MobARDoc: Mobile Augmented Printed Documents

Tom Holland
The University of St. Andrews
tom.e.holland@gmail.com

Aaron Quigley
The University of St. Andrews
aquigley@cs.st-andrews.ac.uk

ABSTRACT
This paper presents work to date on the MobARDoc system for providing digital features for printed content through an augmented view on a mobile device. MobARDoc matches the printed text seen through the device's camera against the digital version of the same document to determine what content is visible and then overlay a touch based interface on top of the camera image. Our goal is to enable digital features for printed content (including academic papers, business documents, books, magazines and newspapers) which includes: the display of version changes, annotations, reference lookup, hyperlinks, content updates, character / author information, textual search, spelling, punctuation and grammar correction, text copy, dictionary and thesaurus support, pronunciation aid and translation, location extraction and mapping, date extraction and contextualisation and currency adaptation.

Author Keywords Augmented Reality; Paper interfaces.

ACM Classification Keywords H.5.1 [Multimedia Information Systems]: Artificial, augmented, and virtual realities

General Terms Algorithms, Experimentation, Design, Human Factors.

INTRODUCTION
Digital content has yet to completely replace printed content. While much of classical printed material comes from non-digital sources, today the majority of printed content comes originally from a digital source. If the printed content can be associated with its digital original, then we can provide some of the functionality of digital documents for their printed counterparts, using a mobile device to bridge this digital physical divide.

MobARDoc does not rely on Optical Character Recognition (OCR). It is instead based on a technique which matches features extracted from blocks of text within a mobile device's camera image against the original digital version of the document. For the purposes of this work “documents” is a generic term for any printed textual content including books, magazine and newspapers.

To enable augmentation to run in real time on the latest generation of smart phones we introduce methods to enhance the image processing phase to reduce the processing and memory costs.

UbiComp'10, September 26–29, 2010, Copenhagen, Denmark.
ACM 978-1-60558-843-8/10/09.

MOBARDOC
Our approach is broken down into a series of stages involving both the original digital content and the mobile device's camera image of the printed content, which form a pipeline.

Preprocessing the digital content
We use a commercial tool [1] to extract the textual content (including coordinates for individual word bounding boxes) from PDF files.

Earlier work [2] has explored the use of CUPS filter to automatically tag documents being printed with a barcode and to generate a PDF version (which can be pushed to a remote server for indexing as required). Commercially printed items (books, newspapers, magazines) are already printed with a visual identifier (a barcode) and come from a digital original (from which a PDF can be generated).

Document and page selection
When wishing to interact with printed content through the mobile device, the document must first be identified; either through decoding a printed visual identifier with the camera (such as a barcode) or through use of existing work in cover recognition ([3]).

Selecting a document loads the data profile for that document (which includes information such as the number of pages and the available augmentation layers).

To reduce the search space, the specific page must be identified. Currently this is a manual process, but existing work in page layout recognition ([4]) can be applied as an additional phase in the process.

Once a page is selected, the specific data profile for that page is requested from the server. This profile contains a map of the page's content, which is based on features extracted from the PDF version. The page map contains the textual content and bounding box coordinates of each word (in the original digital document's coordinate system). This data may have the textual context excluded (but leaving the bounding box coordinates of each word), which can be preferable in circumstances where exposing the original textual content would not be desirable (such as with copyrighted content).

Binarisation and filtering of the camera image
To identify the pixels from the device's camera image which form the printed text, we first binarise the image (pixels which form the content we are interested in are classified as '1', pixels which form the background, in our case the remainder of the page, are classified as '0'). We

convert the input image to greyscale (using the NTSC RGB ratios), resulting in a value between 0 and 255 for each pixel. Binarisation requires selecting a threshold, a value in the scale used to determine if a pixel is part of the content or background. Consider the situation of dark text on a light background, here values below the threshold are classified as content, while values above are background. A desirable threshold value varies, depending on the lighting and other external factors (e.g. shadows cast on the page, such as those by the mobile device being held over it). Global thresholding uses one threshold value for all pixels in the image. This threshold is calculated by first examining all pixels values in the image and uses the distribution to determine a suitable threshold value. Global thresholding is not robust to real-world conditions (due to differences in lighting across the image). Adaptive thresholding is capable of calculating different thresholds for pixels within the same image, by using the values of pixels surrounding the target pixel. While significantly increasing processing time, this does allow for lighting differences to be compensated for.

Our approach is to use different thresholds across the image based on a grid system to separate blocks of pixels. A histogram of the pixel greyscale values in each block is generated and the distribution of values within that block used to determine a suitable threshold. The histograms for blocks which contain text conform to similar patterns in distribution.

This approach is similar in its initial stages to [5] and [6]. Our grid block sizes are larger and their dimensions vary based on the text size in the source document. From a series of sample images of text in documents, books, newspapers and magazines, collected using a mobile device held in a realistic manner (an example can be found in Figure 1), we found that the average text size did not result in distinct

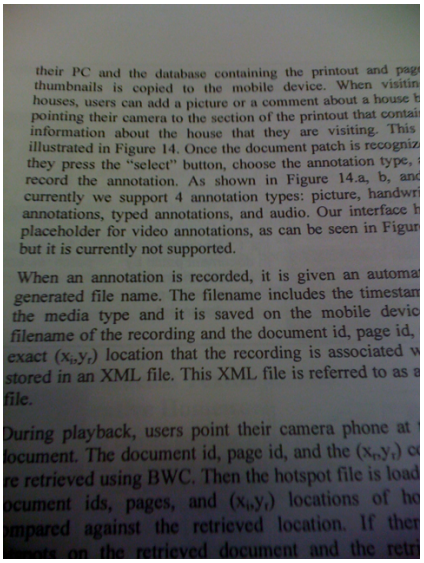

Figure 1. Printed text content through a mobile device camera.

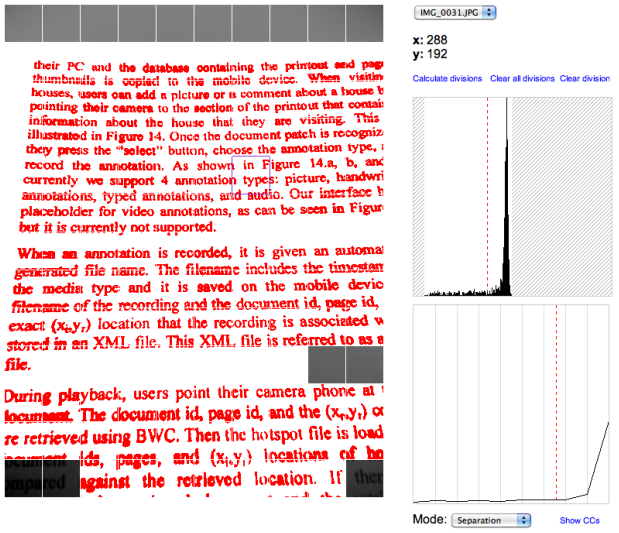

Figure 2. Sample data exploration interface showing pixel blocks, histogram generation, histogram value sampling, threshold calculation and resulting binarised output.

peaks in the greyscale range for content and background pixels (a peak representing the background was preceded with a series of low values across wide range). As a result, identifying bimodality in the form of two Gaussian distributions could only be used to detect and process large pieces of text (such as headings).

Using these sample images (not specifically the target page's or document's content), we pre-generate appropriate threshold values for different histogram distribution patterns (see Figure 2). This has the added benefit of filtering out blocks which do not contain text (or contain non-text items), since blocks not matching a known distribution pattern are ignored (this would require additional stages if standard binarisation techniques were employed).

Text histogram distribution patterns can vary, depending on lighting, the device used and the size of the text. The size of the blocks that compose the grid must also be determined. Although major differences between histograms generated from the digital original suggest that direct comparison cannot be made, the digital original can be used to determine the size of text within the page (and if the page contains pieces of text of different sizes, such as headings and paragraphs). These are used to determine the appropriate size of grid blocks and select suitable histogram distribution patterns from the general pre-generated set to match against. This information is supplied to the device when the data profile for a page is requested. Similarly, device differences which affect the histograms are compensated for by pre-generating sets of general matching data using that device; when a device requests the data profile for a page, the relevant details of the device are sent in the request, allowing specific sets of device customised matching data to be returned.

This technique is less accurate than adaptive binarisation techniques, the resulting classifications containing more noise. Camera images of printed text have the effect of increasing the greyscale range over which pixels are spread. From a digital original of black text on a white background, white becomes varying shades of grey. Black text also becomes shades of grey, but these greys are darker and spread over a narrower range than the white background (although with some overlap). What begins as a crisp edge between black content and white background in a digital file becomes a fuzzy grey transition once printed and viewed through the camera of a mobile device. Our technique results in some of this fuzzy transition being classified as content (causing letters to appear thicker and some to merge into one another). This is acceptable, since we are not performing standard OCR techniques on the content and does not result in an amount of error sufficient to prevent the features extracted from being correctly matched against the digital original.

Connected components and normalisation

Once the pixels have been classified as textual content or other, we use a standard 8 connected components technique to join neighbouring content pixels into 'blobs'. These blobs may contain a single character, multiple characters or whole words (the result of our binarisation is that some letters may have become merged with others). The connected blobs of pixels must now be formed into words (since the features we wish to extract for comparison to the digital original are based on words). The connected pixels are inconsistent; the same camera image may contain blobs of pixels that represent single or multiple letters (see Figure 3). We use a series of domain specific stages to assemble the blobs into words for feature extraction:

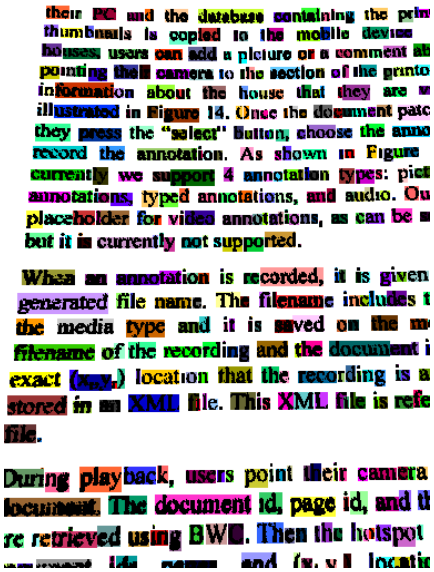

Figure 3. Bounding boxes of connected components after binarisation (visualised with a random background colour assigned to each 'blob' of joined pixels).

1. Calculate the centroid of each blob
2. Select the closest blob corner to each blob corner (excluding its own corners)
3. Using the blob centroids, calculate the bearing between each pair of 'nearby' blobs from the previous step
4. Using the distribution of all these bearings, discount those blocks which do not fit with a line pattern (this does not require that the text forms horizontally aligned lines in the camera image)
5. With the remaining pairs of nearby blobs, calculate the Euclidean distance between the blob edges. Eliminate nearby blobs above a threshold (calculated using the distribution of these distances)
6. Blobs that still have nearby blocks are merged into a single blob to become a word.

Feature extraction and comparison against digital content

We are experimenting with a variety of features that can be extracted from the pattern of words exposed through the previously described stages. Locally Likely Arrangement Hashing (LLAH), an extension of Geometric Hashing, has been built upon in a series of papers ([7], [8], [9]). The use of the ratio of triangle areas within small subsets of words has proven to be resilient against perspective differences between the digital original and the camera's view of the printed content, albeit at the cost of generating all possible combinations within a subset of 'close' words (measured by Euclidean distance, which varies depending on the perspective). Given that we are searching a reduced space (we wish to locate the words visible in the device's camera image within the known page) and existing approaches having been geared towards recognition within ten of thousands of pages, we are looking at less intensive methods which produce a more generalised signature for a block of words (we only require that a block of words is uniquely identifiable by pattern within a page).

Distortion calculation and interface overlay

Once the pattern of words visible to the device has been matched to words within the digital document page, the difference in perspective between the digital original and mobile device's view on the printed document must be calculated to enable the interface to be overlaid on the mobile device's view. Using the centroids of matched word bounding boxes as points on a plane, and using the matched points in the original digital document as the base, a homography is calculated. The visible portion of the relevant overlay interface (which is stored in the original digital document's perspective) can then be distorted accordingly to match the view through the mobile device.

User interaction and augmentation layer switching

Augmentation layers are the overlays available for a document. These are composed of content to be overlaid on top of the camera image and interface elements which can be triggered from this content (which may include the ability to display further content related to the items being interacted with).

Different types of documents will benefit from different augmentation layers. For a business document or research paper this may include differences between versions, coworker / coauthor annotations and hyperlinks for URLs or references. For books these may include annotations by the original author (similar to the concept of "DVD extras"), updates to the content (such as more recent information in a travel guide) or summaries attached to each character name when displayed in a novel, which become increasingly detailed the further into the book the page belongs, to help a reader keep track of characters and their history (without exposing as yet unrevealed plot details). For magazines: author profiles and hyperlinks; for newspapers: updated article content (such corrections and late additions to a story). Media (such as audio and video) can be attached to existing content with hyperlinks or indicated by iconography in the document borders.

Human authored augmentation layers by an authority (such as the author or editor) could be made available to all; some augmentation layers could be determined by the reader's social network (allowing a user to see a friend's / colleague's comments on an article if desired, allowing digital comments to be attached to printed articles).

Numerous automatically generated augmentation layers can also be leveraged. These could include: textual search, hyperlinks, spelling, punctuation and grammar corrections, text copy functionality, reference extraction, dictionary and thesaurus lookup, pronunciation and translation (using online services), location extraction (linking to maps), date extraction (what else happened then; add to my calendar) and monetary values (conversions between different currencies; value now in comparison to when printed).

The inclusion or exclusion of the original textual content in the data profile affects which augmentation layers can be made available. Augmentation layers based on a person adding content (such as annotations) are unaffected. Automatically generated augmentation layers (such as spelling, textual search, text copy, dictionary or thesaurus) require that the original textual content must be exposed to allow them to function.

CONCLUSIONS AND FUTURE WORK

We have presented our work in progress on the MobARDoc system to provide digital document functionality to printed digital documents through a mobile device. We are specifically using features extracted from text patterns, rather than OCR, to match the portion of content visible through the mobile device's camera within the original digital document page to allow a variety of functionality to be made available through an interface overlaid on the device's camera image. Our work can be differentiated from existing approaches in our specific targeting of the real-world use of mobile devices and adapting or replacing parts of the pipelines used in similar work to meet our goals.

We plan to continue to develop our pipeline and produce a functional prototype on a mobile device suitable for a mixture of automatic testing and human-centered experiments in two of our target domains: research papers and books.

REFERENCES

1. Pdflib text extraction toolkit. http://www.pdflib.com/products/tet/
2. Holland, T. and Quigley, A. Doctrack: automatic printed digital document tagging and remote retrieval. Late Breaking Results of the Sixth International Conference in Pervasive Computing, May 2008.
3. Tsai, S.S., Chen, D., Singh, J. and Girod, B. Rate Efficient Real Time CD Cover Recognition on a Camera Phone. In ACM Multimedia, Vancouver, Canada, October 2008.
4. Van Beusekom, J. Document Layout Analysis. Diploma Thesis, Image Understanding and Pattern Recognition Group, Technical University of Kaiserslautern, Kaiserslautern, Germany (under Prof. Dr. Thomas Breuel).
5. Chow, C.K. and Kaneko, T. Automatic detection of the left ventricle from cineangiograms. Computers and Biomedical Research, vol. 5, pp. 388-410, 1972.
6. Taxt, T., Flynn, P.J. and Jain, A.K. Segmentation of document images," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 11, no. 12, pp. 1,322-1,329, 1989.
7. Nakai, T., Kise, K. and Iwamura, M. Use of affine invariants in locally likely arrangement hashing for camera-based document image retrieval. Document Analysis Systems VII, pp. 541–552, 2006.
8. Nakai , T., Kise, K. and Iwamura, M.. Camera based document image retrieval with more time and memory efficient LLAH. In Proceedings of Second International Workshop on Camera- Based Document Analysis and Recognition (CBDAR2007), pp. 21–28, 2007.
9. Uchiyama, H., Pilet, J. and Saito, H. On-line document registering and retrieving system for AR annotation overlay. In Proceedings of the 1st Augmented Human International Conference (AH '10), pp. 1-4, 2010.