

# 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，一款带压力信息的触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), 3 pages. <https://doi.org/10.1145/1122445.1122456>

## 1 INTRODUCTION

物理键盘上的按键有三种状态，分别是released, touched和pressed，而触屏键盘上的按键只有released和pressed两个状态。Touched状态的缺失是触屏输入法不如物理键盘的重要原因，原因有三点：首先，用户可以通过触摸物理按键上的纹理来align他们的手指；第二，用户可以将手指休息在物理键盘上。为了弥补触屏键盘上touched状态的缺失问题，我们提出了TypeBoard，一个在压敏触屏键盘上的防误触算法，能够区分type-touches和unintentional-touches，判断用户的触摸意图。TypeBoard使得用户可以在键盘上休息手指，也为相关工作中在触屏上加上纹理[xx]和触觉反馈[xx]的构想提供了可行性。

在触碰键盘上区分Type-Touches和Unintentional-Touches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，而将其它较轻的触摸事件视为误触，算法上使用了时间和位移的阈值在区分打字和误触。TapBoard先设计了基于阈值的防误触算法，再证明用户可以适应这个算法。这个研究方法有三个缺陷：一是仅通过阈值方法来防误触的准确率不高；二是正常键盘会在pressed以后即作出判断，而TapBoard只能在released时判断，这会影响用户体验和输入效率[xx]。第三，也是最重要的是，键盘的防误触能力会影响用户的心理模型，比如，用户在防误触能力很强的触屏键盘上，可能更倾向于把手指休息在键盘上。TapBoard无法理解用户在防误触能力很强的触屏上的输入行为。另

---

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

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. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](mailto:permissions@acm.org).

© 2018 Association for Computing Machinery.

2476-1249/2018/8-ART111 \$15.00

<https://doi.org/10.1145/1122445.1122456>

外, TapBoard在实验设计上有一个错误, 他们通过一个基于誊写任务的对比实验, 证明了TapBoard的输入效率和传统触屏键盘无异, 这是以偏概全的。我们通过Pilot-study发现, 在誊写等快速打字的任务中, 用户很少将手指休息在触屏上, 因此TapBoard无法证明用户在写日记等任务中不会因为手指的休息造成误触问题。

我们提出了TypeBoard, 我们的目标是窥探用户在一个“完美的防误触触屏键盘”上打字的行为, 并根据用户的心理模型, 来设计一款能完美防误触的触屏键盘。由于防误触能力会影响用户的心理模型, 我们采用了迭代的方法来逼近“完美的防误触键盘”及其上的用户行为, 实现方法是迭代地采集用户数据和使用机器学习来优化防误触算法。我们设计实验一采集了用户在一块没有任何反馈的普通触屏上“打字”的数据, 我们要求用户在输入过程中想象该触屏能够“完美防误触”, 实验后让用户标注每个触点是否有意, 我们通过机器学习设计了初版的TypeBoard算法。接着, 我们设计实验二采集了用户在初版TypeBoard上的文本输入数据, TypeBoard根据自己的预测给出了视觉和听觉反馈, 我们更新数据集以及改进了机器学习模型来设计了终版TypeBoard算法。实验二采集到的数据是一个严格的数据集(有很多非有意触碰点), TypeBoard的准确率达到了xx.x%, 预测时间是touch之后的xx-ms以内。相比之下, baseline-TapBoard的准确率只有xx.x%, 且只能在release时做出判断。最后, 我们在写日记、填问卷、誊写等多个输入任务下对比了TypeBoard、TapBoard和普通触屏键盘, 我们发现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。

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“近乎完美的防误触触屏键盘”上的输入行为, 也公开了该数据集。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, 提高了用户体验。

(可选) 我们在实验方法的创新上有贡献, 我们提倡采用迭代的方法来处理技术和心理模型会互相影响的情况, 并提倡用在鲁棒技术上采集的实验数据来做机器学习准确率的评测, 因为这种情况下用户的数据更加苛刻。

## 2 RELATED WORK

### 1. 触屏上的防误触算法

防误触的算法可以按两种方法分类, 一是按正例的类型分类, 二是按输入模态分类。

正例可以是广义的触摸输入, 也可以是特定的任务, 如打字、触屏笔写字等等; 输入的模态可以是传统的触摸屏, 也可以在触碰上加装压力、xx等信息。甚至是结合了肢体语言、眼动、头动等触屏之外的信息。

在这份工作中, 我们考虑用带有压力的触摸屏, 来实现打字任务下的防误触问题。一方面, 这一问题是很重要的, 因为大多数触屏输入法还没有防误触算法, 使得用户输入的效率和自然性受到影响; 另一方面, 这一收窄过的问题是理论上可解的, 在我们的试验中, 有xx.x%的数据是可以人工标注的, 只要我们使用机器学习去接近这一准确率, 该技术就能实现。

与先前工作不同的是, 我们这份工作首次考虑到了技术和用户行为之间的影响, 有的工作在采集数据时不给出反馈, 或者是给出了未经防误触算法过滤的反馈, 这都不能正确反映用户在最终技术上的心理模型。在这份工作中, 我们采用迭代式的方法, 重复地采集用户数据和优化机器学习算法, 最终总结出用户在一个“近乎完美防误触键盘”上打字的行为, 并根据此用户行为, 设计了一款“近乎完美防误触的键盘”。

### 2. 触屏键盘上支持防误触的好处

#### (1) 有机会弥补触屏键盘上没有触摸状态的问题

物理键盘上有release/touch/press三态, 而触屏键盘上只有release/press两态。touch状态在物理键盘上承担着重要的作用, 具体有三点: 1、物理键盘上的触觉纹理可以让用户不看键盘的情况下对齐手指, 从而支持盲打; 2、键帽能够给用户提供触觉反馈, 确认一次点击操作, 这不仅增强了用户体验, 也显

著地提升了打字速度；3、在物理键盘上，用户可以将手指休息在键位上，降低了疲劳程度。触屏键盘上touch状态的缺少，使得触屏上的打字效率显著低于物理键盘打字的效率。

有不少工作尝试弥补物理键盘和触屏键盘之间的gap。xx通过加装xx硬件，提供了纹理信息，达到了xx效果。xx通过加装xx硬件，提供了用户在打字时的触觉反馈，达到了xx的效果。xx采用简单的时间和距离阈值方法来区分打字和误触，从而允许用户将手指休息在触屏键盘上。然而，以上对触屏键盘的体验提升，都是以能够正确区分打字和误触为前提的。而还没有哪份工作做到了很高的识别准确率，这也是我们这份工作的重要动机。

(2) 在touchscreen上无缝区分keyboard和其它有意输入

如果能够在touchscreen上identify-typing-action，就可以仅使用触摸屏的同一块区域，同时支持文本输入和其它交互，如触摸屏控制鼠标pointing，又如其它的一些手势命令等等。TapBoard2能够有效地将打字的触摸事件和pointing的触摸事件正确区分，使得同一块触摸屏及支持了文本输入，又支持了触摸板控制鼠标。xx将手掌的误触作为输入通道。

还可以区分typing和stylus输入

[2014-PenUnint]

[2014-PenMightier]

[2014-PalmRejection]

[Exploring and Understanding Unintended Touch during Direct Pen Interaction]

(3) 将原本应该是误触的触碰利用为输入信息

Matulic et al. [2017-HandContact] extended hand interactions from fingertips to the whole hand in hand-shape based interaction. Tabletop interaction can be enriched by considering whole hands as input instead of only fingertips. We describe a generalised, reproducible computer vision algorithm to recognise hand contact shapes, with support for arm rejection, as well as dynamic properties like finger movement and hover. A controlled experiment shows the algorithm can detect seven different contact shapes (such as fist, flat palm and spread hand) with roughly 91% average accuracy如果能将type点击事件和手掌的触摸事件区分开来，就可以支持hand contact shape所支持的交互方式。

Zhang et al. [77] proposed to leverage various hand postures such as using the palm to augment pen and touch interactions.

[2018-PalmTouch]

将大鱼际的误触作为模式切换。

We present PalmTouch, an additional input modality that differentiates between touches of fingers and the palm. We present different use cases for PalmTouch, including the use as a shortcut and for improving reachability. We have developed a model that differentiates between finger and palm touch with an accuracy of 99.53% in realistic scenarios.

[??] 比如处理漂移问题。