Markerless Visual Fingertip Detection for Natural Mobile Device Interaction

Matthias Baldauf
FTW - Telecommunications Research Center Vienna
Donau-City-Strasse 1
Vienna, 1220, Austria
baldauf@ftw.at

Peter Reichl
Université Européenne de Bretagne
5 Boulevard Laënnec
Rennes, 35000, France
peter.reichl@irisa.fr

Sebastian Zambanini
Computer Vision Lab
Vienna University of Technology
Favoritenstrasse 9/183-2
Vienna, 1040, Austria
zamba@caa.tuwien.ac.at

Peter Fröhlich
FTW - Telecommunications Research Center Vienna
Donau-City-Strasse 1
Vienna, 1220, Austria
froehlich@ftw.at

MobileHCI 2011, Aug 30–Sept 2, 2011, Stockholm, Sweden.
ACM 978-1-4503-0541-9/11/08-09.

Abstract
The vision-based detection of hand gestures is one technological enabler for *Natural User Interfaces* which try to provide a natural and intuitive interaction with computers. In particular, mobile devices might benefit from such a less device-centric but more natural input possibility. In this paper, we introduce our ongoing work on the visual markerless detection of fingertips on mobile devices. Further, we shed light on the potential of mobile hand gesture detection and present several promising use cases and respective demo applications based on the presented engine.

Keywords
Mobile natural interaction, finger detection, mobile computer vision

ACM Classification Keywords
H.5.2 [**Information Interfaces and Presentation**]: User Interfaces - *Input devices and strategies, Interaction styles*; I.4.9 [**Image Processing and Computer Vision**]: Applications

General Terms
Human Factors, Algorithms, Design

Figure 1. *g-speak* includes sensor gloves for gesture recognition

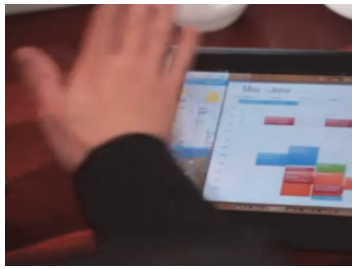

Figure 2. *eyeSight* recognizes waving gestures

Figure 3. *6th Sense* uses colored markers to ease fingertip detection

Introduction

So-called *Natural User Interfaces* (NUIs) aim at enabling more intuitive and direct communication between users and machines, e.g. through the recognition of human speech or hand gestures. Even though pioneering research in the field of 'natural interaction' was already done in the 1980s (cf. [6]), NUIs gained increasing interest and publicity in the last few years. Popular recent solutions include *g-speak* (Figure 1) enabling gestural commands through sensor gloves and *Kinect*, a mass-market gaming tool visually recognizing body gestures.

We argue that especially mobile devices benefit from such NUIs since these gadgets usually provide limited and less comfortable input capabilities than desktop computers with full keyboards and mouse. Further, less device-centric interaction techniques allow a mobile user to focus her attention on the task and its content instead of on the device while on the move. While speech recognition and acceleration-based gestures for mobile devices are well-investigated and respective applications are already available on today's smartphones, specific work on vision-based gestural interaction for mobile devices is scarce despite its manifold potential use cases.

This paper presents our ongoing work on a mobile vision-based fingertip detection engine and the investigation of enabled novel interaction techniques. First, we report on existing approaches for visual hand and finger detection and on respective work in the mobile context. We then describe our prototypical implementation for a smartphone. Based on this engine we present several promising use cases and respective demo applications.

Related Work

Estimation and tracking of hand pose is an active research topic in computer vision [2]. Besides sophisticated methods for recovering the full kinematic hand structure [15], more simple and thus computationally less expensive methods, which are able to provide partial pose estimation for specific tasks (e.g. finger detection), have been proposed. Basically, these methods rely on appearance-specific 2D image analysis and consist of a hand localization step and a subsequent feature analysis to estimate the position of the fingertips. Apart from methods that depend on strong assumptions like a static background [6] [11], hand localization in a more general setup is commonly achieved by skin color segmentation. Skin color segmentation methods detect regions in the image with skin-characteristic color values and comprise lookup table matching [8], thresholding in chromaticity space [15], pattern classification [16] or multiscale aggregation [4]. Another type of methods for hand pose estimation involves classification-based object detection approaches that apply classifiers on features extracted from sliding windows to detect the hand position and pose [3][9].

As mentioned above, appropriate work addressing mobile devices is scarce so far. However, a recently filed patent by *Apple* [12] indicates the increasing relevance of vision-based mobile gesture detection. The respective application deals with command input by closely sweeping a finger above the built-in camera at

Figure 4. Hand segmentation through thresholding

Figure 5. Contour finding and fingertip analysis

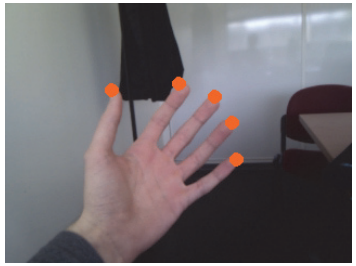

Figure 6. Fingertip visualization in the live video

the backside of the mobile, e.g. for controlling the play-back of voice mail while listening. A similar existing product restricted to motion gestures is *eyeSight* (Figure 2) which detects a waving hand through the mobile camera to trigger dragging or scrolling operations.

Recent projects investigating the usage of more complex finger gestures in a mobile context exploit colored markers attached to the fingers and/or utilize prototypical technical realizations run on powerful portable computers. Examples include the investigation of gestural interaction with projected content in the project *6th Sense* [7] (Figure 3) and the study of gestural interaction with *Imaginary Interfaces* [5] completely dispensing from visual feedback. Other relevant work such as *Behand* [1] aiming to control 3D objects in a mobile augmented virtuality scenario only exists in the form of design studies. A current markerless hand segmentation approach decidedly oriented on mobile devices [4] takes four seconds for a 160x160 pixels image and thus does not allow for real-time applications.

Prototypical Implementation

To be able to explore the possibilities of visually spotted hand gestures on a mobile device in depth, we implemented a prototypical fingertip detection engine for *Android* smartphones. Our test device was an *HTC Desire* phone equipped with a 1 GHz processor, 576 MB RAM, and a 5 megapixel camera and running Android 2.1. The image processing tasks were realized using the popular open source computer vision library *OpenCV*. We made use of Android's Native Development Kit and SWIG wrappers (http://www.swig.org/) to implement performant native C++ code and integrate it into a common Android application written in Java.

Our current implementation follows a lightweight pixel-based skin detection approach in order to enable real-time fingertip detection even on resource-restraint devices. Pixel-based approaches classify each pixel individually by its color value without considering spatial coherence of the output. Thus they are inherently invariant to rotation as well as robust to partial occlusion.

The following method is applied to each 320x240 pixels-sized camera frame at interactive frame rates. We start with a thresholding algorithm to segment skin-colored pixels from the background and create a respective binary image depicted in Figure 4. As threshold values we use the skin-specific values identified by Peer et al. [10]. Morphological operations then remove noisy pixels and fill possible holes in skin-colored areas. From the result of *OpenCV*'s contour finding operation, we assume the largest connected component to be the hand. We then try to identify the hand palm through the largest possible inner circle found by a distance transformation. The largest enclosing circle is calculated by *OpenCV* means. Figure 5 shows both circles and their centers in green and red, respectively. Iterating over the hand's contour points, consequent points with a distance above a certain threshold from the inner circle's center (with regard to the radius of the outer circle) are identified as upper finger parts (in Figure 2 marked in blue). The vector

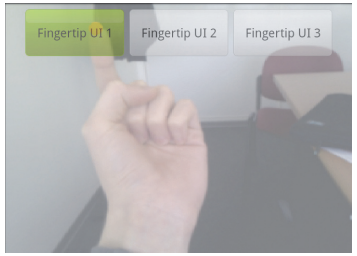

Figure 7. Semi-transparent user interface with camera view

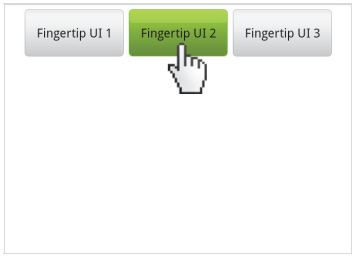

Figure 8. Detected fingertip indicated by traditional mouse cursor

Figure 9. Segmented hand mask blended into user interface

spanned by the centers of the circles indicates the hand's approximate orientation. Thus, potential candidates located towards the underarm can be ignored. Finally, the local distance maxima of the respective contour segments are considered as actual fingertips. Figure 6 shows the detected fingertip points as overlay on top of the live camera view.

Use Cases

On top of the presented mobile fingertip detection engine, we started to implement several demo applications in order to hereafter investigate its possibilities and limitations and to illustrate and explore different use cases.

Back-of-Device-Interaction

Visually spotted fingertips can be used to realize a novel type of so-called *back-of-device-interaction*, i.e. the capturing of user input at the usually unused rear side while holding the device. So far, traditional back-of-device interaction makes use of custom hardware prototypes with touch-sensitive surfaces attached to the device's back [13]. Though our visual approach does not allow for direct contact with the device, it enables a related interaction technique for off-the-shelf mobile devices. The technique might be useful for applications where the common occlusion of display areas during touching needs to be absolutely avoided, yet gestural interaction is preferred. For a non-touchscreen device, it even enables gestural interaction via the built-in camera. Mobile games exploiting this novel interaction style are another promising use case.

Several design variants are conceivable for visual back-of-device interfaces. Figure 7 shows our prototype for a semi-transparent graphical interface on top of the camera view to follow the fingertip. Buttons and interface elements can now be 'touched from the backside'. Another approach depicted in Figure 8 shows a traditional mobile interface while a camera-detected fingertip moves an additional cursor borrowed from desktop systems. Finally, Figure 9 depicts a combined design alternative that overlays the segmented hand on the user interface being not as abstract as the cursor approach and not as obtrusive as the semi-transparent interface.

Real-World Interaction

Traditional *augmented reality* applications make use of the so-called *magic lens* metaphor and overlay live-video with corresponding points-of-interest layers or similar. Taking the steadily increasing amount of georeferenced information into account, hand gestures might serve as a powerful alternative to currently used icons and touchscreen interaction. The visual detection of a hand gesture in the video enables the gesture-based selection of objects in the real world, e.g. by pointing at the object (Figure 10) to retrieve more information about it. Respective objects could be identified via the extraction of natural image features and the comparison with a database of known objects. This approach not only avoids the presentation of a myriad of information icons and the respective occlusions of the live video and the display when touching them. It is also robust to minor shakes of the mobile device and eases the selection of real-world

Figure 10. Precise real-world pointing in a *magic lens* application

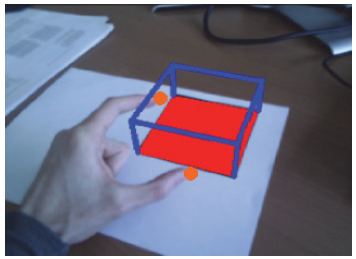

Figure 11. A *pinch* gesture in an *augmented reality* scenario

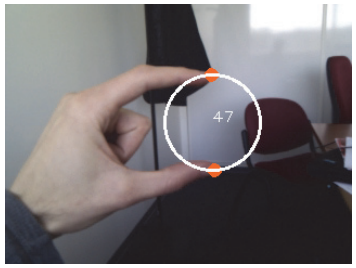

Figure 12. Gestural volume control in a mobile music player

objects which are too small to be correctly touched on the display.

Virtual Object Interaction

Similar to the interaction with real-world objects, the proposed fingertip detection may be used to manipulate virtual 3D objects which are integrated into a real-world scene by a mobile *augmented reality* application. Basic use cases include the selection of one virtual object out of several displayed ones by pointing at it. With a similar gesture or a *pinch* gesture with thumb and index finger virtual objects might be grabbed or dragged and dropped as shown in our respective demo application depicted in Figure 11. In case of animated virtual characters in mobile *augmented reality* games, the detection of the user's fingertips allows for gestural interaction with the character which may react on these gestures in different ways, e.g. by following a fingertip.

Gestural Application Control

In this use case scenario, the mobile device is not held and used as a *magic lens* but rather worn, e.g. with a lanyard around the neck with its camera facing the user's field of view. In such a scenario, the mobile device allows for a more seamless interaction serving as an unobtrusive assistant analyzing and reacting on spotted hand gestures. An example is our prototype of a gesture-sensitive music player. By forming the aforementioned *pinch* gesture and turning the wrist, the user is able to control an imaginary volume regulator and turn up or down the volume. Figure 12 shows the detected rotation gesture and the rotation angle as video overlay.

In more sophisticated hardware setups, alternative feedback channels can be utilized such as pico-projectors attached to or integrated into mobile phones. Thereby, the detection of finger gestures allows for the intuitive interaction with projected content (cf. [7]). The fingertip detection engine presented in this paper enables the first markerless solution for projector interaction in a truly mobile setup.

Conclusions and Future Work

We introduced an engine for fingertip detection in real-time which is especially targeted at mobile devices for the realization of mobile NUIs. Based on this engine, we presented several novel use cases and respective demo applications.

Concerning the use cases, we are going to complete the presented application prototypes in order to conduct early user tests comparing these novel interaction styles with established ones and identify the most promising applications from the wide range of novel interaction possibilities. Interesting research issues e.g. include the cognitive effort of using hand gestures in a mobile *magic lens* application. Future work on the detection engine will include the improvement of the engine's robustness by investigating confidence-based image segmentation methods in order to discard the use of a strict threshold. Another promising improvement is the consideration of coherence: spatial coherence of the pixels classified as skin can be exploited by using hierarchical approaches; temporal coherence of the detected fingertip positions can be assumed and hence used to reduce the search region for possible positions in the next camera frame.

Acknowledgements

This work has been carried out within the project PRIAMUS financed by FFG and A1. FTW is supported by the Austrian Government and by the City of Vienna within the competence center program *COMET*. Additional funding from the SISCom International Research Chair *Future Telecommunication Ecosystems* at Université Européenne de Bretagne Rennes and from INRIA Rennes Bretagne-Atlantique is gratefully acknowledged.

References

[1] Caballero, M. L., Chang, T.-R., Menéndez, M., and Occhialini, V. Behand: Augmented Virtuality Gestural Interaction for Mobile Phones. In *Proc. of MobileHCI '10*, pages 451–454. ACM, 2010.

[2] Erol, A., Bebis, G., Nicolescu, M., Boyle, R. D., and Twombly, X. Vision-based hand pose estimation: A review. *Computer Vision and Image Understanding*, 108:52–73, October 2007.

[3] Fang, Y., Wang, K., Cheng, J., and Lu, H. A real-time hand gesture recognition method. In *Proc. of International Conf. on Multimedia and Expo*, pages 995–998. IEEE, 2007.

[4] García-Casarrubios Muñoz, Á., Sánchez Ávila, C., de Santos Sierra, A., and Guerra Casanova, J. A Mobile-Oriented Hand Segmentation Algorithm Based on Fuzzy Multiscale Aggregation. In *International Symposium on Visual Computing*, pages 479–488. Springer, 2010.

[5] Gustafson, S., Bierwirth, D., and Baudisch, P. Imaginary Interfaces: Spatial Interaction with Empty Hands and without Visual Feedback. In *Proc. of the 23nd ACM symposium on User interface software and technology, UIST '10*, pages 3–12. ACM, 2010.

[6] Krueger, M.W. *Artificial Reality*. Addison-Wesley, 1983.

[7] Mistry, P., Maes, P., and Chang, L. WUW - Wear Ur world: a wearable gestural interface. In *Ext. Abstr. of the International Conf. on Human Factors in Computing Systems, CHI '09*, pages 4111–4116. ACM, 2009.

[8] Mo, Z., Lewis, J., and Neumann, U. SmartCanvas: a gesture-driven intelligent drawing desk system. In *Proc. of International Conference on Intelligent User Interfaces*, pages 239–243. ACM, 2005.

[9] Nguyen, T., Binh, N., and Bischof, H. An active boosting-based learning framework for real-time hand detection. In *Proc. of International Conference on Automatic Face & Gesture Recognition*, pages 1–6. IEEE, 2009.

[10] Peer, P., Kovac, J., and Solina, F. Human skin colour clustering for face detection. In *EUROCON 2003, Int. Conf. on Computer as a Tool*, 2003.

[11] Segen, S., and Kumar, S. Gesture VR: vision-based 3d hand interace for spatial interaction. In *Proc. of International Conference on Multimedia*, pages 455–464. ACM, 1998.

[12] US Patent & Trademark Office. Camera as input interface. Website, 2010. http://bit.ly/gTZr7a.

[13] Wigdor, D., Forlines, C., Baudisch, P., Barnwell, J., and Shen, C. LucidTouch: A See-Through Mobile Device. In *Proc. of UIST 2007*, pages 269–278, 2007.

[14] Wu, Y., Lin, J., and Huang, T. Analyzing and capturing articulated hand motion in image sequences. *Transactions on Pattern Analysis and Machine Intelligence*, pages 1910–1922, 2005.

[15] Xu, Y., Gu, J., and Tao, Z. Bare Hand Gesture Recognition with a Single Color Camera. *In Proc. of International Congress on Image and Signal Processing*, pages 1–4. IEEE, 2009.

[16] Yin, X., and Xie, M. Finger identification and hand posture recognition for human-robot interaction. *Image and Vision Computing*, 25(8):1291-1300, 2007.