

MADIS: A Multimedia-Aided Disaster Information Integration System for Emergency Management

Yimin Yang, Wenting Lu, Jesse Domack, Tao Li, Shu-Ching Chen, Steven Luis, Jainendra K Navlakha
School of Computing and Information Sciences

Florida International University
Miami, FL 33199, USA

{yyang010, wlu, jdoma001, taoli, chens, luiss, navlakha}@cs.fiu.edu

Abstract—With the proliferation of smart devices, disaster responders and community residents are capturing footage, pictures and video of the disaster area with mobile phones and wireless tablets. This multimedia disaster situation information is critical for assisting emergency management (EM) personnel to effectively respond in a timely manner. Currently, however the data is not integrated in incident command systems where situation reports, incidence action plans, etc. are being held. Therefore, we have designed and developed a **Multimedia-Aided Disaster information Integration System (MADIS)**, which utilizes advanced data mining techniques to analyze situation reports and pictures as well as text captured in the field and automatically link the reports directly to relevant multimedia content. Specifically, a dynamic hierarchical image classification approach is proposed to categorize disaster images into different subjects by fusing image and text information. Situation reports are analyzed using advanced document processing techniques and then associated with processed multimedia data. In order to seamlessly incorporate user interactive activities for improving information integration, a user feedback processing scheme is proposed to refine the association between situation reports and images as well as the affinity among images. The system is developed on Apple’s mobile operating system (iOS) and runs on iPad tablets, and its usefulness is evaluated by domain experts from the local EM department.

Index Terms—Multimedia, disaster information management, emergency management, mobile application, hierarchical classification.

I. INTRODUCTION

In recent years, disasters such as hurricanes and earthquakes have caused huge damages in terms of both property loss and human lives. In 2005, hurricane Katrina reported a total property damage of \$81 billion. Thousands of people died in the actual hurricane and in subsequent floods. In 2010, the Haiti earthquake affected billions of people, and an estimated 550,000 buildings collapsed or were severely damaged. In order to reduce such loss, emergency managers are required to not only be well prepared but also provide rapid response activities [1].

For fulfilling a response plan in a disaster event, emergency management (EM) personnel should integrate jurisdictional resources, coordinate multi-agency responses, and establish executing processes among the EM community. However, currently decision makers responsible for emergency responses rely mostly on situation reports, which are usually textual

description of the disaster scene. The limitation of plain textual information provided by situation reports lowers the efficiency of assessing the disaster situation; hence the urgent need of additional multimedia information, such as pictures and videos taken at the disaster scene, for enhancing the text-based reports and providing more details of the disaster event. A system that integrates multi-source information such as textual reports and multimedia data would greatly assist emergency managers in making a better assessment of a disaster situation and performing efficient and timely responses correspondingly. Furthermore, due to its portable and ease-of-use characteristics [2], mobile devices have been proven to be a must-have utility in disaster management areas, especially when considering quick emergency response.

In this paper, built on our previous work [3], [4], we have designed and developed a **Multimedia-Aided Disaster information Integration System (MADIS)** that semantically associates situation reports with disaster-related multimedia data and is implemented within an iPad-specific application that conveys all such information via a unified and intuitive graphical interface. The mobility of the iPad device provides the EM personnel with free and fast interaction in communicating between both the command centers and the actual disaster sites. Compared with the original prototype, the advanced system has improved from both back-end techniques and front-end user experience perspectives. Specifically, a dynamic weighting scheme is introduced for automatically integrating multi-source multimedia information; a more comprehensive user feedback mechanism is designed for improving integration results; user interfaces are refined and more functionalities are included for better user experiences. The proposed MADIS system tries to solve the following problems and challenges:

- **Classification of images into different subjects by fusing image and text information:** Images taken at disaster scenes usually come along with descriptive information which is of great help for better understanding of the imagery data. However, how to effectively fuse text information with visual data for identifying the subjects in images is a challenging problem. In order to solve this issue, a dynamic hierarchical classification mechanism is proposed to classify images into various subjects

using semantic analysis techniques based on Multiple Correspondence Analysis (MCA) [5] and a self-adaptive weighting scheme for information fusion.

- **Associating situation reports with classified images:** After the images have been properly classified, the next problem is how to analyze the situation reports and build the relationship between the report and multimedia data. An intuitive solution is to identify the same subjects as assigned to the images. For solving this problem, advanced text and document processing techniques such as GATE [6] system and WordNet [7] are utilized to analyze and extract location and subject-related information, which is further used to build the association between situation reports and classified images.
- **Incorporating user feedback for better association:** User feedback plays an important role in refining data integration results and helps to improve the system and provide better services. There are different types of feedback regarding the targeted resources. For example, users may not only show interest in the relationship between images and reports but also in the affinities among images. In our proposed system, a comprehensive user feedback processing mechanism is presented to refine both report-image association and image-image affinity based on the Markov Model Mediator (MMM) [8] mechanism inspired by the Markov model theory [9].

The rest of the paper is organized as follows. Section 2 presents the overview of MADIS. Section 3 discusses the dynamic hierarchical image classification based on MCA in detail. Section 4 describes the document processing procedure and how to associate classified images with situation report. Section 5 discusses the incorporation of user feedback for better association. Section 6 describes the system evaluation. Finally, section 7 concludes the paper.

II. MADIS OVERVIEW

MADIS is a multi-source information integration framework designed and developed on mobile platform for enhancing situation report and enabling quick emergency response. The system adopts advanced data mining techniques for multimedia content analysis and document processing, which semantically associates situation reports with multimedia data. The developed iPad application provides the EM personnel with an intuitive and interactive solution for fast and efficient disaster situation assessment.

As depicted in Figure 1, MADIS takes as input images, text, situation reports, and user feedback for multi-source information integration and renders a user-friendly mobile platform for effective and timely emergency responses. To fuse image and textual information for image classification, the system performs multimedia analysis based on the collected data and categorizes the images into different subjects via a hierarchical structure and dynamic weighting schema. At the same time, document analysis is conducted upon the situation reports, which tries to build the association with the classified images. The MMM mechanism is applied to

incorporate user feedback for adjusting the affinity between images and obtaining better association results. At the front-end of the system, a series of controllers are used to control different views of the system, including report lists, related images for reports, images filter, image timelines, and related images for image, which will be touched in the next section.

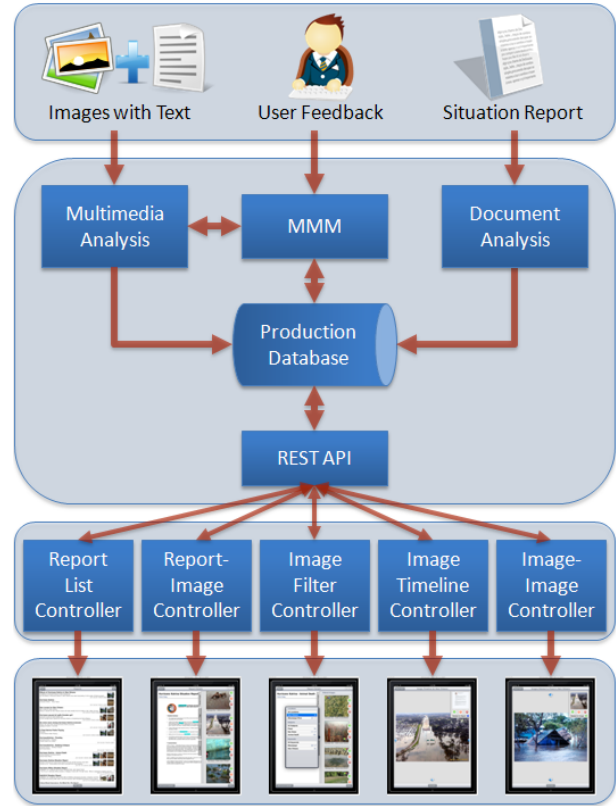


Fig. 1. MADIS Overview

A. MADIS Architecture

The implementation of MADIS follows a three-tiered architecture: (1) the client (iPad application); (2) the RESTful, JSP-based API; and (3) the production database.

The production database is a relational database built using PostgreSQL. It stores all the data related to the situation reports, including multimedia data as well as user-feedback. The relational schema of the database models the semantic relationship between the situation reports and the multimedia data as shown in Figure 2. In their contents, situation reports may reference one or more geographic locations and subjects, which are in turn described by pictures taken at the disaster area. For example, in the scenario of a hurricane that affects South Florida, geographic locations may be Miami-Dade or Miami Beach. Such locations are represented by images, which can be categorized into before or after the natural hazard. The subjects of images are the damages affecting the corresponding locations, such as "building collapse" and "flooding" in the hurricane disaster.

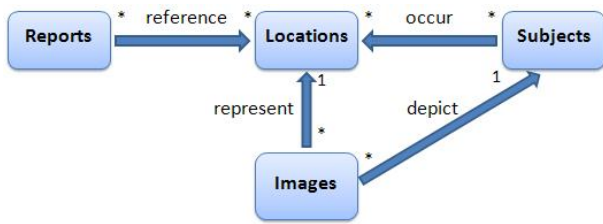


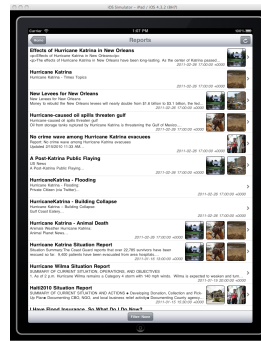
Fig. 2. MADIS Database Relational Schema

The RESTful, JSP-based API answers requests from the user interface by accessing the production database via structured queries. The REST API is implemented as a Java Tomcat servlet and follows the Model-View-Controller (MVC) design pattern. All the requests and responses are in XML format. For example, through this RESTful API, the front-end application can retrieve situation report related information, such as the list of reports, the list of locations and subjects associated with the reports, and the list of images that related to such locations and subjects, etc. It can also send user feedback to the back end and re-arrange related images based on feedback processing results. Over the above two layers, the top tier is implemented in iOS, specifically for Apple’s iPad devices. The iPad application communicates with the server layer via RESTful API and XML-based responses, finally presenting a user-friendly graphical interface for information retrieval and active interaction.

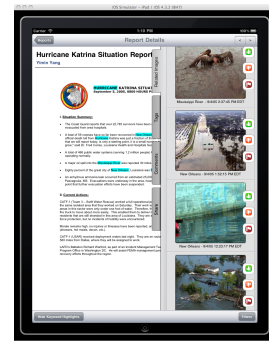
B. MADIS Major Components

The major components of MADIS are illustrated in Figure 3 and described as follows.

- **Report List:** This component shows the main report list, which displays one to three related images next to each entry. They are the most recently taken pictures, each of a different subject associated with the report as shown in Figure 3(a).
- **Related Images for Report:** Once the user enters a specific report page, he/she can browse the related images associated with current report. Long press on any one image will bring up the voting options for user feedback as shown in Figure 3(b).
- **Image Filter:** This component allows a user to filter the image list based on several factors simultaneously. The images can be filtered by locations, subjects, or keywords (which are the synonyms of locations and subjects, being highlighted in the report) as shown in Figure 3(c). In each case, the user can select multiple values to filter on and the image list is updated dynamically. This feature can be useful for displaying only the images that pertain to a specific aspect of the report.
- **Image Timeline:** Users may enter the image page and view the timeline by selecting an image from the related image list. The timeline is a set of images that depict the same location and are organized by date from earliest to latest. Users are allowed to vote for an image to report relationship under this view as shown in Figure 3(d).



(a) Report List



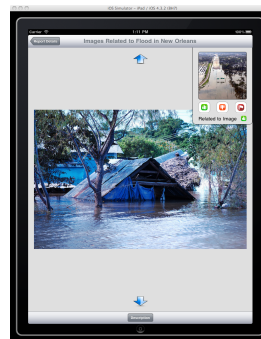
(b) Related Images for Report



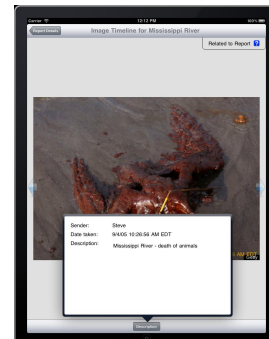
(c) Image Filter



(d) Image Timeline



(e) Related Images for Image



(f) Image Description

Fig. 3. MADIS Major Components

- **Related Images for Image:** Besides the report-image association, the system also presents the image-image relationships and provides the user with voting options as shown in Figure 3(e), where the anchor image is selected from the report page. In addition, the user can tap the description button to get a basic description of the image and additional metadata we may have on the picture, such as taken date and author as shown in Figure 3(f).

III. DYNAMIC HIERARCHICAL IMAGE CLASSIFICATION

The adaptive hierarchical image classification framework addresses multi-source data fusion via MCA and dynamic weighting scheme. MCA has been proven to be effective for multimedia semantic analysis, especially for video concept detection [10]. In the disaster image classification scenario,

the MCA algorithm is introduced for mining the correlation between multi-source data (i.e., image and text) and subjects, such as “building collapse” and “flooding”. This section describes the component of dynamic hierarchical image classification based on data fusion and MCA. Depicted in Figure 4, the framework is composed of two main components: multi-source model training and hierarchical classification. During the model training process, visual and text features are extracted respectively and fused based on the dynamic weighting scheme presented in Section III-C3. Then the models at different granularity levels are trained based on the MCA mechanism, generating thresholds for classification.

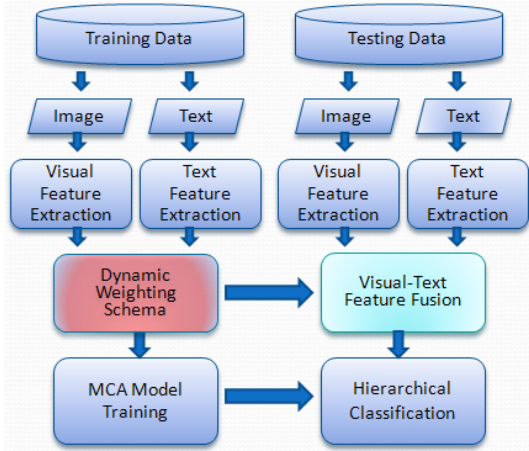


Fig. 4. Dynamic Hierarchical Image Classification framework

A. MCA for Multimedia Content Modeling

MCA is an exploratory data analytic technique designed to analyze multi-way tables for some measure of correspondence between the rows and columns [11]. The input to MCA is a set of nominal variables composed of several levels coded as a binary value. MCA can accommodate quantitative variables by recording them as bins, which inspires the idea that it could be applied to numerical data, such as multimedia feature instances. For example, each image feature variable could be discretized into several intervals (items); then an image can be presented by a series of nominal values. Motivated by its quantitative analysis ability, MCA has been explored to analyze the data instances described by a set of low-level features to capture the relationship between items and classes (subjects).

The multimedia data are projected into a new space by using the first two principal components obtained from the MCA calculation. The similarity of every item and every class can be presented by the inner product of each item and class, i.e., the cosine of the angle between each item and class. A smaller angle indicates a higher correlation between the item and class.

B. MCA Model Training

The training of MCA model can be summarized into the following four steps:

- **Angle Generation:** As a representation of the similarity between the items and classes as described in Section III-A, the angle is calculated using the equation below:

$$\text{angle}_k^j = \arccos\left(\frac{T_k^j \cdot C}{|T_k^j| |C|}\right), \quad (1)$$

where T and C are two-dimensional principal components denoting items and classes respectively, and j, k are indicators of items and features.

- **Weight Calculation:** The generated angles are applied to weight conversion as shown in the following equation:

$$\text{weight}_k^j = \pm(1 + \cos(\text{angle}_k^j \times \pi/180)). \quad (2)$$

The weight here is a measure of the similarity between each item and class.

- **Score Calculation:** The sum of all of the weights within one instance is denoted as *score* (shown in Equation (3)), which is the final evaluation of the relationship between each instance and class. A higher score implies a higher possibility that the instance belongs to the class. This implies the existence of a cut point (threshold), which determines the positive or negative attribute of one instance for certain class (subject).

$$\text{score}_i = \sum_{k=1}^K \text{weight}_k^j, i \in \{1, 2, \dots, N\} \quad (3)$$

- **Threshold Generation:** The threshold is determined by an interactive algorithm that finds out the best cut point based on the F1 measurement of training data.

C. Dynamic Visual-Text Information Fusion

This section describes the feature extraction processes for both visual and text data as well as the model training procedure based on the MCA algorithm. An iterative threshold determination algorithm is also presented to discover the most appropriate threshold for classification.

1) *Visual Feature Extraction:* There are mainly three steps for visual feature extraction: feature extraction, normalization, and discretization. The discretization step is special for the MCA model since it requires nominal input. The first two steps for both training images and testing images are the same; however, the discretization of the features of the testing images is based on the discretized intervals resulted from training image instances.

In order to capture the visual contents of images, two types of features are extracted: low-level color features and mid-level object location features shown as follows:

- *12 color features:* black, white, red, red-yellow, yellow, yellow-green, green, green-blue, blue, blue-purple, purple, and purple-red; the above color features for each image are generated from its HSV color space according to the combinations of different ranges of the hue, saturation and the intensity value [12], [13].
- *9 object location features:* In our work, we utilize the SPCPE algorithm [14] to extract object location features.

Specifically, each image is divided into 3×3 equal-sized regions, *i.e.*, nine locations are ordered from left to right and from top to bottom: L_1, \dots, L_9 , where $L_i = 1$ if there is an object in the image whose centroid falls inside L_i , $1 \leq i \leq 9$, otherwise $L_i = 0$. And the object with its area less than 8% area of the total region can be ignored. In order to effectively determine whether there is an object inside a designated region or not, we adopt the minimal bounding rectangle (MBR) concept in R-tree to guarantee that each object can be covered by a rectangle.

Therefore a total number of 21 features are obtained for each image, and these visual features will be integrated with corresponding text feature.

2) *Text Feature Extraction*: As for text feature extraction, it requires more preprocessing than visual features. First, the punctuation characters are removed and then the stop words, thus obtaining a list of valid words for each image instance. After that we analyze the above valid words related to each image, and then obtain the top N high-frequency words in the list by using MALLET [15], a Java-based package for statistical natural language processing. Since the extracted visual feature is a 21-dimension vector, in order to balance the contribution of different features to the subsequent classification results, we choose the top 21, (*i.e.*, $N=21$) words with high frequencies as the text features. Finally, each original text should be represented as an N-dimension feature vector by the combination of these high-frequency words according to the *tf-idf* schema discussed by Salton and McGill in [16]. Each dimensionality represents the number of times the high-frequency word appears in the text. The feature extraction process of the testing data set is similar to that of the training data set except for the “get word frequency” step.

3) *Dynamic Weighting Scheme*: Among the above 21 visual features, some of them might carry significant semantic information about the image, whereas some others might be less important. Particularly in the classification, the extracted features should be more representative and carry more significance. For example, when identifying sun and grass, color features red and green will play more important roles than other color features, such as yellow, blue, etc.; whereas when distinguishing sky from sea, the object location features might be more crucial than the color feature blue. Therefore, it might be helpful to dynamically assign different weights to different visual features so that the features with more importance can be captured and play more meaningful roles on the classification. In order to find out a suitable weight for each feature, a possible solution is to take the metric learning [17]–[19] into consideration. In this paper, we utilize the idea of metric learning and incorporate the concept of dynamic feature weighting into our solution. Fig. 5 presents the framework of dynamic weighting.

Specifically in the classification, given that human perception of an image is well approximated by its text, a good weighting schema for the extracted visual features guided by text information may lead to a high-quality similarity measurement, and therefore better classification results. Let

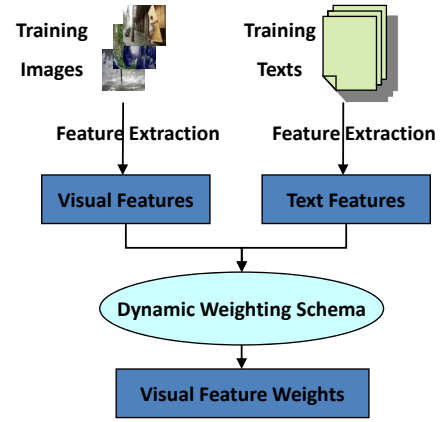


Fig. 5. Framework of Dynamic Weighting

$S_f(\mathbf{f}_i, \mathbf{f}_j; \alpha) = \sum_l f_{i,l} f_{j,l} \alpha_l$ be the image-based similarity measurement between the i -th and the j -th images when the parameterized weights are given by α , where $f_{i,l}$ is the l -th feature in the visual feature set f_i and $f_{j,l}$ is the l -th feature in the visual feature set f_j . Let $S_t(\mathbf{t}_i, \mathbf{t}_j) = \sum_k t_{i,k} t_{j,k}$ be the similarity measurement between the i -th and the j -th text features, in general, the words with high frequency extracted from texts. Here for each k , $t_{i,k}$ denotes whether the k -th word appears in the i -th text or not. To learn appropriate weights α for visual features, we can enforce the consistency between similarity measurements $S_f(\mathbf{f}_i, \mathbf{f}_j; \alpha)$ and $S_t(\mathbf{t}_i, \mathbf{t}_j)$. The above idea leads to the following optimization problem:

$$\alpha^* = \underset{i \neq j}{\operatorname{argmin}} \sum (S_f(\mathbf{f}_i, \mathbf{f}_j; \alpha) - S_t(\mathbf{t}_i, \mathbf{t}_j))^2 \quad \text{s.t. } \alpha \geq 0. \quad (4)$$

By rewriting and calculating the summation in Equation (4), the optimization problem can be addressed using quadratic programming techniques [20]. After obtaining the optimal weighting information for each visual feature, we can get the weighted visual features.

Similar to the visual feature, among those high-frequency words, some of them also might be more significant to the subsequent classification, whereas some others might be less important. Therefore, in order to learn appropriate weights β for text features, we can perform the similar weighting procedure to the text features. Note that the consistency is enforced between the similarity measurements of the weighted visual features under known weights α and vice versa. After obtaining the optimal weighting information for each text feature, both of the optimal weights α and β can be utilized in the subsequent hierarchical classification.

D. Hierarchical Classification

Much work has been done in the field of hierarchical classification [21]–[25]. For example, Fan *et al.* have built hierarchical mixture models for semantic image classification [22]; then extended the work by incorporating concept

ontology for hierarchical image concept organization [24]. Fei-Fei *et al.* have also incorporated prior knowledge to improve hierarchical image classification [23]. In order to explore the extensive relationship between various subjects and perform the classification in a more efficient way, an intuitive and simple hierarchical classification mechanism is adopted for our system [3]. Specifically, a topology tree is designed and used to classify images into pre-defined subjects in a top-down manner. Based on the fact that visual and text features at different layer may have unequal importance, the dynamic weighting scheme is applied at each granularity level to obtain a better integration result.

IV. DOCUMENT ANALYSIS AND IMAGE ASSOCIATION

This section addresses the problem of how to associate locations and subjects to documents, hence the association of situation report and classified images. Specifically, the GATE system is used to extract entities from document, and the WordNet tool is used to explore the synonyms of subjects in order to overcome the exact match limitation.

A. GATE System and Entity Extraction

Natural language processing for information retrieval plays an important role in the proposed system. The GATE system is applied to identify certain types of entities, such as date and location.

The GATE system requires three main processing resources: Tokenizer, Gazetteer and Grammar. GATE's annotation API communicates these resources by a directed graph. The implementation of the processing resources focuses on the robustness and usability of the system, as well as the clear distinction between declarative data representations and finite state algorithms. The Tokenizer splits text into simple tokens, for example, symbols, numbers, or words in different types, such as words with an initial capital, and so on. The Gazetteer is used to group entities and names of useful indicators such as IP, cities, organizations, or names of people. As for Grammar, it is constructed from hand-crafted rules to represent patterns by analyzing a specific text string or annotations previously attached to tokens.

B. WordNet for Synonym Extraction

WordNet is a lexical database for the English language. It groups English words into sets of synonyms called synsets, provides short, general definitions, and records the various semantic relations between these synonym sets. The development of synonym extraction component is based on the open source package which uses synonyms defined by WordNet [7]. The usage of the package requires users to download the WordNet prolog database. Inside this archive is a file named `wn_s.pl`, which contains the WordNet synonyms. We mainly use two classes in the package, i.e., `Syns2Index` and `SynLookup`. Specifically, the class `Syns2Index` is used for converting the prolog file `wn_s.pl` into a Lucene index suitable for looking up synonyms, and the class `SynLookup` is for looking up synonyms.

C. Report-Image Association

The purpose of document analysis is to associate locations and subjects with situation reports. The procedure for location-subject association is illustrated in Figure 10. First, the document (situation reports) is processed by GATE system [6] and a list of locations and tokens are obtained. Then we find out the matched locations and the candidate subjects by comparing the extracted locations with our records stored in database. Considering the fact that the same location in different document may involve different subjects, each candidate subject should be checked in the document to verify their existence. However, different documents may use variant words for the same subject. Therefore the candidate subjects are extended by including their synonyms. Then all the candidate subjects as well as the synonyms are matched with the set of tokens extracted by GATE system to retrieve the matched subjects (synonyms). Finally, we format the matched location-subject pairs by converting the synonyms to the original subject names.

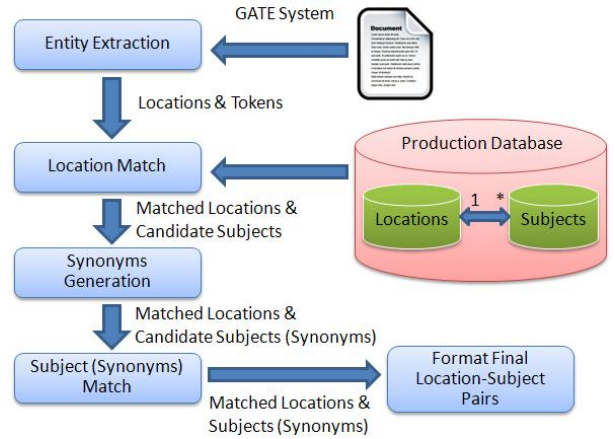


Fig. 6. Document Analysis

V. INCORPORATION OF USER FEEDBACK

Considering the user and application domain of the proposed framework, it would be extremely useful to incorporate domain knowledge and user interaction. One effective type of user interaction is user feedback. In this work, a system improvement mechanism based on user feedback is designed to refine the association between situation reports and multimedia data, i.e., images in current version. There are two feedback situations and four types of feedbacks in the current scenario. This section provides a detailed description of the feedback as well as its usage.

A. Feedback Description

There are two categories of user feedback: (1) feedback for image-document; and (2) feedback for related images.

The feedback for image-document indicates user impression for image-document relationships, i.e., whether a particular image matches the content of the document (situation report).

On the other hand, the feedback for related images implies the fondness of a particular image regarding the target image, which to some sense reflects the similarity (affinity) of image pairs. There are four types of feedback impressions described as follows: (1) no action: system made a correct match, no changes should be made; (2) thumbs up: system made a correct match, but some image(s) is/are more relevant than others; (3) thumbs down: system made a correct match, but some image(s) is/are less relevant than others; (4) flag: image is completely inappropriate, should be hidden from all future image lists.

The processing for flag feedback is the same for both the situations, i.e., hidden from all future image lists. The following two sections will discuss the processing of thumbs up/down feedback in both scenarios.

B. Feedback for Report-Image

The processing or image-document feedback is based on a rather simple counter mechanism. Specifically, a counter is created for each image belonging to certain situation reports and update the counter by increasing 1 (for thumbs up feedback) or decrease 1 (for thumbs down feedback). Then the image list is re-ranked for each report.

C. Feedback for Image-Image

The processing for related images feedback is based on the simplified MMM mechanism [8], which is used to model the searching and retrieval process for content-based image retrieval. It is different from the common image retrieval methods in that the MMM model carries out the searching and similarity computing process dynamically, taking into consideration not only the image content features but also other characteristics of images such as their access frequencies and access patterns.

1) *Markov Model Mediator (MMM)*: MMM is a probability-based mechanism that adopts the Markov model framework and the mediator concept. The MMM mechanism models an image database by a 5-tuple $\lambda = (S, F, A, B, \pi)$, where S is a set of images called states; F is a set of distinct features of the images; A denotes the states transition probability distribution, where each entry (i, j) indicates the relationship between image i and j captured through the off-line training procedure; B is the feature matrix of all images; and π is the initial state probability distribution.

All the disaster images and their relationships in our system are modeled by an MMM, where S represents the whole image set. F is a set of distinct features of the images, i.e., 12-dimension color descriptor and 9-dimension location descriptor in the MADIS. A describes the relationships among all the images in the database based on user's feedback. B consists of the 21-dimension feature vectors for all the images. π indicates how likely an image would be accessed with any prior knowledge of user preference.

The training of MMM basically involves the construction of the two statistical matrices, A and π . A sequence of user feedback characterizing access patterns and access

frequencies is used to train the model parameters. Suppose $R = \{R_1, R_2, \dots, R_T\}$ is a collection of user feedback during a period of time, where each R_t ($t = 1 \dots T$) is a list of user feedback for an anchor image. Let $P_{m,t}$ denote the feedback pattern of image m with respect to collection item R_t per time period, where the value of $P_{m,t}$ is 1 when m appears in R_t and zero otherwise. The value of AC_t denotes the access frequency of R_t per time period. The pairs of user feedback pattern $P_{m,t}$ and access frequency AC_t provides the capability of capturing user preference. Specifically, the training of the two parameters, A and π , are described as follows:

- **Matrix A**: Intuitively, the more frequently two images are accessed together, the more closely related they are. In order to capture the relative affinity measurements among all the images, a matrix AF is constructed with each element $af_{m,n}$ denoting the relative affinity relationship between two images m and n :

$$af_{m,n} = \sum_{t=1}^T P_{m,t} \times P_{n,t} \times AC_t. \quad (5)$$

Each entry in the state transition probability distribution matrix (A) is obtained by normalizing AF per row as in

$$a_{m,n} = \frac{af_{m,n}}{\sum_{n \in d} af_{m,n}}. \quad (6)$$

- **Matrix π** : The preference of the initial states for user feedback can be obtained from the training data set. For any image m , the initial state probability is defined as the fraction of the number of occurrences of image m with respect to the total number of occurrences for all the images in the image database D from the training data set.

$$\pi_m = \frac{\sum_{t=1}^T P_{m,t}}{\sum_{l=1}^N \sum_{t=1}^T P_{l,t}}, \quad (7)$$

where N is the total number of images in database.

2) *Off-line Training Based on User Feedback*: Since the related image retrieval is within a specific subject which utilizes the classification results and implies the cooperation of domain knowledge, the MMM model is refined to keep just the affinity matrix to describe the relationship between image pairs. The challenge then becomes how to update the affinity matrix based on different types of user feedback. In this work, we propose to process positive feedback (thumbs up) and negative feedback (thumbs down) separately. Specifically, two affinity matrices are created for positive and negative feedback respectively, and they are summed up and normalized to form the final affinity matrix.

VI. SYSTEM EVALUATION

The evaluation of the proposed system was carried out from two perspectives. First, experiments are conducted to validate the effectiveness of the proposed dynamic hierarchical classification framework. Second, an evaluation activity is initiated at Miami-Dade Emergency Management (MDEM) department, where the personnel are asked to perform a set of

specific tasks using the developed application and then answer a series of questions based on their experience.

A. Algorithm Evaluation

In this section, two sets of experiments are designed for demonstrating the effectiveness of the dynamic weighting scheme and the MCA-based hierarchical classification model respectively.

1) *Real World Dataset*: Our experiments are based on a collection of disaster images with their corresponding texts downloaded from the Flickr website. The dataset covers six disaster-related subjects as follows: (1) building collapse; (2) Flooding; (3) human Relief; (4) earthquake damage; (5) damage to sea grass; (6) death of animals.

2) *Dynamic Weighting Scheme Evaluation*: To demonstrate the effectiveness of the dynamic weighting scheme for fusing the visual and text features, the performance of different models is compared, i.e., *visual model*, *text model*, and *visual-text model*. The precision, recall, F1, and accuracy [26] are calculated as the measurements of performance using 3-fold cross validation. As can be seen from Tables I through III, the average F1 score of the whole classification framework is 83.9%, which is about 13% and 6% more than the visual model and text model respectively; in addition, compared with the single-course model, the overall accuracy of visual-text model has also improved by 9% and 7% respectively. The promising results verify the significance of the proposed dynamic weighting algorithm, which effectively integrates different sources of information and enhances the performance of the whole framework.

TABLE I
PERFORMANCE OF VISUAL-BASED MODEL.

Subjects	Precision	Recall	F1	Accuracy
(1)	74.9%	67.9%	71.1%	71.2%
(1)	68.0%	62.2%	63.9%	74.8%
(3)	81.0%	81.0%	81.0%	77.4%
(4)	70.2%	64.5%	65.0%	75.7%
(5)	79.5%	70.8%	72.4%	77.8%
(6)	84.2%	63.4%	71.3%	72.6%
Average	76.3%	68.3%	70.8%	74.9%

TABLE II
PERFORMANCE OF TEXT-BASED MODEL.

Subjects	Precision	Recall	F1	Accuracy
(1)	79.8%	93.8%	86.1%	84.0%
(2)	67.9%	88.8%	74.3%	75.3%
(3)	60.8%	92.7%	73.4%	60.2%
(4)	43.5%	58.1%	49.7%	56.7%
(5)	94.5%	85.9%	90.0%	91.7%
(6)	92.9%	93.5%	93.2%	92.3%
Average	73.2%	85.5%	77.8%	76.7%

3) *MCA Model Evaluation*: On the other hand, to illustrate the efficacy of the hierarchical MCA Modeling mechanism, we first implement the above 3 classification models by LibSVM [27], one of the most popular classification tools, and then compare their classification performance with those of the MCA model, as shown in Figure 7. From the results,

TABLE III
PERFORMANCE OF VISUAL-TEXT MODEL.

Subjects	Precision	Recall	F1	Accuracy
(1)	83.6%	94.9%	88.9%	87.5%
(2)	81.8%	86.8%	83.6%	87.5%
(3)	69.5%	95.0%	80.0%	71.4%
(4)	60.4%	75.1%	66.1%	71.0%
(5)	97.1%	86.7%	91.6%	93.1%
(6)	94.9%	91.5%	93.0%	92.4%
Average	81.2%	88.3%	83.9%	83.8%

we have the following observations: (1) compared with the single text model and visual model, the classification results are improved using the dynamic visual-text information fusion method, which demonstrates the effectiveness of our proposed approach; (2) compared with the classification performance of LibSVM, our proposed MCA model outperforms the other on each type of features. The reason for the overall performance of the MCA model is better than that of the LibSVM is that the MCA model could effectively integrate text features and visual features by the dynamic weighting schema and hierarchical classification structure, consequently achieving better classification performance.

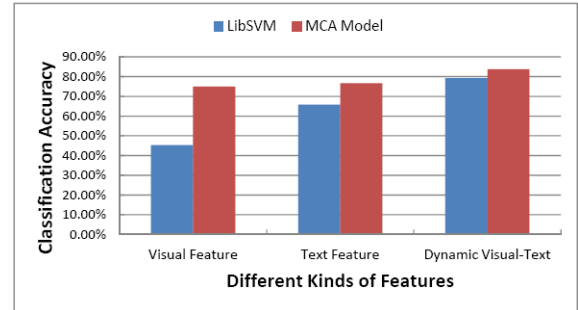


Fig. 7. Comparison between MCA and LibSVM

B. Application Evaluation

To validate the usability and performance of the proposed system, the EM personnel at MDEM department are requested to perform the following tasks and answer twelve questions, where ten of them are multiple choice questions with a 5-level agreement scale (Strongly Agree, Agree, Not Sure, Disagree, and Strongly Disagree) and the other two are open-ended questions.

The set of tasks includes (1) finding hurricane Katrina situation report; (2) reviewing associated images and select thumbs up/down or flag as needed; (3) filtering the images based on one of the locations/subjects; (4) viewing the description and timeline of one selected image; and (5) browsing the related images and select thumbs up/down or flagging as needed. **Some of the multiple choice questions are:** (1) I was able to locate the situation report I was interested in; (2) I found that the images are correctly associated with the report; (3) I was able to give feedback (thumbs up/down, flag) to the associated images; (4) I was able to filter the images based

on location/subject; and (5) I found that the system useful in enhancing the situation report for emergency management.

Several personnel at the MDEM department participated in the evaluation. It is worth noting that all the participants were new to the application and there was no training process involved. The evaluation results indicate that most of the personnel are satisfied with the performance of the system. Specifically, eight out of ten questions receive “Strongly Agree” or “Agree” from all of the participants, which implies a high level of satisfaction with the system performance.

Other feedbacks collected from the opening questions are summarized as follows, and some of them suggests our potential future work.

- **Positive feedbacks:** (1) the concept is extremely helpful and will be very useful for emergency managers; (2) the system is very friendly and easy to use; and (3) the abilities provided by the system is impressive, such as filtering by location and subject, association of reports with images, image timeline, pre-classification of images, and so on.
- **Suggestions:** (1) the disaster ontology could be extended for categorizing images; (2) extra functionalities such as group selection and de-selection of images are welcomed; and (3) labor intervene should be reduced to enhance automated function.

VII. SYSTEM OPERATION AND CONCLUSION

Florida International University (FIU) has spent over \$170K in the development and maintenance of the system, which is managed in a version control system and run through a test suite that validates key functionalities such as report list control, image filtering, feedback processing, and so on. By interacting with MDEM personnel through evaluation and exercise activities, the system has constantly been being updated by improving user interface experience and back-end support techniques.

Feedback from our collaborative partners at MDEM and the potential users suggests that our system will be very useful for emergency managers to gain insight of the situation at actual disaster scene and make quick response. We are encouraged to further develop the system into an operational pilot and promote the commercialization of the system for benefitting the whole EM community.

ACKNOWLEDGMENT

This work is partially supported by the U.S. Department of Homeland Security under grant Award Number 2010-ST-062-000039, U.S. Department of Homeland Security’s VACCINE Center under Award Number 2009-ST-061-CI0001, NSF HRD-0833093, and Army Research Office under grant number W911NF-10-1-0366 and W911NF-12-1-0431.

REFERENCES

- [1] L. Zheng, C. Shen, L. Tang, T. Li, S. Luis, S.-C. Chen, and V. Hristidis. Using data mining techniques to address critical information exchange needs in disaster affected public-private networks. In *ACM SIGKDD*, pages 125–134, 2010.
- [2] L. Zheng, C. Shen, L. Tang, T. Li, S. Luis, and S.-C. Chen. Applying data mining techniques to address disaster information management challenges on mobile devices. In *ACM SIGKDD*, pages 283–291, 2011.
- [3] Y. Yang, H.-Y. Ha, F. Fleites, S.-C. Chen, and S. Luis. Hierarchical disaster image classification for situation report enhancement. In *IEEE International Conference on Information Integration and Reuse*, pages 181–186, 2011.
- [4] S. Luis, F. Fleites, Y. Yang, H.-Y. Ha, and S.-C. Chen. A visual analytics multimedia mobile system for emergency response. In *IEEE International Symposium on Multimedia*, pages 337–338, 2011.
- [5] N. J. Salkind. *Encyclopedia of measurement and statistics*. Sage, Newbury Park, CA, 2006.
- [6] H. Cunningham, D. Maynard, K. Bontcheva, and V. Tablan. Gate: A framework and graphical development environment for robust nlp tools and applications. In *Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics*, Jul. 2002.
- [7] Princeton University. Wordnet, a lexical database for english. <http://wordnet.princeton.edu/>, Jul. 2011.
- [8] M.-L. Shyu, S.-C. Chen, M. Chen, and C. Zhang. Affinity relation discovery in image database clustering and content-based retrieval. In *Proceedings of ACM Multimedia*, pages 372–375, 2004.
- [9] L. R. Rabiner and B. H. Huang. An introduction to hidden markov models. *IEEE ASSP Magazine*, 3(1):4–16, 1986.
- [10] 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*, pages 418–421, Jul. 2009.
- [11] H. Abdi and D. Valentin. Multiple correspondence analysis. *Encyclopedia of measurement and statistics*, 2007.
- [12] K. Van de Sande, T. Gevers, and C. Snoek. Evaluation of color descriptors for object and scene recognition. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008.
- [13] J.C. Femiani and A. Razdan. Interval hsv: Extracting ink annotations. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 2520–2527, 2009.
- [14] S.-C. Chen, S. Sista, M.-L. Shyu, and R. L. Kashyap. An indexing and searching structure for multimedia database systems. In *SPIE Conference on Storage and Retrieval for Media Databases*, pages 262–270, 2000.
- [15] A.K. McCallum. MALLETT: A machine learning for language toolkit. <http://mallet.cs.umass.edu>, 2002.
- [16] G. Salton and M.J. McGill. Introduction to modern information retrieval. 1986.
- [17] J.V. Davis, B. Kulis, P. Jain, S. Sra, and I.S. Dhillon. Information-theoretic metric learning. In *ICML*, pages 209–216. ACM, 2007.
- [18] K.Q. Weinberger and G. Tesauero. Metric learning for kernel regression. In *Eleventh international conference on artificial intelligence and statistics*, pages 608–615, 2007.
- [19] B. Shao, M. Ogihara, D. Wang, and T. Li. Music recommendation based on acoustic features and user access patterns. *IEEE Transactions on Audio, Speech, and Language Processing*, 17(8):1602–1611, 2009.
- [20] P.E. Gill, W. Murray, and M.H. Wright. Practical optimization. 1981.
- [21] K. Barnard and D. Forsyth. Learning the semantics of words and pictures. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 408–415, 2001.
- [22] J. Fan, H. Luo, Y. Gao, and M.-S. Hacid. Mining image databases on semantics via statistical learning. In *ACM SIGKDD*, pages 22–31, 2005.
- [23] L. Fei-Fei and P. Perona. A bayesian hierarchical model for learning natural scene categories. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 524–531, 2005.
- [24] J. Fan, Y. Gao, H. Luo, and R. Jain. Mining multilevel image semantics via hierarchical classification. *IEEE Transactions on Multimedia*, 10(2):167–187, 2008.
- [25] B. Geng, L. Yang, C. Xu, and X.-S. Hua. Collaborative learning for image and video annotation. In *ACM SIGIR*, pages 443–450, 2008.
- [26] D.L. Olson and D. Delen. *Advanced data mining techniques*. Springer, USA, 2008.
- [27] C.C. Chang and C.J. Lin. LIBSVM: a library for support vector machines. 2001.