What Auto Completion Tells Us About Sketch Recognition

Ozan Can Altıok, Kemal Tugrul Yesilbek, Tevfik Metin Sezgin

Intelligent User Interfaces Laboratory, Koç University, Turkey

Abstract

Auto completion is generally considered to be a difficult problem in sketch recognition as it requires a decision to be made with fewer strokes. Therefore, it is generally assumed that the classification of fully completed object sketches should yield higher accuracy rates. In this paper, we report results from a comprehensive study demonstrating that the first few strokes of an object are more important than the lastly drawn ones. Once the first few critical strokes of a symbol are observed, recognition accuracies reach a plateau and may even decline. This indicates that less is more in sketch recognition. Our results are supported by carefully designed computational experiments using Tirkaz et. al.'s sketch auto completion framework on the dataset of everyday object sketches collected by Eitz et. al..

Categories and Subject Descriptors (according to ACM CCS): I.5.0 [Pattern Recognition]: General—H.5.2 [Information Interfaces and Presentation]: User Interfaces—Input devices and strategies

1. Introduction and Related Work

Sketching is an incremental process. Users add strokes to the drawing surface one at a time. This online aspect of sketching makes it amenable to auto-completion – online predictive recognition with partial data. Although the concept of auto-completion has been studied and exploited for other online processes – most notably for typing – such systems for sketching have started to appear only recently.

Tirkaz et al. introduced an elaborate system for auto completion and evaluated it on two relatively simple databases [TYS12]. In other work, Costagliola et. al. investigated the concept of auto completion with a focus on usability [CDRF13]. They reported that it improves usability of sketch-based systems by reducing the time required for sketching through accurate interpretation of partially drawn sketches. Other pieces of work have also contributed methods for auto completion [CDRF14] and [SW14]. However, existing systems in the literature have been evaluated with relatively simple data sets and they haven't examined the relationship between the completeness of classified sketches and the classification accuracies of these.

In this paper, we present a detailed evaluation of the sketch auto completion framework proposed by Tirkaz et. al. [TYS12] on the data set collected by Eitz et. al. [EHA12] to shed light on these issues. The Eitz database contains sufficiently complex symbols, and the overall recognition rates reported for this database are rather low. Hence, it serves as good database for a thorough evaluation.

Our experiments show that, contrary to our expectation, we do not always observe a progressive improvement in recognition accuracies as symbols get closer to being "complete." While the sketches are getting closer to their completed ones, after a while, the recognition accuracies become steady and sometimes even decline. This suggests that strokes that are added later on in the drawing process, which mostly serve as optional ornaments and details, don't necessarily lead to a better classification performance. Earlier strokes, which usually map out the rough boundaries of symbols appear to be sufficient for most classification purposes.

2. Method

We used the methodology of auto completion described by Tirkaz et. al.. The framework has 3 main steps. In the first step, partial sketches are generated from given full sketches through an order that humans follow while drawing. In the second step, both full and partial sketches are clustered using constrained K-means clustering [WCR*01] with that in each cluster there is at most one class having full sketches. Finally, an SVM [HCL*03] classifier is trained for every cluster containing instances belonging to different classes.

In this work, we used the everyday object sketch dataset collected by Eitz et. al. [EHA12]. To represent the sketches in feature domain, IDM feature representation [OD09] was used as it has recently shown the best recognition performance [TAS10]. Example sketches from the dataset are presented in Figure 1.

3. Experiment

We conducted repeated experiments to evaluate the auto completion framework. In order to keep the experiment durations manageable, the experiments were run with the following constraints:

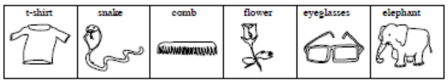

Figure 1: *Some sample sketches from Eitz data set [EHA12]*

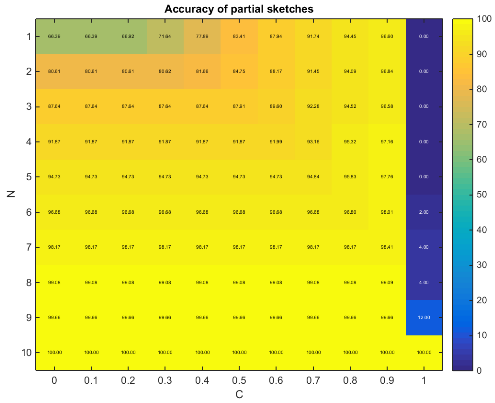

Figure 2: *Partial sketch prediction accuracy rate*

1. We used the first 100 classes.
2. Selected classes were randomly divided into 10 folds, each containing unique classes (spanning all the first 100 classes). In each experiment, one of the folds was selected.
3. We selected the first 50 instances of every class.
4. For testing a fold of 10 classes, 5-fold cross validation was used.

4. Results

In the experiments, we kept two different free parameters belonging to the auto completion framework:

1. N: Number of classes to be present to the user (the most confident ones)
2. C: Threshold for the sum of confidence rates deciding whether the classes should be presented to the user.

In Figure 2, for different combinations of N and C values, mean of the accuracies acquired from partial sketches is presented. In Figure 3, the completeness of the sketches versus the mean and standard deviations of the all experiments' accuracies are presented.

5. Discussion

As shown in Figure 2, auto completion achieved decent performance even for complex, everyday object sketches. Considering the achievement in usability improvement, we can say that auto completion can greatly help users to use sketch-based interfaces, especially in applications where the number of classes supported is large.

As Figure 3 suggests, the first few strokes play the most important role. After the first few strokes were drawn, regardless of the value of C, after a while, completeness of the sketches went up and the recognition performances became steady, sometimes declined. When N was defined as 1, the impact of the first strokes was observed the best.

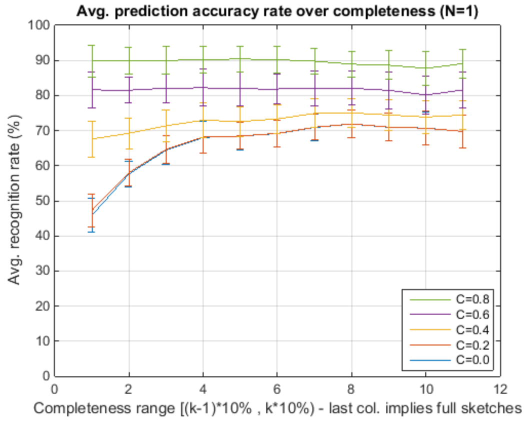

Figure 3: *Prediction accuracy rates of the sketches over the completeness rate, for some C values ($N = 1$, except the last column, the kth column implies the range $[(k-1)*10\%, k*10\%)$)*

6. Acknowledgment

The authors gratefully acknowledge the support and funding of TUBITAK under grant numbers 113E325 and 113E059.

References

[CDRF13] COSTAGLIOLA G., DE ROSA M., FUCCELLA V.: Investigating human performance in hand-drawn symbol autocompletion. In *Systems, Man, and Cybernetics (SMC), 2013 IEEE International Conference on* (2013), IEEE, pp. 279–284. 1

[CDRF14] COSTAGLIOLA G., DE ROSA M., FUCCELLA V.: Recognition and autocompletion of partially drawn symbols by using polar histograms as spatial relation descriptors. *Computers & Graphics 39* (2014), 101–116. 1

[EHA12] EITZ M., HAYS J., ALEXA M.: How do humans sketch objects? *ACM Trans. Graph. 31*, 4 (2012), 44–1. 1, 2

[HCL*03] HSU C.-W., CHANG C.-C., LIN C.-J., ET AL.: A practical guide to support vector classification. 1

[OD09] OUYANG T. Y., DAVIS R.: A visual approach to sketched symbol recognition. 1

[SW14] SCHMIDT M., WEBER G.: Prediction of multi-touch gestures during input. In *Human-Computer Interaction. Advanced Interaction Modalities and Techniques*. Springer, 2014, pp. 158–169. 1

[TAS10] TUMEN R. S., ACER M. E., SEZGIN T. M.: Feature extraction and classifier combination for image-based sketch recognition. In *Proceedings of the Seventh Sketch-Based Interfaces and Modeling Symposium* (2010), Eurographics Association, pp. 63–70. 1

[TYS12] TIRKAZ C., YANIKOGLU B., SEZGIN T. M.: Sketched symbol recognition with auto-completion. *Pattern Recognition 45*, 11 (2012), 3926–3937. 1

[WCR*01] WAGSTAFF K., CARDIE C., ROGERS S., SCHRÖDL S., ET AL.: Constrained k-means clustering with background knowledge. In *ICML* (2001), vol. 1, pp. 577–584. 1