Poet, R., and Renaud, K. (2007) A mechanism for filtering distractors for graphical passwords. In: International Graphonomics Society (IGS2007), 11-14 Nov 2007, Melbourne, Australia.

http://eprints.gla.ac.uk/71377

Deposited on: 22 November 2012

# A Mechanism for Filtering Distractors for Graphical Passwords

Ron POET [a] and Karen RENAUD [a]

[a] *Department of Computing Science (University of Glasgow)*
*University Avenue, Glasgow G20 8QQ, Scotland*
ron@dcs.gla.ac.uk, karen@dcs.gla.ac.uk

**Abstract.** Graphical passwords hold some promise in a world of increasing numbers of passwords, which computer users are straining to remember. However, since these kinds of mechanisms are a fairly new innovation many design and implementation problems exist which could well prevent these mechanisms from realising their full potential. This paper addresses the problem of choosing distractors for graphical passwords. It is important that the distractors should not be too similar to the target image, since that could cause confusion and seriously reduce the potential for enhanced memorability and usability. We cannot use the obvious approach of screening out confusing or similar images by means of classification because images are difficult to categorise unambiguously and also because such classification is time-intensive and therefore does not scale. The mechanism of interest for this paper makes use of user-produced doodles, which makes distractor choice by means of categorisation even more difficult. In this paper we present an algorithm that automatically recognises pertinent features in target doodles and then identifies similar doodles in order to disqualify these doodles as distractors for those targets within a particular challenge set. This algorithm produces few false negatives, which makes it particularly suitable for use in this domain.

## 1. Introduction

In a world increasingly controlled by computer applications, the computer user regularly has to enter secret codes in order to gain access to physical and interactive spaces. There is an expectation and demand for users to remember increasing numbers of these passwords, and because of human memory limitations, the user will either choose weak passwords, use the same password for all systems, or write the passwords down (DeAngeli et al, 2005). This behaviour undermines the overall security of the system since an intruder can often gain access to many accounts by means of one particular account protected by a weak password. The problem is not the *password* per se, but rather the sheer abundance of passwords.

Researchers have been trialing various alternatives to passwords to alleviate this situation, and many of these are based on the well-known fact that it is easier for people to recognise than to recall. Hence alternative systems will often ask users to *recognise* their secret code and point it out, rather than *recall* and enter their password without any assistance. An example of such as system is DynaHand (Renaud & Olsen 2007). Most recognition-based authentication will make use of images, since people tend to remember pictures more successfully than alphanumeric strings.

One type of graphical authentication mechanism that offers some potential is the visuo-biometric mechanism, which records a user's handwriting or hand-drawn doodle at enrolment, which the user is then required to identify amongst a group of distractor doodles at authentication (Renaud, 2005). This mechanism has been used with some success for an extended period but one of the problems that is emerging as the system scales is that some users are being confused by their doodle being too similar to distractor doodles. Hence there is a need for a scalable mechanism to filter distractors so that such confusion does not occur.

Graphical systems that use representational images sometimes deal with this problem by classifying all images by using meta-tags and then using distractors from different semantic groups as distractors. This approach is bound to fail for doodles since many doodles are indeterminate in origin and impossible to classify. This paper presents an algorithm we developed specifically to filter out doodles that could cause confusion.

Our algorithm is not perfect and may produce both false positives and false negatives. A false positive occurs when two doodles that a human would classify as different are classified as similar by the algorithm. Conversely, a false negative occurs when two doodles that a human would classify as similar are automatically classified as different. These two types of errors have different consequences when the algorithm is used to accept or reject distractors. If we were comparing the real password with a potential distractor, then a false positive would result in a perfectly good distractor not being chosen. This is acceptable, since there are many other good potential distractors to choose from. On the other hand, a false negative would allow a potentially confusing distractor to be chosen. This is a much more undesirable state of affairs, and so we have aimed to reduce the number of false negatives as much as possible, even at the expense of more false positives.

## 2. The Automatic Classification Algorithm

Each doodle is analysed in three different ways to produce three different scores. When two doodles are compared, the differences between their scores is calculated and then combined to form a weighted sum. If this weighted sum is below a threshold then they are considered similar. The three different measures are:

number of separate white regions; number of separate black regions and number of joins between lines.  The white and black regions are considered in the connectivity section, while the calculation of joins is discussed in the joins section.

## 2.1. Connectivity

A very simple way of analysing a doodle is to look at the connectivity of the lines and white space that make up the picture.  There are two ways in which this can be done.  Firstly, we can count the number of distinct white regions and secondly the number of distinct black regions.  This will give us two different measures.  The number of different black regions will count the number of times the pen was lifted, which is a rough count of the number of different features in the doodle.  On the other hand, the number of different white regions measures the style of drawing.  A smiley face with closed eyes and ears will be different from one with dots for eyes, or indeed no eyes.

Calculating that number of different black and white regions is straightforward.  We used a 4-connect flood fill algorithm.  The procedure for counting white regions is similar, and in fact we implemented it by starting with a negative copy of the image, one where black pixels became white and white ones black, and then counting black regions as before.

### 2.1.1. Fattening

One feature that we noticed when studying the doodles in our experimental base was how many features made of several lines were constructed. In some cases, the lines did not quite meet, or the lines forming enclosed regions did not completely join up.  In other cases, the lines went slightly too far and protruded on the other side of a join.  This is illustrated in Figure 1.  The first problem was corrected by fattening the image before counting regions and joins, while the second problem was corrected by shaving the image, removing short lines, before counting joins.

Fattening the picture means adding extra black pixels around the edge of all black regions.  The algorithm we used was based on 8-connectedness.  If one of the 8 neighbours of a white pixel were black then that pixel also became black.  Adding pixels in this way is dependent on the resolution of the scanned image.  If we doubled the resolution then the fattening algorithm described would add a thinner line round the outside of the black regions.  We countered this by applying several rounds of fattening, with the number of rounds being dependent on the resolution.

## 2.2. Joins

Counting the number of joins in a doodle is expected to be independent of the black and white region counts.  Counting black regions counts the number of distinct features, while counting the number of joins is a measure of the overall complexity.  Detecting joins is, however, considerably harder than counting black and white regions.  The first difficulty is caused by lines being several pixels wide and another problem is caused by large black shaded regions in a doodle.  In practice, however, shaded regions do not cause a problem in this application because they are rare and distinctive.  Thus even a very inaccurate classification scheme will not cause many false negatives.

The problem of thick lines can be addressed by applying a thinning algorithm to the doodle.  This will shrink all lines to single pixels width to provide a skeleton of the doodle.  In the algorithm we use, the lines remain 4-connected throughout the thinning process.  This makes it easy to detect joins.  If a black pixel has three or four black neighbours in the east, west, north or south positions then it is a join, while if it has just one black neighbour in the east, west, north or south positions then it is the end of a line.

There are two problems caused by the thinning algorithm.  Firstly, one join can become two after the thinning process.  This is especially true when two lines cross at an oblique angle.  Thus, a thick line crossroad becomes two thin line junctions.  This problem is fairly ubiquitous, since lines crossing at oblique angles are very common.  This problem is alleviated by combining joins that are close together into one super-join and not counting them as separate joins.

The second problem is caused by bulges in a thick line being thinned down into a short line segment. This adds an additional spurious join to the join count.  This type of problem can be caused in several ways.  One line joining another can overshoot slightly, causing a bulge opposite the join.  Alternatively, the drawer can pause while drawing the line, and then resume in a slightly different direction.  This is corrected by shaving the thinned drawing before counting joins.

A straightforward thinning algorithm was used.  The doodle was first fattened, as described earlier. Then it was scanned repeatedly from different directions and black pixels removed if they were not needed to maintain connectivity.  The ends of line segments and isolated black pixels were not removed.  The shaving algorithm removed short line segments, where the definition of "shortness" depended on the resolution of the image.  Similarly, joins that were close together were combined, with the definition of "closeness" again being

dependent on the resolution. Figure 2 shows an original doodle and its thinned version with small hair-like artefacts before shaving.



**Figure 1.** Three different smiley doodles. Each has 2 white and 4 black regions after fattening.
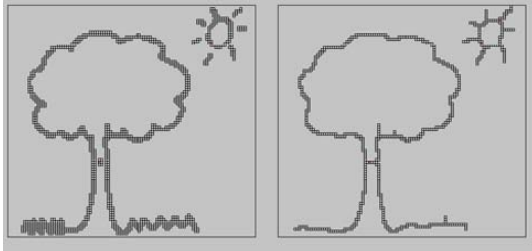


**Figure 2.** An original doodle and the thinned version. Three bulges have produced small hair like lines which are removed by shaving.
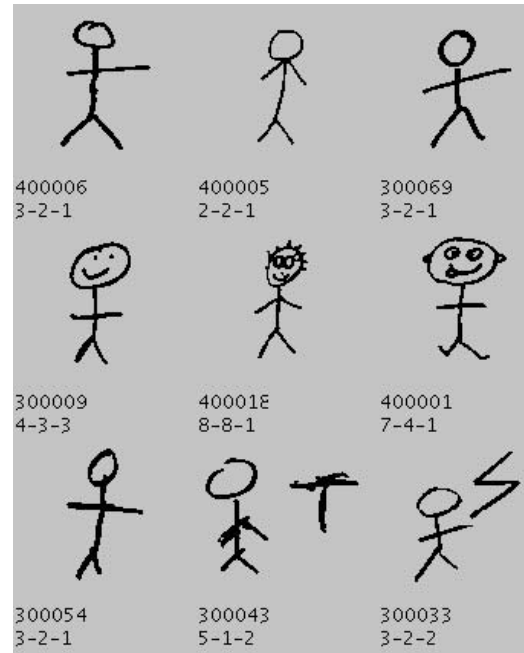


**Figure 3.** A variety of different stick men.

## 2.3. Examples of Automatic Classification

Figure 3 shows a number of similar stick-men doodles and their corresponding automatic classification values. The number under each doodle, for example 400006, is the doodle id in our system. The three numbers under that, for example 3-2-6, are the number of joins, number of white segments and number of black segments. The automatic classification numbers are quite similar, so we should not expect to get any of the more serious false negative similarities in these cases.

Now let us move on to an analysis of the statistical properties of the three measures over all 525 doodles in our collection. They are presented in Table 1. One thing to note is that around half of the doodles have only one connected black region. Very few have more than 4. Thus the number of different black regions is not a very good discriminator, as shown by the low standard deviation. The standard deviation for the number of joins and white regions is much larger, so we should expect them to be better discriminators.

| Value | Joins | White | Black |
|-------|-------|-------|-------|
| 0 | 52 | 0 | 0 |
| 1 | 33 | 53 | 231 |
| 2 | 33 | 92 | 92 |
| 3 | 47 | 67 | 66 |
| 4 | 47 | 52 | 62 |
| 5 | 38 | 41 | 31 |
| 6 | 35 | 39 | 20 |
| 7 | 29 | 38 | 13 |
| 8 | 36 | 27 | 3 |
| 9 | 29 | 23 | 6 |
| 10 | 18 | 22 | 5 |
| >10 | 149 | 92 | 17 |
| mean | 8.1 | 6.7 | 2.9 |
| std | 7.8 | 7.2 | 2.9 |

**Table 1.** Frequency of values for the three doodle measures, together with mean and standard deviation.

| threshold | choice | popular |
|-----------|--------|---------|
| 3 | 315 | 14,13,12,12 |
| 4 | 254 | 14,14,14,14 |
| 5 | 202 | 15,14,14,14 |
| 6 | 140 | 19.16.15.15 |
| 7 | 106 | 18,17,16,16 |
| 8 | 76 | 20.19.19.18 |
| 9 | 54 | 22,22,21,20 |
| 10 | 35 | 27,26,25,24 |
| 11 | 27 | 32,30,30,27 |
| 12 | 16 | 36,35,34,34 |
| 13 | 9 | 40,37,37,37 |
| 14 | 5 | 44.43.43.41 |
| 15 | 0 | not applic |

**Table 2.** Choice and repetition for various values of threshold.

## 2.4. Comparing Doodles

Two doodles are compared by calculating the weighted sum of the absolute differences for the three automatically calculated classification numbers. Experience and the statistics presented in Table 1 show that the number of white regions is the most valuable discriminator. The number of black regions is less discriminatory because there are fewer options, while the number of joins is also less effective. With this in mind we have weighted the three automatic classification numbers in the ratio 1 : 4 : 1. If this sum is less than a threshold value then the doodles are considered to be similar.

In our system, each user provides 4 doodles as their passwords. When they log in they are presented with four screens one after the other. Each screen contains a password doodle and 15 distractors in a four by four grid. They must successfully choose all four pass doodles before gaining access to the system. Thus we must choose 60 distractors for each user. This choice is constrained as follows:

1. No distractor can be the same as one of the user's pass doodles.
2. All 60 distractors must be different.
3. No pairs of doodles on each screen should be similar.

Our system has 50 users and 525 different doodles. The algorithm to achieve this is straightforward. The main tuneable parameter is the similarity threshold.

We have conducted an experiment to vary the value of this threshold to see how it affects the choice of doodles and the use of common doodles. Table 2 records the value of threshold and the minimum number of candidate doodles available when the last distractor is chosen in all 200 screens. The popular values list the frequency of the 4 most popular doodles out of 3200 doodles on 200 screens.

Notice that choice drops steadily until there are no choices for some of the screens when threshold is 15. The choice of most frequent doodles varies up to a threshold of around 10, but then the choice narrows down and some doodles always appear with a high frequency. We have chosen threshold=10 as the optimum value for the size and diversity in our system.

This system was evaluated by asking a researcher who had not worked on this project to examine al 200 screen and find similarities. He found 3 pairs that were superficially similar but not likely to confuse a user, which is encouraging. Further evaluation is under way.

## 3. Conclusions

An examination of the password doodles chosen by the users of our system shows a wide variety of styles. Some users have been very inventive with their sketches and spending time designing and drawing them. We can confidently expect these users to be able to recognise their doodles again quite easily. At the other extreme, some users have just dashed off a very simple drawing with very little imagination or original thought. Their doodle is likely to be very similar to many others. There can be many reasons for this casual approach, including a failure to understand the need for secret codes but it is not up to us to blame users for choosing to provide such simple doodles.

Thus the purpose of this research was to develop an algorithm that could detect doodle similarity. We focused mostly on simple doodles. It would be nice to be able to detect similarities between more complex doodles, but this is both harder from an algorithm design perspective and less important when choosing distractors since the very complexity that challenges the algorithm makes the distractor sufficiently different to make it less likely that the user will confuse the distractor with the target.

We have examined the usefulness of three such measures: the number of joins, the number of white regions and the number of black regions. By far the most useful is the count of the white regions. This measure alone can rule out almost all similar distractors. The number of black regions is moderately useful when this number is greater than one. The number of joins is also moderately useful, although being able to calculate this number accurately is a surprisingly difficult task.

As a result, we have developed an algorithm that can eliminate most similar distractors when the target doodle is simple. This is also the most useful case because the users who have created a simple doodle are least likely to remember any distinguishing features. Simple figures such as stick men may, in fact, not have any distinguishing features at all.

## References

DeAngeli A., Coventry L., Johnson G., & Renaud K. (2005). Is a picture really worth a thousand words? Reflecting on the usability of graphical authentication systems. *IJHCI, 63(1-2),* 128-152.

Renaud K. (2005). A visuo-biometric authentication mechanism for older users, *Proc British HCI 2005, Sept 5-9, Edingburgh.*

Renaud K. & Olsen E. S. (2007). DynaHand: Observation-resistant Recognition-based Web Authentication, *Technology and Soociety (to appear).*