

PhonePoint Pen: Using Mobile Phones to Write in Air

Sandip Agrawal
Dept. of ECE, Duke University
Durham, NC, USA
sandip.agrawal@duke.edu

Ionut Constandache
Dept. of CS, Duke University
Durham, NC, USA
ionut@cs.duke.edu

Shravan Gaonkar
Dept. of CS, UIUC
Urbana-Champaign, IL, USA
gaonkar@ieee.org

Romit Roy Choudhury
Dept. of ECE, Duke University
Durham, NC, USA
romit@ee.duke.edu

ABSTRACT

The ability to note down small pieces of information, quickly and ubiquitously, can be useful. This paper proposes a system called PhonePoint Pen that uses the in-built accelerometer in mobile phones to recognize human writing. By holding the phone like a pen, an user should be able to write short messages or even draw simple diagrams in air. The acceleration due to hand gestures can be converted into an image, and sent to the user's Internet email address for future reference. We motivate the utility of such a system through simple use-cases and applications, and present the design and implementation challenges towards a functional prototype. Our early results show that the PhonePoint Pen is feasible if the user is restricted to a few simple constraints.

Categories and Subject Descriptors

C.3 [Special-Purpose and Application-Based Systems]: Real-time and embedded systems, Signal processing systems; H.4.3 [Information System Applications]: Communications Applications—*Information Browsers*; H.5.2 [Information Interfaces and Presentation]: User Interfaces—*Input devices and strategies*; J.7 [Computers in Other Systems]: Consumer Products, Real time

General Terms

Design, Experimentation, Human Factors, Measurement

Keywords

Writing in Air, Mobile Phones, Accelerometers, Input Device, Gestures, Strokes, Handwriting, Sketching, Electronic Pen, Mobile Computing Applications

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

MobiHeld'09, August 17, 2009, Barcelona, Spain.

Copyright 2009 ACM 978-1-60558-444-7/09/08 ...\$10.00.

1. INTRODUCTION

Imagine this scenario. A person parks her car in one of the many levels of an airport parking lot. While rushing to catch her flight, she glances towards the ceiling, and catches the parking lot number – “Level 5, Row A”. Walking briskly with her luggage in one hand, she takes out her mobile phone using the other. Holding the phone like a pen, she writes the word “5A” in air, and puts the phone back into her pocket. When inside her flight, she checks her email one last time before turning off the phone. An email in her mailbox says “PhonePoint Pen – 5A”. She is now assured to remember her parking lot number when she returns to the airport a week later.

The above is a fictional scenario, however, representative of a niche in the space of mobile computing applications. To be specific, we believe that there is a class of applications that will benefit from a technology that can ubiquitously and quickly “note down” short pieces of information. Although existing technologies have made creative advances towards this direction, deficiencies remain. We discuss some of these deficiencies, and motivate the potential of PhonePoint Pens.

(1) Typing an SMS, while popular among the youth, has been unpopular among a large section of society. Many studies report user dissatisfaction with mobile phone typing [1, 2, 3]. The major sources of discomfort arise from small key sizes, short inter-key spacings, and the need for multi-tapping in some phone keyboards. With increasingly smaller phones, keyboard sizes will decrease, exacerbating the problem of physical typing¹.

Even if keyboard innovations improve the typing experience (virtual keyboards [4]), problems may still arise while noting information on the fly. While driving or walking fast, or while having one hand occupied, typing information may not be feasible. Using the mobile phone accelerometer to capture hand gestures, and carefully laying them out in text or geometric images, can be useful. The utility may be greater

¹One may argue that a large cross-section of people seen in airports seem to be comfortable with typing. However, this population may not be a valid representation. Frequent flyers are perhaps in greater need for continuous communication, and hence, tolerate the difficulties of typing.

due to the *always-with-me* nature of phones, and their seamless connectivity to the Internet.

(2) Keyboards do not permit drawing. While styluses have been the proposed approach, they have not been successful because they are tedious to pull out and push back after every usage. Moreover, the palette for drawing is limited by the phones display, making the interface unattractive. The ability to draw on air, with natural movements of the hand, can circumvent these problems.

(3) One may argue that voice recorders may be an easy way to audio-record “5A”. The burden of carrying an additional device, especially when the need for recording is infrequent, deems them impractical. Voice recordings on the phone may solve this problem. However, browsing the content of multiple voice messages is time-consuming, in addition to its limitations in conveying diagrams.

(4) Current approaches are largely ad hoc. People use whatever is quickly reachable, including pen-and-paper, sticky notes, one’s own palm, etc. None of these scale because they are not always handy, and more importantly, not always connected to the Internet. Thus recorded information gets scattered, making information search and retrieval hard.

This paper proposes to use the in-built accelerometer in modern mobile phones as a quick and ubiquitous way to capture (short) written information. This may include handwritten text or simple diagrams. Even if an user writes in air without visual feedback of what she is writing, we find that the distortions in the alphabets are not excessive². Our early experiments demonstrate that when a random user writes different words, other people are easily able to read the image without knowing what the user intended to write. This confirms the legibility of gesture based images, even when character recognition is not executed on the phone. The ability to email these images to an Internet location (such as the user’s email address) adds to the promise of PhonePoint Pens.

The rest of the paper presents the design challenges and proposed approaches to translate the basic idea into an usable system. Early results are encouraging, but more importantly, they point to a range of new directions in this emerging field of mobile phone-based human-centric applications.

2. USE CASES

We envision a number of use-cases that can benefit from a PhonePoint Pen.

Sketching: Phone conversations often suffer from the lack of visual explanations (the way blackboards are useful in face to face discussions). PhonePoint Pens may be convenient in sketching very simple figures and exchanging them instantaneously between the caller/callee. While visiting a restaurant in a foreign country, drawing out a desired food item in air and showing it on the phone’s display, may also be useful.

Mashing with Cameras: While attending a seminar, imagine the ability to take a picture of a particular slide, and scribble

²Horizontal alignment among alphabets can also be achieved through simple software corrections.

related notes in air. The phone can superimpose the text on the slide, and email it to the user. Photos may similarly be captioned immediately after they are taken.

One-handed Use: People often have one of their hands occupied, perhaps because they are carrying a suitcase, a baby, or holding onto the rails in a moving train. Some people with physical disabilities may find it difficult to move one of their hands. Even while driving, it is strongly recommended that the driver avoids typing into their phones. The PhonePoint Pen allows one-handed actions, much like the benefits of a real pen/pencil.

These are few of the personal applications that do not rely on sophisticated handwriting recognition. If handwriting can be recognized, then user-generated content can be transformed into globally readable and searchable text. Clearly, a broader class of applications can be enabled. However, the first step is to validate the feasibility of a phone pen using commodity mobile phones and simple signal processing algorithms.

3. DESIGN CHALLENGES

Existing devices, such as the Wii remote [5], have the ability to identify hand gestures with reasonably good accuracy. However, most of these devices are more resourceful in terms of hardware and battery, and several offer visual cues to their users (perhaps through a monitor or a TV). Commodity mobile phones are embedded with low-cost sensors, and are constrained by limited battery power. Additionally, while writing in air, visual cues may not be practical because the phone’s screen may be continuously moving. Coping with these constraints create non-trivial research challenges. In the following, we discuss a variety of these design challenges, and propose our approaches to address them. A basic prototype on Nokia N95 phones offers promising results, justifying the validity of our main claims. However, a publicly usable system will require considerable sophistication and extensive testing. We will discuss these issues later in section 5.

(1) The Lack of a Gyroscope

Issue: Nokia N95 phones are equipped with a 3-axis accelerometer that detects acceleration in the X, Y, and Z directions. Figure 2(b) shows an example of raw accelerometer readings on each of the 3 axes. Although accelerometers measure linear movements (along the 3 axes), they cannot detect rotation. Hence, if the human grip rotates while writing, the reference frame of acceleration gets affected. Existing devices like “Wii Motion Plus” and Airmouse employ a gyroscope to capture rotation [5, 6]. In the absence of gyroscopes in phones, avoiding the rotational motion is a primary requirement.

Proposed Approach: We begin with a brief functional explanation of the gyroscope. Consider the position of a gyroscope-enabled phone (GEP) at time $t = t_0$ in 2D space (shown in the left side of Figure 1). At this initial position, the figure shows that the GEP’s axes are aligned with the earth’s reference axes (i.e., gravity is exactly in the negative Y direction). The accelerometer reading at this position is $\langle I_x(t_0), I_y(t_0) - g \rangle$, where $I_x(t_0)$ and $I_y(t_0)$ are the instantaneous acceleration along the x and y axes at time t_0 respectively, and g is the Earth’s gravity. Now, the phone may rotate at the same physical position at time t_1 also shown in Figure 1 (right). The

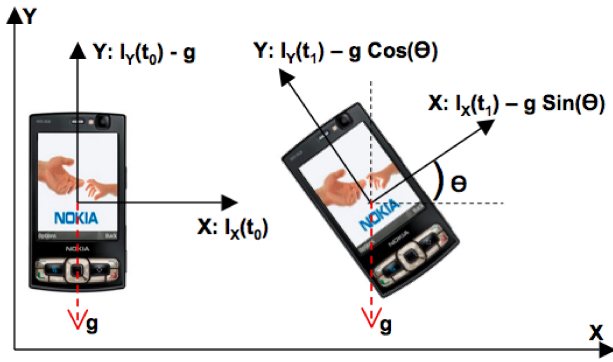


Figure 1: Earth's gravity projected on the XY axes; the axes are a function of the phone's orientation.

phone now makes an angle θ with the earth's reference frame, and the accelerometer readings are $\langle I_x(t_1) - g \sin(\theta), I_y(t_1) - g \cos(\theta) \rangle$. However, it is possible that the phone moved along the XY plane in a manner that induced the same acceleration as caused by the rotation. This leads to an ambiguity that gyroscopes and accelerometers can together resolve (using angular velocity detection in gyroscopes). However, based on the accelerometer readings alone, linear movements and rotation cannot be easily discriminated.

We have two plans to address this issue. (i) The simpler one is to pretend that one of the corners of the phone is the pen tip, and to hold it in a non-rotating grip (shown in Figure 2(a)). (ii) Alternately, while writing an alphabet, users may briefly pause between two "strokes". The pause is often natural because the user changes the direction of movement (from one stroke to another). For example, while writing an "A", the pause after the "/" and before starting the "\" can be exploited. An accelerometer reading at this paused time-point can identify the components of gravity on each axes, and hence, the angular orientation θ can be determined. Knowing θ , the phone's subsequent movement can be derived. To be safe, the user may be explicitly requested to pause briefly between two strokes. Of course, we assume that the phone rotates only in between strokes and not within any given stroke (i.e., while writing each of "/", "\", or "-"). In this paper, we have adopted both approaches.

(2) Coping with Background Vibration (Noise)

Issue: Accelerometers are sensitive to small vibrations. Figure 2(b) reports acceleration readings as the user draws a rectangle using 4 strokes (around -350 units on the Z-axis is due to earth's gravity). A significant amount of jitter is caused by natural hand vibrations. Furthermore, the accelerometer itself has measurement errors. It is necessary to smooth this background vibration (noise), in order to extract jitter-free pen gestures.

Proposed Approach: To cope with vibrational noise, we smooth the accelerometer readings by applying a moving average over the last n readings (in our current prototype $n = 7$). The results are presented in Figure 2(c) (The relevant movements happen in the XY plane, so we removed the Z axes from the subsequent figures for better visual representation).

(3) Computing Displacement of Phone

Issue: The phone's displacement determines the size of the character. The displacement δ is computed as a double integral of acceleration, i.e., $\delta = \int (\int a dt) dt$, where a is the instantaneous acceleration. In other words, the algorithm first computes the velocity (the integration of acceleration), followed by the displacement (the integration of velocity). However, due to errors in the accelerometer, the cumulative acceleration and deceleration values may not sum to zero even after the phone has come to rest. This offset translates into some residual constant velocity. When this velocity is integrated, the displacement and movement direction become erroneous.

Proposed Approach: In order to reduce velocity-drift errors, we need to reset the velocity to zero at some identifiable points. For this, we use the stroke mechanism described earlier. Characters are drawn using a set of strokes separated by short pauses. Each pause is an opportunity to reset velocity to zero and thus correct displacement. Pauses are detected by using a moving window over n consecutive accelerometer readings ($n = 7$) and checking if the standard deviation in the window is smaller than some threshold. We chose this threshold empirically, based on the average vibration caused when the phone was held stationary. All acceleration values during a pause are suppressed to zero. Figure 3(a) shows the combined effect of noise smoothing and suppression. Further, velocity is set to zero at the beginning of each pause interval. Figure 3(b) shows the effect of resetting the velocity. Even if small velocity-drifts are still present, they have a small impact on the displacement of the phone. As seen in figure 3(c) the shape drawn (intended to be a rectangle) is captured reasonably well.

(4) Need to Lift Pen from Slate

Issue: The imaginary slate in air has no global reference frame for position. While writing character "A", assume the writer has already drawn the "/" and "\", and now lifts the pen to draw the "-". Observe that the phone has no idea about the global position of "\". Hence, upon drawing the "-", the pen does not know whether it is meant to be added in the center (to indicate an "A"), or at the bottom (to indicate a triangle, Δ). This ambiguity underlies several other characters and shapes.

Proposed Approach: This is a difficult problem, and we plan to jointly exploit the acceleration along the X, Y, and Z axes. Consider the intent to write an "A". Also assume that the user has just finished writing "\". The pen is now at the bottom of the "\". The user will now lift the pen and move it towards the up-left direction, so that it can write the "-". The lifting of the pen happens in 3D space, and generates an identifiable impulse on the Z axis. When the acceleration in Z axis is above a certain threshold, we label that stroke as "lifting of the pen". This pen-lifting can be used as a trigger to indicate that the user "has gone off the record". User movements in the XY plane are still monitored for pen repositioning, but do not get included in the final output. A second impulse in the Z axis ends the pen repositioning phase and provides an indication for going back "on the record".

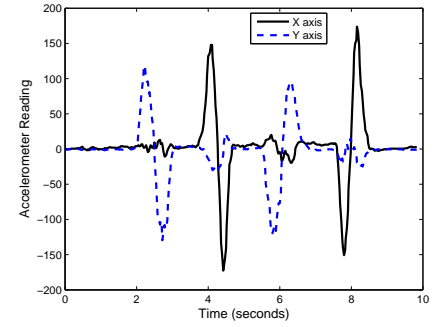
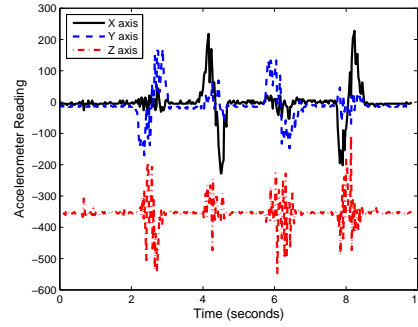


Figure 2: (a) Pretending the phone's corner to be the pen-tip reduces rotation. (b) Raw accelerometer data while drawing a rectangle (note gravity on the Z axis). (c) Accelerometer noise smoothing.

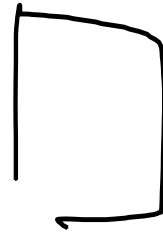
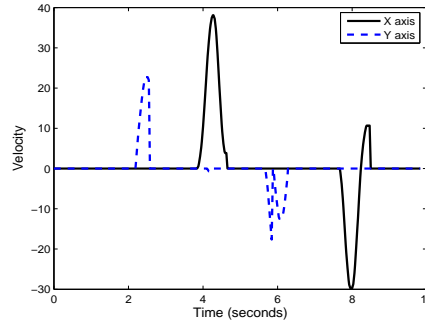
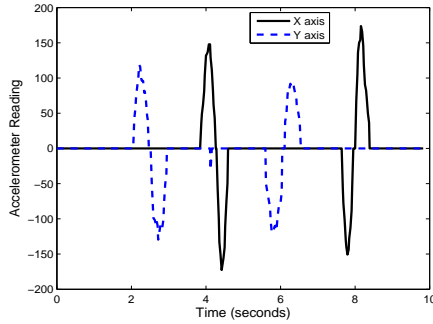


Figure 3: (a) Accelerometer readings after noise smoothing and suppression. (b) Resetting velocity to zero in order to avoid velocity-drifts. (c) The rectangle as the final output.

4. PROTOTYPING AND EVALUATION

We prototyped the PhonePoint Pen on a Nokia N95 mobile phone, equipped with a software accessible 3-axes accelerometer. A Python script, running on the phone, was used to obtain 30 to 35 instantaneous acceleration readings per second. The phone was used to draw shapes and write English alphabets in air. The accelerometer readings were processed using MATLAB scripts in the sequence of steps described in the prior section (with reference to Figures 2(b) to 3(c)).

Evaluation

We present the experimental results of the PhonePoint Pen using Nokia N95 phones. Of particular interest is the legibility of the characters. This is a qualitative metric and we intend to present a more quantitative evaluation in future. Figure 3 (c) shows the result when a rectangle was drawn in air. The figure is a reasonable representation of a rectangle. It is not a closed figure because there is no reference frame while writing on air and thus the user does not necessarily end up at the same point at which she started.

We were also able to write English alphabets. Figure 4 shows a few of them. Though the alphabets are not totally refined, they are legible. The raw accelerometer readings for alphabets "M", "i", and "h" are shown in Figure 5. Alphabet "i" is representative for the case where we need to lift the pen. We can observe that there is a significant spike in the Z axis before the 4th second when the pen was lifted off the writ-

ing plane. The system computes the movement (not shown in output image) till the 6th second, when a second spike on Z-axis triggers the pen to go back to writing mode. This accomplishes the repositioning of the phone, thereby placing the dot of the "i" in the correct position.

We evaluated how the PhonePoint pen performs from a user perspective. We asked few students to write random alphabets and words, and generated their visual representation. However, holding the phone in a non-rotating grip proved to be difficult. Even if we compensate for angular movement in between strokes, most users rotated the phone during individual strokes. As a result, the characters were deformed. In our second trial we decided to confine the user movements to one physical plane represented by a flat surface. Figure 6 shows some of the words written.

Our evaluation metric was the readability of the notes. For this, we asked both the test persons and non-users to read the notes. Both groups were satisfied with the results as the words could be distinguished easily. We think many further improvements are possible. For instance, adding word recognition techniques can significantly improve the readability of the system. Nevertheless, the results are promising for an initial prototype.

We also evaluated the energy consumption on Nokia N95 mobile phones. Our energy measurements show that contin-

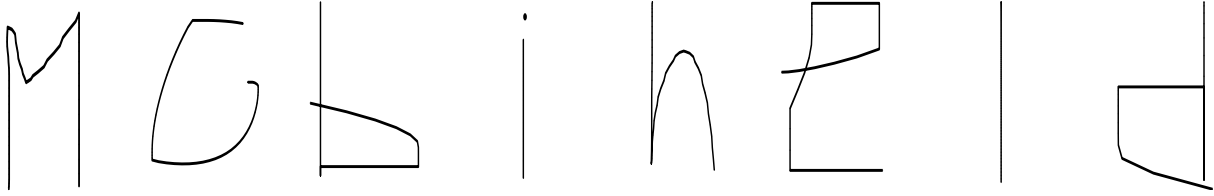


Figure 4: Alphabets M, o, b, i, h, e, l, d, as outputs of the PhonePoint Pen. Although distorted, the characters are still legible.

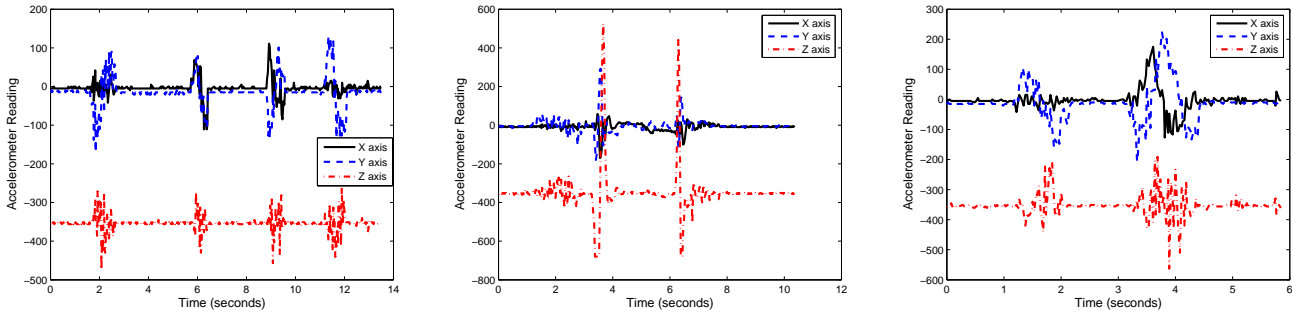


Figure 5: Raw accelerometer readings from the characters M, i, and h.

uous accelerometer readings can be sustained for more than 40 hours starting with a fully charged battery. Therefore, the energy required to write few characters is negligible. However, the ease of use may increase the frequency of use. We plan to investigate this in much greater detail in the future.

5. ISSUES AND ONGOING WORK

Building a usable PhonePoint Pen on commodity mobile phones is non-trivial. Several Issues exist between a functional prototype and a widely usable system. We discuss some of these issues, and other ideas that may be helpful in making the pen usable.

Real Time Display: The present prototype performs all the processing offline using MATLAB. We envision that the user's hand-movements on air can be displayed on the phone screen in real time. The real time feedback can enhance the user's experience and also make the system less prone to errors. We plan to address this issue in two ways: (i) Send the acceleration data to a server and require it to respond in real time with the output image. (ii) Port the accelerometer data analysis code from MATLAB to Python so that the hand-gestures can be locally processed and displayed on the phone itself. This is the main focus of our ongoing work as we believe that real time feedback adds robustness to the entire system and offers users visual feedback on what they write.

Deleting in air: A user may wish to delete a character that she has written by mistake. We plan to allow the user to choose a signature movement that indicates deletion (e.g. few brisk horizontal shakes). For every character the user writes, the phone checks whether it matches with the delete signature. If it does, the phone eliminates the previously drawn character. Deletion, in addition to real time feedback,

can enhance the user experience. A user might simply remove the alphabets that the phone did not decode correctly.

Character Recognition: As phones become more powerful, machine learning (ML) techniques may be employed to associate accelerometer readings with certain characters. Previous research [7, 8, 9] reported training and inferring simple hand gestures using accelerometers. However, writing on air needs to connect such simple gestures to form specific alphabets, digits, or figures. This may be difficult because alphabets require pen lifting and repositioning. Further, ML techniques require a gesture database, and thus, are confined to a fixed set of matching signatures. PhonePoint Pens can avoid the need for such gesture databases.

Background Movements: If a person is moving while writing with the phone, the accelerometer will reflect that movement. Movements due to driving or running, need to be eliminated in order to extract the alphabets. While rhythmic motion may be amenable to removal, arbitrary movements of the hand are harder to discriminate and prune. All these movements will be blended with the characters, distorting the desired output. We are addressing these issues as a part of our ongoing work.

Typing on Paper: The PhonePoint Pen is a quick and ubiquitous mean of jotting down small pieces of information. However, we believe that a complete system should be able to note long messages as well. We envision the use of a piece of paper as an input device for the mobile phone. The basic idea is that the user writes all the alphabets and digits on a piece of paper, points the phone camera towards the paper and types on it, much like on an actual keyboard. The phone goes first through a calibration phase in which it recognizes the characters on the paper. Then, the phone camera captures



Figure 6: Words written by 3 test persons. The words are “N95”, “730PM” and “Duke”.

which characters are obstructed by the user’s fingertips. Together, this information conveys the user input to the phone. Providing the ability to write short messages in air and type long messages on paper is part of our ongoing work towards a complete system.

6. RELATED WORK

A popular device capable of tracking hand movement is the Wii remote (or “Wiimote”) used by the Nintendo Wii console [5]. The Wiimote uses a 3-axes accelerometer to infer forward and backward rapid movements. In addition, optical sensors aid in positioning, accurate pointing, and rotation of the device relative to the ground. The optical sensor is embedded on the Wiimote and relies on a fixed reference (and a sensor bar) centered on top of the gameplay screen. The “Wiimote” can be augmented with the “Wii Motion Plus”, a pluggable device containing an integrated gyroscope. Using this feature rotational motion is captured. These three sensors – the accelerometer, the gyroscope, and the optical sensor – can reproduce motions similar to real arm-motion. The Nokia N95 consists of only a (low-cost) accelerometer, and limited processing capabilities, in comparison to the Wii. Developing the pen on this platform entails a variety of new challenges.

The Logitech Air Mouse [6] targets people who use computers as multimedia devices. The Air Mouse provides mouse-like functionalities but the device can be held in air similar to a remote control. The presence of accelerometers and gyroscopes together allow accurate linear and rotational motion of the pointer on the screen. Unlike the Air Mouse, PhonePoint Pen are not equipped with gyroscopes and relies on just the accelerometers. Furthermore, the proposed phone-based pen does not have a screen on which one may see the pen movement in real time. The absence of gyroscopes and visual cues makes positioning of the pen a difficult problem.

A series of applications for the Nokia N95 use the built-in accelerometer. The NiiMe [10] project transformed the N95 phone into a bluetooth PC mouse. The PyAcceleREMOTER [11] project developed a remote control for the Linux media player MPlayer. By tilting the phone, the play, stop, volume, fast-forward, and rewind functions of the player are controlled. Lastly many video games for the N95 phone make use of the accelerometer, e.g., to guide a ball through a maze. Being able to write in air, we believe, is a more challenging problem than the ones in existing systems.

Livescribe Smartpen [12] is a pen-like device capable of tracking a person’s writing. The device requires a special finely dotted paper to monitor movement of the pen. The pen recognizes the alphabets and numbers, and hence, can be downloaded to a PC. However, the dotted paper may not be always accessible, making ubiquitous note-taking difficult. Tablet PCs also suffer from this problem of ubiquitous accessibility.

7. CONCLUSION

We argue that the ability to quickly and ubiquitously note down information can be useful. Towards this goal, we propose a PhonePoint Pen, that uses the in-built accelerometer in mobile phones to capture human handwriting. An user should be able to hold the phone like a pen and write short messages in air. Our proposed solution translates the hand gestures into an image, and emails it to a specified Email address. Our basic prototype validates the feasibility of the pen. Our ongoing work plans to augment the pen with real time feedback, character recognition, and with better support for drawing diagrams.

8. REFERENCES

- [1] C. Soriano, G. K. Raikundalia, and J. Szajman, “A usability study of short message service on middle-aged users,” in *Proceedings of the 19th conference of the computer-human interaction special interest group (CHISIG) of Australia*, 2005.
- [2] V. Balakrishnan and P. H.P. Yeow, “A study of the effect of thumb sizes on mobile phone texting satisfaction,” in *Journal of Usability Studies*, 2008.
- [3] V. Balakrishnan and P. H.P. Yeow, “Sms usage satisfaction: Influences of hand anthropometry and gender,” in *Human IT 9.2*, 2007.
- [4] Nokia, “Virtual keyboard,” <http://www.unwiredview.com/wp-content/uploads/2008/01/nokia-virtual-keyboard-patent.pdf>.
- [5] Nintendo, “Wii Console, Wii Motion Plus,” <http://www.nintendo.com/wii>.
- [6] Logitech, “Air mouse,” <http://www.logitech.com>.
- [7] J. Mäntyjärvi, J. Kela, P. Korpipää, and S. Kallio, “Enabling fast and effortless customisation in accelerometer based gesture interaction,” in *Proceedings of the 3rd international conference on Mobile and ubiquitous multimedia*, 2004.
- [8] P. Korpipää, E. J. Malm, T. Rantakokko, V. Kyllonen, J. Kela, J. Mäntyjärvi, J. Hakkila, and I. Kansala, “Customizing user interaction in smart phones,” *IEEE Pervasive Computing*, 2006.
- [9] J. Liu, Z. Wang, L. Zhong, J. Wickramasuriya, and V. Vasudevan, “uWave: Accelerometer-based personalized gesture recognition and its applications,” in *IEEE PerCom*, 2009.
- [10] A. Arranz, “Niime,” <http://www.niime.com/>.
- [11] “Pyacceleremoter project,” <http://serk01.wordpress.com/pyacceleremoter-for-s60/>.
- [12] LiveScribe, “Smartpen,” <http://www.livescribe.com/>.