Camera Phones with Pen Input as Annotation Devices

Michael Rohs
Institute for Pervasive Computing
Department of Computer Science
ETH Zurich, Switzerland
rohs@inf.ethz.ch

Christof Roduner
Institute for Pervasive Computing
Department of Computer Science
ETH Zurich, Switzerland
rodunerc@inf.ethz.ch

ABSTRACT

This paper explores the use of camera phones with pen input as a platform for generating digital annotations to real-world objects. We analyze the client-side requirements for a general annotation system that is applicable in mobile as well as stationary settings. We outline ways to create and interact with digital annotations using the camera and pen-based input. Two prototypically implemented annotation techniques are presented. The first technique uses visual markers for digital annotations of individual items in printed photos. The second technique addresses the annotation of street signs and indication panels. It is based on image matching supported by interactively established 4-point correspondences.

1. INTRODUCTION

Digital annotations link user-generated digital media to physical objects. This allows users to combine the persistency and availability of physical objects with the flexibility and versatility of digital media. Physical media, like printed photographs and street signs, are tangible and permanent, but can typically store only a limited amount and type of information. Digital media, like text, graphics, audio, and video, are immaterial and volatile, but are virtually unlimited in terms of the amount and type of information they can represent. They can be automatically processed and shared across space and time. Using physical media as "entry points" [11] to digital annotations is a way to structure information and to embed it into the real world. In comparison to other types of augmentation the distinct feature of annotations is that users can freely create them and that they are not predefined by the system or any content provider.

Many projects have looked into annotating physical media with online information and services [2, 6, 7, 9, 11, 13, 14, 15]. Our goal in this paper is to explore the interaction possibilities of camera phones (or camera-equipped PDAs) with pen-based input as a platform for generating digital annotations. We present ideas of how to create and interact with annotations using phonecam-specific features. More generally, we are interested in how a mobile user interface for a generic annotation system could be structured that allows for the creation, access, sharing, and organization of digital annotations. This system shall be usable in stationary and mobile settings and exclusively rely on mobile devices as user interfaces.

Camera phones fulfill several essential requirements of a mobile annotation device. First, the camera in combination with image processing algorithms allows for identifying annotatable objects and for determining their orientation. There are multiple options for visually identifying physical objects, including image recognition techniques and visual marker detection. RFID tagging and near-field communication (NFC) for mobile devices [8] offer non-visual alternatives. Determining the orientation of objects in the camera image enables the registration of graphical overlays in the camera image in the sense of augmented reality [1, 4]. Second, wireless connectivity allows for sharing annotations with others, persistently storing and organizing them on a back-end server, and getting up-to-date information. Third, camera phones combine the ability to create annotations in multiple media types with the ability to play them back. Fourth, they are ubiquitously available in users' everyday situations.

A distinct feature of pen-based input devices is that they enable users to make more fine-grained annotations of objects captured with the camera. Users can encircle objects and create specific annotations to them, draw arrows to give directions, or put predefined icons onto the captured image. In addition to allowing for more fine-grained annotations, pen-input can also support image processing algorithms by telling the system which objects are relevant and which ones are not. In section 3 we show how this can be used to accurately segment street signs in images from the background.

Digital annotations can take many forms, such as text, graphics, audio, video, hyperlinks, vCard and vCalendar items, drawings and predefined graphical objects on the captured image. All of these media types can be created and presented on camera phones. If the semantics of the annotated object or its classification in a taxonomy are known to the system, then users might be provided with forms for rating objects or widgets for entering specific parameters. This supports users in minimizing the amount of data they have to enter into their mobile device in order to create an annotation. Annotations can further be supported by context data that is automatically gathered from the mobile phone [3], such as the current location or the time of day.

In section 2 we discuss the annotation of physical media with visual codes [10] in stationary settings. In section 3 we discuss the annotation of signs – or other areas with four clearly distinguishable corners – in outdoor environments, where attaching visual markers might not always be feasible.

2. DIGITAL ANNOTATIONS WITH VISUAL CODES

Even though digital photography has spread rapidly over the past few years, printed photographs are still omnipresent. To explore novel ways of attaching digital content to these artifacts, we have implemented a prototype application that allows for the annotation of pictures in a physical photo album (see Figure 1). We use a Windows Mobile 2003 based smartphone with an integrated camera to enable users to attach text or multimedia content (e.g. voice notes) to arbitrary parts of album pictures. In our approach, we stick a two-dimensional marker on every page of a conventional photo album. Each marker contains a unique number that we use to identify the album page. The markers can either be pre-printed onto pages or they can be supplied to users as individual stickers that they can put on album pages.

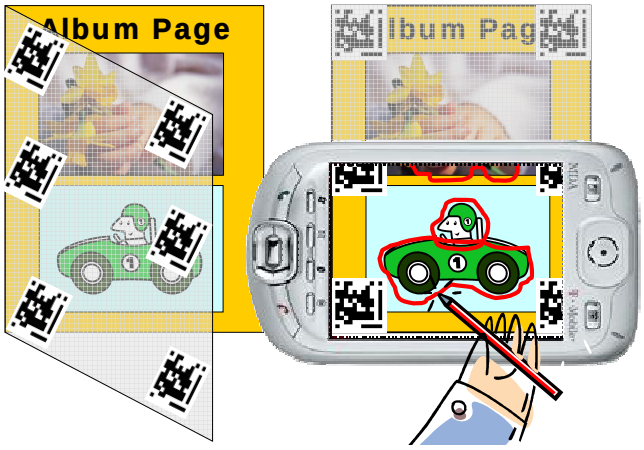

Figure 1: Digital annotation of a photo album with visual codes.

In order to annotate a photo on an album page, users take a picture of that page with their camera-equipped smartphone. Our annotation application running on the phone then extracts the marker from this snapshot and yields the numeric page identifier that it contains. At the same time, the snapshot is presented to the user on the phone's display. By drawing a polygon with the pen, users can then specify the part of the photo that they would like to annotate.

The phone then maps the polygon drawn on the display to a corresponding polygon in the physical photo's plane. In order to achieve this, we use the primitives available in the visual code system [10]. A visual code is a two-dimensional marker that provides a coordinate system which is independent of the camera's orientation. On top of that, the camera's distance, rotation angle, and amount of tilting relative to the code can be determined. These features allow us to transform the display coordinates of the user's drawing into a coordinate system in the physical marker's plane.

In our prototype application, users can attach plain text, hyperlinks, voice recordings, and files to a polygon by encircling an object of interest in a photograph. Along with the code's value identifying the album page and the polygon's coordinates in the code coordinate system, this annotation is sent over GPRS, WLAN, or Bluetooth to a back-end server, on which it is stored in a database.

Obtaining the annotations for a given album page works analogously: When the user takes a snapshot of an album page, the page identifier stored in the code tag is read. The application then fetches from the back-end service the coordinates of all polygons that are available for the given album page. These coordinates relative to the code tag are mapped to the corresponding pixels in the snapshot, which allows the application to highlight the polygons on the phone's display. Users can then read or play back the annotations for an object by tipping the polygon surrounding it.

The size of a printed code is currently 2x2 cm. The camera provides a resolution 640x480 pixels. With higher resolution cameras smaller codes (1x1 cm) can be used, which are less obtrusive. We experienced some difficulties regarding the placing of visual codes that we put on album pages. Depending on the size of an album page, the smartphone has to be placed relatively far away from it in order to take a snapshot of the whole page. As a result of this, the application occasionally could not recognize the visual code any more. We thus attached up to six visual codes to a single album page. This ensures that, when the camera is placed closer to the album and covers only a part of it, there is still at least one code located in this part. This, however, incurs the problem of an album part that, depending on how the camera is held, can be seen with a certain code first and another code later. Since each visual code has its own coordinate system, we needed a way to transform the coordinate systems into each other. This could be achieved by pre-printing the visual codes at fixed positions on album pages. In our prototype, we opted for another approach that still allows users to freely place stickers on a page. However, we introduced an additional step to initialize album pages before annotation. In this step, users have to take a few snapshots of a page that contain several codes at the same time. The application can thus learn about the arrangement of the markers and obtain the data needed to transform the coordinate systems of the different codes into each other.

The idea of annotated photographs was presented before in the Active Photos project [6]. However, the annotation and viewing process is different than with our prototype: The annotation of an Active Photo is done in a Web browser and relies on the availability of a digital version of the photo. Whereas we use a standard off-the-shelf smartphone, a special lap-mounted appliance is needed to interact with Active Photos. A third difference lies in the way regions with annotations are shown in a picture. While Active Photos are placed in a transparent envelope where marking objects offers additional content, we overlay the image shown on the smartphone's display with polygons.

Since our prototype builds upon the generic platform of smartphones and does not require the annotated object to be available in a digital version, it has a wide array of potential applications. It would be possible to annotate not just photo albums, stamp collections, or elements in a newspaper, but also all kinds of everyday objects ranging from product packages to posters and places in a city. Another field are applications where professionals such as the police or insurers need to annotate, for example, a crime scene, accident

or damage. Architects could attach visual code stickers onto construction plans in order to add digital annotations (e.g. drawings with the stylus) while at the construction site. Yet another application area is medical diagnosis, in particular the annotation of printed X-ray images onto which visual code stickers could be pasted.

3. SIGN ANNOTATIONS WITH IMAGE MATCHING

Annotating objects by attaching visual markers is sometimes not an option, since the objects may not be under the control of the annotator, physically not reachable, or visual markers might be too obtrusive. This could be the case at public places, for example. Yet many objects, like street signs, shop signs, restaurant signs, indication panels, and even facades of buildings are sufficiently regular and have clear-cut borders to the background in order to be used as annotation anchors. Our idea for using signs as annotation anchors is based on interactive support by users and simple image matching, which makes the approach suited for execution on camera phones with pen-based input. Additionally, context data that is automatically gathered from the mobile phone is taken into account.

In order to attach an annotation to a sign or to retrieve annotations, users take a photo of the sign including any background with their camera phone. The result might be a picture as shown in Figure 2a. Users now tap the four corners of the sign on the device screen with their stylus (Figure 2b). A frame around the sign appears, whose corners have handles to allow for readjustment. This interactively supported sign selection approach solves two problems. First, if multiple candidate objects are present in the image, the one of interest to the user is selected. Second, the image segmentation process becomes trivial.

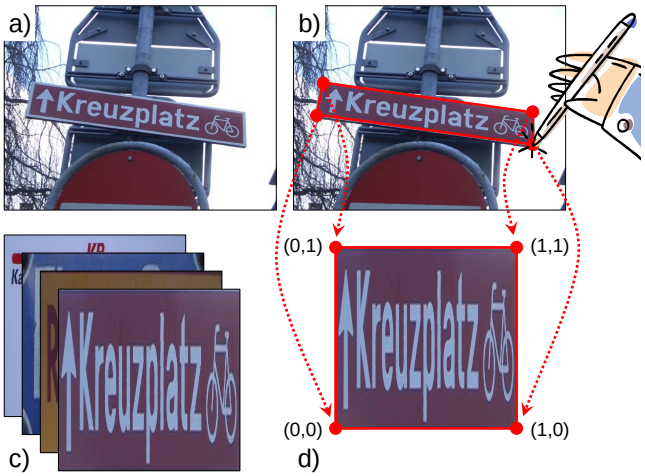

Figure 2: Annotating signs using camera phones with pen-based input: (a) captured photo, (b) framing a sign with the pen, (c) set of templates, (d) mapping framed area to unit square.

To enable simple matching of the framed part of the image (the sign) against a set of templates (Figure 2c), the framed part is projected into the unit square (Figure 2d). Depending on the orientation of the user towards the sign when taking the photo, the sign may appear perspectively distorted in the photo. This perspective distortion can be removed and the framed part projected into the unit square as follows. The four corners of the frame are set as correspondences to the corners of the unit square (Figure 2d). Since the frame corners are coplanar, there exists a unique homography (projective transformation matrix) between the pixel coordinates of the framed area and the unit square [5]. By scaling the unit square, we can thus produce a square request image of a predefined size (in our current implementation 480x480 pixels), which is sent to a server for matching against a set of template images. If the mobile device stores the relevant set of templates (Figure 2c), then the matching algorithm can also be run on the mobile device.

To further facilitate image matching and to make it more robust, we take a number of context parameters into account that are automatically gathered from the phone at the time of capture and sent to the server together with the request image. The context parameters comprise the current GSM cell id(s) for spatially restricting the search and the time of day (morning, noon, afternoon) to restrict matching to images taken under similar light conditions. The server may optionally add the current weather conditions (sunny, cloudy, rainy) to further restrict the search.

In our current implementation, the actual matching algorithm on the selected subset of templates is executed on a background server, which stores the shared annotations and templates. It computes the sum of differences between the request image and each template image by adding up pixel-by-pixel differences of the hue value. If saturation is below a certain threshold for a pixel, it adds the (gray) value difference. The server returns a list of matching annotations to the phone, ordered by increasing image difference values.

We expect that if we take contextual parameters, such as the current cell id, the time of day, and the current weather conditions into account, the remaining number of relevant templates will be a few dozen. Preliminary experiments on phonecam-generated images show promise that the matching algorithm can correctly distinguish that number of objects. A problem is of course that images of street signs are very similar – having a common text color and a common background color. Shop signs and restaurant signs typically show more variation in terms of color and visual appearance. Since street names are not unique, different physical signs with the same contents may appear multiple times along a street. In this case, annotations that refer to a particular location are not possible, but only annotations that refer to the street as a whole. This is true for all media that are reproduced multiple times – like flyers, product packages, images in newspapers, or logos.

The approach is beneficial for users, if it requires less effort to take a photo and tap the four corners than to enter some unique textual description of the annotated object (which of course needs to be identical across different persons if annotations are to be shared). Secondly, if the algorithm is not performed on the phone itself, the approach requires the upload of an image part and context data to the server via the phone network, which takes some time. We still need to investigate, how accurately users typically draw frames on a

mobile phone with pen-based input and in what way imprecise frames degrade matching performance. In addition to simple pixel-to-pixel color comparisons, better image matching approaches need to be investigated [12]. For signs that have a clear visual border against the background it might suffice if users specify a single point on the sign. Image processing algorithms could then automatically extend the region based on color similarity and find the corners.

The presented approach is applicable if it is not desirable or impossible to attach visual markers to an object. Objects are recognized based on their unmodified visual appearance. Thus the facade of a building can be annotated even from a distance. A disadvantage is that a conscious effort is required for the user to retrieve annotations. Annotations are not discovered automatically, as is the goal in augmented reality systems [1, 4]. Still, there are a number of compelling applications, like pervasive gaming, in which the proposed sign annotation approach can be a component.

4. SUMMARY AND FUTURE WORK

We have investigated interaction possibilities that camera phones with pen-based input provide for creating digital annotations of physical objects. Camera phones fulfill the technical requirements of object identification and orientation detection, online connectivity for sharing annotations, the ability to create and play back annotations, and ubiquitous availability. Pen-based input allows for more fine-grained annotations. We have presented two digital annotation approaches that are applicable under different circumstances. The first one relies on visual code stickers and enables the annotation of individual items on a printed page. The second approach is based on the interactive establishment of a 4-point correspondence, which helps separating a selected area from the background and thus simplifies image matching.

We are going to investigate especially the second approach further, which is still at an early stage. Topics that need to be explored include user acceptance of tapping the corner points, the typical accuracy of the area framed by the user, better image matching techniques, a larger set of test images taken under different lighting conditions, as well as target applications that can be based on this approach. Applications that we intend to implement are restaurant recommenders as well as pervasive urban games that involve looking for sign annotations within a scavenger hunt. Another aspect is the creation of a coarse taxonomy of annotated objects that would allow for automatic processing of images and annotations. The background system could then automatically provide the user with other relevant shared annotations and related objects.

Acknowledgments

We thank Kaspar Baltzer for implementing the annotated photo album as part of his semester project.

5. REFERENCES

[1] R. T. Azuma. A survey of augmented reality. *Presence: Teleoperators and Virtual Environments*, 6(4):355–385, August 1997.

[2] J. Barton, P. Goddi, and M. Spasojevic. Creating and experiencing ubimedia. HP Labs Technical Report HPL-2003-38, 2003.

[3] M. Davis, S. King, N. Good, and R. Sarvas. From context to content: Leveraging context to infer media metadata. In *MULTIMEDIA '04: Proc. 12th ACM conf. on Multimedia*, pages 188–195. ACM Press, 2004.

[4] S. K. Feiner. Augmented reality – a new way of seeing. *Scientific American*, 4 2002.

[5] P. S. Heckbert. Fundamentals of texture mapping and image warping. Master's Thesis, Department of Electrical Engineering and Computer Science, University of California, Berkeley, 1989.

[6] T. Kindberg, E. Tallyn, R. Rajani, and M. Spasojevic. Active photos. In *DIS '04: Proceedings of the 2004 conference on Designing interactive systems*, pages 337–340. ACM Press, 2004.

[7] P. Ljungstrand and L. E. Holmquist. WebStickers: Using physical objects as WWW bookmarks. In *CHI '99: CHI '99 extended abstracts on Human factors in computing systems*, pages 332–333. ACM Press, 1999.

[8] Nokia Corporation. Nokia Mobile RFID Kit and Nokia NFC (Near Field Communication). www.nokia.com/rfid and www.nokia.com/nfc.

[9] J. Rekimoto, Y. Ayatsuka, and K. Hayashi. Augment-able reality: Situated communication through physical and digital spaces. In *ISWC '98: Proceedings of the 2nd IEEE International Symposium on Wearable Computers*, pages 68–75, 1998.

[10] M. Rohs. Real-world interaction with camera-phones. In *2nd International Symposium on Ubiquitous Computing Systems (UCS 2004)*, pages 39–48, Tokyo, Japan, November 2004.

[11] M. Rohs and J. Bohn. Entry points into a smart campus environment – overview of the ETHOC system. In *ICDCSW '03: Proc. 23rd Intl. Conf. on Distributed Computing Systems*, pages 260–266, 2003.

[12] Y. Rui, T. S. Huang, and S.-F. Chang. Image retrieval: Current techniques, promising directions, and open issues. *Journal of Visual Communication and Image Representation*, 10(1):39–62, March 1999.

[13] M. Smith, D. Davenport, H. Hwa, and L. Mui. The annotated planet: A mobile platform for object and location annotation. In *1st Int. Workshop on Ubiquitous Systems for Supporting Social Interaction and Face-to-Face Communication in Public Spaces at UbiComp 2003*, October 2003.

[14] M. A. Smith, D. Davenport, H. Hwa, and T. Turner. Object auras: A mobile retail and product annotation system. In *EC '04: Proc. 5th ACM conf. on Electronic commerce*, pages 240–241. ACM Press, 2004.

[15] R. Want, K. P. Fishkin, A. Gujar, and B. L. Harrison. Bridging physical and virtual worlds with electronic tags. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 370–377. ACM Press, 1999.