Enhancing Map Viewing on Symbian Series 60 Smartphones using Camera as 2D Input Device

AH KUN, LEONARD MARTIN
and
LEE, KUO-HUNG

This paper describes a simple implementation of a map viewing application for mobile smartphones that is enhanced by using the camera as a 2D input device. Due to the small screens and tedious input devices found on most mobile phones, this research intends to give users an insightful way of navigating through maps on their mobile phones using a new input technique. The idea is to use your mobile device as a peephole /virtual window [3] to a large virtual workspace. An evaluation of this implementation was conducted and the results presented here. The results reject our hypothesis that our new input technique would make navigating maps on small screens easier. However, this result was due mainly to the input mechanism not being responsive enough. The device that this implementation was tested on was an entry level smartphone and thus testing was done using the poorest available camera and slowest processor available. Using higher-end smartphones with higher quality cameras and faster processors could prove to provide better results.

Categories and Subject Descriptors: HCI, Mobile Devices, Map Viewing, Small Screens
General Terms:
Additional Key Words and Phrases: Peephole, Virtual window, Image Correlation

1. INTRODUCTION

Maps are useful to help pinpoint your location and destination, and are often needed when exploring a new area. However, it can be frustrating to have to constantly carry a map book with you. As most people own a mobile phone, using it to store and view maps seems to be an ideal solution.

Mobile phones are limited with respect to their input and output hardware. The input devices found on mobiles phones are often tedious to use as the buttons are often too small and the small navigation joystick often responds incorrectly to a user s actions. The small screens that display information are also often too small to view all necessary information at once. This necessitates navigation through information using the joystick, which can be frustrating.

Most modern mobiles have a built-in camera. The camera is essentially another type of input device which is typically used to capture video or static images. Alternatively, we could use the camera as a motion input device. Motion analysis in computer vision is well-studied with numerous applications. We therefore propose to use motion analysis to analyse the images captured by the mobile phone s camera to estimate the physical motion of the device. This estimation can be interpreted as an input to be used for various applications. Our aim is to use this input technique as a method to pan/scroll through a map.

2. BACKGROUND

Various techniques have been attempted for the motion analysis. Techniques discussed in previous research papers have proposed various ways of detecting motion of mobile devices using the camera input. Due to the lack of performance [4] [6] and background texture limitations [8] [7] [4], these techniques used in past research are unacceptable.

Standard image processing algorithms [6] [5] [2] considered most suitable were, edge detection [2] and the optical flow algorithm [6]. Whilst these techniques are more stable and robust, they require extensive processing power to calculate the motion of the mobile phone in real time. Due to the limited resources available on the handset, these techniques were deemed unsuitable for the purpose of this project.

Using pixel correlation [5] is more efficient than the two techniques discussed above. It works by comparing pixel blocks in adjacent frames to detect movement. The problem with using this technique is that the accuracy is affected by noise in the captured image and is also very sensitive to changes in the image. However, its efficiency makes it the most acceptable approach for this application.

3. IMPLEMENTATION

3.1 Map Viewer Application

To handle the loading of maps of any size, the map consisted of an array of constant size images. At any time, 9 of these images were cached into memory and swapped/replaced when necessary. The images contain overlapping data. As a result, only one of nine images in the cache needs to be displayed at any time. During image swaps, calculations of the relative positions of the image were computed.

The Halo [1] technique was implemented to mark off-screen locations. This technique gives the user a general idea of where a location is situated at a zoomed-in view.

3.2 Movement Detection Engine

A smoothing filter [5] is applied to the captured image to reduce the noise produced by the low quality camera on the handset. The smoothing filter works by applying a convolution mask. This blurs the image and as a result eliminates noise in the image.

There are multiple sets of movement output from the movement analysis engine when multiple search areas are used. The motion output module takes this input, analyse it and find the most suitable output from these inputs. It aggregates individual movement calculations to determine the most common movement from all the inputs. The actual movement value is then calculated as the average value for all the inputs which have the same movement directions.

A comparison between current movement value and previous movement value is performed to reduce the chance of false direction detection. If the movement values between two frames are different, the output value gets set to zero. This modification to the output causes the movement detection output to pause for one frame if the direction is changed, but it eliminates the output errors caused by hand jolt and the randomly occurred false movement detection mentioned at the beginning of this section.

3.3 Usability Test

A series of usability experiments were performed on the movement detection engine to determine: the most suitable background texture; the amount of search areas to be sampled and the moving speed one should use.

Usable background texture is an important factor to the usability of the application. For the image correlation method used in the application, a lower returned value means higher background similarity. If the image correlation calculation is performed on the background to return the image correlation average, the result can identify how self-similar the background is. The algorithm was then evaluated by performing phone movements on background evaluated as excellent , good , usable and poor to determine how the application respond to the movement.

For backgrounds evaluated as poor, the movement detection engine produces mostly false movement. In some cases, it produced correct results from the poor background, but it is very rare (about 5%). For backgrounds rated as usable, movement direction detected is roughly correct, but it sometimes (about 20%) gives false detection. Movements in good and excellent backgrounds are mostly detected correctly false detections are rare. However, after this experiment, it is believed that users only need to be notified if the background is usable. Therefore, a flag on the interface to show if the background is good/bad should be sufficient.

Search area size could affect the accuracy and usability of the system. A usability test was performed on the system with small search areas and large search areas to determine the area size most suitable for the map viewing application. For a smaller search area used, the number of search areas will increase, and vice versa.

There were three sets of search areas used in this experiment. The first set contains 3 search areas: each search area is a 40x40 pixels block, and each search pattern is sized 16x16 pixels. The second set contains 7 search areas: each search area is sized at 16x16 pixels, and each search pattern sized at 6x6 pixels. The third set contains 11 search areas: each search area sized at 10x10 pixels, and the search pattern for each search area is 4x4 pixels.

The experiment is performed by moving the phone 50cms at different rates. If the movement detected by the movement detection engine shows the correct movement during the movement, then the result is correct.

The experiment shows that systems with 3 sets of 40x40 search areas do not respond well to movements. The rate of processing data is about 800ms per frame. This is very slow and not usable. For the other two sets of search areas, the impacts of various search area sizes do not affect the result. However, because of the higher image processing rate for small search areas, the map movement in the map viewing application seem smoother.

Having settled on screen area and pattern size, we now wanted to test the effect of speed. The moving speed of the camera affects the accuracy of the movement detection. An experiment was performed to determine the ideal moving speed for operating the system. In this experiment, the phone was moved at four different speeds approximately, 0.005m/s, 0.1m/s, 0.2m/s and 0.5m/s.

The experiment for moving speed shows that for speeds between 0.05m/s (travel 1 meter in 20 seconds) and 0.2m/s (travel 1 meter in 5 seconds), the application can recognise the movements without returning false movement detection. As the moving speed increases, (as the speed hits 0.5m/s), about 30% of the movement detection is incorrect. As the speed keeps increasing, the image captured by the camera becomes totally distorted.

From the result of the usability evaluations, it was demonstrated that the background evaluation module does produce useful information to the users about the background usability. Smaller search areas produce better results and make the map scrolling smoother. The moving speed is limited to a lower speed, however if the speed increases, the image will become distorted and become unusable.

4. EXPERIMENT

Having configured the settings of the motion detection algorithm, it remained to test the usability of the camera input map viewer when completing a set of map search tasks. The performance of the camera-based system was compared with a key-input system on the same handset.

4.1 Hypothesis

Our hypotheses were:

- *our camera input device supports faster map navigation than the conventional interface*: we expect that our new input technique would be intuitive and simple to use when following paths with our map viewing application.

- *our map viewing application could be used in a real world situation:* we expect that given locations of places on the map, a real path can be found to go to that location.

4.2 Conditions

The device used for this evaluation was the Nokia 6600. This is an entry-level Symbian Series 60 Smartphone for which we developed our application. The independent variable that was manipulated in the evaluation was the interface type: either the camera input device or the standard joystick.

4.3 Subjects

24 subjects took part in our user experiment. Each subject was an undergraduate or postgraduate student. 6 of the 24 subjects were female and 18 male. All subjects owned and used a mobile phone on a daily basis. None had previously used our camera input device. None of the subjects have used mapping software before.

4.4 Tasks

The tasks consisted of two sets of three tasks. Two of these tasks were to find a route between two locations and one of these tasks was to navigate directly from a certain location to another location. Task one of a set was a short distance find route; task two, a long distance find route and task three, a long distance direct navigate. Each set was carried out by either the traditional joystick or our camera input device.

12 subjects attempted task set 1 using the camera input device and task set 2 using the joystick. The other 12 subjects attempted task set 1 using the joystick and task set 2 using the camera input device.

Before each evaluation, the user was notified about what the application was meant for and generally how to use it. The Halo technique that we implemented was also explained as we were not sure how intuitive it was. Finally, the users were also notified on which background the camera interface works best.

Before each task, the start and end locations were marked using the Halo technique. The user was then asked to find a route between the two locations using a either our camera interface or the joystick. Each task was timed individually using a stopwatch. The behaviour or subjects were also observed during evaluation.

After both sets of tasks were completed, the subjects were asked various questions about the application and the interface. The total time taken on average to complete the test was 20 minutes.

5. RESULTS AND ANALYSIS

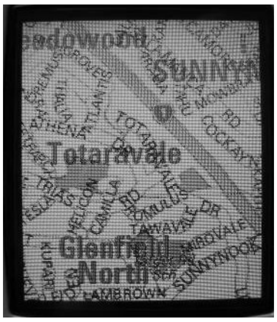

Figure 1: Map Viewing Application

5.1 Task completion times

From the task completion times the average completion times for the camera input device is always higher than that of the traditional input methods. With some tasks, the completion time of the task by some subjects using the camera input device gets close to the average completion times achieved using the traditional input mechanisms. This result seems to arise for short distance navigations (Task 1s). These results could be related to the subject s skill or knowledge of using maps.

The two sets of results from both task sets were combined for each interface used. For each type of task (i.e. short path find, long path find and long path direct navigation), the completion times for each task type were compared between interfaces (i.e. the joystick and the camera input device). Using ANOVA (Analysis of Variance between groups), we test the difference between the means of the groups of data. Table 1 (see appendix) compares the completion times of task type 1 using the interface as the changing factor. The calculated F value is larger than 1 and thus we can accept that there is a difference between the means of the two data sets. The F value calculated in Table 2 and Table 3 (see appendix) are larger and thus the difference is more distinct.

From this analysis, we can accept that there is a difference in task completion times using our new interface. From the results, we can see that the task completion time using our new interface is slower than the average time using the traditional methods. The difference in completion time increases as the distance that is navigated increases.

5.2 Observations

From the observations, the traditional input method that the majority of the subjects used was the keypad. This was largely due to the ability to pan/scroll in diagonal directions. None of the users had any problems using the traditional methods due to their familiarity.

Task 2 of task set 1 seems to be the most difficult task as there was a fork in the road where the subject had to make a decision which to choose. Since the incorrect road seemed to go more in the direction of the Halo, that road was usually chosen. The subject then had to back track once they realised that it was the incorrect choice.

The Halo technique seemed to be understood by most subjects. As we did not want any confusion to affect the results, we ensured that the subject understood the Halo technique before conducting our evaluation.

Initially, when subjects used the camera input device, it was commonly observed that users typically hold the mobile in a particular way. Holding the mobile this way naturally positions the subject s index finger in front of, or close to, the camera. When this was observed, we notified the subject which caused them to hold the mobile correctly , keeping a clear view of the camera.

Much frustration was observed when users used the camera input technique. This was observed mostly by the long distance navigation tasks. Most subjects moved their arms frantically and eventually started spinning around. The majority of the subjects also moved the device faster than the input technique was capable of detecting.

5.3 Feedback

After the completion of all the tasks, the users were asked a few questions to get some feedback about our map viewing application and our new camera input technique.

The first question that was asked, was which of the input techniques was easier to use. Most subjects preferred the traditional joystick. This was due to the familiarity of the input technique. Some subjects mentioned that it took less effort to use the joystick as all you have to do is push the joystick in the direction that you want to move. There were complaints about the difficulty of following a path when having to physically move the device. The difficulty in using the camera was mainly due to the unresponsiveness/sensitivity and also the fact that the subject has to physically move their arm all the time. The subjects also felt that they had more control when using the traditional input methods.

One subject enjoyed using the camera input for the short distance task. He commented that it was easier to find roads using this technique, which required only small slow movements.

In general, all the users believed that the map viewing application could be useful in a real world situation. They commented that the roads were clear to read and very detailed. The majority of the subjects liked the Halo technique to identify locations. However, some subjects found the Halo technique a bit confusing. They commented that the application needed an index of locations and the ability to zoom which they thought would improve the application usability. Some subjects commented on a possible feature to hi-light a path to the destination.

All subjects were asked if they would use the input technique in public. The majority of the responses were negative as they all felt that it looks ridiculous to wave your hand in the air with your mobile phone.

6. DISCUSSION

These results have the following implications on our hypothesis:

- We reject our hypothesis that *our camera input device supports faster map navigation than the conventional interface*. On average, there were no results that suggested that our camera input device was faster than the conventional interface

- We accept our hypothesis that *our map viewing application could be used in a real world situation*. The tasks that were used for evaluation were simulated tasks that would be needed in real world situations. All the tasks could be completed and thus locations and routes could be found using this map viewing application.

7. CONCLUSION

Map viewing applications for mobile phones can be useful in real world situations. Many people do not always have a map book with them and it can be frustrating carrying map books around. Since many people carry their mobile phone with them, having a map viewing application on these devices can be very useful.

The input devices on these mobile phones can be tedious and frustrating to use, however, using our camera input device for this type of application can be even more frustrating. This result seems to be largely due to the camera input not being responsive enough. The camera input device responsiveness is limited to the camera s specification and the processing power supplied by the phone. The responsiveness could be improved if the image quality and frame rate were improved. These higher quality images would require more processing to detect more accurate movement and thus could improve the overall performance of camera input technique. The next generation of handset provide these features and we are hopeful that camera input may yet prove to be a viable technique.

8. REFERENCE

[1] Baudisch, P. and Rosenholtz, R. Halo: A Technique for Visualizing Off-Screen Locations . In Proc. CHI 03, pp.481-488.

[2] J. R. Parker, Algorithms For Image Processing And Computer Vision , John Wiley & Sons, Inc. 1997.

[3] Ka-Ping Yee (2003) Peephole displays: pen interaction on spatially aware handheld computers . Proceedings of the SIGCHI conference on Human factors in computing systems, pp 1-8

[4] Martin Hatchet, Joachim Pouderoux, Pascal Guitton, A Camera-Based Interface for Interaction with Mobile Handheld Computers . Proceedings of I3D'05 - ACM SIGGRAPH 2005 Symposium on Interactive 3D Graphics and Games - 2005

[5] Milan Sonka, Vaclav Hlavac, Roger Boyle, Image Processing, Analysis, and Machine Vision Second Edition, Brooks.Cole Publishing Company 1999.

[6] Rafael Ballagas, Michael Rohs, Jennifer G. Sheridan, Mobile Phones as Pointing Devices . Pervasive 2005 workshop on Pervasive Mobile Interaction Devices (PERMID), Munich, Germany, May 2005

[7] Richard Sharp. New Ways to Maximise Camera Phone Technology , Technology@Intel Magazine, July 2004

[8] Symball. Available at http://www.vtt.fi/multimedia/

APPENDIX

Completion Time Comparison

Anova: Single Factor

SUMMARY

Groups	Count	Sum	Average	Variance
Joystick	24	722	30.08333	190.4275
Keypad	24	1301	54.20833	1991.737

ANOVA

Source of Variation	SS	df	MS	F	P-value	F crit
Between Groups	6984.188	1	6984.188	6.401155	0.014893	4.051749
Within Groups	50189.79	46	1091.082			
Total	57173.98	47				

Table 1. Short Route Path Find Completion Times

Anova: Single Factor

SUMMARY

Groups	Count	Sum	Average	Variance
Joystick	24	2428	101.1667	2729.449
Keypad	24	3708	154.5	3834.087

ANOVA

Source of Variation	SS	df	MS	F	P-value	F crit
Between Groups	34133.33	1	34133.33	10.4009	0.00232	4.051749
Within Groups	150961.3	46	3281.768			
Total	185094.7	47				

Table 2. Long Route Path Find Completion Times

Anova: Single Factor

SUMMARY

Groups	Count	Sum	Average	Variance
Column 1	24	1344	56	326.087
Column 2	24	2332	97.16667	963.1884

ANOVA

Source of Variation	SS	df	MS	F	P-value	F crit
Between Groups	20336.33	1	20336.33	31.54692	1.0829E-06	4.051749
Within Groups	29653.33	46	644.6377			
Total	49989.67	47				

Table 3. Long Path Direct Navigation Completion Times