파지 형태 인식을 통한 휴대 단말용 사용자 인터페이스 설계

장욱, 김기응, 이현정, 조준기, 소병석, 심정현, 양경혜, 조성정, 박준아
삼성종합기술원 인터랙션 랩
E-mail: {wook.chang, kekim, hyem.lee, handle.cho, bssoh, junghyun.shim, gh727.yang, sung-jung.cho, joonah}@samsung.com

Designing Mobile User Interface with Grip-Pattern Recognition

Wook Chang, Kee Eung Kim, Hyunjeong Lee, Joon Kee Cho, Byung Seok Soh, Jung Hyun Shim, Gyunghye Yang, Sung Jung Cho, Joonah Park
Interaction Lab, Samsung Advanced Institute of Technology
E-mail: {wook.chang, kekim, hyem.lee, handle.cho, bssoh, junghyun.shim, gh727.yang, sung-jung.cho, joonah}@samsung.com

요약 본 논문에서는 휴대 단말을 위한 새롭고 직관적인 응용 프로그램 구동 방식을 제안한다. 핵심 아이디어는 사용자가 휴대 단말을 사용할 때 자연스럽게 발생하는 파지 형태를 응용 프로그램 구동의 근거로 활용하는 것이다. 이를 위해 정전 용량 방식 터치 센서 시스템을 제작하고 이를 휴대 단말의 케이스 하부에 장착을 해 사용자의 파지 형태를 취득한다. 획득된 파지 형태의 인식을 위해 제안한 시스템에 특화된 인식기 및 전처리와 후처리 알고리즘을 개발하였다. 제안된 사용자 인터페이스 시스템의 효용성을 검증하기 위해 인식률 테스트를 수행한다.

Abstract A novel and intuitive way of accessing applications of mobile devices is presented. The key idea is to use grip-pattern, which is naturally produced when a user tries to use the mobile device, as a clue to determine an application to be launched. To this end, a capacitive touch sensor system is carefully designed and installed underneath the housing of the mobile terminal to capture the image of the user's grip-pattern. The captured data is then recognized by a recognizer with dedicated preprocessing and postprocessing algorithms. The recognition test is performed to validate the feasibility of the proposed user interface system.

Keywords Mobile terminal, capacitive touch sensing, recognition, context-awareness, motion sensing, accelerometer, support vector machine, Bayesian network

1. Introduction

As technology evolves, powerful computing power is redirected away from the desktop and into mobile devices. Consequently, users can now perform many tasks which could be done only on the desktop computing environments in the past. Therefore, now, limited interactivity of buttons and keyboards become severe problem.

In an effort to resolve this problem, several approaches have been taken in the industrial and academic fields. The classical main stream in the design of mobile user interface is to refine physical and graphical interfaces or improve the menu navigation structures while sticking to conventional input devices such as buttons, keyboards, mice, touch screens, etc. The second approach, which is relatively immature, is to totally redesign or invent an input device by employing additional sensors to increase the number of interaction channels [1–4].

The major role of input systems is to robustly deliver user's intention to the device, which is not problematic for the first case since the user's actions and the response of input systems of the devices are well defined. The second approaches, however, suffers from many technical and non-technical barriers. First, one has to first implement reliable hardware and software systems for the channel, which are mainly issues of sensing technology. Regarding another critical issue, the user acceptance of the developed interface cannot be guaranteed.

Turning to the issues regarding human-computer interaction, our hands are undoubtedly dominant tools to use our mobile devices. Therefore, touch, which is one of the five human senses and the integral feeling of various physical quantities such as pressure, texture, vibration, etc. arouse from a contact between objects, is popular alternative interaction channel. The most well-known touch-based user interface systems are touch pads and touch screens. They have been primarily used for notebook computers, tablet PCs, and PDAs, whose interaction metaphor follow that of desktop PCs.

The primary interaction metaphor of touch pads and touch screens is based on that of desktop PCs. However, there are large amounts of mobile devices which do not resemble the interface of desktop PCs, such as cellular phones and MP3 players. There are research efforts to use touch control for such systems. Buil *et al.* presented a touch-controlled headphones [5]. Apple's ipod presented a new interaction scheme with touch by employing touch sensors configured as a rotary wheel. Its simple but ergonomic design makes it easy for users to search large amount of audio files and navigate through the menu structure with their

(a)

(b)

Figure 1: Grip patterns and their relevant functions (a) each tool requires its own grip-pattern. (b) each function in a multi-function mobile device requires its own grip-pattern.

Figure 2: A mobile device with GripLaunch system in idle mode.

thumb. Since late nineties, Sony has also investigated and introduced various types of user interfaces using capacitive touch sensors. [6–8]. Based on their efforts on touch user interface, they released an portable audio player VGF-AP1 June 2004. The surface of VGF-AP1 is called G-sense and is the combination of buttons and capacitive touch pads. The similar concept can be also found in Motorola's A668 cellular phone. It employs touch-enabled keypads to allow users to input handwritten characters directly on the keypad.

Although the previous academic and industrial efforts on the touch user interface allow users to enjoy increased usability with his/her mobile devices, their touch interaction schemes are based on single finger touch interaction. Due to the advent of convergence era, the authors believe that more interaction channels should be provided.

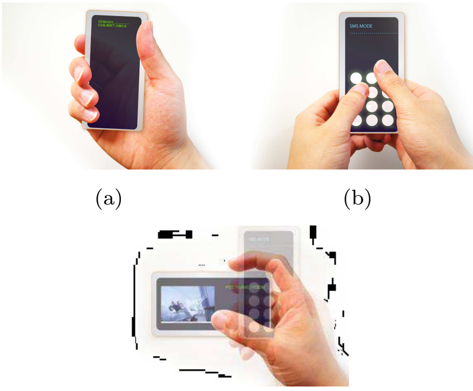
Based on the aforementioned observations, this paper aims at developing a new interaction technique for mobile devices. Among various issues on mobile human-computer interaction, the presented work focuses on accessing (launching) a specific application in a multi-function mobile device. As illustrated in Fig. 1, the proposed system utilizes grip-pattern information to provide quick and easy access to embedded applications without multi-step menu navigation.

In order to realize the proposed concept, we first develop a capacitive touch sensor system with commercially available touch sensors ICs, whose sensing portion is made by flexible printed circuit board (FPCB) to be easily installed beneath the housing of the device. In addition, the sensor system is designed to detect multi-point touch for sensing the grip-pattern. Totally 64 electrodes are arranged on the FPCB and the size of each electrode is designed to be 8 mm by 8 mm. Thus, the synthesized mobile device can sense the user's grip with sufficient resolution. Since the capacitive touch sensor has a feature of non-contact sensing, the system also detects the user's grip without revealed sensing portions on the surface of the device. The obtained sensor data is then mapped to an appropriate application by an embedded recognizer system. Since the whole surface of the device except for the display portion senses the contact of the human body, the unintentional touch can trigger the system to falsely operate, which leads to degraded reliability of the system. In order to resolve this problem, we also present systematic preprocessing and postprocessing procedures to reject signals from the unintentional touch actions and improve the recognition performance. The preprocessing is composed of two stages. In the first stage, the simple heuristic rule base is adopted to reject the noise of the sensor system. In the second stage of the preprocessing, a Bayesian network-based recognizer is trained and used to reject the unintentional touch signals. The actual recognition is done by a support vector machine (SVM) based recognizer. Finally, postprocessing is performed for the confused grip-pattern classes to more improved recognition performance.

The presented interaction method should be expressly distinguished from other touch user interfaces in that: 1) the proposed interface takes into account the grip-pattern, which has been ignored in the mobile human-computer interaction, to control mobile device; and 2) the problem of the robustness of the touch user interface to unintentional touch action is tackled in a systematic way.

2. Interaction Scenario

Before going further, this section briefs the ideal scenario of mobile human-computer interaction with the proposed scheme. Figure 2 shows a mobile device with a touch-sensitive screen and housing with no externally disclosed buttons or switches. Although the system is literally touch-sensitive, it does not respond to the unintentional touch signals. In other words, a user can hold or put the system in his/her pocket without worrying about false operation of the system. When a specific function of the device is required, the user simply takes hold of the device. The embedded recognizer classifies the sensed grip-pattern to launch an appropriate application. Selected examples are described in Fig. 3. During the use of the service, the part of the housing and screen is assigned as a control surface [9]. For example, in the camera mode, the half of the right side of the device becomes a linear slider and a switch with slight illumination for feedforward guide. Figure 4 illustrates the brief overview of the data flow for GripLaunch system.



(a) (b)

(c)

Figure 3: (a) When a user holds the device to make a call, the system automatically launches the voice dialing service (b) When a user holds the device to send a short message, the system smoothly launches the SMS composing application and the touch screen is activated for text entry. (c) A user just grasps the device to take a picture, then the device is functioned as a camera. The internal motion sensor further determines the proper orientation of the display.

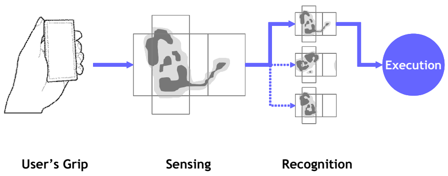


Figure 4: Data flow.

3. Implementation

3.1 Hardware

In this section, we attempt to show that the proposed concept can be implemented with current available components with reasonable size, cost and performance.

We chose the capacitive type touch sensor for sensing the grip-pattern. The main reason of our choice is that it is the only sensor that can sense user's touch beneath the housing. The touch sensor system is composed of sensing part and processing circuits.

The system employs flexible capacitive touch sensor array installed inside the housing to capture the grip image of the user. Unlike the traditional touch sensor interface, the interaction surface of the prototype is not confined within a specific region. Ideally speaking, a user just need to naturally grip the device to transfer his/her control intention to the device. Then, system automatically launch a desirable application. By using this interaction approach, the system and the user can naturally interact without traditional mechanical buttons.

The system hardware is composed of the main board and the separate sensor board. Samsung S3C2410X01 ARM-compatible CPU is used as a main processing unit. Recognition software and user interface programs are stored in the 2 MB flash memory area and loaded to 16 MB SDRAM during the execution. In the current version, the user interface is just in charge of guiding users the grip-pattern and notifying recognition results. We virtually implement a 1:2 ratio screen with two Samsung LTS180S3-H1 1.8 inch TFT LCDs. In addition, we adopt a Kionix KXM52 accelerometer. The measured motion information is used as auxiliary information for the recognizer. The touch electrode array (TEA), i.e. touch pad, is made of flexible printed circuit board and consists of 64 square electrodes so that the single flexible TEA (FTEA) is enough to make a touch-sensitive housing. The size of each electrode is 8 mm by 8 mm, which is sufficient to sense the contact of the human body through 1.8 mm thickness housing. Totally eight 8 channel ESSD SS01 touch sensor ICs are deployed and two 32 channel ADG732 multiplexors are used to transfer the sensed data to the CPU through two analogue input ports of the CPU. The CPU controls the two multiplexors with five GPIO ports. Figure 5 shows photos of the sensor board and the FTEA.

3.2 Software Algorithms

Figure 6 illustrates the structure of recognition system.

Since many electrodes are integrated into a small area and capacitive touch sensors respond to all conductive materials including the human body, the sensor system often suffers from inevitable sensor noises. The misfired signal rejection (MSR) stage is dedicated to solve this problem. MSR model is based on first-order-predicate calculus shown in Fig. (1)

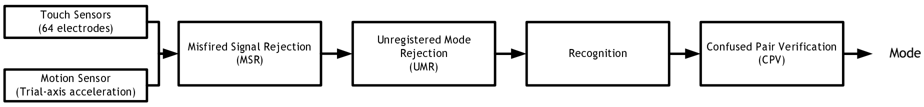
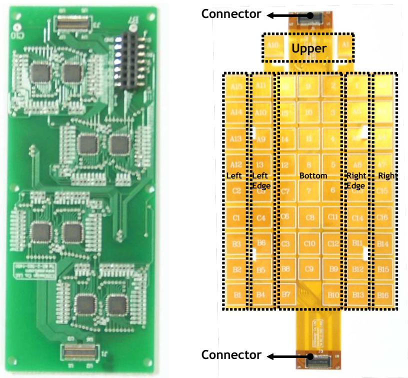


Figure 6: Structure of recognition system



(a) (b)

Figure 5: Sensor System (a) Sensor board with eight ESSD SS01 touch sensor ICs. (b) Flexible Touch Electrode Array (FTEA) printed on FPCB.

$$\text{IF} \left\{ \begin{array}{l} \text{(1) the number of fired electrodes are less than } N \\ \text{OR} \\ \text{(2) the system is just placed on the surface} \\ \text{OR} \\ \text{(3) the system is just carried by the user} \end{array} \right\}$$
$$\text{THEN reject the signal from the sensor system.} \tag{1}$$

where $N = 5$ is the threshold value.

In order to evaluate the second rule, the roll and pitch angles, which indicate the tilting of the system, are computed as follows:

$$\phi = \tan^{-1}\left(\frac{-A_{by}}{-A_{bz}}\right) \tag{2}$$

$$\theta = \sin^{-1}\left(\frac{A_{bx}}{g}\right) = \tan^{-1}\left(\frac{A_{bx}}{\sqrt{A_{by}^2 + A_{bz}^2}}\right). \tag{3}$$

where ϕ is roll angle, θ is pitch angle, A_{bx}, A_{by}, A_{bz} are x, y, and z axis outputs of the accelerometer, respectively, and $g = 9.8m/s^2$ is the gravity constant.

Therefore, the second antecedent part of (1) can be rewritten as follows:

$$(|\phi| < \phi_t) \wedge (|\theta| < \theta_t) \tag{4}$$

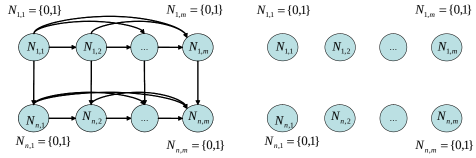
where $\phi_t = 10°$ and $\theta_t = 10°$ are threshold values of roll and pitch angles, respectively.

Another important type of signals to be filtered is signals produced by unintentional touch of the user. By the empirical observation, we concluded it is impossible to intentionally touch only the single surface or both of the lateral sides at the same time. This observation leads to the following Boolean rule:

$$\begin{array}{l} (\mathcal{T} \subseteq \mathcal{S}_1 \wedge \mathcal{T} \nsubseteq \mathcal{S}_i (i = 2,3,4,5,6)) \vee \\ (\mathcal{T} \subseteq \mathcal{S}_2 \wedge \mathcal{T} \nsubseteq \mathcal{S}_i (i = 1,3,4,5,6)) \vee \\ (\mathcal{T} \subseteq \mathcal{S}_3 \wedge \mathcal{T} \nsubseteq \mathcal{S}_i (i = 1,2,4,5,6)) \vee \\ (\mathcal{T} \subseteq \mathcal{S}_4 \wedge \mathcal{T} \nsubseteq \mathcal{S}_i (i = 1,2,3,5,6)) \vee \\ (\mathcal{T} \subseteq \mathcal{S}_5 \wedge \mathcal{T} \nsubseteq \mathcal{S}_i (i = 1,2,3,4,6)) \vee \\ (\mathcal{T} \subseteq \mathcal{S}_6 \wedge \mathcal{T} \nsubseteq \mathcal{S}_i (i = 1,2,3,4,5)) \vee \\ (\mathcal{T} \subseteq (\mathcal{S}_4 \cup \mathcal{S}_6) \wedge \mathcal{T} \nsubseteq \mathcal{S}_i (i = 1,2,3,5)) \end{array} \tag{5}$$

where $\mathcal{T}$ is the set of fired electrodes, $\mathcal{S}_1$ is the set of electrodes on the rear surface of the system, $\mathcal{S}_2$ is the set of electrodes on the upper surface of the system, $\mathcal{S}_3$ is the set of electrodes on the right edge of the system, $\mathcal{S}_4$ is the set of electrodes on the right side of the system, $\mathcal{S}_5$ is the set of electrodes on the left edge of the system, and $\mathcal{S}_6$ is the set of electrodes on the left side of the system.

Although the rule-based filtering of the noise is simple to implement, the designer should construct the whole rule-base with trial-and-error method and its optimality is not guaranteed. In order to relieve this problem, in the second stage, a simple classifier is trained deployed. This stage is called unregistered mode rejection (UMR) stage. In this paper, the unregistered mode is defined as a set of all grip-patterns except for the predefined grip-patterns for system operation. We evaluated one-class support vector machine (SVM) and Bayesian network (BN) for UMR, and finally constructed UMR system with BN as shown in Fig. 7.



(a) (b)

Figure 7: BN for rejecting unregistered modes (a) Each node has high inter-dependency (registered mode) (b) Each node has low inter-dependency (unregistered mode).

Specifically speaking, the measured signal is allowed to pass the UMR stage if the score of the BN for registered mode (BN_R) is higher than that for unregistered mode (BN_U).

$$Score(BN_U) \geq \max_i Score(BN_R^i), \quad (i = 1, 2, \cdots, N) \tag{6}$$

where N is the number of registered modes.

The SVM-based recognizer actually classifies the measured grip-pattern signals. The recognition model of the relationship between the grip-pattern I and the relevant

Figure 8: Example of confused pairs

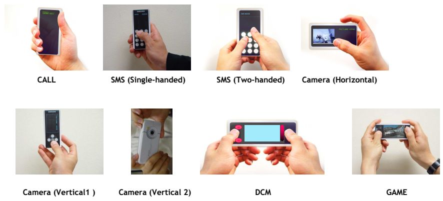


Figure 9: Examples of graip-pattern for data collection.

function M is represented as probability $P(M|I)$ and proportional to (7) according to Bayes' rule

$$P(M)P(I|M) \tag{7}$$

where $P(M)$ is the probability of selecting an arbitrary mode M and can be calculated as (8):

$$P(M) = \frac{Y}{X} \tag{8}$$

where X is the counted number of grips for a mode M and Y is the number of counts that the function M is selected when the user holds the system.

$$P(I|M) = \prod_i P(I_i|M) \tag{9}$$

where $P(I|M)$ is the probability of a specific grip-pattern I for an arbitrary function M, I_i is information from the touch sensor and the accelerometer. The mode selected by the SVM satisfies the following equation:

$$\arg\max_{m\in M} P(m|I) = \arg\max_{m\in M} P(m)\prod_i P(I_i|m) \tag{10}$$

where M is the arbitrary function or mode of the system, m is the specific mode.

In the final stage, the confused pair verification (CPV) is performed for better recognition result by using optimal rejection algorithm [10]. The CPV process is necessary since similar grip-patterns might exist even in the well designed pool of grip-patterns as shown in Fig. 8.

4. Experimental Results

For the training of recognizers, the authors collect data from 50 users. During the data collection, the developed system was attached to a PC through the RS232C serial interface cable. The sensor signals are captured and transmitted to the PC and save to the database. Users were guided to take hold of the device according to examples shown in Fig. 9. The services or situations for the defined modes include call, one-handed SMS, two-handed SMS, camera function with horizontal pose, two camera functions with vertical pose, digital contents management (DCM), gaming, and unregistered mode. For the unregistered mode, users are asked to hold the device naturally. Five grip-pattern data sets was collected for each of nine modes from 50 users.

The sensor data is composed of 64 binary touch sensor outputs and three analogue accelerometer outputs. The data sampling rate is set to 30 Hz.

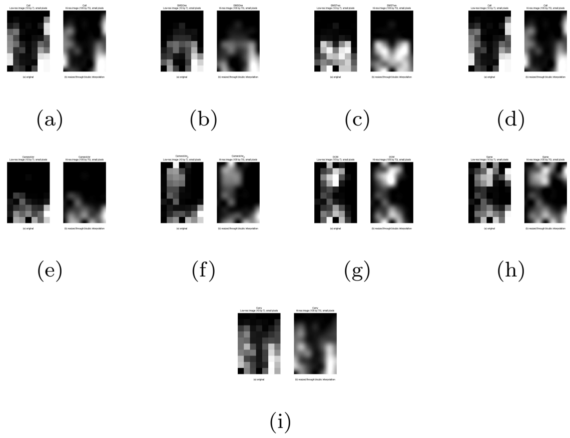


Figure 10: Averaged images of grip-patterns (a) Call (b) One-handed SMS (c) Two-handed SMS (d) Camera (horizontal) (e) Camera (vertical 1) (f) Camera (vertical 2) (g) DCM (h) Game (i) Unregistered.

Before training recognizers, we qualitatively evaluate the reliability of the sensor system by displaying the captured grip-pattern. Due to the low resolution of the capture images grip-patterns, we resize the images by using bicubic interpolation algorithm [11, 12]. Averaged images of each mode are shown in Fig. 10, which clearly indicate that the performance of the sensor system is acceptable.

The recognizer training and testing is performed on the PC environment. The performance of recognizer system is evaluated by dividing the data set of 50 users into four partitions according to users. At the first time, The first three subsets are used to train recognizers and the remaining set is used for testing. Secondly, the next three subsets and the remained set are used for training and testing, respectively. In this way, four different configurations of training and test sets are constructed to measure the user-independent recognition rate. The recognizers use averaged value of the sensor data of the last 10 samples from the collected grip-pattern database. In the recognition test, MSR stage is omitted since the collected data base does not contain unintentional touch signals. The averaged recognition rate of all the user is 93.8 %. From the recognition test, the authors claim that the technical feasibility of implementing the proposed graip-pattern-based user interaction scenario is validated. The trained recognizer is then ported to the developed system for stand-alone operation, where the MSR stage is implemented. The final stand-alone system is currently used as a demo system of our interaction scheme.

5. Conclusions

In this article, we presented an innovative mobile user interface based on grip-pattern recognition. The working implementation of the proposed system is constructed and its feasibility is evaluated through recognition rate test. The key contributions of this paper can be summarized as follows:

1. It adopts the capacitive-type touch sensing technique to capture the image of the user's grip-pattern. Due to the non-contact sensing feature of the capacitive-type touch sensors, the housing of the device itself works as a sensing mean and the resulting user interface does not require separate control surface. In addition, the single, flexible touch pad allows the sensor system to be easily installed inside the mobile device.
2. It utilizes improved pre-processing and post-processing algorithms to avoid unintentional execution of the device. The suggested structure of the recognizer effectively prevents users from suffering from unexpected operations of the device and eliminates the need of using the *hold* button.
3. Unlike most commercial and/or academic works on the mobile touch-based user interface, it addresses the effective way of utilizing multi-touch information. The grip, which is the inevitable action in using mobile devices, seamlessly integrated in the flow of interaction between the user and the device.

Finally, the authors admit that there are plenty amount of industrial and research works on capacitive touch sensing techniques, touch-based user interfaces, and bare-hand human-computer interaction schemes. However, it is worth while to note that the proposed system is one of the very few systems that offer an systematically organized framework to fusing those three items for the mobile human-computer interaction.

References

[1] G. Abowd, E. D. M. T., and Rodden, "The human experience," *IEEE Pervasive Computing*, vol. 1, no. 1, pp. 48–57, Jan.-March 2002.

[2] K. Hinckley, J. S. Pierce, M. Sinclair, and E. Horvitz, "Sensing techniques for mobile interaction," in *UIST*, 2000, pp. 91–100. [Online]. Available: citeseer.ist.psu.edu/hinckley00sensing.html

[3] K. Hinckley, J. Pierce, E. Horvitz, and M. Sinclair, "Foreground and background interaction with sensor-enhanced mobile devices," *ACM Trans. Comput.-Hum. Interact.*, vol. 12, no. 1, pp. 31–52, 2005.

[4] K. Hinckley and M. Sinclair, "Touch-sensing input devices," in *CHI '99: Proceedings of the SIGCHI conference on Human factors in computing systems.* New York, NY, USA: ACM Press, 1999, pp. 223–230.

[5] V. Buil, G. Hollemans, and S. van de Wijdeven, "Headphones with touch control," in *MobileHCI '05: Proceedings of the 7th international conference on Human computer interaction with mobile devices & services.* New York, NY, USA: ACM Press, 2005, pp. 377–378.

[6] J. Rekimoto, H. Oba, and T. Ishizawa, "SmartPad: a finger-sensing keypad for mobile interaction," in *CHI '03: CHI '03 extended abstracts on Human factors in computing systems.* New York, NY, USA: ACM Press, 2003, pp. 850–851.

[7] J. Rekimoto, "ThumbSense: automatic input mode sensing for touchpad-based interactions," in *CHI '03: CHI '03 extended abstracts on Human factors in computing systems.* New York, NY, USA: ACM Press, 2003, pp. 852–853.

[8] J. Rekimoto, T. Ishizawa, C. Schwesig, and H. Oba, "PreSense: interaction techniques for finger sensing input devices," in *UIST '03: Proceedings of the 16th annual ACM symposium on User interface software and technology.* New York, NY, USA: ACM Press, 2003, pp. 203–212.

[9] M. Gold, J. T. Chahin, J. Clark, A. Ledin, and M. Schwirn, "Recent innovations in user interface," SRI Consulting, Tech. Rep., July 2004.

[10] C. Chow, "On optimum recognition error and reject tradeoff," *IEEE Transactions on Information Theory*, vol. 16, no. 1, pp. 41–46, Jan. 1970.

[11] J. Rekimoto, "SmartSkin: an infrastructure for freehand manipulation on interactive surfaces," in *CHI '02: Proceedings of the SIGCHI conference on Human factors in computing systems.* New York, NY, USA: ACM Press, 2002, pp. 113–120.

[12] P. Bourke. (2005, May) Bicubic interpolation for image scaling. [Online]. Available: http://astronomy.swin.edu.au/ pbourke/colour/bicubic/