Florida International University - University of Miami TRECVID 2013

Tao Meng[1], Guiru Zhao[1], Fausto Fleites[2], Miguel Gavidia[3], Tarek Sayed[1], Yang Liu[1], Mei-Ling Shyu[1],
Hsin-Yu Ha[2] and Shu-Ching Chen[2]

[1]Department of Electrical and Computer Engineering
University of Miami, Coral Gables, FL 33146
[2]School of Computing and Information Sciences
Florida International University, Miami, FL 33199
[3]Department of Computer Science
University of Miami, Coral Gables, FL 33124
*tmeng@umiami.edu, zhaoguiru@gmail.com, fflei001@cs.fiu.edu, {m.gavidia, t.sayed}@miami.edu,
y.liu39@umiami.edu, shyu@miami.edu, {hha001, chens}@cs.fiu.edu*

Abstract

This paper demonstrates the framework and results from the team "Florida International University - University of Miami (FIU-UM)" in TRECVID 2013 Semantic Indexing (SIN) task [14, 15]. Four runs were submitted, and the summary of these four runs are as follows:

- M_A_FIU-UM-1_1*: SVM - Support vector machine (SVM) based ranking using key frame (KF) features.*
- M_A_FIU-UM-2_2*: SVM+DASD - SVM based ranking using KF features, domain adaptive semantic diffusion (DASD) is applied to refine results of this round.*
- M_A_FIU-UM-3_3*: SVM+AAN - SVM based ranking using KF features, concept association network (CAN) is applied to refine results of this round.*
- M_A_FIU-UM-4_4*: Fusion of the results generated from three models corresponding to the aforementioned three runs. In addition, the results of deformable part model for some concepts are also integrated.*

In run 1, 2 and 3, the same baseline SVM-based model is applied. Different re-ranking approaches such as DASD and CAN are applied to refine the initial results. In this way, we want to test whether re-ranking approach could help the baseline results. In Run 4, the results from the aforementioned three runs are fused. The deformable part model [7] is utilized to detect some concepts such as chair. As a result, from the submission results, run 4 outperforms other three runs.

1 Introduction

In TRECVID 2013 project, the semantic indexing (SIN) task aims to recognize the semantic concept contained within a video shot which has several challenges such as data imbalance, scalability, and semantic gap. The automatic annotation of semantic concepts in video shots can be an essential technology for retrieval, categorization, and other video exploitations. The semantic concept retrieval research directions contain developing robust learning approaches that adjust to the increasing size and the diversity of the videos, fusing information from other sources such as audio and text, and detecting low-level and mid-level features that have high discriminability.

Compared to SIN task of the last year, the size of high-level semantic concepts decreases in this year, totally 60 concepts. For each of the 60 semantic concepts, the participants are allowed to submit a maximum of $2,000$ possible shots, and the submission result is rated by using mean inferred average precision (mean xinfAP) [18].

This paper is organized as follows. Section 2 describes our proposed framework and the specific approaches utilized for each run. Section 3 shows submission results in detail. Section 4 summarizes the whole paper and proposes some future directions to pursue.

2 The Proposed Framework

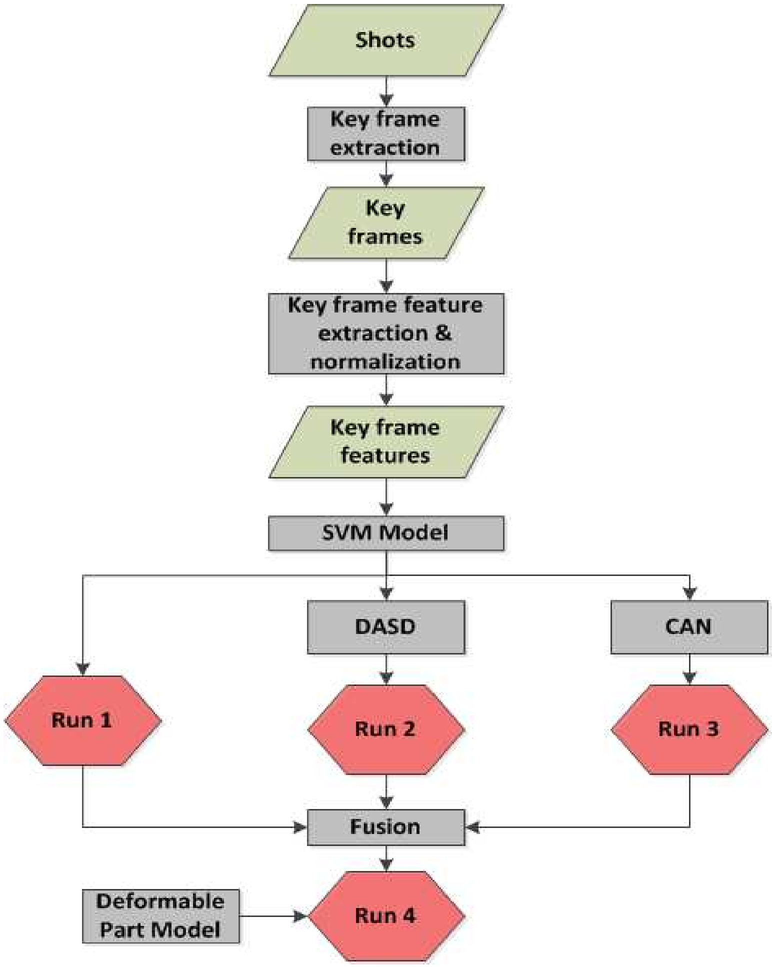

Figure 1. The whole framework for semantic indexing

The global framework of TRECVID 2013 SIN task is shown in Figure 1. Key frame level features (KF) are extracted and normalized. For Run 1, the SVM model is applied and no re-ranking approach is utilized. For Runs 2 and 3, the DASD and CAN are applied. The xinfAP values are calculated from models trained on TRECVID 2012 training data and evaluated on TRECVID 2012 testing data. In Run 4, the results from Run 1 to 3 are fused.

2.1 Data Pre-processing and Feature Extraction

A key frame for each shot is provided to SIN task participates in both training and testing videos. Ten kinds of KF features are extracted from each frame in training and testing data, including color moment in the YCbCr space [16], color histogram in the HSV space, canny edge histogram, sobel edge histogram, texture co-occurrence, color and edge directivity descriptor (CEDD) [5], histogram of oriented gradients (HOG) [6], Gabor wavelets [9], haar wavelets [17], and local binary patterns (LBP) [13]. Before extracting features, histogram equalization is employed to regulate the contrast of frames.

2.2 Support Vector Machine

The SVM is a supervised pattern classification model [3]. It is extended from the maximal margin classifier and utilizes the kernel trick to map the data instances to a higher dimensional space, where they could be classified easier than the original space. The main idea of SVM is to find the optimal hyperplane to maximize the distance between the hyperplane and the support vectors of the two classes. Specifically, given the training data set of the instance-label pairs $(x^{(i)}, y^{(i)})$, $i = 1, 2, ..., m$, where m is the total number of data instances, $x^{(i)} \in R^n$ and $y^{(i)} \in \{1, -1\}$ indicating the instance is positive or negative. The SVM model requires the solution of the following optimization problem:

$$\underset{w,b,\xi}{\text{argmin}} \frac{1}{2} w^T w + Q \sum_{i=1}^{m} \xi^{(i)} \tag{1}$$

$$\text{subject to } y^{(i)}(w^T \varphi(x^{(i)}) + b) \geq 1 - \xi^{(i)}$$

$$\xi^{(i)} \geq 0$$

The training vectors $x^{(i)}$ are mapped to a higher dimensional space using the function φ. Q and $\xi^{(i)}$ are two parameters. In order to apply the kernel trick, the kernel function $H(x^{(i)}, x^{(j)}) \equiv \varphi(x^{(i)})^T \varphi(x^{(j)})$ is defined. The RBF kernel in Equation (2) is applied in this paper. More details of SVM could be found in [3] and are skipped here.

$$H(x^{(i)}, x^{(j)}) = exp(-\gamma \left\| x^{(i)} - x^{(j)} \right\|^2). \tag{2}$$

The implementations of the SVM classification model used in this paper is the LIBSVM [4] package which is an off-the-shelf SVM software implementation.

2.3 Domain Adaptive Semantic Diffusion

In order to model the inter-concept relationship, one re-ranking approach which is the domain adaptive semantic diffusion approach (DASD) is applied [8]. The general idea of DASD is that the detection scores of contextually correlated concepts should be harmonic. The authors proposed a graphic model so that each node represents a concept detector and the edge between two nodes represents the connection between two concepts. Such a graphic model is then applied to refine the detection scores output from SVM using a function level diffusion process,

Table 1. Label matrix

Instance	C_1	C_2	...	C_k	...	C_n
Instance 1	C_1^0	C_2^1	...	C_k^1	...	C_n^0
Instance 2	C_1^0	C_2^1	...	C_k^0	...	C_n^1
...	...	...	...	...	...	...
Instance i	C_1^1	C_2^0	...	C_k^0	...	C_n^1
...	...	...	...	...	...	...
Instance m	C_1^0	C_2^1	...	C_k^1	...	C_n^0

where the purpose is to enhance the consistency of detection scores. Formally, the cost function of DASD is defined as:

$$E(\boldsymbol{g}, \boldsymbol{W}) = \frac{1}{2} \sum_{ij} W_{ij} \left\| g(c_i) - g(c_j) \right\|^2 \tag{3}$$

Here, $g(c_i)$ and $g(c_j)$ are the decision score vectors over a set of testing samples. W_{ij} indicates the affinity, which quantifies the connection between concepts. The minimization of this function leads to the consistency of neighboring nodes. The weights could be learned using the gradient descent algorithm. The specific details of DASD can be found in [8].

2.4 Concept Association Network

The concepts have some correlations in the TRECVID 2013 data sets since they do not occur independently. In order to model the correlations among different concepts, we utilize the concept association network [10][11][12] in this project. The proposed framework is shown in Figure 2 (training phase) and Figure 3 (testing phase). The training phase consists of the *Concept Based Classifiers Training Component* and the *Concept Association Network Training Component*. The former is the architecture of the concept detection framework proposed in our work. For example, in a training data set, there are m instances and n high-level ($n = 60$ in our case) concepts to detect. The training instances are pre-processed and a set of features are extracted. Next, n binary content-based classifiers such as the subspace-based models or the MCA-based classifiers are trained for n concepts, so that each model k ($1 \leq k \leq n$) outputs m scores for the $k-th$ concept, represented by C_k in the figure. The *Concept Association Network Training Component* receives the scores from the *Concept Based Classifiers Training Component* and discovers the frequent itemsets in the label matrix to build a Concept Association Network (CAN). The detailed steps of building the CAN are introduced as follows.

First, all the labels of the training instances are organized into a label matrix. Specifically, the labels of all the m instances for the n high-level concepts are organized into a label matrix $\boldsymbol{L} = \{l_{ik}\}$, $i = 1, 2, ..., m$ and $k = 1, 2, ..., n$, where $l_{ik} = C_k^1$ or $l_{ik} = C_k^0$ indicates the i-th instance is labeled as positive or negative for the k-th concept. Table 1 shows an example of a label matrix.

Next, the association links among different concepts are generated by mining the significant rules from the label matrix. The the Apriori algorithm [1] is applied to the label matrix to discover the association rules. The specific algorithm to generate all the 2-item rules works as follows. First, all 1-$itemsets$ are generated for $\boldsymbol{L}$. Only the 1-$itemsets$ $\{C_k^1\}$ consisting of positive concept-class pairs are retained. Second, all the candidate 2-$itemsets$ are generated by combining the 1-$itemsets$ with a minimum support of one. Afterwards, the candidate 2-$itemsets$ which contain the concept of interest are organized together. Based on these 2-$itemsets$, a set of candidate rules which draw the conclusion that the concept of interest is positive are generated. In order to select the most significant rules, two rule pruning modules are incorporated into the framework.

In this work, we only take into consideration of the binary cooccurence relationships between concepts. Therefore, only the 2-item rules are considered. In order to retain the most significant rules, the support ratio and the interest ratio are used to select rules. Formally, let C_t represent the target concept which is the concept of interest and C_r represent the reference concept which is the concept used to help the detection of the target concept; $sup(X)$ represent the support value of the itemset X. The support ratio and interest ratio are defined in Equation (4) and Equation (5), respectively. Intuitively, these criteria represent the rule selection from the target concept point of view and the reference concept point of view. The theoretical justification of these rules is in [10]. The thresholds for the support ratio and the interest ratio are determined using the cross validation process.

$$R_s = \frac{sup(\{C_t^1, C_r^1\})}{sup(\{C_t^1\})}. \tag{4}$$

$$R_i = \frac{sup(\{C_t^1, C_r^1\})}{sup(\{C_t^1\}) \times sup(\{C_r^1\})}. \tag{5}$$

From the network point of view, if all the relationships among concepts are modeled in a network G=$\{V, A, W\}$, where V is a set of nodes (each node representing a concept), A represents a set of links, and each link has a weight in set W to model the relationship between two nodes. The selected significant rules could be viewed as the significant links from the reference concepts to the target concept. These links are defined as the association links and form the core of the concept association network.

Because the output scores of different models could fall into different ranges, the raw scores are preprocessed to feed into the concept association network. In addition, the information of the credibility of the score is not included in the raw score. Therefore, the raw scores are converted to the probability based scores using the Bayes Rule. Assuming for an instance i, the detection score of C_k is $O(k, i)$. The output score $O'(k, i)$ for the C_k which encompasses the information of the credibility of the model is given in Equation (6).

$$O'(k, i) = \frac{p(O(k, i)|C_k = 1) \times p(C_k = 1)}{\sum_{z=0}^{1} p(O(j, i)|C_k = z) \times p(C_k = z)}, \tag{6}$$

where $p(C_k = 1)$ is the prior probability of C_k appeared in a data instance and is estimated by dividing the $sup(\{C_k^1\})$ by the total number of training instances. $p(C_k = 0)$ is one minus $p(C_k = 1)$ because there are only two possible cases. $p(O(k, i)|C_k = 1)$ is the conditional probability density function (pdf) $f_P(x) = p(x|C_k = 1)$ evaluated at $x = O(k, i)$, and $p(O(k, i)|C_k = 0)$ is the conditional pdf $f_N(x) = p(x|C_k = 0)$ evaluated at $x = O(k, i)$. To estimate $f_P(x)$ and $f_N(x)$, the Parzen-Window [2] approach is applied here.

The last step is to integrate the posterior probability scores from the reference concepts and the target concept properly to generate the final score. This process is called the fusion process, which is extremely important for the overall performance of the framework. In this work, the logistic regression model is utilized to fuse the outputs together. The details could be found in [10] and the weights for the links are learned utilizing the cross validation procedure. Then with all the learned wieghts, the concept association network is built by using the training instances.

In the testing phase, the set of features that are identical as in the training phase is first extracted. For each testing instance, it receives one score from each content-based classifier. For a target concept, by leveraging the concept association network, a new score which integrates the information from reference concepts is generated as the final output score.

2.5 Deformable Part Model

In this project, we utilized deformable part model [7] to detect some concepts. This model is one of the most popular models for object detection. Different from "bag of features" and "hog matching", which are based on object level conceptually, the deformable part model describes the location relationship between parts of models by part and deformable configuration. This model includes both a coarse global template covering an entire object

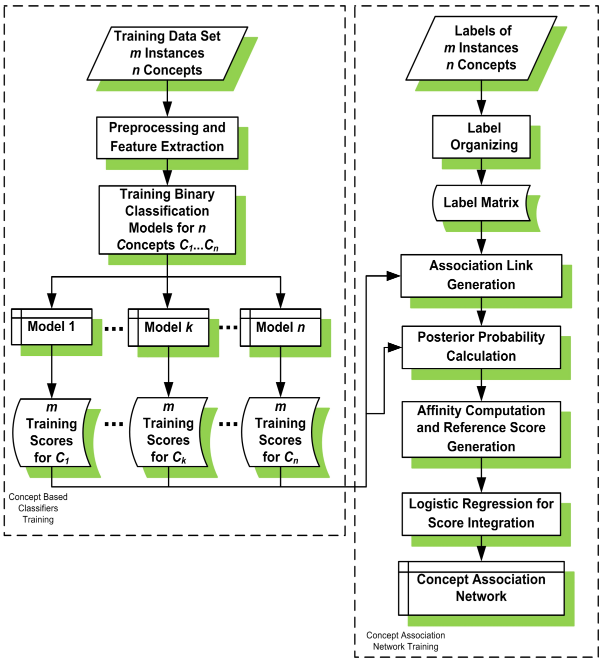
Training Data Set
m Instances
n Concepts
Preprocessing and Feature Extraction
Training Binary Classification Models for n Concepts $C_1...C_n$
Model 1
Model k
Model n
m Training Scores for C_1
m Training Scores for C_k
m Training Scores for C_n
Concept Based Classifiers Training
Labels of
m Instances
n Concepts
Label Organizing
Label Matrix
Association Link Generation
Posterior Probability Calculation
Affinity Computation and Reference Score Generation
Logistic Regression for Score Integration
Concept Association Network
Concept Association Network Training

Figure 2. Training phase of the proposed framework

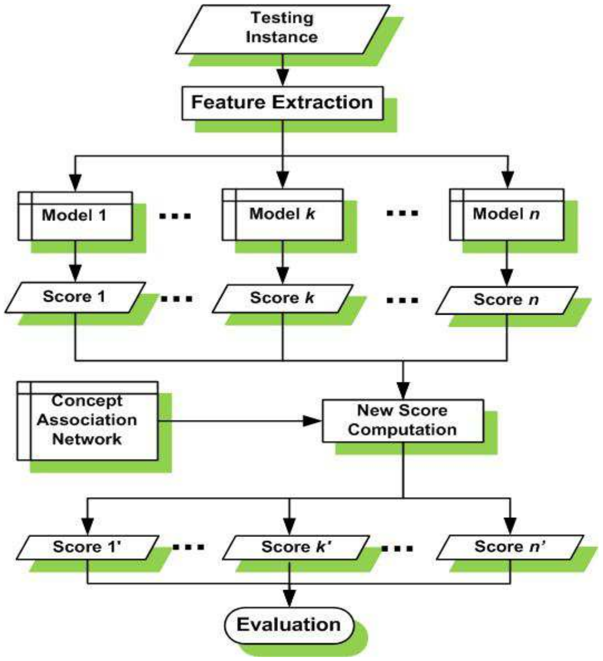

Figure 3. Testing phase of the proposed framework

and higher resolution part templates. In this model, an object is detected by considering two kinds of information, which are “a collection of parts” and “connection between parts”. Imaging for a spring model, each part of an object is linked by a spring. An energy function, which contains two parts: the matching degree of every part and the changings degree of the connection parts (like the deformation of the corresponding spring), is defined. The best image that matches the model is the one that can minimize the energy function. In the graphical representation, an undirected graph $G = (V, E)$ can be used to represent the object model. $V = \{V_1, V_2, \cdots, V_n\}$ is for n parts, while $(V_i, V_j) \in E$ represents the connection between two parts. A specific example of configuration of an object can be represented as $L = (l_1, l_2, \cdots l_n)$, l_i is for the location of v_i. Given an image, $m_i(l_i)$ can be used to measure the degree of mismatch between the model and v_i in location l_i, and $d_{ij}(l_i,l_j)$ can be used to measure the variety of v_i and v_j when they are located at l_i and l_j. Therefore, the best configuration of matching an image with a model is the one that can both match every part well and let the relative relationship among parts match the model best. The best configuration can be calculated by the following equation: $L^* = \arg\min_L(\sum_{i=1}^{n} m_i(l_i) + \sum_{(v_i,v_j)\in E} d_{ij}(l_i, l_j))$. We used 2009 template for Concept 13 (bicycling) for final submission in Run 4.

3 Experimental Results

The overall framework of TRECVID 2013 SIN task contains three stages:

1. Model training: use TRECVID 2012 training videos as training data.

2. Model evaluation: use TRECVID 2012 testing videos as testing data to evaluate the framework and tune parameters of the models.

3. Model testing: use TRECVID 2012 training + TRECVID 2012 testing videos as TRECVID 2013 training data, and TRECVID 2013 testing videos as testing data to generate the ranking results for submission.

Figure 4 to Figure 7 present the performance of our semantic indexing results. The x axis is the concept number while the y axis is the inferred average precision. More clearly, Table 2 shows the inferred mean average precision (MAP) values of the first 10, 100, 1000 and 2000 shots. The inferred true shots and mean xinfAP are shown in Table 3.

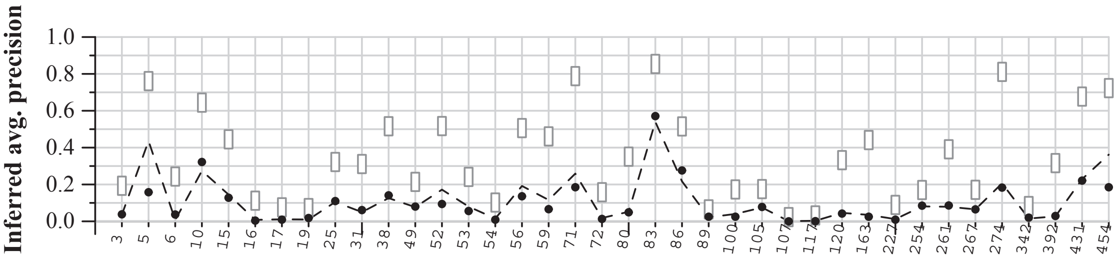

Figure 4. Run scores (dot) versus median (—) versus best (box) for *M_A_FIU-UM-1_1*

Evaluation results demonstrate that the performance of our model is around the median of all the groups. For some concepts such as concept 10 and 83, we give better performance. It is also noticed that the proposed CAN re-ranking model performs better than DASD, which is the state-of-the-art model in re-ranking field.

Based on the overall process of the task, We have the following obervations:

- The proper re-ranking strategy could improve the performance of the overall system.

- Our proposed CAN gives relatively good performance in terms of score re-ranking.

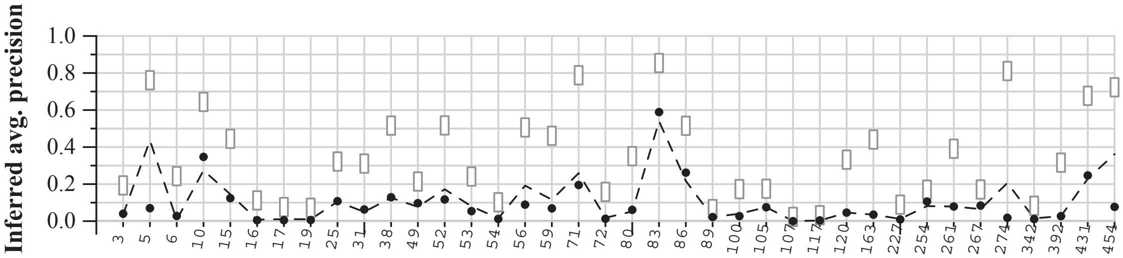

Figure 5. Run scores (dot) versus median (—) versus best (box) for *M_A_FIU-UM-2_2*

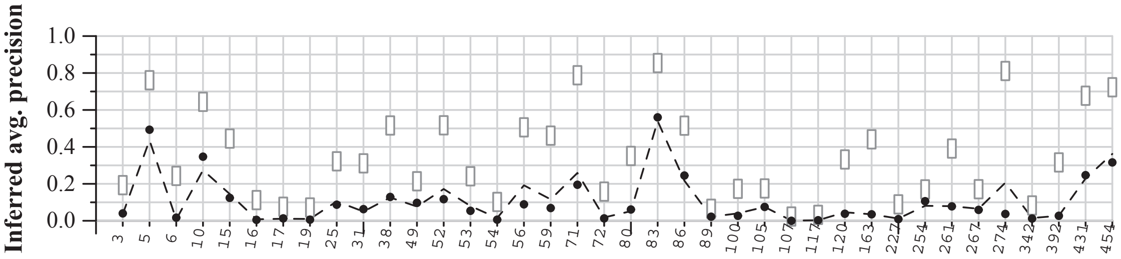

Figure 6. Run scores (dot) versus median (—) versus best (box) for *M_A_FIU-UM-3_3*

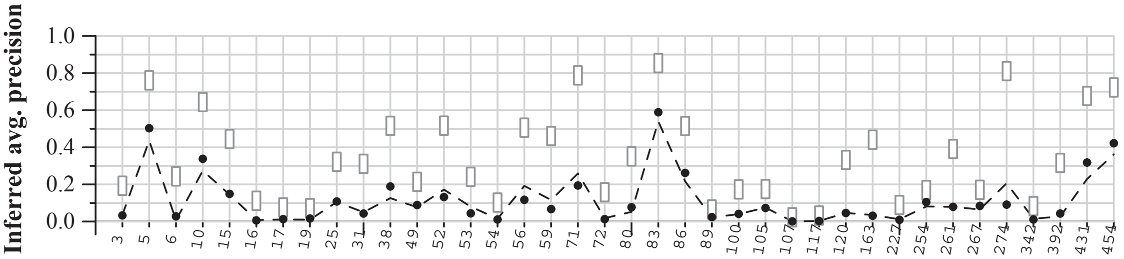

Figure 7. Run scores (dot) versus median (—) versus best (box) for *M_A_FIU-UM-4_4*

Table 2. The MAP values at first n shots for all four runs

Framework	10	100	1000	2000
M_A_FIU-UM-1_1	47.1%	36.4%	16.6%	11.9%
M_A_FIU-UM-2_2	47.6%	34.4%	15.9%	11.4%
M_A_FIU-UM-3_3	46.1%	35.0%	17.8%	12.6%
M_A_FIU-UM-4_4	47.4%	36.5%	19.5%	14.0%

Table 3. Inferred true shots and mean xinfAP

Framework	Inferred true shots	Mean xinfAP
M_A_FIU-UM-1_1	9046	0.096
M_A_FIU-UM-2_2	8675	0.088
M_A_FIU-UM-3_3	9592	0.103
M_A_FIU-UM-4_4	10632	0.116

4 Conclusion and Future Work

In this notebook paper, the framework and results of team FIU-UM in TRECVID 2013 SIN task are summarized. We can tell there are still a lot of improvements need to be done based on the results. Some important directions are desired to be investigated:

- In our framework, only global features are utilized. Object-level and mid-level features need to be explored.
- The proper re-ranking strategy needs to be explored in depth to further improve the retrieval accuracy.
- The proper filtering strategy needs to be adopted to address the data imbalance issue.

It is also necessary to exchange ideas and thoughts with other groups to come up with novel approaches to further improve performance.

Acknowledgements

The authors would like to thank Ming Ma and Yilin Yan from Department of Electrical and Computer Engineering in University of Miami for their help in this work.

References

[1] R. Agrawal and R. Srikant. Fast algorithms for mining association rules. In *Proceedings of the 20th International Conference on Very Large Data Bases*, pages 487–499, September 1994.

[2] C. Archambeau, M. Valle, A. Assenza, and M. Verleysen. Assessment of probability density estimation methods: Parzen window and finite Gaussian mixtures. In *Proceedings of the 2006 IEEE International Symposium on Circuits and Systems*, pages 3245–3248, May 2006.

[3] C. J. C. Burges. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2(2):121–167, June 1998.

[4] C.-C. Chang and C.-J. Lin. Libsvm: A library for support vector machine. *ACM Transactions on Intelligent Systems and Technology*, 2(3):27:1–27:27, 2011.

[5] S. A. Chatzichristofis and Y. S. Boutalis. Cedd: color and edge directivity descriptor: a compact descriptor for image indexing and retrieval. In *Proceedings of the 6th international conference on Computer vision systems*, ICVS'08, pages 312–322, Berlin, Heidelberg, 2008. Springer-Verlag.

[6] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 886–893. IEEE, 2005.

[7] P. Felzenszwalb, D. McAllester, and D. Ramanan. A discriminatively trained, multiscale, deformable part model. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008.

[8] Y.-G. Jiang, J. Wang, S.-F. Chang, and C.-W. Ngo. Domain adaptive semantic diffusion for large scale context-based video annotation. In *2009 IEEE 12th International Conference on Computer Vision*, pages 1420–1427, 2009.

[9] T. S. Lee. Image representation using 2d gabor wavelets. *IEEE Trans. Pattern Anal. Mach. Intell.*, 18(10):959–971, Oct. 1996.

[10] T. Meng and M.-L. Shyu. Leveraging concept association network for multimedia rare concept mining and retrieval. In *IEEE International Conference on Multimedia and Expo (ICME12)*, pages 860–865, July 2012.

[11] T. Meng and M.-L. Shyu. Model-driven collaboration and information integration for enhancing video semantic concept detection. In *The 13th IEEE International Conference on Information Integration and Reuse (IRI2012)*, pages 144–151, Las Vegas, Nevada, August 2012.

[12] T. Meng and M.-L. Shyu. Concept-concept association information integration and multi-model collaboration for multimedia semantic concept detection. *Information Systems Frontiers*, pages 1–13, 2013.

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

[14] P. Over, G. Awad, M. Michel, J. Fiscus, G. Sanders, W. Kraaij, A. F. Smeaton, and G. Quenot. Trecvid 2013 – an overview of the goals, tasks, data, evaluation mechanisms and metrics. In *Proceedings of TRECVID 2013*. NIST, USA, 2013.

[15] A. F. Smeaton, P. Over, and W. Kraaij. *High-Level Feature Detection from Video in TRECVid: a 5-Year Retrospective of Achievements*. Springer US, first edition, 2009.

[16] S. Sural, G. Qian, and S. Pramanik. Segmentation and histogram generation using the hsv color space for image retrieval. In *in International Conference on Image Processing (ICIP). 2002: p. 589-592. VIIth Digital Image Computing: Techniques and Applications, Sun C., Talbot H., Ourselin*, pages 589–592, 2002.

[17] D. Verma and V. Maru. An efficient approach for color image retrieval using haar wavelet. In *Methods and Models in Computer Science, 2009. ICM2CS 2009. Proceeding of International Conference on*, pages 1–5. IEEE, 2009.

[18] E. Yilmaz, E. Kanoulas, and J. A. Aslam. A simple and efficient sampling method for estimating ap and ndcg. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '08, pages 603–610, New York, NY, USA, 2008. ACM.