

Design and Evaluation of a Hand-held Device for Recognizing Mid-air Hand Gestures

Kyeongun Seo[†] · Hyeonjoong Cho^{**}

ABSTRACT

We propose AirPincher, a handheld pointing device for recognizing delicate mid-air hand gestures to control a remote display. AirPincher is designed to overcome disadvantages of the two kinds of existing hand gesture-aware techniques such as glove-based and vision-based. The glove-based techniques cause cumbersomeness of wearing gloves every time and the vision-based techniques incur performance dependence on distance between a user and a remote display. AirPincher allows a user to hold the device in one hand and to generate several delicate finger gestures. The gestures are captured by several sensors proximately embedded into AirPincher. These features help AirPincher avoid the aforementioned disadvantages of the existing techniques. We experimentally find an efficient size of the virtual input space and evaluate two types of pointing interfaces with AirPincher for a remote display. Our experiments suggest appropriate configurations to use the proposed device.

Keywords : Remote Control, Mid-air Pointing Devices, Hand Gesture Recognition

공중 손동작 인식을 위한 핸드 헬드형 기기의 설계 및 평가

서 경 언[†] · 조 현 중^{**}

요 약

본 논문에서는 원격디스플레이를 제어하기 위하여 공중에서 섬세한 손동작을 인식할 수 있는 핸드 헬드형 포인팅 디바이스인 AirPincher를 제안한다. AirPincher는 현재 존재하고 있는 손동작 인식 기술 중 장갑을 이용한 기술과 디스플레이 근처에 설치하는 카메라를 이용한 기술의 단점을 극복하기 위해 고안되었다. 장갑을 이용한 기술은 사용 시 매번 탈착을 해야 하는 불편함이 존재하고 설치형 카메라를 이용한 기술은 사용자와 원격 디스플레이 간의 거리에 따라 제스처 인식 효율이 상이한 단점이 있다. 이러한 단점을 극복하기 위하여 AirPincher는 사용자가 한 손으로 디바이스를 잡고 사용하도록 설계되었다. 손가락 동작은 AirPincher에 탑재되어있는 근접한 센서들에 의해 인식되어 섬세한 손가락 움직임을 인식할 수 있다. 본 논문에서는 AirPincher를 사용하는 두 가지 포인팅 방법을 제안하고 이들의 성능을 분석하기 위한 실험을 진행하였다. 실험은 AirPincher를 사용하기 위한 공중에서의 효율적인 가상 입력공간의 크기를 찾는 실험과 AirPincher를 사용한 연속 포인팅 방법과 불연속 포인팅 방법을 비교하는 실험으로 구성되었다.

키워드 : 원격제어, 공중 포인팅 디바이스, 핸드 제스처 인식

1. Introduction

A large amount of effort has focused on replacing the traditional button-based interfaces with hand gesture-aware devices to control remote displays more intuitively and interactively. The hand gesture recognition approaches are divided into two categories, i.e., glove-based and

vision-based[1]. The glove-based techniques utilize sensors embedded into gloves to read hand and finger gestures. The close proximity between hands and sensors enables recognizing various delicate hand poses and movements. However, it is cumbersome to wear gloves whenever to use[2] and sharing gloves with others could cause concern about hygiene. On the other hand, vision-based techniques utilize cameras which free users from aforementioned inconvenience. Especially, several depth cameras recently commercialized, e.g., Kinect, provide three-dimensional visualization of objects, which allows barehanded control for remote displays. A drawback of

* 이 논문은 BK21플러스사업의 지원을 받아 수행된 연구임.

[†] 비 회 원 : 고려대학교 컴퓨터정보학과 석사과정

^{**} 종신회원 : 고려대학교 컴퓨터정보학과 부교수

Manuscript Received : July 17, 2014

First Revision : December 2, 2014

Accepted : December 2, 2014

* Corresponding Author : Hyeonjoong Cho(raycho@korea.ac.kr)

these approaches is that the resolution performance of distinguishing small gestures depends on the distance between a camera and a user.

In order to alleviate those drawbacks of both approaches for utilizing hand gestures as interfacing terminology, we consider using a new form factor, named AirPincher, in combination with Kinect. AirPincher here is

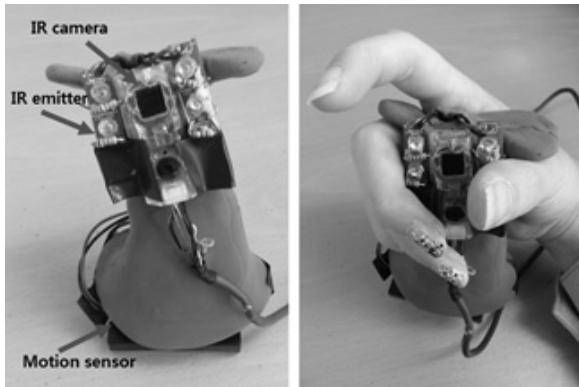


Fig. 1A. A prototype of AirPincher (left), 1B. A user hand posture to use the device (right)

a handheld device that is a slight modification of the previous apparatus in[3]. A handheld device embedded with a low-cost Infrared(IR) camera, IR emitters, and low-cost motion sensors. The proposed device, named AirPincher, allows a user to hold the device in one hand and to generate several finger gestures, e.g., pinching, rotating, etc., with a user's thumb, index and middle fingers of the holding hand. The close proximity between AirPincher and a user hand makes its resolution performance independent from the distance between a user and a remote display and allows for accommodating several delicate hand gestures. In addition, the handheld devices are one of the common form factors for remote control and thus, we believe that it incurs less hesitation for use than glove-based devices.

We apply AirPincher for controlling a cursor on a remote display. Cockburn et al. classified air pointing interfaces into three categories, such as raycasting, 2D plane and 3D volume[4]. They found that 2D plane interface where the positional movement of a user hand in an imaginary mid-air 2D plane is projected to the remote display outperforms the other air pointing interfaces in terms of speed and accuracy. Their observation motivated us to consider two types of air pointing interfaces with AirPincher based on 2D plane.

We also assume that AirPincher is used together with Kinect located by a remote display. AirPincher here is responsible of detecting delicate hand gestures while

Kinect monitors the positional transition of AirPincher.

In this paper, we experimentally find the comfortable size of the imaginary mid-air 2D plane for AirPincher. Then, we evaluate two types of air pointing interfaces with AirPincher based on 2D plane to find out their tradeoffs over various conditions. Our experiments suggest appropriate configurations to use the proposed device.

2. Related Works

While various mid-air hand gestures have been studied as moving and selecting commands, the pinch gesture is found to have a number of attractive features[5-6]. Our hand is able to quickly bring a thumb and a forefinger together and apart and additionally, the tactile feedback from our pinch fingers makes us ensure whether the pinch is performed or not. It enables precise and stable interaction[7]. The pinch gestures have been used by various gesture command systems[5, 8]. For example, Disits was a wrist-worn device for recovering the full 3D pose of a user hand. Kim et al. introduced a use case that a user initiates zooming with a mid-air pinch gesture of a hand wearing Disits.

Cockburn et al. investigated three different air pointing techniques for spatial target acquisition, i.e., raycasting, 2D plane, and 3D volume[4]. They experimentally showed that raycasting relying on small angular movement of a hand was fast but inaccurate. 3D volume was found to be expressive but slow and inaccurate. On the other hand, 2D plane was found to be appropriate as the air pointing interface in speed and accuracy.

Casiez et al. investigated control of display gain (CD gain) that affects the performance of pointing tasks. CD gain is defined as the ratio of the pointer velocity to device velocity. If CD gain is 1, the display pointer moves at exactly the same speed as the control device does. In this paper, we test different CD gains with AirPincher in order to search for an appropriate value for our purpose.

3. Airpincher

Fig. 1A and Fig. 1B show our prototype of AirPincher and the posture of a user's holding hand. The device is incorporated with a single low-cost IR camera, an array of IR emitters on both sides of the camera, and a low-cost motion sensor consisting of accelerometers, gyroscopes, and magnetometers. The IR camera and IR emitters are used to detect the pinch gesture of a user's holding hand

and the motion sensor is used to measure the angular posture of AirPincher. While a user grabs the device with the ring and little finger of his holding hand, AirPincher allows the user to generate pinch gestures with the remaining three fingers. Thus, several combinations of delicate hand gestures are permitted, e.g., pinch with a thumb and an index finger, pinch with a thumb and a middle finger, double pinch, pinch and rotating gestures, etc. The shape of AirPincher is designed to maintain IR camera to view those pinch gestures of three fingers while a user holds the device.

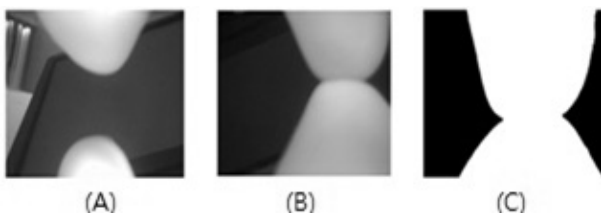


Fig. 2A. Normal state image from IR camera, 2B. Pinch state image from IR camera, and 2C. Pinch state image after binarization

Fig. 2A and 2B show a normal state image and a pinch state image captured by IR camera, respectively. IR light from IR emitters helps the fingers distinct from their background under different environmental light. After binarizing the images as in Fig. 2C, the pinch gestures are recognized by a simple post-processing.

In the post-processing, a bundle of vertical lines only consisting of white pixels are established as for Fig 2C. Based on the number of the lines that exceeds a predefined threshold, a pinch gesture occurrence is detected. The horizontal location of the bundle differentiates two pinch gestures, i.e., a pinch with a thumb and an index finger and a pinch with a thumb and a middle finger. A pinch and a double pinch gestures are distinguished using a dwell time between subsequent pinches.

To use two types of air pointing interfaces with AirPincher for 2D plane pointing interface, i.e., continuous and non-continuous. The continuous technique allows the positional movement of a user hand in 2D plane to be projected continuously to the cursor movement on the remote display. On the other hand, the non-continuous technique permits the positional movement of a user hand in 2D plane to be projected to the cursor movement only when the user is performing pinch gestures. the location of AirPincher should be tracked. The location of the devices is continuously acquired using Kinect placed by a remote display. The cursor movement on the remote

display is smoothened by Kalman filter.

4. Experiment

4.1 Experiment 1

The objective of this experiment was to determine a comfortable size of the imaginary mid-air 2D plane in which a user moves AirPincher. We recruited 8 participants including 4 women and 4 men whose ages ranged from 21 to 25. Participants were seated 3.4 meter away from a remote screen where the distance was known to be suitable for a 42 inch-sized screen we used[10]. The resolution of the screen was 1920 1080.

Each participant was asked to define their preferred input plane by double-pinchng the four corners of the imaginary mid-air plane three times[11]. The positions of the four corners were collected by our experimental setup with AirPincher and Kinect.

Table 1. Lengths of input plane sides(SD : Standard Deviation)

Input plane	Average lengths	Minimum lengths	Maximum lengths
Top side	502.9cm (SD=165)	408.4cm	597.5cm
Bottom side	537.4cm (SD=160)	445.5cm	629.4cm
Left side	394.8cm (SD=91)	342.9cm	446.7cm
Right side	412.1cm (SD=136)	334.2cm	490.1cm

Table 1 shows the experimental results. Those lengths were used to compute three CD gains for our further experiments. CD gain was calculated as the ratio of the display size to imaginary space size. According to our measurements in this experiment, the average, minimum, and maximum CD gains were determined to be 1.5, 1.23, and 1.73, respectively.

4.2 Experiment 2

The major goal of this experiment was to investigate how different CD gains affect the performance of AirPincher in terms of movement time and accuracy. We compare two pointing techniques with AirPincher, i.e., continuous and non-continuous.

We evaluated those techniques with three different CD gains which were obtained in the previous experiment. The experiment for 2D target selection was established based on the multi-directional tapping test in ISO9241-9

[12]. We also used the device assessment questionnaire to assess several subjective metrics as described in ISO9241-9. The same participants and experimental setup as Experiment 1 were reused.

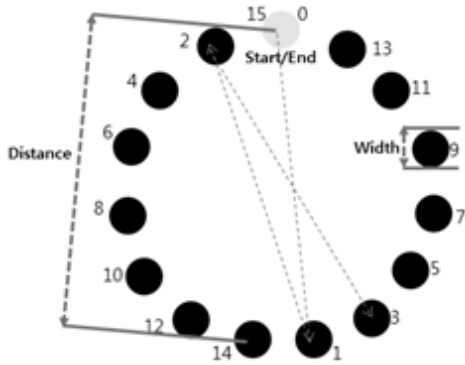


Fig. 3. Circular array of targets on the remote display for second experiment

1) Procedure

We compared the two pointing techniques over three different CD gains while adjusting the difficulty degree of the target selection task that is quantitated as the pointing task’s index of difficulty (ID)[11]. ID is determined by the width(W) of each target and the distance(D) between two farthest targets as shown in Fig. 3. In total, five ID values including 2.4 (D=430, W=100), 2.8 (D=600, W=100), 3.2 (D=820, W=100), 3.6 (D=721.5, W=65), and 4 (D=300, W=50) were tested. The total number of targets was 15. The uppermost yellow circle needs to be selected at the beginning and the end as well. The next target becomes highlighted from black to red for participants to notice. The double click gesture was used to select each target.

Participants were divided into four groups for a fully counterbalanced experiment. The first and second groups perform non-continuous and continuous pointing techniques respectively over ascending CD gain values. The third and fourth groups perform non-continuous and continuous pointing techniques respectively over descending CD gain values. Each experiment was conducted while increasing ID values. After the series of experiments finished, participants were asked to select the most preferred CD gain and to fill up the questionnaire.

2) Result

The movement time was used as a metric for evaluating the performance of the target selection task. It was defined as the time it took to move from the current target and select the next target. Fig. 4 and 5 show the movement

time of the continuous and non-continuous pointing techniques, respectively. For RM ANOVA analysis, two pointing techniques and three CD values were combined into six subgroups. The analysis indicated that they were significantly different ($F_{5,35}=9.219, p<0.001$).

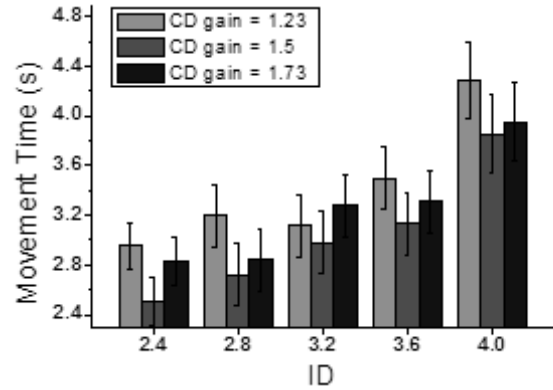


Fig. 4. Movement time of continuous pointing technique with AirPincher

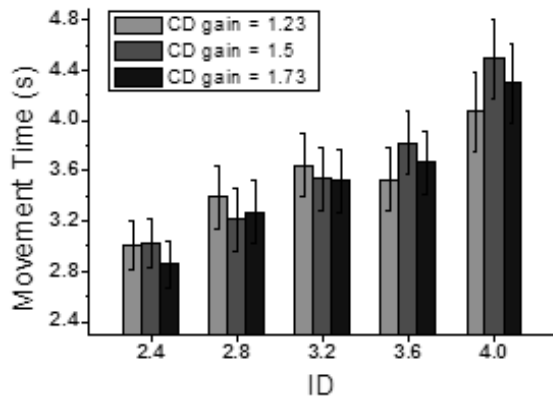


Fig. 5. Movement time of non-continuous pointing technique with AirPincher

Post hoc multiple means comparison tests revealed the significantly different subgroups in detail as follows. The continuous pointing technique with CD gain 1.23 was 0.3597 sec slower than that with CD gain 1.5 ($p<0.05$). The continuous pointing technique with CD gain 1.5 was 0.4882sec faster than the non-continuous one with CD gain 1.23 ($p<0.001$). The continuous pointing technique with CD gain 1.5 was 0.5579 sec faster than the non-continuous one with CD gain 1.5 ($p<0.001$). The continuous pointing technique with CD gain 1.5 was 0.4813 sec faster than the non-continuous one with CD gain 1.73 ($p<0.001$). The continuous pointing technique with CD gain 1.73 was 0.3757 sec faster than the non-continuous one with CD gain 1.5 ($p<0.05$). Therefore, the shortest movement time is

with the continuous pointing technique with CD gain 1.5.

The error rate defined as the number of erroneous selections over the number of the entire selections was measured to compare two pointing techniques. The continuous and non-continuous pointing techniques incurred 4.1% and 6.69% error rates, respectively. However, RM ANOVA indicated that there was no significant difference($F_{1,7}=1.068, p>0.3$).

Table 2. Result of Questionnaire(SD : Standard Deviation)

Question	Continuous	Non-continuous
Accurate pointing	3.38(SD=0.9)	3.75(SD=1.2)
Finger fatigue	3.38(SD=0.9)	2.88(SD=1.4)
General comfort	3.39(SD=0.8)	3.39(SD=0.6)

The preferable CD gain of participants was found to be 1.5. And their replies to the questionnaire about accuracy, fatigue, and comfort were summarized in Table 2. Their replies were measured in 5-point Likert scale.

5. Discussion

When the distance between an AirPincher user and the remote display was set, the experimental results showed that the continuous pointing technique at CD gain 1.5 performed better than the others in movement time. Participants replied that they felt almost the same comfort for both pointing techniques. They also replied that the continuous pointing technique with AirPincher is more effortless and less accurate than its counterpart. We suppose that it is because the non-continuous pointing technique has a clear delimiter between the pinch gesture for moving a cursor and the double pinch gesture for selection, which made the selection task slower but more accurate.

Our experimental configuration though has limitations that we only considered the absolute mapping between the imaginary input plane and the remote display in which an absolute location of AirPincher in the input plane is translated to an absolute location on the remote display. In future, the relative mapping that translates a positional transition of AirPincher to the positional transition of a cursor on the remote display will be studied.

6. Conclusion

We propose AirPincher, a handheld pointing device for

recognizing delicate hand gestures to control a remote display. The design of AirPincher aimed at recognizing several hand gestures made by the user's thumb, index, and middle fingers. AirPincher was applied to two types of pointing techniques for a remote display, continuous and non-continuous. The first experiment suggested the proper size of the imaginary input plane for the pointing techniques. Then, the second experiment showed which pointing technique is faster than the other at which CD gain. We also found that the continuous pointing technique yielded less fatigue, and lower accuracy than its counterpart.

References

- [1] Shastry, K. R., Ravindran, M., Srikanth, M. V. V. N. S., and Lakshmikanth, N., "Survey on various gesture recognition techniques for interfacing machines based on ambient intelligence," arXiv preprint, 1012.0084, 2010.
- [2] Osunkoya, T., Chern, J. C., "Gesture-Based Human-Computer-Interaction Using Kinect for Windows Mouse Control and PowerPoint Presentation," in *Proceedings of the MIC Symposium*, 2013.
- [3] Kyeongun Seo, Hyeonjoong Cho. AirPincher: A HandHeld Device for Recognizing Delicate Mid-air Hand Gestures in *Proceedings of UIST*, pp.83-84, 2014.
- [4] Cockburn, A., Quinn, P., Gutwin, C., Ramos, G., and Looser, J., "Air pointing: Design and evaluation of spatial target acquisition with and without visual feedback," *International Journal of Human-Computer Studies*, Vol.69, No.6, pp. 401-414, 2011.
- [5] Schwaller, M., Lalanne, D., "Pointing in the air: Measuring the effect of hand selection strategies on performance and effort. In *Human Factors in Computing and Informatics*," *Human Factors in Computing and Informatics*, Vol.7946, pp.732-747, 2013.
- [6] Van de Camp, F., Schick, A., and Stiefelhagen, R., "How to Click in Mid-Air," *Distributed, Ambient, and Pervasive Interactions*, pp.78-86, 2013.
- [7] Wilson, A. D., "Robust computer vision-based detection of pinching for one and two-handed gesture input," in *Proceedings of the ACM Symposium on UIST*, pp.255-258, 2013.
- [8] Kim, D., Hilliges, O., Izadi, S., Butler, A. D., Chen, J., Oikonomidis, I., and Olivier, P., "Digits: freehand 3D interactions anywhere using a wrist-worn gloveless sensor," in *Proceedings of the ACM Symposium on UIST*, pp.167-176, 2012.
- [9] Cockburn, A., Quinn, P., Gutwin, C., Ramos, G., and Looser, J., "Air pointing: Design and evaluation of spatial target

acquisition with and without visual feedback,” *International Journal of Human-Computer Studies*, Vol.69, No.6, pp. 401-414, 2011.

- [10] Lee, D. S., “Preferred viewing distance of liquid crystal high-definition television,” *Applied ergonomics*, Vol.43, No.1, pp.151-156, 2012.
- [11] Winkler, C., Pfeuffer, K., and Rukzio, E., “Investigating mid-air pointing interaction for projector phones,” in *Proceedings of the ACM Symposium on UIST*, pp.85-94, 2006.
- [12] ISO 9241-9:2000: Ergonomic Requirements for Office Work with Visual Display Terminals(VDTs)-Part 9: Requirements for Non-keyboard Input Devices. ISO Standard(2000).



서 경 은

e-mail : chemd111@korea.ac.kr

2012년 고려대학교 컴퓨터정보학과(학사)

2013년~현 재 고려대학교 컴퓨터정보학과 석사과정

관심분야: Human computer interaction



조 현 중

e-mail : raycho@korea.ac.kr

1996년 경북대학교 전자공학부(학사)

1998년 포항공과대학교 전자전기공학과(석사)

2006년 Computer Engineering, Virginia Polytechnic Institute and State University, Blacksburg VA(Ph.D)

2009년~현 재 고려대학교 컴퓨터정보학과 부교수

관심분야: Real-time computing for embedded systems, Human-computer interactions for smart devices