1. 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 to a degree, but still requires visually intrusive changes on the document with an opaque black-white code locator. 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.

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. 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.

Figure 1. Partial image of an EMM enhanced paper page. In this EMM usage example, the circle shape EMM printed on paper signifies that there is a video corresponding to the location the child is looking at. (the box shape callouts are not parts of an EMM)

Both marker-based methods and document-appearance-based methods fall short in providing visual guidance for users. Although bar codes and Data Glyphs are visible, they do not directly indicate the existence or type of media associated with them. When appearance-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 appearance. 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. Unlike other appearance-based approaches, EMMs clearly indicate signified document patches and locations. 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 efficiency.

Figure 1 shows an EMM added to the Fuji-Xerox Sustainability Report cover page. The EMM signifies that there is a video corresponding to the location the child looks at. The portion of the document inside the big EMM feature boundary circle is named EMM-signified document patch. 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.

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, followed by the EMM system and several applications. We end the paper with an EMM identification evaluation.

2. EMM CONSTRUCTION

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

1. EMMs should be visible to the human.
2. EMMs should be meaningful to human.
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.
5. EMMs should not significantly change the document patch appearance, as identification is mainly based on the visual features of document appearance, which must be well preserved.

2.1 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 [2] or stroke-center arrangements in a text patch [11]. These features are derived from the local appearance 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 an augmented paper system may capture a document patch without sufficient features for the system to identify the patch. To solve 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 the accuracy and speed of an EMM identification.

To guarantee sufficient features in different capturing directions (we assume the camera optical axis is perpendicular to the paper), we chose to use a feature boundary circle, illustrated in Figure 2, as an artificial feature boundary. Other shapes are possible, but we currently focus on circles for simplicity. From document patch identification aspect, the larger the circle is, the more features we can use to facilitate the document patch identification. On the other hand, the larger the circle is, the less EMMs we can put in every page and the less benefit we can get from feature reduction. Moreover overly large circles cannot be fully used by camera phones because of field-of-view and resolution limitations.

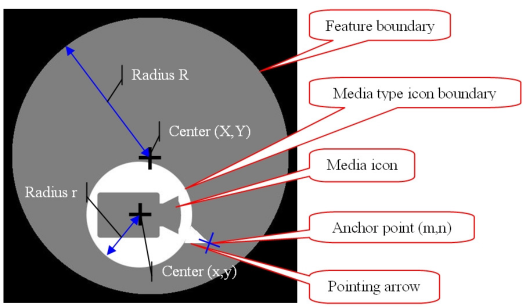

Figure 2.Basic components of an EMM

Besides circle size (radius R in Figure 2), 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.

2.2 Media Type Mark

Besides cameraphone capture guidance, the media type (e.g. audio, video, weblink, 3D model, or company mark etc.) information is also useful for readers of enhanced paper document. For different media associated with an EMM, different media type icons are adopted to reflect the distinction. The EMM in Figure 2 has a video icon to graphically depict a video link associated with this EMM. The iconic information of media type is within a small fixed-size circle, which aims to emphasize the iconic information. In this paper, we provide a method for finding an optimal iconic mark center (x,y).

2.3 Anchor Point and Pointing Arrow

The relatively small media type mark is difficult to convey information to a user at an exact position in the document. 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.

2.4 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, with less than 1/3 of SIFT's storage space and much shorter construction and search time.

3. 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 document patch identification process. To achieve this goal, an algorithm is needed to adjust parameters for an EMM arrangement.

3.1 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 feature boundary circles have the following benefits to our system:

1) The EMM will create less distortion in the original document appearance.
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 query image identification accuracy and 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 size of the capture region 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 document 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 media-type-icon-circle 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.

3.2 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.

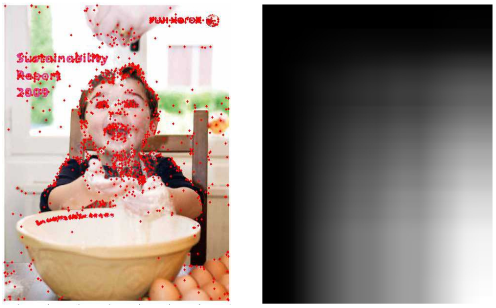

Figure 3. (left) keypoints overlaid on top of a brochure image, and (right) cumulative keypoint distribution of the same page.

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.

The number of keypoints, n, of a normally scanned 200DPI (dots per inch) 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 arithmetic 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 arithmetic 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 image. Figure 3 (left) shows keypoints overlaid on top of a brochure image. Corresponding to the image in Figure 3 (left), Figure 3 (right) shows the cumulative keypoint distribution map, in which 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.

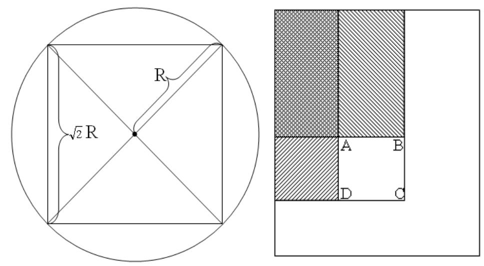

Figure 4. (left) Use the number of keypoints in the square to approximate the number of keypoints in the circle. (right) Use a cumulative keypoint distribution to compute the number of keypoints in a square.

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 4 (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:

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

The computation consists of only one addition and two subtractions. This is much more efficient than the brute force approach. With this fast approach, we can 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 4 (left). Denote Nc as the number of keypoints in the circle and Ns as the number of keypoints in the square, we can get

Ns≤Nc. 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.

3.3 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
```

3.4 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 90º. We take the second approach in our current implementation.

4. 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 5, the system consists of three major interaction entities, namely Authoring Client, EMM Server and Retrieving Client.

The authoring client is a PC application for authors to add EMMs to a document with the previously presented algorithm. With this tool, an EMM author can open a document file, specify an anchor point in a 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, or view keypoint distribution on the image. Currently, our system supports five types of media links: audio, video, text, webpage, and image.

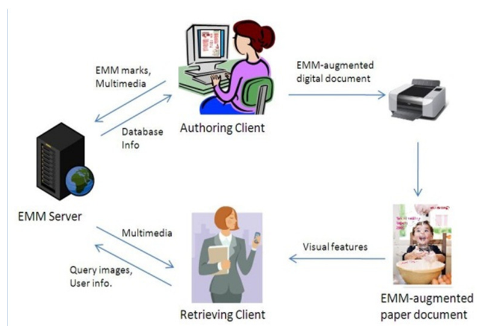

Figure 5. The architecture of the EMM system

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.

5. EMM EVALUATION

For early stage deployment and evaluation, we currently focus on using EMMs for pages within a publication such as a book or magazine and use our image local features FIT [8] for EMM identification. Users scan the book ISBN, or capture the cover page, or input other similar publication index. Then, EMMs within the publication can be used. For this purpose, we choose the ICME06 proceedings as our target document for an early stage performance evaluation of the EMM design and the semi-automatic EMM construction algorithm. The proceeding has 2188 letter-size (8.5”x11”) document pages with text, images, and figures. With the large page collection and content variation in the experiment, we believe that our test results will be reliable EMM-performance evaluations at the book level.

Before giving the demo system to many users for a more complete testing, we need to make sure the system is robust to commonly used printing devices, printing software, image coding mechanisms, and lighting conditions. Therefore, our preliminary EMM evaluation will focus on impacts of these factors.

In practice, EMM enhanced documents may be rendered with different printing software, and print on different size papers with different printers. People may also use EMMs in indoor environments or outdoor environments.

For testing these factors, 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 for indexing. From these 2188 pages, we also randomly selected 110 EMM overlaid pages for real query image capture. These 110 testing pages were saved in JPEG format with 1700 by 2200 resolution.

For testing the EMM sensitivity to printing software, we tried the IrfanView and the Windows XP Photo Printing Wizard. To check the EMM sensitivity to paper size, we print 90 pages on letter-size paper, and fit 4 pages on each letter-size paper for all other 20 pages. Because the EMM sizes on these 4-page/paper hardcopies are relatively small, filling the captured image with an EMM is a little difficult, as the camera phone was too close to the paper to be focused. 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.

Table 1. EMM testing results for different setups

Setup \ Batch #	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
Printing 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 reports our EMM performance evaluations when we change various factors described in this section. 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.

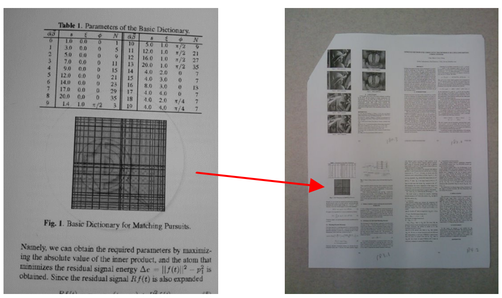

Figure 6.(left) The EMM capture that is failed to find its corresponding link. (right) The hardcopy used for this capture.

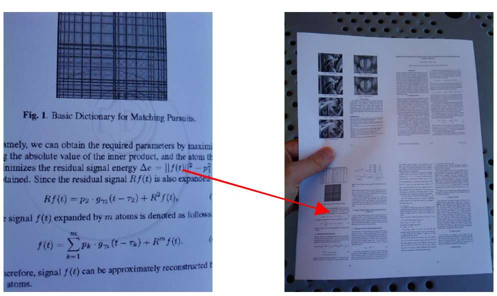

Figure 7.(left) The capture of an adjusted EMM. (right) The hardcopy used for this capture.

In all these 110 EMM query images, the system failed on only one image. This image is shown in Figure 6. According to the log of our code, the failure is caused by insufficient matching points. This could be solved by providing feedback information for the author to relocate the EMM.

5. CONCLUSION

This paper describes the Embedded Media Markers (EMM) printed on paper to signify the existence of media associated with documents and guide users to interact with the printout. We present a procedure and algorithm to semi-automatically arrange on paper EMMs that are friendly to users, machines and documents. With the EMMs users can retrieve the associate media by capturing an image of the EMM-signified document patch. We also discuss the potential applications and performance evaluation of EMMs.

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. In Proceedings of ACM 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, 2008.

[5] Hecht, B., M. Rohs, J. Schöning, and A. Krüger. Wikeye –Using Magic Lenses to Explore Georeferenced Wikipedia Content. PERMID 2007.

[6] Henze, N. and S. Boll. Snap and share your photobooks. In Proceedings of ACM 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 IEEE ICAT 2007, 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. In Proceedings of IEEE ICME'09, pp. 386-389.

[9] Liu, X. and D. Doermann, Mobile Retriever: access to digital documents from their physical source. Int. J. Doc. Anal. Recognit., 2008. 11(1): pp. 19-27.

[10] Lowe, D.G., Distinctive Image Features from Scale-Invariant Keypoints. Int. J. Comput. Vision, 2004. 60(2): pp. 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. LNCS, Vol. 3872, 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 ACM CHI'06, pp. 551-560.

[13] Rekimoto, J. and Ayatsuka, Y. 2000. CyberCode: designing augmented reality environments with visual tags. In Proceedings of ACM DARE 2000, pp. 1-10.

[14] Rohs, M. Real-world interaction with camera-phones. LNCS, Vol. 3598, pp. 74-89.

[15] Ke, Y. and Sukthankar, R., PCA-SIFT: A More Distinctive Representation for Local Image Descriptors. Proceedings of IEEE CVPR 2004.

[16] Hecht D. L., Embedded Data Glyph Technology for Hardcopy Digital Documents. SPIE -Color Hard Copy and Graphics Arts III, Vol. 2171, pp. 341-352.

[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): pp. 215-226.

[19] Costanza, E., J. Huang., Designable Visual Markers, Proceedings of ACM CHI'09, pp. 1879-1888.

[20] NODA TSUGIO, MOROO JUN, CHIBA HIROTAKA, Print-type Steganography Technology, Fujitsu 2006, Vol. 57. No. 3, pp. 320-324.

[21] Morrison, A., Oulasvirta, A., Peltonen, P., Lemmela, S., Jacucci, G., Reitmayr, G., Näsänen, J., and Juustila, A., Like bees around the hive: a comparative study of a mobile augmented reality map. In Proceedings of ACM CHI '09, pp. 1889-1898.

[22] Microsoft Tag. http://www.microsoft.com/tag/

Author's Introductions

Qiong Liu
FX Palo Alto Laboratory
Area of specialty: Electrical Engineering and Computer Science (Ph.D), Multimedia Information Access

Chunyuan Liao
FX Palo Alto Laboratory
Area of specialty: Electrical Engineering and Computer Science (Ph.D), Multimedia Information Access

Lynn Wilcox
FX Palo Alto Laboratory
Area of specialty: Electrical Engineering and Computer Science (Ph.D), Multimedia Information Access

Anthony Dunnigan
FX Palo Alto Laboratory
Area of specialty: Art Design

Bee Liew
FX Palo Alto Laboratory
Area of specialty: Electrical Engineering and Computer Science (MS), Multimedia Information Access