A Multimedia Semantic Retrieval Mobile System Based on HCFGs

Yimin Yang, Hsin-Yu Ha, Fausto C. Fleites, and Shu-Ching Chen
Florida International University

A multimedia semantic retrieval system based on hidden coherent feature groups (HCFGs) can extract the semantics from multimedia data and perform effective retrieval.

The proliferation of Internet-connected mobile devices such as mobile phones and tablet computers has made it common practice for people to upload all kinds of multimedia data to social sites such as Flickr, YouTube, and Facebook. By 2014, the rate of data sharing via mobile devices will be 14 times greater than in 2008.[1] The multimedia research community has addressed this challenge by developing systems that allow the semantic retrieval of multimedia data. Nevertheless, research on this problem remains active given the semantic gap between the low-level representation of multimedia data and their high-level semantic meaning.

A typical concept retrieval framework is built on the tasks of feature extraction, model training, classification, and ranking. Although much research has been done on each of these tasks,[2,3] significant challenges still remain such as the effective analysis and utilization of multisource, high-dimensional features. To effectively retrieve meaningful semantics from rapidly growing multimedia data, it is essential to capture the correlations among features to enhance the effectiveness of model training and classification tasks.

To tackle this problem, researchers usually perform either a linear combination of the original features from different modalities or use statistical techniques such as principle component analysis (PCA) and independent component analysis (ICA) to transform the original features into another space and select the most "important" features. However, these statistical methods try to make each feature independent in the transformed space, and as a result, some information is lost during model training on the transformed feature set. Overall, these methods do not thoroughly explore the correlation between features with different types and may not fully utilize the complementary information from various features. For instance, the tag "tree" implies the color "green" for the semantic concept "forest," which is considered a hidden correlation between features.

In addition to the feature-analysis problem, another issue is the integration of multiple models in the semantic space by fusing the decisions (scores) from different models. The challenges lie in how to select the training models for different feature types and how to evaluate the confidence of the decision from different models and take that into account when performing final fusion. (See the "Related Work in Multimedia Semantic Retrieval Systems" sidebar for more details on previous research.)

Given these problems and challenges, we propose a correlation-based feature-analysis method to explore hidden coherent feature groups (HCFGs) and present a novel, multimodel fusion scheme. Specifically, we analyze the correlation between each feature pair and use the affinity propagation algorithm to separate the original feature set into different feature groups (HCFGs), maximizing the intragroup correlation and minimizing intergroup correlation. Subsequently, one model is trained for each of the HCFGs, and the HCFGs with best performance in the training phase are chosen for the final score fusion. This article presents a mobile system that utilizes the proposed framework as its retrieval engine and features a user feedback mechanism for improving retrieval performance.

Proposed Framework

Figure 1 depicts the proposed multimedia semantic retrieval mobile system. The system

1070-986X/14/$31.00 © 2014 IEEE

Published by the IEEE Computer Society

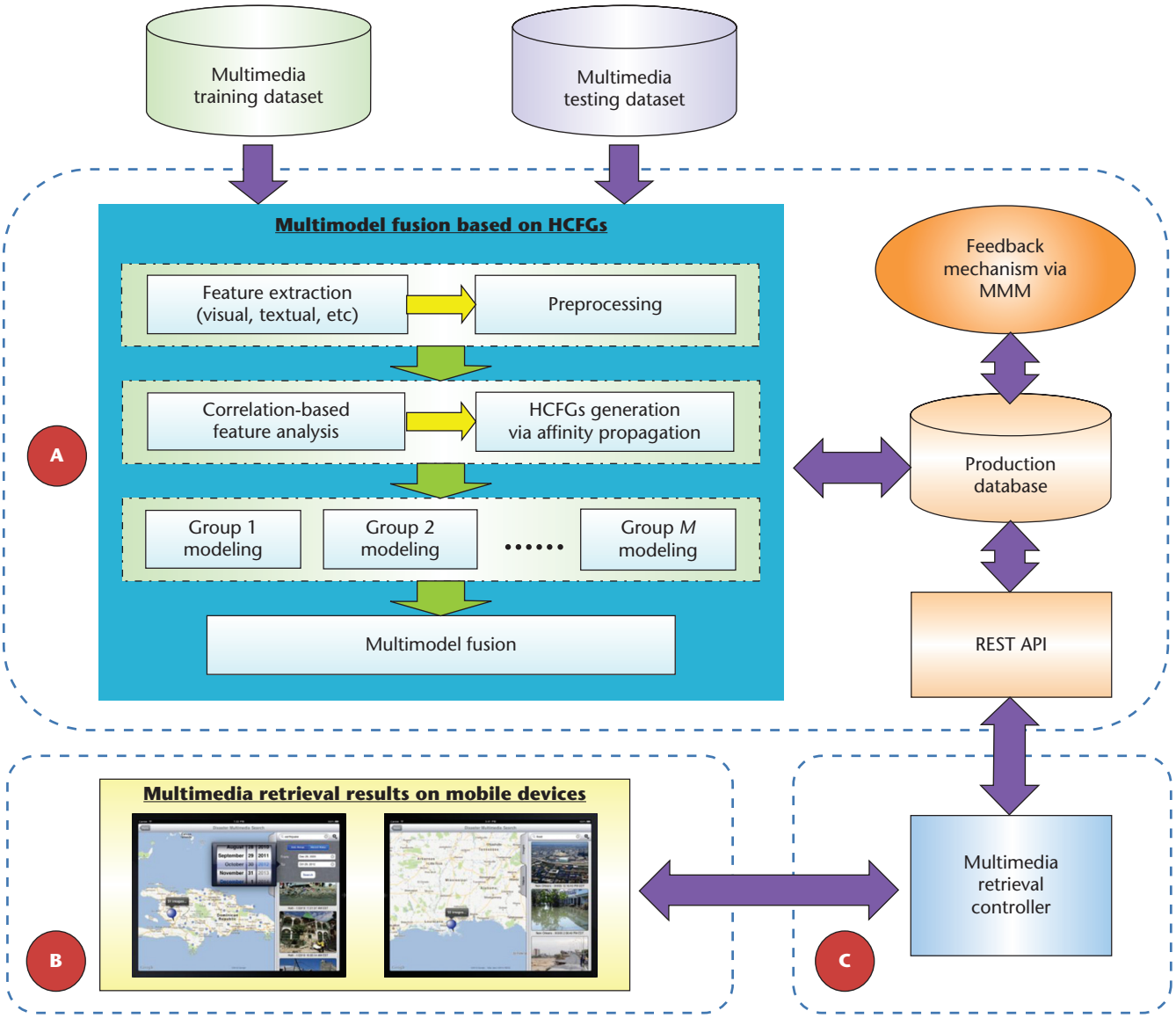


Figure 1. Multimedia semantic retrieval mobile system based on hidden coherent feature groups (HCFGs). Parts A, B, and C correspond to the main system logic, system output, and multimedia retrieval controller, respectively.

design follows a model-view-controller (MVC) pattern. The model part (labeled A) implements the main logic of the system; it is a retrieval model built from the fusion of multiple classification models, which are based on hidden feature groups. The retrieval model consists of training and testing phases. During the training phase, the metamodel is trained based on training data with ground-truth information, and unknown multimedia data are classified using the learned model during the testing phase. All the processed data and the trained models are stored in the production database. The multimedia retrieval controller part (labeled C) translates user input into operations on the model and controls the data transfer between the front-end user interface and the back-end server through a REST API. Finally, the view part (labeled B) generates and presents output to the users.

The proposed system builds the retrieval model following a five-step process:

1. feature extraction,
2. preprocessing,
3. correlation-based feature analysis and clustering,
4. model training, and
5. model fusion.

Related Work in Multimedia Semantic Retrieval Systems

The related work in multimedia semantic retrieval systems can be generally categorized based on the following two perspectives: the back-end algorithm and system performance. For the first category, Michael Lew and his colleagues provided a thorough overview of the state of the art in multimedia semantic retrieval and identify several prevalent research topics that have potential for improving multimedia retrieval by bridging the well-known semantic gap.[1] These topics include human-centered computing, learning and semantics, new features and similarity measures, new media, browsing and summarization, indexing, and evaluation and benchmarking. In this article, we mainly focus on extracting the semantics from multimedia data by exploring the correlations in feature space and present a multimodel fusion scheme for effective retrieval. There are two subtopics involved in our work: feature space analysis and multimodel fusion.

To effectively retrieve semantic concepts from multimedia data, much research has been done to project the original feature space to a low-dimensional space using linear or nonlinear mapping methods[2] and further derive the Euclidean distance for each instance pair to represent the pairwise similarity. For example, Jing Huang and his colleagues proposed an image retrieval system using only the Euclidean distance of image color features to calculate the ranking score for each image per specific concept.[3] In another work,[4] Paris Smaragdis and Michael Casey proposed employing the subspace projection on all the features by using PCA and ICA to determine the maximally independent subspaces. Other works use statistical techniques to capture multimedia correlation at the feature level. Ara Nefian and his colleagues adopted an early fusion approach to combine audio and visual features for speech recognition by using the coupled hidden Markov model (CHMM) and dynamic Bayesian networks.[5] Recently, canonical correlation analysis (CCA), another powerful statistical technique, has found its application in linear mapping that maximizes the cross-correlation between two feature sets.[6] However, besides the correlation among multimedia data instances, the complementary and mutual information among features from multiple modalities should also be extensively exploited to determine how to integrate them to improve the performance and avoid possible information loss during the transformation between different feature spaces.

Other than correlation captured at the feature level, the correlation among different models and model confidence toward extracting semantic concepts should also be learned. In one work,[7] separate generative probabilistic models were learned for different classifiers. The scores were combined afterward to yield a final detection score. Chao Chen and his colleagues proposed a fusion strategy to combine ranking scores from both tag- and content-based models that considers the adjustment, reliability, and correlation of ranking scores from different models.[8] To leverage the correlation from the feature and model levels, Azzedine Bendjebbour and his colleagues performed fusion at both levels.[9] At the feature level, the mass of a given pixel based on two sensors is computed and fused, while at the decision level, the HMM outputs are combined. In another work,[10] CCA is used to fuse audio-visual features with joint subspace learning at different granularity, and the final decision is made based on the Bayesian decision fusion of

In the first two steps, the system extracts visual features from the training data and performs preprocessing to normalize the features and remove those with relatively low variance. Second, in the correlation-based feature analysis and clustering step, the system computes a feature similarity matrix based on correlation coefficients for all pairs of retained features and applies the Affinity Propagation (AP) algorithm to cluster the feature set to obtain multiple HCFGs that exhibit low intergroup correlation and high intragroup correlation. Subsequently, the model-training step builds a classification model for each discovered feature group. Finally, the model fusion step combines the individual models using the proposed multimodel fusion strategy. Such a partition of the feature set into HCFGs aims at "untapping" hidden feature groups that will enhance the fused model's predictive power.

When a query is issued, the system performs feature extraction and preprocessing and groups the features into the same HCFGs identified in the training phase. The HCFGs are then fed to the trained models obtained during the model-training step. The generated testing scores are fused and ranked afterward. The ranked results are shown in the mobile application. In addition, the system contains a user-feedback component that incorporates user interactions in the retrieval process to refine the retrieval results.

Visual Feature Extraction

The feature set utilized in the proposed system consists of histogram of oriented gradients

multiple HMM-based classifiers. Although researchers have made many attempts to utilize two kinds of correlation among multimedia data, the performance is far from satisfactory.

For the second category (the system performance point of view), most of the mobile multimedia retrieval systems mainly focus on improving performance in terms of transmission time. Researchers proposed first compressing low-level feature descriptors using techniques such as compressed histogram of gradients (CHoG) and progressively transmitting compressed data to avoid network transmission latency.[11] Another way to expedite multimedia retrieval process is to unify the approach of retrieving and processing various multimedia data. A multimedia query language called the MPEG Query Format (MPQF) saves complex interpretation among all kinds of description formats by generally expressing multimedia requests.[12] Different from the earlier-mentioned research work, our proposed framework performs off-line training on the server side and uploads them periodically with the corresponding concept relationship. Thus, users can retrieve a set of well-trained models in real time without end-to-end network latency.

References

1. M.S. Lew et al., "Content-Based Multimedia Information Retrieval: State of the Art and Challenges," *ACM Trans. Multimedia Computing, Communications, and Applications,* vol. 2, no. 1, 2006, pp. 1–19.
2. J. Yu and Q. Tian, "Learning Image Manifolds by Semantic Subspace Projection," *Proc. 14th Ann. ACM Int'l Conf. Multimedia,* ACM, 2006, pp. 297–306.
3. J. Huang et al., "Image Indexing Using Color Correlograms," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition,* IEEE CS, 1997, pp. 762–768.
4. P. Smaragdis and M. Casey, "Audio/Visual Independent Components," *Proc. 4th Int'l Symp. Independent Component Analysis and Blind Source Separation* (ICA), 2003, pp. 709–714; www.kecl.ntt.co.jp/icl/signal/ica2003/.
5. A.V. Nefian et al., "Dynamic Bayesian Networks for Audio-Visual Speech Recognition," *EURASIP J. Advances in Signal Processing,* vol. 1900, no. 11, 2002, pp. 1274–1288.
6. D. Liu et al., "IR and Visible-Light Face Recognition Using Canonical Correlation Analysis," *J. Computational Information Systems,* vol. 5, no. 1, 2009, pp. 291–297.
7. T. Westerveld et al., "A Probabilistic Multimedia Retrieval Model and its Evaluation," *EURASIP J. Applied Signal Processing,* vol. 2003, 2003, pp. 186–198.
8. C. Chen et al., "Web Media Semantic Concept Retrieval via Tag Removal and Model Fusion," *ACM Trans. Intelligent Systems and Technology,* vol. 4, no. 4, article no. 61.
9. A. Bendjebbour et al., "Multisensor Image Segmentation Using Dempster-Shafer Fusion in Markov Fields Context," *IEEE Trans. Geoscience and Remote Sensing,* vol. 39, no. 8, 2001, pp. 1789–1798.
10. M.E. Sargin et al., "Audiovisual Synchronization and Fusion Using Canonical Correlation Analysis," *IEEE Trans. Multimedia,* vol. 9, no. 7, 2007, pp. 1396–1403.
11. V.R. Chandrasekhar et al., "Low Latency Image Retrieval with Progressive Transmission of Chog Descriptors," *Proc. ACM Multimedia Workshop on Mobile Cloud Media Computing,* ACM, 2010, pp. 41–46.
12. M. Doller et al., "The MPEG Query Format: Unifying Access to Multimedia Retrieval Systems," *IEEE MultiMedia,* vol. 15, no. 4, 2008, pp. 82–95.

(HOG), color and edge directivity descriptor (CEDD), and other low-level visual features.

The essential idea behind the HOG descriptors is that local object appearance and shape within an image can be characterized by the distribution of local intensity gradients or edge directions, even without precise knowledge of the corresponding gradient or edge positions. The HOG descriptor maintains a few key advantages over other descriptor methods. It captures the local edge or gradient structure that is invariant to a low degree of geometric and photometric transformations in the local area.

CEDD is a popular low-level feature descriptor that incorporates both color and texture features in a histogram. The size of a CEDD is limited to 54 bytes per image, making this descriptor suitable for large image databases. The CEDD histogram consists of $6 \times 24 = 144$ regions, where the six regions are determined by the texture component and the 24 regions are originated from the color component.

The extracted low-level features include 48-dimension features for a color histogram in the HSV space, 120-dimension local features for color moment in the YCbCr space, and 260-dimension features for texture wavelet.

Feature Correlation Analysis (FCA)

The proposed FCA method explores the interrelationships among the features to establish the basis for identifying HCFGs.

Let $X = \{x_i\}_{i=1}^{N}$ be a given dataset, where $x_i \in R^L$ represents each instance in the dataset and N and L are the number of instances and the

dimension of the feature set $\left\{\mathbf{f}^i\right\}_{i=1}^{L}$, respectively. Then, the feature matrix $\mathbf{F}$ of $\mathbf{X}$ is represented as

$$\begin{bmatrix} f_1^1 & f_1^2 & \cdots & f_1^L \\ f_2^1 & f_2^2 & \cdots & f_2^L \\ \vdots & \vdots & \ddots & \vdots \\ f_N^1 & f_N^2 & \cdots & f_N^L \end{bmatrix}$$

where the ith column represents $\mathbf{f}^i$ and rows are instances in $\mathbf{X}$. Let $\left(\mathbf{f}^j, \mathbf{f}^k\right)$, $1 \le j, k \le L$, be a feature pair. Then the correlation coefficient between them can be calculated as follows:

$$C_{\mathbf{f}^j,\mathbf{f}^k} = \frac{\sum_{i=1}^{N}(f_i^j - \overline{\mathbf{f}^j})(f_i^k - \overline{\mathbf{f}^k})}{\sqrt{\sum_{i=1}^{N}(f_i^j - \overline{\mathbf{f}^j})^2}\sqrt{\sum_{i=1}^{N}(f_i^k - \overline{\mathbf{f}^k})^2}} \quad (1)$$

where $\overline{\mathbf{f}^j}$ and $\overline{\mathbf{f}^k}$ are the mean values of $\mathbf{f}^j$ and $\mathbf{f}^k$, respectively.

This correlation coefficients analysis method is based on the calculation of the Pearson product-moment correlation coefficient, which assumes normally distributed data and the linear relationship between feature variables. However, this is not always the case. To account for situations where the feature variables follow a nonlinear relationship, we propose another correlation estimation method based on the Spearman's rank correlation coefficients, which use the ranks of the observations instead of their values:

$$C_{\mathbf{p}^j,\mathbf{p}^k} = \frac{\sum_{i=1}^{N}(p_i^j - \overline{\mathbf{p}^j})(p_i^k - \overline{\mathbf{p}^k})}{\sqrt{\sum_{i=1}^{N}(p_i^j - \overline{\mathbf{p}^j})^2}\sqrt{\sum_{i=1}^{N}(p_i^k - \overline{\mathbf{p}^k})^2}} \quad (2)$$

where $\mathbf{p}$ is the rank representation of the feature vector $\mathbf{f}$. (The rank representation means the rank of a variable in a feature vector with a specific order—for example, by value.)

Finally, the feature correlation matrix $\boldsymbol{C}$ is constructed as

$$\begin{bmatrix} C_{\mathbf{v}^1,\mathbf{v}^1} & C_{\mathbf{v}^1,\mathbf{v}^2} & \cdots & C_{\mathbf{v}^1,\mathbf{v}^L} \\ C_{\mathbf{v}^2,\mathbf{v}^1} & C_{\mathbf{v}^2,\mathbf{v}^2} & \cdots & C_{\mathbf{v}^2,\mathbf{v}^L} \\ \vdots & \vdots & \ddots & \vdots \\ C_{\mathbf{v}^L,\mathbf{v}^1} & C_{\mathbf{v}^L,\mathbf{v}^2} & \cdots & C_{\mathbf{v}^L,\mathbf{v}^L} \end{bmatrix}$$

where $\mathbf{v}$ could be the feature vector $\mathbf{f}$ or its rank vector $\mathbf{p}$. Each element in the matrix presents the correlation coefficient between each feature pair, creating a symmetric matrix—that is, $C_{\mathbf{v}^j,\mathbf{v}^k}$ equals $C_{\mathbf{v}^k,\mathbf{v}^j}$.

All the correlation coefficients are calculated based only on the positive instances, thus identifying relationships between the features in a supervised manner (that is, per concept). In addition, the inclusion of the negative instances may hinder the discovery of correlations between feature pairs. An added benefit is the improved computational efficiency of the system, which is an important requirement in mobile systems.

Feature Grouping via Affinity Propagation

Because of its simplicity, general applicability, and performance, the AP algorithm has found application in science and engineering fields,[4] which inspired us to adapt it to our framework for feature clustering. Specifically, we choose to use AP algorithm for the following reasons:

- AP generates clusters with much lower error than other clustering methods, such as k-means and mixtures of Gaussian.
- AP is deterministic—that is, its clustering results do not depend on initialization, unlike most clustering methods such as k-means.
- AP can automatically determine the number of clusters.

Considering each feature as a data point, the input for AP is the similarity matrix **S,** with each element computed as

$$s(\mathbf{v}^j, \mathbf{v}^k) = C_{\mathbf{v}^j,\mathbf{v}^k} \quad (3)$$

The AP algorithm propagates affinities by passing two types of messages between two data points (for example, features $\mathbf{v}^j$ and $\mathbf{v}^k$)[5] as follows:

- The responsibility $r(\mathbf{v}^j, \mathbf{v}^k)$ is sent from $\mathbf{v}^j$ to $\mathbf{v}^k$, representing how well $\mathbf{v}^j$ serves as the exemplar of v^k considering other potential exemplars for $\mathbf{v}^j$.
- The availability $a(\mathbf{v}^j, \mathbf{v}^k)$ is sent from $\mathbf{v}^k$ to $\mathbf{v}^j$, reflecting how appropriate $\mathbf{v}^j$ chooses $\mathbf{v}^k$ as its exemplar considering other potential features that may choose $\mathbf{v}^k$ as their exemplar.

The responsibility and availability are updated iteratively using the following equations:

$$r(\mathbf{v}^j, \mathbf{v}^k) \leftarrow s(\mathbf{v}^j, \mathbf{v}^k) - \max_{l:l\neq k}(a(\mathbf{v}^l, \mathbf{v}^j) + s(\mathbf{v}^j, \mathbf{v}^l)) \quad (4)$$

$$a(\mathbf{v}^k, \mathbf{v}^j) \leftarrow \min \left(0,\ r(\mathbf{v}^k, \mathbf{v}^k) + \sum_{l:l\notin\{k,j\}} \max\{0, r(\mathbf{v}^l, \mathbf{v}^k)\}\right) \quad (5)$$

The self-availability is updated as

$$a(\mathbf{v}^k, \mathbf{v}^k) \leftarrow \sum_{l:l\neq k} \max\{0, r(\mathbf{v}^l, \mathbf{v}^k)\} \quad (6)$$

This message reflects an accumulated confidence that feature $\mathbf{v}^k$ is an exemplar, based on

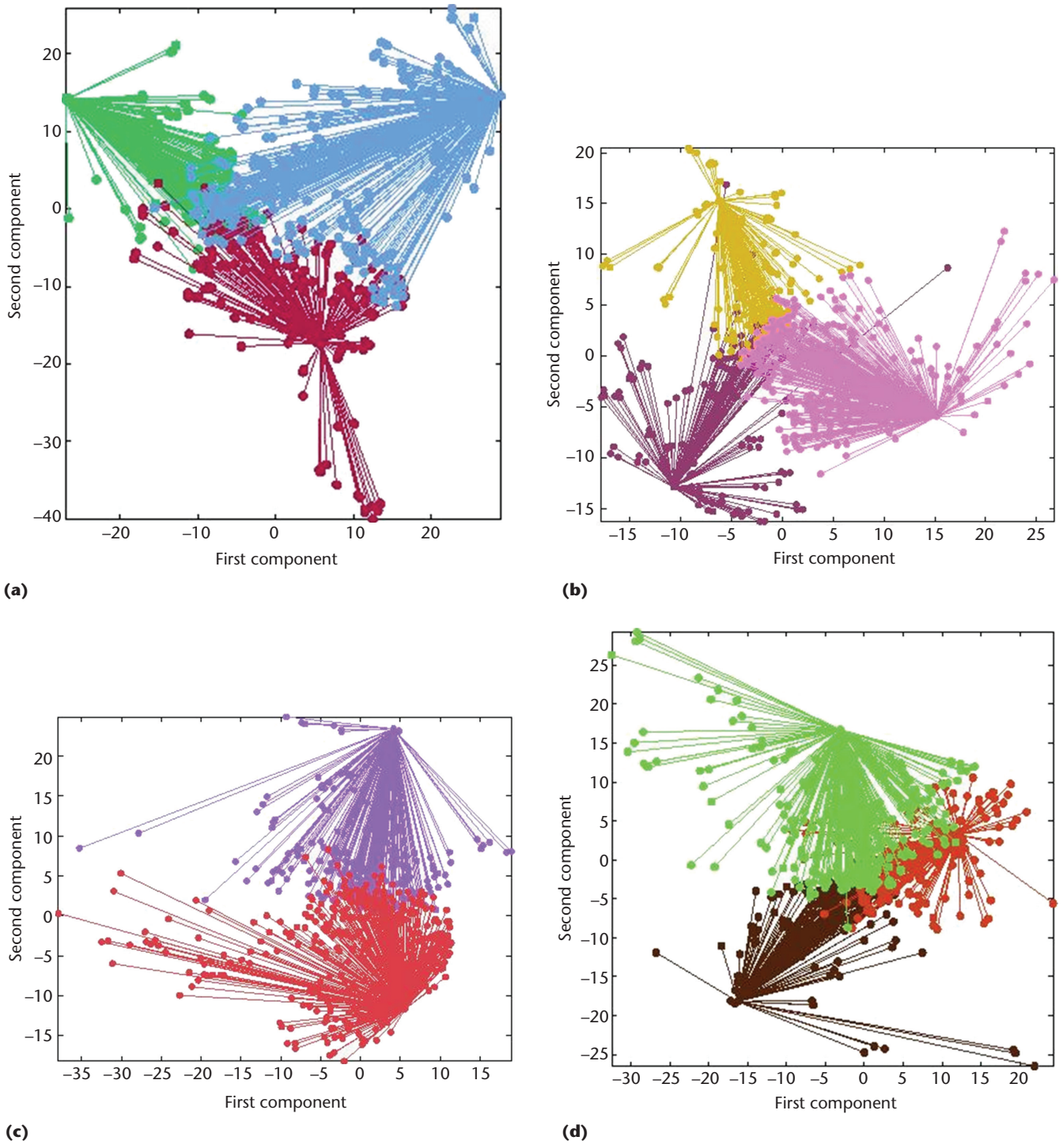


Figure 2. Feature grouping results for four disaster semantic concepts. (a) Road debris, (b) earthquake, (c) flood, and (d) volcano.

the positive responsibilities sent to the candidate exemplar k from other features.

Finally, the exemplar for feature $\mathbf{v}^j$ is chosen as follows:

$$e_j^* \leftarrow \arg\max_{\mathbf{v}^k}(r(\mathbf{v}^j, \mathbf{v}^k) + a(\mathbf{v}^k, \mathbf{v}^j)) \quad (7)$$

Figure 2 illustrates the feature grouping results for four disaster topics (with a preference value set to 30 times the minimum similarity and using the visual features described earlier), where the x and y axes represent the first and second component of the features in the

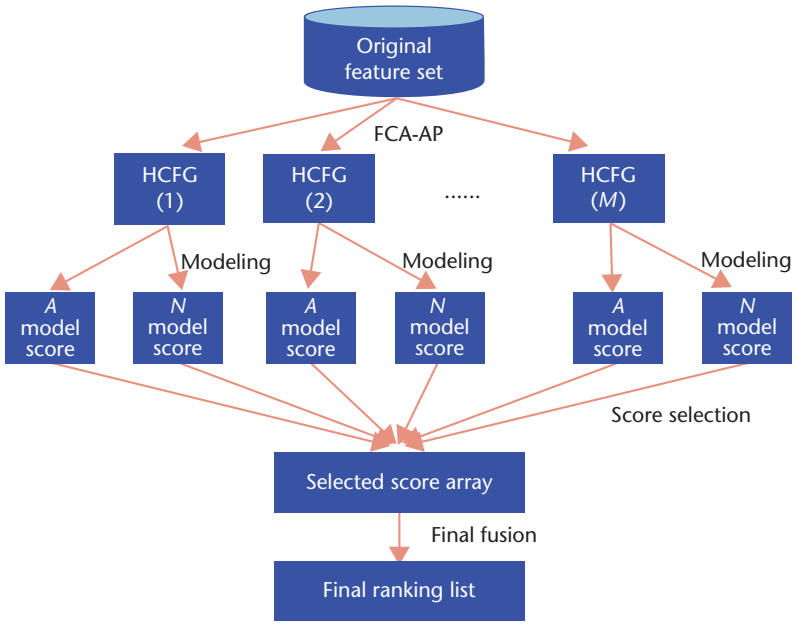


Figure 3. Multimodel fusion procedure. This procedure ensures the best HCFGs are selected for the fusion to optimize the final retrieval performance.

projected subspace using PCA. Each colored point in the plots represents one feature. All the feature points belonging to the same group are of the same color, and there is a line between the exemplar feature point and each member of the feature group. This figure demonstrates that the proposed features grouping method is capable of capturing the underlying correlation among all the features and separates them into different feature groups. Each feature group potentially implies distinct contexts relating the disaster topic.

Model Fusion

Figure 3 depicts the multimodel fusion procedure. First, the feature correlation analysis and affinity propagation (FCA-AP) algorithm is applied to the original feature set, obtaining M HCFGs. Then each HCFG is modeled by a series of classifiers, named A through N, generating a score array, denoted as $\left[\text{Score}(t)_g^m\right]$, where t represents each concept and g and m denote the HCFG group ID and the model used for training, respectively. The score array is sorted against the training performance evaluated using MAP measurement. Only the top Q scores are kept for the final fusion. This procedure ensures the best HCFGs are selected for the fusion to optimize the final retrieval performance.

The selected scores from multiple models are combined using the refined formula from earlier work,[10] expressed as

$$\text{Score}(x) = \sum_{q=1}^{Q} \frac{\gamma_q \cdot \beta_q}{\gamma_q + \beta_q} \cdot \left(\frac{\text{Score}_q(x)}{\alpha_q}\right) \qquad (8)$$

The parameters are explained as follows:

- α_q denotes the refined scale factor for balancing the ranking score from the qth model. It is calculated as the absolute mean score for all the training instances for that model. We refine this parameter by taking the absolute value to accommodate negative scores.
- β_q expresses the relationship between the testing score for the qth model and the target concept, which is measured based on the correlation value between the testing score interval and the related concept.[6]
- γ_q represents the reliability of model q based on training performance. Specifically, it is calculated as the average precision of the qth model evaluated on the instances in the training set.

User Feedback Mechanism

One important component of our proposed system is the user feedback system based on the Markov model mediator (MMM).[7] The objective is to improve the multimedia semantic retrieval performance by incorporating user interaction. The MMM mechanism is used to model the searching and retrieval process for content-based image retrieval. One distinctive characteristic of the MMM model is that it carries out the searching and similarity computing process dynamically, taking into consideration not only the image content features but also other properties of multimedia data instances such as their access frequencies and access patterns.

MMM is a probability-based mechanism that adopts the Markov model framework and the mediator concept. The MMM mechanism models a multimedia database by a five-tuple $\lambda = (S, F, A, \boldsymbol{B}, \pi)$, where S is a set of instances called states; F is a set of distinct features of the instances; A denotes the state transition probability distribution, where each entry (i, j) indicates the relationship between instances i and j captured through the offline training procedure; $\boldsymbol{B}$ is the feature matrix of all instances; and π is the initial state probability distribution.

Table 1. Disaster image dataset.

ID	Disaster topic	No. of images
1	Avalanche	624
2	Drought	599
3	Earthquake	884
4	Flood	1,009
5	Ice storm	1,078
6	Mudflow	266
7	Oil spill	1,847
8	Volcano	800
9	Tornado	266
10	Gas explosion	1,019
11	Road debris	2,009
Total		10,401

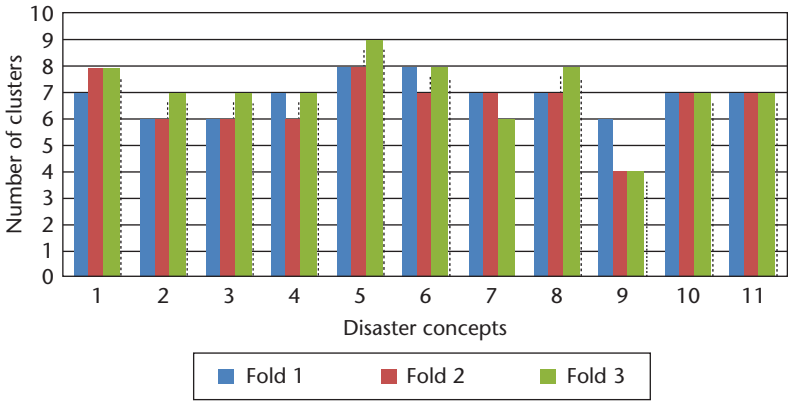


Figure 4. Number of groups for each concept. The fusion scheme is evaluated using three-fold cross validation, and the number of groups per fold ranges from four to nine.

The training of MMM involves the construction of the two statistical matrices: $\boldsymbol{A}$ and π. A sequence of user feedback characterizing access patterns and access frequencies is used to train the model parameters. Specifically, the training of the two parameters are described as follows.

The training of matrix $\boldsymbol{A}$ is based on the intuition that the more frequently two images are accessed together, the more closely related they are. To capture the relative affinity measurements among all the instances, a matrix $\boldsymbol{AF}$ is constructed with each element $af_{i,j}$ representing the relative affinity relationship between two instances i and j as

$$af_{i,j} = \sum_{d=1}^{D} P_{i,d} \times P_{j,d} \times AC_d \qquad (9)$$

where $P_{i,d}$ denotes the feedback pattern of instance i in time period d and AC_d represents the access frequency in that time period.

For matrix π, the preference of the initial states for user feedback can be obtained from the training dataset. For any instance i, the initial state probability is defined as the fraction of the number of occurrences of instance i with respect to the total number of occurrences for all the images in the image database from the training dataset.

Experimental Analysis

To evaluate our proposed framework, we used a disaster dataset that contains more than 10,000 images with the associated tags and descriptions covering 11 disaster topics. The images, which were crawled from Flickr, include both natural disasters such as earthquakes and floods and manmade disasters like road debris and oil spills. Table 1 shows the composition of the dataset.

To thoroughly evaluate the effectiveness of the proposed framework, we conducted a series of experiments. First, we analyzed the significance of the feature grouping approach by discussing the number of feature groups. Second, the multimodel fusing scheme was evaluated using the disaster image dataset under three-fold cross validation. Finally, we compared the overall performance of our fusion framework with the other modeling methods.

The evaluation criterion is the well-known mean average precision (MAP), which is widely used in the information retrieval community. The MAP is calculated as

$$\mathrm{MAP}(T) = \frac{1}{|T|} \sum_{i=1}^{|T|} \frac{1}{n_i} \sum_{j=1}^{n_i} \mathrm{Precision}(R_{ij}) \qquad (10)$$

where $|T|$ is the total number of queried concepts and R_{ij} is the top j ranked results for concept i.

Evaluation on the Disaster Image Dataset

The AP algorithm has a heuristic parameter P, or preference, that indicates the preference that an instance is chosen as an exemplar. Previous work showed that the number of groups monotonically increases with P polynomially.[4] The value of P is empirically set to –10 in the following experiments. Figure 4 shows the number of groups for each concept in each of the three folds, which range from four to nine. Our experimental analysis shows the advantages of our proposed feature grouping method—that is, the decomposition of features

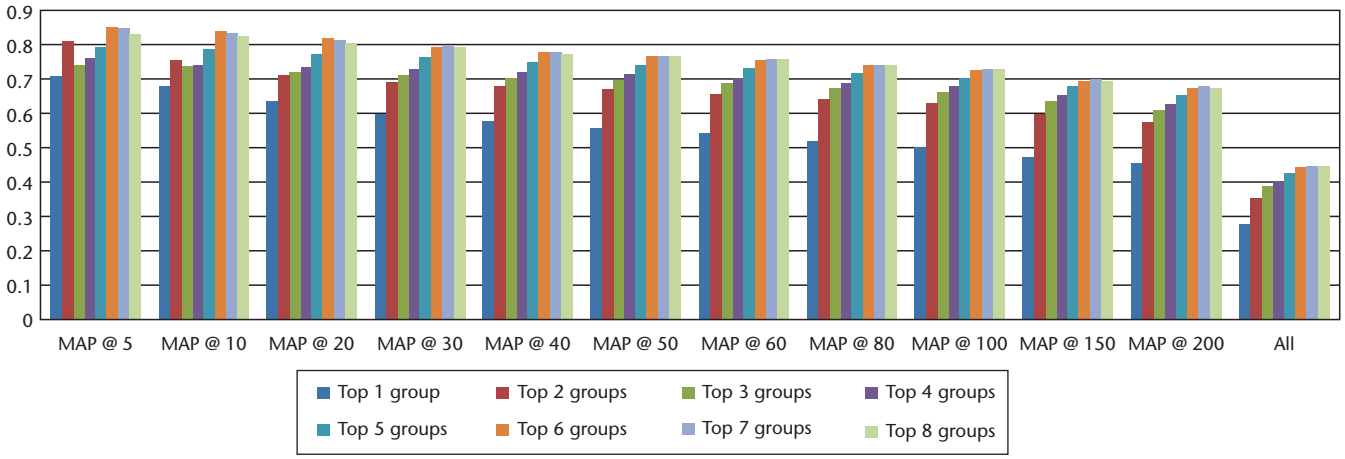


Figure 5. Mean average precision (MAP) values for different number of hidden coherent feature groups (HCFGs). Results show the performance stabilizes when the number of models reaches the top six groups.

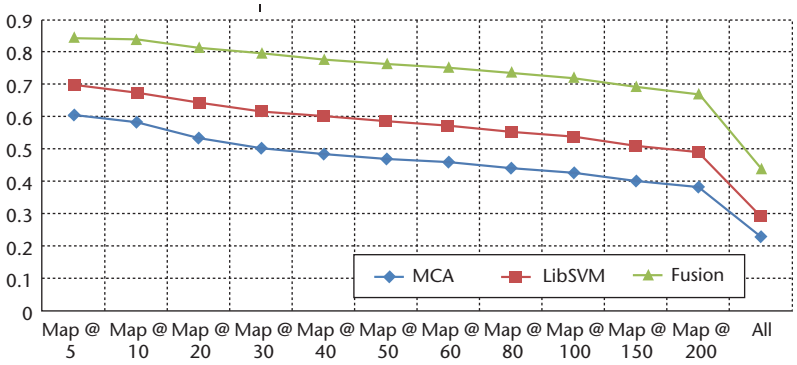


Figure 6. Mean average precision (MAP) values for different modeling methods and the proposed fusion scheme. The fused scheme outperforms single models by taking advantages of both the LibSVM[8] and multiple correspondence analysis (MCA) modeling methods.[6]

enables parallel processing, which is an important characteristic for mobile applications. In addition, the feature grouping method keeps all the original information, thus avoiding potential information loss by using the previously discussed subspace analysis methods.

Figure 5 shows the MAP values when selecting different numbers of models for multimodel fusion. The MAP values increase as more groups (models) are selected for final fusion, which is intuitive because we add more valuable information for the final decision. In addition, the performance stabilizes when the number of models reaches certain point—in this case, the top six groups—which indicates that we capture the most important information for the final decision with a subset of the original features. This also means that our framework can automatically filter out the irrelevant information that is not useful for the final decision making.

We further compared the final fusion results with the average performance for all the groups using the LibSVM[8] and multiple correspondence analysis (MCA) modeling methods,[6] as Figure 6 shows. The results demonstrate that the fused scheme outperforms single models by taking advantages of both models. It is worth noting that our framework can adapt to multiple training models and can optimize the overall performance by fusing the most promising HCFGs from different models.

Multimedia Retrieval via Mobile Devices

Based on our proposed framework, we developed an iPad application that follows a three-tiered architecture (see Figure 7). The production database is implemented as a PostgreSQL database that stores all the processing results of the backend system. Accessing the database and performing complicated data queries is done through the REST API, implemented as a Java Tomcat servlet (using the Restlet framework). On top of these two layers, the client is implemented in iOS, specifically for Apple's iPad devices.

Figure 8 shows two search results with the developed application tested on the disaster image dataset. The application lets a user search for multimedia content based on one or more keywords. Upon submission of the search terms in the mobile application, these terms are sent to our backend server, which dynamically generates a query to search our database for images that match the given keywords. Relevant

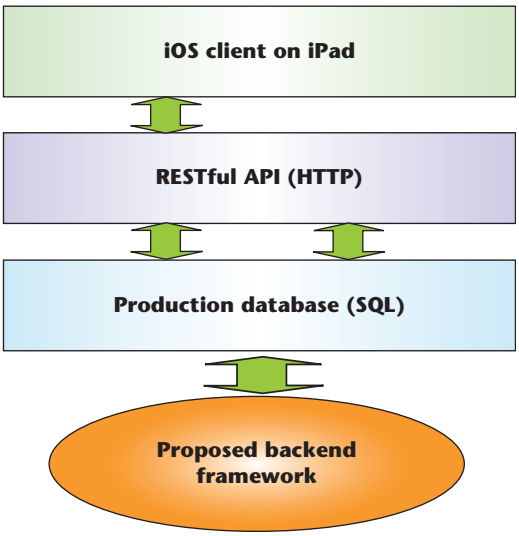
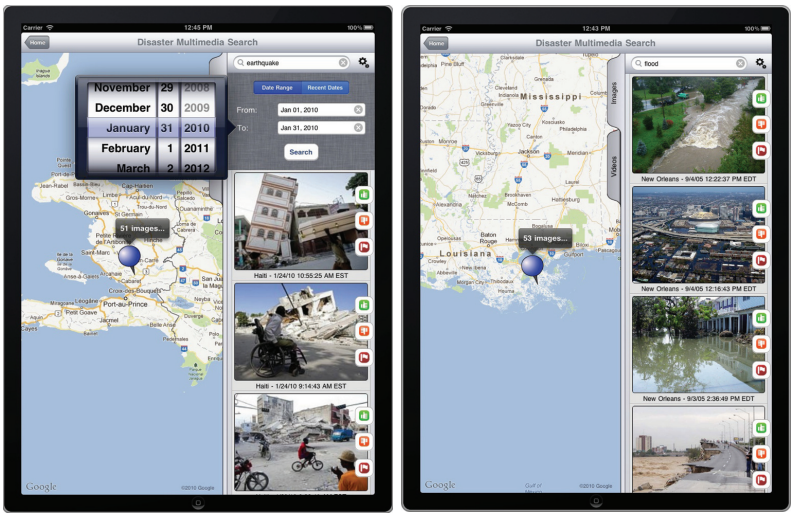


Figure 7. Application architecture. The iPad application follows a three-tiered architecture.



(a) (b)

Figure 8. Application interface. (a) Search results using keyword "earthquake." (b) Search results using keyword "flood."

information about each image is then sent to the mobile application. This information includes the keywords (concept names and their synonyms) associated with the image, its subject, its location, its description, and a URL for retrieving the image for display. The mobile application is designed with a built-in image cache. If the system receives a request to display an image multiple times, the cache is checked first, before a call is made to retrieve the image from the servers. This reduces overhead when retrieving and displaying images.

In addition to simply searching based on keywords, the system also lets the user specify a date range for the search. This enables the user to search for images that are relevant to a specific disaster event. Once the user has submitted a search, the mobile application groups all the images based on location and displays them on the map to the left. Selecting one of the push pins on the map filters the list of images, showing only the images at the specific location.

Moreover, users can provide feedback on the retrieval results with the following three options: "thumbs up," meaning the system made a correct match but some images are more relevant than others; "thumbs down," meaning the system made a correct match but some images are less relevant than others; and "flag," meaning the image is completely inappropriate and should be hidden from all future image lists. This user feedback is collected and processed by the MMM component to further refine the retrieval results.

Conclusion

This article only focuses on visual features and presents a novel correlation-based feature analysis method to derive HCFGs for multimedia semantic retrieval on mobile devices. In the future, we will incorporate textual information and further improve the retrieval performance by fusing information from multiple modalities. In addition, we will also conduct more experimental analyses on video retrieval using the improved framework. From application perspective, our proposed framework will be integrated into a disaster management system and play an important role in enhancing the situation report and benefitting the decision-making process. **MM**

Acknowledgments

This research was supported by the US Department of Homeland Security under grant 2010-ST-062-000039, the US Department of Homeland Security's VACCINE Center under grant 2009-ST-061-CI0001, and the National Science Foundation under grant HRD-0833093. Many thanks to Jesse Domack for developing the front end of the system.

References

1. S. Cherry, "Cloud Computing Drives Mobile Data Growth," *IEEE Spectrum*, vol. 46, no. 10, 2009, p. 52.
2. M.S. Lew, et al., "Content-Based Multimedia Information Retrieval: State of the Art and Challenges,"

ACM Trans. Multimedia Computing, Communications, and Applications, vol. 2, no. 1, 2006, pp. 1–19.
3. P.K. Atrey, et al., "Multimodal Fusion for Multimedia Analysis: A Survey," *Multimedia Systems,* vol. 16, no. 6, 2010, pp. 345–379.
4. Y. Yang and S.-C. Chen, "Disaster Image Filtering and Summarization Based on Multi-layered Affinity Propagation," *Proc. IEEE Int'l Symp. Multimedia,* IEEE CS, 2012, pp. 100–103.
5. B.J. Frey and D. Dueck, "Clustering by Passing Messages Between Data Points," *Science,* vol. 315, no. 5814, 2007, pp. 972–976.
6. C. Chen, et al., "Web Media Semantic Concept Retrieval via Tag Removal and Model Fusion," *ACM Trans. Intelligent Systems and Technology,* vol. 4, no. 4, article no. 61.
7. M.-L. Shyu, et al., "Affinity Relation Discovery in Image Database Clustering and Content-Based Retrieval," *Proc. ACM Multimedia,* ACM, 2004, pp. 372–375.
8. C.C. Chang and C.J. Lin, "Libsvm: A Library for Support Vector Machines," *ACM Trans. Intelligent Systems and Technology,* vol. 2, no. 3, 2011, article no. 27.

Yimin Yang is a doctoral candidate in the School of Computing and Information Sciences at the Florida International University (FIU), Miami. Her research interests *include multimedia data mining,* multimedia systems, and image and video processing. Yang has an MS in computer science from FIU. Contact her at yyang010@cs.fiu.edu.

Hsin-Yu Ha is a doctoral candidate in the School of Computing and Information Sciences at the Florida International University (FIU), Miami. Her research interests include multimedia data mining. Ha has an MS in computer science from FIU. Contact her at hha001@cs.fiu.edu.

Fausto C. Fleites is a doctoral candidate in the School of Computing and Information Sciences at the Florida International University (FIU), Miami. His research interests include multimedia indexing, multimedia data mining, and distributed systems. Fleites has an MS in computer science from FIU. Contact him at fflei001@cs.fiu.edu.

Shu-Ching Chen is a professor in the School of Computing and Information Sciences at the Florida International University. His research interests include distributed multimedia database management systems and multimedia data mining. Chen has a PhD in electrical and computer engineering from Purdue University. Contact him at chens@cs.fiu.edu.

Selected CS articles and columns are also available for free at http://ComputingNow.computer.org.