

基于字符级纠错的盲人智能键盘

石伟男 喻 纯
清华大学

关键词：盲人智能键盘 字符级纠错

介绍

在当今的移动计算时代，智能手机对于盲人或视障用户而言，和对于明眼人用户一样，是不可或缺的存在。然而，对于视障用户来说，许多交互上的障碍影响了他们在移动计算方面的体验，其中一项便是文本输入。在智能手机上进行文本输入是一项有挑战的任务，甚至对明眼人用户来说也是如此，原因在于人们使用手指去精确地选择一个比较小的目标键是非常困难的。幸运的是，对于明眼人用户来说，几乎所有的现代触屏键盘都是所谓的“智能”键盘，因为它们有词级别的自动纠错能力。当用户输入一个词语分隔符（例如空格）之后，键盘会根据语言的上下文和空间的点击位置将用户的输入纠正为词典中的一个词。

但不幸的是，视障用户却无法享受这些便利，他们进行文本输入是一件十分困难的事情。其主要原因在于，现有的词级别的自动纠错键盘并不符合视障用户的输入行为。明眼人用户可以忽略输入过程中的错误字符，等待自动纠错算法来纠正这些错误。而视障用户无法这样做，他们使用读屏软件（例如安卓系统上的TalkBack和iOS系统上的VoiceOver）将字符逐个输入，直到当前字符被确认输入了才会继续输入下一个字符。为了输入某个特定的字母，视障用户首先在键盘上使用手指摸索着找到目标键所在的位置，然后抬起手指（或双击屏幕）确认输入。他们需要确保每个字母都被正确输入，因为（1）这样可以避免之后再修改这些错误所

带来的较高成本^[1]；（2）语音反馈中的错误相对于视觉反馈更加明显，视障用户更难以忽略输入过程中的错误。综上，读屏键盘并没有词级别的自动纠错能力，视障用户只能忍受较低的文本输入速度（低于每分钟5个词^[2,3]）。

针对上述问题，我们提出了VIPBoard，一个在不改变用户输入行为的基础上把自动纠错能力带给视障用户的“智能”读屏键盘。VIPBoard的主要功能由两个机制实现：（1）它可以根据语言模型和手指的位置预测用户最可能输入的字符，并自动调整键盘布局，以使该字符所对应的按键处于手指接触的位置。这样，用户无须移动手指就能纠正输入，节省了时间和精力。（2）键盘布局会进行缩放调整来确保所有的按键都能通过移动手指来访问，以保证在预测和布局调整不正确的情况下仍可以正常使用键盘。这两个机制提供了和传统非智能键盘一致的用户体验，并将学习成本最小化。VIPBoard的优势建立在大多情况下对字符预测正确的基础上。

相关工作

为视障用户设计的输入法

对视障用户而言，传统的使用软键盘的方式是在读屏软件的帮助下完成的，而这种方式是非常低效的（文献[2]中每分钟输入0.66词，文献[27]中每分钟输入1.32词）。视障用户在智能手机上使用软键盘的主要挑战在于精确选择目标的困难性^[2]。

因此, 研究者们提出了很多不同的技术来解决这一问题。

一种直接的解决方案是降低目标数量, 把目标安置在基于物理设备易被找到的位置, 并保持目标静止。桑切兹 (Sánchez) 等人^[25] 在屏幕上放了9个类似于T9键盘的虚拟按键, 这种方式使得用户每输入一个字符需要进行多次点击。奥利维拉 (Oliveira) 等人^[23] 提出的 BrailleType 技术把屏幕分成了6个大的按钮, 并允许用户通过盲文编码输入文本。这些方法能够提供更准确的结果, 然而由于目标数量较少, 输入效率会比较低。

另一种可行的解决方案是基于手势的方法, 因为它们并不需要用户点击任何目标。这些方法通常允许用户通过不同的键盘布局在所有的字符中导航以选择目标。例如, 这种布局可以是三个不同的转盘菜单 (pie menu)^[29], 通过元音分隔的字母表^[15] 或者环形布局的8组按键^[2] 选择目标。盲文编码的文本也可以通过简单的手势来进行输入^[7,10,12,20,21]。这些方法的主要成本是学习手势的成本以及在目标中导航需要较长的时间成本。

还有一些基于硬件的方法可以辅助视障用户选择目标, 例如, 使用外部设备^[11,18] 或者触觉反馈^[8]。由于对特殊设备的要求, 这些方法既不具有普适性, 在日常使用中也不够方便。

这些方法的目标都在于通过新的、不同的界面来增强用户获取每一个字符的绝对能力。然而, 这些工作并非基于传统的 QWERTY 键盘布局所设计, 因此既无法融入到现有的商业键盘中为盲人用户所使用, 又增加了盲人使用新键盘的学习成本。同时, 也没有工作利用现在比较成功的智能输入法中的自动纠错算法, 将其强大的纠错能力融入到盲人输入法中。VIPBoard 通过使用字符级的自动纠错方法来增强读屏键盘, 以填补研究中的空白, 尽可能减少视障用户的输入噪声所带来的影响。

现代智能纠错输入法

自动纠错方法被广泛应用于词级别^[27, 31] (或句子级别^[26]) 的智能输入法中。这些方法通常在用户

完成所有输入, 并输入一个分隔符 (例如空格) 后对输入进行解码, 根据用户的输入提供一个概率较高的候选词列表。古德曼 (Goodman) 等人^[16] 首次提出了将触摸模型和语言模型结合到一起以减少用户输入噪声的方法。这种方法已经被拓展到各种不同的上下文场景中, 例如不同的握姿^[14]、运动能力^[13] 和屏幕大小^[31]。然而这些工作并不适合视障用户使用, 原因是视障用户会由于纠正错误的高昂代价和明显的语音反馈, 在输入过程中需要确认每一个字母都被正确输入。这启发了我们将字符级的自动纠错方法应用到视障用户的软键盘上。

字符级别的智能输入法在用户每输入一个字符时进行预测。这些技术通常根据预测所得的概率值, 在每次触摸输入时改变键盘的布局 (可见的或者不可见的): 波动最优字符布局 (Fluctuating Optimal Character Layout)^[6] 重新排布键盘以使得最有可能的下一个字符距离光标更近; 鱼眼 (Fisheye) 键盘^[24] 和 BigKey^[5] 拓展即将触摸的按键的面积, 从而为用户提供更强的目标选取能力; 键-目标重新排布算法 (key-target resizing)^[4,17] 根据计算出的概率值改变每个键下面的目标区域。

显式改变键盘可能会影响明眼人用户: 他们熟悉一个新的布局需要学习成本^[6], 或者需要根据布局改变自己的行为^[5,24]。然而, 这一问题对于视障用户并不存在, 因为他们并不能直接“看”到布局的变化。因此, 在软键盘上进行键盘布局的自适应对于视障用户是可行的。除此之外, 和以往的在每次输入结束后进行预测的算法不同, 在输入过程中纠正错误的字符级的自动纠错方法也是值得探索的。本文将字符级的自动纠错方法应用到盲人输入法中以增强传统的读屏键盘, 提高输入效率。

VIPBoard 键盘设计

在本部分, 我们首先介绍视障用户如何使用 VIPBoard, 之后介绍 VIPBoard 的两个主要部分: 用于预测用户意图的贝叶斯算法和布局调整策略。预测算法基于用户的输入历史和一个预先定义好的

词库来预测用户的输入意图，布局调整策略会根据预测出的结果调整键盘布局，为视障用户提供一致的交互体验。

交互设计

VIPBoard 的交互基于传统的读屏键盘。这一做法的目的是将用户的学习成本最小化。传统的读屏键盘通过读出用户触摸的界面元素的名字为视障用户提供语言反馈。用户通过抬起手指¹来确认选择。

要想在一个传统读屏键盘上输入一个字符的过程和使用读屏软件保持一致，需要经过三个阶段：

尝试阶段（touch down 事件）。用户将手指放到键盘上，读屏软件读出手指所在键的名字（图 1(a)）。

校准阶段（touch move 事件，可省略）。如果读屏软件读出的第一个键就是用户的目标键，则跳过这一步。否则，用户需要根据键盘的布局和语音反馈来移动手指以找到目标键（图 1(b)）。用户的触摸点每次进入一个新键的边界都会收到相应的语音反馈。

确认阶段（touch up 事件）。用户从屏幕上抬起手指以输入当前字符。

每当屏幕接收到 touch down 事件后，VIPBoard 会估计用户想输入每个字符的概率。之后系统会读出预测的概率最高的字母。在这种情况下，用户无须进行下一步的校准操作，因此打字速度得到了提

升（图 1(c)）。为了与传统键盘的交互保持一致，在预测结果出错的情况下，用户仍然能进行校准操作。VIPBoard 根据触摸位置和预测出的结果调整键盘布局以实现这一效果。在这种情况下，键盘的相对布局保持不变（图 1(d)）。VIPBoard 的确认操作和传统的键盘一样。

输入预测算法

我们通过一种类似于文献 [16] 的方法来预测用户最可能输入的键。令 pos 表示用户输入的触摸位置， pre 表示用户已经输入的历史字符。对于任意字符 c ，我们可以通过如下方式计算在已知 pos 和 pre 的情况下用户输入 c 的概率：

$$\begin{aligned} P(c|pos, pre) &= \frac{P(pos, c, pre)}{P(pos, pre)} \\ &= \frac{P(pos|c, pre)P(c, pre)}{P(pos, pre)} \quad (1) \\ &\propto P(pos|c, pre)P(c, pre) \end{aligned}$$

在公式 (1) 中， $P(c, pre)$ 为语言模型，可通过一个提前定义的词库计算得到，计算方法与文献 [32] 类似：

$$P(c, pre) = \frac{\sum_{W \in S(pre) \wedge W_{n+1}=c} P(W)}{\sum_{W \in S(pre)} P(W)} \quad (2)$$

其中 n 是 pre 的长度， $S(pre)$ 表示所有满足 $P(pre|W_1W_2 \cdots W_n) > 0$ 的单词 W 构成的集合。换言之， $S(pre)$ 代表所有以 pre 为前缀的词

的集合。 $P(pos|c, pre)$ 称为触摸模型，反映了用户的输入噪声。同大部分以往工作一样^[16, 22]，我们认为每一次触摸都是独立的，因此它可以被简化成 $P(pos|c)$ ，并通过一个二维高斯分布计算。在计算了所有字符

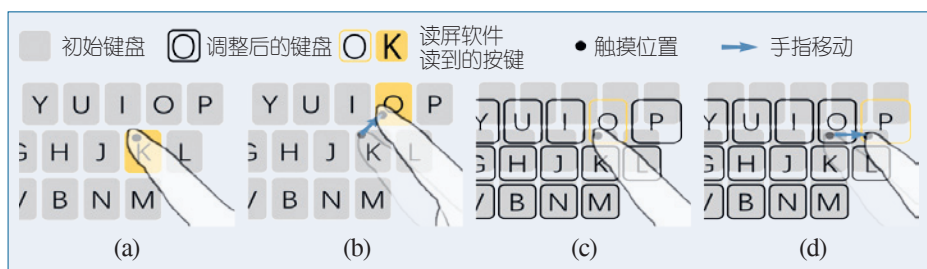


图 1 当“hell”已经被输入后输入“o”的示例。(a~b) 在传统的读屏键盘上，用户通常点击在错误的位置，并通过一次校准操作找到目标键，这一过程通常十分耗时。(c) VIPBoard 预测出了目标字符“o”并自动调整了键盘布局，此时用户无须校准。(d) 即使预测是错误的，用户仍然可以通过在新布局上移动手指来输入其他字符（例如“p”）

¹ 其他可能的方式包括双击确认或者在不抬起屏幕上手指的同时用另一只手指点击屏幕。

c 的概率 $P(c|pos, pre)$ 后, 我们可以找到概率最大的字符作为用户的输入意图。

布局调整策略

布局调整的主要目的是使交互和布局与传统读屏键盘保持一致。这一步骤的主要优点是如果预测出错, 用户仍然可以进行校准操作, 因此用户可以在键盘上输入任意字符。布局调整在每次 touch down 事件被触发并且按前述算法计算出用户最有可能输入的键之后起作用。当字符被确认输入之后 (即触发一次 touch up 事件) 布局会重新回到原始状态。这一设计主要是为了避免过度的调整^[17], 其结果可能导致界面上有很多按键非常小而难以被用户输入。布局调整的主要步骤如下:

步骤 1: 计算出从最可能输入的键在初始布局中的位置到用户触摸位置的位移向量。该位移向量是从键边界到触摸位置的最短路径。

步骤 2: 根据位移向量将最可能输入的键平移到新位置, 平移之后的触摸位置应被新的键边界所包含。

步骤 3: 将其他按键相对于最可能输入的键进行平移和缩放, 并确保所有的键都在键盘边界内。

步骤 4: 如果键盘上存在过小而不易输入的键, 退回到原始的布局并读出触摸位置所对应的键。否则, 采用新的布局并且读出该用户最可能输入的键。

图 2(a) 展示了初始布局及调整后的新布局的一个示例。

性能评估实验

我们进行了一项用户实验来评估 VIPBoard 的文本输入性能。在评估中, 我们首先实现了一个支持英文和中文输入的 VIPBoard 原型系统, 之后进行了一个被试内设计²的用户实验来比较 VIPBoard 和传统键盘的性能。



图 2 调整键盘布局的示例。(a) 预测出的最可能输入的键 (“o”) 的位移以及其他键 (例如 “p”) 的缩放。(b) 未经优化的位移向量 (dx, dy) 及优化后的位移向量 (dx', dy')

Sample English phrase: call for more details
Sample Chinese phrase: 希望 你 可以 做 一个 好梦
Corresponding pinyin string: xiwang ni keyi zuo yige haomeng

图 3 英文、中文测试句子和中文句子对应的拼音串的示例

我们从一所特殊教育学院招募了 14 名视障学生 (7 名男性, 7 名女性), 他们的平均年龄为 23.7 岁 (标准差为 1.1 岁)。8 名用户全盲, 其他用户有非常低的视力。这些用户中没有人参与过实验。他们汇报自己使用触摸键盘的平均时间为 4.9 年 (标准差为 1.6 年)。所有实验用户均熟知 QWERTY 键盘布局, 并且知道每一个键的大概位置。

我们使用了双因素 (变量为输入阶段和使用的技术) 被试内设计来测试 VIPBoard 的性能。我们要求用户通过传统键盘和 VIPBoard 分别用英文和中文进行 4 个阶段的输入, 每个阶段输入 5 个句子。英文输入的测试句子集是文献 [30] 中的 T-40 集合, 中文输入的测试句子集是从文献 [9] 中解析出的长度合适的句子集合。实验中, 一共输入了 1120 个测试句子 (14 名用户 $\times$ 20 句子 / 条件 $\times$ 2 种键盘 $\times$ 2 种语言), 如图 3 所示。

总体输入速度

我们通过词每分钟 (WPM) 来衡量文本输入的

² 被试内设计 (within-subjects design), 亦称 “组内设计” 或 “重复测量设计”, 是指每个或每组被试接受所有自变量水平的实验处理的真实实验设计, 其特点是所有的被试都受到每一个水平自变量的影响。

速度,其计算方法如下^[19]:

$$WPM = \frac{|T| - 1}{S} \times 60 \times \frac{1}{5} \quad (3)$$

其中 $|T|$ 是最后输入的字符串的长度, S 是一次输入中从第一次到最后一次触摸的时间间隔(以秒为单位)。对于中文输入,我们使用拼音串作为最后的输入字符串。文本输入的速度按词计算,并且没有考虑选择候选词的时间。

RM-ANOVA³结果表明,在两种语言下,使用VIPBoard的文本输入速度明显比使用传统触屏键盘的速度高(对于英文输入 $F_{1,13} = 23.67, p < .01$;对于中文输入 $F_{1,13} = 9.51, p < .01$)。如表1所示,输入英文(中文)时,使用VIPBoard的平均文本输入速度比使用传统键盘高12.6%(13.7%)。在VIPBoard上,用户平均1分钟大约能多输入5个字符。

总体输入错误率

我们使用已纠正错误率(CER)和未纠正错误率(UER)来衡量字符级的输入错误率^[28]。用户倾向于修正输入过程中的大部分错误,在最后输入的字符串中只留下少量错误,如表2所示。RM-ANOVA的分析表明两个键盘上的已纠正错误率(对于英文输入 $p = .99$;对于中文输入 $p = .95$)和未纠正错误率(对于英文输入 $p = .11$;对于中文输入 $p = .37$)均没有显著区别。这一结果表明对于视障用户来说,两种键盘上的错误处理没有区别。

点击错误率

如表3所示,VIPBoard相对于传统键盘可以减少英文(中文)输入中63.0%(60.0%)的点击错误。RM-ANOVA分析结果表明,在两种语言下VIPBoard相比于传统键盘在点击正确率方面有显著的提升(对于英文输入 $F_{1,13} = 152.73, p < .01$;对于中文输入 $F_{1,13} = 67.34, p < .01$)。我们同样计算了两种

种技术中每个键的点击错误率,VIPBoard中每个键的点击错误率均比传统键盘上的低。

主观反馈

主观打分:在完成了文本输入部分后,每名用户被要求在李克特7级量表⁴上对两种键盘进行打

表1 两种键盘的平均文本输入速度

	VIPBoard	传统键盘
英文输入	8.14(标准差为1.62)	7.23(标准差为1.67)
中文输入	8.61(标准差为2.04)	7.57(标准差为1.95)

表2 两种键盘上的平均已纠正错误率和未纠正错误率

		VIPBoard	传统键盘
已纠正错误率	英文输入	3.51% (标准差为1.92%)	3.54% (标准差为2.25%)
	中文输入	4.46% (标准差为2.30%)	4.48% (标准差为2.60%)
未纠正错误率	英文输入	1.30% (标准差为2.36%)	1.94% (标准差为2.28%)
	中文输入	1.28% (标准差为1.88%)	1.03% (标准差为1.52%)

表3 两种键盘的平均点击错误率及标准差

	VIPBoard	传统键盘
英文输入	14.2%(8.10%)	32.8%(12.9%)
中文输入	14.8%(6.30%)	34.6%(14.1%)

表4 两种键盘的平均主观评分(标准差)。分数越高,键盘性能越好

	VIPBoard	传统键盘
脑力需求	5.79(1.05)	3.79(1.37)
体力需求	5.36(1.01)	4.00(1.04)
时间需求	5.79(0.89)	3.79(1.81)
付出努力	5.64(0.84)	3.71(1.86)
疲劳程度	5.57(0.65)	3.36(1.34)
总体偏好	6.14(0.77)	3.36(1.28)

³ RM-ANOVA: Repeated Measures-Analysis of Variance, 重复测量的方差分析。

⁴ 李克特量表(Likert scale)是属评分加总式量表最常用的一种,是由美国社会心理学家李克特(Likert)于1932年在原有的总加量表基础上改进而成的。7级量表分为非常不满意、不满意、比较不满意、一般、比较满意、满意、非常满意。

分, 打分项分别为脑力需求、体力需求、时间需求、付出努力、疲劳程度和总体偏好。每一项的平均得分和标准差如表4所示。Wilcoxon 测试⁵表明 VIPBoard 在每一项上都显著优于传统键盘 ($p < .01$)。

对于布局调整的感知: VIPBoard 中的布局调整对于视障用户来说是“透明的”, 因为他们仅依赖于语音反馈来完成输入。当前的调整策略是为了使用的一致性而设计的, 且其中的可视化部分仅供展示使用。几乎所有的实验用户 (12/14) 都能感觉到点击错误率的下降, 但是他们并没有感知到我们的算法在其中所起的作用。在他们的认知中, 他们感觉到能够在交互相同的其中一个键盘上“点击得更加准确”。然而, 还有极少数的非常熟练的用户 (1/14) 可以感知到不同触摸间变化的按键边界。这种变化可能会给这些用户带来按键位置上的困惑, 并稍微影响到他们的信心以及打字的表现。然而, 根据采访结果, 该用户可以非常快地克服这种不适, 适应当前的界面, 因此我们仍然认为 VIPBoard 是视障用户在软键盘上输入文本的一种较好的解决方案。我们可能需要更进一步的长期的实验来研究由布局调整引起的用户心理模型的变化。

结论

VIPBoard 是一个为视障用户设计的应用字符级自动纠错算法来增强读屏键盘的智能键盘。实验表明, 即使只是在字符级, VIPBoard 仍能可以提升 12.6% 的文本输入速度, 降低 63% 的点击错误率, 并被视障用户广泛接受。我们希望 VIPBoard 可以将智能带到读屏软键盘上, 为更多的视障用户造福。 ■



石伟男

清华大学计算机系博士生。主要研究方向为基于语言的交互技术、智能终端技术、文本输入技术等。曾获 ACM CHI 2019 最佳论文提名奖。
swen18@mails.tsinghua.edu.cn



喻 纯

CCF 高级会员, CCF 人机交互专委会秘书长。清华大学计算机系副研究员。曾入选 CCF “青年人才托举工程”、CCF-Intel “青年学者提升计划”、达摩院青橙奖等。主要研究方向为人机交互。
chunyu@tsinghua.edu.cn

参考文献

- [1] Banovic N, Grossman T, Fitzmaurice G. The Effect of Time-based Cost of Error in Target-directed Pointing Tasks[C]// *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI'13)*, 2013: 1373-1382. <https://doi.org/10.1145/2470654.2466181>.
- [2] Bonner M N, Brudvik J T, Abowd G D, et al. No-Look Notes: Accessible Eyes-Free Multi-touch Text Entry[C]// *Proceedings of Pervasive Computing*, 2010: 409-426.
- [3] Oliveira J, Guerreiro T, Nicolau H, and et al. Blind People and Mobile Touch-based Text-entry: Acknowledging the Need for Different Flavors[C]// *Proceedings of the 13th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS'11)*, 2011:179-186. <https://doi.org/10.1145/2049536.2049569>.
- [4] Baldwin T and Chai J. Towards Online Adaptation and Personalization of Key-target Resizing for Mobile Devices[C]// *Proceedings of the 2012 ACM International Conference on Intelligent User Interfaces (IUI'12)*, 11-20. <https://doi.org/10.1145/2166966.2166969>.
- [5] Khaldoun Al Faraj, Mustapha Mojahid, Nadine Vigouroux. BigKey: A Virtual Keyboard for Mobile Devices[C]// *Human-Computer Interaction. Ambient, Ubiquitous and Intelligent Interaction*, 13th International Conference, HCI International 2009, 3-10.
- [6] Bellman T and Mackenzie S. A Probabilistic Character Layout Strategy for Mobile Text Entry. In *Graphics Interface*, 1998:168-176. Retrieved from <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.11.898>
- [7] Azenkot S, Wobbrock J O, Prasain S, et al. Input finger detection for nonvisual touch screen text entry in Perkininput[C]// *Proceedings of Graphics Interface 2012*. 2012:121-129.
- [8] Brewster S A, Chohan F, Brown L M. Tactile feedback for mobile interactions[C]// *Proceedings of the 2007 Conference on Human Factors in Computing Systems*, 2007: 159-162. <https://doi.org/10.1145/1240624.1240649>.

⁵ Wilcoxon 测试是由美国统计学家弗兰克·威尔科克森 (Frank Wilcoxon) 于 1945 年提出的。

- [9] Chen T, Kan M Y. Creating a live, public short message service corpus: the NUS SMS corpus[J]. *Language Resources and Evaluation*, 2013, 47(2):299-335. <https://doi.org/10.1007/s10579-012-9197-9>.
- [10] Chomchalerm G, Rattanakajornsak J, Samsrisook U, et al. Braille dict: Dictionary application for the blind on android smartphone[C]// *Student Project Conference. IEEE*, 2014: 143-146. <https://doi.org/10.1109/ict-ispc.2014.6923237>.
- [11] Cecil D' silva, Vickram Parthasarathy, and Sethuraman N. Rao. 2016. Wireless Smartphone Keyboard for Visually Challenged Users. In *Proceedings of the 2016 Workshop on Wearable Systems and Applications (WearSys'16)*, 13-17. <https://doi.org/10.1145/2935643.2935648>.
- [12] Brian Frey, Caleb Southern, and Mario Romero. 2011. BrailleTouch: Mobile Texting for the Visually Impaired. In *Universal Access in Human-Computer Interaction. Context Diversity*, 19-25.
- [13] Mayank Goel, Leah Findlater, and Jacob Wobbrock. 2012. WalkType: Using Accelerometer Data to Accomodate Situational Impairments in Mobile Touch Screen Text Entry. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI'12)*, 2687-2696. <https://doi.org/10.1145/2207676.2208662>.
- [14] Mayank Goel, Alex Jansen, Travis Mandel, Shwetak N. Patel, and Jacob O. Wobbrock. 2013. ContextType: Using Hand Posture Information to Improve Mobile Touch Screen Text Entry. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI'13)*, 2795-2798. <https://doi.org/10.1145/2470654.2481386>.
- [15] D. Gonçalves, J. A. Jorge, H. Nicolau, T. Guerreiro, and P. Lagoá. 2008. From Tapping to Touching: Making Touch Screens Accessible to Blind Users. *IEEE MultiMedia* 15: 48-50. <https://doi.org/10.1109/mmul.2008.88>.
- [16] Joshua Goodman, Gina Venolia, Keith Steury, and Chauncey Parker. 2002. Language Modeling for Soft Keyboards. In *Proceedings of the 7th International Conference on Intelligent User Interfaces (IUI'02)*, 194-195. <https://doi.org/10.1145/502716.502753>.
- [17] Asela Gunawardana, Tim Paek, and Christopher Meek. 2010. Usability Guided Key-target Resizing for Soft Keyboards. In *Proceedings of the 15th International Conference on Intelligent User Interfaces (IUI'10)*, 111-118. <https://doi.org/10.1145/1719970.1719986>.
- [18] D. Kocieliński and J. Brzostek-Pawłowska. 2013. Improving the accessibility of touchscreen-based mobile devices: Integrating Android-based devices and Braille notetakers. In *2013 Federated Conference on Computer Science and Information Systems*, 655-658.
- [19] I Scott Mackenzie. 2002. A note on calculating text entry speed. Unpublished work. Available online at <http://www.yorku.ca/mack/RN-TextEntrySpeed.html>.
- [20] Sergio Mascetti, Cristian Bernareggi, and Matteo Belotti. 2012. TypeInBraille: Quick Eyes-Free Typing on Smartphones. In *Computers Helping People with Special Needs*, 615-622.
- [21] Hugo Nicolau, Kyle Montague, Tiago Guerreiro, João Guerreiro, and Vicki L. Hanson. 2014. B#: Chord-based Correction for Multitouch Braille Input. In *Proceedings of the 32Nd Annual ACM Conference on Human Factors in Computing Systems (CHI'14)*, 1705-1708. <https://doi.org/10.1145/2556288.2557269>.
- [22] Azenkot S and Zhai S. Touch Behavior with Different Postures on Soft Smartphone Keyboards[C]// *Proceedings of the 14th International Conference on Human-computer Interaction with Mobile Devices and Services (MobileHCI'12)*, 2012: 251-260. <https://doi.org/10.1145/2371574.2371612>.
- [23] João Oliveira, Tiago Guerreiro, Hugo Nicolau, Joaquim Jorge, and Daniel Gonçalves. 2011. BrailleType: Unleashing Braille over Touch Screen Mobile Phones. In *Human-Computer Interaction – INTERACT 2011*, 100-107.
- [24] Mathieu Raynal and Philippe Truillet. 2007. Fisheye Keyboard: Whole Keyboard Displayed on PDA. In *Human-Computer Interaction. Interaction Platforms and Techniques*, 452-459.
- [25] Jaime Sánchez and Fernando Aguayo. 2007. Mobile Messenger for the Blind. In *Universal Access in Ambient Intelligence Environments*, 369-385.
- [26] Keith Vertanen, Haythem Memmi, Justin Emge, Shyam Rey, and Per Ola Kristensson. 2015. VelociTap: Investigating Fast Mobile Text Entry Using Sentence-Based Decoding of Touchscreen Keyboard Input. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI'15)*, 659-668. <https://doi.org/10.1145/2702123.2702135>.
- [27] Daryl Weir, Henning Pohl, Simon Rogers, Keith Vertanen, and Per Ola Kristensson. 2014. Uncertain Text Entry on Mobile Devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI'14)*, 2307-2316. <https://doi.org/10.1145/2556288.2557412>.

- [28]Jacob O. Wobbrock and Brad A. Myers. 2006. Analyzing the Input Stream for Character- Level Errors in Unconstrained Text Entry Evaluations. *ACM Trans. Comput.-Hum. Interact.* 13, 4: 458–489. <https://doi.org/10.1145/1188816.1188819>.
- [29]Georgios Yfantis and Grigori Evreinov. 2006. Adaptive blind interaction technique for touchscreens. *Universal Access in the Information Society* 4, 4: 328-337. <https://doi.org/10.1007/s10209-004-0109-7>.
- [30]Xin Yi, Chun Yu, Weinan Shi, Xiaojun Bi, and Yuanchun Shi. 2017. Word Clarity As a Metric in Sampling Keyboard Test Sets. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI'17)*, 4216-4228. <https://doi.org/10.1145/3025453.3025701>.
- [31]Xin Yi, Chun Yu, Weinan Shi, and Yuanchun Shi. 2017. Is it too small?: Investigating the performances and preferences of users when typing on tiny QWERTY keyboards. *International Journal of Human-Computer Studies* 106: 44-62. <https://doi.org/10.1016/j.ijhcs.2017.05.001>.
- [32]Xin Yi, Chun Yu, Weijie Xu, Xiaojun Bi, and Yuanchun Shi. 2017. COMPASS: Rotational Keyboard on Non-Touch Smartwatches. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI'17)*, 705-715. <https://doi.org/10.1145/3025453.3025454>.