Unsupervised Domain Adaptation by Subspace Alignment

Rémi Emonet

Talk at XRCE – 2015-11-27


$ whoami
CC by Bernd Thaller (Flickr)

Overview

- Introduction to Domain Adaptation
- Domain Adaptation by Subspace Alignment
- Landmarks-based Kernelized Subspace Alignment
- More?
  - Contextually Constrained Deep Networks for Scene Labeling
  - Semantic Scene Parsing Using Inconsistent Labelings

Overview

- Introduction to Domain Adaptation
- Domain Adaptation by Subspace Alignment
- Landmarks-based Kernelized Subspace Alignment
- More?
  - Contextually Constrained Deep Networks for Scene Labeling
  - Semantic Scene Parsing Using Inconsistent Labelings

Domain Adaptation: What and Why?

When do we need Domain Adaptation (DA)?

- The training distribution is different from the testing distribution

Example Domain Adaptation task?

- Given: labeled images (e.g., from a Web image corpus)
- Task: is there a Person in unlabeled images (e.g. from a Video corpus)

Person

not-Person

⇒

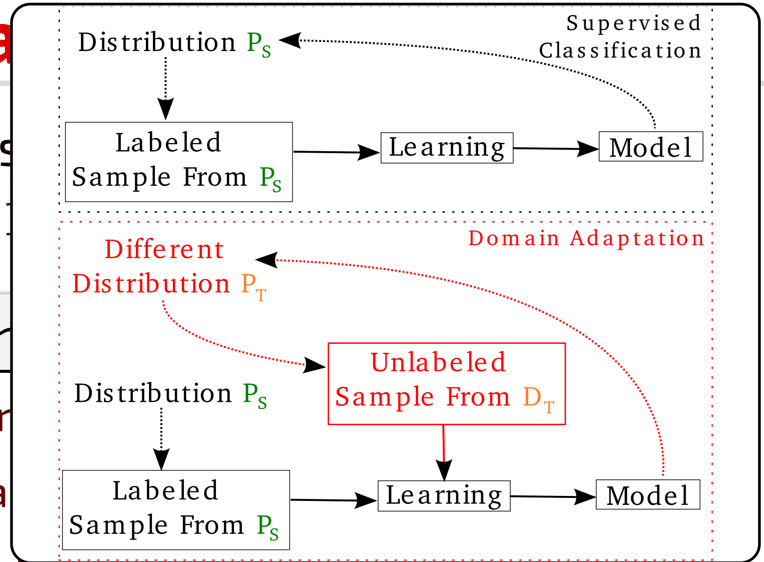
Person?

Person?

- How can we learn, from one distribution,
  a low-error classifier on another distribution?

Domain Adaptation: ta

- Typical binary classification tas
  - X : input space, $Y = \{-1, +1$



Typical supervised classification

- P_S source domain: distribution over
- $S = \{(x_i^s, y_i^s)\}_{i=1}^{m_s} \sim (P_S)^{m_s}$: a sa
- Goal: Find a classifier $h \in \mathrm{H}$ with a low source error

$$R_{P_S}(h) = \mathbf{E}_{(x^s, y^s) \sim P_S} \; \mathbf{I}\left[h(x^s) \neq y^s\right]$$

Domain Adaptation

- P_T target domain: distribution over $X \times Y$, (D_T: marginal over X)
- $T = \{(x_i^t)\}_{j=1}^{m_t} \sim (D_T)^{m_t}$: a sample of unlabeled target points
- Goal: Find a classifier $h \in \mathrm{H}$ with a low target error

$$R_{P_T}(h) = \mathbf{E}_{(x^t, y^t) \sim P_T} \; \mathbf{I}\left[h(x^t) \neq y^t\right]$$

Link the Target Risk to the Source?

$$
\begin{aligned}
R_{P_T}(h) &= \mathbf{E}_{(x^t,y^t)\sim P_T}\mathbf{I}\left[h(x^t) \neq y^t\right] \\
&= \mathbf{E}_{(x^t,y^t)\sim P_T}\frac{P_S(x^t,y^t)}{P_S(x^t,y^t)}\mathbf{I}\left[h(x^t) \neq y^t\right] \\
&= \sum_{(x^t,y^t)} P_T(x^t,y^t)\frac{P_S(x^t,y^t)}{P_S(x^t,y^t)}\mathbf{I}\left[h(x^t) \neq y^t\right] \\
&= \mathbf{E}_{(x^t,y^t)\sim P_S}\frac{P_T(x^t,y^t)}{P_S(x^t,y^t)}\mathbf{I}\left[h(x^t) \neq y^t\right]
\end{aligned}
$$

Domain Adaptation – Covariate Shift?

- $$R_{P_T}(h) = \mathbf{E}_{(x^t,y^t)\sim P_S} \frac{P_T(x^t,y^t)}{P_S(x^t,y^t)} \mathbf{I}\left[h(x^t) \neq y^t\right]$$
- The target risk can be rewritten as an expectation on the source

Covariate Shift

- When $P_S(y^t|x^t) = P_T(y^t|x^t)$ (covariate shift assumption)
- Very strong assumption
- We can estimate a ratio between unlabeled data

$$\begin{aligned} R_{P_T}(h) &= \mathbf{E}_{(x^t,y^t)\sim P_S} \frac{D_T(x^t)P_T(y^t|x^t)}{D_S(x^t)P_S(y^t|x^t)} \mathbf{I}\left[h(x^t) \neq y^t\right] \\ &= \mathbf{E}_{(x^t,y^t)\sim P_S} \frac{D_T(x^t)}{D_S(x^t)} \mathbf{I}\left[h(x^t) \neq y^t\right] \end{aligned}$$

⇒ **Approach**: density estimation and instance re-weighting

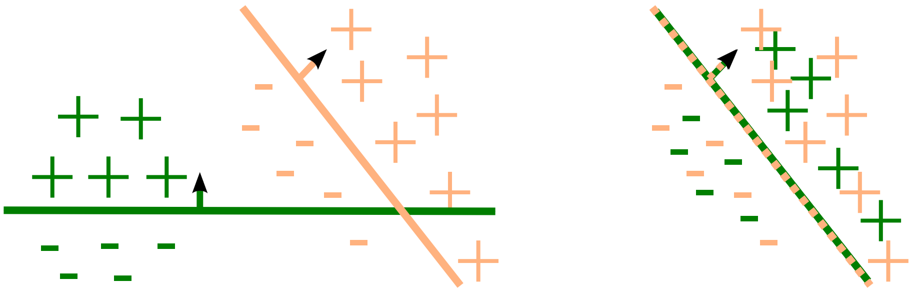
Domain Adaptation – Domain Divergence

Labeled source samples S drawn *i.i.d.* from P_S

Unlabeled target samples T drawn *i.i.d.* from P_T

- h is learned on the source, how does it perform on the target?

  ⇒ it depends on the closeness of the domains



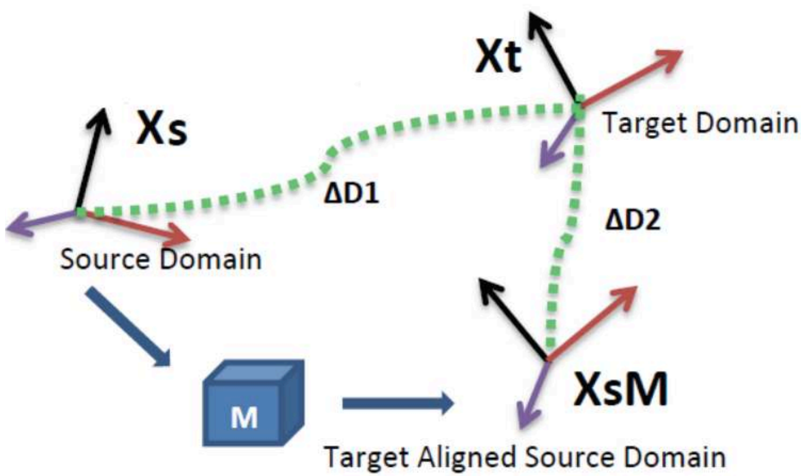
Adaptation Bound [Ben-David et al., MLJ'10, NIPS'06]

- $\forall h \in \mathrm{H}, \quad R_{P_T}(h) \leq R_{P_S}(h) + \frac{1}{2} d_{\mathrm{H}\Delta\mathrm{H}}(D_S, D_T) + \nu$
- Domain divergence: $d_{\mathrm{H}\Delta\mathrm{H}}(D_S, D_T) = 2 \sup_{(h,h')\in\mathrm{H}^2} \left| R_{D_T}(h,h') - R_{D_S}(h,h') \right|$
- Error of the joint optimal classifier: $\nu = \inf_{h'\in\mathrm{H}} \left(R_{P_S}(h') + R_{P_T}(h') \right)$

Overview

- Introduction to Domain Adaptation
- Domain Adaptation by Subspace Alignment
- Landmarks-based Kernelized Subspace Alignment
- More?
    - Contextually Constrained Deep Networks for Scene Labeling
    - Semantic Scene Parsing Using Inconsistent Labelings

- Intuition for unsupervised domain adaptation
  - principal components of the domains may be shared
  - principal components should be re-aligned
- Principle
  - extract a source subspace (d largest eigen vectors)
  - extract a target subspace (d largest eigen vectors)
  - learn a linear mapping function
    that aligns the source subspace with the target one



Subspace Alignment – Algorithm

Algorithm

- **Input:** Source data S, Target data T, Source labels L_S

  Input: Subspace dimension d

  Output: Predicted target labels L_T
- $X_S \leftarrow PCA(S, d)$ *(source subspace defined by the first d eigenvectors)*
- $X_T \leftarrow PCA(T, d)$ *(target subspace defined by the first d eigenvectors)*
- $M \leftarrow {X_S}' X_T$ *(closed form alignment)*
- $X_a \leftarrow X_S M$ *(operator for aligning the source subspace to the target one)*
- $S_a = S X_a$ *(new source data in the aligned space)*
- $T_T = T X_T$ *(new target data in the aligned space)*
- $L_T \leftarrow Classifier(S_a, L_S, T_T)$

- A natural similarity: $Sim(\mathbf{x}_s, \mathbf{x}_t) = \mathbf{x}_s X_S M X_T' \mathbf{x}_t' = \mathbf{x}_s A \mathbf{x}_t'$

Subspace Alignment – Experiments

- Comparison on visual domain adaptation tasks
  - adaptation from Office/Caltech-10 datasets (four domains to adapt)
  - adaptation on ImageNet, LabelMe and Caltech-256 datasets: one is used as source and one as target
- Other methods
  - Baseline 1: projection on the source subspace
  - Baseline 2: projection on the target subspace
  - 2 related methods:
    - GFS [Gopalan et al.,ICCV'11]
    - GFK [Gong et al., CVPR'12]

Subspace Alignment – Results

- Office/Caltech-10 datasets

Method	C→ A	D→ A	W→ A	A→ C	D→ C	W→ C
NA	21.5	26.9	20.8	22.8	24.8	16.4
Baseline 1	38.0	29.8	35.5	30.9	29.6	31.3
Baseline 2	**40.5**	33.0	**38.0**	33.3	31.2	31.9
GFS [8]	36.9	32	27.5	35.3	29.4	21.7
GFK [7]	36.9	32.5	31.1	**35.6**	29.8	27.2
OUR	39.0	**38.0**	37.4	35.3	**32.4**	**32.3**

Method	A→ D	C→ D	W→ D	A→ W	C→ W	D→ W
NA	22.4	21.7	40.5	23.3	20.0	53.0
Baseline 1	34.6	37.4	71.8	35.1	33.5	74.0
Baseline 2	34.7	36.4	72.9	36.8	34.4	78.4
GFS [8]	30.7	32.6	54.3	31.0	30.6	66.0
GFK [7]	35.2	35.2	70.6	34.4	33.7	74.9
OUR	**37.6**	**39.6**	**80.3**	**38.6**	**36.8**	**83.6**

Table 2. Recognition accuracy with unsupervised DA using a NN classif er (Off ce dataset + Caltech10).

Method	C→ A	D→ A	W→ A	A→ C	D→ C	W→ C
Baseline 1	44.3	36.8	32.9	36.8	29.6	24.9
Baseline 2	44.5	38.6	34.2	37.3	31.6	28.4
GFK	44.8	37.9	37.1	38.3	31.4	29.1
OUR	**46.1**	**42.0**	**39.3**	**39.9**	**35.0**	**31.8**

Method	A→ D	C→ D	W→ D	A→ W	C→ W	D→ W
Baseline 1	36.1	38.9	73.6	**42.5**	34.6	75.4
Baseline 2	32.5	35.3	73.6	37.3	34.2	80.5
GFK	37.9	36.1	74.6	39.8	34.9	79.1
OUR	**38.8**	**39.4**	**77.9**	39.6	**38.9**	**82.3**

Table 3. Recognition accuracy with unsupervised DA using a SVM classif er(Off ce dataset + Caltech10).

- ImageNet (I), LabelMe (L) and Caltech-256 (C) datasets

Method	L→ C	L→ I	C→ L	C→ I	I→ L	I→ C	AVG
NA	46.0	38.4	29.5	31.3	36.9	45.5	37.9
Baseline1	24.2	27.2	46.9	41.8	35.7	33.8	34.9
Baseline2	24.6	27.4	**47.0**	**42.0**	35.6	33.8	35.0
GFK	24.2	26.8	44.9	40.7	35.1	33.8	34.3
OUR	**49.1**	**41.2**	**47.0**	39.1	**39.4**	**54.5**	**45.0**

Table 4. Recognition accuracy with unsupervised DA with NN classif er (ImageNet (I), LabelMe (L) and Caltech-256 (C)).

Method	L→ C	L→ I	C→ L	C→ I	I→ L	I→ C	AVG
NA	49.6	40.8	36.0	45.6	41.3	58.9	45.4
Baseline1	50.5	42.0	39.1	48.3	44.0	59.7	47.3
Baseline2	48.7	41.9	39.2	48.4	43.6	58.0	46.6
GFK	52.3	43.5	39.6	49.0	45.3	61.8	48.6
OUR	**52.9**	**43.9**	**43.8**	**50.9**	**46.3**	**62.8**	**50.1**

Table 5. Recognition accuracy with unsupervised DA with SVM classif er (ImageNet (I), LabelMe (L) and Caltech-256 (C)).

Subspace Alignment – Recap.

- Good
  - Very simple and intuitive method
  - Totally unsupervised
  - Theoretical results for dimensionality detection
  - Good results on computer vision datasets
  - Can be combined with supervised information (future work)
- Bad
  - Cannot be directly kernelized to deal with non linearity
  - Actually assumes that spaces are relatively close
- Ugly
  - Assumes that all the source and target examples are relevant

Overview

- Introduction to Domain Adaptation
- Domain Adaptation by Subspace Alignment
- Landmarks-based Kernelized Subspace Alignment
- More?
  - Contextually Constrained Deep Networks for Scene Labeling
  - Semantic Scene Parsing Using Inconsistent Labelings

Subspace Alignment – Recap.

- Good
  - Very simple and intuitive method
  - Totally unsupervised
  - Theoretical results for dimensionality detection
  - Good results on computer vision datasets
  - Can be combined with supervised information (future work)
- Bad
  - Cannot be directly kernelized to deal with non linearity
  - Actually assumes that spaces are relatively close
- Ugly
  - Assumes that all the source and target examples are relevant
- **Idea:** *Select landmarks from both source and target domains to project the data in a common space using a kernel w.r.t those chosen landmarks. Then the subspace alignment is performed.*

Principle of Landmarks

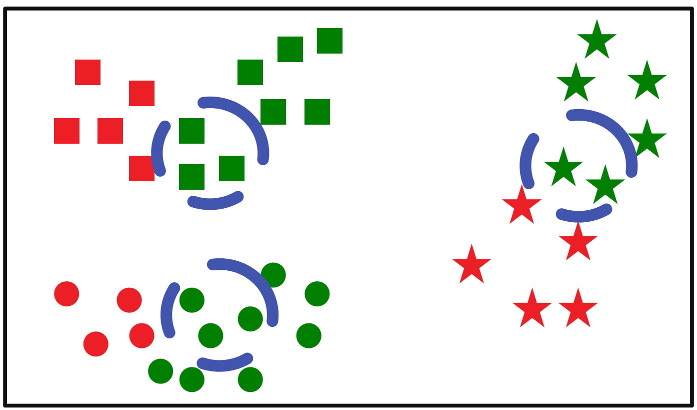
JMLR 2013 – *Connecting the Dots with Landmarks: Discriminatively Learning Domain-Invariant Features for Unsupervised Domain Adaptation*

- Boqing Gong, Kristen Grauman, Fei Sha

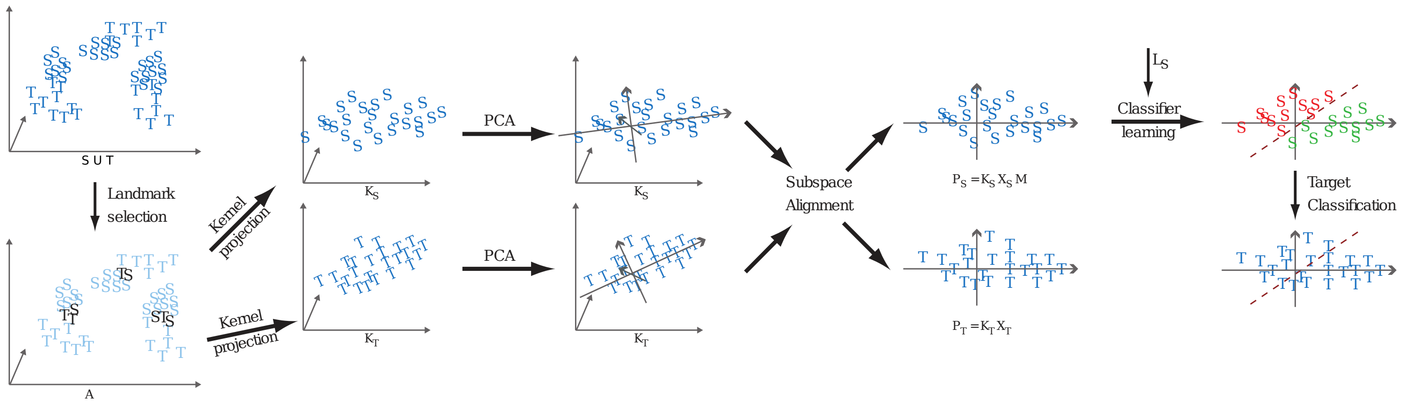
- Principle: find source points (the landmarks) such that the domains are similarly distributed "around"

- Optimization problem: $\min_{\alpha} \left\| \frac{1}{\sum_m \alpha_m} \sum_m \alpha_m \phi(x_m) - \frac{1}{N} \sum_n \phi(x_n) \right\|^2$
  - α: binary landmark indicator variables
  - $\phi(.)$: nonlinear mapping, maps every x to a RKHS
  - minimize the difference in sample-means
  - + a constraint: *labels should be balanced among the landmarks*

- Intuition for landmarks-based alignment
  - subspace alignment does not handle non-linearity
  - subspace alignment cannot “ignore” points
  - landmarks can be a useful to handle locality and non-linearity
- Challenges
  - selecting landmarks in a unsupervised way
  - choosing the proper Gaussian-kernel scale



Proposed Approach – Workflow



- Overall approach
  - 2 new steps: *landmark selection*, *projection* on landmarks
  - subspace alignment

Multiscale Landmark Selection

- Select landmarks among all points, $S \cup T$
- Greedy selection
  - consider each candidate point c and a set of possible scales s
  - criteria to promote the candidate
    - after projection on the candidate
    - the overlap between source and target distributions is above a threshold
- Projection: a point is projected with $K(c,p) = \exp\left(\frac{-\|c-p\|^2}{2s^2}\right)$
- Overlap
  - project source and target points
  - fit two Gaussians (one for each)
  - $overlap(\mu_S, \sigma_S; \mu_T, \sigma_T) = \frac{\mathrm{N}\left(\mu_S - \mu_T \mid 0, \sigma_{sum}^{\;2}\right)}{\mathrm{N}\left(0 \mid 0, \sigma_{sum}^{\;2}\right)}$
    - normalized integral of product
    - with $\sigma_{sum}^2 = {\sigma_S}^2 + {\sigma_T}^2$, and $\mathrm{N}\left(. \mid 0, \sigma_{sum}^2\right)$ centered 1d-Gaussian

Landmark-Based Alignment – Overall

- Select landmarks among all points, $S \cup T$
  - greedy selection
  - multi-scale selection
  - maximize domain overlap
- Project all points on the landmarks
  - use a Gaussian kernel
  - $\sigma \leftarrow median\_distance(S \cup T)$
- Subspace-align the projected points
  - PCA on source domain
  - PCA on target domain
  - compute the alignment M

Landmark-Based Alignment – Results

- Is landmark-based kernelization useful?

Comparison (in terms of accuracy) of unsupervised DA methods. C: Caltech, A: Amazon, W: Webcam, D: Dslr. NA: No Adaptation; KPCA+SA: two independent KPCA are performed on the source and target data, then a subspace alignment is applied; GFK: Geodesic Flow Kernel; SA: one step Subspace Alignment; TJM: Joint Matching Transfer; LSSA: our approach.

Method	A→W	A→D	A→C	C→D	C→W	C→A	W→D	W→A	W→C	D→W	D→C	D→A	Avg
NA	31.5	40.7	45.4	38.2	30.2	50.1	80.2	32.4	31.2	67.8	28.3	30.8	42.2
KPCA+SA	10.1	5.1	7.7	7.6	10.5	10.4	7.6	10.4	11.8	7.2	8.5	7.5	8,7
GFK	38.6	35.7	40.1	44.6	39.0	54.1	81.2	36.6	28.9	80.3	39.2	33.1	45.9
SA	40.7	46.4	41.6	49.0	42.7	52.7	78.9	39.4	34.7	83.4	44.8	38.0	49.3
TJM	42.0	45.8	45.7	49.0	48.8	58.6	83.4	40.8	34.8	82.0	39.6	35.1	50.5
LSSA	42.4	47.2	44.8	54.1	48.1	58.4	87.2	39.4	34.7	87.1	45.7	38.1	52.6

- Is our landmark-selection any good?

Table 1. Comparison (in terms of accuracy) of 5 landmark selection methods on 12 unsupervised DA subproblems. C: Caltech, A: Amazon, W: Webcam, D: Dslr. RD: Random Selection; All: all the source and target examples are used; σ-LS: our selection method with a f xed σ; CDL: Connecting Dots with Landmarks; MLS: our approach. In red, one reports the best method.

Method	A→W	A→D	A→C	C→D	C→W	C→A	W→D	W→A	W→C	D→W	D→C	D→A	Avg
RD	40.3	38.8	42.3	41.2	40.6	47.5	84.0	32.9	28.4	81.8	36.8	32.3	45.6
All	41.0	39.4	44.7	41.4	41.6	49.6	85.3	33.0	29.2	82.7	38.6	31.3	46.5
σ-LS	39.3	37.5	43.8	42.7	31.5	52.4	80.3	32.6	29.5	82.0	38.6	31.2	45.1
CDL	38.3	38.8	43.9	45.8	45.4	51.7	77.7	35.3	30.9	72.5	33.9	33.3	45.6
MLS	41.1	39.5	45.0	45.2	44.1	53.6	84.7	35.9	31.6	82.4	39.2	34.5	48.1

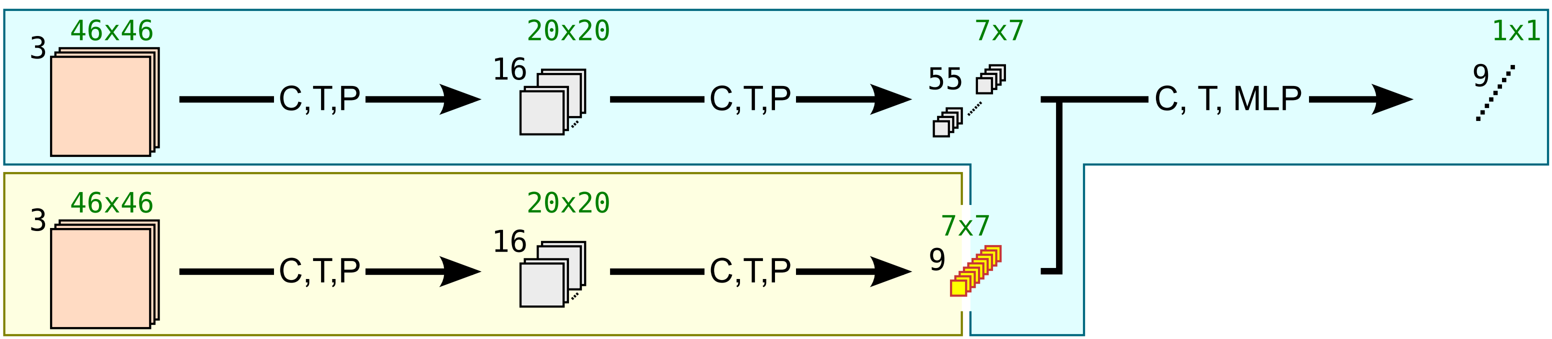
Overview

- Introduction to Domain Adaptation
- Domain Adaptation by Subspace Alignment
- Landmarks-based Kernelized Subspace Alignment
- More?
  - Contextually Constrained Deep Networks for Scene Labeling
  - Semantic Scene Parsing Using Inconsistent Labelings

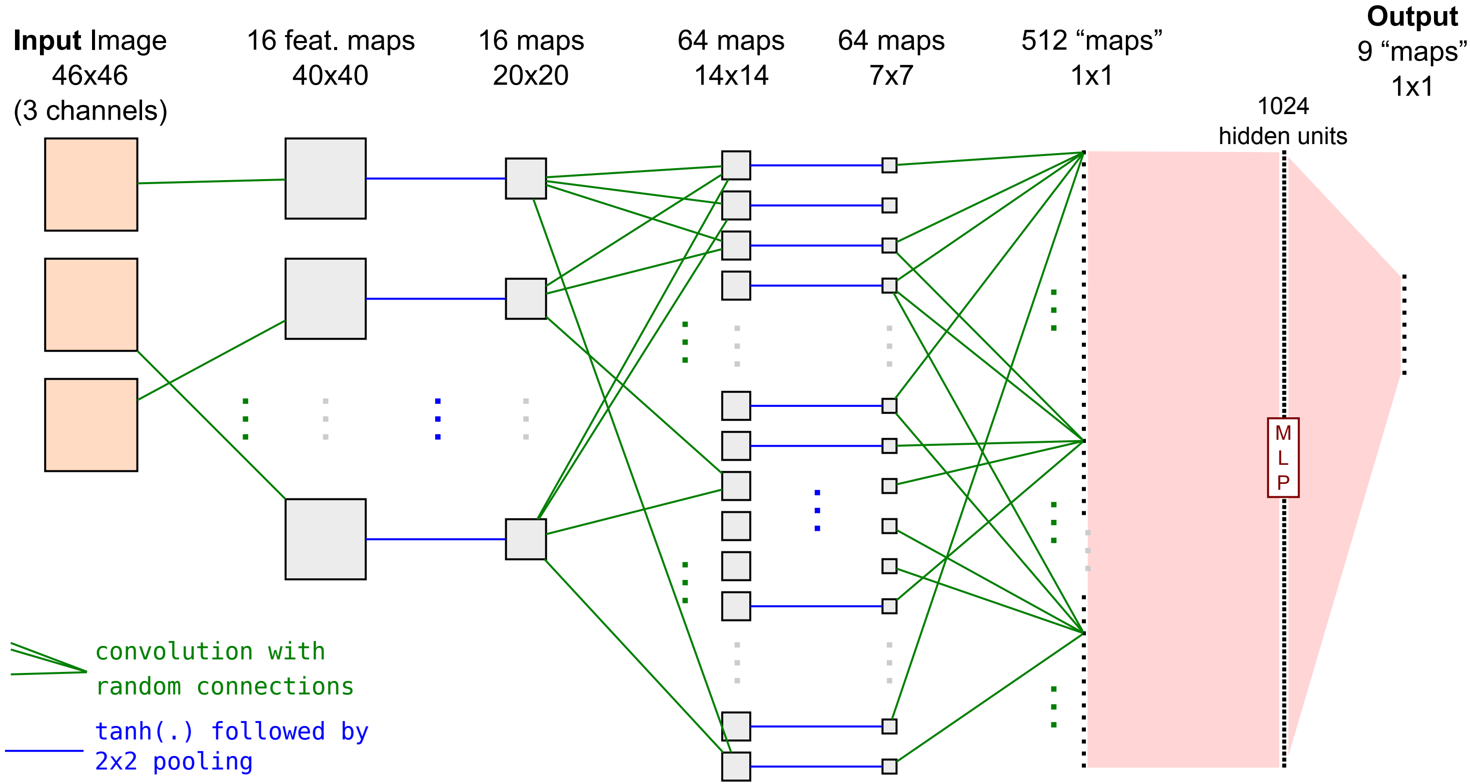
Task: Semantic Scene Labeling

- For each pixel in an image (or video), predict its class
    - e.g., building, road, car, pedestrian, sign, ...

- Observation
  - state of the art uses deep CNN (conv. net.)
  - learning is patch-based, using the center label
  - training images are densely labeled
- Idea
  - use labels in the patch to guide the network
  - force a part of the network to use the context (like an MRF)

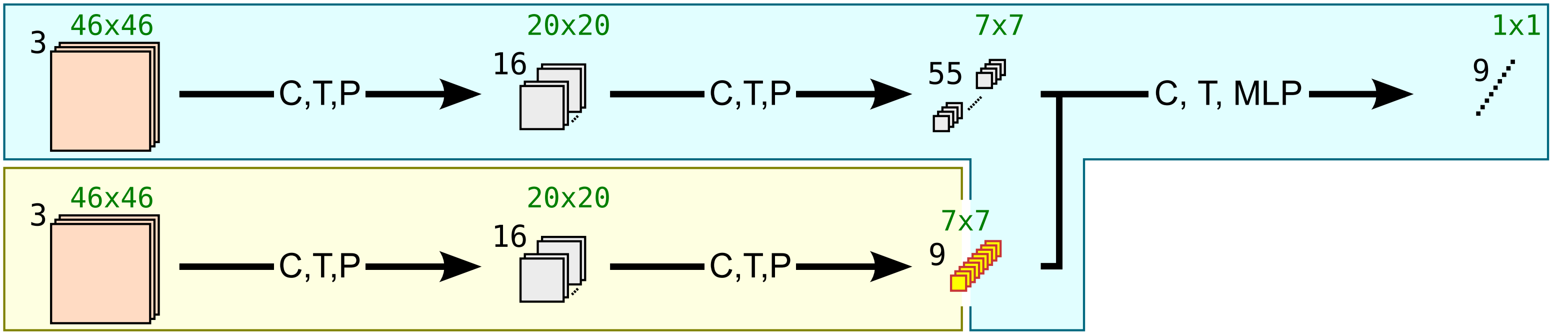


The Network



Input Image
46x46
(3 channels)
16 feat. maps
40x40
16 maps
20x20
64 maps
14x14
64 maps
7x7
512 “maps”
1x1
1024
hidden units
Output
9 “maps”
1x1
M
L
P
convolution with
random connections
tanh(.) followed by
2x2 pooling

Multi-Step Learning



- Learn the context net (yellow)
- Learn the dependent net (blue)
  - freeze the context net
  - use prediction, mixed with some ground truth (probability τ)
- Fine tuning
  - unfreeze the context net
  - no intermediate supervision
  - allow for co-adaptation

Contextually Constrained – Results

Architecture	Stanford Dataset Pixel Acc.	Stanford Dataset Class Acc.	SIFT Flow Dataset Pixel Acc.	SIFT Flow Dataset Class Acc.	number of # param.	train speed
ContextL	54.19	45.12	42.52	9.89	4.4k	0.75x
ConvNet	69.72	66.24	48.02	44.04	700k	1x
AugL ($\tau = 0$)	72.06	67.22	48.93	44.53	701k	1.1x
AugL ($\tau = 0.05$)	71.97	66.16	49.39	44.54	701k	1.1x
msContextL	55.39	50.06	44.71	10.20	4.4k	2.1x
msConvNet	75.67	67.1	69.93	45.65	1224k	2.70x
msAugL ($\tau = 0$)	76.05	68.01	70.88	44.82	1225k	2.85x
msAugL ($\tau = 0.05$)	76.36	68.52	70.42	45.80	1225k	2.85x

- Context: KITTI dataset
  - urban scenes recorded from a car
  - many sensors (RGB, stereo, laser, ...), different tasks
- Observation (scene labeling on KITTI)
  - different groups labeled frames
  - they used different frames (mostly) and different labels
  - the quality/precision of annotations varies
- Goal
  - leverage all these annotations
  - improve segmentation on individual labelsets/datasets

Authors	Train	Validation	Test	Total
He et al. [9]	32	7	12	51
Kundu et al. [15]	28	7	12	50
Ladicky et al. [16]	24	6	30	60
Ros et al. [20]	80	20	46	146
Sengupta et al. [21]	36	9	25	70
Xu et al. [29]	56	14	37	107
Zhang et al. [32]	112	28	112	252
Total	368	91	277	736

Labels

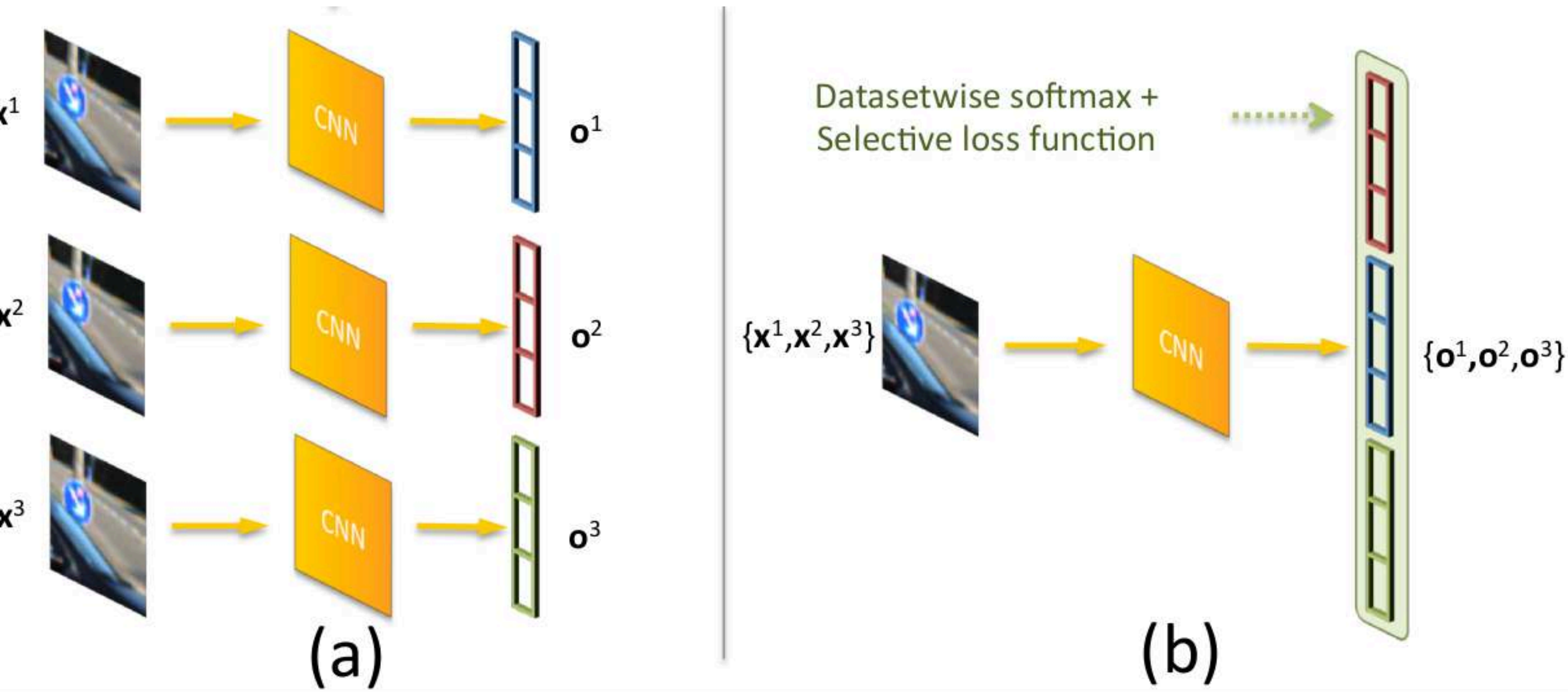
- 7 different label sets

He et al.	Road	Building	Sky	Tree	Sidewalk	Car	Pedestrian	Bicyclist				VegetationMisc
Kundu et al.	Road	Building	Sky	Vegetation	Sidewalk	Car	Pedestrian	Cyclist	Pole	Sign	Fence	
Ladicky et al.	Road	Building	Sky	Tree	Sidewalk	Car	Pedestrian	Bike	Column	Sign	Fence	Grass
Ros et al.	Road	Building	Sky	Vegetation	Sidewalk	Car	Pedestrian	Cyclist	Pole	Sign	Fence	
Sengupta et al.	Road	Building	Sky	Veg.	Pavement	Car	Pedestrian		Poles	Signage	Fence	
Xu et al.	Ground	Infrastructure	Sky	Vegetation		Movable						
Zhang et al.	Road	Building	Sky	Vegetation	Sidewalk	Car	Pedestrian	Cyclist		Signage	Fence	

Authors	Train	Validation	Test	Total
He et al. [9]	32	7	12	51
Kundu et al. [15]	28	7	12	50
Ladicky et al. [16]	24	6	30	60
Ros et al. [20]	80	20	46	146
Sengupta et al. [21]	36	9	25	70
Xu et al. [29]	56	14	37	107
Zhang et al. [32]	112	28	112	252
Total	368	91	277	736

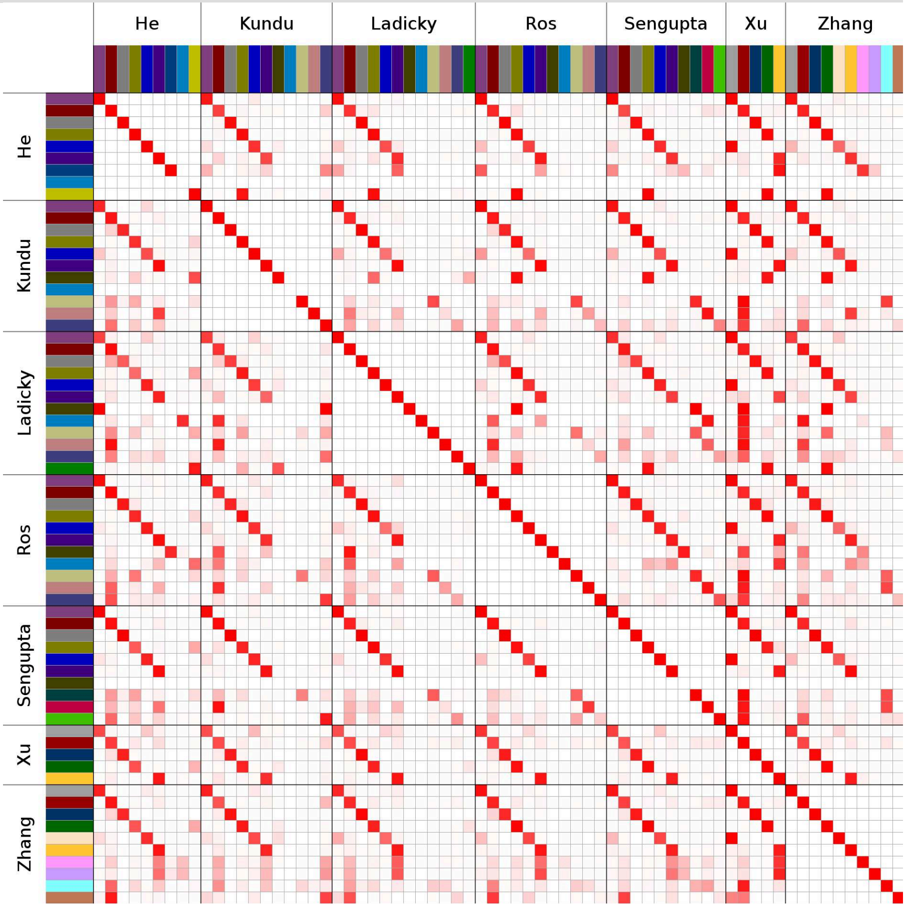
First Approach

- a) Baseline: separate training
- b) Joint Training
  - with datasetwise soft-max
  - with selective loss function



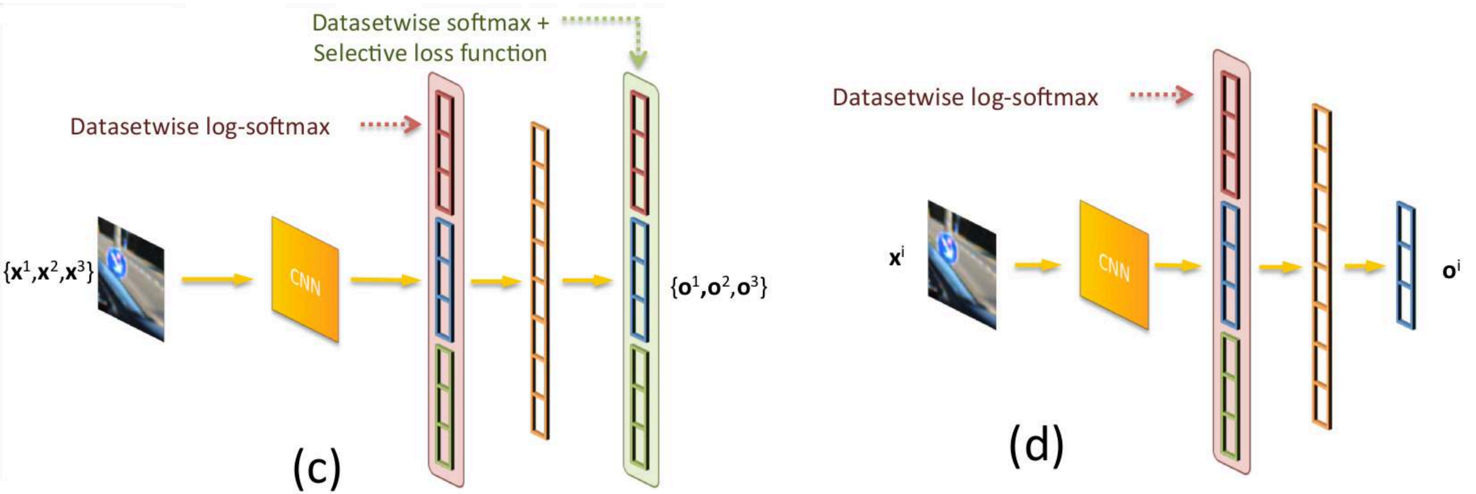
Labels Correlation After Joint Training

- Observing outputs after joint training
  - correlation across datasets
  - clear correspondence for some labels
  - one-to-many correspondences



Exploiting Correlations after Joint Training

- c) Joint Training with shared context
  - a single network to learn all correlations
- d) Joint Training with individual context
  - a specialized network per labeling



Joint Training – Results

Results with all classes available in the ground truth										
		He	Kundu	Ladicky	Ros	Sengupta	Xu	Zhang	Total	
No Fusion	Global	74.67	72.48	72.94	76.96	78.71	86.97	84.98	80.94	-
	Average	58.56	56.04	43.16	48.76	71.26	83.11	57.39	57.14	-
Joint training	Global	78.68	77.20	75.86	78.22	81.48	88.02	86.89	82.75	(+1.81)
	Average	64.41	60.61	46.52	52.06	75.64	85.14	60.54	60.99	(+3.85)
Joint training with shared context	Global	78.61	77.76	76.00	78.40	81.97	88.43	87.54	83.16	(+2.22)
	Average	62.87	59.13	45.22	51.16	75.55	84.94	59.75	60.03	(+2.89)
Joint training with individual context	Global	79.31	77.53	76.81	78.41	80.98	88.35	86.76	83.19	(+2.25)
	Average	64.15	59.77	47.92	52.35	77.19	85.09	59.84	61.24	(+4.10)

Table 3. Pixel (Global) and Class (Average) accuracy results for the 7 used sub-datasets with 4 different training strategies: NF=No Fusion (see Fig. 2a) ; JT= Joint training (see Fig. 2b); JTSC=Joint training with shared context (see Fig. 2c); JTIC=Joint training with individual context (see Fig. 2d). Best results are highlighted in bold.

Overview

- Introduction to Domain Adaptation
- Domain Adaptation by Subspace Alignment
- Landmarks-based Kernelized Subspace Alignment
- More?
  - Contextually Constrained Deep Networks for Scene Labeling
  - Semantic Scene Parsing Using Inconsistent Labelings

Thanks! More Questions?

Unsupervised Domain Adaptation by Subspace Alignment

Rémi Emonet

Talk at XRCE – 2015-11-27