Geodesic Flow Kernel for Unsupervised Domain Adaptation

Boqing Gong
University of Southern California

Joint work with Yuan Shi, Fei Sha, and Kristen Grauman
Motivation

Mismatch between different domains/datasets

- Object recognition
  - Ex. [Torralba & Efros’11, Perronnin et al.’10]
- Video analysis
  - Ex. [Duan et al.’09, 10]
- Pedestrian detection
  - Ex. [Dollár et al.’09]
- Other vision tasks

Performance degrades significantly!

Images from [Saenko et al.’10].
Unsupervised domain adaptation

• **Source domain (labeled)**

\[ D_S = \{(x_i, y_i), i = 1, 2, \ldots, N\} \sim P_S(X, Y) \]

• **Target domain (unlabeled)**

\[ D_T = \{(x_i, ?), i = 1, 2, \ldots, M\} \sim P_T(X, Y) \]

• **Objective**

Train classification model to work well on the target

*The two distributions are not the same!*
Challenges

• How to optimally, w.r.t. target domain, define discriminative loss function; select model, tune parameters

• How to solve this ill-posed problem? impose additional structure
Examples of existing approaches

• Correcting sample bias
  – Ex. [Shimodaira’00, Huang et al.’06, Bickel et al.’07]
  – Assumption: marginal distributions are the only difference.

• Learning transductively
  – Ex. [Bergamo & Torresani’10, Bruzzone & Marconcini’10]
  – Assumption: classifiers have high-confidence predictions across domains.

• Learning a shared representation
  – Ex. [Daumé III’07, Pan et al.’09, Gopalan et al.’11]
  – Assumption: a latent feature space exists in which classification hypotheses fit both domains.
Our approach: learning a shared representation

Key insight: bridging the gap

- Fantasize infinite number of domains
- Integrate out analytically idiosyncrasies in domains
- Learn invariant features by constructing kernel
Main idea: geodesic flow kernel

1. Model data with **linear subspaces**
2. Model domain shift with **geodesic flow**
3. Derive domain-invariant features with **kernel**
4. Classify target data with the **new features**
Modeling data with linear subspaces

Assume low-dimensional structure

Ex. PCA, Partial Least Squares (source only)
Characterizing domains geometrically

Grassmann manifold $G(d, D)$

- Collection of $d$-dimensional subspaces of a vector space $\mathbb{R}^D$ ($d < D$)
- Each point corresponds to a subspace
Modeling domain shift with geodesic flow

Geodesic flow on the manifold

- starting at source & arriving at target in unit time
- flow parameterized with one parameter
- closed-form, easy to compute with SVD
Modeling domain shift with geodesic flow

\[
\Phi(0) \rightarrow \Phi(t), 0 \leq t \leq 1 \\
\Phi(1) \rightarrow \mathbb{G}(d,D)
\]

Subspaces:

Domains:

Source

Target
Modeling domain shift with geodesic flow

Along this flow, points (subspaces) represent intermediate domains.
Domain-invariant features

\[ z^\infty = \begin{bmatrix} \Phi(0)^T x, \cdots, \Phi(t)^T x, \cdots, \Phi(1)^T x \end{bmatrix} \]

More similar to source.
Domain-invariant features

\[ z^\infty = [\Phi(0)^T x, \cdots, \Phi(t)^T x, \cdots, \Phi(1)^T x] \]

More similar to target.
Domain-invariant features

\[ z^\infty = [\Phi(0)^T x, \ldots, \Phi(t)^T x, \ldots, \Phi(1)^T x] \]

Blend the two.
Measuring feature similarities with inner products

\[ z_i^\infty = [\Phi(0)^T x_i, \ldots, \Phi(t)^T x_i, \ldots, \Phi(1)^T x_i ] \]

\[ z_j^\infty = [\Phi(0)^T x_j, \ldots, \Phi(t)^T x_j, \ldots, \Phi(1)^T x_j ] \]

More similar to source.

More similar to target.

\[ \langle z_i^\infty, z_j^\infty \rangle : \text{Invariant to either source or target.} \]
Learning domain-invariant features with kernels

We define the **geodesic flow kernel (GFK)**:

\[
\langle z_i^\infty, z_j^\infty \rangle = \int_0^1 (\Phi(t)^T x_i)^T (\Phi(t)^T x_j) dt = x_i^T G x_j
\]

- **Advantages**
  - Analytically computable
  - Robust to variants towards either source or target
  - Broadly applicable: can kernelize many classifiers
Contrast to discretely sampling

GFK (ours)

\[ \langle z_i^\infty, z_j^\infty \rangle = \int_0^1 (\Phi(t)^T x_i)^T (\Phi(t)^T x_j) dt = x_i^T G x_j \]

No free parameters

[Gopalan et al. ICCV 2011]

GFK is conceptually cleaner and computationally more tractable.

Dimensionality reduction

Number of subspaces, dimensionality of subspace, dimensionality after reduction
Recap of key steps

1. \( \mathbb{G}(d,D) \)  

Target subspace

2. Source subspace

3. 

\[
\mathbf{z}^\infty = \begin{pmatrix}
\Phi(0)^T \\
\Phi(t)^T \\
\Phi(1)^T
\end{pmatrix} \mathbf{x}
\]

4. \( H^\infty \)  

\[
\langle \mathbf{z}_i^\infty, \mathbf{z}_j^\infty \rangle = x_i \mathbf{G} x_j
\]
Experimental setup

- Four domains
- Features
  - Bag-of-SURF
- Classifier: 1NN
- Average over 20 random trials
Classification accuracy on target

Accuracy (%)

Source $\rightarrow$ Target

- W $\rightarrow$ C
- W $\rightarrow$ A
- C $\rightarrow$ D
- C $\rightarrow$ A
- A $\rightarrow$ W
- A $\rightarrow$ C
- D $\rightarrow$ A

No adaptation

Referenced: Gopalan et al.'11, GFK (ours)
Classification accuracy on target

Accuracy (%)

Source $\rightarrow$ Target

No adaptation  [Gopalan et al.'11]

W$\rightarrow$C  W$\rightarrow$A  C$\rightarrow$D  C$\rightarrow$A  A$\rightarrow$W  A$\rightarrow$C  D$\rightarrow$A
Classification accuracy on target

Accuracy (%)

Source → Target

- W --> C
- W --> A
- C --> D
- C --> A
- A --> W
- A --> C
- D --> A

Legend:
- Blue: No adaptation
- Red: [Gopalan et al.'11]
- Purple: GFK (ours)
Which domain should be used as the source?

- DSLR
- Caltech-256
- Webcam

Amazon
We introduce the **Rank of Domains** measure:

\[ R(S, \mathcal{T}) = \frac{1}{d^*} \sum_{i}^{d^*} \theta_i \left[ KL(S_i \| T_i) + KL(T_i \| S_i) \right] \]

**Intuition**

- Geometrically, how subspaces disagree
- Statistically, how distributions disagree
Automatically selecting the best

<table>
<thead>
<tr>
<th>Possible sources</th>
<th>Our ROD measure</th>
</tr>
</thead>
<tbody>
<tr>
<td>Caltech-256</td>
<td>0.003</td>
</tr>
<tr>
<td>Amazon</td>
<td>0</td>
</tr>
<tr>
<td>DSLR</td>
<td>0.26</td>
</tr>
<tr>
<td>Webcam</td>
<td>0.05</td>
</tr>
</tbody>
</table>

Caltech-256 adapts the best to Amazon.
Semi-supervised domain adaptation

Label three instances per category in the target

Source → Target

- W → C
- W → A
- C → D
- C → A
- A → W
- A → C
- D → A

No adaptation
[Saenko et al.'10]
[Gopalan et al.'11]
GFK (ours)

Accuracy (%)
Analyzing datasets in light of domain adaptation

Cross-dataset generalization [Torralba & Efros’11]
Analyzing datasets in light of domain adaptation

Cross-dataset generalization [Torralba & Efros’11]

Caltech-101 generalizes the worst. Performance drop of ImageNet is big.
Analyzing datasets in light of domain adaptation

Cross-dataset generalization [Torralba & Efros’11]

Caltech-101 generalizes the worst (w/ or w/o adaptation). There is nearly no performance drop of ImageNet.

Performance drop becomes smaller!
Summary

• Unsupervised domain adaptation
  – Important in visual recognition
  – Challenge: no labeled data from the target

• Geodesic flow kernel (GFK)
  – Conceptually clean formulation: no free parameter
  – Computationally tractable: closed-form solution
  – Empirically successful: state-of-the-art results

• New insight on vision datasets
  – Cross-dataset generalization with domain adaptation
  – Leveraging existing datasets despite their idiosyncrasies
Future work

• Beyond subspaces
  Other techniques to model domain shift

• From GFK to statistical flow kernel
  Add more statistical properties to the flow

• Applications of GFK
  Ex., face recognition, video analysis
Summary

- Unsupervised domain adaptation
  - Important in visual recognition
  - Challenge: no labeled data from the target
- Geodesic flow kernel (GFK)
  - Conceptually clean formulation
  - Computationally tractable
  - Empirically successful
- New insight on vision datasets
  - Cross-dataset generalization with domain adaptation
  - Leveraging existing datasets despite their idiosyncrasies