# Classifying Images with Visual/Textual Cues

By Steven Kappes and Yan Cao

# Motivation

- Image search
- Building large sets of classified images
- Robotics

# Background

- Object recognition is unsolved
- Deformable shaped matching
- Classification based on purely textual information
- SVMs, PCA

Google monkey Search Images Search the Web Advanced Image Search Preferences

Moderate SafeSearch is on

New! Google Image Labeler

**Images** Showing: All image sizes

Results **61** - **80** of about **8,260,000** for **monkey**. (**0.07** seconds)

« View all web results for **monkey**


got **monkey**? baseball hat
532 x 800 - 271k - jpg
www.monkeygoods.com


Flying with the **Monkeys**
1500 x 2119 - 705k - jpg
www.monkey-bum.com

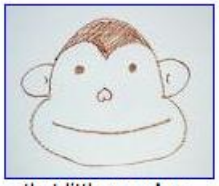

Flying with the **Monkeys**
354 x 500 - 33k - jpg
www.monkey-bum.com

Puki the Baby **Monkey**
400 x 400 - 32k - jpg
www.thegreenhead.com

The Hundredth **Monkey**
370 x 490 - 92k - jpg
pure-research.net

Indeed, the **monkeys'** feces revolts ...
491 x 342 - 21k - jpg
users.cihost.com

**Monkey's** Audio configured with ...
626 x 500 - 9k - png
wiki.hydrogenaudio.org



that little **monkey**.
482 x 398 - 18k - jpg
infinity-simplified.blogspot.com

Funky **Monkey** Buttons
380 x 372 - 46k - jpg
www.istockphoto.com

enjoy the **monkey**, be entertained by ...
698 x 599 - 276k - gif
michaelontheair.com

YAHOO! SEARCH

Web | **Images** | Video | Local | Shopping | **more »**

monkey Search

Advanced Search
Preferences

SafeSearch is **on** (turn off)

**Image Results** 61 - 80 of about 2,261,773 for **monkey** - 0.04 sec.

**Show: All** | Wallpaper - Large - Medium - Small | Color - Black & White

Also try: cartoon **monkey**, family guy evil **monkey** More...

**monkey** loves the car .
By delibelli on Flickr
www.flickr.com

**monkey** magic from defacto .
By What What on Flickr
www.flickr.com

**monkey** eating mango in belize
By princessangel on Flickr
www.flickr.com

**monkey** socks from knitty .
By mommymoshpit on Flickr
www.flickr.com

cute **monkey** cupcakes
By Yano on Flickr
www.flickr.com

yep a **monkey** with...
By Glass Slipper... on Flickr
www.flickr.com

planet **monkey** mia eye
By willster on Flickr
www.flickr.com

marmoset **monkey**
By tr33lo on Flickr
www.flickr.com

squirrel **monkey** at everland zoo
By floridapfe on Flickr
www.flickr.com

**monkey** made using slightly modified
By Angry Angel on Flickr
www.flickr.com

# Image and Textual Feature Based Matching

- Goal: Large dataset of labelled images of animals found online
- Animals are difficult to classify

# Dataset

- Web pages from Google text search on 10 different animals used as the data set
- Also gathered data based on searches related to monkeys

# Training

- Use LDA to discover highly likely words for each topic
- Rank images based on their word likelihoods to get a set of 30 exemplars for each topic
- Supervision

# Latent Dirichlet Allocation

- Nearby words are likely to be relevant to the image
- Probabilistic generative model to find 10 latent topics for each category
- Discover 50 highly likely words for each topic

# LDA cont.

- Assign each image to to it’s most likely topic
- Select the top 30 images for each topic as exemplars

# Supervision

- Each topic is labelled as relevant or background
- Topics merged to form a single relevant group and background group
- Optional: Allow the user to swap incorrectly labelled exemplars

# Testing

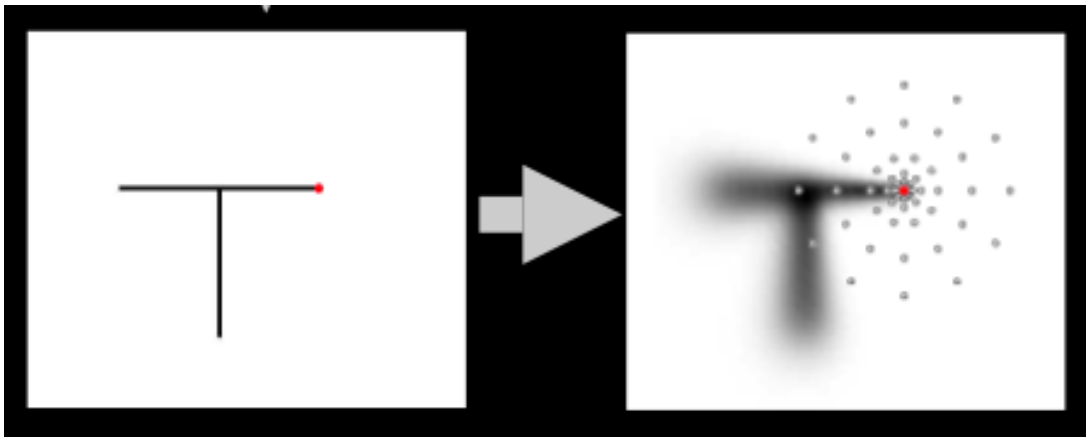
- Independent voting based on textual information and 3 image types: shape, color, texture
- Image feature similarity is computed by nearest neighbour comparing to features from positive and negative exemplar groups
- Compute the sum of the similarities of the image features matching the positive group

# Textual

- Sum the likelihood the image belongs to the relevant topics of a category
- Normalize based on the maximal score over all images

# Shape - Geometric Blur

- Local appearance descriptor
- Apply spatially varying blur
- Robust to small affine distortions



# Shape

- Compare geometric blur features
- Ensure there is a high similarity
- Gating of geometric blur features: ensure the local color matches

# Color

- Subdivide the image into 9 regions
- Compute a normalized color histogram with 8 bins per color channel
- Compute color histograms around geometric blur features with a radius of 30 pixels

# Texture

- Compute histograms of the output many different filters

# Voting

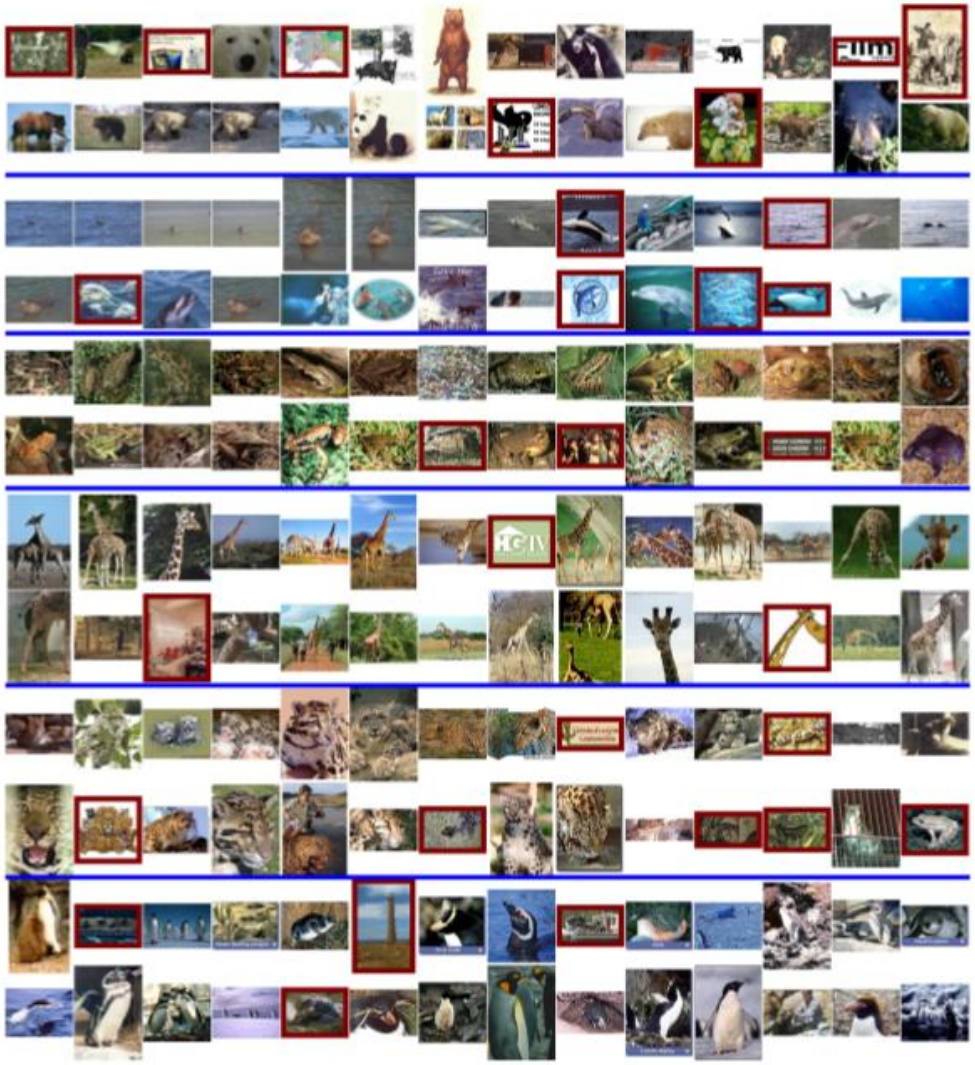
- Linear combination of the 4 voting features
- All votes have equal weight

# Supervision

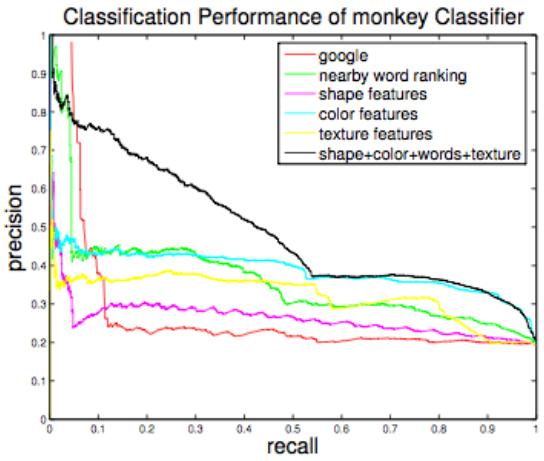
- Supervision helps accuracy for smaller categories
- Excluding exemplars selected by LDA can result in worse accuracy

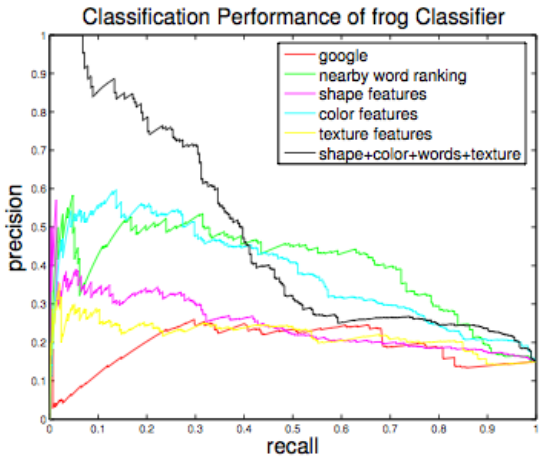
# Results

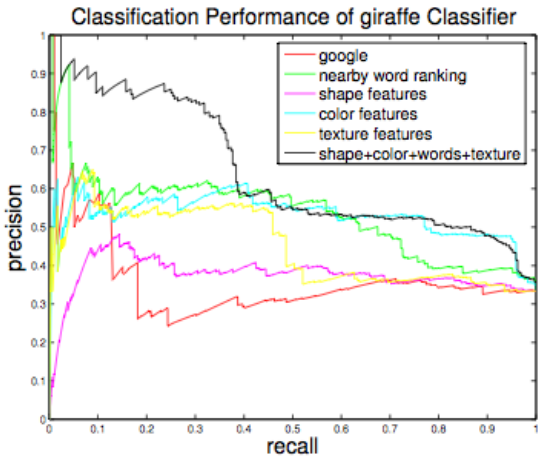
- Always more accurate than google image search
- False positives are often reasonable
- Image features greatly improves accuracy over purely textual classification
- Multiple features help recognize a wide range of images



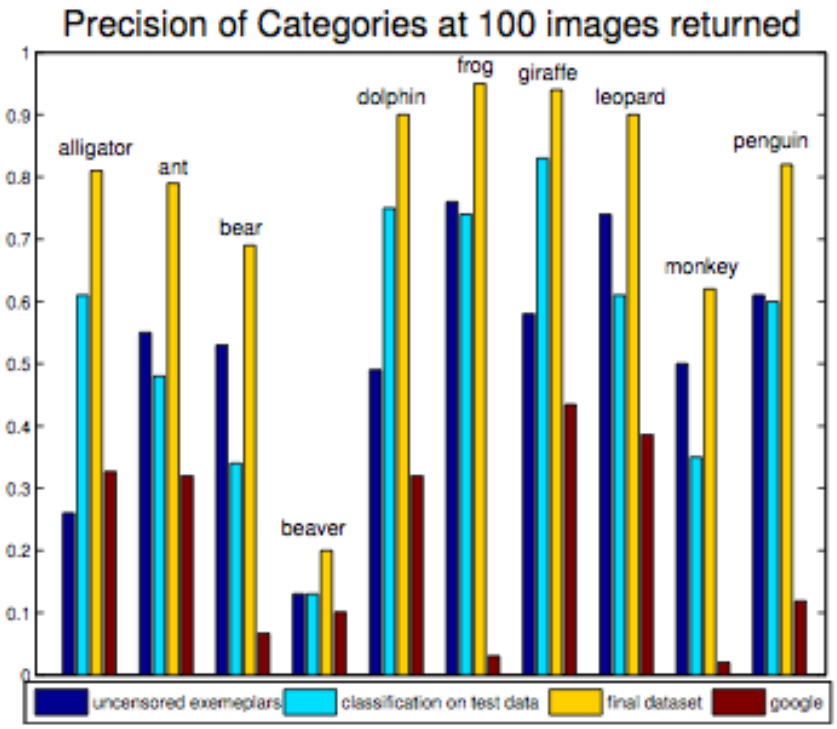
# Voting Accuracy

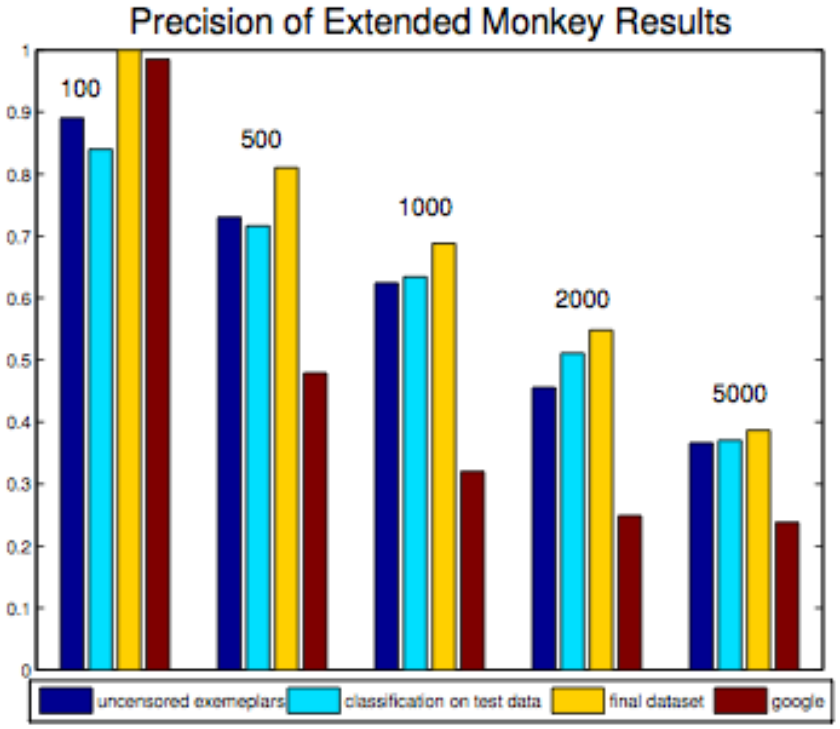


# Accuracy


Precision of Categories at 100 images returned
alligator
ant
bear
beaver
dolphin
frog
giraffe
leopard
monkey
penguin
uncensored exemeplars
classification on test data
final dataset
google


Precision of Extended Monkey Results
100
500
1000
2000
5000
uncensored exemeplars
classification on test data
final dataset
google

# Limitations

- Requires light supervision
- Based on textual information, only can be applied to certain situations when there is associated text

# Names and Faces

- Goal: Given an input image and an associated caption, detect the face(s) in the image and label it with the correct name(s) detected in the caption
- Motivation: Build a rich, reasonably accurate collection of labeled faces

# Names and Faces


**President George** W. Bush makes a statement in the Rose Garden while Secretary of **Defense Donald Rumsfeld** looks on, July 23, 2003. Rumsfeld said the United States would release graphic photographs of the dead sons of **Saddam Hussein** to prove they were killed by American troops. Photo by Larry Downing/Reuters


British director **Sam Mendes** and his partner actress **Kate Winslet** arrive at the London premiere of 'The Road to Perdition', September 18, 2002. The films stars **Tom Hanks** as a Chicago hit man who has a separate family life and co-stars **Paul Newman** and Jude Law. REUTERS/Dan Chung


World number one **Lleyton Hewitt** of Australia hits a return to **Nicolas Massu** of Chile at the Japan Open tennis championships in Tokyo October 3, 2002. REUTERS/Eriko Sugita


German supermodel **Claudia Schiffer** gave birth to a baby boy by Caesarian section January 30, 2003, her spokeswoman said. The baby is the first child for both Schiffer, 32, and her husband, British film producer **Matthew Vaughn**, who was at her side for the birth. Schiffer is seen on the German television show 'Bet It...?!' ('Wetten Dass...?!') in Braunschweig, on January 26, 2002. (Alexandra Winkler/Reuters)

# Names and Faces

- Names: Extract names found in the captions; Identify two or more capitalized words followed by a present tense verb
- Faces: “Face Detector”; Rectification
  - Use SVM to detect 5 feature points on the face
  - Do affine transformation

# Face Representation

- All faces are resized to 86*86 pixels
- RGB values from each face are concatenated into a long vector
- Vectors in a space where same faces are close and different faces are far apart

# Background

- Kernel PCA (Principal Component Analysis)
  - Discard components that are similar for all faces to reduce dimensions
  - In the coordinate system set up by the basic principal components, the images have the widest distribution
- LDA (Linear Discrimination Analysis)
  - Provide a linear space that best separates different faces for discrimination

# Kernel PCA in Names and Faces

- Compute a kernel matrix, K
- Kij = value of kernel function (Gaussian kernel here) comparing image i and image j
- Due to the huge image set, NxN Kernel Matrix will have about 2*10^9 elements
- Nystrom Approximation is used to calculate the eigenvectors of K. C is estimated by Ĉ

$$K = \begin{bmatrix} A & B \\ B^T & C \end{bmatrix} \Rightarrow \hat{K} = \begin{bmatrix} A & B \\ B^T & \hat{C} \end{bmatrix} \quad \hat{C} = B^T A^{-1} B$$

# LDA in Names and Faces

- After applying kernel PCA, the dimensions of the data points (here are the faces) are sharply reduced
- The size of covariance matrix for the reduced input vectors is not huge
- Project all images into a linear space where different faces are separated best

# Modified k-Means

- Step 1. Randomly assign each face to a name
- Step 2. for the faces in each name, calculate the means of the image vectors
- Step 3. Reassign each image to the name whose vector means is closest to it.
- Repeat step 2 and 3 to convergence

# Prune and Merge

- Remove clusters with fewer than three
- Remove points with low likelihood to get low error rates
  - Likelihood=P(face is from assigned cluster)/P(face is not from assigned cluster)
- Merge clusters with small distances between their means

# Example

# Evaluation

- Fairly good assignment of names to faces using simple models for images, names

| #Images | #Clusters | error rate |
|---|---|---|
| 19355 | 2357 | 26% |
| 7901 | 1510 | 11% |
| 4545 | 765 | 5.2% |
| 3920 | 725 | 7.5% |
| 2417 | 328 | 6.6% |

# Limitations

- Random assignment of faces in k-means
- Use of RGB pixel values to discriminate between faces of different people

# Bag-of-features for scene categorization

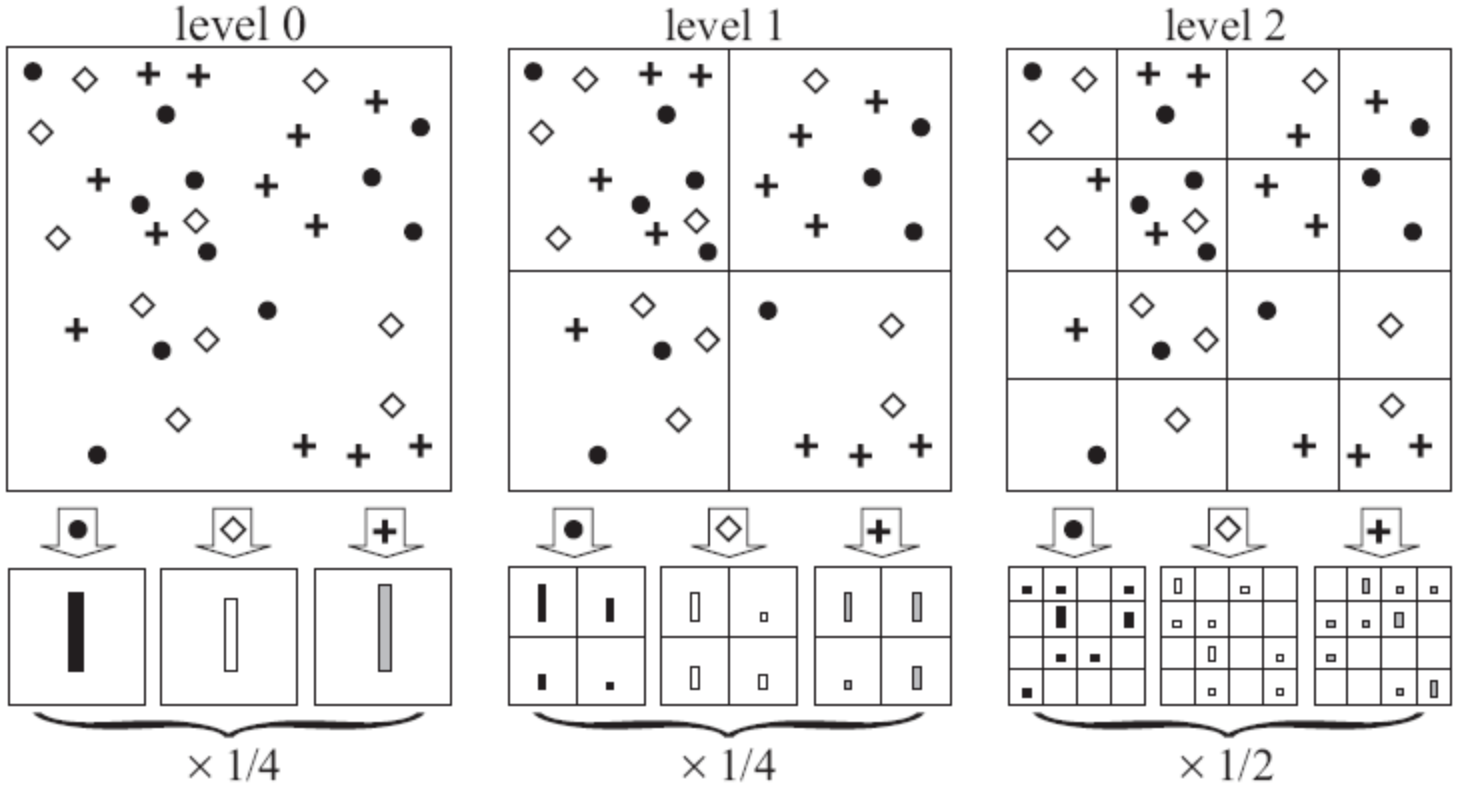
- Bag-of-features method represents an image as an orderless collection of local features
- Disregards all information about the spatial layout of the features
- Incapable of capturing shape or of segmenting an object from its background

# Spatial Pyramid Matching (review)

- Compute rough geometric correspondence on a global scale
- Repeatedly subdivide the image and compute histograms of local features at increasingly fine resolutions.
- It is a “Locally orderless matching” method which achieves good performance in estimating overall perceptual similarity between images

# Spatial Pyramid Matching (review)

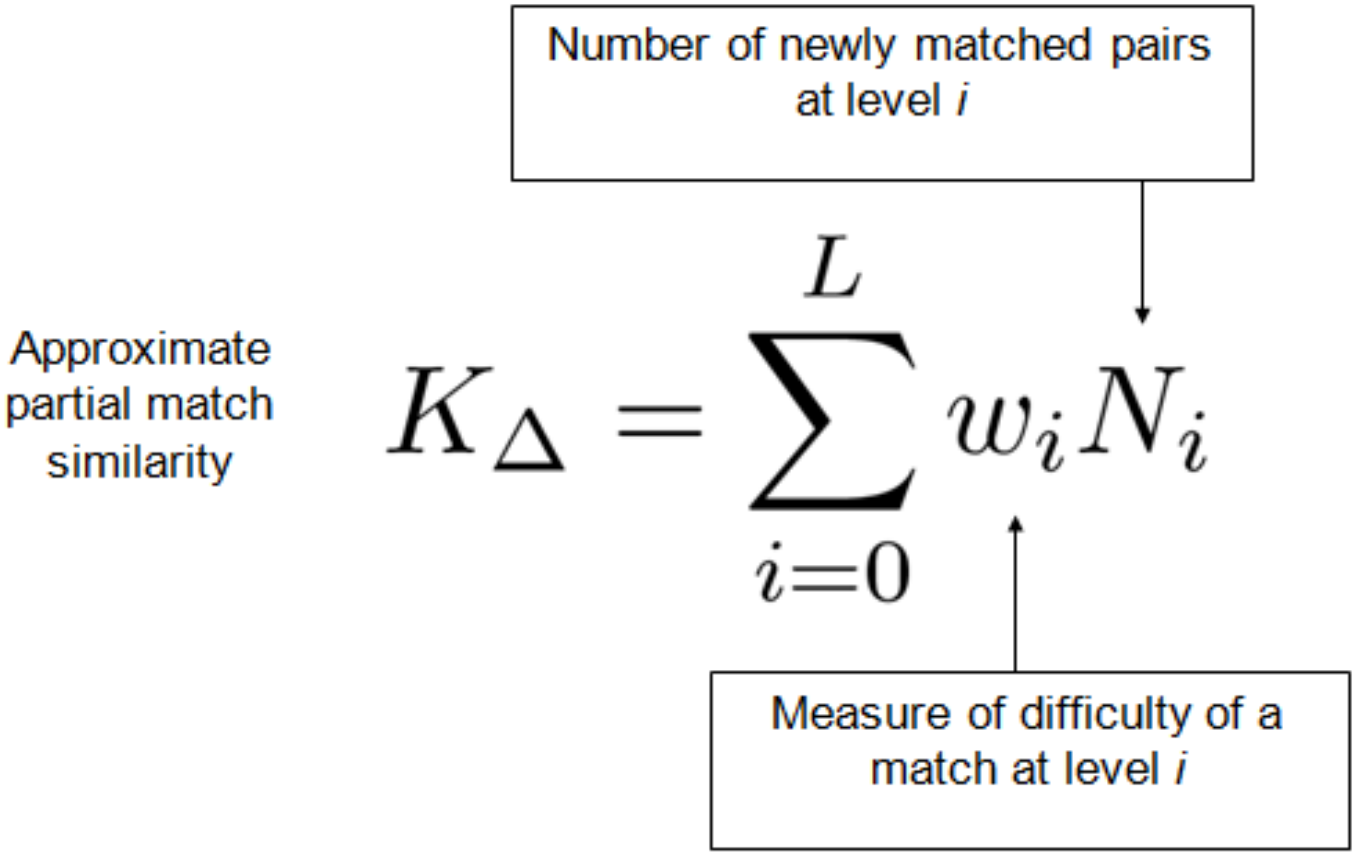
Example



# Pyramid Matching Mechanism (review)

- Place a sequence of increasingly coarser grids over the feature space
- Takes a weighted sum of the number of matches that occur at each level of resolution
- Two points are said to match if they fall into the same cell of the grid
- matches found at finer resolutions are weighted more highly than matches found at coarser resolutions

# Equation for Kernel



# Spatial Pyramid Matching in Labeling Images (Preparation stage)

- Change images to gray scale
- Feature Extraction
  - Feature as a dense regular grid instead of interest points
  - SIFT descriptors of 16*×16 pixel patches*
- Set up vocabulary (classify features)
  - k-Means is to merge similar features
  - k=200, k=400

# Spatial Pyramid Matching in Labeling Images (Matching)

- Each type of feature as a channel
- For each channel, apply Pyramid Matching to get kernel values of each pair of images
- Sum of the kernel values between image X and Y in all channels to get the final kernel between X and Y

# Spatial Pyramid Matching in Labeling Images (Discriminating)

- SVM. One-versus-all rule: a classifier is learned to separate each class from the rest, and a test image is assigned the label of the classifier with the highest response.

# Spatial Pyramid Matching in Labeling Images

office

kitchen

living room

bedroom

store

industrial

tall building*

inside city*

street*

highway*

coast*

open country*

mountain*

forest*

suburb

# Spatial Pyramid Matching in Labeling Images Classification Result

The authors' dataset

| | Weak features ($M = 16$) | | Strong features ($M = 200$) | | Strong features ($M = 400$) | |
|---|---|---|---|---|---|---|
| $L$ | Single-level | Pyramid | Single-level | Pyramid | Single-level | Pyramid |
| 0 ($1 \times 1$) | 45.3 ±0.5 | | 72.2 ±0.6 | | 74.8 ±0.3 | |
| 1 ($2 \times 2$) | 53.6 ±0.3 | 56.2 ±0.6 | 77.9 ±0.6 | 79.0 ±0.5 | 78.8 ±0.4 | 80.1 ±0.5 |
| 2 ($4 \times 4$) | 61.7 ±0.6 | 64.7 ±0.7 | 79.4 ±0.3 | **81.1** ±0.3 | 79.7 ±0.5 | **81.4** ±0.5 |
| 3 ($8 \times 8$) | 63.3 ±0.8 | **66.8** ±0.6 | 77.2 ±0.4 | 80.7 ±0.3 | 77.2 ±0.5 | 81.1 ±0.6 |

Caltech 101 dataset

| | Weak features | | Strong features (200) | |
|---|---|---|---|---|
| $L$ | Single-level | Pyramid | Single-level | Pyramid |
| 0 | 15.5 ±0.9 | | 41.2 ±1.2 | |
| 1 | 31.4 ±1.2 | 32.8 ±1.3 | 55.9 ±0.9 | 57.0 ±0.8 |
| 2 | 47.2 ±1.1 | 49.3 ±1.4 | 63.6 ±0.9 | **64.6** ±0.8 |
| 3 | 52.2 ±0.8 | **54.0** ±1.1 | 60.3 ±0.9 | 64.6 ±0.7 |

# Spatial Pyramid Matching in Labeling Images Classification Result (continued)

The Graz dataset

| Class | $L = 0$ | $L = 2$ | Opelt [14] | Zhang [25] |
|---|---|---|---|---|
| Bikes | 82.4 ±2.0 | 86.3 ±2.5 | 86.5 | 92.0 |
| People | 79.5 ±2.3 | 82.3 ±3.1 | 80.8 | 88.0 |

# Discussion

- Simple
  - Using global cues as indirect evidence about the presence of an object
  - Explicit object models are not necessary
- Accurate
  - Achieve improvement over “bag-of-features” image presentation

# Applications

- Stand-alone Scene categorizer

- “Context” modular integrated in larger object recognization system