# Learning to Type on Mobile Keyboards: A Theory on the Formation of Spatial Memory

ANDREAS KOMNINOS, University of Patras, Greece
IOULIA SIMOU, University of Patras, Greece
JOHN GAROFALAKIS, University of Patras, Greece

Typing on mobile soft keyboards is an acquired skill that users learn through experience. Previous research has modelled learning of keyboard layouts via statistical approximation of the probability of correct recall of a target's (key) location from long and short-term memory, however, there are no insights as to how the mental model of the keyboard may be formed inside the brain. We construct a theory that could explain this process sufficiently, highlighting the importance of mental landmarks for efficient storage and recall of spatial information. The theory is explored and validated through simulation of a cognitive architecture and comparison with extant empirical data, generating insights that could inform the design of new layouts and facilitate the learning process for novice users.

CCS Concepts: • **Human-centered computing** → **HCI theory, concepts and models**; **Text input**; **Touch screens**; **Mobile devices**.

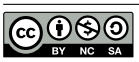
Additional Key Words and Phrases: Mobile text entry, Smartphones, Soft Keyboards, Spatial Memory

**ACM Reference Format:**
Andreas Komninos, Ioulia Simou, and John Garofalakis. 2025. Learning to Type on Mobile Keyboards: A Theory on the Formation of Spatial Memory. *Proc. ACM Hum.-Comput. Interact.* 9, 5, Article pn4279 (September 2025), 27 pages. https://doi.org/10.1145/3743720

## 1 Introduction

Spatial memory, the cognitive ability to encode, store, and recall the positions of elements in an environment, is a fundamental mechanism that supports efficient and effective interaction with user interfaces (UIs). The ability to internalize spatial arrangements through memory development can significantly reduce cognitive load and improve task performance, leading to the operational skill that is crucial for digital literacy [29]. A core component of digital literacy is keyboarding skill, which requires the construction and internalisation of the keyboard's spatial layout[15]. Although keyboarding skills with physical (desktop) keyboards remain important, the primacy of mobile devices over desktop computers means that learning to use a smartphone or tablet touchscreen (virtual) keyboard is usually the first keyboarding challenge that the wider population encounters (e.g. children), and for many, it remains either the primary or only challenge, as a touchscreen device is the only, or most available computing device in their daily life [9, 50]. Beyond novice users, scenarios involving foreign language interfaces or systems with adaptive layouts present additional challenges where spatial learning is vital. For example, learning to type on a foreign language keyboard often requires users to memorize new spatial arrangements of characters while

Authors' Contact Information: Andreas Komninos, Computer Engineering and Informatics Department, University of Patras, Rio, Greece, akomninos@ceid.upatras.gr; Ioulia Simou, Computer Engineering and Informatics Department, University of Patras, Rio, Greece, simo@ceid.upatras.gr; John Garofalakis, Computer Engineering and Informatics Department, University of Patras, Rio, Greece, garofala@ceid.upatras.gr.



This work is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License.
© 2025 Copyright held by the owner/author(s).
ACM 2573-0142/2025/9-ARTpn4279
https://doi.org/10.1145/3743720

reinforcing new motor and cognitive behaviors. Spatial memory enables users to rely less on slow, effortful visual searches and instead utilize automatic recall of locations, fostering smoother transitions to expert behavior over time [21, 55].

Novice mobile keyboard users typically depend on visual search strategies during their initial interactions, but with repetition, they build recall-based memory that allows faster and more fluid interactions [33, 34, 63]. For example, the transition from visually guided typing to touch typing on keyboards relies heavily on spatial memory, as users progressively internalize the positions of individual keys through stable layouts like QWERTY [34, 70]. This progression is not limited to typical text-entry tasks but extends to adaptive interfaces, such as sparse or invisible keyboards, where effective learning becomes critical in the absence of consistent visual cues [61, 71]. Spatially consistent designs and interfaces enhanced with landmarks, both natural (e.g., edges, corners) and artificial (e.g., visual anchors, unique icons), have demonstrably improved the learning curve for such users, especially in complex or high-density layouts [54, 62, 63]. Keyboards, on the other hand, lack such landmarks, as the majority of keys are almost identical in appearance, while visually distinct keys (e.g. shift, delete, space, language or layout modifiers) are grouped towards the bottom of the keyboard, and thus offer limited assistance for orientation during visual searches. Computational models have shown how users adapt to such unfamiliar keyboard layouts by combining visual attention with gradual memory development [33, 34, 59]. State-of-the-art models use probabilistic approaches to infer the time required for a spatial location be recalled from memory, but without the ability to account for recall error, or being able to provide a plausible explanation of how spatial memory is formed, the type of information it contains, and how it evolves over practice.

In this paper, we propose a theory on the formation of spatial memory, which may shed light on the process of how spatial memory is formed while learning new keyboard layouts. We compare a computational implementation of this model with previous models and findings from empirical studies, discovering evidence for the theory's validity. Further from providing an extension to state-of-the-art models that may be used to computationally evaluate novel input method designs, this understanding may help in the development of new designs of keyboard features to assist learning.

## 2 Related Work

### 2.1 Spatial Memory

Human long term memory encodes information about the world, with some researchers arguing that the capacity to store is massive, if not unlimited [6], unlike short-term memory which is known to have capacity limits, often related to the quantity and complexity of information stored there [41]. Information recall from long-term memory usually involves using one of the encoded features as a retrieval cue, and successful recall depends on both the availability of such encoded features, as well as how specific a feature is to the particular encoded information [68]. The term "spatial memory" is used to describe the part of the memory system responsible for the encoding and recall of spatial information about a person's environments and their orientation within that environment [30]. This information is not only vital to our survival in the real world, but plays a central role in our ability to learn graphical computer interfaces, by providing the ability to recall and utilize the organization of elements in an environment. The construction of gradually improved mental spatial models of a user interface, helps users transition from novice to expert behaviors by reducing cognitive load and visual search times, leading to interaction automaticity [57].

Spatial memory has been investigated in the context of 2D maps (cartography), with empirical evidence pointing towards the positive effect of increasing map complexity to the ability to recall object locations, up to some asymptote on which further complexity ceases to have positive effects.

As in the real world, map annotations (e.g. pictograms or other salient visual features) and map boundaries, serve as prominent landmarks, on which a mental model of the location of objects, in relation to these landmarks, can be constructed. Landmarks serve as the retrieval cues which can help speed up visual search, by leveraging mental spatial associations of search targets with known landmarks (e.g. first find the stadium, and the café is somewhere west of the stadium) [4, 35]. Spatial memory is then more a case of forming mental "neighbourhoods", based on salient features of the information space, and associations of objects with these, rather than remembering the precise location of every single object. Naturally, information on these associations can be improved over time to become more precise, so, progressing between *"near the stadium"* → *"west of the stadium"* → *"two blocks west of the stadium"* → *"two blocks west of the stadium on Findlay Street"*. In user interfaces, landmarks, both natural (e.g., screen boundaries, window boundaries) and artificial (e.g., icons, grid anchors), have been shown to enhance spatial memory by serving as salient reference points. Empirical studies demonstrate that artificial landmarks improve spatial recall efficiency in large grid-based layouts and linear control widgets, such as sliders and scrollbars [45, 63, 64]. Similarly, hand-based landmarks are leveraged in innovative designs such as HandMark menus, enabling rapid command selection on touch interfaces [62, 65]. Landmark-based designs mitigate cognitive load and enable users to develop spatial schemas more quickly, particularly in complex or large-scale interfaces [54, 64]. Interestingly, spatial information encoding and recall can be helped not just through distinct visual landmarks, but also by displaying visual cues that organise the information space in segments. In 2D cartography, the display of map grid lines has been shown to improve object recall, even when only parts of the grid lines are shown. Further, larger cell sizes improve performance in spatial memory. An exploration of different types of grid visualisation (hexagons) showed that they are more helpful for spatial memory compared to rectangular grids, only in cases where the underlying cartographic complexity is low (e.g. rural, vs. urban area maps) [14, 19, 20].

### 2.2 Learning of keyboard layouts

Typing is a complex cognitive task associated with vision, memory and motor control functions of the brain. Past studies with fMRI methods, which measure blood flow to various parts of the brain, demonstrate activation in several areas of the brain associated with these functions, while typing on full-sized (desktop) keyboards (e.g. [25, 27, 52]). Particularly with regard to the role of memory in typing, while such studies are able to provide some insight about *where* in the brain the user's spatial knowledge of the keyboard layout is stored, they cannot uncover *how* this information is stored, i.e. the structure of the actual mental model of a keyboard, as developed by users. To acquire the user's mental model, an empirical experimentation approach with cognitive tasks such as verbal recall, sketching or location judgement tasks might be more appropriate [1, 56]. An alternative approach is to use cognitive modelling in order to simulate cognitive functions of the brain.

A variety of computational models have been proposed to simulate and predict the mechanisms underlying spatial memory development. ACT-R cognitive models have been widely used to explain how users transition from visually guided search to recall-based interaction, capturing the effects of memory activation, interference, and decay [13, 21]. Reinforcement learning models extend this framework by simulating reward-driven optimization of spatial recall in dynamic contexts, such as touchscreen typing, where interactions balance visual-motor coordination and memory retention [33, 59]. Spatial memory is especially crucial in text-entry systems such as soft keyboards, as users rely on stable layouts like QWERTY to internalize key positions for touch typing [34, 70]. Probabilistic approaches similarly model spatial recall errors and adaptation mechanisms under conditions like invisible or sparse keyboards and show that spatial models already formed in the brain can be used even in contexts where the actual interface is not visible [61, 71].

Previous literature on text entry employs evaluations of novel keyboards over several sessions conducted on the same day (e.g. [16, 40]), or through longitudinal studies over multiple days, in an effort to improve external validity (e.g. [10, 11, 26]). Usually, text entry evaluations employ transcription tasks, with phrases presented to participants that are randomly selected from standardised phrase sets (e.g. [42, 66]). To reduce the number of sessions required to show performance differentials between novel and baseline input methods, other researchers have designed evaluation tasks where participants need to input the same phrase or word repeatedly [5, 24, 47, 48]. However, studies devoted to the learning progress of entirely novice mobile keyboards are quite few, as most input method evaluations focus on variations or support mechanisms for the QWERTY layout, in order to leverage the spatial knowledge and expertise participants already have with classic implementations of this keyboard type. Notably, we mention the early studies in [43, 69], as well as more recently in [17, 31, 47, 48], as examples of investigating learning in various novel layouts, or optimisations (re-configurations of keys) of the classic QWERTY layout. In all studies, an exponential increase in participant performance across sessions is observed. In fact, some studies fitted exponential functions to predict participant performance past the confines of the actual study [43, 47]. This finding is congruent with other research in learning user interfaces (e.g. [39]), and suggests more practice should yield better performance but with diminishing returns. It is, however, known from other domains of human skill acquisition, that learning curves are not all the same shape [2], and that it is possible for them not to be curves at all (e.g. exhibiting phase transitions [67]). Empirical data from large scale data collections of typing performance data, seem to suggest that in contrast with exponential models, which may have unlimited upper bounds, human typing skill with smartphones is, in fact, bounded [51]. Early simulation work in [69], also hinted that performance plateaus are likely in mobile typing, though the authors did not attempt to fit regression models to the simulated learning data.

Beyond this empirical evidence, less is known about how this pattern of human performance improvement is achieved. Input speed, translated into time to hit a keyboard target, is a function of motor skill (finger movement speed and accuracy) as well as the ability to correctly and speedily recall target locations from long-term memory. Previous computational modelling was mostly concerned with motor performance, using Fitt's law [43, 53, 70]. Other early computational models substituted the recall process for Visual Search Time, an aggregate metric that includes both time to recall, and the need for effortful visual search, since recall may be inexact. Early models assumed this time would be zero for expert users, while using the Hick-Hyman law to model it for novice users [60]. This was later refuted by empirical data in [28, 58]. Later work in [34], presented a model focusing exclusively on cognitive aspects of learning to type on soft keyboards, modelling long-term and short-term memory, as well as visual attention with ocular movements as key factors of the typing process, while leaving motoric aspects out of the model. Though this model is the most complete effort in understanding the cognitive process of learning, it is not concerned with how spatial memory is encoded and refined over practice. It is assumed that the first time a target is encountered, its precise location is stored in memory, and from there on, the model is only concerned with modelling the time required to forget this information, as well as the time required to retrieve it from LTM.

### 2.3 Summary

In summary, the process of learning how to type with mobile keyboards remains still underexplored. We know very little about how spatial information is encoded in long-term spatial memory, why retrieval inaccuracies that cause effortful visual searches occur, and how these inaccuracies are resolved over time, leading to expert and effortless performance. To address this gap in knowledge, we propose a theory for the encoding of spatial information related to typing with fixed layouts in

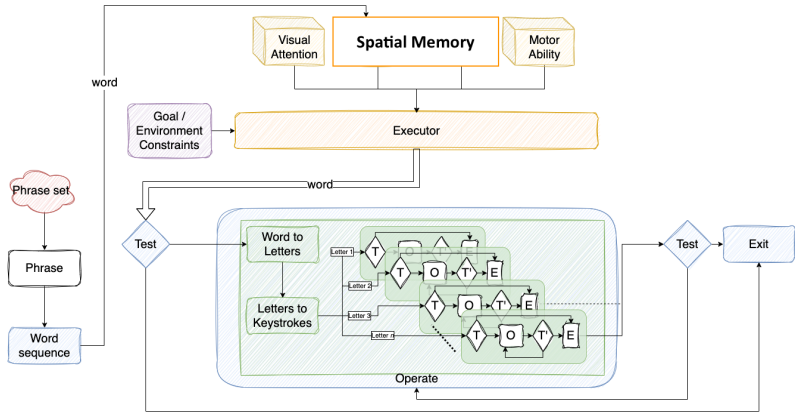



Fig. 1. The cognitive architecture presented in [37], adapted to reflect improvements effected for this paper. We extend the architecture by implementing a Spatial Memory module, which was missing from the original implementation.

mobile keyboards, which is discussed in the next sections. The theory is based on the concept of imaginary zones on the keyboard, towards which a user may direct their gaze in order to search for a target key. For example, in a $2 \times 1$ imaginary zone arrangement, the user will probabilistically search for the target either at the left or right half of the keyboard. Finer grained arrangements are also possible, for example in a $2 \times 2$ arrangement, the user may search at the top-left, top-right, bottom-left or bottom-right area of the keyboard, and so forth.

## 3 Modelling Spatial Memory for Simulated Text Entry

This paper builds on our previous open-sourced simulation approach for mobile typing, presented initially in [36]. More recently, we presented an updated version of this simulator in [37], which significantly revised our original work, by adopting a cognitive architecture that follows Miller's Test, Operate, Test, Exit model, as well as being inspired by Norman's Action Cycle Theory [44, 49]. A limitation of this latest version of our simulator is that it did not focus on aspects of short or long term memory, and simply assumed perfect spatial knowledge of the layout. For this paper, we extended the existing code in order to simulate long-term spatial memory, the development of memory, and recall of information from there. As the main topic of this paper is to examine how spatial memory is formed, we do not delve into the separation of long and short term memory, nor do we consider aspects of memory loss (decay) due to lack of recency and frequency of use of the information it contains. Therefore, we consider long and short term memory as a unified faculty of the brain, and, more specifically, we consider only spatial memory aspects (the brain's spatial model of the keyboard) for the input process. The overall architecture of the updated cognitive model is shown in Fig. 1). Next, we will explain the general mechanics of our implemented extensions to the simulator. Details can be seen in the code, which is openly available, see Section 9 for availability.

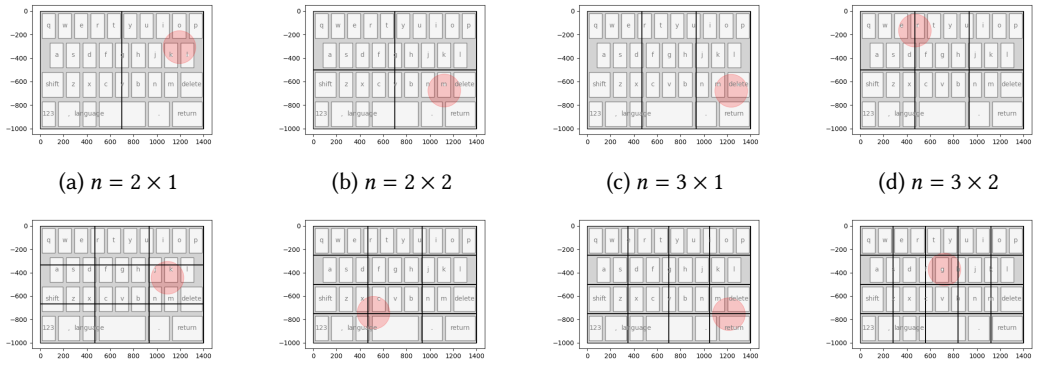
In order to execute a keypress task, the modelled brain's central executor module takes the following actions. To type a sentence, the sentence is first broken down into tokens (words), and this sequence forms the main execution plan. For each token, a subplan is formed for the sequence of characters to be input in the form of an array, for example the plan for the token "*test*" becomes ['t','e','s','t']. Then for each step in the plan (each character), the executor requests the location of the target key from the Spatial Memory module. Target locations recall does not concern precise key coordinate pairs, but rather the "general areas" of the keyboard where the keyboard may be

found, i.e. a region. This region is chosen probabilistically, and learning concerns the update of the association probabilities between keys and regions, as well as the bounding boxes of the regions associated with each target, as will be explained in the next sections. Upon retrieval, the executor module instructs the user's visual attention (eyes) to be directed towards the recalled region of the target. Therefore recall concerns the bounds of an area in which the target is likely to be found, and a saccade places the gaze at some random coordinates within that area. Simulating foveal vision, several targets can be "read" from these landing coordinates at once. If the target is not found from this process, the executor attempts more recalls, until the target is found. Then, the executor instructs the finger to move to that target in order to hit it. This action can be successful, in which case the executor moves to the next step in the plan. If the action fails, the plan is revised by inserting corrective actions at the current plan stage. The simulator considers two possibilities of failure. The first one is when the finger hits an inactive area of the keyboard. In this case, the plan step counter is not incremented, and the eye is not moved, therefore the user attempts a finger motion again. The second case involves hitting a proximal target that was not the intended one. In this case, the plan needs to be modified by inserting a deletion attempt, e.g. if the plan is ['t','e','s','t'], at step 2, instead of 's' the user accidentally types 'a'. The plan is then modified to ['t','e','←','s','t']. Of course, attempting to hit the backspace (←) character may also result in error, in which case more corrective actions are inserted into the plan. The TOTE architecture ensures that as many actions as are needed in order to continue with the original plan, are taken during simulated input.

In the following sections, we present the outcomes of simulations conducted with this cognitive model and our additions to it, demonstrating how we constructed our theoretical framework and, ultimately, compared the final model against results from previous, independent emprical studies. The simulator software in [36, 37] includes a range of virtual participants whose kinematic dexterity is pre-modelled by the authors, after real data from human participants in [32]. We use these pre-configured virtual participants in our own experimentation. Note that the parameter values in the simulations, as mentioned in the text, were empirically selected for illustration purposes after experimentation with a range of values. Throughout this paper, we simulate use of a keyboard with a resolution of $1400 \times 1000$ pixels, which is representative of mainstream smartphones [32]. Foveal vision of 2 degrees is simulated by returning all keys whose bounding boxes are within one keywidth (140px) from the current gaze coordinates. This setting equates to a viewing distance of approximately 23cm for a device with a 450ppi pixel density screen, and is typical of seated posture with two elbows rested on a table (see Fig. 2). It is important to mention that the simulator only supports single-digit (index finger) typing, therefore the simulated posture is holding the device with a non-dominant hand, and typing with the dominant hand index finger. Readers are welcome to try out alternative parameter values, by modifying and running the provided code (see Section 9 for availability).

## 4 Study 1 - How Imaginary Zones affect Learning

In contrast with other user interfaces or 2D information visualisations like maps, virtual keyboards lack distinguishing features that can function as visual anchors. Most keys have very similar appearance (e.g. character keys). The space bar and special function keys, which differ in size and thus might play the role of the landmark, are located in the bottom part of the keyboard, therefore are not adequately spaced around the keyboard in order to promote learning. Based on the lack of distinctive visual landmarks, as well as evidence from 2D cartography that grid layouts may improve learning and recall from spatial memory, we theorise that users may begin the learning process by making a simple mental model of the spatial layout, by dividing the keyboard area into imaginary key groupings ("zones"), akin to the neighbourhoods depicted in a map. This zone layout is stored in long-term memory and various targets (keys) can then be associated with one or more



(a) $n = 2 \times 1$ (b) $n = 2 \times 2$ (c) $n = 3 \times 1$ (d) $n = 3 \times 2$

(e) $n = 3 \times 3$ (f) $n = 3 \times 4$ (g) $n = 4 \times 4$ (h) $n = 5 \times 4$

Fig. 2. Illustrative divisions of keyboard area into imaginary zones. Labels denote columns x rows. The red cirlce indicates coverage of foveal vision.

zones. Initially, because the user has not learned these associations, every key is equally associated with every imaginary zone - when the task "*retrieve the zone for target t*" is presented by the user, neural synapses leading to the result that represents each zone fire equally strongly and at the same rate, therefore the probability that the correct zone will be retrieved from long-term memory, is the same for all zones. After a target's zone has been retrieved from long-term memory, the user then scans inside that zone to find the desired target. If the target is not found, the user selects another zone, and the process is repeated until the target is found.

As such, for each target $t$ in a keyboard layout, the user builds a pairwise association with every zone $z$, representing the probability $P(z, t)$ that the key can be found in that zone. The user holds the belief that if they have mentally divided the keyboard in $n$ zones (e.g. as per Fig. 2), then the probability of finding target $t$ in one of these zones $\sum_{z=1}^{n} P(z, t) = 1$ (which is reasonable when intending to enter an alphabetic character using the main screen of a virtual QWERTY keyboard). This process is inspired by the building of pairwise associations of spatial objects in hippocampal place cells [22], or, in our case, a target key $t$ and the each of the imaginary zones. We model Spatial Memory as a dictionary data structure in Python, where the key "regions" contains the bounding boxes of the imaginary zone arrangement in the user's mental model of the keyboard. In code listing 1 (lines 2-4), we illustrate a mental model with two zones (e.g. left and right, or top and bottom of the keyboard).

Every time the user locates a desired target, they gradually learn the zone where that target is located, by increasing the probability $P(z_k, t)$ that zone $k$ will be the first to be visited during a search, by some quantum $q$, and, simultaneously reducing the probability for each of the other zones by an amount equal to $q/(n-1)$ (see code listing 1, lines 9-10). This is analogous to the correct set of place cell synapses activating with a stronger signal and faster firing rate, while synapses leading to undesirable results will gradually fade and will be eventually eliminated, as so happens in neural synaptic plasticity [12, 46]. This quantum $q$ represents the *learning progress* on the task of locating a target, therefore its magnitude is determined by the velocity $U$ at which a user learns (i.e. how quickly new neurotransmitters and neuroreceptors are added/removed to the synapse) and thus improves their chances of selecting the correct zone in which to initiate the search.

```
{
  "regions":{
    "q1":[<left>,<bottom>,<right>,<top>] //bounding box coordinates
    "q2":[<left>,<bottom>,<right>,<top>] //bounding box coordinates
  }
  "probabilities":{
    "a":{
      "regions":{
        "q1":0.8,
        "q2":0.2
      },
      "visitations":0, //count of gaze visitations for statistics
      "home_zone":{    //the target's home zone and its boundaries
        "name":"q1",
        "bbox":[<left>,<bottom>,<right>,<top>] //static or dynamically adjusted
      }
    },
    "b":{..},
    "c":{..},
    ...
    "delete":{..},
    "space":{..},
    ...
  }
}
```

Listing 1. Pseudo-code representation of the data structure used to represent Spatial Memory in our extension of the model in [37]

The quantum increase can be linear, by some absolute amount (e.g. every time a target is located, increase the probability to find it again by a constant value $\lambda$), therefore the probability to begin the visual search in the correct zone (hence forth the "home zone") $z_h$ for target $t$ after the $i^{th}$ selection of $t$ is

$$P(z_h, t)_i = P(z_h, t)_0 + \lambda \times i \tag{1}$$

where $P(z_h, t)_0 = 1/n$ is the initial probability. The main issue with this approach is that it implies that human ability in learning a new task is constant, while it is known that human learning is not a linear process - learning begins somewhat slowly and then rapidly increases. This phenomenon (i.e. the increase of $P(z_h, t)_i$ at a variable velocity) can be modeled non-linearly by setting $P(z_h, t)_i$ as a function of exposures to the task at a constant rate of change (e.g. increase $P(z_h, t)_i$ by $q = 3\%$). An equation such as

$$P(z_h, t)_i = P(z_h, t)_0 \times \lambda^i \tag{2}$$

can provide an exponential increase of the probability. However, this approach does not correspond to the biology of neuroplasticity, which shows that synaptic strength and firing rates in neurons

increase as learning progresses, up to a fundamental limit of receptor and transmitter densities on neural synapses, imposed by the biological structure and chemistry of the brain. Another way to model this process is by applying a decay function to the learning velocity, therefore it begins at some value of $P(z_h, t)_0$ and then gradually slows down until some threshold of $P(z_h, t)_i$ is reached. It is reasonable to assume that this decay would be inversely related to the number of imaginary zones - more zones would mean a slower decay because, simply, learning is a more complex process when more zones are involved. Learning stops later ($U = 0$) for a person learning to associate a target with a zone, when the number of possible zones is large. For example, learning the assignment of any of 26 possible keys to one of $N$ zones should take longer when $N = 10$, compared to $N = 2$, because there are more ways to fail while learning this task. A decay process could be modeled by an equation such as

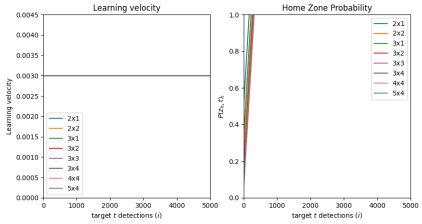
$$P(z_h, t)_i = P(z_h, t)_0 \times (1 + e^{(-\frac{i}{k \times n})}) \tag{3}$$

where $k$ is a scaling constant. The issue with this approach is that it would imply that, depending on $k$ and the number of imaginary zones in the users' mental model, users may never achieve perfect recall of a target's location, as the resulting cumulative probability is asymptotic to some upper bound. Perhaps a better way thus to model such a process of change in the rate of learning would be to calculate $P(z_h, t)_i$ through a logistic function, which can better approximate velocity "growth" with an upper asymptote. This way, a user learns rapidly initially, until they reach the upper limit of the learning velocity, and by that velocity achieve perfect recall. Therefore, we can use an equation such as

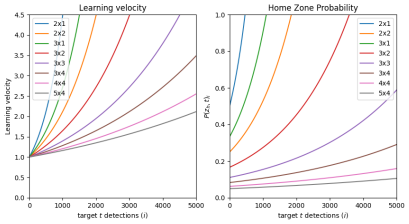
$$P(z_h, t)_i = P(z_h, t)_0 \times \frac{\lambda}{(1 + e^{(-k*(i-i_0))})} \tag{4}$$

where $\lambda$ is the function's carrying capacity (upper bound), $k$ is the logistic growth rate, and $i_0$ is the functions inflection point. Again, the function should be contingent on the number of imaginary zones, with the number of exposures required to reach the upper threshold related to the the number of zones. To illustrate the differences, Fig. 3 shows an example of the behaviour of the different functions used to calculate learning velocity and the resulting cumulative probability, for a number of imaginary zones.

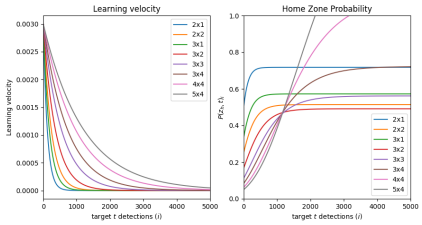
At this point it is worth pausing to reflect on some of the findings illustrated in Fig. 3. First, zones in this example are deterministic with hard boundaries, which do not necessarily fit the bounds of the visual targets (keys). We will return to this problem in the next sections but for now, since our simulation only considers key center coordinates as the finger's target, the groupings can be deemed appropriate at least for the purpose of exploration. As expected, a constant velocity yields rather unrealistic results, as probability of having learned a target's home zone increases linearly. Constant and logistic rate of change require the enforcement of hard upper bounds of 1.0 to the probability. The main difference between the two is that the learning velocity is unbound with constant rate of change (until the point where $P(z_h, t)_i = 1$), therefore the user is assumed to be able to learn ever more quickly, without limits to how fast they learn. With logistic rate of change, learning velocity increases until an upper bound, demonstrating an increasing ability to learn until that limit. This is closer to existing understanding of learning functions in the brain. Finally, with decaying rate of change, the user is assumed to learn fast initially, while this ability is reduced to zero (no learning gains) after a while. As a consequence, the decaying rate of change shows that for some zone sizes, learning cannot be perfected, which contravenes common sense: it should be, for example, perfectly possible for a user to learn whether a key is to the left or to the right of the keyboard. As shown in Fig 3c, with $k = 60$, the model makes it impossible for a user to perfectly learn whether a key is to the left or right of a keyboard (a $2 \times 1$ arrangement), which of course is unrealistic. Finally, since we discarded the decaying rate, and the linear increase approach



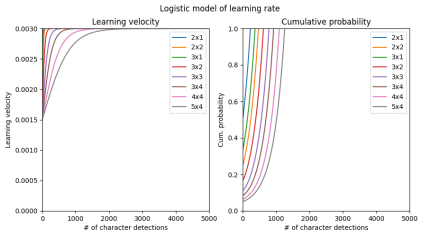

(a) Constant velocity




(b) Constant velocity rate of change




(c) Decaying velocity rate of change, $k = 60$



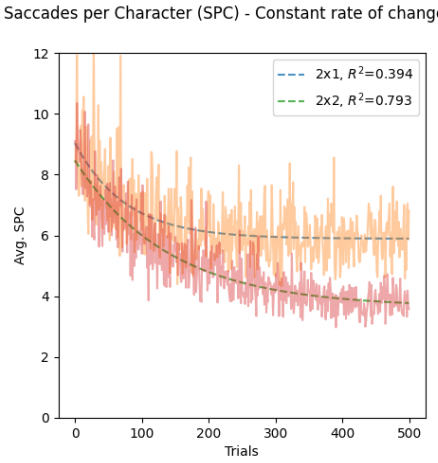

(d) Logistic velocity rate of change, $k = 1/n^2$, $i_0 = 10 \times \lambda$

Fig. 3. Illustrative progression of learning velocities and the probability of initiating a target search in the target's home zone, for 5000 exposures to the search task. In all illustrations, $\lambda = 0.003$

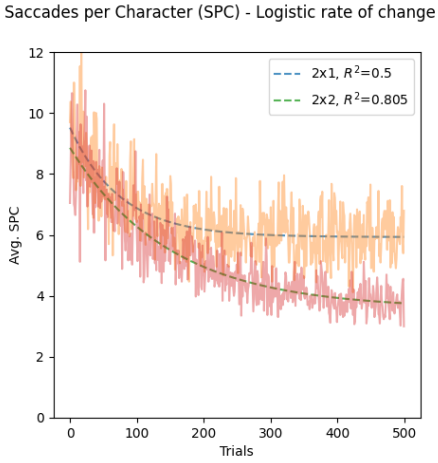
is incompatible with the non-linear nature of human learning, the most plausible models are the *constant* rate and *logistic* rate of change.

To investigate what effect these models have on actual learning, we next simulate a process of repeated typing of an English pangram ("*the quick brown fox jumps over the lazy dog*"), using a randomly selected virtual participant from [36]. From this simulation, we can explore how various models of probability assignment lead to learning by reducing the number of saccades required to locate a character, on average. Initially, we show that both learning models result in similar learning behaviour, as exemplified in Fig. 4. The user's average number of gazes to find a specific character gradually reduces until a performance bound is reached (lower is better). The process can be modelled reasonably well with a decay function of the general form $f(x) = \alpha \times e^{(-\beta+x)} + \gamma$, as demonstrated by the $R^2$ scores after curve fitting. From this toy example, we observe that a mental division of the keyboard in more zones ($2 \times 2$) shows that users can learn how to locate keys with better precision, leading to fewer visual searches. It is plausible then to assume that learning might begin with fewer imaginary zones initially, but in order to become perfect, the user must then proceed to increase the number of imaginary zones and re-configure the associations between keys and zones.

Based on this insight, we plot the learning curves for six mental zone layouts defined as low-resolution ($2 \times 1$ & $2 \times 2$), mid-resolution ($3 \times 3$ & $3 \times 4$) and high resolution ($4 \times 4$ & $5 \times 4$) as per Fig. 2. The results presented in Fig. 5 demonstrate that this assumption holds under simulation. In Table 1, we show the computed theoretical limits $\lim_{x \to \infty} f(x)$ using the fitted function parameters for each zone layout as the number of trials approaches infinity. As can be seen, using the Constant rate of change model, the optimal strategy for the user is to begin with a $2 \times 2$ layout, as this yields the best progress until approximately 150 trials, after which the user would only continue to learn if they switched to a $3 \times 3$ layout. Further learning can be obtained by moving to a $3 \times 4$ layout after about 500 trials. In the Logistic model, the optimal learning strategy is again to begin




(a) Constant rate of change, $\lambda = 0.003$




(b) Logistic rate of change, $\lambda = 0.003, k = 1/n^2, i_0 = 10 \times \lambda$

Fig. 4. Illustrative progression of learning a keyboard layout, by reducing the average number of saccades required to find a target while typing an English pangram, after 500 trials.

with a $2 \times 2$ layout, as this yields the best progress until approximately 300 trials, after which the user would only continue to learn if they switched to a $3 \times 3$ layout. Notably, finer-grain layouts do not seem to offer any significant learning advantages even at the theoretical limit (see Table 1). This is commensurate with the fact that due to foveal vision, users are able to identify (read) multiple targets with one fixation, therefore remembering the precise location of each target is not a requirement for optimal recall. Note also that for our simulation, the optimal number of saccades to locate a target is one. This can occur even if the gaze coordinates are already on the target, e.g. when typing the same character twice, because a zero-length saccade needs to be made to continue fixating on the target after the finger has moved and tapped it the first time. For different targets, the optimal number of saccades is also one, due to moving gaze to the home zone, and happening to land precisely on the target. We note also that the time to complete tasks drops significantly as learning progresses, a direct result of the fewer saccades required to find the desired targets (spikes are due to occasions where the simulated finger movements result in target misses and the requirement to correct these).

From these results, we conclude that the mental division of the keyboard area in zones is conducive to learning, and that finer zones are required in order for learning to progress. These findings are consistent with [28], where it was found that visual search time of novice users with ambiguous keyboards that group multiple characters in one key, are longer when the groupings are larger. Therefore the performance plateau for low-resolution zonal division of the keyboard is to be expected, since users need to visually search within the zones to locate the precise target.

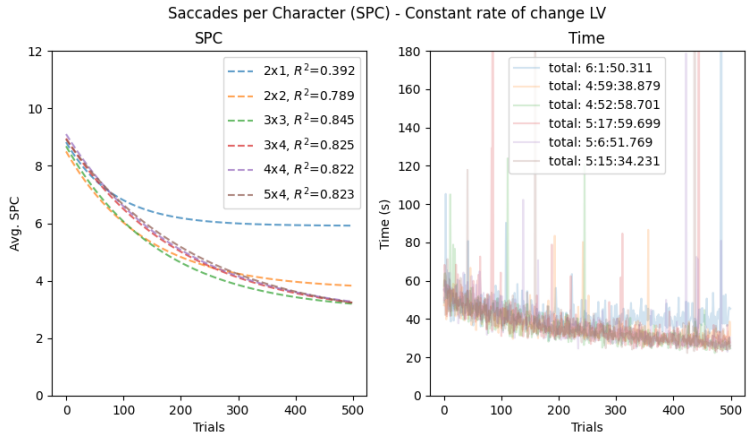
In the next section, we discuss an implementation of the model that describes dynamic zone development and adjustment for each key to promote accurate recall of its position.

## 5 Study 2 - Dynamic zone adjustment

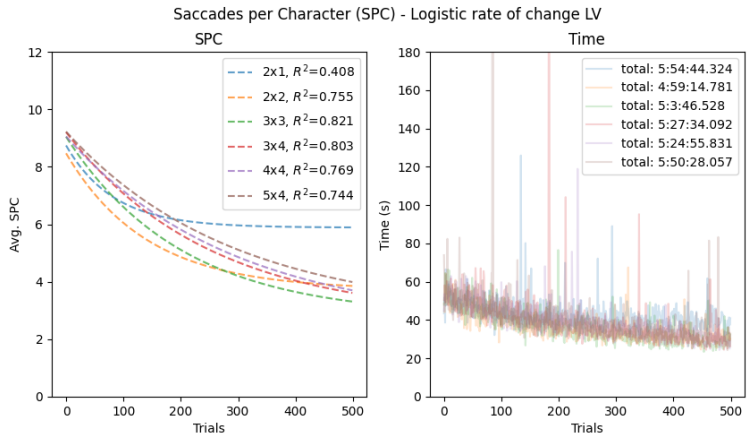
In the previous study, we demonstrated that a greater number of imaginary zones can yield better learning outcomes, at the cost of slower learning progress. A large number of zones, however, is an

| Method | Zones | | | | | |
|---|---|---|---|---|---|---|
| | $2 \times 1$ | $2 \times 2$ | $3 \times 3$ | $3 \times 4$ | $4 \times 4$ | $5 \times 5$ |
| Constant | 5.909 | 3.698 | 2.919 | 2.733 | 2.758 | **2.525** |
| Logistic | 5.833 | 3.719 | 2.779 | 2.734 | **2.518** | 2.855 |

Table 1. Fitted function limits (lowest values highlighted bold).




(a) Constant rate of change, $\lambda = 0.003$




(b) Logistic rate of change, $\lambda = 0.003, k = 1/n^2, i_0 = 10 \times \lambda$

Fig. 5. Illustrative progression of learning a keyboard layout, by reducing the average number of saccades required to find a target while typing an English pangram, after 500 trials, for six mental zone layouts.

inefficient and expensive way to store spatial information, bearing in mind the large number of pairwise associations that would need to be formed, at least initially in spatial memory (e.g. the 26 keys grouped into a $5 \times 5$ zone layout means 650 pairwise associations to be maintained initially, until some of these fade and only 26 remain). If we accept that imaginary zones serve as the recall cue, as landmarks do in 2D maps and other user interfaces, then expert performance means that the

list of recall cues must be able to fit in working memory, otherwise items from the list will need to be swapped between working and long-term memory, creating a significant performance overhead. Since working memory capacity is limited by 7±2 items (e.g. digits, groups of digits, letters, words etc.), it is therefore logical to assume that humans could work with only up to 9 imaginary zones (e.g. $3 \times 3$).

The mechanism for retrieving the location of target $t$ would then be first to recall the cue (the *home zone* for $t$), then to recall the boundaries (location) of that zone, and then to perform an (effortful) visual search within that zone. However this process provides a bound to learning that amounts to approximately three saccades per character under our previous simulation, something that does not constitute expert performance. Although it could be imagined that the brain could start with a few zones (e.g. $2 \times 1$) and then gradually develop a finer zone resolution by breaking these zones down, it is clear that such a process would be incompatible with known performance bounds. Therefore we must consider another mechanism.

An answer may come from a slightly modified interpretation of the previous results. Having more zones means that each zone (including the home zone for every key), covers a smaller surface area, therefore searches benefit not only from increases in the probability of initiating a search in the home zone, but also inherently by the increased probability that the gaze will land somewhere close to the target within the home zone, minimising the need for effortful visual searches within the home zone. If the user could recall not just the bounds of the home zone, but also *where* to look in that home zone, then we would achieve the effect of combining a coarse layout that can serve as recall cues and can fit in working memory, and the effect of having progressively smaller home zones that would guide visual searches to maximise the chances of immediately locating the target. Therefore, if a user starts with a coarse zone model, to perfect learning, it is enough for the brain to improve its knowledge of the search area boundaries within the home zone of each key, while ignoring the boundaries of other zones it might be initially associated with. This way, both probabilities will increase in parallel, leading to the perfection we associate with expert typists. Although this model of spatial memory storage requires a third level of information item storage, it is not incompatible with known memory recall processes. First, long-term memory can hold vast amounts of information, thus the additional information required to store the search zone boundaries is not an excessive burden. Second, the main bottleneck for efficient retrieval in rich episodic memories is the identification of a recall cue - after that, retrieval of the whole set of information relating to an event, episode or other construct, is very rapid [38]. This process may therefore lead to accurate recall of key positions on the keyboard, as evidenced for example in [40]. In this study, the authors asked participants to tap on a keyboard that was entirely blank, except for two horizontal bars corresponding to the location of the F and J keys (as per the tactile cues in physical keyboards). It was found that participants were able to recall the positions of keys with remarkable accuracy.

Based on this, we implement a modification to the model, by which the search zone has initially the same bounds as the home zone (see line 15 of code listing 1). With each successful key stroke, the horizontal and vertical adjustment of the search zone boundaries are adjusted proportionally to the probability of initiating searches in the home zone $q$. The search zone bounds are also adjusted proportionally to the target's central coordinates relative position to the initial zone bounds, until the adjusted zone fits the target's dimensions precisely. As such, the search zone is reduced vertically and horizontally by $\Delta_H = |q \times H/n|$ and $\Delta_W = |q \times W/n|$, where $H, W$ are the keyboard's height and width (in pixels) and $n$ the number of zones. Accordingly, $\Delta_H, \Delta_W$ are split into top, bottom, left and right bound displacements $\Delta_{W_L}, \Delta_{W_R}, \Delta_{H_T}, \Delta_{H_B}$ where

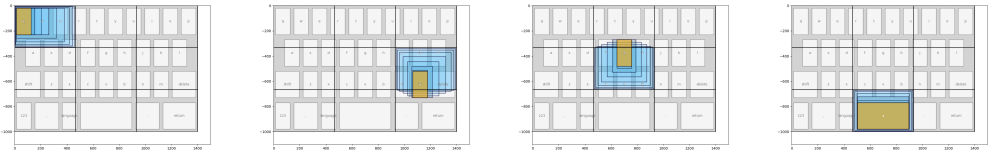


Fig. 6. Illustrative progression of gradual search zone adjustment from an initial $3 \times 3$ division, for the targets "Q", "M", "G", "SPACE" (from left to right). The final area of the zone is represented in orange.

$$L = \begin{cases} \text{W:} & \frac{|t_{x_0} - x_0| + t_x - t_{x_0}}{x_1 - x_0} \\ \text{H:} & \frac{|t_y - y_0|}{y_1 - y_0} \end{cases} \tag{5}$$

$$\Delta_{W_L} = \Delta_W \times L, \Delta_{W_R} = (1 - \Delta_{W_L}) \times \Delta_W$$
$$\Delta_{H_T} = \Delta_H \times L, \Delta_{H_B} = (1 - \Delta_{H_T}) \times \Delta_H$$

having $t_{x_0}, t_{y_0}$ as the target's left and bottom boundary coordinates, $t_x, t_y$ as the coordinates of target $t$'s center, $x_0, x_1, y_0, y_1$ the coordinates of the home zone's left, right, bottom and top bounds. Hard limits to search zone bound adjustments are implemented so that any adjustment cannot make the zone smaller than the actual size of the target itself. The process and resulting search zone adjustments are illustrated in Fig. 6.

Using this method, we repeat the analysis from Study 1, in order to examine how learning might progress when the probability to initiate a search in the home zone increases simultaneously with an adjustment of the search zone's boundaries. The results are shown in Fig. 7. As can be seen, in both constant and logistic rates of change of the learning velocity, users can achieve optimal performance regardless of the initial mental division of the keyboard into zones (see Table 2). Apparently, the best performance in terms of saccades per character can be achieved with a very low spatial resolution of $2 \times 1$ or $2 \times 2$, and in fact, the $2 \times 1$ layout offers the best performance overall, requiring the fewest number of trials in order to reach the limit of performance, as well as consequently requiring the shortest amount of time to complete the 500 simulated trials. To illustrate further, Fig. 8 shows the development of search zones from an initial $2 \times 1$ and $2 \times 2$ division after these 500 trials. As shown, the simulated user has learned the search zones almost perfectly for most keys in the $2 \times 1$ layout. Some letters are more frequent in the pangram (e.g. 'r', 't', 'h' appear twice), therefore for these, learning has reached an optimal state as they have been typed Using this method, we repeat the analysis from Study 1, in order to examine how learning might progress when the probability to initiate a search in the home zone increases simultaneously with an adjustment of the search zone's boundaries. The results are shown in Fig. 7. As can be seen, in both constant and logistic rates of change of the learning velocity, users can achieve optimal performance regardless of the initial mental division of the keyboard into zones (see Table 2). Apparently, the best performance in terms of saccades per character can be achieved with a very low spatial resolution of $2 \times 1$ or $2 \times 2$, and in fact, the $2 \times 1$ layout offers the best performance overall, requiring the fewest number of trials in order to reach the limit of performance, as well as consequently requiring the shortest amount of time to complete the 500 simulated trials. To illustrate further, Fig. 8 shows the development of search zones from an initial $2 \times 1$ and $2 \times 2$ division after these 500 trials. As shown, the simulated user has learned the search zones almost perfectly for most keys in the $2 \times 1$ layout. Some letters are more frequent in the pangram (e.g. 'r', 't', 'h' appear twice), therefore for these, learning has reached an optimal state as they have been

| Method | Zones | | | | | |
|---|---|---|---|---|---|---|
| | $2 \times 1$ | $2 \times 2$ | $3 \times 3$ | $3 \times 4$ | $4 \times 4$ | $5 \times 5$ |
| Constant | 2.090 | 1.942 | 2.260 | 2.705 | 2.578 | 2.635 |
| Logistic | 2.092 | 2.141 | 1.904 | 2.277 | 3.006 | 2.368 |

Table 2. Fitted function limits for dynamic zone adjustment (lowest values highlighted bold).

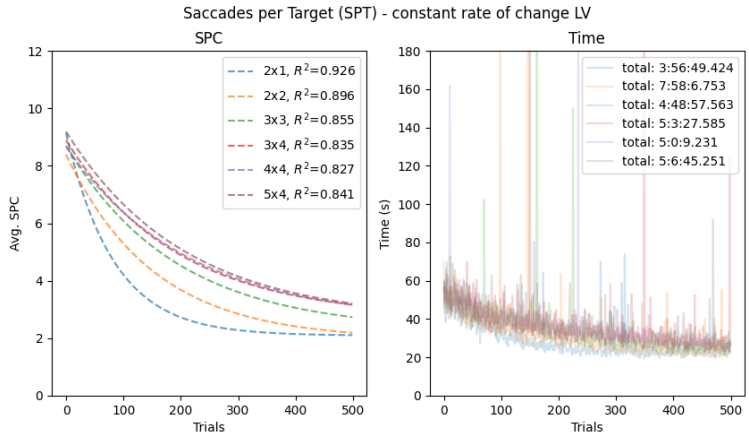
typed Using this method, we repeat the analysis from Study 1, in order to examine how learning might progress when the probability to initiate a search in the home zone increases simultaneously with an adjustment of the search zone's boundaries. The results are shown in Fig. 7. As can be seen, in both constant and logistic rates of change of the learning velocity, users can achieve optimal performance regardless of the initial mental division of the keyboard into zones (see Table 2). Apparently, the best performance in terms of saccades per character can be achieved with a very low spatial resolution of $2 \times 1$ or $2 \times 2$, and in fact, the $2 \times 1$ layout offers the best performance overall, requiring the fewest number of trials in order to reach the limit of performance, as well as consequently requiring the shortest amount of time to complete the 500 simulated trials. To illustrate further, Fig. 8 shows the development of search zones from an initial $2 \times 1$ and $2 \times 2$ division after these 500 trials. As shown, the simulated user has learned the search zones almost perfectly for most keys in the $2 \times 1$ layout. Some letters are more frequent in the pangram (e.g. 'r', 't', 'h' appear twice), therefore for these, learning has reached an optimal state as they have been typed $2 \times 500$ times. Home zone learning is less perfect for the $2 \times 2$ layout, however, as shown in Fig. 7, performance in terms of saccades per character is almost equal to that of the $2 \times 1$ layout. This is a result of users having foveal vision, i.e. being able to read more than one target labels in one fixation.

These findings demonstrate that users do not need to remember many mental zones - in fact, the best learning strategy is to remember whether a key is to the left or to the right of the keyboard. This alone provides a significant boost to learning in the initial stages. Simultaneous improvement by adjusting the size of the search zone of each key in memory, brings performance to the optimum level of just 2 saccades per characters, on average.

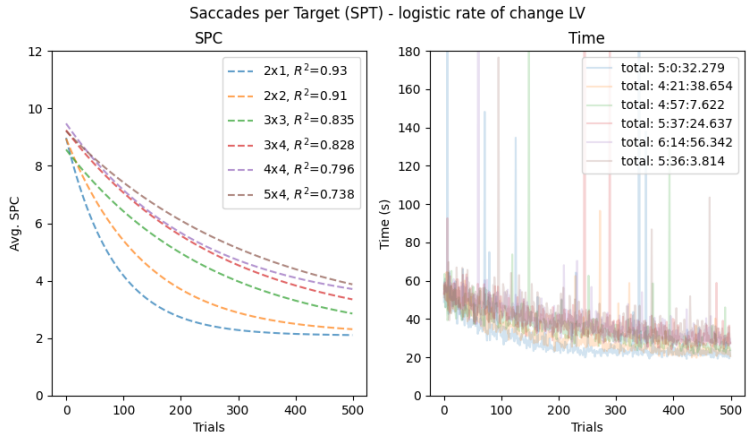
## 6 Study 3 - Performance in Simulated Experiments

The previous two studies demonstrated that human spatial memory for learning a keyboard layout can be modelled plausibly using a structure to represent a) the probability of a user initiating visual search for a target in the target's home zone and b) the spatial bounds of the target's search zone within the home zone. We can now examine if these findings and assumptions can be held to truth, by performing simulated transcription experiments and comparing the outcomes, in terms of metrics of proficiency, to known empirical results from previous studies. Note here that we should be interested only in the *patterns* of emergent behaviour through the simulation, and not the absolute values of metrics, as this is a matter of fitting model parameters rather than proving correct model behaviour. Therefore, we are not interested in, say, the final WPM that users could reach after a finite amount of trials, but are rather concerned with the shape of the emergent learning curve, as captured by increasing WPM through the trials, and in seeing if this shape would match empirical data.

Unfortunately, there are very few studies with longitudinal data showing the learning process for keyboard layouts. In this section, we will compare against the results from Jokinen et al. [34] and their model of learning new keyboard layouts, an empirical transcription task study by Mackenzine




(a) Constant rate of change, $\lambda = 0.003$




(b) Logistic rate of change, $\lambda = 0.003, k = 1/n^2, i_0 = 10 \times \lambda$

Fig. 7. Illustrative progression of learning a keyboard layout, by reducing the average number of saccades required to find a target while typing an English pangram, after 500 trials, for six mental zone layouts (with dynamic adjustment).

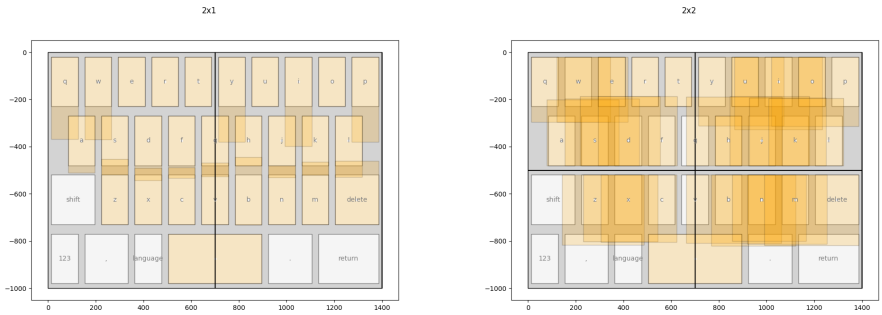



Fig. 8. Development of adjusted search zone sizes (orange) after 500 trials (with constant rate of change).

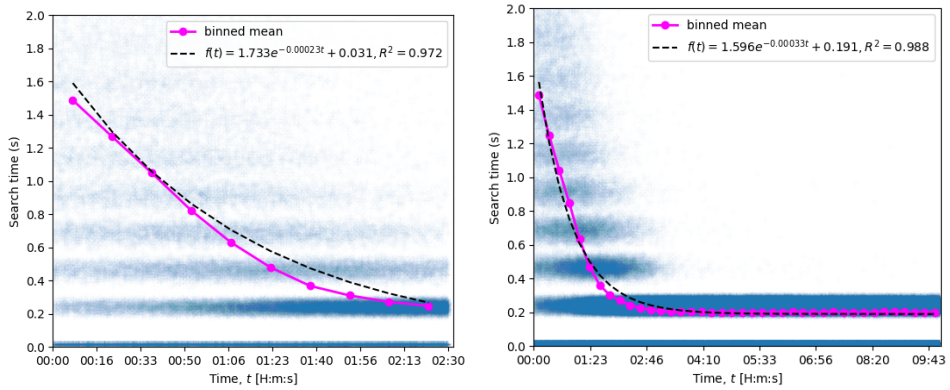
and Zhang [43] and a more recent study by Mutasim et al. [47]. In Study 2, we did not observe any notable differences between the constant and logistic learning velocity approaches, so in the next sections we employ the latter for evaluation. During the simulated experiments, we attempted to replicate the conditions, as described in the original studies, in as much detail as these studies contained. The simulations involved virtual participants, who are modelled after typing kinematics of real users (see [36, 37]), and we compared their performance against the results of actual user studies carried out in past research. This is akin to replicating the experiment carried out by other researchers, with the major exception, of course, being that we could not use the same human participants. Replication is usual methodological practice for the purpose of checking generalisability (though, sadly, not often encountered in HCI [18]). Therefore, we employ this replication process to contrast our model's result *patterns* (not actual results) to those in previous empirical studies with real human participants. Similar approaches are recently beginning to emerge in other communities, e.g. interactive information retrieval and others [3, 7, 8].

### 6.1 Comparison with Jokinen et al.

In Jokinen et al. [34], a computational model for learning new keyboard layouts was developed on the basis of the ACT-R cognitive architecture. This work presents the most complete up-to-date model in this area. To validate the model, the authors recruited 33 participants and asked them to perform a visual search task with a randomly shuffled English QWERTY keyboard, in multiple trials (10 blocks over four days, each block lasting 15 minutes of visual search tasks) . Participants did not need to type anything, but rather were asked to hit a reaction time button every time they located the assigned target. The main metric of performance in this study was search time in seconds. The experiment was performed on a computer, however dimensions of the screen size are not mentioned. The authors found an approximate halving of the search time (assumedly, mean) by the final session with learning progress following what appears to be an exponential decay curve. Unfortunately the data from this study are not provided by the authors. The model in this paper matches observed performance quite well ($R^2 = 0.97$), however, there are some concerns with these findings. As shown by simulations in typing on a mobile keyboard, the model predicts a maximum search time performance of approximately 0.38s per target even after 50+ hours of training, which, excluding finger movement time, results in a maximum theoretical input speed of 31.5WPM. This means that according to the model, users may reach the actual performance in [51] only if they do not need to move their fingers to type, and of course, never make any input errors.

To replicate the study, we include all virtual participants from [36] (n=30). Each participant is assigned a randomly shuffled English QWERTY keyboard, and performs as many simulated visual search-and-tap tasks of randomly selected letters from the English alphabet as possible, within $10 \times 15$ minute blocks, introducing a random 2.5-3.5 second delay between tasks, as per the original paper. The results are shown in Fig. 9a. As seen, search times are increasingly reduced as the model learns the spatial layouts assigned to each participant. After 2.5 hours of learning, the mean visual search time is reduced to 0.248s, which equates a theoretical maximum input speed (without finger movement) of 48.387WPM. This leaves adequate space to include motor function into the final achievable performance, e.g. if the average motor movement requires 0.15s, the resulting input speed would be more realistic 30WPM. In Fig. 9, an interesting artifact is the band of search times close to zero. These are generated by instances where the user has located the target successfully, but fails to hit the key and has to repeat the process, as error detection and correction behaviour is included in our simulation.

Fitting an exponential decay function of the general form $f(x) = \alpha \times e^{-\beta x} + \gamma$ to the raw data shown in Fig. 9a yields a very good fit ($R^2 = 0.972$) and as implied by the $\gamma$ parameter, the theoretical lower bound for search time is $\lim_{x\to\infty} f(x) = 0.031s$ (30ms), which is close to the known average




(a) 2.5 hours of training (b) 10 hours of training

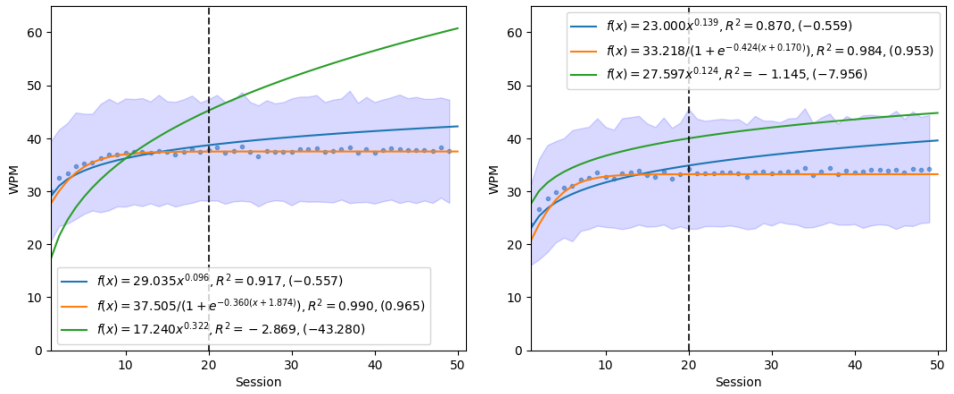
Fig. 9. Simulated search time for single characters in random QWERTY layouts ($\lambda = 0.015$). Search time means are calculated on time bins equivalent to the number of sessions (left:10, right:40)

duration single saccadic movements. This is somewhat unrealistic given that our model implements a saccadic latency for the first saccade between [120,170] milliseconds. By expanding the training to 10 hours ($40 \times$ 15-min sessions, see Fig. 9b), we note that $\lim_{x\to\infty} f(x) = 0.191s$ (191ms), which is more realistic and according to the saccadic parameters in our model. This "flattening" of the search time curve is also observed in simulation by Jokinen et al., therefore both models seem to produce equivalent behaviour predictions. In the next section, we compare our model behaviour against input speed metrics (which by definition include visual search and motor performance) from other studies.

### 6.2 Comparison with Mackenzie & Zhang

The previous experiment by Jokinen et al. [34] did not involve actual typing, which is usually conducted in lab studies via transcription tasks. We now consider the replication of a seminal transcription task study by Mackenzie & Zhang [43], where the OPTI layout was tested against QWERTY on a stylus-based touch screen. Five participants took 20 sessions of entry, with each session lasting approximately 20 minutes, with as many blocks of 10 phrases being administered as possible within each session (min 5, max 11 blocks). From these results, the authors extrapolate performance up to 50 sessions, fitting an exponential function to the OPTI data with a resulting $R^2 = 0.9974$, and $R^2 = 0.9802$ for QWERTY.

We perform a simulation where five randomly chosen virtual participants from [36] perform 50 sessions, fitting as many blocks of 10 phrases in each session as time allows (as per the original study), using a $2 \times 1$ imaginary zone initial layout. Unfortunately the original 70 phrase set used in the original study is not available, however we use the expanded set from the same authors [42]. To compare with the original study, we fit an exponential function to the data of the first 20 sessions. There is, however, a fundamental issue with the fitting of exponential functions, which is not mentioned in the original study. Any exponential function $f(x) = ax^b$ has $\lim_{x\to\infty} f(x) = \infty$, therefore fitting such a function implies that, given enough training, users will reach improbably high speeds. For example, using the fitted parameters in Mackenzie's & Zhang's study, at 236 sessions, a user would reach speeds of 100WPM with OPTI, while at the same number of sessions, QWERTY performance would be at 54WPM. Meanwhile, data from real human performance of




(a) OPTI (b) QWERTY

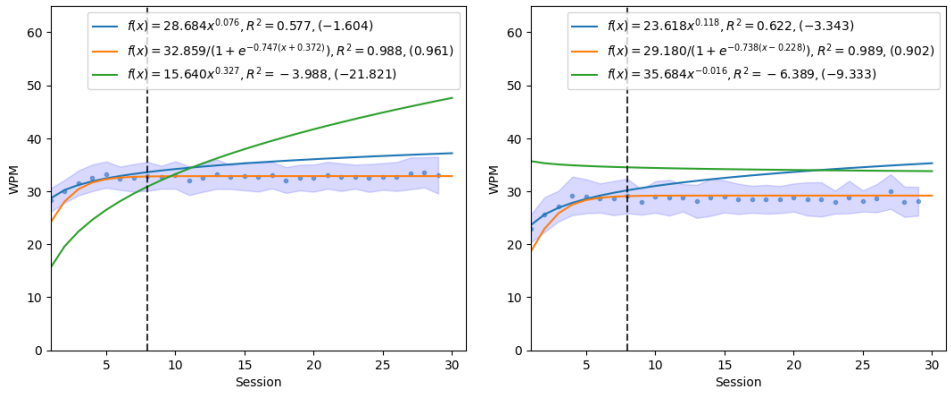
Fig. 10. Simulated performance with OPTI and QWERTY keyboard, error band at 95% c.i., as per the study protocol in Mackenzie & Zhang. $R^2$ values show the fit of the model produced with 20 sessions (in parentheses, the fit of the same model over data from 50 sessions) Green line shows model from Mackenzie & Zhang.

experienced users with QWERTY keyboards on smartphones (hours per day typing $\bar{x} = 6.5, \sigma = 6.2$), indicates that average typing speed with a single digit is between 25WPM ($\sigma = 11.4$) - 30.8WPM ($\sigma = 12.4$) (Palin et al. [51]), well below the predicted value of 54WPM. These limits to human performance are predicted also by our model and that of Jokinen et al. [34]. Therefore, we also fit a logistic function to the data of the first 20 sessions, as logistic functions are bound by their carrying capacity $L$.

In Fig. 10a we observe that for the OPTI keyboard, both the exponential and logistic functions fit the data quite well ($R^2 = 0.917, 0.990$ respectively). For comparison, we also plot the model curve in [43], which has a very low fit ($R^2 = -0.557$). This is to be expected as the virtual participants are modelled after other human subjects who may be considerably different from the participants in [43]. However, the fact that an exponential function fits the data well, as found in the original study, provides evidence that our learning model produces realistic results. The situation becomes more interesting, however, when examining the data after the 20th session. As shown in Fig. 10a, the exponential function predicts an increase in performance during our simulation, which is not matched by the produced data, resulting in $R^2 = -0.557$ for the whole data set. On the other hand, the logistic function fitted to the first 20 sessions, predicts a plateau in performance which matches the actual data ($R^2 = 0.965$).

On the other hand, QWERTY simulation for the first 20 sessions shows exponential and logistic functions fit the data quite well ($R^2 = 0.870, 0.984$ respectively). Again, the model function in [43] shows a bad fit ($R^2 = -1.147$) but interestingly the function parameters, and especially the exponent, are not that far off the parameters of the fitted exponential function. In terms of performance up to the 50th session, only the logistic function fits the observed data well ($R^2 = 0.953$), demonstrating an upper bound to performance with QWERTY at 34.246WPM ($\sigma = 11.543$), which is within the range of real performance reported by Palin et al [51]. Note here that the study of Mackenzie & Zhang used students from a CS department as subjects, therefore we might assume some familiarity with QWERTY keyboards, whereas our simulation assumes no prior familiarity with the layout, therefore the comparison is not direct.

In the original study, participant performance improved by 42% for QWERTY (from 28 to 40WPM) and 164% for OPTI (from 17 to 45WPM) at the end of the 20 sessions. With our simulation, participant




(a) OPTI (b) QWERTY

Fig. 11. Simulated performance with OPTI and QWERTY keyboard, error band at 95% c.i., as per the study protocol in Mutasim et al ($\lambda = 0.015$). $R^2$ values show the fit of the model produced with 8 sessions (in parentheses, the fit of the same model over data from 30 sessions). Green line shows model from Mutasim et al.

performance increased by 64.98% for QWERTY (14.799 to 24.415WPM) and 37.09% for OPTI (from 20.693 to 28.369WPM).

### 6.3 Comparison with Mutasim et al.

A more recent study by Mutasim et al. [47] aimed at investigating whether repeated typing of the same phrase may provide adequate data to demonstrate performance differentials between novel (OPTI) and QWERTY layouts, without the need for extensive longitudinal studies. In this study, eight participants performed eight sessions of repeated typing of a single phrase randomly chosen (unique for each participant) from the Mackenzie & Soukoreff set [42], which was unique to each participant, using both the OPTI and QWERTY keyboard. Participants were experienced in using soft QWERTY keyboards (>9 years of experience each). Each session consisted of 12 trials. The authors found that performance with QWERTY remained mostly constant at approximately 33WPM throughout the study ($\bar{x} = 33.7WPM, \sigma = 6.37$ on the first and $\bar{x} = 33.0WPM, \sigma = 5.94$ on the last session), with minor performance detriments in the final sessions attributed to user fatigue and boredom. This provides further evidence that performance plateaus are real and that exponential functions to model performance may lead to unrealistic expectations. Nevertheless, the authors fitted exponential functions to the data, to predict performance up to the 30th session.

We replicated this study using eight randomly selected virtual participants from [36], and, as per Mutasim et al., a random single phrase for each of them from the Mackenzie & Soukoreff set [42]. As per Mutasim et al., we also simulated user input for 30 sessions. We used the data from the first eight sessions to fit exponential and logistic curves. The results are shown in Fig. 11.

In the original study, participant performance decreased by 2% for QWERTY (from 33.7 to 33WPM) and increased by 63% for OPTI (from 19.1 to 31WPM) at the end of the eight sessions. With our simulation, participant performance increased by 53.63% for QWERTY (from $18.690WPM, \sigma = 2.944$ to $28.714WPM, \sigma = 4.660$) and 34.86% for OPTI (from $24.123WPM, \sigma = 2.733$ to $32.532WPM, \sigma = 3.744$). As expected, OPTI proves faster than QWERTY in the simulation, although the effect is not as pronounced as in the study by Mutasim et al.

# 7 Discussion

## 7.1 Overview of findings

In this paper, we proposed a theory for the formation of spatial memory in learning to type on mobile keyboards. Our theory stipulates that, in the absence of diverse and sparsely positioned salient visual landmarks, users may benefit from spatial learning through associating keys with imaginary zones (i.e. a grid-like mental division of the keyboard area). The theory was implemented in simulation code, and outputs from simulated experiments following, as closely as possible, the protocols of three previous studies, were compared against published empirical results.

Our findings indicate that our theoretical model is able to reproduce behaviours that are similar to those encountered in studies with real participants. We did not attempt to optimise or otherwise fit the model parameters to match the findings from these studies, since this would require full access to the data from these studies and the modelling of virtual participants after the kinematic parameters of the actual participants. Nevertheless, important findings such as the shape of learning curves and theoretical performance advantages of optimised layouts are replicated in our simulation, yielding confidence that the theory is plausible and may be used to explain the process of spatial memory development and recall during the learning process.

We also highlighted some important issues in previous studies, namely the use of exponential functions to predict future performance and performance limits. We demonstrate that such models may provide unrealistic expectations, since both our theory, and empirical evidence from large scale data collections show that input speed performance has hard upper bounds that were hinted at in [69] but were not predicted in other previous simulations.

## 7.2 Implications for design

This work bears some interesting implications for the design of novel keyboards as well as the development of training protocols to assist novice users. As mobile devices overtake desktop computing, increasing numbers of ICT users will become familiar with the soft keyboard input paradigm long before they actually get to use a physical keyboard. For large parts of the global population, soft keyboards will likely be the only touch-based text entry method they may encounter. Thus, design change in soft keyboards may be beneficial for the purposes of learning basic text entry skills, or for transitioning to the text entry in a new language.

The introduction of visual landmarks or layout grids in mobile keyboards may assist the quicker development of spatial memory in novice users, or users transitioning to QWERTY-like layout in foreign languages. For example, as shown in Fig. 12, colour-coding can help the user more readily associate key with keyboard zones, therefore promoting the initial formation of the mental layout. This type of visual assistance can be gradually faded out, as the user becomes more proficient and metrics on their performance are gathered automatically by the keyboard software. These findings can be extended not just to QWERTY-type layouts, but also to the design of ambiguous or semi-ambiguous keyboards that offer multiple character groupings in one key, using optimisation to reconfigure the character arrangement to a non-QWERTY-like grouping. Training and practice regimes could start with helping users remember the rough locations of keys at first, instead of normal procedure where new keys are gradually introduced to words that need to be transcribed. Knowledge of how users form and maintain spatial memory may also be helpful in implementing design mechanisms to support its decline or neurodiverse function, for example, in elderly populations, patients with early dementia onset, persons with mild cognitive impairment or learning disabilities. As text entry is a core digital skill, supportive and inclusive design is essential to narrow the digital divide across diverse population segments.



Fig. 12. Visual augmentation of keyboards in a 3 × 1 configuration. From top to bottom and left to right: Grid lines; Colour-coded keys; Colour-coded zones; Colour-coded tooltips (activated on key press); Landmark keys; Abstract visual landmarks.

Other types of multimodal feedback can also be considered to support spatial memory learning, for example, different vibrotactile feedback patterns or stereophonic auditory cues played back alongside keypresses could help users associate the rough location of target keys. Such feedback might be especially helpful for low-vision or blind users who may require more assistance to learn how to type on soft keyboards, while maintaining privacy during both learning and actual use.

### 7.3 Limitations and Future Work

In this paper, we focused our exploration of the theory on novel keyboard layouts, which, though they may differ from QWERTY, still adopt the paradigm of one-character-per-key. The proposed theory can be investigated in the context of learning to use ambiguous keyboards, such as 9-key layouts, circular or chorded keyboards. The simulator can be extended to support the modelling and interaction with such keyboards. More generally, the theory might find application in the formation of spatial models of any 2D user interface, in desktop, mobile or VR/AR environments. In this work we treated memory as a unified faculty of the brain, although the distribution of cognitive load in terms of the key location recall task, could be modelled more accurately by distinguishing between working and long-term memory functions, as well as modelling memory decay across typing sessions that are temporally separated. We leave this as future work to follow.

Although our theory is validated through comparison to extant literature and empirical findings, further work is required to obtain more empirical data to confirm the results of the simulation. In particular, eye-tracking and possibly EEG data could offer more insights into the visual search patterns with unfamiliar keyboard layouts, in order to confirm the development of mental spatial knowledge of the keyboard, as well as the effect of adding visual cues to assist its development, as e.g. suggested in Fig. 12. Other approaches, such as asking users to sketch or draw the derived mental models of the keyboard as they learn, may be conducive to assisting the understanding of the process of forming spatial memory and to further validate our theory [1, 56].

As mentioned, we did not proceed to fit model parameters to empirical data. Capture of data as mentioned previously would assist in creating more realistic synthetic users, and to validate the simulation further. The current model is based solely on single (index) finger input. Further work will be required to introduce kinematic models for thumb entry with one hand, or dual finger (thumb) entry, as is common in proficient users. Single thumb entry requires a different kinematic modelling approach, as the thumb's reach and accuracy vary according to the distance

of the target from the thumb's anchoring position. Dual thumb typing implies an inherent initial pre-mapping of keys in a $2 \times 1$ zone arrangement, to correspond with each thumb's range of reach. Empirical evidence from previous studies suggest that the initial or subsequently developed zones may not necessarily be of equal size, as in our experiment, as hand dominance or other cognitive factors may play a role in the preference of thumb use, e.g. prevalance of dominant thumb use, with opportunistic use of the non-dominant thumb [32]. As a result, we might expect to see different learning outcomes for users who choose to employ this typing strategy, or subjects with ambidextrous abilities.

Finally, we must mention here that for the comparison with Jokinen et al., the ocular movement model in the simulator includes a saccadic delay between 120-170ms as a priming delay for the first ocular movement during a search, and an ocular movement speed between 30-100px/msec, as per [23]. In the other comparisons, the saccadic delay is reduced to 10-20ms, as we assume the user does not need to look at the target before entering it (the whole phrase to be input is memorised) and ocular movement speed was set to 25-35msec, assuming greater familiarity with the experimental interface. Finally, the simulator assumes immediate detection and correction attempts for input errors, which is not necessarily the case with real text entry (i.e. users may not notice input errors until after a few characters have been input, or may not notice them at all), and gaze-shifts between presented target phrases, the input area and the keyboard are also not modelled. Although similar limitations apply to all state-of-the-art computational models in extant literature, further work on the simulator is needed to include these known aspects of behaviour during text entry.

## 8 Conclusion

This paper addressed the gap in understanding how users develop spatial memory for mobile keyboard layouts. While previous models could predict the probability of recalling key locations, they offered limited insight into the underlying cognitive processes of how this mental map is formed and refined through practice. The core problem investigated was the mechanism by which users transition from slow, search-based typing to faster, recall-based interaction on visually uniform soft keyboards lacking distinct landmarks. We proposed a theory centered on the idea that users initially form a coarse mental model by dividing the keyboard into a small number of imaginary "zones." Learning begins by associating keys with these zones. Through simulation using a cognitive architecture, we explored how these associations strengthen over time. Our findings suggest that while starting with finer-grained zone divisions might seem beneficial, a simple initial division (e.g., $2 \times 1$ or $2 \times 2$) combined with a dynamic adjustment mechanism proves most effective. This mechanism involves progressively refining the recalled search area within the initial "home zone" for each key, eventually approximating the key's precise location. This dual process—strengthening the zone association and refining the search area—allows the model to replicate empirically observed learning curves, including the characteristic performance plateaus often missed by models using unbounded exponential functions. Comparisons with simulated experiments mimicking protocols from prior empirical studies [34, 43, 47] demonstrated that our model produces plausible learning patterns and performance limits consistent with real-world user data.

The key takeaways from this research are threefold. First, the formation of spatial memory for interfaces like keyboards may not require complex, high-resolution mental maps from the outset; rather, a simple zonal division coupled with dynamic refinement of search boundaries for individual items appears sufficient and efficient. Second, this suggests that interface design and training could leverage this by initially emphasizing coarse landmarking or zonal cues (e.g., color-coding, subtle grid lines) that fade as proficiency increases. Third, computational modeling of skill acquisition, particularly for text entry, should incorporate mechanisms that account for performance saturation,

potentially using logistic functions rather than simple exponential models, to provide more realistic long-term predictions. Our findings suggest that any keyboard layout will inherently impose an upper bound on performance during use, and therefore it is inappropriate to derive exponential functions from empirical data to suggest predictions of performance after continued use.

Understanding this process of spatial memory formation offers valuable insights for designing more intuitive keyboard layouts, developing effective training strategies for novice users or those learning new languages, and potentially creating adaptive interfaces that better support cognitive learning patterns. Future work could explore the theory's applicability to different keyboard types (e.g., ambiguous, chorded) and other 2D interfaces, incorporate finer distinctions between memory systems, and gather further empirical data (e.g., eye-tracking, EEG, sketching) from real users, to refine and validate the model.

## 9 Data & Code Availability

Complete Python code for the model implementation and data analyses is provided at https://github.com/komis1/KeyboardSpatialMemory and as supplementary material for this paper.

## References


[1] Negar Ahmadpoor and Alastair D. Smith. 2020. Spatial Knowledge Acquisition and Mobile Maps: The Role of Environmental Legibility. *Cities* 101 (June 2020), 102700. doi:10.1016/j.cities.2020.102700

[2] Michel Jose Anzanello and Flavio Sanson Fogliatto. 2011. Learning Curve Models and Applications: Literature Review and Research Directions. *International Journal of Industrial Ergonomics* 41, 5 (Sept. 2011), 573–583. doi:10.1016/j.ergon.2011.05.001

[3] Leif Azzopardi, Timo Breuer, Björn Engelmann, Christin Kreutz, Sean MacAvaney, David Maxwell, Andrew Parry, Adam Roegiest, Xi Wang, and Saber Zerhoudi. 2024. SimIIR 3: A Framework for the Simulation of Interactive and Conversational Information Retrieval. In *Proceedings of the 2024 Annual International ACM SIGIR Conference on Research and Development in Information Retrieval in the Asia Pacific Region (SIGIR-AP 2024)*. Association for Computing Machinery, New York, NY, USA, 197–202. doi:10.1145/3673791.3698427

[4] Anne-Kathrin Bestgen, Dennis Edler, Christina Müller, Patrick Schulze, Frank Dickmann, and Lars Kuchinke. 2017. Where Is It (in the Map)? Recall and Recognition of Spatial Information. *Cartographica* 52, 1 (March 2017), 80–97. doi:10.3138/cart.52.1.3636

[5] Xiaojun Bi and Shumin Zhai. 2016. 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 on Human Factors in Computing Systems (CHI '16)*. Association for Computing Machinery, New York, NY, USA, 49–58. doi:10.1145/2858036.2858421

[6] Timothy F. Brady, Talia Konkle, George A. Alvarez, and Aude Oliva. 2008. Visual Long-Term Memory Has a Massive Storage Capacity for Object Details. *Proceedings of the National Academy of Sciences* 105, 38 (Sept. 2008), 14325–14329. doi:10.1073/pnas.0803390105

[7] Timo Breuer, Christin Katharina Kreutz, Norbert Fuhr, Krisztian Balog, Philipp Schaer, Nolwenn Bernard, Ingo Frommholz, Marcel Gohsen, Kaixin Ji, Gareth J. F. Jones, Jüri Keller, Jiqun Liu, Martin Mladenov, Gabriella Pasi, Johanne Trippas, Xi Wang, Saber Zerhoudi, and ChengXiang Zhai. 2025. Report on the 1st Workshop on Simulations for Information Access (Sim4IA 2024) at SIGIR 2024. *SIGIR Forum* 58, 2 (March 2025), 1–14. doi:10.1145/3722449.3722460

[8] Timo Breuer and Maria Maistro. 2024. Toward Evaluating the Reproducibility of Information Retrieval Systems with Simulated Users. In *Proceedings of the 2nd ACM Conference on Reproducibility and Replicability (ACM REP '24)*. Association for Computing Machinery, New York, NY, USA, 25–29. doi:10.1145/3641525.3663619

[9] Christina Bröhl, Peter Rasche, Janina Jablonski, Sabine Theis, Matthias Wille, and Alexander Mertens. 2018. Desktop PC, Tablet PC, or Smartphone? An Analysis of Use Preferences in Daily Activities for Different Technology Generations of a Worldwide Sample. In *Human Aspects of IT for the Aged Population. Acceptance, Communication and Participation*, Jia Zhou and Gavriel Salvendy (Eds.). Springer International Publishing, Cham, 3–20. doi:10.1007/978-3-319-92034-4_1

[10] Steven J. Castellucci, I. Scott MacKenzie, Mudit Misra, Laxmi Pandey, and Ahmed Sabbir Arif. 2019. TiltWriter: Design and Evaluation of a No-Touch Tilt-Based Text Entry Method for Handheld Devices. In *Proceedings of the 18th International Conference on Mobile and Ubiquitous Multimedia (MUM '19)*. Association for Computing Machinery, New York, NY, USA, 1–8. doi:10.1145/3365610.3365629

[11] Edward Clarkson, James Clawson, Kent Lyons, and Thad Starner. 2005. An Empirical Study of Typing Rates on Mini-QWERTY Keyboards. In *CHI '05 Extended Abstracts on Human Factors in Computing Systems (CHI EA '05)*. Association

for Computing Machinery, New York, NY, USA, 1288–1291. doi:10.1145/1056808.1056898

[12] Luis Fernando Cobar, Li Yuan, and Ayumu Tashiro. 2017. Place Cells and Long-Term Potentiation in the Hippocampus. *Neurobiology of Learning and Memory* 138 (Feb. 2017), 206–214. doi:10.1016/j.nlm.2016.10.010

[13] Arindam Das and Wolfgang Stuerzlinger. 2013. Unified Modeling of Proactive Interference and Memorization Effort: A New Mathematical Perspective within ACT-R Theory. *Proceedings of the Annual Meeting of the Cognitive Science Society* 35, 35 (2013). https://escholarship.org/uc/item/1p05s7db

[14] Frank Dickmann, Dennis Edler, Anne-Kathrin Bestgen, and Lars Kuchinke. 2017. Exploiting Illusory Grid Lines for Object-Location Memory Performance in Urban Topographic Maps. *The Cartographic Journal* 54, 3 (July 2017), 242–253. doi:10.1080/00087041.2016.1236509

[15] Denise Donica, Peter Giroux, and Young Joo Kim. 2019. Effectiveness of Two Keyboarding Instructional Approaches on the Keyboarding Speed, Accuracy, and Technique of Elementary Students. *The Open Journal of Occupational Therapy* 7, 4 (Oct. 2019), 1–15. doi:10.15453/2168-6408.1599

[16] Tafadzwa Joseph Dube and Ahmed Sabbir Arif. 2019. Text Entry in Virtual Reality: A Comprehensive Review of the Literature. In *Human-Computer Interaction. Recognition and Interaction Technologies*, Masaaki Kurosu (Ed.). Springer International Publishing, Cham, 419–437. doi:10.1007/978-3-030-22643-5_33

[17] Mark Dunlop and John Levine. 2012. Multidimensional Pareto Optimization of Touchscreen Keyboards for Speed, Familiarity and Improved Spell Checking. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12)*. Association for Computing Machinery, New York, NY, USA, 2669–2678. doi:10.1145/2207676.2208659

[18] Florian Echtler and Maximilian Häußler. 2018. Open Source, Open Science, and the Replication Crisis in HCI. In *Extended Abstracts of the 2018 CHI Conference on Human Factors in Computing Systems (CHI EA '18)*. Association for Computing Machinery, New York, NY, USA, 1–8. doi:10.1145/3170427.3188395

[19] Dennis Edler, Frank Dickmann, Anne-Kathrin Bestgen, and Lars Kuchinke. 2014. The Effects of Grid Line Separation in Topographic Maps for Object Location Memory. *Cartographica* 49, 4 (Dec. 2014), 207–217. doi:10.3138/carto.49.4.2674

[20] Dennis Edler, Julian Keil, Lars Kuchinke, and Frank Dickmann. 2019. Correcting Distortion Errors in Memory of Object Locations: The Example of Grid Line Spacing in Topographic Maps. *International Journal of Cartography* 5, 1 (Jan. 2019), 92–109. doi:10.1080/23729333.2018.1532651

[21] Brian D. Ehret. 2002. Learning Where to Look: Location Learning in Graphical User Interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '02)*. Association for Computing Machinery, New York, NY, USA, 211–218. doi:10.1145/503376.503414

[22] Howard Eichenbaum, Paul Dudchenko, Emma Wood, Matthew Shapiro, and Heikki Tanila. 1999. The Hippocampus, Memory, and Place Cells: Is It Spatial Memory or a Memory Space? *Neuron* 23, 2 (June 1999), 209–226. doi:10.1016/S0896-6273(00)80773-4

[23] John Enderle. 2005. 12 - PHYSIOLOGICAL MODELING. In *Introduction to Biomedical Engineering (Second Edition)*, John D. Enderle, Susan M. Blanchard, and Joseph D. Bronzino (Eds.). Academic Press, Boston, 693–798. doi:10.1016/B978-0-12-238662-6.50014-8

[24] Sanjay Ghosh, Anirudha Joshi, Manjiri Joshi, Nagraj Emmadi, Girish Dalvi, Shashank Ahire, and Swati Rangale. 2017. Shift+Tap or Tap+LongPress? The Upper Bound of Typing Speed on InScript. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*. Association for Computing Machinery, New York, NY, USA, 2059–2063. doi:10.1145/3025453.3025944

[25] A. M. Gordon, J.-H. Lee, D. Flament, K. Ugurbil, and T. J. Ebner. 1998. Functional Magnetic Resonance Imaging of Motor, Sensory, and Posterior Parietal Cortical Areas during Performance of Sequential Typing Movements. *Experimental Brain Research* 121, 2 (June 1998), 153–166. doi:10.1007/s002210050447

[26] Hiroyuki Hakoda, Buntarou Shizuki, and Jiro Tanaka. 2016. QAZ Keyboard: QWERTY Based Portrait Soft Keyboard. In *Design, User Experience, and Usability: Technological Contexts*, Aaron Marcus (Ed.). Springer International Publishing, Cham, 24–35. doi:10.1007/978-3-319-40406-6_3

[27] Yuichi Higashiyama, Katsuhiko Takeda, Yoshiaki Someya, Yoshiyuki Kuroiwa, and Fumiaki Tanaka. 2015. The Neural Basis of Typewriting: A Functional MRI Study. *PLOS ONE* 10, 7 (July 2015), e0134131. doi:10.1371/journal.pone.0134131

[28] Yu-Chun Huang and Fong-Gong Wu. 2015. Visual and Manual Loadings with QWERTY-like Ambiguous Keyboards: Relevance of Letter-Key Assignments on Mobile Phones. *International Journal of Industrial Ergonomics* 50 (Nov. 2015), 143–150. doi:10.1016/j.ergon.2015.08.002

[29] Catalina Iordache, Ilse Mariën, and Dorien Baelden. 2017. Developing Digital Skills and Competences: A Quick-Scan Analysis of 13 Digital Literacy Models. *Italian Journal of Sociology of Education* 9, Italian Journal of Sociology of Education 9/1 (Feb. 2017), 6–30. https://ijse.padovauniversitypress.it/2017/1/2

[30] Lucia F. Jacobs. 2003. Memory, Spatial. In *Encyclopedia of the Neurological Sciences*, Michael J. Aminoff and Robert B. Daroff (Eds.). Academic Press, New York, 86–90. doi:10.1016/B0-12-226870-9/01375-7

[31] Xinhui Jiang, Jussi P. P. Jokinen, Antti Oulasvirta, and Xiangshi Ren. 2022. Learning to Type with Mobile Keyboards: Findings with a Randomized Keyboard. *Computers in Human Behavior* 126 (Jan. 2022), 106992. doi:10.1016/j.chb.2021.1

06992

[32] Xinhui Jiang, Yang Li, Jussi P.P. Jokinen, Viet Ba Hirvola, Antti Oulasvirta, and Xiangshi Ren. 2020. How We Type: Eye and Finger Movement Strategies in Mobile Typing. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. Association for Computing Machinery, New York, NY, USA, 1–14. https://doi.org/10.1145/3313831.3376711

[33] Jussi Jokinen, Aditya Acharya, Mohammad Uzair, Xinhui Jiang, and Antti Oulasvirta. 2021. Touchscreen Typing As Optimal Supervisory Control. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems (CHI '21)*. Association for Computing Machinery, New York, NY, USA, 1–14. doi:10.1145/3411764.3445483

[34] Jussi P. P. Jokinen, Sayan Sarcar, Antti Oulasvirta, Chaklam Silpasuwanchai, Zhenxin Wang, and Xiangshi Ren. 2017. Modelling Learning of New Keyboard Layouts. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*. Association for Computing Machinery, New York, NY, USA, 4203–4215. doi:10.1145/3025453.3025580

[35] Julian Keil, Dennis Edler, Lars Kuchinke, and Frank Dickmann. 2020. Effects of Visual Map Complexity on the Attentional Processing of Landmarks. *PLOS ONE* 15, 3 (March 2020), e0229575. doi:10.1371/journal.pone.0229575

[36] Andreas Komninos. 2023. RoboType: Realistic Mobile Text Entry Evaluations with Synthetic Users. In *Proceedings of the 25th International Conference on Mobile Human-Computer Interaction*. ACM. doi:10.1145/3565066.3608693

[37] Andreas Komninos, Ioulia Simou, and John Garofalakis. 12/2024 2024. Towards Psycho-Cognitive Modelling of Mobile Text Entry Behaviour. In *28th Pan-Hellenic Conference on Progress in Computing Informatics (PCI 2024)*. Association for Computing Machinery, Athens, Greece. doi:10.1145/3716554.3716581

[38] Talia Konkle, Timothy F. Brady, George A. Alvarez, and Aude Oliva. 2010. Scene Memory Is More Detailed Than You Think: The Role of Categories in Visual Long-Term Memory. *Psychological Science* 21, 11 (Nov. 2010), 1551–1556. doi:10.1177/0956797610385359

[39] Sébastien Lallé, Dereck Toker, Cristina Conati, and Giuseppe Carenini. 2015. Prediction of Users' Learning Curves for Adaptation While Using an Information Visualization. In *Proceedings of the 20th International Conference on Intelligent User Interfaces*. ACM, Atlanta Georgia USA, 357–368. doi:10.1145/2678025.2701376

[40] Frank Chun Yat Li, Richard T. Guy, Koji Yatani, and Khai N. Truong. 2011. The 1line Keyboard: A QWERTY Layout in a Single Line. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology (UIST '11)*. Association for Computing Machinery, New York, NY, USA, 461–470. doi:10.1145/2047196.2047257

[41] Roy Luria, Paola Sessa, Alex Gotler, Pierre Jolicœur, and Roberto Dell'Acqua. 2010. Visual Short-term Memory Capacity for Simple and Complex Objects. *Journal of Cognitive Neuroscience* 22, 3 (March 2010), 496–512. doi:10.1162/jocn.2009.21214

[42] I. Scott MacKenzie and R. William Soukoreff. 2003. Phrase Sets for Evaluating Text Entry Techniques. In *CHI '03 Extended Abstracts on Human Factors in Computing Systems (CHI EA '03)*. Association for Computing Machinery, New York, NY, USA, 754–755. doi:10.1145/765891.765971

[43] I. Scott MacKenzie and Shawn X. Zhang. 1999. The Design and Evaluation of a High-Performance Soft Keyboard. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '99)*. Association for Computing Machinery, New York, NY, USA, 25–31. doi:10.1145/302979.302983

[44] George A. Miller, Galanter Eugene, and Karl H. Pribram. 1968. Plans and the Structure of Behaviour. In *Systems Research for Behavioral Science*. Routledge.

[45] Ehsan Sotoodeh Mollashahi, Md. Sami Uddin, and Carl Gutwin. 2018. Improving Revisitation in Long Documents with Two-Level Artificial-Landmark Scrollbars. In *Proceedings of the 2018 International Conference on Advanced Visual Interfaces (AVI '18)*. Association for Computing Machinery, New York, NY, USA, 1–9. doi:10.1145/3206505.3206554

[46] May-Britt Moser, David C. Rowland, and Edvard I. Moser. 2015. Place Cells, Grid Cells, and Memory. *Cold Spring Harbor Perspectives in Biology* 7, 2 (Feb. 2015), a021808. doi:10.1101/cshperspect.a021808

[47] Aunnoy K Mutasim, Anil Ufuk Batmaz, Moaaz Hudhud Mughrabi, and Wolfgang Stuerzlinger. 2023. Does Repeatedly Typing the Same Phrase Provide a Good Estimate of Expert Text Entry Performance?. In *Extended Abstracts of the 2023 CHI Conference on Human Factors in Computing Systems (CHI EA '23)*. Association for Computing Machinery, New York, NY, USA, 1–8. doi:10.1145/3544549.3585647

[48] Aunnoy K Mutasim, Anil Ufuk Batmaz, Moaaz Hudhud Mughrabi, and Wolfgang Stuerzlinger. 2024. The Guided Evaluation Method: An Easier Way to Empirically Estimate Trained User Performance for Unfamiliar Keyboard Layouts. *International Journal of Human-Computer Studies* 190 (Oct. 2024), 103317. doi:10.1016/j.ijhcs.2024.103317

[49] Donald A. Norman. 1990. *The Design of Everyday Things*. Doubleday.

[50] Enass Oliemat, Fathi Ihmeideh, and Mustafa Alkhawaldeh. 2018. The Use of Touch-Screen Tablets in Early Childhood: Children's Knowledge, Skills, and Attitudes towards Tablet Technology. *Children and Youth Services Review* 88 (May 2018), 591–597. doi:10.1016/j.childyouth.2018.03.028

[51] Kseniia Palin, Anna Maria Feit, Sunjun Kim, Per Ola Kristensson, and Antti Oulasvirta. 2019. How Do People Type on Mobile Devices? Observations from a Study with 37,000 Volunteers. In *Proceedings of the 21st International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '19)*. Association for Computing Machinery,

New York, NY, USA, 1–12. doi:10.1145/3338286.3340120

[52] Jeremy J. Purcell, Eileen M. Napoliello, and Guinevere F. Eden. 2011. A Combined fMRI Study of Typed Spelling and Reading. *NeuroImage* 55, 2 (March 2011), 750–762. doi:10.1016/j.neuroimage.2010.11.042

[53] Jochen Rick. 2010. Performance Optimizations of Virtual Keyboards for Stroke-Based Text Entry on a Touch-Based Tabletop. In *Proceedings of the 23nd Annual ACM Symposium on User Interface Software and Technology (UIST '10)*. Association for Computing Machinery, New York, NY, USA, 77–86. doi:10.1145/1866029.1866043

[54] Md. Sami Uddin and Carl Gutwin. 2021. The Image of the Interface: How People Use Landmarks to Develop Spatial Memory of Commands in Graphical Interfaces. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems (CHI '21)*. Association for Computing Machinery, New York, NY, USA, 1–17. doi:10.1145/3411764.3445050

[55] Joey Scarr, Andy Cockburn, and Carl Gutwin. 2013. Supporting and Exploiting Spatial Memory in User Interfaces. *Foundations and Trends® in Human–Computer Interaction* 6, 1 (Dec. 2013), 1–84. doi:10.1561/1100000046

[56] Steven Schirra, Shraddhaa Narasimha, Sasha Volkov, and Justin Owens. 2022. Understanding User Mental Models Through App Sketches from Memory. In *Extended Abstracts of the 2022 CHI Conference on Human Factors in Computing Systems (CHI EA '22)*. Association for Computing Machinery, New York, NY, USA, 1–6. doi:10.1145/3491101.3519753

[57] Richard A. Schmidt and Timothy Donald Lee. 2005. *Motor Control and Learning: A Behavioral Emphasis*. Human Kinetics.

[58] Andrew Sears, Julie A. Jacko, Josey Chu, and Francisco Moro. 2001. The Role of Visual Search in the Design of Effective Soft Keyboards. *Behaviour & Information Technology* 20, 3 (Jan. 2001), 159–166. doi:10.1080/01449290110049790

[59] Danqing Shi, Yujun Zhu, Jussi P. P. Jokinen, Aditya Acharya, Aini Putkonen, Shumin Zhai, and Antti Oulasvirta. 2024. CRTypist: Simulating Touchscreen Typing Behavior via Computational Rationality. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems (CHI '24)*. Association for Computing Machinery, New York, NY, USA, 1–17. doi:10.1145/3613904.3642918

[60] R. William Soukoreff and I. Scott MacKenzie. 1995. Theoretical Upper and Lower Bounds on Typing Speed Using a Stylus and a Soft Keyboard. *Behaviour & Information Technology* 14, 6 (Nov. 1995), 370–379. doi:10.1080/01449299508914656

[61] Ke Sun, Chun Yu, and Yuanchun Shi. 2019. Exploring Low-Occlusion Qwerty Soft Keyboard Using Spatial Landmarks. *ACM Trans. Comput.-Hum. Interact.* 26, 4 (June 2019), 20:1–20:33. doi:10.1145/3318141

[62] Md. Sami Uddin. 2016. Use of Landmarks to Design Large and Efficient Command Interfaces. In *Proceedings of the 2016 ACM Companion on Interactive Surfaces and Spaces (ISS '16 Companion)*. Association for Computing Machinery, New York, NY, USA, 13–17. doi:10.1145/3009939.3009942

[63] Md. Sami Uddin, Carl Gutwin, and Andy Cockburn. 2017. The Effects of Artificial Landmarks on Learning and Performance in Spatial-Memory Interfaces. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*. Association for Computing Machinery, New York, NY, USA, 3843–3855. doi:10.1145/3025453.3025497

[64] Md. Sami Uddin, Carl Gutwin, and Alix Goguey. 2017. Using Artificial Landmarks to Improve Revisitation Performance and Spatial Learning in Linear Control Widgets. In *Proceedings of the 5th Symposium on Spatial User Interaction (SUI '17)*. Association for Computing Machinery, New York, NY, USA, 48–57. doi:10.1145/3131277.3132184

[65] Md. Sami Uddin, Carl Gutwin, and Benjamin Lafreniere. 2016. HandMark Menus: Rapid Command Selection and Large Command Sets on Multi-Touch Displays. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. Association for Computing Machinery, New York, NY, USA, 5836–5848. doi:10.1145/2858036.2858211

[66] Keith Vertanen and Per Ola Kristensson. 2011. A Versatile Dataset for Text Entry Evaluations Based on Genuine Mobile Emails. In *Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services (MobileHCI '11)*. Association for Computing Machinery, New York, NY, USA, 295–298. doi:10.1145/2037373.2037418

[67] Günter Vetter, Michael Stadler, and John D. Haynes. 1997. Phase Transitions in Learning. *The Journal of Mind and Behavior* 18, 2/3 (1997), 335–350. jstor:43853827 https://www.jstor.org/stable/43853827

[68] Danielle Yanes, Emily Frith, and Paul D. Loprinzi. 2019. Memory-Related Encoding-Specificity Paradigm: Experimental Application to the Exercise Domain. *Europe's Journal of Psychology* 15, 3 (Sept. 2019), 447–458. doi:10.5964/ejop.v15i3.1767

[69] Shumin Zhai, Michael Hunter, and Barton A. Smith. 2002. Performance Optimization of Virtual Keyboards. *Human–Computer Interaction* 17, 2-3 (Sept. 2002), 229–269. doi:10.1080/07370024.2002.9667315

[70] Shumin Zhai, Alison Sue, and Johnny Accot. 2002. Movement Model, Hits Distribution and Learning in Virtual Keyboarding. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '02)*. Association for Computing Machinery, New York, NY, USA, 17–24. doi:10.1145/503376.503381

[71] Suwen Zhu, Tianyao Luo, Xiaojun Bi, and Shumin Zhai. 2018. Typing on an Invisible Keyboard. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. Association for Computing Machinery, New York, NY, USA, 1–13. doi:10.1145/3173574.3174013