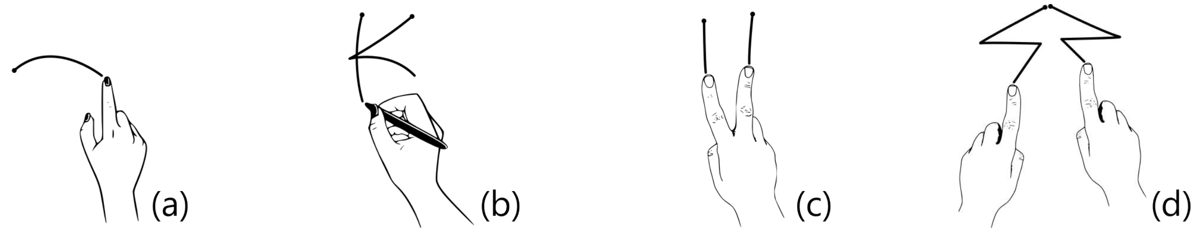
GATO: Predicting Human Performance with Multistroke and Multitouch Gesture Input

Luis A. Leiva
Sciling, SL
Valencia, Spain
name@sciling.com

Daniel Martín-Albo
Independent researcher
Barcelona, Spain
dmartinalbos@gmail.com

Radu-Daniel Vatavu
MintViz Lab | MANSiD
University Ştefan cel Mare of Suceava, Romania
vatavu@eed.usv.ro



True time: 438 ms
Our estimation: [436, 442] ms

True time: 1630 ms
Our estimation: [1642, 1725] ms

True time: 662 ms
Our estimation: [648, 662] ms

True time: 1461 ms
Our estimation: [1459, 1484] ms

Figure 1. Examples of articulation patterns representative for pen and touch stroke gesture input: unistrokes (a), multistroke gestures (b), multitouch (c), and bimanual input (d). GATO advances the state-of-the-art in predicting human performance with gesture input, currently limited to unistrokes (a), by providing accurate user-independent estimations of multistroke and multitouch gesture production times for all these articulation patterns and more.

ABSTRACT
We introduce GATO, a human performance analysis technique grounded in the Kinematic Theory that delivers accurate predictions for the expected user production time of stroke gestures of all kinds: unistrokes, multistrokes, multitouch, or combinations thereof. Our experimental results obtained on several public datasets (82 distinct gesture types, 123 participants, ≈36k gesture samples) show that GATO predicts user-independent gesture production times that correlate $r_s > .9$ with groundtruth, while delivering an average relative error of less than 10% with respect to actual measured times. With its accurate estimations of users' *a priori* time performance with stroke gesture input, GATO will help researchers to understand better users' gesture articulation patterns on touchscreen devices of all kinds. GATO will also benefit practitioners to inform highly effective gesture set designs.

ACM Classification Keywords
H.5.2 Information Interfaces and Presentation: User Interfaces; I.5.2 Pattern Recognition: Design Methodology

Author Keywords
Gesture Input; Stroke Gestures; Touch Gestures; Human Performance; Production Time; Kinematic Theory.

MobileHCI '18, September 3–6, 2018, Barcelona, Spain

ISBN 978-1-4503-5898-9/18/09…$15.00

DOI: https://doi.org/10.1145/3229434.3229478

INTRODUCTION
Stroke gestures, in the form of flicks, swipes, letters, digits, and symbols drawn on touchscreens with either the finger or a stylus, are prevalent on mobile devices, enabling users to perform a variety of tasks fast and confidently [48]. For example, stroke gestures are employed to enter text quickly [28,49,69], call app functions directly [6,54,71], or get access to content much faster than by navigating the menus of the graphical user interface [34,35]. Moreover, gesture input is among the few modalities for people with motor or visual impairments to employ mobile touchscreen devices effectively [27,57,67], e.g., the Apple VoiceOver and Google TalkBack gesture sets were specifically designed to assist users with visual impairments to benefit from mobile smart technology [41,68].

Stroke gesture input has therefore been adopted on mobile devices for reasons of efficiency: a quick swipe on the screen executes a command directly or gets users instantaneous access to content. The efficiency of human performance with generic input has been evaluated in the community with *task times*, the best known and longstanding example being Fitts' law [17]. In the case of gesture input, the *production time* of a gesture, i.e., how long it takes users to produce a path on a touchscreen, is therefore an essential aspect of user performance [70] that has been used to inform gesture design [6,12,14] and derive key insights on how difficult stroke gestures are to articulate [52,61].

In this context, it is important for gesture user interface designers to estimate, as accurately as possible, users' performance with stroke gesture input. If such insightful information were available during the early stages of design, it would represent a valuable asset for practitioners, enabling them to explore various gesture set designs with minimum effort (i.e., no experiments or user studies required) to find out which gestures are faster [6,14] or easier to articulate [52,61]. Such valuable information can only be obtained by having access to models of user performance with stroke gesture input, such as CLC [12] and KeyTime [32], that were introduced to help the community model and predict the production times of unistroke gestures (see Figure 1a). Unfortunately, such models do not exist for gestures that are more complex than a mere stroke (such as those illustrated in Figures 1b, 1c, and 1d), leaving designers with no information about their users' *a priori* time performance with multistroke or multitouch input.

Our contributions in this work are as follows:

1. We introduce GATO (the Gesture Articulation Time predictOr), a new human performance analysis and estimation technique that predicts user performance with stroke gesture input of all kinds: unistrokes, multistrokes, multitouch, or combinations thereof. GATO is based on theoretical and empirical evidence from the Kinematic Theory regarding the production of human movement as the effect of multiple coupled neuromuscular subsystems [44,45].
2. We evaluate GATO on 6 public datasets consisting of roughly 36k samples of 82 distinct gesture types collected from 123 participants. Our experimental results show that GATO predicts user-independent gesture production times that correlate $r > .9$ (and up to $r_s = .99$) with groundtruth, while delivering an average relative error of 10% with respect to actual measured times.
3. We release an online application that practitioners can readily use to estimate accurate, user-independent production times of multistroke and multitouch gestures by providing just one gesture example. Moreover, to assist researchers in their analyses of large gesture datasets, we also release a RESTful JSON API, which enables easy integration of GATO with third party applications over the web.
4. During our design and implementation of GATO, we had to reconsider how multistroke and multitouch gestures have been defined and represented in the community, and found those representations unsatisfactory for our purpose. Thus, we also contribute with a new perspective of representing multistroke gestures as *a sequence of strokes performed both on the touchscreen and in air*, which we believe to be useful for other gesture investigations as well.

MOTIVATING EXAMPLES WITH GATO

In the following, we present two examples for GATO to show how easy it is for designers to get predictions of users' production times with our online web application.

Informing gesture design

Imagine a designer who wants to decide which of three stroke gesture types to use for launching the Twitter app on a smartphone. Knowing that the task will be executed frequently, the gesture command should be easy to remember, fast, and easy to perform. The options are: (a) the "reply" symbolic icon from the Twitter user interface; (b) uppercase letter T, which refers to the app name and, thus, acts as a mnemonic shortcut [54]; and (c) the hashtag symbol #, commonly employed

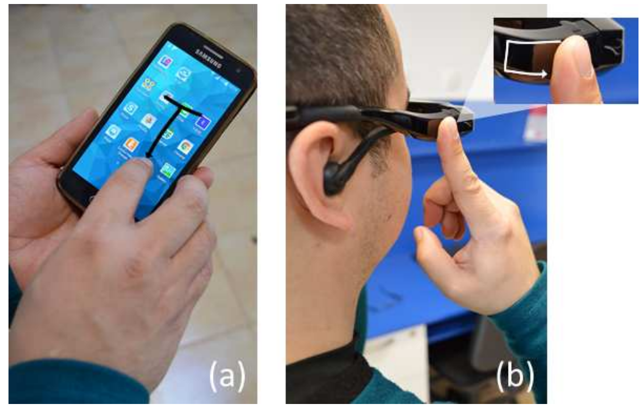


Figure 2. GATO is a generic technique that predicts the production time of any stroke gesture produced on any touch input device, such as the letter "T" performed on a smartphone (a) or the small rectangle drawn on the side touchpad of a pair of smart glasses (b). These are mockup examples to support the discussion of our two use cases.

for user-generated tagging on microblogging services and, thus, easily memorable due to association [43]; see Figure 2a. All the three gesture candidates are easy to remember due to intuitive associations, so the remaining question is how fast they will be produced by users. The reply icon can be produced with a single stroke, letter T is always produced with two strokes, while the four-stroke hashtag # can be produced with one finger drawing each of the four strokes or as a multitouch gesture with two fingers drawing two strokes at once. Using GATO, the designer enters each of these articulation options in the online interface and receives the following predictions: $[1833, 1868]$ ms for the unistroke symbol , $[1028, 1044]$ ms for the two-stroke letter T, and $[860, 877]$ ms for the multitouch #.[1] The designer notes that both multistroke candidates, T and #, are about twice as fast as the unistroke symbol with just small differences between them, and decides to implement both options to accommodate various user preferences [50].

Predicting variation in gesture input

Now imagine a designer that already has a good gesture to enable users to quickly capture a photograph with their smart glasses by drawing on the side touchpad; see Figure 2b. The gesture is the rectangle shape (reflective of a picture or a camera symbol), and the designer wishes to know how long it will take users to produce it using one stroke , two strokes , and four strokes , respectively, but also how much production times will vary across users. The designer enters each articulation pattern in the GATO web application and gets the following estimates: $\mu_1 = 1258$ ms, $\sigma_1 = 476$ ms for the unistroke , $\mu_2 = 1660$ ms, $\sigma_2 = 758$ ms for the two-stroke , and $\mu_4 = 2843$ ms, $\sigma_4 = 1499$ ms for the four-stroke rectangle , respectively. The designer notes that the four-stroke version takes not only twice as long as the unistroke, but also that the variation in its production time is three times larger, which might impact the recognition accuracy of statistical classifiers that employ temporal features to discriminate between gesture types [11]. The design decision is to implement the unistroke version to make both users and the system efficient.

[1] These ranges represent 95% confidence intervals for the mean production time estimated by GATO from one articulation of each gesture type.

RELATED WORK

We review prior work that examined user performance with stroke gesture input on multitouch surfaces. We discuss applications of this prior work to gesture synthesis and analysis, highlighting the importance of production time as a key performance aspect for gesture user interfaces. We also review methods and techniques that are currently available for estimating unistroke gesture production times.

Stroke gesture input

There are many ways to produce a stroke gesture, depending on the number of strokes that decompose the gesture or the number of fingers that touch the screen; see Figure 1 for a few examples of multistroke and multitouch input. Common touchscreen technology, such as the one found on commodity smartphones, can detect up to ten discrete touch points at once, which enables designers to use gesture types from a rich space of unistrokes, multistrokes, multitouch, and bimanual input. Therefore, using more than one finger for stroke gesture input has become common. For example, pinching the screen with five fingers takes users directly to the home screen on an iPad. Expert gesture designs often involve the use of more fingers [8, 9,21,36], various finger parts [23], or even the entire hand for expressive input [40]. At the same time, users are known for their variations in articulating multistroke and multitouch gestures in terms of the number of strokes and fingers [4,52] when there are no constraints imposed [25,53].

Stroke gesture performance

Researchers have employed a variety of measures to characterize users' performance with stroke gesture input. For example, Blagojevic et al. [11] examined 114 distinct gesture features to inform the design of an accurate feature-based statistical classifier. Other researchers looked for representative features to depict various aspects of users' performance. For example, Anthony et al. [4] evaluated gesture articulation consistency, and reported high within-users consistency, but also less consistency for gestures produced with more strokes. Gesture features and measures have been also used to inform the design of gesture sets. For example, Long et al. [2] found that users' perceptions of gestures' visual similarity correlated with several features (such as length, area, or various angles), and derived a model for perceived gesture similarity.

Researchers have employed gesture measures to understand differences in performance between users or input conditions. Vatavu et al. [58,59,60] used accuracy measures to quantify deviations from "ideal" gestures produced by various user categories. Kane et al. [27] and Tu et al. [55] examined specific gesture features, such as "line steadiness" or "axial symmetry," to understand the differences between stroke gestures produced with either the pen or the finger [55], or by users with and without visual impairments [27]. Such gesture measures have proven very useful to characterize various aspects of gesture input as well as to inform gesture-based UI design. However, another line of work has focused on a more fundamental understanding of human movement during stroke gesture production by relating to key aspects from the motor control theory. We discuss this work in the following section.

Handwriting, gesture input, and the Kinematic Theory

Viviani et al. [62,63] were among the first to investigate the fundamentals of human handwriting and drawing behavior. Since then, an auspicious line of research has been the application of minimization principles to motor control, such as Flash and Hogan's minimum-jerk theory [18]. Further investigations showed that lognormal-based models, such as those postulated by the Kinematic Theory [44,45,46], are arguably the most accurate descriptors of human movement known today, compared to which "other models can be considered as successive approximations" according to Djioua and Plamondon [16].

In the context of the Kinematic Theory, stroke gestures are planned in advance in the form of an "action plan" described by a map of "virtual targets." The overall gesture trajectory is the result of the time superimposition of several velocity profiles of the action plan, approximating each gesture stroke with one or more "stroke primitives," i.e., one primitive for each velocity profile. The gesture articulation is directly linked to the quality of this superimposition. In a later section, we provide a more detailed introduction to this framework.

The Kinematic Theory has recently found applications to gesture input. For example, the "Gestures à Go Go" (G3) application [30,37] was introduced to synthesize stroke gestures from just a single example provided by the designer. Leiva et al. [29,30,31,33,56] evaluated the articulation characteristics of synthetic stroke gestures under various conditions, such as pen vs. finger input, slow vs. fast speed, or for various user categories. In this work, we rely on the principles of the Kinematic Theory to introduce the GATO technique.

Time estimation models for stroke gestures

Simple forms of stroke-based input, such as pointing and item selection from menus, have been extensively studied with Fitts' law and its variations [10,17,66], the steering law [1], or the Keystroke-Level Model (KLM) [13]. However, more complex stroke input, such as handwriting or free-form gesture paths drawn on touchscreens, need more sophisticated models to be able to characterize human performance effectively. Comprehensive surveys in this area are provided by Quinn and Zhai [49] and Müller et al. [42].

Isokoski [26] proposed a first-order rank model for stroke gestures that used the number of approximating straight line segments as a predictor of a gesture's shape complexity. Although Isokoski's model did not attempt to quantify production time explicitly, it was nevertheless found to predict the relative ranking of gesture types by their production times with reasonable accuracy [26,32].

The problem of predicting gesture production times has been addressed in the community with various techniques, from simple estimation rules [26] and training procedures [61] to complex models of the geometry of stroke gesture paths [12]. Among these, the recent KeyTime technique [32] has shown excellent performance for predicting production times. However, all these time estimation techniques were specifically designed for unistroke gestures, the simplest kind of stroke gesture input (see Figure 1) and, therefore, their performance on more complex gesture types is uncertain.

STROKE GESTURE TIME PREDICTION WITH GATO

We introduce in this section GATO, our technique for estimating the production times of any kind of gestures.

For all following discussion, we formalize multistroke and multitouch gestures as *sequences of touch and in-air strokes*. Let $\mathcal{G}$ denote a gesture composed of M touch strokes, for the production of which fingers need to land on and lift off of the touchscreen several times, such as the multistroke illustrated in Figure 3. We define $\mathcal{G}$ as a tuple of two sets, the list of touch strokes ($\mathcal{S}$) and the list of in-air movements ($\mathcal{Z}$) occurring between those strokes, i.e., $\mathcal{G} = (\mathcal{S},\mathcal{Z})$. Note that for some gesture applications, such as gesture recognition or analysis [30,51,52], such a rigorous representation may not be important, because the touchscreen can detect only the fingers moving on the screen and is ignorant of what happens in-between. However, for other applications, such as air+touch input [7,15] and especially for our investigation on predicting production times for multistrokes, in-air movements need to be considered. The distinction between touch and in-air movements is thus related to a crucial existence criteria for multistroke input: without the in-air movements between strokes, a multistroke gesture simply could not exist.

Each touch stroke $S_m \in \mathcal{S}$ may be produced with one or multiple fingers that touch the screen at the same time, such as the first stroke of the example shown in Figure 3. We formalize this aspect as $S_m = \{s_i \,|\, i = 1..F_m\}$, where F_m represents the number of fingers touching the screen simultaneously and each finger trace s_i is composed of K_i points x, y with associated timestamps t, i.e., $s_i = \{(x_{ik}, y_{ik}, t_{ik}) \,|\, k = 1..K_i\}$. The in-air strokes $\mathcal{Z} = \{\zeta_m \,|\, m = 1..M-1\}$ represent the movement of the hand between two lift-off and land-on events on the touchscreen. With these notations, the gesture from Figure 3 can be described as $\mathcal{G} = \{\{s_1, s_2, s_3\}, \{s_4\}, \{s_5, s_6\}\} \cup \{\zeta_1, \zeta_2\}$. Note that $|\zeta_m| = |S_m| - 1$, since in-air movements always take place between consecutive touch strokes.

Using this formalism, we can easily distinguish between the following types of stroke gestures relevant for our work, in increasing order of their complexity of articulation:

1. $\mathcal{G}$ is a *unistroke* gesture i.i.f. $M = 1$ and $F_1 = 1$, i.e., $\mathcal{G}$ is composed of only one stroke that is performed with one finger only; see Figure 1a on the first page for an example.
2. $\mathcal{G}$ is a *multitouch unistroke* gesture i.i.f. $M = 1$ and $F_1 > 1$, i.e., $\mathcal{G}$ is composed of one stroke performed with multiple fingers, all touching the screen at once; see Figure 1c.
3. $\mathcal{G}$ is a *multistroke* gesture i.i.f. $M > 1$ and $F_m = 1\, \forall m$, i.e., there are many strokes, each performed with one finger only; see Figure 1b.
4. $\mathcal{G}$ is a *multitouch multistroke* gesture i.i.f. $M > 1$ and $F_m > 1\, \exists m$, i.e., $\mathcal{G}$ is composed of multiple strokes, but at least one touch stroke is performed with multiple fingers, all touching the screen at the same time.

The production time of a multistroke multitouch gesture

The production time of a gesture $\mathcal{G} = (\mathcal{S},\mathcal{Z})$ is composed of the production times of all its strokes performed on the touchscreen ($S_m \in \mathcal{S}$), but also of the time during which the hand moves in air ($\zeta_m \in \mathcal{Z}$) between those strokes:

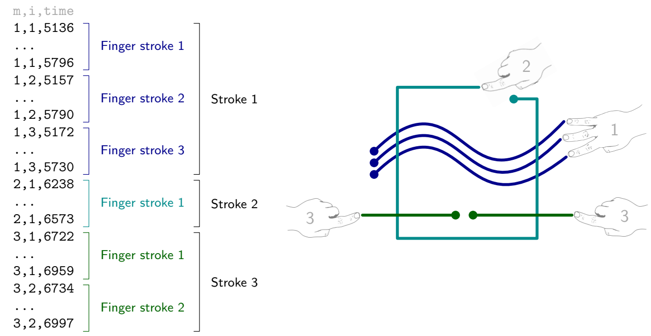


Figure 3. A multistroke multitouch gesture (right) and its numerical representation showing strokes S_m and finger traces s_i (left). Point coordinates x,y are omitted from this representation for the sake of brevity. Small dots denote the starting points of each trace s_i.

$$t(\mathcal{G}) = \sum_{m=1}^{M} t(S_m) + \sum_{m=1}^{M-1} t(\zeta_m) \qquad (1)$$

The production time of a stroke S_m is computed as the difference between the maximum and minimum timestamps of its finger traces $s_i \in S_m$.[2] For our example, $t(S_3) = \max\{6997, 6959\} - \min\{6734, 6722\} = 275$ ms. The time needed to move in air between strokes can be computed from the land-on and lift-off timestamps recorded by the touchscreen for the adjacent touch strokes. For our example, $t(\zeta_1) = \min\{6238\} - \max\{5796, 5790, 5730\} = 442$ ms. The overall production time for the gesture illustrated in Figure 3 is therefore: $660 + 442 + 335 + 149 + 275 = 1861$ ms. Note that we would have reached the same result simply by subtracting the maximum and minimum timestamps across all touch points (i.e., $6997 - 5136 = 1861$ ms), but the calculation of the production times of individual strokes is important for how GATO employs stroke gesture synthesis algorithms under the hood [30,31,32,37] to model the way strokes S_m are articulated by users in the time domain; see next section.

GATO and the Kinematic Theory

GATO employs the principle of "gesture synthesis" [30,31,32,37,56] and the concepts of the Kinematic Theory [44,45]. Therefore, we feel that a cursory introduction of the Kinematic Theory would be beneficial for readers.

The Kinematic Theory is a solid framework for studying human movement production, which has been recently adopted in HCI for stroke gesture synthesis and recognition [30,31,37]. The latest instantiation of this framework is the Sigma-Lognormal ($\Sigma\Lambda$) model [47], which was demonstrated to outperform many other approaches [16,46]. The Kinematic Theory assumes that a complex handwritten trace, e.g., a character, word, signature, or stroke gesture, is composed of a series of primitives[3] connecting a sequence of virtual targets, such as

[2] An alternative computation consists of averaging the contribution of each finger, but the differences between both approaches are very small in practice.

[3] The Kinematic Theory uses the term "stroke" to denote what we call a "primitive" in this paper. In HCI, we refer to a gesture stroke as the sequence of points bewteen two consecutive touch-down and touch-up events.

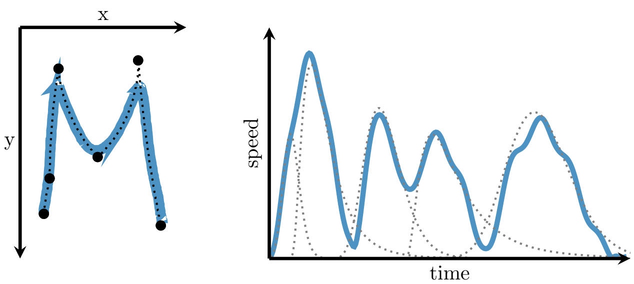


Figure 4. Left: A gesture stroke (thick line) is described by a series of primitives (dotted arcs) that connect virtual targets (black dots). Right: primitives are described by their lognormal velocity profiles.

those illustrated in Figure 4. The virtual targets correspond to near-zero-velocity peaks in the gesture strokes and are automatically computed by the $\Sigma\Lambda$ model [37]. These primitives form the action plan of the user for a specific gesture that, by means of the neuromuscular network, will produce a path on the touch-sensitive surface.

The $\Sigma\Lambda$ model computes the velocity profile of each primitive ($\vec{v}_i$) according to a lognormal function (Figure 4), which is defined by a set of central (D,t_0,θ) and peripheral (μ,σ) model parameters [44]. The summation of all velocity profiles enables reconstruction of the original gesture path, the quality of which is measured with the signal-to-noise ratio and the number of lognormals. For mathematical details, we refer the interested reader to Plamondon et al. [44,45,47] and to Leiva et al. [30,31,33] and Martín-Albo et al. [38,39] for applications to stroke gesture input and handwriting analysis, respectively.

GATO applies the concepts of the Kinematic Theory to model stroke gestures with lognormal velocity profiles in a 2-step procedure: (1) automatic synthesis of new timestamps for the points making up a gesture path and (2) estimation of gesture production times based on the synthesized data.

Step 1: Synthesis of stroke gesture timestamps

GATO predicts user-independent production times for multistroke and multitouch gestures as follows. For each stroke $S_m \in \mathcal{S}$ and each finger trace $s_i \in S_m$, GATO generates new timestamps (t_i) for all the K_i points of s_i:

$$t_i = \sum_{j \in s_i} \max_j \exp(\mu_j^* + 3\sigma_j^*) - t_{0_j} \tag{2}$$

where j denotes the j-th synthesized version of s_i, t_0 is the start time of each stroke primitive, and μ and σ are the peripheral parameters of the $\Sigma\Lambda$ model employed by the Kinematic Theory to synthesize human movements [44].

To introduce variability into the synthesized timestamps, each stroke primitive is distorted using the following noise model:

$$p_j^* = p_j + \mathcal{U}\left(-p_j, p_j\right) \tag{3}$$

where $p_j = \{\mu_j, \sigma_j\}$ are the peripheral $\Sigma\Lambda$ parameters and $\mathcal{U}$ the noise applied to each parameter. The noise function $\mathcal{U}$ is a uniform distribution centered around the value of each peripheral parameter (see Leiva et al. [31] for details) with

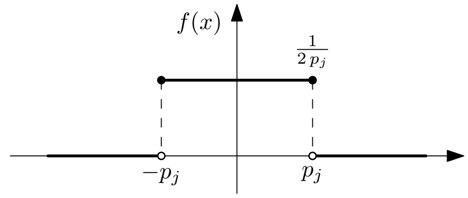


Figure 5. The uniform probability density function $\mathcal{U}$ of the peripheral noise applied to each primitive to induce variability for stroke gesture synthesis, to account for the natural variability that is present in user articulations.

the probability density function depicted in Figure 5. Peripheral noise introduces variation in the synthesized production times, reflective of articulation variation of the same gesture type by different users. To this effect, GATO employs user-independent noise values for p_j, empirically derived and validated by prior work [20,30,33,38]. Concretely, GATO uses $\mathcal{U}_\mu = 0.15$ and $\mathcal{U}_\sigma = 0.35$.

Step 2: Estimation of gesture production times

Assume we have n synthetic versions of gesture $\mathcal{G}$, for which the corresponding production times are denoted by t_i, $i = 1..n$. Starting from these values, GATO computes a prediction of the expected production time of gesture $\mathcal{G}$ as follows:

$$\hat{t}(\mathcal{G}) = \mathcal{F}(t_1, t_2, \ldots, t_n) \tag{4}$$

where $\mathcal{F}$ is a positive, real-valued, multivariable function. We refer to $\mathcal{F}$ as the TIME-ESTIMATOR function. The most immediate and simplest instance of a TIME-ESTIMATOR is the arithmetic mean of the production times of all synthetic versions of $\mathcal{G}$, i.e., $\hat{t}_{\text{M}} = \frac{1}{n}\sum t_i$. As we show in this paper, this approach delivers very accurate results. However, to control for cases in which the distribution of production times deviates from normality, we also evaluate other variants of TIME-ESTIMATORS, such as the median $\hat{t}_{\text{Mdn}}$, the 20%-trimmed mean $\hat{t}_{.20}$, and the winsorized mean $\hat{t}_{\text{W}}$. These measures of location are known for their robustness to outliers compared to the sensitivity of the mean due to their higher finite sample breakdown points of 0.2 and 0.5, respectively [64].[4]

The generic formalization of Equation (4) can be used to estimate measures of variation as well. For example, GATO computes predictions for both the variance and standard deviation of the expected production time for a given gesture type by instantiating $\mathcal{F}$ to $\hat{\sigma} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(t_i - \hat{t}_{\text{M}}\right)^2}$ for the standard deviation and to $\hat{\sigma}^2$ for the variance, respectively.

EVALUATION

We conducted a controlled experiment to evaluate the accuracy of GATO for predicting the production times of multistroke and multitouch gestures performed with either the stylus or the finger. To this end, we compared the predictions delivered by GATO with the production times of gestures actually articulated by users (i.e., groundtruth) with the following measures:

[4] The arithmetic mean has a finite breakdown point of $1/n$, which means that a single outlier can alter its value, making it arbitrarily small or large; see Wilcox [64]. The median has the highest breakdown point of 0.5.

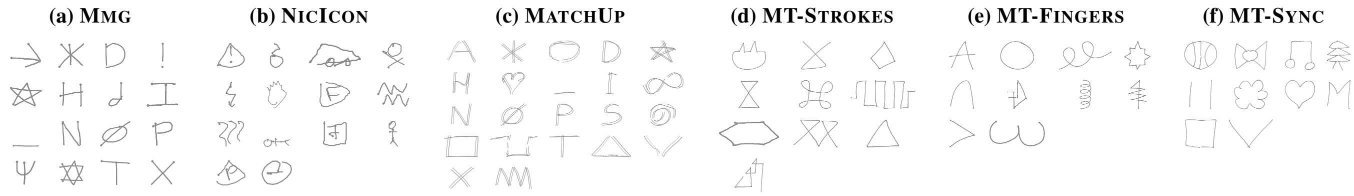


Figure 6. All the 82 multistroke and multitouch gesture types from the 6 public datasets used to evaluate the prediction performance of GATO.

1. RANKING-ACCURACY evaluates the extent to which GATO delivers the correct *ranking* of gesture types according to their production times. For example, if the mean production times of gestures A and B are 2000 ms and 2500 ms, respectively, and their predicted production times also respect this relative order, i.e., $\hat{t}_{\mathrm{M}}(\mathrm{A}) < \hat{t}_{\mathrm{M}}(\mathrm{B})$, then GATO is accurate. For more than two gestures, the ranking accuracy can be evaluated against groundtruth times using Spearman's rank correlation coefficient r_s. The closer r_s to 1, the more accurate GATO is for reporting the relative order of gesture production times.
2. ABSOLUTE-ERROR evaluates the extent to which GATO delivers the correct *magnitude* of the expected production time of a given gesture. For example, if the predicted production time of gesture A is 2100 ms, but the groundtruth time is 1989 ms, the absolute error is $|2100 - 1989| = 111$ ms.
3. RELATIVE-ERROR evaluates the extent to which GATO's predictions of production times deviate from groundtruth, percentage-wise. The relative error for the previous example is $100 \cdot |2100-1989| / 1989 = 5.6\%$.

Datasets

We evaluated GATO on six public multistroke and multitouch gesture datasets (see Figure 6):

1. MMG: Comprises 16 multistroke gesture types performed by 20 participants on a Tablet PC with 9,600 samples in total [5]. Each participant provided 10 executions per gesture type at three different speeds: slow, medium, and fast. Half of the participants used their fingers for input, while the other half used a stylus. Because participants were asked to produce gestures at three different speeds, we evaluated GATO separately for each articulation condition, which corresponds to having six sub-datasets of 1,600 gesture samples each (10 participants, 16 gestures, 10 repetitions) corresponding to all 6 combinations of {stylus, finger} $\times$ {slow, medium, fast} speed.
2. NICICON: Comprises 14 multistroke gesture types performed by 33 participants with a stylus on a Wacom Intuos2 tablet with 13,860 gesture samples in total [65]. Each participant provided 30 executions per gesture type.
3. MATCHUP: Comprises 22 multistroke and multitouch gesture types performed by 16 participants on a 3M C3266PW 32" multitouch display. For each gesture type, participants were asked to produce as many different variations as possible, and each variation was articulated for 5 times. This dataset comprises 5,155 gesture samples [51].
4. MT-STROKES: Comprises 10 multistroke multitouch gesture types performed by 18 participants on a 3M C3266PW 32" multitouch display. Each gesture type was articulated for 5 times under three conditions: using one stroke, two strokes, and three or more strokes. This dataset comprises 2,700 gesture samples in total [52].
5. MT-FINGERS: Comprises 10 multistroke multitouch gesture types performed by 18 participants on a 3M C3266PW 32" multitouch display. Each gesture was articulated with 5 repetitions under three conditions: using one finger, two fingers, and three or more fingers touching the screen at once. This dataset comprises 2,700 gesture samples in total [52].
6. MT-SYNC: Comprises 10 multistroke multitouch gesture types performed by 18 participants on a 3M C3266PW 32" multitouch display. Each gesture type was articulated for 5 times under two conditions: using one hand (sequential input) and two hands (parallel, bimanual input). This dataset comprises 1,800 gesture samples in total [52].

These datasets include 82 distinct gesture types that represent a good mixture of geometrical shapes and symbols with a large variety and wide range of complexity [5,51,65], and a good balance between familiar (i.e., known and practiced) and non-familiar (i.e., first time seen) symbols [52]. In total, we evaluate the prediction performance of GATO on 35,815 samples collected from 123 participants under various conditions.

Methodology

We evaluated GATO with a user-independent, leave-one-out cross-validation procedure [32], as follows. For each execution e of each gesture $\mathcal{G}$ produced by each participant $p \in \mathcal{P}$ (e.g., $|\mathcal{P}| = 18$ participants and e takes 2,700 values for the MT-STROKES dataset), GATO used that specific execution as the sample from which to predict the production time of gesture type $\mathcal{G}$ according to Equation 1 and using $n = 100$ synthetic values in Equation (4). The estimated time was compared to the groundtruth time, computed as the average production time of all the gestures of type $\mathcal{G}$ produced by the rest of the participants from $\mathcal{P} - \{p\}$; i.e., participant p was excluded from the computation of groundtruth data.

RESULTS

We report the prediction performance of GATO in terms of the RANKING, ABSOLUTE, and RELATIVE error measures.

Table 1 shows Spearman correlation coefficients computed between the time predictions delivered by GATO and groundtruth for each dataset. On average, GATO predictions correlated

Characteristics of the evaluation datasets					Spearman correlation				Absolute error (ms)				Relative error (%)			
Dataset	Num. classes	Num. samples	Num. users	Groundtruth 95% CI (ms)	$\hat{t}_{\mathrm{M}}$	$\hat{t}_{\mathrm{Mdn}}$	$\hat{t}_{.20}$	$\hat{t}_{\mathrm{W}}$	$\hat{t}_{\mathrm{M}}$	$\hat{t}_{\mathrm{Mdn}}$	$\hat{t}_{.20}$	$\hat{t}_{\mathrm{W}}$	$\hat{t}_{\mathrm{M}}$	$\hat{t}_{\mathrm{Mdn}}$	$\hat{t}_{.20}$	$\hat{t}_{\mathrm{W}}$
MMG stylus fast	16	1,600	10	[328, 746]	.788	.788	**.802**	.791	76	**48**	56	61	13.5	**8.5**	9.9	10.7
MMG stylus medium	16	1,600	10	[425, 911]	.900	**.902**	**.902**	**.902**	**4**	36	24	16	**0.7**	5.2	3.5	2.3
MMG stylus slow	16	1,600	10	[686, 1335]	.932	**.958**	.932	.941	**9**	48	21	**9**	**0.9**	4.5	2.0	**0.9**
MMG finger fast	16	1,600	10	[341, 766]	.775	**.832**	.828	.807	188	**163**	165	171	31.9	**27.6**	28.1	29.0
MMG finger medium	16	1,600	10	[441, 935]	.914	**.946**	.917	.917	124	**80**	91	102	16.9	**10.9**	12.5	13.9
MMG finger slow	16	1,600	10	[697, 1378]	**.950**	**.950**	**.950**	**.950**	**104**	153	134	125	**9.4**	13.9	12.1	11.3
NICICON	14	13,860	33	[715, 1082]	**.907**	**.907**	**.907**	**.907**	198	**118**	135	150	20.7	**12.3**	14.1	15.6
MATCHUP	22	5,155	16	[1657, 2251]	**.997**	**.997**	**.997**	**.997**	**32**	191	154	113	**1.6**	9.4	7.5	5.5
MT-STROKES	10	2,700	18	[2249, 4709]	**.999**	**.999**	**.999**	**.999**	**35**	304	232	181	**0.9**	8.0	6.1	4.7
MT-FINGERS	10	2,700	18	[1298, 4079]	**.987**	**.987**	**.987**	**.987**	**80**	169	125	82	**2.7**	5.7	4.2	2.8
MT-SYNC	10	1,800	18	[1577, 3808]	**.999**	**.999**	**.999**	**.999**	**39**	263	212	148	**1.3**	8.9	7.2	5.0
Overall	82	35,815	123		.914	**.933**	.929	.927	**80**	143	122	105	**9.1**	10.4	9.7	9.2

Table 1. Spearman correlation coefficients (r_s) and absolute and relative errors computed for GATO predictors with respect to groundtruth times. Notes: The highest correlation coefficients are highlighted for each dataset. All correlations are statistically significant at $p < .01$.

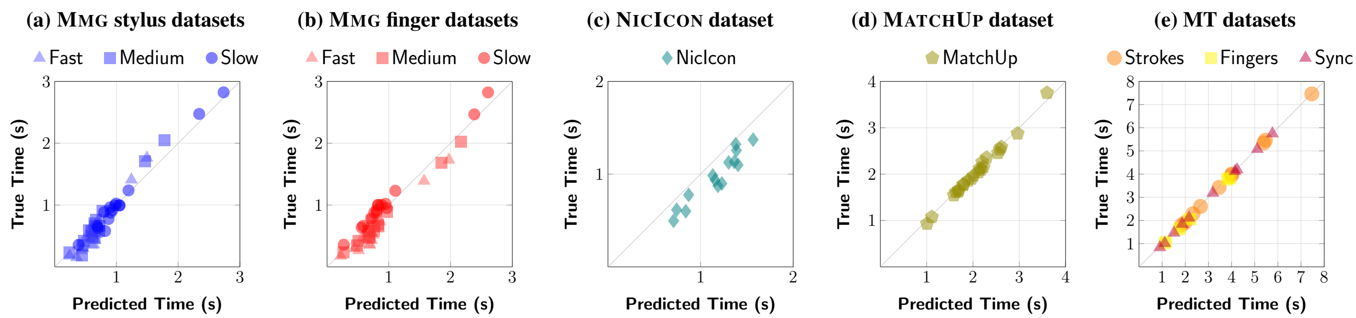


Figure 7. Production times ($\hat{t}_{\mathrm{M}}$) predicted by GATO vs. groundtruth (t_{True}), averaged on a per-gesture basis.

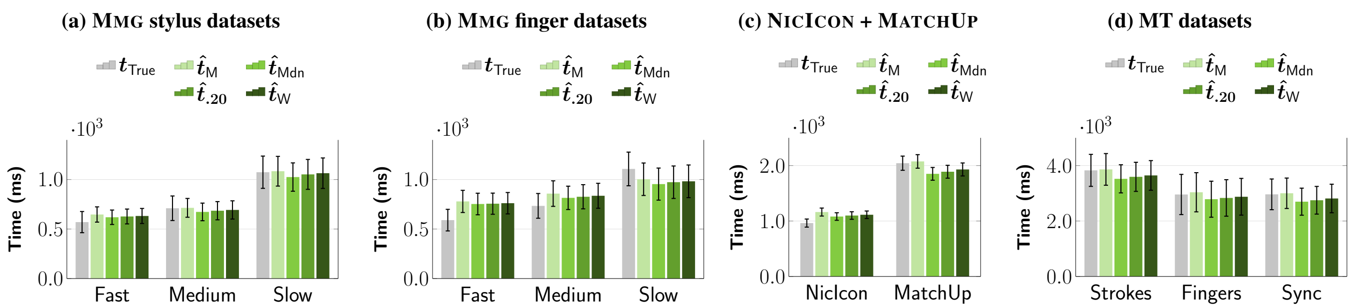


Figure 8. All GATO's production time estimators vs. groundtruth (t_{True}). Error bars denote the standard error of the mean.

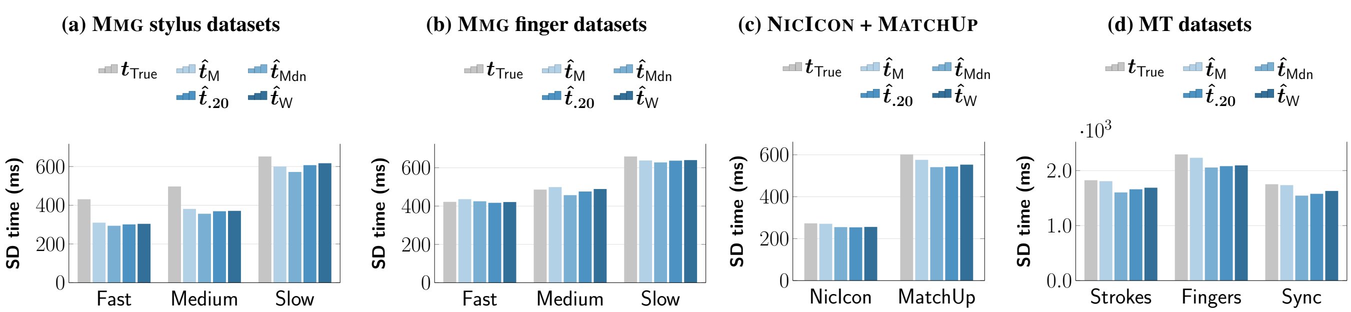


Figure 9. Standard deviations of GATO's production time estimators vs. groundtruth (SD t_{True}).

$r_s > .9$ with groundtruth (all correlations are statistically significant at $p < .01$) and the best performance was delivered by $\hat{t}_{\text{Mdn}}$ (avg $r_s = .933$, max $r_s = .999$). The ranking accuracy of GATO reached .999 for the MT-STROKES and MT-SYNC datasets, and .997 for the MATCHUP dataset. Figure 7 plots predicted times vs. groundtruth for each dataset, on a per-gesture basis. As it can be observed, GATO delivers time predictions that are close to the diagonal line. (The diagonal represents the performance of an ideal time predictor.)

Figure 8 shows the magnitudes of production times predicted by GATO for each dataset and each TIME-ESTIMATOR. A one-way ANOVA procedure showed a non-significant effect of TIME-ESTIMATOR on ABSOLUTE-ERROR for any of our evaluation datasets $(0.02 < F < 1.11, p > .05)$. These results suggest that GATO's predictions are on par with users' actual time performance with multistroke and/or multitouch gesture input. Furthermore, we found low effect sizes $\left(\eta_p^2 < 0.1\right)$ for all datasets, showing that the magnitude difference between predicted and measured times is of small practical importance; i.e., GATO estimations are very close to the actual production times. For example, the average absolute difference between the time predictions and groundtruth was 4 ms (relative error 0.7%) for the MMG-stylus-medium dataset, 118 ms (12.3%) for the MATCHUP dataset, and 32 ms (1.6%) for the MT-STROKES dataset.

We should point out that GATO is a flexible predictor of users' stroke gesture time performance. For example, besides predicting the magnitude of production times, GATO can also predict measures of variation. In support of this claim, Figure 9 shows the standard deviations of gesture production times predicted by GATO for each dataset. As it can be observed, GATO delivers very similar variation compared to groundtruth; e.g. SD $\boldsymbol{t}_{\text{True}}$ of 273 ms vs. SD $\hat{t}_{\text{M}}$ of 271 ms for the NICICON dataset, 602 vs. 576 ms for the MATCHUP dataset, or $1{,}825$ vs. $1{,}808$ ms for the MT-STROKES dataset. Considered together with the previously reported aspects of performance, these additional results build our confidence that GATO produces accurate estimations of stroke gesture production times.

GATO APPLICATION AND WEB SERVICE

As a service to the community, we deliver GATO implemented as a web application and a RESTful web service at the web address `https://luis.leiva.name/gato/`.

Using the GATO user interface, designers draw the gesture type for which they wish to obtain time prediction data, and GATO computes several estimators of location and dispersion. Our application can be used directly on any device that can run a modern browser; see Figure 10 for several examples.

For other touch-capable devices, such as touchpads on watch straps [19], smart glasses [22], or smart textiles [24], to name only a few recent trends in touch input on mobile and wearable devices, gestures can be collected by the designer and sent to the GATO web service, which will respond with JSON-encoded time prediction data; see Figure 11 for the response received from GATO for the two-stroke letter "T" gesture.

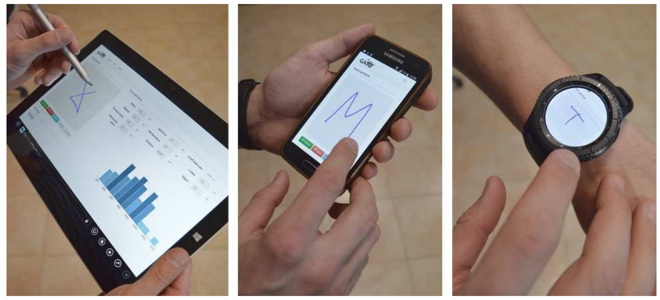


Figure 10. The GATO user interface, exemplified in these pictures, allows free-form drawing on a mobile-first, responsive UI, accessible from any touchscreen device with a modern web browser. In these examples, the designer gets real-time estimates of production times for gestures produced with a pen and the finger on three different mobile devices.


```
HTTP/1.1 200 OK
Connection: close

{
  "errors": null,
  "result": {
      "confidence_intervals": {
          "90%": [1925, 1954],
          "95%": [1922, 1957],
          "99%": [1916, 1962]
      },
      "max": 5055,
      "mean":1939,
      "median": 1863,
      "min": 501,
      "range": 4554,
      "standard_deviation": 848,
      "standard_error": 87,
      "trimmed_mean": 1862,
      "values": [1791, 2429, ..., 3841, 1228],
      "variance": 718366,
      "winsorized_mean": 1882
  }
}
```


Figure 11. Example of JSON API response that makes GATO practical for gestures collected on any touch input device without a web browser.

DISCUSSION, LIMITATIONS, AND FUTURE WORK

GATO requires only one gesture example (e.g., produced by the designer) to deliver predictions of that gesture's production times. Our experiments revealed that GATO is an accurate user-independent time predictor, reporting production times that are very close in magnitude to the actual groundtruth data. This performance is due to the fact that the gesture synthesizer employed by GATO under the hood [30] uses generic, user-independent value ranges for the $\Sigma\Lambda$ model parameters, which were empirically derived and validated for a wide range of users by prior work [20,30,33,38].

One requirement of GATO is that the gesture example should be reconstructable with high quality, as defined by the signal-to-noise ratio (SNR) measure of performance of the Kinematic Theory [30]. Previous work suggests that SNR values below 15 dB denote poor articulation quality [3,30,31] and, in such cases, the input gesture should be discarded and a new one provided. To address this aspect, the GATO web application alerts the designer when the provided example does not have enough quality to generate synthetic gestures effectively. This validation represents an important feature of GATO, which helps to confirm that the ultimate impulse response of a human movement follows a lognormal velocity curve [46].

An intuitive explanation for our accurate empirical results comes from the fact that GATO computes a numerical approximation for a given estimator of location, such as the mean, based on a bootstrapping approach [30], i.e., GATO computes what is known as "the sample mean" for $n = 100$ possible articulations for a given gesture type. Considering a large number of samples (as implemented by our cross-validation evaluation procedure with ≈36k trials), the central limit theorem indicates that the average sample mean should converge to the population mean, or the groundtruth mean in our case. However, for practical applications of GATO, using more than one gesture example is recommended and, under the above considerations, we believe that the accuracy of GATO may improve if more samples were used. That includes both user-dependent (i.e., the designer enters multiple articulations of the gesture) and user-independent predictions (i.e., the designer asks a few colleagues or participants to produce one articulation of the gesture). While we provide empirical results in this paper for estimating production times based on one gesture sample only, interesting future work will look at considering the effect of larger sample sizes (user-dependent and user-independent) on the prediction performance of stroke gesture production times, but also at theoretical argumentations to explain the accuracy performance of GATO.

More complex approaches to prediction, such as based on more gesture examples, will probably benefit from an adaptation of our evaluation procedure as well. For example, a more rigorous evaluation scenario for such cases would be picking one sample of each gesture type $\mathcal{G}$ based on some best/worse performance criteria, such as the highest or the lowest signal-to-noise ratio among all the reconstructed exemplars, and use that sample for prediction. This procedure would probably resemble well to how an end-user would test our web application to understand the limits of time performance, i.e., the best-case and worst-case scenarios.

We also need to point out that the datasets that we considered during evaluation include gestures that were performed under laboratory conditions. Thus, participants were able to focus entirely on their gesture performance. We expect that small mobile devices, such as smartphones or smartwatches, which need to be held during input by adopting particular hand poses, or other contexts of use, such as walking or situational impairments, might affect the hand kinematics. Thus, further investigation is needed to validate GATO for small screen devices and mobile or wearable contexts of use.

CONCLUSION

GATO delivers very accurate user-independent predictions of multistroke and multitouch gesture production times with minimum effort required from designers. Specifically, GATO requires just one gesture example that designers can draw themselves, and is readily available as an online application on the web and a RESTful JSON API. GATO provides researchers and practitioners with unprecedented levels of accuracy and sophistication to characterize their users' *a priori* time performance with stroke gesture input of all kinds: unistrokes, multistrokes, multitouch, or combinations thereof. We expect that GATO's time predictions will advance our capacity as a community to model, analyze, and understand users' stroke gesture articulations on touchscreen devices and, consequently, will foster more effective and efficient gesture-based user interface designs.

ACKNOWLEDGMENTS

We thank the anonymous reviewers for their constructive and useful feedback to strengthen the final version of our paper. R.-D. Vatavu acknowledges support from the project no. PN-III-P2-2.1-PED-2016-0688 (209PED/2017) financed by UEFISCDI, Romania.

REFERENCES

1. Johnny Accot and Shumin Zhai. 1997. Beyond Fitts' law: Models for trajectory-based HCI tasks. In *Proc. CHI '97*.
2. Jr. Allan C. Long, James A. Landay, Lawrence A. Rowe, and Joseph Michiels. 2000. Visual similarity of pen gestures. In *Proc. CHI '00*.
3. Abdullah Almaksour, Eric Anquetil, Réjean Plamondon, and Christian O'Reilly. 2011. Synthetic handwritten gesture generation using Sigma-Lognormal model for evolving handwriting classifiers. In *Proc. IGS '11*.
4. Lisa Anthony, Radu-Daniel Vatavu, and Jacob O. Wobbrock. 2013. Understanding the consistency of users' pen and finger stroke gesture articulation. In *Proc. GI '13*.
5. Lisa Anthony and Jacob O. Wobbrock. 2012. $N-protractor: a fast and accurate multistroke recognizer. In *Proc. GI '12*.
6. Caroline Appert and Shumin Zhai. 2009. Using strokes as command shortcuts: Cognitive benefits and toolkit support. In *Proc. CHI '09*.
7. Ilhan Aslan, Ida Buchwald, Philipp Koytek, and Elisabeth André. 2016. Pen + mid-air: An exploration of mid-air gestures to complement pen input on tablets. In *Proc. NordiCHI '16*.
8. Shiri Azenkot, Kyle Rector, Richard Ladner, and Jacob Wobbrock. 2012. PassChords: Secure multi-touch authentication for blind people. In *Proc. ASSETS '12*.
9. Gilles Bailly, Jörg Müller, and Eric Lecolinet. 2012. Design and evaluation of finger-count interaction: Combining multitouch gestures and menus. *Int. J. Hum.-Comput. Stud.* **70**(10).
10. Xiaojun Bi, Yang Li, and Shumin Zhai. 2013. Ffitts law: Modeling finger touch with Fitts' law. In *Proc. CHI '13*.
11. Rachel Blagojevic, Samuel Hsiao-Heng Chang, and Beryl Plimmer. 2010. The power of automatic feature selection: Rubine on steroids. In *Proc. SBIM '10*.
12. Xiang Cao and Shumin Zhai. 2007. Modeling human performance of pen stroke gestures. In *Proc. CHI '07*.
13. Stuart K. Card, Thomas P. Moran, and Allen Newell. 1980. The keystroke-level model for user performance time with interactive systems. *Commun. ACM* **23**(1).

14. Steven J. Castellucci and I. Scott MacKenzie. 2008. Graffiti vs. Unistrokes: An empirical comparison. In *Proc. CHI '08*.
15. Xiang Anthony Chen, Julia Schwarz, Chris Harrison, Jennifer Mankoff, and Scott E. Hudson. 2014. Air+touch: Interweaving touch & in-air gestures. In *Proc. UIST '14*.
16. Moussa Djioua and Réjean Plamondon. 2009. Studying the variability of handwriting patterns using the Kinematic Theory. *Hum. Mov. Sci.* **28**(5).
17. Paul M. Fitts. 1954. The information capacity of the human motor system in controlling the amplitude of movement. *J. Exp. Psychol.* **47**(6).
18. Tamar Flash and Neville Hogan. 1985. The coordination of arm movements: an experimentally confirmed mathematical model. *J. Neurosci.* **5**(7).
19. Markus Funk, Alireza Sahami, Niels Henze, and Albrecht Schmidt. 2014. Using a touch-sensitive wristband for text entry on smart watches. In *Proc. CHI '14*.
20. Javier Galbally, Réjean Plamondon, Julián Fierrez, and Javier Ortega-García. 2012. Synthetic on-line signature generation. Part II: Experimental validation. *Pattern Recogn.* **45**(7).
21. Emilien Ghomi, Stéphane Huot, Olivier Bau, Michel Beaudouin-Lafon, and Wendy E. Mackay. 2013. Arpège: Learning multitouch chord gestures vocabularies. In *Proc. ITS '13*.
22. Tovi Grossman, Xiang Anthony Chen, and George Fitzmaurice. 2015. Typing on glasses: Adapting text entry to smart eyewear. In *Proc. MobileHCI '15*.
23. Chris Harrison, Julia Schwarz, and Scott E. Hudson. 2011. Tapsense: Enhancing finger interaction on touch surfaces. In *Proc. UIST '11*.
24. Florian Heller, Stefan Ivanov, Chat Wacharamanotham, and Jan Borchers. 2014. Fabritouch: Exploring flexible touch input on textiles. In *Proc. ISWC '14*.
25. Uta Hinrichs and Sheelagh Carpendale. 2011. Gestures in the wild: Studying multi-touch gesture sequences on interactive tabletop exhibits. In *Proc. CHI '11*.
26. Poika Isokoski. 2001. Model for unistroke writing time. In *Proc. CHI '01*.
27. Shaun K. Kane, Jacob O. Wobbrock, and Richard E. Ladner. 2011. Usable gestures for blind people: Understanding preference and performance. In *Proc. CHI '11*.
28. Per Ola Kristensson and Shumin Zhai. 2004. SHARK^2: A large vocabulary shorthand writing system for pen-based computers. In *Proc. UIST '04*.
29. Luis A. Leiva. 2017. Large-scale user perception of synthetic stroke gestures. In *Proc. DIS '17*.
30. Luis A. Leiva, Daniel Martín-Albo, and Réjean Plamondon. 2016. Gestures à Go Go: Authoring synthetic human-like stroke gestures using the kinematic theory of rapid movements. *ACM T. Intel. Syst. Tec.* **7**(2).
31. Luis A. Leiva, Daniel Martín-Albo, and Réjean Plamondon. 2017a. The Kinematic Theory produces human-like stroke gestures. *Interact. Comput.* **29**(4).
32. Luis A. Leiva, Daniel Martín-Albo, Réjean Plamondon, and Radu-Daniel Vatavu. 2018. KeyTime: Super-accurate prediction of stroke gesture production times. In *Proc. CHI '18*.
33. Luis A. Leiva, Daniel Martín-Albo, and Radu-Daniel Vatavu. 2017b. Synthesizing stroke gestures across user populations: A case for users with visual impairments. In *Proc. CHI '17*.
34. Yang Li. 2010. Gesture Search: A tool for fast mobile data access. In *Proc. UIST '10*.
35. Hao Lü and Yang Li. 2011. Gesture avatar: A technique for operating mobile user interfaces using gestures. In *Proc. CHI '11*.
36. Yuexing Luo and Daniel Vogel. 2015. Pin-and-cross: A unimanual multitouch technique combining static touches with crossing selection. In *Proc. UIST '15*.
37. Daniel Martín-Albo and Luis A. Leiva. 2016. G3: bootstrapping stroke gestures design with synthetic samples and built-in recognizers. In *Proc. MobileHCI '16*.
38. Daniel Martín-Albo, Réjean Plamondon, and Enrique Vidal. 2014. Training of on-line handwriting text recognizers with synthetic text generated using the Kinematic Theory of rapid human movements. In *Proc. ICFHR '14*.
39. Daniel Martín-Albo, Réjean Plamondon, and Enrique Vidal. 2015. Improving sigma-lognormal parameter extraction. In *Proc. ICDAR '15*.
40. Fabrice Matulic, Daniel Vogel, and Raimund Dachselt. 2017. Hand contact shape recognition for posture-based tabletop widgets and interaction. In *Proc. ISS '17*. 3–11.
41. Meredith Ringel Morris, Annuska Zolyomi, Catherine Yao, Sina Bahram, Jeffrey P. Bigham, and Shaun K. Kane. 2016. "with most of it being pictures now, i rarely use it": Understanding twitter's evolving accessibility to blind users. In *Proc. CHI '16*.
42. Jörg Müller, Antti Oulasvirta, and Roderick Murray-Smith. 2017. Control theoretic models of pointing. *ACM Trans. Comput.-Hum. Interact.* **24**(4).
43. Miguel A. Nacenta, Yemliha Kamber, Yizhou Qiang, and Per Ola Kristensson. 2013. Memorability of pre-designed and user-defined gesture sets. In *Proc. CHI '13*.
44. Réjean Plamondon. 1995a. A kinematic theory of rapid human movements. Part I: Movement representation and control. *Biol. Cybern.* **72**(4).
45. Réjean Plamondon. 1995b. A kinematic theory of rapid human movements. Part II: Movement time and control. *Biol. Cybern.* **72**(4).

46. Réjean Plamondon, Adel M. Alimi, Pierre Yergeau, and Franck Leclerc. 1993. Modelling velocity profiles of rapid movements: a comparative study. *Biol. Cybern.* **69**(1).
47. Réjean Plamondon and Moussa Djioua. 2006. A multi-level representation paradigm for handwriting stroke generation. *Hum. Mov. Sci.* **25**(4–5).
48. Benjamin Poppinga, Alireza Sahami Shirazi, Niels Henze, Wilko Heuten, and Susanne Boll. 2014. Understanding shortcut gestures on mobile touch devices. In *Proc. MobileHCI '14*.
49. Philip Quinn and Shumin Zhai. 2018. Modeling gesture-typing movements. *Hum.-Comput. Interact.* **33**(2).
50. Yosra Rekik, Laurent Grisoni, and Nicolas Roussel. 2013. Towards many gestures to one command: A user study for tabletops. In *Proc. INTERACT '13*.
51. Yosra Rekik, Radu-Daniel Vatavu, and Laurent Grisoni. 2014a. Match-up & conquer: A two-step technique for recognizing unconstrained bimanual and multi-finger touch input. In *Proc. AVI '14*.
52. Yosra Rekik, Radu-Daniel Vatavu, and Laurent Grisoni. 2014b. Understanding users' perceived difficulty of multi-touch gesture articulation. In *Proc. ICMI '14*.
53. Yosra Rekik, Radu-Daniel Vatavu, and Laurent Grisoni. 2016. *Spontaneous Gesture Production Patterns on Multi-touch Interactive Surfaces*. Springer, Cham.
54. Quentin Roy, Sylvain Malacria, Yves Guiard, Eric Lecolinet, and James Eagan. 2013. Augmented letters: Mnemonic gesture-based shortcuts. In *Proc. CHI '13*.
55. Huawei Tu, Xiangshi Ren, and Shumin Zhai. 2012. A comparative evaluation of finger and pen stroke gestures. In *Proc. CHI '12*.
56. Ovidiu-Ciprian Ungurean, Radu-Daniel Vatavu, Luis A. Leiva, and Réjean Plamondon. 2018. Gesture input for users with motor impairments on touchscreens: Empirical results based on the kinematic theory. In *Proc. CHI EA '18*. Article LBW537, 6 pages.
57. Radu-Daniel Vatavu. 2017. Improving gesture recognition accuracy on touch screens for users with low vision. In *Proc. CHI '17*.
58. Radu-Daniel Vatavu, Lisa Anthony, and Jacob O. Wobbrock. 2013. Relative accuracy measures for stroke gestures. In *Proc. ICMI '13*.
59. Radu-Daniel Vatavu, Lisa Anthony, and Jacob O. Wobbrock. 2014. Gesture heatmaps: Understanding gesture performance with colorful visualizations. In *Proc. ICMI '14*.
60. Radu-Daniel Vatavu, Gabriel Cramariuc, and Doina Maria Schipor. 2015. Touch interaction for children aged 3 to 6 years: Experimental findings and relationship to motor skills. *Int. J. Hum.-Comput. Stud.* **74**(1).
61. Radu-Daniel Vatavu, Daniel Vogel, Géry Casiez, and Laurent Grisoni. 2011. Estimating the perceived difficulty of pen gestures. In *Proc. INTERACT '11*.
62. Paolo Viviani and Tamar Flash. 1995. Minimum-jerk, two-thirds power law, and isochrony: converging approaches to movement planning. *J. Exp. Psychol.* **21**(1).
63. Paolo Viviani and Carlo Terzuolo. 1982. Trajectory determines movement dynamics. *Neuroscience* **7**(2).
64. Rand Wilcox. 2012. *Modern Statistics for the Social and Behavioral Sciences*. Taylor & Francis Group, LLC, Boca Raton, FL, USA.
65. D. Willems, R. Niels, M. van Gerven, and L. Vuurpijl. 2009. Iconic and multi-stroke gesture recognition. *Pattern Recogn.* **42**(12).
66. Jacob O. Wobbrock, Edward Cutrell, Susumu Harada, and I. Scott MacKenzie. 2008. An error model for pointing based on fitts' law. In *Proc. CHI '08*.
67. Jacob O. Wobbrock, Brad A. Myers, and John A. Kembel. 2003. Edgewrite: A stylus-based text entry method designed for high accuracy and stability of motion. In *Proc. UIST '03*.
68. Shaomei Wu and Lada A. Adamic. 2014. Visually impaired users on an online social network. In *Proc. CHI '14*. 10.
69. Shumin Zhai and Per Ola Kristensson. 2012. The word-gesture keyboard: Reimagining keyboard interaction. *Commun. ACM* **55**(9).
70. Shumi Zhai, Per Ola Kristensson, Caroline Appert, Tue H. Anderson, and Xiang Cao. 2012. Foundational issues in touch-surface stroke gesture design — an integrative review. In *Foundations and Trends in Human-Computer Interaction*. Vol. 5.
71. Chi Zhang, Nan Jiang, and Feng Tian. 2016. Accessing mobile apps with user defined gesture shortcuts: An exploratory study. In *Proc. ISS '16*.