DESCRIBING OBJECTS WITH ATTRIBUTES

Visual recognition course
By Yarin Didi & Guy Ezer
DESCRIBING OBJECTS BY THEIR ATTRIBUTES
Concept – simplify recognition

- Reduction the problem of recognition to smaller and more simple tasks.
Problem – recognition lack with details

- Naming objects alone is not very informative.
What do we want to know?

- recognition program: is a dog.
- Human in the scene:
  - “puppy”
  - “running”
  - “move forward”
  - “looks happy and calm”
  - “bright”
Attributes – infer properties

Can I poke with it?
What shape is it?
Does it have a tail?

Is it alive?

Can I put stuff in it?
Is it soft?
Will it blend?
Principle - General attribute

- Attributes might and should be shared with relevant categories.

Snout should be shared by all relevant classes
What do we want to know?

- Attributes formation:
  - is alive
  - Has four legs
  - Has ears
  - Texture: Is furry
  - Color: #yellow
  - Has snout
Motivation - detailed information

Lets us to perform some more complex queries.

<table>
<thead>
<tr>
<th>Detailed information</th>
<th>Recognition</th>
</tr>
</thead>
<tbody>
<tr>
<td>is alive</td>
<td>Dog/puppy</td>
</tr>
<tr>
<td>Has four legs</td>
<td></td>
</tr>
<tr>
<td>Has ears</td>
<td></td>
</tr>
<tr>
<td>Texture: Is furry</td>
<td></td>
</tr>
<tr>
<td>Color: #yellow</td>
<td></td>
</tr>
<tr>
<td>Has snout</td>
<td></td>
</tr>
<tr>
<td>Bright puppy</td>
<td></td>
</tr>
<tr>
<td>running</td>
<td></td>
</tr>
<tr>
<td>move</td>
<td></td>
</tr>
<tr>
<td>forward</td>
<td></td>
</tr>
<tr>
<td>looks happy</td>
<td></td>
</tr>
<tr>
<td>and calm</td>
<td></td>
</tr>
</tbody>
</table>
Motivation – describe unknown object

- The ability to infer some properties of unknown abject is better than conclude nothing.

<table>
<thead>
<tr>
<th>Information</th>
<th>Zero information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Has horn</td>
<td>unknown</td>
</tr>
<tr>
<td>Has snout</td>
<td></td>
</tr>
<tr>
<td>Has strips</td>
<td></td>
</tr>
<tr>
<td>Has tail</td>
<td></td>
</tr>
<tr>
<td>Has two legs</td>
<td></td>
</tr>
<tr>
<td>Has hands</td>
<td></td>
</tr>
<tr>
<td>Has eyes</td>
<td></td>
</tr>
<tr>
<td>standing</td>
<td></td>
</tr>
</tbody>
</table>
Motivation - report about unusual things
Abilities

- Naming: Aeroplane
  - Description: Unknown
    - Unusual attributes: Has Wheel, Has Wood
    - Unexpected attributes: Bird, No Head, No Beak
  - Unexpected attributes: Motorbike, Has Cloth
- Textual description: Has Horn, Has leg, Has Head, Has Wool
Scheme

Feature extraction

Feature Selection

Attribute Predictions

Category Models

Bird

Has Beak, Has Eye, Has foot, Has Feather

Attribute Classifiers
Semantic attribute

Colors

Textures

Parts

Shape
Semantic attributes

• Bag of words for each:
  • Texture descriptors – (for materials)
    • Computed for each pixel and quantized to NN 256 kmeans, extracted with a texton filterbank.
  • Visual words – (parts)
    • HOG spatial pyramid quantized to NN 1000 kmeans.
  • Edges –
    • standard canny edge detector.
  • Color descriptors – (materials)
    • Sampled for each pixel, and quantized to the NN 128 kmeans.
Bag of words – reminder

1. sampling “images”
2. learning a dictionary - kmeans
3. test image
Correlated attribute problem

- The classifier might be accidentally learn correlated feature instead the feature.
Bounding box problem

Bounding-box might influence with irrelevant issues

One possible solution is appropriate segmentation
Conventional feature selection will not apply to the problem because confused with semantically irrelevant correlation.

Instead of choosing all features we will choose a subset of it that perform well distinguishing.
Deal with correlation

- select features that perform well at distinguishing examples of cars with and without wheel.
- Run same procedure to for bikes, motorbikes, boats…
- Learn the “wheel” classifier with the selected features.

“Has Wheels” vs. “No Wheels”
Correlation between ground-truth “wheel” and metallic

<table>
<thead>
<tr>
<th></th>
<th>PASCAL</th>
<th>Yahoo</th>
</tr>
</thead>
<tbody>
<tr>
<td>whole</td>
<td>0.71</td>
<td>0.17</td>
</tr>
</tbody>
</table>

We might notice that the generalizing worked because it

Testing on Yahoo

<table>
<thead>
<tr>
<th></th>
<th>Whole features</th>
<th>Selection features</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.56</td>
<td>0.28</td>
</tr>
</tbody>
</table>

Training with on PASCAL
Influence of feature selection

<table>
<thead>
<tr>
<th></th>
<th>With feature selection</th>
<th>Without feature selection</th>
</tr>
</thead>
<tbody>
<tr>
<td>learning from textual description</td>
<td>25.2%</td>
<td>32.5%</td>
</tr>
<tr>
<td>reporting the absence of typical attributes</td>
<td>54.8%</td>
<td>68.2%</td>
</tr>
<tr>
<td>presence of atypical attributes</td>
<td>24.5%</td>
<td>47.3%</td>
</tr>
</tbody>
</table>
Localization using a grid

Furniture-leg tend to appear in the lower part of the image.
Differences from parts based model

- The parts based model is not flexible to other classes because it used the latent parts on an object.
- This paper define “parts” which can be shared with other classes.
- Side effect of “feature selection” is localizing.

\[ L^* = \arg \min_L \left( \sum_{i=1}^{n} m_i(l_i) + \sum_{(v_i,v_j) \in E} d_{ij}(l_i, l_j) \right) \]
Datasets

- **a-Pascal**
  - 20 categories from PASCAL 2008 train dataset (10K object images)
  - airplane, bicycle, bird, boat, bottle, bus, car, cat, chair, cow, dining table, dog, horse, motorbike, person, potted plant, sheep, sofa, train, tv monitor
- **Annotation via Amazon’s Mechanical Turk**
- **a-Yahoo**
  - 12 new categories from Yahoo image search
  - bag, building, carriage, centaur, donkey, goat, jet ski, mug, monkey, statue of person, wolf, zebra
  - Categories chosen to share attributes with those in Pascal
Predicting attributes

- Train on 20 object classes from a-Pascal train set
  - Feature selection for each attribute
  - Train a linear SVM classifier
- Test on 12 object classes from Yahoo image search (cross-category) or on a-Pascal test set (within-category)
  - Apply learned classifiers to predict each attribute
Experiments: description
Experiments: naming - absence of typical attributes

752 reports 68% are correct
Experiments: presence of atypical attributes

951 reports 47% are correct
Experiments : localization of attributes

Feature selection is necessary to localize attributes.
Experiments: localization of atypical attributes
Experiments: localization using selected features
Experiments – naming – directly

<table>
<thead>
<tr>
<th>PASCAL 08</th>
<th>Base Features</th>
<th>Whole Features</th>
<th>Selected Features</th>
</tr>
</thead>
<tbody>
<tr>
<td>SVM</td>
<td>58.5 (35.5)</td>
<td>56.1 (34.3)</td>
<td>58.3 (38.1)</td>
</tr>
<tr>
<td>Logistic Regression</td>
<td>54.6 (36.9)</td>
<td>51.2 (31.4)</td>
<td>53.4 (33.5)</td>
</tr>
</tbody>
</table>
Attribute prediction

• Attribute prediction for attribute classifiers trained on a-Pascal and tested on a-Pascal.
• Mid-plastic, 3/2 D boxy, Metal, Shiny
• Bad: Feather, tail, Horn, close

This means that the correlation statistics are not changing during training and testing. So generalize don’t need.
Attribute prediction - across category protocols
Experiment – Naming

Accuracy vs. Number of Training Examples Per Category

From Textual Description
- Sel.Feat.
- WholeFeat.

Sem + Dis Attributes
Sem. Attributes
Base Features

Chance
Conclusions

• Inferring object properties should be an important goal of object recognition
• Learning attributes enables several new abilities:
  • Predicting properties of new types of objects
  • Identifying unusual about a familiar object
  • Learning from verbal description
• Raises an important issue concerning dataset biases while learning.
Attributes and Simile Classifier for face verification

Agenda:

- Face verification using attributes
- Face verification using simile
Face Verification

Given two face images – determine if they show the same individual.
Introduction

There is large variability in ways which the same face presents itself to the camera, for example:

• Pose
• Hairstyle & Skin
• Expression
• Illumination
Recognition using visual attributes

- Different objects can be described by attributes:
  - Stripes
  - Four-legged
  - Furry
  - Male
  - Child
  - Black hair
Attributes can define categories

Female  Eyeglasses  Middle-aged  Dark hair
The Attributes Approach

• The first step in our approach is to extract “low-level” features – such as SIFT, HOG, the pixel values.

• The second step is to use these low-level features to compute “high-level” visual features, or traits, which are insensitive to changes in pose, illumination and expression.

• The third step is preforming the verification, based on the traits values we get.
The Attributes Approach

Images ➔ Low-level features ➔ Attributes ➔ Verification

RGB
HOG
SIFT
...

RGB
HOG
SIFT
...

Different

Male
Asian
Dark hair
Round jaw

+ + + - -
Low-level features

• We first detect faces and fiducial point locations using a commercial face detector.
• The faces are then recertified to a common coordinate system using an affine wrap.
• The low-level features are constructed by choosing a face region, a feature type to extract from this region, and a normalization and aggression of these values.
Low-level features – cont.

The regions are constructed by hand-labeling different parts of the rectified face images – nose, eyes, mouth, etc.

- Regions are defined manually, but only once.

- This division takes advantage of the common geometry of human faces, while still allowing differences.

- Extracted values are normalized to generalize the image.
Low-level features – cont.

A sample face split into regions of interest:
Low-level features – cont.

- There are many features types – as RGB and HSV color spaces, edge magnitudes and gradient directions.

- We get a large number of low-level features $\{f_i\}$ - those features are used for each trait classifier $C_i$ as described next.
Traits Classifiers

- For each attribute, we build a classifier $C_i$ to detect the describable attribute.
Traits Classifiers – cont.

• We train several attribute classifiers, each requires a set of positive and negative examples for each attribute.

• Uses a simplified version of adaboost.

• Best features for classification chosen automatically from our low-level features, these are used to train the final classifiers.
Learning an attribute classifier

Training images \rightarrow Low-level features \rightarrow Feature selection \rightarrow Train classifier

Males: RGB, HoG, HSV, Edges, Mouth
Females: RGB, HoG, Eyes, HSV, Hair

Feature selection: RGB, Nose, HoG, Eyes, HSV, Hair, Edges, Mouth

Gender classifier: Male 0.87
# Traits Classifiers

65 attributes classifiers were trained using the system.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Asian</td>
<td>92.32%</td>
</tr>
<tr>
<td>Attractive Woman</td>
<td>81.13%</td>
</tr>
<tr>
<td>Baby</td>
<td>90.45%</td>
</tr>
<tr>
<td>Bags Under Eyes</td>
<td>86.23%</td>
</tr>
<tr>
<td>Bald</td>
<td>83.22%</td>
</tr>
<tr>
<td>Bangs</td>
<td>88.70%</td>
</tr>
<tr>
<td>Black</td>
<td>88.65%</td>
</tr>
<tr>
<td>Black Hair</td>
<td>80.32%</td>
</tr>
<tr>
<td>Blond Hair</td>
<td>78.05%</td>
</tr>
<tr>
<td>Blurry</td>
<td>92.12%</td>
</tr>
<tr>
<td>Brown Hair</td>
<td>72.42%</td>
</tr>
<tr>
<td>Child</td>
<td>83.58%</td>
</tr>
<tr>
<td>Chubby</td>
<td>77.24%</td>
</tr>
<tr>
<td>Color Photo</td>
<td>95.50%</td>
</tr>
<tr>
<td>Curly Hair</td>
<td>68.88%</td>
</tr>
<tr>
<td>Double Chin</td>
<td>77.68%</td>
</tr>
<tr>
<td>Environment</td>
<td>84.80%</td>
</tr>
<tr>
<td>Eye Width</td>
<td>90.02%</td>
</tr>
<tr>
<td>Eyebrow Shape</td>
<td>80.90%</td>
</tr>
<tr>
<td>Eyebrow Thickness</td>
<td>93.40%</td>
</tr>
<tr>
<td>Eyeglasses</td>
<td>91.56%</td>
</tr>
<tr>
<td>Eyes Open</td>
<td>92.52%</td>
</tr>
<tr>
<td>Flash Lighting</td>
<td>72.33%</td>
</tr>
<tr>
<td>Frowning</td>
<td>95.47%</td>
</tr>
<tr>
<td>Goatee</td>
<td>80.35%</td>
</tr>
<tr>
<td>Gray Hair</td>
<td>87.18%</td>
</tr>
<tr>
<td>Harsh Lighting</td>
<td>78.74%</td>
</tr>
<tr>
<td>High Cheekbones</td>
<td>84.70%</td>
</tr>
<tr>
<td>Indian</td>
<td>86.47%</td>
</tr>
<tr>
<td>Male</td>
<td>81.22%</td>
</tr>
<tr>
<td>Middle-Aged</td>
<td>78.39%</td>
</tr>
<tr>
<td>Mouth Closed</td>
<td>89.27%</td>
</tr>
<tr>
<td>Mouth Partially Open</td>
<td>85.13%</td>
</tr>
<tr>
<td>Mouth Wide Open</td>
<td>89.63%</td>
</tr>
<tr>
<td>Mustache</td>
<td>91.88%</td>
</tr>
<tr>
<td>No Beard</td>
<td>89.53%</td>
</tr>
<tr>
<td>No Eyewear</td>
<td>93.55%</td>
</tr>
<tr>
<td>Nose Shape</td>
<td>86.87%</td>
</tr>
<tr>
<td>Nose Size</td>
<td>87.50%</td>
</tr>
<tr>
<td>Nose-Mouth Lines</td>
<td>93.10%</td>
</tr>
<tr>
<td>Obstructed Forehead</td>
<td>79.11%</td>
</tr>
<tr>
<td>Oval Face</td>
<td>70.26%</td>
</tr>
<tr>
<td>Pale Skin</td>
<td>89.44%</td>
</tr>
<tr>
<td>Posed Photo</td>
<td>69.72%</td>
</tr>
<tr>
<td>Receding Hairline</td>
<td>84.15%</td>
</tr>
<tr>
<td>Rosy Cheeks</td>
<td>85.82%</td>
</tr>
<tr>
<td>Round Face</td>
<td>74.33%</td>
</tr>
<tr>
<td>Round Jaw</td>
<td>66.99%</td>
</tr>
<tr>
<td>Semi-Obscured Forehead</td>
<td>77.02%</td>
</tr>
<tr>
<td>Senior</td>
<td>88.74%</td>
</tr>
<tr>
<td>Shiny Skin</td>
<td>84.73%</td>
</tr>
<tr>
<td>Sideburns</td>
<td>71.07%</td>
</tr>
<tr>
<td>Smiling</td>
<td>95.33%</td>
</tr>
<tr>
<td>Soft Lighting</td>
<td>67.81%</td>
</tr>
<tr>
<td>Square Face</td>
<td>81.19%</td>
</tr>
<tr>
<td>Straight Hair</td>
<td>76.81%</td>
</tr>
<tr>
<td>Sunglasses</td>
<td>94.91%</td>
</tr>
<tr>
<td>Teeth Not Visible</td>
<td>91.64%</td>
</tr>
<tr>
<td>Teeth Visible</td>
<td>91.64%</td>
</tr>
<tr>
<td>Visible Forehead</td>
<td>89.43%</td>
</tr>
<tr>
<td>Wavy Hair</td>
<td>64.49%</td>
</tr>
<tr>
<td>Wearing Hat</td>
<td>85.97%</td>
</tr>
<tr>
<td>Wearing Lipstick</td>
<td>86.78%</td>
</tr>
<tr>
<td>White</td>
<td>91.48%</td>
</tr>
<tr>
<td>Youth</td>
<td>85.79%</td>
</tr>
</tbody>
</table>
Traits Classifiers – collecting examples

• Each attribute had at least 1,000 examples – at least 500 positive and 500 negative.

• That means, that for 65 attributes, at least 65,000 images needed to be obtained and labeled!
Trait Classifiers – cont.

We can see a correlation between the trait values of photos of the same person.

- The “Flash” and “Shiny skin” attributes produce differently.
- The rest of the attributes are still agree despite the change in pose.
Simile Classifiers

• One can imagine that there are many visual cues to people’s identities that cannot be described – at least not concisely.

• In order to use this information, we use the concept of a “simile” classifier.

• The basic idea is that we can describe a person's appearance in terms of the similarity of different parts of their face to a limited set of “reference” people.
Simile Classifier

Female

Black hair

Female

Mouth closed

White

Young
Simile Classifier

Penelope Cruz

Angelina Jolie
Simile Classifier – cont.

- For each reference person, several simile classifiers are trained for each face region – one per feature type, yielding a large set of total classifiers.
Simile Classifier – cont.

We can see correlation between the similies values of photos of the same person:
Performing verification

To decide if two face images, $I_1$ and $I_2$ are the same person, we compare their trait vectors using a final classifier $F$ which defines our verification function:

$$v(I_1, I_2) = D(C(I_1), C(I_2))$$

This function is positive when the two images show the same person and negative otherwise. $D$ is a verification classifier.
Verification Classifier

- The trait vectors are obtained by the attributes method, or by the similies method (or both).

- As said, corresponding values of $C_i(I_1)$ and $C_i(I_2)$ from the $i$-th iteration should be similar if the images are of the same person.

- In addition, the SVM classifier optimize for separating data at the separation boundary – so differences in values close to 0 are more important than differences between those with greater absolute values.
Verification Classifier – cont.

- Let \( a_i \) and \( b_i \) be the outputs of the \( i \)-th trait classifier for each image \((I_1, I_2)\) - \( a_i = C_i(I_1), b_i = C_i(I_2) \).

- The absolute value of \( a_i - b_i \) nets us the similarity result, and the product \( a_i \times b_i \) gives us the signal.

- We save the pair \( p_i = \left( |a_i - b_i|, (a_i \times b_i) \times g\left(\frac{1}{2}(a_i + b_i)\right) \right) \) where \( g \) is a gaussian with mean 0 and variance 1.
Verification Classifier – cont.

- These pairs are concatenated to form the $2n$ dimension vector that we actually classify:

$$v(I_1, I_2) = D(\langle p_1, p_2, ..., p_n \rangle)$$

- Training D requires pairs of positive examples and negative examples.

- SVM with an RBF kernel was used for D.
Experiments and Results

Tested on two datasets:


- PubFig – a more challenging dataset. Build by the writers of the article.
Performance on LFW

An improvement of 31.68% in error rates.
Human face verification performance

Human still perform well even with inverse crop – strong context cues available on the LFW dataset!
PubFig dataset & benchmark

Public figures:
- Politicians
- Celebrities

Larger & deeper:
- 60,000 Images
- 200 People
- 300 Images per person

Subsets:
- Pose
- Illumination
- Expression

http://www.cs.columbia.edu/CAVE/databases/pubfig/
Performance on PubFig

[Graph showing ROC curves for different conditions with annotations]

- Full PubFig Benchmark (77.78%)
- "Easy" Pose (80.81%)
- "Difficult" Pose (77.50%)
- "Easy" Lighting (79.54%)
- "Difficult" Lighting (75.32%)
- "Easy" Expression (78.40%)
- "Difficult" Expression (77.66%)

All results computed using our Attribute Classifiers
Future work

• Further improvements can be made by using:
  • More attributes
  • Improving the training process
  • Combining attributes and simile classifiers with low level image cues.

• Applying attributes to other objects?
Observations

• Taken also background as a feature, will error-rate on the LFW dataset decrease even more?

• Can similar results be achieved if only low-level data were used?
Summary

- Two approaches for face verification using traits computed in face images
  - Attributes
  - Similie

- Improvement on the LFW dataset – without the background or the context, but still there is room for improvement.

- PubFig dataset – larger and deeper than previous datasets.
Resources

• Attribute and Simile Classifiers for Face Verification, N. Kumar, A. Berg, P. Belhumeur, S. Nayar. ICCV 2009
• Reading group presentation by Esa Rahtu http://www.robots.ox.ac.uk/~vgg/rg/slides/faceverification.pdf
• Attribute and Simile Classifiers for Face Verification http://www.cs.columbia.edu/CAVE/projects/faceverification/
• Labeled Faces in the Wild http://vis-www.cs.umass.edu/lfw/
• PubFig http://www.cs.columbia.edu/CAVE/databases/pubfig/