TypeBoard: Identifying Unintentional Touch on Force-Sensitive Touchscreen Keyboard

ANONYMITY, address, Country

用户在触屏键盘上打字时，触摸屏幕即触发点击事件。因此，触屏用户不能像在物理键盘上一样通过触摸来align fingers positions[xx]，也不能将手指休息在键盘上，影响了触屏打字的效率和舒适度。在这篇论文中，我们提出了TypeBoard，一款带压力屏键盘上的防误触算法，我们也研究了用户使用TypeBoard时的打字行为。用户的打字行为和防误触能力之间是相互影响的，比如，在防误触的触屏键盘上，用户会更倾向于将手指休息在触碰上，造成更多的、更多样的非有意触摸点；而更多的、更多样的非有意触碰点会对防误触提出更高的要求。为此，我们通过迭代的数据采集和机器学习方法来设计了TypeBoard防误触算法。在一个使用TypeBoard写日记的评测实验中，用户的非有意触点个数占触点总数的xx.x%，我们的算法能在点击事件发生100 ms的时间内以xx.x%的准确率判断出其输入意图，而相关工作中基于压力和时间阈值的方法[xx]的识别准确率只有xx.x%，且必须在release以后做出判断。这份工作说明，第一，压敏触屏键盘有能力准确地、低延迟地防止用户休息、轻触等行为造成的误触。第二，用户在有效防误触的键盘上打字时，其用户行为会发生改变，比如手指休息的行为会显著增加。第三，防误触键盘和传统触碰键盘相比，能够防止用户疲劳，提高用户体验，且显著降低了输入任务的完成时间。

CCS Concepts: • **Computer systems organization → Embedded systems**; *Redundancy*; Robotics; • **Networks** → Network reliability.

Additional Key Words and Phrases: Smart watch, text entry, touch input.

ACM Reference Format:
anonymity. 2018. TypeBoard: Identifying Unintentional Touch on Force-Sensitive Touchscreen Keyboard. *Proc. ACM Meas. Anal. Comput. Syst.* 37, 4, Article 111 (August 2018), 13 pages. https://doi.org/10.1145/1122445.1122456

1 INTRODUCTION

随着平板电脑的普及[xx]，在平板电脑上进行文本输入的需求也在增长。平板电脑用户需要使用文本输入来完成搜索、笔记等简单的功能，有的用户甚至需要用平板电脑办公。触屏软键盘是平板电脑上通用的文本输入方案，然而，软键盘在输入效率（xx-WPM[xx] vs. xx-WPM[xx]）、疲劳程度[xx]和视觉注意力负担[xx]等诸多方面上与和物理键盘存在巨大的差异。在物理键盘上打字时，用户可以将手指休息在键盘上，这有两大优势：第一，用户可以将手指休息在键盘上，减轻疲劳；第二，用户可以通过触摸按键纹理来对齐他们的手指，从而实现盲打，降低视觉注意力的占用，大大提高输入效率。目前，用户不能将手指休息在软键盘上，因为这会导致误触。在篇论文中，我们提出了TypeBoard，一款软键盘上的防误触算法，使得用户可以将手指休息在触屏上。在TypeBoard的帮助下，我们有机会通过添加膜层[xx]或可变形触屏[xx]等方案给软键盘添加触觉纹理，从而支持软键盘上的盲打，弥补软硬键盘之间的差距。

在触屏键盘上区分打字点击和“误触”并不是我们的首创。在2013年，TapBoard就曾经提出将Tapping动作看作打字点击，而将其它触摸事件视为误触。Tapping的定义是“触摸时间低于xx毫秒，位移低于xx毫米的点击”。用户需要主动去适应TapBoard的技术方案，这存在着准确率低、影响用户自然性和舒适性

Author’s address: anonymity, anonymity@anonymity.com, address, P.O. Box 1212, Dublin, Ohio, Country, 43017-6221.

2476-1249/2018/8-ART111 $15.00
https://doi.org/10.1145/1122445.1122456

等等诸多问题，我们会在相关工作中详细描述。为了克服TapBoard的局限性，在这份工作中我们从用户的角度出发，将打字点击定义为"表达键入意图的点击"。特别地，我们希望理解用户在防误触触屏键盘上自然的打字行为，并根据用户的行为模式，来设计一款鲁棒的防误触触屏键盘。

"理解用户在防误触触屏键盘上自然的打字行为"是困难的，这涉及到人机交互研究领域中一个广泛存在的问题，即交互技术和用户行为之间可能是互相影响的。这一现象在防误触算法的设计上尤为突出：一方面，防误触算法的能力会显著影响用户的行为，举例而言，用户在防误触能力强的软键盘上会将手指休息在触屏上，带来更多更有挑战性的误触点；另一方面，我们对用户行为的理解又能帮助防误触算法的设计。然而，几乎所有相关论文都忽略了这一现象，有的论文在收集用户行为数据时没有给出反馈[xx,xx,xx]，其它论文给出了替代性的反馈[xx,xx,xx]。为了弥补这个缺陷，我们采用迭代的方式来设计TypeBoard，逐步理解用户的自然打字行为、设计与行为相符的防误触算法。我们共组织了三个用户实验，分别回答了以下三个研究问题：

(1) **RQ1):** 用户在一个想象中的防误触触屏键盘上的打字行为如何？我们组织实验一收集了用户在无反馈触摸板上打字的数据，用户想象该键盘能够完美地防止误触。由于触摸板没有反馈，用户不能真的打字，而是想象字母上屏了。用户完成了多种文本输入任务，并人工标注了数据。实验一共采集了xx个数据点，其中误触点占xx%，远大于我们在正常触屏上的误触数量。我们分析了用户的行为，并开发了针对性算法，识别误触的准确率达到了xx.x%，延迟为手指落下后xx毫秒。作为比较，TapBoard[xx]在该数据集上的准确率仅为xx.x%，且只能在手指抬起时识别。
(2) **RQ2):** 用户在防误触触屏键盘上的打字行为如何？在实验一以后，我们得到了初步的防误触算法，我们组织实验二收集了用户在该防误触键盘上打字的数据。在实验二中，用户能够键入字母，同时得到声音反馈。用户完成了多种文本输入任务，并人工标注了数据。共有25名用户参与了实验二，他们被分成了五组。在每五个用户完成实验之后，实验者都会更新数据集，添加必要的特征向量，重新训练防误触算法。最终，实验二共采集到xx个数据点，其中误触误触点占xx%，显著高于实验一的误触点数量，这暗示真正的防误触算法会提高用户对触屏的信任感。在该数据集上，我们的算法的识别准确率达到了xx.x%，进一步拉大了与TaoBoard（xx.x%）的差距。
(3) **RQ3):** 防误触键盘对用户体验和效率有和影响？防误触键盘上加上纹路有何影响？ 我们组织了实验三，在多个输入任务下对比了TypeBoard和普通触屏键盘的用户体验和输入效率。结果表明，TypeBoard在写日记、填问卷等任务下显著提升了用户体验，降低了疲劳，而在誊写任务下和普通触屏键盘没有差异。实验三同时对比了在有无纹理反馈两种设置下用户的行为规律，和TypeBoard性能变化，结果发现，纹理反馈使得用户更频繁地将手指放置在触摸屏上，误触点数大大增加，在这一极端的数据集下，我们的算法准确率为xx.x%，而baseline的准确率仅为xx.x%。纹理反馈大大提升了用户的输入效率，观察实验视频我们发现，这一提升可能是因为防误触算法+纹理反馈合力使能了用户在触屏键盘上的盲打。

这份工作有三个贡献点：第一，TypeBoard准确地、低延迟地区分了触屏打字时的typing和误触。第二，在TypeBoard的支持下，我们理解并总结了用户在防误触触屏键盘上的输入行为，我们公开了该数据集。第三，我们通过评测实验证明，TypeBoard与传统触屏键盘相比，提高了输入效率，降低了用户疲劳程度。

2 RELATED WORK

2.1 Unintentional Touch Rejection on Touchscreen

Touch is the main input channel on touchscreen devices such as smartphone, tablet and tabletop, but not all contacts on the touchscreen are intended to trigger a digital response. Those touches that do not contribute to any interaction goal are known as unintentional touches [25], or namely accidental/unwanted touches [16, 17]. Since unintentional touches trigger unwanted interaction, the user's on-going behavior will be interrupted by the unintentional touch [xx]. Moreover, the user needs to spend extra time to cancel the accidentally triggered response [xx], which affects the efficiency [xx] and naturalness [xx] of the interaction [2, 25].

However, unintentional touch is inevitable in touchscreen interaction. For example, the thenar eminence on the human hand will constantly contact the touchscreen during the daily use of smartphones [xx]. Fortunately, we can identify and filter out these unintentional touches by software techniques. In the literature, the methods of preventing unintentional touches has been extensively studied. We compare existing work in two aspects: (1) the use scenario, and (2) the sensor.

2.1.1 Unintentional Touch Rejection over use scenarios. The definition of unintentional touch varies in different use scenarios. When the application is not limited, unintentional touches refer to those touches that do not contribute to any interaction goal [25]. The boundary of intentional and unintentional touches will be more clear in specific task. For examples, in the text entry task, unintentional touches are those touches that do not intended to entry words [xx].

A few studies [17, 25] and a large amount of patents [5, 6, 10, 19, 20] attempt to identify unintentional touch over applications. Metero et al. presented guidelines to reduce the amount of unintentional touches on smartphones [17]. Their filtering criteria rejected 79.6% of unintentional touches. Xu et al. identify and filter out unintentional touches on interactive tabletop using gaze direction, head orientation and screen contact data [25]. The accuracy was 91.3%. These approaches suffered from low recognition rate.

In the literature, more studies were conducted to identify unintentional touch in specific scenarios. TapBoard and TapBoard2 discussed this issue in text entry tasks [11, 12]. TapBoard regarded short-term tapping actions as keystrokes and other contacts as unintentional touches. The system reported a keystroke when the touch duration is shorter than 450 ms and the touch movement is shorter than 15 mm. Users adapted their behaviors to these thresholds, so that TapBoard achieved an accuracy of roughly 97%. Based on TapBoard, TapBoard2 was able to disambiguate typing and pointing actions with an accuracy of greater than 95%. While inking on tablets, unintentional touch resulted in a great effect on user behavior and was one of the most prominent features identified as problematic by participants [2], e.g., users were forced to write in an uncomfortable position to avoid the 'palm touch' screen. Several studies were proposed to reject unintentional touch in pen and tablet interaction [3, 9, 21]. Schwarz et al. achieved the best performance [21] by leveraging spatiotemporal touch features, reducing accidental palm inputs to 0.016 per pen stroke, while correctly passing 98% of stylus inputs. In smartphone interaction, palm touches are often considered as unintentional touches. PalmTouch used these "unintentional touches" as intentional input methods [13], such as a shortcut. PalmTouch differentiated between finger and palm touch with an accuracy of 99.53% in realistic scenarios. GestureOn enabled gesture shortcuts in the standby mode by which a user can draw a gesture on the touchscreen before the screen is turned on [16]. GestureOn acquired 98.2% precision and 97.6% recall on detecting gestures from accidental touches. The studies above explored the feasibility of rejecting unintentional touch in specific scenarios. Their performances were generally high.

2.1.2 Unintentional Touch Rejection over sensors. A mass of studies have been conducted to recognize unintentional input on touchscreen devices, including smartphones [13, 14, 16, 17], tablets [3, 10–12, 21] and tabletops [25]. Most techniques leverage spatiotemporal features of touchpoints [10–12, 17, 21] and capacitive images to identify unintentional touches [3, 13]. Metero et al. explored the feasibility of rejecting unintentional touch on smartphones using touchpoint patterns [17]. They analyzed user behavior in three typical tasks including swipe interactions in the home view, phone call interaction, and general device handling. The authors proposed filtering criteria such as touch duration, position and trajectory pattern that rejected 79.6% of unintentional touches whilst rejecting 0.8% of intentional touches. Schwarz el al. presented a probabilistic touch filtering approach that distinguish between legitimate stylus with palm touches on tablet computers [21]. The method extracted features from touchpoints and used the decision forest model, reducing accidental palm inputs to 0.016 per pen stroke, while correctly passing 97.9% of stylus inputs. PalmTouch [13] is an additional input modality that differentiates between touches of fingers and the palm. The intended palm touch supports different use cases, including the use as a shortcut and for improving reachability. PalmTouch used the raw capacitive image of the touchscreen as input and used Convolutional Neural Network (CNN) as the method, resulting in an accuracy of 99.53% in realistic scenarios. The above approaches rely on built-in sensors, and can be immediately applied to most of the existing smartphones and tablet computers. As a trade-off, they are limited by low recognition accuracy or fixed application scenarios.

Some related techniques enhanced the sensing ability to improve the performance of rejecting unintentional touches [8, 14, 16, 25]. GestureOn [16] distinguishes between intended gesture input with unintentional touches in the standby mode of smartphones. The user can trigger gesture shortcuts before the screen is turned on. GestureOn used most of the built-in sensors on the smartphone including proximity sensor, light sensor, IR sensors and Inertial Measurement Unit (IMU). The system also leverage the pressure associated with the touch event, which is not popularized on smartphones yet. Based on sensor fusion, GestureOn acquired 98.2% precision and 97.6% recall on detecting gestures from accidental touches. Xu et al. leveraged gaze direction, head orientation and screen contact data to identify and filter out unintentional touches on interactive tabletop [25]. Result showed that the patterns of gaze direction and head orientation improved the accuracy of identifying unintentional touches by 4.3%, reaching 91.3%. The above approaches used additional sensors, and unsurprisingly improved the recognition rate.

2.1.3 Summary. As table xx shows, we proposed a classification of unintentional touch rejection methods into two broad categories: *universal method* and *specific method*. We define *universal methods* as those techniques that identifies and filters out unintentional touches over different scenarios. For *universal methods*, unintentional touches refer to those touches that do not contribute to any interaction goal. We define *specific methods* as those techniques that identifies unintentional touches in specific scenarios. For example, unintentional touches in the text entry task are those touches that do not represent typing intent. As table xx shows, the recognition rate of *specific methods* are generally higher that the performance of *universal methods*. We argue that this gap is significant. As the performances of *universal methods* were not higher then 92%, the lowest achievable error rate (Bayes error rate [23]) might be high. *Universal methods* do not significantly change user's behavior, i.e., users were forced to perform unnatural and uncomfortable actions to avoid unintentional touches [2] [xx]. For comparison, the performances of *specific methods* are much higher. Users are willing to change their behavior to acquire benefits from the techniques [11, 12].

Beside use scenario, the sensing ability of the devices also affect the recognition rate of unintentional touch. Several studies have shown that additional input channel such as xx, xx and xx significantly improve the accuracy. In this work, we used a device with strong sensing ability (force-sensitive touchscreen) to identify unintentional touch in a specific task (text entry). There was no doubt that we were able to reach a high recognition rate. In this situation, we were interested in a deeper question, which is the interrelationship between technique and user behavior. On the one hand, a perfect technique of rejecting unintentional touch will enable natural and relaxing user behavior on the touchscreen. On the other hand, a perfect technique should adapt to the natural user behavior. To our knowledge, previous studies did not analyze natural user behavior on a unintentional touch rejecting touchscreen. They either collected user data in a system with no feedback [xx,xx,xx] or emulational feedback [xx]. In this paper, we introduce a iterative progress to reveal the interrelationship between unintentional touch rejection technique and user behavior.

TapBoard [12] was similar to our work. We both investigate unintentional touch rejection on touchscreen keyboard. However, TapBoard used thresholds to distinguish between typing and unintentional touch, thus users need to adapt their behaviors to the technique. This could not provide natural and relaxing user experience [xx, xx]. Moreover, TapBoard suffered from modest accuracy (97%) and high recognition delay (recognized at the touch up moment). In this paper, we cover TapBoard's shortage by considering the interrelationship between technique and user behavior.

	universal method	specific method
off-the-shelf sensors	xxx (xx.x%) [xx] xxx (xx.x%) [xx] xxx (xx.x%) [xx]	xxx (xx.x%) [xx] xxx (xx.x%) [xx] xxx (xx.x%) [xx]
additional sensors	xxx (xx.x%) [xx] xxx (xx.x%) [xx] xxx (xx.x%) [xx]	xxx (xx.x%) [xx] xxx (xx.x%) [xx] xxx (xx.x%) [xx]

Table 1. Classification of unintentional touch rejection techniques.

2.2 Benefits from Unintentional Touch Rejection

Since unintentional touch affects the efficiency and naturalness of the touch interaction, a straightforward benefit from unintentional touch rejection is to improve the user experience. Beside this, we summarize other advantages of unintentional touch rejection as follows.

2.2.1 Bridge the Gap Between Soft Keyboard and Physical Keyboard. Though touchscreen devices are getting popular in the past ten years [xx], users still prefer to use personal computers and physical keyboards in the office work [xx]. This is because physical keyboards are better than touchscreen keyboards in usability [xx] and efficiency (xx-WPM vs. xx-WPM [xx]). Users can rest their fingers on physical keyboards, which is an important factor in the usability of a keyboard [12]. First, users avoid fatigue by resting their fingers in a long text entry task. Second, users align their fingers by touching the tactile landmarks on physical keyboards. This alignment behavior enable efficient touch type.

Our work support unintentional touch rejection on touchscreens, that is, users are able to rest their fingers on the touchscreen. This improvements has three advantages. The first is avoiding fatigue. Second, our studies show that the resting behavior improved the accuracy of touch position in each keystroke, because the users' fingers got closer to the desired keys. Third, we added stickers on the touchscreen keyboard to provide tactile landmarks and found it significantly improve the accuracy and efficiency. In real use, deformable touchscreen surfaces [1, 4, 15] and tangible keyboard objects on the touchscreen [22, 24] can provide such tactile landmarks. Thus, unintentional touch rejection does bridge the gap between soft keyboard and physical keyboard.

2.2.2 Leveraging Unintentional Touch as Input Channels. The goal of our work is filtering out unintentional touches on the touchscreen. However, there are some literature that tried to "use" unintentional touches as situational awareness contexts [xx] or explicit interactions [xx]. For situational awareness, Zhang et al. leveraged unintentional touches such as the palm touch to augment pen and touch interactions, e.g., providing "Palm Tools" that teleport to a hand-centric location when the user plants their palm on the screen while drawing [26]. For explicit interactions, Matulic el at. enriched tabletop interaction by considering whole hands as input [18]. The algorithm detects seven different contact shapes with 91% accuracy and can be used to trigger, parameterise, and dynamically control menu and tool widgets. PalmTouch is an additional input modality that differentiates between touches of fingers and the palm [13]. PalmTouch can be used as a shortcut for improving reachability. TapBoard2 [11] distinguishes typing from pointing using a threshold method, thereby unifying the keyboard and mouse control spaces. This depresses the burden of frequently switching between devices [7].

3 STUDY 1: USER BEHAVIOR ON IMAGINARY TYPEBOARD

In this study, we collected data from participants' typing actions on a force-sensitive touchpad. The motivation was to investigate users' typing behavior on an imaginary touchscreen keyboard that can "perfectly reject unintentional touch", based on which we can design the algorithm of rejecting unintentional touch. Because (incorrect) feedback will affect users' behaviors, in this study, participants typed on a touchpad without any feedback. Participants could not enter words by typing on the touchpad, instead they imagined that the desired words are entered. Besides, participants need to adapts their typing behaviors according to the imagination that the keyboard can perfectly reject unintentional touch.

3.1 Participants

We recruited 16 participants from the campus (aged from xx to xx, M = xx.xx, SD = xx.xx, xx females). All the participants were right handed and native Chinese speaker. They have used software keyboards on smartphones for not less than xx years. xx of the 16 participants were familiar with software keyboards on tablets.

3.2 Design and Procedure

Figure xx illustrate the experimental setting. There was a Morph Sensel [xx] force-sensitive touchpad on the desk. We placed a windows surface tablet on the touchpad, covering less than half of the pad. We drew a QWERTY layout on the touchpad using highlighter pens. This combination of touchpad and table took place of the force-sensitive touchscreen, which is expected to be commercialized in the future. Participants filled in a Microsoft Word document to complete the experimental tasks. They touched on the tablet for pointing, and typed on the touchpad to "enter words". The system recorded every touches and the screencast during the experiment.

The experiment included four sessions of text entry tasks: (1) filling in personal information, (2) describing personal tastes, (3) open-book examination, and (4) picture writing. We counterbalanced the order of the four tasks using a balanced latin square. We did not include transcription as a task as many other studies of text entry do [xx,xx,xx], because our pilot study showed that participants seldom rested their fingers on the touchpad in a transcription task, resulting in low efficiency of obtaining unintentional touches. The details of the four tasks are as follows:

(1) **Filling in personal information:** In the task document, there was a table of xx question about personal information, such as name, gender, and so on. Participant direct touched on the tablet to select what to fill, and typed on the touchpad the "enter words". To protect users' privacy, users were allowed to fill in fake information. This assignment represented those text entry tasks that require frequently switching between pointing and typing.
(2) **Describing personal tastes:** There was another table of xx question about personal tastes, such as "the favorite city", "the favorite fruit", and so on. Participant touched on the tablet for pointing and pressed on the touchpad for typing, as if they were doing office work. Participant were allowed to fill in fake information.
(3) **Open-book examination:** The "exam" consisted of five hard questions such as "What is the 50th element of the periodic table?". Participants could hardly know the answer, so that they needed to use the search engine. The system recorded both the behaviors of answering questions and searching in the Internet. Because the participants could not enter words in the search engine, they needed to speak out when typing, so that the experimenter could replaced them to enter the words. This assignment represented those text entry tasks that require using the search engine.
(4) **Picture writing:** There was a picture (as figure xx shown) in the task document. Participants needed to describe the picture in five sentences and wrote down the story in the document. We asked participants to speak out while typing, so that the experimenter could replaced the participants to enter the words. This assignment represent the text entry tasks that the users need to think while writing.

Before the experiment started, the participant had five minutes to familiarize himself with the tasks and the requirements of the experiment. During the warm-up phase, the participant "typed" on the touchpad freely, while the experimenter reminded the user of two points. First, the keyboard did not provide any feedback. Participants could not enter words, but imaged that they entered words. Because the users' first language are Chinese, which involved word selection in the text entry method, users assumed that the desired word is always the first one of the candidate words. Second, users needed to imagine that the keyboard can perfectly prevent unintentional touches, and adjust their behavior according to this assumption. For example, they could rest their fingers on the keyboard while thinking. This is not mandatory. Participants could make choices as they wished.

After finishing each session of task, the participant labeled the data through an interactive program. The program showed the capacitive images of touchpad and the screencast of tablet at the same time. As figure xx shows, there were some red points on the capacitive images that showed the touchpoints reported by the touchpad. Participants labeled the intended touches as green points. Because participants got context information from the screencast, they were able to identify most intentional touches. If participants were not sure, they could label the touchpoint as a blue point to remove the data. In average, participants spent 15 minutes to finish the text entry tasks and spent 45 minutes to label the data. Participants rested for five minutes between two sessions to avoid fatigue. The study was generally completed within 80 minutes.

3.3 Apparatus

As figure xx shows, we placed a Windows surface tablet and a Morph Sensel force-sensitive touchpad [xx] together to simulate a tablet computer that contains force-sensitive touchscreen. The Sensel Morph is a multi-touch and force-sensitive touchpad, which sense the position and the pressure level of touches. The Sensel Morph contains 185 x 105 sensor elements ("sensels") at a 1.25mm pitch. Each contact can sense approximately 30000 levels, ranged from 5g to 5kg. The upper limit of frequency is 125Hz (8ms latency), while we slowed it down to 50Hz to fetch stable data. The Sensel Morph provides capacitive images and touchpoint information including position, timestamp, touch area, pressure level and shape. The recognition of touchpoint is sensitive that almost every contacts are reported as touchpoints, so in this paper, we identified unintentional touches among reported touchpoints, while did not consider missing touches by the Morph Sensel.

The sensing area of the Morph Sensel is 240mm x 138mm. We used highlighter pen to draw a Qwerty layout on the touchpad as shown in figure xx. The width of the sensing area (240mm) is a little shorter than the Qwerty layout on a 15 inches MacBook (270mm).

传感区域的大小是240mm*138mm，我们在传感区域中央用记号笔画上了Qwerty布局，用来提示用户每个按键所在的位置。Morph Sensel的传感区域比15寸MacBook上的键盘（xx-mm*xx-mm）小一些，为了能在触摸板上画上Qwerty布局，而不对用户的打字行为造成太大的影响，如图xx所示，我们维持了MacBook键盘上每个按键的大小，并去除了本次实验中不会用到的符号键（如中括号和分号）。由于软键盘的布局可以多种多样，而我们希望做一个应用范围更广的防误触算法，因此我们不会将键盘布局作为先验知识应用在防误触算法当中。

我们使用的平板电脑是Windows surface xx，xx核，cpu，程序运行的帧率稳定在50FPS，我们采集到的报点数据也是50FPS。

3.4 Result

原始数据共包含xx个数据点，其中用户无法区分的点的占比为xx.x%，刨去这些数据点之后正例（打字事件）的比例是xx.x%，负例（误触点）的比例是xx.x%。我们先使用简单的机器学习方法对数据进行验证，当机器学习结果和用户标注结果有大量不同时，我们会将用户召回，让她重新标注这些分类错误的数据，其中，有xx.x%的数据的确是用户标错了（有的用户对误触的理解出现了偏差），修正了用户标错的数据之后，实验共采集了xx个数据点，其中xx%是误差。

在该数据集上，简单的机器学习方法的错误率为xx.x%，其中xx.x%的误触和xx.x%的漏报。我们对这些简单的机器学习不能正确分类的数据点进行了人工观察，总结出来这些无触点的分类如下表所示。

【表格：简单机器学习容易分类出错的误触点类型，简单的机器学习方法是采取报点后3 5帧的数据作为数据集，采用压力、面积和压强的时序特征和SVM训练的模型】

从表格xx能够看出来，大多数误触点都是有明显的特点的，我们针对几乎每种错判，都设计了专门的特征向量，用于机器学习。只有表中加粗的两类错判很难找到特征，这一类数据的占比为xx.x%，用户仅能通过任务的上下文来标注，而系统不可能准确知道用户想要输入的文字，因此我们认为这一部分数据的分类问题是不可解的。因此在该数据集上，人工判断的准确率不超过xx.x%，这也是我们给自己算法定的目标。

3.5 Recognition

训练集是报点后4到6帧的数据，测试集是报点后第5帧的数据，如果报点提前结束，则选取结束那一帧的数据。使用第5帧作为测试集的原因是，模拟发现，5帧的延迟能较好地平衡识别准确率和识别延迟，我们会在稍后讨论识别延迟和准确率之间的tradeoff。使用一个时间窗口4 6帧的数据作为训练集的原因是，将数据对齐的问题交给机器学习来解决，通俗地讲，某次点击中第五帧的数据，可能和其它同类点击中第4、第5或第6帧的数据比较相似。

我们采用支持向量机来实现二分类模型，特征向量如表格xx中间的一列所示，每一段特征向量独立的预测准确率也公布在表格中了。将所有特征结合起来的准确率为xx.x%。

最终，Leave-one-out准确率达到了xx.x%，相比之下，baseline[xx]的准确率仅为xx.x%。这一结果说明两点：首先，用户在想象中能防误触的键盘上打字时，会引发很多难以仅通过阈值方法区分有意与否的触摸点，对防误触算法带来挑战；第二，结合压力触摸屏上众多的传感信息、结合时间、空间信息，能够大幅度提高触屏上误触点的识别。

3.6 Discussion

研究者将初版TypeBoard算法实现成一个可以真正打字的Demo，然后亲身体验。研究者发现，这一版的TypeBoard已经能防止大部分的误触，但是有的行为会导致可复现的误触，比如左手休息、右手打字的情况。其中可能有两方面原因，一是实验一得到的数据量样本数不够多，没有全面覆盖各种各样的误触行为；第二个原因是，用户在真正可以防误触的键盘上打字时，其行为可能和实验一所采集到的行为有差异。因此，我们有必要设计实验二，来调查用户在（有反馈的）防误触触屏键盘上的打字规律。

在用户行为的分析方面，有许多值得讨论的地方，比如误触点的构成（休息、手掌误触、输入间误触）、不同实验任务对用户行为的影响、不同识别延迟对识别准确率的影响等等。但由于实验一采集到的数据仅代表“用户在想象中防误触键盘上的行为”，不如实验二中“用户在防误触键盘上的行为”有价值，因此我们只是通过表格的形式展示了实验一中这些问题的结果，而更多的讨论会在实验二中展开。

相关讨论的结果，误触点的构成，不同任务对用户行为的影响，不同识别延迟对识别准确率的影响。

4 STUDY 2: USER BEHAVIOR ON TYPEBOARD

实验二的目的是研究用户在防误触键盘上的打字行为，并根据分析此行为优化触屏键盘上的防误触算法。在实验一之后，我们已经得到了初版的TypeBoard算法，在本实验中，我们采集了用户在TypeBoard防误触触屏上的打字数据。实验共采集了5组*4人=20人的数据，在采集完每4名被试之后，我们都会利用新的数据去优化TypeBoard，并在之后的实验中使用优化后的TypeBoard，这样做的目的是迭代地逼近鲁棒的防误触算法及其上的用户行为模型。

4.1 Participants

我们从本地的校园中邀请了20名用户参与实验，他们的年龄从xx岁到xx岁不等，平均数是xx，标准差是xx，其中有xx名女性。所有的用户都是右撇子，所有用户有超过xx年的手机文本输入经历，xx名用户常用平板电脑进行文本输入。这些用户没有参与过实验一。如上面所说，我们将用户分成了五组，为了

保证五组用户的平衡性，我们通过程序随机模拟了10000次分组，最终确定了平均年龄最相近的分组方式，这五组的年龄平均数分别为xx,xx,xx,xx,xx，每组分别有xx,xx,xx,xx,xx名女性。

4.2 Design and Procedure

在实验之前有一个训练阶段，用户有五分钟时间通过誊写例句熟悉该键盘，例句随机选自xx例句库[xx]。由于所有用户都没有过使用防误触触屏键盘的体验，在用户的试用过程中，实验者提醒用户该键盘有防误触的功能，平时可以把手休息在触屏上。这不是强制的要求，具体行为由用户根据具体的任务和自己的喜好决定。

正式实验分为四个session，分别收集了用户在填写个人信息、描述个人爱好、模拟开卷考试和看图写话这四个任务下的打字数据，我们通过拉丁方来平衡每个用户做这四个任务的顺序。如图xx所示展示了实验二的实验设置，桌面上的实验设备包含morph-sensel压力触摸板、鼠标、显示器和耳机。在本实验中，用户同样通过填写word文档的方法来完成四个文本输入任务（图xx）。和实验一不一样的是，用户在本实验中真的能够通过TypeBoard来键入字母，且打字的过程中能够听到"啪啪"的声音反馈。在极少数情况下，用户会发现一类机器学习总是错判的情况，字母不能正常上屏。举例来说，第x名用户经常在单手五指都休息的情况下，另一只手打字，系统总是漏报。在这种情况下我们鼓励用户维持这种系统会误判的用户行为，并像实验一中一样想象字母已经正确上屏，然后在标注的时候标注出这些错误。

在每个session结束后，用户通过交互式程序标注刚刚完成的session的报点。和实验一的标注过程相同，所有的报点一开始都被标为红色（误触），用户需要将每一个有意的点击标注成绿色（正例）。实验者在一旁观察标注的过程，并通过另一台电脑事先得知机器学习的结果，当发现机器学习结果和用户人工标注不符的情况时，实验者会人工分析其中的原因，如果实验者不能通过独立思考弄明白其中的原因，或者他认为被试标注出错时，他会立即和被试进行讨论。

在本实验中，用户完成输入任务的总时间大约为30分钟，比实验一稍长，这是因为实验二的输入过程涉及键入字母和删改操作；用户完成标注的时间大约为45分钟；每两个session之间用户休息5分钟时间以避免疲劳，实验总时长为90分钟。

由于用户的打字行为和触屏防误触能力会互相影响，我们将20名被试平均分为五组，在每组被试完成实验后，我们都会更新数据集和添加必要的特征向量，以训练新的防误触算法，迭代地增强触屏键盘的防误触能力。这就是说，我们期望每组用户所用到的TypeBoard键盘的防误触能力都会更强，也期望每组用户的打字行为更接近防误触键盘上的自然输入行为。

4.3 Apparatus

实验二的设备与实验一相比，只多出了一个普通的有线耳机，用于提供打字时的声音反馈。实验二所运行的程序包含两个功能，一是区分打字事件和误触，二是将打字事件触点位置上的字母上屏，其运行速度为50FPS。

4.4 Result

在实验的过程中，每五个用户的实验结束之后，我们都会采用新的数据来优化防误触算法，如图xx所示是防误触算法随着完成实验人数的增加的变化，其中每个测试点的数据集包含实验二过程中已经收集到的数据加上实验一的所有数据，评测方法是leave-one-out检测。结果表明，随着实验人数的增加，迭代更新TypeBoard、初版TypeBoard和baseline[xx]的差距在显著拉大，这说明TypeBoard的防误触能力在增强，我们采集到的数据也越来越接近用户在一个完美防误触键盘上的打字行为。

【图：baseline、初版TypeBoard、迭代更新TypeBoard随着实验人数增加，准确率的变化】

在20名用户的实验都完成了之后，抛去用户无法区分的xx个数据点，我们共收集了xx个有效的数据点。在有效的数据中，正例（打字）的比例是xx.x%，负例（误触）的比例是xx.x%。我们总结了一些实验一的机器学习算法不能很好处理的fail-cases（如表xx左侧所示），我们针对这些情况专门设计了

相应的特征组（如表xx右侧所示）。最终，我们的算法在实验二的数据集上达到了xx.x%的准确率，而baseline的准确率仅为xx.x%，TypeBoard和baseline的差距与实验一相比显著增大。

【表：实验二新总结出来的fail-cases和应对方案】

这一结果说明：第一，用户在防误触键盘上打字时，存在更多、更难分辨的误触点，对防误触算法带来更多的挑战；第二，在该用户行为下，结合压力信息的机器学习方法能够鲁棒地防误触，而baseline-基于电容屏信息和阈值地方法变得不可用。

用户的主观评分和采访。评分包括主观的好评率over 任务，主观的休息手指频率over 任务。

4.5 Discussion

误触点的构成：休息、手掌误触、输入时误触的比例。

实验任务显著影响了用户行为。这说明了考虑不同实验任务的必要性。

TypeBoard算法在不同延迟下的识别准确率问题。

TypeBoard算法在不同数据组合下的准确率问题。motivation，比如有的设备没有压力信息，能否正确区分误触与否?

技术实现和用户行为之间的互相影响，及应对方案。可以列举很多相关工作进行讨论，比如其它防误触算法、文本输入相关算法。

5 STUDY 3: EVALUATION ON TYPEBOARD

实验三的目的有两点：第一，评测TypeBoard的性能，包括在不同的文本输入任务下的输入效率和主观用户体验，baseline是没有防误触算法的触屏键盘。第二，相关工作中我们提到，在防误触触屏键盘上，有望通过加上纹理反馈帮助用户盲打，提高输入效率，本实验探索了这一假设是否成立。

5.1 Participants

我们从本地的校园中邀请了15名用户参与实验，他们的年龄从xx岁到xx岁不等，平均数是xx，标准差是xx，其中有xx名女性。所有的用户都是右撇子，所有用户有超过xx年的手机文本输入经历，xx名用户常用平板电脑进行文本输入。这些用户没有参与过前两个实验。

5.2 Design and Procedure

我们采取了within-subject的实验方法，对比了三种不同的实验设置（如图xx所示）：

(1) **Config. 1):** *Regular Soft Keyboard.* 没有防误触功能的软键盘。
(2) **Config. 2):** *TypeBoard.* TypeBoard，有防误触功能的软键盘，这也是实验二的设置。
(3) **Config. 3):** *TypeBoard + Tactile Landmarks.* 在TypeBoard的基础上，在每个按键的位置上贴上了0.05毫米厚的贴纸，其中F键和J键上各再加上了一条0.05毫秒厚的横杠，用于模拟物理键盘上的触觉Lankmakrs。0.05毫米的厚度可以让用户明显地摸到每个键的边缘，同时符合可变形触摸屏的工艺要求[xx]。

每名用户使用这三种实验设置来完成五个文本输入任务，除了前两个实验中所述的四个任务（填写个人信息、描述个人爱好、模拟开卷考试、看图写话）以外，实验三中新增了文本誊写任务，具体是誊写一个中文句子5次（某中文打字测速网站的第一句话）。文本誊写任务在文本输入相关工作中很常见，一般用于测试文本输入法的输入效率上限[xx,xx,xx,xx]。为了平衡学习效应，我们采用拉丁方来规定用户使用这三种实验设置的顺序，和完成五个输入任务的顺序。

【图：实验三要对比的三种实验设置】

用户使用每一种实验设置之前，都有5分钟的热身时间，通过输入例句来热身。用户在切换实验设置的时候休息5分钟时间，以避免疲劳。用户在每个实验设置下需要完成与实验二相同的五个文本输入任务，每种实验设置下的实验时长大约为xx分钟。实验的总耗时为xx分钟。

5.3 Appartus

除了触摸屏存在三种不同的设置以外，其余设备与实验二完全一样。桌面上只有触摸板，鼠标，显示器和耳机这些设备，用户通过耳机来获取打字瞬间的声音反馈。

5.4 Result: TypeBoard vs. Regular Board

我们首先讨论有/无触觉反馈对用户打字可用性、效率的影响，纹理landmark的问题会在稍后讨论。

我们采用了什么样的统计分析方法，对于xx等违反正态分布的量，我们通过xx方法来进行校正。如果一个独立变量对结果有显著性影响，我们采用xx方法来检验变量两两之间的显著性。

对用户行为、用户体验、输入效率的影响。

要注意的点：1. 有无landmarks两种情况下，用户盲打的时间对比；每个按键点云标准差的对比；误触数量的对比；错误率。

5.4.1 *Completion Time.* English.

在誊写任务下可以对比打字速度。

5.4.2 *Error Rate (Text Entry).* CER

UER

5.4.3 *Detected Unintentional Touches Percentage.*

5.4.4 *Time Components.*

5.4.5 *Subjective Rating and Feedback.* English.

物理负担（疲劳程度），心理负担，主观输入速度，主观输入准确率（误触准确率，而非选词准确率）。

将手指休息在键盘上的频次over 任务，技术。

5.4.6 *Percentage of EyeFree time.*

5.5 Result: TypeBoard with/without tactile feedback

在上面的分析中我们已经看到，自然的防误触算法对用户打字的可用性、效率和用户体验都有好处。接下来，我们来讨论在防误触的情况下，有/无触觉landmarks对文本输入可用性的影响。

对用户行为、用户体验、输入效率的影响。

5.6 Discussion

6 DISCUSSION

此外，这份工作还有两点关于人机交互工作的思考。第一，在用户意图推理的相关工作中，存在的一个普遍的问题，即用户行为和技术实现之间存在相互影响的作用，而大部分的先前工作都忽略了这一效应的存在[xx]。第二，在文本输入研究相关领域，誊写任务是默认的评测标准，然而，誊写任务与常见的文本输入相比缺失了手指休息和思考这两个成分，许多工作在这两个未被评测的步骤上存在明显的局限性[xx]，在这些工作中仅评测誊写任务是有失偏颇的。我们认为以上两点值得学术界的更多关注。

不同语言的问题。分析一下不同按键对防误触算法带来的挑战，并表面英文中只涉及到字母、空格、逗号句号的输入需求，不会受到很大的影响。

7 LIMITATION AND FUTURE WORK

7.1 Limitation

实验二中，每个用户使用TypeBoard的时长只有20分钟，也就是说，我们采集了用户在初次使用防误触键盘时的用户行为，但没有采集到长期影响下的用户行为。在未来，一个长期的实验是有必要和有价值的。

7.2 Future Work

有可能的follow up works:

键盘、触摸板一体for个人笔记本电脑?

触屏+纹理=触屏盲打?

利用手掌位置来处理触屏输入法中的漂移问题。

8 CONCLUSION

一些结论。

REFERENCES

[1] Michel Amberg, Frédéric Giraud, Betty Semail, Paolo Olivo, Géry Casiez, and Nicolas Roussel. 2011. STIMTAC: a tactile input device with programmable friction. In *Proceedings of the 24th annual ACM symposium adjunct on User interface software and technology*. 7–8.

[2] Michelle Annett, Fraser Anderson, Walter F Bischof, and Anoop Gupta. 2014. The pen is mightier: understanding stylus behaviour while inking on tablets. In *Proceedings of Graphics Interface 2014*. 193–200.

[3] Michelle Annett, Anoop Gupta, and Walter F Bischof. 2014. Exploring and understanding unintended touch during direct pen interaction. *ACM Transactions on Computer-Human Interaction (TOCHI)* 21, 5 (2014), 1–39.

[4] Olivier Bau, Ivan Poupyrev, Ali Israr, and Chris Harrison. 2010. TeslaTouch: electrovibration for touch surfaces. In *Proceedings of the 23nd annual ACM symposium on User interface software and technology*. 283–292.

[5] Jörg Brakensiek and Raja Bose. 2013. Method and apparatus for precluding operations associated with accidental touch inputs. US Patent App. 13/221,344.

[6] Olumuyiwa M Durojaiye and David Abzarian. 2016. Classification of touch input as being unintended or intended. US Patent 9,430,085.

[7] Wolfgang Fallot-Burghardt, Morten Fjeld, C Speirs, S Ziegenspeck, Helmut Krueger, and Thomas Läubli. 2006. Touch&Type: a novel pointing device for notebook computers. In *Proceedings of the 4th Nordic conference on Human-computer interaction: changing roles*. 465–468.

[8] Gregg S Goyins and Mark F Resman. 2001. Palm pressure rejection method and apparatus for touchscreens. US Patent 6,246,395.

[9] Jason Tyler Griffin. 2013. Touch screen palm input rejection. US Patent App. 13/469,354.

[10] Terho Kaikuranta. 2006. Method for preventing unintended touch pad input due to accidental touching. US Patent 6,985,137.

[11] Sunjun Kim and Geehyuk Lee. 2016. Tapboard 2: Simple and effective touchpad-like interaction on a multi-touch surface keyboard. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. 5163–5168.

[12] Sunjun Kim, Jeongmin Son, Geehyuk Lee, Hwan Kim, and Woohun Lee. 2013. TapBoard: making a touch screen keyboard more touchable. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 553–562.

[13] Huy Viet Le, Thomas Kosch, Patrick Bader, Sven Mayer, and Niels Henze. 2018. PalmTouch: Using the Palm as an Additional Input Modality on Commodity Smartphones. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. 1–13.

[14] Huy Viet Le, Sven Mayer, Benedict Steuerlein, and Niels Henze. 2019. Investigating Unintended Inputs for One-Handed Touch Interaction Beyond the Touchscreen. In *Proceedings of the 21st International Conference on Human-Computer Interaction with Mobile Devices and Services*. 1–14.

[15] Vincent Levesque, Louise Oram, Karon MacLean, Andy Cockburn, Nicholas D Marchuk, Dan Johnson, J Edward Colgate, and Michael A Peshkin. 2011. Enhancing physicality in touch interaction with programmable friction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 2481–2490.

[16] Hao Lu and Yang Li. 2015. Gesture on: Enabling always-on touch gestures for fast mobile access from the device standby mode. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. 3355–3364.

[17] Juha Matero and Ashley Colley. 2012. Identifying unintentional touches on handheld touch screen devices. In *Proceedings of the Designing Interactive Systems Conference*. 506–509.

[18] Fabrice Matulic, Daniel Vogel, and Raimund Dachselt. 2017. Hand contact shape recognition for posture-based tabletop widgets and interaction. In *Proceedings of the 2017 ACM International Conference on Interactive Surfaces and Spaces*. 3–11.

[19] Valentin Popescu. 2015. Touch Screen with Unintended Input Prevention. US Patent App. 14/764,742.
[20] Luca Rigazio, David Kryze, Philippe Morin, and YUN Tiffany. 2013. System and method for differentiating between intended and unintended user input on a touchpad. US Patent 8,502,787.
[21] Julia Schwarz, Robert Xiao, Jennifer Mankoff, Scott E Hudson, and Chris Harrison. 2014. Probabilistic palm rejection using spatiotemporal touch features and iterative classification. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 2009–2012.
[22] TouchFire. 2015. The screen-top keyboard for ipad. Website. https://www.kickstarter.com/projects/touchfire/touchfire-the-screen-top-keyboard-for-ipad.
[23] Kagan Tumer and Joydeep Ghosh. 1996. Estimating the Bayes error rate through classifier combining. In *Proceedings of 13th International Conference on Pattern Recognition*, Vol. 2. IEEE, 695–699.
[24] Malte Weiss, Roger Jennings, Julie Wagner, Ramsin Khoshabeh, James D Hollan, and Jan Borchers. 2008. Slap: Silicone illuminated active peripherals. *Ext. Abstracts of Tabletop* 8 (2008).
[25] Xuhai Xu, Chun Yu, Yuntao Wang, and Yuanchun Shi. 2020. Recognizing Unintentional Touch on Interactive Tabletop. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 4, 1 (2020), 1–24.
[26] Yang Zhang, Michel Pahud, Christian Holz, Haijun Xia, Gierad Laput, Michael McGuffin, Xiao Tu, Andrew Mittereder, Fei Su, William Buxton, et al. 2019. Sensing posture-aware pen+ touch interaction on tablets. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. 1–14.