



HAL
open science

Designer Led Computational Approach to Generate Mappings for Devices with Low Gestural Resolution

Roberto Montano-Murillo, Teng Han, Pourang Irani, Diego
Martinez-Plasencia, Sriram Subramanian

► **To cite this version:**

Roberto Montano-Murillo, Teng Han, Pourang Irani, Diego Martinez-Plasencia, Sriram Subramanian. Designer Led Computational Approach to Generate Mappings for Devices with Low Gestural Resolution. 17th IFIP Conference on Human-Computer Interaction (INTERACT), Sep 2019, Paphos, Cyprus. pp.623-643, 10.1007/978-3-030-29381-9_38 . hal-02544535

HAL Id: hal-02544535

<https://inria.hal.science/hal-02544535v1>

Submitted on 16 Apr 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

Designer Led Computational Approach to Generate Mappings for Devices with Low Gestural Resolution

Roberto Montano-Murillo¹, Teng Han², Pourang Irani², Diego Martinez-Plasencia¹ and Sriram subramanian¹

¹ Department of Informatics / Interact Lab, University of Sussex, Brighton, United Kingdom
{R.Montano-Murillo, D.Martinez-Plasencia, Sriram}@sussex.ac.uk

² Department of Computer Science, University of Manitoba, Winnipeg, Manitoba, Canada
{hanteng, Pourang.Irani}@cs.umanitoba.ca

Abstract. We present an approach for the semi-automatic generation of gesture mappings for devices with low gestural resolution such as the Myo Armband, an off-the-shelf EMG capture device. As an exemplar interactive task, we use text-entry: a pervasive and highly complex interaction. We quantify data related to interaction combining systematic studies (i.e., error, speed, accuracy) and semi-structured workshops with experts (e.g., cognitive load, heuristics). We then formalize these factors in a mathematical model and use optimization algorithms (i.e. simulated annealing) to find an optimum gesture mapping. We demonstrated our method in a text-entry application (i.e., complex interactive dialogue) comparing our approach with other computationally determined mappings using naive cost functions. Our results showed that the designers mapping (with all factors weighted by designers) presented a good balance on performance in all factors involved (speed, accuracy, comfort, memorability, etc.), consistently performing better than purely computational mappings. The results indicate that our hybrid approach can yield better results than either pure user-driven methodologies or pure data-driven approaches, for our application context featuring a large solution space and complex high-level factors.

Keywords: gestural interaction, semi-automated interaction design, optimization, computational approaches.

1 Introduction

Gestures play an inherent role in our everyday communication, to the extent that we make use of them even when our interlocutor is not present, such as when speaking on the phone [26]. Gestures can be used to communicate meaningful information (*semiotic*), manipulate the physical world (*ergotic*) or even to learn through tactile exploration (*epistemic*) [4]. *Semiotic* gestures have been of particular interest to the HCI community as a powerful way to communicate with computers [23, 27].

The creation of interfaces involving gestural interaction remains a challenge. On one hand, advances in hardware have been remarkable. Gestural interaction is no longer restricted to data-gloves [7, 16, 34], and there is an increasing range of potential devices, allowing gesture tracking on un-instrumented hands or even in mobile formats. On the other hand, the methods and approaches to design these experiences have followed a much slower progression, not copying with the increasing number of devices available, and still relying on iterative methods and designers' expertise [9, 30].

As a result, interaction designers are faced with a very challenging task, with many factors involved in the creation of the gestural interface. While some factors will be

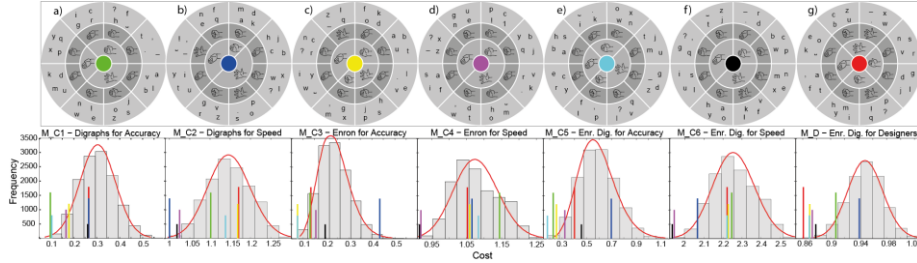


Fig 1: From left to right (Top), the resulting mappings from the full optimization using different training database and cost function’s factors. Below each layout, its histogram is shown. The cost per layout is represented along all histograms using color code (M_C1=green, M_C2=blue, M_C3=yellow, M_C4=magenta, M_C5=cyan, M_C6=black and M_D=red).

easy to assess (e.g., device’s comfort, accuracy, speed), others will be more complex (e.g., social acceptability and cognitive load). Particularly challenging is the elicitation of the most appropriate gestures and their mapping to tasks, which can easily lead to a combinatorial explosion. For instance, our example case study (text entry) offers more than 35K ways to map gestures to input commands and more than 12K ways to map these to actual letters. While iterative methodologies, designers’ intuition and heuristics might help, it will be costly to navigate this vast solution space and identify the optimum interactive dialogue. In contrast, computational approaches might struggle to capture the complex subjective factors (i.e. social acceptability or cognitive load).

Unlike previous methods, we propose a hybrid approach, merging designer-led methods and computational approaches for the generation of robust gestural mappings under such challenging conditions (i.e. large solution space involving complex high-level factors). More specifically, we present an expert-guided, semi-automated design of interactive dialogues for low gestural resolution devices. Our approach consists of four steps: **i) quantify low-level factors** (gesture error rates, speed or accuracy); **ii) semi-structured workshops with designers** (identify *higher-level* factors, such as cognitive load and experts’ heuristics); **iii) formalization & optimization** (using objective and designers’ knowledge to produce a mathematical model, and compute an optimum mapping); and **iv) comparative evaluations** (to guide the iterative interface design, in a cost-effective manner).

We demonstrate this approach applying it to the design of a text entry technique using a Myo device. Fig 1(g) shows the result – a multi-level mapping between the input gestures and characters for text entry. To assess the value of our approach, we compared the mapping produced from our hybrid approach (incorporating designers’ *high-level* factors) to several purely computational, naïve mappings. Particularly, we defined 6 alternative cost functions (i.e. models to assess the quality of a mapping) optimizing for time and accuracy, and explored up to 2.7 billion possible mappings, finding the optimum mapping for each of the 6 naïve cost functions.

Fig 1 shows histograms for all these mappings according to: the naïve computational metrics (a-f) and our approach (g). The optimum mappings computed are also highlighted within each histogram (bars). These show that, while naïve functions are highly ranked according to the designers-led metric (i.e. low scores, in Fig 1g), the designers-led mapping ranked relatively poorly according to each of the 6 naïve cost functions

used (red bar showing high values in Fig 1(a-f)). This could either point towards designers' insight being irrelevant (or even harmful) or to computational methods failing to capture the complexity of the task. The results from our study show that the designers-led mapping actually showed a good balance on performance in all factors involved (speed, accuracy, comfort, memorability, etc.), consistently performing better than purely computational mappings. This reveals an untapped power in the designers' ability to identify a good cost function, with our approach helping to produce a suitable formalization to exploit the exploratory potential of computational approaches.

We finish the paper reflecting on these results and on how they should open a discussion on the added value of designers' intuition and heuristics when exploring gestural interfaces, and the need to make these an integral part of current design methodologies, for large solution spaces.

2 Related Work

2.1 Gestural Input Devices: A growing landscape

An increasing number of device options are available to support gestural interaction. Early instances included data gloves and tracking systems, mostly used for Virtual Reality [34] and multimodal interaction. These provide high gestural resolution (i.e. high number of distinct gestures), but require user instrumentation, hindering their applicability (i.e. users cannot simply walk-up and use them, wires limit mobility, etc). Wireless tracking systems (e.g., Leap Motion, Kinect, Project Soli) can improve applicability [6, 33] but their sensors are typically fixed, constraining the user to specific working spaces.

Mobile solutions have also been proposed. Kim et al. [17], presented a wrist-mounted optical system, allowing for hand gestural interaction. Myo armbands use Electromyography (EMG) to record and analyse electrical activity, allowing lightweight mobile gestural input, without hindering the use of our hands and avoiding self-occlusion problems. EMPress [22], combines EMG and pressure sensors, providing the same affordances of Myo bands, but with improved gestural resolution. Solutions to extend smartwatch interaction with around device gestural interaction have also been explored [20], but they either provide limited gestural resolution [15] or involve instrumenting the user's gesturing hand [36].

2.2 Gestures and mappings: point studies

The HCI literature has produced a plethora of studies, which can help designers deal with the increasing number of device options available. Sturman et al. [31] explored and provided guidelines to improve gestural interaction in VR. Studies from Rekimoto [25], Wu & Balakrishnan [35] provide insight in the context of interactive surfaces, and Grossman et al. [12] explored the topic in the context of 3D volumetric displays, just to mention some. However, these illustrate how information related to gestural interaction is scattered across individual point studies, focused on specific tasks and contexts.

A more general approach to designing gestural interaction has been to formalize user elicitations [10, 14]. Designers seek end-user input on mapping gestures to tasks, classifying gestures into *high-level* groupings based on salient properties (e.g., the direction of movement, finger poses, etc). Elicitation studies have been successfully used in a number of contexts, but have also been criticized for biasing results by basing them on input from populations unfamiliar with the task or capabilities of a device [5, 10].

Alternatively, designers can gain insight about the mapping between gestures and tasks from related literature. Focusing on text entry (closest to our case study), the QWERTY keyboard serves as a preeminent example of discrete mapping, enforcing a 1:1 mapping between each key (gesture) and a letter (task). It also illustrates a mapping designed around the mechanical limitations of past typing machines, rather than its appropriateness for human input.

Computational approaches have proved to be valid tools to identify better mappings. Zhai et al. showed clear improvements for clarity (avoid gesture ambiguity) and typing speed for the most common digraphs in English [2] by simply swapping two keys (I and J). Bi et al. [1] explored alternative mappings by swapping a few neighbouring keys, to get a layout with better performance on speed, while retaining QWERTY similarity. Smith et al. showed a similar approach, improving clarity, speed and QWERTY similarity for 2D gesture typing. Alternatives for situations where 1:1 mappings are not available (e.g., mobile phones) have also been tackled using computational approaches, mostly through predictive text entry models [11, 24]. Other works have focused on exploring the extent of human hand’s dexterity, creating mappings that benefit from all its bandwidth. Oulasvirta et al. [29] explored the biomechanical features of the hand (flexion levels, inter-digit dependencies), while PianoText [8] leverages users’ musical skills, using a piano keyboard and chords to create an ultrafast text-entry system. In all cases, the benefits of computational approaches are limited by the use of low-level, quantifiable factors.

This situation motivates our approach. Interface designers might rely on methods that introduce biases into the process and will struggle to iteratively explore large solution spaces. Alternatively, computational approaches have great exploratory power, but they might fail to capture *higher-level* aspects of such complex tasks as they tend to bias/limit their results towards quantifiable factors that are easy to assess. Our approach intends to bridge this gap, being the first one to put together the benefits of both approaches (designers-led vs computational solutions), by blending designers’ methods/insight and computational approaches.

3 Our approach: Semi-automatic Mappings for Low Input Resolution

Our method aims to bridge the differences between designer-led and computational solutions, capturing designers’ tacit knowledge of the domain, and formalizing it to be exploited by computational approaches. We thus combine *quantitative parameterization of relevant factors* with *domain expert knowledge elicitation*, into a structured approach. We refine these into a *formal model* quantifying the quality of each mapping

and using a *global optimization algorithm* to explore the solution space, finding (potentially) the best solution. Our approach is compatible with iterative methodologies and can be seen as the tasks required for one iteration cycle. The outline of our approach can be divided into four stages:

(i) Quantification of Low-level Factors and constraints

This stage involves the experiments and in-lab tests required to measure and quantify *low-level factors* and *constraints*. *Low-level factors* are simple parameters (e.g., time, errors) associated with the device or modality that might influence the design of the mapping and are easily quantifiable. *Low-level constraints* represent limitations within the device or the way it is used. Using our case study as an example, *factors* can include time to perform each Myo gesture, while excluding the double tap gesture due to its low accuracy can be an example of a *constraint*.

These quantified values will be used in the two following stages: First, they will inform designers, to help produce mappings and formulate heuristics; Second, they provide quantifiable data, used by our optimization methods.

(ii) Domain Expert knowledge elicitation

We use small teams of experts as a way to elicit the relevant factors that need consideration to design the interactive dialogue. Different methodologies can be used (e.g., workshops, elicitation studies, prototypes), which help on addressing a broad spectrum of aspects that cannot be covered by computational approaches alone (e.g., interface design, feedback elements, definition of the interactive dialogue, etc.).

However, while designers must consider the mapping of gestures to tasks, the ultimate intent of this process is *not* the specific mapping they create (computational searches will help make this specific choice). Instead, we focus on the designers' rationale that they use to determine what might be a good choice of gestures and mapping.

We reflect this rationale as *constraints* (i.e. conditions that must be obeyed) and *high-level factors* (i.e. non-obvious aspects or heuristics affecting interaction, such as social acceptance). These will help our following formalization process and the weighting of the relative importance of each of these factors.

(iii) Formalization & Optimization

In order to optimize our mappings, we first need to provide a metric for the quality of any given mapping. We formalize the quality of a mapping M as a cost function C computed as a weighted average of the factors identified by the experts, with lower values identifying better mapping:

$$C(M) = \sum k_i \cdot \text{Factor}_i(M) \quad (1)$$

The different factors are all normalized to a homogeneous range $[0, 1)$, according to the maximum and minimum values observed from the quantification. The value for k_i (influence of a given Factor_i in the mapping M) needs to be estimated from the experts'

impressions and analysis (further details follow). This assures that the contribution of each factor to the quality of M is the result of the designer’s insight, and not the result of the factors’ relative orders of magnitude. In our example, the sum of factor weights ($\sum k_i$) equals one (factor as a ratio), but any other weight distribution reflecting the expert’s impressions can be used. We then use a global optimization method to explore the solution space, converging towards an optimized solution given the factors and weighing values identified. Although our case study used Simulated Annealing [18], other optimization approaches can also be used.

(iv) Comparative-Summative Evaluation

While the normalization of the factors identified follows quantitative criteria, the estimation of the weight distribution (k_i) does not, and it relies on the subjective assessment of domain experts. Different weight distributions might reveal different ways of thinking about the solution (e.g., how more relevant is minimizing time over cognitive load?). Computing optimized mappings, according to different weight distributions, and comparing them through summative evaluations can allow for the best mapping to be identified. This reduces the exploration of the solution space to a few candidates (each resulting from a different weighting strategy), and integrates easily with iterative methodologies for gestural interaction, such as [9].

4 Case Study with Myo: Compute Vs Design

We tested our approach using a Myo device (i.e. very low gestural resolution) for a text-entry task, both as a worst-case scenario and as an obvious match to Foley’s analogy between natural language and a general interactive dialogue. The in-built IMU was not used and only the muscle activation was considered. This reduces our gestural resolution even further (more challenging solution space) but it also lends itself to interesting application scenarios. IMU-based gestures are defined relative to the body, and might be restricted during our daily life (i.e. while sitting in a bus, walking or inside a busy elevator). In contrast, our gestures remain relative to the hand, being still available in any situation where the wrist can be moved.

Finally, we also wanted to assess the added value of our designers’ guided approach when compared to unconstrained computational approaches, based on observable and quantifiable factors alone. We replace the last stage of the method (iv), by a description of the naïve computational mappings used, and a comparison against the results provided by these alternative approaches.

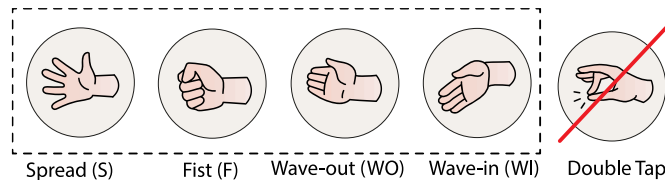


Fig. 2. Gestures possible with a Myo armband. We used the enclosed gestures in this work.

4.1 Problem Delimitation

Although Myo supports up to five gestures, at the time when this work was carried out “Double tap” was a recent addition with known inconsistencies in its detection [32]. Also, any fast and consecutive pair of gestures was detected as “Double tap” (i.e. false positives), conflicting with the use of other potential gesture chains. For that reason, only the four remaining gestures were used (see spread (*S*), fist (*F*), wave-out (*WO*) and wave-in (*WI*), in Fig 2). We quantified the performance of 16 possible 2-step chain gestures (consecution of two gestures, as in Fig 3). Such 2-step chains require an intermediate relax action (i.e. hand returning to a neutral status between gestures) to be recognized by the system.

We asked our designers to categorize the 2-step chain gestures and they identified three different groups: *opposite*, *orthogonal* and *repeat*. *Opposite* chains combine gestures that activate opposing muscles. *Orthogonal* chains invoke orthogonal muscle groups; and *Repetitive* chains contain two instances of the same gesture (see Fig. 3). For example, WI+WI is a *Repeat*, WI+WO is of type *Opposite*, and WI+F is of type *Orthogonal*. We will borrow this for the analysis in this section (even if the distinction only appeared during the later workshops), as its analysis allows us to assess to what extent designers’ insight reflects trends in data, or if some aspects pointed by designers would be likely to be included or ignored by alternative purely computational approaches. Finally, we also conducted a similar study for 3-step chain gestures. However, designers soon disregarded these chains during the later workshop (only use 2-chain gestures – C1), so our results for 3-step chains are omitted here for brevity.

(i) Quantification of Relevant Factors. We conducted a quantitative study, where participants performed a series of 2-step chain gestures under different input speeds to evaluate potentially relevant factors (i.e. errors, ergonomics, and preferred 2-step chain gestures). We calibrated the Myo for each individual participant and allowed them to become familiar with the 4 Myo gestures (Fig. 2) and our 2-step chain gestures (Fig. 3). They were then asked to perform the 2-step chain gestures shown on a display, which changed at regular speeds (i.e. each single gesture shown during 0.6s, 0.8s, 1.0s or 1.2s). Participants were asked to complete the gestures accurately and within the length of the prompts, which helped us identify the appropriate “typing speed”.













Opposite	Orthogonal	Repeat
WI - WO 	WI - S 	WI - F 
WO - WI 	WO - S 	WO - F 
S - F 	S - WI 	S - WO 
F - S 	F - WI 	F - WO 

Fig. 3. Two-step chain gestures under designers’ categories.

Accuracy per gesture (F2). Fig. 4 (b) shows our results for accuracy, revealing overall accuracy is low (70% - 90%). An ANOVA analysis revealed an effect of gesture on accuracy (used as factor **F2**). Again, significant differences were found between specific pairs of gestures, but we focus the analysis on designers' categories. We only found significant differences between Repeat ($M=86.8\%$; $SD=21.57\%$) and Orthogonal categories ($M=81.28\%$; $SD=24.45\%$; $p=0.032$), but with reduced effect size. Also, no clear patterns could be observed by looking at the categories (values well above and below the mean are present in all categories, in Fig. 4 (b)).

Gesture Comfort (F3). Comfort was rated by participants using a Borg CR10 Scale [3] questionnaire (Fig. 4 (c) shows the average of participants' effort per gesture). According to their answers, we found *Repeat* gestures as the most comfortable ($M=1.5$, $SD=0.33$) followed by *opposite* gestures ($M=1.66$ BCR10 and $SD=0.2$) and the most uncomfortable reported were *orthogonal* gestures ($M=2.35$ BCR10, $SD=0.38$). It is worth mentioning that due to the number of trials (192) during the experiment, fatigue could potentially affect participants' performance. However, as shown in Fig. 4 (c), the maximum score of effort was about 3.2 (in a scale from 0 to 10) suggesting that although we could observe differences in effort (e.g. *orthogonal* gestures were more uncomfortable), participants gave generally low scores in effort and therefore we considered unlikely that these low scores represent a negative effect on participants' performance during the experiment.

Typing speed of 1 second (C2). The effects of typing speed on gesture time (Fig. 5 (a)) and accuracy (Fig. 5 (b)) were also analyzed. This revealed the first gesture ($M=0.783s$; $SD=0.119s$) is significantly shorter than the second one ($M=0.843s$; $SD=0.109s$), and also more accurate ($p=0.012$). Using an input speed of 0.8s users barely could keep up with the input speed (first gesture $>0.8s$, accuracy significantly smaller than input at 1.2s ($p<0.001$)). It is interesting how users (even if allowed more time) did not take more than 0.97s to perform each gesture. No significant differences were found for typing speeds of 1s or 1.2s. Thus, we included typing speed of 1s (**C2**) as a *low-level constraint* (i.e. fastest speed allowing sustained typing).

(ii) Designer's Workshop. After obtaining the relevant low-level factors, we carried out a workshop with interaction designers, as a way to identify the design rationale they

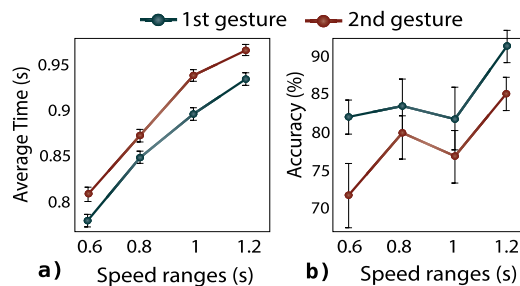


Fig. 5. a) Average time for the first and second gesture; b) Average accuracy for the first and second gesture. Error bars represent standard error.

use in producing their mappings. We motivated the workshop around the concept of gestural text-entry, a challenging context forcing them to explore the topic in depth.

We recruited four UX designers (no specific expertise on text-entry) from Anonymous University HCI group (other than where the main study was conducted), to produce a design scheme for the system. The workshop session lasted four hours. To encourage a broad perspective towards the design of an effective interactive dialogue, designers were encouraged to think about these four questions: *How to map gestures with letters? What is a good interface layout? What feedback elements are required? Is the operation easy to remember?* The workshop was kept open-ended to encourage creative thinking, but one researcher stayed in the room, to answer designers' questions. It must be noted that the quantitative results from (i) (e.g., speed, accuracy) were only provided if and when specifically requested by designers, to not bias their thoughts.

At the beginning of the workshop, designers considered using chained gestures right away. Three-chain gestures were soon discarded by designers, due to their high cognitive load (too many potential gestures to remember) and discomfort (orthogonal and opposite gestures). Thus, they limited their search to 2 step-chain gestures (C1) and a predictive text entry. This used 8 categories, mapping 4 letters to each gesture/category and addressing 32 characters: the 26 letters from the English alphabet and the 6 most common punctuation characters (space, period, comma, question mark, exclamation mark and hyphen). They also felt inclined to explore alternatives beyond the constraints defined (such as using both hands or using continuous gestures, using the duration of the gesture as a variable). At the end of the workshop, designers were asked to present their interface layout and to reflect on it, as a way to verbalize their rationale. In the next subsection, we report these observations as *high-level factors* and *constraints*.

From designers' rationale to factors and constraints. Designers soon got interested in the time (F1) and accuracy (F2) of each gesture and experimented the level of comfort (F3) afforded by each gesture by performing them casually. They considered the *WI* gesture to be the most 'natural' gesture, and *WO* as the least comfortable. They also found the *F* and *S* gestures hard to perform. Designers also became interested in the frequency of using each letter, using the ENRON corpus [19] to inform this aspect.

At the end of the workshop, they presented their proposed interface design (see Fig. 6 (a)), reflecting both the appropriate interface design and the way the interactive dialogue should work. The UI layout consisted of several concentric circles, working as a decision tree with choices at each node. Users would identify the target letter in the external level/ring and then follow the path through the ring from the inside out, performing the gestures to reach the chosen letter. The interface should highlight the rings, as gestures are recognized, e.g., Fig. 6 (b), shows *Fist + Spread* gestures used to type 'q', and feedback displayed.

The final scheme presented reflected aspects of their rationale (*high-level factors*), highly relevant for our approach. For instance, they attempted to maximize the usage of *WI* (F4), while avoiding *WO* (F5) and *S* gestures (F6). They also found the use of orthogonal gestures very uncomfortable and suggested avoiding them (C3).

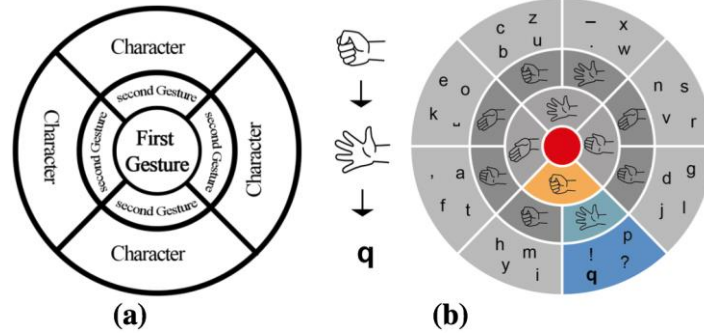


Fig. 6. a) Interface layout proposed by designers; b) Final design using their factors and our search method. Typing a “q” requires to perform the chain gesture fist (F) - spread (S).

As a second major concern, designers also attempted to reduce the cognitive load of the mapping, by applying several heuristics. For instance, they suggested keeping all vowels clustered together (in two categories only) (F7). They also placed alphabetically adjacent letters in the same categories (e.g., “abcd”), which was considered as a relevant factor (F8). These techniques were meant to facilitate users' ability to remember the layout.

Designers also tried to assign the comfortable and fast gestures to the most frequent characters. They attempted to build a mapping solving the problem in an optimal way, and including all identified factors. However, they failed to find a clear candidate mapping, illustrating the challenge designers face when addressing large solution spaces.

(iii) Formalization & Optimization. We used the *constraints* (C1-C3) and *factors* (F1-F8) identified in the previous stages to refine our definition of the problem and to formalize the description of our candidate mappings. Due to our constraints, we limited our search to 2-step chain gestures (C1), with typing speed 1s (C2) and used only “opposite” and “repeated” gestures (C3), resulting in only 8 possible gesture chains (see Fig. 3).

Each factor was formalized (quantified), with the common criteria that lower values represent a better mapping. Let D be our dictionary (we use the ENRON database [19], with duplicates to represent word frequency). Let W be a word and L a letter. Let $Time(L)$, $Accuracy(L)$ and $Exertion(L)$ be the mean time, accuracy and effort (i.e. the inverse of comfort) of the gesture associated with letter L , as measured from our quantitative studies from (i).

Time factor (F1). This factor favours fast typing speeds, by quantifying the “average time to input a letter according to our dictionary”.

$$F1(M) = \sum W \text{ in } D \sum L \text{ in } W \frac{Time}{|D||W|} \quad (2)$$

Accuracy factor (F2). This factor enforces mappings with gestures of high accuracy recognition, by quantifying the “probability to make one (or more) errors in a word”.

$$F2(M) = \sum W \text{ in } D \sum L \text{ in } W \frac{1 - \text{Accuracy}(L)}{|D|} \quad (3)$$

Comfort factor (F3). This factor measures the “amount of exertion required to input a letter”, to minimize effort.

$$F3(M) = \sum W \text{ in } D \sum L \text{ in } W \frac{\text{Exertion}(L)}{|D||W|} \quad (4)$$

Wave-in factor (F4). This factor encourages the use of *WI* gesture, considered comfortable by designers. This factor computes "the average density of non-*WI* gestures per letter”.

$$F4(M) = \sum W \text{ in } D \sum L \text{ in } W \frac{\text{isNot } Wi(L)}{|D||W|} \quad (5)$$

Wave-out factor (F5). This factor discourages the use of *WO* gesture, as it was considered less comfortable. Particularly, it quantifies "average density of *WO* gestures per letter”.

$$F5(M) = \sum W \text{ in } D \sum L \text{ in } W \frac{\text{is } Wo(L)}{|D||W|} \quad (6)$$

Spread factor (F6). This factor penalizes the use of *S* gestures, as they were considered less comfortable. This factor computes the "average density of *S* gestures per letter”.

$$F6(M) = \sum W \text{ in } D \sum L \text{ in } W \frac{\text{is } S(L)}{|D||W|} \quad (7)$$

Vowels factor (F7). This factor counts the “number of categories containing vowels”, to favour vowels are grouped in a few categories.

$$F7(M) = \max(|V|), V \subset C / \forall c \in V, \{a, e, i, o, u\} \cap c \neq \emptyset \quad (8)$$

Consecution factor (F8). This factor benefits mappings where letters are assigned to categories in consecutive order. Thus, it measures the "number of non-consecutive (NC) letter per category (C)”.

$$F8(M) = \frac{NC(C[0],C[1]) + NC(C[1],C[2]) + NC(C[2],C[3])}{3} \quad (9)$$

Determining the weight of each factor and optimization. Each factor was normalized to a $[0, 1)$ range, as in Table 1. This allows the relevance of each factor to be assessed in terms of weight alone (and not according to the factor’s scale). Constants s_w and l_w represent the length of the shortest and longest words in D , respectively; m_t and M_t stand for the minimum and maximum gesture times, and m_a and M_a stand for the minimum and maximum gesture accuracy respectively. Weights were then determined based on

Table 1. Factors used use for MDes (our proposed mapping), ranges and weights (ki).

	F1	F2	F3	F4	F5	F6	F7	F8
Min	$ s_w \cdot m_t$	$ s_w \cdot (1 - m_a)$	0.125	0	0	0	2	0
Max	$ l_w \cdot M_t$	$ 1w \cdot (M_a)$	1.75	1	1	1	5	1
Ki	0.35	0.20	0.1	0.05	0.05	0.05	0.1	0.1

the designers' insight. It must be noted that this was the interpretation of the research team (i.e. two transcribing and cross-validating notes from the experiment, and two translating them into the weights described in Table 1), as we had no further access to the designers involved in **(ii)**.

We used these weights (cost function as described by Eq(1)) and simulated annealing (SA) [18] to find the optimum mapping. Initially, letters were randomly assigned to the 8 categories (only "opposite" and "repeated" gestures, see Fig. 3) and neighbour states were computed by permutation of single letters between two random categories (diameter=32). Transition acceptance between states follows the traditional method by Kirkpatrick [18]. The cooling schedule was empirically tuned with $N_s = 20$ step adjustments per temperature step, $N_t = 7$ temperatures steps per temperature change, $R_t = 0.85$ (Cooling factor). The initial temperature was set in $T(0)=180$. The final mapping is shown in Fig 7 (b) and Fig 7 (c).

Given the designers' constraints (no *Orthogonal* gestures), the solution space was limited to $\binom{32}{4} = 35960$ mappings and a full search would have been feasible. However, this was not feasible for the pure computational solutions we compared against (larger solution space), and we used the same schedule to aid fairness in comparison.

(iv) Computing alternative approaches. Some of the factors and observations made by designers were hard to justify purely looking at the data. The categories identified (*Repeat*, *Opposite* and *Orthogonal*) show weak differences and, given any performance metric, all of them have gestures both well above and below the sample mean. Even in the case of time per gesture (clearer distinctive behaviour for *Opposite*), the use of clustering techniques would not result in the categories identified.

Picking specific data could seem to back up the designers' insight. For instance, *WI+WI* was the most comfortable gesture ($M=1.15$ Borg CR10 Scale –BCR10) and *WO+S* as the least comfortable ($M=3.15$ BCR10), followed by *S+WO* ($M =2.6$ BCR10). While *WO+WI* resulted the fastest 2-step chain gesture ($M=1.947s$, $SD=0.228$), *WI+WI* was second fastest ($M=1.949s$, $SD=0.242$), the most accurate ($M=95.13\%$, $SD=13.73$) and the most comfortable gesture performed ($M=1.15$ BCR10), whilst *WO+S* the least comfortable ($M=3.15$ BCR10).

These point observations could support designers' factors **F3** and **F4**, but observational bias and the limited size of the sample would make for weak evidence. This was found worrying, as it could point towards a weak ability of the designers to analyse the complexity of the problem. On the other hand, factors could also reflect designers' *tacit knowledge*, that is, understanding of complex mechanics of the task which were difficult to articulate, but still relevant. Thus, we decided to compare the designer's guided solution against six naïve computational solutions, not considering designers' *high-level factors* and *constraints* (e.g., 8 categories used to allow comparison, but not constrained to *Repeat* and *Opposite* gestures alone). These naïve solutions will both help us assess the added value introduced by feeding the designers' insight into the optimization method; and also challenge their decisions/constraints.

These six solutions were generated as a combination of two elements: **a**) the training dataset: the Enron (E) dataset [19]; the most common Digraphs (D) in English language [21], and a combination of both (E+D); and **b**) the cost functions: two were defined,

Table 2. The percentile per mapping (0 to 100) across the seven cost functions (CF) used in the optimization process. On the right columns, AVG and SD for the data per CF condition are shown. The best mappings for speed (M_C4) and accuracy (M_C5) are highlighted in green while our proposed mapping (M_Des) is highlighted in blue.

Cost/Mapping	M_C1	M_C2	M_C3	M_C4	M_C5	M_C6	M_Des	Average	SD
C1-A _{Acc}	0	33	5	3	1	31	33	0.304	0.077
C2-A _{Sp}	21	0	65	1	45	1	67	1.147	0.052
C3-E _{Acc}	4	100	0	11	1	29	5	0.238	0.066
C4-E _{Sp}	81	50	45	0	60	1	40	1.077	0.064
C5-D+E _{Acc}	1	80	1	3	0	20	10	0.587	0.014
C6-D+E _{Sp}	49	4	42	1	40	0	41	2.258	0.110
D-D+E _{Mix}	10	40	1	1	1	1	0	0.949	0.025

one assessing time per gesture (factor *F1*) and another one assessing accuracy (*F2*). e.g., M_C1 represents the mapping obtained with the best *Accuracy* assessed by Di-graphs dataset. For each of the six combinations, we generated all the possible subsets of 8 gestures (from the 16 different 2-step gestures possible) and used Simulated Annealing to compute the best letter combinations. We explored $\binom{16}{8} \cdot \binom{32}{4} \cdot 6 = \sim 2.8$ billion possible mappings, with Fig. 1 showing the best mapping for each of the 6 naïve cost functions.

5 Analytical and Summative Evaluation

Fig. 1 shows histograms for all possible mappings according to our seven metrics (the naïve computational metrics (a-f) and designer-led (g)). The best mappings per metric are also highlighted (as colour bars) in the remaining histograms, for comparison. Table 2, shows this information in a numerical format. The best results for Accuracy mappings (i.e. M_C1, M_C3 and M_C5) was M_C5 (best average percentile across the 6 naïve functions, within its category), while the selected mapping for Speed (i.e. M_C2, M_C4 and M_C6) was M_C4. For clarity, during the comparative evaluation, we will refer to these as time (M_{Ti}), and accuracy mappings (M_{Acc}), instead of (M_C4 and M_C5).

It was also interesting to see how the designers-led layout (M_{Des}), rated against the other mappings. While computational mappings consistently scored well using the designers’ cost function (see the last row), the designers mapping scored much more mediocre results (see column M_{Des}), being usually in fourth or fifth position (or even last) among the mappings considered. We then carried out a user study to evaluate the performance of the generated mappings: M_{Ti}, M_{Acc} and M_{Des}. We added one additional mapping for text-entry i.e. a simple alphabetical distribution (M_{Abc}) shown in Fig. 7 (d), as a baseline comparison (minimum cognitive load, not optimized).

5.1 Experiment Setup

At the beginning of the session, we calibrated the Myo for each individual participant. Subsequently, each mapping was shown on the screen with its different layout and letter distribution (see Fig. 7). Participants were then instructed to “type” a sentence shown

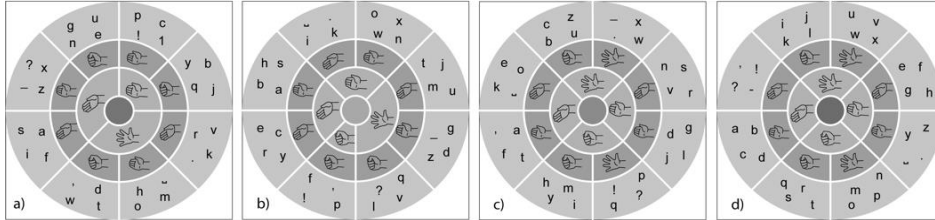


Fig. 7. a) Gesture mappings of time factor (M_{Ti}), (b) accuracy factor (M_{Acc}), (c) mixed mapping according to designers' factors (M_{Des}) and d) alphabetical gesture mapping (M_{Abc}).

above the circle by performing the specific chain of gestures (i.e. identifying the two gestures they need to perform to select a given letter). The system included feedback cues i.e. visual highlights in the category selected at each step (see Figure 6 (b)), and auditory effects.

Participants were allowed to practice the chain gestures in a training stage to complete 4 sentences before each block, in order to get familiar with the layouts. Participants performed 4 blocks of 3 sentences each, completing 28 sentences in total (700 letters/gesture chains). The sentences in the blocks had from 4 to 6 words, and 4 to 6 letters per word, being selected by using the Levenshtein algorithm [13] to compute representative sets of sentences from our dictionary. The full experiment duration was 45min. Similarly, as described in the first study, each block was designed to be completed in about 8min giving participants a 3min break between blocks to avoid fatigue. Moreover, since *orthogonal* gestures (the most uncomfortable gestures found in the first study and rated on average ~ 3.2 in a scale from 0 to 10) were not employed in this study, we considered unlikely that fatigue negatively affects participants' performance during the experiment. We counterbalanced the order of the sets (i.e. sentences) and mappings using a 4×4 Latin Square design. Fig. 8 shows our experimental setup.

The system collected the time per letter and error rate automatically. User-satisfaction questionnaires after each block (mapping), collected information about typing comfort and how easy each it was to remember each mapping. Finally, at the end of the experiment, participants also chose their favourite mapping according to 4 aspects (easy to type, comfort, speed and easy to remember). Sixteen right-handed participants took part in the experiment (4 Females, average age of 29.33, $SD= 3.86$), which was approved by the local ethics board. The recruitment criteria were the same as in the first experiment.

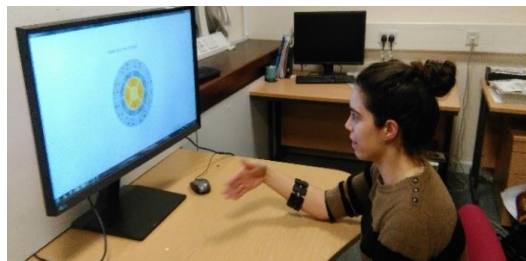


Fig. 8. Experimental setup for the typing task.

An a priori statistical power analysis was performed for sample size estimation in G*Power. Running a power analysis on a repeated measures ANOVA between mapping conditions (i.e., M_{Ti} , M_{Acc} , M_{Des} and M_{Abc}), repeated 28 times corresponding to the 28 sentences on the experiment, a power of 0.95, an alpha level of 0.05, and a medium effect size ($f=0.196$, $\eta^2=0.037$, critical $F=1.1$), required a sample size of approximately 8 participants. Thus, our proposed sample of 12 participants was adequate for the purposes of this study.

5.2 Analysis of Results

A Repeated Measure ANOVA was conducted to compare the effect of the four types of mappings (M_{Ti} vs M_{Acc} vs M_{Des} vs M_{Abc}) on the time of chain of gestures. Results revealed a significant effect on the average time, $F_{(3,45)}=25.82$, $p<.001$ depending on the type of mapping, with the designers' mapping providing best results. Post-hoc comparisons using Bonferroni correction showed statistically significant differences in time, specifically between M_{Des} ($M=1.577s$, $SD=0.622s$) compared to M_{Abc} ($M=1.785s$, $SD=0.674s$; $p<0.001$), but also M_{Des} and M_{Ti} ($M=1.782s$, $SD=0.653s$; $p<0.001$). No such difference was found compared to M_{Acc} ($M=1.64s$, $SD=0.71s$), $p=0.279$. Surprisingly, M_{Ti} did not provide the best results for time, which seems to indicate it failed to capture the complexity of the typing task.

The average error per mapping was small for all conditions. As expected, M_{Acc} got the lowest error score as it was computed to minimize errors. Repeated Measure ANOVA showed a significant effect of the type of mapping (M_{Ti} vs M_{Acc} vs M_{Des} vs M_{Abc}) on the number of errors $F_{(3,45)}=7.71$, $p<.001$ ($\eta^2 = 0.009$ small effect). Post-hoc comparisons showed statistically significant differences for errors, specifically between M_{Acc} ($M=0.072$ errors, $SD=0.293$ errors) compared to M_{Des} ($M=0.139$ errors, $SD=0.444$ errors), $p=0.001$ and M_{Abc} ($M=0.149$ errors, $SD=0.520$ errors), $p=0.001$; but no such difference was found compared to M_{Ti} ($M=0.087$ errors, $SD=0.369$ errors), $p=1$. Additionally, we found a significant difference in M_{Ti} compared to M_{Des} , and M_{Abc} , $p<=0.035$. These results suggest that M_{Acc} and M_{Ti} produced the lowest number of errors when participants performed the gesture chains to "type" the sentences.

Fig. 11 shows the score given by participants after each block in relation to memorability (left) and comfort (right). In both cases, participants gave higher scores to M_{Acc}

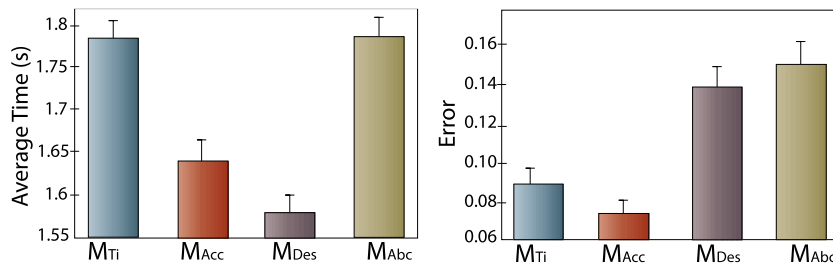


Fig. 9. Scatter plot of average gesture time (left) and errors (right) per mapping. Bars represent standard error of mean.

and M_{Des} , with worse results for M_{Ti} . These results align with the user’s final appreciations at the end of the experiment, in which participants compared among all mappings (see Fig. 10). In this case, most of the participants reported M_{Des} as the most comfortable (50%) and easiest to type mapping (44%), followed by M_{Acc} (31% and 25%, respectively). Although M_{Des} allowed for faster typing (Fig. 9, left), M_{Acc} was perceived as faster by participants. As expected, participants also reported M_{Abc} , as the easiest to remember (44%), followed by M_{Des} (31%).

6 Discussions

Our results seem to indicate the designer-led semi-automatic mapping M_{Des} provided better results in terms of time, comfort and users’ preference when compared to the remaining mappings. It consistently appeared as the best or second-best option, only performing worse in terms of accuracy, where very small differences (effect size) were present among mappings. This suggests that users preferred the mappings created by the combinations of experts’ knowledge (proposed weights for M_{Des}) and the computational optimization. This might reflect the difficulty to model all aspects related to interaction using only low-level factors, and how these might be misleading when the complexity of the task increases. Even for our naïve cost functions, M_{Ti} did not actually lead to faster typing speeds; and they also failed to predict the performance of M_{Des} (expected to be poor, as shown in Table 2), even for the specific factors (i.e. time) they measured. The results also highlight the value of designers’ higher-level insight, even if it cannot be directly justified from data. For instance, the categories identified (*Orthogonal, Repeat, Opposite*) guided constraint **C3**, but they could not be identified from clustering techniques. During the workshop, we pointed out that the *high-level factors* **F4**, **F5** and **F6** were already covered by *low-level* ones, but designers still decided to keep them. We understand these reflect *tacit knowledge* which, even if hard to verbalize/rationalize, was still relevant to the task. The results obtained by the designers’ mapping should highlight the relevance of such designers’ insight (i.e. high-level factors identified), but it also illustrates the value of our hybrid approach, exploiting computational methods to keep this human knowledge in the optimization loop.

The resources required for both the designers’ workshops and the brute-force exploration of alternative mappings must also be considered. The full search to create our

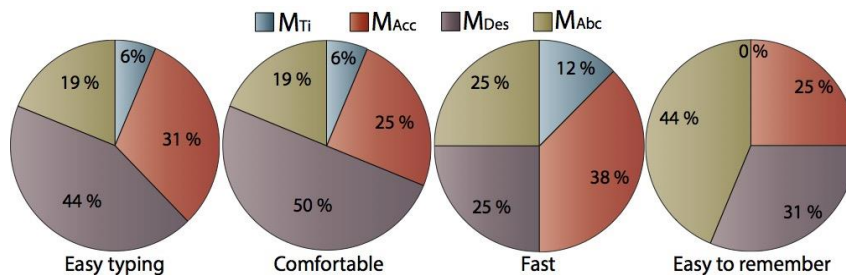


Fig. 10. Participants' preference per mapping (M_{Ti} , M_{Acc} , M_{Des} and M_{Abc}) regarding their task experience (ease typing, comfort, speed and ease to remember).

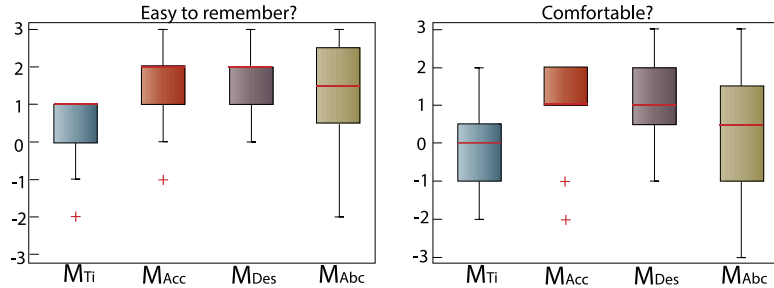


Fig. 11. Box plots for memorability (left) and comfort (right) per mapping. Horizontal red bars and boxes represent medians and IQRs. Whiskers stretch to points within median ± 1.5 IQR. Outliers shown as single red crosses.

alternative mappings (2.7 billion combinations explored, for the 6 alternatives) required 5 standard desktop machines running over 5 days (development costs for software not considered). In comparison, the designers' feedback was gathered during a single workshop of 4 hours and still managed to identify relevant *high-level factors*, *constraints*, and provided good results for the final mapping. This seems to indicate designers' involvement can be easily justified, producing relevant input to underlying computational approaches and potentially reducing development costs.

Finally, our use-case must be considered as an illustrative example of our approach, rather than an exemplar text-entry system. Text entry systems can leverage extensive task-specific knowledge (e.g., digraph transitions, predictive models, etc.), which can allow defining effective mappings even from low-level factors. Instead, our case study provides an example that is generalizable to a broader spectrum of applications using gestural interaction; illustrates the challenges related to creating complex interactive dialogues from low-level factors; and highlights the benefits related to designers' insight into the process.

7 Conclusion

We presented an approach for semi-automatic generation of gesture mappings for devices with low gestural resolution. Our approach consists of quantifying observable *low-level factors* such as individual gesture error rates, speed and accuracy; identifying how designers weigh different factors to create a weighted cost function, which is then fed into a computational approach to find the optimum gesture set and its mapping to tasks. Comparing the results of our mapping with the mappings obtained from other naïvely constructed cost functions shows that overall users perform consistently well with our mapping in terms of speed, comfort and memorability. These results highlight the value of our approach, as a tool to guide the designer-led computational approach to generate complex mappings. This approach should not stand as a replacement for traditional HCI methods, but as a tool to help such iterative processes to converge faster towards satisfying solutions, particularly within complex application domains featuring large solution spaces and complex/subjective factors influencing interaction.

References

1. Bi, X., Smith, B.A., and Zhai, S. Quasi-qwerty soft keyboard optimization. in Proceedings of the SIGCHI Conference. ACM, 2010.
2. Bi, X. and Zhai, S. IJQwerty: What Difference Does One Key Change Make? Gesture Typing Keyboard Optimization Bounded by One Key Position Change from Qwerty. in Proceedings of the 2016 CHI Conference. ACM, 2016.
3. Borg, G. Psychophysical scaling with applications in physical work and the perception of exertion. *Scandinavian journal of work, environment & health*: p. 55-58. 1990.
4. Cadoz, C. *Les réalités virtuelles*. 1994.
5. Davey, B. and Parker, K.R. Requirements elicitation problems: a literature analysis. *Issues in Informing Science and Information Technology*. 12: p. 71-82. 2015.
6. Dutta, T. Evaluation of the Kinect™ sensor for 3-D kinematic measurement in the workplace. *Applied ergonomics*. 43(4): p. 645-649. 2012.
7. Fabiani, L., Burdea, G.C., Langrana, N.A., and Gomez, D. Human interface using the Rutgers Master II force feedback interface. in *VRAIS*. 1996.
8. Feit, A.M. and Oulasvirta, A. Pianotext: Redesigning the piano keyboard for text entry. in Proceedings of the 2014 conference DIS. ACM, 2014.
9. Gabbard, J.L., Hix, D., and Swan, J.E. User-centered design and evaluation of virtual environments. *IEEE computer Graphics and Applications*. 19(6): p. 51-59. 1999.
10. Gelain, M., Pini, M.S., Rossi, F., Venable, K.B., and Walsh, T. Elicitation strategies for soft constraint problems with missing preferences: Properties, algorithms and experimental studies. *Artificial Intelligence*. 174(3): p. 270-294. 2010.
11. Gong, J., Haggerty, B., and Tarasewich, P. An enhanced multitap text entry method with predictive next-letter highlighting. in *CHI'05*. ACM.
12. Grossman, T., Wigdor, D., and Balakrishnan, R. Multi-finger gestural interaction with 3d volumetric displays. in Proceedings of the 17th annual ACM UIST. ACM, 2004.
13. Haldar, R. and Mukhopadhyay, D. Levenshtein distance technique in dictionary lookup methods: An improved approach. *arXiv e-print (arXiv:1101.1232)*. 2011.
14. Kammer, D., Wojdziak, J., Keck, M., Groh, R., and Taranko, S. Towards a formalization of multi-touch gestures, in *ACM ISS*, ACM. p. 49-58. 2010.
15. Kerber, F., Lessel, P., Kr, A., #252, and ger. Same-side Hand Interactions with Arm-placed Devices Using EMG, in Proceedings of the ACM CHI, ACM: Seoul, Republic of Korea. p. 1367-1372. 2015.
16. Kessler, G.D., Hodges, L.F., and Walker, N. Evaluation of the CyberGlove as a whole-hand input device. *ACM TOCHI'95*. 2(4): p. 263-283.
17. 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 ACM UIST. ACM, 2012.
18. Kirkpatrick, S., Gelatt, C.D., and Vecchi, M.P. Optimization by simulated annealing. *science*. 220(4598): p. 671-680. 1983.
19. Klimt, B. and Yang, Y. The enron corpus: A new dataset for email classification research. in *European Conference on Machine Learning*. Springer, 2004.

20. Knibbe, J., Martinez Plasencia, D., Bainbridge, C., Chan, C.-K., Wu, J., Cable, T., Munir, H., and Coyle, D. Extending interaction for smart watches: enabling bimanual around device control. in CHI'14. ACM.
21. Lyons, J. A branch of both mathematics and computer science, cryptography is the study and practice of obscuring information. . <http://practicalcryptography.com/>,
22. McIntosh, J., McNeill, C., Fraser, M., Kerber, F., Löchtefeld, M., and Krüger, A. EMPress: Practical Hand Gesture Classification with Wrist-Mounted EMG and Pressure Sensing. in Proceedings of the 2016 CHI Conference. ACM, 2016.
23. McNeill, D. Hand and mind: What gestures reveal about thought. University of Chicago Press. 1992.
24. Pavlovych, A. and Stuerzlinger, W. Model for non-expert text entry speed on 12-button phone keypads. in Proceedings of the SIGCHI conference. ACM, 2004.
25. Rekimoto, J. SmartSkin: an infrastructure for freehand manipulation on interactive surfaces. in Proceedings of the SIGCHI conference. ACM, 2002.
26. Rimé, B. The elimination of visible behaviour from social interactions: Effects on verbal, nonverbal and interpersonal variables. *European journal of social psychology*. 12(2): p. 113-129. 1982.
27. Rimé, B. and Schiaratura, L. Gesture and speech. 1991.
28. Robertson, R.J., Goss, F.L., Rutkowski, J., Lenz, B., Dixon, C., Timmer, J., Frazee, K., Dube, J., and Andreacci, J. Concurrent validation of the OMNI perceived exertion scale for resistance exercise. *Medicine and science in sports and exercise*. 35(2): p. 333-341. 2003.
29. Sridhar, S., Feit, A.M., Theobalt, C., and Oulasvirta, A. Investigating the Dexterity of Multi-Finger Input for Mid-Air Text Entry. in Proceedings of ACM CHI Conference. 2015.
30. Sturman, D.J. and Zeltzer, D. A design method for “whole-hand” human-computer interaction. *ACM Transactions on Information Systems (TOIS)*. 11(3): p. 219-238. 1993.
31. Sturman, D.J., Zeltzer, D., and Pieper, S. Hands-on interaction with virtual environments. in Proceedings of the 2nd annual ACM SIGGRAPH UIST. ACM, 1989.
32. Thalmic-Labs. <https://support.getmyo.com/hc/en-us/articles/205180865-Double-Tap-gesture-is-not-recognized>,
33. Weichert, F., Bachmann, D., Rudak, B., and Fisseler, D. Analysis of the accuracy and robustness of the leap motion controller. *Sensors*. 13(5): p. 6380-6393. 2013.
34. Weissmann, J. and Salomon, R. Gesture recognition for virtual reality applications using data gloves and neural networks. in *Neural Networks, 1999. IJCNN'99. International Joint Conference on. IEEE, 1999*.
35. Wu, M. and Balakrishnan, R. Multi-finger and whole hand gestural interaction techniques for multi-user tabletop displays. in Proceedings of ACM UIST. ACM, 2003.
36. Zhang, Y., Zhou, J., Laput, G., and Harrison, C. SkinTrack: Using the Body as an Electrical Waveguide for Continuous Finger Tracking on the Skin. in Proceedings of the 2016 CHI Conference. ACM, 2016.