2, \| w^{(1,1)} \|_2, \cdots, \| w^{(F,F)} \|_2 \right] \right\|_p^2 ,$$

Loss functions

$$\ell^H(\vec{w}, x, y) = \max \left\{ 0, 1 - y \vec{w} \cdot \vec{\phi}(x) \right\} ,$$

$$\ell^{MC}(\vec{w}, x, y) = \max_{y' \neq y} \left\{ 0, 1 - \vec{w} \cdot (\vec{\phi}(x, y) - \vec{\phi}(x, y')) \right\} .$$

Use an efficient p-norm MKL solver (Orabona et al., 2010)
Experimental Evaluation

Visual Object Category Detection
Binary problems: Object vs Non-Object

Multiclass problems

Data sets:
• Caltech 256

Image Features:
• SIFT, PHOG, Reg. Covariance, LBP, VIS.
• Linear Kernel (sources)
• Gaussian Kernel (target)
Experimental Evaluation

Visual Object Category Detection
Binary problems: Object vs Non-Object

Multiclass problems

Data sets:
• Caltech 256

Image Features:
• SIFT, PHOG, Reg. Covariance, LBP, VIS.
• Linear Kernel (sources)
• Gaussian Kernel (target)
Experimental Evaluation

**Visual Object Category Detection**
Binary problems: Object vs Non-Object

**Multiclass problems**

**Data sets:**
• Caltech 256

**Image Features:**
• SIFT, PHOG, Reg. Covariance, LBP, VIS.
• Linear Kernel (sources)
• Gaussian Kernel (target)
Experimental Evaluation

**Visual Object Category Detection**
Binary problems: Object vs Non-Object

**Multiclass problems**

**Data sets:**
- Caltech 256

**Image Features:**
- SIFT, PHOG, Reg. Covariance, LBP, VIS.
- Linear Kernel (sources)
- Gaussian Kernel (target)
Experimental Evaluation

**Visual Object Category Detection**
Binary problems: Object vs Non-Object

**Multiclass problems**

**Data sets:**
- Caltech 256

**Image Features:**
- SIFT, PHOG, Reg. Covariance, LBP, VIS.
- Linear Kernel (sources)
- Gaussian Kernel (target)
Binary Results

30 mixed classes, 1 target and 29 sources.
Binary Results

30 mixed classes, 1 target and 29 sources.

Target

\[ \hat{w}_1 \ldots \hat{w}_{29} \]
Binary Results

30 mixed classes, 1 target and 29 sources.

Training set:
1-30 positive, 30 negative

Test set:
50 positive, 50 negative

no transfer: $\beta_i = 0$

average-TL: $\beta_i = 1/J$

prior features:

| $S_1$ | $S_2$ | $\cdots$ | $S_{29}$ |
Multiclass Results

- SIFT, LBP + AdaBoost
- REGCOV, SIFT, V1S + SVM
- SIFT + AdaBoost
- SIFT, LBP + SVM

- PHOG
Multiclass Results

SIFT, LBP + AdaB

Recognition Rate

# of positive training samples

no transfer
prior-features
KT
MKTL

LBP + SVM

KT: one-vs-all extension. Homogeneous setting.
Analysis

**KT**
- Binary problems.
- Homogeneous tasks.
- Two optimization steps.

**MKTL**
- Binary and Multiclass problems.
- Homogeneous and Heterogeneous tasks.
- Single optimization step.

**Batch approaches**
The target model and the relevance of each source knowledge must be re-calculated every time a new training sample arrives.
Wrapping up

• General consensus that it is possible to exploit prior knowledge to boost learning of new incoming classes

• Prior knowledge can take different forms and meaning, depending on the task and the designer’s choices

• Research is taking off