Describing objects with attributes

Attribute and Simile Classifiers for Face Verification, N. Kumar, A. Berg, P. Belhumeur, S. Nayar. ICCV 2009

Describing Objects by Their Attributes, A. Farhadi, I. Endres, D. Hoiem, and D. Forsyth, CVPR 2009

Recognition and Classification in Images and Video, Haifa university

By: Bahjat Musa

26 May 2014

Outline

- Introduction
- Attribute and Simile Classifiers for Face Verification
- Describing Objects by Their Attributes
- Conclusion
- References

Outline

- **Introduction**
- Attribute and Simile Classifiers for Face Verification
- Describing Objects by Their Attributes
- Conclusion
- References

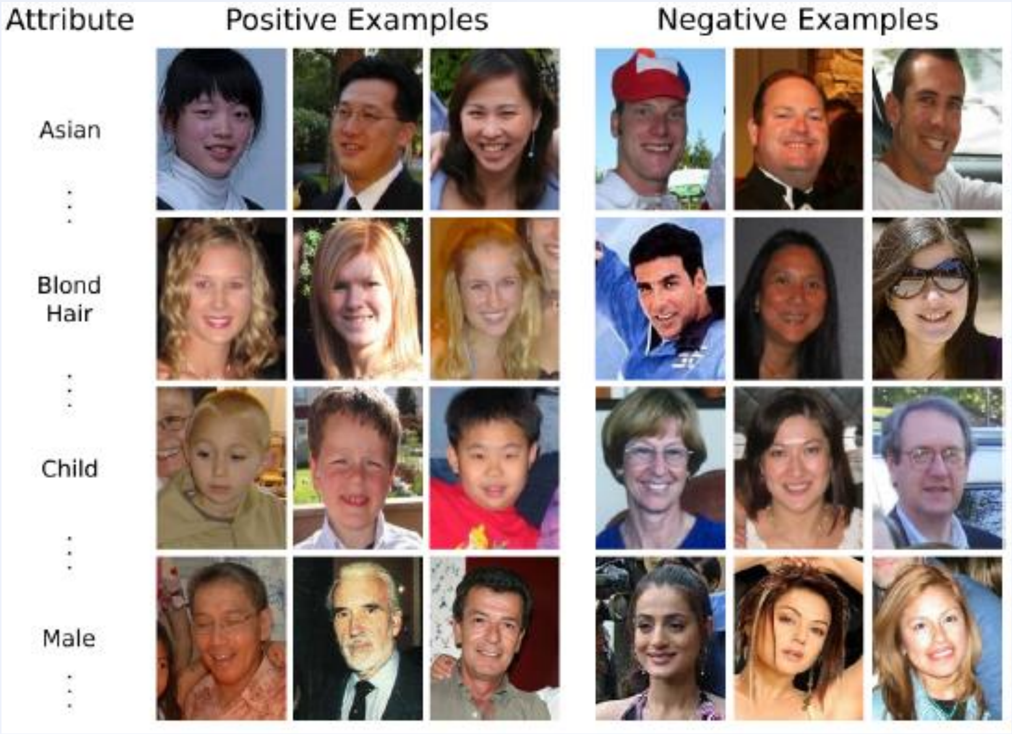
Introduction

- A lot of methods and papers where published with different approaches, giving many solutions for problems like naming objects and recognizing faces.
- Today we will discuss a new approach, “Describing objects by it’s attributes”.
- Two new attribute based frameworks, one for face verification and the other for recognizing objects.
- Main concept: recognizing objects by detecting appearance or absence of it’s attributes.

Introduction

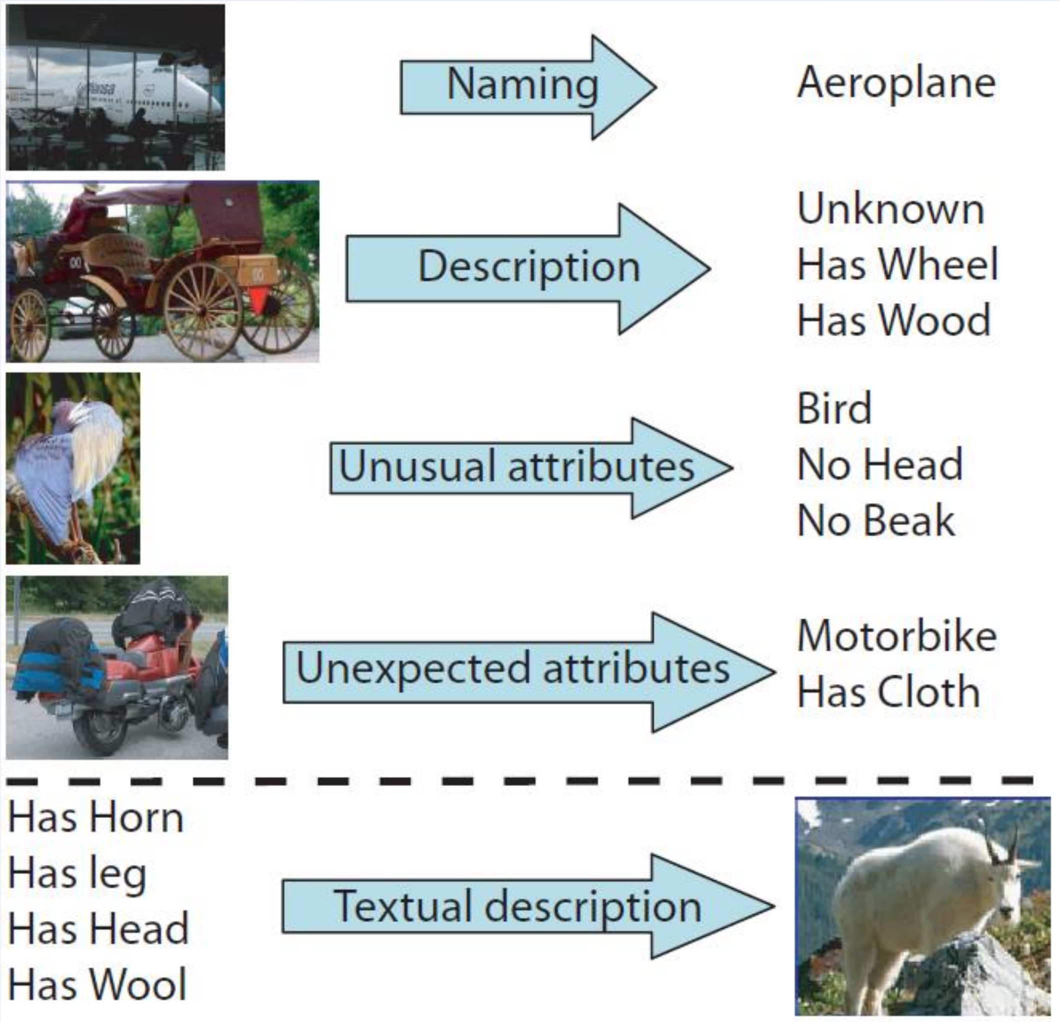
Attribute and Simile Classifiers for Face Verification

- Two novel and complementary methods for face verification.
- Common to both methods is the idea of extracting and comparing "high-level" visual features, or traits, of a face image.
- Insensitive to pose, illumination, expression, and other imaging conditions.
- Easier training data requirement.



Introduction
Describing Objects by their Attributes

- Rather than focusing on identity assignment, inferring attributes will be the core problem of recognition.
- Shifting the goal of recognition from naming to describing allows:
  - ➢ Naming objects
  - ➢ Reporting unusual aspects of a familiar object
  - ➢ Saying something about unknown objects, not just “unknown”
  - ➢ learning to recognize new objects with few examples or textual description



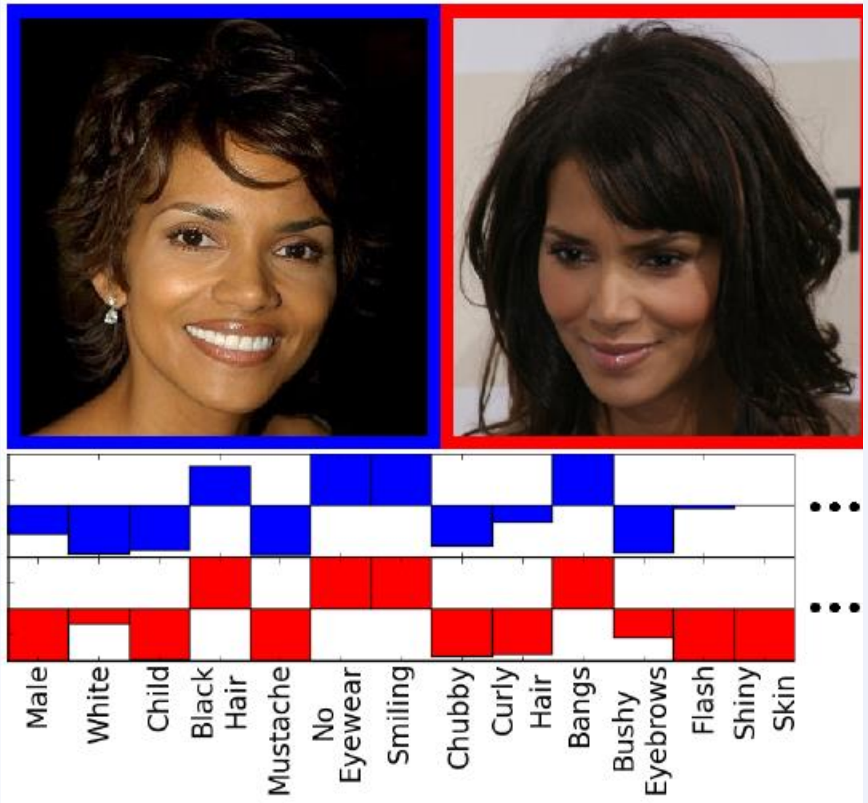
Outline

- ✓Introduction
- ➢**Attribute and Simile Classifiers for Face Verification**
  - ➢**Steps to perform face verification**
  - ➢**Attribute classifier**
  - ➢**Simile classifier**
  - ➢**Experiments**
- Describing Objects by Their Attributes
- Conclusion
- References

Two trait classifiers:

The first – based on attribute classifiers

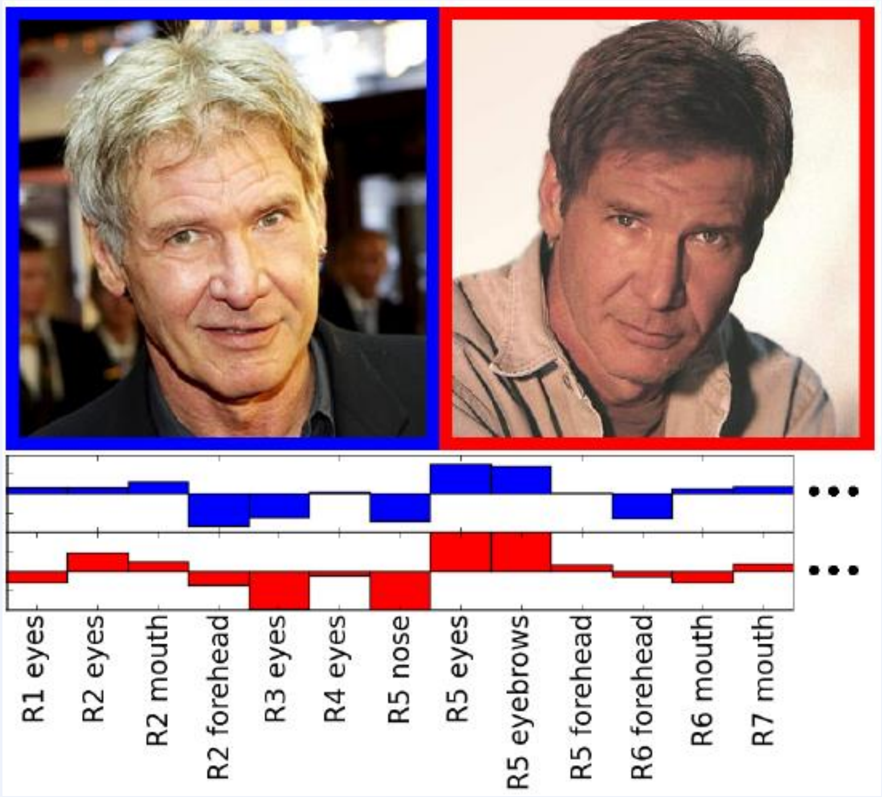
- Uses binary classifiers trained to recognize the presence or absence of describable aspects of visual appearance.
- With 65 describable visual traits such as gender, age, race, hair color, etc… the classifiers improve on the state-of-the-art, reducing overall error rates by 23.92% on LFW.



Two trait classifiers:

The second – based on simile classifiers

- Binary classifiers trained to recognize the similarity of faces, or regions of faces, to specific reference people.
- Unknown face might be described as having a mouth that looks like Barack Obama's and a nose that looks like Owen Wilson's.
- With similarities to a set of 60 reference faces, the classifiers improve on the state-of-the-art, reducing overall error rates by 26.34% on LFW.
- Does not require the manual labeling of training sets.



Steps to perform face verification
on a pair of images

1. Extract low-level features

   Extract the output of k low-level features and concatenate these vectors to form a large feature vector - $F(I) = \{f_1(I), \ldots, f_k(I)\}$.

2. Compute visual traits

   For each extracted feature vector $F(I)$, compute the output of n trait classifiers in order to produce a “trait vector” - $C(I) = \{C_1(F(I)), \ldots, C_n(F(I))\}$.

3. Perform verification

   To decide if two face images I_1 and I_2 are of the same person, we compare their trait vectors using a final classifier D which defines our verification function – $v(I_1, I_2) = D(C(I_1), C(I_2))$, where positive result means match.

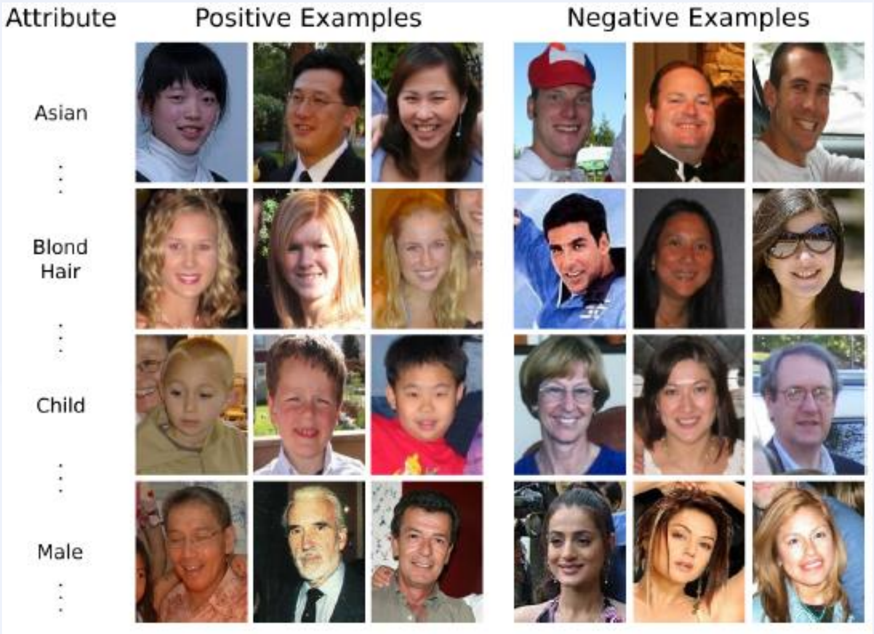
1. Extract low-level features

1. Detect faces and fiducial point locations using a commercial face detector.
2. Rectify the faces to a common coordinate system using an affine warp based on the fiducials.
3. Construct low-level features by choosing a face region and a feature type to extract from this region

This produces a large number of possible low-level features, a subset of which is automatically chosen and used for each trait classifier.

2. Compute visual traits
Attribute classifier

- Build classifiers to detect the describable attributes of faces.
- Train the classifiers with a set of positive and negative example images for each attribute.
- Use a simplified version of adaboost to choose from the set of low-level features, up to 6 features per classifier.
- Each attribute classifier is an SVM with an RBF kernel.



2. Compute visual traits

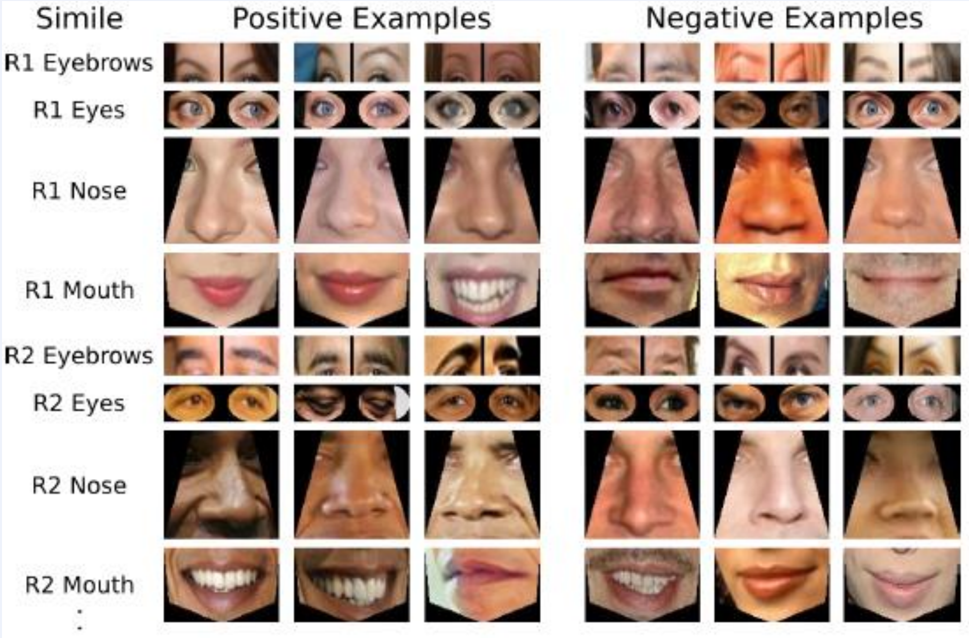
Attribute classifier - results

Accuracies of the 65 attribute classifiers trained using the procedure described earlier

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

2. Compute visual traits
simile classifier

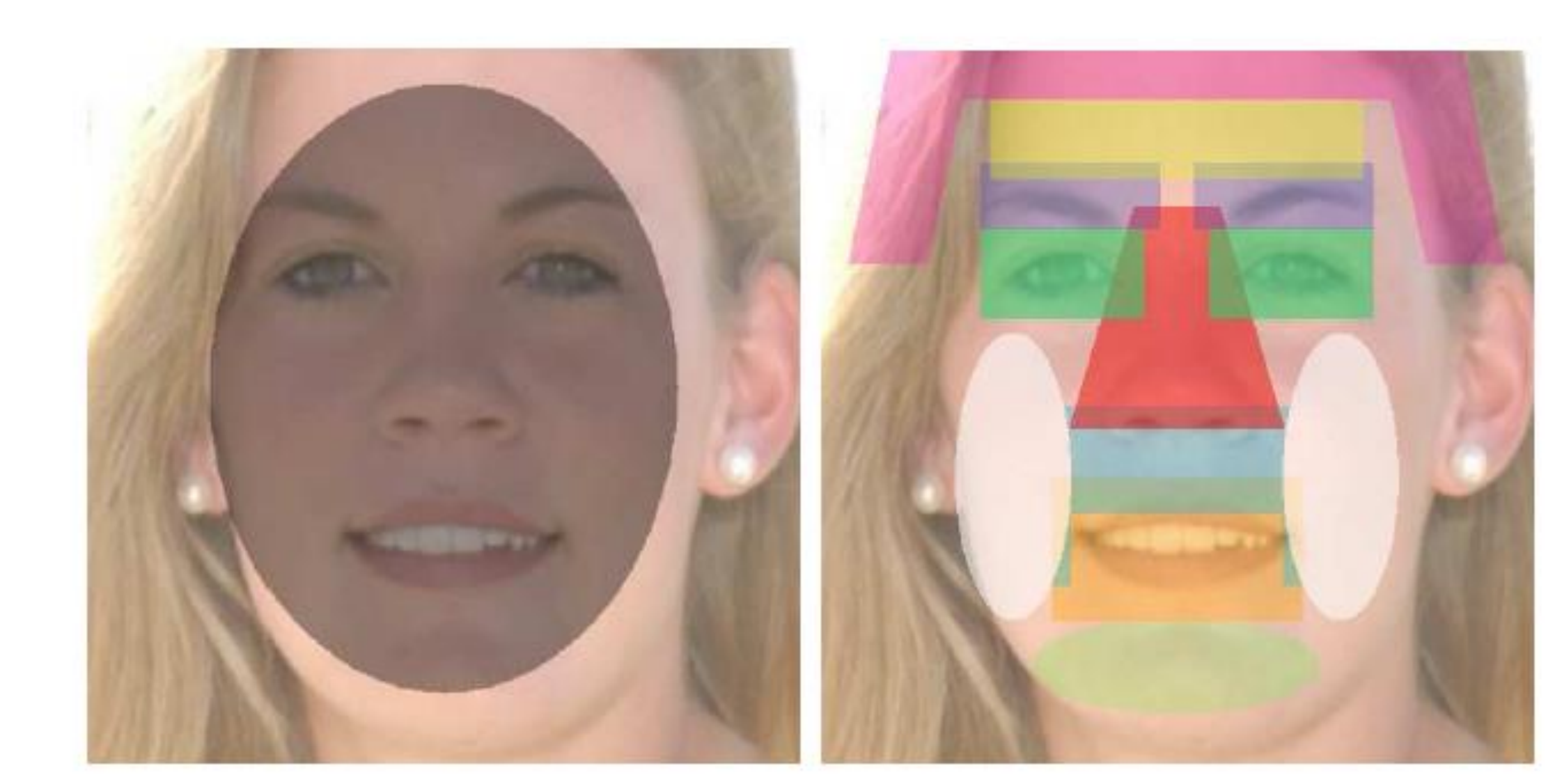
- There are many visual cues to people's identities that cannot be described – at least not concisely.
- We can describe a person's appearance in terms of the similarity of different parts of their face to a limited set of "reference" people.
- Dissimilarities also provide useful information.
- For each reference person in the training set, several simile classifiers are trained for each face region, yielding a large set of total classifiers.



2. Compute visual traits
simile classifier - continues

- The reference people do not appear in benchmarks on which we produce results.
- Train the simile classifiers to recognize similarity to part of a reference person’s face in many images, not similarity to a single image.
- Choose eight regions and six feature types from the set of possible features, and train classifiers for each simile using at most 600 positive example face images of the reference person, and at most 10 times as many negative examples.

2. Compute visual traits
Example



A sample face discovered and split into regions of interest.

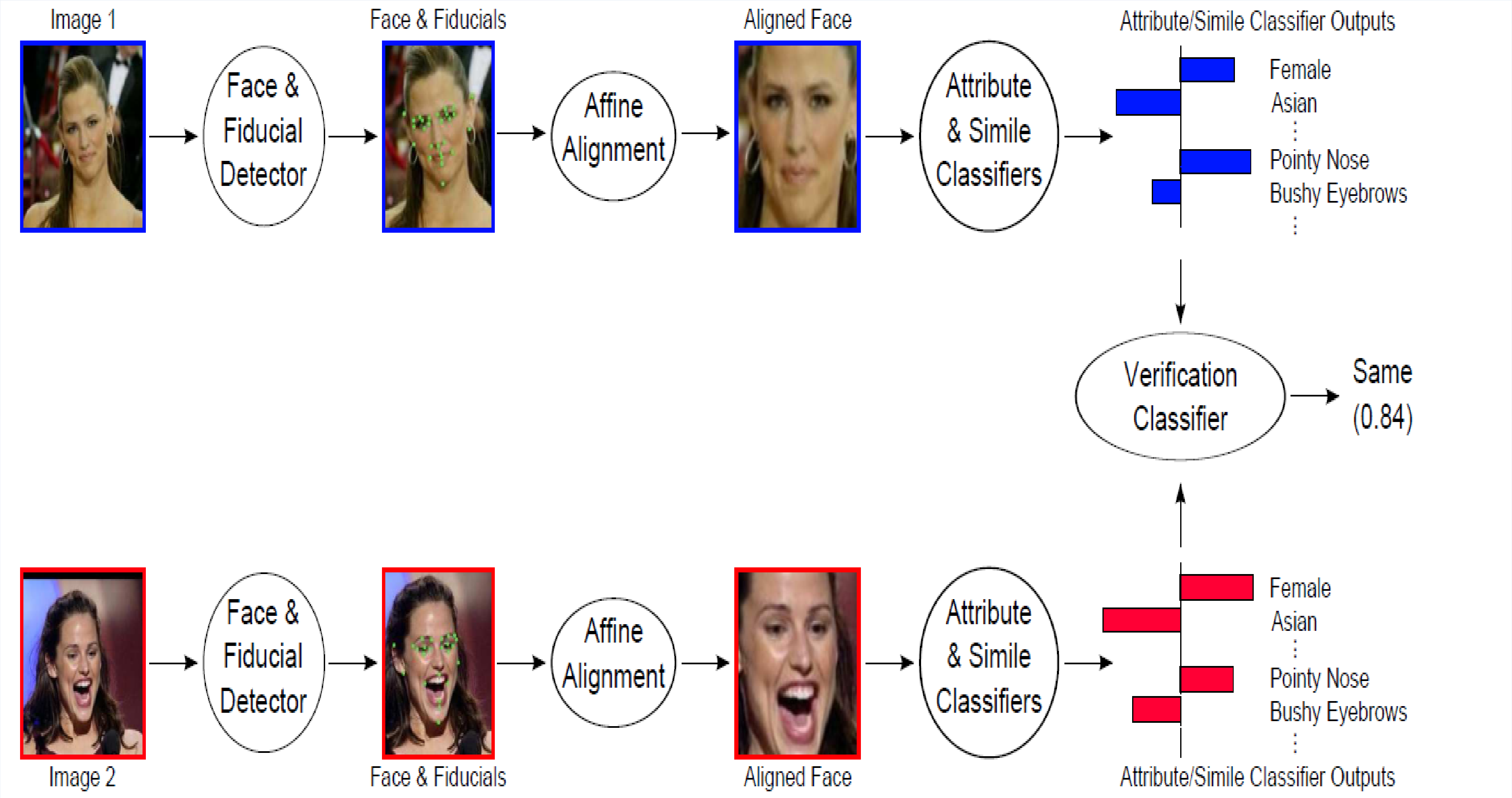
3. Perform verification

- In order to make a decision about whether two face images I_1 and I_2 show the same person, we use the final classifier D to compare the trait vectors $C(I_1)\ and\ C(I_2)$ obtained by one or both of the methods discussed earlier.
- Assuming that:
  - ✓Corresponding values $C(I_1)\ and\ C(I_2)$ from the ith trait classifier should be similar if the images are of the same individual.
  - ✓Trait values are raw outputs of binary classifiers (in the range [-1, 1]), and so the signs of values should be important.
  - ✓The choice of classifier, SVMs, optimize for separating data at the separation boundary, and so differences in values close to 0 are more important than differences between those with greater absolute values.

3. Perform verification - continues

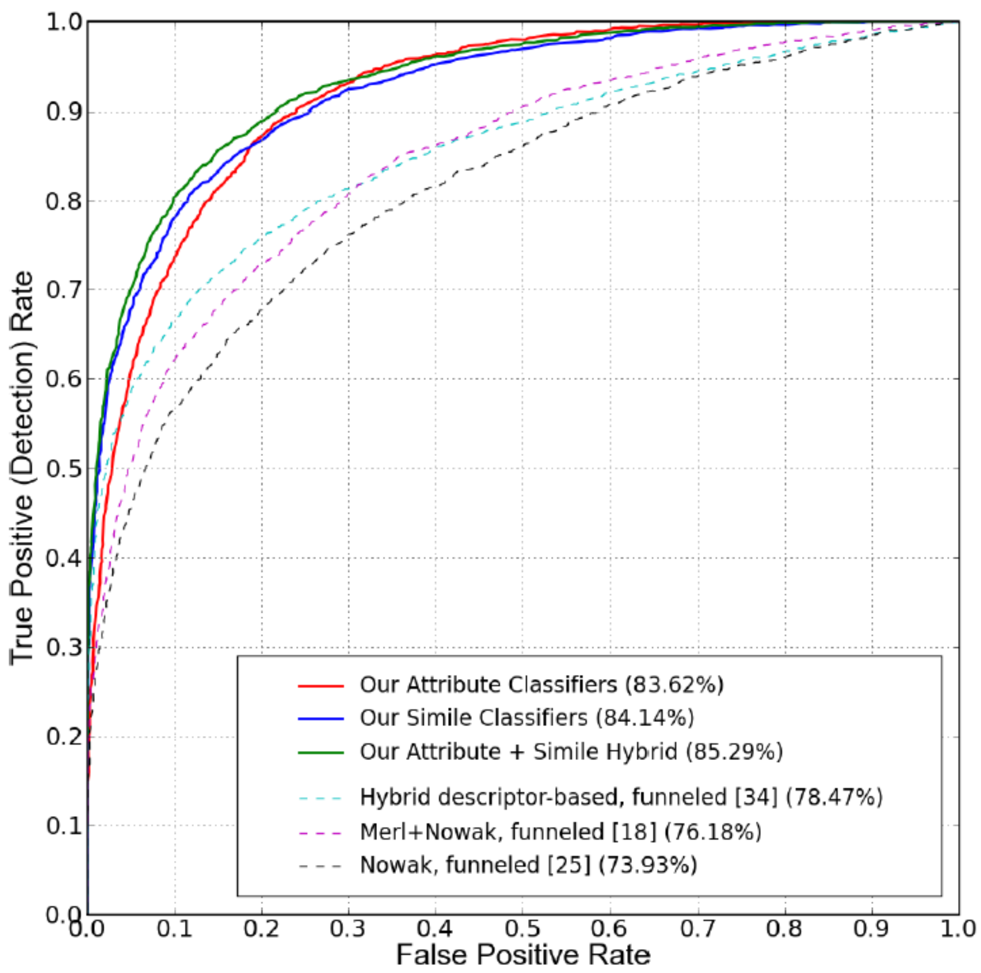
- For each of the n trait classifiers, we compute a pair
  $p_i = (|a_i - b_i|, (a_i \cdot b_i)) \mid a_i = C_i(I_1), b_i = C_i(I_2)$
- These pairs are concatenated to form the 2n dimensional vector that we actually classify:
  $v(I_1, I_2) = D(\{p_1, \ldots, p_n\})$
- Training D requires pairs of positive examples (both images of the same person) and negative examples (images of different people).

Sample of face verification



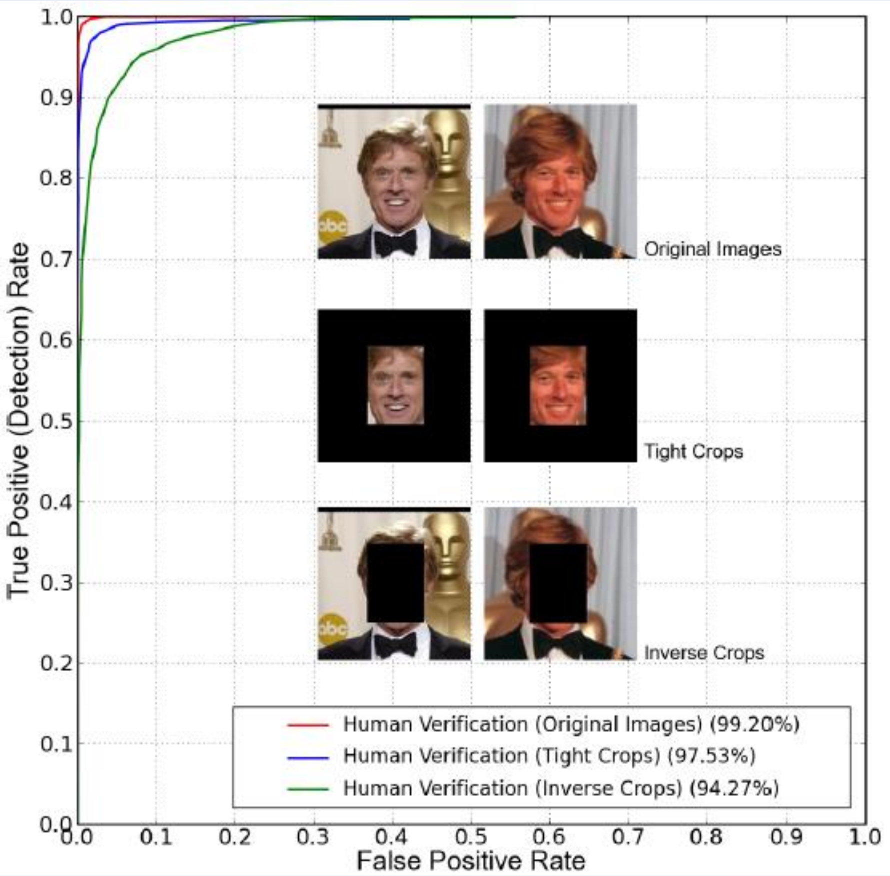
Experiments

- All the experiments evaluate performance on a face verification task, given two images of faces, determine if they show the same individual.
- LFW data set:
  - 13,233 images of 5,749 people.
  - Divided into 2 views:
    - A development set of 2,200 pairs for training and 1000 pairs for testing, used to build models and choose features.
    - 10-fold cross-validation set of 6,000 pairs, where final performance is evaluated on each fold.
  - The chart shows the results, comparing with several previous methods.
- The highest performance is with the hybrid method, which achieves a 31.68% drop in error rates from the previous state-of-the-art.



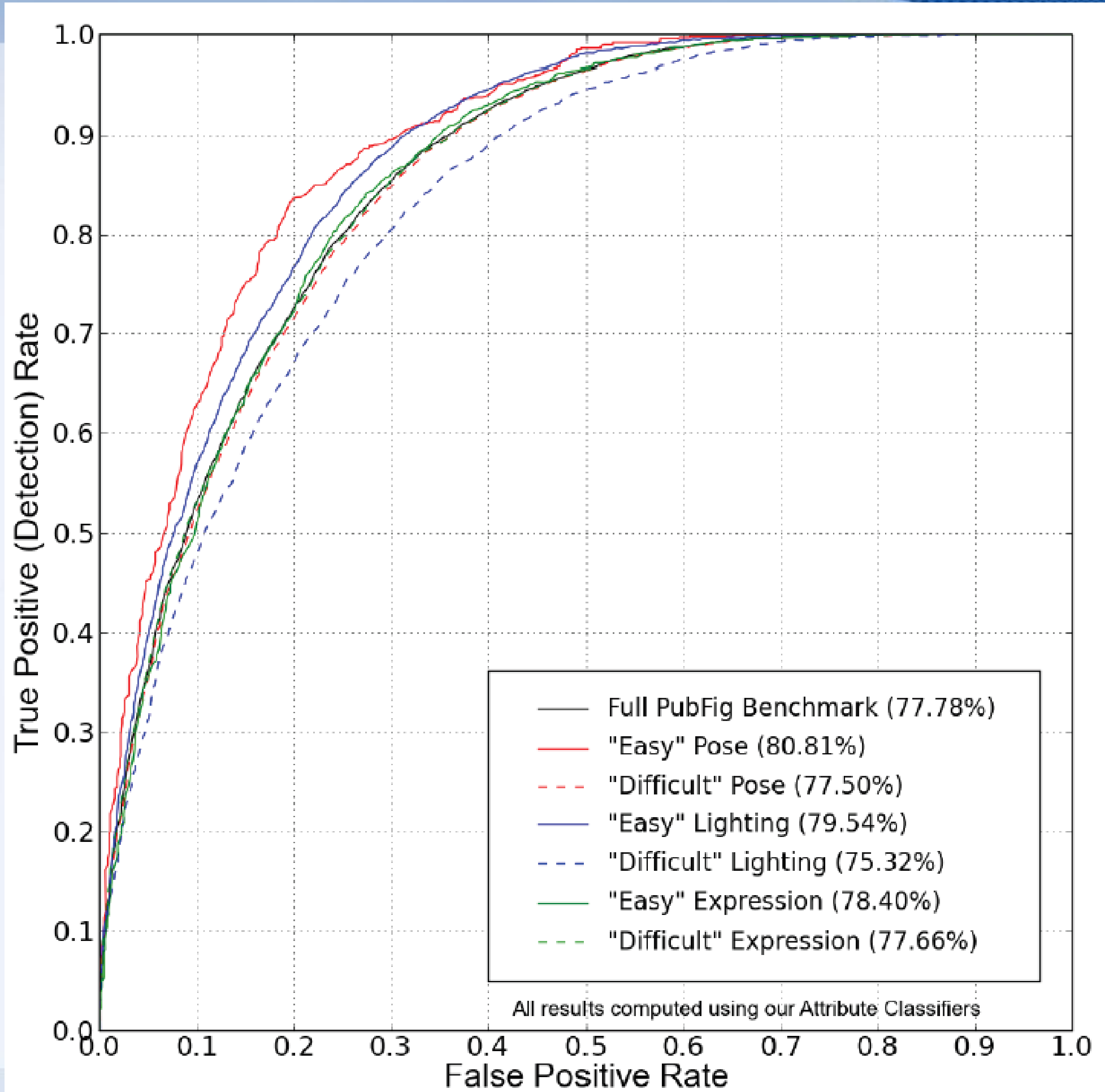
Experiments - continues

- Human Performance on LFW
  - The “same” task was given to humans. (Amazon Mechanical Turk)
  - Their performance was almost perfect (99.2%).
  - Two additional tasks where given:
    - Showing only cropped image of the face (97.53%).
    - Showing all the rest of the image but the face (surprisingly 94.27%).
- PubFig data set:
  - New data set created by the authors, consists 60,000 images of 200 people.
  - Allows constructing large subsets of the data across different poses, lighting conditions, and expressions.



Experiments - continues

Face Verification Results on PubFig:

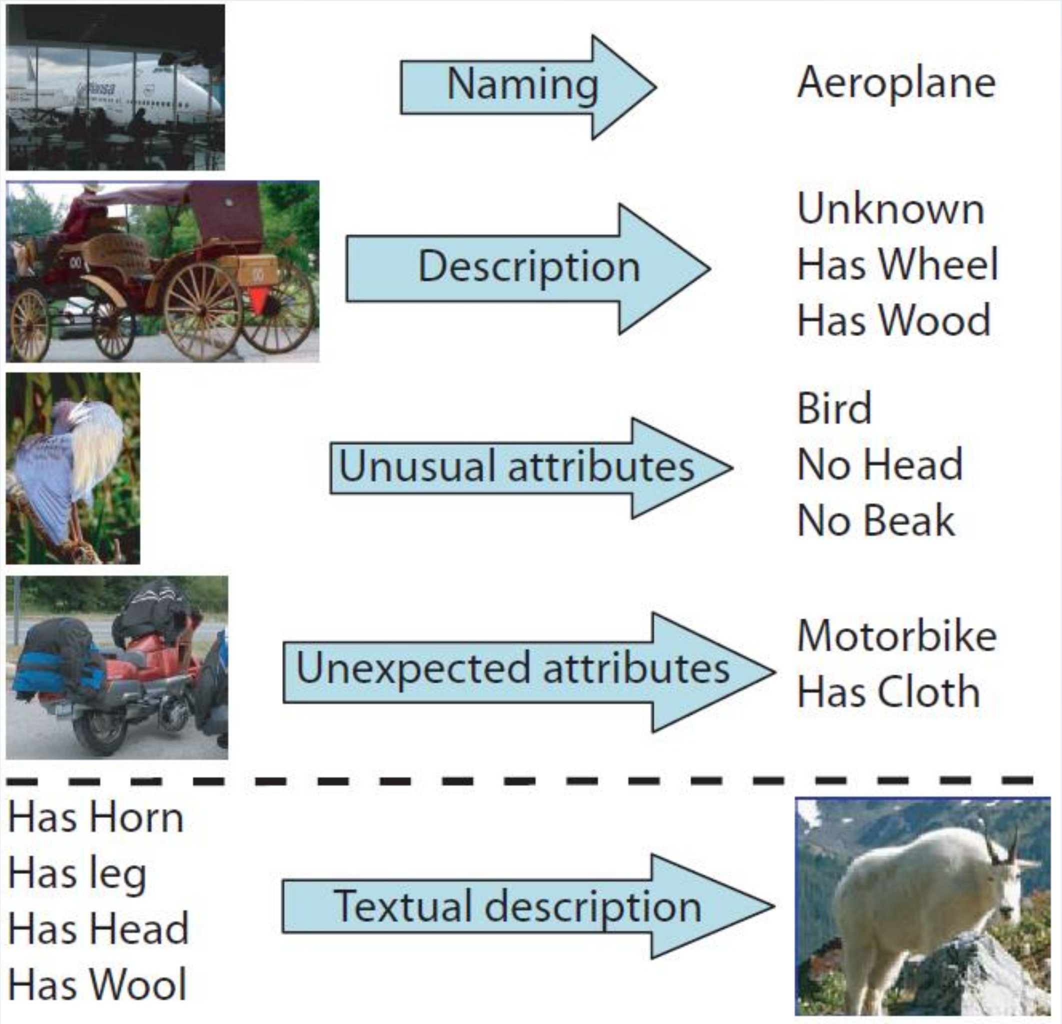


Outline

- ✓Introduction
- ✓Attribute and Simile Classifiers for Face Verification
- ➢**Describing Objects by Their Attributes**
    - ➢**Recognition steps**
    - ➢**Attributes**
    - ➢**Features selection**
    - ➢**Experiments**
- Conclusion
- References

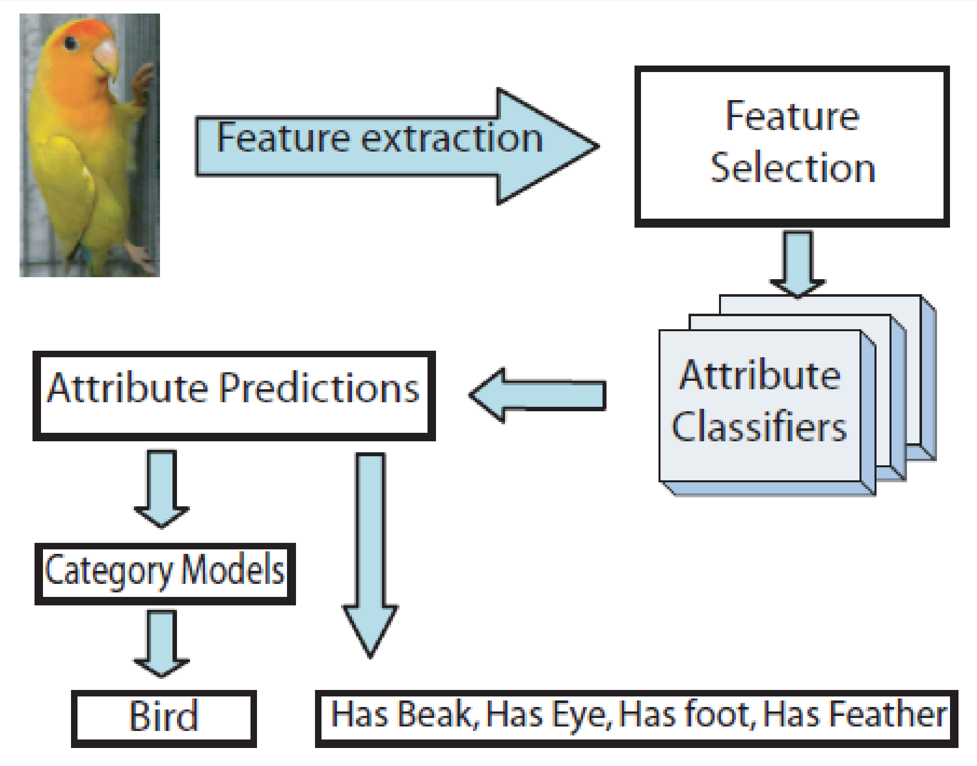
Reminder
Describing Objects by their Attributes

- Rather than focusing on identity assignment, inferring attributes will be the core problem of recognition.
- Shifting the goal of recognition from naming to describing allows:
  - Naming objects
  - Reporting unusual aspects of a familiar object
  - Saying something about unknown objects, not just “unknown”
  - learning to recognize new objects with few examples or textual description



Recognition steps

1. Feature extraction
2. Feature selection
3. Attribute classifiers prediction
4. Result:
   - ✓Naming
   - ✓Reporting additional description/info
   - ✓learn new categories from very few examples



Base features

4 types of bag of words style base features:

- Texture – good for materials

  Computed for each pixel and quantized to the nearest 256 kmeans, extracted with a texton filterbank.

- Color – good for materials

  Sampled for each pixel, and quantized to the nearest 128 kmeans centers.

- visual words – good for parts

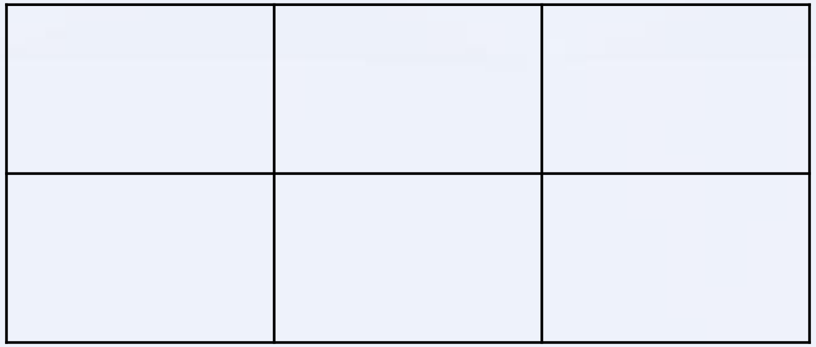
  Constructed with an HOG spatial pyramid, the descriptors are quantized to 1000 kmeans centers.

- Edges – good for shapes

  found using a standard canny edge detector and their orientations are quantized into 8 unsigned bins.

Base features

- Local texture, HOG, edge, and color descriptors inside the bounding box are binned into individual histograms.
- Generate histograms for each feature type for each cell in a grid of three vertical and two horizontal blocks.
- These seven histograms are stacked together resulting in a 9751 dimensional feature, which we refer to as the base features.



Attributes

- Semantic attributes - describes parts (“has nose”), shape (“cylindrical”), and materials (“furry”).
  - Can be learned from annotations and allows describing objects and identifying them based on textual descriptions.
  - But, both cats and dogs can share all available semantic attributes.
- Discriminative attributes - take the form of comparisons, “cats and dogs have it but sheep and horses don’t”.
  - Each comparison splits a portion of the data into two partitions.
  - Each split is further defined by a subset of base features, such as texture or color, to use for learning.
  - Then using linear SVM, learn tens of thousands of these splits and pick those that can be well predicted.

Across Category Generalization by Within Category Prediction

- When learning the attributes, we want to be able to generalize to new types of objects.
- Training attribute classifiers in the traditional way (use all features) leads to poor generalization for some attributes across categories.
- Instead, select features that can predict attributes within an object class and use only those to train the attribute classifier, subset of features that can generalize well.
- Conventional feature selection criteria will not apply to our problem because they are still confused by semantically irrelevant correlations.
- Therefore, we'll use novel feature selection criterion that de-correlates attribute predictions.

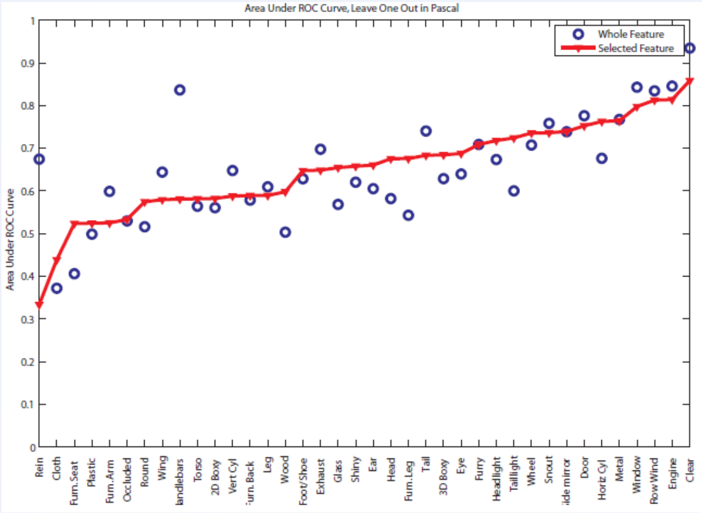
Features selection -
Selection procedure

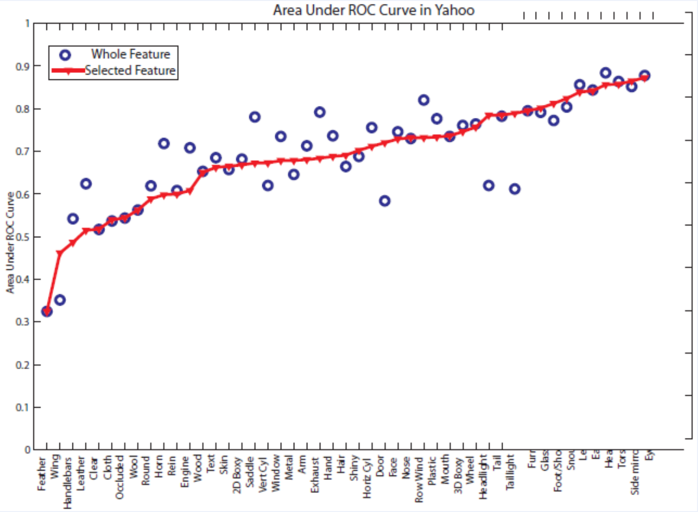
- For example, to learn a “wheel” classifier:
  1. Select features that perform well at distinguishing examples of cars with wheels and cars without wheels.
  2. Use the same procedure to separating motorbikes, buses and trains with and without wheels.
  3. Learn the “wheel” classifier over all classes using those selected features.

The subset selected from the “whole features” by this procedure is called “selected features”.

Experiments – describing objects

- Assigning attributes to objects:
  - predicting attributes for the within category protocol with the area under the curve of 0.834.
  - The chart shows that predicting attributes for across category protocols are fairly reliably.





Experiments – describing objects – cont.

Experiments – describing objects – cont.

- Unusual attributes of object:
  - If a reliable attribute classifier predicts one of these typical attributes is absent, we report that it is not visible in the image, 68.2% of the reported attributes are correct.
  - On the other hand, if a reliable attribute classifier predicts an attribute which is not expected to be in the predicted class, we can report that too, and try to localize it.

Experiments – describing objects – cont.

Reporting unusual attributes.

Localizing unusual attributes.

Experiments – naming objects

- Naming familiar objects:
  - this table indicates the method's performance of the naming task with different classifiers.

PASCAL 08	Base Features	Whole Features		Selected Features	
		Sem. Attr.	All Attr.	Sem. Attr.	All Attr.
SVM	58.5 (35.5)	56.1 (34.3)	58.3 (38.1)	54.6 (28.4)	**59.4** (37.7)
Logistic Regression	54.6 (36.9)	51.2 (31.4)	53.4 (33.5)	51.8 (32.3)	53.5 (35.1)

- Learning to identify new objects:
  - If we recognize classes in a-Yahoo set using attribute classifiers trained on a-Pascal, we get an overall accuracy of 69.8%.
  - If we train attributes on a-Yahoo as well, we get an overall accuracy of 74.7%, comparing to 72.7% using base features.

Experiments – naming objects

- Learning new categories from textual description:
  - For example, can learn new categories by describing new classes to the algorithm as this new class is “furry”, “four legged”, “has snout” and “has head”.
  - Could get an accuracy of 32.5%.
- Rejection:
  - When presented with an object from a new category, we want the model to recognize that it doesn’t belong to any known category.
  - By rejection using confidences of one-vs.-all SVM’s used to learn a-Pascal object models, we get chance performance (0.5). However, by using attributes we reject significantly better (0.6).

Outline

- ✓Introduction
- ✓Attribute and Simile Classifiers for Face Verification
- ✓Describing Objects by Their Attributes
- ➢**Conclusion**
- References

Conclusion

- We reviewed two approaches for face verification using traits computed on face images:
  - Based on describable attributes and the novel simile classifiers.
  - Both approaches result in error rates significantly lower (23.92% to 31.68%) than the state-of-the-art for face verification on the LFW data set.
- Also, we reviewed another similar approach, for recognizing objects:
  - Shifts the goal of recognition from naming to describing, by doing so, it allows not only to naming familiar objects, but also, reporting additional info about objects.
  - Learning attributes presents a major new challenge, generalization across object categories, not just across instances within a category.
  - Introduce a novel feature selection method for learning attributes that generalize well across categories.

Outline

- ✓Introduction
- ✓Attribute and Simile Classifiers for Face Verification
- ✓Describing Objects by Their Attributes
- ✓Conclusion
- ➢**References**

References

- Describing Objects by their Attributes; Ali Farhadi, Ian Endres, Derek Hoiem, David Forsyth; Computer Science Department, University of Illinois at Urbana-Champaign.
- Attribute and Simile Classifiers for Face Verification; Neeraj Kumar Alexander C. Berg Peter N. Belhumeur Shree K. Nayar; Columbia University.
- Describable Visual Attributes for Face Verification and Image Search; Presentation by Gregory Teodoro.