Image Analogies for Domain Adaptation

Jeff Donahue

CS294-69 – Final Project Presentation

November 30th, 2011

Problem

- Image classification datasets are biased
- Training classifiers on one dataset results in much lower accuracy when testing on others [1]

task	Test on: / Train on:	SUN09	LabelMe	PASCAL	ImageNet	Caltech101	MSRC	Self	Mean others	Percent drop
"car" classification	SUN09	**28.2**	29.5	16.3	14.6	16.9	21.9	28.2	19.8	**30%**
	LabelMe	14.7	**34.0**	16.7	22.9	43.6	24.5	34.0	24.5	**28%**
	PASCAL	10.1	25.5	**35.2**	43.9	44.2	39.4	35.2	32.6	**7%**
	ImageNet	11.4	29.6	36.0	**57.4**	52.3	42.7	57.4	34.4	**40%**
	Caltech101	7.5	31.1	19.5	33.1	**96.9**	42.1	96.9	26.7	**73%**
	MSRC	9.3	27.0	24.9	32.6	40.3	**68.4**	68.4	26.8	**61%**
	Mean others	10.6	28.5	22.7	29.4	39.4	34.1	53.4	27.5	48%

[1] Torralba, A. and Efros, A. Unbiased Look at Dataset Bias. CVPR 2011.

Problem (2)

- Lots of examples of classes in a source domain A

- Few/no examples of target domain B

- Want to classify B using what we know about A

Main Idea

- Use **image analogies [2]** to synthesize examples of novel classes in target domain

[2] Aaron Hertzmann, Charles E. Jacobs, Nuria Oliver, Brian Curless and David H. Salesin. Image Analogies. SIGGRAPH 2001. *(Figure taken from project page.)*

Image Analogies [2]

[2] Aaron Hertzmann, Charles E. Jacobs, Nuria Oliver, Brian Curless and David H. Salesin. Image Analogies. SIGGRAPH 2001. *(Figures taken from paper.)*

Caveats

- Won’t work in general
  - Assumes the domain transform is simply a filter
  - Requires a pair of “matching” images from the two domains
  - Doesn’t extend to, e.g., bias in image recognition benchmark datasets
- But still has potential real-world applications
  - Different cameras produce different textures (particularly for low-quality vs. high-quality cameras)

Approach

- DSLR images from office dataset are domain A
- Reduced resolution, added Gaussian blur and Gaussian noise to create synthetic domain B

```
B = imnoise(imfilter(imresize(A, [400 400]), fspecial
('gaussian', 6, 6), 'gaussian', 0, 0.003);
```

Real DSLR Image (A)

Synthetic "Webcam" Image (B)

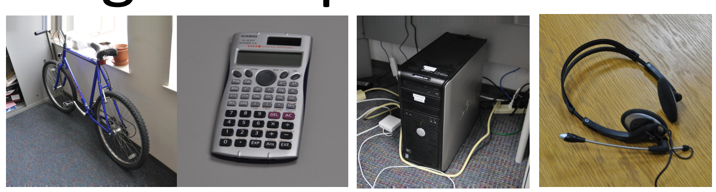
Approach (2)

- Use single pair of matching examples from A and B to get analogous images for rest of data

Approach (3)

- Inputs to system:
  - Single pair of matching images from domains A and B:

  - More training examples from all classes in domain A:



- Output: classifiers for classes in domain B

Algorithm Overview

1) Use the one known domainA/domainB image pair to generate analogous images for all other domainA images in the training set

2) Compute bag-of-words histograms of SIFT features for generated analogous images

3) Train one-vs.-all linear SVMs on the BoW histograms to get classifiers for domain B

Experimental Design

- **“Ours”**: run algorithm from previous slide to get domain B classifiers
- **Baseline**: just train SVMs on domain A and apply directly to classifying domain B
- Compare classification accuracy for domain B
- 13 classes
- 5 training examples per class – same images per trial in ours, baseline (test on the rest)

Experimental Design (2)

Original Images (Train Exes for Baseline) | Generated by Analogy (Train Exes for Ours) | Synthetic Domain (Test Examples)

Results

- Results over 50 trials of random train/test splits:
  - **Ours**: 64.33% accuracy
  - **Baseline**: 25.50% accuracy
  - **"Ideal"**: 72.07% accuracy
  - **Chance**: 07.69% accuracy
- 152% improvement over baseline (significant with $\alpha = 10^{-45}$ in paired t-test)
- Not too far off from "ideal" performance (training on actual target domain)

Conclusions

- A domain transform in which textures are added can significantly confuse a classifier
- Image analogies can effectively capture certain types of domain transforms and aid in classification
- Potentially useful in applications where it is known that the domain transform is mainly explained by a texture change

Future Work

- Apply to a real-world domain transform by taking same photo with two different cameras/lenses/etc.
  - Somewhat difficult setup, but only need one pair of matching images

- Extend image analogies to capture more general types of domain transforms?