# Gesture Synthesis for Human-Computer Interaction

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

Daniel Martín-Albo
*Wiris Math, SL*
Barcelona, Spain
dmas@wiris.com

Réjean Plamondon
*École Polytechnique de Montréal*
Montréal, QC, Canada
rejean.plamondon@polymtl.ca

***Abstract*—Gesture recognizers require a large pool of training data to achieve good accuracy. However, recruiting participants, data collection and labeling, etc. necessary for achieving this goal are usually time-consuming and expensive. Fortunately, the Kinematic Theory allows to easily bootstrap gesture generation. In this paper, we show that the synthesized gestures not only perform equally similar to gestures generated by human users but also they look and feel the same. Ultimately, this work benefits researchers and designers who wish to prototype gesture-driven applications.**

***Index Terms*—Gesture Synthesis; Bootstrapping; Gesture Recognition; Strokes; Marks; Symbols; Unistrokes; Multistrokes; Multitouch; Kinematics; User Interfaces; Rapid Prototyping**

## I. Introduction

Gestures are increasingly becoming a predominant input modality in today's graphical user interfaces (GUIs). Gesture interaction is possibly one of the most researched areas in Human-Computer Interaction (HCI), with a long history that started as early as 1960, with the Sketchpad project [40] and the RAND tablet [10]. Gestures can be mid-air (more prominent in gaming applications) or stroke based (more prominent in mobile applications). We are particularly interested in the latter type, since stroke gestures are becoming more and more relevant to mainstream products such as touchscreen-capable devices like smartphones and tablets.

Stroke gestures represent the movement trajectory of one or more contact points on a sensitive surface. Stroke gestures are sometimes also called "pen gestures", "hand drawn marks", "hand drawn gestures", "hand markings", or "markings" [45]. Stroke gestures tend to give richer perceptual cues to the user, to form an association between the shape of the gesture and the meaning of the command [5]. Stroke gestures also may improve the usability of UIs, by replacing standard shortcuts by more accessible triggers.

Today, stroke gestures are mostly used in consumer devices for executing simple actions, such as pinching a picture to zoom in/out, swiping to reveal an options menu, or panning to switch between apps. Nevertheless, stroke gestures are increasingly being incorporated to facilitate random access to smartphone contents, such as invoking a command hidden in an advanced settings menu or quickly searching for a friend's email in the contacts list. Therefore, it is expected that stroke gestures will make a notable impact in consumers' lives.

This paper agglutinates ideas and results previously published by the authors at scientific journals and conferences [21], [22], [23], [27].

In general, any application that is driven by gestures must rely on some recognition-based techniques. These techniques often require expert knowledge in pattern recognition or machine learning, something that is typically beyond the reach of many developers and UI designers. Furthermore, recruiting participants, data collection and labeling, etc. necessary for using these techniques are usually time-consuming and expensive. Thus, it is important to investigate how to empower developers to (1) quickly collect gesture samples and (2) create recognizers; both for improving GUI usage and user experience.

## II. Related Research

We review core areas that resemble the most to our work: approaches to gesture recognition and gesture bootstrapping.

### A. Gesture Recognition

Gesture recognition has its own roots in sketching and handwriting recognition [9], [11], [29], [38]. In HCI, most gesture recognizers for prototyping GUIs are based on the template matching (or instance-based) approach [20]: a query gesture is geometrically compared against a number of stored templates, using 1 nearest-neighbor for classification and either Euclidean distance or a Mean Square Error (MSE) score as dissimilarity measures. Template matchers are a very viable and a relatively simple solution for recognizing gestures, and can be adapted to personalized user gestures.

Popular examples of these template-based recognizers among the HCI literature are part of the so-called "$ family": $1 [44], $N [3], and their newer versions Protractor [24] and $N-Protractor [4], respectively. More recently, Vatavu [42] introduced $P, a sequential-agnostic recognizer where strokes are treated as a cloud of 2D points, discarding thus stroke number, order, and direction.

### B. Gesture Boostrapping

Example-based approaches like GRANDMA [38], Agate [18], or Gesture Studio [26] allow developers to create and test gestures by recording examples. There are a number of similar systems tailoring end-users, like EventHurdle [16], A CAPpella [12], or GestIT [39]. They support designers' explorative prototyping through programming by demonstration environments. Another strand of research is aimed at simplifying the process of designing gesture sets. For example, Gesture Script [25], Gesture

Marks [31], Gestalt [32], or CrowdLearner [2]. Finally, we can find a number of competing systems aimed at creating synthetic 3D gestures as a means to improve gesture recognizers, including e.g. MAGIC [6], [17] and Gesture Follower [8].

Overall, training data is the key factor to build a competitive gesture recognizer, for which most of the previously reviewed approaches have contributed to generate their own, without having to recruit participants and perform time-comsuming user evaluations. They also have contributed to decreasing the number of iterations needed to build a fast and stable gesture recognition interface. However, there is no evidence that any of the previous works can produce human-like samples. Further, artificially generated samples usually perform poorly since they do not illustrate sufficient variation required for high-quality training [1], [36], [37].

## III. Synthesizing Gestures

Many models have been proposed to study human movement production; e.g. [7], [41], [30], [43], among which the Kinematic Theory [33] provides a well-established and solid framework for the study of the production of human movements This framework takes into account different psychophysiological features, such as the neuromuscular response time, and has been shown to outperform many other approaches [34]. The Sigma-Lognormal ($\Sigma\Lambda$) model [35] is the latest instantiation of this framework, and very recently has been used to explore gesture recognition.

At a high-level representation, the Kinematic Theory assumes that a complex handwritten trace (such as a stroke gesture) is composed of a series of *primitives* (circular arcs) connecting a sequence of *virtual targets*. This series of primitives conform the "action plan" of the user, which is fed through the neuromuscular network to produce a trajectory that leaves a handwritten trace.

Under this framework, each gesture primitive is modeled according to a lognormal function of their velocity profile, defined by a set of *central* parameters $(D, t_0, \theta)$ and *peripheral* parameters $(\mu, \sigma)$ [33]. Then, an extractor computes the parameter values that best explain the observed velocity profiles [28]. Once the gesture primitives are modeled, perturbations can be added to the model parameters in order to produce different gesture variations [22]:

$$p_i^* = p_i + n_{p_i} \tag{1}$$

where $p_i = \{\, \mu_i, \sigma_i, D_i, \theta_i \,\}$ denote the $\Sigma\Lambda$ parameters, with $n_{p_i} = \mathcal{U}(-n_i, n_i)$ being the noise applied to each primitive, according to a uniform distribution (i.e., a rectangular distribution with constant probability) centered around the expected human variability ranges [15], [22]: $n_\mu = n_\sigma = 0.1$, $n_D = 0.15$, $n_\theta = 0.06$. Figure 1 shows some examples of the synthetic gestures produced with the Kinematic Theory.

Previous works have demonstrated the connection between the distortion of the Sigma-Lognormal parameters and the intra-variability found in human handwriting [13]. Combining both types of variations reflects real-life situations like performing the same movement under different psychophysiological conditions. For example, perturbations in $\mu$ and $\sigma$ mimic peripheral noise, e.g., a user who articulates the same gesture slightly different each time; perturbations in $D$ and $\theta$ refer to central fluctuations that occur in the position of the virtual targets of the action plan from one articulation to the next [21], [22].

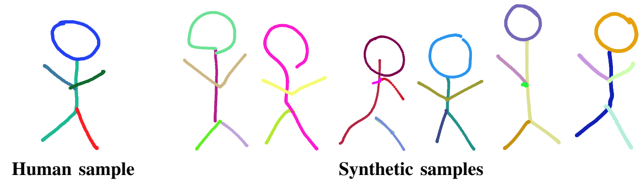


Fig. 1: Examples of gestures synthetized with the Kinematic Theory, using a single human example as input.

## IV. Gesture Performance Analysis

We compared the performance of synthetic gestures with that of human samples under user-independent tests in terms of articulation speed, input device, and gesture variability. The interested reader may consult user-dependent tests and a follow-up evaluation in our previous work [21].

We synthesized two popular datasets in HCI: GDS [44] and MMG [4]. On the one hand, the GDS dataset comprises 5,280 *unistroke* gestures (16 classes). Ten users provided 10 samples per class at 3 articulation speeds (slow, medium, fast) using an iPAQ Pocket PC (stylus as input device). On the other hand, the GDS dataset comprises 5,280 *multistroke* gestures (16 classes). Twenty users provided 10 samples per class at 3 articulation speeds (same as in GDS) using either finger (half of the users) or stylus as input device on a Tablet PC.

### A. Impact of Articulation Speed

We sought to analyze whether gesture articulation speed leads to a difference in classification error rates between human and synthetic templates. The GDS dataset was analyzed with the \$1 recognizer, whereas MMG was analyzed with the \$P recognizer. Both recognizers were fed with 10 templates. Table I summarizes this experiment. A two-tailed paired $t$-test (Bonferroni corrected) revealed no statistically significant differences for any of the articulation speeds, suggesting thus that synthetic gestures perform the same as their human counterparts.

TABLE I: Effect of articulation speed on error rates (in %)

| **Type** | **GDS** | | | **MMG** | | |
|---|---|---|---|---|---|---|
| | Slow | Med. | Fast | Slow | Med. | Fast |
| Human | 0.09 | 0.06 | 0.18 | 0.03 | 0.02 | 0.06 |
| Synthetic | 0.20 | 0.27 | 0.90 | 0.17 | 0.36 | 0.56 |

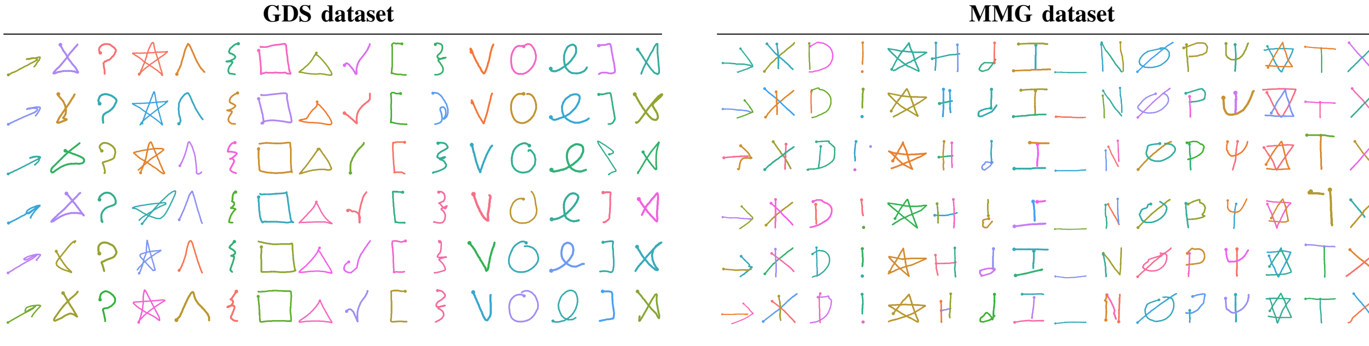


Fig. 2: Five examples of each synthesized gesture using the human examples (top row) as input.

### *B. Impact of Input Device*

We also sought to analyze whether the input device leads to a difference in classification error rates between human and synthetic templates. This analysis was performed over the MMG dataset, which is the one that provides two data splits: finger and stylus. We used the $P recognizer with 10 templates. Table II summarizes this experiment. A two-tailed paired $t$-test (Bonferroni corrected) revealed no statistically significant differences, suggesting thus that gestures can be successfully synthesized with both a stylus and the finger.

TABLE II: Effect of input device on error rates (in %)

| **Human** | | **Synthetic** | |
|---|---|---|---|
| Finger | Stylus | Finger | Stylus |
| 0.03 | 0.04 | 0.43 | 0.29 |

### *C. Impact of Gesture Variability*

Finally, we sought to analyze whether an increase in the amount of noise $\xi$ introduced to the Sigma-Lognormal model parameters leads to more variable synthetic gestures. We computed the mean squared error between human and synthetic gestures for different number of synthesized samples using $\xi$ from 0.0 (no variability) to 1.0 (maximum variability, in the allotted human ranges [21]). Table III summarizes this experiment. As expected, it was found that synthetic samples are more variable as $\xi$ increases. Interestingly, variability was found to increase as the number of requested synthetic samples increases.

TABLE III: Gesture variability (mean squared error)

| **N** | **GDS** | | **MMG** | |
|---|---|---|---|---|
| | $\xi = 0.0$ | $\xi = 1.0$ | $\xi = 0.0$ | $\xi = 1.0$ |
| 10 | 554.9 | 593.6 | 169.7 | 490.4 |
| 100 | " | 622.1 | " | 493.4 |
| 1000 | " | 621.4 | " | 498.7 |

## V. Gesture Similarity Analysis

To provide further evidence on the value of the Kinematic Theory as a means to generate stroke gestures, we conducted an online survey that measured the user perception toward gestures' human-likeness. We used the same datasets depicted in the previous section, both in their original and synthesized form. The survey is still available online at https://g3.prhlt.upv.es/guessit/. Eventually, 236 participants took part in this study.

We defined the guessing accuracy as the user's ability to distinguish between human and synthetic samples; i.e., the proportion of gestures that were successfully classified by the user. Then, two types of errors can be committed [14], [15]: (i) a synthetic gesture is mistaken with a real sample, measured by the False Real Rate (Type I error); and (ii) a real gesture is marked as synthetic, measured by the False Synthetic Rate (Type II error). The results are presented in Table IV.

TABLE IV: Accuracy and error rates (in %)

| **Accuracy** | **False Real Rate** | **False Synthetic Rate** |
|---|---|---|
| 49.65 | 29.84 | 20.40 |

A paired two-sample $t$-test (two-tailed alternative hypothesis) revealed that there is no difference between classified and misclassified gestures, proving thus the human-like appearance of synthetic samples. In sum, participants could not tell human and synthetic gestures apart. Follow-up analyses [22], [19] provided further evidence that synthesized gestures are actually reflective of how users produce stroke gestures. We concluded therefore that the visual appearance of the synthetic samples is very similar and close to that of human gestures.

## VI. Discussion

Users tend to be reluctant to invest time and effort upfront to train or adjust software before using it [5]. Further, users are unwilling to provide more than a small set of samples for training [24]. Consequently, synthesizing techniques like are of high value, as they help to lower time and costs associated to recruiting users and subsequent data labeling.

Until now the “human likeness” of synthesized gestures was measured indirectly, intermediated by classification/recognition accuracy performance. Our studies are important because recent research has shown how different users produce different gesture articulations in various conditions.

Yet, finally researchers and practitioners can be confident that synthesized gestures using the Kinematic Theory are actually reflective of how users produce stroke gestures.

## VII. Conclusion

We have shown that the Kinematic Theory generates stroke gestures that can be useful to researchers and practitioners in many ways. The synthesized gestures not only perform equally similar to their human couterparts but also they look and feel the same. In sum, the Kinematic Theory provides the HCI community with a reliable way to synthesize gesture sets without having to expressly collect them from a large pool human subjects. However, we do not to encourage the substitution of human gestures by synthetic ones, but rather to provide an automated way to lower the need of recruiting a large number of users and subsequent data labeling. Our online application and accompanying web service (JSON RESTful API) is available at https://g3.prhlt.upv.es/.

## References


[1] A. Almaksour, E. Anquetil, R. Plamondon, and C. O'Reilly, "Synthetic handwritten gesture generation using Sigma-Lognormal model for evolving handwriting classifiers," in *Proc. IGS*, 2011.

[2] S. Amini and Y. Li, "CrowdLearner: Rapidly creating mobile recognizers using crowdsourcing," in *Proc. UIST*, 2013.

[3] L. Anthony and J. O. Wobbrock, "A lightweight multistroke recognizer for user interface prototypes," in *Proc. GI*, 2010.

[4] ——, "$N-protractor: a fast and accurate multistroke recognizer," in *Proc. GI*, 2012.

[5] C. Appert and S. Zhai, "Using strokes as command shortcuts: Cognitive benefits and toolkit support," in *Proc. CHI*, 2009.

[6] D. Ashbrook and T. E. Starner, "MAGIC: A motion gesture design tool," in *Proc. CHI*, 2010.

[7] D. Bullock and S. Grossberg, "The VITE model: a neural command circuit for generating arm and articulator trajectories," in *Dynamic Patterns in Complex Systems*, 1988.

[8] B. Caramiaux, N. Montecchio, A. Tanaka, and F. Bevilacqua, "Adaptive gesture recognition with variation estimation for interactive systems," *ACM TiiS*, vol. 4, no. 4, 2014.

[9] S. D. Connell and A. K. Jain, "Template-based on-line character recognition," *Pattern Recogn.*, vol. 34, no. 1, 2000.

[10] M. R. Davis and T. O. Ellis, "The RAND tablet: A man-machine graphical communication device," in *Proc. AFIPS*, 1964.

[11] V. Deepu, S. Madhvanath, and A. G. Ramakrishnan, "Principal component analysis for online handwritten character recognition," in *Proc. ICPR*, 2004.

[12] A. K. Dey, R. Hamid, C. Beckmann, I. Li, and D. Hsu, "A CAPpella: Programming by demonstration of context-aware applications," in *Proc. CHI*, 2004.

[13] M. Djioua and R. Plamondon, "Studying the variability of handwriting patterns using the Kinematic Theory," *Hum. Mov. Sci.*, vol. 28, no. 5, 2009.

[14] J. Galbally, R. Plamondon, J. Fierrez, and J. Ortega-García, "Synthetic on-line signature generation. Part I: Methodology and algorithms," *Pattern Recogn.*, vol. 45, no. 7, 2012.

[15] ——, "Synthetic on-line signature generation. Part II: Experimental validation," *Pattern Recogn.*, vol. 45, no. 7, 2012.

[16] J.-W. Kim and T.-J. Nam, "EventHurdle: Supporting designers' exploratory interaction prototyping with gesture-based sensors," in *Proc. CHI*, 2013.

[17] D. Kohlsdorf and T. E. Starner, "MAGIC Summoning: Towards automatic suggesting and testing of gestures with low probability of false positives during use," *J. Mach. Learn. Res.*, vol. 14, no. 1, 2013.

[18] J. A. Landay and B. A. Myers, "Extending an existing user interface toolkit to support gesture recognition," in *Proc. INTERCHI*, 1993.

[19] L. A. Leiva, "Large-scale user perception of synthetic stroke gestures," in *Proc. DIS*, 2017.

[20] L. A. Leiva, V. Alabau, V. Romero, A. H. Toselli, and E. Vidal, "Context-aware gestures for mixed-initiative text editing UIs," *Interact. Comput.*, vol. 27, no. 1, 2015.

[21] L. A. Leiva, D. Martín-Albo, and R. Plamondon, "Gestures à Go Go: Authoring synthetic human-like stroke gestures using the kinematic theory of rapid movements," *ACM T. Intel. Syst. Tec.*, vol. 7, no. 2, 2016.

[22] ——, "The kinematic theory produces human-like stroke gestures," *Interact. Comput*, 2017, advance online publication. DOI: 10.1093/iwc/iww039.

[23] L. A. Leiva, D. Martín-Albo, and R.-D. Vatavu, "Synthesizing stroke gestures across user populations: A case for users with visual impairments," in *Proc. CHI*, 2017.

[24] Y. Li, "Protractor: a fast and accurate gesture recognizer," in *Proc. CHI*, 2010.

[25] H. Lü, J. A. Fogarty, and Y. Li, "Gesture Script: Recognizing gestures and their structure using rendering scripts and interactively trained parts," in *Proc. CHI*, 2014.

[26] H. Lü and Y. Li, "Gesture Studio: Authoring multi-touch interactions through demonstration and declaration," in *Proc. CHI*, 2013.

[27] D. Martín-Albo and L. A. Leiva, "G3: bootstrapping stroke gestures design with synthetic samples and built-in recognizers," in *Proc. MobileHCI*, 2016.

[28] D. Martín-Albo, R. Plamondon, and E. Vidal, "Improving sigma-lognormal parameter extraction," in *Proc. ICDAR*, 2015.

[29] R. Marzinkewitsch, "Operating computer algebra systems by hand-printed input," in *Proc. ISAAC*, 1991.

[30] D. E. Meyer, J. E. Keith-Smith, S. Kornblum, R. A. Abrams, and C. E. Wright, "Speed-accuracy tradeoffs in aimed movements: Toward a theory of rapid voluntary action," *Atten. and Perform.*, vol. 13, no. 23, 1990.

[31] T. Ouyang and Y. Li, "Bootstrapping personal gesture shortcuts with the wisdom of the crowd and handwriting recognition," in *Proc. CHI*, 2012.

[32] K. Patel, N. Bancroft, S. M. Drucker, J. Fogarty, A. J. Ko, and J. Landay, "Gestalt: Integrated support for implementation and analysis in machine learning," in *Proc. UIST*, 2010.

[33] R. Plamondon, "A kinematic theory of rapid human movements. Part I: Movement representation and control," *Biol. Cybern.*, vol. 72, no. 4, 1995.

[34] R. Plamondon, A. M. Alimi, P. Yergeau, and F. Leclerc, "Modelling velocity profiles of rapid movements: a comparative study," *Biol. Cybern.*, vol. 69, no. 1, 1993.

[35] R. Plamondon and M. Djioua, "A multi-level representation paradigm for handwriting stroke generation," *Hum. Mov. Sci.*, vol. 25, no. 4–5, 2006.

[36] R. Plamondon, C. O'Reilly, J. Galbally, A. Almaksour, and E. Anquetil, "Recent developments in the study of rapid human movements with the Kinematic Theory: Applications to handwriting and signature synthesis," *Pattern Recogn. Lett.*, vol. 35, 2014.

[37] M. Reznakova, L. Tencer, R. Plamondon, and M. Cheriet, "The generation of synthetic handwritten data for improving on-line learning," in *Proc. IGS*, 2015.

[38] D. Rubine, "Specifying gestures by example," *Proc. SIGGRAPH*, vol. 25, no. 4, 1991.

[39] L. D. Spano, A. Cisternino, F. Paternò, and G. Fenu, "GestIT: A declarative and compositional framework for multiplatform gesture definition," in *Proc. EICS*, 2013.

[40] I. E. Sutherland, "Sketchpad: A man-machine graphical communication system," Lincoln Laboratory, MIT, Tech. Rep. 296, 1963.

[41] J. J. D. van der Gon and J. P. Thuring, "The guiding of human movements," *Kybernetik*, vol. 14, no. 1, 1965.

[42] R.-D. Vatavu, L. Anthony, and J. O. Wobbrock, "Gestures as point clouds: a $P recognizer for user interface prototypes," in *Proc. ICMI*, 2012.

[43] P. Viviani and T. Flash, "Minimum-jerk, two-thirds power law, and isochrony: converging approaches to movement planning," *J. Exp. Psychol.*, vol. 21, no. 1, 1995.

[44] J. O. Wobbrock, A. D. Wilson, and Y. Li, "Gestures without libraries, toolkits or training: A $1 recognizer for user interface prototypes," in *Proc. UIST*, 2007.

[45] S. Zhai, P. O. Kristensson, C. Appert, T. H. Anderson, and X. Cao, "Foundational issues in touch-surface stroke gesture design — an integrative review," in *Foundations and Trends in Human-Computer Interaction*, 2012, vol. 5, no. 2.