

Embedded Media Markers: Marks on Paper that Signify Associated Media

Qiong Liu, Chunyuan Liao, Lynn Wilcox, Anthony Dunnigan, Bee Liew

FX Palo Alto Laboratory

3400 Hillview Avenue, Bldg.4

{liu, liao, wilcox, tonyd, bee}@fxpal.com

ABSTRACT

Embedded Media Markers, or simply EMMs, are nearly transparent iconic marks printed on paper documents that signify the existence of media associated with that part of the document. EMMs also guide users' camera operations for media retrieval. Users take a picture of an EMM-signified document patch using a cell phone, and the media associated with the EMM-signified document location is displayed on the phone. Unlike bar codes, EMMs are nearly transparent and thus do not interfere with the document contents. Retrieval of media associated with an EMM is based on image local features of the captured EMM-signified document patch. This paper describes a technique for semi-automatically placing an EMM at a location in a document, in such a way that it encompasses sufficient identification features with minimal disturbance to the original document.

Author Keywords

Augmented paper, camera phone, vision-based paper interface, marker on paper, document recognition.

ACM Classification Keywords

Perceptual & Vision-based UIs, Tangible UIs, Handheld Devices and Mobile Computing.

INTRODUCTION

Although paper is one of the most widely used devices for viewing information, it cannot play dynamic media such as video and audio. On the other hand, cell phones are increasingly used to play audio and video but cannot match paper's high resolution, large display size, flexibility in spatial organization, outdoor-readability and robustness for static content. It is now possible to combine the two, using image recognition technology to link paper documents to corresponding dynamic media. A cell phone camera is used to capture an image of a document patch. The document

patch is identified using features in the image, and digital media linked to that location in the document is retrieved and then played on the cell phone.

A common method for creating this type of media link on a paper document is to print markers on the document. Examples are 2D bar codes [13] or printed grids of dots [5]. However, these markers are visually obtrusive and interfere with the document content layout. Microsoft Tag [22] alleviates this issue by merging data cells with the user-specified image background, but still requires an opaque black-white border for the decoder to locate the data cells. DataGlyphs [16] overcome these problems by printing a nearly invisible machine-recognizable pattern on the paper. However, this type of marker requires high resolution printers and cameras to identify document locations. Electronic markers like RFID can be used too [18], but this approach increases the production costs.

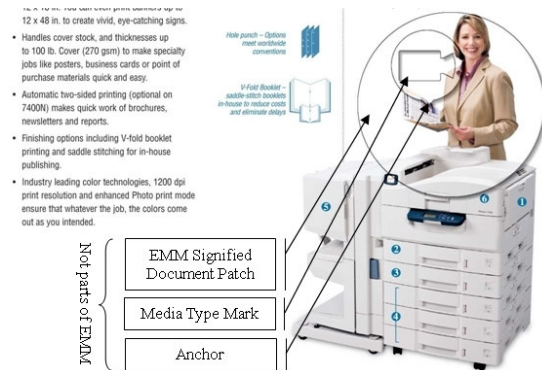


Figure 1. An EMM signifies that there is a video corresponding to the magazine in the person's hand. (callouts are not parts of an EMM)

Other systems compute features of the document content itself for identifying the document patch and thus creating a media link. HotPaper [2] and Mobile Retriever [9] use features based on document text such as the spatial layout of words. Other systems such as Bookmarkr [6] and MapSnapper [4] use pixel level image features, such as the SIFT [10] algorithm, to recognize generic document content such as pictures and graphic elements. With these systems, visually obtrusive marks are not required for identification.

Both marker-based methods and content-based methods lack media type indicators. Although bar codes and Data

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

CHI 2009, April 4–9, 2009, Boston, MA, USA.

Copyright 2009 ACM 978-1-60558-246-7/08/04...\$5.00

Glyphs are visible, they do not directly indicate the existence or type of media associated with them. When content-based feature are used, there is no on-paper indication at all to the user that there is media linked to the document. As a result, a HotPaper [2] user has to pan a camera phone over the paper document to look for hotspots until feedback such as a red dot or vibration is presented on the cell phone.

To solve this problem, we augment paper with meaningful awareness-marks, called Embedded Media Markers (EMMs) that indicate the existence, type, and capture guidance of media links. On seeing an EMM, the user knows to capture an image of the EMM-signified document patch with a cell phone in order to view associated digital media. This is analogous to Web pages that use underlines, font differences, or image tags to indicate the existence of links that users then click for additional information. Unlike barcodes, EMMs are nearly transparent and thus do not interfere with the document content. Unlike Embedded Data Glyphs [16] or Anoto patterns [17], EMMs can be printed with a regular low-resolution printer and identified from an image captured by a normal cell phone camera. The design of EMMs also indicates what type of media (e.g. audio, video, or image) is associated with the EMM-signified document location. Furthermore, by requiring the captured image to cover the whole mark, we can improve feature construction accuracy, matching accuracy, and efficient resource usage.

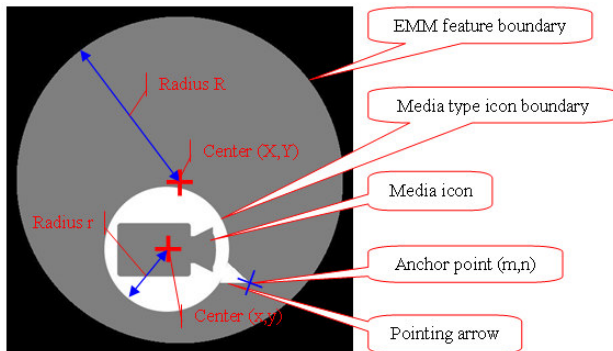


Figure 2. Basic components of an EMM

Figure 1 shows an EMM added in a printer description brochure. The EMM signifies that there is a video corresponding to the magazine in the person's hand. The portion of the document inside the big EMM feature boundary circle is named EMM-signified document patch (Figure 2). This patch should be completely included in a snapshot for successful retrieval. Within the EMM-signified document patch is the media type mark, a smaller boundary (a circle in this implementation) containing a graphic that indicates the type of media associated with this EMM, in this case video. The arrow connected to the smaller circle points to the exact location in the document that is associated with the media, and is called the anchor, or the EMM-signified document location (Figure 2).

In the next section, we explain the requirements for EMMs construction. We then describe the algorithm for semi-automatic arrangement of the EMM on the document. The EMM system and several applications are described, followed by an evaluation. We end with conclusions and discussions for future work.

EMM CONSTRUCTION

EMMs are marks on paper that signify the existence of media associated with that location. For usability purposes, EMMs should have the following properties:

1. EMMs should be visible to the human. An EMM should be a visible mark that signifies the existence of multimedia information associated with the document
2. EMMs should be meaningful to human. An EMM should indicate the type of media, e.g. audio, video, text, image, and annotation, that is associated with it.
3. EMMs should not take up extra space on the paper, nor should the document layout be changed to accommodate the EMMs.
4. EMMs should minimize the semantic change to original paper content. The design and implementation of the EMM should blend with the original document content.
5. EMMs should not significantly degrade the document patch. Since identification is mainly based on document content features, document contents must be well preserved.

Feature Boundary Mark

When barcodes are not available, features in the image can be used to identify the document patch. These features may be generic image features such as SIFT [10], PCA-SIFT [15], SURF [1], FIT [8] etc. They may also be features based on word center relations in a text patch [2], or stroke-center arrangements in a patch [11]. These features are derived from the local content of the document image and are distributed non-uniformly. Thus a feature boundary is needed to tell users what part of the document to capture. Without a clear feature boundary, users of a paper document enhancement system may capture a document patch without sufficient features for a system to identify the patch. To tackle this problem, we guide the user's capture with an artificial boundary. More specifically, we use this boundary to set the minimum capture region for patch identification. With the help of this capture region guidance, we can dramatically reduce indexed features in our database. This feature reduction is very helpful for improving identification accuracy and identification speed.

To guarantee sufficient features in different capturing angle (we assume the camera optical axis is perpendicular to the paper), we tentatively use a feature boundary circle, illustrated in Figure 2, as an artificial feature boundary. From paper patch identification aspect, the larger the circle

is, the more features we can use to facilitate the paper patch identification. On the other hand, the larger the circle is, the less EMMs we can put in every page; the larger the circle is, the less benefit we can get from feature reduction. Moreover large circle beyond certain size cannot be fully used by camera phones because of field-of-view and resolution limitations.

Besides circle size (radius R in Figure 2) consideration, the circle position (center (X,Y) in Figure 2) is also important for patch identification. If a circle is positioned at a place with dense feature distribution, the circle radius R can be greatly reduced while the identification accuracy is not compromised. On the other hand, a large circle placed at a blank location may lead to degraded recognition accuracy. The procedure and algorithm proposed here provides a method for finding an optimal circle center (X,Y) and radius R .

Media Type Mark

Besides cameraphone capture guidance, the media type (e.g. audio, video, weblink) information is also useful for readers of enhanced paper document. The EMM in Figure 2 has a video icon to graphically depict a video link associated with this EMM. For different media (e.g. audio) associated with an EMM, the media type icon is changed to reflect the associated media. The iconic information of media type is within a small interior circle. This circle is useful for emphasizing the iconic information. On the other hand, it is also visually consistent with the feature boundary mark.

A media type information mark must have a certain size to attract user's attention. On the other hand, this mark shouldn't obscure features for document patch identification. In this paper, we provide a method for finding an optimal iconic mark center (x,y) .

Anchor Point and Pointing Arrow

The relatively small circle in Figure 2 does not include any specific position on a paper. This makes it difficult to convey information to a user at an exact position in the document. For example, if there are several machine parts close to each other in a figure and we have 3D models for the parts, it is difficult for people to associate the circle surrounded iconic marks with their corresponding parts. To solve this problem, we allow the document creator to select a specific location as an anchor point (m,n) and add an arrow that points to the anchor point from the circle surrounded icon. The arrow's appearance exactly matches that of the circle it is associated with so that the arrow and the circle surrounded icon form a callout for the specific location (m,n) .

Graphical Effects

To reduce the impacts to the original document, we use skeletons instead of colored regions to represent media type icons. Since human are good at separating alpha-blended images, alpha-blending media-type-icons with the original

document can further reduce semantic impacts to the original document. By changing the alpha-blending coefficients for different color channels and regions, we can also change the text color or picture color in an EMM-signified region. Moreover, because graphical objects in a paper document are mainly in 2D space, adding 3D graphical effects to EMMs may further facilitate the separation of EMMs and other graphical objects in the original document.

In our current implementation, shown in Figure 2, all EMMs have a similar appearance: a large circle surrounding a smaller circle. The smaller circle features a simple icon and an attached arrow (Figure 2). This homogeneity in appearance is designed to allow users to find the EMMs on a page very quickly. The function of an EMM is similar to the simple colored and underlined hyperlink on a traditional web page. Unlike a hyperlink, an EMM is printed on physical paper and uses a meaningful icon to indicate available media associated with the EMM-signified document location.

In our current implementation, occlusion of images and text has been mitigated by adjusting only the luminance channel of the file used to print a page containing EMMs. The EMM varies the luminance of the underlying image in a limited range. Luminance information was chosen because it is reproduced with good fidelity by both color and monochrome copy machines and printers. In the implementation both the large bounding circle and the smaller circle add to the underlying image's luminance while a drop shadow effect diminishes it.

Image Feature Descriptor

Many image local features such as SIFT [10], PCA-SIFT [15], SURF [1], and FIT [8] can be used for EMM indexing. We use FIT[8] in our current system. FIT is an image local feature descriptor related to the well known SIFT[10] descriptor. Similar to SIFT, FIT finds keypoints (locations for feature computation) based on extremes in the Difference of Gaussian (DOG) image pyramid. Unlike SIFT which accumulates histograms of Gaussian weighted gradients at a key point level, the FIT descriptor directly computes its features at multiple scales higher than the key point scale. This approach can greatly reduce the number of image-pixel-operations involved in feature extraction. Moreover, FIT uses the pre-computed pyramid to save computational cost on the expensive Gaussian weighting process [8]. Through early comparison between SIFT and FIT, FIT was reported to have comparable recognition accuracy to SIFT on document recognition task. On the other hand, FIT uses less than 1/3 of SIFT's storage space. Its construction time and search time are also much shorter than SIFT.

EMM PLACEMENT PROCEDURE AND ALGORITHM

The EMM construction mainly focuses on readers' requirements for multimedia enhanced paper. We also need to consider the requirements from the identification algorithm. More specifically, we want to arrange an EMM

to improve instead of degrading a paper patch identification process. To achieve this goal, an algorithm is needed to adjust parameters for an EMM arrangement.

EMM Placement Rules and Procedures

There are three basic sets of parameters for an EMM adjustment: feature-boundary-circle center (X,Y) , feature-boundary-circle radius R , and media-type-circle center (x,y) . Since the patch identification accuracy does not change much if the number of patch-covered feature points is over a certain threshold, the boundary-circle optimization goal is to achieve good patch-identification accuracy with minimum cost of paper surface area. Small paper surface occupation by an EMM has following benefits to our system:

- 1) The EMM will create less distortion in original content.
- 2) Small surface area occupation makes it easier to put more separated EMMs in every page.
- 3) Since we only need to index the keypoints, such as SIFT[10]/SURF[1]/FIT[8] keypoints, in the circle for patch identification, small surface area occupation reduces the number of keypoints saved on the index server. The reduced number of keypoints is helpful for increasing identification accuracy and identification speed.
- 4) The small marked area may guide a reader to submit a patch without too much computation (the number of keypoints is controlled by the circle), and require less time for retrieving multimedia data.
- 5) Keeping the capture-area-low-limit small also makes it more convenient for cell phone capture. More specifically, capturing a large circle needs a large distance between the cell phone and paper while capturing a small circle is more flexible in a large dynamic range.

To get a small radius for the feature-boundary-circle, our algorithm will locate the boundary-circle at a place with high keypoint density and shrink the radius to meet the minimum keypoint number requirement.

With paper patch identification algorithms described in the previous sections, the feature-boundary-circle is normally much larger than the minimum visibility requirement. In our algorithm, we set the circle surrounding the media-type-icon inside the feature-boundary-circle to make sure readers consider them as one EMM. To reduce the disturbances caused by the icon-circle, our algorithm tries to move this circle to a place with minimum keypoint density. More specifically, with the icon-circle size fixed, we expect the circle to include a minimum number of keypoints in the original document. This procedure has the following benefits for the patch identification process.

- 1) The media-type-icon and its surrounding circle create less distortion to important features. We think this is

helpful for a reader to separate the original content and the icon so that the reader can have a better understanding to the original document.

- 2) Since the media-type-callout is mainly formed with semi-transparent skeletons and shadows, the callout and original document content may form more distinctive keypoints in sparse-keypoint regions. This keypoint density balancing process is useful for removing matching outliers and further improving patch identification accuracy.

Besides these basic optimization approaches, we also set some rules to make an EMM look more attractive and more consistent in various situations. These rules include:

- 1) The anchor point should be inside the feature-boundary-circle. It makes the EMM more intuitive. It reduces our circle searching space. Moreover, it avoids merging multiple EMMs on the same page to the same location (a global optimal position).
- 2) The media-type-icon and its surrounding circle shouldn't cover the anchor point. In this way, we can always have the arrow pointing outside from the circle and that gives EMMs more consistent look at various locations.
- 3) To make the arrow shorter, the algorithm forces the iconic callout to move closer to the anchor point.
- 4) All EMMs uses the same lighting source at infinite distance for their shadows.
- 5) In our current implementation, we assume an EMM model has a three-level construction in a 3D space for shadows. This construction model is illustrated in Figure 5.

In our current design, the light for shadows comes from up-left at an infinite distance.

A Fast Algorithm for Estimating the Number of Points in a Circle

To get an optimal location and size for a feature-boundary-circle or an optimal location for an iconic callout circle, the system has to count the number of keypoints inside a circle. If an EMM changes the image local features dramatically, the system has to re-compute all features in an EMM-signified patch when a set of new parameters (i.e. location and size) is tested. This kind of procedure will make it difficult to get an optimal parameter set in reasonable time. Since our EMM mainly includes edges and shadows, we believe that adding an EMM in a document patch will not decrease the number of keypoints much. Since an EMM edge can form new features with original contents close to the edge and an EMM transparent region will not have much impact to the original features, it is more probable that an EMM will increase the number of keypoints in its local region. With this consideration in mind, we can safely use keypoint distribution in a page without an EMM to

estimate the number of real features in an EMM feature boundary circle or an EMM media-type-icon boundary.

Even though our system can skip feature re-computation for testing each set of EMM parameters, the system still has to count the number of keypoints inside a circle with many different circle parameters. More specifically, the system may need to try the position of every pixel as a circle center. Moreover, it may also try multiple radiuses before an optimal solution is reached. Therefore, the algorithm for estimating the number of points in a circle has to be fast for a practical application.

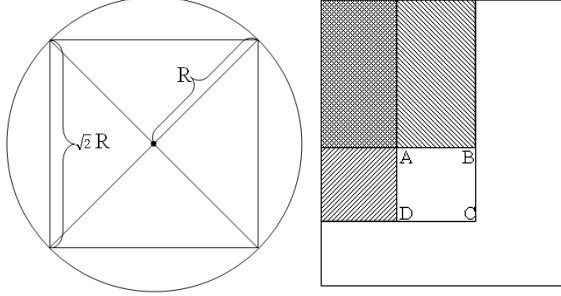


Figure 3. (left) Use the number of keypoints in the square to approximate the number of keypoints in the circle. (right) Use a cumulative keypoint distribution to compute the number of keypoints in a square.

The number of keypoints, n , of a normally scanned 100DPI page may reach several thousands. Assume we use a brute-force searching approach, the computational complexity for estimating the number of keypoints in a circle will be $O(n)$. For example, if we want to search an optimal circle center in a 500 by 500 patch and the number of keypoints in the whole page is 5000, the number of math operations used by the system will be over 10^9 . If we try multiple circle radiuses and consider the computation for the icon circle, the number of math operations may reach 10^{10} or more.

To overcome this computational complexity, we designed a fast algorithm for estimating the number of points in a circle. The algorithm is based on the integral image of a keypoint distribution histogram. To use this algorithm, we approximate the number of keypoints in a circle with the number of keypoints in a square inside the circle. The relationship between the circle and the square is illustrated in Figure 3 (left). Denote N_c as the number of keypoints in the circle and N_s as the number of keypoints in the square, we can get $N_s \leq N_c$. This approximation can guarantee enough keypoints in the circle for patch identification when the number of keypoints in the square reaches the patch identification low limit.

Figure 4 (left) shows keypoints overlaid on top of a brochure image. Corresponding to the image in Figure 4 (left), Figure 4 (right) shows the cumulative keypoint distribution map. In the cumulative keypoint distribution map, the value at each point equal to the number of keypoints in its top-left region. Assume the number of pixels in an image is N , the computational complexity for

getting this cumulative keypoint distribution map is $O(N)$. Since the algorithm only needs to compute this map once and the system can pre-compute this map for each image, the computational complexity of this map will not affect the optimization much when a document creator uses this approach to get an optimal EMM arrangement.

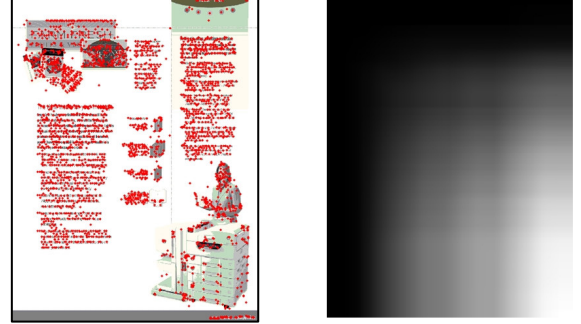


Figure 4. (left) keypoints overlaid on top of a brochure image, and (right) cumulative keypoint distribution of the same page.

With the cumulative keypoint distribution map, the system can compute the number of keypoints in a square in constant time. Assume a square, ABCD in Figure 3 (right), has its sides parallel to one image boundary or the other and the values for the points A, B, C, and D on the cumulative distribution map are N_A , N_B , N_C , N_D respectively. The system can get the number of keypoints in the square ABCD, N_{SQ} , with the following equation:

$$N_{SQ} = N_A + N_C - N_B - N_D$$

From this equation, the system can get the number of keypoints in a square with one addition and two subtractions. This is much more efficient than the brute force approach.

Optimal Feature-boundary-circle

With the fast algorithm for estimating the number of keypoints in a circle, the system can try a circle center at the position of every pixel. Moreover, the system also needs to know the optimal radius for the best feature-boundary-circle. To get the optimal radius, we used the following binary search approach for optimization:

```
while ((radiushigh-radiuslow)>SMALLMARGIN)
```

```
    Get a circle center location that allows the circle to
    include the anchor point and the maximum number of
    keypoints with this radius;
```

```
    if maximum number of keypoints with this radius >
    KEYNUMLOWLIMIT
```

```
        radiushigh = currentradius;
```

```
    else
```

```
        radiuslow = currentradius;
```

```
    end
```

```
    currentradius = (radiushigh + radiuslow)/2;
```

```
end
```


Optimal surrounding circle for an media-type-icon

Since a media-type-icon has a fixed size, getting the optimal location of this circle is a circle location that allows the circle to include minimum number of keypoints. Besides this optimization, we also need to consider rules in the previous paragraphs to make an EMM look nicer. In other words, the distance between the surrounding circle center and the anchor point should be larger than the radius of the surrounding circle. Moreover, the surrounding circle should be 'close' to the anchor point for a short pointing arrow. There are several ways to make the arrow short. One way is to set the maximum distance between the anchor point and the surrounding circle center. Another approach is to compute a vector from (X,Y) to (x,y) and a vector from (X,Y) to (m,n) , and force the angle between these two vectors smaller than 90° . We take the second approach in our current implementation.

Generate graphical effect for EMM

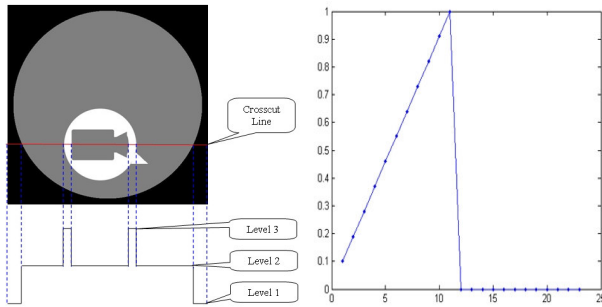


Figure 5. (left) A three-level EMM construction in a 3D space. (right) The filter diagonal coefficient shape for shadow generation.

Because graphical objects in a paper document are mainly in 2D space, adding 3D graphical effects to EMMs may further facilitate the separation of EMMs from other graphical objects in original document. In our current implementation, we use a three-level EMM model in 3D space for shadows. The three-level EMM model is illustrated in Figure 5 (left). For shadows, we assume the light comes from the top-left of a document. To mimic this effect, we used a sobel filter followed by a 2D filter. The sobel filter has coefficients $[1\ 0; 0\ -1]$. The 2D filter has the shape shown in Figure 5 (right) in its diagonal direction, and has zero coefficients for all other filter positions.



Figure 6. Generate graphical effect for an EMM. (a) A black and white multimedia icon. (b) An EMM decorated with graphical effects.

By combining the filtering result with an EMM edge, the system can generate proper EMM graphical effects based on each black and white multimedia icon. Figure 6 showed a typical black and white multimedia icon and an EMM

decorated with graphical effects. Because the arrow is dynamically added to the 3-level EMM model after the optimal EMM parameter set is found, the arrow's graphical effects are properly blended with graphical effects for other EMM parts. From Figure 1, readers may see the effect of alpha blending an EMM with a document page.

THE EMM SYSTEM AND APPLICATIONS

To demonstrate the feasibility and applications of EMMs, we have built a system to support the complete workflow for creating and using EMMs. As illustrated in Figure 7, the system consists of three major interaction entities, namely Authoring Client, EMM Server and Retrieving Client.

The EMM Authoring Client

The authoring client is a PC application for authors to add EMMs to a document with the previously presented algorithm. With this tool, a document editor can open a file, specify an anchor point on the document page via a mouse-click, type the URL of associated media, and get an EMM overlaid image. This tool also allows users to change the media type mark, adjust the EMM alpha-blending coefficient, adjust the document DPI (dots per inch), or view keypoint distribution on the image. Currently, our system supports five types of media links: audio, video, text, webpage, and image.

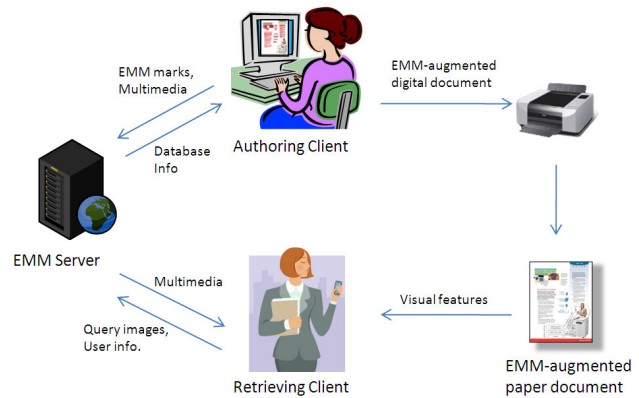


Figure 7. The architecture of the EMM system

The generated EMMs and their associated multimedia URLs are uploaded to the EMM server, which indexes the marks in a database and stores the link information. The authoring client might retrieve information from the server, e.g. the existing features in the database, for verification.

The resulting EMM-augmented digital document can then be printed and delivered to end users, who run a retrieving client application on a camera phone. With the application, an end user takes a snapshot of an EMM on paper to capture its visual features. The snapshot and the user information are then submitted to the EMM server to fetch the associated multimedia. Upon successful retrieval, the user can review and interact with the multimedia data. Based on the system, we have developed several applications in the areas of maps, manuals and advertisements.

Augmented Maps

A paper map offers a large high-quality display of geographic information. It is outdoor readable, foldable, and easy to share with other people [4, 21]. However, it lacks dynamic location-specific information, such as video clip about a restaurant, weather forecast for a park, and currently available discounts for a hotel.



Figure 8. (left) A map augmented with a video EMM (right) The associated YouTube video

With EMMs, such dynamic multimedia can be easily brought to paper maps. As illustrated in Figure 8, on a map of Tokyo in Japan, a famous restaurant Ginza Sky Lounge is augmented by an EMM, which points to a YouTube video clip about the history and highlights of the place. Similarly, the EMM may point to a customer review site or the detailed menus, and allow the user to make a reservation right away. Note that the EMM does not change the original map layout and has minimal appearance interference, which differentiates EMMs from the existing barcode-based techniques [13, 14]. Besides, overlaying an in-place EMM right on top of a POI (point of interest) is especially useful to retain the context of the POI in the map.

In-situ Multimedia Manuals



Figure 9. (left) An EMM in a manual for cartridge installation. (right) The associated step-by-step video tutorial

Due to paper's high display quality, robustness, instant accessibility, and spatial flexibility, printed manuals are extensively used by in-the-field workers such as automobile mechanics and airplane electricians, who often bring to the working space the manual as quick references for specific parts. This in-situ use of manuals is also quite common with average customers who are learning to operate appliances or perform maintenance, say replacing cartridge of a printer. The paper manuals, however, often falls short of presenting intuitive instructions (e.g. a video clip demonstrating the cartridge installation steps) or up-to-date

information about the parts and devices (e.g. the availability of a type of tires at local stores). And a computer is not always accessible in the working spaces.

An EMM-enhanced manual and a mobile phone can help address the dilemma. Figure 9 exemplifies such a printer manual, with which a user can easily retrieve and play the video tutorial of cartridge inspection. If a new cartridge is necessary, the user can also immediately place an order through the web site associated with an EMM.

Multimedia Advertisement on Paper

EMMs open the door to a wide range of interactive paper applications. Besides their use in maps and manuals, EMMs can also be applied to multimedia advertisements (Ads) on paper bills, flyers or catalogs. EMMs not only bring to paper more expressive media for Ads (e.g. video, audio and animations), but also allow users to proactively interact with the Ads and start the shopping workflow immediately. For example, after watching the associated video of a cash bonus promotion for opening a checking account, the user can activate the enrollment by simply clicking the video (Figure 10). Moreover, the Ads can be customized based on the user's personal information, including preferences, shopping history and events (e.g. birthdays and anniversaries). EMMs can be also used to augment textbooks, kids' story books, calendars, magazines etc.

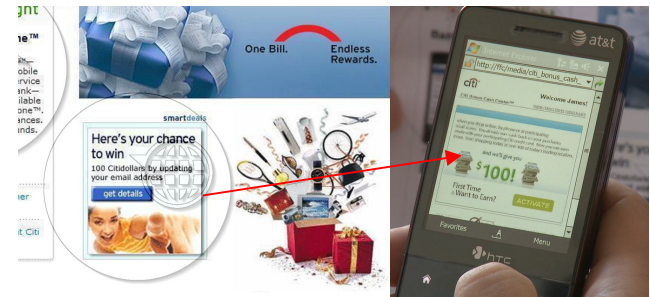


Figure 10 (left) An EMM-augmented Ad on a bill. (right) The personalized web page for the user to interact with the Ad.

EMM EVALUATION

We conducted an early stage performance evaluation of the EMM design and the semi-automatic EMM construction algorithm. In this evaluation, we use the ICME06 proceeding as our target document. The proceeding has 2188 letter-size (8.5"x11") document pages with text, images, and figures.

The feature-boundary-circle of an EMM is used to guide a user's capture of the EMM image. We assume that EMMs are normally used in the way shown in Figure 11. In other words, when a user captures an EMM, the user roughly has the EMM feature-boundary-circle filled the captured image. With this assumption, we may compute the features within the neighborhood of the feature-boundary-circle instead of all document features for EMM retrieval. This simplification can save us memory, disk space and

Batch # Setup	1	2	3	4	5	6	7	8	9	10
Printer	BW	BW	BW	BW	Color	Color	Color	Color	Color	Color
Capture Environment	Office	Patio	Office	Patio	Office	Patio	Office	Patio	Office	Patio
Rendering Software	Irfan-View	Irfan-View	Printing Wizard	Printing Wizard	Irfan-View	Irfan-View	Printing Wizard	Printing Wizard	Printing Wizard	Printing Wizard
Pages / Paper	1	1	1	1	1	1	1	1	4	4
Query #	20	10	10	10	10	10	10	10	10	10
Correct Identification #	20	10	10	10	10	10	10	10	10	9

Table 1. EMM testing results for different setups.

computation time. For each EMM, we crop a square region that includes the EMM feature-boundary-circle for visual feature indexing. Because most cameras have 4:3 aspect ratio and an EMM needs 6% more space for shadows, we set the square side $1.06 \times 4/3 \approx 1.42$ times the EMM feature-boundary-circle diameter. We also align the center of the square with the center of the feature-boundary-circle for consistent margin around the circle.



Figure 11. A typical EMM usage scenario.

A main purpose of an EMM is to signify the existence of media associated with that part of the document. For that purpose, an EMM must occupy a sufficiently large region on a page to attract user's attention. We set the diameter size of the EMM inner circle to 1" for clear indication of the media type. Moreover, the minimum diameter size of an EMM feature-boundary-circle is set to 2.5". Based on our experience, this minimum size or a larger size is easy for a user to focus a camera phone on a marker and fill the captured image with a small neighborhood of the marker. For example, when we use an AT&T FUZE to capture a 2.5" EMM, the distance between the camera phone and the paper is about 4". This is a little bit above the shortest recommended distance (about 3") that a FUZE camera can focus on an object. If we reduce this distance further, the FUZE camera phone will show a red 'x' on its screen. We also tested the Google G1 phone. Its shortest distance for auto focus is also around 3". Therefore, we think that the 2.5" diameter setup is reasonable for existing technology.

Another important factor for practical applications of the EMM is its recognition accuracy. EMM designers are responsible for reasonable retrieval accuracy and speed. On the other hand, designers should also control the memory consumption on the server for computation and cost savings. In our EMM construction algorithm, we use the minimum number of keypoints within the feature-boundary-circle to address this issue. In our experiment, we set this minimum number to 100.

Since most camera phones have the 640-by-480 resolution, we assume the captured EMM image has this resolution. To efficiently use captured EMM images at this resolution, we use similar resolution images in our database. With this setup, our database has all needed scales for feature matching with the captured image. On the other hand, we do not need to save features on unreachable scales for mismatches, or computation time/memory space cost. By combining this consideration with the square crop consideration in a previous paragraph, we use 200-DPI (dot per inch) resolution to sample each document page. With this sampling rate, each document page is converted to a 2200 by 1700 image. Corresponding to each 2.5" diameter EMM, the square-crop-size for visual feature computation will be 707 pixel by 707 pixel in our database. If the EMM diameter size is larger than 2.5", we currently increase the square size proportionally. If memory space or computation cost really becomes tight, we may consider down-sampling larger-images to 707-by-707 size. For space saving, all images in our database are saved in JPEG format.

In practice, EMM overlaid documents may be rendered with different printing software, and output to different printers. People may use EMMs in indoor environments or outdoor environments. To mimic these application scenarios, we randomly generate 2188 2D locations for 2188 document pages. By using these 2188 locations as EMM anchor points, our EMM construction algorithm creates 2188 non-overlapping EMMs and computes FIT features in all EMM-neighborhood squares. We randomly selected 110 EMM overlaid pages for testing. These testing

pages were saved in JPEG format with 1700 by 2200 resolution. For image rendering, we tried the IrfanView and the Windows XP Photo Printing Wizard. In these 110 selected pages, we print 90 pages on letter-size paper, and fit 4 pages on each letter-size paper for all other 20 pages. By printing these pages with two quite different scales, we can check the EMM sensitivity to paper size. Because the EMM sizes on these 4-page/paper hardcopies are relatively small, filling the captured image with an EMM neighborhood is a little difficult. Our AT&T FUZE camera showed the red 'x' for all these captures at the camera focus stage. For data collection, we forced these captures on our camera phone. The 110 hardcopies were generated with either Xerox Workcentre 255 black-and-white printer or Xerox Docucolor 240 color printer. On the black-and-white printer generated hardcopies, we can clearly see halftone effects. The 110 testing query images were captured either in an office or on an open patio. By capturing EMMs in these different environments, we can learn the EMM sensitivity to normal indoor and outdoor lightings. Because we assumed that EMMs are used in normal reading conditions, we did not capture EMMs in extreme lighting conditions for this test. The EMM testing results are listed in Table 1.

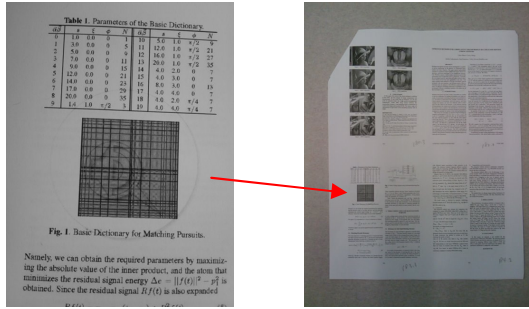


Figure 12. (left) The EMM capture that is failed to find its corresponding link. (right) The hardcopy used for this capture.

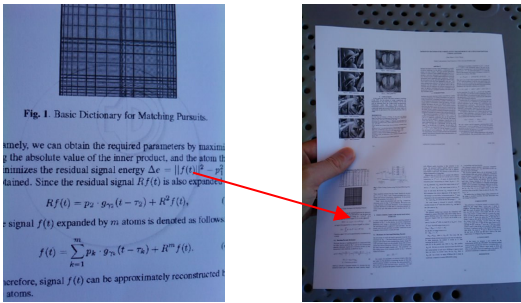


Figure 13. (left) The capture of an adjusted EMM. (right) The hardcopy used for this capture.

In the table, we use 'BW' to indicate the Xerox Workcentre 255 black-and-white printer and use 'Color' to indicate the Xerox Docucolor 240 color printer. In all these 110 EMM query images, the system failed on only one image. This image is shown in Figure 12. According to the log of our code, the failure is caused by insufficient matching points. In several million keypoints saved in our database, this

image input only finds two matching points. To handle similar problems in practice, we may consider increasing the EMM diameter size or forcing the EMM feature-boundary-circle center to a different location for EMM rearrangement. We tried the second approach in our experiment. In this experiment, we kept the anchor point unchanged and adjusted the EMM center location for the failed document. Our EMM arrangement algorithm re-computed all other EMM parameters, such as the feature-boundary-circle diameter, and the media type mark location, and regenerated the EMM overlaid page (Figure 13). After that, we captured the regenerated EMM under similar conditions of the previous capture, and the captured EMM was correctly recognized. For future research, we will explore the causes of insufficient matches in this case.

The FIT paper tested the algorithm by randomly rotating and scaling training-images in a certain range. The test dataset was the ICME2006 proceeding, and recognition test was on the page-level. In the test presented in this paper, we used the same proceeding. However, we used real camera captures instead of synthetic images. Moreover, the EMM recognition is performed on patch-level instead page-level. People may expect much lower recognition rate for our test. However, after the EMM adjustment, our document recognition result is even better than the result presented in the FIT paper [8]. We think this good result is caused by following reasons:

1. Even though the real EMM patch is smaller than a physical page, the indexing image we used in this experiment has a 707x707 size instead of a 306x396 size, which was used in the previous experiment in [8]. This image size increase gives the system more details for matching.
2. Because the EMM feature-boundary-circle guidance, we roughly know the image matching scales. With this knowledge, we prepared indexing images at similar resolutions of the captured images. This strategy can help us to reduce the feature-construction interpolation error, the matching noise caused by irrelevant scales.

In an early experiment which allows novice users to interact with paper via an unguided (without EMM) 320x240 image capture, the first snapshot success rate is about 81.6% (return nothing with less than six matching points). The result we get from the EMM experiment is much improved over that early experiment. We think this improvement is also caused by increasing the indexing image size, the query image size, and providing EMM capture guidance.

CONCLUSION AND FUTURE WORK

This paper describes the use of Embedded Media Markers printed on paper to signify the existence of media associated with documents. To facilitate the use of EMMs, we introduce a procedure and algorithm to semi-automatically arrange visible EMMs on paper. The procedure and algorithm is for paper enhancement systems that use natural content features on paper to recognize

document patches. More specifically, we can use this procedure and algorithm as the base of an authoring tool, with which a media editor can select an anchor point on a page of paper for adding multimedia information and the algorithm can automatically arrange EMMs based on the editor-selected anchor point. The algorithm can help editors to build quality EMMs that are user friendly, machine friendly, and document friendly. The EMMs can then be overlaid on other document contents for printout of enhanced paper. A user can then get extra multimedia information on an active screen by capturing an EMM-signified document patch on paper.

Besides the EMM idea and the semi-automatic EMM construction tool, we also discussed potential EMM applications and evaluated the EMM performance. Through performance evaluation experiments, we find that EMMs are not sensitive to the JPEG storage format. We also found that EMMs are not sensitive to the Xerox Workcentre 255 black-and-white printer and the Xerox Docucolor 240 color printer. Moreover, EMMs are not very sensitive to paper size, indoor/outdoor ambient lighting, and two popular printing software packages. We also found a failed case during these experiments. For this case, we proposed a practical solution and identified an interesting future research topic.

Additionally, we also want to modify the EMM authoring tool so that the author can always override the automatic placement for artistic design or other reasons, such as the failure case we experienced in our experiments. We also want the authoring client to retrieve information from the server, e.g. the existing features in the database, so that it can verify the uniqueness of a mark that is being placed in the current working document. Finally, studying semi-automatically arranging irregular-shape EMMs is another interesting research topic.

REFERENCES

1. Bay, H., Ess, A., Tuytelaars, T., Van Gool, L. "SURF: Speeded Up Robust Features", Computer Vision and Image Understanding (CVIU), Vol. 110, No. 3, pp. 346-359, 2008.
2. Erol, B., Emilio Antunez, and J.J. Hull. HOTPAPER: multimedia interaction with paper using mobile phones. Proceedings of Multimedia'08, pp. 399-408.
3. Pixazza, Now a picture is worth more than a thousand words. <http://pixazza.com/>.
4. Hare, J., P. Lewis, L. Gordon, and G. Hart. MapSnapper: Engineering an Efficient Algorithm for Matching Images of Maps from Mobile Phones. Proceedings of Multimedia Content Access: Algorithms and Systems II pp.
5. Hecht, B., M. Rohs, J. Schöning, and A. Krüger. Wikeye – Using Magic Lenses to Explore Georeferenced Wikipedia Content. Proceedings of the 3rd International Workshop on Pervasive Mobile Interaction Devices (PERMID), pp.
6. Henze, N. and S. Boll. Snap and share your photobooks. Proceedings of Multimedia'08, pp. 409-418.
7. Hull, J.J., B. Erol, J. Graham, Q. Ke, H. Kishi, J. Moraleda, and D.G.V. Olst. Paper-based Augmented Reality. Proceedings of Int. Conf. on Artificial Reality and Telexistence, pp. 205-209.
8. Liu, Q., H. Yano, D. Kimber, C. Liao, and L. Wilcox. High Accuracy and Language Independent Document Retrieval With A Fast Invariant Transform. Proceedings of ICME'09, pp.
9. Liu, X. and D. Doermann, Mobile Retriever: access to digital documents from their physical source. Int. J. Doc. Anal. Recognit., 2008. 11(1): p. 19-27.
10. Lowe, D.G., Distinctive Image Features from Scale-Invariant Keypoints. Int. J. Comput. Vision, 2004. 60(2): p. 91-110.
11. Nakia, T., K. Kise, and M. Iwamura. Use of affine invariants in locally likely arrangement hashing for camera-based document image retrieval. Proceedings of 7th Int'l Workshop DAS'06, pp. 541-552.
12. Parikh, T.S., P. Javid, S. K., K. Ghosh, and K. Toyama. Mobile phones and paper documents: evaluating a new approach for capturing microfinance data in rural India. Proceedings of CHI'06, pp. 551-560.
13. Rekimoto, J. and Ayatsuka, Y. 2000. CyberCode: designing augmented reality environments with visual tags. In *Proceedings of DARE 2000 on Designing Augmented Reality Environments* (Elsinore, Denmark). DARE '00. ACM, New York, NY, pp. 1-10.
14. Rohs, M. Real-world interaction with camera-phones. Proceedings of UCS. IPSJ Press (2004).
15. Ke, Y. and Sukthankar, R., PCA-SIFT: A More Distinctive Representation for Local Image Descriptors. *Computer Vision and Pattern Recognition*, 2004.
16. Hecht D. L., Embedded Data Glyph Technology for Hardcopy Digital Documents. SPIE -Color Hard Copy and Graphics Arts III, Vol. 2171. Feb 1994, pp341-352.
17. Wikipedia, Digital paper. http://en.wikipedia.org/wiki/Digital_paper
18. Reilly, D., M. Rodgers, R. Argue, et al., Marked-up maps: combining paper maps and electronic information resources. Personal and Ubiquitous Computing, 2006. 10(4): p. 215-226.
19. Costanza, E., J. Huang., Designable Visual Markers, Proceedings of CHI'09, pp 1879-1888.
20. NODA TSUGIO, MOROO JUN, CHIBA HIROTAKA, Print-type Steganography Technology, <http://sciencelinks.jp/j-east/article/200612/000020061206A0425989.php>.
21. Morrison, A., A. Oulasvirta, P. Peltonen, S. Lemmela, G. Jacucci, G. Reitmayr, J. N., #228, s\, nen, and A. Juustila, Like bees around the hive: a comparative study of a mobile augmented reality map, in CHI'09. 2009, ACM: Boston, MA, USA. p. 1889-1898.
22. Microsoft Tag. <http://www.microsoft.com/tag/>