# Mining GPS Traces and Visual Words for Event Classification

Junsong Yuan[1], Jiebo Luo[2], Henry Kautz[3], Ying Wu[1]

[1] EECS Dept., Northwestern University
2145 Sheridan Road
Evanston, IL, USA, 60208

[2] Kodak Research Labs
1999 Lake Avenue
Rochester, NY, USA,14650

[3] Department of Computer Science
University of Rochester
Rochester, NY, USA,14627

{j-yuan, yingwu}@northwestern.edu jiebo.luo@kodak.com kautz@cs.rochester.edu

## ABSTRACT

It is of great interest to recognize semantic events (*e.g.*, hiking, skiing, party), in particular when given a collection of personal photos, where each photo is tagged with a timestamp and GPS (Global Positioning System) information at the capture. We address this emerging multi-class classification problem by mining informative features derived from traces of GPS coordinates and a bag of visual words, both based on the entire collection as opposed to individual photos. Considering that semantic events are best characterized by a compositional description of the visual content in terms of the co-occurrence of objects and scenes, we focus on mining feature-combinations (equivalent to word combinations) that have better discriminative and descriptive abilities than individual features for improved event classification. In order to handle the combinatorial complexity in discovering such compositional features, a novel data mining method based on frequent itemset mining (FIM) is proposed. Complementary features are also derived from GPS traces and mined to characterize the underlying *movement patterns* of various event types. Upon feature mining, we perform multi-class AdaBoost to solve the multiclass problem. Based on a dataset of eight event classes and a total of more than 3000 geotagged images from 88 events, experimental results using leave-one-out cross validation have shown the synergy of all of the components in our proposed approach to event classification.

## Categories and Subject Descriptors

H.2.8 [**Database Management**]: Database Applications—*Data mining, Image databases*

## General Terms

Algorithms, Performance

## Keywords

event categorization, image data mining

## 1. INTRODUCTION

Personal photo collections are pervasive. Mining semantically meaningful information from such collections has been an area of active research in machine learning and computer vision communities. Most existing systems perform image labeling on individual images. However, images are often not independent of each other due to the temporal and spatial correlation among the images that belong to the same event. More specifically, in personal image collections, there is rich contextual information besides the image features, and such contextual information is usually complementary to the image features for the purpose of semantic understanding. When GPS sensors are installed in digital cameras, we can collect the following information for each individual image: (1) low-level visual features: e.g., color and edge histograms, (2) timestamp and GPS information: time, latitude, and longitude of the photo capture, (3) semantic object and scene recognition: output from object detectors (*e.g.*, faces) and image classifiers (indoor/outdoor detector, beach detector, etc.), (4) collective context information: images taken at the same time and same place. With extra information available besides the visual features, it becomes possible to improve the visual event recognition by considering all of the available information and all of the related images together.

Semantic event recognition requires a compositional description of the time, place, subject, and event. Simple individual features (*e.g.*, color or texture) provide limited representation and discrimination power and thus usually fail to generate reliable classification results. In order to provide a comprehensive description for events, we are interested in mining compositional features that possess rich representation power for event classification. By allowing different event classes to share common features, we employ a compositional representation instead of individual features for classification. For example, both events "hiking" and "beach" may contain "blue sky" scene. Although with only the observation of "blue sky" scene it is difficult to tell whether it is "hiking" or "beach," "blue sky" is still an informative feature when combined with other features appropriately. For instance, "blue sky" and "sea" together may imply a "beach" event while "blue sky" and "mountain" together may imply a "hiking" event.

To build an effective classification system based on compositional features, one must address the following issues:

1. *how to evaluate compositional features?*
   Although efficient data mining methods exist for discovering frequent patterns and extracting features for classification tasks [2] [3] [17] [1], frequent patterns

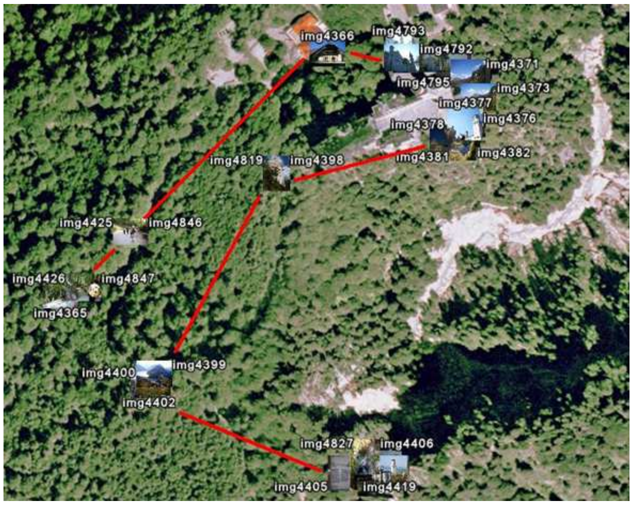


**Figure 1: A collection of geotagged photos: connecting the images creates a trace (red line).**

such as frequent itemsets may not always be discriminative features that are useful for classification [19]. It is not uncommon that a discovered frequent pattern appears in both positive and negative training samples, and thus its discrimination power is limited. Therefore, a better criterion for selecting compositional features beyond its frequency is needed.

2. *how to find good compositional features efficiently?* Mining compositional features is difficult due to its combinatorial complexity. Previous data mining methods apply the idea of branch-and-bound to speed up the search process. However, with a classic data mining technique such as frequent itemset mining (FIM), it is nontrivial to select appropriate data mining parameters such as the support (frequency) and confidence (discrimination) of the patterns that are useful for classification.

3. *how to apply the discovered compositional features for final classification?* After compositional features (rules) are discovered, we need to further combine them or select the most reliable ones for classification. This is also a challenging problem due to the imperfectness of each individual rule and the high redundancy among the rules [10].

We address the above challenges by first showing under what conditions frequent patterns can serve as good features and then present a general guidance for selecting data mining parameters in order to discover the best features efficiently. Specifically, we show that a good compositional feature suitable for classification should not only appear frequently, but also possess good prediction ability. We thus require compositional features to be both (1) descriptive (*i.e.* high frequency in training data), and (2) discriminative (*i.e.* high accuracy in classification). An efficient mining scheme using FIM is proposed to efficiently discover the desired compositional features. Most importantly, we show that the discovered compositional features have guaranteed classification ability in terms of the bounded training error. This theoretical foundation gives us the guidance in selecting appropriate parameters (*e.g.*, support and confidence) for data mining.

For the final event classification, we employ a confidence-based fusion method to combine the GPS and visual prediction results, which are obtained through multiclass AdaBoost based on the corresponding mined features.

The main contributions of this work are many fold. First, we address semantic event recognition at photo collection level instead of individual photos. Second, we propose to apply compositional visual word features over the entire single-event photo collection for multiclass recognition and use a novel mining procedure to find these features efficiently. We also prove theoretic training error bound of the discovered compositional features. Third, we employ GPS traces to characterize the underlying movement patterns of various event type features, again using the mining procedure over the entire collection. Finally, we utilize confidence-based fusion to produce the final event classification based on both visual and movement cues. To give a concrete example of our visual event, We show a collection of geotagged photos in Fig. 1, where a trace is created by connecting all of the images in the temporal order.

## 2. RELATED WORK

There is a large body of work focusing on problems of object recognition, for instance, detecting objects of certain types such as faces, cars, grass, water, sky, and so on. Most of the work relies on using low-level vision features (such as color, textures and edges) available in the image. In recent years, there has been an increasing interest in extracting semantically more complex information such as scene detection and event recognition [16] [4] [14] [15]. For example, one might want to cluster pictures based on if they were taken outdoors or indoors, or separate work pictures from leisure pictures. The solution to such problems primarily relies on derived features such as people present in the image, the presence or absence of certain kinds of objects in the image, and so on. This line of research aims to revolutionize the way people perceive the digital photo collection - from a group of pixel values to complex and meaningful objects, scenes, and events that can be queried or automatically organized in ways that are meaningful to the user. In all of the aforementioned prior art, traditional image clustering and classification is performed on individual images using image-based features only, for example, color and edge histograms, or "bag of visual features" [8]. Existing systems have attempted to recognize events through visual classification of scenes and objects. For example, [9] reported moderate success in recognizing a number of peculiar sports events, such as polo, rowing, and boche, thanks to the unique visual characteristics that can be observed from pictures of such events. However, for other event types that are more relevant to people's lives, such as sightseeing, hiking, playing on the beach, and wedding, it is difficult to obtain satisfactory clustering and classification results solely based on those image features due to the well-known "semantic gap" between low-level features and high-level semantic concepts.

## 3. OVERVIEW

We formulate the event recognition as a $K$-class classification problem. Each individual event $\mathbf{E}$ contains a collection of images with GPS tags: $\mathbf{E}_i = \{\mathcal{G}_j, I_j\}_{j=1}^{|\mathbf{E}_i|}$, where $I_j$ denotes an individual image; $\mathcal{G}_j = \{x_j, y_j, t_j\}$ is the GPS tag

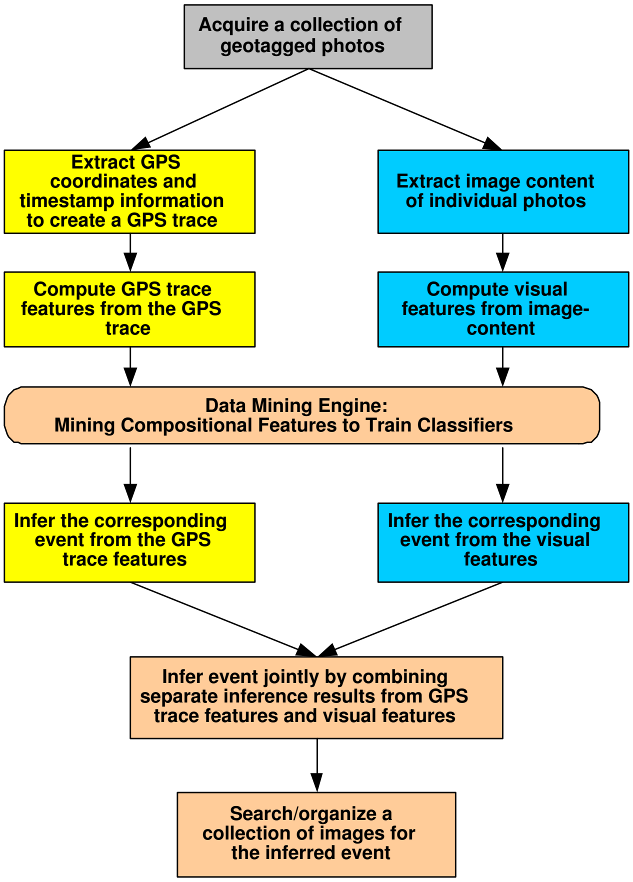

Acquire a collection of geotagged photos
Extract GPS coordinates and timestamp information to create a GPS trace
Extract image content of individual photos
Compute GPS trace features from the GPS trace
Compute visual features from image-content
Data Mining Engine: Mining Compositional Features to Train Classifiers
Infer the corresponding event from the GPS trace features
Infer the corresponding event from the visual features
Infer event jointly by combining separate inference results from GPS trace features and visual features
Search/organize a collection of images for the inferred event

**Figure 2: The flowchart of the proposed method.**

recording the spatial location where $x_j$ is the latitude, $y_j$ is the longitude, and $t_j$ is the time stamp of the image $I_j$.

After feature extraction of each event $\mathbf{E}$, the classification problem can be defined as follows. Suppose we are given a training dataset containing in total $N$ events of $K$ classes: $\mathcal{D}_N = \{\mathbf{X}_t, C_t\}_{t=1}^N$, where $\mathbf{X}_t \in \mathbb{R}^P$ denotes the GPS or visual features describing $\mathbf{E}_t$ and $C_t \in \{1, 2, ..., K\}$ is the label of $\mathbf{X}_t$. The task is to find a classifier $\mathbf{g}(\cdot) : \mathbf{X} \to C$ from the training data, such that given a new query event $\mathbf{X}$, we can assign it a class label $C \in \{1, 2, ..., K\}$. In the experiment, we predefine 8 event classes: $C$ = {hiking, wedding, city-tour, ball-game, backyard, beach, ski, road-trip}.

We explain the architecture of our method in Fig. 2. The whole system contains 3 major components

1. Compositional Feature Mining
   Given a pool of primitive features $\mathbf{\Omega} = \{\mathbf{f}_i\}$, $\mathbf{f}_i$ is a binary classification rule, our goal is to discover *compositional feature* $\mathcal{F} = \{\mathbf{f}_i\} \subset \mathbf{\Omega}$ for multiclass classification. Each $\mathcal{F}$ is a combination of primitive rules $\mathbf{f}_i\}$ and performs better than individual $\mathbf{f}_i\}$.

2. Multiclass AdaBoost for Classification
   After compositional features are discovered, we linearly combine them through multiclass AdaBoost.

3. Fusion of GPS and Visual Classification
   After pattern classification using GPS and visual features separately, we combine the two modalities through confidence-based fusion.

# 4. FEATURE EXTRACTION

## 4.1 Visual Feature Extraction

To extract the visual features of a single-event collection $\mathbf{E}$, first, for each image $I_j \in \mathbf{E}$, if its size is larger than $200,000$ pixels, we scale its size down to $200,000$ pixels. Second, based on the resized image, we extract image grids of fixed-size $16 \times 16$ with sampling interval $8 \times 8$ (such that the neighboring blocks overlap by 50%). Hence a typical image in our dataset can generate $117 \times 87$ such grids. Then for each grid, we extract both color and texture features. For the color features, we partition an image grid into $2 \times 2$ equal size subgrids. For each sub-grid, we extract the mean R, G, and B values to form a $4 \times 3$ feature vector, which characterizes the color information. For the texture features, we apply the SIFT descriptor to describe the edge distribution information. Similarly, we apply a $2 \times 2$ array of histograms with 8 orientation bins in each. Thus our SIFT feature for each image grid is of 32-dimension instead of 128-dimension as the original SIFT [11]. Finally, an image grid is presented by a 44-$d$ feature vector containing both color and texture information.

After extracting raw visual features from all grids in the dataset, we apply the "bag of words" method to represent each event. The $K$-means algorithm is used to cluster the color and SIFT features, respectively, where we obtain two visual vocabularies $\mathbf{\Delta}^c$ and $\mathbf{\Delta}^t$. In our experiments, we set both vocabularies of size 500, thus the combined vocabulary $\mathbf{\Delta} = \mathbf{\Delta}^c \cup \mathbf{\Delta}^t$ contains 1000 visual words. By accumulating all of the grids in an event (a collection of images), we obtain two normalized histograms for an event, $\mathbf{X}^c$ and $\mathbf{X}^t$, corresponding to the word distribution of color and texture vocabularies, respectively. Concatenating $\mathbf{X}^c$ and $\mathbf{X}^t$, we end up with a normalized word histogram: $\sum_{i=1}^{1000} \mathbf{X}(i) = 1$. Each bin $\mathbf{X}(i)$ in the histogram indicates the occurrence frequency of the corresponding word.

## 4.2 GPS Trace Feature Extraction

Based on the GPS coordinates recorded for each individual image, an event is represented by a GPS-time trace: $\mathbf{T}_i = \{(x_j, y_j, t_j)\}_{j=1}^{|\mathbf{T}_i|}$. Given such a GPS-time trace, we extract 11 temporal and 11 spatial features to characterize the trace. We call these 22 GPS features **structure similarity features**.

In terms of temporal features, we obtain a sequence of timestamps $\{t_j\}$ and extract statistical features including entropy, total duration, variance, skewness, and kurtosis of $\{t_j\}$. Furthermore, we utilize several features to characterize the photo-taking behavior: (1) the maximum and median time between two photos; (2) the maximum and median distance between two photos; and (3) the maximum and median traveling speed between two photos.

To extract additional spatial features, we first perform principal component analysis (PCA) over a collection of points $\{x_i, y_i\}$ in the spatial domain in order to take out the orientation variability. This is necessary normalization because the absolute direction of movement should not matter. Based on the normalized distribution of the 2-$d$ points after PCA, we extract seven statistical features, which include the entropy of the 2-$d$ distribution, two variances in the two dimensions determined by PCA, two range values (long and short axes) in the two dimensions, and the product (area) and ratio (aspect) of the two eigenvalues. To distinguish whether the trace is circular movement or linear movement, we calculate the distance of each point to the center (the average of all points). We then calculate the entropy of this distance distribution. A small value of the entropy indicates the event undergoes a circular movement. Finally, we also compute the trace length, average moving speed, and the area of the covered spatial extent.

It is interesting to notice that even without checking the image content, such GPS trace features can help distinguish some events, as qualitatively summarized in Table 1. Furthermore, Fig. 3 shows sample traces of hiking and city-tour events, respectively. Although these two types of events have similar characteristics in terms of covered spatial range, they are still distinguishable due to different trace shapes. We notice that hiking usually has a more random pattern in the spatial domain, where city tours may generate piecewise linear traces due to the constraint of the streets.

**Table 1: How GPS traces can help distinguish different types of events.**

| GPS Trace feature | Event classes |
|---|---|
| large spatial ranges | city-tour, hiking, road-trip |
| small spatial ranges | backyard, beach,<br>ball-game, wedding |
| high moving speed | road trip |
| low or medium moving speed | city-tour, hiking, backyard,<br>beach, ball-game, wedding |

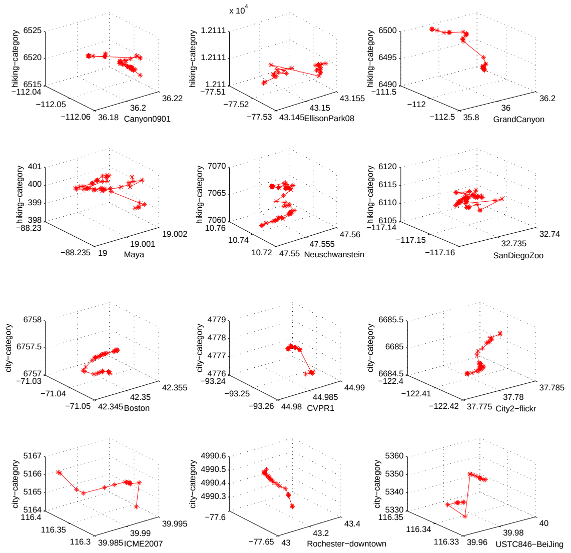


**Figure 3: GPS traces of hiking (1st and 2nd rows) and city-tour events (3rd and 4th rows). Each row shows 3 event traces.**

# 5. FEATURE DISCOVERY

## 5.1 From Feature Vectors to Transactions

Instead of using the word histogram feature $\mathbf{X}$ directly to estimate the event class $C$, we consider a collection of induced binary features, where each $\mathbf{f}_i : \mathbf{X} \to \{0,1\}$ is a *feature primitive* associated with an individual histogram bin $\mathbf{X}(i)$. For example, $\mathbf{f}_i$ can be a decision stump:

$$\mathbf{f}_i(\mathbf{X}) = \begin{cases} f_i^+ & if\ \mathbf{X}(i) \geq \theta_i \\ f_i^- & if\ \mathbf{X}(i) < \theta_i \end{cases}, \quad (1)$$

or

$$\mathbf{f}_i(\mathbf{X}) = \begin{cases} f_i & if\ \mathbf{X}(i) \geq \theta_i \\ \emptyset & if\ \mathbf{X}(i) < \theta_i \end{cases}, \quad (2)$$

when only positive response is considered. Here $\mathbf{f}_i(\mathbf{X})$ indexes whether the $i_{th}$ word appears frequently in the histogram; $\theta_i \in \mathbb{R}$ is the quantization threshold for $\mathbf{f}_i$. We call $f_i$ the *feature item* associated with the feature primitive $\mathbf{f}_i$.

Suppose $\mathbf{X}$ contains $P$ words, if only positive responses are indexed, we have a *item vocabulary* $\mathbf{\Omega} = \{f_1, f_2, ..., f_P\}$ containing $P$ items; otherwise $\mathbf{\Omega} = \{f_1^+, f_1^-, ..., f_P^+, f_P^-\}$ containing $2P$ items. For each word distribution $\mathbf{X}$, we can generate a transaction representation:

$$\mathcal{T}(\mathbf{X}) = \{\mathbf{f}_1(\mathbf{X}), \mathbf{f}_2(\mathbf{X}) ..., \mathbf{f}_P(\mathbf{X})\} \subseteq \mathbf{\Omega},$$

according to the responses of $P$ feature primitives. The induced transaction dataset $\mathbf{T} = \{\mathcal{T}_t\}_{t=1}^N$ contains a collection of $N$ training samples, where each $\mathcal{T}$ corresponds to a data sample $\mathbf{X}$. By transforming continuous features $\mathbf{X} \in \mathbb{R}^P$ into discrete transactions, we can perform a traditional data mining algorithm, such as frequent itemset mining [6].

## 5.2 Mining Compositional Rules

For each feature primitive $\mathbf{f}_i$, we can use it to predict the class label. A *primitive classification rule* is thus in the form:

$$\mathbf{f}_i(\mathbf{X}) = f_i \qquad \Longrightarrow \qquad \hat{C}(\mathbf{X}) = k,$$

where $k \in \{1, 2, ..., K\}$ when considering $K$-class problem, and $\hat{C}(\mathbf{X})$ is the predicted label of $\mathbf{X}$. Since a classification rule based on an individual $\mathbf{f}$ is usually of low accuracy, it is our interest to find compositional feature $\mathcal{F} = \{f_i\} \subseteq \mathbf{\Omega}$, which can be more accurate. We denote by $\mathcal{F}(\mathbf{X}) = k$ a compositional rule for predicting class $k$:

$$\mathcal{F} \subseteq \mathcal{T}(\mathbf{X}) \qquad \Longrightarrow \qquad \hat{C}(\mathbf{X}) = k, \quad (3)$$

An optimal compositional rule $\mathcal{F}^* \subseteq \mathbf{\Omega}$ should have the best discriminative power, such that can minimize the training error. In spite of its clear definition, unfortunately, exhaustive search for $\mathcal{F}^*$ is computationally demanding due to the combinatorial complexity. For example, if each $\mathbf{f}_i$ can generate three possible outcomes: $f_i^+$, $f_i^-$, or $\emptyset$, the total number of all possible compositional rules becomes $3^P$, considering $P$ feature primitives. Thus efficient search methods are required to make the feature selection process computationally feasible. Even worse, such a perfect rule $\mathcal{F}^*$ may not always exist in the case of noisy training data [13], where positive and negative samples are not perfectly separable. In such a case, we need to sacrifice the strict conditions of selecting optimal $\mathcal{F}^*$ for suboptimal ones. In other words, instead of searching for perfect rule $\mathcal{F}^*$, we search for a collection of weaker rules $\mathbf{\Psi} = \{\mathcal{F}_i\}$. To this end, we follow the compositional rule definition in [18].

Definition 1. *($\lambda_1$, $\lambda_2$)*-**compositional rule**
*A compositional rule $\mathcal{F} \subset \mathbf{\Omega}$ is called ($\lambda_1$, $\lambda_2$)-compositional rule if $\exists\ k \in \{1, 2, ..., K\}$, such that:*

$$\begin{array}{rrcl} sup.: & P(\mathcal{F}) & \geq & \lambda_1 \\ conf.: & P(C(\mathbf{X}) = k | \mathcal{F}(\mathbf{X}) = k) & \geq & \lambda_2 \times P(C(\mathbf{X}) = k) \end{array}$$

In the above definition, $\lambda_1, \lambda_2 \in \mathbb{R}$ are parameters, and $P(C(\mathbf{X}) = k)$ denotes the prior distribution of class $k$. Following the terms in data mining literature, we also call $\mathcal{F}$ as a *feature itemset*, which is equivalent to a word combination in the case of our visual vocabulary representation. Given $\mathcal{F}$, the transaction $\mathcal{T}_t$, which includes $\mathcal{F}$, is called an *occurrence* of $\mathcal{F}$, i.e., $\mathcal{T}_t$ is an occurrence of $\mathcal{F}$, if $\mathcal{F} \subseteq \mathcal{T}(\mathbf{X}_t)$. We denote by $\mathbf{T}(\mathcal{F})$ the set of all occurrences of $\mathcal{F}$ in $\mathbf{T}$, and the *frequency* of an itemset $\mathcal{F}$ is:

$$frq(\mathcal{F}) = |\mathbf{T}(\mathcal{F})| = |\{t : \mathcal{F} \subseteq \mathcal{T}(\mathbf{X}_t)\}|.$$

Suppose there are in total $N$ transactions, the first condition in Definition 1 requires that $P(\mathcal{F}) = \frac{frq(\mathcal{F})}{N} \geq \lambda_1$, which is the support requirement in mining frequent patterns [6]. A rule of low support covers few training samples. Such a classification rule has limited ability to generalize, even if it can predict accurately on a low number of training samples. The second condition requires that the rule is accurate enough for prediction, such that most covered samples are correctly classified. This condition corresponds to the confidence of a rule in data mining literature [6]. Different from traditional data mining methods, which usually set a fixed confidence threshold, we take the class prior into consideration to handle unbalanced training data.

As a justification of $(\lambda_1, \lambda_2)$-compositional rule, [18] shows that Definition 1 can be developed into two weak conditions:

$$\begin{aligned} P(\mathcal{F}(\mathbf{X}) = k | C(\mathbf{X}) = k) &\geq \lambda_2 \times P(\mathcal{F}(\mathbf{X}) = k), \quad (4) \\ P(C(\mathbf{X}) = k | \mathcal{F}(\mathbf{X}) = k) &\geq \lambda_2 \times P(C(\mathbf{X}) = k). \quad (5) \end{aligned}$$

Such weak rules in Definition 1 have both descriptive and discriminative power. It is proved in [18] that the training error of $(\lambda_1, \lambda_2)$-compositional rules is bounded by

$$\epsilon_{\mathcal{F}} \leq \frac{1}{\lambda_2} - \lambda_1 \lambda_2 P(C(\mathbf{X}) = k).$$

Besides the guarantee of effectiveness, the other advantage of defining the $(\lambda_1, \lambda_2)$-compositional rules derives from the efficiency. As shown in [18], there is an efficient two-step mining scheme to discover the qualified compositional rules, despite of the combinatorial complexity. First, we perform closed frequent itemset mining algorithm [6] to find candidates $\mathbf{\Psi}' = \{\mathcal{F}\}$, where each $\mathcal{F} \in \mathbf{\Psi}'$ is a frequent pattern that satisfies the first condition in Definition 1, *i.e.*, appearing frequently in the whole dataset. After mining the frequent patterns, we end up with $(\lambda_1, \lambda_2)$-compositional rules $\mathbf{\Psi} \subseteq \mathbf{\Psi}'$ by further checking the prediction accuracy of $\mathcal{F}$, *i.e.* the confidence requirements in Definition 1.

Specifically, the FIM algorithms tackle the combinatorial complexity by using the monotonic property of frequent itemsets, where an infrequent itemset implies infrequent super itemsets. In this paper we apply the FP-growth algorithm to implement closed-FIM [5] for discovering frequent patterns.

# 6. CLASSIFICATION AND FUSION OF GPS AND VISUAL FEATURES

## 6.1 Multiclass AdaBoost

After discovering compositional features, we need to combine them for final classification. Given a $(\lambda_1, \lambda_2)$-compositional rule $\mathcal{F}$ predicting for class $k$, we can easily transfer it to be a $K$-class classifier by randomly guessing the rest of the classes. Formally, its $K$-class classification rule is:

$$\mathcal{F}(\mathbf{X}) = \begin{cases} k & if\ \mathcal{F} \subseteq \mathcal{T}(\mathbf{X}) \\ random & otherwise \end{cases} . \quad (6)$$

Despite its accuracy in predicting class $k$, each $\mathcal{F}$ is still a weak $K$-class classifier. Thus our target is a strong final classifier based on weak classifiers $\mathcal{F}$. Given a pool $\mathbf{\Psi}$, we follow the stagewise additive modeling with exponential loss (SAMME) formulation for multiclass AdaBoost [20], which selects a few $\mathcal{F} \in \mathbf{\Psi}$ and linearly combines them for the final classification.

The target of SAMME is the regression function $\mathbf{g} : \mathbf{X} \rightarrow \mathbb{R}^K$, namely, $\mathbf{g}(\mathbf{X}) = [g_1(\mathbf{X}), ..., g_K(\mathbf{X})]^T$, where $\mathbf{X}$ is the training data. In our formulation, the structure of the regression function is a linear combination of some discovered rules, where $\alpha^m \in \mathbb{R}$ is the weight:

$$\mathbf{g}(\mathbf{X}) = \sum_{m=1}^{M} \alpha^m \mathcal{F}^m(\mathbf{X}). \quad (7)$$

The advantage of SAMME is its consistency with the Bayes classification rule in minimizing the misclassification error [20]:

$$\arg\max_k g_k^*(\mathbf{X}) = \arg\max_k Prob(c = k|\mathbf{X}). \quad (8)$$

Moreover, compared with traditional multiclass boosting such as AdaBoost.MH, which performs $K$ one-against-all classifications, SAMME performs $K$-class classification directly. It only needs weak classifiers better than random guess (for example, correct probability larger than $1/K$), rather than better than $1/2$ as AdaBoost.MH requires. As a result, our discovered $\mathcal{F}$ can naturally satisfy the requirement.

Conceptually, in the high-dimensional feature space $\mathbb{R}^d$, each compositional rule $\mathcal{F}$ covers a *hyper-rectangle* region in $\mathbb{R}^d$. Such a $\mathcal{F}$ is selected because (1) it covers enough training samples, both negative and positive; and (2) it has high prediction accuracy, i.e., most samples in the covered region belong to the same class. These compositional rules (subspace rectangles) will be boosted to approximate the data distribution in $\mathbb{R}^d$. Therefore the classification boundary is piecewise linear in the $d$-dimensional space.

## 6.2 Confidence-Based Fusion

Once individual classifiers are built for each of the GPS trace features and visual features, we can further combine the results of such parallel classification through information fusion. Given a query example $\mathbf{X}$, the multiclass AdaBoost classifier generates a vector output $\mathbf{g}(\mathbf{X})$, which we can conceptually treat as probabilities. To fuse the two output vectors from the GPS and visual modalities, we account for the reliability of each modality through confidence-based fusion. For a learned classifier $\mathbf{g}(\cdot)$, we first estimate its confidence in predicting each of the $K$ classes: $W(k) = P(C = k|\mathbf{g}(\mathbf{X}) = k)$. The weight can be computed based on the confusion matrix of the corresponding modality (GPS trace or visual) of classification obtained through the training phase. Given a query $\mathbf{X}$, suppose its prediction is $\hat{C} = \arg\max_k g(\mathbf{X})$, then we weight its prediction as $W(\hat{C})\mathbf{g}_k(\mathbf{X})$. The fusion of GPS and visual results is:

$$\mathbf{g}(\mathbf{X}) = W^G(\hat{C}^G)\mathbf{g}^G(\mathbf{X}) + W^V(\hat{C}^V)\mathbf{g}^V(\mathbf{X}), \quad (9)$$

where $\mathbf{g}^G(\mathbf{X})$ and $\mathbf{g}^V(\mathbf{X})$ denote the output vector from GPS and visual classifiers, respectively; $\hat{C}^G, \hat{C}^V \in \{1, 2, ..., K\}$ is the prediction from the GPS and visual modality, respectively; $W^G(\hat{C}^G), W^V(\hat{C}^V) \in \mathbb{R}$ are confidence weights.

# 7. EXPERIMENTS

Our goal is to recognize typical events reflected in personal photo collections, where each event corresponds to a specific human activity taking place in a certain environment, and captured by a collection of images taken during the event: $\mathbf{E}_i = \{I_j\}_{j=1}^{|\mathbf{E}_i|}$, where $I_j$ denotes an image. We chose eight types of frequently occurring events with reasonably distinctive visual characteristics, inspired by the tag statistics revealed by Flickr.com: $C = \{$*backyard (including park), ball-game, beach, city-tour, hiking, road-trip, skiing, wedding*$\}$. They include both outdoor and indoor events. In general, event recognition is more challenging and complicated than scene recognition due to the higher semantics involved - the visual content can vary dramatically from one instance to another. Fig. 5 shows some event examples. For each event $\mathbf{E}$, it can be uniquely labeled with one of the eight event classes: $l(\mathbf{E}_i) \in C$. The experimental dataset contains 88 individual events, where each event contains a variable number of 7 to 108 images. There are 3359 images in total in the dataset, where 3126 of them contain geotags. We use all of the images for mining visual features and those having geotags for mining GPS trace features. Due to the limited number of events (because accurate geotagging required a few customized GPS-capable digital cameras unavailable in the market), we perform a leave-one-out test to report all of the results.

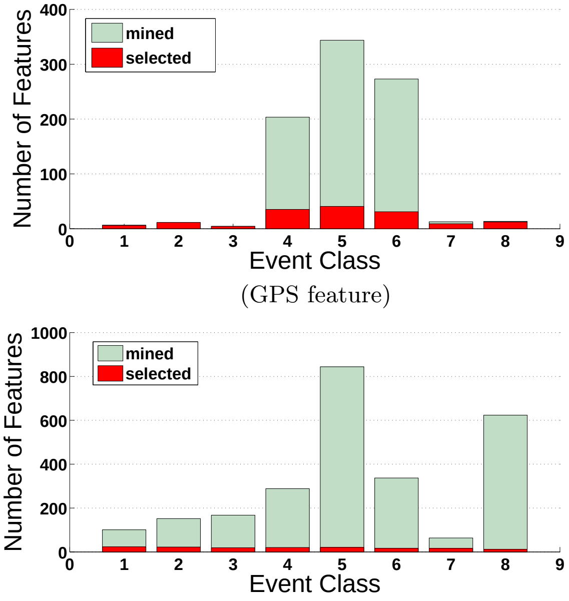
## 7.1 Compositional Feature Discovery

For both GPS and visual features, we set support threshold $\lambda_1 = 0.08 = \frac{1}{2}\max_k r_k$ and the confidence threshold is set as $\lambda_2 = 3$, such that $\lambda_1\lambda_2 = 0.24$. For multiclass AdaBoost, the iteration number of boosting is 150 for both GPS and visual features. Namely our final classifier selects 150 compositional features from the discovered pool $\mathbf{\Psi}$ and linearly combines them using the weights determined through boosting. The quantization parameters $\theta_i$ determine the transactions and thus have large influences on the mining and classification results. To carefully select $\theta_i$, for each feature dimension $\mathbf{X}(i)$, we estimate its mean $\mu_i = E[\mathbf{X}(i)]$ and variance $\sigma^2 = Var[\mathbf{X}(i)]$. We then set $\theta_i = \mu_i + \tau \times \sigma_i$, where $\tau \geq 0$ is a global parameter decided though leave-one-out cross validation. For the GPS features, considering there are only 22 features, we select $\tau = 0$ and apply standard decision stumps as in Eq. 1. For the visual features, we apply the simplified decision stumps (Eq. 2), which consider only positive responses to generate transactions for data mining. By considering only positive responses in each decision stump, we only apply a positive rule when a word (or word combination) appears frequently enough in the event, while negative rules are ignored. We select $\tau = 1.2$ ($\theta_i = \mu_i + 1.2\sigma_i$) through cross-validation.

Since the discovered compositional feature $\mathcal{F}$ can contain an arbitrary number ($\leq |\mathbf{\Omega}|$) of items (or visual words), we analyze the order distribution of the mined feature pool $\mathbf{\Psi}$ in Table 2. It is interesting to notice that all of the discovered and finally selected GPS features are high-order compositional features, where each $\mathcal{F}$ is composed of at least two or more feature primitives ($|\mathcal{F}| \geq 2$). This validates that compositional features are more powerful compared with individual feature primitives. On the other hand, over 80% of the discovered visual features are high-order ones, but they only contribute around $\frac{1}{3}$ in the final committee upon boosting.

**Table 2: Order distribution of the discovered compositional features and the finally used ones. The values are averaged by all of the leave-one-out tests.**

| **Order** | # selected / mined (**G**) | # selected / mined (**V**) |
|---|---|---|
| 1 | **0**/0 | **95.1**/502.9 |
| 2 | **0.9**/1.9 | **40.5**/782.5 |
| 3 | **3.1**/15.4 | **7.0**/488.7 |
| 4 | **10.4**/42.5 | **3.0**/277.7 |
| 5 | **14.3**/68.1 | **1.4**/161.6 |
| 6 | **14.0**/81.2 | **0.9**/89.1 |
| 7 | **12.0**/85.6 | **0.3**/52.9 |
| $\geq 8$ | **95.3**/ 574.2 | **1.8**/221.8 |
| total | **150**/869.0 | **150**/2577.1 |



(GPS feature)

(visual feature)

**Figure 4: Distribution of mined and finally selected compositional features regarding to eight event classes.**

In Fig. 4, we show the feature distributions regarding different event classes. In both GPS and visual cases, the selected compositional features are roughly evenly distributed with respect to the eight classes.

**Table 3: Distribution of visual compositional features: (1) in color modality, (2) in texture (SIFT) modality, and (3) across color and texture (SIFT) modalities. The values are averaged over all of the leave-one-out tests.**

| | **texture** | **color** | **cross** | **total** |
|---|---|---|---|---|
| **# selected** | 79.3 | 69.9 | 0.8 | 150 |
| **# mined** | 1941.7 | 519.7 | 115.7 | 2577.1 |

In the visual mode, since both color and texture features are extracted, the composed $\mathcal{F}$ may fuse feature primitives from these two different modalities. To investigate how color and texture (SIFT) information affect the visual classification, we present in Table 3 the distribution of compositional features in color, texture, respectively, and also across the two modalities. In terms of the number of selected features, both color and texture provide important information for the visual classification. Also, it is interesting to note that although a few compositional features across color and texture modalities are selected, very few of them eventually contribute to the final committee through boosting.

## 7.2 Multiclass AdaBoost

In Table 4, we compare the performance of using only GPS information and only visual information. Visual information produces more reliable results (76.1%) compared with GPS information (39.8%). The confusion matrix represents how are samples of each class (in each row) classified into each of the possible classes (each column). Therefore, a good classifier should have a confusion matrix with most nonzero values concentrated along the diagonal of the matrix.

**Table 4: Boosting compositional features: class confusion matrix of leave-one-out test results. Each row indicates the classification results of the corresponding class. Each element in the table shows the GPS/Visual results. The accuracy of using GPS and visual features is 39.8% and 76.1%, respectively.**

| **G/V** | byd | bgm | bea | cit | hik | rtp | ski | wed |
|---|---|---|---|---|---|---|---|---|
| byd | **6/4** | 1/3 | 1/0 | 0/1 | 0/2 | 0/0 | 0/0 | 2/0 |
| bgm | 3/3 | **2/5** | 2/0 | 1/0 | 0/1 | 0/1 | 0/0 | 2/0 |
| bea | 3/0 | 4/1 | **0/10** | 0/0 | 0/0 | 0/0 | 1/1 | 3/0 |
| cit | 0/0 | 0/0 | 0/0 | **8/14** | 6/0 | 0/0 | 0/0 | 0/0 |
| hik | 0/1 | 0/0 | 0/0 | 6/0 | **7/13** | 0/0 | 0/0 | 1/0 |
| rtp | 0/1 | 0/1 | 0/1 | 0/0 | 0/0 | **11/8** | 0/0 | 0/0 |
| ski | 2/0 | 1/1 | 1/0 | 2/1 | 1/0 | 0/0 | **1/7** | 1/0 |
| wed | 3/1 | 5/2 | 0/0 | 0/0 | 0/0 | 0/0 | 1/0 | **0/6** |

Based on the confusion matrices, we calculate the confidence weights of GPS and visual classifiers in Table 5. We use these weights for the confidence-based fusion (akin to using a Bayesian network [12]).

**Table 5: Confidence regarding different event classes using GPS and visual features.**

| | byd | bgm | bea | cit | hik | rtp | ski | wed |
|---|---|---|---|---|---|---|---|---|
| **G** | 0.35 | 0.15 | 0.00 | 0.47 | 0.50 | 1.00 | 0.33 | 0.00 |
| **V** | 0.40 | 0.38 | 0.91 | 0.88 | 0.81 | 0.89 | 1.00 | 1.00 |

## 7.3 Confidence-Based Fusion

The overall performance of fusing GPS and visual information is presented in Table 6. Road-trip is an example where the GPS trace feature can help the visual feature. Since road trips cover great distances, they have unique trace features compared with the rest of the seven events. However, in terms of visual features, road-trip can be confused with other classes. Skiing is a good example where the visual features help the GPS features. In terms of GPS features, skiing can cover either large or small spatial ranges depending on the actual activity. However, in terms of visual features, it is easy to distinguish skiing from the other events. Overall, the confusion mainly comes from the backyard and ball-game events, which are difficult to distinguish by either GPS or visual features.

**Table 6: Overall performance by fusion GPS and visual results: class confusion matrix of leave-one-out test results. The overall accuracy is 81.8%.**

| **G+V** | byd | bgm | bea | cit | hik | rtp | ski | wed |
|---|---|---|---|---|---|---|---|---|
| byd | **5** | 2 | 0 | 1 | 2 | 0 | 0 | 0 |
| bgm | 3 | **5** | 0 | 0 | 1 | 1 | 0 | 0 |
| bea | 0 | 1 | **10** | 0 | 0 | 0 | 0 | 0 |
| cit | 0 | 0 | 0 | **14** | 0 | 0 | 0 | 0 |
| hik | 0 | 0 | 0 | 0 | **14** | 0 | 0 | 0 |
| rtp | 0 | 0 | 0 | 0 | 0 | **11** | 0 | 0 |
| ski | 0 | 0 | 0 | 2 | 0 | 0 | **7** | 0 |
| wed | 1 | 2 | 0 | 0 | 0 | 0 | 0 | **6** |

To further explain why fusing GPS and vision information can boost the overall performance, we present the detailed analysis of the 88 leave-one-out testing results in Table 7. When both GPS and visual features produce correct results, the final fusion is always correct. Moreover, since visual information is more reliable in our experiments, visual information always helps GPS to get the correct results (41 out of 41 times). On the other hand, five out of nine times, GPS helps out when visual results are wrong, which validates that our confidence-based fusion is effective in allowing weaker predictions from different modalities to reinforce each other. Unsurprisingly, when neither GPS nor visual information is correct, fusion of them could not generate correct results.

**Table 7: Fusion of GPS trace and visual features.**

| correct # / total # | GPS correct | GPS wrong |
|---|---|---|
| visual correct | **26**/26 | **41**/41 |
| visual wrong | **5**/9 | **0**/12 |

A few event examples are presented in Fig. 5 to illustrate how our overall approach outperforms its counterpart that uses only visual information. our confidence-based fusion is effective in allowing weaker predictions from different modalities to reinforce each other (the correct classes are not at the top but close to the top by the visual classifier in these cases). In the backyard example (# 1), children played on the lawn in the backyard. Using visual information alone, the visual classifier misclassified the whole event as ball-game because grass field is also common in ball games. However, with the GPS trace carried by the pictures, the GPS trace classifier is able to recognize that this is a backyard. After information fusion, the final decision is that this event is a backyard because our method learned that it is possible for a backyard event to contain pictures of green grass. In two other examples (#3 and #4), the top class by visual features is backyard, GPS traces help correct the final results (road-trip and hiking) because the events covered a large spatial extent. Another example (#2) is a hiking event where the visual feature actually helps GPS. This hiking event was indeed within a smaller than usual area and thus is classified as wedding according to the GPS trace.

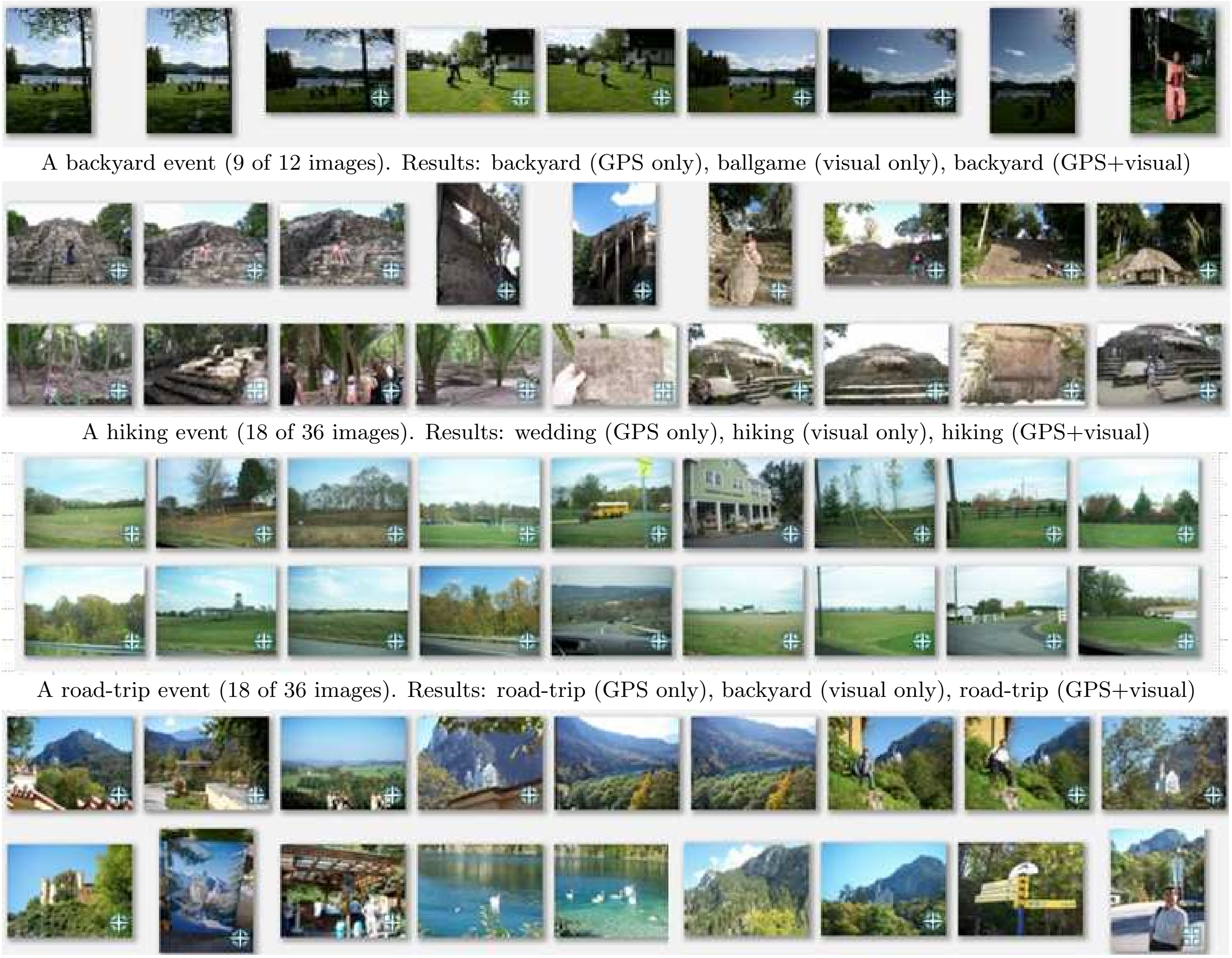
# 8. CONCLUSION AND FUTURE WORK

We present a novel approach to event recognition that utilizes both compositional visual features and GPS trace features that characterize the underlying movement of an event. This approach is effective for personal photo collections that contain geotagged photos. As digital cameras with GPS capability become available in the market and more and more people are geotagging their photos using other means, this approach can potentially revolutionize photo annotations. We are in the process of expanding both the event ontology and dataset in order to fully reap the benefits. We also plan to integrate this work with other brave new ways of managing the ever proliferating digital photos. For example, the elegant work on ZoneTag [7], although designed for single images instead of collections, is an effective way to suggest annotation tags for GPS-capable camera-phones based on the existing community tagging effort.

# 9. REFERENCES

[1] Y. M. Bing Liu, Wynne Hsu. Integrating classification and association rule mining. In *Proc. SIGKDD*, 1998.

[2] H. Cheng, X. Yan, J. Han, and C.-W. Hsu. Discriminative frequent pattern analysis for effective classification. In *Proc. of Intl. Conf. on Data Engineering*, 2007.



A backyard event (9 of 12 images). Results: backyard (GPS only), ballgame (visual only), backyard (GPS+visual)

A hiking event (18 of 36 images). Results: wedding (GPS only), hiking (visual only), hiking (GPS+visual)

A road-trip event (18 of 36 images). Results: road-trip (GPS only), backyard (visual only), road-trip (GPS+visual)

An example of hiking event (18 of 23 images). Results: hiking (GPS only), backyard (visual only), hiking (GPS+visual)

**Figure 5: Example event recognition results. Geotagged images are marked by a compass icon (as in Google Picasa).**

[3] H. Cheng, X. Yan, J. Han, and P. S. Yu. Direct discriminative pattern mining for effective classification. In *Proc. of Intl. Conf. on Data Engineering*, 2008.

[4] S. Ebadollahi, L. Xie, S.-F. Chang, and J. R. Smith. Visual event detection using multi-dimensional concept dynamics. In *Proc. IEEE Conf. on Multimedia Expo*, 2006.

[5] G. Grahne and J. Zhu. Fast algorithms for frequent itemset mining using fp-trees. *IEEE Transaction on Knowledge and Data Engineering*, 2005.

[6] J. Han, H. Cheng, D. Xin, and X. Yan. Frequent pattern mining: current status and future directions. In *Data Mining and Knowledge Discovery*, 2007.

[7] L. Kennedy, M. Naaman, S. Ahern, R. Nair, and T. Rattenbury. How flickr helps us make sense of the world: Context and content in community-contributed media collections. In *Proc. ACM Multimedia*, 2007.

[8] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: spatial pyramid matching for recognizing natural scene categories. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 2006.

[9] L.-J. Li and L. Fei-Fei. What, where and who? classifying events by scene and object recognition. In *Proc. IEEE Intl. Conf. on Computer Vision*, 2007.

[10] D. Liu, G. Hua, , P. Viola, and T. Chen. Integrated feature selection and higher-order spatial feature extraction for object categorization. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 2008.

[11] D. Lowe. Distinctive image features from scale-invariant keypoints. *Intl. Journal of Computer Vision*, 2004.

[12] J. Luo and M. Boutell. Automatic image orientation detection via confidence-based integration of low-level and semantic cues. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 27(5):715–726, 2005.

[13] T. Mitchell. *Machine Learning.* McGraw Hill, 1997.

[14] L. Xie and S.-F. Chang. Pattern mining in visual concept streams. In *Proc. IEEE Conf. on Multimedia Expo*, 2006.

[15] D. Xu and S.-F. Chang. Visual event recognition in news video using kernel methods with multi-level temporal alignment. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 2007.

[16] R. Yan, J. Tesic, and J. R. Smith. Model-shared subspace boosting for multi-label classification. In *Proc. ACM SIGKDD*, 2007.

[17] X. Yin and J. Han. Cpar: classification based on predictive association rules. In *SIAM International Conference data mining (SDM)*, 2003.

[18] J. Yuan, J. Luo, and Y. Wu. Mining compositional features for boosting. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 2008.

[19] J. Yuan, Y. Wu, and M. Yang. From frequent itemsets to semantically meaningful visual patterns. In *Proc. ACM SIGKDD*, 2007.

[20] J. Zhu, S. Rosset, H. Zou, and T. Hastie. Multi-class adaboost. *Technique Report*, 2005.