VISUAL SALIENCY WITH SIDE INFORMATION

Wei Jiang[1], *Lexing Xie*[2], *Shih-Fu Chang*[1]

[1] Columbia University, New York, NY [2] IBM T.J. Watson Research Center, Hawthorne, NY

ABSTRACT

We propose novel algorithms for organizing large image and video datasets using both the visual content and the associated side-information, such as time, location, authorship, and so on. Earlier research have used side-information as pre-filter before visual analysis is performed, and we design a machine learning algorithm to model the join statistics of the content and the side information. Our algorithm, Diverse-Density Contextual Clustering (D_2C_2), starts by finding *unique* patterns for each sub-collection sharing the same side-info, e.g., scenes from winter. It then finds the *common* patterns that are shared among all subsets, e.g., persistent scenes across all seasons. These unique and common prototypes are found with Multiple Instance Learning and subsequent clustering steps. We evaluate D_2C_2 on two web photo collections from Flickr and one news video collection from TRECVID. Results show that not only the visual patterns found by D_2C_2 are intuitively salient across different seasons, locations and events, classifiers constructed from the unique and common patterns also outperform state-of-the-art bag-of-features classifiers.

Index Terms— Pattern clustering methods, Image classification

1. INTRODUCTION

The proliferation of digital photos and videos calls for effective tools for organizing them. Such organization usually makes use of two types of information: the visual content, e.g., pixels or pixel sequences in images and videos; and media meta data, such as the time, location, tags, or source of capture, which often comes attached to the media file and provides additional information other than what is in the image – the latter is often dubbed as *side information*. This paper focuses on the joint analysis of visual content and side information in image/video collections, so as to discover meaningful visual themes, improve automatic recognition and help browsing and navigation of large collections.

The use of content and side information have been explored by prior work. Meta-tags alone can reveal semantic sub-clusters, such as the three distinct meanings of *apple* as "fruit, Mac computer, or New York City/Big Apple" revealed by Flickr tag co-occurrences [1]. Significant recent efforts are also devoted to using visual content alone for recognition, leading to large-scale public research benchmarks [2]. Recent work in web image analysis has effectively leveraged both visual content and side information. There, text keywords [3] or geotags [4] are used to pre-filter a much larger collection, and then an image analysis step performs either keyword annotation with region analysis [3], or landmark discovery using interest point linking [4]. Note, however, that the use of visual content and side information is essentially disjoint in these existing systems. Furthermore, existing work handles only one type of side information, e.g., tag, or location, and there is no consistency exploited across different sub-slices of the collection, e.g., [4] discovers the appearance of the golden gate bridge, but does not recognize similar bridges elsewhere.

We propose an algorithm to discover salient patterns from joint statistics of visual content and side information. The proposed algorithm is flexible enough to handle many types of side information simultaneously (e.g. time, location, author, source, tags). It also exploits both the distinctions and the consistencies among diverse sub-collections (e.g. tennis courts around the world). An overview of our approach is shown in Fig.1. We first partition a topic-based broad collection into subsets, according to different values in side information, e.g. date and/or location of tennis Grand Slams that naturally form four different events: Wimbledon, US Open, French Open and Australian Open. We propose a *Diverse-Density Contextual Clustering* (D_2C_2) algorithm that analyzes the statistics within and across different subsets and finds the unique and common visual patterns on image regions. D_2C_2 can be seen as an extension of the well-known Multiple Instance Learning algorithm for comparing collections separated by side information with multiple features. The resulting clusters intuitively reveal distinct visual patterns in the collection, e.g., tennis courts versus audience in Fig.1. We also successfully use D_2C_2 patterns as visual codebooks for semantic classification. Our experiments show that codebook constructed with D_2C_2 outperforms existing content-only approaches for 0.1∼0.15 in average precision.

In the rest of this paper, Sec.2 describes the D_2C_2 algorithm in detail, Sec.3 presents two evaluations on three datasets, and Sec.4 concludes this work.

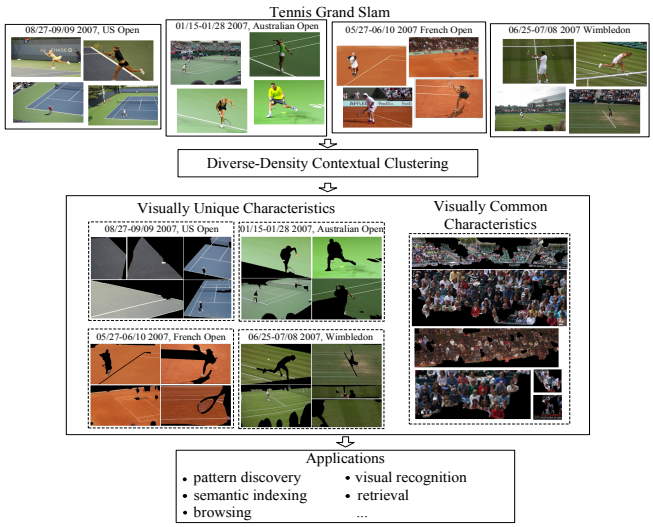

Fig. 1. Visual pattern discovery overview.

2. DISCOVER UNIQUE AND COMMON PATTERNS WITH DIVERSE-DENSITY CONTEXTUAL CLUSTERING

Let $\mathcal{X}$ denote a large data set containing L images $x_1, \ldots, x_L$, each has associated side information $y_1, \ldots, y_L$. We start by segmenting

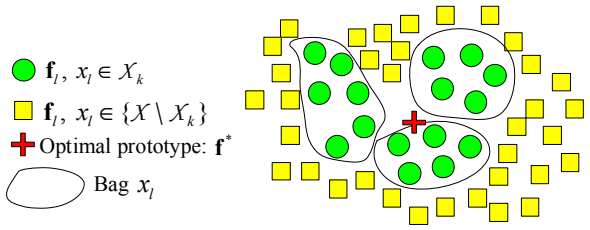

Fig. 2. Diverse Density prototype learning (see Sec 2.1).

an image x into N different regions based on color and texture coherence, i.e. $x = \{r_1, \ldots, r_N\}$. For each region r we extract M different visual features measuring its color, texture and shape statistics, denote $r = [\mathbf{f}_1, \ldots, \mathbf{f}_M]^T$, where each feature $\mathbf{f}_h$ is a d_h-dimensional vector, i.e., $\mathbf{f}_h \in \mathbb{R}^{d_h}$. We assume that the side information variable y takes one among K discrete values, $y_l \in \{1, \ldots, K\}$, e.g., possible cross-products of a finite set of dates, locations and/or broadcast channels. We then partition the image set $\mathcal{X}$ into K subsets based on the values of y, i.e., $\mathcal{X} = \mathcal{X}_1 \cup \ldots \cup \mathcal{X}_K$, and $y_l = k$ if $x_l \in \mathcal{X}_k$.

The goal of our algorithm is to find "patterns", i.e. representative regions r and their corresponding feature points $\mathbf{f}$ in their respective feature spaces, that are *unique* for each subset $\mathcal{X}_k$, or *common* among all subsets $\mathcal{X}_1, \ldots, \mathcal{X}_K$. *Uniqueness* and *commonality* can be thought of as a statistical *diff* operator on image collections, and therefore can reveal what are *persistent*, *new*, or *fading* among subcollections. And the $\mathrm{D_2C_2}$ is one method for achieving this.

The $\mathrm{D_2C_2}$ algorithm consists of three steps: (1) Learn unique pattern prototypes for each of the K subsets on each feature using the diverse-density criteria with their contextual side-information y; (2) Learn unique patterns for each subset by fusing information from all M features; (3) Learn common patterns among all K subsets based on the results of unique patterns. Finally, each image region is mapped into either a *unique* or a *common* codeword learned above, and the cookbook is in turn used for visualization and classification.

2.1. Learning unique prototypes

We treat each image x_l, $l \in \{1, \ldots, L\}$ as a "bag-of-regions" $\{r_{l1}, \ldots, r_{lN_l}\}$. To learn the unique patterns for a subset $\mathcal{X}_k$, it is natural to introduce an auxiliary variable $\tilde{y}_l$ that turns "on" for images in subset $\mathcal{X}_k$, and "off" otherwise. i.e., $\tilde{y}_l = 1$ iff. $y_l = k$, and $\tilde{y}_l = -1$ if $y_l \neq k$. Note that $\tilde{y}$ are labels over bags x rather than over instances r. For a "positive" image bag $x_l \in \mathcal{X}_k$, it is sensible to have at least one of its instances being "unique" to subset $\mathcal{X}_k$, i.e. similar to other unique instances in other positive bags, and dissimilar to all instances in all "negative" bags $\mathcal{X} \setminus \mathcal{X}_k$. As illustrated in Fig. 2. This formulation is known as Multiple Instance Learning (MIL) [5] in the literature, and here we repeat an MIL-type procedure K-times in order to obtain the unique pattern prototypes $(\mathbf{f}^*_{kh}, \mathbf{w}^*_{kh})$, $k = 1, \ldots, K$, consisting of a prototype point (or centroid) $\mathbf{f}^*_{kh} = [f^*_{kh1}, \ldots, f^*_{khd_h}]^T$ in the h^{th} feature space for subset $\mathcal{X}_k$, and the corresponding weights for each dimension $\mathbf{w}^*_{kh} = [w^*_{kh1}, \ldots, w^*_{khd_h}]^T$, also of dimension d_h.

Among the flavors of MIL objective functions, the Diverse Density (DD) objective function is one that fits our intuitive objective above and also with efficient inference algorithm available [6] via expectation-maximization (EM). In the rest of Sec 2.1, we omit subscripts k, h without loss of generality, as each $\mathbf{f}^*$ will be independently optimized over the different image bags $l \in \{1, \ldots, L\}$ and different instances $j \in \{1, \ldots, N_l\}$ in each bag x_l. The DD objective function for one image bag x_l is simply written as:

$$Q_l = \frac{1+\tilde{y}_l}{2} - \tilde{y}_l \prod_{j=1}^{N_l} (1 - e^{-||\mathbf{f}_{lj} - \mathbf{f}^*||^2_{\mathbf{w}^*}}) \quad (1)$$

where $\mathbf{f}_{lj}$ is the feature vector of region r_{lj}, and $\|\mathbf{f}\|_{\mathbf{w}}$ denotes the weighted 2-norm of vector $\mathbf{f}$ by $\mathbf{w}$, i.e., $\|\mathbf{f}\|_{\mathbf{w}} = (\sum_{i=1}^{d} (f_i w_i)^2)^{\frac{1}{2}}$. For a positive bag x_l, Q_l will be close to 1 when $\mathbf{f}^*$ is close to any of its instances, and Q_l will be small when $\mathbf{f}^*$ is far from all its instances. For a negative bag x_l, Q_l will be large when $\mathbf{f}^*$ is far from all its instances. By aggregating Eq (1) over all bags the optimal $\mathbf{f}^*$ will be close to instances in the positive bags and far from all of the instances in the negative bags.

For each positive image bag $x_l \in \mathcal{X}_k$, there should be at least one instance to be treated as a positive sample to carry the label of that bag. This instance, denoted by $L(x_l)$, is identified as the closest instance to the prototype $\mathbf{f}^*$ and is given by Eq (2). For each negative image bag $x_l \in \{\mathcal{X} \setminus \mathcal{X}_k\}$, on the other hand, all instances are treated as negative samples, whose contribution to Q_l are all preserved.

$$L(x_l) = \arg\max_{j=1}^{N_l} \{\exp[-\|\mathbf{f}_{lj} - \mathbf{f}^*\|^2_{\mathbf{w}^*}]\} \quad (2)$$

This leads to the max-ed version of Eq (1) on the positive bags:

$$Q_l = \begin{cases} e^{-||\mathbf{f}_{lL(x_l)} - \mathbf{f}^*||^2_{\mathbf{w}^*}} & , \ x_l \in \mathcal{X}_k \\ \prod_{j=1}^{N_l} (1 - e^{-\|\mathbf{f}_{lj} - \mathbf{f}^*\|^2_{\mathbf{w}^*}}) & , \ x_l \in \{\mathcal{X} \setminus \mathcal{X}_k\} \end{cases} \quad (3)$$

The DD function in Eq (3) is used to construct an objective function over all bags, which in turn is maximized by an EM algorithm [6]. In the E-step, the aggregate DD function is given by $Q = Q^+ \cdot Q^-$ where Q^+ and Q^- are DD functions for all positive bags and all negative bags, respectively:

$$\begin{aligned} Q^+(\mathbf{f}^*, \mathbf{w}^*) &= \prod_{x_l \in \mathcal{X}_k} e^{-\|\mathbf{f}_{lL(x_l)} - \mathbf{f}^*\|^2_{\mathbf{w}^*}} \\ Q^-(\mathbf{f}^*, \mathbf{w}^*) &= \prod_{x_l \in \{\mathcal{X} \setminus \mathcal{X}_k\}} \prod_{j=1}^{N_l} (1 - e^{-\|\mathbf{f}_{lj} - \mathbf{f}^*\|^2_{\mathbf{w}^*}}) \end{aligned}$$

In the M-step, we use gradient ascent to maximize Q. The update equations for the prototype points and weights are as follows, with learning rate η empirically set to 5×10^{-3}.

$$\begin{aligned} \mathbf{f}^* &\leftarrow \mathbf{f}^* + \eta \frac{\partial \log Q}{\partial \mathbf{f}^*} \\ \mathbf{w}^* &\leftarrow \mathbf{w}^* + \eta \frac{\partial \log Q}{\partial \mathbf{w}^*}, \end{aligned}$$

Define a notation, $\mathrm{Diag}(W)$, which generates a diagonal matrix by using the corresponding diagonal entries from matrix W, i.e. $\tilde{W} = \mathrm{Diag}(W)$ so that entry $\tilde{W}_{ii} = W_{ii}, \forall i$ and $\tilde{W}_{ij} = 0, \forall i, j, i \neq j$. The partial vectors are computed as:

$$\begin{aligned} \frac{\partial \log Q}{\partial \mathbf{f}^*} &= 2\mathrm{Diag}(\mathbf{w}^* \mathbf{w}^{*T}) [\sum_{x_l \in \mathcal{X}_k} (\mathbf{f}_{lL(x_l)} - \mathbf{f}^*) \\ &\quad - \sum_{x_l \in \{\mathcal{X} \setminus \mathcal{X}_k\}} \sum_{j=1}^{N_l} (\mathbf{f}_{lj} - \mathbf{f}^*) \frac{e^{-\|\mathbf{f}_{lj} - \mathbf{f}^*\|^2_{\mathbf{w}^*}}}{1 - e^{-\|\mathbf{f}_{lj} - \mathbf{f}^*\|^2_{\mathbf{w}^*}}}] \\ \frac{\partial \log Q}{\partial \mathbf{w}^*} &= -2\mathbf{w}^* [\sum_{x_l \in \mathcal{X}_k} \mathrm{Diag}\left((\mathbf{f}_{lL(x_l)} - \mathbf{f}^*)(\mathbf{f}_{lL(x_l)} - \mathbf{f}^*)^T\right) \\ &+ \sum_{x_l \in \{\mathcal{X} \setminus \mathcal{X}_k\}} \sum_{j=1}^{N_l} \mathrm{Diag}\left((\mathbf{f}_{lj} - \mathbf{f}^*)(\mathbf{f}_{lj} - \mathbf{f}^*)^T\right) \frac{e^{-\|\mathbf{f}_{lj} - \mathbf{f}^*\|^2_{\mathbf{w}^*}}}{1 - e^{-\|\mathbf{f}_{lj} - \mathbf{f}^*\|^2_{\mathbf{w}^*}}}] \end{aligned}$$

We repeat DD-optimization above from each instance in each positive bag, and prototypes with DD values smaller than a threshold T (that equals to the mean of DD values of all learned prototypes) are excluded. Regions closest to the remaining prototypes are obtained. These optimal prototypes $\mathbf{f}^*$ and the regions close to them together form a pool of candidate regions to represent the unique patterns in data subset $\mathcal{X}_k$ with feature h.

2.2. Unique Pattern Learning with Multiple Features

We learn unique patterns $U_{k1}, \ldots, U_{km_k}$ from the candidate regions using multiple features. Let $r_{h1}, \ldots, r_{hn_h}$ be n_h candidate regions learned for the h^{th} feature. A pairwise similarity matrix S is constructed over all the candidate regions from all different types of features as follows: For each type of feature h, we calculate the pairwise similarities among all the candidate regions $r_{h1}, \ldots, r_{hn_h}$ from this feature type, and we add these similarities $s(r_{hi}, r_{hj})$ to the corresponding entry in the overall similarity matrix by multiplying with a weight v_h (which measures the importance of this feature type). $s(r_{hi}, r_{hj}) = \exp\{-\mathcal{E}(r_{hi}, r_{hj})\}$ where $\mathcal{E}(r_{hi}, r_{hj})$ is the Euclidean distance of r_{hi}, r_{hj}. Spectral clustering [7] is then used on S to cluster all candidate regions from different features into a set of region clusters $U_{k1}, \ldots, U_{km_k}$, essentially re-organizing the set of candidate regions from Sec 2.1

2.3. Learning of Common Patterns

The final step of D_2C_2 involves learning a set of *common* patterns based on the outcome of *unique* patterns. This is done by excluding the regions that belong to all unique pattern sets $\{U_{k1}, \ldots, U_{km_k}\}$, $k = 1, \ldots, K$ and clustering the remaining regions in the entire dataset $\mathcal{X}$, using a clustering algorithm such as K-means. This gives us a set of cluster centers $C_1, \ldots, C_m$ and their member regions, and we treat each C_j as a common pattern for describing characteristics shared across data subsets.

2.4. Vocabulary-based Feature Representation

The unique and common patterns form unique and common visual codebooks respectively, where each unique or common pattern is a unique or common codeword. Using unique patterns as an example, for an image x_l, each segmented region $r_{lj} \in x_l$ can be mapped to each unique codeword U_{ki} by using the maximum similarity between r_{lj} and the prototype regions in the codeword, i.e., $m(r_{lj}, U_{ki}) = \max_{r_{ki} \in U_{ki}}\{s(r_{lj}, r_{ki})\}$. Then image x_l can be mapped to codeword U_{ki} by using the maximum similarity between U_{ki} and all regions in x_l, i.e., $m(x_l, U_{ki}) = \max_{r_{lj} \in x_l} m(r_{lj}, U_{ki})$. That is, the unique codebook spans a feature space for representing each image x_l as a feature vector formed by the mapping score of this image to each codeword $m(x_l, U_{ki})$. Based on this feature vector, classifiers such as SVM can be used for concept classification. Similarly, image x_l can also be mapped to common patterns C_j to form vocabulary-based features with entry $m(x_l, C_j)$.

3. EXPERIMENTAL RESULTS

We examine the outcome of the D_2C_2 algorithm in two ways: visualize the unique and common patterns, and evaluate the resulting codebook for visual concept classification.

The evaluations are carried out on 3 different data sets. (1) 2000 Flickr images retrieved using the keyword "Jefferson Memorial". Using the taken date as the side information, these images are separated into four seasons in 2007: spring, summer, fall and winter. For example, images for spring were taken between 2007-03-01 and 2007-05-31. (2) 2000 Flick images downloaded with query keyword "Tennis". Using the taken dates and location tags, these images are separated into four events: US Open 2007, French Open 2007, Australian Open 2007, and Wimbledon 2007. For example, images for Australian Open were taken between 2007-01-15 and 2007-01-28 in Melbourne, Australia. (3) 6112 keyframes taken from TRECVID 2005 dataset [2]. They were aired in 10 news broadcasts during a one-week period in November 6–12, 2004. The channel information (CCTV4 or NBC) are treated as side-information and the videos are partitioned accordingly. For both of the two Flickr image data sets,

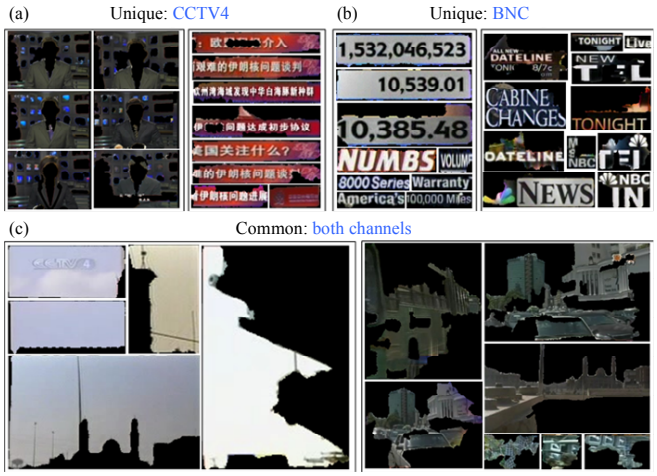

Fig. 4. Unique and common patterns from news videos. (a) and (b): two unique clusters from CCTV4 and NBC, respectively; (c): two common clusters across both channels.

1000 images are randomly selected for training and the rest 1000 images are used for testing. For the TRECVID data set, 5 videos (3009 keyframes) are randomly selected for training and the rest are used for testing. 6 types of low-level visual features are used: 3 color features including color histogram, color correlogram and grid-based color moments; 2 texture features including wavelet texture and tamura texture; and edge direction histogram.

3.1. Visualizing the unique and common patterns

Fig. 3 and Fig. 4 show representative unique and common patterns in the three datasets. We visualize each pattern by looking at a sample of image regions closest to the centroid of the corresponding codebook. A segmentation mask is overlayed on each region segment, with pixels that belong to the region shown in their original color and intensity, while pixels not in the region shown in black.

For example, the unique patterns "Jefferson Memorial" dataset turns out to be the seasonal vegetation changes in the surroundings of the landmark: cherry blossom in the spring, thriving tree leaves in the summer, foliage colors in the fall, and white snows in the winter. The four common patterns in this set are the interior and exterior of the memorial building itself, as well as the blue sky and waterfront segments that are "seasonally-invariant". For the four tennis tournaments in Grand Slam, their different court surfaces are the most salient among the unique clusters, i.e. plexicushion for Australian Open, clay court for French Open, grass for Wimbledon, and deco turf for US Open. As for the common patterns, sponsorship logos around the sport field, the blue sky, and the green stadium ground are shared by most tournament scenes.

For news data, the different channel logos and studio backgrounds are identified as unique, i.e., the CCTV studio setting with a screen wall and their overlay captions with a red backdrop, as well as the NBC logo, different section headings and caption with unique fonts. There was significant coverage on the military actions in Iraq in November 2004, the common patterns across these two channels included the building ruins and sky in these stories.

3.2. Classification Performance

We use a set of binary classification tasks on each dataset to evaluate the effectiveness of codebooks from unique+common patterns. The classes on the two Flickr datasets are custom-designed for this experiment, four for "Jefferson Memorial" and eight for "Tennis". For TRECVID news videos, we selected a subset of 10 generic concepts

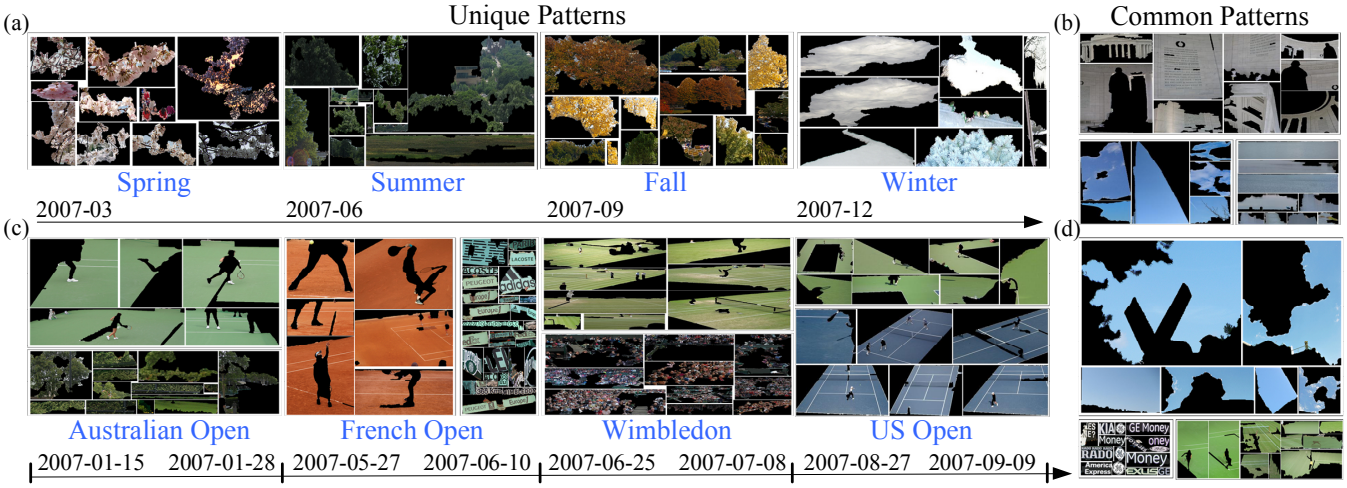

Fig. 3. Unique and common patterns from the Flickr images.

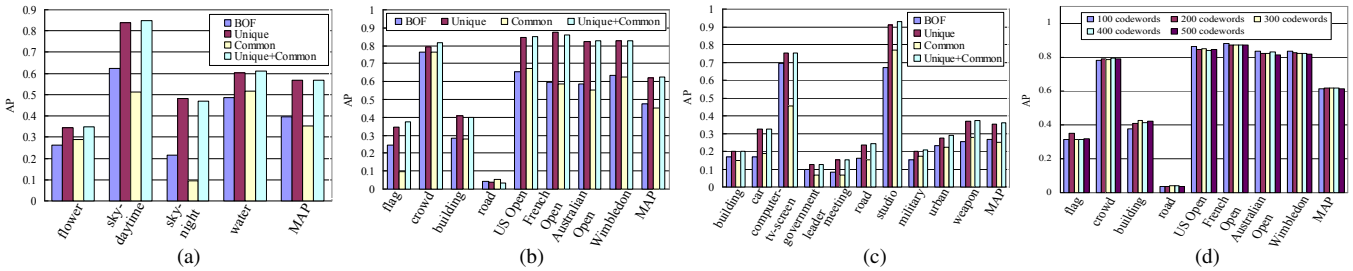

Fig. 5. Per-concept performance over (a) Jefferson Memorial (b) Tennis 2007, and (c) TRECVID datasets. (d) Influence of codebook size on the tennis dataset.

from the LSCOM ontology [8] that has significant presence in the ten newscasts. We generate vocabulary-based codebook representation (Sec. 2.4) from the unique and common patterns, and we also concatenate the two features and dub it as "unique+common" patterns. The codebooks generated by D_2C_2 is compared with a "bag-of-features" (BoF) baseline [9], where a cluster codebook is generated from all regions from the entire data set with the Kmeans algorithm, without any side information. SVMs are trained on these histograms features, and the average precision (AP, area under precision-recall curve) was measured for each concept, and mean-average-precision (MAP) is reported over all concepts [2].

Fig.5 (a–c) gives an overview of concept detection performance in the three data sets, with 200 codewords for each different codebook. We can see that the unique patterns alone outperforms BoF by a margin over all concepts. While the common patterns alone perform much worse, it improves the overall performance when used together with the unique patterns. Fig.5(d) shows the performance variations with respect to the codebook size of the unique codebook. We can see that the performance only undergoes slight change as the size of the codebook varies from 100 to 500. Such results demonstrates that the unique and common patterns not only discover visually meaningful themes, they are also helpful for building better classifiers. Furthermore, there is a large range of codebook sizes where the proposed D_2C_2 algorithm can perform well.

4. CONCLUSION

We propose a D_2C_2 algorithm for visual pattern discovery by joint analysis of visual content and side information. A content collection is partitioned into subsets based on side information, and the unique and common visual patterns are discovered with multiple instance learning and clustering steps that analyzes across and within these subsets. Such patterns help to visualize the data content and generate vocabulary-based features for semantic classification. The proposed framework is rather general which can handle all types of side information, and incorporate different common/unique pattern extraction algorithms. One future work is to improve the generation of common patterns by emphasizing the shared consistencies, instead of the current heuristic clustering. Another future work is to explore other applications using the unique+common patterns, such as those listed in Fig.1.

5. REFERENCES

[1] Flickr: Tagged with apple, retrieved Sept 24, 2008 http://www.flickr.com/photos/tags/apple/clusters.

[2] NIST TRECVID http://www-nlpir.nist.gov/projects/trecvid/.

[3] Y.Q. Sun *et al.*, "Visual pattern discovery using web images", in *ACM Workshop on MIR*, 2006, pp. 127–136.

[4] L. Kennedy and M. Naaman, "Generating diverse and representative image search results for landmarks", in *Int'l Conf. on WWW*, 2008, pp. 297–306.

[5] O. Maron *et al.*, "A framework for multiple-instance learning", *Advances in Neural Info. Proc. Sys.*, 1998, vol. 10.

[6] Y.X. Chen and J.Z. Wang, "Image categorization by learning and reasoning with regions", in *Journal of Mach. Learn. Res.*, 2004, 5:913–939.

[7] A.Y. Ng *et al.*, "On spectral clustering: Analysis and an algorithm", in *Advances in Neural Info. Proc. Sys.*, 2001.

[8] LSCOM lexicon definitions and annotations V1.0, DTO Challenge Workshop on Large Scale Concept Ontology for Multimedia, Columbia Univ. ADVENT Tech. Report, 2006.

[9] W. Jiang *et al.*, "Similarity-based online feature selection in content based image retrieval", in *IEEE Trans. Image Proc.*, 2006, (15)7:702-712.