Participation at TRECVID 2011
Semantic Indexing & Content-based Copy Detection Tasks

Wan-Lei Zhao
Department of Computer Science, University of Kaiserslautern
zhao@iupr.com

Damian Borth
Department of Computer Science, University of Kaiserslautern
d_borth@cs.uni-kl.de

Thomas M. Breuel
Department of Computer Science, University of Kaiserslautern
tmb@cs.uni-kl.de

Abstract

Semantic Indexing Task (SIN)

Run No.	Run ID	Run Description	infMAP (%)
1	F_A_IUPR-DFKI_1	Fisher Kernel + SVMs	2.86
2	F_A_IUPR-DFKI_2	Color Correlogram + SVMs	5.38
3	F_A_IUPR-DFKI_3	Fisher Kernel fused with Color Correlograms + SVMs	5.0
4	F_A_IUPR-DFKI_4	Fisher Kernel + kNN	0.71

Content-based Copy Detection (CCD)

Run No.	Run ID	Run Description	Opt.NDCR
1	*iupr-dfki.fsift	F-SIFT+BoW+HE+EWGC	0.776
2	*iupr-dfki.fsift2	F-SIFT+BoW+HE+EWGC	0.923
3	SIFT	SIFT+BoW+HE+EWGC	0.884
4	SIFT+PV	SIFT+BoW+HE+EWGC+PV	0.501
5	F-SIFT+PV	F-SIFT+BoW+HE+EWGC+PV	0.446

*: officially submitted run.

This paper describes the TRECVID 2011 participation of the IUPR-DFKI team in the semantic indexing task (SIN) and content based copy detection task (CCD) task. For **SIN**, this years participation was dominated by an significant increase of vocabulary concept size from 130 to 346 concepts. In particular the system setup has been changed to last year's participation [6] with respect to computational demands employing less computational costly features for classification and no usage of external training sources like YouTube. For **CCD**, this years participation is aimed at testing the flip invariant SIFT applied in video-only CCD. At the same time, we investigated how well we could achieve by relying on one keypoint feature alone.

1. Introduction

While the web-based video continues to grow rapidly with respect to user communities and numbers of views [1], the demand for robust search and retrieval tools increases: For example, YouTube as the market leader still struggles to grant efficient access to vast parts of their content. Correspondingly, the majority (99%) of video views on YouTube are generated by a few (30%) highly popular videos [2]. This lack of information describing the content of videos is one of the key challenges of today's web based video platforms.

Another challenge of such platforms is the protection of content owership i.e. prevention of copyright infringement. Although tool as YouTube's Content ID[1] exist, such systems are fare from being perfect and require further development [3].

In this year's TRECVID benchmark we participated in the Semantic Indexing task (SIN) and the Content-based Copy Detection task (CCD). This paper will first describe the system setup for SIN and then continue with an outline of our system for CCD.

2. Semantic Indexing

This section describes the semantic indexing task, set up of our system and the resulting runs. The goal of semantic indexing task is to predict the presence of semantic concepts like locations, objects, or actions appearing in a dataset of unknown videos [18]. One successful approach to provide such indexing is concept detection [19]. This year the size of the concept vocabulary was increased to 346 concepts[2].

2.1. Datasets

In this year's participation the Internet Archive Creative Common (IACC.1) dataset from 2010 benchmark was used. The dataset is splitted into training data and test data. For both datasets the shot boundary reference information is provided for temporal segmentation into shots - the main unit of interest. The datasets are defined as follows:

1. **Training**: This dataset consists of last year's test (IACC.1.A) and training (IACC.1.tv10.training) datasets containing over $400h$ of video material. The $11.5k$ videos vary in video length ranging from $10s$ to 3.5 minutes and segment into *266*k shots. For this dataset concept labels have been acquired by manual inspection through TRECVID's collaborative annotation effort [5].

2. **Testing:**: A dataset (IACC.1.B) videos containing $200h$ of video ranging in duration from $10s$ to 3.5 minutes define the 8k unknown test videos with 137k shots, for which concept appearance should be predicted and submitted for evaluation to NIST.

2.2. Approach

In this year's TRECVID participation we performed a major change in our feature extraction setup. With the goal to speed up the entire process we changed from a dense sampling of SIFT patches to an interest point detection based sampling and a post-processing to Fisher BoW features. As compared to our last year's participation and because of its robust performance and its computationally less expensive extraction, color correlograms have been evaluated as additional visual features. For classification the system depends on SVMs. However, we also evaluated kNN classification in this year's participation. The system is describes in more detail in following sections.

2.2.1 Keyframe Extraction

Regarding shot representation we extract keyframes for each shot which serve as samples for further processing. First, the standard shot boundary reference - as given from NIST - was used for temporal segmentation of the videos. Then an intra-shot diversity based approach was applied for keyframe extraction [7]. For each shot, a K-Means clustering is performed over MPEG7 Color Layout Descriptors [16] extracted from all frames. The number of clusters is fitted using the Bayesian Information Criterion [17]. For each cluster the frame closest to the cluster center is chosen as a keyframe providing $1...k$ keyframes per shot. Sample keyframes for the concepts "cityscape" and "mountain" can be seen in Figure 1.

2.2.2 Features

For each keyframe the following visual features are extracted:

[1] `http://www.youtube.com/t/contentid`

[2] `http://www-nlpir.nist.gov/projects/tv2011/tv11.sin.346.concepts.simple.txt`

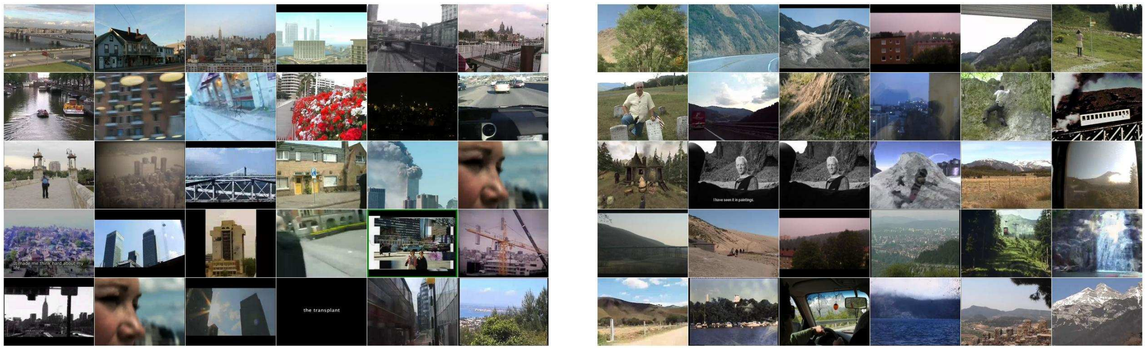


Figure 1. Sample keyframes for the concept “cityscape” (left) and “mountain” (right) in the IACC.1 dataset.

- **Fisher on BoW**: Visual words are extracted by first performing Harlap point detection with SIFT features [15], this results in 260 features per keyframe on average. Keypoint features in one frame are aggregated by Fisher kernel with a small visual vocabulary (16 words). The equation of Fisher aggregation [13] is shown in Eqn. 1.

$$V = \sum_{s=1}^{N} [x_s^d - w_i^d], \qquad (1)$$

  where N is the number of keypoints in a frame, w_i is the closest visual word to keypoint x_s^d. This results in 2048-dimensional feature. This feature is further reduced to 128 dimensions by PCA mapping. As a consequence, one frame is finally represented by an 128-dimensional aggregated BoW feature.

- **Color Correlograms**: To capture color information, color correlograms [11] have been extracted. The descriptor forms a 600-dim. vector and is normalized to 1 as in [10].

2.2.3 Statistical Model

For classification the following statistical models have been applied:

- **Support vector machines**: SVMs were used as a standard approach for concept detection, forming the core of numerous concept detection systems [19]. We used the LIBSVM [8] implementation with a χ^2 kernel, which has empirically been demonstrated to be a good choice for histogram features [20]:

$$K(x,y) = e^{-\frac{d_{\chi^2}(x,y)^2}{\gamma^2}} \qquad (2)$$

  where $d_{\chi^2}(.,.)$ is the χ^2 distance. γ and the SVM cost of misclassifications C were estimated separately for each concept using a grid search over the 3-fold cross-validated average precision. One problem is that training sets are *imbalanced*, i.e. the number of negative samples outnumbers the number of positive ones. Such setups cause problems for many classifiers, including SVMs [4]. To overcome this problem, the dominant class is subsampled to obtain roughly balanced training sets. For each run, SVMs were trained on the given small-scale training sets with maximal 3000 positive and and 6000 negative training examples from the available data set.

  In all cases, SVM scores were mapped to probability estimates using the LIBSVM standard implementation.

- **kNN Classification**: Because of the redundancy in the dataset we also employ kNN classification. In this light-weight method, we simply predict a sample based on its top-*80* nearest neighbors. A confidence score is assigned to the sample by normalizing sum of all the *Cosine* similarities this sample to the training instances from one class.

2.2.4 Late Fusion

Finally, scores obtained from several keyframes are fused:

- Having several keyframes for each shot, the corresponding scores are simply averaged, providing a single score for each shot and feature.

- For fusing different features, we perform a weighted sum fusion, whereas concept-specific

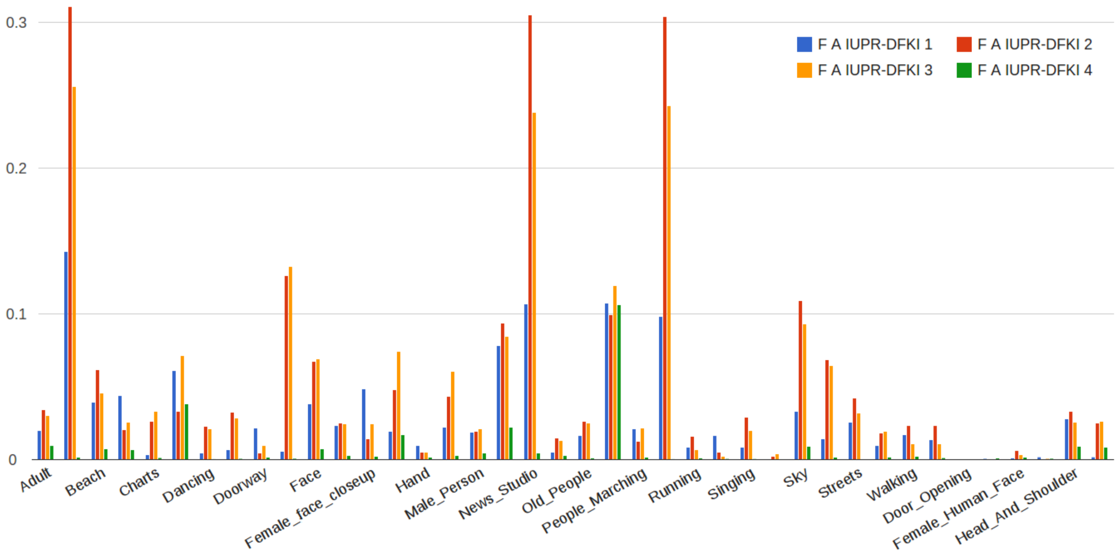


Figure 2. Quantitative results for our runs. The first three runs evaluate different features with SVM based classification whereas the third run uses kNN classification.

weights are learned using a grid search maximizing average precision on the TRECVID data set.

2.3. Results

We submitted 4 runs for the full submission including all 346 concept detections. In particular, our runs are described as follows:

1. **F_A_IUPR-DFKI_1** In this run, we used the SVM approach in combination with Fisher BoW features.
2. **F_A_IUPR-DFKI_2** As in F_A_DFKI-MADM_1, the second run used SVMs but now only color correlograms as features for keyframe description are employed.
3. **F_D_IUPR-DFKI_3** Here, we combine Fisher on BoW and color correlograms features used as in F_A_DFKI-MADM_1 and F_A_DFKI-MADM_2 setup in a late fusion approach.
4. **F_B_IUPR-DFKI_4** In contrast to the previous runs, we perform concept detection based on kNN classification using Fisher on BoW features.

Quantitative results are displayed in Figure 2. It can be seen that concept detection using color correlograms (infMAP of **5.38%**, for F_A_IUPR-DFKI_2) outperforms pure Fisher BoW based concept detection (infMAP of **2.86%** for F_A_DFKI-MADM_1) Regarding concept detection using multiple features, the combined run (infMAP of **5.0%** for F_A_DFKI-MADM_3) is not reaching a performance as similar as one of the single feature runs. Finally, the kNN based concept detection (infMAP of **0.7%** for F_A_DFKI-MADM_4) is not able to outperform SVM based concept detection. Probably the amount of samples being in the training set is not sufficient for an accurate nearest neighbor classification.

3. Content-based Copy Detection

This section describes the content-based copy detection task [14], our system set up and the resulting runs.

3.1. Datasets

Our system is tested on TRECVID sound and vision dataset 2010. The dataset consists of *11,525* web videos with a total duration of *400* hours. There are *1,608* video-only queries which are artificially generated by eight different transformations ranging from camcording, picture-in-picture, re-encoding, frame dropping to the mixture of different transformations including flip, blurring and etc.

In our pre-processing step, dense keyframe sampling is performed on both query and reference videos with the same rate: one keyframe per *1.6* seconds. This results in an average of *46* keyframes per query, and a total of *903,656* keyframes in the reference dataset. Image local features are extracted from the frames by Harris-Laplacian keypoint detector with flip invariant SIFT (F-SIFT) [21] descriptor[3]. On average, there are *309* keypoints per frame.

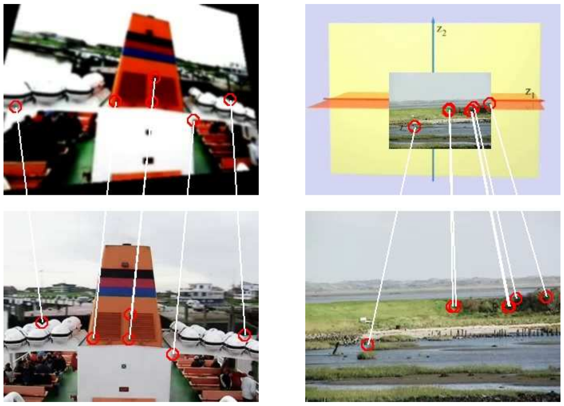
3.2. CCD Framework

In general, our detection framework is based on BoW search which is composed of an online retrieval and an offline indexing part. For the offline indexing part, keypoints extracted from each sampled frame are represented by a *20*K visual vocabulary. Each keypoint is attached with a 32-bit Hamming embedding [12] code, x, y location, characteristic scale and the dominant orientation [15].

For the online retrieval, the sampled query frames are similarly represented with BoW, and searched against the inverted file (IF). To alleviate the information loss due to vector quantization, we impose both visual (Hamming Embedding (HE) [12]) and geometric verification (EWGC [23]) on the visual word matches. Together with EWGC, SR-PE is employed to perform reciprocal validation, which is elaborated in next section.

Additionally, we perform dominant curl verification on visual word matches which is derived from F-SIFT [21] . Visual word pair is accepted as a valid match only when their signs of dominant curl meet up. Besides filtering out false matches, this scheme reduces the total number of matches fed into geometric verification (EWGC and SR-PE), which is the processing bottle-neck, this operation also speeds up the overall detection process.

Finally we employ 2D Hough transform (HT) to aggregate the scores from matched frame pairs and localize the copy segments. The aggregated score is further normalized by the identified duration on the query side. This normalized value is treated as confidence score in the final output. For each query, top k ranked reference videos (K=*1*, *2* in our case) are considered. Since we only conduct video-only copy detection, for video+audio detection task, we simply submit the same detection results for the corresponding queries.



(a) Heavy skew (b) Large scaling

Figure 3. Examples of copies with very few true positive matches due to heavy transformation.

3.3 Reciprocal Geometry Verification

In BoW based framework, given the valid visual word matches returned by IF and E-WGC verification, the similarity between a query Q and a reference image R is given by

$$Sim(Q,R) = \frac{\sum h(q,p)}{\|BoW(Q)\|_2 \cdot \|BoW(R)\|_2}, \qquad (3)$$

where $h(q,p)$ is the weighted distance [12] between Hamming signatures of q and p. The notation $BoW(Q)$ denotes the bag-of-words of Q. In order to evaluate the similarity between query and reference video, similarities of matched query and reference keyframes are aggregated on $Sim(Q,R)$ via Hough transform [9, 23].

In practice, Eqn. 3 is not robust to heavy transformation which often causes few matches between two keyframes. Figure 4 shows an example where there are only six matches being identified due to large skew and scale resulting in low similarity scores by Eqn. 3. To alleviate this problem, we revise $h(p,q)$ in Eqn. 3 such that the similarity is not only dependent on the weighted Hamming distance but also the confidence of matching between two words. The $h(p,q)$ is revised as

$$H(q,p) = (\alpha - \Delta) \times \log_\alpha \Delta \times h(q,p), \qquad (4)$$

where Δ indicates the confidence of matching, and α is an empirical parameter which is set to 0.9 in our experiment. Eqn. 4 basically amplifies $h(q,p)$ when the matched pair holds high confidence score (low Δ in other words).

[3] Code available at: http://www.cs.cityu.edu.hk/ wzhao2/˜lip-vireo.htm

Table 1. Optimized Video-only performance of five “BALANCED” runs

Transform	Measures	1	2	3	4	5	6	8	10
*iupr-dfki.fsift	NDCR	1.099	0.984	0.456	0.945	0.505	0.664	0.755	0.797
	TP	1	2	97	7	77	43	45	26
	F1	0.898	0.856	0.596	0.626	0.565	0.551	0.524	0.501
*iupr-dfki.fsift2	NDCR	1.099	0.984	0.859	0.945	0.932	0.828	0.885	0.852
	TP	1	2	59	7	36	22	42	19
	F1	0.898	0.856	0.535	0.626	0.516	0.522	0.512	0.468
SIFT	NDCR	0.984	0.992	0.654	0.906	0.776	0.852	0.958	0.953
	TP	2	1	58	12	56	19	19	6
	F1	0.518	0.898	0.516	0.568	0.528	0.525	0.495	0.424
SIFT+PV	NDCR	0.552	0.992	0.318	0.628	0.417	0.398	0.622	0.508
	TP	71	1	101	75	102	77	62	63
	F1	0.632	0.898	0.550	0.613	0.544	0.601	0.558	0.536
F-SIFT+PV	NDCR	0.552	0.886	0.172	0.622	0.240	0.258	0.302	0.539
	TP	71	83	106	62	111	95	103	59
	F1	0.622	0.673	0.597	0.620	0.593	0.595	0.609	0.599

*: officially submitted run.

We estimate Δ by reciprocal geometric verification. On one hand, given two matched words p and q from keyframes Q and R respectively, the scale $\hat{s}$ and rotation $\hat{\theta}$ between them can be approximated by SR-PE [22] approach. On the other hand, EWGC [23] is able to recover these two parameters ($\widetilde{\theta}$ and $\widetilde{s}$) via dominant orientation and characteristic scale. Notice that $\hat{s}$ and $\hat{\theta}$ could be different from the values $\widetilde{\theta}$ and $\widetilde{s}$ estimated in EWGC [22] model. However, in general the closer their values are, the higher chance that the match between p and q is correct. We thus define Δ as the discrepancy value between them as $\Delta = max\{|\hat{\theta} - \widetilde{\theta}|, |\hat{s} - \widetilde{s}|\}$. Referring back to equations Eqn. 3 and Eqn. 4, the similarity between two keyframes is revised by weighting the significance of matched words based on their Hamming distance and matching confidence.

3.4. Results

We officially submitted two ‘BALANCED’ runs for this year’s CCD task. Besides that, we carried out three internal runs. The configurations of these runs are described below.

- **iupr-dfki.fsift** F-SIFT is adopted as the descriptor in this run. The detection follows the framework presented in Section 3.2. Score is aggregated on Eqn. 3 by Hough transform. This score is further normalized by the detected duration of one query. For each query, the detected video ranked at top *1* is returned (if there is any).
- **iupr-dfki.fsift2** This run is similar to the first run except that we choose top *2* ranked candidates for each query.
- **SIFT** This run adopts SIFT [15] as the descriptor and follows the same processing flow as “iupr-dfki.fsift” except that verifying visual word matches via dominant curl is not applied. This run acts as the comparison baseline for our submissions.
- **SIFT+PV** This run is generated based on “SIFT” run. The difference lies in two aspects. First, we didn’t normalize the aggregated score by Hough transform. Second, for each detected copy video pair, the matched keyframe pair with highest score on Eqn. 3 among all matched keyframe pairs is selected and undergone OOS+SR-PE [22] verification. This verification helps to confirm whether the most confident matched pair are really near-duplicate. The video pairs fail in the verification are removed from the resulting list. This post validation (PV) only takes less than *0.25s* when it is performed between a query and one reference video.
- **F-SIFT+PV** This run is built upon “iupr-dfki.fsift”. However, unlike the first run, we didn’t normalize the aggregated score by the identified duration of one query. In addition, similar to “SIFT+PV”, OOS+SR-PE is adopted for post verification.

Since all these runs are video-only detection, their performances are under the inspection of video transformations in the paper. Table 1 and Table 2 illustrate the performances of these four runs under thresholds which optimize NDCRs and the thresholds suggested by us respectively. In the evaluation, we already ignore *48* queries which

Table 2. Actual Video-only performance of five “BALANCED” runs

Transform	Measures	1	2	3	4	5	6	8	10
*iupr-dfki.fsift	NDCR	8.327	4.673	3.381	2.274	3.214	2.000	5.100	4.902
	TP	19	63	119	42	113	77	104	61
	F1	0.437	0.632	0.588	0.545	0.589	0.567	0.581	0.545
*iupr-dfki.fsift2	NDCR	6.886	5.162	7.439	1.505	5.556	2.573	10.348	7.374
	TP	12	55	119	31	114	72	102	59
	F1	0.468	0.616	0.588	0.528	0.586	0.567	0.588	0.543
SIFT	NDCR	8.884	6.009	6.975	4.555	4.587	5.305	10.439	7.436
	TP	16	56	110	37	115	64	63	51
	F1	0.443	0.594	0.556	0.541	0.551	0.569	0.507	0.532
SIFT+PV	NDCR	0.612	1.055	0.703	0.802	0.719	0.602	1.086	0.508
	TP	77	75	120	80	118	92	71	63
	F1	0.622	0.641	0.556	0.609	0.559	0.623	0.543	0.536
F-SIFT+PV	NDCR	0.552	0.969	0.703	0.763	0.826	0.440	0.560	0.552
	TP	71	86	120	85	118	99	111	71
	F1	0.622	0.666	0.591	0.592	0.592	0.596	0.594	0.573

*: officially submitted run.

Table 3. Time costs for SIFT and F-SIFT runs (s)

	1	2	3	4	5	6	8	10
SIFT	88.7	139.5	143.6	116.9	117.2	107.9	134.8	113.1
F-SIFT	65.1	86.4	90.5	86.1	79.5	80.4	85.6	79.8

have more than one duplicates in the reference set as suggested by the organizer. We also ignore *7* queries which are visually duplicate to the same reference video however they are not included in the ground-truth[4]. The performances are evaluated with NDCR, the number of true positives (TP) and F1 on the localization accuracy.

As seen from Table 1 and Table 2, as to the runs without post verificaton, “iupr-dfki.fsift” achieves apparently better performances in transformation 8 in which flip transformation is included. Moreover, this F-SIFT run outperform SIFT across most of the transformation types. This is mainly due to the use of dominant curl to filter out false visual word matches. On the other hand, as indicated in “iupr-dfki.fsift2” choosing top *2* detected videos turns out to be an unsuccessful try, it simply brings in more false positives.

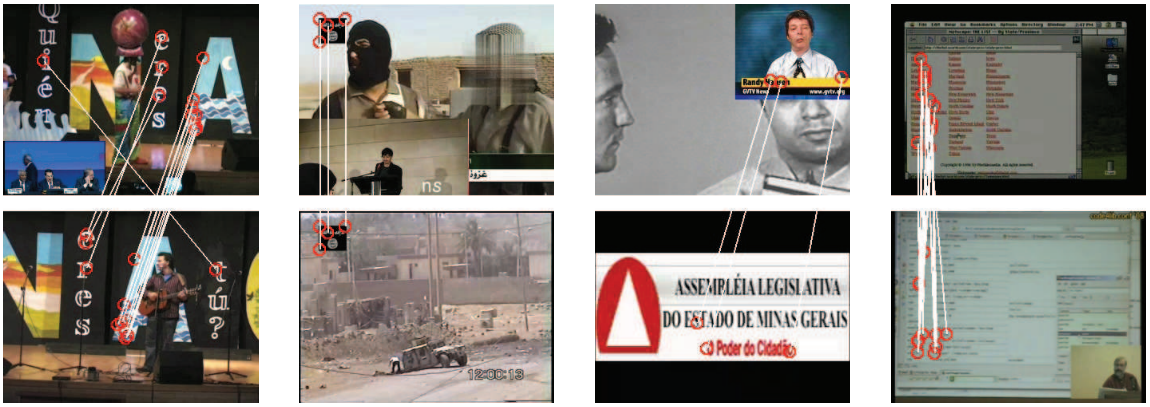
According to NDCR measure, the system runs exhibit high NDCR scores when they are mixed with few false positives with high confidence value. Typically, there are four types of false alarms in the runs when post validation by OOS+SR-PE is not involved. They are illustrated in Figures (a)-(d). Among these types of false alarm, type c and d can be easily removed after OOS+SR-PE verification. As demonstrated in the runs with post verification (“SIFT+PV” and “F-SIFT+PV”), NDCR scores have been dropped to pretty low level. Although, “SIFT+PV” and “F-SIFT+PV” are both benefit from the post verification, “F-SIFT+PV” still outperforms ‘SIFT+PV’ across most of the transformations as it is able to return more true-positives.

All our runs were carried out on a PC with *7.8G* memory and four processors with *2.8GHz* for each. We only open up one process throughout all our simulations and no parallel computing is considered in the code. During the simulation, it takes up less than *4G* memory to accommodate the whole reference set. Since all our five runs are actually built upon two individual runs, namely SIFT and F-SIFT runs, Table 3 summarizes the processing time for SIFT and F-SIFT runs. Due to the use of dominant curl verification, F-SIFT run only takes *84.6s* on average for one query which is close to the average duration of the query (*71.9s*). If we only consider the time cost at BoW retrieval stage, we found F-SIFT run only takes *43.8s* in average.

4. Discussion

For SIN - as compared to last year’s participation - we changed our feature extraction to be more speed efficient by using low dimensional Fisher BoW features instead of a high dimensional bag of viswords constructed from pure SIFT descriptors. The results indicate that the gained performance speedup is bought with a less robust content de-

[4] Query ID: 262/532/667/700/893/981/1553, Reference ID: 5757.



(a) Duplicate in background (b) Duplicate Logo (c) Similar text pattern (d) Similar texture pattern

Figure 4. Typical false alarms returned by the system. For false alarm types a and b, they turn out to be inevitable in our system since they are partially duplicate due to same backgrounds or logos. In terms of false alarms from types c and d, they are mainly due to imprecise visual word matching introduced by quantization errors in BoW.

scription and lower performance. Further, the evaluation of the color correlogram based concept detection indicate a reasonable balance between computational performance and detection robustness.

In this year's CCD task, we tested the flip invariant SIFT, it turns out to be a more favorable descriptor than SIFT for video copy detection in terms of both detection effectiveness and speed efficiency. We also demonstrated how well we can really achieve when only a single image local feature is involved. We found that with the presented framework, F-SIFT and OOS+SR-PE post validation, low NDCR score is still achievable with relatively low computing resources comparing with many other existing systems in the literature.

References

[1] ComScore August 2011 U.S. Online Video Rankings. available from `http://www.comscore.com/Press_Events/Press_Releases/2011/9/comScore_Releases_August_2011_U.S._Online_Video_Rankings` (retrieved: Sep'11).

[2] YouTube Blog: YouTube Videos now served in WebM. available from `http://youtube-global.blogspot.com/2011/04/mmm-mmm-good-youtube-videos-now-served.html` (retrieved: Sep'11).

[3] YouTube's Content ID (C)ensorship Problem Illustrated. available from `https://www.eff.org/deeplinks/2010/03/youtubes-content-id-c-ensorship-problem` (retrieved: Nov'11).

[4] R. Akbani, S. Kwek, and N. Japkowicz. Applying Support Vector Machines to Imbalanced Datasets. In *Proc. Europ. Conf. Machine Learning*, pages 39–50, 2004.

[5] S. Ayache and G. Quenot. Video Corpus Annotation using Active Learning. In *Proc. Europ. Conf. on Information Retrieval*, pages 187–198, 2008.

[6] D. Borth, A. Ulges, M. Koch, and T. Breuel. DFKI and University of Kaiserslautern Participation at TRECVID 2010 - Semantic Indexing Task. In *Proc. TRECVID Workshop*, December 2010.

[7] D. Borth, A. Ulges, C. Schulze, and T. Breuel. Keyframe Extraction for Video Tagging and Summarization. In *Proc. Informatiktage 2008*, pages 45–48, 2008.

[8] C.-C. Chang and C.-J. Lin. *LIBSVM: A Library for Support Vector Machines*, 2001.

[9] M. Douze, A. Gaidon, H. Jégou, M. Marszatke, and C. Schmid. INRIA-LEAR's video copy detection system. In *NIST TREVCID Workshop*, 2008.

[10] J. Hofmann and M. Ali. An extensive approach to content based image retrieval using low- and high-level descriptors. Diploma Thesis, University of Gteborg, Sweden, 2006.

[11] J. Huang, S.R. Kumar, M. Mitra, W.J. Zhu, and R. Zabih. Image Indexing Using Color Correlograms. In *Proc. Int. Conf. on Pattern Recognition*, page 762, 1997.

[12] H. Jegou, M. Douze, and C. Schmid. Hamming embedding and weak geometric consistency for large

scale image search. In *Proc. European Conf. on Computer Vision*, pages 304–317, Oct. 2008.

[13] H. Jégou, M. Douze, C. Schmid, and P. Pèrez. Aggregating local descriptors into a compact image representation. In *CVPR*, pages 3304–3311, 2010.

[14] W. Kraaij. TRECVID-2008 Content-based Copy Detection Task: Overview. In *Proc. TRECVID Workshop*, November 2008.

[15] D. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *Int. J. Comput. Vis.*, 60(2):91–110, 2004.

[16] B. Manjunath, J.-R. Ohm, V. Vasuvedan, and A. Yamada. Color and Texture Descriptors. *IEEE Trans. Circuits and Systems for Video Technology*, 11(6):703–715, 2001.

[17] G. Schwarz. Estimating the Dimension of a Model. *Ann. of Stat.*, 2(6):461–464, 1978.

[18] Alan F. Smeaton, Paul Over, and Wessel Kraaij. High-Level Feature Detection from Video in TRECVid: a 5-Year Retrospective of Achievements. In *Multimedia Content Analysis, Theory and Applications*, pages 151–174. 2009.

[19] C. Snoek and M. Worring. Concept-based Video Retrieval. *Foundations and Trends in Information Retrieval*, 4(2):215–322, 2009.

[20] J. Zhang, M. Marszalek, S. Lazebnik, and C. Schmid. Local Features and Kernels for Classification of Texture and Object Categories: A Comprehensive Study. *Int. J. Comput. Vis.*, 73(2):213–238, 2007.

[21] W.-L. Zhao. A comprehensive study over flip invariant SIFT. *Technical report*, May 2011.

[22] W.-L. Zhao and C.-W. Ngo. Scale-rotation invariant pattern entropy for keypoint-based near-duplicate detection. *IEEE. Trans. on Image Processing*, 18(2):412–23, Feb. 2009.

[23] Wan-Lei Zhao, Xiao Wu, and Chong-Wah Ngo. On the annotation of web videos by efficient near-duplicate search. *IEEE Trans. on Multimedia*, 12(5):448–461, 2010.