

# Keystroke-Level Model for Advanced Mobile Phone Interaction

**Paul Holleis**

Research Group  
Embedded Interaction  
University of Munich  
paul@hciab.org

**Friederike Otto, Heinrich Hußmann**

Media Informatics Group  
University of Munich  
friederike.otto@ifi.lmu.de  
heinrich.hussmann@ifi.lmu.de

**Albrecht Schmidt**

Fraunhofer IAIS, Sankt Augustin  
B-IT, University of Bonn  
<http://uie.bit.uni-bonn.de/>  
albrecht.schmidt@acm.org

## ABSTRACT

The design of applications using mobile devices needs a different quality assessment than those known for desktop applications. Of the many aspects that have to be taken into account, one important criterion is the average time users need to complete a task. For interactions with the mouse, keyboard or touch screens, there exist models that predict interaction times like Fitts' law or the Keystroke-Level Model (KLM). This paper shows parallels to these models for advanced interactions with mobile phones targeted at pervasive services, including near field communication as well as built-in cameras and sensors. Applications can be evaluated with respect to user performance time without having a prototype running on the phone. To accomplish that, we extend the known KLM by identifying basic interaction elements for mobile phones and give estimates for expert user performance derived from several user tests.

## Author Keywords

Keystroke-Level Model (KLM), user performance, design decisions, real world interaction, mobile phone interaction

## ACM Classification Keywords

H.1.2 User/Machine Systems: Human factors. H.5.2 User Interfaces: Evaluation/methodology, Interaction styles

## INTRODUCTION

Experience has shown that it is essential to assess designs and applications early in the development phase. The phone company NYNEX probably saved millions of dollars [14] because the Keystroke-Level Model (KLM) was used to find out that the interaction performance of a newly designed workstation would have been worse than the existing system. This was possible without having to actually build and test the new system at all.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

CHI 2007, April 28–May 3, 2007, San Jose, California, USA.  
Copyright 2007 ACM 978-1-59593-593-9/07/0004...\$5.00.

Even though time to completion of a task is only one aspect of a promising application, it is an important factor for a large set of applications ranging from small games to reservation systems, from sub tasks of larger systems to support and search systems. This is especially the case for applications designed as side tasks or that exploit people's precious and short amount of spare time, e.g., between two tasks or while waiting for a meeting or the bus. These fundamentally rely on quick and hassle-free interactions. In addition to games and entertainment, mobile phones are increasingly used to enhance productivity and throughput in various fields like security or ticket sale.

In this paper we focus on time/performance predictions for the evolving domain of mobile phone interactions building upon KLM. This choice is motivated by the large number of publications in the CHI environment using KLM in a variety of emerging application domains. Many projects in cognitive modelling such as ACT-R [2] rely on such data in ongoing research areas like in-vehicle interfaces. We adopt and define a set of operators giving sound and study-based estimates of performance measures for each of them. Developers of mobile applications, which possibly include identification tags and smart objects, can then describe tasks as a sequence of these operators and predict user interaction times without even needing to create prototypes. Table 2 shows an annotated excerpt of a model resulting from the new mobile phone KLM developed in this paper.

As an additional application area, we propose cross-platform evaluation as ongoing work in our lab. It is easy to enhance any type of prototype like a paper or interactive HTML/Flash prototype to generate a KLM of a given task sequence. Our model estimates execution time of those tasks on, e.g., a mobile phone without the need to have a single line of code actually running on a phone.

Although it sometimes seen as a drawback of such models that these assume nearly error-free expert user interaction, KLM has revealed remarkably precise prediction results in several projects (e.g. [5,15,25,37]). Even in cases where experimental studies indicated that estimates in fact were considerably off the actual values, the estimated difference between two examined designs still proved to be a strong basis for making a choice between them (e.g. [9,23,28]).

## BACKGROUND

Modelling user tasks and processes in general attained much attention from researchers, not only in the last decades. The pioneering work of Card, Moran and Newell introduced the GOMS (Goals, Operators, Methods, Selection rules) model in 1980 [7,8]. It is often cited as one of the most mature engineering models of human performance. The GOMS elements enable designers to model expert user behaviour during a given task and therefore to analyse user complexity for interactive systems including cognitive information processing activities. Of course, even with certain extensions developed later, this family of models can only account for a small subset of aspects that decide whether a product is successful or not. However, many empirical studies have confirmed that it suffices to support justified choices between similar approaches to the same problem (e.g., [12,13, 14,20]).

In its beginning, GOMS was targeted mainly at text editing tasks on office desktop computers. For that purpose, the Keystroke-Level Model (KLM, [7,21]), a tailored instance of GOMS, was developed. A task can be described using operators that model unit tasks like key presses, pointing, hand switches between mouse and keyboard, mental acts, system response times and others. With a set of user studies the authors were able to give estimates for the duration of these actions and evaluate those estimates.

Several projects (e.g., [5,15,23,37]) successfully used and validated those values in various application areas. Others slightly adjusted or added one or more operators for a specific application setting (e.g., [25]). Variants of GOMS have been applied to many different application domains: phone operators [14], visual and auditory perception (John, [19]), a tool selection technique for tablet computers [16] (Hinckley et al., CHI'06), and several others mentioned below and in [8].

## RELATED WORK

In this paper, we are building upon KLM, revisiting the timing specifications for existing operators and extending it with new operators necessary to describe interaction with mobile phones. The number of possible types of mobile phone interaction have increased and achieved much attention from various sources in the last years. However, most research on performance measures for phone users has yet been limited to the input of text for short messages: An initial work by Dunlop and Crossan [9], shows KLM operator sequences for three different text entry methods (traditional, predictive, and word completion). However, the authors adopted the original operator values used for desktop interaction which proved to be imprecise in this new environment. This is improved in How and Kan [17] where the presented model is more fine-grained. They define 13 operators that more directly map onto the phone keyboard interface according to the different input methods (multi-tab etc.). New times are gathered from video taped sessions with a small set of subjects and a message typing task. In a complementary approach pursued in 2004 by

Pavlovych and Stuerzlinger [29], non-perfect users are considered using the cognitive load operator to model input verification. Although we do not focus on text messages, our model supports this view using *Micro Attention Shifts* explained later. [29] also compares several existing variants of the KLM and evaluates them with actual times from user studies. The authors conclude that the models give only rough approximations to real user behaviour for text input. However, the models can correctly predict which input methods are faster than others (e.g., predictive over multi-tab). There is also work reporting on time measurements for key presses and the mental act operator for text input in different languages (e.g. Myung for Korean [28]).

In addition to text input, Mori et al. [26] studied how the time values of the original KLM operators apply to mobile phone menu navigation and conclude that the operator values fit quite well and suggest only minor modifications.

There is to our knowledge no published research yet that includes new mobile interaction techniques in its model. A beginning is indicated in Luo and John [24] and its follow-up (Teo and John [37], 2006) where the authors show that the method can be soundly applied to handheld devices using stylus-based interfaces. They also present a tool they developed to automatically generate KLM models from storyboard descriptions and state that they plan to apply such research to novel interfaces like speech or gestures.

After a short treatment of mobile phone interaction types we first briefly describe the set of new, adapted and adopted parameters of our model and then present the studies that led to updated time values. Their results are summarised and an evaluation of the values is described. We close the paper with a discussion of the results.

## Mobile Phone Interactions

Mobile phones have mainly been used to make phone calls, send text messages, and sometimes as calendar. However, other uses are becoming more and more popular. Taking pictures, surfing the web, storing data, and playing music as well as videos are some of them. Additionally, researchers started to use it as universal remote control (e.g., Myers [27]) and suggest further interactions with the world (see Figure 1). This adds several new interaction styles that have



Figure 1: A sample of physical mobile phone interactions: using tags and taking pictures (of visual markers).

not yet been treated by any interaction model. In [32], Rukzio et al. define *physical mobile interactions* as being interactions between a user, a mobile device, and a smart object in the real world. The user interacts with the mobile device and the mobile device interacts with the smart object. This allows the implementation of systems envisioned, e.g., in [22,33] and allows bridges to be built between the physical and virtual worlds using devices that many people carry with them, c.f. Want et al. [40].

Although many people still think that mobile phone applications include games and entertainment only, phones are increasingly used to enhance productivity. In Japan, for example, it is common to buy tickets for public transport with the phone. Security personnel can use a tag reading mobile device to quickly log the places they have checked.

Up to now there is no user performance model available for physical mobile interactions. Rukzio et al. [32] studied general preferences of people to use a specific physical mobile interaction method (touching, pointing and scanning). They also show that timing is an issue for users. However, no quantitative performance numbers have been measured and only individual opinions of subjects are given. Ballagas et al. [3] describe, explore and categorise a multitude of interaction types with mobile phones but do not give any timings or comparisons in that respect.

### Interaction Types and Technology

Besides number entry and menu selection there are several ways physical mobile interactions can be implemented. [1] and [3] give an overview of current technology for physical selection (visual patterns, electromagnetic and infrared methods) and compare them according to several characteristics like transfer rate and operating range. In [38] and [39], as well as in [32], projects are described using prototypical implementations of three basic physical selection techniques, *Touching* (using RFID [40], Near Field Communication [32], or proximity sensors [38]), *Pointing* (visual codes like Semacodes and QR Codes [30,31], laser pointer and light sensors [38], IrDA (Deutsche Post: Mobilepoint), or object recognition [10]), and *Scanning* (WLAN, GPS, Bluetooth [32]).

Another interaction method we investigated is performing gestures. The underlying technology is based on tracking the phone by an external camera, using the phone's camera [4], or reading built-in sensors (found, e.g., in the Nokia Fun Shell or the Samsung SGH-E760 and S4000 phones).

We keep the revised KLM as general as possible to be able to offer operators to model most of these types of actions and give accurate estimates for some special cases.

### MODEL PARAMETERS

In this section we show differences and similarities between the KLM used for desktop interaction and the new KLM for mobile phone interactions. The original KLM defines 6 operators and assigns time values to each of them: *Keystroke* (**K**, key and button presses), *Pointing* (**P**, mouse

movements), *Drawing* (**D**( $n_D, I_D$ ), straight lines drawings with the mouse), *Homing* (**H**, hand movement between keyboard and mouse), *Mental Act* (**M**, pauses needed for reflection, choice, etc), and *System Response Time* (**R**( $t$ ), user waits for the system). Some operators have to be added to describe interactions that do not exist in the standard desktop metaphor. Others have to be examined closely to ensure that the original timing specifications are still applicable or to be able to derive new values. Others again are not applicable to the phone setting at all. The execution time of a task in the new model is then given by

$$T_{execute} = \sum_{op \in OP} (n_{op} + d_{op} \cdot X_{slight} + D_{op} \cdot X_{strong}) \cdot op$$

where  $OP = \{A, F, G, H, I, K, M, P, R, S_{Micro}, S_{Macro}\}$  is the set of available operators and  $n_{op}$ ,  $d_{op}$ ,  $D_{op}$  are the numbers of occurrences of the operator  $op$  in the model without distraction, with slight, and with strong distraction, respectively. Distractions are modeled with the **X** operator.

### New Operators

#### Macro Attention Shift ( $S_{Macro}$ )

One major difference from desktop to phone interaction is that the attention of users may be split between the phone and the real world surrounding them (Figure 3).



Figure 3: Attention shift ( $S_{Macro}$ ) between the mobile phone and objects in the real world.

Thus, a *Macro Attention Shift* operator models the time needed to shift the focus between the contents on the screen of the mobile device to an object (e.g., a poster) in the real world and vice versa. The original KLM does not need to consider this case since it assumes that the whole interaction session takes place on one single screen.

#### Micro Attention Shift ( $S_{Micro}$ )

$S_{Micro}$  models the time needed to look from the display to the keypad and hotkey regions and vice versa (Figure 2).



Figure 2: Regions of a standard mobile phone: keypad, hotkeys, and display. The  $S_{Micro}$  operator measures eye movements between those regions.

Although this can also happen in the desktop setting, this has not been mentioned in the original KLM. A possible explanation is the expert user assumption: users were not expected to need to look at the keyboard at all and therefore the time was incorporated into the *Keystroke* operator. This is different on mobile phones since the mapping of the keys is considerably more complex. Even experienced users tend to spend some time to confirm their input. Thus, the *Micro Attention Shift* operator allows a much more fine grained control over user interaction. It can also model uncertainty when, e.g., entering critical data like credit card numbers.

#### *Distraction (X)*

Since interactions with mobile phones take place in the real world, people are likely to be distracted from their main task by approaching people, passing cars, conversations, etc. This is accounted for by the *Distraction* operator. In contrast to all other operators, distraction is modelled as a multiplicative factor modifying the times of other operators.

#### *Action (A(t))*

This general operator models the time needed to execute a certain complex action with the phone that cannot sensibly be subdivided into smaller tasks and modelled with a combination of other operators. Possible actions include touching RFID tags, or focus to take a picture of a marker or other objects. The time for this operator highly depends on the type of action and, similar to the response time operator  $R(t)$ , this must be input to the model (indicated by the  $(t)$  notation). We give values for some typical actions.

#### *Gesture (G)*

This operator models the time needed when using a system that recognises mobile phone gestures like rotating, shaking, or drawing numbers in the air.

#### *Finger Movement (F)*

This operator models the time needed for a user to move a finger from one place (especially key or button) to another one on the device. It will in most models be subsumed in the *Keystroke* operator but allows designers a more fine-grained modelling, e.g., for predicted repeated key presses.

#### *Initial Act (I)*

In the KLM for the desktop, it is generally assumed that users are already sitting in front of their keyboard, mouse and monitor, ready to initiate the next task. The phone introduces a completely different setting since people have to carry out some preparations (e.g., locating it in a bag) before being able to use it in most circumstances. The value depends on whether the interaction was initiated by the user or externally, e.g., by an incoming call.

### **Adapted Operators**

#### *Keystroke or Button Press (K)*

Card et al. [7] originally define the *Keystroke* operator  $K$  to be the average time needed to push a button. It is to be measured by dividing the time needed for a longer sequence

of button presses by the number of these presses. Even though KLM is targeted at expert users, immediate corrections of incorrectly pressed buttons (e.g., by hitting backspace) have explicitly been allowed and incorporated.

There are 4 factors influencing the value of  $K$  in our setting. Distances between buttons are much smaller on a phone than on a standard keyboard which removes the need for head and larger eye movements and indicates a smaller value for  $K$ . However, buttons are in general harder to spot and to press and people use only one or two fingers to type (in contrast to up to ten fingers for keyboard input). Finally, all but the most experienced users check and validate their input at some points needing some *Micro Attention Shifts*. The last three aspects suggest a higher value.

For text input, we concentrate on multi-tap which, based on figures presented from industry in a panel at MobileHCI 2006, is still used by about every second user. In addition, multi-tap proves useful for comparisons with previous research. Variants like T9, which can also be easily modelled by KLM, are often seen as too complex and do not work well for names or addresses.

#### *Pointing (P)*

*Pointing* has originally been defined to model the time used to move a cursor to a target area using the mouse. This is in general not applicable for mobile phone applications except in rare applications in which a cursor can be controlled using the joystick or special buttons. Such interactions can be modelled using appropriate *Keystrokes* since they are not based on Fitts' Law as is the original interpretation of  $P$ .

For larger screens or handheld devices using stylus input, we refer to Luo and John [24] who updated the values for *Pointing* for touch and stylus use. In our context, this models the time needed to move the phone from one place to another possibly to perform some *Action* at that point (which itself is not included in the pointing action and has to be represented by an  $A$  following the  $P$ ). This operation is similarly based on Fitts' Law as the original operator.

#### *Homing (H)*

In the original KLM, this modelled the movement of the hand from the keyboard to the mouse or back. For mobile phone interactions, this is not relevant. However, the action of moving the phone from a position where one can read the screen to one's ear or back is an analogous motion and similarly important. Therefore, we use the *Homing* operator whenever the user changes from listening and speaking to reading the screen or vice versa. In this setting  $H$  can be expected to be somewhat smaller but close to *Pointing P*.

### **Unchanged Operators**

#### *Mental Act (M)*

The Mental Act operator "is based on the fact that when reasonably experienced users are engaged in routine operation of a computer, there are pauses in the stream of actions that are about a second long and that are associated

with routine acts such as remembering a filename or finding something on the screen” (Kieras [21]). It can be adopted as defined, and existing usage guidelines, e.g. from [21], can be used. Since we use new operators, we give additional guidelines in a later section on the *Mental Act* operator.

Most studies adopted the original choice of  $M = 1.35$  seconds for their applications (e.g., [9,15,18]). [26] and [28] propose a smaller value of 0.38 and 0.57 seconds. However, these values were taken from much specialised applications. Myung [28], e.g., examined Korean text input only. For general settings, a higher average value can be assumed. Larger values than the original value are reported in Manes et al. [25] which mainly result from studying an explicit scenario (a car navigation system) and users who do not have the routine of expert users assumed by KLM. Current cognitive architectures like ACT-R [2] confirm the original value. We also found no evidence to justify a change.

#### Response Time ( $R(t)$ or $W(t)$ )

The *Response Time* operator models the time the system needs to react to user input as long as it blocks the user from executing further actions. It can be adopted as defined and must be input to the model (hence the  $(t)$  notation) since it is highly variable and dependent on the application.

#### Not Applicable Operator

##### Drawing ( $D(n_D, l_D)$ )

The *Drawing* operator models manual drawings of  $n_D$  straight line segments with a total length of  $l_D$  cm with the mouse. This is not applicable in our setting. However, this might change with possible applications for drag and drop operations done with the phone in some future systems.

### USER STUDIES FOR TIME MEASUREMENTS

To be able to use the model in practice and to predict the time required for certain complex tasks based on the model, the duration of a single application of an operator must be known. In 7 studies we acquired data to estimate the times. We recruited volunteers of various backgrounds (about 50% students) on a study by study basis (between 9 and 19 participants per study) and did not see any differences caused by gender with, altogether, 41% female participants.

Before conducting each of the studies, questionnaires were given to the users to clarify their experience with mobile phones in general and more specifically the mobile phone interaction technique under observation. We also aimed to adhere to the expert user assumption by running one or several training sessions with each user. Participants had to repeat the same or similar tasks until they and we were confident that they will to make only minimal errors. Erroneous trials were discarded. All but one studies were executed in various, every day, non-laboratory situations.

#### Initial Act ( $I$ ), Homing ( $H$ )

Without preparation or creating specific situation, we observed people in everyday settings receiving and answering phone calls and specifically asked them to pull

out their mobile phone and execute a phone call. We taped all these actions and extracted timing information from 11 people, aged 25-54 with an average of 34.6 years, 4 female, all used to standard mobile phone interaction.

The found value for the *Initial Act* operator depends on whether the interaction was initiated by the users themselves (leading to a median value of  $I = 5.32$  seconds) or externally, e.g., by an incoming call (median  $I = 3.89$  seconds)<sup>1</sup>. Those values are highly diverse, however, since people have extremely different ways to store the phone (trouser pocket, handbag, pocket attached to the belt). A best-case study in which the phone was placed in front of the users on the table who initiate the action themselves or expect a call gives a median of  $I = 1.18$  seconds. Thus, if no assumptions can be made, we suggest an average value of  $I = 4.61$  seconds. For repeated or expected interaction the  $I = 1.18$  seconds estimate should be used.

In the same study we measured times needed to switch from a phone position where the screen can be read to one close to the ear and back (*Homing*). The times of all people under observation were very similar and we extracted a median of  $H = 0.95$  seconds. As expected this is only slightly smaller than the found value of *Pointing*  $P = 1.00$  second described below. To model the fact that people have to refocus on the phone’s screen and continue their interrupted action, we strongly suggest that a *Mental Act* operator be placed after a *Homing* away from the ear as specified in the heuristics given in the section about the *Mental Act* operator.

#### Pointing ( $P$ ), Action ( $A(t)$ )

To measure execution times for *Pointing* and *Action*, we needed an application where such interactions occur quite often. In some countries like Japan, visual markers and near field communication (NFC) are already very wide-spread technologies in the public [6,11]. This is not yet the case in Europe. Therefore, in conjunction with other projects run in our lab (e.g., [34]), we prepared a movie poster acting as user interface for several interaction methods. Users can select and use different services by, e.g., touching NFC tags or taking pictures of visual markers. We asked users to follow the brief instructions on the poster and let them buy tickets for their favourite movies in a theatre close to them. From the videotaped footage we were able to extract timing measurements regarding the movement ( $P$ ) and alignment ( $A(t)$ ) of the phone to the NFC tag, and the approach ( $P$ ) and focus ( $A(t)$ ) of the phone to take a picture of a marker.

The user study was carried out with 9 persons, aged 22-46, with an average of 28.6 years, 2 female. From a set of 64 error free video taped actions, we deduced *Pointing*  $P = 1.00$  second. The 37 NFC interactions showed that aiming at the NFC tag itself did not need any separate action besides the phone movement and we define  $A_{NFC} = 0$ .

<sup>1</sup> If not explicitly stated otherwise, we present the median of the measured values and list 1<sup>st</sup> and 3<sup>rd</sup> quartiles in Table 1.



The remaining 27 photographs of visual markers led to a value of  $A_{picture} = 1.23$  seconds for correct positioning and focussing (note that this does not include *Pointing*).

Note also that the time needed by the system to recognise the tag or interpret the marker and initiate the appropriate action is not included in the *Pointing* or *Action* operator but must be modelled by with the *Response Time* operator  $R(t)$ .

#### Macro Attention Shift ( $S_{Macro}$ )

Using a careful frame-by-frame manual analysis of the video tapes from the study presented in the last section, we counted the number and determined the duration of head and eye movements that indicate an attention switch from the phone to the poster and vice versa. We extracted a total of 121 attention shifts. The times of the shifts in one direction do not differ significantly at all from those in the other direction ( $t=0.57$ ,  $p>0.56$ ). Thus, we propose a common value of  $S_{Macro} = 0.36$  seconds.

#### Micro Attention Shift ( $S_{Micro}$ )

Mobile phones in general suggest a split into three regions: display, hot keys, and keypad (Figure 2). Finding out when and to which section people looked proved to be infeasible with conventional video taping. Therefore we used an eye gaze tracker from Eye Response Technologies that samples images with a sufficient rate of 60Hz. The participants had to run three pre-set tasks that included writing a text message (mainly text input), changing the ring-tone (mainly menu navigation), and setting the time of the alarm-clock (menu navigation and number input).



Figure 4: Eye gaze positions during a task, overlaid over the phone in use. Left: write a text message. Right: set alarm time.

All 10 people (aged 24-34 with an average of 27.5 years, 6 female) were allowed to use their own mobile phone and we ran several sessions to ensure error free interaction. We then automatically calculated the number and time of gaze position changes between the regions from the logged data. Figure 4 shows data overlaid on some phones. We counted more than 1500 shifts between the three regions and found the following values: display  $\leftrightarrow$  hotkeys 0.12 seconds, display  $\leftrightarrow$  keypad 0.14 seconds, and keypad  $\leftrightarrow$  hotkeys 0.04 seconds. If no distinction should or can be made between the single sections of the phone, we suggest using the median of all values of  $S_{Micro} = 0.14$  seconds.

#### Gesture ( $G$ )

To measure gesture input, we used a Samsung SGH-E760 phone with built-in acceleration sensors and a few games and standard applications that can be controlled using simple gestures (Figure 5). The times for each gesture was extracted from videos of 6 different types of gestures, each done by 10 people (aged 23-33 with an average of 26.3 years, 5 female). Since the possible gestures were quite similar in type and time, the measurements resulted in one value for all gestures and we set  $G = 0.80$  seconds.

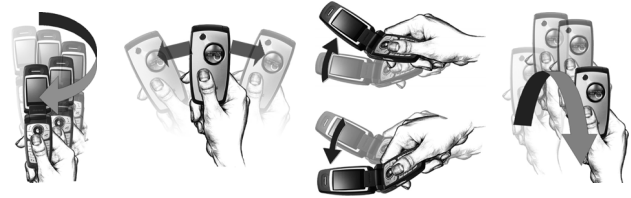


Figure 5: The Samsung SGH-E760 phone and some possible gestures recognised by the built-in acceleration sensor system.

#### Keystroke ( $K$ )

Keystrokes were measured with a small J2ME program that logs timestamps of key presses and releases into a file on the mobile phone (using the File API on a Nokia N90). We invited 19 people, aged 25-40 with an average of 27.8 years, 9 female. Each person entered two mobile phone numbers of their own choice. All of them used the wide-spread one-hand thumb entry method. During the study we observed that no errors were made.

The Nokia N90 phone features a standard 12 button keypad. The average of the whole interaction times is 4.63 seconds and the time per keystroke was calculated to be  $K = 0.39$  seconds. For the five most experienced users we got a value of  $K = 0.33$  seconds per keystroke.

Another interesting value we measured is the mere physical action of pressing and releasing a key. It was measured by the key logger to be 0.13 seconds (the most experienced users were only 10 milliseconds faster) and will be used to calculate *Finger Movement* time in the next section.

We measured standard keypad input separately from the hotkeys, although we did not take additional special hotkeys into account that can be found on several phone models on the side or top of the phone. For the hotkeys of the N90 (4 buttons and a 5 button joystick),  $K = 0.16$  seconds were measured as a median. The smaller value can be easily explained with the smaller distance between buttons and the larger average size as well as the more direct and known semantic mapping of the buttons.

The findings are close to those of other research. Mori et al. [26] for example measured 0.39 seconds. The original KLM suggests values between 0.08 and 1.20 with 0.28 seconds for a user with average routine on a standard sized desktop QWERTY keyboard. An average value for typing random characters is also mentioned. This better resembles text input on mobile phones. The suggested value of 0.50 seconds again comes quite close to our estimate.

These values are meant for individual button presses or number input only. Several projects already verified and improved Keystroke-Level Modelling of more complex variants of text entry. The results are quite diverse: Dunlop and Crossan [9] predict a value of 2.01 and 1.84 seconds on average for multi-tap and predictive text input, respectively. How and Kan [17] specify 1.32 and 1.00 for the same techniques, assuming an average SMS length of 60 characters. Silfverberg et al. [36] also examined multi-tap and predictive text input and give values of 0.57 and 0.30 seconds for optimal expert use. The comparatively very small values result from only modelling the pointing component with the help of Fitts' Law, neglecting the time needed to find and actually press the buttons as well as verification. Pavlovych and Stuerzlinger [29] calculate values ranging from 2.04 to 1.58 seconds for different input methods and suggest how those times should be adjusted for different states of routine.

These results might indicate that KLM does not work too well in this respect. However, James and Reischel show in [18] that although the predicted times can differ from actual performance times, relative relations between different designs prove to be correct and significant. Because of the rich set of publications in that area, we have not conducted detailed studies for text entry. Some values were taken from the study for the *Distraction* operator described later.

#### Finger Movement ( $F$ )

From our observations during the tests for the *Keystroke* parameter we can report that most users verified less than every second number they typed. This means that the average total time needed to enter a 11 digit number was actually composed of 11 physical key presses, on average 4 *Micro Attention Shifts*, and 10 *Finger Movements*. Since we know the values of the other operators, we can calculate  $F$  to be 0.24 seconds for all but the 5 quickest users. According to our experience, full experts tend to only check their typing once during writing. Modelling that behaviour for the 5 quickest users results in the median value of  $F=0.22$ . To additionally verify those assumptions, we ran an extra 10 tests using a mobile phone with a blinded display eliminating the use of *Micro Attention Shifts*. The upshot of this study was a median of  $F=0.23$  seconds. These results from the 323 key strokes done in the tests make it a very stable parameter. Figure 6 shows movement paths of three sample phone numbers types in the tests.

The value is also very close to what others like Silfverberg et al. [36] found who measured 0.27 seconds with the thumb and 0.31 seconds using the slower two-handed index finger input. In [26], Mori et al. specify 0.19 seconds for  $F$ .

When movements occur in the hotkey region *only* (as is the case for menu navigation sequences and when starting an application),  $F$  is smaller. Our studies indicate that the time drops to  $F=0.16$  seconds on average. Depending on the interaction, designers can choose which value fits better or use an average according to an assumed ratio of key uses.



Figure 6: Approximate finger movements ( $F$ ) occurred while typing three different phone numbers on the Nokia N90.

#### General Parameters

Some parameters cannot be measured in a single specific setting. The system response time, for example, differs strongly depending on the phone model, the application running on it, and the action invoked. Also, the influence of mental preparation and the appropriate placement of the *Mental Act* operator has always been a complex issue in KLM models. Kieras [21] gives several suggestions and heuristics specifying where and in what quantity the operator should be placed that also apply to the model as used in this paper. Another parameter that belongs to the same category is the new *Distraction* operator  $D$ . It has not been treated in previous research on task models but we found that it has a considerable impact on time performance and there is a whole set of applications especially in the area of mobile interactions that are influenced by distractive and disruptive factors.

#### Response Time ( $R(t)$ or $W(t)$ )

As already discussed, information on system response times is supposed to be input to the model since these are highly diverse. We can, however, support the assumption of Silfverberg, MacKenzie and Korhonen [36] that key presses in general have immediate feedback. Menu browsing and selection was also running in negligible amounts of time on the phones we investigated. Starting applications needed anything between 0 and 6 seconds. For our setting, we only explicitly give values for the special cases when tags are detected (NFC) or pictures taken.

Our 10 tests with a Nokia 3220 with a built-in NFC reader showed that the processing of a tag takes on average  $R_{NFC} = 2.58$  seconds. Using a Nokia N71, measurements for visual marker processing resulted in  $R_{marker} = 2.22$  seconds.

#### Distraction ( $X$ )

Interactions in the real world have to take into consideration various events that divert the concentration on the task at hand. This includes approaching people, passing cars, traffic lights, conversations, etc. Situations in which it is known that such distractions occur frequently (like at a bus stop) can be modelled with the *Distraction* operator.

People cope with such situations in different ways. They use their peripheral view, make quick glances, or introduce pauses. Initial tests showed that the behaviour also depends on the type of task. Thus, distraction can not be easily modelled as certain specific actions. Through our tests we found that it is more appropriate to model distraction as a multiplicative factor rather than an additive operator.

Although the type and consequences of distractions can be manifold, several studies of distraction and multi-tasking, e.g., by D. D. Salvucci [35], proved feasible in cognitive modelling. We give a simplified and rough but nevertheless justified idea how a task is probably slowed down by common side activities.

We ran three experiments, each with the same 10 people, aged 24-33, with an average of 26.7 years, 3 female. Subjects had to write a short message (about 90 characters) on their own phone in 3 different settings: a silent room (experiment 1), standing on a busy street (experiment 2) and walking along and crossing that street (experiment 3). To obviate the possibility of sequence effects, we varied the order in which the participants conducted those trials. They also freely chose the contents of the message to write.

Typing speed of the participants varied considerably (0.76 to 3.49 characters per second in experiment 1). However, relative changes in the performance of each user are quite consistent. Experiments 2 and 3 revealed in an analysis of variance that the expected increase in time demand for distracted tasks is relevant well beyond the 5% level ( $t=2.23$ ,  $p<0.03$  and  $t=3.28$ ,  $p<0.01$ ). The results suggest to add  $X_{Slight} = 6\%$  of the modelled time to the whole interaction if there is an anticipated slight distraction and  $X_{Strong} = 21\%$  for distractions that force persons to deviate from their task in a more rigorous or regular fashion.

#### Mental Act (M)

As said earlier, we found no reason to change the value of  $M = 1.35$  seconds from the original KLM. However, since we added and slightly changed the interpretation of some operators, we update the heuristics from Kieras [21] to place  $M$ 's: Use rule 0 to place  $M$ 's and then cycle through rules 1 to 5 for each  $M$  to see whether it should be deleted.

- Rule 0** Place  $M$ 's in front of all  $K$ 's,  $H$ 's,  $S_{Macro}$ 's and  $G$ 's.
- Rule 1** If an operator following an  $M$  is anticipated in the operator before  $M$ , delete the  $M$  (e.g.,  $PMK$  becomes  $PK$ ).
- Rule 2** If a string of  $M$ 's belongs to a cognitive unit (e.g., writing a known number), then delete all  $M$ 's but the first.
- Rule 3** If a  $K$  is a redundant terminator (e.g., the selection key for entering submenus), then delete the  $M$  in front of it.
- Rule 4** Delete the  $M$  in front of a  $H$  which describes the movement from the reading to the listening position.
- Rule 5** If unsure, emphasise more the number than the placement of the occurrences of the  $M$  operator.

#### PARAMETER VALUES OVERVIEW

Table 1 shows the results of the studies. The median of each operator is given in the "Time" column. If applicable, the other two columns contain the 1<sup>st</sup> and 3<sup>rd</sup> quartiles, respectively. All times are specified in seconds.

#### EVALUATION

To validate the values we found for the mobile phone KLM we set up a scenario, modelled it with our parameters and compared it to actual, observed timing data.

Operator		Time	Quartile 1	Quartile 3
<b>A, Action</b>	picture/marker	<b>1.23</b>	0.61	1.44
	NFC	<b>0.00</b>	-	-
	in general	<i>variable, input to model</i>		
<i>D, Drawing</i>		<i>not applicable</i>		
<b>F, Finger Movement</b>		<b>0.23</b>	0.20	0.29
<b>G, Gestures</b>		<b>0.80</b>	0.73	0.87
<i>H, Homing</i>		<b>0.95</b>	0.81	1.00
<b>I, Initial Act</b>	externally	<b>5.32</b>	3.98	7.51
	internally	<b>3.89</b>	2.23	4.89
	optimal setting	<b>1.18</b>	1.10	1.26
	no assumptions	<b>4.61</b>	-	-
<b>K, Keystroke</b>	keypad average	<b>0.39</b>	0.37	0.48
	keypad quick	<b>0.33</b>	0.32	0.37
	hot Key	<b>0.16</b>	0.15	0.20
<i>M, Mental Act</i>		<b>1.35</b>	-	-
<i>P, Pointing</i>		<b>1.00</b>	0.84	1.20
<b>R, System Response Time</b>	NFC	<b>2.58</b>	2.46	2.80
	visual marker	<b>2.22</b>	2.09	2.82
	general	<i>variable, input to model</i>		
<b>S<sub>Macro</sub>, Macro Attention Shift</b>		<b>0.36</b>	0.28	0.44
<b>S<sub>Micro</sub>, Micro Attention Shift</b>	keypad ↔ display	<b>0.14</b>	0.14	0.19
	hotkey ↔ display	<b>0.12</b>	0.02	0.14
	keypad ↔ hotkey	<b>0.04</b>	0.02	0.12
	in general	<b>0.14</b>	0.10	0.16
<b>X, Distraction</b>	slight	<b>6 %</b>	3 %	13 %
	strong	<b>21 %</b>	11 %	25 %

**Table 1: Overview of the proposed times for all operators. Names of newly defined operators are set in bold.**

The scenario was based on a ticket service for public transportation in Munich, Germany. The tasks included the download of a service from a poster augmented with NFC tags. The interaction device was a Nokia 3220 with built-in NFC reader. People had to order a ticket to a pre-defined target for three persons and write a text message to a friend about the expected time of arrival. In order to illustrate the applicability of the modelling approach, two different ways of accomplishing the main task were implemented: Using a form input on the phone's web browser, and using the NFC capabilities of the phone.

The participants were either routinely working with the used technology or trained before the study. None of them had taken part in any of the earlier studies.

#### KLM Prediction

KLM predicts 122.77 seconds for the first variant of the scenario using direct input and browsing with a total of 110 operators. The model of the NFC version of the scenario uses 198 operators and predicts 174.84 seconds. Distractions were neither observed nor modelled. Table 2 shows some excerpts of the latter model. The prediction was calculated independently of and before the validation based on a detailed analysis of the scenario and the heuristics given in an earlier section.



Pick up the mobile phone	<i>I</i>	1.18 sec.
Enter main menu	<i>M, H</i> [Hotkey]	1.35 sec. + 0.16 sec.
Go to 'Programs'	<i>M, 3H</i> [Hotkey]	1.35 sec. + 3*0.16 sec.
Select 'Programs'	<i>K</i> [Hotkey]	0.16 sec.
Go to 'Collection'	<i>M, H</i> [Hotkey]	1.35 sec. + 0.16 sec.
Select 'Collection'	<i>K</i> [Hotkey]	0.16 sec.
Select 'Choose program'	<i>K</i> [Hotkey]	0.16 sec.
...	...	...
Process tag	<i>R<sub>NFC</sub></i>	2.58 sec.
Find next tag	<i>M</i>	1.35 sec.
Movement to tag 'To'	<i>P</i>	1.00 sec.
NFC interaction	<i>A<sub>NFC</sub></i>	0.00 sec.
Process tag	<i>R<sub>NFC</sub></i>	2.58 sec.
Find next tag	<i>M</i>	1.35 sec.
...	...	...
	Sum:	174.84 sec.

**Table 2: Selected sequences from the proposed mobile phone Keystroke-Level Model of a scenario based on NFC tags.**

### Empirical Validation

Both alternatives of the scenario have been completed by 9 people (aged 23-34 with an average of 27.6 years, 3 female). The times needed for the first version ranged from 106 to 133 seconds with an average of 117 seconds ( $\sigma=9.38$ ). The values are remarkably close to the predicted value of roughly 123 seconds. The upshots of the second, NFC version of the scenario are similar in magnitude: The average duration of the task was 170 seconds (times ranged between 147 and 190 seconds,  $\sigma=13.72$ ) which is also very close to the KLM estimate of 175 seconds.

This means that the deviations of the KLM predictions to each data sample are in the small range of -15% to +8%. The measured averages actually deviate only 5% and 3%, respectively. Furthermore, the speed loss of 30% of the NFC implementation predicted by the KLM is confirmed by the study with a measured average decrease of 31%.

### SUMMARY AND DISCUSSION

We introduced a Keystroke-Level Model (KLM) that can be applied to most interactions currently available for mobile phones including advanced interactions involving identification tags or gestures. We described considerations and several user tests to get sound estimates of time values for the operators of that model. We also presented detailed explanations and guidelines for the use of these operators to enable application designers and teams to quickly model such interactions and compare different designs ahead of any implementation efforts.

It is evident that average users handle complex interaction styles differently and with different speed. It can also be hard to get into a routine for tasks that are new to a user even after several repetitions. This may render the expert user assumption difficult to support. The complexity of the interactions adds to this problem. However, our experience and evaluation show that for a set of interaction methods known to its users through at least sporadic use, estimates given by the mobile phone KLM are very good indeed. Especially when target users are young and 'hip' people or professional workers, it is very likely that these learn and adapt quickly and reach a state of experience that can be modelled close enough to make sound predictions.

We presented models of two different implementations of a real world scenario that also indicate that well grounded design decisions can be reached purely based on the model predictions. Nevertheless, we strongly encourage other researchers to expand our initial set of studies and examine our results through additional measurements and scenarios.

The collection of introduced operators is necessary to apply KLM to interactions such as those described in this paper. According to the experiences in our lab and after reviewing relevant publications, we conclude that this set also suffices to capture the interactions possible with mobile phones at the current state of the art. Of course, time will inevitably bring different and additional types of interactions in the future for which new operators might have to be defined. Others might need adjustments when new or radically more time consuming variants are introduced (like complex gestures). It can also happen that some interactions become considerably easier (for example by getting rid of the need to accurately aim and focus for visual marker recognition). Nevertheless, after having conducted our tests, we are very positive that those changes are easy to integrate into the model and predictions can be made that lead to an early and valuable basis for design decisions.

### REFERENCES

1. Ailisto, H., Korhonen, I., Plomp, J., Pohjanheimo, L., Strömmer, E., Realising Physical Selection for Mobile Devices. In *Proc. PI'03*. 38-41. 2003
2. Anderson, J. R., Lebiere, C., The Atomic Components of Thought. Lawrence Erlbaum Associates, Inc. 1998
3. Ballagas, R., Borchers, J., Rohs, M., Sheridan, J.G., The Smart Phone: A Ubiquitous Input Device. *IEEE Pervasive Computing* 5(1). 70-77. 2006
4. Ballagas, R., Rohs, M., Sheridan, J. G., Sweep and Point and Shoot: Phocam-based Interactions for Large Public Displays. In *Extended Abstracts CHI'05*. ACM Press. 1200-1203. 2005
5. Bälter, O., Keystroke Level Analysis of Email Message Organization. *Proc. CHI'00*. ACM Press. 105-112. 2000
6. Boyd J., Here Comes the Wallet Phone, *IEEE Spectrum* 42(11). 2005
7. Card, S. K., Moran, T. P., Newell, A., The Keystroke-Level Model for User Performance Time with Interactive Systems. *Comm. ACM* 23, 7. 396-410. 1980
8. Card, S. K., Newell, A., Moran, T. P., The Psychology of Human-Computer Interaction. Lawrence Erlbaum Associates, Inc. 1983
9. Dunlop, M.D., Crossan, A., Predictive Text Entry Methods for Mobile Phones. *Personal Technologies*, 4(2-3), 2000
10. Föckler, P., Zeidler, T., Brombach, B., Bruns, E., Bimber, O., PhoneGuide: Museum Guidance Supported by On-device Object Recognition on Mobile Phones. In *Proc. MUM'05*. 2005

11. Fowler, G. A., QR Codes: In Japan, Billboards Take Code-crazy Ads to New Heights. *Wall Street Journal* 10.10.2005. 2005
12. Gong, R., Elkerton, J., Designing Minimal Documentation Using a GOMS Model: a Usability Evaluation of an Engineering Approach. In *Proc. CHI'90*. ACM Press. 99-107. 1990
13. Gong, R., Kieras, D., A Validation of the GOMS Model Methodology in the Development of a Specialized, Commercial Software Application. In *Proc. CHI'94*. ACM Press. 351-357. 1994
14. Gray, W. D., John, B. E., Atwood, M. E. The Precipitous Project Ernestine or an Overview of a Validation of GOMS. In *Proc. CHI'92*. ACM Press. 307-312
15. Haunold, P., Kuhn W., A Keystroke Level Analysis of a Graphics Application: Manual Map Digitizing. In *Proc. CHI'94*. 337-343. 1994
16. Hinckley, K., Guimbretière, F., Baudisch, P., Sarin, R., Agrawala, M., Cutrell, E., The Springboard: Multiple Modes in one Spring-loaded Control. In *Proc. CHI'06*. ACM Press. 181-190. 2006
17. How Y., Kan M.Y., Optimizing Predictive Text Entry for Short Message Service on Mobile Phones. In *Proc. HCI'05*. 2005
18. James, C. L., Reischel, K. M., Text Input for Mobile Devices: Comparing Model Prediction to Actual Performance. In *Proc. CHI'01*, 365-371. 2001
19. John, B. E., Extensions of GOMS Analyses to Expert Performance Requiring Perception of Dynamic Visual and Auditory Information. In *Proc. CHI'90*. ACM Press. 107-115. 1990
20. John, B. E., Vera, A. H., A GOMS Analysis of a Graphic Machine-paced, Highly Interactive Task. In *Proc. CHI'92*. ACM Press. 251-258. 1992
21. Kieras, D., Using the Keystroke-Level Model to Estimate Execution Times. The University of Michigan, *Unpublished Report, Online Version as of August 2006*, <http://www.pitt.edu/~cmlewis/KSM.pdf>. 1993
22. Kindberg, T., Barton, J., Morgan, J., Becker, G., Caswell, D., Debaty, P., Gopal, G., Frid, M., Krishnan, V., Morris, H., Schettino, J., Serra, B., Spasojevic, M., People, Places, Things: Web Presence for the Real World. *Mobile Networks and Applications* 7(5). 365-376. 2002
23. Koester, H. H., Levine, S. P., Validation of a Keystroke-Level Model for a Text Entry System Used by People with Disabilities. In *Proc. Assets '94*. ACM Press. 115-122. 1994
24. Luo, L., John, B. E., Predicting Task Execution Time on Handheld Devices Using the Keystroke-Level Model. In *CHI'05*. ACM Press. 1605-1608. 2005
25. Manes, D., Green, P., Hunter, D., Prediction of Destination Entry and Retrieval Times Using Keystroke-Level Models. *Technical Report UMTRI-96-37*. The University of Michigan Transportation. 1996
26. Mori, R., Matsunobe, T., Yamaoka, T., A Task Operation Prediction Time Computation Based on GOMS-KLM Improved for the Cellular Phone and the Verification of that Validity, In *Proc. ADC'03*. 2003
27. Myers, B. A., Mobile Devices for Control. In *LNCS 2411*. Springer. 1-8. 2002
28. Myung R., Keystroke-Level Analysis of Korean Text Entry Methods on Mobile Phones, *International Journal of Human-Computer Studies* 60, 5-6. 545-563. 2004
29. Pavlovych, A., Stuerzlinger, W., Model for Non-expert Text Entry Speed on 12-button Phone Keypads. In *Proc. CHI'04*. ACM Press. 351-358. 2004
30. Rekimoto, J., Nagao, K., The World Through the Computer: Computer Augmented Interaction with Real World Environments. In *Proc. UIST'95*. 29-36. 1995
31. Rohs, M., and Gfeller, B., Using Camera-equipped Mobile Phones for Interacting with Real-world Objects. *Advances in Pervasive Computing*. Austrian Computer Society (OCG). 265-271. 2004
32. Rukzio, E., Leichtenstern, K., Callaghan, V., Schmidt, A., Holleis, P., Chin J., An Experimental Comparison of Physical Mobile Interaction Techniques: Touching, Pointing and Scanning. In *Proc. Ubicomp'06*. 2006
33. Rukzio, E., Paolucci, M., Wagner, M., Berndt, H. H., Hamard, J., Schmidt, A., Mobile Service Interaction with the Web of Things. In *Proc. ICT'06*. 2006
34. Rukzio, E., Schmidt, A., Hussmann, H., Physical Posters as Gateways to Context-aware Services for Mobile Devices. In *Proc. WMCSA'04*. 2004
35. Salvucci, D. D. Modeling Driver Distraction from Cognitive Tasks. In *Proc. CogSci2002*, 792-797. Lawrence Erlbaum Associates, Inc. 2002
36. Silfverberg M., MacKenzie, I. S., and Korhonen, P. Predicting Text Entry Speed on Mobile Phones. In *Proc. CHI'00*, 9-16. 2000
37. Teo, L., John, B. E., Comparisons of Keystroke-Level Model Predictions to Observed Data. In *Extended Abstracts CHI'06*. ACM Press. 1421-1426. 2006
38. Välikkynen, P., Korhonen, I., Plomp, J., Tuomisto, T., Cluitmans, L., Ailisto, H., Seppä, H., A User Interaction Paradigm for Physical Browsing and Near-object Control Based on Tags. In *Proc. PI'03*. 38-41. 2003
39. Välikkynen, P., Tuomisto, T., Physical Browsing Research. In *Proc. PERMID'05*. 35-38. 2005
40. Want, R., Fishkin, K. P., Gujar, A., Harrison, B. L., Bridging Physical and Virtual Worlds with Electronic Tags. In *Proc. CHI'99*. ACM Press. 370-377. 1999