Visual Language Modeling for Image Classification

Lei Wu
MOE-MS Key Lab of MCC
University of Science and Technology of China
+86-551-3600681
leiwu@live.com

Mingjing Li, Zhiwei Li, Wei-Ying Ma
Microsoft Research Asia
49 Zhichun Road
Beijing 100080, China
+86-10-58968888
{mjli, zli, wyma}@microsoft.com

Nenghai Yu
MOE-MS Key Lab of MCC
University of Science and Technology of China
+86-551-3600681
ynh@ustc.edu.cn

ABSTRACT

Although it has been studied for many years, image classification is still a challenging problem. In this paper, we propose a visual language modeling method for content-based image classification. It transforms each image into a matrix of visual words, and assumes that each visual word is conditionally dependent on its neighbors. For each image category, a visual language model is constructed using a set of training images, which captures both the co-occurrence and proximity information of visual words. According to how many neighbors are taken in consideration, three kinds of language models can be trained, including unigram, bigram and trigram, each of which corresponds to a different level of model complexity. Given a test image, its category is determined by estimating how likely it is generated under a specific category. Compared with traditional methods that are based on bag-of-words models, the proposed method can utilize the spatial correlation of visual words effectively in image classification. In addition, we propose to use the absent words, which refer to those appearing frequently in a category but not in the target image, to help image classification. Experimental results show that our method can achieve comparable accuracy while performing classification much more quickly.

Categories and Subject Descriptors

I.5.2 [**Pattern Recognition]**: Design Methodology; I.2.10 **[Artificial Intelligence]:** Vision and Scene Understanding; I.4.8 **[Image Processing and Computer Vision]:** Scene Analysis

General Terms

Algorithms，Experimentation.

Keywords

Image classification, visual language model, absent word criterion.

1. INTRODUCTION

Image classification aims at automatically classifying images into one of a number of predefined categories. Although it has been studied for many years, it is still a challenging problem within multimedia and computer vision. The major difficulties come from the influence of complex background, occluding objects, the semantic gap between low-level visual features and high-level semantic concepts, and so on. These difficulties as well as the heavy computational burden have limited the practical application of image classification in many scenarios, such as web image search, video surveillance, medical image system, etc. Techniques for effective and efficient image classification are called for.

Current approaches for image classification are based on either global features or local features. The former kind usually performs classification using global features extracted from whole images, such as color histogram, texture histogram, etc. Although it is successful in some cases, like indoor/outdoor classification [24] and city/landscapes classification [20], this kind of approaches is susceptible to local and global variations, e.g. slight changes of the viewpoint or illumination. Meanwhile, the later kind attempts to exploit local properties of images in classification [13]. For example, Gorkanai et al. [25] proposed to classify city/suburb scenes by dividing the images into 16 non-overlapping equal-sized blocks. The image category is determined by the dominant orientation of each block. In 2001, a graph/photograph classification is successfully achieved using the wavelet coefficients in high frequency band of image sub-blocks [26]. As the image regions are not equally informative, some methods try to perform classification based on regions of interest [9][22][23]. First, informative regions are either obtained by image segmentation, or generated around interest points in the image. Then a feature vector is extracted for each region according to its color or texture information. Sometimes these feature vectors are further condensed by dimensionality reduction methods (e.g. PCA) or mapped into visual words via clustering. For classification, two frameworks are usually adopted. Some methods, like pLSA [23][5] and LDA [5], adopt the generative model framework. Others adopt the multiple instance learning framework [21][22]. No matter what kinds of features are extracted, most approaches only utilize the co-occurrence of image features in classification, while ignore the spatial correlation of them, which might provide additional information to help image classification.

In this paper, we propose to build visual language models for image classification, which capture both co-occurrence and spatial proximity of local image features. This approach first segments images into patches, then maps each patch into a visual word. It also assumes that each visual word is only conditionally

MIR'07, September 28-29, 2007, Augsburg, Bavaria, Germany.

dependent on its neighbors. Based on a collection of training images for a category, the spatial correlation of visual words is modeled as the conditional probability by using statistical language modeling algorithms. Given a novel image, its label is determined by how likely it is generated under a specific category.

The rest of the paper is organized as follows. In Section 2, we introduce some related work on current image classification methods. In Section 3, we give an overview of the mature statistical language models, which have been proved to be successful in text classification. The proposed visual language modeling method is elaborated in Section 4. Moreover, a new concept of absent words is provided in Section 5, along with its application in image classification. The evaluation of the proposed models and comparison with some state-of-the-art image classification methods are reported in Section 6. Finally, we conclude in Section 7.

2. RELATED WORK

Recently, some state-of-the-art image categorization methods, such as pLSA and LDA, are based on the generative models. In 2005, Sivic et al. [5] proposed to take the relation between images and visual words as that of documents and terms, and applied pLSA [10] and LDA [11] to handle the classification problem. pLSA assumes that there is a number of hidden topics within a category, and adopts a mixture of models to describe the distribution of visual words. EM algorithm is used to iteratively estimate the posterior probabilities of an image, which in turn determine the image's category. This process makes the method complex and time-consuming. Taking a further step, LDA assumes that the distribution of visual words and hidden topics are conditionally dependent on some hidden variables. To best estimate these hidden variables, Gibbs sampling is often adopted to solve the problem. This makes the method more complex. Both methods are based on the bag-of-words approach, and ignore the spatial information of visual words.

In order to use the spatial information for classification, TSI-pLSA (translation and scale invariant pLSA) was proposed [12], which considers the distribution of visual words within sub-windows of the image. In fact, it is a kind of localized bag-of-words approach. Later, the concept of visual phrase was proposed [18], which refers to frequently co-occurred visual words in the image. It can be treated as an extended set of visual words. Both methods only partially utilize the spatial correlation of visual words in image classification.

3. OVERVIEW OF STATISTICAL LANGUAGE MODELING

Statistical language modeling (SLM) has been widely used in natural language processing applications, such as automatic speech recognition [1], machine translation [2], automatic spelling correction [3], and text classification [27]. SLM employs statistical estimation technique as a computational mechanism to obtain the conditional probability of a word sequence. The basic model can be represented as follows:

$$p(w_i^n) = p(w_i)\prod_{k=2}^{n} p(w_{i+k-1}|w_i^{k-1}) \quad (1)$$

w_k represents the k-th word in the sequence. w_i^n represents the string $w_i w_{i+1} \cdots w_{i+n-1}$. $p(w_i^n)$ is the probability that the word sequence appears. Usually, if SLM is adopted to estimate the conditional probability of word sequence of length n, it is called n-gram language model. Given a vocabulary V of size|V|, a unigram model has $|V|-1$ independent parameters, the one is removed by the constraint that all of the probabilities add up to 1. In the unigram case, the conditional probability becomes the word probability $p(w_k) = \mathrm{Count}(w_k)/T$, where $\mathrm{Count}(w_k)$ counts the number of times w_k occurs in the category and T is the total number of words in the category. Accordingly, an n-gram model has$|V|^n - 1$ independent parameters. N-gram model estimates its parameters by counting the number of n-gram occurring in the document.

$$p(w_{i+n-1}|w_i^{n-1}, C) = \frac{\mathrm{Count}\,(w_i^{n-1}w_{i+n-1}|C)}{\sum_w \mathrm{Count}\,(w_i^{n-1}w|C)} \quad (2)$$

SLM is usually used in the context of a Bayesian classifier for text classification. Given a document, which is represented as a sequence of words, its category C^* can be determined by the following equation.

$$C^* = arg\,max_{C_j}\, p(C_j)\prod_{w_i^n \in D} p(w_{i+n-1}|w_i^{n-1}, C_j) \quad (3)$$

if n=1, Eq. (3) is a unigram classifier; if n=2, it is called a bigram classifier; if n=3, it is called a trigram classifier. Unigram and trigram classifier are mostly adopted in text classification.

4. VISUAL LANGUAGE MODELING

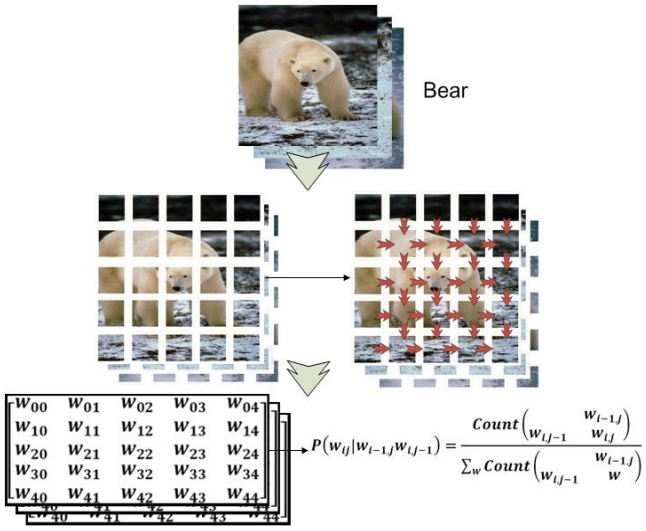


Figure 1. Process of trigram language model training

Inspired by the successful application of SLM in text classification, we propose to build visual language models for image classification. These are theoretical parity with nature language and visual language. Nature language consists of words, and visual language consists of image patches. In nature language, there are grammar, which restricts the words' distribution and order. In an image, if divided into patches, there are absolutely some constraints on how the patches are combined to form a meaningful object. Randomly combination of the patches will not construct a meaningful image. Visual language modeling is based on the assumption that there are implicit visual grammars in a meaningful image. To build visual language models, a number of factors should be considered. Firstly, a text document consists of words, each of which has its own semantic meanings, while an image consists of pixels. A pixel alone does not make any sense at all. A group of pixels together may make some sense. Thus, a larger unit is required to represent the

content of an image. Here, we adopt local patches, each of which covers a sub-window of an image. Secondly, considering the variance of pixels within a local patch, the number of different patches is huge even if the patch size is small. Therefore, local patches should be properly quantized into a limited number of visual words, typically via clustering, in order to facilitate the visual language model training. Lastly, images are 2-D signals while text documents are 1-D in nature. We can assume that each visual word is conditionally dependent on its neighbors. However, there are too many parameters that need to be estimated. In case that the size of visual vocabulary is n and only 4-neighborhood is considered, the number of conditional probabilities is n^5. To train a reasonably good model, a large number of training images are required. To alleviate the dependence on training data, we further simplify the model. We assume that an image is generated in a Markov process and its visual words are generated in the order from left to right and top to bottom. Thus a visual word is only conditionally dependent on its previous words. In this way, traditional SLM training methods can be applied to train a visual language model for an image category.

4.1 Image Representation

As afore mentioned, each image should be transformed into a matrix of visual words. For this purpose, many schemes may be used. A straightforward way is to segment images into equal-sized patches, then cluster all patches in the training set into groups based on their features, and use the cluster IDs as visual words. Here we represent visual words by a hashing based method which has been proven to be efficient and effective for large-scale duplicate image detection [14].

Four steps are performed to transform an image into a matrix of visual words: image subdivision, feature vector extraction, redundancy reduction, and hash coding. To preserve the spatial information between patches, uniformly sampled equal-sized sub-window (e.g 8×8) is chosen to divide each image into a matrix of patches. Then within each patch, some kinds of features are extracted to describe its property. In this paper, we have tried four kinds of features separately: the raw pixel values in RGB or HSI color space, SIFT descriptor [6] (SD), and texture histogram (TH). For RGB and HSI features, each patch is described by a 192-dim vector (8pixels×8pixels×3colors). For SD, each patch is represented by a 128-dimensional vector. For texture histogram, each patch is represented by an 8-dimensional vector, which records the gradient magnitude in eight directions. To make the feature more robust to rotation, the directions are measured by the angle to the maximum gradient within the patch. Different kinds of features suit different categories of images. RGB and HSI focus on difference in colors, while it is sensitive to rotation and scaling. SD and TH focus on texture information and are insensitive to rotation, but do not contain color information. As there is much redundancy in high dimensional local features, PCA is adopted to project the former three features to lower dimensional space. A low dimensional feature vector $V = [v_0, v_1, \cdots, v_k]$ is further transformed to a more compact hash code $H = [h_0 h_1 \cdots h_k]$ using the similar method proposed in [14]. Each bit of the hash code indicates whether the corresponding dimension of the feature vector is above the average level or not. If v_i is larger than the mean value within the patch, h_i is set to 1 otherwise 0. In this way, the image is transformed into a matrix of visual words represented by hash codes. This kind of format will improve the efficiency of the training and classification process. The visual word matrix is named visual document.

4.2 Language Model Training

To simplify the training process, we make the following assumptions. First, we assume that each word in the visual document is not independent, but correlated with all the other words in the document. However, when the size of the vocabulary is large, it is difficult to model so many relations. So we make the second assumption that visual words are generated in the order from left to right, and top to bottom. Each word is conditionally dependent on its previous words. In fact, we can also assume the visual words are generated from right to left, bottom to top, or from both directions. For simplicity, we only take the first assumption, which is enough to measure the visual word proximity.

$$p(w_{ij} | w_{00} w_{01} \cdots w_{mn}) = p(w_{ij} | w_{00} w_{01} \cdots w_{ij-1}) \quad (4)$$

The calculation of this conditional probability is still not efficient enough. Further inspired by 2D HMM [19][29] used in face recognition and indexing, we make the third assumption that each patch depends only on its immediate vertical and horizontal neighbors. There may be some statistical dependency on remote visual words. The description of this dependency will make the model too complex to implement. In this paper, we would ignore the remote words dependency, just as the language model does for text classification.

According to how much dependency information is considered in the model, we propose three kinds of visual language models, unigram, bigram and trigram. In unigram model, the visual words in an image are considered independent to each other. In bigram model, the proximity between two neighboring words is calculated, and in trigram model the words are assumed to depend on two immediate vertical and horizontal neighbors.

These three models are expressed in Eq. (5)~(7) respectively.

$$p(w_{ij} | w_{00} w_{01} \cdots w_{mn}) = p(w_{ij}) \quad (5)$$

$$p(w_{ij} | w_{00} w_{01} \cdots w_{mn}) = p(w_{ij} | w_{i-1,j}) \quad (6)$$

$$p(w_{ij} | w_{00} w_{01} \cdots w_{mn}) = p(w_{ij} | w_{i-1,j} w_{i,j-1}) \quad (7)$$

Where w_{ij} represents the visual word at Row i, Column j in the word matrix. In the following, we will discuss the training process for the three kinds of models one by one.

4.2.1 Unigram model

For each category, a unigram model characterizes the distribution of individual visual words under the category.

$$p(w_k | C) = \frac{Count\ (w_k | C)}{\sum_{w \in V} Count\ (w | C)}, \quad C = C_1, C_2, \cdots, C_n \quad (8)$$

The training process is described by Eq. (8), where C represents a predefined category, and w_k is the k-th visual word in the vocabulary. $Count(w_k | C)$ represents the number of times that the word w_k appears in the training images of category C. To avoid zero probability which would cause the classifier to fail, each unseen word for the category is assigned a small prior probability. Accordingly, the amount of this prior probability should be discounted from the appearing words to meet the

condition that the sum of probability is 1. So the smoothed words distribution is represented by Eq. (9).

$$p(w_k|C) = \begin{cases} \frac{Count\ (w_k|C)\times(1-\frac{1}{R})}{\sum_{w\in V} Count\ (w|C)} & Count(w_k|C) > 0 \\ 1/R & otherwise \end{cases} \quad (9)$$

R is the total number of words in the training set. This probabilistic model will tell how likely each word is generated from the category.

4.2.2 Bigram model

Unlike the unigram model, a bigram model assumes that each visual word is conditionally dependent on its left neighbor only. So the training process is to learn the conditional probability by Eq. (10).

$$p(w_{ij}\ |\ w_{i,j-1}, C) = \frac{Count\ (w_{ij}, w_{i,j-1}|C)}{Count\ (w_{i,j-1}|C)} \quad (10)$$

$w_{i,j-1}$ is the horizontal neighbor of $w_{i,j}$ in the visual words matrix. Bigrams, however, are sparsely distributed in the image, and the maximum likelihood estimation is usually biased higher for observed samples and biased lower for unobserved samples. Thus smoothing technique is required to provide better estimation of the infrequent or unseen bigrams. Instead of just assigning a small constant prior probability, we adopt more accurate smoothing method [28], which combines back-off and discounting [15].

$$p(w_{ij}\ |\ w_{i,j-1}, C) = \begin{cases} \beta(w_{i,j-1}) \times p(w_{ij}|C), & if\ Count(w_{i,j-1}w_{ij}|C) = 0 \\ \hat{p}(w_{ij}|w_{i,j-1}, C), & otherwise \end{cases} \quad (11)$$

$$\beta(w_{i,j-1}) = \frac{1-\sum_{Count\ (w_{i,j-1}w)>0} \hat{p}(w|w_{i,j-1}, C)}{1-\sum_{Count\ (w_{i,j-1}w)>0} p(w|C)} \quad (12)$$

$$\hat{p}(w_{ij}|w_{i,j-1}, C) = d_r \times \frac{Count\ (w_{i,j-1}w_{ij}|C)}{Count\ (w_{i,j-1}|C)} \quad (13)$$

Back-off method is represented in Eq. (11) and (12), and discounting is represented in Eq. (13). If the bigram does not appear in the category, back-off method is applied to calculate the bigram model from the unigram model by Eq. (11) and (12). $\beta(w_{i,j-1})$ is called back-off factor. If bigram $w_{i,j-1}w_{ij} \in D$ appears in category C, the discounting method is used to depress the estimation of its conditional probability. d_r is called the discounting coefficient. There are many discounting methods, such as linear discounting (Eq. 14) and absolute discounting (Eq. 15).

$$d_r = 1 - \frac{n_1}{R} \quad (14)$$

$$d_r = \frac{r-b}{r} \quad (15)$$

$$b = \frac{n_1}{n_1+2n_2} \quad (16)$$

r is the number of times bigram $w_{i,j-1}w_{ij}$ appears; R is the total number of words in the training set, and n_i is the number of visual words that appear i times in the category.

4.2.3 Trigram model

The above two modeling processes are almost the same with building of statistical language models used in text classification, while in trigram model, there are obvious differences. In traditional text document, trigram is a sequence of three words $< w_{i-2}, w_{i-1}, w_i >$, while in visual words document, as it is arranged in matrix form, we assume each word is conditionally dependent on its previous vertical and horizontal patches. So these three words form a trigram $< w_{i-1,j}, w_{i,j-1}, w_{ij} >$.

The training process of a trigram model is illustrated in the following equation.

$$p(w_{ij}\ |\ w_{i-1,j}, w_{i,j-1}, C) = \frac{Count\ (w_{i-1,j}w_{i,j-1}w_{ij}|C)}{Count\ (w_{i-1,j}w_{i,j-1}|C)} \quad (17)$$

For the same reason with bigram model, discounting and back-off methods are also defined in trigram model.

$$p(w_{ij}|w_{i-1,j}w_{i,j-1}, C) = \begin{cases} \beta(w_{i-1,j}w_{i,j-1})p(w_{ij}|w_{i,j-1}, C), & if\ Count(w_{ij}{}^3|C) = 0 \\ \hat{p}(w_{ij}|w_{i-1,j}w_{i,j-1}, C), & otherwise \end{cases} \quad (18)$$

$$\beta(w_{i-1,j}w_{i,j-1}) = \frac{1-\sum_{Count\ (w_{i-1,j}w_{i,j-1}w)>0} \hat{p}(w|w_{i-1,j}w_{i,j-1}, C)}{1-\sum_{Count\ (w_{i-1,j}w_{i,j-1}w)>0} \hat{p}(w|w_{i,j-1}, C)} \quad (19)$$

$$\hat{p}(w_{ij}|w_{i,j-1}, C) = d_r \times \frac{Count\ (w_{i,j-1}w_{ij}|C)}{Count\ (w_{i,j-1}|C)} \quad (20)$$

$w_{ij}{}^3$ represents the trigram $w_{i-1,j}w_{i,j-1}w_{ij}$. The spatial correlation between visual words is evaluated in the distribution of the trigrams.

It is worth noting that not all visual words are useful for classification. As a result, feature selection is adopted in the language model training process. In this paper, words are selected by the document frequency (DF) scheme. Only those words which often appear in different images are selected. This approach can depress the influence of random background and reduce the size of vocabulary. As the visual words are in the compact hash code form, the calculation of the conditional probability is especially swift. All the word distribution and conditional probability is also stored in a hash table.

The training procedure is as follows:

1. Load training images
2. Divide each image into patches
3. Generate a hash code for each patch to form a visual document
4. Build visual language models for each category by calculating the conditional distribution of unigram, bigram and trigram

The process of building trigram visual language model is illustrated in Fig. 1.

4.3 Image Classification

In the classification phase, each novel image is transformed to a matrix of words in the same way as the training process. The image is assigned to the most probable category by maximizing the posterior probability.

$$C^* = argmax_{C_j} p(C_j|D) \quad (21)$$

D represents a novel visual document. The maximum probability over all categories C_j , $j = 1,2,\cdots,K$ is chosen as its label C^*.

For unigram model, visual words in the document are assumed independent to each other. Thus the classification process can be transformed into the form of Eq. (22).

$$C^* = argmax_{C_j} \prod_{w_{ij} \in D} p(w_{ij}|C_j)\, p(C_j) \quad (22)$$

For bigram model, words are dependent to its left neighbor. So the classification is formulated as the following maximizing process.

$$C^* = argmax_{C_j} \prod_{w_{i,j-1}w_{ij} \in D} p(w_{ij}|w_{i,j-1}, C_j)\, p(C_j) \quad (23)$$

Accordingly, a trigram model based classifier is illustrated by the subsequent equation.

$$C^* = argmax_{C_j} \prod_{w_{ij}{}^3 \in D} p(w_{ij}|w_{i-1,j}\, w_{i,j-1}, C_j)\, p(C_j) \quad (24)$$

As the word distribution and conditional probability is stored in a hash table. The classification process can be implemented by an efficient hash table lookup.

The classification procedure is as follows:

1. Load a novel image
2. Generate a visual document from the image
3. Estimate the novel image's category by maximizing the conditional distribution of n-grams (n=1,2,3) over all categories.

5. IMAGE CLASSIFICATION BY ABSENT WORDS

The classification in the previous section uses only the information of words that appear in the image. Words that are not contained in the test image still have distinguishing information among different categories. In this section, we will discuss how to use this information to help classification.

5.1 Basic Idea of Absent Words (AW)

The words that contained in a test image are called *appearing words*. The words appearing in the training set while absent in a test image are defined as *absent words*. For example, if the visual word indicating human eye appears in the face category in the training set, but it does not appear in an image, we call it an absent word of this image. And the confidence of classifying it to face category should be discounted. That is to say, if an image does not contain visual word of "eye", it is less likely to be a face image.

The information of the absent words is named AWI and is measured by the entropy. If absent words of a test image have wavy distribution in a category, they are considered informative to this category. In other words, the test image is less likely to be of this category. To achieve a better estimation of the category, we combine absent words information with appearing words information by the extended document model (EDM). When information in appearing words is not adequate to judge an image's category, the classifier will refer to the absent words in EDM.

5.2 Extended Document Model

The extended document model (EDM) is used to combine the appearing words information with absent words information in image classification. In EDM, a visual document is represented by two parts. First part is the visual words matrix, which contains all appearing words with order. Trigram model is adopted to calculate the word frequency as well as the spatial correlation. The second part is formed by a set of non-order absent words. As the neighboring information is unavailable for these absent words, unigram model is used for this part. Suppose that we have vocabulary $V = \{w^1, w^2, \cdots, w^{|V|}\}$ of size $|V|$, which contains all the high frequency words in the dataset. Each of the documents is represented as $D = \{w_{00}, w_{01}, \cdots, w_{mn}\}$, where $|D|$ is the size of the document. We also define a document vocabulary:

$$\langle D \rangle, \quad s.t.\, w^i \in D \Leftrightarrow w^i \in \langle D \rangle,$$
$$\forall w^i, w^j \in \langle D \rangle, i \neq j, \Rightarrow w^i \neq w^j$$

All the words in document vocabulary $\langle D \rangle$ are specific and appearing in the document D. The supplementary document is $\bar{D} = V - \langle D \rangle$, which contains all the absent words of the document. Finally, an *extent document* is defined as the combination of the two parts, denoted as $D^e = D|\bar{D}$.

Instead of classifying images by maximizing $p(C_j|D)$, we now maximize $p(C_j|D^e)$. The classification phase is represented in the following three forms (25)~(27) corresponding to unigram, bigram and trigram models separately.

$\bar{p}(w_k|C_j)$ is the normalized absent words distribution. λ is a small constant used as a trade-off parameter to balance the two parts. Generally, the visual document classification is mainly based on the appearing words information. When the first part is almost equal in ambiguous cases, the AWI part will take effect.

$$\begin{aligned} C^* = arg\max_{C_j} p(C_j|D^e) &= arg\max_{C_j} \left(p(D^e|C_j)p(C_j)\right) \\ &\approx arg\max_{C_j} \left(\sum_{w_{ij} \in D} \log p(w_{ij}|C_j) \right. \\ &\left. + \lambda \sum_{w_k \in \bar{D}} \log \bar{p}(w_k|C_j) + \log p(C_j) \right) \end{aligned} \quad (25)$$

$$\begin{aligned} C^* \approx arg\max_{C_j} \left(\sum_{w_{i,j-1}w_{ij} \in D} log\, p(w_{ij}|w_{i,j-1}, C_j) \right. \\ \left. + \lambda \sum_{w_k \in \bar{D}} log\, \bar{p}(w_k|C_j) + log\, p(C_j) \right) \end{aligned} \quad (26)$$

$$\begin{aligned} C^* \approx arg\max_{C_j} \left(\sum_{w_{i-1,j}w_{i,j-1}w_{ij} \in D} log\, p(w_{ij}|w_{i-1,j}\, w_{i,j-1}, C_j) \right. \\ \left. + \lambda \sum_{w_k \in \bar{D}} log\, \bar{p}(w_k|C_j) + log\, p(C_j) \right) \end{aligned} \quad (27)$$

$$\bar{p}(w_k|C_j) = \frac{p(w_k|C_j)}{\sum_{w \in \bar{D}} p(w|C_j)} \quad (28)$$

5.3 How can AWI help?

The mechanism of AWI in a common ambiguous test image is discussed in the following two cases. In Case 1, the probability of appearing words in two categories equals to each other. In

Case 2, the appearing words distribution is different while their product equals, which would also cause identical posterior probability under more than one category.

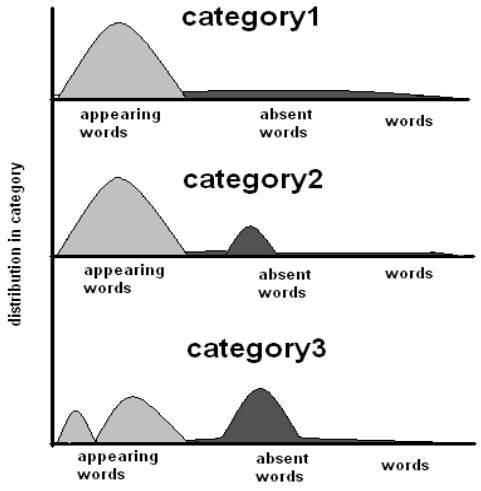


Figure 2. Word distribution in three categories. Light color represents the distribution of appearing words; Dark color is the distribution of absent words.

The ambiguity in Case 1 can be represented in following equation.

$$p(w_{ij}|w_{i-1,j}w_{i,j-1}, C_0) = p(w_{ij}|w_{i-1,j}w_{i,j-1}, C_1) \quad w_{ij}{}^3 \in D \tag{29}$$

As a result, the first part of the classifier equals. The classification can be simply achieved by maximizing the AWI part.

$$C^* = arg\max_{C_j}\left(\lambda\prod\nolimits_{w_k\in\bar{D}} p(w_k|C_j)\right)$$

$$= arg\max_{C_j}\left(\lambda\sum_{w_k\in\bar{D}}\log p(w_k|C_j)\right), j = 0,1$$

$$s.t.\sum_{w_k\in\bar{D}} p(w_k|C_0) = \sum_{w_k\in\bar{D}} p(w_k|C_1) \tag{30}$$

As the appearing words have identical distribution under the two categories, the sum of absent words distribution also equals. So this is a constrained optimization. The more uniformly these absent words distribute, the larger their product will be. That is, the absent words with peek distribution are considered more important than those with uniform distribution. The above ambiguous case is discussed under rigor assumption. However, some cases may not meet the assumption while still meet ambiguity in classification. For the second case, ambiguity is formulated by Eq. (31).

$$\prod_{w_{ij}{}^3\in D} p(w_{ij}|w_{i-1,j}w_{i,j-1}, C_0) = \prod_{w_{ij}{}^3\in D} p(w_{ij}|w_{i-1,j}w_{i,j-1}, C_1) \tag{31}$$

In this case, the first part of the classifier also equals for the two categories. However, the sum of absent words distribution under the two ambiguous categories may not equal. Thus, we need to normalize the absent words distribution. After normalization, the problem becomes the same with the previous case.

In sum, the EDM classifier can adaptively choose appearing words information and absent words information for classification. Ordinarily, as λ is small, it performs as a trigram classifier based on appearing words information; in ambiguous cases, it classifies mainly based on AWI.

Fig.2 provides a visual explanation. The horizontal axis represents all the words in the vocabulary. The vertical axis is the word distribution under different categories. For the appearing words in an image, their distributions in Category 1, 2 and 3 are illustrated in light gray color in Fig. 2. By chance, the product of the appearing words distribution happened to be the same. However, when the distribution of absent words (as shown in dark gray color in Fig. 2) is used, the ambiguity can be cleared. For Category 1 and 2, although the appearing words distribution is identical, the absent words distribution in Category 2 is more wavy. So the test image is more likely generated by Category 1. Comparing Category 1 and 3, the product of the appearing words distribution happens to be the same. In this case, normalization is needed to make the sum of the absent words distribution equal. Then again, for the same reason, Category 1 wins.

6. EXPERIMENT

The goal of the experiment is to evaluate the performance of the proposed visual language modeling method. The comparison with pLSA [5] and LDA [11] is performed on Caltech dataset to compare the efficiency and effectiveness. Then we compare the classification precision of trigram model with that of unigram model to find out how much the performance has been improved by modeling spatial correlation between image patches. Furthermore, to evaluate the contribution of AWI, we also compare the performance of trigram model with that of trigram-AWI model, which adopt EDM instead of common visual documents. Finally, the experiment to test the parameter sensitivity is performed to give an idea of how much patch size can affect the results.

6.1 Dataset

There are two dataset for the experiment. In order to compare the classification results, the first dataset we use is the same with [5], which consists of 7 categories and background from Caltech image dataset [7] and 101 category dataset [8]. Another smaller set containing 6 difficult categories from Corel dataset is used to test the robustness of the methods, as there are more variation in illumination, scaling, rotation, viewpoint change and fewer images for each category. Most of the samples in Caltech dataset contain single object (Fig. 4). A majority of Corel images have multiple objects (Fig. 5).

Table 1. Categories and data distribution

Dataset 1: Caltech dataset		**Dataset 2: Corel dataset**	
Category	**Images**	**Category**	**Images**
Faces	435	Balloon	100
Motorbikes	800	Car	100
Airplanes	800	Fitness	100
Cars rear	1155	Horse	100
Leopards	200	Sunset	100
Watch	241	Waterfall	100
Ketch	114		
Background	1370		

Dataset 1, which is the same dataset used in [5], is used to compare the efficiency and accuracy between different models and methods. Dataset 2, is used to test the robustness of the method. For this dataset, we have chosen six difficult categories from Corel dataset. There are multiple objects in one image, various background, illumination changes, pose changes, large scaling, rotation and view point changes.

6.2 Features

For the visual language modeling method, the effective uniformly sampled patches are used to preserve the spatial correlation between patches. Each patch of the image is represented by the texture histogram (TH) feature in hash code form. RGB, HSI and SD features are also tried in experiment D to make a comparison between features.

For LDA and pLSA methods, more complex features are adopted to make sure of the best performance. Salient feature extraction is adopted in the pLSA and LDA methods to obtain the most informative regions in the image. Each region is centered at the salient point and represented by SIFT feature [6]. Visual words are formed by clustering these informative patches using k-means algorithm.

6.3 Experimental Settings

To reduce the influence of image size variation, all the following experiments have resized the image to 640*640. We conduct 5 experiments to evaluate the proposed models. We denote methods based on unigram assumption, trigram assumption as "Unigram" and "Trigram" separately. The trigram method which adopts EDM instead of visual document is denoted as "AWI".

A. Caltech experiment

This experiment is mainly designed to evaluate the efficiency of the proposed visual language modeling method in image classification application. The comparison with other state-of-the-art methods pLSA and LDA is conducted on Dataset 1. The distribution of the data is shown in Table 1 on the left two columns.

In the methods of pLSA and LDA, two kinds of local feature extraction methods are adopted. One is to construct the affine co-variant regions by elliptical shape adaptation about an interest point [17]; the other is constructing the regions using the maximally stable procedure of [16]. Watershed image segmentation is adopted in obtaining the region areas. The regions of the image are represented by ellipses and further mapped to a circle to compute the SIFT descriptor. Then the SIFT descriptors are quantized by k-means clustering to form visual words. The total vocabulary has 2,237 words. For pLSA, EM algorithm is used. For LDA, the Gibbs sampling is adopted.

For the proposed method, we choose 8x8 patch size and each patch is coded in 8-bit hash code. The feature used here is TH. It is worth noting that there is neither image segmentation nor interest point detection in visual language modeling method, which makes the method very efficient. We intend to see how well the language modeling method can still perform even when the complex and time-consuming processes are avoided.

B. Model comparison

The model comparison experiment is designed to test the priority and robustness of the language models to various backgrounds and multiple objects. We conduct the experiment on two datasets. One is the Caltech dataset of 7 categories + background used in Experiment A, and the other is the Corel dataset of 6 categories, which contains more variant backgrounds, pose changes and multiple objects. In the experiment, two kinds of visual language models (unigram model, trigram model) are compared. Unigram model ignores the spatial information of patches and takes the image as "bag-of-word". Trigram model has adopted the spatial correlation between patches to help classification. The comparison between these two kinds of models can show how much improvement is gained by using spatial correlation. All images are sub-divided into uniformly distributed patches of size 8*8 pixels.

C. AWI experiment

Experiment C is designed to test the effectiveness of AWI for image classification. We evaluate the effectiveness of AWI by comparing two trigram based classifiers on Dataset 1. One is original trigram modeling method (denoted as "Trigram"); the other adopts EDM and use trigram model for appearing words and unigram model for absent words (denoted as "AWI"). In both methods, the image is divided into 8x8 patches and the TH feature for each patch is coded in 8-bit hash code. We divided the dataset into five folds randomly, any four of which is used for training, and the left one is used for testing.

D. Feature comparison

The goal of Experiment D is to test the influence of different features on the classification results. We divide the image into patches of size 8x8, and represent each patch with four kinds of features: RGB, HSI, SD, and TH. Based on each feature setting, a separate trigram classifier is trained. In classification, we compare the precision of the four classifiers on each category. As different features suit different kinds of images, the precision over image categories will reflect the property of these features. We also intend to see how dramatically feature settings will affect the classification performance.

E. Parameter property

In this experiment, we try to test the influence of patch size on the classification performance. If the patch size is set too big, each patch will become too specific. The size of visual vocabulary will be large. To build language models under such big vocabulary space will front sparseness problem. Otherwise, there should be an unbelievably large training set. If the patch size is set too small, the proximity between neighboring patches becomes meaningless. In order to take advantage of visual grammar, more surrounding patches should be taken into account, which will make the model too complex to describe and implement.

To reveal the relation between classification accuracy and patch size settings, we conduct this experiment. We resize the images to 640x640, and then set the patch size to 8x8, 16x16, 32x32, and 64x64 separately. For each patch size, we calculate the classification accuracy and plot the relation curve as Fig. 3.

6.4 Evaluation and Discussion

Table 2. Result of experiment A

Method	pLSA	LDA	Trigram
Time (sec/image)	0.2	1.2	0.02
Accuracy (%)	59	64	82

Experiment A shows an appealing advantage of both accuracy and efficiency with trigram method. Comparing with pLSA, trigram method gains 39% on accuracy. Comparing with LDA, the gain is 28%. Table 2 also shows trigram method takes only 1/10 of the classification time of pLSA and only 1/60 of the time by LDA. It should be noticed that pLSA, LDA are implemented on a PC with 2GHz CPU [4]. And the trigram model method is implemented on a PC with 1.5GHz CPU. Considering inferior position of CPU speed, the efficiency of the proposed method is more noticeable. This advantage comes from several intrinsic properties of the visual language modeling method. First, it has separate training phase and classification phase. So it can learn offline, and once the models are built, the online classification task takes little time. Second, complex processes, such as image segmentation and interest point detection, are avoided. Third, in the classification phase, it only needs to look up a hash table to get the conditional probability of each trigram. This superior of trigram model makes classification on huge image database possible. The larger the image database is, the better performance trigram method will achieve, because large amount of samples will help to model an unbiased estimation of the underlying trigram distributions.

Table 3. Result of experiment B and C (precision)

Categories		Unigram	Trigram	AWI
Caltech dataset	airplanes	77.14	**93.08**	92.54
	background	74.31	84.19	**88.14**
	cars	67.32	**77.36**	75.74
	faces	41.67	**90.24**	88.89
	ketch	**9.36**	0.00	0.00
	leopards	28.26	88.89	**90.00**
	motorbikes	**88.16**	74.42	75.47
	watch	14.71	87.5	**100.0**
Corel dataset	Balloon	35.71	**38.89**	35.71
	Car	25.00	50.00	**60.00**
	Fitness	**90.00**	58.33	60.00
	Horse	32.61	32.08	**36.17**
	Sunset	0.00	28.57	**100.0**
	Waterfall	32.43	52.17	**56.00**

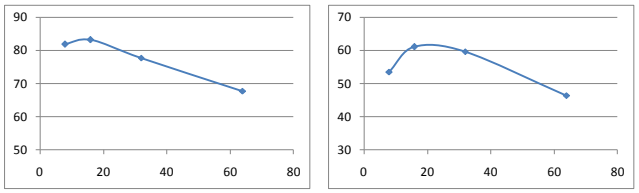
Table 4. Precision comparison of different features on Caltech dataset

Category	RGB	HSI	SD	TH
Airplane	28.75	15.44	54.74	**93.08**
Cars	**95.24**	0.00	55.26	77.36
Faces	33.33	65.22	72.73	**90.24**
Ketch	0.71	**33.33**	19.15	0.00
Leopards	2.58	13.01	53.85	**88.89**
Motorbikes	60.56	16.74	59.65	**74.42**
Watch	9.64	5.26	19.35	**87.5**

Table 3 gives both comparison results of experiment B and C. The comparison between the performance of "Unigram" and that of "Trigram" shows trigram has obvious improvement on most of the categories in the two datasets, since it uses additional information of spatial correlation. The improvement is especially obvious on "airplanes", "faces", "leopards", "watch", "Car", "Sunset" and "Waterfall". Further comparison between "Trigram" and "AWI" shows that the performance on some categories has improved, since AWI is taken into consideration. According to the performance on the more difficult Corel data, we find the advantage of "AWI" is more obvious. Since in the difficult dataset, there are more ambiguous cases, in which AWI will take effect.

Table 4 gives the comparison of precision under different feature settings. The highest precision is marked in bold font. We find that some categories are distinguished by colors, such as Cars, Ketch, while others are better discerned by texture.

Generally speaking, classification results vary dramatically with the change of feature settings, and TH feature is relatively more robust on all categories than other features. There are three reasons for TH's superiority in visual language modeling. First, as the same objects can have different colors, color feature is less meaningful than texture feature. Second, as the image is divided into uniform equal-sized patches, SD is no longer scale invariant. Third, high dimensional features will over-fit the specific patch.



(a) (b)

Figure 3. Influence of patch size. (a) and (b) are the performances of unigram and trigram with different patch size. The horizontal axis represents the patch size in pixel. The vertical axis represents the classification accuracy.

In experiment E, the influence of patch size on classification accuracy is studied. The relation between the patch size and accuracy is plotted in Fig 3. In the experiment, we also compare the performance curves of trigram model with that of unigram model to dig out some interesting properties. Comparing (a) and (b), we find that unigram model and trigram model have tiny different appetites on patch size. Although both models perform worse when patch size became larger than 16x16, unigram decreases rapidly when patch size is over small; while trigram model tends to keep relatively stable performance with small patch size. This is because small patch contains little meaningful information. Unigram model who classifies based on information within each patch cannot work at all with small patch. Trigram model, who classifies mainly based on the proximity between patches, however, can still work.

6.5 Discussion on Robustness

There are six main factors, which challenge image representation, scaling, rotation, translation, illumination variance, viewpoint change and complex backgrounds. Current image representation used in this paper fit all the factors except object scaling and rotation.

For translation, the visual language modeling method considers only the relative proximity of the image patch to its neighbors, and does not use the absolute coordinate of the patches. So the representation is invariant to object translation. For in-patch rotation, as the gradient histogram is oriented to the maximum gradient direction, which is a kind of rotation invariant feature.

However, it cannot deal with the whole image rotation. The hash code form of TH feature is somewhat robust to illumination change. Each bit of the hash code records only the relative intensity of the gradient magnitudes in each direction. If the illumination changes over the whole image simultaneously, there is no influence on the results. If the illumination changes within a small region, there may have some influence. However, since the small region can not dominate the visual document, the influence is limited. This is demonstrated by the experiment, since there are many illumination variation images in the dataset. Although different view point may lead to dramatic shape variation, the texture within local patches changes little. So the influence of viewpoint change is somewhat mitigated by the local features. For complex background, since visual language model is a statistic model. Only those frequent visual words will take effect. Random backgrounds are not able to affect the results. The frequent backgrounds themselves are correlated to the object. So if they can affect the results, it will be positive effect.

However, the main contribution of this paper focuses on how spatial information can be efficiently used by a statistical model, and concerns only the fundamental mechanism of the model. For simplicity, large object scaling and rotation is not discussed in this paper. To handle large scaling and rotation, more complicated features and segmentation strategy will be adopted, which may further improve the classification performance. This work is considered as important and independent to the fundamental mechanism. It will be discussed in the consecutive papers.

7. CONCLUSION

In this paper, we have proposed a visual language modeling method to classify images. It takes an image as a matrix of visual words and trains visual language models for each category. On classification, it labels the image with maximum posterior probability. As the method avoids image segmentation and interest point detection, it is especially efficient on huge image dataset. By using visual language models, both the visual word frequency and the spatial correlation have been considered simultaneously, which achieves obvious better performance than model ignoring spatial information (unigram model) in the experiment. The more words appear in training set, the better estimation of the underlying word distribution will be modeled. As a result, more superior properties of the method may be exposed when the size of image dataset increases.

With the proposed extended document model, we use not only appearing words information but also absent words information for classification. Appearing words information dominates the image label in common cases, and absent words information takes effect in ambiguous cases. This stratagem is tested effective on some of the categories in the experiment.

Meanwhile, we demonstrate two interesting and useful facts by experiment. One is that the feature settings can dominate the classification results. Some classes of images cannot be described properly by features with only color and textural information. The second fact is that patch size can somewhat influence the result. From the experiment, we find unigram model cannot work well with small patch size, while trigram model is relatively stable with small patch size.

These findings guide our future work. We will further investigate on the performance of novel image features such as some shape descriptors. More study on invariant representation of visual words matrix will proceed. Automatic choosing of patch size and feature switching according to image local information will also be studied in the future.

8. ACKNOWLEDGMENTS

The research is supported in part by National Natural Science Foundation of China (60672056) and Microsoft Research Asia Internet Services in Academic Research Fund. This work was performed at Microsoft Research Asia.

9. REFERENCES

[1] Bahl, L. R., Jelinek, F., and Mercer, R. L. A maximum likelihood approach to continuous speech recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1983.

[2] Brown, P. F., Cocke, J., DellaPietra, S. A., Mercer, R. L. and Roossin, P. S. A statistical approach to machine translation. *Computational Linguistics*, 1990.

[3] Mays, E., Damerau, F. J. and Mercer, R. L. Context-based spelling correction. IBM *Natural Language* ITL, 1990.

[4] Chatterjee, S., Hadi, A. and Price, B. Simple Linear Regression. *Regression Analysis by Example*, 3rd ed. New York: Wiley, 2000.

[5] Sivic, J., Russell, B., Efros, A., Zisserman, A. and Freeman, W. *Discovering object categories in image collections.* Technical Report A. I. Memo 2005-005, MIT, 2005.

[6] Lowe, D. Object recognition from local scale-invariant features. In *Proceedings of the International Conference on Computer Vision (ICCV'99)*, 1999, 1150-1157.

[7] Fergus, R., Perona, P. and Zisserman, A. Object class recognition by unsupervised scale-invariant learning. In *Proceedings of the Computer Vision and Pattern Recognition (CVPR'03)*, 2003.

[8] Li, F. F., Fergus, R., and Perona, P. Learning generative visual models from few training examples: An incremental Bayesian approach tested on 101 object categories. In *IEEE CVPR Workshop of Generative Model Based Vision*, 2004.

[9] Csurka, G., Bray, C., Dance, C. and Fan, L. Visual categorization with bags of keypoints. In *Workshop on Statistical Learning in Computer Vision, (ECCV'04)*, 2004, 1-22.

[10] Hofmann, T. Probabilistic latent semantic indexing. In *Proc. ACM SIGIR (SIGIR'99)*, ACM Press, 1999.

[11] Blei, D., Ng, A. and Jordan, M. Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, Jan 2003.

[12] Fergus, R., Li, F. F., Perona, P. and Zisserman, A. Learning object categories from Google's image search. In *Proc. Tenth IEEE International Conference on Computer Vision, (ICCV'05)*, 2005.

[13] Maree, R., Geurts, P., Piater, J. and Wehenkel, L. Random Subwindows for Robust Image Classification. In *Proceedings of the Computer Vision and Pattern Recognition (CVPR'05)*, 2005.

[14] Wang, B., Li, Z. W., Li, M. J. and Ma, W. Y. Large-Scale Duplicate Detection for Web Image Search. In *Proceedings*

of IEEE International Conference on Multimedia & Expo (ICME'06), 2006.

[15] Katz, S. M. Estimation of Probabilities from Sparse Data for the Language Model Component of a Speech Recognizer. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 35(3):400-401, 1987.

[16] Matas, J., Chum, O., Urban, M. and Pajdla, T. Robust wide baseline stereo from maximally stable extremal regions. In *Proceedings of The British Machine Vision Conference (BMVC'02)*, 2002, 384-393.

[17] Mikolajczyk, K. and Schmid, C. An affine invariant interest point detector. In *Proceedings of European Conference on Computer Vision (ECCV'02)*, Springer-Verlag, 2002.

[18] Zheng, Q., Wang, W. and Gao, W. Effective and efficient object-based image retrieval using visual phrases. In *Proc. of the 14th Annual ACM international Conference on Multimedia, (MM '06)*, 2006.

[19] Otluman, H. and Aboulnasr, T. Low Complexity 2-d Hidden Markov Model for Face Recognition. In *Proceedings of International Symposium on Computer Architecture. (ISCAS'00)*, 2000.

[20] Vailaya, A., Jain, A. K. and Zhang, H. J. On image classification: City images vs. landscapes. Pattern Recognition, Vol. 31, pp. 1921--1936, 1998.

[21] Maron, O. and Lozano-Perez, T. A framework for multiple-instance learning. In *M.I. Jordan, M.J. Kearns, and S.A. Solla, Eds. Advances in Neural Information Processing Systems 10*, Cambridge, MA: MIT Press, pp.570--576, 1998

[22] Bi, J., Chen, Y. and Wang, J.Z. A Sparse Support Vector Machine Approach to Region-Based Image Categorization. In *Proceedings of the Computer Vision and Pattern Recognition (CVPR'05),* 2005.

[23] Quelhas, P., Monay, F., Odobez, J., Gatica-Perez, D., Tuytelaars, T. and Gool, L. Modeling Scenes with Local Descriptors and Latent Aspects. In *Proc. Tenth IEEE International Conference on Computer Vision, (ICCV'05)*, 2005.

[24] Szummer, M. and Picard, R. Indoor-Outdoor Image Classification. In *Proc. IEEE Workshop on Content-Based Access of Image and Video Databases*, 1998, 42 – 51.

[25] Gorkani, M. M. and Picard, R. W. Texture orientation for sorting photos 'at a glance'. In *Proc. 12th Int. Conf. on Pattern Recognition (ICPR'94)*, 1994, 459-464.

[26] Wang, J., Li, J. and Wiederhold, G. SIMPLIcity: Semantics-sensitive integrated matching for picture libraries. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2001, 23(9):947-963.

[27] Peng, F. and Schuurmans, D. Combining Naive Bayes and n-Gram Language Models for Text Classification. In *Proc. of The 25th European Conference on Information Retrieval Research (ECIR'03)*, 2003.

[28] Clarkson, P. R. and Rosenfeld, R. Statistical Language Modeling Using the CMU-Cambridge Toolkit. In *Proceedings ESCA Eurospeech*, 1997.

[29] Li, J. and Wang, J. Z. Automatic Linguistic Indexing of Pictures by a Statistical Modeling Approach. IEEE Trans. Pattern Anal. Mach. Intell. 25, 9 (Sep. 2003), 1075-1088. 2003.

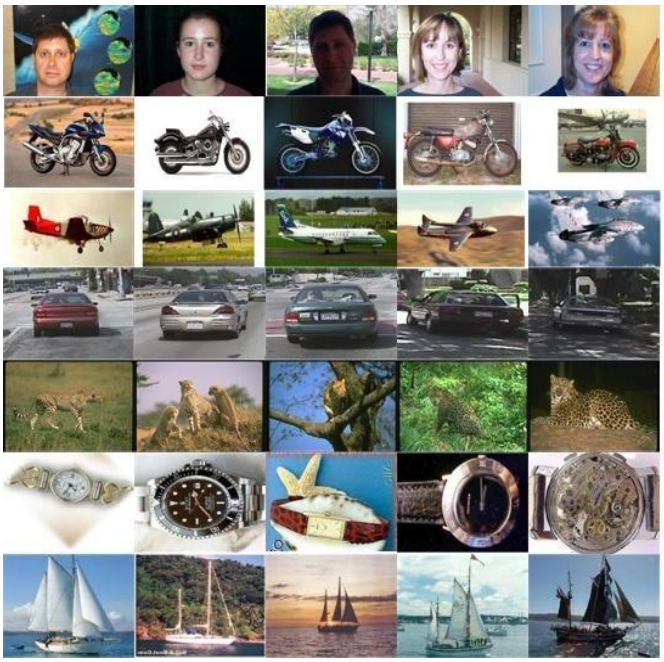


Figure 4. Illustration of Caltech dataset (Dataset 1)

Figure 5. Illustration of Corel dataset (Dataset 2)