UNIVERSITY
of
GLASGOW

Masters Thesis
Level 5, MSci, 2005/2006

Department of Computing Science
University of Glasgow
Lilybank Gardens
Glasgow, G12 8QQ

Textual Associations from Images Applied to Authentication

by

Tony McBryan

I hereby give my permission for this project to be shown to other University of Glasgow students and to be distributed in an electronic format.

Tony McBryan

Abstract

In this research project show the need for improved authentication systems. An authentication system utilising textual associations formed from images as passwords is created which relies on associative memory to recover the password at a later date. We detail a number of hypotheses and scientific questions we set out to answered as part of the research, describe the experimental design used to answer these questions and discuss our results.

Contents

Chapter 1

Introduction

1.1 Overview

This research project performs an analysis of methods of assisting the user in generating and remembering secure passwords through the use of visual cues. The studies detailed in the report seeked to determine whether using images as cues would help the user to generate a secure password through the use of associations, and determine if the user can then recreate the association at a later time to remember the password.

The basis of the work rests in exploiting superior visual memory [She67] to assist a legitimate user in remembering secure passwords while minimising the predictability of the password. A number of user studies were performed using a variety of different image types to gauge their effectiveness.

This is an important issue as users forgetting their passwords to systems can be costly to a large business. Gartner Research [Res02] estimated an average of 3.8 forgotten passwords per year. RSA Security[Inc04] estimated the cost of a forgotten password, including the cost of support personnel and lost productivity, to be approximately $58 per incident. This results in an average cost of $220 per user/per year. Reducing this cost would have tremendous cost savings for large organisations.

1.2 Problem Statement

The overall goal of this project was to devise a way of supporting the process of securely authenticating a user to allow or deny their access to a restricted resource so that users do not forget their authentication details as frequently. This most commonly done by using a secret password known only to the user. We extended this model such that this secret was a textual description of an image which the user can recreate at will by reobserving the image.

To help a user create and remember a secure password a system presents the user with an image; this image is then described by the user in textual form resulting in a password, an association will then exist between the image and the textual description meaning that the user will be reminded of their textual association when they view the image. When the user authenticates at a later date, the image will be displayed as a cue to help them recreate the association which will allow them to enter the correct password.

The images used for authentication need to be carefully chosen such that they are suitable for colour blind or impaired users and large enough for visually challenged users. The images must be easy to describe to prevent the users from experiencing unnecessary difficulty in forming associations but still be difficult for other people to guess the association. Creating large, colour blind safe images is an easy task compared to the final two constraints. The purpose of this report is to find image types that can fulfil these latter constraints.

For this type of authentication system to be successful a number of further constraints must be fulfiled. An image needs to be found that triggers associations in the user that have good properties for a password. The image should trigger a single highly memorable textual association which should be as difficult to guess as possible. The user should not need to specifically remember the association; it should be possible for the user to recreate the same association on each viewing of the image. If such an image type can be found this will enable its use as a cue for a user to enter a description of the image without giving undue assistance to an attacker.

Additionally it has been shown by Adams and Sasse[AS99] that users do not like multiple passwords which encourages them to use the same password at multiple sites. We seek to prevent this by making it easier for users to remember individual passwords through the use of cues.

By using the textual description of the image as the password we will not be trying to replace the traditional password paradigm; instead we seek to augment it, this will simplify the implementation of the system in real world applications and provide a fall back mechanism for blind users who are unable to see the image - instead of providing a description they can simply provide a password as normal. Ideally images should be parameterisable such that an entire image does not need to be stored for each user, instead a seed storing the parameters for generation of the image can be stored instead and ideally the generation of the image can take place on the client to minimise network transfer.

This approach does have a number of drawbacks which need to be understood by anyone using associative based systems. The descriptions of the images will unfortunately be more predictable than truly random passwords due to the presence of English *bigraphs* and *trigraphs* (groups of common 2 and 3 character sequences). To combat this the user must be encouraged to enter long descriptions to attain a similar level of security.

Another potential drawback with all associative authentication techniques is that a malicious website may present the user with duplicates of his images from another site and thus get them to enter the association for the image - obtaining the password for the targeted site in the process, this is a form of social engineering related to phishing[DOK04] which is still a problem with traditional password authentication systems.

A final attack would be to decrease the problem space of the password by soliciting help from a large user population to create associations for an image. This would provide clues to the attacker on likely attack vectors. However this is a much more infeasible approach than a traditional bruteforce or dictionary attack due to the requirement for large employment of manpower since computers are unable to create associations.

Our aim is to determine the viability of using image associations to effectively remember a password and to find which images are most effective for the task. We explore how each image type affects the strength of the description as well as the memorability of the different descriptions.

Our final aim is to measure the viability of this type of authentication system in an actual system and compare it to the existing password model in terms of functionality and usability.

1.3 Purpose and Significance of the Study

Developers of user authentication systems must ensure that their systems are both user friendly and secure from attack. Our goal in this project is to create a system that satisfies both these goals. We hope this study will have implications for further research on association based passwords as well as influencing new live systems to be developed using these theories.

The purpose behind this study is to improve the ability of users to easily authenticate themselves with secured systems. Security and authentication are well researched fields and several alternative graphical approaches have been developed which rely on recognition memory. We, however, seek to take advantage of associative memory. In previous graphical authentication systems the user has been asked to select an area of an image or a combination of images they previously chose, unfortunately this form of authentication is vulnerable to an "over the shoulder" attack where a malicious user can simply observe which area or images are selected. This vulnerability is especially helpful to a determined attacker who may use sophisticated eavesdropping techniques to observe the computer monitors output at a distance[Kuh02, vE85].

This study is novel as creating associations from images has not been well explored. Microsoft [SS04] conducted studies in this direction using Inkblots to create associations for authentication but a study of associations created by different image types for the purposes of supporting authentication has not yet been conducted.

It has been said that the user is the "weakest link" in the security of a system, Sasse et al.[SBW01] suggest reasons for the users' behaviour in such circumstances such as social issues, limits on memory and large numbers of passwords the user is required to remember. We deal with two reasons in particular - users have limited recall memory capacity and recognition of a familiar item is easier than unaided recall. By assisting the user to remember the password we hope to alleviate the effects of this weakness during authentication.

We hope that our results will lead to further research in this area and influence the creation of more user friendly authentication processes in addition to encouraging industrial and academic developers to introduce new techniques to real user populations and encourage the adoption of more advanced cueing techniques during authentication.

1.4 Outline

The remainder of this report is organised as follows. In Section 2 we provide a summary of different authentication mechanisms and provide a critique of their advantages and drawbacks. This encompasses a discussion of the current traditional password mechanism, graphical authentication techniques and associative techniques, as well as a discussion of some general issues used when designing authentication systems. Section 3 details our preliminary hypotheses and the questions we seek to answer in our proposed study. In Section

4 we detail the proposed design of the study and the experimental methods and policies we intend on using throughout the study then in Section 5 we discuss our implementation of the system. Section 6 provides our results which are then discussed further in Section 7. Section 8 concludes and summaries the report along with our recommendations. We finalise this report with a listing of suggested future work in Section 9.

Chapter 2

Background (Related Work)

2.1 Overview

There are many applications which require the user to authenticate themselves with a central authority before being permitted to perform certain tasks. In this section we will provide a summary of common authentication techniques and drawbacks associated with each technique and go on to make specific conclusions in respect to our proposed system.

2.2 Properties of a good authentication system

The four stages of authentication are enrolment, authentication, replacement and deregistation [Ren06]. This is a formalisation of how an authentication system works. During enrolment with the system the user shares a secret, to authenticate themselves with the system at a later date this secret is repeated to confirm that it is the same person. The replacement and deregistration stages are to allow the ability for the user to change the password or have their details (and their secret) deleted from the system.

To properly evaluate the effectiveness of an authentication system we must first define the properties it requires. Jobusch and Oldenhoeft[JO89] define the main goals of an authentication system. They essentially state that the system must control access based on the user's identity and must be able to allow or restrict access based on this.

To do so the system must prevent people from pretending to be someone else, that is there must be a minimal number of false positives (ideally zero) allowing access to the system. To obtain this it is necessary that the system is strong against attack. This is defined as the system's ability to resist a repeated non-random attack by a malicious party.

The system must properly allow access to an authorised user after they identify themselves and share their secret with the system to confirm their identity; that is there must be a minimal number of false negatives (ideally zero). To fulfil this criteria the system must be designed such that the user does not forget their login credentials; a forgotten password is of no use to a user.

The system must be as simple as possible to minimise the burden on the user of identifying themselves. If a system requires a user to follow a complicated series of steps or enter routine information during every authentication procedure then the system will frustrate the user and may result in responses being written down or abandonment of use of the application.

The cost of the authentication system must be reasonable in both monetary and CPU resources. If expensive specialist equipment is required it is unlikely the system would be able to be implemented in a general sense, likewise if the authentication process requires hours of CPU time it is unlikely to be used. The cost of the system must be relative to the value of the information it protects.

2.3 Physical access restriction

The simplest and most effective method of establishing computer security is simply to make the system physically inaccessible to unauthorised users. To effectively isolate a machine from unauthorised users it is necessary to secure the computer from any physical attempts to gain access.

The most obvious attempt is a user gaining access to the computer itself. This would be typically prevented by policies and protection mechanisms designed to authenticate a user before they can gain access to the system. In a highly classified system this may be as extreme as placing the computer in a locked room protected by armed security guards.

Emissions security deals with the threat of unauthorised users intercepting and analysing electrical emissions from secured equipment from a distance to obtain classified data. To protect against this it is necessary to shield the equipment to block electromagnetic radiation. Spurious emissions are covered under FCC regulations[FCC] for standard commercial equipment but these regulations are insufficient for highly secure systems as it is necessary that any radiated emissions have minimal correlation to the data that is being processed which is not covered in the FCC regulations.

Additional threats become possible when the system is connected to a computer network. In addition to ensuring that the unauthorised users are unable to gain access to the physical system itself it is necessary to restrict access across the network or make it otherwise impossible for the attacker to use the network through encryption. Ensuring physical security to a single system is a much easier task than ensuring security of a distributed network across multiple sites potentially across large distances.

Physical security is clearly not appropriate for the majority of systems where it is necessary for a user to access the system remotely or in the trivial case of a user wishing to purchase an item from an online retailer. The Department of Defense provides criteria[oD85] on how to properly secure against physical attempts to gain access to the system. The National Institute of Standards and Technology provide guidance on how to apply the DOD standard in a number of different environments[oST].

2.4 Traditional password systems

The traditional authentication system is the password system. In this a user has a userid which is used to identify the user and a secret password combination which is used to authenticate the user. The system will typically prompt for the userid/password combination and check it against an internal list of authorised users. The secret password is the login credential used by the user to prove their identity to the system.

The password model has proven itself throughout history as a simple and secure method of controlling access to a system when correctly implemented and used. The password technique has been documented in use for centuries, from guards asking for passwords for entry[Smi02] to Ali Baba's "Open, Sesame"[Gal10]. When a good password is used the attacker will have a much harder time breaking into the system [YBAG00].

Unfortunately passwords have some drawbacks which cause researchers to look for new and improved methods of authentication. Essentially, users pick bad passwords. If the user is allowed to choose their own password they commonly choose passwords from within a limited domain. Morris[MT79] found that large numbers of passwords were 3 characters or less or present within electronic dictionaries. This was confirmed by Klein[Kle90] who found that even checking in only a small dictionary and using variations of the username almost 25% of passwords were found.

The obvious solution to this would be to assign users precomputed passwords we know to be secure. Unfortunately, if the user is assigned a random password by the system then they are liable to forget it [YBAG00]. To prevent forgetting the password the user may write the password down which is just as bad as it then allows anyone who finds the password list to access the system and masquerade as the user. This is especially bad if the written down password includes the name of the system, the userid or takes the form of a “post-it note” on the monitor.

A second technique to reduce the frequency of “bad” password choices would be for the administrator to run their own attacks on the password lists to try and find problems in advance [RU88]. This would, at first glance, seem to be an ideal method for eliminating bad choices for passwords; however it has several drawbacks. The first drawback is that it simply takes a long time to check the list, secondly the user can still choose a weak password that is not in the list you are using to check the passwords. A neat solution is the OPUS method[Spa92] for preventing weak password choices using a fast hash based lookup table to screen passwords as they are entered although it still cannot guarantee a user has not chosen a bad password if the password list is not complete. The final problem is that if the user continually chooses bad passwords the user may become frustrated if every password they think of is on the banned words list. We wish users to choose to use a secure password as a matter of course without needing to force them into it.

A brute force attack [CDW04] is the process of trying every possible combination (or a subset of all possible combinations) of characters and digits to break a password. This leads us to the conclusion that the rate that someone can check if a password is correct is a large factor in the security of a system, thus an offline attack where the attacker can devote large amounts of processor power to the task is most effective for attacking passwords. Kedem[KI99] shows that a SIMD machine created with modern technology can break any UNIX password within two days of processor time. Brute force attacks can only be reliably defeated by restricting the rate at which someone can attempt the password, this can only be done reliably by preventing offline attacks (where the attacker can attempt to break the passwords without continual access to the original system - for example by obtaining the hashed password file on a UNIX system) and extending the amount of time it takes to present a password to the system by an arbitrary amount of time (enforced wait periods between attempts) or enforcing a fixed limit on the number of failed password attempts per account.

As passwords are often used in protecting very confidential information it is essential that a user uses a password that is both secure and highly memorable but it has been found to be difficult to encourage users to do this with the plain password model.

2.5 Passphrases

Passphrases is an extension of the passwords model where the user remembers a phrase of words instead of a single password. This results in a much longer identity credential which is suitably harder to break using brute force or dictionary attacks as the number of combinations is significantly higher. The modern idea of passphrases began with Porter[Por82].

An extension on the passphrases idea is to use reconstructed passwords[Has84], that is an algorithm which you use to turn a passphrase into a password. The user then remembers the passphrase and applies the algorithm to retrieve the password when required. An example would be musician who remembers first few bars of a musical score and enters it in a textual format to reconstruct the password.

Passphrases and reconstructed passwords are both highly memorable however the user may be tempted to use the same highly memorable passphrase for all of their passwords when we wish to encourage them to use different passwords for each system.

2.6 Single use passwords

The next technique we will examine is the tactic of using single-use passwords (One time passwords). To use these you create a method of constructing a series of passwords which are shared between the user and the system. When the user logs into the system they use the next password on the list. This is extremely secure, as long as the password list is kept secure, as even if someone learns the password used to login to the system they cannot log into the system as that password has expired.

The single use password system is extremely secure providing the password list is not divulged to anyone. The downside is that it is extremely inconvenient for the user and expensive for the system administrator. The administrator must generate a list of passwords and distribute them to the users in advance while ensuring the list of passwords does not fall into the wrong hands. When the list runs out an updated list must then be generated. An implementation for this exists within the UNIX system called S/KEY[Hal95] (sometimes referred to as Lamports scheme).

2.7 Pass-algorithms

An extension to the single use system is to use, again, single use passwords but instead of creating a list of the passwords in advance use an algorithm the user knows to generate the passwords, as described by Haskett[Has84]. The system provides the user with a prompt which the user applies the algorithm to provide the new password, the password may be as simple as advancing each character in the prompt by one letter or may be more complicated. This has the same advantage as the single use password system in that even if a particular password is discovered it is not useful for logging into the system. Adding several factors into the algorithm such that the solution to the algorithm is different depending on which machine is used to login, can add to the difficulty in cracking the algorithm through observation alone. This does, however, suffer a similar problem to single use passwords - it is hard for users as it forces users to perform complex mental tasks to login to a system.

2.8 Biometrics

The most ancient authentication system was the simple biometric of “I know that person”. Computers are now beginning to be able to take advantage of the same techniques people have had for hundreds of years. Biometric systems are now able to distinguish between users with high accuracy using several biometric techniques. Available techniques include face recognition, fingerprint scanning, hand geometry, iris and retinal scans, hand written signatures, voice analysis and facial thermograms. It is beyond the scope of this paper to discuss the advantages of each method but suffice to say they are very good at authenticating users. For a detailed discussion of biometric identification systems we recommend the paper by Jain et al.[JHP00].

Biometric systems do, however, have some drawbacks which make them difficult or impossible to use in many circumstances. The first and most obvious problem is that they typically require specialist devices to perform the authentication itself; some systems also involve high amounts of processor usage in order to perform comparisons between the biometric details stored on record and those presented by the user.

A more subtle point is that users simply don’t like biometric systems. Users are accustomed to using passwords and similar systems; they do not want to deal with systems which record personal data about themselves citing them as examples of “big brother”. Furnell[FDIR92] shows that although users are willing to experiment with new techniques they still prefer the password model by a large margin over biometric techniques.

2.9 Graphical authentication systems

Studies in 1960s by Shepard[She67] and Nickerson[Nic65, Nic68] show that people have a very high ability to recognise large numbers of images both immediately after viewing and after a long term delay. This shows us that graphical authentication systems utilising the ability of a person to recognise an image can be used to ensure high memorability authentication systems.

Several alternative graphical based authentication schemes have been presented by research groups. Perhaps the first graphical authentication system was suggested by Blonder who presented a system which used images with hotspots known as "tap areas" which the user recognises and selects in order to authenticate themselves[Blo96]. Issues with this technique include the accuracy the user requires in order to select the tap area correctly again, when used with a small tolerance to force the user to click very close to the correct tap areas the performance of the system drops significantly[WWB$^+$05].

The Passface[BS00] system has a user create a password from a selection of human faces, at login the user is presented with a grid of human faces including the chosen face and a number of distractor faces where the user had to select the correct the face. This process was then repeated 4 times. The Passface system was found to have good memorability and low error rates for login attempts.

Motivated by PDA systems the Draw A Secret[JMM$^+$99] system was developed by Jermyn et al. where the user draws a symbol or sequence of symbols on a touch screen. They show that the images drawn by the users are both highly memorable and possess a large domain of images that are memorable. Since the image drawn is highly visible on the display screen it is necessary to shield the display screen from onlookers thus restricting the technique to portable devices such as PDA's.

The Déjà Vu system [DP00] implements the graphical authentication technique by providing the user with "random art" images and are asked to select a set of images they will use for logging into the system. When the user logs into the system they then select their chosen image from a set of decoy images.

Goldstein and Chance[GC70] tested memory recall using Faces, Inkblots and Snowflakes. They discovered that faces had superior recognition followed by Inkblots and then Snowflakes. These studies, although not concerned with the password domain, are relevant to our planned associative system as they offer us some clues as to which images may perform best for memory recall purposes.

The biggest problem with the above systems is that they are vulnerable to “over the shoulder” attacks where someone can simply observe the combination and ordering of images used to login to the system. This problem is not unique to graphical authentication systems as it is still possible to observe finger movement of a slow typist to discern a password in a traditional password system although the traditional password model avoids direct over the shoulder attack by hiding the entered password behind asterixes or a similar typographical symbol or glyph which is not possible when the user needs to click on an images. The traditional password model also fails in this respect as keypresses can be logged by specialised hardware or software to reconstruct the password.

An additional problem is the small number of combinations available. If you have 4 stages of 16 images each there is a mere 65536 combinations which is trivial for a computer to brute force if no restriction is placed on maximum number of tries. It is therefore required to protect the system further using either delays between attempts, a maximum number of attempts or increasing delays between attempts to discourage automated brute forcing of the combination.

2.10 Cognitive passwords

The use of cognitive passwords is the process of asking the user questions based on their personal facts, interests and opinions. The system can keep asking the user questions until it is “convinced” that the user is who they say they are. This method is by its very nature easy for the user as they are being asked questions they already know the answer to rather than having to try and remember a password. Zviran and Haga[ZH90] found that users had a very easy time in answering the questions and high correctness rate.

Again, this technique has drawbacks. The simplest of which is that the attacker can research the details of the person who owns the account. Most of the answers for the questions will likely be easily found or obtained, it is possible that you could obtain the answers to some of the cognitive questions by simply asking the account owner as they are not likely to protect the information about their favourite colour as heavily as they would a password. Podd et al.[PBH96] found cognitive passwords were unacceptably easy to guess by an attacker.

The second problem is that because the questions deal with the user’s interests and opinions is that the answers to these questions may change over time, they may switch allegiance to another political party or simply change their mind on what their favourite colour is. Thus you need to ensure that the list of questions is kept up to date to prevent expiry of the answers.

2.11 Associative systems

The system we intend to build is closest to an associative system but seeks to take advantage of memorability features within the graphical authentication systems. An associative system provides a cue to a user who then creates a single association, typically a description or “feeling” created by the cue which is then used to decide the password for the system. When the user then views the same cue at a later date they can recreate the same association which helps them to remember the password.

Smith[Smi87] presented a system where the users created a list of 20 cue words and associated responses and were later tested 6 and 18 months later. Although it was a small sample size the cued responses had a very high correctness rate.

Studies on word association using different formulation techniques for choosing how the cue words chosen[PPBH00] has been carried out which concluded that the method of choosing the word associations resulted in no significant difference to the effectiveness of the association implying that choosing associations from images should be just as effective as word associations.

Liddel et al.[LRA03] attempted a system where the user performed Sound to Image associations. Experiments were conducted which showed poor performance for their proposed system resulting in a dissatisfied user base. They showed that the authentication system required much more effort and took much longer than comparable systems. It was shown that the process of creating an association from a sound to an image caused users to commonly choose the same resulting image - in one case 1/3rd of users chose the same image from a grid of 10). We believe this to be due to the limited number of choices the association was forced to be represented in. Associations from music represented in textual form may be more secure.

Microsoft developed an associative system based on inkblots[SS04]. The original Rorschach inkblot test was developed as a projective test which encourages descriptive associative responses however its results in therapy are no longer trusted[Ana82]. Additionally precise details of the test have been leaked to the Internet which allows people to prepare for the test which was not previously possible[S.P05].

In the Microsoft system users are presented with a set of inkblots where the user takes the first and last letter of their description of each inkblot and concatenates them to create a textual password. This is therefore immune to the “over the shoulder” attack as the authentication can be entered textually. A drawback in the Microsoft study is that it requires the user to be able to recreate multiple associations for every password which would be more difficult for the user than recreating a single association. An additional drawback with the Microsoft system is that it is cognitively demanding on each login attempt - requiring users to create multiple associations and then perform a mental hashing function on them. The Microsoft study on inkblots is the only system in the literature that used associations from images to help users to generate textual passwords.

It has been found[PBH96] that associative based passwords are hard for an attacker to guess, despite their memorability. However, most implementations upon associative methods rely on the user recreating many associations, which may be difficult. An additional constraint is that the associations must be non-trivial[ZH93]. For example if a user was given a cue of “good” then the obvious association of “bad” would result in the system be easy to penetrate. This result implies that if these problems can be solved then associative based passwords could be therefore highly resistant to attack while maintaining high memorability.

2.12 Usability Issues

ISO Standard 9241-11 [Org98] defines the usability measures used to evaluate a system as effectiveness, efficiency and satisfaction. These measure the accuracy and completeness that a user performs a task, the amount of work that needs to be done to complete the task and the attitude of the user towards the system. These three points measure the essentials of what needs to be done to make a system usable and will be used during user evaluations to measure the ease of use of the system.

Brostoff and Sasse demonstrated the difficulty of implementing alternative authentication systems by performing a usability study of the Passfaces system[BS00] which highlighted the requirement of performing field trials. They showed that although users had a lower error rate when using the Passfaces system they also took longer to login and logged in less frequently than systems using the traditional password; indicating that the passfaces system was more work than the traditional system and discouraged people from logging in. We will need to take this into account when evaluating our proposed system.

Smith and Mosier defined a set of guidelines for user authentication[SM86] where they direct that user authentication procedures should be as simple as possible. They also provide several security related recommendations such as private entry and storage of passwords and limitations on failed login attempts which we will use as a heuristic guideline to ensure our system is easy to use.

2.13 Conclusions

We found that all of the above mentioned systems had advantages and drawbacks. Some systems could be implemented without problems within specialised areas (physical security and biometrics), some of the systems had drawbacks that we would prefer to avoid but allow implementation almost anywhere (traditional, passphrases, single use, pass-algorithms, graphical, associative) and cognitive systems were generally too insecure for general use.

Physical access restrictions are unsuitable for the majority of systems but can be implemented on a number of highly secure systems which do not need remote access. It is therefore not a general solution to the authentication problem.

The literature discussing traditional password systems finds that passwords are unsatisfactory, mostly due to user choice of passwords[MT79]. We need a way to encourage the use of longer or more difficult to attack passwords, but traditional password systems are the most widely accepted mechanism by users[FDIR92] so we do not wish to deviate much from this model.

Single use passwords and Pass algorithms are highly secure however result in high amounts of work and inconvenience put upon the user and administrator which is undesirable. Implementation of either of these systems would need to have a valid justification for imposing this load such as large financial loses for breach of security.

Biometrics are very promising however they have three major hurdles to overcome before they can be put into more general use. The first hurdle is the cost of the specialist devices required to measure biometric data from the user. Secondly the general population do not like overly invasive systems, however it has been shown by Furnell[FDIR92] that users may accept these techniques more readily in some sectors such as Defense or Government so biometrics may have applications within these areas. Use of biometrics may then encourage their uptake by other sectors. The final hurdle is the impossibility of implementing biometrics within an insecure environment where the biometric devices can be tampered with (for example where fingerprints can be retrieved from the scanner and reused).

It has been shown that graphical systems have very high memorability due to the human ability to recognise images. However, graphical authentication systems suffer from the "over-the-shoulder" security vulnerability which renders them unusable for high security applications.

Cognitive passwords have easily the highest memorability due to the fact that the password is a detail about the persons life, however this also results in the lowest security as these details can be trivially researched about the user resulting in it being unsuitable for use for most applications[PBH96].

Associative passwords have high memorability although if trivial associations are created it can reduce the security of the system. Additionally recreating high numbers of associations may have a high workload. Associative systems may work best with the user creating a single non-trivial association.

A study by Zviran[ZH93] compares system generated passwords, self generated passwords, passphrases, cognitive and associative passwords and finds that associative and cognitive systems are the best for the purposes of second level passwords. As previously shown cognitive answers are far too insecure to use. Thus we come to the conclusion that associative systems are the most valid path for research.

We therefore concluded that there was a need to investigate a system which allows users to generate textual associations from images in an attempt to capture all the advantages of graphical, associative and traditional passwords. Our system provides the user with an image as a cue which is then used to create a textual association to be used as a password. Accurate reproduction of this textual association at a later date will then allow access to the system.

The presence of images should help to exploit graphical recognition memory while the security of an associative system is obtained by allowing the user to create their own personal associations from the image to use as their password. Since the associations come from an image they should be non-trivial and therefore unpredictable. The use of textual associations allows the ease of use of the traditional password model to be apparent; additionally the familiarity with traditional password systems should minimise the learning curve of a new user to the system. By doing this we would expect to take advantage of the memorability of graphical systems, the security of associative systems and the ease of use of the traditional password model.

We will detail a number of hypotheses we created for the behaviour of such a system in the following section, followed by construction of a sequence of scientific questions derived from these hypotheses, we will detail an experimental design to answer these questions and detail our results.

Chapter 3

Hypotheses/Questions

3.1 Hypotheses

From the work detailed in the previous section we hypothesised that our proposed system would have a series of expected properties. Firstly, and most critically we hypothesised that the use of images as cues is useful for helping the user to construct associations and that when shown the same image at a later point in time the user is able to recreate the association.

> *Images can be used as cues to help users to generate and remember textual associations. (Hypothesis 1)*

We further hypothesised that these associations will be useful in the authentication domain. This hypothesis is reliant upon the associations chosen by users being strong enough to resist repeated non-random attack due to the unpredictability of the association.

> *Associations chosen will be strong enough to be used as passwords. (Hypothesis 2)*

We believed that certain types of images would be superior for the purposes described above. This hypothesis is influenced by the work of Goldstein and Chance[GC70] who showed that different types of images have different rates of recognition and association.

> *Some types of images will be superior to other types of images. (Hypothesis 3)*

Based on cited research in Section 2.9 we hypothesised that due to the projective nature of inkblots they would encourage high strength descriptions as each person's description should be unique. We hypothesised that human faces would have a high memorability of associations due to the human ability to recognise other people by face quickly and reliably.

Inkblot-type images will encourage high strength descriptions. (Hypothesis 3a)
Human faces will have high memorability of associations. (Hypothesis 3b)

We then further hypothesised that variations within an image class will affect the performance of the image.

Variations within the image class will result in differing performance. (Hypothesis 4)

To test this we decided to measure the strength of descriptions generated by users and the memorability of the same descriptions. We believed that these properties will not necessarily be exclusive and that some image types will show evidence of encouraging both high strength descriptions and high memorability of descriptions at the same time. These images would then be suitable for use within an authentication system and offer superior performance than the traditional password model. To do this the image type needs to match, or exceed, the strength and memorability of the traditional password without compromising ease of use for the user.

Some image types or variations will have superior memorability and unpredictability over the traditional password model. (Hypothesis 5)

This can be represented as two sub-hypotheses as follows:

There will be images with high memorability of associations. (Hypothesis 5a)
There will be images with high strength. (Hypothesis 5b)

Since this system is based upon the well used password model we would expect to find that the system would be no more difficult to use than a traditional password based system. To measure this accurately we would be required to create a live system and monitor the frequencies and success rate of logins to the system compared to traditional passwords. This would also incorporate measurements of the time it takes to login.

The proposed system will be no more difficult to use than traditional passwords. (Hypothesis 6)

To assess the effectiveness of these techniques we must define what properties of the image would be required to create a useful authentication system. Clearly for this system to be incorporated into real systems the properties of security and ease-of-use must be no worse than the traditional password model and at least one, or both, must be better by a degree large enough to warrant implementation. To do this we will measure the performance of each property compared to the traditional password system.

Our research is then based around the task of checking the correctness and extent of these hypotheses.

3.2 Questions

In order to design experiments to check our hypotheses we needed to define the scientific questions that we intended to answer. These questions are based directly upon the hypotheses in the previous section.

1. *When presented with an image as a cue are users able to generate textual associations?*
2. *When presented with the same image after a specific duration of time can the user recreate their textual association?*
3. *Are textual descriptions based on images unpredictable enough to be used as passwords?*
4. *Do some image types offer better memorability than other image types?*
5. *Do some image types offer better security than other image types?*
6. *Do inkblot-type image encourage higher strength descriptions than other image types?*
7. *Do human faces encourage higher memorability than other image types?*
8. *Do variations within the image class result in altered memorability than other images within the class?*
9. *Do variations within the image class result in altered security than other images within the class?*
10. *Is the system more difficult to use than the traditional password systems?*

We attempted to answer these questions in the order provided above. Images help authentication and then seek to discover how the strength of the textual descriptions compares to randomly generated passwords, user generated passwords and passphrase based passwords. We then investigate their memorability followed by varying the properties of the image to check if any specific parameters involved in the creation of the image type affect its results.

Finally we tested the usability of the system in a live environment and assess the level of work required to use the system.

3.3 Image Type Selection

Human perception of shapes and groups of shapes is governed primarily by the Gestalt[WB58] laws which explain how grouping works for perception and recognising patterns within an image. In the Gestalt laws the five factors which affect still images are Closure, Continuity, Proximity, Similarity and Symmetry. These are governed by the law of *prägnanz* which states that the image will be experienced in the "best" Gestalt way possible. The better the Gestalt laws apply to an image the more likely the image is to be experienced as a whole rather than as a collection of individual elements. The laws of Gestalt are summarised below.

Law	Preferred Images
Closure	Images which produce "closed" rather than "open" figures are preferred.
Continuity	Images which produce smooth contours rather than sudden changes of direction are preferred.
Proximity	Images where features are close together are preferred.
Similarity	Images that have similar sections or self-similarity are preferred.
Symmetry	Images that are symmetrical are preferred.

In order to allow users to make associations more easily we chose image classes that typically exhibit a set of Gestalt laws. We examine the following image classes and list the Gestalt laws that they exhibit and we believe make them viable experimental candidates.

Image Type	Closure	Continuity	Proximity	Similarity	Symmetry
Faces	√	√			√
Fractals		√		√	
Inkblot	√	√	√	√	√
Snowflakes	√	√	√	√	√
Textures	√		√	√	√

We believe that these were the best choices for performing our experiments as they should encourage good associations due to each image type exhibiting at least two Gestalt laws.

In the next section we will show how we evaluated these image types by building a framework application to determine the best image type to use for authentication. We will also demonstrate how we compared this image type to the traditional password model in the third stage of experiments.

Chapter 4

Design - Methods & Procedures

4.1 Introduction

In the previous section we developed a number of hypotheses and scientific questions which needed to be answered as part of this research. In this section we will detail the methods we used to answer these questions. In order to answer the questions from the previous section we performed several experiments as detailed below.

Question	Method
1	Present users with an image and capture a textual association.
2	Present users with an image, capture a textual association and test if they can reproduce it later when prompted with the same image.
3	Present a large number of users with images, capture their associations and measure the unpredictability of individual associations.
4	Perform the studies in questions 2 and 3 using multiple image types and test if any particular image type has superior memorability or unpredictability.
5	Using the data from question 4 measure if inkblot-type images have higher strength descriptions than other image types.
6	Using the data from question 4 measure if human faces have higher memorability than other image types.
7	Perform the study from question 4 with multiple image types and variations within the image types to test if different parameters used when generating the image cause different memorability or unpredictability.
8	Create a live system containing content of interest to the user and protect it with 50% of the users using traditional passwords and 50% using the proposed system to measure users comparative ease-of-use.

For practical purposes some of the experiments will be performed in conjunction with other experiments. We performed these studies in two distinct phases consisting of a total of three experiments to answer all the above questions. Each phase of the project consists of building and testing of a system designed to answer a question or set of questions resulting in two systems.

We quantify the “goodness” of the image type using a number of factors; primarily quality of textual association in terms of password strength and memorability of the association. We then judge the best image type by finding the image with the best average performance of password strength and memorability.

For the live system evaluation we introduce additional satisfaction metrics to measure the performance of the system in comparison to the traditional password system; these are: time taken to login, number of failures, number of consecutive failures, number of successes and number of consecutive successes.

4.2 Phases

4.2.1 Phase One

The first phase of the project investigates the types of images and their effectiveness at allowing users to create effective associations that are usable as passwords, this corresponds to the scientific questions 1 to 7 in Section 3.2. This was done by creating a framework application which allows us to perform user evaluations and measure the performance of the user.

We used the framework application to gather a large number of associations created by users from sets of images. This allows us to analyse the strength of passwords generated from these images to determine which image types are worthy of further study.

We then performed a user evaluation using the same images from the previous experiment to measure memorability of images to measure the users ability to recreate the association.

The above experiments took place using the five classes of image discussed in Section 3.3 with six variations created by varying the parameters of the image to allow us to determine the performing image types, determine if any image types significantly outperform the other images and determine how any variations in the image type affect the security or memorability of associations generated from it. This resulted in a total of 30 images which are shown in Section 5.1.

4.2.1.1 Framework

Framework Overview
This frame work will take the form of an application which presents the user with an image and prompts for a the user to create an association in the form of a description of the image.

The framework is suitable for tasks which can be carried out by the users at a single session with the framework application and which can be verified by an evaluator at a later date without needing further contact with the user. This will be used for gathering a large number of descriptions of images to test their strength. Uniqueness of results is obtained by calculating a one-way hash of the users details and using this as the key for the results, if a user tries to resubmit a second run of results for the same experiment they will be disregarded. Implementation of the framework will also incorporate the writing of algorithms suitable for generating the images we wish to study or obtaining sample databases of suitable images such as human faces.

The framework initially provides a help screen for the user which will provide instructions for the task and then display a number of images for the user to enter their descriptions. The user will be timed in several respects, including total time taken as well as time taken to create and enter the association. Results will be stored in an SQL database for analysis.

Although some of the image types can be generated in real time by an algorithm the framework instead stores the actual images used in the evaluation requiring the images to be created in advance.

4.2.2 Phase Two

The second phase of the research consists of the implementation of a live system utilising the “best” image type to compare its performance against the traditional password system. This required the implementation of a more complete system as Brostoff et al.[BS00] demonstrated that the authentication system can only be adequately measured for usability in a full implementation where the users have the choice of how often they use it.

We implemented an authentication system utilising the inkblot image type alongside a traditional password authentication system. We can then use this to compare the ease of use, security and memorability of the system compared to passwords in a live setting. We developed an RSS Feed aggregation system to encourage users to use the site; clearly a content-free site would not be used by either the traditional password system or our proposed system. We also offered a monetary incentive for using the site in the form of a lottery.

The inkblot images in this stage are generated in real time allowing a unique inkblot to be given to each user of the system. It has been suggested by Microsoft[SS04] researchers to use Computer Generated Watercolour [CAS+97] to make generated inkblots look more authentic but this uses several orders of magnitude more computing power than is feasible for real time generation.

4.2.2.1 User Study

We believe that a user study and evaluation are a necessary and sufficient method of comparing this authentication system to the traditional password system. This is because the system relies heavily on user abilities and preferences which cannot be adequately modelled in any other fashion.

4.3 Diagrammatic Overview of Phases

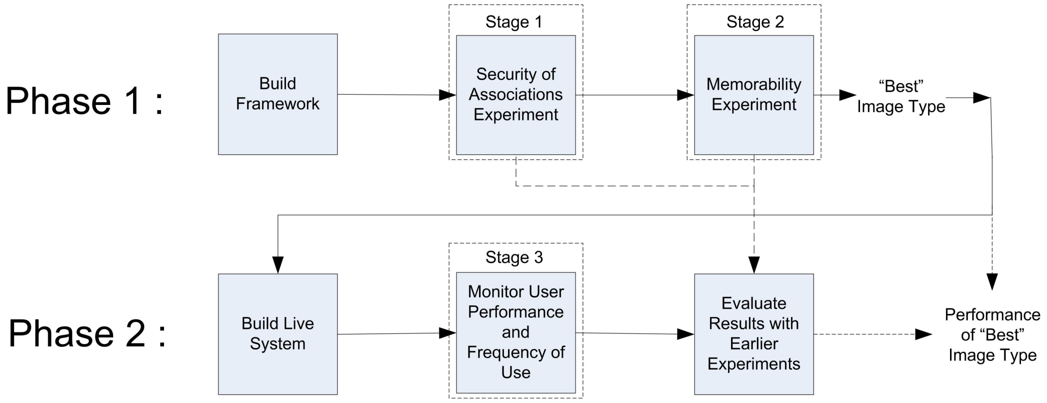

Phase 1 :
Build Framework
Stage 1
Security of Associations Experiment
Stage 2
Memorability Experiment
"Best" Image Type
Phase 2 :
Build Live System
Stage 3
Monitor User Performance and Frequency of Use
Evaluate Results with Earlier Experiments
Performance of "Best" Image Type

Figure 4.1: Diagrammatic Overview of Phases

4.4 Sampling

As a large proportion of our work was heavily user study oriented we must carefully sample the population to match the target population of an authentication system which will be the general PC using population. Our accessible portion of this is, however, considerably smaller. We sampled from as large an age-range as possible. For practical purposes we did not pursue users are not native English speakers to ensure that the description of the task is adequately conveyed.

We attempted to use as large a sample population as feasibly possible. We distributed the framework application as widely as possible to obtain a large number of image descriptions to evaluate strength however we were forced to use a smaller sample for the memorability tests due to time constraints.

The study was approved by the Departmental Ethical Review board to ensure that Ethical considerations were met while performing this study. All test users were informed fully as to the amount and type of information being gathered and allowed to end the evaluation at any time with any partial results to that point in time being disregarded. We ensured that all users are exercising informed consent and provided an ethical statement to each user prior to the evaluation which informs them of the purposes of the study and their right to end it at any time. The ethical statement will take the form of the introduction screens provided during the evaluations.

4.5 Data collection

Data was collected and processed on a central server running an SQL compatible DBMS to facilitate ease of analysis.

Factors we identified as variables affecting the outcomes of the study are listed here. The first is the description of the task for the users, this must be carefully created to ensure that the user fully understands the purpose of the task and follows instructions correctly. Additionally the description of the task cannot change between user populations or it may negatively affect the reliability of the results. The clarity of the words used in the description and the meaning of the task is important in this regard and care was taken to ensure that the descriptions were not needlessly complex. Another variable was the knowledge possessed by the individual user performing the evaluation; this was countered by collecting data from a large enough sample set for this to be statistically insignificant.

4.6 Analysis

We analysed the results primarily by measuring statistical significances between the performances of types of images. Each image type will be compared against competing types of images and a standard password system. We will obtain statistics on traditional password usability from the relevant literature as well as from our study in phase two. By comparing the results to a traditional password model we can judge the effective improvement that a user gains by using the proposed system.

4.7 Conclusion

In this section we presented a design for the purposes of answering the scientific questions presented in Section 3; the next stage of the research is therefore be the actual implementation of our design and its use to answer the questions.

In the next section we will discuss the implementation of this design followed by the results we obtained from using the system.

Chapter 5

Implementation

5.1 Images

This section details the source of the images used in the first phase of the experiment as well as displaying the resulting images. For the sake of conciseness the exact details of the images and parameters used to generate the images are located in appendix 10.1.

5.1.1 Faces

The face images used in this study were collected from the Essex University Computer Vision Facial Databases[Spa06] "face94" and "face95" and were chosen to represent an equal mix of male and female faces with a range of physical features. Only images that were clearly visible with similar scale and without distracting backgrounds were considered.

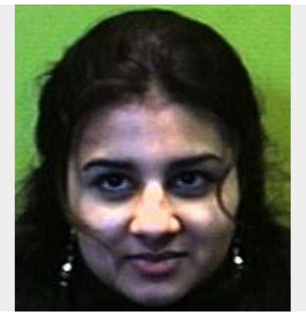


Figure 5.1: Face 1 (Image 1)

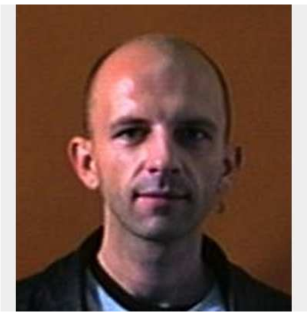


Figure 5.2: Face 2 (Image 2)

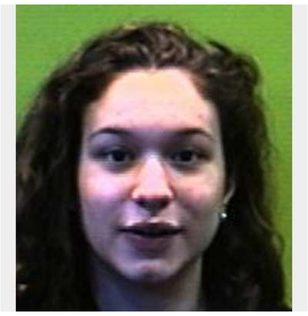


Figure 5.3: Face 3 (Image 3)

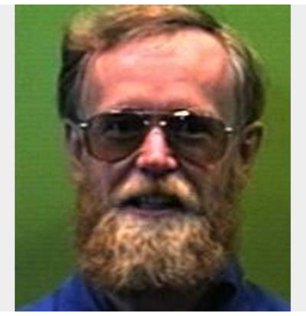


Figure 5.4: Face 4 (Image 4)

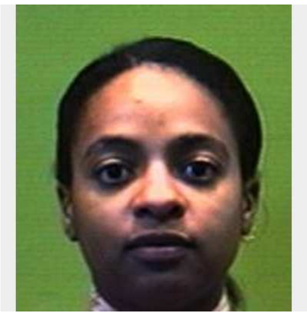


Figure 5.5: Face 5 (Image 5)

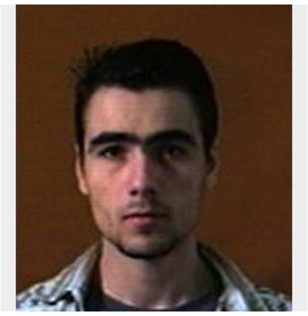


Figure 5.6: Face 6 (Image 6)

5.1.2 Fractal

The Fractals were generated using a commercial program Ultra Fractal[Sli05]. Variations within the image class were obtained by changing the algorithm used to generate the fractal in addition to varying the viewing position and colouring algorithms.

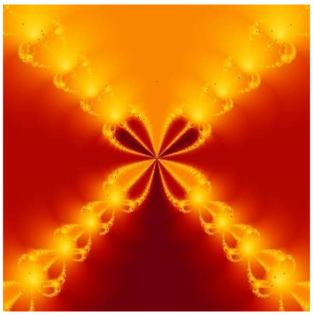


Figure 5.7: Fractal 1 (Image 7)

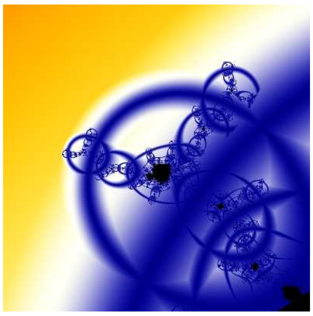


Figure 5.8: Fractal 2 (Image 8)

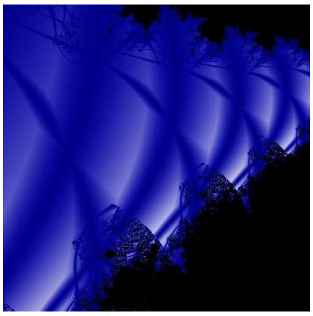


Figure 5.9: Fractal 3 (Image 9)

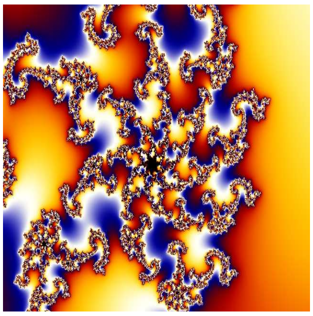


Figure 5.10: Fractal 4 (Image 10)

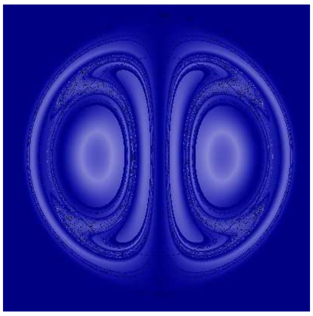


Figure 5.11: Fractal 5 (Image 11)

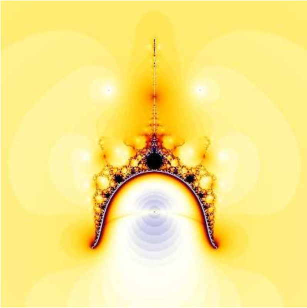


Figure 5.12: Fractal 6 (Image 12)

5.1.3 Inkblots

The inkblots were generated by a custom PHP script. The inkblots were built by dropping ”blots” onto a canvas and ensuring the next blot landed within a fixed area of the previous blot. The images were varied by changing the values of variables which control the number of blots, blot diameter, colour and distance between the blots.

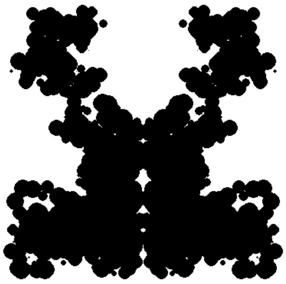


Figure 5.13: Inkblot 1 (Image 13)

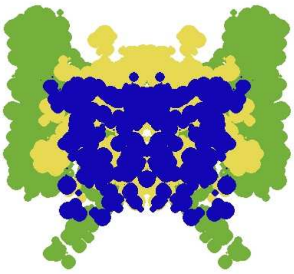


Figure 5.14: Inkblot 2 (Image 14)

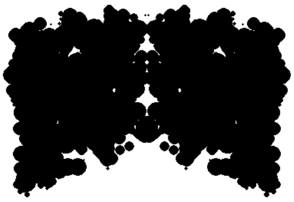


Figure 5.15: Inkblot 3 (Image 15)

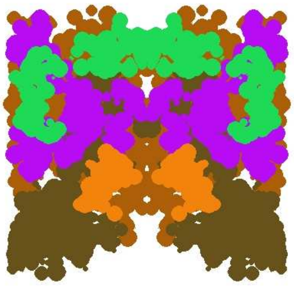


Figure 5.16: Inkblot 4 (Image 16)

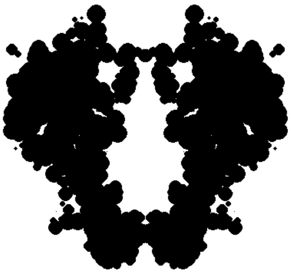


Figure 5.17: Inkblot 5 (Image 17)

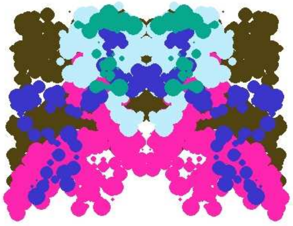


Figure 5.18: Inkblot 6 (Image 18)

5.1.4 Snowflakes

The Snowflake images were generated using A.I. Studio Snowflake Generator[A.I04] and variations within the images were acheived primarily by varying the number and complexity of the rays along with scaling and position details.

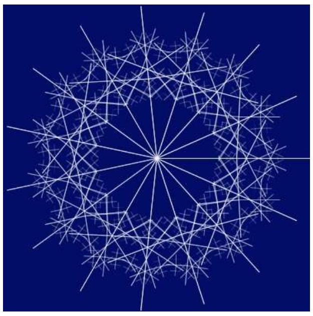


Figure 5.19: Snowflake 1 (Image 19)

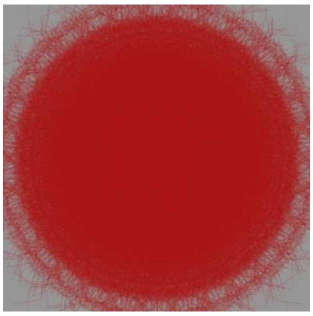


Figure 5.20: Snowflake 2 (Image 20)

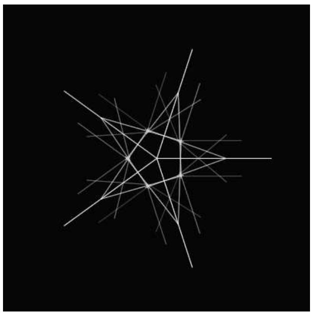


Figure 5.21: Snowflake 3 (Image 21)

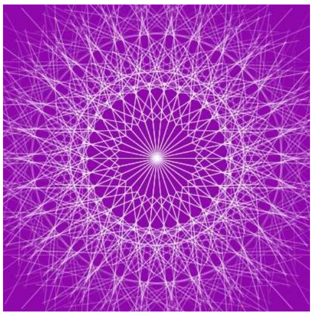


Figure 5.22: Snowflake 4 (Image 22)

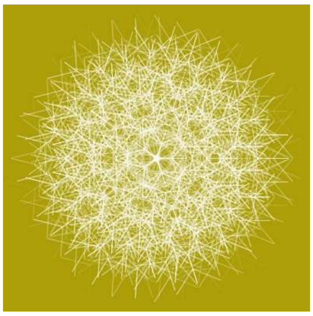


Figure 5.23: Snowflake 5 (Image 23)

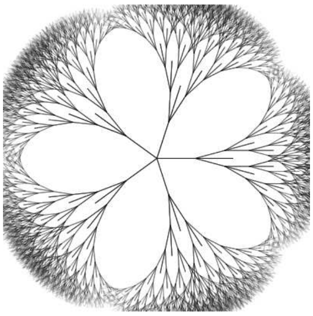


Figure 5.24: Snowflake 6 (Image 24)

5.1.5 Textures

The Texture images were obtained from the CUReT[DGNK05] texture database and were chosen to represent a range of different textures including both man-made and natural textures.

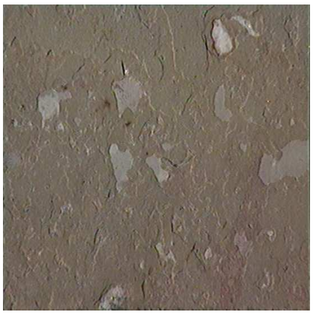


Figure 5.25: Texture 1 (Image 25)

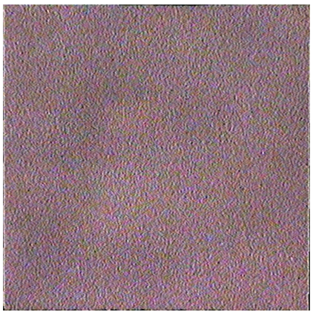


Figure 5.26: Texture 2 (Image 26)

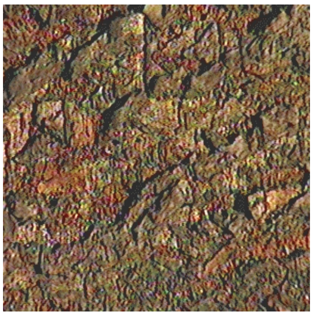


Figure 5.27: Texture 3 (Image 27)

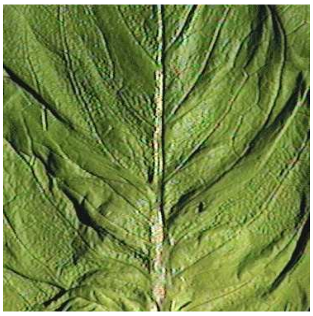


Figure 5.28: Texture 4 (Image 28)

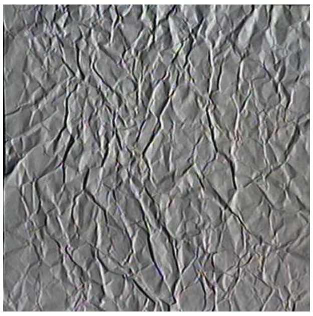


Figure 5.29: Texture 5 (Image 29)

Figure 5.30: Texture 6 (Image 30)

5.2 Deployment

5.2.1 Framework

The framework used in phase one of the research was implemented using PHP and MySQL. To satisfy ethical requirements the framework stored all state on the client side and confirmed all entries with the user before committing them to the database.

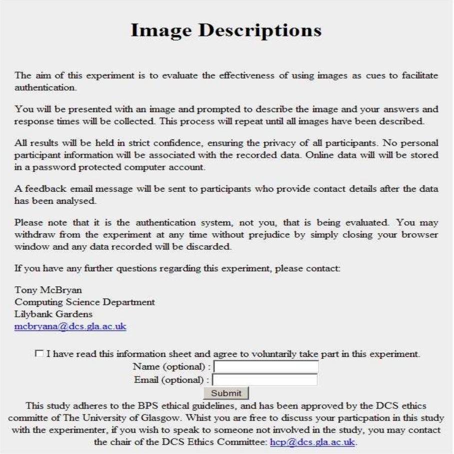


Image Descriptions

The aim of this experiment is to evaluate the effectiveness of using images as cues to facilitate authentication.

You will be presented with an image and prompted to describe the image and your answers and response times will be collected. This process will repeat until all images have been described.

All results will be held in strict confidence, ensuring the privacy of all participants. No personal participant information will be associated with the recorded data. Online data will will be stored in a password protected computer account.

A feedback email message will be sent to participants who provide contact details after the data has been analysed.

Please note that it is the authentication system, not you, that is being evaluated. You may withdraw from the experiment at any time without prejudice by simply closing your browser window and any data recorded will be discarded.

If you have any further questions regarding this experiment, please contact:

Tony McBryan
Computing Science Department
Lilybank Gardens
mcbryana@dcs.gla.ac.uk

□ I have read this information sheet and agree to voluntarily take part in this experiment.
Name (optional) :
Email (optional) :
Submit

This study adheres to the BPS ethical guidelines, and has been approved by the DCS ethics committe of The University of Glasgow. Whist you are free to discuss your particpation in this study with the experimenter, if you wish to speak to someone not involved in the study, you may contact the chair of the DCS Ethics Committee: hcp@dcs.gla.ac.uk.

Figure 5.31: Consent Screen

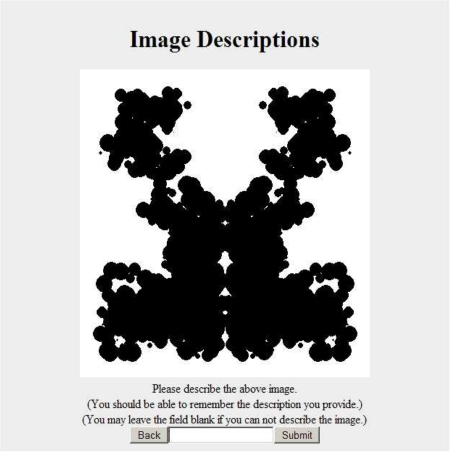


Image Descriptions

Please describe the above image.
(You should be able to remember the description you provide.)
(You may leave the field blank if you can not describe the image.)

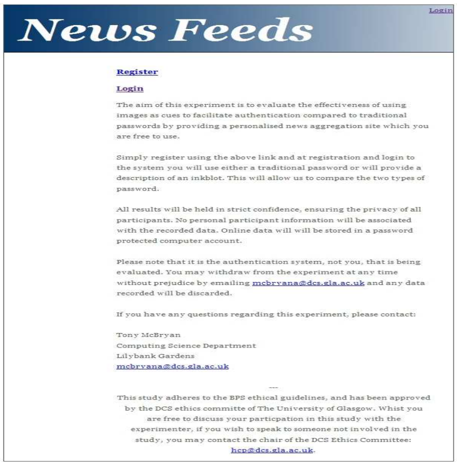
Figure 5.32: Image Description Screen

Users signalled informed consent at the start of the experiment after reading the ethical consent information screen. They were then presented with a series of images for which they would enter descriptions. At the end of the experiment the descriptions were repeated to the user and the user was thanked for their time.

5.2.2 Live System

To create a live system to compare actual use of passwords and image description based authentication systems we created an RSS[Ham03] feed aggregator in PHP/MySQL which is able to read XML based RSS feeds from a variety of news sites in order to allow a central source of news.

The aggregator allows users to choose from presupplied news feeds or to add their own. When users registered with the system they were assigned to the inkblot or password group based on the order they joined. We chose to use inkblots as the image type in this system as they performed consistently well in our previous experiments, as detailed in Section 6, and were simple to generate. Password users would register a password as normal while inkblot users would use the inkblot designer to create an inkblot which they would then describe to form an image description.



The aim of this experiment is to evaluate the effectiveness of using images as cues to facilitate authentication compared to traditional passwords by providing a personalised news aggregation site which you are free to use.

Simply register using the above link and at registration and login to the system you will use either a traditional password or will provide a description of an inkblot. This will allow us to compare the two types of password.

All results will be held in strict confidence, ensuring the privacy of all participants. No personal participant information will be associated with the recorded data. Online data will will be stored in a password protected computer account.

Please note that it is the authentication system, not you, that is being evaluated. You may withdraw from the experiment at any time without prejudice by emailing mcbryana@dcs.gla.ac.uk and any data recorded will be discarded.

If you have any questions regarding this experiment, please contact:

Tony McBryan
Computing Science Department
Lilybank Gardens
mcbryana@dcs.gla.ac.uk

This study adheres to the BPS ethical guidelines, and has been approved by the DCS ethics committe of The University of Glasgow. Whist you are free to discuss your particpation in this study with the experimenter, if you wish to speak to someone not involved in the study, you may contact the chair of the DCS Ethics Committee: hcp@dcs.gla.ac.uk.

Figure 5.33: Consent Screen

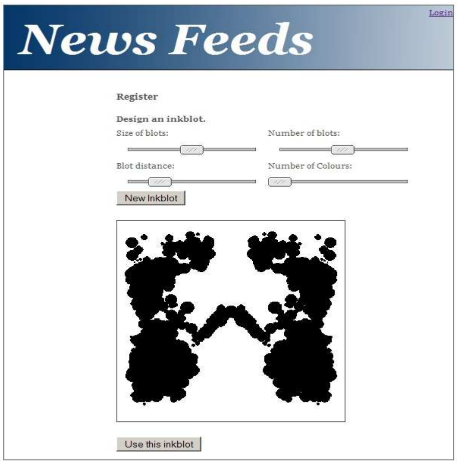
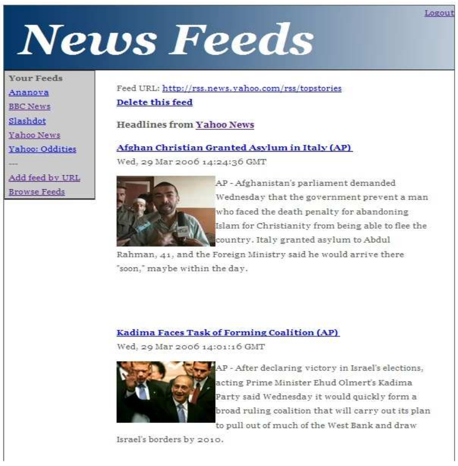


Figure 5.34: Inkblot Designer



Feed URL: http://rss.news.yahoo.com/rss/topstories

Headlines from Yahoo News

Afghan Christian Granted Asylum in Italy (AP)
Wed, 29 Mar 2006 14:24:36 GMT

AP - Afghanistan's parliament demanded Wednesday that the government prevent a man who faced the death penalty for abandoning Islam for Christianity from being able to flee the country. Italy granted asylum to Abdul Rahman, 41, and the Foreign Ministry said he would arrive there "soon," maybe within the day.

Kadima Faces Task of Forming Coalition (AP)
Wed, 29 Mar 2006 14:01:16 GMT

AP - After declaring victory in Israel's elections, acting Prime Minister Ehud Olmert's Kadima Party said Wednesday it would quickly form a broad ruling coalition that will carry out its plan to pull out of much of the West Bank and draw Israel's borders by 2010.

Figure 5.35: A Sample Feed

Chapter 6

Results

6.1 About these results

In this section we detail the results of our experiments for Stages 1, 2 and 3. Where charts are provided the error bars will signify Standard Error and are generated using Microsoft Excel[Mic06].

Our statistical analysis of the results was undertaken by using the SPSS[Inc06] statistics package and custom PHP scripts for transforming the data into a form suitable for SPSS.

Since some of the data is unbalanced by nature, due to the users ability to provide a null response, we have performed EM missing value analysis where appropriate to ensure valid results.

6.2 Phase 1 / Stage 1

6.2.1 Response Rate

In the results for the experiment undertaken during stage 1 of the project we released the framework application designed to gather textual associations from images. This study involved 49 users who each provided 30 associations resulting in a maximum number of 1470 textual associations being gathered from the user base. In reality since we allowed users to enter an empty (0 character) entry if they could not form an association we actually gathered a total of 1355 useful textual associations.

The breakdown of response rate is important as it provides us with a comparative measure in which images users find easiest to create associations for - images which were difficult to provide an association for will have lower response rates.

Image Class	Image	Responses
Faces	1	46
	2	46
	3	46
	4	49
	5	45
	6	46
Total		278

Image Class	Image	Responses
Fractals	7	47
	8	46
	9	43
	10	46
	11	47
	12	43
Total		272

Image Class	Image	Responses
Inkblots	13	47
	14	46
	15	44
	16	43
	17	48
	18	42
Total		270

Image Class	Image	Responses
Snowflakes	19	44
	20	46
	21	42
	22	42
	23	40
	24	43
Total		257

Image Class	Image	Responses
Textures	25	48
	26	46
	27	47
	28	48
	29	46
	30	43
Total		278

The following figures represent the mean response rates we would expect to receive for images. A response rate of 0.9 indicates that 90% of the cases in which we present an image to a user would result in the user providing a textual response (i.e. 0.9 responses received for each image shown in that class).

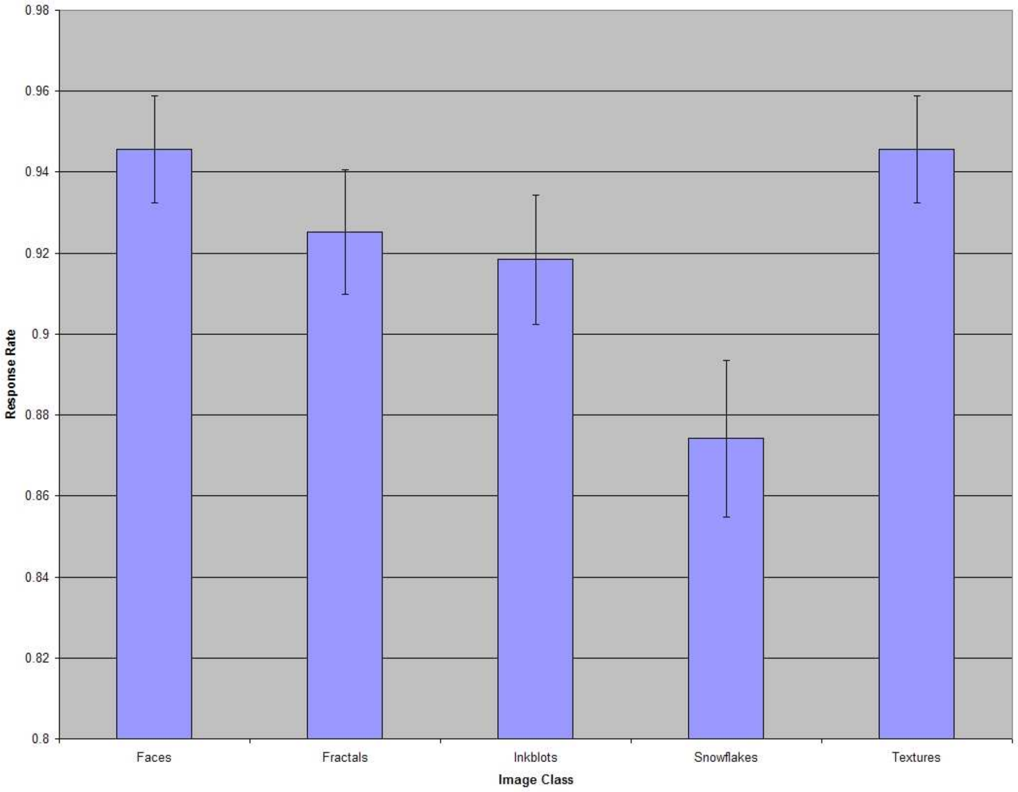


Figure 6.1: Rate of Responses for Image Classes

The results show that the number of responses was significantly affected by the image class shown, F(1.83,535.4)=15.53, $p < 0.05$. Bonferroni post hoc tests of each image class revealed that Snowflakes had significantly lower rate of responses than all other image classes (Faces: $CI_{.95}$ = -.114 (lower) -.029 (upper), Fractals: $CI_{.95}$ = -.087 (lower) -.015 (upper), Inkblots: $CI_{.95}$ = -.078 (lower) -.010 (upper), Textures: $CI_{.95}$ = -.114 (lower) -.029 (upper), $p < 0.05$ for all). We observed that Faces and Textures provided significantly higher rate of responses than Inkblots, $CI_{.95}$ = 0.00 (lower) 0.05 (upper), $p < 0.05$. No other comparisons were significant (all $p > 0.5$).

Analysis of the response rate for individual images within the same image class using Bonferroni post hoc tests do not indicate that images within classes differ signicantly in response rate (for all $p > 0.05$).

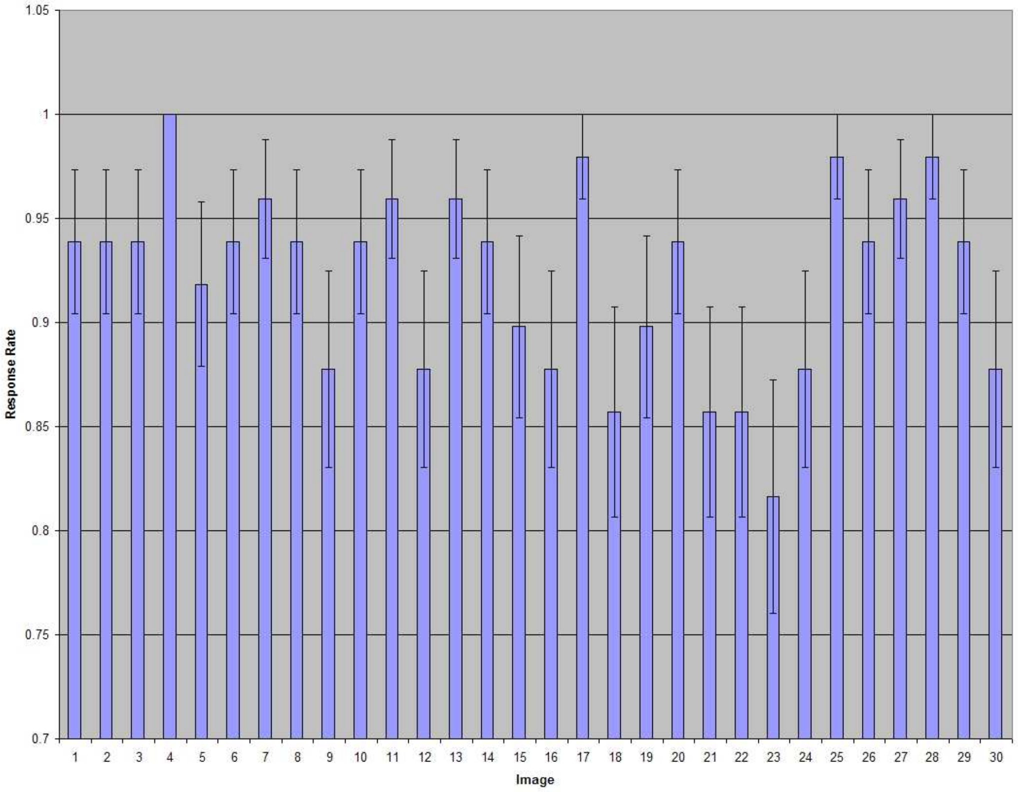


Figure 6.2: Rate of Responses for Individual Images

6.2.2 Length of Response

Since the password length is often the defining measure of security of traditional textual passwords we were interested in seeing the length of the response for each image and determining if any particular images or image classes encouraged longer responses.

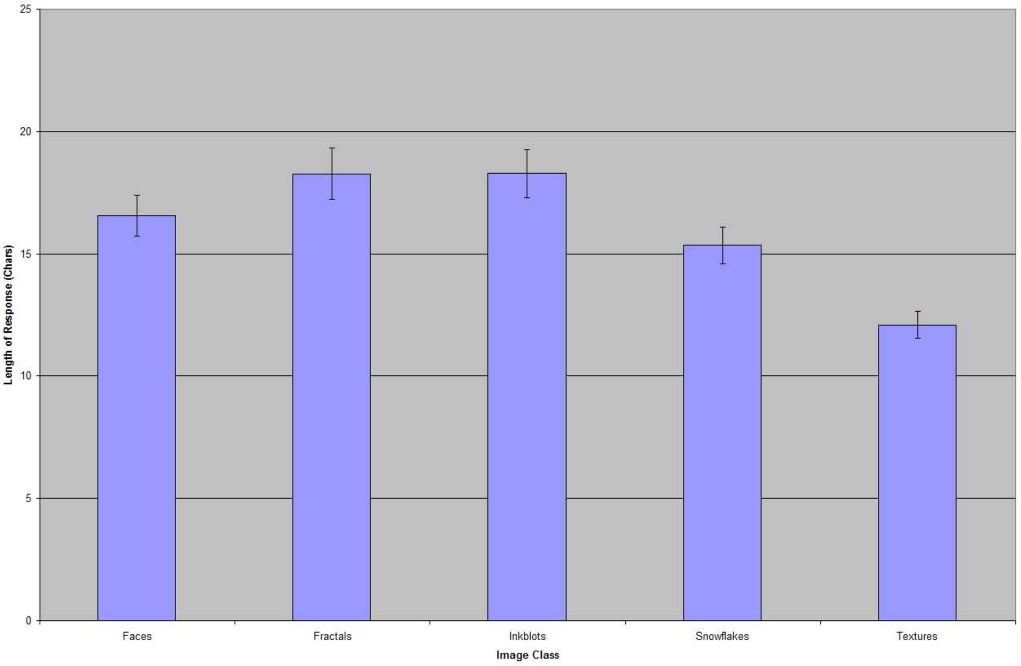


Figure 6.3: Mean Length of Responses for Image Classes

The results show that the length of the responses was significantly affected by the image class, F(3.61,1000.5)=12.414, $p < 0.05$. Bonferroni post hoc tests of each image class showed that Textures had significantly shorter response lengths than all other image classes, Faces: $CI_{.95}$ = -7.06 (lower) -1.87 (upper), Fractals: $CI_{.95}$ = -8.94 (lower) -3.25 (upper), Inkblots: $CI_{.95}$ = -8.787 (lower) -3.43 (upper), Snowflakes: $CI_{.95}$ = -5.2 (lower) -.963 (upper), $p < 0.05$ for all. Inkblots had significantly longer responses than Snowflakes, $CI_{.95}$ = .14 (lower) 5.92 (upper), $p < 0.05$. Fractals also had longer responses than Snowflakes but it was insufficient to be statistically significant and merely poses as an interesting detail, $CI_{.95}$ = -.16 (lower) 6.19 (upper), $p < 0.08$.

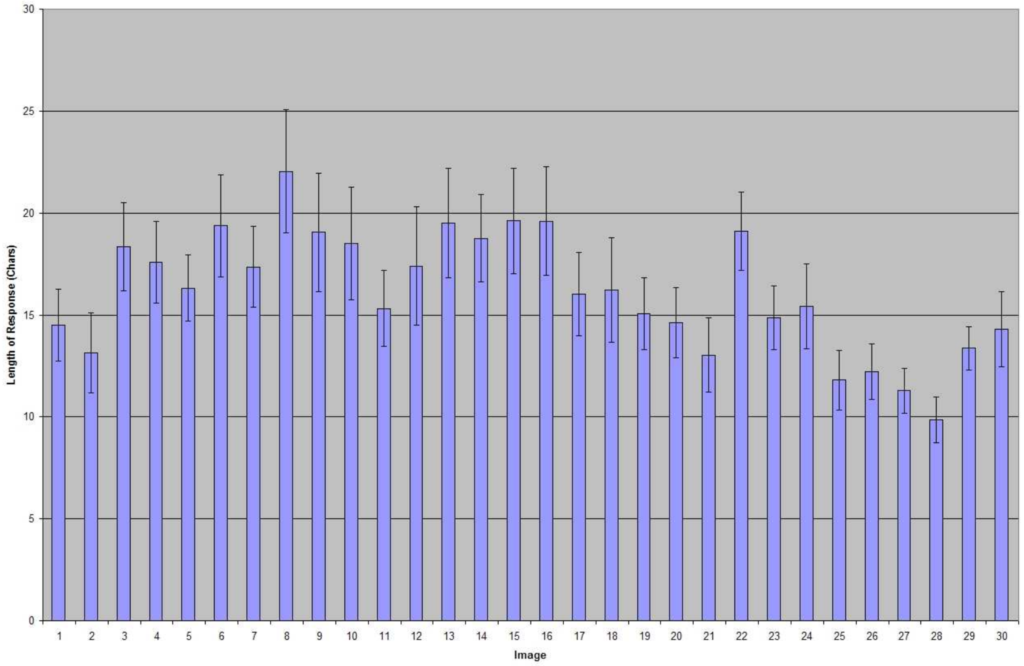


Figure 6.4: Mean Length of Responses for Individual Images

Analysis of the response length indicated a significant difference of images within image classes between images 21 and 22, F=(4.16,99.96)=2.85, $CI_{.95}$ = -11.72 (lower) -1.56 (higher), $p < 0.05$. Otherwise there were no significant differences ($p > 0.05$ in all cases).

6.2.3 Character Distribution

For illustrative purposes we provide a character frequency analysis. We do not perform statistical analysis as character frequency count is dependant on the length of the responses which we have covered in the previous section and we deal with predictability of responses in Section 6.2.4.

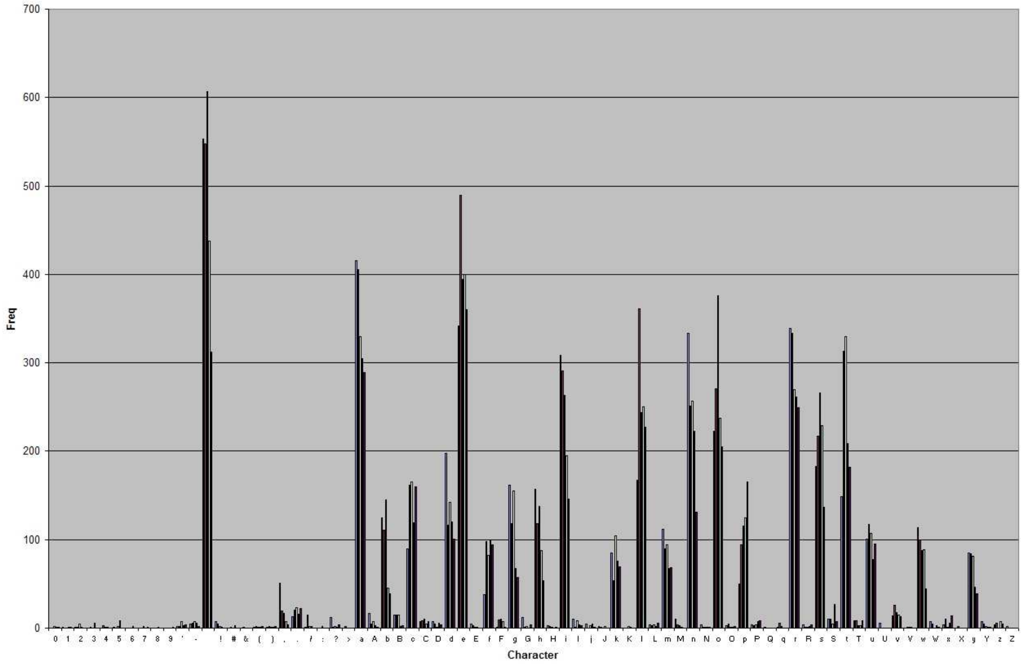


Figure 6.5: Character Frequency Analysis by Image Class

We note informally that the character frequency analysis matches what we would expect from English text.

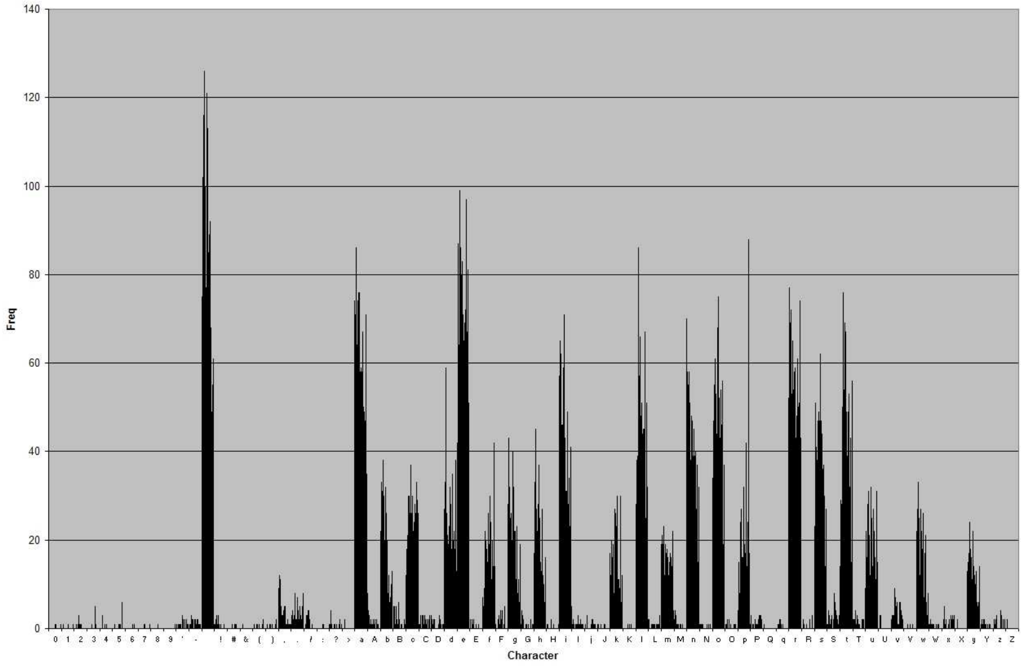


Figure 6.6: Character Frequency Analysis by Image

6.2.4 Entropy

The entropy of a piece of information defines how much uncertainty or "randomness" there is in a signal[Sha48]. This is a useful measure for us as when we increase the entropy in a password it makes the password less predictable and therefore harder to attack in an automated fashion. We will represent the strength of a password in two measures, first the average number of bits per character and the total number of bits for the entire string. This represents the minimum number of bits it would take to encode this string if you knew the character set and the frequency of each character and can be compared to how much you can compress the string assuming a perfect algorithm.

For comparison, on a standard ASCII keyboard there are 95 printable characters (including the space character), this is a maximum entropy of lg(95) = 6.57 bits per character and requires that every character is used in equal distribution to obtain this value. A single character string, or a string made entirely of the same character, would have a maximum entropy of lg(1) = 0 bits. This provides us with an upper (6.57 bits per character) and lower (0 bits per character) bound on entropy for textual associations. Note that the entropy figures essentially measure the use of the character set and as such the more varied the characters the higher the entropy.

Initially we look at individual responses and calculate the number of bits required to represent the response and the average number of bits per character. The bits per character value represents the situation in which an attacker knows both the character set and character frequency of the response while preparing an attack. We then present the means of these values for each image class and individual image.

The only significant effect observed after Bonferroni post hoc analysis was that the Textures class has a significantly lower value of the number of bits per character than all other image classes, F(4,1108)=5.49, $p < 0.05$, Faces: $CI_{.95}$ = -.351 (lower), -.05 (upper), Fractals: $CI_{.95}$ = -.345 (lower) -0.49 (upper), Inkblots: $CI_{.95}$ = -.363 (lower) -.067, Snowflakes: $CI_{.95}$ = -.317 (lower) -.023 (upper). No other significance was observed ($p > 0.05$).

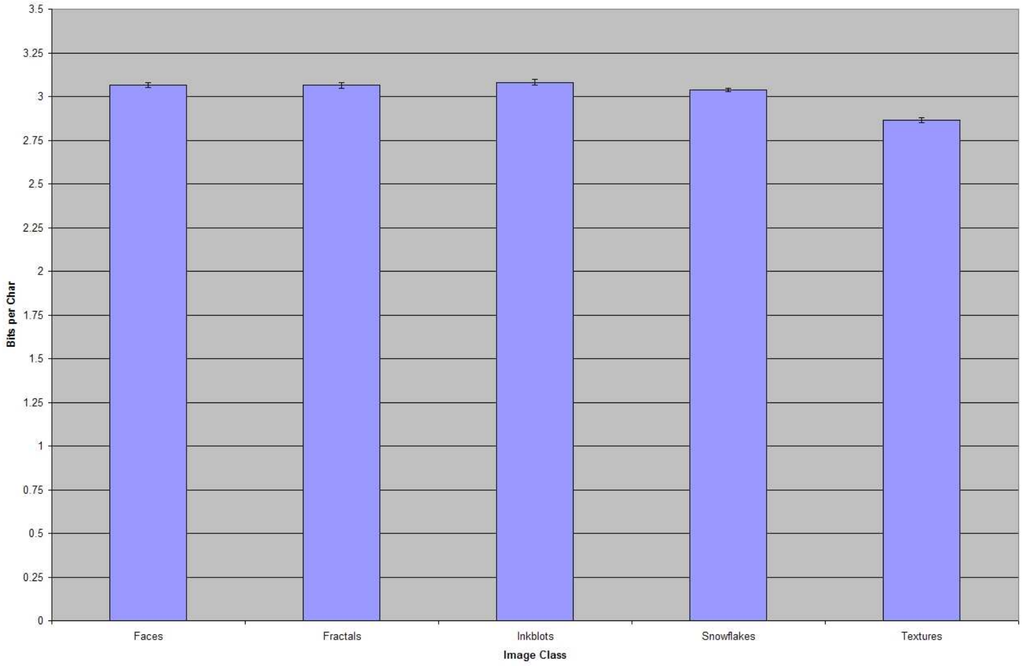


Figure 6.7: Mean Number of Bits per Character for Image Classes

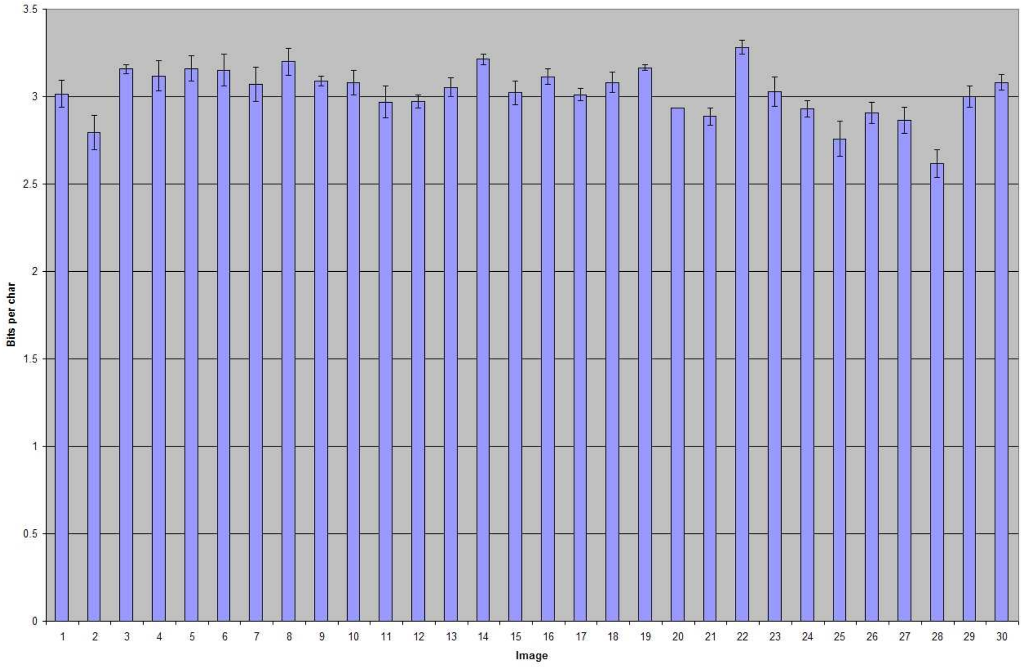


Figure 6.8: Mean Number of Bits per Character for Individual Images

We then performed Bonferroni post hoc analysis on images within the image classes to ascertain if there was significant effects between individual images in the same image class.

In the Snowflakes class we found that image 22 had significantly higher mean bits per character than three other images in its class, F(4.11,197.285)=4.083, $p < 0.05$, Image 20: $CI_{.95}$ = .01 (lower) .693 (upper), Image 21: $CI_{.95}$ = .174 (lower) .738 (upper), Image 24: $CI_{.95}$ = .117 (lower) .608 (upper). We believe this increase in entropy is caused at least partially by the longer response length of this image as it would increase the character set used in the responses.

We observed that within the Textures class there were a number of significant effects between individual images, F=(3.753,180.12)=5.508, $p < 0.05$. This was mostly caused by image 28 which was significantly lower than images 29 and 30, Image 29: $CI_{.95}$ = -.727 (lower) -.042 (upper), Image 30: $CI_{.95}$ = -.773 (lower) -.169 (upper), $p < 0.05$. Image 25 had a significantly lower number of bits per character than image 30, $CI_{.95}$ = -.623 (lower) -.002 (upper), $p < 0.05$.

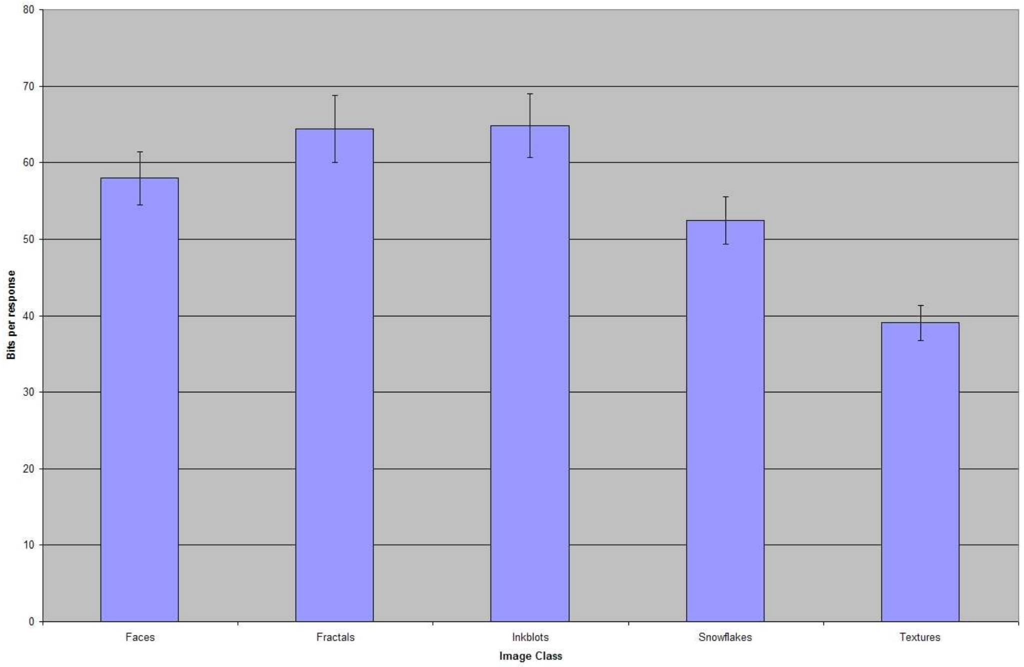


Figure 6.9: Mean Number of Bits per Response for Image Classes

The bits per response measure is the product of the length of the response and the number of bits per character required to encode it and represents the number of bits necessary to represent the response on its own assuming you know the exact character set and frequency distribution in advance. The results showed that the number of bits per response was significantly affected by the image class, F(3.62,1003.2) = 9.954, $p < 0.05$. Bonferroni analysis indicated that Textures had significantly lower bits per response average than all other classes, Faces: $CI_{.95}$ = -30.85 (lower) -6.97 (upper), Fractals: $CI_{.95}$ = -38.86 (lower) -11.75 (upper), Inkblots: $CI_{.95}$ = -37.64 (lower) -13.7 (upper), Snowflakes: $CI_{.95}$ = -23.59 (lower) -2.76 (upper). The Snowflakes class was almost significantly different from the Inkblots class but p was insufficiently small for it to be statistically significant ($p \sim 0.1$).

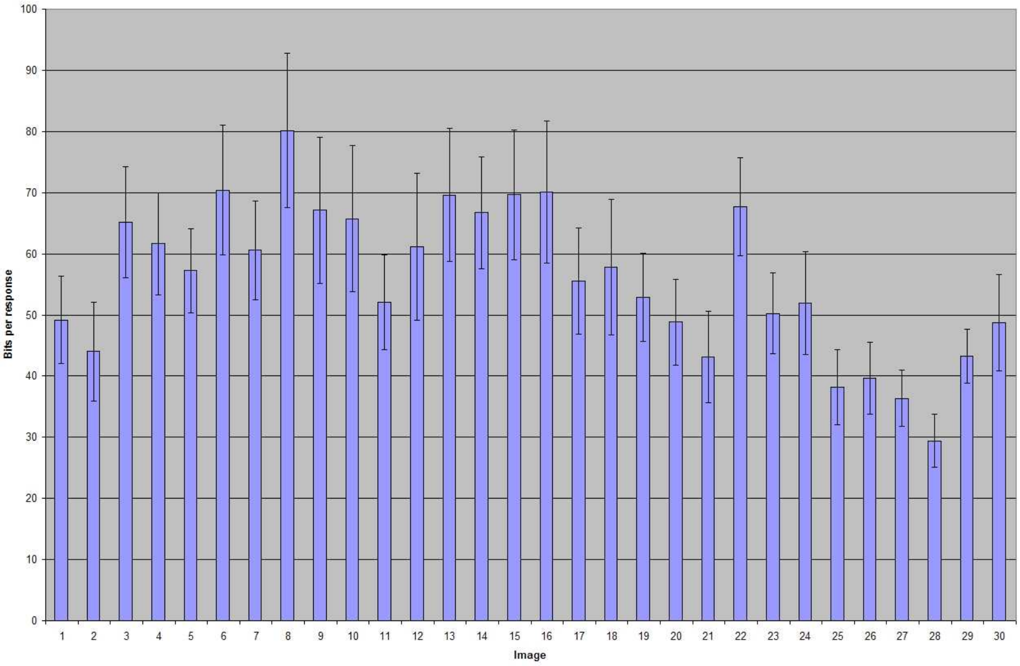


Figure 6.10: Mean Number of Bits per Response for Individual Images

The only significant difference in bits per pass for individual images compared to images within the same class was in the snowflakes class where images 21 and 22 differed significantly, F(4.15,199.3)=2.879, $p < 0.05$, $CI_{.95}$ = -47.02 (lower) -6.401 (upper). There were no other signficant differences ($p > 0.05$ in all other cases).

The previous results showed the optimal encoding assuming known character set and frequencies for a particular response, however that situation is unlikely. Our second approach to entropy measures the entropy of the entire image class or individual image by concatenating all the responses for that image or class together and calculating the bits per character needed to represent all the responses. This represents an attacker that knows only the character frequencies of the general population for a specific image class or image rather than the distribution for a particular response.

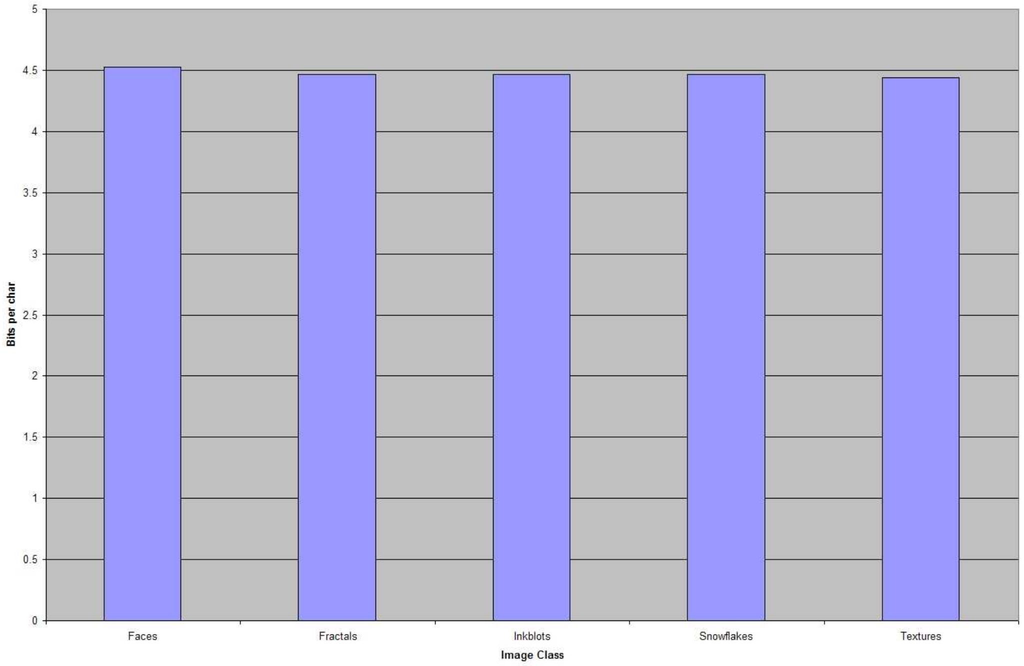


Figure 6.11: Number of Bits per Character for General Population by Image Class

We then envisioned a scenario in which an attacker had a number of associations from a user and wished to break into a site for which he did not have the association, to do this we concatenated together all the users responses and calculated the Bits per char for this, the results indicated that 4.33 (SE: 0.004) bits per char would be required if you knew the distribution of characters favoured by a particular user.

All the responses were then concatenated to represent the general case of an attacker who does not know the image class which resulted in a mean 4.52 bits required to represent each character, this is the worst case for an attacker as they are likely to be able to gather this distribution data without specific information about the user or image.

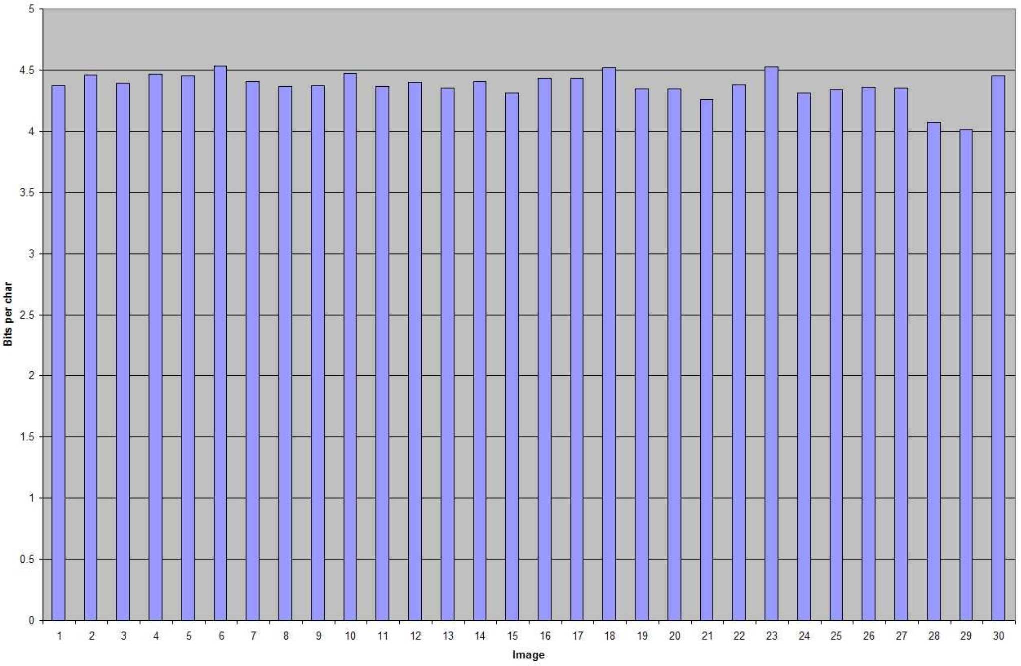


Figure 6.12: Number of Bits per Character for General Population by Individual Image

6.2.5 Levenshtein Distance

Levenshtein Distance[Lev65] is a measure of the least number of transformations that need to be applied to a string to convert it into another string. We can use this to measure the *difference* between two strings. This allows us to test responses provided by users to determine how different they are to each other. In the results presented a high Levenshtein distance is better. Levenshtein Distance is highly correlated with Length thus we use the ratio of :

$$d = \frac{AverageLevenshteinDistance}{AverageLength}$$

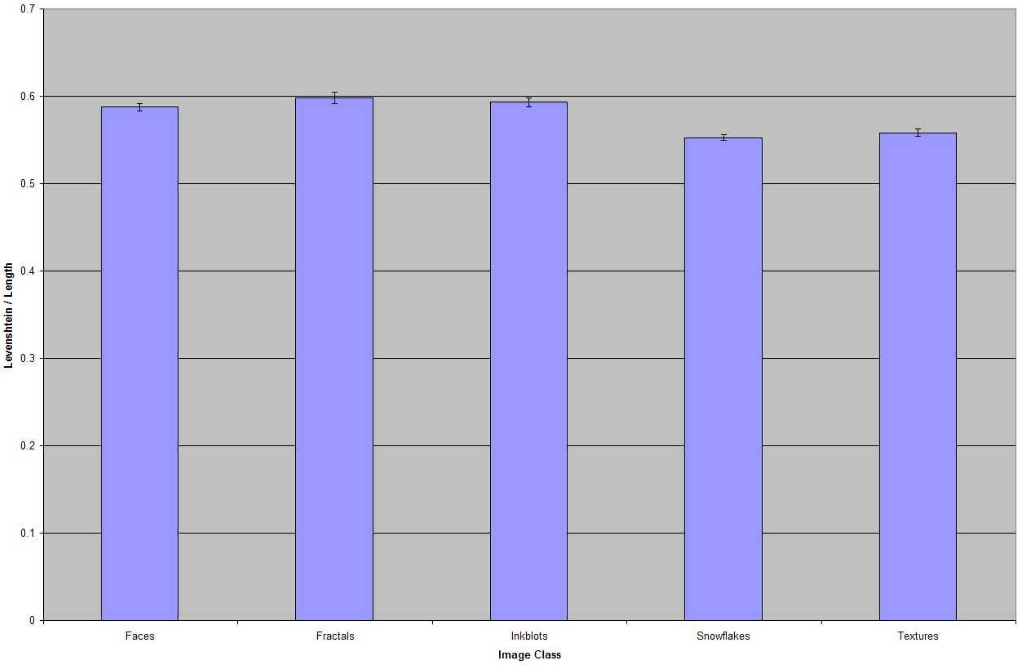


Figure 6.13: Average Levenshtein Distance / Average Length for Image classes

From these results we observe that the image class significantly affects the ratio of Levenshtein to Length, F(2.35,651.5)=52.246, $p < 0.05$. Snowflakes have a significantly lower ratio than all other image classes and Textures have a significantly lower ratio than all other image classes excluding Snowflakes. No other significant effects were observed.

Significant differences for Image Classes : (all $p < 0.05$)

	$CI_{.95}$	
Image Class Pair	(Lower)	(Upper)
Face – Snowflake	.037	.054
Face – Texture	.016	.042
Fractal – Snowflake	.034	.068
Fractal – Texture	.019	.051
Inkblot – Snowflake	.034	.057
Inkblot – Texture	.017	.042
Snowflake – Texture	-.032	.000

We then performed an analysis of images within their image class to determine if the Levenshtein ratio is affected by the individual images within the class. The results of a Bonferroni analysis showed some images with significant differences within their classes which are summarised here.

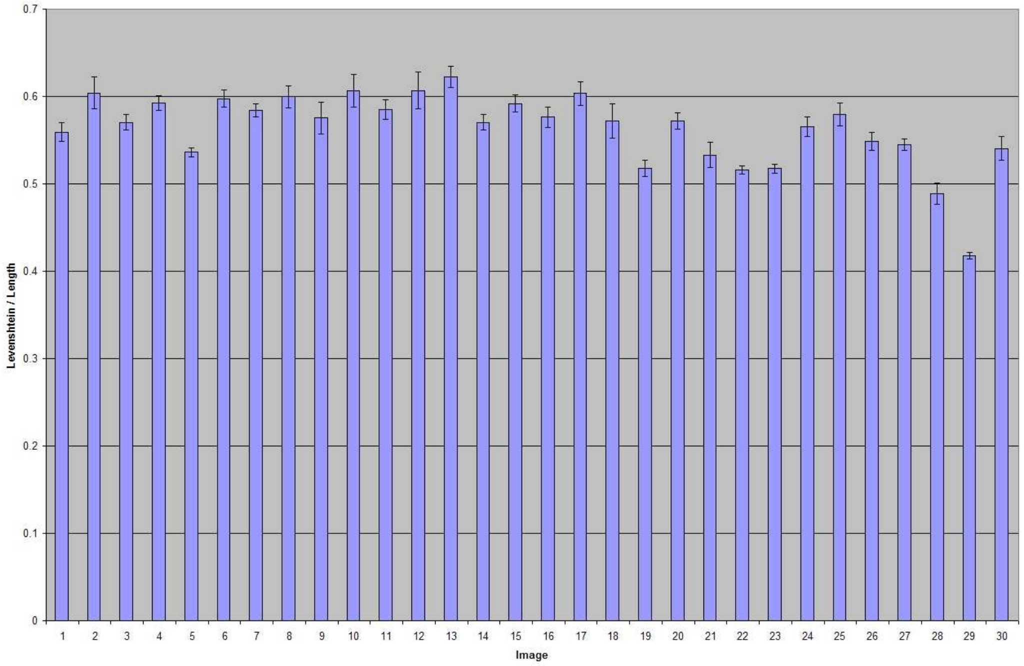


Figure 6.14: Average Levenshtein Distance / Average Length for Individual Images

In the Faces image class the variations were caused by division of sex along images 1,3 and 5 (Female faces) and images 2,4,6 (Male faces). The images of female faces had significantly lower ratios than the images of male faces, F(1.519,75.93)=9.084, $p < 0.05$.

Significant differences for Faces : (all $p < 0.05$)

	$CI_{.95}$	
Image Pair	(Lower)	(Upper)
1–2	-.085	-.001
1–4	-.063	-.025
1–6	-.066	-.009
3–4	-.051	-.013
3–5	0.002	0.071
4–5	.038	.099
5–6	-.098	-.026

None of the Fractals differed significantly within the image class.

In the Inkblot class we observed that Black inkblots (13,15,17) have higher ratios than coloured inkblots (14,16,18), F(1.585,79.268)=5.342, $p < 0.05$.

Significant differences for Inkblots : (all $p < 0.05$)

	$CI_{.95}$	
Image Pair	(Lower)	(Upper)
13–14	.023	.086
13–15	.013	.067
13–16	.029	.091
14–15	-.027	-.002
16–17	-.090	-.002
17–18	.024	.076

Our results from the Snowflake class indicate that the two highest scoring snowflakes (20 and 24), which were both generated using the maximum complexity options, had significantly higher ratios than the other snowflakes, F(2.354,117.72)=6.151, $p < 0.05$.

Significant differences for Snowflakes : (all $p < 0.05$)

	$CI_{.95}$	
Image Pair	(Lower)	(Upper)
19–20	-.093	-.035
19–24	-.067	-.023
20–21	.009	.096
20–22	.031	.100
20–23	.013	.114
22–24	-.089	-.005

In the textures class image 25 had a significantly higher Levenshtein ratio than images 28,29 and 30. Images 28 and 29 also showed a significantly lower performance, we attribute this to the fact that textures 28 and 29 were both easily identifiable textures and as such prompted very similar associations while the other textures were harder to identify, F(2.077,103.84)=22.936, $p < 0.05$.

Significant differences for Textures : (all $p < 0.05$)

	$CI_{.95}$	
Image Pair	(Lower)	(Upper)
25–28	.037	.145
25–29	.093	.238
25–30	.032	.073
26–28	.018	.090
26–29	.079	.177
27–29	.095	.160
28–29	.013	.136
29–30	-.179	-.046

We further observed that the minimum Levenshtein Distance between pairs of strings was either 0 or 1 for all images - meaning that the exact, or almost exact, same textual association had been made. This is equivalent in the traditional password model to users using the same password.

6.2.6 Smith Waterman

A Smith-Waterman[SW81] score can be used as a measure of the ability to perform a local alignment on two strings for finding similar areas of the string. We can use this to measure the *similarity* of the strings by way of common phrases or sequences of characters shared by the strings. Unlike the Levenshtein Distance we want the Smith-Waterman score to be as close to zero as possible, a zero score indicates the strings do not share any common sequences. Like Levenshtein Distance the Smith-Waterman score is correlated with the length of the strings it measures, hence we use the ratio of :

$$s = \frac{AverageSmithWatermanScore}{AverageLength}$$

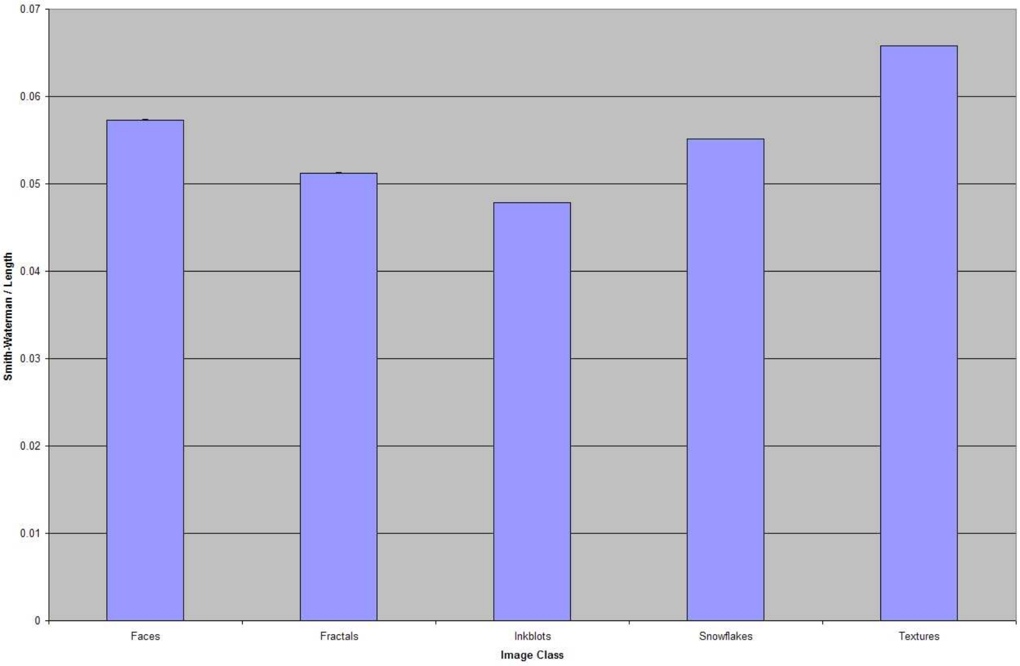


Figure 6.15: Average Smith-Waterman Score / Average Length for Image Classes

The results of the images class analysis indicate that the Smith-Waterman ratio is significantly affected by the type of image shown, F(1.79,496)=1487, $p < 0.05$. (Note that the Standard Error bars on Figure 6.15 are too small to be visible). Bonferroni post hoc analysis showed that Inkblots had a lower Smith-Waterman ratio than all other image classes, followed by Fractals, Snowflakes, Faces and then Textures.

Significant differences for Image Classes : (all $p < 0.05$)

	$CI_{.95}$	
Image Class Pair	(Lower)	(Upper)
Faces–Fractals	.006	.007
Faces–Inkblots	.009	.011
Faces–Snowflakes	.003	.005
Faces–Textures	-.009	.008
Fractals–Inkblots	.003	.004
Fractals–Snowflakes	-.003	.004
Fractals–Textures	-.016	-.015
Inkblots–Snowflakes	-.007	-.006
Inkblots–Textures	-.019	-.018
Snowflakes–Textures	-.013	-.012

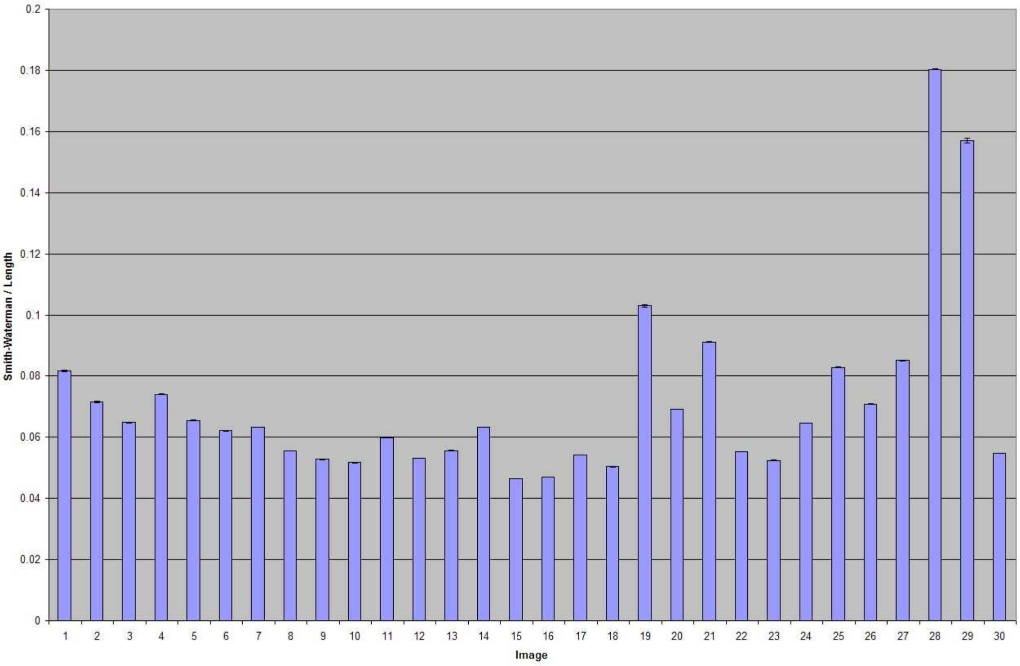


Figure 6.16: Average Smith-Waterman Score / Average Length for Individual Images

Our analysis of the Faces class indicates that faces with distinctive features or items of clothing (example visible earrings in image 1 and beard in image 4) tend to score higher Smith-Waterman ratios as these features are more readily commented on within the response, F(1.905,91.45)=36.567, $p < 0.05$.

Significant differences for Faces : (all $p < 0.05$)

	$CI_{.95}$	
Image	(Lower)	(Upper)
1–2	.007	.015
1–3	.012	.021
1–5	.008	.022
1–6	.014	.023
2–4	-.011	-.004
2–6	.002	.013
3–4	-.016	-.008
4–5	.003	.019
4–6	.010	.019
5–6	-.001	.009

The two fractals in images 7 and 11 demonstrated significantly higher Smith-Waterman ratios than the other fractals, we believe this may be caused by the high level of symmetry however would not explain why image 12 does not score a similarly high Smith-Waterman ratio, F(1.725,85.62)=23.66, $p < 0.05$.

Significant differences for Fractals : (all $p < 0.05$)

	$CI_{.95}$	
Image Pair	(Lower)	(Upper)
7–8	.006	.011
7–9	.005	.017
7–10	.006	.016
7–12	.004	.016
8–11	-.010	-.002
9–11	-.012	-.005
11–12	.004	.011

We found that there was a significant decrease in Smith-Waterman ratios for images 15,16 and 18 compared to images 13, 14 and 17 within the Inkblots class, F(1.74,83.52)=42.196, $p < 0.05$. It is possible this is related to the high blot density of these images, whereas images 13, 14 and 17 were more spread out on the inkblot canvas.

Significant differences for Inkblots : (all $p < 0.05$)

	$CI_{.95}$	
Image Pair	(Lower)	(Upper)
13–14	-.009	-.003
13–15	.005	.014
13–16	.004	.013
13–18	.000	.010
14–15	.010	.021
14–16	.009	.021
14–17	.003	.010
14–18	.005	.018
15–17	-.011	-.006
15–18	-.007	-.001
16–17	-.011	-.005
16–18	-.005	-.002
17–18	.001	.008

We found that there was a significant difference in the Smith-Waterman ratio for the different snowflake images, F(1.53,73.565)=66.757, $p < 0.05$. The highest Smith-Waterman ratios were images 19 and 21 which were the least complex of the snowflakes and had significantly higher ratios than all other snowflakes. They were followed by the two most complex images highlighted in the Levenshtein Distance analysis. The lowest Smith-Waterman ratios were for images 22 and 23 which were generated using the maximum number of rays (image 22) or the maximum complexity (image 23) but not both.

Significant differences for Snowflakes : (all $p < 0.05$)

	$CI_{.95}$	
Image Pair	(Lower)	(Upper)
19–20	.017	.047
19–21	.002	.024
19–22	.028	.062
19–23	.032	.065
19–24	.022	.052
20–21	-.027	-.011
20–22	.007	.019
20–23	.009	.023
20–24	.001	.009
21–22	.024	.041
21–23	.027	.044
21–24	.018	.031
22–24	-.012	-.004
23–24	-.016	-.006

The images within the texture class were also found to display significant differences by image, F(1.268,60.878)=193.984, $p < 0.05$. The main comment to make on the analysis of the texture images is the significantly higher Smith-Waterman ratios of images 28 and 29 which were previously highlighted as having poor performance in the Levenshtein Distance measures. Image 30 had the lowest Smith-Waterman score in this class and was the hardest to identify what the texture actually was.

Significant differences for Textures : (all $p < 0.05$)

	$CI_{.95}$	
Image Pair	(Lower)	(Upper)
25–26	.007	.020
25–28	-.107	-.088
25–29	-.088	-.045
25–30	.020	.037
26–27	-.020	-.010
26–28	-.121	-.101
26–29	-.106	-.053
26–30	.010	.021
27–28	-.106	-.086
27–29	-.088	-.041
27–30	.024	.037
28–29	.011	.051
28–30	.112	.140
29–30	7.066	.124

This concludes the results of the first stage of our experiments. In the next section we will present the results obtained in the second stage where users were asked to recall the textual associations they had entered previously.

6.3 Phase 1 / Stage 2

6.3.1 Answer Rate

The study for stage two involved 15 users who had also undertaken the previous stage of the study who each provided 30 associations at each stage resulting in a maximum of 900 textual associations (450 for each stage), again because textual associations were optional the actual number of associations gathered from these users was only 410 in the first stage and 395 in the second.

The following tables list the number of responses gathered for each image and the totals for the image classes.

Image Class	Image	Both	First Only	Second Only
Faces	1	13	0	0
	2	12	0	1
	3	13	0	0
	4	15	0	0
	5	13	1	0
	6	13	0	0
Total		**79**	**1**	**1**

Image Class	Image	Both	First Only	Second Only
Fractals	7	13	1	1
	8	14	0	0
	9	12	0	0
	10	13	0	0
	11	13	1	0
	12	13	0	0
Total		**78**	**2**	**1**

Image Class	Image	Both	First Only	Second Only
Inkblots	13	14	0	1
	14	15	0	0
	15	14	1	0
	16	13	1	0
	17	13	1	0
	18	13	1	0
Total		**82**	**4**	**1**

Image Class	Image	Both	First Only	Second Only
Snowflakes	19	13	1	0
	20	12	2	0
	21	13	0	0
	22	14	0	0
	23	12	1	0
	24	13	1	0
Total		**77**	**5**	**0**

Image Class	Image	Both	First Only	Second Only
Textures	25	13	1	0
	26	13	1	0
	27	13	1	0
	28	13	1	0
	29	13	1	0
	30	11	1	0
Total		**76**	**6**	**0**

We found that there was no significant effects between the image classes or individual images as regards response rates in this trial (all $p > 0.05$. We attribute the fact that there was a significant difference in stage 1 to the greater sample size in that experiment.

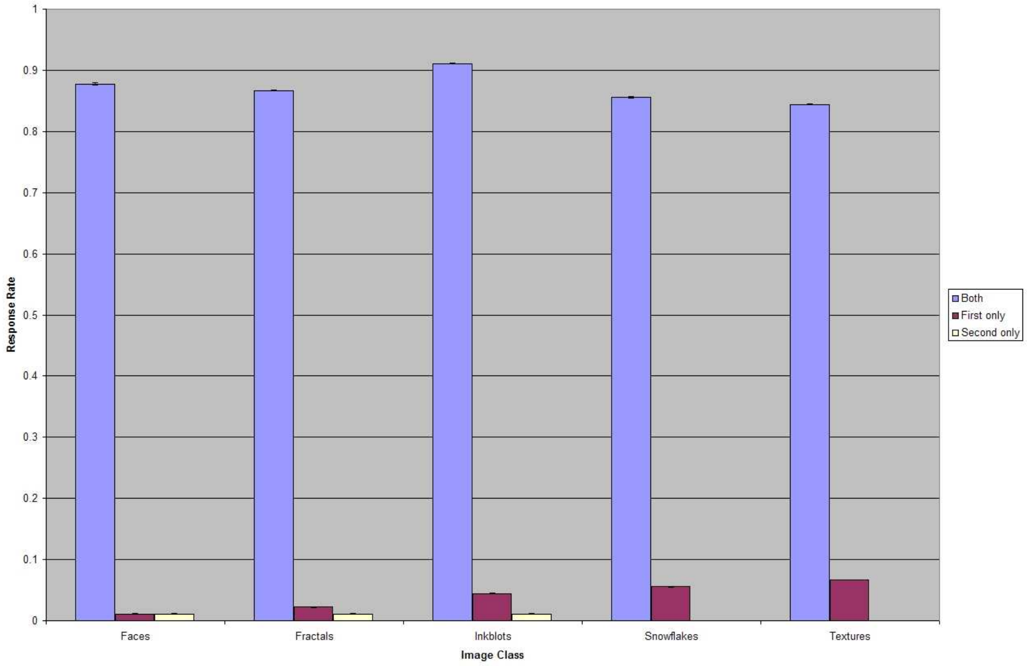


Figure 6.17: Response Rate for Image Classes

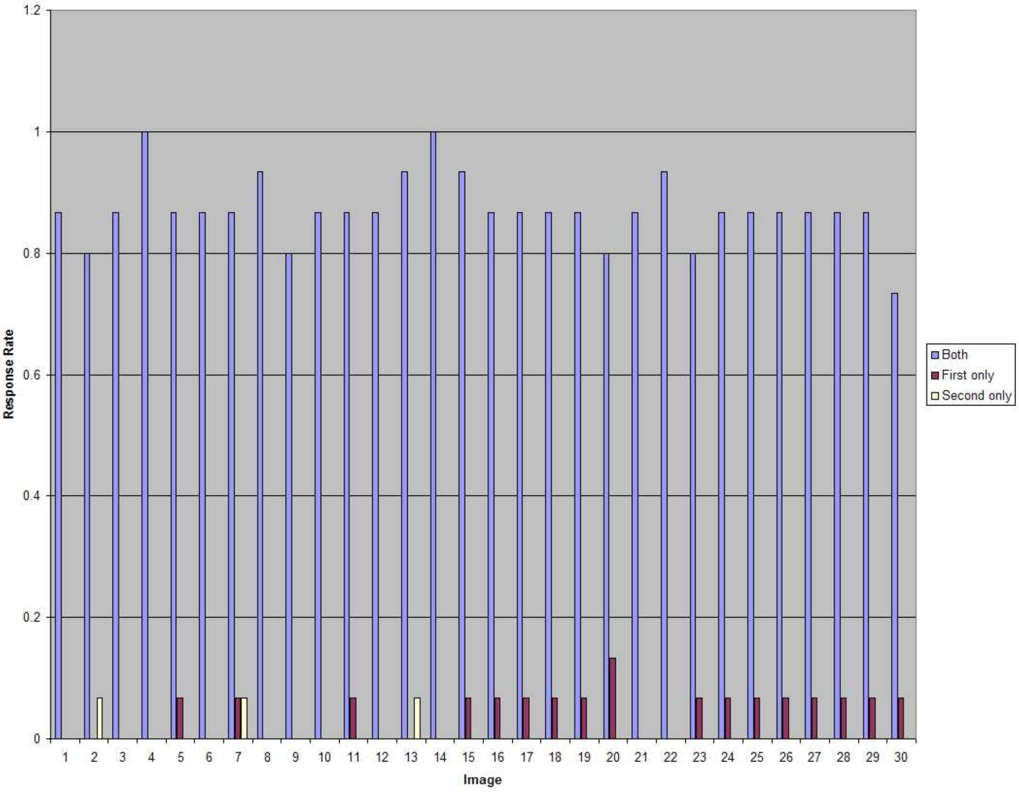


Figure 6.18: Response Rate for Individual Images

6.3.2 Memory vs Time

In this section we will investigate the performance of the users recall rate over time by using the Levenshtein ratio to measure the difference between the users two responses plotted for each image class over time. We present this data in the form of scattercharts with smoothed lines for better readability; each data point on the scatter chart represents one user's responses.

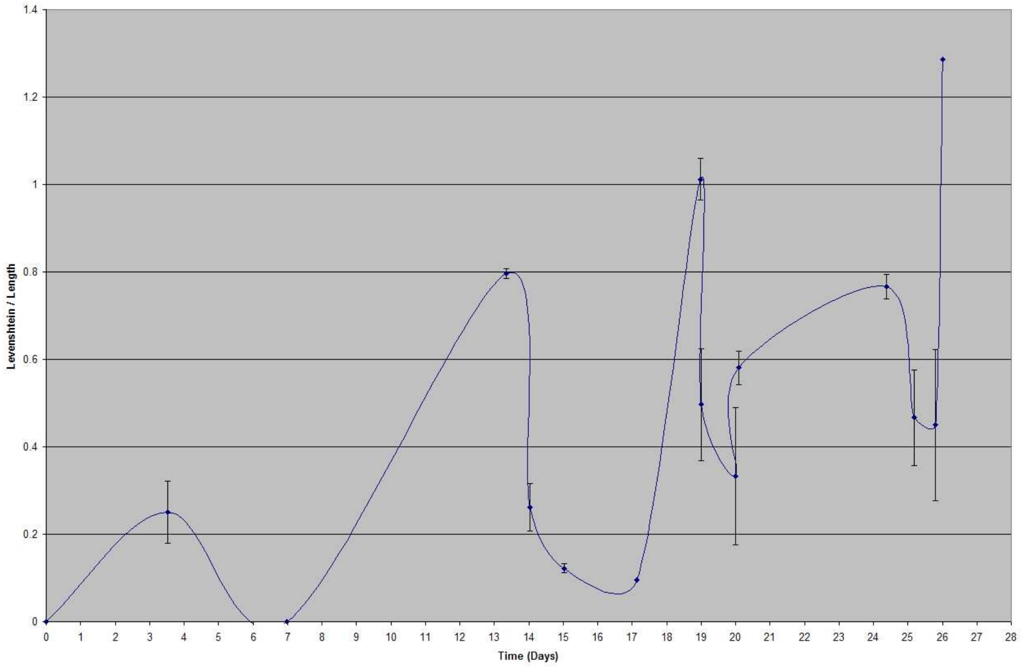


Figure 6.19: Levenshtein Ratio over Time for Face Class

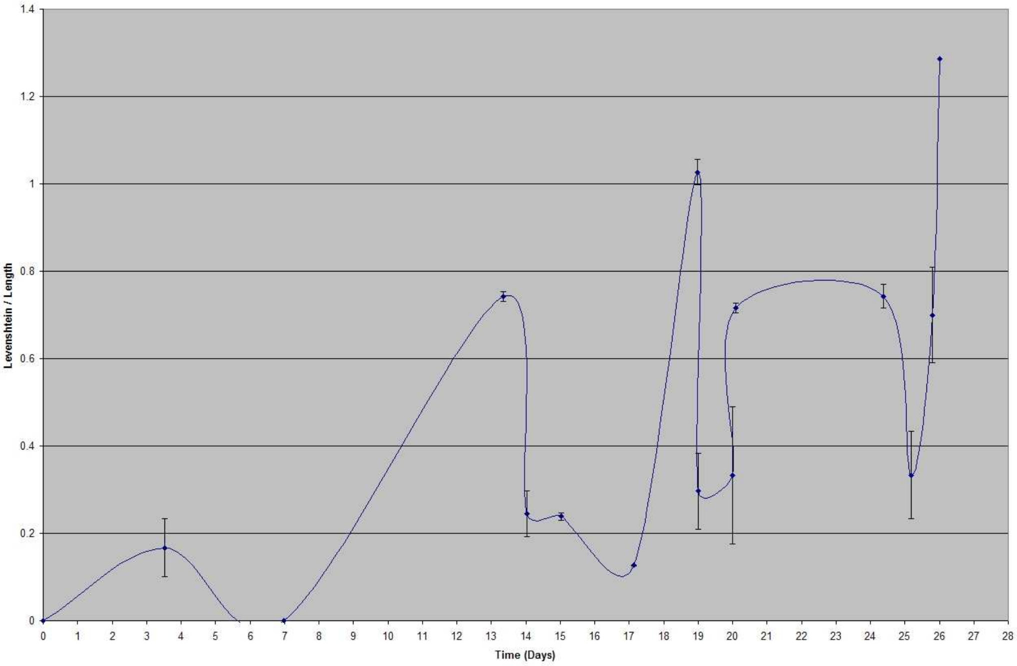


Figure 6.20: Levenshtein Ratio over Time for Fractals Class

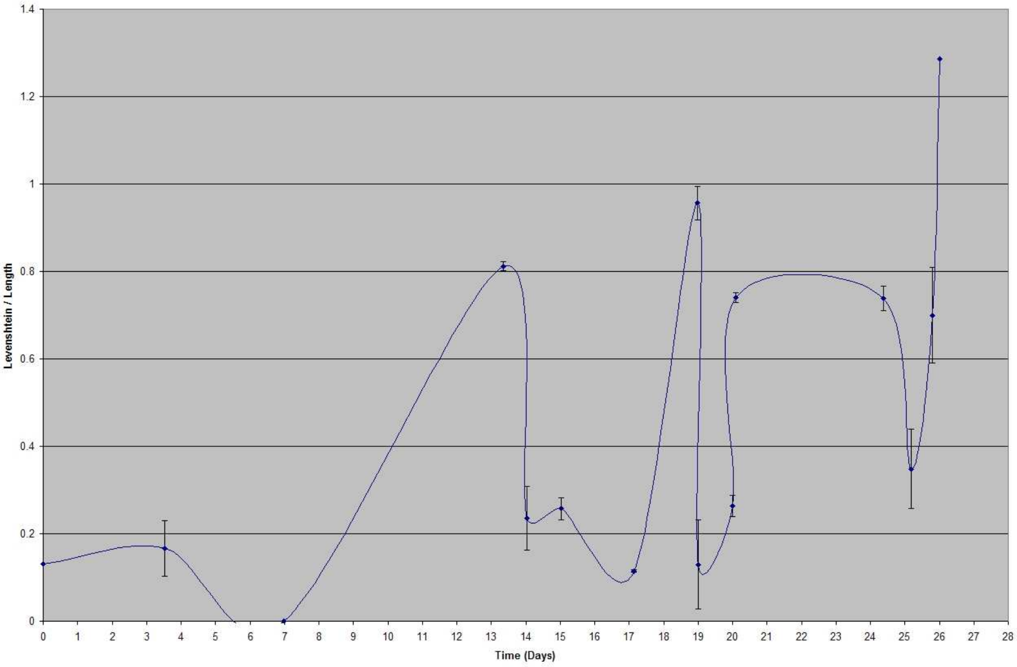


Figure 6.21: Levenshtein Ratio over Time for Inkblots Class

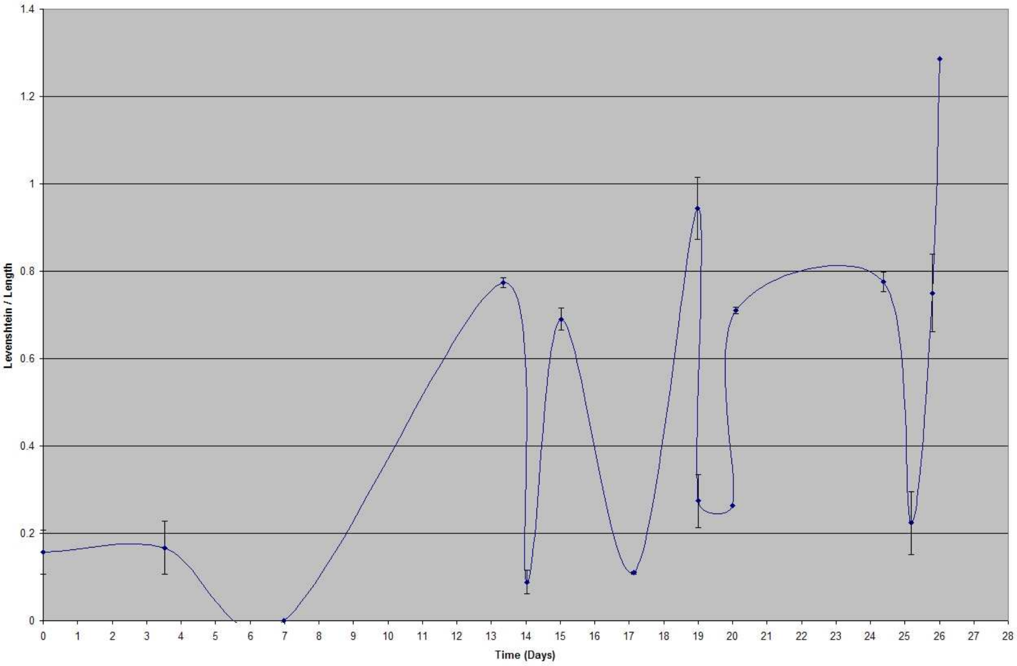


Figure 6.22: Levenshtein Ratio over Time for Snowflakes Class

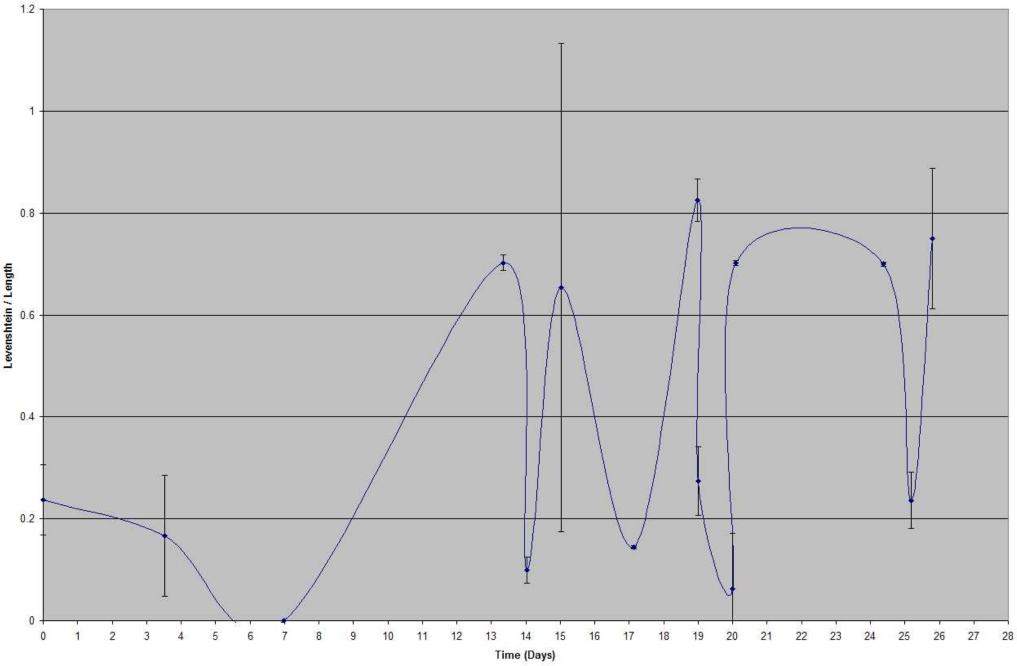


Figure 6.23: Levenshtein Ratio over Time for Textures Class

As demonstrated by the above images there are no statistical significance for Levenshtein ratio over time by the image class. There is also no significance for Levenshtein ratio over time by individual images, all $p > 0.05$. For the sake of conciseness we omit scattercharts of Levenshtein ratio over time for each individual image. It is apparent that time is the major factor for accuracy and that the drop off verses time would appear to be linear.

We provide Figure 6.24 which demonstrates the Levenshtein ratio over time for all images with a linear trendline.

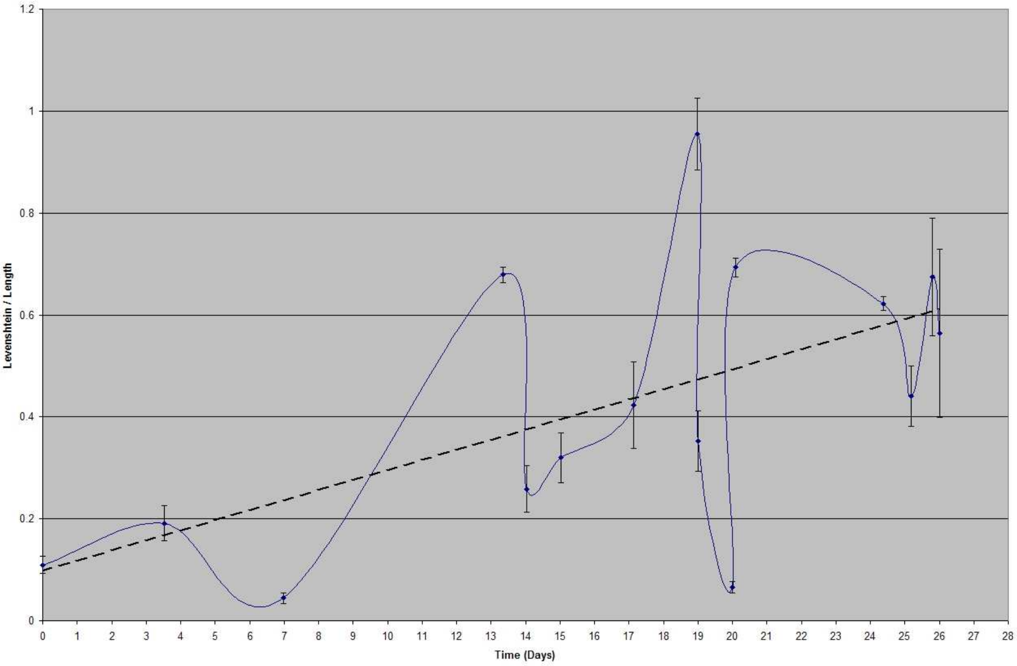


Figure 6.24: Levenshtein Ratio over Time for all Images

Based on the results from Sections 6.2 and 6.3 we decided that it would be best to implement a live system using the inkblot image class as it performed consistently well compared to the other image classes for the security of the image description. Since there was no effect on memory for image classes we only considered their relative security when choosing inkblots to be used. We discuss the results we obtained from this live system in the next section.

6.4 Phase 2 / Stage 3

6.4.1 Answer Rate

In the third stage of the study we created a live system, protected it with a joint inkblot and password authentication system and invited users to register. Users were alternately allocated between the inkblot and password group based on order of registration. In total we had 17 users of each type who each registered an inkblot description or password. In order to ensure that inkblot users would be able to form a textual association the inkblot users were asked to design their own inkblots.

6.4.2 Comparison of Password Security to Inkblot Security

To compare the security of the Inkblot login method we chose to perform the bits per character, length, bits per response and Levenshtein ratio tests on the two authentication methods to determine which had the most effective security. For the purposes of these charts the inkblot descriptions and passwords are both referred to as responses.

We found that the number of bits per character for inkblots (M = 2.91, SE: 0.15) was significantly higher than passwords (M = 2.08, SE: 0.19), t(32)=3.38, $p < 0.05$. We also calculated the bits per character for the entire inkblot group as 4.63 and the bits per character for the entire password group as 4.28.

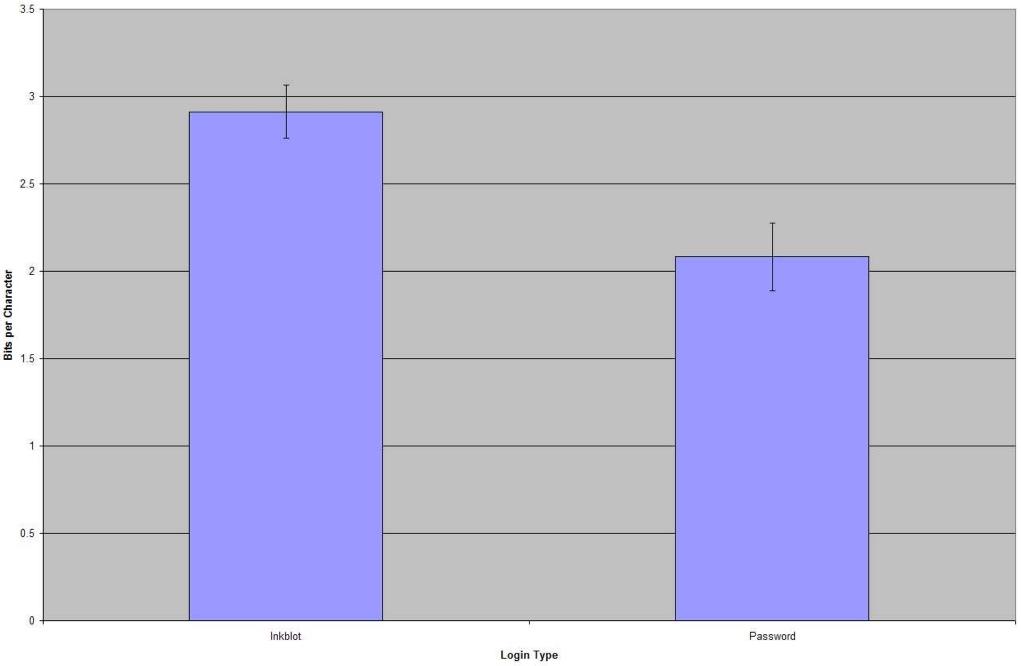


Figure 6.25: Bits per Character

Analysis then indicated that the length of the response in characters was significantly affected by the login type and that the length of inkblot responses (M=11.12, SE: 1.71) was significantly higher than the length of passwords (M: 6.00, SE: 0.6),t(19.88)=2.823, $p < 0.05$.

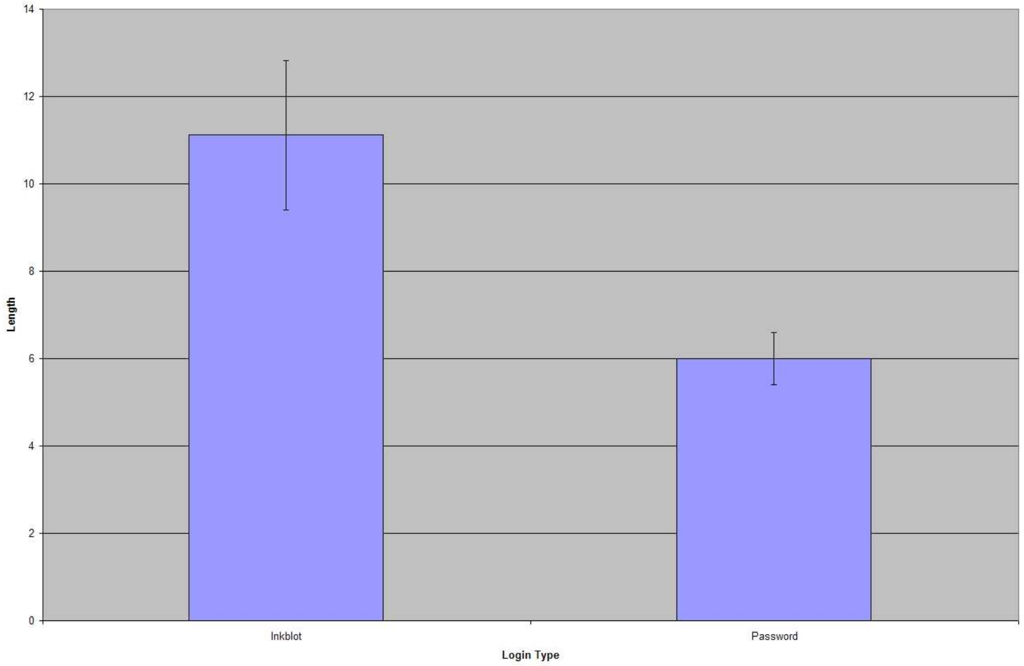


Figure 6.26: Response Length

Since the number of bits per response is the product of the number of bits per character and the number of characters we would expect the number of bits per response to also be significantly affected by the login type. In fact this was the case, and we found that the number of bits in the inkblot response (M: 36.01, SE: 7.13) was significantly higher than the number of bits in the password response (M: 14.10, SE: 2.08), t(32)=2.947, $p < 0.05$.

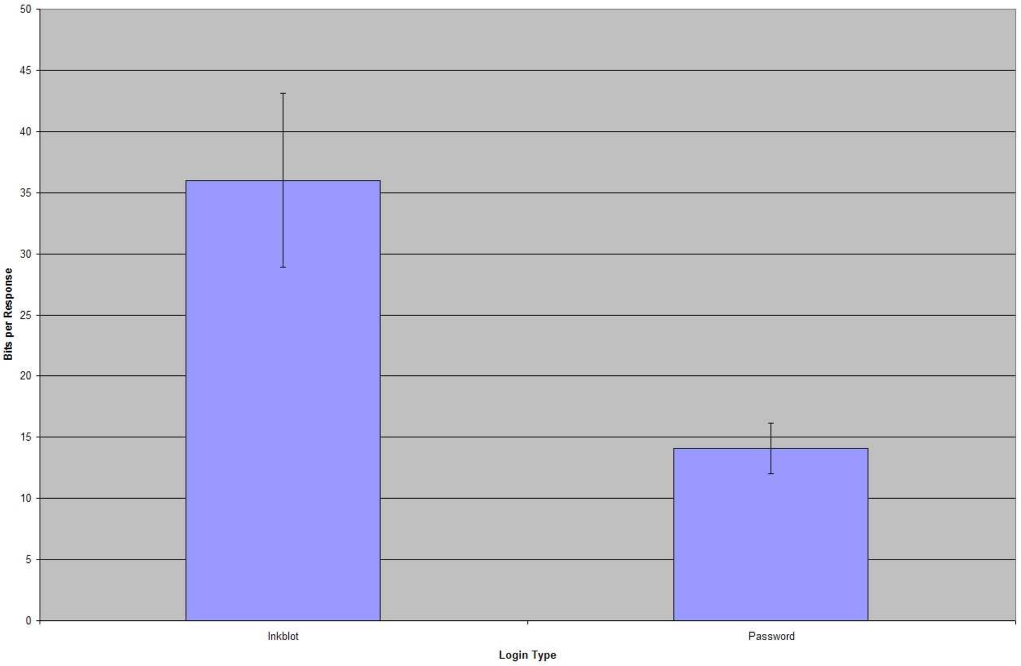


Figure 6.27: Bits per Response

We performed the Levenshtein ratio analysis as in the previous sections and discovered that there was no significant difference between inkblot (M: 0.75, SE: 0.09) and password (M: 0.72, SE 0.15) Levenshtein ratios ($p > 0.05$), this is likely because the inkblot users were all describing different inkblots and as such there were few common features in the blots.

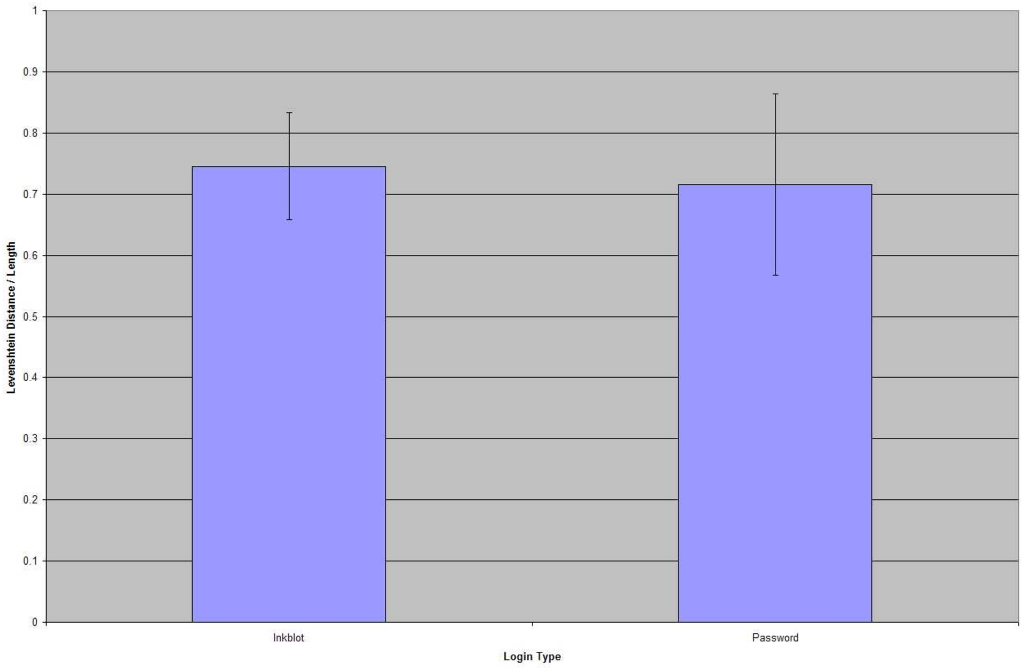


Figure 6.28: Levenshtein Ratio

The final point to make regarding the security of the password is regards some very poor password choices. 3 out of the 17 password users used their exact username as the password and one user used a single character "h" as their password.

6.4.3 Time to login (Usability)

To obtain a measure of comparative usability of the two systems we measured the amount of time it took users to login to the system and found that inkblots (M: 11.59, SE: 1.87) took significantly longer to login than passwords (M: 6.29, SE: 11.59), t(106)=2.3, $p < 0.05$. It should be noted that the ratio of time to login for the two login types is similar to the ratio of response length obtained in Section 6.4.2.

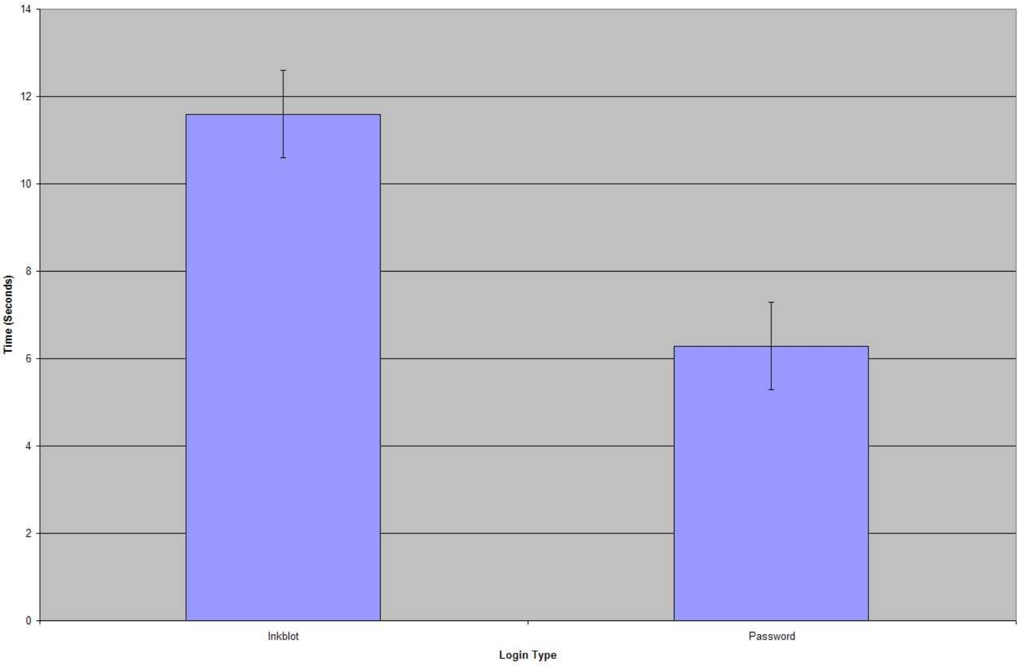


Figure 6.29: Time to Login (Seconds)

6.4.4 Session Details

For comparison purposes we categorised sessions into five types as follows :

- Single successful attempt
- Single failed attempt
- Single failed attempt followed by success
- Several failed attempts followed by success
- Several failed attempts with no later success

During the course of our experiments we had no actual incidences of sessions that were not eventually successful thus the only session types that actually occurred were a single successful attempt, a single failure then success and several failures followed by success. We believe this shows how users will generally be able to recover from failures.

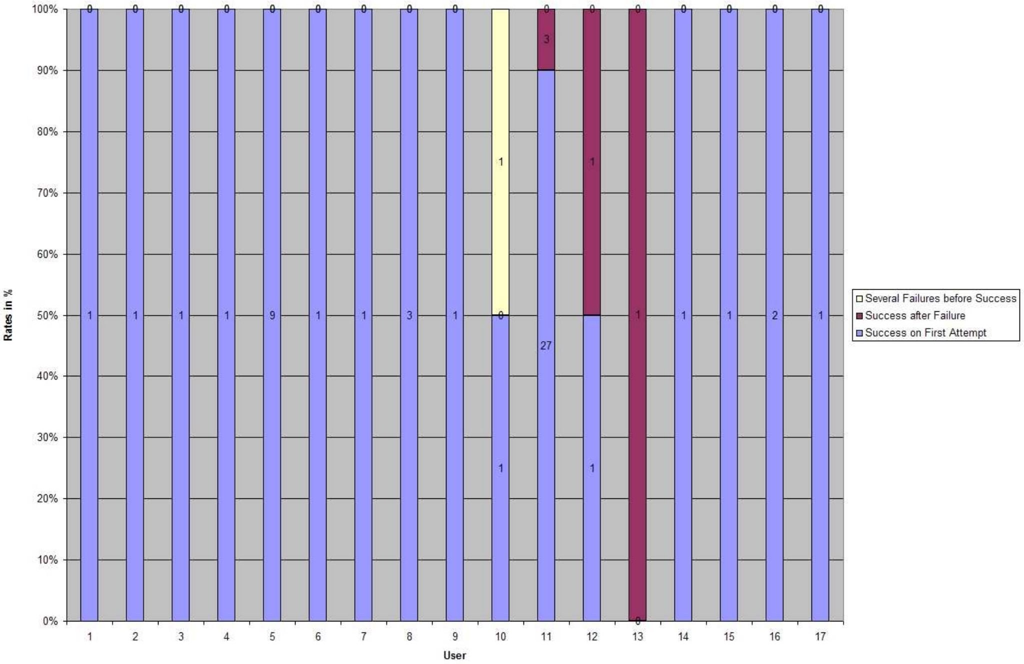


Figure 6.30: Session Data for Inkblots

Our results indicate that there is a much lower rate of errors for the password login type than in the inkblot login type; we hypothesise that this may be due to our users familiarity with the traditional password model.

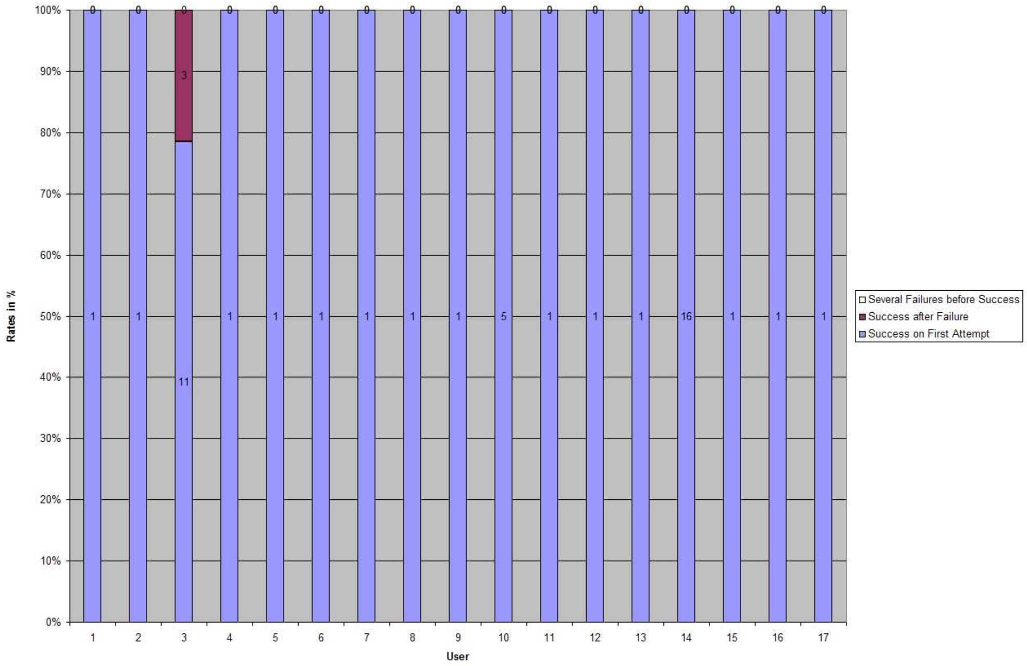


Figure 6.31: Session Data for Passwords

In the next section we will discuss the results we have found in relation to our original research questions followed by comparisons of usability and quality coefficients for the password and inkblot login types.

Chapter 7

Discussion

7.1 Summary

In this section we will summarise the results found in Section 6 before considering how these results will affect the original questions we proposed in Section 3.2.

In Section 6.2 we performed an experiment to gather textual associations from images and compared the resulting associations across image class and individual image boundaries. The results in Section 6.2.1 showed that the Texture and Faces classes were the best classes for response rate followed by the Fractals and Inkblots class and finally the Snowflakes class which had the worst performance. Furthermore we found that the individual images did not vary significantly within a class.

Section 6.2.2 was concerned with the length of the textual responses gathered and demonstrated that the Inkblot and Fractals class had the longest responses and Textures the shortest. The other classes had lengths between the two extremes. We found that examining the length of the images within the image class resulted in only a single significant difference between images which was for images 21 and 22 in the Snowflakes class.

We then looked at the character distribution for associations in Section 6.2.3 and observed that the character frequencies resembled those of English. Although this is to be expected with English textual descriptions it is worth noting that this is actually the case.

We then measured the level of entropy, or randomness, within the textual descriptions in Section 6.2.4 and found that images in the Textures class were significantly less random than the other image classes. There were no other significant effects within the image classes however two images showed significantly different levels of randomness than other images within their classes. Image 22 (Snowflake) had significantly higher randomness than the images 20,21 and 24 while image 28 (Texture) had significantly lower randomness than images 29 and 30.

We then looked at the overall randomness of the textual association by measuring the number of bits per response and found that Textures had significantly lower number of bits per response than all other classes and included two images that differed from each other significantly (Images 22 and 25) in terms of number of bits per response.

We then looked at how different the responses given for each image were from each other in Section 6.2.5 by measuring the Levenshtein distance between strings and then using this in a ratio of Levenshtein distance over string length. The results indicated that the descriptions in the Snowflakes class were the least different followed by the Textures class while the remaining three classes were roughly the same performance. Looking at the images within the classes we found a number of significant effects on certain images. In the Faces class we found that there was a division among the sex of the face shown and that the descriptions for female faces were less different from each other than the male faces. There was no such division in the Fractals although Inkblots had a division along colour where the simple black and white images had higher average difference. The Snowflakes class had a division along complexity where it seemed high complexity images encouraged results to have higher average difference. The final class, Textures, showed a division along identifiability where easily identifiable images had lower average difference. Our final observation for Levenshtein distance was that in each image there was at least one case of people choosing the same textual association ± 1 character.

In Section 6.2.6 we measured the similarity of responses by measuring local sequence alignment to find matching sequences of characters within the strings. This enabled us to find substrings of the responses that were highly similar. We found that there was a significant difference between all image classes in this analysis and that the Smith-Waterman ratio was lowest (best) for Inkblots followed by Fractals, Snowflakes, Faces and then Textures. We found that images in the Faces class were more similar if they had distinctive features such as visible earrings or distinctive facial hair. Some of the Fractals images had higher similarity which we believe may be caused by higher symmetry but the tests had a counter-example to this theory which had high symmetry but did not score a similarly high Smith-Waterman ratio. The Snowflakes image class divided into three levels of Smith-Waterman ratio, the low to mid complex images had high ratios which the most complex and images generated with maximum numbers of rays had much lower ratios. Like the Levenshtein results the most easily identifiable images had the highest Smith-Waterman ratios making them the most similar.

In Section 6.3 we analysed the results of a second experiment where we asked the previous participants to repeat the study and attempt to enter the same associations they had previously. We found that there was no difference on image recall for either image class or for individual images and that declines in accuracy were primarily attributable to time which appeared to have a roughly linear drop-off rate.

Section 6.4 detailed the results gathered from a third experiment where we implemented a live system using Inkblots and traditional Password login types and compared the results of a user base of 34 people using the system (17 in each group). In Section 6.4.2 we compared the security of the inkblot derived responses gathered by the system to the passwords used in the system and found that inkblots had more bits per character than passwords as well as a larger number of characters and more bits per response. We also found that the Levenshtein ratio for both login types was the same.

In Sections 6.4.3 and 6.4.4 we investigated the usability of the system by measuring the time taken to login and login success rates for each user session. We found that inkblots users take much longer to login on average, however the increased login time was in the same ratio as ratio of the difference of the length of the response, so it is possible that the extra login time is attributable to the extra time needed to key the longer inkblot response in. We then found that there were more failures to login in the inkblot group but despite initial failures all the users were able to login after retrying.

7.2 Revisiting the Research Questions

To determine the outcomes of this research we need to revisit our original scientific questions we posed in Section 3.2. Our first question was to determine if when users are presented with an image as a cue if they are able to generate textual associations and we can make the assertion that this is the case and users can generate textual associations from images based on the response rates we obtained in Sections 6.2.1 and 6.3.1.

The second question we posed was if when presented with the same image after a specific duration of time is the user able to recreate their textual association. The answer to this is that they can although the accuracy of the recreation is decreased linearly by the amount of time that has passed.

We then asked if textual descriptions based on images were unpredictable enough to be used as passwords and we believe that our comparison with the password login type in Section 6.4.2 shows that this is the case.

Following up on our question on memory we asked if some image types offer better memorability than other image types to which the reply is in the negative. We found that time was the only appreciable factor in memorability and that the image type did not make any statistical difference in this regard.

We then asked if some image types offer better security than other image types and found that this was the case. In Sections 6.2.4, 6.2.5 and 6.2.6 we performed a number of comparisons between image classes and found that Textures and Snowflakes were consistently poor performers while Inkblots and Fractals performed consistently well across all tests.

We had originally asked if inkblot type images encouraged higher strength descriptions than other image types and as discussed above we found that they did. We also asked if human faces encourage higher memorability than other image types but since the image classes did not affect memorability this was not the case.

The next question posed was if variations within images in an image class resulted in different results for memorability than other images within the class but our results in Section 6.3 indicate this is not the case, we found that individual images had no bearing on memorability of the textual associations.

We had asked if variations within the image class resulted in different levels of security for the textual association than other images within the class and found that this was in-fact the case for some images. In Section 6.2 we found that some of our image classes had improved or decreased performance when generated or chosen with certain parameters. For example we found that the descriptions of female faces had lower Levenshtein ratios meaning they had lower average difference between strings.

We also asked questions regarding the usability of the system, in particular if the system was more difficult to use than the traditional password system. In Section 6.4 we found that users using the image login system tended to make more mistakes and take longer to login than users using the traditional password system.

We provide a short summary of the results in the following table.

	Question	Result
1.	When presented with an image as a cue are users able to generate textual associations?	√
2.	When presented with the same image after a specific duration of time can the user recreate their textual association?	√
3.	Are textual descriptions based on images unpredictable enough to be used as passwords?	√
4.	Do some image types offer better memorability than other image types?	×
5.	Do some image types offer better security than other image types?	√
6.	Do inkblot-type image encourage higher strength descriptions than other image types?	√
7.	Do human faces encourage higher memorability than other image types?	×
8.	Do variations within the image class result in altered memorability than other images within the class?	×
9.	Do variations within the image class result in altered security than other images within the class?	√
10.	Is the system more difficult to use than the traditional password systems?	√

7.3 Gestalt Laws

When we originally chose our image classes to be used in this experiment in Section 3.3 we applied the Gestalt laws to choose images which exhibited Closure, Continuity, Proximity, Similarity and Symmetry. We now wish to evaluate whether these laws had any effect on the results of the experiment.

The images were evaluated and the following Gestalt laws were observed in the images chosen.

Image Type	Closure	Continuity	Proximity	Similarity	Symmetry
Faces	√	√			√
Fractals		√		√	
Inkblot	√	√	√	√	√
Snowflakes	√	√	√	√	√
Textures	√		√	√	√

We observe that Snowflakes and Inkblots image classes both exhibited elements of all 5 Gestalt laws however performed very differently in tests. We further observe that Fractals which had the lowest number of Gestalt laws performed well.

Looking at each rule in turn we find that for each of the five rules the image classes that were found to exhibit that rule varied in performance as did the image classes that did not meet the rule. As such we do not believe that the Gestalt laws had any measurable effect on the results of our image classes.

7.4 Usability

In Section 2.12 we listed the usability measures defined in ISO 9241-11[Org98] as effectiveness, efficiency and satisfaction. The effectiveness metric is designed to measure the ability of a user to accomplish a task and how many errors they make during the process, it is typically goal oriented. The efficiency metric is the speed at which the user can accomplish the goal and the satisfaction metric measures how satisfied the user was with the system and their performance in reaching their goal. We did not perform an evaluation of user satisfaction due to time constraints but we examined the remaining two aspects in our stage three experiment by comparing inkblots to passwords.

In the third stage of the experiments our users were presented with the goal of successfully logging into the system. In Section 6.4.4 we obtained the success rates of attempted logins which provides us an effectiveness (or efficacy) metric for the users login attempts to measure how well users accomplished the goal of logging in. Section 6.4.3 provides measurements of the amount of time to login which provides us with a usability efficiency metric which we can use to evaluate the efficiency of the login goal for each login type.

In both cases we found that the passwords had the higher usability score, in the effectiveness tests we found that the users made more mistakes when logging in (failed login attempts) and took longer to login. Since the mechanics of providing responses are the same as passwords it may be possible that the differing performances may be simply due to high user familiarity with the traditional password system. We also note that the length of time to login (efficiency score) is higher for inkblots but remind the reader that the increase in login time is proportional to the increase in number of characters in the response.

7.5 Quality Coefficients

To further compare the inkblot login method with the traditional password method we introduce the idea of Quality Coefficients as described by Renaud[Ren04].

In the following metrics we use *ad*, *md*, *sd* and *vd* to represent the root sum squares of the 3 metrics ($\sqrt{x^2+y^2+z^2}$) of the three coefficients for Accessibility, Memorability, Security and Vulnerability respectively.

We first examine the inkblot authentication system, the coefficient score indicates the relative severity of the problem using the metrics established in the referenced.

```
Accessibility (ad = 0.6)
   Special Requirements    0      None
   Convenience             0.25   Time consuming to design inkblots
   Inclusivity             0.33   Cognitive

Memorability (md = 0.83)
   Retrieval               0.5    Inkblot provides cue
   Meaningfulness          0      Meaningful descriptions
   Depth of Processing     0.67   Visual images

Security (sd = 1.41)
   Predictability          1      Some users choose the same associations
   Abundance               0      [a-zA-Z0-9 ]^{11.12}
   Disclosure              1      Can be written down

Vulnerability (vd = 1.41)
   Confidentiality         1      Full
   Privacy                 0      No personal details required
   Breakability            1      Vulnerable to keyboard tapper
```

The deficiency value ($\overline{d_{web}}$) weighted for web site based systems is calculated as below. The environmental quality ($\overline{eq_{web}}$) indicator is simply the maximum score (13) subtracting the deficiency score for that environment.

$$\begin{aligned}
\overline{d_{web}} &= ad*1.5+md+sd*1.5+vd \\
&= 5.3 \\
\overline{eq_{web}} &= 7.7
\end{aligned}$$

We then repeat this exercise for traditional password authentication systems using the same weights.

```
Accessibility (ad = 0.33)
   Special Requirements    0      None
   Convenience             0      Very convenient
   Inclusivity             0.33   Congnitive

Memorability (md = 1.52)
   Retrieval               1      Recall
   Meaningfulness          0.33   Usually meaningful
   Depth of Processing     1      Cursory

Security (sd = 1.58)
   Predictability          1      Usually self-assigned
   Abundance               0.5    [a-zA-Z0-9]^{6}
   Disclosure              1      Passwords are easy to record

Vulnerability (vd = 1.41)
   Confidentiality         1      Full
   Privacy                 0      Not necessarily private
   Breakability            1      Vulnerable to keyboard tapper and Dictionary attack
```

$$\begin{aligned} \overline{d_{web}} &= ad * 1.5 + md + sd * 1.5 + vd \\ &= 5.8 \\ \overline{eq_{web}} &= 7.2 \end{aligned}$$

We therefore find that the environmental quality metric ($\overline{eq_{web}}$) for inkblot based systems is higher than the comparative password based system which indicates its greater suitability in this regard.

Chapter 8

Conclusion

In this research we initially introduced the idea of authentication where we showed the inconvenience and monetary cost of forgotten and lost passwords.

Our literature review reveals that the current authentication systems in existence are lacking due to insufficient memorability or security and showed that despite widespread adoption the traditional password model had a number of drawbacks. These drawbacks were primarily due to the choice of passwords resulting in either passwords which were very random which were difficult to remember or passwords which were easy to remember but easier to guess.

The individual advantages of graphical, associative and traditional passwords were discussed and analysis of the properties of these existing systems led us to conclude that to advance the state of the art it would be necessary to combine the advantages of these systems to create an authentication system which used textual associations derived from images as passwords.

After we detailed our improved authentication system this led us to detail a number of hypotheses that define the properties we expect the system to adhere to based on the results of our literature review. These hypotheses then guided the creation of a list of scientific questions which need to be answered for testing the performance of the proposed system and comparing it against traditional techniques.

These scientific questions prompted us to design a series of experiments designed to determine the validity of the hypotheses and to answer the questions. We explained our design goals and methods which we would intend to use to test our authentication system.

We detailed our results in Section 6 and provided a summary of the results in Section 7.1. We believe that we have found some interesting results, including the importance of image class choice for the quality of the textual association and its non-effect on memorability. Our finding of linear memory drop-off was also unexpected but doubtless of note.

The results of our experiments indicate that the inkblot style authentication is certainly feasible although has some usability issues relating to the length of time to login and the number of errors made while logging in. We feel that by presenting this new inkblot authentication system we have provided additional options for system designers and researchers.

Our final comment is that we believe the task of performing three user studies within the allotted time-period was trying to do too much but that despite some minor timescale issues the research was clearly a success and addressed all the questions we set out to answer.

Chapter 9

Future Work

We would like to see further research into alternative authentication systems to improve password choice and hope that this research has shown that such systems are both possible and competitive to traditional authentication mechanisms. Such improved systems could also attempt to improve usability of the authentication system. We would be very interested in any system that could improve both security and usability.

Investigation into different types of cues or different image classes would seem prudent to determine if any other type of cue can enduce better password choices or higher usability.

We would be especially interested in further studies concerning memory effects on other authentication systems such as passwords to determine if they demonstrate the same behaviour of linear drop-off over time.

Chapter 10

Appendixes

10.1 Image Parameters

10.1.1 Faces

Images taken from Essex University Computer Vision Facial Databases[Spa06].

Face	Description	Image
1	Female with neutral expression	faces94/female/asamma.6.jpg
2	Male with neutral expression	faces95/sirmcb/sirmcb.14.jpg
3	Female smiling	faces94/faces/phughe/phughe.15.jpg
4	Male smiling, beard, glasses	faces94/malestaff/spacl.15.jpg
5	Female frowning	faces94/female/vstros/vstros.16.jpg
6	Male frowning, beard	faces95/mizli/mizli.15.jpg

10.1.2 Fractals

Images generated using Ultra Fractal[Sli05].

Fractal 1	
Location	
Center (Re)	0
Center (Im)	0
Magnification	1
Rotation Angle	0.1
Stretch (X/Y)	1
Skew Angle	0
Left Top (Re)	-2.003487611
Left Top (Im)	1.996506297
Right Top (Re)	1.996506297
Right Top (Im)	2.003487611
Right Bottom (Re)	2.003487611
Right Bottom (Im)	-1.996506297
Formula	
Newton	
Drawing Method	Guessing
Periodicity Checking	Normal
Additional Precision	0
Maximum Iterations	100
Exponent (Re)	4
Exponent (Im)	0
Root (Re)	1
Root (Im)	0
Colouring Algorithm	
Orbit Traps	
Color Density	1
Transfer Function	Log
Solid Color	Black
Gradient Offset	0
Repeat Gradient	
Trap Shape	egg
–Diameter	1.0
–Order	4.0
Trap Colouring	distance
Trap Mode	closest
–Threshold	0.25
Trap Center (Re)	0
Trap Center (Im)	0
Aspect Ratio	1.0
Rotation	0.0

Fractal 2	
Location	
Center (Re)	-0.91875000019
Center (Im)	1.1375
Magnification	11.428571
Rotation Angle	-0.1532
Stretch (X/Y)	1
Skew Angle	0
Left Top (Re)	-1.0932814589
Left Top (Im)	1.3129673032
Right Top (Re)	-0.74328269697
Right Top (Im)	1.3120314588
Right Bottom (Re)	-0.74421854144
Right Bottom (Im)	0.96203269678
Formula	
Mandelbrot	
Drawing Method	Guessing
Periodicity Checking	Normal
Additional Precision	0
Maximum Iterations	100
Starting Point (Re)	0
Starting Point (Im)	0
Power (Re)	2.5
Power (Im)	0
Bailout value	4
Colouring Algorithm	
Orbit Traps	
Color Density	1
Transfer Function	Log
Solid Color	Black
Gradient Offset	0
Repeat Gradient	
Trap Shape	ring
–Diameter	1.0
Trap Colouring	distance
Trap Mode	closest
–Threshold	0.25
Trap Center (Re)	0
Trap Center (Im)	0
Aspect Ratio	1.0
Rotation	0.0

Fractal 3	
Location	
Center (Re)	-0.59249094145
Center (Im)	0.56742365795
Magnification	25.511836
Rotation Angle	-0.1532
Stretch (X/Y)	1
Skew Angle	0
Left Top (Re)	-0.6706760313
Left Top (Im)	0.6460279793
Right Top (Re)	-0.5138866201
Right Top (Im)	0.6456087479
Right Bottom (Re)	-0.5143058516
Right Bottom (Im)	0.4888193366
Formula	
Mandelbrot	
Drawing Method	Guessing
Periodicity Checking	Normal
Additional Precision	0
Maximum Iterations	100
Starting Point (Re)	0
Starting Point (Im)	0
Power (Re)	2.5
Power (Im)	0
Bailout value	4
Colouring Algorithm	
Orbit Traps	
Color Density	1
Transfer Function	Log
Solid Color	Black
Gradient Offset	0
Repeat Gradient	
Trap Shape	ring
–Diameter	1.0
Trap Colouring	distance
Trap Mode	closest
–Threshold	0.25
Trap Center (Re)	0
Trap Center (Im)	0
Aspect Ratio	1.0
Rotation	0.0

Fractal 4	
Location	
Center (Re)	0.10000000059
Center (Im)	0.2796874990935
Magnification	7.5294117
Rotation Angle	0
Stretch (X/Y)	1
Skew Angle	0
Left Top (Re)	-0.16562500169
Left Top (Im)	0.47890625081
Right Top (Re)	0.36562500287
Right Top (Im)	0.47890625081
Right Bottom (Re)	0.36562500287
Right Bottom (Im)	0.080468747381
Formula	
Julia	
Drawing Method	Guessing
Periodicity Checking	Normal
Additional Precision	0
Maximum Iterations	100
Julia Seed (Re)	-0.84302
Julia Seed (Im)	0.22093
Bailout value	4.0
Colouring Algorithm	
Smooth (Mandelbrot)	
Colour Density	1
Transfer Function	Linear
Solid Colour	Black
Gradient Offset	0
Repeat Gradient	
Exponent (Re)	2
Exponent (Im)	0
Bail-out value	128.0

Fractal 5	
Location	
Center (Re)	-0.029965048
Center (Im)	-0.0200523295
Magnification	1.2987013
Rotation Angle	0.1
Stretch (X/y)	1
Skew Angle	0
Left Top (Re)	-1.572650507
Left Top (Im)	1.517257518
Right Top (Re)	1.507344799
Right Top (Im)	1.522633129
Right Bottom (Re)	1.512720411
Right Bottom (Im)	-1.557362177
Formula	
Unity	
Drawing Method	Guessing
Periodicity Checking	Normal
Additional Precision	0
Maximum Iterations	149
Colouring Algorithm	
Gaussian Integer	
Colour Density	0.25
Transfer Function	Linear
Solid Colour	Black
Gradient Offset	0
Repeat Gradient	
Integer Type	round(z)
Color By	minimum distance
Normalization	none

Fractal 6	
Location	
Center (Re)	-0.5
Center (Im)	0
Magnification	1.5
Rotation Angle	90
Stretch (X/y)	1
Skew Angle	0
Left Top (Re)	-1.833333333
Left Top (Im)	-1.833333333
Right Top (Re)	-1.833333333
Right Top (Im)	1.333333333
Right Bottom (Re)	0.8333333333
Right Bottom (Im)	1.333333333
Formula	
Nova (Mandelbrot)	
Drawing Method	Guessing
Periodicity Checking	Off
Additional Precision	0
Maximum Iterations	1000
Start Value (Re)	1
Start Value (Im)	0
Exponent (Re)	3
Exponent (Im)	0
Bailout	0.00001
Relaxation (Re)	1
Relaxation (Im)	0
Colouring Algorithm	
Smooth (Mandelbrot)	
Colour Density	0.5
Transfer Function	Log
Solid Colour	Black
Gradient Offset	0
Repeat Gradient	
Exponent (Re)	2
Exponent (Im)	0
Bail-out value	128.0

10.1.3 Inkblots

Images generated using the custom PHP script below.

```php
<?

/*
 * Simple PHP script to generate random inkblots
 * Copyright: Tony McBryan 2005 onwards
 * File History:
 *           11/6/2005 - First written
 *           11/6/2005 - Blots now cluster together
 *                     - Blots much larger
 *                     - Fewer blots
 *           01/3/2006 - Colours
 *                     - Parameratised from GET input
 *                     - Seed parameratised
 */

function validInt($var)
{
        return isset($var)&&ctype_digit($var);
}

// random number seed
if (validInt($_GET['seed']))
{
        mt_srand($_GET['seed']); // seed
}
// else if mt_srand not called then its implied

// Variables to change how blots look
$imgWidth=250;  // the output image will actually be double this width
$imgHeight=500;
// blot stats (can be parameratised from get input)
// defaults : $maxDiameter=30;  $minDiameter=0; $numbBlots=1500;
$maxDiameter=(validInt($_GET['maxDiameter']))?$_GET['maxDiameter']:30;
$minDiameter=(validInt($_GET['minDiameter']))?$_GET['minDiameter']:0;
$numbBlots=(validInt($_GET['numbBlots']))?$_GET['numbBlots']:1500;

// distance between blots
// defaults : $maxSeperationH=15; $maxSeperationV=15;
$maxSeperationH=(validInt($_GET['maxSepH']))?$_GET['maxSepH']:15;
$maxSeperationV=(validInt($_GET['maxSepV']))?$_GET['maxSepV']:15;

// number of colours
// default = 1; (black and white)
$numbpasses =(validInt($_GET['numbColors']))?$_GET['numbColors']:1;

// Define .PNG image
header("Content-type: image/png");
// Create image
// actual image = 2* imgWidth as ink blot is mirrored
$image = imagecreate($imgWidth*2,$imgHeight);

$colourWhite = imagecolorallocate($image,255,255,255); // bkg colour


for ($currentpass=0;$currentpass<$numbpasses;$currentpass++)
{
        if ($numbpasses != 1)
        {
            $colourBlack = imagecolorallocate( $image,
                                               mt_rand(0,255),
                                               mt_rand(0,255),
                                               mt_rand(0,255)); // blot colour
        }
        else
        {
                $colourBlack = imagecolorallocate($image, 0,0,0); // black
        }

        // make a starting position
        $hPosPrev = mt_rand(0+$maxDiameter,$imgWidth);
        $vPosPrev = mt_rand(0+$maxDiameter,$imgHeight-$maxDiameter);

        // make $numbBlots worth of blots
        for ($i=0;$i<$numbBlots;$i++)
        {

                // set bounds
                $leftBound = $hPosPrev - $maxSeperationH;
                $rightBound = $hPosPrev + $maxSeperationH;

                // establish shift amount
                if ($leftBound<$maxDiameter)
                {
                        // shift right
                        $shift=$maxDiameter-$leftBound;
                }
                else if ($rightBound>$imgWidth)
                {
                        // shift left
                        $shift=$imgWidth-$rightBound;
                        // note this will be -ve number
                }
                else
                {
                        // no shift
                        $shift=0;
                }

                $leftBound=$leftBound+$shift;
                $rightBound=$rightBound+$shift;

                // set V bounds
                $topBound = $vPosPrev + $maxSeperationV;
                $bottomBound = $vPosPrev - $maxSeperationV;

                // establish V shifts
                if ($bottomBound<$maxDiameter)
                {
                        // shift up
                        $shift = $maxDiameter;
                }
                else if ($topBound>($imgHeight-$maxDiameter))
                {
                        // shift down
                        $shift = 0-$maxDiameter;
                }
                else
                {
                        // no shift
                        $shift = 0;
                }

                $topBound=$topBound+$shift;
                $bottomBound=$bottomBound+$shift;

                // blot position
                $hPos = mt_rand($leftBound, $rightBound);
                $vPos = mt_rand($bottomBound,$topBound);

                // blot diameter
                $diameter = mt_rand($minDiameter,$maxDiameter);
                $height = $diameter;
                $width = $diameter;

                // note last position
                $hPosPrev = $hPos;
                $vPosPrev = $vPos;


                // draw left hand blot
                imagefilledellipse($image,$hPos,$vPos,$width,$height,$colourBlack);

                // now make the mirrored version (right hand side) of it
                $hPos = $imgWidth*2-$hPos;
                imagefilledellipse($image,$hPos,$vPos,$width,$height,$colourBlack);
        }
        $numbBlots = $numbBlots/2;
```

```
}

// Output image and destroy from memory
imagepng($image);
imagedestroy($image);

?>
```

The following parameters were used to generate the images.

Inkblot 1	
Colours	1 (Black)
Max Blot Diameter	25
Min Blot Diameter	0
Numb Blots (per colour)	1250
Max Seperation H	15
Max Seperation V	15

Inkblot 2	
Colours	3
Max Blot Diameter	25
Min Blot Diameter	0
Numb Blots (per colour)	1000,500,250
Max Seperation H	15
Max Seperation V	15

Inkblot 3	
Colours	1 (Black)
Max Blot Diameter	25
Min Blot Diameter	0
Numb Blots (per colour)	1250
Max Seperation H	15
Max Seperation V	15

Inkblot 4	
Colours	5
Max Blot Diameter	25
Min Blot Diameter	0
Numb Blots (per colour)	1250,625,312,156,78
Max Seperation H	15
Max Seperation V	15

Inkblot 5	
Colours	1 (Black)
Max Blot Diameter	25
Min Blot Diameter	0
Numb Blots (per colour)	1000
Max Seperation H	15
Max Seperation V	15

Inkblot 6	
Colours	5
Max Blot Diameter	25
Min Blot Diameter	0
Numb Blots (per colour)	750,375,187,93,46
Max Seperation H	15
Max Seperation V	15

10.1.4 Snowflakes

Images generated using Fractal Snowflake Generator[A.I04].

Snowflake 1 (medium settings)	
rays	15
complexity	6
scaling	52
position	52
angle	87
rotate	0.0
scale	100
background	blue
foreground	white
anti-alias	on
fade	on
randomise	on

Snowflake 2 (max rays, max complexity)	
rays	30
complexity	12
scaling	106
position	100
angle	75
rotate	0.0
scale	10
background	gray
foreground	white
anti-alias	on
fade	on
randomise	on

Snowflake 3 (min rays, min complexity)	
rays	5
complexity	3
scaling	75
position	60
angle	160
rotate	0.0
scale	75
background	black
foreground	white
anti-alias	on
fade	on
randomise	on

Snowflake 4 (max rays, min complexity)	
rays	30
complexity	5
scaling	100
position	50
angle	50
rotate	0.0
scale	50
background	purple
foreground	white
anti-alias	on
fade	on
randomise	on

Snowflake 5 (min rays, max complexity)	
rays	5
complexity	10
scaling	100
position	50
angle	60
rotate	0.0
scale	100
background	yellow
foreground	white
anti-alias	on
fade	on
randomise	on

Snowflake 6 (min rays, max complexity)	
rays	5
complexity	10
scaling	80
position	50
angle	15
rotate	0.0
scale	50
background	white
foreground	black
anti-alias	on
fade	on
randomise	on

10.1.5 Textures

Images taken from the CUReT texture database[DGNK05].

Texture	Image
1	Sample image of Limestone http://www1.cs.columbia.edu/CAVE/curet/html/about36.html
2	Sample image of Sandpaper http://www1.cs.columbia.edu/CAVE/curet/html/about06.html
3	Sample image of Tree Bark http://www1.cs.columbia.edu/CAVE/curet/html/about58.html
4	Sample image of Lettuce Leaf http://www1.cs.columbia.edu/CAVE/curet/html/about23.html
5	Sample image of Crumpled Paper http://www1.cs.columbia.edu/CAVE/curet/html/about28.html
6	Sample image of Styrofoam http://www1.cs.columbia.edu/CAVE/curet/html/about20.html

Bibliography

[A.I04] A.I.Studio. Fractal Snowflake Generator. http://a-i-studio.com/snowflake/, 2004.

[Ana82] Anne Anastasi. *Psychological Testing*. New York: McMillen and Co., 1982.

[AS99] Anne Adams and Martina Angela Sasse. Users Are Not The Enemy. *Communications of the ACM*, 42(12):40–46, 1999.

[Blo96] G. Blonder. Graphical Passwords - United States Patent 5559961, 1996.

[BS00] Sacha Brostoff and M. Angela Sasse. Are Passfaces More Usable Than Passwords? A Field Trial Investigation. In *Proceedings of HCI 2000*, pages 405–424, 2000.

[CAS$^+$97] C Curtis, S Anderson, J Seims, K Fleischer, and D Salesin. Computer Generated Watercolor. In *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, pages 421–430, 1997.

[CDW04] Art Conlin, Glenn Dietrich, and Diane Walz. Password-Based Authentication: A System Perspective. In *Proceedings of the 37th Annual Hawaii International Conference on Systems Sciences*, 2004.

[DGNK05] Kristin J. Dana, Bram Van Ginneken, Shree K. Nayar, and Jan J. Koenderink. Columbia-Utrecht Reflectance and Texture Database. http://www1.cs.columbia.edu/CAVE/curet/, 2005.

[DOK04] Christine E. Drake, Jonathan J. Oliver, and Eugene J. Koontz. Anatomy of a Phishing Email. In *Conference on Email and Anti Spam (CEAS)*, 2004.

[DP00] Rachna Dhamija and Adrian Perrig. Déjà Vu: A User Study Using Images for Authentication. In *9th USENIX Security Symposium*, 2000.

[FCC] Federal Communications Commission. Code of Federal Regulations 47 CFR § 15.

[FDIR92] S.M. Furnell, P.S. Dowland, H.M. Illingworth, and P.L. Reynolds. Opus: Preventing Weak Password Choices. *Computers and Security*, 11(3):273–278, 1992.

[Gal10] Antonine Galland. *Ali Baba and the Forty Thieves from The Thousand and One Nights aka. Arabian Nights*. Public Domain, Circa 1710.

[GC70] Alvin G. Goldstein and June E. Chance. Visual recognition memory for complex configurations. *Perception and Psychophysics*, 9(2B), 1970.

[Hal95] Neil M. Haller. The S/KEY one-time password system. Technical report, Internet RFC 1760, 1995.

[Ham03] Ben Hammersley. *Content Syndication with RSS*. O'Reilly, 2003.

[Has84] James A. Haskett. Pass-algorithms: a user validation scheme based on knowledge of secret algorithms. *Communications of the ACM*, 27(8):777–781, 1984.

[Inc04] RSA Security Inc. Are passwords really free?, 2004. White paper.

[Inc06] SPSS Inc. Spss statistics package. http://www.spss.com/, 2006.

[JHP00] Anil Jain, Lin Hong, and Sharath Pankanti. Biometric identification. *Communications of the ACM*, 43(2):90–98, 2000.

[JMM$^+$99] Ian Jermyn, Alain Mayer, Fabian Monrose, Michael K. Reiter, and Aviel D Rubin. The design and analysis of graphical passwords. In *Proceedings of the 8th USENIX Security Symposium*, 1999.

[JO89] David L. Jobusch and Arthur E. Oldenhoeft. A survey of password mechanisms: Weaknesses and potential improvements. part 1. *Computers and Security*, 8(7):587–601, 1989.

[KI99] Gershon Kedem and Yuriko Ishihara. Brute Force Attack on UNIX Passwords with SIMD Computer. In *Proceedings of the 8th USENIX Security Symposium*, 1999.

[Kle90] Daniel V. Klein. Foiling the Cracker : A survey of, and improvements to, password security. In *Proceedings of the USENIX Second Security Workshop*, pages 5–14, 1990.

[Kuh02] Markus G. Kuhn. Optical Time-Domain Eavesdropping Risks of CRT Displays. In *Proceedings of the 2002 IEEE Symposium on Security and Privacy*, pages 3–18, 2002.

[Lev65] Vladimir I. Levenshtein. Binary codes capable of correcting deletions, insertions and reversals. *Doklady Akademii Nauk SSSR*, 163(4):845–848, 1965.

[LRA03] Jim Liddell, Karen Renaud, and Antonella De Angeli. Using A Combination of Sound and Images to Authenticate Web Users. In *17th Annual Human Computer Interaction Conference. Designing for Society*, 2003.

[Mic06] Microsoft. Microsoft Office. http://office.microsoft.com/en-gb/default.aspx, 2006.

[MT79] Robert Morris and Ken Thompson. Password Security: A Case History. *Communications of the ACM*, 19(11):594–597, 1979.

[Nic65] R.S. Nickerson. Short term memory for complex meaningful visual confiurations. A demonstration of capacity. *Canadian Journal of Psychology*, 19:155–160, 1965.

[Nic68] R.S. Nickerson. A note on long-term recognition memory for pictorial material. *Psychonomic Science*, 11:58, 1968.

[oD85] Department of Defense. Trusted Computer System Evaluation Criteria - DoD 5200.28-STD, 1985.

[Org98] International Standards Organisation. ISO Standard 9241-11 - first edition, 1998.

[oST] National Institute of Standards and Technology. Computer Security Requirements : Guidance for applying the Department of Defense Trusted Computer System Evaluation Criteria in Specific Environments - CSC-STD-003-85.

[PBH96] John Podd, Julie Bunnell, and Ron Henderson. Cost-Effective Computer Security: Cognitive and Associative Passwords. In *Proceedings of the 6th Australian Conference on Computer-Human Interaction*, 1996.

[Por82] Sigmund N. Porter. A password extension for improved human factors. *Computers and Security*, 1(1):54–56, 1982.

[PPBH00] Rachael Pond, John Podd, Julie Bunnell, and Ron Henderson. Word Association Computer Passwords: The Effect of Formulation Techniques on Recall and Guessing Rates. *Computers and Security*, 19(7):645–656, 2000.

[Ren04] Karen Renaud. Quantifying the Quality of Web Authentication Mechanisms - A Usability Perspective. *Journal of Web Engineering*, 3(2):95–123, 2004.

[Ren06] Karen Renaud. Authentication – The Final Frontier. In *IST-Africa 2006 Conference*, 2006.

[Res02] Gartner Research. The Cost of a non-automated Help Desk, 2002.

[RU88] T. Raleigh and R. Underwood. CRACK: A Distributed Password Advisor. In *USENIX UNIX Security Workshop Proceedings*, 1988.

[SBW01] M A Sasse, S Brostoff, and D Weirich. Transforming the 'weakest link' – a human/computer interaction approach to usable and effective security. *BT Technology Journal*, 19:122–131, 2001.

[Sha48] C.E. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27:379–423 and 623–656, 1948.

[She67] R.N. Shepard. Recognition memory for words, sentences and pictures. *Journal of Verbal Learning & Verbal Behaviour*, 6:156–163, 1967.

[Sli05] Frederik Slijkerman. Ultra Fractal 4.02 (Standard Edition). http://www.ultrafractal.com/, 2005.

[SM86] Sidney L. Smith and Jane N. Mosier. Guidelines for Designing User Interface Software. Technical report, The MITRE Corporation, 1986.

[Smi87] Sidney L. Smith. Authenticating users by word associations. *Computers and Security*, 6(6):464–470, 1987.

[Smi02] Richard E. Smith. *Authentication: From Passwords to Public Keys*. Addison Wesley, 2002.

[S.P05] S.P.A.R.C. The Rorschach Test. http://www.deltabravo.net/custody/rorschach.php, 2005.

[Spa92] E.H. Spafford. Opus: Preventing Weak Password Choices. *Computers and Security*, 11(3):273–278, 1992.

[Spa06] Libor Spacek. Computer Vision Research Face Database. http://cswww.essex.ac.uk/mv/allfaces/index.html, 2006.

[SS04] Adam Subblefield and Dam Simon. Inkblot authentication. Technical report, Microsoft Research, 2004.

[SW81] Temple F. Smith and Michael S. Waterman. Identification of Common Molecular Subsequences. *Journal of Molecular Biology*, 147:195–197, 1981.

[vE85] Wim van Eck. Electromagnetic radiation from video display units: an eavesdropping risk? *Computers and Security*, 4(4):269–286, 1985.

[WB58] Michael Wertheimer and David Beardslee. Principles of perceptual organization. *Readings in Perception*, pages 115–135, 1958.

[WWB$^+$05] Susan Wiedenbeck, Jim Waters, Jean-Camille Birget, Alex Brodskiy, and Nasir Memon. Authentication Using Graphical Passwords: Effects of Tolerance and Image Choice. In *Proceedings of the 2005 symposium on Usable Privacy and Security*, 2005.

[YBAG00] Jianxin Yan, Alan Blackwell, Ross Anderson, and Alasdair Grant. The Memorability and Security of Passwords - Some Empirical Results. Technical report, Cambridge University Computer Laboratory, 2000.

[ZH90] Moshe Zviran and William J. Haga. User Authentication by Cognitive Passwords: An Emperical Assessment. In *Proceedings of the 5th Jerusalen Conference on Information Technology. 'Next Decade in Information Technology'*, 1990.

[ZH93] Moshe Zviran and William J. Haga. A Comparison of Password Techniques for Multilevel Authentication Mechanisms. *The Computer Journal*, 36(3):227–237, 1993.