ECL-LIRIS at TRECVID 2011: Semantic Indexing

Chao Zhu, Boyang Gao, Charles-Edmond Bichot, Emmanuel Dellandréa, Liming Chen, Ningning Liu, and Yu Zhang

Université de Lyon, CNRS,
Ecole centrale de Lyon, LIRIS, UMR5205, F-69134, France.
{chao.zhu, boyang.gao, charles-edmond.bishot, emmanuel.dellandrea, liming.chen, ningning.liu, yu.zhang}@ec-lyon.fr

ABSTRACT

This is the first time that our team participate TRECVID. This paper summarizes our approach submitted to Semantic Indexing (SIN) task in TRECVID 2011. Our approach adopts bag-of-features method to transform original visual and audio features into histogram features, using pre-trained codebook. After feature transformation, one-versus-others SVMs with Chi-square kernel are trained. In decision step, averaged probability is calculated as a final score to rank shots. Under this framework, we tested 4 visual features including dense grid SIFT, color SIFT, OLBPC and DAISY together with 1 audio feature consisting of MFCC with delta and acceleration. Our audio visual combination model achieves best results in terms of mean xinfAP. Besides, considering the huge amount of data this year, we employed several speedup strategies such as k-means clustering with GPU acceleration and homogeneous kernel map. All these efforts rank us at the 12th out of 19 teams in full run and the 13th out of 27 teams in the light run test.

1. INTRODUCTION

The aim of semantic indexing (SIN) task in TRECVID[1][2] is to automatically analyze the meaning conveyed by videos and tag videos with semantic concept labels. To evaluate various systems TRECVID provides carefully labeled corpus to participants. The biggest challenges of semantic index task on one hand reside in the extra-large scale of dataset not only in length of videos, but also in number of semantic concepts to cope with. In SIN task of TRECVID 2011, development and test set reach up to 400 hours and 200 hours respectively. Corresponding concept number increases to 346. On the other hand, all videos are collected from Internet, which means we are facing real world data with unpredictable quality and content. All of these demand an effective, efficient and robust system. Bag-of-features framework has demonstrated its efficiency in image classification and has been widely used by teams participating TRECVID [3][4][5]. In the following sections our bag-of-features based approach is explained in detail.

2. FEATURE EXTRACTION

2.1. Data pre-processing

FFmpeg[6] is used to decode video files. According to master shot boundary and mp7 descriptions, for each shot, we keep the single key frame and 2 seconds audio wave around the key frame, 1 second before and after. Because of the time limitation we did not try multi-frame approach which preserves video's sequential information. Instead we use the 2 seconds audio waves to compensate.

2.2. SIFT

Scale invariant feature transform (SIFT) proposed by Lowe [7] has demonstrated strong ability in image classification task. However, the original SIFT depends largely on the quality of key points (interest points). According to our experiment on TRECVID 2010 dataset, dense grid SIFT outperforms original key points SIFT by more than 5 times in terms of mean xinfAP. Therefore, we applied dense grid SIFT in this year's task. Color SIFT[8] is also used to incorporate color information.

2.3. OLBPC

LBP[9] is an important texture feature, However, in original LBP, the size of histogram grows exponentially with respect to number of neighbor pixels. For example, the size of the LBP histogram will be 256/65536 if 8/16 neighboring pixels are considered. Thus, a dimensionality reduction method for the LBP is needed to investigate when more neighboring pixels are concerned. We propose a new orthogonal local binary pattern combination (OLBPC) [10] to reduce the histogram size and at the same time preserve information on all neighboring pixels. As illustrated in Figure 1, instead of encoding local patterns on 8 neighbors, we perform encoding on two sets of 4 orthogonal neighbors, resulting two independent codes [OLBP1 OLBP2]. Concatenating

and accumulating two codes leads to a final LBP histogram of 32 dimensions, which is much more compact than the original one (256 dimensions).

Since the neighboring pixels used in each unit LBP operator are orthogonal in position, we denote this method as orthogonal LBP combination. It could be generalized to the LBP operators with more neighboring pixels, and the general process is as follows. The neighboring pixels of the original LBP is firstly split into several non-overlapped orthogonal groups, then the LBP code is computed separately for each group, and finally these codes are concatenated together as the new LBP code.

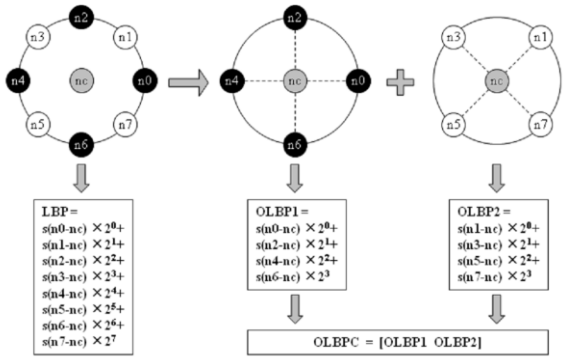

Figure 1. Calculation of the LBP and OLBPC with 8 neighboring pixels

2.4. DAISY

Similar to SIFT, DAISY[11] descriptor is a 3D histogram of gradient locations and orientations. The differences between them lie in two aspects. One is that DAISY replaces the weighted sums of gradient norms used in SIFT by convolutions of gradients in specific directions with several Gaussian filters. The other is that DAISY uses a circular neighborhood configuration instead of the rectangular one used in SIFT, as the comparison shown in Figure 2.

As argued in [12], DAISY outperforms SIFT with a shorter descriptor length. When having similar classification accuracy, DAISY's size is only 1/6 of SIFT, resulting extraction runs 12 times faster than SIFT.

In our submission, all 4 visual features are extracted on dense grid with multi-scale neighborhood.

2.5. MFCC

Mel-frequency cepstral coefficients (MFCC)[13] have been successfully applied in many branches of audio signal processing for example speech recognition, speaker identification and audio similarity measure. MFCC simulates human ears' sensation of sound. Together with delta and acceleration, MFCC preserves both static and dynamic information of audio signal. OpenSMILE[14][15] is used for computing 39 dimensional MFCC with delta and acceleration features.

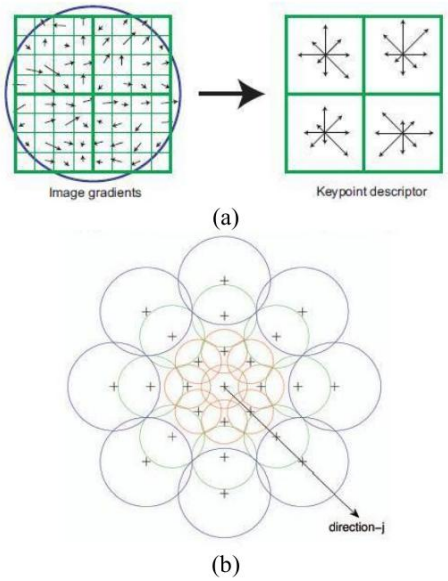

Figure 2. Comparison of SIFT and DAISY descriptor shapes. (a) SIFT uses a rectangular grid [4], (b) DAISY considers a circular configuration [7], where the radius of each circle is proportional to its distance from the center.

2.6. Codebook training and looking-up

For visual features, classical k-means clustering is used and histogram feature is accumulated by hard assignment after looking up codebook. To take advantage of our 32-cored server, k-means clustering was performed by Yael[16] with multithreading implementation.

Audio feature codebook generation and codebook looking-up are slightly different from visual ones. Because of the effectiveness of Gaussian mixture model (GMM) for describing audio feature space, we use GMM as codebook and perform soft assignment according to posterior probability. However, training GMM with EM consumes unaffordable amount of time on TRECVID data, even with GPU acceleration. Instead, we omit EM step and generate GMM directly from k-means clustering centers, since means, variances and weights for GMM can be estimated directly from clusters. Thanks to our GPU accelerated k-means implementation, audio codebook training and looking-up finished within one day.

3. CLASSIFICATION

We choose support vector machine (SVM) with Chi-square kernel as our classifier for each of 5 features. To avoid pair wise Chi-square kernel function computing which consumes tremendous amount of time and space, we adopted

homogenous kernel map (HKM)[17][18] method. Homogeneous kernel map transforms original vector into higher dimensional space where vector inner product approximately equals to Chi-square kernel function calculation in original space. VLFeat[19] provides a neat implementation of HKM.

After performing HKM on histogram feature vectors, a fast linear SVM solver like liblinear[20] suffices to train classifiers. For multiclass classification tasks unbalanced training data is inevitable. To compensate highly insufficient positive instances, we cast penalty weights according to positive and negative proportion.

In the final step, we adopt a simple post-fusion strategy which directly averages probabilities output by all SVMs.

4. RESULTS

Our overall and individual mean xinfAP results of full-run are shown in Figure 3 and 4 in which "I" denotes visual (image) feature result, "I+A" denotes visual (image) plus audio feature result, "mean" denotes the average result of all submissions and "max" denotes maximum result achieved by some submission. From returned results we can find that our submission A_ecl_liris_IA_4 which incorporates visual and audio features achieves best result in terms of mean xinfAP. In 50 categories tested in full-run, 42 have been improved by fusing audio feature classifier. A similar improvement can also be observed in light-run results shown in Figure 5 and 6.

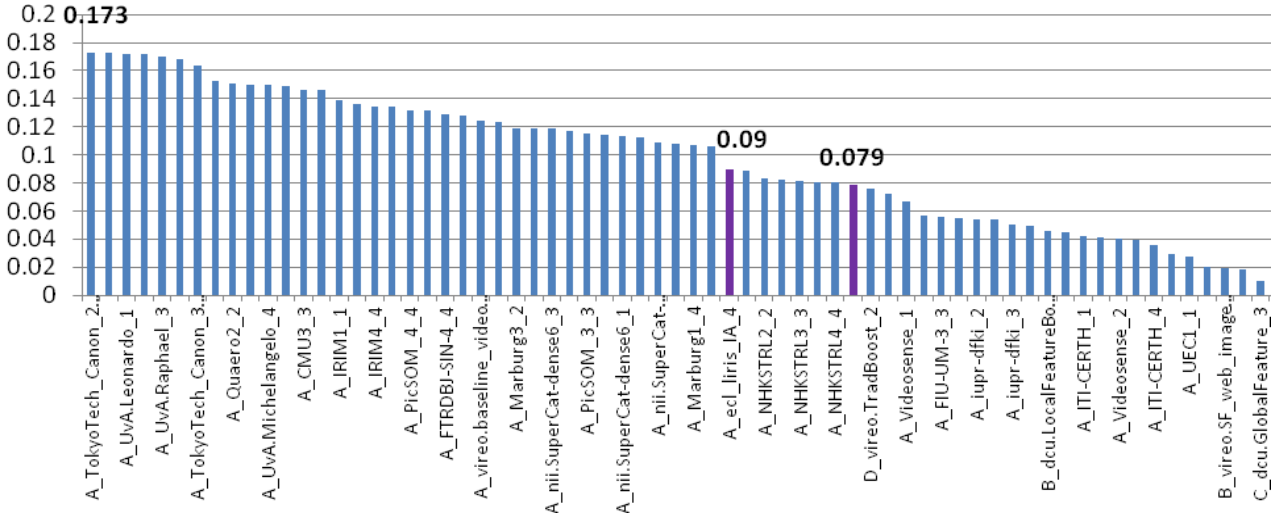

Figure 3 Full-run submission (two in purple) mean xinfAP (Y axis) in all 68 submissions (X axis)

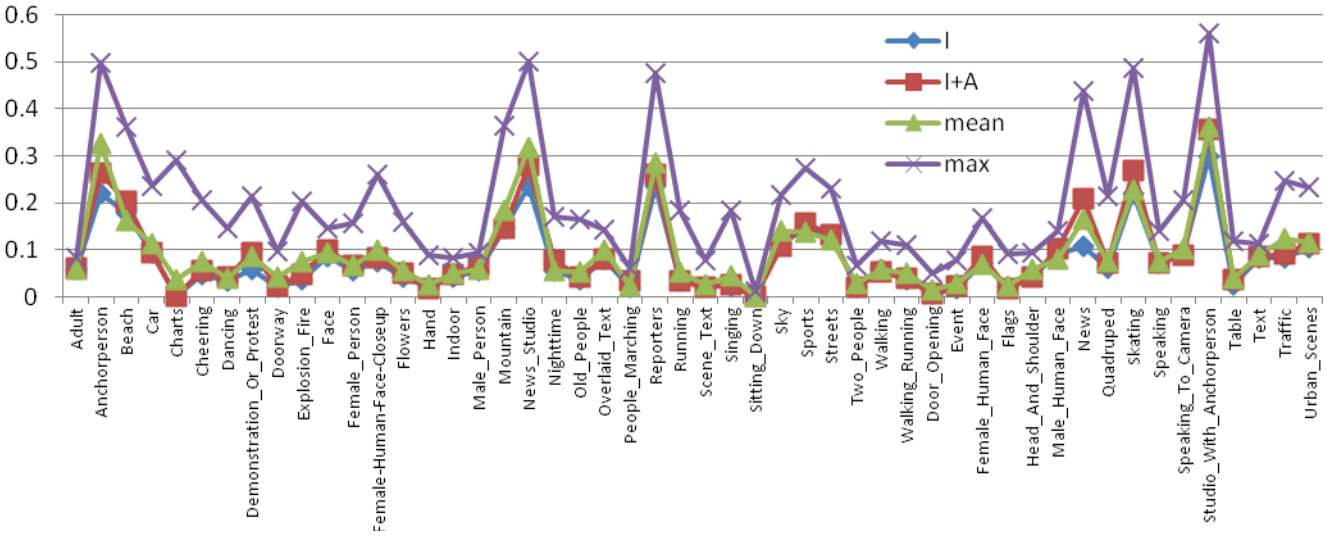

Figure 4 Full-run submission mean xinfAP (Y axis) of all 50 categories (X axis)

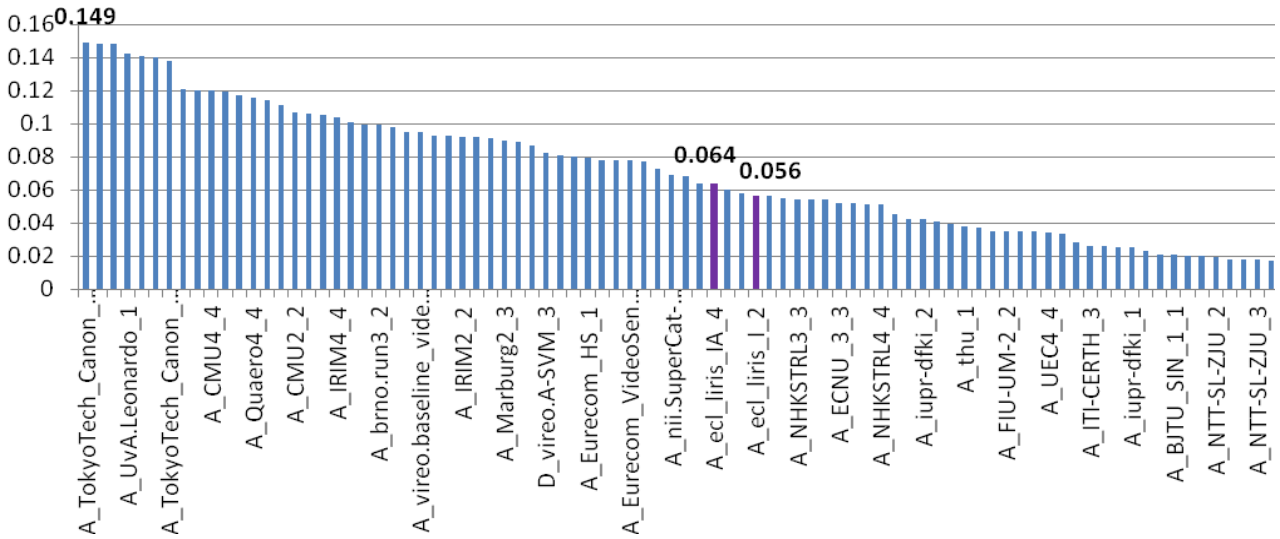

Figure 5 Light-run submission (two in purple) mean xinfAP (Y axis) in all 102 submissions (X axis)

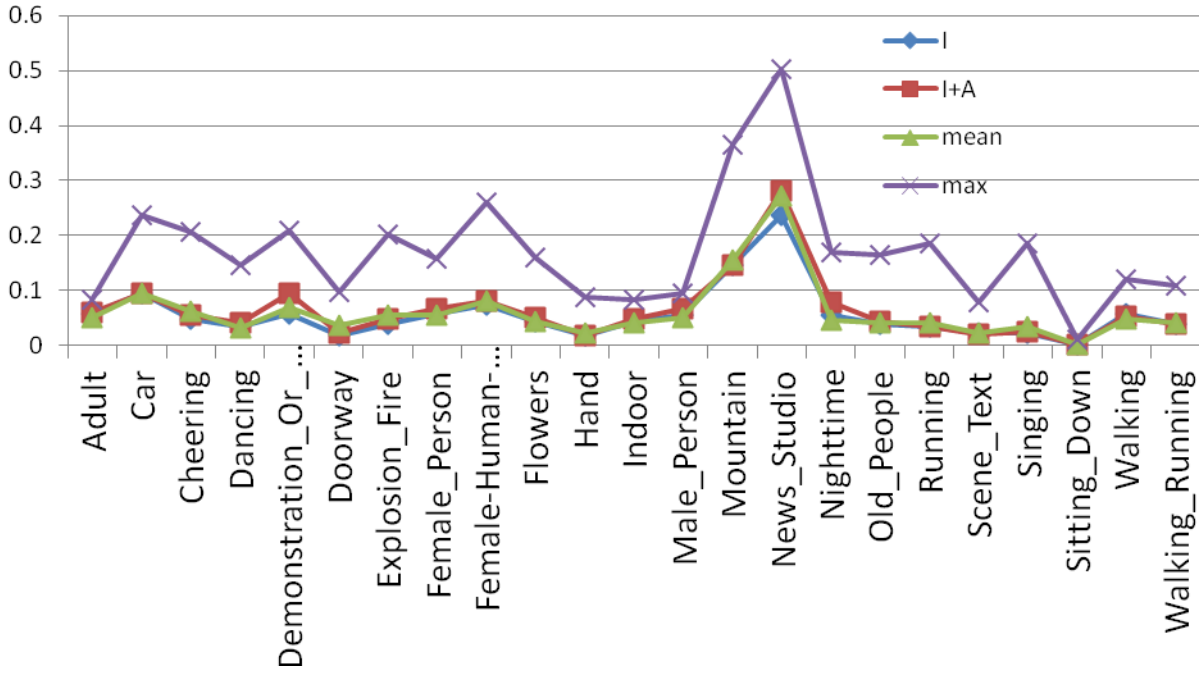

Figure 6 Light-run submission mean xinfAP (Y axis) of all 20 categories (X axis)

5. REFERENCES

[1] A. F. Smeaton, P. Over, W. Kraaij, "Evaluation campaigns and TRECVid," In *Proc. the 8th ACM International Workshop on Multimedia Information Retrieval*, pp.321-330, 2006

[2] A. F. Smeaton, P. Over, W. Kraaij, "High-Level Feature Detection from Video in TRECVid: a 5-Year Retrospective of Achievements," in *Multimedia Content Analysis, Theory and Applications*, pp. 151-174, 2009

[3] D. Gorisse1, F. Precioso, P. Gosselin1, L. Granjon, D. Pellerin, M. Rombaut, H. Bredin, L. Koenig, R. Vieux, B. Mansencal, J. Benois-Pineau, H. Boujut, C. Morand, H. Jégou, S. Ayache, B. Safadi, Y. Tong, F. Thollard,

G. Quénot, M. Cord, A. Benoît, PLambert, "IRIM at TRECVID 2011: Semantic Indexing and Instance Search," TREVID 2011 notebook paper.
[4] L. Bao, S.-I. Yu, Z.-Z. Lan, A. Overwijk, Q. Jin, B. Langner, M. Garbus, S. Burger, F. Metze, A. Hauptmann, "Informedia @ TRECVID 2011," TREVID 2011 notebook paper.
[5] C.G.M. Snoek, K.E.A. van de Sande, X. Li, M. Mazloom, Y.-G. Jiang, D.C. Koelma, A.W.M. Smeulders, "The MediaMill TRECVID 2011 Semantic Video Search Engine," TREVID 2011 notebook paper.
[6] http://ffmpeg.org/
[7] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," in *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91-110, 2004
[8] G. J. Burghouts, J. M. Geusebroek. "Performance evaluation of local colour invariants," in *Computer Vision and Image Understanding*, 113:48-62, 2009.
[9] T. Ojala, M. Pietikainen, D. Harwood, "A comparative study of texture measures with classification based on feature distribution," in *Pattern Recognition* 29 (1996) 51-59.
[10] C. Zhu, C.-E. Bichot, L. Chen, "Color orthogonal local binary patterns combination for image region description," in *Technical Report*, LIRIS UMR5205 CNRS, Ecole Centrale de Lyon, 2011.
[11] E. Tola, V. Lepetit, P. Fua, "Daisy: an Efficient Dense Descriptor Applied to Wide Baseline Stereo," *IEEE PAMI*, vol. 32, no. 5, pp. 815-830, 2010.
[12] C. Zhu, C.-E. Bichot and L. Chen, "Visual object recognition using daisy descriptor," In *Proc. IEEE International Conference on Multimedia and Expo (ICME)*, Jul. 2011.
[13] S. B. Davis, P. Mermelstein, "Comparison of Parametric Representations for Monosyllabic Word Recognition in Continuously Spoken Sentences," in *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 28(4), pp. 357–366, 1980.
[14] F. Eyben, M. Wöllmer, B. Schuller: "openSMILE - The Munich Versatile and Fast Open-Source Audio Feature Extractor," in *Proc. ACM Multimedia (MM)*, ACM, Firenze, Italy, 25.-29.10.2010.
[15] http://www.openaudio.eu/
[16] https://gforge.inria.fr/projects/yael
[17] A. Vedaldi, A. Zisserman, "Efficient additive kernels via explicit feature maps," In *Proc. CVPR*, 2010.
[18] A. Vedaldi, B. Fulkerson, "VLFeat: An open and portable library of computer vision algorithms," 2008.
[19] http://www.vlfeat.org/
[20] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin. "LIBLINEAR: A library for large linear classification," in *Journal of Machine Learning Research* 9(2008), 1871-1874.