Short-Term Audio-Visual Atoms for Generic Video Concept Classification

Wei Jiang[1] Courtenay Cotton[1] Shih-Fu Chang[1] Dan Ellis[1] Alexander C. Loui[2]

[1] Electrical Engineering Department, Columbia University, New York NY

[2] Kodak Research Labs, Eastman Kodak Company, Rochester NY

ABSTRACT

We investigate the challenging issue of joint audio-visual analysis of generic videos targeting at semantic concept detection. We propose to extract a novel representation, the Short-term Audio-Visual Atom (S-AVA), for improved concept detection. An S-AVA is defined as a short-term region track associated with regional visual features and background audio features. An effective algorithm, named Short-Term Region tracking with joint Point Tracking and Region Segmentation (STR-PTRS), is developed to extract S-AVAs from generic videos under challenging conditions such as uneven lighting, clutter, occlusions, and complicated motions of both objects and camera. Discriminative audio-visual codebooks are constructed on top of S-AVAs using Multiple Instance Learning. Codebook-based features are generated for semantic concept detection. We extensively evaluate our algorithm over Kodak's consumer benchmark video set from real users. Experimental results confirm significant performance improvements – over 120% MAP gain compared to alternative approaches using static region segmentation without temporal tracking. The joint audio-visual features also outperform visual features alone by an average of 8.5% (in terms of AP) over 21 concepts, with many concepts achieving more than 20%.

Categories and Subject Descriptors

H.3.1 [**Information Storage and Retrieval**]: Content Analysis and Indexing; H.4 [**Information Systems Applications**]: Miscellaneous; H.3.m [**Information Storage and Retrieval**]: Miscellaneous

General Terms

Algorithms, Experimentation

Keywords

Semantic concept detection, joint audio-visual analysis, short-term audio-visual atom, audio-visual codebook

MM'09, October 19–24, 2009, Beijing, China.

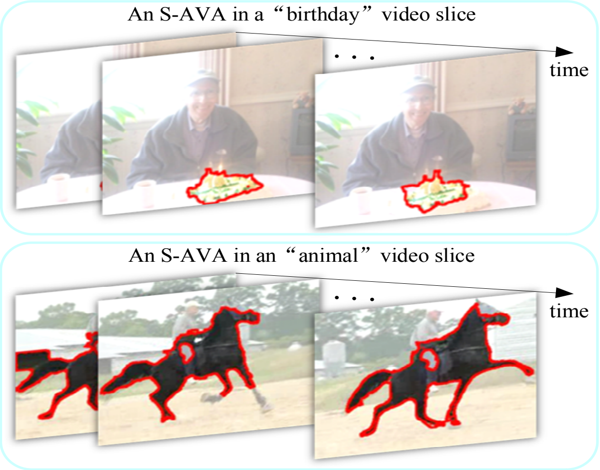

An S-AVA in a "birthday" video slice
time
An S-AVA in an "animal" video slice
time

Figure 1: Examples of S-AVAs. The short-term region track of a birthday cake and the background birthday music form a salient audio-visual cue to describe "birthday" videos. The short-term region track of a horse and the background horse running footstep sound give a salient audio-visual cue for the "animal" concept.

1. INTRODUCTION

This paper investigates the problem of automatic detection of semantic concepts in unconstrained videos, by joint analysis of audio and visual content. These concepts include generic categories, such as scene (*e.g.*, beach, sunset), event (*e.g.*, birthday, wedding), location (*e.g.*, museum, playground) and object (*e.g.*, animal, boat). Unconstrained videos are captured in an unrestricted manner, like those videos taken by consumers on YouTube. This is a difficult problem due to the diverse video content as well as the challenging condition such as uneven lighting, clutter, occlusions, and complicated motions of both objects and camera.

To exploit the power of both visual and audio aspects for video concept detection, *multi-modal fusion* approaches have attracted much interest [1, 6, 35]. Visual features over global images such as color and texture are extracted from image frames, and audio features such as MFCC coefficients are generated from the audio signal in the same time window. In early fusion methods [6, 35], such audio and visual raw features are either directly fused by concatenation to learn classifiers or used to generate individual kernels which are then added up into a fused kernel for classification. In late fusion approaches [1, 6], concept detectors are first trained over audio and visual features respectively and then fused to generate the final detection results. These fusion methods have shown promising results with performance improvements. However, the global visual feature is insufficient to capture the object information, and the disjoint process of extracting audio and visual features limits the ability to generate joint audio-visual patterns that are useful for concept detection. For example, as illustrated in Fig. 1, the joint pattern of a birthday cake region and the birthday music is an intuitive strong audio-visual cue for the "birthday" concept but has never been explored in prior works.

On the other hand, there are many recent works exploring audio-visual analysis for object detection and tracking. In audio-visual speech recognition [17, 19], visual features obtained by tracking the movement of lips and mouths are combined with audio features for improved speech recognition. In audio-visual object detection and tracking [3, 8], synchronized visual foreground objects and audio background sounds are used for object detection [8]. By using multiple cameras to capture the object motion, the joint probabilistic model of both audio and visual signals can be used to help tracking [3]. In audio-visual localization [2, 16], under the assumption that fast moving pixels make big sounds, temporal patterns of significant changes in the audio and visual signals are found and the correlation between such audio and visual temporal patterns is maximized to locate sounding pixels. Such joint audio-visual object tracking methods have shown interesting results in analyzing videos in a controlled or simple environment where good foreground/background separation can be obtained. However, both object detection and tracking (especially for unconstrained objects) are known to be difficult in generic videos. There usually exist uneven lighting, clutter, occlusions, and complicated motions of both multiple objects and camera. In addition, the basic assumption for tight audio-visual synchronization at the object level may not be valid in practice. Multiple objects may make sounds together in a video without large movements, and sometimes the objects making sounds do not show up in the video.

In this work, we investigate the challenging issue of audio-visual analysis in generic videos aiming at detecting generic concepts. We propose a novel representation, *Short-term Audio-Visual Atom* (*S-AVA*), by extracting atomic representations over short-term video slices (*e.g.*, 1 second). We track automatically segmented regions based on the visual appearance within a short video slice and decompose the corresponding audio signal into most prominent bases from a time-frequency representation. Regional visual features (*e.g.*, color, texture, and motion) can be extracted from the short-term region track, which are combined with the audio feature generated from the decomposition bases to form a joint audio-visual atomic representation. Based on S-AVAs, joint *audio-visual codebooks* can be constructed, and the codebook-based features can be used for concept detection.

Our method provides a balanced choice for exploring audio-visual correlation in generic videos: compared to the previous audio-visual fusion approach using coarsely aligned concatenation of global features, we generate a short-term atomic representation in which a moderate level of synchronization is enforced between local object tracks and ambient sounds; compared to the tight audio-visual synchronization framework focusing on object detection and tracking, we do not rely on precise object extraction. Compared with alternative methods using static image frames without temporal tracking, less noisy atomic patterns can be found by the short-term tracking characteristics of S-AVAs. As illustrated by the S-AVA examples in Fig. 1, the temporal region track of a birthday cake associated with the background birthday music gives a representative audio-visual atomic cue for describing "birthday" videos. Similarly, the temporal horse region track together with the horse running footstep sound form a joint audio-visual atomic cue that is salient for describing the "animal" concept. Fig. 1 also indicates that the joint audio-visual correlation captured by our S-AVA is based on co-occurrence, *e.g.*, frequent co-occurrence between a birthday cake and the birthday music. Accordingly, the audio-visual codebooks constructed from salient S-AVAs can capture the representative audio-visual patterns for classifying individual concepts, and significant detection performance improvements can be achieved.

2. OVERVIEW OF OUR APPROACH

Fig. 2 shows the framework of our system. We will briefly summarize our work in this section. More details about the visual and audio processes can be found in Section 3 and Section 4, respectively.

In the visual aspect, we develop an effective algorithm, named *Short-Term Region tracking with joint Point Tracking and Region Segmentation* (*STR-PTRS*), to extract short-term visual atoms from generic videos. STR-PTRS accommodates the challenging conditions in generic videos by: conducting tracking within short-term video slices (*e.g.*, 1 second); getting meaningful regions by image segmentation based on the color and texture appearance; and jointly using interest point tracking and region segments to obtain short-term region tracks. The short-term region tracks are not restricted to foreground objects. They can be foreground objects or backgrounds, or combinations of both, all of which carry useful information for detecting various concepts. For example, the red carpet alone or together with the background wedding music are important for classifying the "wedding" concept.

With temporal tracking in short-term video slices, better visual atomic patterns can be found compared to the static-region-based alternatives where no temporal tracking is involved. Tracking of robust regions can reduce the influence of noisy regions. Such noise usually comes from imperfect segmentation, *e.g.*, over segmentation or wrong segments due to sudden changes of motion or illumination. By finding trackable short-term regions and using such region tracks as whole units to form the short-term visual atoms, the influence of erroneous segments from a few frames can be alleviated through averaging across good segments as majorities.

The audio descriptors are based on a *Matching Pursuit* (*MP*) representation of the audio data. MP [24] is an algorithm for sparse signal decomposition from an over-complete set of basis functions, and MP-based audio features have been used successfully for classifying ambient environmental sounds [9]. MP basis functions correspond to concentrated bursts of energy localized in time and frequency and span a range of time-frequency tradeoffs, allowing us to describe an audio signal with the basis functions that most efficiently explain its structure. The sparseness of the representation makes this approach robust to background noise, since a particular element will remain largely unchanged even as the surrounding noise level increases; this is related to a highly robust audio fingerprinting approach explored in [28]. The

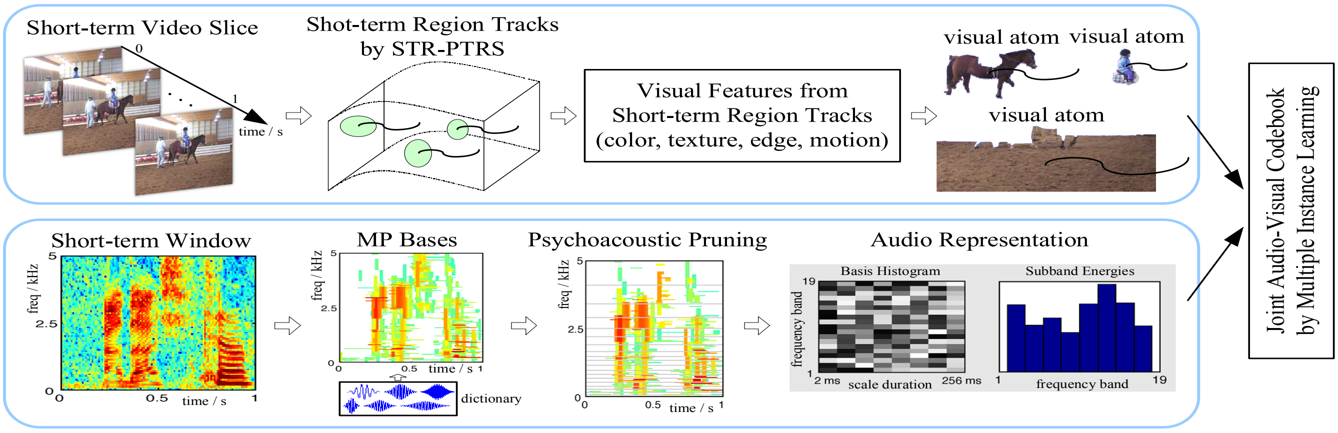


Figure 2: The overall framework of the proposed approach.

composition of an MP representation should allow discrimination among the various types of structured (*e.g.* speech and music) and unstructured audio elements that are relevant to concept detection. We extract the audio feature from each short-term window corresponding to the short-term video slice for visual tracking. Each window is decomposed into its most prominent elements, and described as a histogram of the parameters of the basis functions. The mean energy in each frequency band is also used.

Based on the S-AVA representation, we construct discriminative audio-visual codebooks using *Multiple Instance Learning* (*MIL*) [25] to capture the representative joint audio-visual patterns that are salient for detecting individual concepts. We extensively evaluate our algorithm over a challenging benchmark: Kodak's consumer video data set from real users [21]. Our method is compared with two state-of-the-art static-region-based image categorization approaches that also use multiple instance learning: the DD-SVM algorithm [7] where visual codebooks are constructed by MIL based on static regions and codebook-based features are generated for SVM classification; and the ASVM-MIL algorithm [36] where asymmetrical SVM classifiers are directly built using static regions under the MIL setting. Experiments demonstrate significant improvements achieved by our joint audio-visual codebooks, *e.g.*, over 120% MAP gain (on a relative basis) compared to both DD-SVM and ASVM-MIL. In addition, the joint audio-visual features outperform visual features alone by an average of 8.5% (in terms of AP) over 21 concepts, with many concepts achieving more than 20%.

3. SHORT-TERM VISUAL ATOM

Detecting and tracking unconstrained objects in generic videos are known to be difficult. As can be seen from the example frames in Fig. 12 (Section 6), there exist dramatic clutter, occlusions, change of shape and angle, and motions of both camera and multiple objects. Most previous tracking algorithms, both blob-based trackers [31, 34] and model-based trackers [15, 18] can not work well. Specifically, blob-based approaches rely on silhouettes derived from variants of background substraction methods, while in generic videos due to the complex motions from both objects and camera, and the occlusion of multiple objects, it is very hard to obtain satisfactory silhouettes. On the other hand, most model-based algorithms rely on manual initialization, while for automatic semantic concept detection, such manual initialization is not available. Object detectors can be used to initialize a tracking process [26] but are restricted to tracking some specific objects like human body or vehicle, since it is unrealistic to train a detector for any arbitrary object.

We propose an effective framework, called Short-Term Region tracking with joint Point Tracking and Region Segmentation (STR-PTRS), to extract short-term visual atoms from generic videos. Tracking is conducted within short-term video slices (*e.g.*, 1 second) to accommodate generic videos, since only during a short period of time, the changes and movements of the camera and objects are relatively small and there is a high chance to find consistent parts in the frames that can be tracked well. To obtain meaningful regions from a video slice, we use the image segmentation algorithm (which relies on the static color and texture appearance) instead of the background substraction or spatial-temporal segmentation methods [11] that rely on motion. This is because it is very hard to separate camera motion from object motion in generic videos and the overall motion is very unstable. In addition, for semantic concept detection not only foreground objects but also backgrounds are useful.

Within each short-term video slice, we jointly use interest point tracking and region segments to obtain short-term region tracks. Robust points that can be locked-on well are tracked through the short-term video slice, and based on point linking trajectories, image regions from adjacent frames are connected to generate region tracks. Compared to other possible alternatives, *e.g.*, connecting regions with the similarity over the color and/or texture appearance directly, our approach is more effective in both speed and accuracy: to track a foreground/background region, matching with raw pixel values is not as reliable as matching with robust interest points, due to the change of lighting, shape, and angle; and extracting higher-level color/texture visual features for region matching is quite slow.

The next subsection will describe the detailed STR-PTRS. Now let's formulate our problem. Let $\mathbf{v}$ denote a video that is partitioned into m consecutive short-term video slices $v_1, \ldots, v_m$ (*e.g.*, each v_i has 1-sec length). A set of frames $I_i^1, \ldots, I_i^T$ are uniformly sampled from each video slice v_i with a relatively high frequency, *e.g.*, one for every 0.1 second. Our task is to extract short-term visual atoms from these short-term video slices v_i, $i=1, \ldots, m$.

3.1 Short-Term Point Track

Image features (corners *etc.*) that can be easily locked-on are automatically found [30] and then tracked by using the Kanade-Lucas-Tomasi Tracker (KLT Tracker) [4] for every short-term video slice v. The result is a set of N_p feature tracks, and each feature track has a trajectory $P_j^t=(x_{1j}^t, x_{2j}^t)$, where $t=1,\dots,T$ is the temporal index (in the unit of frames), j is the index of feature tracks, and x_1, x_2 are the image coordinates.

The KLT tracker is used because of its potent to balance reliability and speed. The KLT tracker defines a measure of dissimilarity that quantifies the change of appearance of a feature between the first and the current image frame, allowing for affine image changes. At the same time, a pure translation model of motion is used to track the selected best features through the sequence. In addition, the maximum inter-frame displacement is limited to improve the reliability and the processing speed. Alternative methods such as tracking with SIFT-based registration [37, 38] generally have limitations in dealing with a large amount of videos (*e.g.*, 1358 videos with 500,000+ frames in our experiments in Section 6) due to the speed problem.

In practice, we initiate the KLT tracker with 3000 initial points. In the next subsection, the extracted point tracking trajectories are used to generate short-term region tracks.

3.2 Short-Term Region Track

Each frame I^t is segmented into a set of n_r^t homogeneous color-texture regions $r_1^t,\dots,r_{n_r^t}^t$ by the JSeg tool developed in [12]. Then from each short-term video slice v, we generate a set of N_r short-term region tracks $\mathbf{r}_1,\dots,\mathbf{r}_{N_r}$ by the algorithm described in Fig. 3. Each region track $\mathbf{r}_j$ contains a set of regions $\{r_j^t\}$, where $t=1,\dots,T$ is the temporal index (in the unit of frames). The basic idea is that if two regions from the adjacent frames share lots of point tracking trajectories, these two regions are considered as matched regions. To accommodate inaccurate segmentation (where a region from the frame at time t may be separated into several regions at time $t+1$, or several regions from time t may be merged at time $t+1$), we use a replication method to keep all the possible region tracks as illustrated in Fig. 4. Such an approach not only retains all possible region tracks to provide rich information for constructing S-AVA-based codebooks in the later section, but also helps to reduce the noise from inaccurate segmentation. By treating the short-term region track as a whole unit, the influence of wrong segments from the few frames can be reduced by averaging across good segments as majorities.

Note that the above STR-PTRS algorithm may miss some region tracks that enter into the screen in the middle of a short-term video slice. However such regions will still be found in the next video slice as long as they stay in the screen long enough. For those regions that enter and exit the screen very fast (*e.g.*, within a video slice), they are negligible in most generic videos for the purpose of semantic concept detection. Similarly, if a shot transition happens within a video slice, most region tracks during the transition may be thrown away, and the final detection performance will hardly be affected. In addition, our STR-PTRS can be extended by adding a backward checking process to overcome this problem. This is also one of our future work.

To select the appropriate length for short-term video slices, we need to consider two aspects. The video slice needs to be short so that a good amount of point tracking trajectories can be found to get region tracks. On the other hand, the longer the video slice is the better information it retains about temporal movements in visual and audio signals.

```
Input: A set of frames $I^1,\dots,I^T$ from a short-term video slice $v$. Regions $r_1^t,\dots,r_{n_r^t}^t$ for each frame $I^t$. A set of $N_p$ point tracks $P_j^t$, $j=1,\dots,N_p$, $t=1,\dots,T$.
1. Initialization: set $\mathcal{R}=\phi$, $N_r=0$.
2. Iteration: for $t=1,\dots,T$
  • Set $\mathcal{U}=\phi$.
  • Calculate $M_{k,g}^{t|t+1}=\sum_{j=1}^{N_p} I(P_j^t\in r_k^t)I(P_j^{t+1}\in r_g^{t+1})$ for each pair of regions $r_k^t\in I^t$, $r_g^{t+1}\in I^{t+1}$.
  • For each region $r_k^t\in I^t$:
      – If $M_{k,l^*}^{t|t+1}>H_{low}$ ($l^*=\arg\max_g M_{k,g}^{t|t+1}$), add matched region pair ($r_k^t$, $r_{l^*}^{t+1}$) to $\mathcal{U}$.
      – If $M_{k,l}^{t|t+1}>H_{high}$ ($l\neq l^*$), add matched region pair ($r_k^t$, $r_l^{t+1}$) to $\mathcal{U}$.
  • Iteration: for the set of $m^t$ region pairs $(r_k^t, r_{g_1}^{t+1}),\dots,(r_k^t, r_{g_{m^t}}^{t+1})$ in $\mathcal{U}$ that start with region $r_k^t$:
      – If there exist $m^r$ region tracks $\mathbf{r}_1,\dots,\mathbf{r}_{m^r}$, $\mathbf{r}_j\in\mathcal{R}$ that end with $r_k^t$, replicate each region track $\mathbf{r}_j$ by $m^t$ times, and extend each region track replication by appending $r_{g_1}^{t+1},\dots,r_{g_{m^t}}^{t+1}$ to the end of each replication respectively. Set $N_r=N_r+m^t\times m^r$.
      – Else, create new tracks $\mathbf{r}_{N_r+1},\dots,\mathbf{r}_{Nr+m^t}$ starting with $r_k^t$ and ending with each $r_{g_1}^{t+1},\dots,r_{g_{m^t}}^{t+1}$ respectively. Set $N_r=N_r+m^t$.
3. Remove region tracks in $\mathcal{R}$ with lengths shorter than $H_{long}$. Output the remaining region tracks in $\mathcal{R}$.
```

Figure 3: The algorithm to generate short-term region tracks. $I(\cdot)$ is the indicator function. In practice, we empirically set $H_{long}=T-2$, $H_{low}=10$, $H_{high}=\frac{1}{2}M_{k,l^*}^{t|t+1}$.

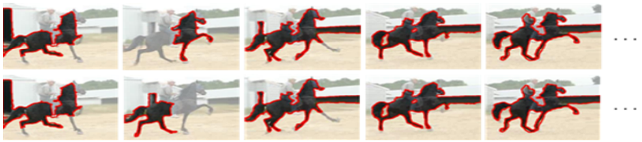


Figure 4: An example of region track replication. In the 2^{nd} frame the horse is separated to 2 parts by inaccurate segmentation. We keep both 2 possible region tracks.

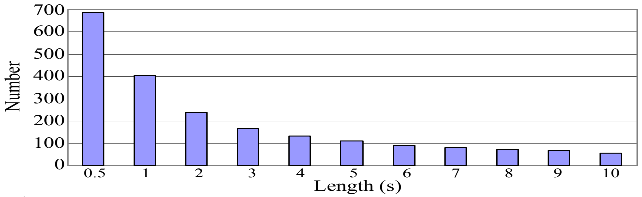


Figure 5: Number of point tracking trajectories with changing lengths of the short-term video slice. 1-sec length gives a balanced choice and is used in practice.

Fig. 5 gives average numbers of point tracking trajectories with changed lengths of video slices. From the figure 1-sec length gives a balanced choice and is used in practice.

3.3 Visual Features over Region Tracks

In this subsection we generate visual feature representations for the short-term region track $\mathbf{r}$. First, several types of visual features are extracted from each region $r^t\in\mathbf{r}$, including color moments in the HSV space (9 dimensions), Gabor texture (48 dimensions), and edge direction histogram (73 dimensions). These features have been shown effective in detecting generic concepts [5]. We concatenate these features into a 130-dim feature vector $\tilde{\mathbf{f}}_{vis}^t$ and then average $\tilde{\mathbf{f}}_{vis}^t$ across time $t=1,\dots,T$ to obtain a 130-dim feature vector $\mathbf{f}_{vis}$ for the region track $\mathbf{r}$. $\mathbf{f}_{vis}$ describes the overall visual characteristics of $\mathbf{r}$. In addition, optical flow vectors are calculated over every pixel of each frame I^t using the Lucas-Kanade method [23], where for a pixel (x_1^t, x_2^t) a motion vector $[m_1(x_1^t, x_2^t), m_2(x_1^t, x_2^t)]$ is obtained. Then for each region $r^t \in \mathbf{r}$, a 4-dim feature vector $\tilde{\mathbf{f}}_{mt}^t$ is computed, where each bin corresponds to a quadrant in the 2-D motion space and the value for this bin is the average speed of motion vectors moving along directions in this quadrant. For example, the first item in $\tilde{\mathbf{f}}_{mt}^t$ is computed as:

$$\frac{1}{R}\sum\nolimits_{(x_1^t,x_2^t)\in r^t: m_1(x_1^t,x_2^t)>0, m_2(x_1^t,x_2^t)>0}\sqrt{m_1^2(x_1^t,x_2^t)+m_2^2(x_1^t,x_2^t)}$$

where R is the total size of region r^t. Then we average $\tilde{\mathbf{f}}_{mt}^t$ across $t = 1, \ldots, T$ to obtain a motion feature vector $\mathbf{f}_{mt}$ for the region track $\mathbf{r}$. $\mathbf{f}_{mt}$ describes the overall moving speed and direction of $\mathbf{r}$. The coarse 4-bin granularity is empirically chosen since for the purpose of semantic concept detection fine granularity of motion directions can be very noisy, *e.g.*, an animal can move towards any direction. The coarse description of motion speed and direction gives relatively robust performance in general.

Note that more visual features can be extracted to describe short-term region tracks, such as local descriptors like SIFT [22] and HOG [10]. In our experiments, we have constructed the "bag-of-words" histogram for the region track $\mathbf{r}$ based on a codebook generated by clustering SIFT features from a set of training videos, following the recipe of [14]. However, for concept detection over our consumer videos, local SIFT can not compete with the global regional visual feature $\mathbf{f}_{vis}$. Due to space limit, we will omit such comparison results in the rest of this paper. This phenomenon intuitively confirms how challenging this consumer collection is. Due to the large diversity in the visual content, there is very little repetition of objects or scenes in different videos, even those from the same concept class. In such a case, it is hard to exert the advantage of local descriptors like SIFT for local point matching/registration. Nonetheless, local features can still be used as additional descriptors to complement the regional global visual features.

4. AUDIO REPRESENTATION

We represent the audio sound using a matching pursuit decomposition [24]. This is done over each short-term window corresponding to the short-term video slice for visual tracking. The bases used for MP are Gabor functions, which are Gaussian-windowed sinusoids. The Gabor function is evaluated at a range of frequencies covering the available spectrum, scaled in length (trading time resolution for frequency resolution), and translated in time. The created functions form a dictionary, which possesses a continuum of time-frequency localization properties. The length scaling creates long functions with narrowband frequency resolution, and short functions (well-localized in time) with wideband frequency resolution. This amounts to a modular STFT representation, with analysis windows of variable length. During MP analysis, functions are selected in a greedy fashion to maximize the energy removed from the signal at each iteration, resulting in a sparse representation. The Matching Pursuit Toolkit [20], an efficient implementation of the algorithm, is used. The dictionary contains functions at eight length scales, incremented by powers of two. For data sampled at 16 kHz, this corresponds to durations ranging from 2 to 256 ms. These are each translated in increments of one eighth of the function length, over the duration of the signal.

To ensure coverage of the audio activity in each short-term window, we extract a fixed number of functions (500) from each window. We then prune this set of functions with post-processing based on psychoacoustic masking principles [29]. This emulates the perceptual effect by which lower energy functions close in frequency to higher-energy signal cannot be detected by human hearing. We retain the 70% of the functions with the highest perceptual prominence relative to their local time-frequency neighborhood. This emphasizes the most salient functions, and removes less noticeable ones.

From this representation, histograms are calculated over the center frequency parameters of the functions extracted from each short-term window. A separate histogram is constructed for each of the eight function scales (durations) in the dictionary. The frequency axis is divided logarithmically into constant-Q frequency bins, one-third of an octave wide, giving 19 bins in total; each scale uses the same frequency bins. These divisions are perceptually motivated, to imitate the frequency resolution of human hearing. Since the histogram does not retain information about the relative amplitude of the functions, the mean energy of functions in each frequency bin is added to the feature set.

Compared to conventional features like MFCCs, these new features are designed to be relatively invariant to background noise and to variations in acoustic channel characteristic, due to the focus on energy peaks, and the normalization implicit in forming the histogram, respectively. The histogram also provides a natural domain for segmenting the representation into portions associated with distinct objects. As will be discussed in Section 7, such ability gives the opportunity to study moderately tight audio-visual synchronization, *i.e.*, sounding regions, as an interesting future work.

By now, a 152-dim audio feature ($\mathbf{f}_{audio}$) is extracted from each short-term window corresponding to the video slice for visual tracking. $\mathbf{f}_{audio}$ can be attached (by concatenation) to each short-term visual atom in this video slice to generate joint audio-visual atoms. Such audio-visual atoms provide candidate elementary data units to learn salient audio-visual patterns for describing individual semantic concepts.

5. JOINT AUDIO-VISUAL CODEBOOK

As illustrated in Fig. 6, each S-AVA contains a short-term region track $\mathbf{r}$ associated with a visual feature vector $\mathbf{f}_{vis}$ (d_{vis} dimensions), a motion feature vector $\mathbf{f}_{mt}$ (d_{mt} dimensions), and an audio feature vector $\mathbf{f}_{audio}$ (d_{audio} dimensions). We can concatenate different types of features into various multi-modal vectors, based on which different multi-modal codebooks can be constructed.

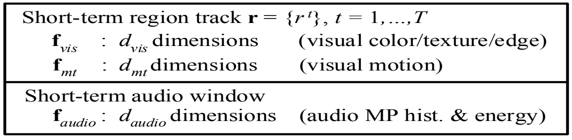
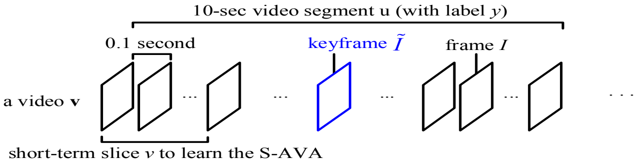

Short-term Audio-Visual Atom (S-AVA)
Short-term region track $\mathbf{r} = \{r^t\}$, $t = 1,\ldots,T$
$\mathbf{f}_{vis}$: d_{vis} dimensions (visual color/texture/edge)
$\mathbf{f}_{mt}$: d_{mt} dimensions (visual motion)
Short-term audio window
$\mathbf{f}_{audio}$: d_{audio} dimensions (audio MP hist. & energy)

Figure 6: Structure of S-AVA in our implementation. This structure can be easily extended to accommodate other types of features.

As described in Fig. 7, a video concept detection task usually has the following formulation: a set of keyframes are sampled from each video $\mathbf{v}$, *e.g.*, one keyframe $\tilde{I}_l$ for every 10 seconds. A binary label $y_l^k = 1$ or -1 is assigned to each keyframe $\tilde{I}_l$ to indicate the occurrence or absence of a concept C^k in the 10-sec video segment u_l centered at the keyframe. Based on this structure, we will use the extracted S-AVAs to construct a discriminative joint audio-visual codebook for each concept C^k.



10-sec video segment u (with label y)
0.1 second
keyframe $\tilde{I}$
frame I
a video v
short-term slice v to learn the S-AVA

Figure 7: Structure for a video concept detection task.

Each 10-sec video segment u can be treated as a "bag-of-S-AVAs", *i.e.*, it consists of a set of S-AVAs generated from the previous sections, and each S-AVA is an instance in the 10-sec bag. Thus y is the label over the bag rather than over instances. For a semantic concept C^k, it is sensible to assume that a "positive" bag u_l (with $y_l^k=1$) must have at least one of its instances being "positive", *e.g.*, a positive 10-sec video segment for concept "animal" must have at least one "animal" S-AVA. On the other hand, a "negative" bag u_l (with $y_l^k=-1$) does not have any "positive" instance. This formulation is known as Multiple Instance Learning (MIL) [7, 25, 36] in the literature.

With different concatenations of $\mathbf{f}_{vis}$, $\mathbf{f}_{mt}$, and $\mathbf{f}_{audio}$, various multi-modal features can be generated to describe an S-AVA. Assume that we have a combined d-dim feature space. For each concept C^k, we repeat an MIL-type procedure P_k-times in order to obtain P_k discriminative prototypes $(\mathbf{f}_p^{k*}, \mathbf{w}_p^{k*})$, $p=1,\ldots,P_k$, consisting of a prototype point (or centroid) $\mathbf{f}_p^{k*}=[f_{p1}^{k}{}^*,\ldots,f_{pd}^{k}{}^*]^T$ in the d-dim feature space, and the corresponding weights for each dimension $\mathbf{w}_p^{k*}=[w_{p1}^{k}{}^*,\ldots,w_{pd}^{k}{}^*]^T$.

5.1 Prototype Learning by MIL

Among the flavors of MIL objective functions, the Diverse Density (DD) is one that fits our intuitive objective above and also with efficient inference algorithm available [7] via expectation-maximization (EM). In the rest of Section 5.1, we omit subscripts k, p without loss of generality, as each $\mathbf{f}^*$ will be independently optimized for different concepts over different video segment bags $l\in\{1,\ldots,L\}$ and different instances $j\in\{1,\ldots,N_l\}$ in each bag u_l. The DD objective function for one bag u_l is simply written as:

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

where $\mathbf{f}_{lj}$ is the feature vector of the j-th S-AVA instance with short-term region track $\mathbf{r}_{lj}$, and $||\mathbf{f}||_{\mathbf{w}}$ is 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 u_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 u_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 video segment bag u_l, there should be at least one S-AVA to be treated as a positive sample to carry the label of that bag. This instance, denoted by $L(u_l)$, is identified as the closest instance to the prototype $\mathbf{f}^*$ and is given by Eq (2). For each negative bag u_l (with $y_l=-1$), on the other hand, all instances are treated as negative samples, whose contributions to Q_l are all preserved.

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

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

$$Q_l = \begin{cases} e^{-||\mathbf{f}_{lL(u_l)}-\mathbf{f}^*||^2_{\mathbf{w}^*}} & ,\ y_l=1 \\ \prod_{j=1}^{N_l}(1-e^{-||\mathbf{f}_{lj}-\mathbf{f}^*||^2_{\mathbf{w}^*}}) & ,\ y_l=-1 \end{cases} \qquad (3)$$

The DD function in Eq (3) is used to construct an objective function Q over all bags, $Q=\prod_{u_l} Q_l$. Q is maximized by an EM algorithm [7].

We use each instance in each positive bag to repeatedly initiate the DD-optimization process above, and prototypes with DD values smaller than a threshold H_{dd} (that equals to the mean of DD values of all learned prototypes) are excluded. Such prototype learning process is conducted for each semantic concept independently, and the final learned prototypes form a codebook to describe the discriminative characteristics of each individual concept.

In practice, since the number of negative bags is usually much larger than that of positive bags, we maintain a balanced number of positive and negative bags for prototype learning by sampling the negative ones. Specifically, the negative bags that come from the same videos as positive bags are all used, and at least one negative bag is randomly selected from the remaining videos.

5.2 Codebook-Based Concept Detection

For each semantic concept C^k, the learned prototypes form a codebook to describe its discriminative characteristics, each prototype corresponding to a codeword. These codewords span a codebook-based feature space to represent S-AVAs. For an S-AVA with a short-term region track $\mathbf{r}$ and a feature $\mathbf{f}$, it can be mapped to each prototype codeword $(\mathbf{f}_p^{k*},\mathbf{w}_p^{k*})$ by the weighted norm-2 distance $||\mathbf{f}-\mathbf{f}_p^{k*}||^2_{\mathbf{w}_p^{k*}}$. Accordingly, each 10-sec video segment u can be mapped to each prototype codeword by using the minimum distance $D(u,\mathbf{f}_p^{k*})_{\mathbf{w}_p^{k*}}=\min_{\mathbf{r}_j\in u}\Big\{||\mathbf{f}_j-\mathbf{f}_p^{k*}||^2_{\mathbf{w}_p^{k*}}\Big\}$. Then the video segment u can be represented by a codebook-based feature $\mathbf{D}^k(u)=\Big[D(u,\mathbf{f}_1^{k*})_{\mathbf{w}_1^{k*}},\ldots,D(u,\mathbf{f}_{P_k}^{k*})_{\mathbf{w}_{P_k}^{k*}}\Big]^T$, base on which classifiers like SVMs [33] can be trained for concept detection.

By using different combinations of $\mathbf{f}_{vis}$, $\mathbf{f}_{mt}$, and $\mathbf{f}_{audio}$, various codebooks can be generated in different multi-modal feature spaces. In general, different types of codebooks have uneven advantages at detecting different concepts. We selectively choose the optimal types of codebooks to use by adopting a boosting feature selection framework similar to [32]. The Real AdaBoost method [13] is used where during each iteration, an optimal codebook is selected to construct an SVM classifier as the weak learner, and the final detector is generated by adding up weak learners from multiple iterations. The boosting algorithm is summarized in Fig. 8.

6. EXPERIMENTS

We evaluate our algorithm over Kodak's consumer benchmark video set [21], which contains 1358 videos from real consumers. 5166 keyframes are uniformly sampled from the videos for every 10 seconds, and are labeled to 21 semantic concepts that are of great interest based on real user study. The concepts fall into several broad categories including activities (*e.g.*, ski, sports), occasions (*e.g.*, birthday, wedding), locations (*e.g.*, beach, playground), scenes (*e.g.*, sunset), or particular objects in the scene (*e.g.*, baby, boat).

We separate the entire data set into two subsets: 60% videos, *i.e.*, 813 videos, are randomly sampled as the training data; and the rest 40% videos are used for testing. This

```
Input: Training set $\mathcal{S}=\{(u_1,y_1),\ldots,(u_n,y_n)\}$. Each 10-sec video segment $u_i$ is represented by several types of codebook-based features learned with different combinations of $\mathbf{f}_{vis}$, $\mathbf{f}_{mt}$, and $\mathbf{f}_{audio}$.
1. Initialization: set sample weights $\sigma_i^1=1/2n^+$ or $1/2n^-$ for $y_i=1$ or $-1$, respectively, where $n^+$ ($n^-$) is the number of positive (negative) samples; set final decisions $H^1(u_i)=0$, $i=1,\ldots,n$.
2. Iteration: for $\tau=1,\ldots,\Gamma$
    • Get training set $\tilde{\mathcal{S}}^\tau$ by sampling $\mathcal{S}$ according to weights $\sigma_i^\tau$.
    • Train an SVM over set $\tilde{\mathcal{S}}^\tau$ by using the $k$-th type of feature.
      Get the corresponding $q_k^\tau(u_i)=p_k^\tau(y_i=1|u_i)$, $i=1,\ldots,n$.
    • Set $h_k^\tau(u_i)=\frac{1}{2}\log\left\{q_k^\tau(u_i)/(1-q_k^\tau(u_i))\right\}$.
    • Choose the optimal $h^{\tau,*}(\cdot)=h_k^\tau(\cdot)$ with the minimum error
      $\epsilon^{\tau,*}=\epsilon_k^\tau$, $\epsilon_k^\tau=\sum_{i=1}^n\sigma_i^\tau e^{-y_i(H^\tau(u_i)+h_k^\tau(u_i))}$, $\epsilon_k^\tau<\epsilon_j^\tau$ if $j\neq k$.
    • Update weights: $\sigma_i^{\tau+1}=\sigma_i^\tau e^{-y_i h^{\tau,*}(u_i)}$, $i=1,\ldots,n$, and
      re-normalize so that $\sum_{i=1}^n\sigma_i^{\tau+1}=1$.
    • Update $H^{\tau+1}(u_i)=H^\tau(u_i)+h^{\tau,*}(u_i)$ for $i=1,\ldots,n$.
```

Figure 8: The algorithm to construct concept detectors by selectively using different codebooks. 10 iterations are empirically taken in our experiments ($\Gamma=10$).

is a multi-label data set, *i.e.*, each keyframe can have multiple concept labels. One-vs.-all classifiers are trained for detecting each individual concept, and the average precision (AP) and mean average precision (MAP) are used as performance measures. AP is an official performance metric for multi-label semantic concept detection in images and videos [27]. MAP is obtained by averaging APs across all concepts.

To extensively evaluate the proposed audio-visual analysis framework, we experiment on two different concept detectors using the short-term atomic representation: (1) Short-term Visual Atoms with MIL codebook construction (S-VA-MIL), where the visual-codebook-based features are directly used to train SVM detectors; and (2) S-AVA with MIL codebook construction and Boosting feature selection (S-AVA-MIL-Boosting), where different types of codebooks are generated and selectively used via Boosting. In addition, we compare S-VA-MIL with two state-of-the-art static-region-based image categorization approaches that also use MIL, *i.e.*, DD-SVM [7] and ASVM-MIL [36]. For static-region-based methods, each 10-sec video bag u contains a set of static regions that come from the center frame of each short-term video slice v in this 10-sec bag. DD-SVM learns visual codebooks with static bags using MIL for individual concepts, and codebook-based features are generated to train SVMs. ASVM-MIL directly builds an asymmetrical SVM over the static regions under the MIL setting. No temporal tracking is involved in both of these two approaches.

6.1 Codebook Visualization

We first show examples of the discriminative prototypes learned in various types of codebooks. Such visualization helps us to subjectively and intuitively evaluate different approaches. To get the short-term region tracks to visualize, for each prototype $(\mathbf{f}^*,\mathbf{w}^*)$ we calculate the distance between all training S-AVA instances and this prototype. Then the S-AVA with the minimum distance is considered as the most appropriate example to visualize this prototype. In addition, prototypes learned for each concept can be ranked according to the DD values Q in descending order. The higher rank a prototype has, the better the prototype describes the discriminative characteristics of this concept. Fig. 12 gives some example prototypes (ranked within top 50) extracted based on short-term visual atoms. From Fig. 12, the visual-atom-based prototypes are very reasonable. For example, in the Location category we get water, sand, and beach facility as representative patterns to describe the "beach" concept; in the Activity category, we get the white snow, bald white trees, and the athlete as representative patterns to describe the "ski" concept; in the Occasion category, we get wedding gown, black suit, and wedding candles as representative patterns to describe the "wedding" concept; and in the Object category we get the baby face, baby hand, and baby toys as representative patterns to describe the "baby" concept.

In comparison, Fig. 13 gives some example prototypes learned by the static-region-based DD-SVM algorithm. In later experiments we will see that our method significantly outperforms static-region-based approaches. The prototype visualization helps to explain such results. The static DD-SVM gets very noisy prototypes in general, *e.g.*, many fragments of human clothes are extracted as representative patterns. Although some good prototypes can also be obtained, the performance suffers from the noisy ones a lot. The results also confirm our motivation that short-term region tracks are more noise-resistent for video concept detection.

By adding audio feature to short-term visual atoms, salient audio-visual patterns can be discovered by the audio-visual codebook for concepts that are expected to have strong cues in both audio and visual aspects. Fig. 14 gives some example prototypes learned by using S-AVAs with the concatenation of $\mathbf{f}_{vis}$ and $\mathbf{f}_{audio}$. These prototypes are salient for concept detection, but are not captured by visual-atom-based codebooks. For example, the salient patterns about the tableware with a piece of birthday cake inside can be discovered by considering audio and visual features jointly but can not be extracted by using visual features alone. This is because tableware also appears in many other videos visually, and only when combined with the background birthday music can the tableware generate salient audio-visual cues to describe "birthday" videos. Similarly, body parts of a dancing person can be discovered by using audio-visual atoms but are missed by using visual features alone, since only when combined with background music can the body parts form salient audio-visual cues to describe "dancing" videos.

6.2 Performance of Concept Detection

In this section, we compare AP and MAP of different algorithms for semantic concept detection. All algorithms use the RBF kernel: $K(\mathbf{x}_i,\mathbf{x}_i)=\exp\{-\theta||\mathbf{x}_i-\mathbf{x}_j||_2^2\}$, and a multiple-parameter technique [6]. That is, the error control parameter C in SVM [33] takes values $C=2^s$, $s=\{0,1,2,3,4\}$, and θ takes values $\theta=(1/d)^{2^t}$, $t=\{-3,-2,-1,0,1\}$ (d is the dimension of the data point $\mathbf{x}$). This gives 25 parameter settings with different combinations of C and θ, based on which 25 SVM classifiers can be constructed and then averagely fused to generate the final classification result. We use this multi-parameter technique instead of tuning parameters by cross-validation due to the findings in [6], *i.e.*, over this challenging consumer video set, parameter tuning tends to over fit resulting from the large diversity of the video content.

6.2.1 Short-term region tracks vs. static regions

As described in Section 5, each short-term region track is associated with a visual feature vector $\mathbf{f}_{vis}$. For a static region, we can also extract a visual feature $\mathbf{f}_{vis}$. Fig. 9 shows the per-concept AP and MAP comparison of S-VA-MIL, DD-SVM, and ASVM-MIL, by using $\mathbf{f}_{vis}$. The results from random guess are also shown for comparison. From the figure, our S-VA-MIL consistently outperforms other methods over every concept, and significant performance improvements, *i.e.*, over 120% MAP gain on a relative basis, can be achieved compared to both DD-SVM and ASVM-MIL. This phenomenon confirms that static regions segmented from generic videos are very noisy. Our short-term region tracks can significantly reduce the noise by not only extracting robust and trackable regions, but also averaging out the outlier noise through the entire tracked sequence.

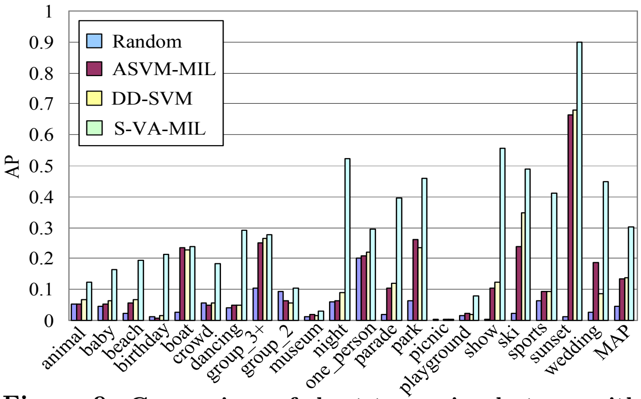


Figure 9: Comparison of short-term visual atoms with static-region-based methods.

6.2.2 *S-AVA-MIL with multi-modal features*

Fig. 10 gives the performance comparison of our S-AVA-MIL by using different codebooks generated from individual and different concatenations of $\mathbf{f}_{vis}$, $\mathbf{f}_{mt}$, and $\mathbf{f}_{audio}$. From the result, $\mathbf{f}_{mt}$ performs badly because of the low-quality motion in unconstrained videos and the lack of discriminative power of motion alone for concept detection. For example, the moving speed and direction of a person can not discriminate "one person" videos. $\mathbf{f}_{audio}$ alone can not compete with $\mathbf{f}_{vis}$, since most of the 21 concepts are visual-oriented. However, $\mathbf{f}_{audio}$ works very well over "museum". The visual content of "museum" videos is very diverse while the audio sound is relatively consistent, *e.g.,* the sound of people talking and walking in a large quiet indoor room. For multi-modal features generated by concatenating $\mathbf{f}_{vis}$ and $\mathbf{f}_{mt}$, or $\mathbf{f}_{vis}$ and $\mathbf{f}_{audio}$, the overall MAP performances are both slightly better than $\mathbf{f}_{vis}$ alone. By adding the noisy motion features ($\mathbf{f}_{vis}$+$\mathbf{f}_{mt}$), most concepts get worse or unchanged performances except for "beach" and "sports", which receive 9.6% and 3.5% AP gains, respectively. This is reasonable since "sports" videos often have fast moving athletes, and "beach" videos have large stable regions like sky, sand and waterfront that do not move. On the other hand, audio features are helpful in many cases. By adding audio features ($\mathbf{f}_{vis}$+$\mathbf{f}_{audio}$), 12 concepts get clear improvements, *e.g.,* "boat" and "sports" get 28.7% and 15.9% AP gains, respectively. However, we also have noticeable AP degradation over some concepts like "crowd" and "park", because the regional color and texture features are much more powerful in detecting these concepts than audio features. The results also indicate the uneven strengths of different modalities in detecting different concepts. Therefore, as will be shown in Section 6.2.3, a more rigorous approach in selecting optimal features from different modalities is needed.

6.2.3 *S-AVA-MIL-Boosting with multi-modal features*

Fig. 11 gives the detection performance of our S-AVA-MIL-Boosting. For better comparison, we show results from random guess and S-VA-MIL using visual feature $\mathbf{f}_{vis}$ alone.

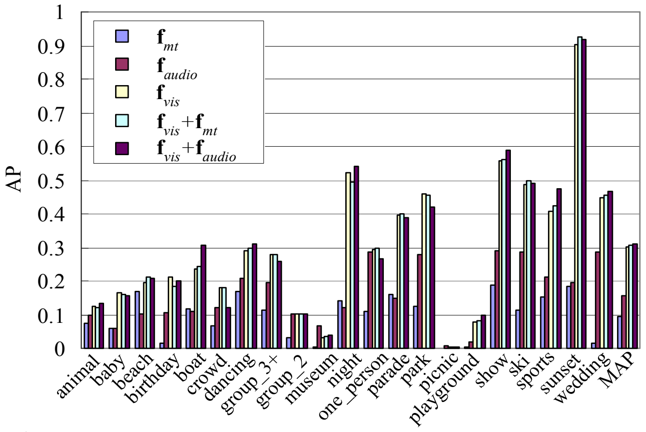


Figure 10: Comparison of S-AVA-MIL with individual and different concatenations of $\mathbf{f}_{vis}$, $\mathbf{f}_{mt}$, and $\mathbf{f}_{audio}$.

Also, we compare with a straightforward fusion approach, where SVM detectors trained using codebooks generated from $\mathbf{f}_{vis}$, $\mathbf{f}_{vis} + \mathbf{f}_{mt}$, and $\mathbf{f}_{vis} + \mathbf{f}_{audio}$ respectively are averagely combined together to give the final detection results. From the figure, we can see that by selectively using the optimal types of codebooks for detecting different individual concepts, our S-AVA-MIL-Boosting can improve the detection performance over most of the concepts (17 out of 21) compared to S-VA-MIL. Significant AP gains are achieved (on a relative basis) for "animal" by 20.9%, "beach" by 28.3%, "boat" by 35.4%, "crowd" by 11.7%, "group of three or more" by 14.6%, "dancing" by 56.7%, "museum" by 106.3%, "playground" by 20.7%, and "sports" by 18.9%. The overall MAP is improved by 8.5%. In comparison, without feature selection, the direct fusion method can not improve the overall performance by simply adding up classifiers from different modalities, which is consistent with the findings in [6].

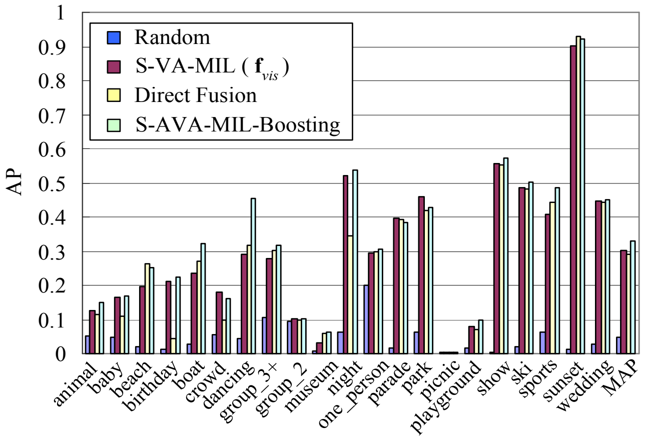


Figure 11: Comparison of S-AVA-MIL-Boosting, S-VA-MIL with visual feature $\mathbf{f}_{vis}$, and direct fusion.

7. CONCLUSION

We propose a framework for audio-visual analysis in generic videos by extracting atomic representations over short-term video slices. Visual atoms are extracted by STR-PTRS and are associated with regional visual and background audio features to generate S-AVAs. Joint audio-visual codebooks are constructed on top of S-AVAs to capture salient audio-visual patterns for effective concept detection. Our method provides a balanced choice for audio-visual analysis in generic videos: we generate a middle-level atomic representation to fuse visual and audio signals and do not rely on precise object extraction. Experiments over the challenging consumer benchmark videos demonstrate the effectiveness.

The performance of our algorithm is limited by the quality of image segmentation. Although by temporal tracking we alleviate the influence of noisy segments, bad segments (caused by sudden movement of camera or sudden lighting change) can still break our temporal tracking. To increase the robustness of the extracted S-AVAs, several approaches can be taken, such as using the multiple segmentation strategy to generate various sets of region segments, or using overlapping video slices with multiple window durations to generate a large pool of candidate short-term tracks.

One major future work is to explore moderately tight audio-visual synchronization, *i.e.*, sounding regions, from generic videos. As discussed in Section 3, with a backward checking process, our STR-PTRS can be extended to find short-term region tracks with their starting/ending time stamps within video slices. Trajectories of such short-term region tracks can be generated. On the other hand, the MP-based audio representation is specifically chosen to be able to find audio features corresponding to identified video objects. By describing audio as a set of parameterized basis functions, distinct features of overlapping sounds can be segmented into disjoint sets of functions. Since each function depends only on a compact neighborhood of time-frequency energy, it is largely invariant to simultaneous energy from other sources, unlike common audio features like MFCCs which reflect the global properties of all energy present in a sound. Moreover, the precise timing associated with each function can provide both for the detection of repeated structure within the sound, and also for detailed synchronous correlation against video features. By modeling relations between visual region trajectories and audio bases, we can study moderately tight audio-visual synchronization for discovering interesting audio-visual events like horse running and people singing. In addition, we may explore temporal patterns beyond co-occurrence and synchronization, such as "audio atom A typically precedes video atom B by a time offset between 1-3 seconds". We may investigate related data mining techniques to discover such pattern rules.

8. ACKNOWLEDGMENTS

This material is based upon work supported by Eastman Kodak and the National Science Foundation under Grant No. 0716203.

9. REFERENCES

[1] J. Anemueller and *et al.*. Biologically motivated audio-visual cue integration for object categorization. In *CogSys*, 2008.

[2] Z. Barzelay and Y. Schechner. Harmony in motion. In *Proc. CVPR*, pages 1–8, 2007.

[3] M.J. Beal and *et al.*. A graphical model for audiovisual object tracking. In *IEEE Trans. PAMI*, 25(7):828-836, 2003.

[4] S. Birchfield. KLT: An Implementation of the Kanade-Lucas-Tomasi Feature Tracker. http://vision.stanford.edu/∼birch

[5] S.F. Chang and *et al.*. Columbia university TRECVID-2005 video search and high-level feature extraction. In *NIST TRECVID workshop*, Gaithersburg, MD, 2005.

[6] S.F. Chang and *et al.*. Large-scale multimodal semantic concept detection for consumer video. In *ACM MIR*, 2007.

[7] Y.X. Chen and J.Z. Wang. Image categorization by learning and reasoning with regions. In *JMLR*, 5:913–939, 2004.

[8] M. Cristani and *et al.*. Audio-visual event recognition in surveillance video sequences. In *IEEE Trans. Multimedia*, 9(2):257-267, 2007.

[9] S. Chu and *et al.*. Environmental sound recognition using MP-based features. in *Proc. ICASSP*, pages 1–4, 2008.

[10] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Proc. CVPR*, pages 886-893, 2005.

[11] D. Dementhon and D. Doermann. Video retrieval using spatial-temporal descriptors. In *ACM Multimedia*, 2003.

[12] Y. Deng and B.S. Manjunath. Unsupervised segmentation of color-texture regions in images and video. In *IEEE Trans. PAMI*, 23(8):800-810, 2001.

[13] J. Friedman and *et al.*. Additive logistic regression: a statistical view of boosting. *Ann. of Sta.*, 28(22):337-407, 2000.

[14] K. Grauman and T. Darrel. The pyramid match kernel: Discriminative classification with sets of image features. In *Proc. ICCV*, 2:1458-1465, 2005.

[15] B. Han and *et al.*. Incremental density approximation and kernel-based bayesian filtering for object tracking. In *Proc. CVPR*, pages 638–644, 2004.

[16] J. Hershey and J. Movellan. Audio-vision: Using audio-visual synchrony to locate sounds. In *NIPS*, 1999.

[17] K. Iwano and *et al.*. Audio-visual speech recognition using lip information extracted from side-face images. In *EURASIP JASMP*, 2007(1):4-4, 2007.

[18] A. Jepson, D.J. Fleet, and T.F. El-Maraghi. Robust online appearence models for visual tracking. In *IEEE Trans. PAMI*, 25(10):1296–1311, 2003.

[19] R. Kaucic, B. Dalton, and A. Blake. Real-time lip tracking for audio-visual speech recognition applications. In *Proc. ECCV*, vol.2, pages 376-387, 1996.

[20] R. Gribonval and S. Krstulovic. MPTK, the matching pursuit toolkit. http://mptk.irisa.fr/

[21] A. Loui and *et al.*. Kodak's consumer video benchmark data set: concept definition and annotation. In *ACM SIGMM Int'l Workshop on MIR*, pages 245–254, 2007.

[22] D. Lowe. Distinctive image features from scale-invariant keypoints. In *IJCV*, 60(2):91-110, 2004.

[23] B.D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *Proc. Imaging understanding workshop*, pages 121-130, 1981.

[24] S. Mallat and Z. Zhang. Matching pursuits with time-frequency dictionaries. In *IEEE Trans. Signal Processing*, 41(12):3397–3415, 1993.

[25] O. Maron and *et al.*. A framework for multiple-instance learning. In *NIPS*, 1998.

[26] J.C. Niebles and *et al.*. Extracting moving people from internet videos. in *Proc. ECCV*, pages 527–540, 2008.

[27] NIST. TREC Video Retrieval Evaluation (TRECVID). 2001 – 2008. http://www-nlpir.nist.gov/projects/trecvid/

[28] J. Ogle and D. Ellis. Fingerprinting to identify repeated sound events in long-duration personal audio recordings. In *Proc. ICASSP*, pages I-233-236, 2007.

[29] F. Petitcolas. MPEG for MATLAB. http://www.petitcolas.net/fabien/software/mpeg

[30] J. Shi and C. Tomasi. Good features to track. In *Proc. CVPR*, pages 593–600, 1994.

[31] C. Stauffer and W.E.L. Grimson. Learning patterns of activity using real-time tracking. In *IEEE Trans. PAMI*, 22(8):747–757, 2002.

[32] K. Tieu and P. Viola. Boosting image retrieval. In *IJCV*, 56(1-2):228–235, 2000.

[33] V. Vapnik. Statistical learning theory. Wiley-Interscience, New York, 1998.

[34] X.G. Wang and *et al.*. Learning Semantic Scene Models by Trajectory Analysis. In *Proc. ECCV*, pages 110-123, 2006.

[35] Y. Wu and *et al.*. Multimodal information fusion for video concept detection. in *Proc. ICIP*, pages 2391–2394, 2004.

[36] C. Yang and *et al.*. Region-based image annotation using asymmetrical support vector machine-based multiple-instance learning. In *Proc. CVPR*, pages 2057–2063, 2006.

[37] G.Q. Zhao and *et al.*. Large head movement tracking using SIFT-based registration. In *ACM Multimedia*, 2007.

[38] H. Zhou and *et al.*. Object tracking using sift features and mean shift. In *Com. Vis. & Ima. Und.*, 113(3):345-352, 2009.

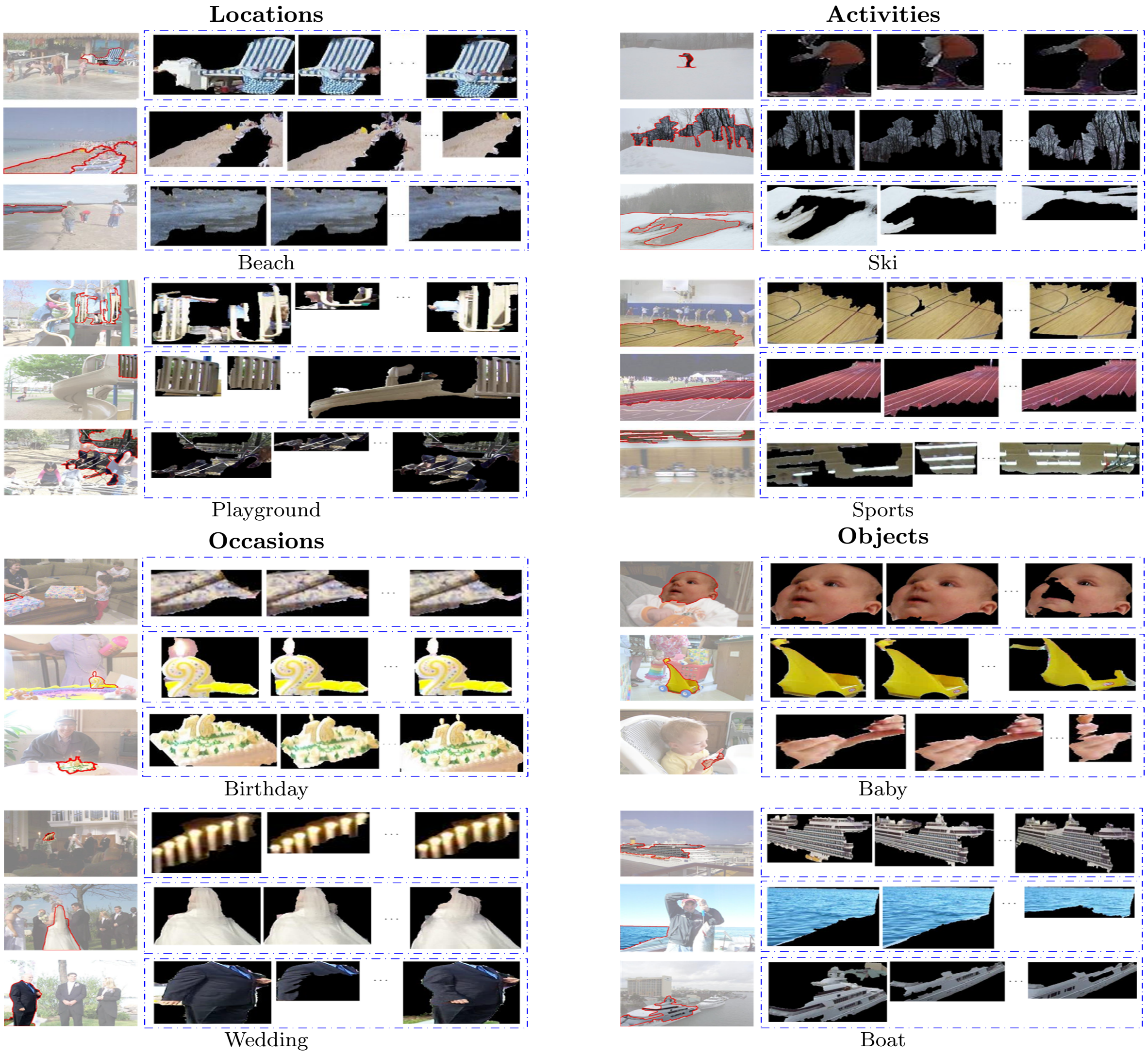


Figure 12: Example prototypes learned with short-term visual atoms for concepts of several broad categories, *i.e.*, Locations, Activities, Occasions, and Objects. Doted blue boxes show the corresponding short-term region track prototypes, where images on the left show example frames where region tracks are extracted.

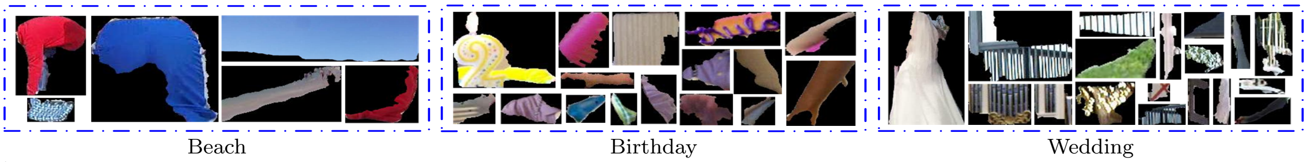


Figure 13: Examples of learned static region prototypes by DD-SVM. The static codebook contains lots of noise, *e.g.*, fragments of human clothes, which cause severe degradation of the concept detection performance.

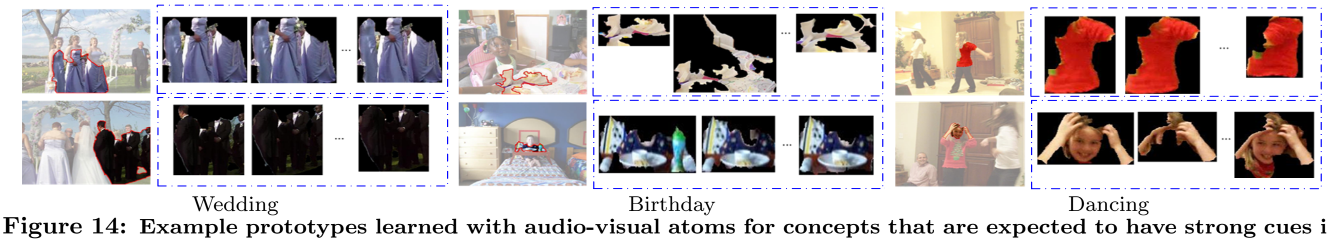


Figure 14: Example prototypes learned with audio-visual atoms for concepts that are expected to have strong cues in both audio and visual aspects. These prototypes capture salient audio-visual patterns for describing the corresponding concepts, and they are not discovered by visual-only codebooks. For example, the salient patterns about the tableware with birthday cake inside can be found by considering audio and visual features jointly but are not learned by using visual features alone. This is because tableware also appears in many other videos visually, and only when combined with the background birthday music can the tableware generate a salient audio-visual pattern for "birthday" concept.