Visual Categorization With Bags of Keypoints. ECCV, 2004.

G. Csurka, C. Bray, C. Dance, and L. Fan.

Shilpa Gulati
2/15/2007

Basic Problem Addressed

- Find a method for Generic Visual Categorization
  - Visual Categorization: Identifying whether objects of one or more types are present in an image.
  - Generic: Method generalizes to new object types. Invariant to scale, rotation, affine transformation, lighting changes, occlusion, intra-class variations etc.

Main Idea

- Applying the bag-of-keywords approach for text categorization to visual categorization.
- Constructing vocabulary of feature vectors from clustered descriptors of images.

The Approach I: Training

- Extract interest points from a dataset of training images and attach descriptors to them.
- Cluster the keypoints and construct a **set of** vocabularies (Why a set? Next slide).
- Train a multi-class qualifier using bags-of-keypoints around the cluster centers.

Why a set of vocabularies?

- The approach is motivated by text categorization (spam filtering for example).
  - For text, the keywords have a clear meaning (Lottery! Deal! Affine Invariance). Hence finding a vocabulary is easy.
  - For images, keypoints don't necessarily have repeatable meanings.
  - Hence find a set, then experiment and find the **best vocabulary and classifier**.

The Approach II: Testing

- Given a new image, get its keypoint descriptors.
- Label each keypoint with its closest cluster center in feature space.
- Categorize the objects using the multi-class classifier learnt earlier:
  - Naïve Bayes
  - Support Vector Machines (SVMs)

Feature Extraction and Description

- From a database of images:
  - Extract interest points using **Harris affine detector**.
    - It was shown in Mikolajczyk and Schmid (2002) that scale invariant interest point detectors are not sufficient to handle affine transformations.
  - Attach **SIFT descriptors** to the interest points. A SIFT description is 128 dimension vector.
    - SIFT descriptors were found to be best for matching in Mikolajczyk and Schmid (2003).

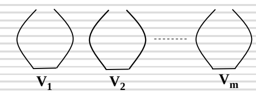
7

Visual Vocabulary Construction

- Use a *k*-means clustering algorithm to form a set of clusters of feature vectors.
- The feature vectors associated with the cluster centers (V_1..V_m) form a vocabulary.
- Find multiple sets of clusters using different values of *k*.

Vocabulary is
V = {V_1, V_2.., V_m}

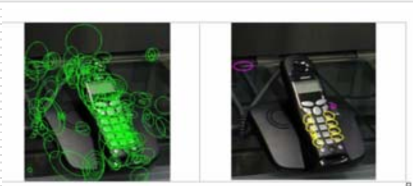
Construct multiple vocabularies.



Slide inspired by [3]

8

Clustering Example



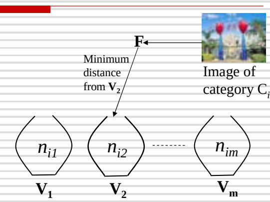
All features　　Clusters

Image taken from [2]

9

Categorization by Naïve Bayes I: Training

- Extract keypoint descriptors from a set of labeled images.
- Put the descriptor in the cluster or "bag" with minimum distance from cluster center.
- Count the number of keypoints in each bag.



n_{ij} is the total number of times a feature "near" V_j occurs in training images of category *i*

If a feature in image *I* is nearest to cluster center *Vj*, we say that keypoint *j* has **occurred** in image *I*

Slide inspired by [3]

10

Categorization by Naïve Bayes II: Training

- For each category C_i,
  - P(C_i) = Number of images of category C_i / Total number of images
- In all images *I* of category C_i ,
  - For each keypoint V_j
    - P (V_j | C_i) = Number of keypoints V_j in *I* / Total number of keypoints in *I*
      = n_{ij} / n_i
    - But use Laplace smoothing to avoid numbers near zero.

    P (*Vj* | C_i) = (n_{ij} + 1) / (n_i + |*V*|)

Slide inspired by [3]

11

Categorization by Naïve Bayes III: Testing

- $P(C_i|Image) = \beta P(C_i)P(Image|C_i)$

$$= \beta P(C_i)P(V_0, V_1, .. ,V_m|C_i)$$

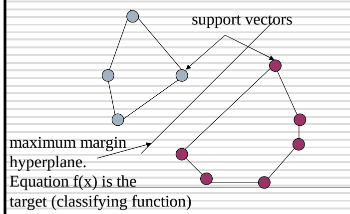
$$= \beta P(C_i)\prod_{i=0}^{m} P(V_i|C_i)$$

Slide inspired by [3]

12

SVM: Brief Introduction

- SVM classifier finds a hyperplane that separates two-class data with maximum margin.



Two class dataset with linearly separable classes.

Maximum margin hyperplane give greatest separation between classes.

The data instances closest to the hyperplane are called support vectors.

Categorization by SVM I: Training

- The classifying function is
  - **f** (**x**) = sign ($\sum_i \boldsymbol{y}_i \boldsymbol{\beta}_i \boldsymbol{K}(\boldsymbol{x}, \boldsymbol{x}_i) + \boldsymbol{b}$)
  - $\boldsymbol{x}_i$ is a feature vector from the training images, $\boldsymbol{y}_i$ is the label for $\boldsymbol{x}_i$ (yes, in category C_i, or no not in C_i), $\boldsymbol{\beta}_i$ and **b** have to be learnt.
  - Data is not always linearly separable (Non linear SVM)
    - A function $\boldsymbol{\Phi}$ maps original data space to higher dimensional space.
    - $\boldsymbol{K}(\boldsymbol{x}, \boldsymbol{x}_i) = \boldsymbol{\Phi}(\boldsymbol{x}).\boldsymbol{\Phi}(\boldsymbol{x}_i)$

Categorization by SVM II: Training

- For an image of category C_i , $\boldsymbol{x}_i$ is a vector formed by the number of occurrences of keypoints V in the image.
- The parameters are sometimes learnt using **Sequential Quadratic Programming**. The approach used in the paper is not mentioned.
- For the *m* class problem, the authors train *m* SVMs, each to distinguish some category C_i from the other *m*-1.

Categorization by SVM III: Testing

- Given a query image, assign it to the category with the highest SVM output.

Experiments

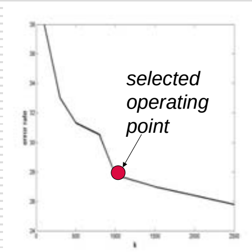
- Two databases
  - DB1: In-house. 1779 images.
    - 7 object classes: faces, buildings, trees, cars, phones, bikes.
    - Some images contain objects from multiple classes. But large proprtion of image is occupied by target image.
  - DB2: Freely available from various sites. About 3500 images.
    - 5 object classes: faces, airplanes, cars (rear), cars(side) and motorbikes(side).

Performance Metrics

- Confusion Matrix, M
  - m_{ij} = Number of images from category *j* identified by the classifier as category *i*.
- Overall Error Rate, *R*
  - Accuracy = Total number of correctly classified test images/Total number of test images
  - *R* = 1 – Accuracy
- Mean Rank, *MR*
  - *MR* for category *j* = E [rank of class *j* in classified output | true class is *j*]

Finding Value of *k*

- Error rate decreases with increasing k.
- Decrease is low after k >1000.
- Choose k = 1000.
  - Good tradeoff between accuracy and speed.



Graph of error rate vs. k for Naïve Bayes for DB1
Graph is taken from [2]

Naïve Bayes Results for DB1

True →	Faces	Buildings	Trees	Phones	Cars	Bikes	Books
Faces	**75**	4	2	2	4	3	9
Buildings	2	**42**	5	0	5	3	3
Trees	2	2	**80**	0	0	5	0
Phones	4	0	0	**76**	3	0	3
Cars	8	**15**	1	15	**67**	**13**	**13**
Bikes	2	**14**	11	0	9	**73**	0
Books	4	**19**	0	5	7	1	**69**
Mean rank	1.49	1.88	1.33	1.33	1.63	1.57	1.57

Confusion Matrix for Naïve Bayes on DB1
Overall error rate = 28%

Table taken from [2]

SVM Results

- Linear SVM gives best results out of linear, quadratic and cubic, except for cars. Quadratic gives best results on cars.
  - How do we know these will work for other categories? What if we have to use higher degrees? Only time and more experiments will tell.

SVM Results Results for DB1

True →	Faces	Buildings	Trees	Cars	Phones	Bikes	Books
Faces	**98**	**14**	**10**	**10**	**34**	0	13
Buildings	1	**63**	3	0	3	1	6
Trees	1	10	**81**	1	0	6	0
Cars	0	1	1	**85**	5	0	5
Phones	0	5	4	3	**55**	2	3
Bikes	0	4	1	0	1	**91**	0
Books	0	3	0	1	2	0	**73**
Mean rank	1.04	1.77	1.28	1.30	1.83	1.09	1.39

Confusion Matrix for SVM on DB1
Error rate for faces = 2%. But increased rate of confusion with other categories due to larger number of faces in the training set.

Overall error rate = 15%

Multiple Object Instances: Correctly Classified

Images taken from [2]

Partially Visible Objects: Correctly Classified

Images taken from [2]

Images with Multi-Category Objects

Images taken from [2]

25

Conclusions

- Good results for 7 category database.
  - However time information (for training and testing) not provided!
- SVMs superior to Naïve Bayes.
- Robust to background clutter.
  - Extension is to test on databases where the target object does NOT form a large fraction of the image.
  - May need to include geometric information.

26

References

1. G. Csurka, C. Bray, C. Dance, and L. Fan. *Visual categorization with bags of keypoints*. In Workshop on Statistical Learning in Computer Vision, ECCV, 2004.
2. Gabriela Csurka, Jutta Willamowski, Christopher Dance. Xerox Research Centre Europe, Grenoble, France. *Weak Geometry for Visual Categorization*. Presentation Slides.
3. R. Mooney. Computer Science Department, University of Texas at Austin. *CS 391L: Machine Learning - Text Categorization.* Lecture Slides.

27

SVM Results Results on DB2

	Faces (frontal)	Airplanes (side)	Cars (rear)	Cars (side)	Motorbikes (side)
Faces (frontal)	**94**	0.4	0.7	0	1.4
Airplanes (side)	1.5	**96.3**	0.2	0.1	2.7
Cars (rear)	1.9	0.5	**97.7**	0	0.9
Cars (side)	1.7	1.9	0.5	**99.6**	**2.3**
Motorbikes (side)	0.9	1.9	0.9	**0.3**	**92.7**
Mean rank	1.07	1.04	1.03	1.01	1.09

Confusion Matrix for SVM on DB2

28