pEncode: A Tool for Visualizing Pen Signal Encodings in Real-time

Felix CÉARD-FALKENBERG, Konstantin KUZNETSOV,
Alexander PRANGE, Michael BARZ, Daniel SONNTAG
German Research Center for Artificial Intelligence (DFKI)
Saarbrücken, Germany

Abstract. Many features have been proposed for encoding the input signal from digital pens and touch-based interaction. They are widely used for analyzing and classifying handwritten texts, sketches, or gestures. Although they are well defined mathematically, many features are non-trivial and therefore difficult to understand for a human. In this paper, we present an application that visualizes a subset from 114 digital pen features in real-time while drawing. It provides an easy-to-use interface that allows application developers and machine learning practitioners to learn how digital pen features encode their inputs, helps in the feature selection process, and enables rapid prototyping of sketch and gesture classifiers.

Keywords. digital pen, gesture recognition, digital pen features, machine learning

1. Introduction

Over the last years, the popularity of digital pen input and touch-based interaction significantly increased. They are used in various tasks such as gesture-based user interfaces (UIs) [28,14,4,8,15,17,2,9], handwriting recognition [11,10], signature verification [26], cognitive state assessments [22,18,7,25,23,19], and multimodal learning analysis [3,16]. Recent studies have also explored the interpretability of such systems [19,12,29,18,5,13]. Processing digital pen signals typically relies on a set of predefined features that aggregate the raw sensor input, i.e., timestamped 2D coordinates, often with an additional pressure signal from digital pen or touch-screen devices. Although they have a well defined mathematical definition, they lack human interpretability. Understanding the exact meaning of important features and how they relate to individual strokes in a potentially complex sketch remains non-trivial.

With *pEncode*, we present a tool that allows developers to interactively visualize a broad range of pen-based features. The feature values are updated as the drawing progresses. This representation allows a user to discern the peculiarities of each feature and get a feeling about the impact of using different shapes or drawing styles on the feature-based sketch encoding.

pEncode can help developing efficient digital pen and touch-based interfaces, which enable natural human-machine interaction and cooperation.

2. Visualization Application

We implement our visualization tool *pEncode* as an Android application, which supports pen-enabled mobile devices. The main screen of the app provides a drawing space and a list showing a subset of digital pen features. We visualize the real-time evolution of each feature during the drawing process using a progress bar. It shows the current value of the feature and immediately updates it as the drawing evolves (see Figure 1). The values are normalized using min-max scaling. For features that do not have an upper bound, the user can specify the maximum value or let the app estimate it, e.g., based on the largest value in a session. In total, the user can select a subset from 114 features as described in Prange et al. [20][1]. They categorize and implement four feature sets introduced by Dean Rubine [21], Willems and Niels [27], Sonntag et al. [24], and the HBF49 feature set by Delaye and Anquetil [6].

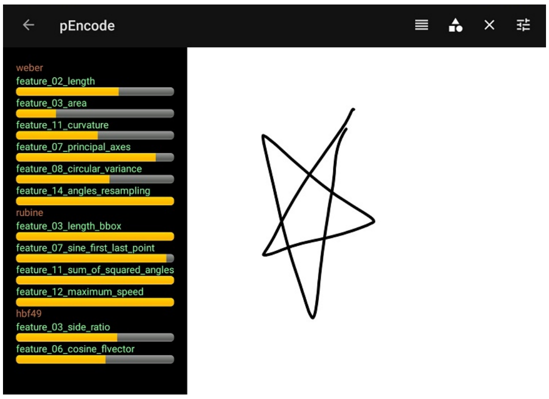

Figure 1.: Screenshot of *pEncode*, showing the drawing space (right) and the feature list (left).

In addition, *pEncode* implements four supervised learning methods for rapid prototyping of gesture recognition or sketch classification models. We integrate the k-nearest neighbors (KNN) classifier, k-means-based classifier, Linear Discriminant Analysis (LDA), and Quadratic Discriminant Analysis (QDA). The user can select a set of features for a model and immediately create a training dataset. For instance, the user could draw a few stars and circles and provide class labels accordingly. After training, each new sketch will be classified using this model, which allows the user to assess the prediction quality and learn about the selected features. To better understand the relevance of individual features, they are rearranged in order of their importance, calculated using permutation importance method [1].

3. Conclusion

We presented the pEncoder[2] tool that helps researchers to learn how digital pen features encode pen input by interactively displaying their values in real time. It enables rapid prototyping of gesture recognition models and provides insights into the importance of individual features.

[1] `https://github.com/DFKI-Interactive-Machine-Learning/ink-features`

[2] Video of the demonstrator: `https://www.youtube.com/watch?v=t80aa2E5jKo`

References

[1] André Altmann, Laura Toloşi, Oliver Sander, and Thomas Lengauer. Permutation importance: a corrected feature importance measure. *Bioinformatics*, 26:1340–1347, 5 2010.

[2] Lisa Anthony and Jacob O Wobbrock. A lightweight multistroke recognizer for user interface prototypes. pages 245–252. Canadian Information Processing Society, 2010.

[3] Michael Barz, Kristin Altmeyer, Sarah Malone, Luisa Lauer, and Daniel Sonntag. Digital pen features predict task difficulty and user performance of cognitive tests, 2020.

[4] Andrew Bragdon, Robert Zeleznik, Brian Williamson, Timothy Miller, and Joseph Jr. Gesturebar: Improving the approachability of gesture-based interfaces. pages 2269–2278, 4 2009.

[5] Mihir Chauhan, Mohammad Abuzar Shaikh, and Sargur N Srihari. Explanation based handwriting verification. *CoRR*, abs/1909.02548, 2019.

[6] Adrien Delaye and Éric Anquetil. Hbf49 feature set: A first unified baseline for online symbol recognition. *Pattern Recognit.*, 46:117–130, 2013.

[7] Peter Drotár, Jiří Mekyska, Irena Rektorová, Lucia Masarová, Zdenek Smékal, and Marcos Faundez-Zanuy. Analysis of in-air movement in handwriting: A novel marker for parkinson's disease. *Computer Methods and Programs in Biomedicine*, 117:405–411, 2014.

[8] Nicholas Gillian and Joseph A Paradiso. The gesture recognition toolkit. *Journal of Machine Learning Research*, 15:3483–3487, 2014.

[9] Tracy Hammond and Brandon Paulson. Recognizing sketched multistroke primitives. *ACM Trans. Interact. Intell. Syst.*, 1, 10 2011.

[10] R Reeve Ingle, Yasuhisa Fujii, Thomas Deselaers, Jonathan Baccash, and Ashok C Popat. A scalable handwritten text recognition system. *CoRR*, abs/1904.09150, 2019.

[11] Dmitrijs Kass and Ekta Vats. Attentionhtr: Handwritten text recognition based on attention encoder-decoder networks. *CoRR*, abs/2201.09390, 2022.

[12] Maciej Marcinowski. Top interpretable neural network for handwriting identification. *Journal of Forensic Sciences*, 1 2022.

[13] Sven Mayer, Valentin Schwind, Huy Viet Le, Dominik Weber, Jonas Vogelsang, Johannes Wolf, and Niels Henze. Effect of orientation on unistroke touch gestures. pages 1–9. Association for Computing Machinery, 2019.

[14] Erin McAweeney, Haihua Zhang, and Michael Nebeling. User-driven design principles for gesture representations. pages 1–13. Association for Computing Machinery, 2018.

[15] David A Mellis, Ben Zhang, Audrey Leung, and Björn Hartmann. Machine learning for makers: Interactive sensor data classification based on augmented code examples. pages 1213–1225. Association for Computing Machinery, 2017.

[16] Sharon Oviatt, Kevin Hang, Jianlong Zhou, Kun Yu, and Fang Chen. Dynamic handwriting signal features predict domain expertise. *ACM Transactions on Interactive Intelligent Systems*, 8:1–21, 7 2018. Publisher: ACM.

[17] Otto Parra, Sergio España, and Oscar Pastor. Gestui: A model-driven method and tool for including gesture-based interaction in user interfaces. *Complex Systems Informatics and Modeling Quarterly*, pages 73–92, 4 2016.

[18] A Parziale, R Senatore, A Della Cioppa, and A Marcelli. genetic programming for diagnosis of parkinson disease through handwriting analysis: Performance vs. interpretability issues. *Artificial Intelligence in Medicine*, 111:101984, 2021.

[19] Alexander Prange, Michael Barz, Anika Heimann-Steinert, and Daniel Sonntag. Explainable automatic evaluation of the trail making test for dementia screening. Association for Computing Machinery, 2021.

[20] Alexander Prange, Michael Barz, and Daniel Sonntag. A categorisation and implementation of digital pen features for behaviour characterisation, 2018.

[21] Dean Rubine. Specifying gestures by example. *SIGGRAPH Comput. Graph.*, 25:329–337, 7 1991.

[22] Daniel Sonntag. Interakt - a multimodal multisensory interactive cognitive assessment tool. *CoRR*, abs/1709.01796, 2017.

[23] Daniel Sonntag. Interactive cognitive assessment tools: a case study on digital pens for the clinical assessment of dementia. *arXiv preprint arXiv:1810.04943*, 2018.

[24] Daniel Sonntag, Markus Weber, Alexander Cavallaro, and Matthias Hammon. Integrating digital pens in breast imaging for instant knowledge acquisition. *Ai Magazine*, 35:26–37, 4 2014.

[25] William Souillard-Mandar, Randall Davis, Cynthia Rudin, Rhoda Au, David Libon, Rodney Swenson, Catherine Price, Melissa Lamar, and Dana Penney. Learning classification models of cognitive conditions from subtle behaviors in the digital clock drawing test. *Machine Learning*, 102, 4 2016.

[26] Rubén Tolosana, Rubén Vera-Rodr{ı́}guez, Carlos Gonzalez-Garcia, Julian Fiérrez, Aythami Morales, Javier Ortega-Garcia, Juan-Carlos Ruiz-Garcia, Sergio Romero-Tapiador, Santiago Rengifo, Miguel Caruana, Jiajia Jiang, Songxuan Lai, Lianwen Jin, Yecheng Zhu, Javier Galbally, Moisés D{ı́}az Cabrera, Miguel Ángel Ferrer, Marta Gomez-Barrero, Ilya A Hodashinsky, Konstantin S Sarin, Artem Slezkin, Marina Bardamova, Mikhail Svetlakov, Mohammad Saleem, Cintia Lia Szücs, Bence Kovári, Falk Pulsmeyer, Mohamad Wehbi, Dario Zanca, Sumaiya Ahmad, Sarthak Mishra, and Suraiya Jabin. Svc-ongoing: Signature verification competition. *CoRR*, abs/2108.06090, 2021.

[27] Don Willems and Ralph Niels. Definitions for features used in online pen gesture recognition. *Nijmegen Institute for Cognition and Information Radboud University Nijmegen, Nijmegen, The Netherlands*, 2008.

[28] Jacob O Wobbrock, Andrew D Wilson, and Yang Li. Gestures without libraries, toolkits or training: a $1 recognizer for user interface prototypes. In *Proceedings of the 20th annual ACM symposium on User interface software and technology*, pages 159–168, 2007.

[29] Jin-Wen Wu, Fei Yin, Y Zhang, Xu-Yao Zhang, and Cheng-Lin Liu. Graph-to-graph: towards accurate and interpretable online handwritten mathematical expression recognition. volume 35, pages 2925–2933, 2021.