SUPER: Towards Real-time Event Recognition in Internet Videos

Yu-Gang Jiang
School of Computer Science
Fudan University, Shanghai, China
ygj@fudan.edu.cn

ABSTRACT

Event recognition in unconstrained Internet videos has great potential in many applications. State-of-the-art systems usually include modules that need extensive computation, such as the extraction of spatial-temporal interest points, which poses a big challenge for large-scale video processing. This paper presents SUPER, a Speeded UP Event Recognition framework for efficient Internet video analysis. We take a multimodal baseline that has produced strong performance on popular benchmarks, and systematically evaluate each component in terms of both computational cost and contribution to recognition accuracy. We show that, by choosing suitable features, classifiers, and fusion strategies, recognition speed can be greatly improved with minor performance degradation. In addition, we also evaluate how many visual and audio frames are needed for event recognition in Internet videos, a question left unanswered in the literature. Results on a rigorously designed dataset indicate that similar recognition accuracy can be attained using only 14 frames per video on average. We also observe that, different from the visual channel, the soundtracks contains little redundant information for video event recognition. Integrating all the findings, our suggested SUPER framework is 220-fold faster than the baseline approach with merely 3.8% drop in recognition accuracy. It classifies an 80-second video sequence using models of 20 classes in just 4.56 seconds.

Categories and Subject Descriptors

I.2.10 [**Artificial Intelligence**]: Vision and Scene Understanding—*Video analysis*; H.3.1 [**Information Storage and Retrieval**]: Content Analysis and Indexing—*Indexing methods*

General Terms

Algorithms, Experimentation, Performance.

Keywords

Internet videos, event recognition, efficiency, multimodal features, frame selection.

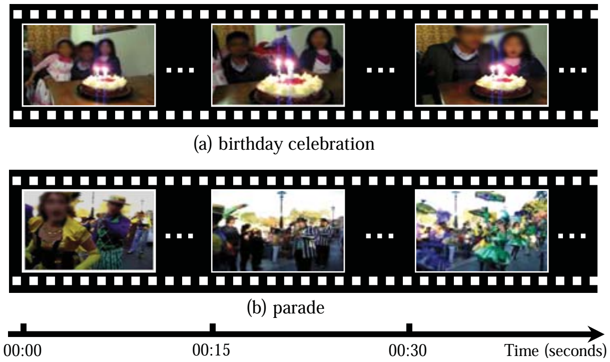


Figure 1: Examples of two events in Internet videos.

1. INTRODUCTION

The amount of videos on the Internet has grown explosively in recent years. This motivates an urgent need of techniques for automatically recognizing high-level complex video events (see examples in Figure 1), which are important in applications such as video search, smart advertising, and intelligence monitoring. State-of-the-art video event recognition systems often deploy a diverse set of features and classifiers in order to achieve a good accuracy. For instance, one of the popularly used features is spatial-temporal interest points (STIP) [7], which typically requires expensive analysis of every frame in a video sequence. While promising results have been reported on several benchmark datasets, such kind of systems are computationally too slow to deal with large-scale data in real-world applications.

This paper discusses and evaluates various options to improve event recognition efficiency, while still maintaining a high degree of accuracy. Our work is built upon a multimodal baseline that is the core of top-performing systems in NIST TRECVID benchmark evaluations [1]. We assess the computational cost and accuracy contribution of each component in this baseline approach, and identify alternative methods and implementations for speed improvement. These evaluations lead to a Speeded UP Event Recognition (SUPER) framework, which cleverly utilizes several multimodal features and classifiers that can be computed efficiently.

Apart from evaluating features and classifiers, another important question we intend to answer in this paper is "how many visual and audio frames are needed for recognizing complex events in Internet videos?". Existing approaches usually compute visual and audio features from entire video sequences [10, 5]. However, human recognition performance indicates that many events may be identified based on only a small fraction of the sequences (or even by viewing a single frame, e.g., the *birthday celebration* event). We therefore conduct an in-depth empirical study on the number of required frames for video event recognition. To the best of our knowledge, this problem has never been thoroughly investigated in the context of Internet video analysis. Our evaluation is performed under a multimodal setting, where we also discuss the similarities and distinctions of visual and audio modalities in terms of the minimum information needed for event recognition. Such a study can provide very useful insights for choosing a suitable number of frames in the design of an efficient event recognition system.

One may argue that the contents of Internet videos are too diverse. As a result it could be very difficult to select representative subsets of frames for event recognition, since an event may happen anywhere. While this is true to a certain extent, it is worth noting that videos recorded and uploaded by ordinary consumers are mostly very short and contain only a single story with fairly consistent scene settings. As a result there may be a significant amount of redundant information if all frames are computed. In this work we focus on consumer videos, the dominant role in Internet video sharing activities. Other types of videos on the Web, such as sports, news, sitcoms and movies, normally have plenty of textual tags and descriptions, and therefore do not really need deep content recognition to facilitate effective search and organization. In addition, applications like intelligence monitoring are basically only interested in the unconstrained consumer videos.

The remaining sections are organized as follows. We discuss related works in Section 2 and introduce a simple and effective baseline system for video event recognition in Section 3. Section 4 elaborates various options for speeded up event recognition and Section 5 discusses experimental results. Finally, Section 6 summarizes this paper.

2. RELATED WORKS

Compared with human action classification, recognizing complex events in unconstrained Internet videos is a new topic receiving significant research attentions. Traditional methods on action analysis purely relied on visual features [8, 10], while recent advances have shown that—for Internet videos—event recognition can benefit from the use of audio features [9, 5]. A typical event recognition process first extracts a number of multimodal features from video files, and then applies machine learning algorithms (e.g., support vector machines – SVM) for classification. In this section we mainly introduce related works that have specifically considered recognition speed.

Due to the high computational cost of extracting well-performing features like SIFT, several works have devised various alternatives to speed up this process. In [2], SURF (speeded up robust feature) was proposed as a fast image keypoint detector and descriptor. Knopp et al. [6] extended SURF to detect and describe 3D spatio-temporal keypoints. Further, researchers found that the time-consuming keypoint detection process can be omitted by using dense sampling, i.e., uniform grid-based selection of local image patches. Dense sampling has been shown to produce comparable or even better performance in visual recognition tasks [17]. In addition, a recent evaluation by Uijlings et al. [26] showed that the dense SIFT and dense SURF descriptors can be extracted more efficiently with a smart engineering design that eliminates repetitive computations of pixel responses of overlapped image regions. Compared with these visual features, acoustic descriptors like Mel-frequency cepstral coefficients (MFCC) can be computed much more efficiently and therefore it is less critical to optimize the speed of audio feature extraction.

The number of feature descriptors (e.g., SIFT or SURF) varies across different videos, posing difficulties for classifiers which normally require fixed-dimensional inputs. One popular solution to this problem is the bag-of-word representation, where descriptors in a set are quantized into a word frequency histogram using a pre-computed vocabulary (e.g., through clustering a subset of the descriptors). The quantization process is expensive using brute-force nearest neighbor search. In [16], Nister et al. showed that quantization can be executed very efficiently if words in the vocabulary are organized in a tree structure. In [14], Moosmann et al. further adopted random forest, a collection of binary decision trees, to achieve fast quantization and more accurate recognition. In [27], Yu et al. extended semantic texton forests [23] from 2D to 3D spatiotemporal analysis for fast action recognition.

The most widely used classifier for video content recognition is SVM. One expensive component in SVM classification is the computation of nonlinear kernels such as Histogram Intersection and χ^2. In [12], Maji et al. showed that histogram intersection kernels can be computed in logarithmic complexity to the number of support vectors. This method has been tested on image and video classification tasks with promising performance [26]. Alternatively, the simple and highly efficient linear kernel has been shown to work well with high-dimensional representations like Fisher vectors [20].

Schindler and van Gool conducted an interesting study to evaluate the number of required visual frames for human action recognition [21]. Their experiments indicate that a single frame is already enough for recognizing many actions. However, this study was performed on Weizmann and KTH datasets, which consist of very short action videos recorded under fully controlled environment with clean background. Therefore their observations are unlikely to be generalizable under the realistic setting of Internet consumer videos. Moreover, in addition to evaluating the number of required visual frames, we also comparatively study the minimum length of audio soundtracks (i.e., the number of audio frames) needed for event recognition, which has never been investigated in prior works.

3. A BASELINE SYSTEM FOR VIDEO EVENT RECOGNITION

To optimize speed we first need a reliable baseline system. For this, we consider a multimodal approach that lies in the heart of top-performing systems at TRECVID evaluations 2010 [5] and 2011 [15]. The best reported results in [5, 15] are slightly better than this simple baseline by using many more features (e.g., dense STIP), classifiers, and/or sophisticated multimodal fusion strategies, which require significant additional computation.

Figure 2 depicts the baseline event recognition pipeline.

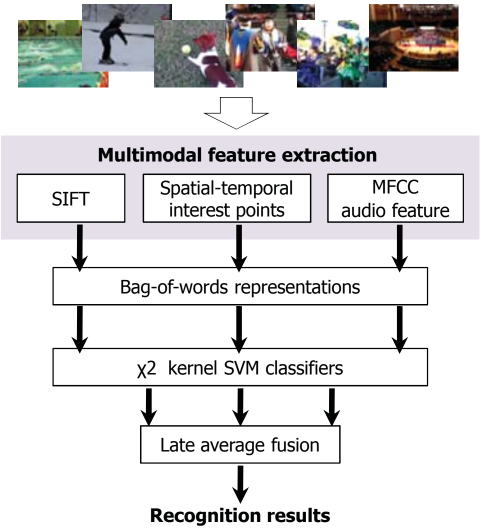

Multimodal feature extraction
SIFT
Spatial-temporal interest points
MFCC audio feature
Bag-of-words representations
χ2 kernel SVM classifiers
Late average fusion
Recognition results

Figure 2: Pipeline of a baseline video event recognition system. Three sets of audio-visual features are extracted and mapped to fixed-dimensional vectors using the popular bag-of-words framework. SVM classifiers are then applied to the three modalities separately, and prediction scores are merged via (late) average fusion.

Three visual and audio features are extracted and represented under the bag-of-words framework. Recognition is then performed by classifying the three features independently using SVM. Finally, prediction scores from the three SVM are consolidated by average fusion. We briefly describe each component of the system below.

SIFT: SIFT feature has been popular for years, exhibiting top-notch performance in many visual recognition tasks. Here two sparse keypoint detectors, Difference of Gaussian [11] and Hessian Affine [13], are adopted to find local invariant image patches from video frames. Each keypoint is described by a 128-dimensional gradient histogram [11]. Since keypoint detection on every frame is computationally expensive and nearby frames are visually very similar, we sample one frame every two seconds. Nonetheless, the uniformly sampled frames from entire video sequences may still contain redundant information, as will be examined later in Section 5.

Spatial-Temporal Interest Points (STIP): STIP captures a space-time volume in which pixel values have large variations in both space and time. Laptev's algorithm is adopted [7] to locate STIPs, which are described by Histograms of Oriented Gradients (HOG) and Histograms of Optical Flow (HOF). HOG and HOF are 72-dimensional features computed in the neighborhood of each detected 3D local volume, and are concatenated as the final descriptor (144 dimensions).

MFCC: Research in neuroscience has revealed that multiple senses can work together to largely enhance human perception [24]. Therefore in the baseline system acoustic features are also considered to complement the visual features for machine recognition of video events. Specifically, the well-known MFCCs are computed for every 32ms time-window (an audio frame) with 50% overlap.

Bag-of-Words Representation: To map the three feature sets with different cardinalities to fixed-dimensional vectors, the bag-of-words representation is adopted. Given a video clip, the features extracted from each frame (SIFT) or an entire sequence (STIP and MFCC) are collapsed into a single bag. For SIFT, two visual vocabularies of 500 words are clustered for DoG and Hessian keypoints separately, and two spatial layouts 1×1 and 2×2 are used to generate bag-of-words histograms of 5,000 dimensions $(2\times500\times(1+2\times2))$. SIFT word histograms of the uniformly sampled frames are averaged to generate a single feature vector for each video sequence. For STIP and MFCC, vocabularies of 5,000 words and 4,000 words are used respectively. No spatial/temporal partitioning is used for either. For all the three features, a soft-weighting scheme is employed to alleviate the quantization effects in generating bag-of-words features [3]. During quantization, the similarities between features and words (cluster centers) are computed by inner product of L-2 normalized vectors (equivalent to using Euclidean distance). As reported by Uijlings et al. [26], this simple trick gives a 43% speed-up over direct computation of Euclidean distances.

Classification and Result Fusion: χ^2 kernel SVM is trained separately using the three bag-of-words features. Because some event categories are not mutually exclusive, one-versus-all strategy is used to train three SVM models for each event. Given a test video, its bag-of-words features are used as input of the SVM models, and prediction scores are combined by average fusion (mean value of the scores). This fusion strategy is commonly referred to as late fusion, since multimodal information is fused after classification.

4. SUPER: SPEEDED UP EVENT RECOGNITION

In this section we discuss several possible ways to speed up the baseline system. While efficiency can be easily improved by adopting fast feature extraction/quantization methods and efficient classification kernels, we do not consider these as the main contribution of this paper as they have been (separately) studied in prior works. Instead, we directly take some of the off-the-shelf findings in feature computation and classifier design, and conduct two new and important investigations: 1) a comparative analysis of a variety of features (in terms of both recognition accuracy and efficiency); and 2) an evaluation of the number of needed visual/audio frames in Internet video event recognition. Details are introduced as follows.

4.1 Frame Sampling

Frame sampling can be regarded as a data preprocessing step. Using too few frames from each video sequence may result in poor performance because of incomplete information, while running on every frame will be computationally expensive. The main purpose of our study is to seek empirical insights in selecting the most suitable number of frames, so that a good tradeoff between speed and accuracy can be achieved. This is partially motivated by a recent analysis of human perception performance in [4], which showed that human annotators could provide very precise labels (around 80% accuracy) with an average working time much shorter

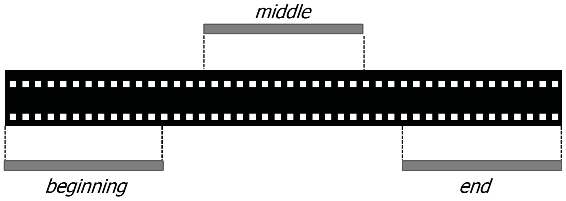


Figure 3: Video segment selection for frame sampling. Given a desired length, we chop three segments in the beginning, centered in the middle, and at the end of each video, respectively.

than mean video length.

As introduced earlier, in the baseline system audio frames are densely sampled over every 32ms window (16ms overlap) and visual frames are extracted uniformly with 2-second intervals. These audio and visual frames form the basis for down-sampling, and we are interested in the relationship between the number of selected audio/visual frames and recognition accuracy. There are mainly two strategies to down-sample the frames: *uniform sampling* – selecting frames uniformly across an entire sequence, and *segment-based sampling* – selecting frames continuously from a short video segment. Both will be evaluated in the experiments. For the latter, we test segments chopped in the beginning, around the middle, and at the end of the videos, in order to study which part contains the most informative/discriminative information. Figure 3 visualizes our simple segment selection method.

4.2 Feature Representation

Due to speed limitation, some well-performing features such as STIP become less appropriate. In this work we mainly focus on evaluating the accuracy and speed of various visual features, which are more expensive to compute compared with audio features like MFCC. In addition to the baseline features, we consider several other popular visual features as listed below.

DIFT and DURF: To get around expensive computation of sparse keypoint detection, we adopt Uijlings' fast implementation of dense SIFT and dense SURF, dubbed DIFT and DURF [26]. Given a video frame, each kind of features are converted to a bag-of-words histogram based on a visual vocabulary of 500 words. Three spatial layouts are used (1×1, 3×1 and 2×2). 3×1 (not in the baseline) is added due to its popularity in several well-performing image/video analysis systems.

Self-Similarities (SSIM): SSIM is computed by quantizing a correlation map of an image patch within a larger circular window [22]. Patch size and the window radius are set as 5×5 and 40 pixels respectively. Similar to DIFT and DURF, SSIM descriptors are computed over densely sampled image patches, and converted to bag-of-word histograms using a visual vocabulary of 500 words and the same spatial layouts.

Color Moment (CM): The first three moments of three channels in *Lab* color space are calculated over a 5×5 grid. The 9 moments from each of the 25 image partitions are concatenated into a 225-d vector.

GIST: It computes the output energy of Gabor-like filters (8 orientations, 4 scales) over a 4×4 image grid [19]. The final descriptor is 512-d ($8 \times 4 \times 4 \times 4$).

Local Binary Patterns (LBP): LBP is a popular texture feature which uses binary numbers to label each pixel of an image by comparing its value to that of neighborhood pixels [18]. We follow the standard settings to use 8 neighbors, which lead to a vector of 256 dimensions per video frame (2^8).

Tiny Images (TINY): TINY [25] is probably the most simple feature one can compute, which directly concatenates image pixel values in RGB space. Images are resized to a very small scale (32×32) in order to reduce feature dimension and the effect of misalignment. This descriptor is 3,072-d ($32 \times 32 \times 3$).

For bag-of-words quantization of the feature sets (i.e., DIFT, DURF, and SSIM), we continue to use inner product of L-2 normalized vectors. Applying random forest can further reduce the quantization time, as observed in several existing works discussed in Section 2.

4.3 Classification and Multimodal Fusion

For both classification and multimodal fusion, we take findings from the state-of-the-arts for speed improvement. Specifically, in SVM classification, we compare Maji's efficient histogram intersection kernel [12] with the baseline χ^2 kernel.

Generally speaking, the fusion of multimodal features can be done at three different stages, namely early fusion, kernel fusion, and late fusion. **Early Fusion** concatenates multiple feature vectors of a video sample into a very long vector, which is then used for classification. **Kernel Fusion** adds kernels computed by different features into one kernel for SVM learning. The difference from early fusion is that features from multiple modalities are used separately to compute kernels. Early fusion and kernel fusion are equivalent when using some simple kernels (see discussions in the next paragraph). For both early and kernel fusion, only one classifier needs to be trained. Instead of combining multiple modalities before classification, **Late Fusion** feeds kernels of different features into separate SVM classifiers and then fuses SVM prediction scores. Throughout this paper, we adopt average fusion (equal fusion weights) to combine kernels or classification scores. Average fusion has been popularly used in multimodal classification due to its simplicity. Although there are many ways for adaptively selecting fusion weights (e.g., by cross validation or multiple kernel learning), the learned weights have been frequently reported to be over-fitted to training data, i.e., they do not generalize well to new test data.

Computationally, early/kernel fusion is more efficient in both model training and testing than late fusion since the latter uses more SVM models. Speed of early and kernel fusion does not differ too much. When complex kernels like χ^2 are in use, kernel fusion is slightly slower as we need to compute the exponential function in kernels multiple times (χ^2 kernel is computed as $\mathcal{K}(\mathbf{x}, \mathbf{y}) = e^{-\rho d_{\chi^2}(\mathbf{x}, \mathbf{y})}$, where $d_{\chi^2}()$ returns the χ^2 distance of the two input vectors). For typical histogram intersection kernel computed as $\mathcal{K}(\mathbf{x}, \mathbf{y}) = \sum_i \min\{x_i, y_i\}$, early and kernel fusion (with equal weights) are apparently identical. We evaluate all the three strategies in our experiments.

Figure 4: Examples of 20 categories in Columbia Consumer Video dataset. 15 categories are events, and 5 (circled) are objects and scenes.

5. EXPERIMENTS

5.1 Dataset and Evaluation

Columbia Consumer Video (CCV) dataset [4][1] is adopted in our experiments. CCV is to our knowledge the largest publicly available dataset on Internet consumer videos. It has 9,317 YouTube videos, which are divided into a training set and a test set, containing 4,659 and 4,658 videos respectively. Average video duration is around 80 seconds (about 210 hours in total). 20 categories were carefully defined based on user studies and annotated using Amazon Mechanical Turk platform. Figure 4 gives an example for each category. Detailed definitions can be found in [4]. Among the 20 categories, 15 are events. Although our focus is on event recognition, all the 20 categories are evaluated in order to facilitate benchmark comparison. On average, there are 394 positive samples per category, which are evenly distributed in the training and test sets.

In testing, SVM prediction scores are used to rank the test set according to the probability that each video contains a category. Recognition accuracy is measured by average precision (AP). We report mean AP (mAP) over all the categories due to space limitation.

Computational efficiency is measured by the time used separately in feature extraction and classification (with various kernels and fusion schemes). Speed of feature extraction is measured by the average time needed for processing a video sequence (30fps) of 80 seconds, the average duration of CCV. Specifically, since one frame is sampled every 2 seconds, for the frame-based features (e.g., SIFT and DURF), we report time for computing 40 frames (320×240 pixels, a standard frame size of YouTube videos). Classification speed of a video is measured by the average time for classifying all the 20 categories.

For software implementation, we use popular public codes for each module, e.g., sparse keypoint detectors from INRIA-LEAR[2], STIP from Laptev [7], DIFT/DURF from Uijlings [26], fast histogram intersection SVM from Maji [12], etc. With optimization, the efficiency of some codes may be improved, which is however beyond the scope of this work. All the speed evaluations are conducted on a regular PC with an Intel Core2 Duo 2.4GHz CPU and 2GB RAM.

Table 1: Recognition accuracy (mAP) and computational efficiency of various features and their combinations. Efficiency is measured in seconds needed for extracting features of an 80-second video sequence (cf. Section 5.1). The combination of features, indicated by "+", is done by late average fusion.

Feature(s)	mAP	Time
SIFT	0.523	82.00
STIP	0.449	916.80
MFCC	0.331	2.36
SIFT+STIP	0.551	998.80
Baseline (SIFT+STIP+MFCC)	0.595	1001.16
DIFT	0.493	8.68
DURF	0.513	6.60
SSIM	0.463	37.16
CM	0.324	4.88
GIST	0.325	5.40
LBP	0.285	0.68
TINY	0.229	0.32
DIFT+DURF	0.514	15.28
DIFT+SSIM	0.525	45.84
DURF+SSIM	0.538	43.76
DIFT+DURF+SSIM	0.539	52.44
CM+GIST	0.407	10.28
CM+GIST+LBP	0.438	10.96
CM+GIST+LBP+TINY	0.434	11.28
DURF+SSIM+CM+GIST+LBP	0.545	54.72
Baseline+DURF+SSIM+CM+GIST+LBP	0.626	1055.88
MFCC+DURF+SSIM+CM+GIST+LBP	0.593	57.08
MFCC+DURF+CM+GIST+LBP	0.584	19.92
MFCC+DURF	0.567	8.96

In the following we experimentally evaluate the factors discussed in Section 4 for speeded up event recognition. We move frame selection to the end of the evaluation since it may be sensitive to other components, particularly the choice of features.

5.2 Selecting Features

First, let us evaluate the recognition accuracy and computational efficiency of multimodal features. Results are summarized in Table 1. Late average fusion of the three baseline features (SIFT, STIP, and MFCC) offers a decent mAP of 0.595. Although the three features are complementary, i.e., significant mAP gain is attained via fusion, SIFT (based on sparse detectors) and STIP are computationally too slow. In addition, the most expensive feature STIP does not show superior performance compared with the frame-based SIFT feature. This is probably due to the fact that most events, even the complex ones like *wedding ceremony*, can be identified based on static objects and scene settings.

As shown in the second group of results in Table 1, the features introduced in Section 4.2 are efficient. Among them the slowest, SSIM, only costs 45% of the time needed for extracting the sparse SIFT. We also test various combinations to see whether these new features are complementary. As

[1] http://www.ee.columbia.edu/dvmm/CCV/

[2] http://lear.inrialpes.fr/software

can be seen, DIFT is not complementary to DURF. Among the four global features (CM, GIST, LBP, TINY), TINY is the worst and combining it with the others results in minor mAP degradation (0.438→0.434). Therefore, DIFT and TINY are discarded, and the fusion of the remaining five features gives an mAP of 0.545.

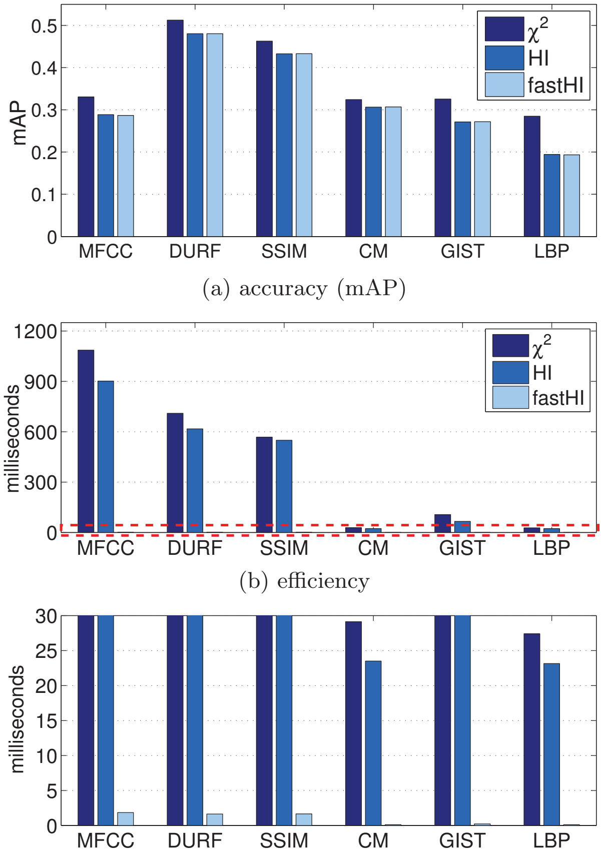
Combining the baseline with the five newly selected features we obtain an mAP of 0.626. Since SIFT and STIP are expensive to compute, we only retain MFCC from the baseline. This leads to three feature sets for fast event recognition, as shown in the bottom three rows of Table 1. Fusing MFCC with all the five produces similar performance to the baseline, but is 17 times faster. By removing the most expensive SSIM feature among these remaining ones, mAP drops marginally to 0.584 with a significant speed-up of 50 times over the baseline system. Further, by only adopting MFCC and DURF, we can still attain a fairly good mAP of 0.567 (vs. baseline mAP 0.595), with a speed-up of 111 times. Notice that for both MFCC and DURF, a considerable amount of computations is spent on bag-of-words quantization (inner product of L-2 normalized vectors is used). As discussed earlier, with tree-based vocabularies like random forest, this quantization time can be largely reduced. Since we will also try to down-sample the number of computed frames, and as a result the quantization time can be further compressed, this option is not implemented in the current version of SUPER.

5.3 Selecting Classifiers and Fusion Strategies

We now evaluate classifiers and fusion strategies. For classifier, we compare SVMs with Maji's fast approximation of histogram intersection kernel (fastHI), standard histogram intersection kernel (HI), and the baseline χ^2 kernel. MFCC and the five newly selected features are used in this experiment. Results are given in Figure 5.

As shown in Figure 5(a), χ^2 kernel outperforms HI and fastHI for all the evaluated features. mAP gap ranges from 0.03 (DURF) to as high as 0.09 (LBP). Recognition accuracy of fastHI is almost the same to that of standard HI, which is consistent with the results reported in [12]. For speed (Figure 5(b-c)), fastHI is around 200-400 times faster than HI, and HI is only slightly faster than χ^2. These are different from observations in [26], where fastHI was reported to be only 18 times faster than χ^2 and 3 times faster than standard HI. One main reason is that precomputed kernels were adopted in [26], which can be reused for multiple classes and therefore the effect of fastHI is limited. Since precomputed kernels are not applicable to Internet-scale data, this strategy is not preferred in practice.

We further compare the three fusion strategies. Figure 6 visualizes the results on two selected feature sets. There are two important observations from this experiment. First, kernel/early fusion outperforms the baseline late fusion for both HI and fast HI kernels. While for χ^2 kernel, kernel fusion is the best, and early fusion performs similar to late fusion. These results reveal that the popular late fusion strategy is probably not a good option for multimodal event recognition. We attribute this to the fact that combining features before model learning may result in a kernel space where event classes are more separable. Second and very interestingly, for all the three fusion strategies, the accuracy gap between the χ^2 and HI/fastHI kernels is generally smaller than that on the individual features (see individual feature results in Figure 5(a)), particularly when more features are used. As shown in Figure 6(a), using kernel fusion, mAPs of fusing the six features are almost the same across all kernel choices (χ^2 0.617 vs. HI/fastHI 0.616). For the fusion of MFCC and DURF (Figure 6(b)), the mAP numbers are 0.594 (χ^2) and 0.559 (HI/fastHI). Such insensitivity to kernel choice under multimodal fusion settings is a very attractive observation since we can adopt the very efficient fastHI kernel without hurting recognition accuracy significantly.



(a) accuracy (mAP)

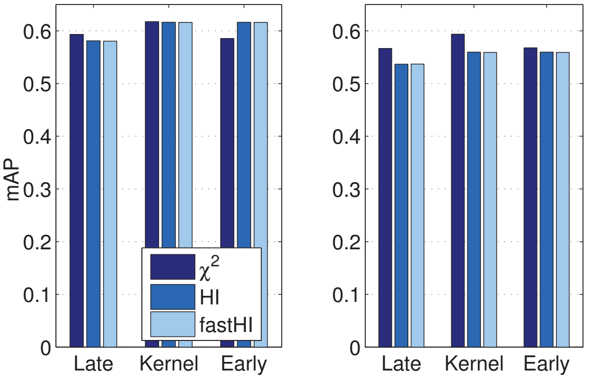
(b) efficiency

(c) zoom-in view of the area in the dotted box in (b).

Figure 5: Recognition accuracy and efficiency of χ^2, HI and fastHI kernels. Efficiency is measured by the average time for classifying a video using classifiers of all the 20 categories.

Computational efficiency of the three fusion strategies is similar. As expected, we only observe a marginal speed-up from early/kernel fusion over late fusion. Take the fusion of the six features as an example, early/kernel fusion requires 2.03 seconds to classify a video using HI kernel, and late fusion needs 2.18 seconds. While for fastHI, the former uses 4.8 milliseconds and latter costs 5.6 milliseconds.

To summarize, fastHI is a preferred kernel for SUPER since it is much faster than the others. Although it shows significantly lower recognition accuracy than χ^2 kernel when classifying several features independently (e.g., LBP), after multimodal fusion we see no significantly accuracy degradation from fastHI. In addition, kernel fusion is recommended



(a) (b)

Figure 6: Comparison of early, kernel and late fusion. (a) Fusion of six features: MFCC, DURF, SSIM, CM, GIST, and LBP; (b) Fusion of MFCC and DURF.

since it shows consistently good performance on both evaluated feature sets using all the three kernels.

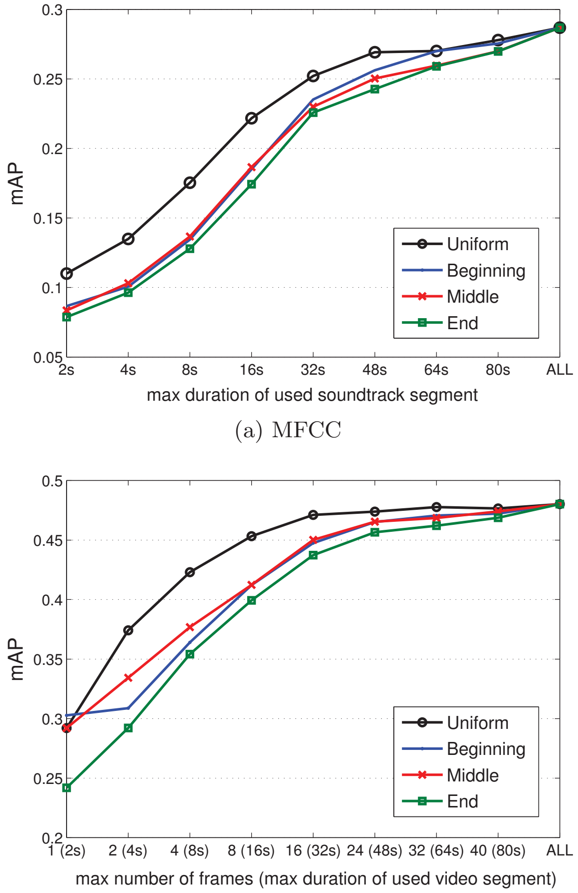
5.4 Selecting Frames

Next we evaluate the suitable number of frames required for event recognition in Internet videos. We use MFCC and DURF that have been found effective and efficient by previous experiments. Using both audio and visual features we can also study the distinctions of the two modalities in terms of the minimum number of needed frames.

Figure 7 shows recognition accuracy versus the *maximum* number of used frames. We use "max number/duration" because some videos may be shorter than the evaluated durations, for which we simply compute features from all the frames. We see that the mAP of MFCC drops rapidly when reducing the number of audio frames, indicating that the entire soundtracks are useful for event recognition. In contrast, the recognition accuracy of the visual feature DURF remains stable until a maximum of just 16 frames (i.e., a total duration of 32 seconds) are used. This shows that the visual frames contain much more redundant information than the audio counterpart and therefore can be down-sampled safely, which is very appealing as the visual features are generally more expensive to be extracted. Note that for the option "max 16 frames", on average there are only 14 visual frames computed per video, which are about 1/3 of all the frames, i.e., those used by the option "ALL" in Figure 7(b). In other words, we can save 2/3 of the visual feature extraction time without loosing recognition accuracy. This observation is different from the results on human action recognition in [21], which showed that a single frame is enough. Such distinction is as expected since both our video data and the visual classes are more complicated than that used in [21].

Among the four sampling strategies, uniform sampling clearly outperforms the three options of continuous sampling. This is not surprising because nearby frames are more likely to be similar, and thus uniform sampling leads to frame sets with less redundant information. In addition, we also find that "End" is not as good as "Beginning" and "Middle", indicating that the last segments of the videos contain (slightly) less informative contents.

Overall, results in this experiment show that event recognition speed can be improved with no performance loss by selecting a subset of visual frames and 1/3 may be a good proportion for down-sampling. By combining DURF (max 16 frames) and MFCC (entire sequence) with early fusion, we obtain an mAP of 0.557 (fastHI kernel). This is very similar to the mAP of fusing entire sequence DURF and MFCC (0.559).



(a) MFCC

(b) DURF

Figure 7: Recognition accuracy by sampling various numbers of audio/visual frames. For visual feature DURF, a good accuracy can be maintained with a maximum of 16 frames per video (uniform sampling), while for audio feature MFCC, we suggest using all the frames.

6. SUMMARY AND DISCUSSION

We have discussed and evaluated various options to improve the speed of event recognition in Internet videos, including visual-audio features, classifier kernels, multimodal fusion strategies, and frame sampling.

For feature representations, our results suggest the use of audio feature MFCC, efficient visual feature descriptors such as DURF (fast implementation of dense SURF), and several efficient global descriptors (e.g., LBP). The two computationally expensive visual features in the baseline system, sparse SIFT and STIP, should be discarded and replaced with the suggested ones.

Table 2: The most favorite components of SUPER, in comparison with the baseline system.

	Baseline	SUPER
Vis. frame sampling	–	Max 16
Aud. frame sampling	–	–
Features	SIFT, STIP, MFCC	DURF, MFCC
Classifier	χ^2 SVM	fastHI SVM
Fusion	Late	Early
mAP	0.595	0.557
Time*	1003s	4.56s

*Classifying a video of 80 s duration, using models of 20 classes.

Several kernels and fusion strategies have been evaluated. One interesting and important observation is that Maji's fastHI kernel performs close to the more expensive χ^2 kernel under multimodal fusion settings (cf. Figure 6), although χ^2 significantly outperforms fastHI on many features individually (cf. Figure 5(a)). The testing/classification time of fastHI is two orders of magnitude faster than that of the χ^2 kernel. For multimodal fusion, we observed that early/kernel fusion offers higher recognition accuracy than the popularly adopted late fusion.

Another important study conducted in this work was to examine the suitable number of visual/audio frames needed for event recognition in Internet videos. Results on CCV benchmark indicate that we can maintain a high degree of accuracy by using an average number of just 14 frames per video, saving 2/3 of the feature extraction time. While for the audio channel, it is always harmful to down-sample the frames.

We summarize our recommendations for SUPER in Table 2. These carefully selected components lead to a 220-fold speed-up with marginal accuracy degradation. By adding a few efficient global features like LBP, recognition accuracy can be further boosted. On the other hand, there is still room to improve efficiency, e.g., by employing random forest for bag-of-words quantization. One important message delivered in this paper is that complex events in Internet videos can be recognized fairly efficiently while maintaining a competitive accuracy, which is critical to the applicability of the automatic techniques in large scale problems.

7. REFERENCES

[1] TRECVID multimedia event detection track. `http://www.nist.gov/itl/iad/mig/med.cfm/`.

[2] H. Bay, A. Ess, T. Tuytelaars, and L. van Gool. SURF: Speeded up robust features. *Computer Vision and Image Understanding*, 110(3):346–359, 2008.

[3] Y. G. Jiang, C. W. Ngo, and J. Yang. Towards optimal bag-of-features for object categorization and semantic video retrieval. In *Proc. of ACM International Conference on Image and Video Retrieval*, 2007.

[4] Y.-G. Jiang, G. Ye, S.-F. Chang, D. Ellis, and A. C. Loui. Consumer video understanding: A benchmark database and an evaluation of human and machine performance. In *Proc. of ACM International Conference on Multimedia Retrieval*, 2011.

[5] Y.-G. Jiang, X. Zeng, G. Ye, S. Bhattacharya, D. Ellis, M. Shah, and S.-F. Chang. Columbia-UCF TRECVID2010 multimedia event detection: Combining multiple modalities, contextual concepts, and temporal matching. In *Proc. of NIST TRECVID Workshop*, 2010.

[6] J. Knopp, M. Prasad, G. Willems, R. Timofte, and L. van Gool. Hough transform and 3D SURF for robust three dimensional classification. In *Proc. of the European Conference on Computer Vision*, 2010.

[7] I. Laptev. On space-time interest points. *International Journal of Computer Vision*, 64:107–123, 2005.

[8] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld. Learning realistic human actions from movies. In *IEEE Conf. on Computer Vision and Pattern Recognition*, 2008.

[9] K. Lee and D. P. W. Ellis. Audio-based semantic concept classification for consumer video. *IEEE Transactions on Audio, Speech, and Language Processing*, 18(6):1406–1416, 2010.

[10] J. Liu, J. Luo, and M. Shah. Recognizing realistic actions from videos "in the wild". In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

[11] D. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal on Computer Vision*, 60:91–110, 2004.

[12] S. Maji, A. C. Berg, and J. Malik. Classification using intersection kernel support vector machines is efficient. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, 2008.

[13] K. Mikoljczyk and C. Schmid. Scale and affine invariant interest point detectors. *International Journal of Computer Vision*, 60:63–86, 2004.

[14] F. Moosmann, E. Nowak, and F. Jurie. Randomized clustering forests for image classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(9):1632–1646, 2008.

[15] P. Natarajan and et al. BBN VISER TRECVID 2011 multimedia event detection system. In *Proc. of NIST TRECVID Workshop*, 2011.

[16] D. Nister and H. Stewenius. Scalable recognition with a vocabulary tree. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, 2006.

[17] E. Nowak, F. Jurie, and B. Triggs. Sampling strategies for bag-of-features image classification. In *Proc. of European Conference on Computer Vision*, 2006.

[18] T. Ojala, M. Pietikainen, and T. Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7):971–987, 2002.

[19] A. Oliva and A. Torralba. Modeling the shape of the scene: a holistic representation of the spatial envelope. *International Journal of Computer Vision*, 42:145–175, 2001.

[20] F. Perronnin, J. Sanchez, and T. Mensink. Improving the fisher kernel for large-scale image classification. In *Proc. of the European Conference on Computer Vision*, 2010.

[21] K. Schindler and L. van Gool. Action snippets: How many frames does human action recognition require? In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, 2008.

[22] E. Shechtman and M. Irani. Matching local self-similarities across images and videos. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, 2007.

[23] J. Shotton, M. Johnson, and R. Cipolla. Semantic texton forests for image categorization and segmentation. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, 2008.

[24] B. E. Stein and T. R. Stanford. Multisensory integration: current issues from the perspective of the single neuron. *Nature Reviews Neuroscience*, 9:255–266, 2008.

[25] A. Torralba, R. Fergus, and W. T. Freeman. 80 million tiny images: a large database for non-parametric object and scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(11):1958–1970, 2008.

[26] J. R. R. Uijlings, A. W. M. Smeulders, and R. J. H. Scha. Real-time visual concept classification. *IEEE Transactions on Multimedia*, 12(7):665–680, 2010.

[27] T.-H. Yu, T.-K. Kim, and R. Cipolla. Real-time action recognition by sptiotemoral semantic and structural forests. In *Proc. of British Machine Vision Conference*, 2010.