Modeling user choice in the PassPoints graphical password scheme

Ahmet Emir Dirik
Polytechnic University
Department of Electrical and Computer Engineering
Brooklyn, NY, USA
emir@isis.poly.edu

Nasir Memon
Polytechnic University
Department of Computer and Information Science
Brooklyn, NY, USA
memon@poly.edu

Jean-Camille Birget
Rutgers University at Camden
Computer Science Department
Camden, NJ, USA
birget@camden.rutgers.edu

ABSTRACT

We develop a model to identify the most likely regions for users to click in order to create graphical passwords in the *PassPoints* system. A PassPoints password is a sequence of points, chosen by a user in an image that is displayed on the screen. Our model predicts probabilities of likely click points; this enables us to predict the entropy of a click point in a graphical password for a given image. The model allows us to evaluate automatically whether a given image is well suited for the PassPoints system, and to analyze possible dictionary attacks against the system. We compare the predictions provided by our model to results of experiments involving human users. At this stage, our model and the experiments are small and limited; but they show that user choice can be modeled and that expansions of the model and the experiments are a promising direction of research.

Categories and Subject Descriptors

H.5.2 [**Interfaces and Representation**]: User Interfaces—*Graphical user interfaces*; K.6.5 [**Computing Milieux**]: Security and Protection—*Authentication*

Keywords

Graphical passwords, password entropy, user behavior, dictionary attack

1. INTRODUCTION

The most common user authentication scheme in computer systems today is the alphanumeric password. Although alphanumeric passwords are used widely, they have certain well known drawbacks such as low memorability of high entropy passwords. These drawbacks are not due to the authentication system itself but arise from the interaction between the users and the system. Since users usually cannot remember high entropy passwords they tend to select short or simple passwords, that can be broken by dictionary attacks [17, 1]. Policies and mechanisms that force users to select high entropy passwords usually result in other unsafe practices, such as the passwords being written down and kept in the open.

In order to improve the security of user authentication, alternatives to alphanumeric passwords have been proposed, e.g., token based authentication, biometrics, graphical passwords, or "multiple factors" based on the simultaneous use of two or more authentication mechanisms. This paper focuses on graphical password systems, as a "single factor" authentication.

The main motivation for graphical passwords is the hypothesis that people are better at remembering images than artificial words. Visual objects seem to offer a much larger set of usable passwords. For example we can recognize the people we know from thousands of faces; this fact was used to implement an authentication system [19]. As another example, a user could choose a sequence of points in an image as a password; this leads to a vast number of possibilities, if the image is large and complex, and if it has good resolution. This is the basis for the graphical passwords in [4, 3, 27]. An excellent survey of the numerous graphical password schemes that have been developed is [23].

Following [27] we classify password systems as

1. Recognition based systems [5, 10, 2, 9, 19, 26],
2. Pure recall based systems [14, 15, 24],
3. Cued recall based systems [4, 22, 3, 27, 13].

In *recognition* based systems, a user chooses images or icons or symbols from a large collection; to be authenticated, the users need to recognize their previous choice among a large set of candidates. Dhamija, et al. [10] presented a scheme based on recognition of computer generated images. Akula and Devisetty [2] provide a variation of this method. The commercial scheme Passfaces [19] uses images of human faces. Davis, et al. [9] studied such systems and found that user password selection is biased by race and gender. Weinshall and Kirkpatrick [26] worked on a similar recognition based scheme in which users were asked to recognize a set of images (100-200) from a database of 20,000 images. Their studies showed that even after one or two months, users could still recognize their graphical passwords with 90% accuracy. This study supports the hypothesis that people remember pictures/images better than alphanumeric strings.

Symposium On Usable Privacy and Security (SOUPS) 2007, July 18-20, 2007, Pittsburgh, PA, USA.

Recognition based graphical passwords seem to be easy to remember, but they have a drawback: On order to provide a sufficiently large password space they require many rounds of image recognition for authentication, which is tedious.

In *pure recall* based graphical password schemes, users need to reproduce their password without being given any hints or cues. Alphanumeric passwords, as well as manuscript signatures, are examples of means of authentication based on pure recall. Jeremyn et al. [14] described a graphical password scheme "Draw a Secret" (DAS), where users draw a shape on a grid. Users need to draw approximately the same shape in order to authenticate themselves. Wei-Chi Ku et al. [15] study a variation of DAS. Recent research by Thorpe and van Oorschot [24] describes possible dictionary attacks against DAS. Overall, graphical password schemes based on pure recall are quick and convenient to use, but they seem to have the same disadvantage as alphanumeric password: They are hard to remember with sufficient precision when they have enough entropy to be secure.

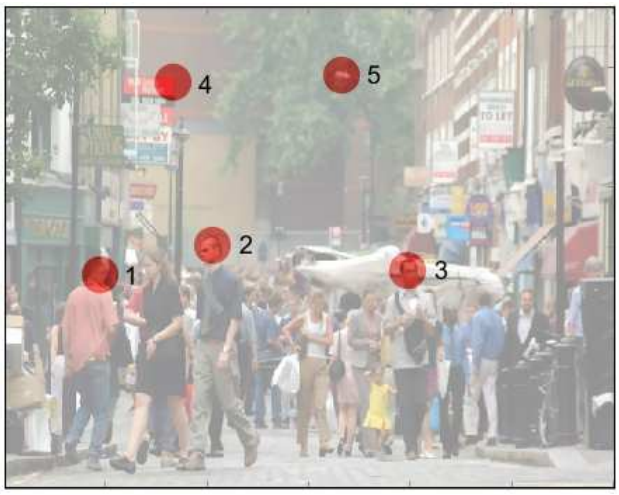
The concept of *cued recall* was introduced in [27]. As the name indicates, users have to recall a password, but the system offers a framework of hints, context, and cues, that help the users reproduce their password or help them make the reproduction more accurate. In the field of computer systems the earliest example of a graphical password scheme based on cued recall was Blonder's patent [4]. Here, the user is shown an image on the screen, and the password consists of a few points that the user chooses in the image (by clicking or pointing). The underlying images in the system help users recall their graphical password click points, but they have no direct role in the password. Authentication is performed by clicking near the previously determined points. In Blonder's scheme the image is partitioned into regions, whose outlines are visible; this results in comics-like images. The user has to click within the correct regions to log in. An extension of Blonder's idea was presented in [3]. This system allows natural images, without visible regions; instead, there are several underlying discretization grids (invisible to the user). Cued recall is intermediate between recognition and pure recall.

Since graphical passwords are relatively new, their security has not been investigated as much as that of alphanumeric passwords. We mentioned already the human tendency to choose weak passwords, which enable dictionary attacks. Is the same true for graphical passwords? It has been said that graphical passwords can resist dictionary attacks due to a very large actually used key space. However, the evidence for this so far is mostly anecdotal. For graphical passwords in the DAS system, mentioned above, it was shown in [24] that users often choose simple and somewhat predictable passwords [24]. On the other hand, as we mentioned already, DAS is a pure recall system, which gives it some similarity to alphanumeric passwords. (Note added in proofs: The editors of the conference drew our attention to [31] which became available only after our submission; that paper devises other dictionary attacks on the PassPoints system.)

This paper addresses the question how one can model the way a user chooses a click point in a Blonder type graphical system. We will focus on the PassPoints system of [3, 27], which uses natural images. We also show how a dictionary attack can be based on such a model, which also leads to suggestions about how the password system could be made more resist against such an attack. A password is a sequence of click points, but in this paper we only model the choice of one click point. The question of how a sequence of click points might be chosen, and the possible correlations between click points in a same password will be investigated in a later paper. The rest of this paper is organized as follows: In Section 2 we review the PassPoints graphical password system. In Section 3 we develop a model that predicts user choice of one click point in a PassPoints password, and we predict the entropy of one click point. In section 4 we give experimental results that validate our model and prediction techniques. In section 5 we investigate how we can exploit the user choice model to launch a dictionary attack on PassPoints, and to strengthen the system. Section 6 has a conclusion and mentions future work.

2. PASSPOINTS, A GRAPHICAL PASSWORD SCHEME

In the PassPoints graphical password scheme a password consists of a sequence of click points (say 5 to 8) that the user chooses in an image. The image is displayed on the screen by the system. The image is not secret and has no role other than helping the user remember the click points. Any pixel in the image is a candidate for a click point.



(a) PassPoints clicks

Figure 1: A screen shot of the PassPoints system

To log in, the user has to click again closely to the chosen points, in the chosen sequence. Since it is almost impossible for human users to click repeatedly on exactly the same point, the system allows for an error tolerance r in the click locations (e.g., a disk with radius $r = 10$ or 15 pixels). This is done by quantizing (discretizing) the click locations, using three different square grids, as described in [3]. Each grid has width $6r$ between grid lines (horizontal or vertical). Each one of the three grids is staggered with respect to the previous grid by a distance $2r$ vertically and a distance $2r$ horizontally; see Fig. 1 (b). If there were only one quantization grid then a selected click point could be close to a grid line and small variations in the user's clicking could lead to a click in a different grid square, thus leading to the wrong password. On the other hand, one can prove (see [3])

that with the three staggered grids every point in a two-dimensional image is at distance at least r from the grid lines of at least one of the three grids; we say that the point is "safe" in that grid.

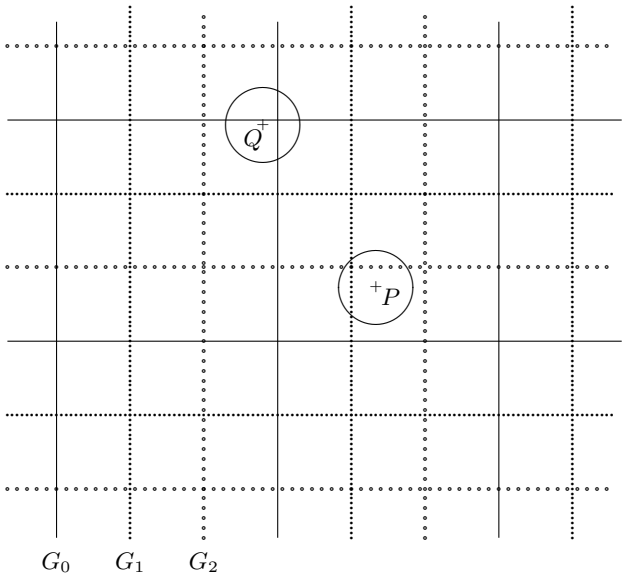


Figure 1 (b): Three staggered grids, G_0, G_1, and G_2. The point P is safe in G_0; Q is safe in G_1 and in G_2.

The simultaneous use of multiple grids makes the click points "robust" against the inevitable small uncertainties in the clicking; hence, this form of discretization is called "robust discretization", or "robust quantization". Click positions are mapped into grid squares. A sequence of click points is represented by a sequence of grids together with a sequence of grid squares. For secure storage of passwords by the system, a cryptographic hash function is applied to the sequence of grid squares.

An important feature of the PassPoints system is that the underlying images for a password are not restricted to simple comics-like drawings. Complex real-world images can be used; users can even install their own images. Natural images help users remember complex passwords better. This suggests that in a human context, the (conditional) entropy of a password will depend on the underlying image, and leads to the question: Given an image, how can we predict the (conditional) entropy of a click point in that image, within the context of PassPoint passwords? We want to develop a model that provides probabilities with which a user clicks on (or near) any point in an image.

3. MODELING USER CHOICE IN PASSPOINTS

Studies on visual attention and eye movements show that most images contain a few portions that most humans focus on [12, 21]. When asked to create a graphical password a user would probably not click with the same frequency on all available pixels, but focus on some specific areas. This is illustrated in Fig. 2 (a), which shows an image along with click points that were actually selected by users; a large amount of clustering of click points is evident.

In the PassPoints scheme the clustering of the users' click points reduces the entropy of these click points. In Section 4

(a) Actual clicks

(b) Predicted clicks

Figure 2: Actual clicks vs. predicted clicks

we will describe experiments that enable us to determine an *observed entropy* of user click points. However, our goal goes beyond observing entropy: We want a model that enables us to predict the entropy of user click points. Such a model would enable us to design automatic dictionary attacks, or to rule out certain images *a priori* (if they lead to low entropy). In this section we develop a model of user choice of a click point in the PassPoints system and give an algorithm that predicts the most likely click locations, along with their probability values. Such a model helps in evaluating the security of the PassPoints system *a priori* (i.e., before any observational user studies), and provides a method for selecting appropriate images that result in higher entropy of the users' click points.

Overview of the a priori entropy prediction:
1. In order to predict the possible click positions, a color-based *mean-shift segmentation* algorithm [6, 7] is applied to the image. This algorithm partitions a digital image into regions, called segments, according to a given criterion in order to locate objects of interest. Among the many possible

segmentation algorithms we want to choose one that detects natural boundaries of visually attractive regions in an image. The mean-shift segmentation was explicitly designed to meet this requirement. It produces an image partition that eliminates redundant information and highlights the important regions.

2. After segmentation, the *centroid* (barycenter) of each segmented region is calculated. All these centroids will be weighted according to their attractiveness to humans. The centroids are mapped to the grid squares of the robust quantization that was described in the previous section; the probability values of all the centroids that are mapped into the same grid square are summed, and the result is taken to be the *attention probability of that square*. This defines the *focus of attention map* which, for each of the three grids and each grid square, gives the probability that this grid square will be clicked in.

3. The entropy of a click point in a given image is calculated by using the attention probabilities of the grid squares. Fig. 3 summarizes the above steps in the form of a block diagram.

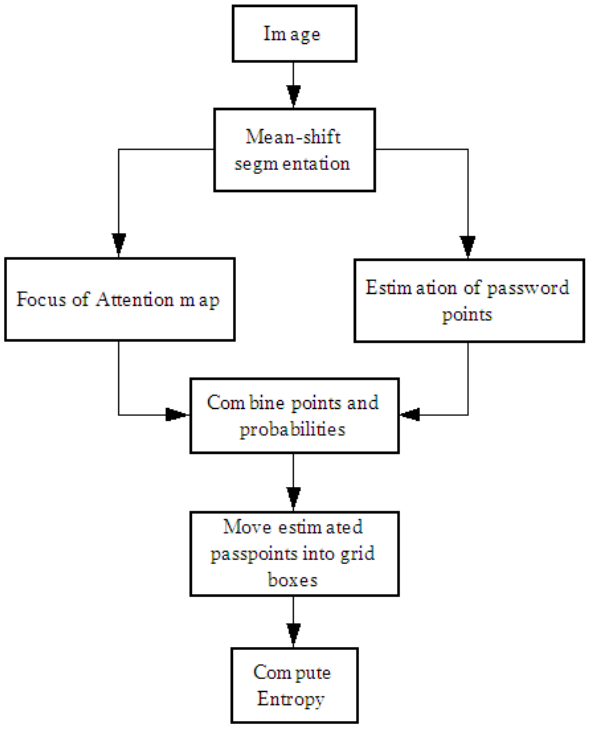


Figure 3: Block diagram of click point entropy prediction

3.1 The probability of click points

Users often select the centers of the objects in an image as password click points. This observation is supported by Fig. 2. In order to predict some of the points that users are likely to select, we first segment the image as mentioned above, using mean-shift segmentation; this preserves the shape and color information of objects in the image and determines important regions and details in the image.

The mean-shift segmentation algorithm uses a five dimensional space (3 dimensions for color and 2 dimensions for cartesian coordinates). Color information is represented in the "L*u*v color space" (see [25]). After segmentation, we compute the centroid of each segment. However, this results in a very large set of predicted points, so this set is reduced a smaller set by ignoring the relatively large and and the relatively small segments, which is justified by some studies, e.g., [18].

Given an image, the prediction of a set of click points is not enough to evaluate the security of the graphical password. We also need to predict the probabilities of the predicted click points. For example, it can be seen in Fig. 2 (a) that the click point distribution is not uniform; some regions in the image are more likely to attract user attention than others. In order to model this behavior, a focus of attention map is computed, as explained in more detail in the next subsection.

3.2 Focus of attention map

Some studies have shown that the user attention is influenced by both "high-level" and "low-level" factors. High-level factors involve image content and memory feedback [18], but these factors are too complicated to be included in a first model. Low-level factors are basic geometric and physical image features, such as contrast, size, shape, color, motion, location, foreground, object category, etc. [18, 30].

One of the most important factors which attracts user attention is *contrast* [29, 11]. The size of the object region is another factor which attracts the attention of a user to a particular object. Some particular colors (e.g., red) also attract our attention, especially if there is a high contrast between the region's color and the background color. Finally, many studies have shown that users generally focus on people in a scene, and in particular on the eyes, mouth and hands [20, 21, 29]. In order to compute the focus of attention map for an image, all the above factors should be considered. In our case, to demonstrate the validity of our modeling approach, we selected some of the above factors and combined them in a fixed way (as opposed to adapting the factors and their combination to the image). The factors used in our study to compute the focus of attention map are *luminosity contrast*, *color contrast*, and *foreground* of segments.

3.2.1 Luminosity contrast

Contrast is the difference in visual properties that makes an object distinguishable from other objects and the background. The luminosity contrast of a segment is calculated by taking the intensity value (i.e., the gray level) of a segment and comparing it with neighboring segments. The luminosity contrast of a segment R_i is calculated as follows:

$$\mathsf{LumContr}(R_i) = \frac{1}{N_i} \cdot \sum_{k=1}^{N_i} |\mathsf{gray}(R_i) - \mathsf{gray}(R_{i,k})|$$

where $\mathsf{gray}(R_i)$ is the gray level of the segment R_i, N_i is the number of neighbors of R_i, and $R_{i,k}$ $(k = 1, \ldots, N_i)$ are the segments that are adjacent to R_i.

3.2.2 Color contrast

In addition to contrast in luminosity, contrast in hue between a segment and its surrounding is a good measure of saliency. It is computed in the HSV domain (Hue Saturation Value) [16]. Hue defines the color value (such as blue, yellow, green) of an area, saturation measures the colorfulness

of the area in proportion to its brightness. The "value" is related to the color luminance or color intensity. Color contrast is computed in the same way as luminosity contrast, but hue values are used instead of gray levels. Before transforming RGB (Red Green Blue) into the HSV domain, RGB values are normalized to remove the brightness of the color. Then, normalized RGB values are transformed into the HSV domain, and the hue contrast is computed as follows (where $R_i, N_i, R_{i,k}$ are as in the previous formula):

$$\mathsf{Color}(R_i) = \frac{1}{N_i} \cdot \sum_{k=1}^{N_i} |\mathsf{hue}(R_i) - \mathsf{hue}(R_{i,k})|$$

3.2.3 Foreground

This feature distinguishes foreground objects from background objects. We use the observation that background objects typically occupy very large segments, compared to foreground objects. Therefore we use the length of the borders of segments to label them as background or foreground. Next, we eliminate very large regions which are likely to belong to the background of the image and have lower probability of selection. The foreground feature is calculated as follows:

$$\mathsf{Foregr}(R_i) = 1 - \min\left\{1, \frac{\mathsf{border}(R_i)^{1.3}}{\mathsf{totalborder}}\right\}$$

where $\mathsf{totalborder}$ is the total number of border pixels (for all the segments) in the image, and $\mathsf{border}(R_i)$ is the number of border pixels of segment R_i. When $\mathsf{border}(R_i)$ is very large then the value of the foreground feature for R_i is close to zero. The exponent 1.3 in the equation is obtained empirically [18].

3.2.4 Combining the saliency features

The three saliency features above are combined into a final **Focus of Attention (FoA) map** for the image which takes a value between 0 and 1; We multiply each feature by a weight W_k $(k = 1, 2, 3)$:

$$\mathsf{FoA}(R_i) = W_1 \cdot \mathsf{LumContr}(R_i) + W_2 \cdot \mathsf{Color}(R_i) + W_3 \cdot \mathsf{Foregr}(R_i).$$

These weights are fixed and obtained empirically in our study but they can also be computed adaptively according to the content of the image. Contrast is the most important factor for determining the most salient regions and it is given higher weight than color and foreground.

Once the FoA map has been computed, it is compared to a threshold which is determined empirically. Attention values under that threshold are set to zero in order to create a better FoA map. Our assumption here is that saliency values under a certain threshold are equally likely and do not attract user attention.

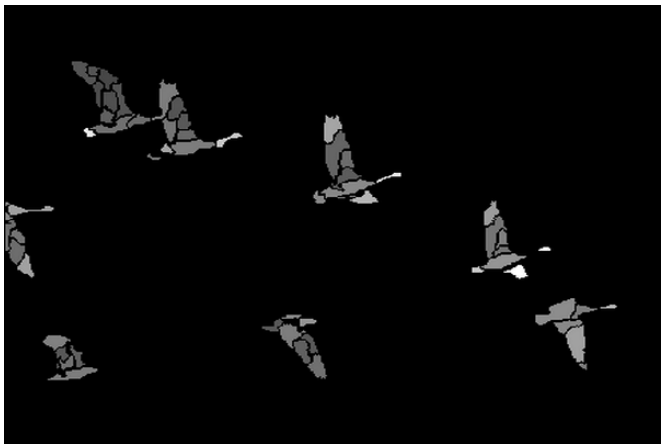
For each one of the three grids, the FoA values of the points that get quantized to the same grid square are summed, which yields an FoA value for each grid square in the grid. The FoA values of the grid squares are turned into probabilities by dividing each FoA value by the sum of all FoA values. These probabilities are then used to predict the entropy $H(I)$ per click point in an image I:

$$H(I) = -\sum_{k=1}^{N} p_i \cdot \log_2 p_i$$

where N is the number of grid squares, p_i is the predicted probability of grid square i.

Remark: In this paper we only study the probability and entropy of a single click point. If the click points of a password with k click points were independent then the total entropy of the graphical password would be $k \cdot H(I)$.

(a) Original image



(b) Focus of attention map

Figure 4: FoA map

However, it is not reasonable to assume independence. So, $k \cdot H(I)$ is an upper bound on the total entropy of the graphical password. Obviously, dependence between click points only makes dictionary attacks easier.

The FoA map and the entropy that we computed in this Section could be called *a priori FoA map* and *a priori entropy*, as opposed to the *observed FoA map and entropy* that will be obtained in the next Section.

4. EXPERIMENTAL RESULTS

In order to collect real graphical password data, a Java based authentication system was developed and tested. Over a hundred users participated in the project. The participants were mostly graduate and undergraduate students who were asked to create a graphical password on one of two different images. Although in PassPoints users could be allowed to import their own images, we used two fixed test images[1] in our experiment in order to compare them. The first one, the Birds Image (Fig. 5), is simple and presumably not very good for the PassPoints system, as it contains only relatively few salient points. The second image, which we

[1]The images were taken from www.freefoto.com

call the People Image, is more complex and appears to have a larger entropy (Fig. 6).

Every user was presented one of the two images and asked to select a password, consisting of 5 clicks, that is not easy to guess, but that they should be able to remember. In order to ensure that the users entered realistic passwords they were asked to re-enter the password. Only correctly re-entered passwords were used in this experiment.

We also used our model (as described in Section 3.1) to predict as many points as were obtained in the experiment. The two images along with the observed and the predicted points are shown in Figures 5 and 6.

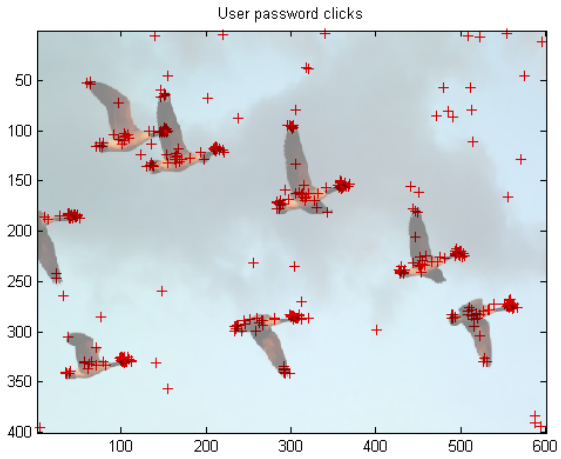
In the Birds Image (Fig. 5 (a)) one observes that the participants generally clicked on the most salient regions in the image such as the flying birds. Our model predicted these regions successfully. User click positions were predicted with **80%** accuracy. The true positive (true prediction rate of password points) and true negative (true prediction rate of non-password points) of our model were 0.79 and 0.80; see Table 2.

Table 1: Parameters of the experiment

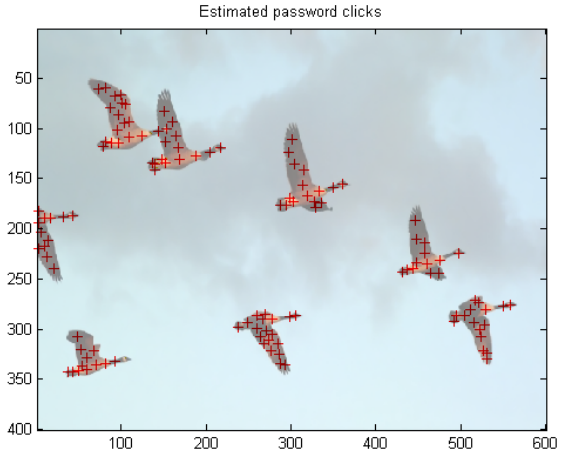
Image name	BIRDS	PEOPLE
Image size (in pixels)	400 × 600	400 × 600
Click tolerance r	10 pixels	10 pixels
Number of user passwords	92	142
Number of grid squares	264	264

The People Image contained many more features than the Birds Image, so it is not surprising that the distribution of user click points is less clustered (Fig. 6 (a)). Also it is seen from Fig. 6 (a) that the participants did not click much on places that are hard to recall such as leaves and the flat background. These regions have low probability and were successfully detected by our model (see Fig. 6 (b)). For the people image, the model's prediction accuracy of user click points was **71%**. The true positive and true negative values are given in Table 2.

In order to test the performance of our prediction model of the probability distributions of click point positions, we applied the Kolmogorov-Smirnov (KS) goodness-of-fit hypothesis test. The predicted and actual click point positions are considered as two independent random variables with the same distribution. The Kolmogorov-Smirnov test measures the maximum difference between the cumulative distributions of two independent random variables. If the KS value is close to zero we can say that these two random variables have similar probability distributions. In others words the null hypothesis at significance level alpha (0.05 in our case) is not rejected if alpha is lower than the asymptotic P-value (the smallest level at which the null hypothesis can be rejected). For the Birds Image, our prediction model worked well and the KS test confirmed that our prediction of the click points distribution is valid (alfa is lower than the P-value). However, the KS test produced a higher KS value for the more complex People Image. This means that our probability prediction result was only fair for complex images and requires further improvement (Table 2). As can be seen from our results, the prediction of the most popular points in the People Image is not as successful as in the Birds Image.



(a) Actual click points



(b) Predicted click points

Figure 5: Predicted vs. actual click points

From the actual user click points we obtain an *observed FoA map* which gives a clicking probability to every grid square; this probability is measured by the number of clicks in the grid square, divided by the total number of clicks made in the image. The observed FoA map is then used to obtain the *observed entropy* of a click point in an image.

Applying this to our two images we found that the observed entropy was **5.2 bits** per click point for the Bird Image and **6.5 bits** per click point for the People Image. This shows that the People Image is better as an underlying image in the PassPoints graphical password system.

By using our entropy prediction we get a similar conclusion. Our entropy prediction results, given in Table 2, are that the predicted click point entropy for bird image is **5.3 bits** and **7.2 bits** per click point for the more complex People Image.

We should note that the observed entropies are also only estimates, since we used relatively few data points: Our images have 264 grid squares (per grid), but only 79 grid squares in the Bird Image, and only 125 squares in the Peo-

Table 2: Click point entropy prediction results

IMAGE NAME	BIRDS	PEOPLE
True positive	34/43 = 0.79	132/194 = 0.68
True negative	176/221 = 0.80	55/70 = 0.79
Kolmogorov-Smirnov statistics (significance level = 0.05)	0.0592 (good est.)	0.1030 (fair est.)
Entropy of a click point	5.2 bits	6.5 bits
Estimated click point entropy	5.3 bits	7.2 bits
Max entropy for a point	8.0 bits	8.0 bits

ple Image had one or more data points. We would need to perform experiments with at least a few thousand users to get more reliable estimates for the observed entropy.

Nevertheless, these small experiments and our simple model demonstrate that user choice can indeed be modeled. In the next section we show how this can be used to strengthen the security of the PassPoints passwords against dictionary attacks.

5. DICTIONARY ATTACK

As an application of our click position prediction model we show how it provides a starting point for designing an automated dictionary attack against PassPoints. For a graphical password consisting of 5 clicks on an image with pixel size 640×480 and an error tolerance of $r = 10$ pixels, there are 264 squares in a grid; hence, the theoretical maximal number of passwords is $264^5 \approx 1.28 \times 10^{12}$. This is not a large password space, and a simple exhaustive search is somewhat feasible. So, the purpose of our experiments is to illustrate a dictionary attack *method*, not to provide a realistic dictionary attack against a realistic system.

For an automated dictionary attack we first apply our PassPoints prediction algorithm to detect the positions in the image that a priori the most likely for the users. Next we sort the grid squares according to their predicted probabilities for each of the three grid systems. We ignore grid squares with probabilities close to zero. We will also consider a dictionary attack in the case where the grid numbers are not known to the attacker.

For the Bird Image, our attack results are given in Fig. 7 (a). The dotted line (Attack 1) refers to the scenario where we have access to the hash value of user password and to the grid numbers. The smooth line (Attack 2) refers to the scenario where the attacker does not know the sequence of grid numbers. The results show that knowledge about the sequence of grid numbers improves the dictionary attack. According to the figure, if we know the users' grid numbers we can discover 61% of 92 user passwords by searching a very small password space of size 31^5.

A dictionary attack against the more complex People Image turned out to be less effective. The predicted entropy per user click point is 7.2 bits for the People Image. This means that the expected number of different locations (per click point) in the People Image is greater than 100. This is high enough to make a dictionary attack difficult, provided that the number of click points is large (e.g., greater than 7). The attack results with 5 click points are given in Fig. 7 (b). According to the figure, even after searching the 80^5 5-point passwords made from the 80 most points, we could crack only 12 user passwords out of the 142 that we had in our data set. Only after 31^5 ($\approx 2.8 \cdot 10^6$) iterations could we crack the first user password. This does not prove that the People Image with 5 click points is safe, but illustrates the influence of the choice of the image.

We can also design a dictionary attack based on the observed FoA map, instead of the FoA map that is predicted by our model. Intuitively we would expect that such an attack to be more successful than an attack based on an a-priori map, since the observed entropies are lower than the predicted ones. An experimental verification of this hypothesis would be interesting. A drawback of attacks based on observed FoA maps is that it does not seem possible to automate such attacks. Since an important purpose of the a-priori FoA map is to rule out images that are bad for PassPoints, automation is very useful.

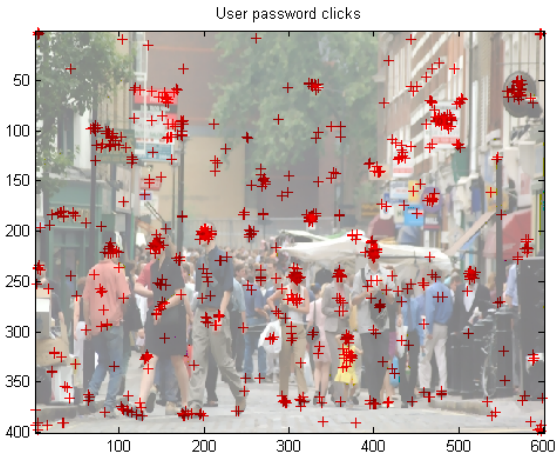
6. CONCLUSION AND FUTURE WORK

In this paper, we investigated the security of the PassPoints graphical password scheme and the suitability of the underlying images, by providing a model that predicts the users' click points and their saliency value. From this we predicted the entropy of a click point in a graphical password.

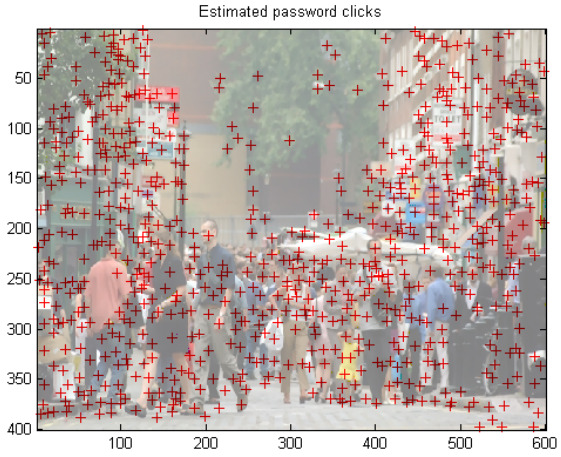
We tested our model experimentally on two images. We analyzed the password security of those underlying images by computing the entropy of a click point, and we compared the predictions produced by our model with data consisting of roughly 100 actual passwords selected by users. In these (very small) images our model was able to predict **70-80%** of the user click positions (Table 2). The results show that our model can be used to evaluate the suitability of an underlying image for the PassPoints system.

Our model could be improved by extending the FoA map so that, in addition to centroids of regions, it includes mid and end points of edges in the image, as well as corner points or tips of pointy regions. Moreover, in image segmentation, texture information may be included to get better results in natural images. A more difficult, but very important improvement of the model would be to include "high-level" factors of attraction (i.e., based on image "content"). In this paper we only considered individual click points. In order to predict entire passwords we must consider the correlations between click points in a graphical password. Finally, for a better experimental test of our model we would need to collect thousands of graphical password data for different types of images.

Even at this point we can say that when users create graphical passwords they should be aware that the most salient regions can be predicted automatically with a significant probability.

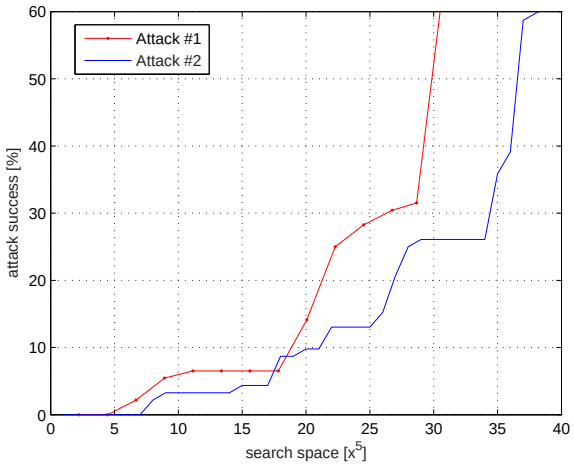


(a) Actual click points

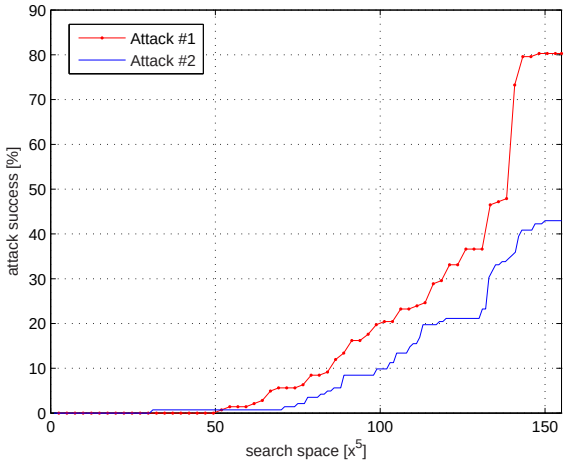


(b) Predicted click points

Figure 6: Predicted vs. actual click points



(a)



(b)

Figure 7: (a) Dictionary attack for Bird Image. (b) Dictionary attack for People Image

7. ACKNOWLEDGMENTS

We would like to thank Alex Brodskiy for programming the web interface used to conduct the human experiments.

8. REFERENCES

[1] A. Adams, M.A. Sasse, "Users are not the enemy: why users comprise computer security mechanisms and how to take remedial measures," *Communications of the ACM* 4 (1999) 41-46.

[2] S. Akula, V. Devisetty, "Image based registration and authentication system," *Midwest Instruction and Computing Symposium* (2004).

[3] J.C. Birget, D. Hong, N. Memon, "Graphical passwords based on robust discretization", *IEEE Transactions on Information Forensics and Security* 1(3) (Sept. 2006) 395-399. (Earlier version: Cryptology ePrint Archive, http://eprint.iacr.org/2003/168, Aug. 2003.)

[4] G.E. Blonder, "Graphical Passwords", United States Patent 5559961 (1996).

[5] M. Boroditsky, "Passlogix Password Schemes" (2002). http://www.passlogix.com

[6] D. Comaniciu, P. Meer, "Mean shift analysis and applications", *7th International Conference on Computer Vision* (1999) 1197-1203.

[7] D. Comaniciu, P. Meer, "Mean shift: A robust approach toward feature space analysis", *IEEE Transactions on pattern analysis and machine intelligence* 24(5) (2002) 603-619.

[8] L. Coventry, A. De Angeli, G. Johnson, "Usability and biometric verification at the ATM interface", *SIGCHI Conference on Human Factors in Computing Systems* (CHI'03) (2003) 153-160.

[9] D. Davis, F. Monrose, M. Reiter, "On user choice in graphical password schemes", *13th Usenix Security Symposium* (2004) 1-14.

[10] R. Dhamija, A. Perrig, "Déjà Vu: User study using images for authentication", *Ninth Usenix Security*

Symposium (2000) 14-17.

[11] G. Elias, G. Sherwin, J. Wise, "Eye movements while viewing NTSC format television", *SMPTE Psychophysics Subcommittee*, white paper (1984).

[12] J. Findlay, "The visual stimulus for saccadic eye movement in human observers", *Perception* (1980) 7-21.

[13] D. Hong, S. Man, B. Hawes, M. Mathews, "A password scheme strongly resistant to spyware", *Proc. International Conference on Security and Management*, Las Vegas NV (2004) 94-100.

[14] I. Jeremyn, A. Mayer, F. Monrose, M.K. Reiter, A.D. Rubin, "The design and analysis of graphical passwords", *Proc. 8th Usenix Security Symposium* (1999)

[15] W. Ku, M. Tsaur, "A remote user authentication scheme using strong graphical passwords", *IEEE Conference on Local Computer Networks* (2005) 351-357.

[16] Jiebo Luo, Amit Singhal, "On measuring low-level saliency in photographic images", *Proc. IEEE Conference on Computer Vision and Pattern Recognition* (2000) 84-89.

[17] R. Morris, K. Thompson, "Password security. A case study", *Comm. ACM* 22 (1979) 594-597.

[18] W. Osberger, A.J. Maeder, "Automatic identification of perceptually important regions in an image", *Proc. 14th International Conference on Pattern Recognition* (1998).

[19] "The Passfaces System", Real User Technology and Products, (2004); http://www.realuser.com/published/RealUserTechnologyAndProducts.pdf

[20] A.S. Patrick, A.C. Long, S. Flinn, "HCI and security systems", *Proc. SIGCHI Conference on Human Factors in Computing Systems* (2004) 24-29.

[21] J. Senders, "Distribution of attention in static and dynamic scenes", *Proc. of SPIE*, 3016 (1997) 186-194.

[22] L. Sobrado, J.C. Birget, "Graphical passwords", *The Rutgers Scholar, An Electronic Bulletin for Undergraduate Research*, vol. 4 (2002).

[23] X. Suo, Y. Zhu, G.S. Owen, "Graphical passwords: A survey", *21st Annual Computer Security Applications Conference* (ACSAC'05) (2005) 463-472.

[24] J. Thorpe, P.C. van Oorschot, " Towards secure design choices for implementing graphical passwords", *Computer Security Applications Conference* (2004).

[25] M. Tkalcic, J.F. Tasic, "Colour spaces: perceptual, historical and applicational background", *EUROCON 2003, Computer as a Tool* (2003) 304-308.

[26] D. Weinshall, S. Kirkpatrick, "Passwords you'll never forget, but can't recall", *Conference on Human Factors in Computing Systems* (CHI) (2004) 1399-1402.

[27] S. Wiedenbeck, J. Waters, J.C. Birget, A. Brodskiy, N. Memon, "Design and longitudinal evaluation of a graphical password system", *International J. of Human-Computer Studies* 63 (2005) 102-127.

[28] S. Wiedenbeck, J. Waters, J.C. Birget, A. Brodskiy, N. Memon, "PassPoints: Design and longitudinal evaluation of a graphical password system", *International Journal of Human Computer Studies* (2005) 102-127.

[29] A. Yarbus, *Eye Movements and Vision*, Plenum Press, New York, NY (1967).

[30] J. Zhao, Y. Shimazu, K. Ohta, R. Hayasaka, Y. Matsushita, "An outstandingness oriented image segmentation and its application", *ISSPA* (1996) 45-48.

[31] J. Thorpe, P.C. van Oorschot, "Human-seeded attacks and exploiting hot-spots in graphical passwords", TR-07-05, School of Computer Science, Carleton University, (Feb. 2007), (Added in proofs).