Gaussian Mixture Model-based Subspace Modeling for Semantic Concept Retrieval

Chao Chen, Mei-Ling Shyu
Department of Electrical and Computer Engineering
University of Miami
Coral Gables, FL 33146, USA
Email: c.chen15@umiami.edu, shyu@miami.edu

Shu-Ching Chen
School of Computing and Information Sciences
Florida International University
Miami, FL 33199, USA
Email: chens@cs.fiu.edu

***Abstract*—Data mining and machine learning methods have been playing an important role in searching and retrieving multimedia information from all kinds of multimedia repositories. Although some of these methods have been proven to be useful, it is still an interesting and active research area to effectively and efficiently retrieve multimedia information under difficult scenarios, i.e., detecting rare events or learning from imbalanced datasets. In this paper, we propose a novel subspace modeling framework that is able to effectively retrieve semantic concepts from highly imbalanced datasets. The proposed framework builds positive subspace models on a set of positive training sets, each of which is generated by a Gaussian Mixture Model (GMM) that partitions the data instances of a target concept (i.e., the original positive set of the target concept) into several subsets and later merges each subset with the original positive data instances. Afterwards, a joint-scoring method is proposed to fuse the final ranking scores from all such positive subspace models and the negative subspace model. Experimental results evaluated on a public-available benchmark dataset show that the proposed subspace modeling framework is able to outperform peer methods commonly used for semantic concept retrieval.**

***Keywords*-Subspace Modeling; Gaussian Mixture Model (GMM); Semantic Concept Retrieval**

I. Introduction

The digital era brings us an explosive amount of data in diverse forms. Nowadays, with the ubiquitous Internet and the prosperity of social media, people upload numerous images and videos to their personal online repositories to share with their families and friends frequently. The content-based retrieval methods [1][2][3][4][5][6][7][8] have achieved great success for various applications in the past decades though they still suffer from the so-called semantic gap issue [8][9]. Researchers have proposed a lot of multimedia content-based retrieval approaches to bridge the semantic gaps and to enhance the retrieval performance in multimedia research [10][11][12][13][14][15][16][17][18][19][20].

Generally speaking, those methods fall into three categories. The first category extracts low-level features (like color, shape, texture, etc.) from images or video frames at pixel, region, and/or object levels and utilizes the keyword-based representation to map the low-level features to keywords or visual words as intermediate features [4][21][22]. Usually, these keywords hold semantic meanings and have a better descriptive ability than those low-level features, and the semantic gaps between high-level concepts and the intermediate features are much smaller.

The second category is called relevance feedback [23][24][25][26]. This category usually involves an interaction between the users and the learning models. For a query issued by a user, the initial results returned by the learning models are sent back to the users to provide their relevance to the query. The learning models can be refined based on the user's feedback to improve the accuracy of the returned results. Usually, such an interactive process needs to undergo several rounds until the results are satisfactory.

Data mining and machine learning-based methods fall into the third category [27][28]. By definition, a positive data instance denotes a data item (a feature vector extracted from an image, a video frame, or a video shot) that contains the target semantic concept and a negative data instance denotes the data item that does not contain the target semantic concept. Semantic concept detection models are built using data mining and/or machine learning algorithms to establish the mapping between the low-level features and the high-level semantic concepts. Although the early success of adopting data mining and machine learning-based methods has greatly encouraged researchers to explore deeper into this area, there are still many scenarios in which these methods find their difficulties in rendering satisfactory retrieval performance. It is not uncommon to see that some semantic concepts are too difficult to be retrieved accurately. Sometimes, it is because the instances of these semantic concepts are so rare in the training set, which makes it (almost) impossible to build a sound model. Under other circumstances, the imbalanced data characteristics in the training set force the trained model to favor negative data instances. In real-world scenarios, we usually encounter imbalance data sets where the positive-to-negative (P2N) ratios are very small, even close to zero. In those scenarios, a model built on the negative training set usually dominates the one trained by the positive training set. Such a data imbalance issue makes it very difficult and challenging for the data mining and machine learning-based methods to retrieve semantic concepts on the imbalanced datasets.

In this paper, we propose a new subspace modeling method, called Gaussian mixture model-based subspace modeling (GMM-based subspace modeling), to attack the problem of retrieving the highly imbalanced semantic concepts. The proposed method employs the Gaussian mixture model (GMM) to generate a number of Gaussian components from the positive training data instances and subsequently assigns every positive data instance to one Gaussian component. The learning models are built on the combination of each Gaussian component and the whole positive training data instances. The idea of utilizing the Gaussian component to partition the data is motivated by realizing the fact that the core idea of subspace modeling is based on the assumption that the underlying data instances loosely satisfy the Gaussian distribution. Therefore, each trained model favors a certain Gaussian component. By ranking an instance using these trained models, we expect to build robust models that are able to capture diverse data characteristics within the subsets of the whole training data.

The paper is organized as follows. The related work is discussed in Section II. Section III elaborates the overall framework and the details of our proposed method. Experimental setup and results are presented in Section IV. Section V concludes the whole paper and explores the future directions.

II. Related Work

Generally speaking, there are a number of ways to address the data imbalance issue in semantic concept retrieval [29]. It is very straightforward to use resampling techniques on the data to manipulate the aforementioned P2N Ratio. The commonly-seen resampling methods include undersampling and oversampling. In an imbalanced dataset where the negative data instances dominate the positive data instances (meaning very small P2N ratio), the oversampling method can generate extra positive data instances by either simply replicating some positive data instances or using synthetic methods such as SMOTE [30] to increase the P2N ratio. The undersampling method increases the P2N ratio in a different way by sampling a portion of the negative data instances from the whole negative training data instances, while keeping all the positive data instances in the dataset. Both methods are common in term of data manipulation to change the P2N ratio of the training data.

Another way to address the data imbalance problem is called cost-sensitive learning [31][32]. Usually, machine learning algorithms are adopted to build learning models, but a cost matrix is introduced to add different penalties to the misclassification of a positive or a negative data instance. Normally, the cost to misclassify a positive data instance is much larger than that of misclassifying a negative data instance. Thus, the penalty values in the cost matrix are therefore larger for the positive data instances than the negative ones.

Boosting can also help deal with the data imbalance problem [33]. Instead of manipulating the P2N ratio or adding different costs inside of the trained learning models, the boosting method is designed to improve the weak learning models. Usually, the learning models trained on the imbalanced dataset are far from satisfactory. By adopting the boosting method at the cost of additional training time, the learning models are expected to render better retrieval accuracy performance.

In addition, kernel-based learning methods are also very popular when learning from imbalanced datasets [34][35]. Kernel-based learning can build more robust learning models from the training set [36]. The idea behind the kernel-based method is that the positive data instances and the negative ones might not be separable in the original feature space. However, they may be separable within a kernel space if such a space is large enough, according to Mercer's theorem [37]. The kernel-based learning method can also be integrated with all aforementioned methods to further improve the retrieval accuracy on imbalanced datasets.

Subspace modeling methods attack the data imbalanced problem in a different way. In subspace modeling, the positive and the negative learning models are trained separately. Each model has its own principal component subspaces. The chi-square distance is used to measure the dissimilarity of a data instance towards the positive or the negative learning models. With regard to ranking, where the learning models are trained using either positive training data instances or the negative training data instances, the data imbalance problem has little impacts on building the subspace learning models.

Subspace modeling methods have shown their effectiveness in semantic concept detection and retrieval, where the positive learning model is built using the whole positive training set [38][39][40][41][42][43][44]. However, the useful patterns in certain subsets of the positive data instances may be shadowed by the dominant patterns reflected by the training set as a whole. This motivates us to improve subspace modeling by constructing a number of positive learning models, each of which is built on the combination of the whole positive training set and a Gaussian-distributed subset derived from it, to improve the retrieval accuracy. In this way, each combined dataset has a portion of the oversampled training data that satisfy some Gaussian distribution. Another way to look at the combined positive data is that a portion of the whole positive data have more weights than the rest of the data that are not duplicated. From either angle, it is beneficial for the subspace modeling method to build the models that favor these oversampled positive data. There are many other methods, such as K-means, that can be used in place of GMM. The reason why GMM is chosen is because it is backed up by the well-established probability theory and it is also a commonly-used generalized solution to deal with real-world datasets.

III. Framework

Figure 1 shows the proposed Gaussian mixture model-based subspace modeling framework. During the training phase, the positive data instances are decomposed into several subsets using Gaussian mixture models (GMMs). Each GMM has its own mean value and standard deviation. Afterwards, the input to each subspace model includes the original positive training set and the positive data instances of each GMM component. Under such circumstances, the input to one subspace model not only covers all positive training data instances but also has the dominant patterns belonging to the selected GMM component strengthened as well, as the data of the GMM component

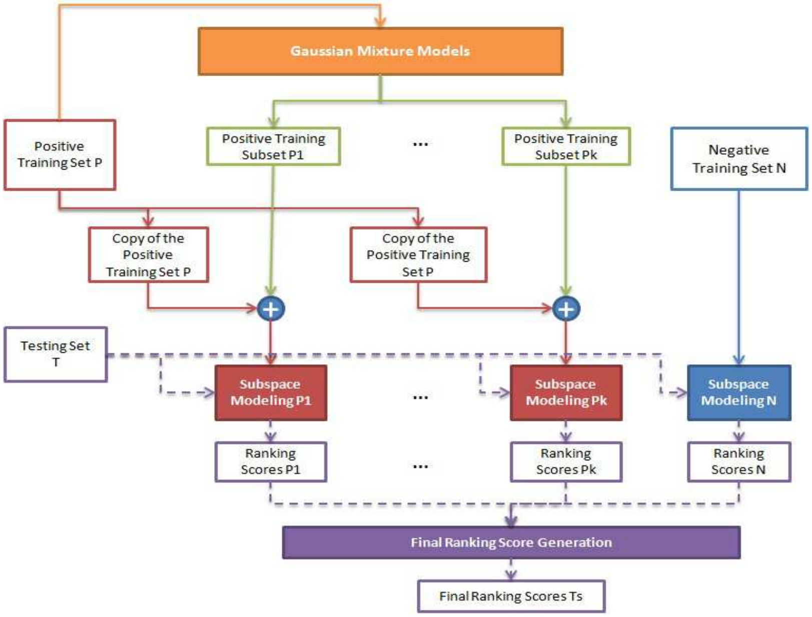

Fig. 1. GMM-based subspace modeling framework

is duplicated and such a duplication makes the center of the learning model move towards the mean value of the selected GMM component. The subspace modeling method, which is based on the assumption that the underlying data generally satisfies the normal distribution, trains a subspace model in the principal component subspace using the assembled input data. For a testing data instance, each subspace model is able to generate a ranking score using the chi-square distance calculated from the subspace model (to be shown in Section III-B). Finally, the ranking scores from all subspace models are consolidated to the final ranking score for the testing data instance. The next subsections attempt to answer the following questions.

- How to dynamically generate the components of the Gaussian mixture models? (To be shown in Section III-A)
- How does subspace modeling generate the ranking scores for a data instance? (To be answered in Section III-B)
- How to consolidate the scores from each model to a final score? (To be answered in Section III-C)

A. Dataset decomposition using Gaussian Mixture Models

For a data instance $x \in \mathbb{R}^d$, Equation (1) shows the density of the Gaussian mixture model (GMM) formed by M Gaussian components. Each component satisfies a Gaussian distribution with the density shown in Equation (2), where μ_i denotes the mean and Σ_i is the covariance. The three parameters (w_i, μ_i, Σ_i) in a GMM can be estimated by the Expectation-Maximization (EM) algorithms [45]. The EM algorithm requires a number of iterations, each of which contains an expectation step and a maximization step. The algorithm starts with a random initial estimation of these parameters and keeps updating the values of these parameters until it converges.

$$p(x|w_i,\mu_i,\Sigma_i) = \sum_{i=1}^{M} w_i \cdot G(x|\mu_i,\Sigma_i), \text{ s.t. } \sum_{i=1}^{M} w_i = 1; \qquad (1)$$

$$G(x|\mu_i,\Sigma_i) = \frac{1}{(2\pi)^{d/2}|\Sigma_i|^{1/2}} e^{-\frac{1}{2}(x-\mu_i)'\Sigma_i^{-1}(x-\mu_i)}. \qquad (2)$$

The GMM does not provide the estimation of the number of components within the data. In fact, the number of Gaussian components (i.e., M) needs to be provided by the users. Although there are some methods that can be used to determine the number of Gaussian components, such as Bayesian information criterion [46] and Akaike information criterion [47], they usually require extra time and space. For simplicity, we simply use the maximum number of Gaussian components allowed by the data as M. In the actual implementation, the Gaussian components cannot be derived if the input matrix is singular. Therefore, we perform principal component analysis on positive training data instances first and keep only those principal components whose eigenvalues are greater than 0.001. After dynamically decomposing the data into M components based on the training data, we assign each training data instance to one Gaussian component based on the maximum probability criterion. In an extreme case where the whole positive training set cannot be decomposed into two or more components, the GMM-based subspace modeling is simply the regular subspace modeling, which is to be elaborated in Section III-B.

It is worth mentioning that the merits of decomposing the data into different Gaussian components lie in two folds. First, the data within each component tend to be similar and the standard deviation in each component is much smaller. Second, the dominant patterns are presented better inside each Gaussian component.

B. Subspace modeling

The subspace modeling methods have been successfully applied in semantic concept detection and retrieval [38][39][40][41][42][43]. The idea of subspace modeling is to derive principal component subspaces separately for the positive and the negative training data instances. Based on the projection on a principal component subspace, the chi-square distance is used to measure the dissimilarity of each data instance towards a subspace model. By comparing the distance towards the positive and the negative learning models, the final ranking scores can be generated. Usually, the whole process requires performing three major steps: normalization, principal component space projection, and ranking score generation. Since the positive and the negative learning models are generated in the same way, we just take the positive learning model as an example here. Similarly, the same process can be applied to the negative learning model.

The normalization step is shown in Equation (3), where μ and σ are the sample mean and standard deviation of the positive training set X. The parameter set $\{\lambda, PC\}$ are derived from the covariance matrix $CovX$ of the normalized positive data instances X_{norm} using singular value decomposition (SVD) (see Equation (4)), λ is the diagonal values in Σ that are greater than a threshold (i.e., 0.001 in our experiment), and PC are the principal components that correspond to those retained eigenvalues in V.

$$\begin{aligned} X_{norm} &= \frac{X-\mu}{\sigma}; & (3)\\ CovX &= U\Sigma V^*; & (4)\\ Y_i &= X_{norm}\cdot PC_i,\ \ i\in[1,numP]; & (5)\\ \chi_p &= \frac{1}{numP}\sqrt{\sum_i \frac{Y_i^p\cdot Y_i^p}{\lambda_i^p}},\ \ i\in[1,numP]. & (6)\end{aligned}$$

The projection of the normalized positive data instances on its principal component subspace satisfies the Gaussian distribution, where PC_i is the i-th PC, Y_i is the projection of X_{norm} on PC_i, and $numP$ is the number of principal components derived from the positive training data instances (shown in Equation (5)). Such a projection is later used to calculate the chi-square distance (shown in Equation (6)) to measure the dissimilarity between a data instance and the positive learning model, which also serves as the ranking scores generated by the learning model for X.

C. Generation of final ranking scores

The Gaussian mixture model may generate a number of Gaussian components on all positive training data instances, each of which is corresponding to a positive learning model. Assume there is a testing data instance Ts which has a ranking score vector RS^p generated by all positive learning models, represented by $RS^p = \{RS_1^p, \ldots, RS_M^p\}$, where M is the number of Gaussian components dynamically derived from Section III-A. Likewise, Ts has a ranking score RS^n generated from the negative learning model, as indicated in Figure 1. The final ranking score RS_{final} of Ts is calculated using Equation (7), which considers the ranking scores from all positive models and the negative model. A large RS^n or a small μ^p indicates that Ts is more likely to be a positive data instance than a negative one. μ^p shows how dissimilar the testing data instance Ts is towards all positive learning models as a whole, which is expected to better depict the relationship between Ts and all positive data instances than using any positive learning model alone.

$$RS_{final} = \frac{RS^n - \mu^p}{RS^n + \mu^p},\ \text{where}\ \mu^p = \frac{1}{M}\sum_{i=1}^{M} RS_i^p. \qquad (7)$$

IV. Experiment

To evaluate the effectiveness of our proposed method on imbalanced datasets, a benchmark dataset is used to compare semantic concept retrieval performance with several other well-known methods including support vector machine, decision tree, etc. The details of the dataset are listed in Section IV-1.

1) Experimental Setup: The dataset used in the experiment is a light version of the NUS-WIDE dataset called NUS-WIDE-LITE [48]. In this dataset, a total of 55,615 images are crawled from the Flickr website. The NUS-WIDE-LITE dataset has predefined the training and testing sets, where 27,807 images are in the training set and another 27,808 images are used as the testing set. Some low-level features of the images (like color histogram, wavelet texture, and etc.) are available for downloading. In the experiment below, we evaluate our proposed method against LibSVM [49], Logistic Regression, and Decision Tree [50] on two feature sets: 64-dimensional color histogram in LAB color space and 128-dimensional wavelet texture. There are a total of 81 concepts in the NUS-WIDE-LITE dataset. The P2N ratio of the training set and testing set for all 81 concepts are drawn in Figure 2 and Figure 3 in a sorted order, respectively. As can be seen in the figures, the mean P2N ratio in the training set is 0.023 and the median value is 0.009. Therefore, it is very difficult and challenging to retrieve semantic concepts in such an imbalanced dataset. Finally, the performance of semantic concept retrieval is evaluated using a commonly used measure called *mean average precision (MAP)*. Let

- C: the number of concepts in the dataset;
- AP_i: the average precision of Concept i (defined in Equation (9));
- $||P_i||$: the number of positive data instances of Concept i;
- K: the number of retrieved data instances;
- r_j: an indicator value, equaling 1 if the retrieved data instance at rank j is positive, zero otherwise (defined in Equation (10)).

Then MAP can be calculated using Equation (8).

$$MAP = \frac{1}{C}\sum_{i=1}^{C} AP_i;\ \text{where} \tag{8}$$

$$AP_i = \frac{1}{\|P_i\|}\sum_{\omega=1}^{K} r_\omega \cdot \frac{1}{\omega}\sum_{j=1}^{\omega} r_j;\ \text{and} \tag{9}$$

$$r_j = \begin{cases} 1, & \text{if the instance } j \text{ is relevant;} \\ 0, & \text{otherwise.} \end{cases} \tag{10}$$

2) Experimental Results and Analyses: Table I shows the experimental results, comparing our proposed GMM-based subspace modeling method with several peer methods such as LibSVM with RBF-kernel (LibSVM), Logistic Regression(LR), and Decision Tree (DTree), in terms of the mean average precision (MAP) on two feature sets. The suggested parameters of these peer methods are used by default, as these parameters usually generate reasonably good results. Table I shows the MAP evaluated on two features and the relative performance gain of our proposed method against peer methods. For example, GMM-based subspace modeling is 9.5% better than logistic regression on the color histogram feature and 13.8% better than LibSVM on the wavelet texture feature set, in terms of relative percentage improvements. Furthermore, by comparing our method with the best peer methods, we found that the proposed method renders the best average precision values on $2/3$ of the concepts in the dataset. For the rest of the concepts, the proposed methods still has room to improve, such as increasing the weight of the samples, if the imbalance is too extreme.

The number of Gaussian components that are dynamically generated for each concept is shown in Figure 4. On average, about 8 Gaussian components are generated per concept and the median value of the generated Gaussian components for each concept is 4. To show the retrieval performance on color histogram and wavelet texture in details, we pick the concept 'whales" and the concept "fish" and draw their ROC curves in Figure 5 and Figure 6, respectively. In Figure 5, the GMM-based subspace modeling method mostly shows better performance than other peer methods on color histogram features with the exception that the LibSVM is better starting from false positive rate more than 0.6. However, it is obvious to see that the area under curve (AUC) of the proposed GMM-based subspace is larger than LibSVM, meaning the overall performance of GMM-based subspace modeling is better. Figure 6 clearly shows our method renders better performance as the AUC of the proposed GMM-based method is much more larger than any other comparative methods. It is worth pointing out is that although the data is so imbalanced that the decision tree model is totally dominated by negative instances (predicting every instance as negative), causing the ROC curve of Decision Tree being a diagonal line. However, our method still shows its effectiveness to retrieve concepts from such an imbalanced dataset.

TABLE I
MAP EVALUATED ON ALL 81 CONCEPTS OF NUS-WIDE-LITE ON COLOR HISTOGRAM (CH64) AND WAVELET TEXTURE (WT128)

	CH64	Relative Gain	WT128	Relative Gain
Ours	4.14%	–	4.44%	–
LibSVM	3.60%	15.0%	3.90%	13.8%
LR	3.78%	9.5%	3.32%	33.7%
DTree	2.80%	47.9%	2.87%	54.7%

V. CONCLUSIONS AND FUTURE WORK

In this paper, we proposed a novel Gaussian mixture model-based subspace modeling method, which built a Gaussian mixture model with a number of dynamically-determined Gaussian components on the positive training set. Then, the positive data instances within each Gaussian component were merged with the whole positive training set as the input to train a positive subspace model. By utilizing GMM to divide the positive training set to several Gaussian-distributed subsets, it was expected that some patterns within the subsets could be revealed. Finally, the final ranking scores were generated by consolidating the score from the negative model with the mean value of the scores from all positive models. Experimental results showed that our proposed method was able to provide better retrieval performance than the other comparative methods in terms of the MAP (mean average precision) measure on a benchmark dataset that was highly imbalanced.

In the future, our research work will explore the following directions. First, the number of replications that should be made on the data belonging to each component will be investigated, when merged with the whole positive training set. In an extreme case in which the data of a component is just a small portion of the whole positive training data instances, the patterns within the Gaussian component would not be obvious enough and therefore cannot be revealed. Second, research work will also be dedicated to the generation of the final ranking scores from the learning models as well, especially on how to consolidate the ranking scores from all the positive models. In this paper, the mean value of the scores from all positive models is adopted, which considers an equal weight from each component. However, this does not take into consideration the distance of a data instance towards each component's center. It would be interesting to generate the consolidated ranking scores by either selecting a few nearest Gaussian components or using a weighted combination based on a data instance's distances toward the centers of the Gaussian components. In addition, kernel PCA can be explored to handle the linearly non-separable data.

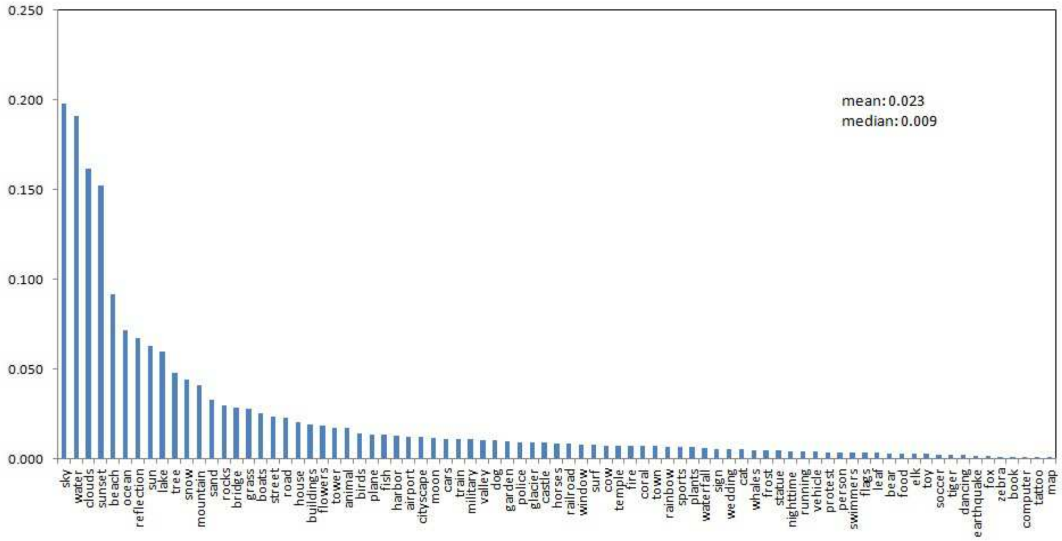

Fig. 2. Positive-to-Negative (P2N) ratios in the training set

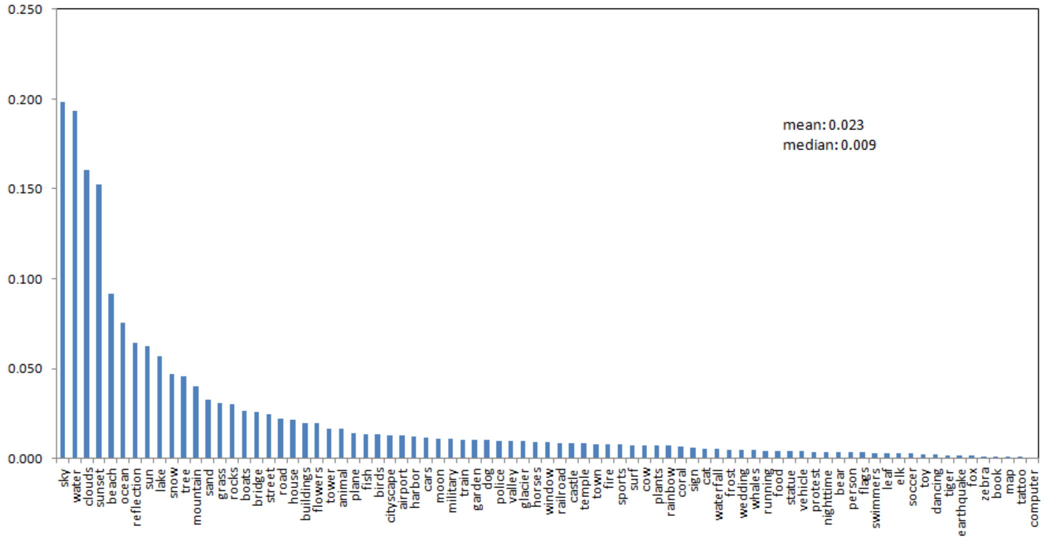

Fig. 3. Positive-to-Negative (P2N) ratios in the testing set

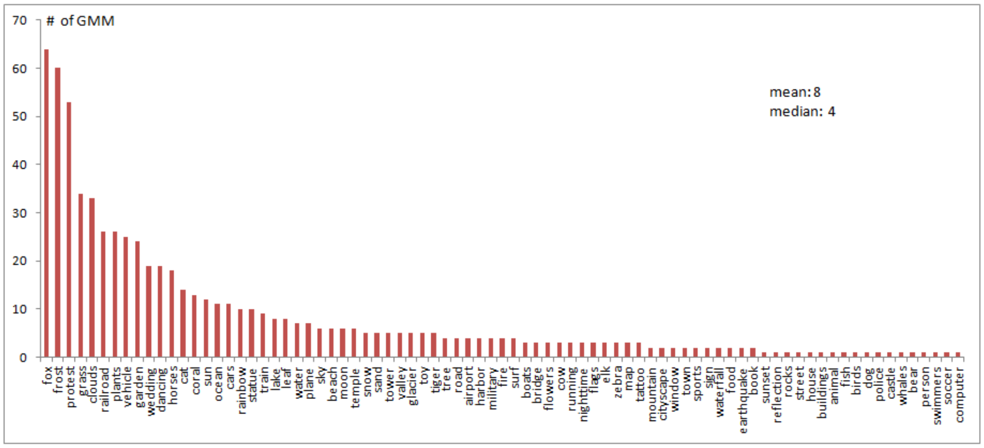

Fig. 4. Numbers of Gaussian components generated for 81 concepts

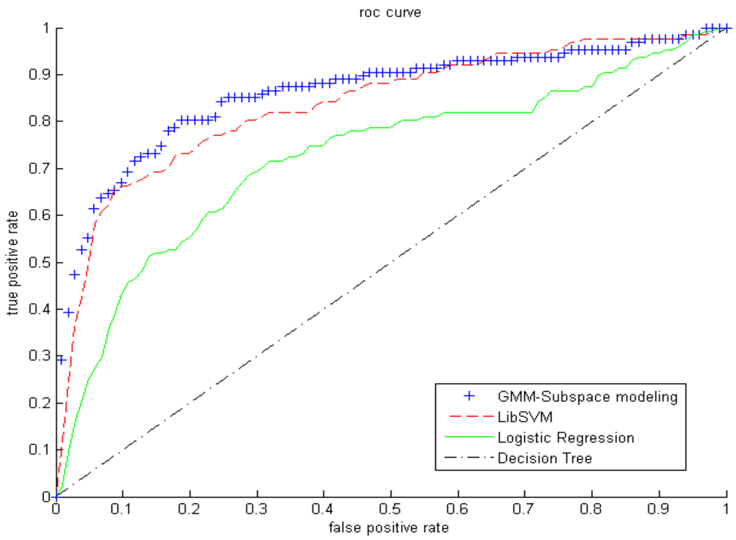

Fig. 5. ROC Curve of Concept “whales” using the color histogram features

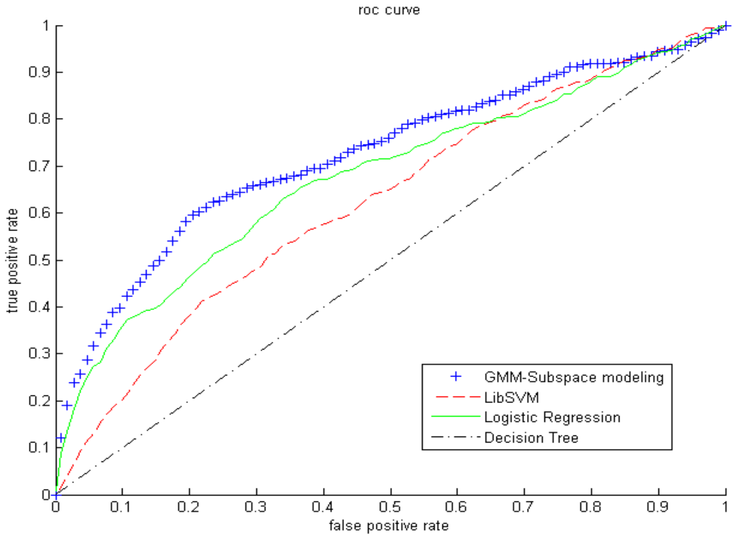

Fig. 6. ROC Curve of Concept “fish” using the wavelet texture features

References

[1] H.-Y. Ha, F. C. Fleites, and S.-C. Chen, “Content-based multimedia retrieval using feature correlation clustering and fusion,” *International Journal of Multimedia Data Engineering and Management (IJMDEM)*, vol. 4, no. 2, pp. 46–64, 2013.

[2] M. Chen, S.-C. Chen, M.-L. Shyu, and K. Wickramaratna, “Semantic event detection via temporal analysis and multimodal data mining,” *IEEE Signal Processing Magazine, Special Issue on Semantic Retrieval of Multimedia*, vol. 23, no. 2, pp. 38–46, October 2006.

[3] S.-C. Chen, S. H. Rubin, M.-L. Shyu, and C. Zhang, “A dynamic user concept pattern learning framework for content-based image retrieval,” *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 36, no. 6, pp. 772–783, 2006.

[4] J. Fan, Y. Gao, H. Luo, and G. Xu, “Automatic image annotation by using concept-sensitive salient objects for image content representation,” in *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR ’04)*, July 2004, pp. 361–368.

[5] M.-L. Shyu, S.-C. Chen, M. Chen, and C. Zhang, “A unified framework for image database clustering and content-based retrieval,” in *ACM International Workshop on Multimedia Databases*, 2004, pp. 19–27.

[6] S.-C. Chen, M.-L. Shyu, C. Zhang, L. Luo, and M. Chen, “Detection of soccer goal shots using joint multimedia features and classification rules,” in *The Fourth ACM International Workshop on Multimedia Data Mining (MDM/KDD2003)*, August 2003, pp. 36–44.

[7] X. Li, S.-C. Chen, M.-L. Shyu, and B. Furht, “An effective content-based visual image retrieval system,” in *IEEE International Conference on Computer Software and Applications Conference, (COMPSAC)*, 2002, pp. 914–919.

[8] A. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain, “Content-based image retrieval at the end of the early years,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 12, pp. 1349–1380, December 2000.

[9] C. Dorai and S. Venkatesh, “Bridging the semantic gap with computa-

tional media aesthetics," *IEEE MultiMedia*, vol. 10, no. 2, pp. 15–17, April 2003.

[10] Y. Chen, H. Sampathkumar, B. Luo, and X.-W. Chen, "ilike: Bridging the semantic gap in vertical image search by integrating text and visual features," *IEEE Transactions on Knowledge and Data Engineering*, vol. 25, no. 10, pp. 2257–2270, October 2013.

[11] X. Hu, K. Li, J. Han, X. Hua, L. Guo, and T. Liu, "Bridging the semantic gap via functional brain imaging," *IEEE Transactions on Multimedia*, vol. 14, no. 2, pp. 314–325, April 2012.

[12] L. Lin and M.-L. Shyu, "Weighted association rule mining for video semantic detection," *International Journal of Multimedia Data Engineering and Management*, vol. 1, no. 1, pp. 37–54, January-March 2010.

[13] ——, "Effective and efficient video high-level semantic retrieval using associations and correlations," *International Journal of Semantic Computing*, vol. 3, no. 4, pp. 421–444, 2009.

[14] L. Lin, M.-L. Shyu, G. Ravitz, and S.-C. Chen, "Video semantic concept detection via associative classification," in *IEEE International Conference on Multimedia and Expo (ICME)*, 2009, pp. 418–421.

[15] L. Lin, G. Ravitz, M.-L. Shyu, and S.-C. Chen, "Correlation-based video semantic concept detection using multiple correspondence analysis," in *IEEE International Symposium on Multimedia (ISM08)*, December 2008, pp. 316–321.

[16] A. Hauptmann, R. Yan, W.-H. Lin, M. Christel, and H. Wactlar, "Can high-level concepts fill the semantic gap in video retrieval? a case study with broadcast news," *IEEE Transactions on Multimedia*, vol. 9, no. 5, pp. 958–966, August 2007.

[17] S.-C. Chen, M.-L. Shyu, C. Zhang, and M. Chen, "A multimodal data mining framework for soccer goal detection based on decision tree logic," *International Journal of Computer Applications in Technology, Special Issue on Data Mining Applications*, vol. 27, no. 4, pp. 312–323, 2006.

[18] S.-C. Chen, M.-L. Shyu, M. Chen, and C. Zhang, "A decision tree-based multimodal data mining framework for soccer goal detection," in *IEEE International Conference on Multimedia and Expo (ICME 2004)*, June 2004, pp. 265–268.

[19] R. Zhao and W. I. Grosky, "Narrowing the semantic gap - improved text-based web document retrieval using visual features," *IEEE Transactions on Multimedia*, vol. 4, no. 2, pp. 189–200, June 2002.

[20] S.-C. Chen, R. L. Kashyap, and A. Ghafoor, *Semantic models for multimedia database searching and browsing*. Springer Science & Business Media, 2000, vol. 21.

[21] A. Kutics, A. Nakagawa, K. Tanaka, M. Yamada, Y. Sanbe, and S. Ohtsuka, "Linking images and keywords for semantics-based image retrieval," in *Proceedings. 2003 International Conference on Multimedia and Expo (ICME '03)*, July 2003, pp. 777–780.

[22] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[23] R. Hong, M. Wang, Y. Gao, D. Tao, X. Li, and X. Wu, "Image annotation by multiple-instance learning with discriminative feature mapping and selection," *IEEE Transactions on Cybernetics*, vol. 44, no. 5, pp. 669–680, 2014.

[24] M.-L. Shyu, S.-C. Chen, M. Chen, C. Zhang, and C.-M. Shu, "Probabilistic semantic network-based image retrieval using mmm and relevance feedback," *Multimedia Tools and Applications*, vol. 30, no. 2, pp. 131–147, August 2006.

[25] S. Hoi, M. Lyu, and R. Jin, "A unified log-based relevance feedback scheme for image retrieval," *IEEE Transactions on Knowl. and Data Engineering*, vol. 18, no. 4, pp. 509–524, April 2006.

[26] C. Zhang, S.-C. Chen, and M.-L. Shyu, "Multiple object retrieval for image databases using multiple instance learning and relevance feedback," in *IEEE International Conference on Multimedia and Expo (ICME)*, 2004, pp. 775–778.

[27] M.-L. Shyu, C. Haruechaiyasak, and S.-C. Chen, "Category cluster discovery from distributed www directories," *Journal of Information Sciences*, vol. 155, no. 3-4, pp. 181–197, 2003.

[28] M.-L. Shyu, S.-C. Chen, M. Chen, C. Zhang, and K. Sarinnapakorn, "A unified framework for image database clustering and content-based retrieval," in *ACM International Workshop on Multimedia Databases*, 2003, pp. 78–85.

[29] H. He and E. Garcia, "Learning from imbalanced data," *IEEE Transactions on Knowledge and Data Engineering*, vol. 21, no. 9, pp. 1263–1284, Sept 2009.

[30] H. Han, W. Y. Wang, and B. H. Mao, "Borderline-smote: A new over-sampling method in imbalanced data sets learning," in *International Conference on Intelligent Computing(ICIC 2005)*, August 2005, pp. 878–887.

[31] H.-Y. Lo, S.-D. Lin, and H.-M. Wang, "Generalized k-labelsets ensemble for multi-label and cost-sensitive classification," *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, no. 7, pp. 1679–1691, July 2014.

[32] J. Wang, P. Zhao, and S. Hoi, "Cost-sensitive online classification," *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, no. 10, pp. 2425–2438, October 2014.

[33] M. Galar, A. Fernandez, E. Barrenechea, H. Bustince, and F. Herrera, "A review on ensembles for the class imbalance problem: Bagging-, boosting-, and hybrid-based approaches," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 42, no. 4, pp. 463–484, July 2012.

[34] X. Hong, S. Chen, and C. Harris, "A kernel-based two-class classifier for imbalanced data sets," *IEEE Transactions on Neural Networks*, vol. 18, no. 1, pp. 28–41, January 2007.

[35] G. Wu and E. Chang, "Kba: kernel boundary alignment considering imbalanced data distribution," *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, no. 6, June 2005.

[36] N. Japkowicz and S. Stephen, "The class imbalance problem: A systematic study," *Intelligent Data Analysis*, vol. 6, no. 5, pp. 429–449, October 2002.

[37] J. Mercer, "Functions of positive and negative type, and their connection with the theory of integral equations," *Philosophical Transactions of the Royal Society*, vol. 209, no. 441-458, pp. 415–446, 1909.

[38] M.-L. Shyu, C. Chen, and S.-C. Chen, "Multi-class classification via subspace modeling," *International Journal of Semantic Computing*, vol. 5, no. 1, pp. 55–78, 2011.

[39] L. Lin, C. Chen, M.-L. Shyu, and S.-C. Chen, "Weighted subspace filtering and ranking algorithms for video concept retrieval," *IEEE Multimedia*, vol. 18, no. 3, pp. 32–43, 2011.

[40] C. Chen and M.-L. Shyu, "Clustering-based binary-class classification for imbalanced data sets," in *The 12th IEEE International Conference on Information Reuse and Integration (IRI 2011)*, August 2011, pp. 384–389.

[41] C. Chen, M.-L. Shyu, and S.-C. Chen, "Data management support via spectrum perturbation-based subspace classification in collaborative environments," in *The 7th International Conference on Collaborative Computing: Networking, Applications and Worksharing*, 2011, pp. 67–76.

[42] C. Chen, T. Meng, and L. Lin, "A web-based multimedia retrieval system with MCA-based filtering and subspace-based learning algorithms," *International Journal of Multimedia Data Engineering and Management*, vol. 4, no. 2, pp. 13–45, 2013.

[43] M.-L. Shyu, Z. Xie, M. Chen, and S.-C. Chen, "Video semantic event/concept detection using a subspace-based multimedia data mining framework," *IEEE Transactions on Multimedia, Special number on Multimedia Data Mining*, vol. 10, no. 2, pp. 252–259, Feb. 2008.

[44] M.-L. Shyu, T. Quirino, Z. Xie, S.-C. Chen, and L. Chang, "Network intrusion detection through adaptive sub-eigenspace modeling in multiagent systems," *ACM Transactions on Autonomous and Adaptive Systems*, vol. 2, no. 3, pp. 9:1–9:37, 2007.

[45] A. Dempster, N. Laird, and D. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of the Royal Statistical Society*, vol. 39, no. 1, p. 138, 1977.

[46] G. E. Schwarz, "Estimating the dimension of a model," *Annals of Statistics*, vol. 6, no. 2, p. 461464, 1978.

[47] H. Akaike, "Information theory and an extension of the maximum likelihood principle," in *2nd International Symposium on Information Theory*, September 1973, pp. 267–281.

[48] T.-S. Chua, J. Tang, R. Hong, H. Li, Z. Luo, and Y.-T. Zheng, "Nus-wide: A real-world web image database from national university of singapore," in *ACM International Conference on Image and Video Retrieval*, July 2009, pp. 48:1–48:9.

[49] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, pp. 27:1–27:27, 2011.

[50] J. R. Quinlan, *C4.5: Programs for Machine Learning*. Morgan Kaufmann, 1993.