Procedural Analysis of a sketching activity: principles and applications

Ney Renau-Ferrer, Céline Rémi
IRISA/Université Rennes 2, LAMIA/Université des Antilles et de la Guyane
Ney.renau-ferrer@irisa.fr, celine.remi@univ-ag.fr

Abstract

The advent of pen-based interfaces has led to the development of many applications concerning the analysis of a drawn shape. But a drawing is not only a shape. It is also all the gestures which constitutes the drawing activity, as well as their planning. Pen-based interfaces give us the possibility to analyze this dimension of sketching. We call this kind of analysis: procedural analysis. In this article, we will show how to identify the privileged drawing procedures of users by identifying a constant pattern despite a huge within-drawer variability in drawing procedures. After stating how to extract this constant pattern, we shall show how we can use those results into a drawing-based user identification method.

1. Introduction

The advent of Pen-based Interface stood out as a logical continuity in the history of handdrawing. Indeed, it allows the automation or assistance by computer of a lot of processes, such as technical drawing for example. Consequently, numerous works apply to analyze shapes sketched by a drawer using Pen-based interfaces. Those kinds of approaches include for example shape and symbols recognition, sketch beautification, Computer Assisted Design... Though, analysis of a handdrawing activity can be operated from another point of view. We can try to analyze the sketching activity which led to the production of the shape rather than to analyze the produced shape. We call the analysis of this dimension of sketching: “procedural analysis”. Online acquisition of drawing activity by devices such as graphic tablets provides access to raw data about pen trajectory over time. Those data could allow automating those types of treatments. We focus here on freehand geometrical sketches. We shall begin by presenting a state of the art where we will show that even if procedural analysis of those kind a shapes finds a lot of applications, for example in order to assess psychological disorder, the attempts for automation are so far too few and the existing approaches lack of genericity.

Then, we shall present a method of extraction and structural modeling of the procedure used to produce a shape. We will also propose a way to compute the dissimilarity between procedures.

Afterward we will define some notions which seem important to us when dealing with procedure analysis. Indeed, one has to make a difference between what we have chosen to call “raw procedure” and “intrinsic procedure”. Most procedural analysis applications rely on the fact that each stroke of the sketch plays a role in the shape construction. Though we will see that most of the time, the raw observed procedure is noisy. Sometime this noise reveals information about user's drawing skills but it can as well prevent one to perform treatments relying on fine procedure analysis. In this article, we will present what is a raw and an intrinsic procedure and we will seek to find a way to discover the intrinsic procedure of any user when drawing a usual geometric shape. Using those results, we will show how we can determinate the favorite procedure of a user for the drawing of a shape. Finally, in order to test the efficiency of our methods, we will show an example of treatments that uses our results to perform a procedure based drawer recognition.

2. State of the art

The analysis of the drawing activity based on the observation of the strategy of realization of a shape is used since a long time in psychology for the assessment of neuropsychological disorders or learning disabilities, or to assess personality[1][6][10]. Unfortunately, those treatments are most of the time performed off-line by a human. Due to a lack of

978-0-7695-4774-9/12 $26.00 © 2012 IEEE
DOI 10.1109/ICFHR.2012.255

methods allowing automatic extraction and modeling of the drawing activity, the human observer must take note during the observation of the sketching activity in order to have information about the execution process involved. However, it has been demonstrated that useful information for some diagnoses can be obtained by observing these dimensions of the sketching activity [6]. According to [2], works of this field can be divided in two categories: Those concerning the analysis of dynamic features and those concerning the analysis of procedural features.

Dynamic analysis is about assessing overall and dynamic properties such as pressure on the pen, sketching speed, etc. Procedural analysis studies the way a drawer produces a drawing from the point of view of the temporal, geometrical and structural properties of the parts composing the sketch. In other words, procedure analysis is about the planning of the gesture while dynamic analysis is about motor control of the gesture. Despite the fact that the stakes of the procedural analysis are many, there is very little work in this field and existing methods present a lack of genericity. In the field of procedural analysis, Guest [2] uses a constructional sequence analysis in order to diagnose Visuo Spatial Neglect. For that he performs an analysis on sequences representing the order of appearance of the parts of the sketch. This method shows the interest and application of procedural analysis, though it presents some gaps: it is dedicated to a particular shape and relies on the assumption that the drawer will draw this exact shape. Also it is sensitive to noise and to inherent issues of free handrawing presented in [7]. Procedural analysis is also used by Sezgin [9] in order to improve a shape recognition process. He uses Hidden Markov Models to build an observation sequence describing ordering and orientations of strokes during the sketching. These observation sequences are then compared to models representing users' favorite sketching style. This approach is efficient for the purpose of drawer-dependent shape recognition. It could however not be used as a generic approach for procedural analysis as all aspect of procedures are not taken into account. Particularly, it does not take into account the sense according to which strokes are drawn. Furthermore, it does not handle events such as when the user lifts the pen or when he puts it on the sheet.

3. Procedural description

We are interested in the analysis of the procedure involved in a sketching activity, according to the definition of the procedure stated in [8], that is gesture planning. To operate our processing, we first perform a segmentation of the sketch. The segmentation method we use is presented in [7]. This segmentation step provides us raw data such as: the temporally ordered list of the geometrical primitives that approximate each part of the sketch, as well as the temporally ordered list of the feature points of the sketch. A feature point is either a break point (when the direction of the stroke changes) or a pen up/down point.

Our procedural description will use two sequences. The first sequence, named O, represents the orientation and direction according to which each part of the sketch delimited by two feature points are sketched. The second sequence, named P, represents the type of each feature point: either break point or pen up/down.

3.1 Extraction of sequence O

For the construction of this sequence O, we will classify each part of the sketch according to their orientation: Vertical, Horizontal, Oblique Left or Oblique Right. But also according to the sense according to which they are done: Positive or Negative. For that we will observe the oriented angle between each part of the sketch and the X-axis as shown in figure 1, where arrows represent the sense of the part of the sketch. Then we divide the space into 8 areas of 45°. If the angle is between -22.5° and +22.5°, it is labeled as a Positive Horizontal: H+. An angle between +157.5° and -157.5° indicates that the user as drawn horizontally from right to left, so the stroke will be labeled as H-., and so on.

3.2 Extraction of sequence P

We take into account for this article the types of feature points: Breakpoint and Pen Up/Down, available as input data from a segmentation step. Breakpoints are points where we can observe a discontinuity in curvature, while Pen Up/Down are points beginning and ending strokes (so when the user lifts the pen or when he puts it on the sheet).

3.3 Dissimilarity Measure

As our procedural descriptor is a set of sequences, we propose to use a string metric such as the Levenshtein distance [5]. This distance is defined as the minimum number of edits needed to transform one string into another one, with allowable edit operations like insertion, deletion, or substitution of a single character. The cost of insertion and deletion will be 2.

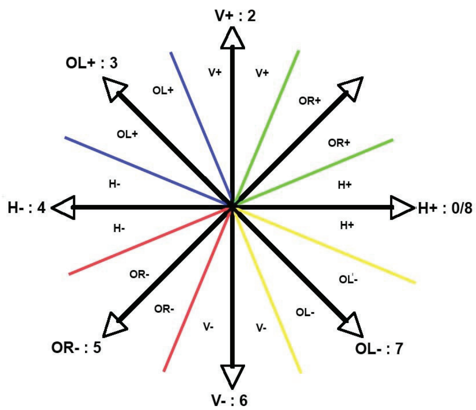

Figure 1 : Orientations

Then we will define a specific and variable cost for substitution. The substitution cost of the sequence O will be calculated using a representation of this sequence based on the Freeman code. To each sense and orientation of strokes will correspond a value included between 0 and 7, as shown on figure 1. The cost of substitution then becomes the absolute value of the difference between the Freeman code of each component: $Ds = \| f1\text{-}f2\|$

Moreover, we have to take into account that every component has 2 distances to 0: one following the clockwise sense and one following the counter-clockwise sense. For example, difference between H+ (0) and OL-(7) should be equal to 1. To overcome this limitation, the cost of a substitution involving a 0 will then be: $Ds = \min(\|0\text{-}f2\|,\|8\text{-}f2\|)$

Using this distance allow us to minimize the problem of this approach: the computation of the O sequences relies on the use of a threshold to determine the direction of a stroke. Unfortunately it can happen that a user draws a clearly oblique segment when trying to draw it horizontally. This induces a bad recognition of the procedure. Using a dissimilarity based on the Freeman code only for substitution allow us to state that :

“If two O sequences, for the same shape, have an edit distance of 1 then the two procedures are exactly the same, as a difference of 1 can only be obtained by the misclassification of one of the stroke composing the shape”.

So, two procedures will be considered as equals if they have a distance of 0 or 1. If a user poorly has drawn one of the strokes composing the sketch, this won’t have any bad repercussion.

4. Procedural Consistency

The genesis of the graphical skills [3] consists in several stages: 1, scrawl and birth of the first shapes, 2, the awareness of the shape and 3, mastery of drawing the shape.

In this final stage, by repetition, sketching becomes an automated gesture which reflects a certain level of "expertise" in the drawing of a shape. So it seems logical to think that the gestures involved in the drawing of a shape, for which the production is an automatism, will always be the same. This offers two possibilities:

- If a drawer controls the drawing of a shape, his procedure, because it is automated, it must be immutable. The procedure of this drawer will then be described as "consistent".
- If it is not possible to identify a preferred procedure in the drawing of a shape by a drawer, it informs us that this drawer does not master the production of this shape.

From these assumptions, we developed the definition of a feature which we call « procedural consistency ». A user's procedural consistency for a shape is equal to the highest utilization rate observed for the procedures that he uses when drawing this form. A user who always uses the same procedure for the drawing of a shape will have a procedural consistency index equal to the maximum observable utilization rate, so 100%. We shall then say that this user has a strong consistency for the drawing of this shape. On the contrary, if a user uses several different procedures to draw the same shape, his consistency will be qualified as weak.

5. Raw procedure / intrinsic procedure: discovering a favored procedure

One of the objectives of this paper is to show the efficiency of our approach of procedural analysis by applying it to the drawer recognition. Of course, the identification of a user by analyzing the way he draws a shape will be possible only if we are able to identify his favored procedure for drawing a shape and if this preferred procedure is strongly consistent.

In order to validate our first intuition that when sketching an usual form, an experienced drawer will most of the time use the same procedure (ie: he will have a high consistency rate), we performed some tests using the Hhreco Dataset [4] in which each of the 19 experimented users were asked to draw at least 30 examples for each shape. We extracted the procedural description and then we calculated their procedural consistencies for the drawing of three shapes: square, parallelogram and pentagon.

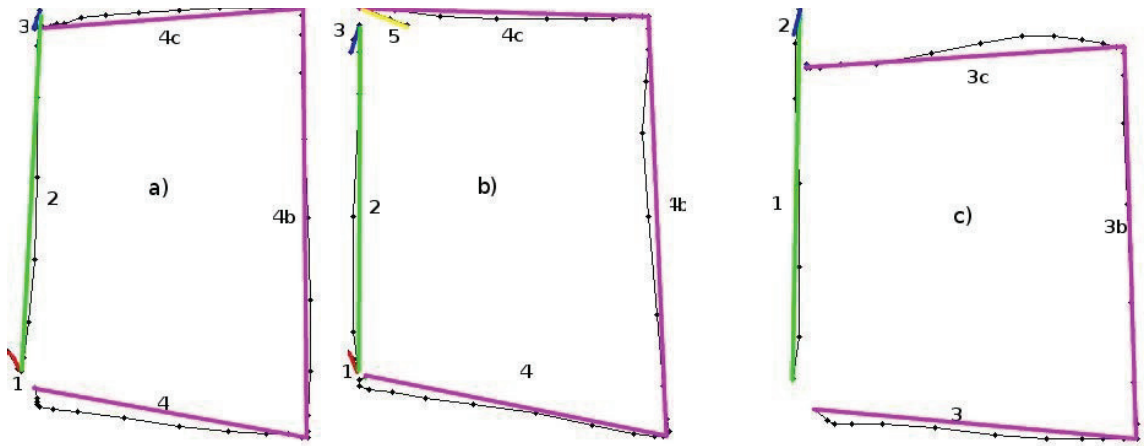

Figure 2: 3 squares drawn by the same user

5.1. Consistency of raw procedures

The procedural consistencies obtained on raw procedural descriptions appeared to be weak (table I). Even if some users may have an acceptable procedural consistency (for example one user has a consistency of 86% for the parallelogram), the average value of 60.37% for the square, for example, means that generally, when drawing 30 squares, users will use their preferred procedure only on 18 of them. However, with a more detailed analysis of those results we found out that the within-drawer variability observed on raw procedures involved to draw an usual shape was usually induced only by significantly smaller strokes than the others of the shape. So, from a strictly axiomatic standpoint, they do not participate in the construction of the shape.

Table 1 : Consistencies on raw procedures

	Square	Parallelogram	Pentagon
Max	83.33%	86%	60%
Min	23.33%	32%	3.33%
Average	60.37%	58.91%	32.09%

Indeed, as shown in [11], when drawing a stroke, drawers tend to miscontrol their movement when they start a stroke or when they try to reach an end point, so they generate artifacts in the sketch at ends or beginnings of strokes. To illustrate this phenomenon, let us take a look at the case of the production of the three squares presented in figure 2. Indeed, the raw procedure varies each time: as we can see, stroke number 5 is present in square (b) but not present in square (a) or in square (c). Also, stroke number 1, which is the first stroke for square (a) and (b), is not present in square (c).

Table 2 : Results of procedural noise filtering

	Square	Parallelogram	Pentagon
Nbr of sketches	602	603	607
Correctly filtered	97.34%	96.52%	99.18%
Over-filtration	1.99%	3.3%	0.32%
Under-filtration	0.67%	0.16%	0.49%

So finally, we get three different sequences. Nevertheless, these different sequences only differ on small and inconstant strokes. Moreover they do not participate in the definition of the shape (ie: they are not needed for the shape construction) but rather seems to be what we have chosen to call « procedural noise ». In truth, in these three sketches, if we filter the procedural noise, the four parts constituting the square and participating in its definition from a geometrical point of view are clearly identifiable and their order of appearance follows a constant pattern. Consequently, if we disregard the procedural noise, the three procedures are strictly the same.

5.2 Procedural noise filtering

In a sketch, all the small artifacts composing the procedural noise share only one characteristic: they are significantly small. Therefore, we propose to filter the procedural noise by eliminating strokes whose length is less than a threshold. Of course we cannot just fix an empirical threshold as it should vary depending on shapes class. An average-based filtering also seems irrelevant. Indeed, the ratio: average length of strokes over length of relevant strokes would also vary depending of the class of the shapes, not to mention that segmentation issues may also impact it. The number of points composing the strokes cannot be used either since this data depends on both acquisition method and user's drawing speed. In order to have an adaptive threshold to decide whether or not a stroke is "small", we take into account the relative length of strokes with regards to the length of the longest stroke of the sketch. Every stroke whose length is under 35% of the length of the longest stroke will be considered as noise and thus will be filtered. Results of this filtering are presented in table II. The "correctly filtered" row is the rate of shapes for which all and only the procedural noise has been eliminated. The over filtration rate represents the percentage of shapes for which, in addition to the noise, we have also erase strokes that should have been kept. Finally, under filtration rate is the percentage of sketches for which some procedural noise remains.

Table 3 : Consistencies with procedural noise filtering

Shape	Square	Parallelogram	Pentagon
Average	98.8%	97.29%	94.70%
Gain	+63%	+65%	+195%

5.3. Consistency of Intrinsic procedures

We extract procedures again for all users of the dataset using the procedural noise filtering described above. We call "intrinsic procedure" the procedure obtained after filtering of a raw procedure. The consistencies we obtain are shown in table 3. Consistency rates proved to be much more interesting than those obtained by raw procedure analysis. However, they can sometime be unsatisfactory: 9 of the 57 scores are under 90%. Non-maximal consistencies (when drawing is an automatism) can be due to Over-filtration, Under-filtration or it can result from Segmentation issues. But most of the time it is due to the fact that users are sometimes going slightly diagonally when trying to draw horizontally for example. This fact is related to user's imprecision.

6. Drawer Identification

As we have seen, once reached a certain degree of control, the drawer acquires an automation of the gesture. Moreover, the procedural noise filtering that we have developed provides a good consistency of intrinsic procedures. Given that, we should now show how we can recognize a drawer by studying the sequences O and P of his intrinsic procedure. In our process, we take into account the finite number of possible procedures for the drawing of a shape. Thus, several users can share the same procedure for the drawing of a shape. The more the number of users is big and the more the complexity of a figure is low, the bigger will be the chances of confusion. This limitation is first lessened by the combined use of O and P sequences. Indeed, different drawers can share the same order of appearance of the various strokes composing the shape. However they are susceptible to handle the pen differently. For example, in our database, we have observed that users 4 and 15 share exactly the same O sequence when drawing a square, however, while user 4 never lift the pen, user 15 always lift it after the first stroke and before the last one. This limitation can also be lessened by increasing the number of figures involved in the recognition process. Indeed, if the chances are great that two users use the same procedure for drawing a shape, there is less chance that they share the same procedure for the drawing of several shapes. In other words, the greater the number of shapes involved in the recognition is, the more we reduce the chances to find multiple users sharing the same procedures. However, the less we use shapes, the more practical the system will be. Also, we have to keep it mind that shapes must be sufficiently common. Indeed, if a Great Stellated Dodecahedron is complex enough to expect that we won't find 2 users sharing the same procedure; its high complexity will prevent us to find any user for which the drawing of this shape is an automatism. To compromise, we decided to test our method with 3 classes of shape: square, parallelogram and pentagon. We have tested this approach on the HHreco dataset [4]. At first, we have separated the data to create a test set and a training set. For each shape and each user, we have randomly selected 4 sketches for the training set, all the remaining sketches belongs to the test set. We extract intrinsic procedures involved by users on those 4 shapes. For each user, the procedure presenting the highest consistency rate is learned as their favorite procedure. Using the test set, we have then generated all possible combinations of a Square-Parallelogram-Pentagon sequence for each user, so at least 17,576 situations by user. We will try to recognize users using the following process:

We extract the procedure used by the drawer to produce a square and we compare it, using our dissimilarity measure, to all the procedures we have learned on the training set. If the one with the smallest distance belongs to the good user, then we count a good recognition. If several procedures are tie, we repeat the process with the parallelogram and we compare it to the learned procedures belonging to users that were tie. If several procedures are tie again, we repeat the process with the pentagon. Results for this step are presented in table 4.

Table 4 : recognition rates

User	Recognition Rate	User	Recognition rate
1	100%	8 to 14	100%
2	96.15%	15	95.56%
3 to 6	100%	16 to 18	100%
7	69%	19	80.77%
Average : 96.92%			

7. Discussion

As we can see, the recognition rates are pretty good. Even for users with a low consistency. For example User2's consistencies are only 63.33% for the square and 70.97% for the pentagon, but his recognition rate is 100%. This is due to the fact that a low consistency rate does not reflect only the use of several construction strategies. It is also an indicator of the precision of a drawer's sketching gestures and an indicator of the consistency of this precision. Our approach for the modeling of the procedure and our dissimilarity measure allow the method to be strong against the drawer's lack of precision. The method is also robust, to some extent, against filtering or segmentation issues. Those issues share a common point: they both can lead to an incomplete or to an overloaded description. Thanks to our dissimilarity measure, in most case, an incomplete or overloaded procedure is still closer to its correct form than it is to any other learned one. For instance, user 8 produced shapes presenting a lot of noise and defects. Thus, about 27% of the procedures extracted from his parallelograms production are incomplete (or overloaded). However, our system can still recognize him with 100% accuracy.

8. Conclusion

After presenting a method of extraction and modeling of the procedure used by a drawer for the construction of a geometrical shape, we proposed a feature for the evaluation of its degree of acquisition of the graphical skills leading to this realization: the procedural consistency. We have shown how to differentiate two types of procedure: raw and intrinsic procedure. By implementing a method for the filtering of the raw procedure, we were able to determine that the intrinsic procedure we obtain can be considered as a favored procedure, for the construction of a shape, when its consistency is high. Finally, to show the efficiency of the notions and methods that we presented, we showed that it was possible to use our procedural analysis approach in order to recognize a user by the observation of his sketching activity with an accuracy of 97%. For future works, we will have to extend our tests to more drawers and to confront our methods to other types of shapes.

References

[1] Bouaziz S. and Magnan A., Contribution of the visual perception and graphic production systems to the copying of complex geometrical drawings: A developmental study, Cognitive Development, Vol. 22, 2007, pp5–15.

[2] Guest R., Chindaro S., Fairhurst M. C. and Potter J. M., "Automatic classification of hand drawn geometric shapes using constructional sequence analysis", in Proc. DAR 2003, pp. 990

[3] Hagenmuller S., Idir-Siegler S., Richert D., Riff E., "Le graphisme à l'école maternelle", IEN Mulhouse1, http://www.crdpstrasbourg.fr/cddp68/maternelle/docdef/.

[4] Hse H., Shilman M., Newton A. R. "Robust Sketched Symbol Fragmentation using Templates", in Proc. IUI 2004, pp. 156

[5] Levenshtein V.I, « Binary codes capable of correcting deletions, insertions, and reversals », Soviet Physics Doklady, Vol. 10, 1965, pp. 707-710.

[6] Muir J.J., Kirk L.C., Casey S.C., Morris M.K., Morris R. D., Neuropsychological predictors of constructional abilities: Drawing and assembling, Archives Of Clinical Neuropsychology, Vol. 14, 1999,pp734-735.

[7] Renau-Ferrer N., Remi C., Automatic analysis of online-sketch based on use of local descriptors.,In Proc IGS 2009, pp. 60-63.

[8] Renau-Ferrer, N., Remi, C. Analyse procédurale : un nouveau paradigme pour l'analyse de tracés manuscrits, In Proc. IHM 2011, Article 16 , pp. 93-96

[9] Sezgin T. and Davis R., "HMM-based efficient sketch recognition", In Proc. IUI 2005, pp.281.

[10] Van Sommers P., A system for drawing and drawing-related neuropsychology,Cognitive Neuropsychology, Vol. 6, 1989, pp117– 164.

[11] Woch A., Plamondon R., "Using the Framework of the Kinematic Theory for the Definition of a Movement Primitive", Motor Control, vol 8, pp.547-557, 2004.