# Image Understanding for Automatic Human and Machine Separation

Cristina Romero Macias

Queen Mary, University of London

School of Electronic Engineering and Computer Science

Supervisor: Prof. Ebroul Izquierdo

PhD Thesis

# Abstract

The research presented in this thesis aims to extend the capabilities of human interaction proofs in order to improve security in web applications and services. The research focuses on developing a more robust and efficient Completely Automated Public Turing test to tell Computers and Human Apart (CAPTCHA) to increase the gap between human recognition and machine recognition. Two main novel approaches are presented, each one of them targeting a different area of human and machine recognition: a character recognition test, and an image recognition test. Along with the novel approaches, a categorisation for the available CAPTCHA methods is also introduced.

The character recognition CAPTCHA is based on the creation of depth perception by using shadows to represent characters. The characters are created by the imaginary shadows produced by a light source, using as a basis the gestalt principle that human beings can perceive whole forms instead of just a collection of simple lines and curves. This approach was developed in two stages: firstly, two dimensional characters, and secondly three-dimensional character models.

The image recognition CAPTCHA is based on the creation of cartoons out of faces. The faces used belong to people in the entertainment business, politicians, and sportsmen. The principal basis of this approach is that face perception is a cognitive process that humans perform easily and with a high rate of success. The process involves the use of face morphing techniques to distort the faces into cartoons, allowing the resulting image to be more robust against machine recognition.

Exhaustive tests on both approaches using OCR software, SIFT image recognition, and face recognition software show an improvement in human recognition rate, whilst preventing robots break through the tests.

# Acknowledgements

First of all, I would like to thank Prof. Ebroul Izquierdo for all his support and advice during my PhD degree that have served me as a guide during these years. Also, I would like to thank Dr Prathap Nair and Dr. Krishna Chandramouli for being there to provide advice and help that made progress in my study feasible.

It was a pleasure to work and share joyful moments with many brilliant colleagues in the MMV Group. They helped me not only professionally but also personally making everyday life easier and happier, for which I am extremely thankful.

I would also like to give a special thanks to my friends in London, those who helped to get here in the first place and those I met afterwards: without you, this whole experience would have been half-empty.

Last but not least, I must thank my family and my friends back home, for the unconditional support in the distance that never let me throw in the towel.

# Contents

# List of tables

# List of figures

# List of Abbreviations

| | |
|---|---|
| ASR | Automatic Speech Recognition |
| BN | Bayesian Network |
| CAPTCHA | Completely Automated Public Turing test to tell Computers and Human Apart |
| DoG | Difference of Gaussians |
| EBN | Embedded Bayesian Network |
| EGM | Elastic bunch graph matching |
| FERET | Facial Recognition Technology |
| FLD | Fisher's Linear Discriminant |
| HIP | Human Interactive Proof |
| ICA | Independent Component Analysis |
| LDA | Linear Discriminant Analysis |
| LEM | Line edge map |
| MDF | Most Discriminating Features |
| OCR | Optical Character Recognition |
| PCA | Principal Component Analysis |
| SIFT | Scale Invariant Features Transform |
| SNoW | Sparse Network of Winnows |
| SOM | Self-Organizing Map |
| SVM | Support Vector Machines |
| URL | Uniform Resource Locator |
| VTK | Visualization Toolkit |
| WWW | World Wide Web |

# Chapter 1

# Introduction

In the last two decades, since the commercialisation in the nineties of the Internet, the number of users has grown exponentially until reaching more than 2.2 billion people [77]. This is the result of its popularisation and incorporation into virtually every aspect of modern human life from daily affairs such as education, web search or goods shopping to more professional oriented tasks. Advances in the protocols and the services have brought a wide variety of services. The most important one is the World Wide Web (WWW) that communicates via the Internet a series of resources such as interconnected documents, linked by hyperlinks and URLs.

Since its creation, the Internet has no centralised governance in either policies for access and usage, technological implementation, or management, and it is maintained by each constituent network with its own standards. Due to this fact, security has become an important issue for the users, companies and services. One of the primary sources of abuse on the Internet is spam, that targets electronic messaging services by sending unsolicited bulk messages indiscriminately, especially advertising, among other actions such as instant messaging spam, web search engine spam, spam in blogs, in wikis, in ads, in forums and in social networks, mobile phone messaging spam, and file sharing network spam. It became a serious problem

when the internet was opened up to the general public in the mid-90s. The fact that people have quick and easy access to the internet network made this problem grow exponentially in the following years, reaching proportions of 85% and 90% of all the emails in the world [122].

Besides the huge expansion it has experienced, spam is also a serious problem because of the property rights and the consumed resources. First of all, spam is difficult to get rid off because property rights in several countries are difficult to enforce. Nowadays in Europe, there is a new legislation that tries to reduce the quantity of spam coming from the continent [59]. Secondly, if we talk about resources, spam consumes shared resources such as bandwidth or the load of the servers, or private resources such as money and time. Finally, another serious issue that derives from the existence of spam is that it has become a tool for malware authors and phishers to abuse the Internet.

Malware or malicious software is the term used for a diverse kind of hostile, intrusive, or annoying software that can be used to gather personal or private information, or to harm computer operations. The most common forms of malware are viruses, worms, trojan horses, spyware, adware, and other malicious programs [149]. On the other hand, phishing is a software used to acquire information such as usernames, passwords, and credit card details by disguising itself as a trustworthy entity in an electronic communication or transaction [159] with the aim of stealing money. Spam can be used by malware authors and phishing software through unsolicited commercial e-mails to spread harmful software with the objective of identity theft or even worse; fee fraud. These software programs take advantage of the victim's inexperience with technology or attempt to call on human greed for money (see Figure 1.1).

One of the most effective methods for reducing the amount of spam circulating on Internet and ensuring safety for users is the use of CAPTCHAs. A CAPTCHA is a program that protects internet companies and human users against spam or bots through the generation of grading tests that most humans can pass but current computers cannot [20].The term CAPTCHA stands for Completely Automated Turing Test to Tell Computers and Humans Apart and was

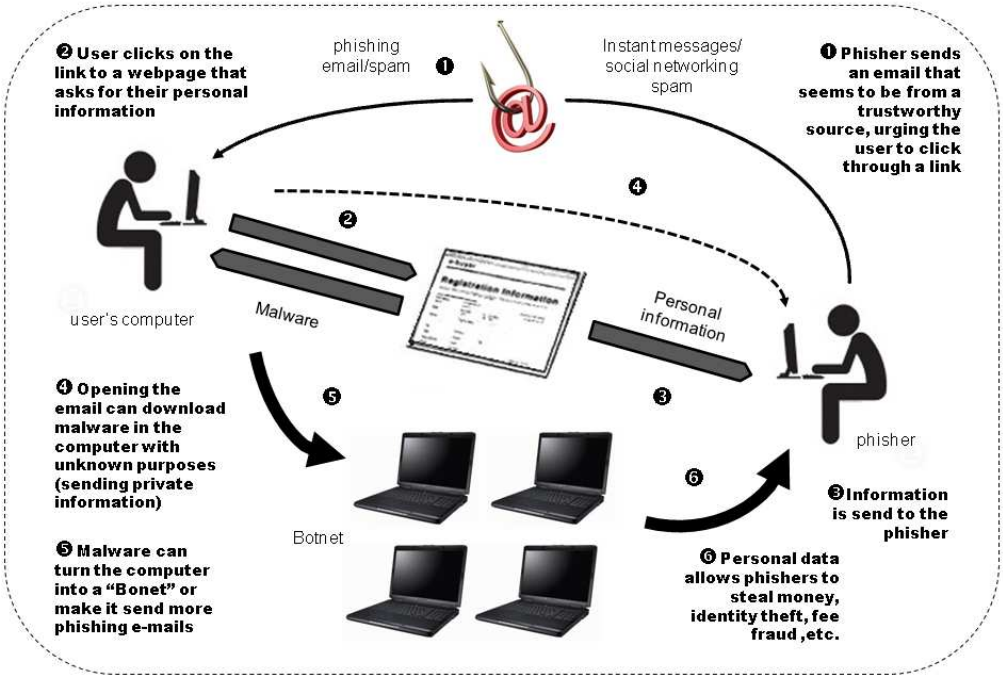


**Figure 1.1:** Flow diagram of a phishing and malware attack through spam.

firstly coined in 2000 by Luis von Ahn, Manuel Blum, Nicholas Hopper and John Langford of Carnegie Mellon University [188].

The primary application of CAPTCHA is to prevent malicious attacks to the systems by spammers. However, they also serve to protect vulnerable systems, such as Yahoo or Hotmail, against e-mail spam, automated posting to forums, blogs and wikis as a result of commercial interests or harassment. Another important function is bit rate limiting when excessive use of a service is observed.

Nowadays, most of the methods to discriminate humans from computers are based on optical character or image recognition, or sound recognition. In a word-based CAPTCHA, the characters are distorted to make its recognition more difficult for the bots. Among the basic distortions, it can use translation, rotation (clockwise or counterclockwise) and scaling, among others such as sight angle, lighting effects, context, and camouflage [38]. A word-based CAPTCHA test consists on an image that contains distorted and noisy characters or words. To

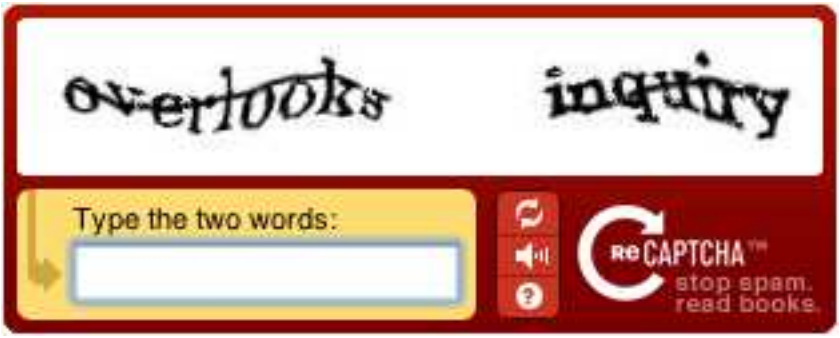


**Figure 1.2:** Word-based CAPTCHA extracted from http://www.captcha.net.

solve this test, the user has to type the characters presented in the image. Usually, the distortions applied to the image are complicated enough to prevent a robot to recognise the word while allowing humans to do so. An example of common CAPTCHA used in current web applications can be appreciated in Figure 1.2.

An image-based CAPTCHA contains primarily an image that the user has to recognise. Amongst these tests, the user can be asked to implement different kinds of actions; solve a quiz, match symbols, recognise faces, etc. Usually, the images do not appear straightforwardly, instead they can contain warping, occlusion or lighting effects to avoid being recognised by machines. The last type is a sound-based CAPTCHA, which was implemented in the first place for those users that cannot solve visual CAPTCHAs due to an impairment. The test presents an audio file that contains words, letters, or numbers, mixed with background noise, that the user has to type correctly.

Even thought there are many CAPTCHA methods available to prevent spam circulating, there are many researchers that have developed techniques to break through them [70, 130, 131, 161] since it means a technological advance in machine learning. Additionally, companies have exploited the fact that users find the tests annoying to create commercial DeCAPTCHAs to break the CAPTCHA tests automatically, without the direct intervention of the users. Due to these facts and the greed of spammers, most of the current tests are becoming obsolete.

In this thesis, the major motivation was the creation of advanced software tools that enables separation of humans and machines in an automated environment and increases the gap of what humans can recognise and machines cannot. The targeted strategies have exploited extremely difficult tasks related to image understanding and human perception. These objectives were established in order to prevent all the security breaches produced by spam and other forms of attack, which are also caused by the inexperience of using computer technology by the majority of users. The primary contributions of this thesis are the development of two efficient and robust CAPTCHA approaches and a categorisation for the current CAPTCHA tests.

## 1.1 Contributions

For the Visual-word based CAPTCHA:

- identification of the issues on the current word-based CAPTCHAs;
- development of a new type of characters based on 3D objects with 3D boundaries delimited by shadows [150];
- design of an efficient algorithm to optimise the distortions applied to the characters and ensure safety against possible external attacks to break the code [150];
- exhaustive experiments to test the efficiency of the approach and improve the human friendliness regarding the current approaches available [150].

For the Image-based CAPTCHA:

- identification of the issues on the current image-based CAPTCHAs;
- development of a database of faces of well known people and a second database with cartoons and animals to create a final image that is the result of the morphing between a selected image from each database [151];

- design of an efficient algorithm to optimise the morphing between images and ensure safety against possible external attacks to break the code [151];
- exhaustive experiments to test the robustness of the approach and improve the human friendliness regarding the current approaches available [151].

Finally, the categorisation gives a classification for every kind of test available and for future techniques since it goes from three general branches to a subclassification that can be enlarge if necessary.

## 1.2 Overview of the thesis

This thesis has been organised in a self-contained manner. The initial chapter presents the fundamental aspects of the addressed technology and the corresponding state of the art, the following three chapters present the techniques used to develop the approaches presented in the thesis. The subsequent two chapters present the proposed approaches, fully explaining the algorithms and the results obtained. The last chapter concludes the work, presenting the conclusions and considerations for future research. This thesis is organised as follows:

Following the introductory chapter, Chapter 2 presents an overview of CAPTCHA methods, as well as a survey of the available CAPTCHA tests. Important evaluation concepts, such as efficiency and robustness, and human friendliness, are explained, as they will be important in the later chapters. Also, several well-known commercial and published CAPTCHAs are presented along with one of the contributions of the thesis; a categorisation of the CAPTCHAs.

Chapter 3 summarises the basic concepts in digital image manipulation used to create visual CAPTCHA tests. Firstly, the digital image warping and morphing tools are presented, which are used to create the pertinent distortions for both approaches. Additionally, a 3D computer graphics study is introduced, since it will play a major role in the development of the new con-

cepts that differentiate the new CAPTCHA tests presented in this thesis with the ones currently available.

In Chapter 4 the digital image recognition tools are presented. These tools are used to evaluate the efficiency and robustness of the approaches created. For the OCR-based CAPTCHA, the SIFT tool is explained, since it will be used to evaluate the grade of machine recognition for characters. It also presents the state-of-the-art study in face recognition techniques, because different techniques will be used to measure the capacity of machines to recognise the distorted faces created by the image-based CAPTCHA.

Human perception and recognition theories are the focus of Chapter 5. The main aim of this chapter is the evaluation of the human friendliness of the approaches presented in this thesis. Human perception theories are explained in the two sections that the chapter is divided. The first section focuses on Gestalt psychology, which defines a branch of psychology than explains how human beings perceive objects when they are incomplete, which is used to create the OCR-based CAPTCHA. The second section focuses on face perception and recognition with the aim of creating a good interactive image-based CAPTCHA.

Chapter 6 introduces the first approach: the visual word-based CAPTCHA. The developed scheme introduces a new concept in the creation of a word-based CAPTCHA: the use of shadows to represent characters. Additionally, it presents both the experiments made to evaluate the efficiency and robustness, and the human friendliness and the results for these experiments, along with a complexity analysis of the test and a brief discussion of these results.

Chapter 7 focuses on the second approach, the image-based CAPTCHA. This scheme is developed with the aim of creating a more interactive and secure test. It uses distorted faces of well known people from diverse cultural sectors, such as politics, sports, cinematographic industry, etc. Following the lead of the first approach, it also presents both the experiments made to evaluate the efficiency and robustness, and the human friendliness and the results for

these experiments, along with a complexity analysis of the test and a brief discussion of these results.

The conclusions are summarised in Chapter 8. The list of author's publications is given at the end of the thesis along with the references used.

# Chapter 2

# CAPTCHA concepts

Internet security has been an important issue since its advent in the 80s. Its rapid evolution and penetration into all business sectors and aspects of life such as education, web search, goods shopping or more professional oriented tasks, has led to an exponential growth in security threats and breaches. Most websites that carry commercial or administrative applications require filling forms to allow people to use the services. Regrettably, some users abuse these services by creating programs, called spam, to register automatically and use them for undesirable purposes. Spam involves sending unsolicited commercial email messages, some with the aim of identity theft or fee fraud. Besides, spam can be used to attack personal computers through viruses, Trojan horses or malicious software [11].

Throughout the thesis, different methods to prevent spam, based on the CAPTCHA methodology, are presented. For this reason, this chapter provides a a state-of-the-art background on CAPTCHA concepts, detailing the basic tools and techniques used to developed the methods. In addition, one of the main novelties of the thesis is introduced: a categorisation of the available CAPTCHAs that allows their classification depending on their characteristics, difficulty and friendliness. The concepts introduced in this chapter can be found in [4].

## 2.1 CAPTCHAs

The concern of security in gaining access to a service over the internet has become a topical issue. To prevent such attacks, diverse systems have been presented recently. These systems are called **HIP** (Human Interactive Proof) and their main objective is to distinguish between various groups of users through a challenge/response protocol, e.g., human versus a machine, one person versus anyone else, etc [11]. The commercial uses of HIPs exploit the gap in ability between human and machine vision systems in reading images that can contain text, faces or symbols. The idea behind these tests comes from a methodology proposed by Alan Turing [144], which tests the intelligence of a computer through an ”imitation game”. In this test, a human judge asks questions to a human person and a computer which are situated in different rooms. If the interrogator cannot determine which room the computer is in and which one the human, the computer has passed the Turing test.

**CAPTCHA** (Completely Automated Public Turing test to tell Computers and Human Apart) is the most expanded branch of HIP systems. A CAPTCHA is a software that generates grading tests that most humans can pass but computers cannot. Its origins reside in 1997 when *Altavista* developed a filter that generated images of printed random characters to avoid automatic submission of URLs to their search engine. Later on Blum et al., created the CAPTCHA project which was developed at Carnegie Mellon University [20]. They articulated the most desirable properties a CAPTCHA test should have [188]:

- the test’s challenges should be automatically generated and graded (the judge is a machine)
- the test should be taken quickly and easily by human users
- the test should accept virtually all human users and will reject virtually all machine users
- the test should resist automatic attack for many years in spite of technology advances or open test algorithms

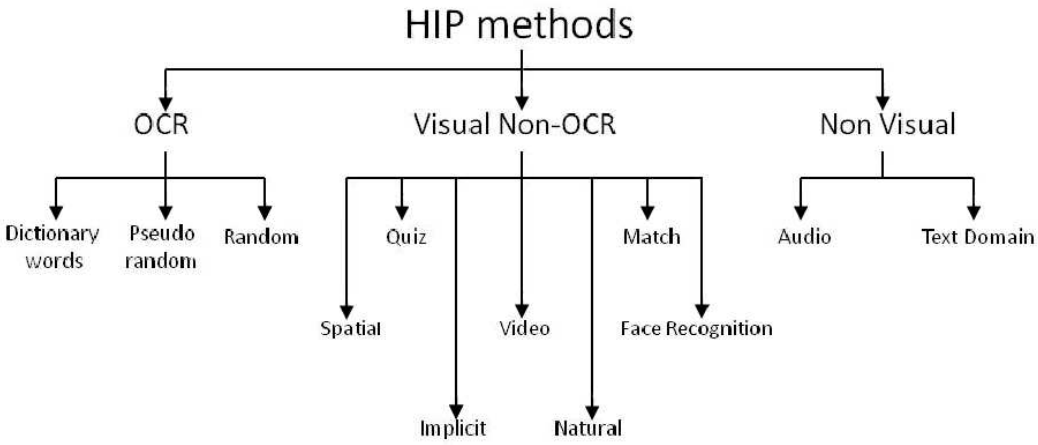


**Figure 2.1:** CAPTCHA methods can be classified as: (left) OCR-based, (middle) Visual Non-OCR based and (right) Non visual, with the subclassifications shown in the figure.

CAPTCHA tests were designed to prevent fake registrations by computer programs in websites [11], but the number of applications has increased since then. Nowadays they are used to prevent email from worms and spam such as [92], [175]. In addition to the email spam problem, CAPTCHA tests are used to prevent fraud in online polls [188], search engine bots reading web pages [5], bots playing online games [71], [204] and dictionary attacks [33]. They are also used for detecting phising attacks [35], [160] or user authentication [64], [176], [171].

In the last decade, different type of methods have been developed to produce CAPTCHA tests. In this thesis, we propose a detailed classification that starts with three main branches and divides into sub-categories (see Figure 2.1). The three main groups are: OCR-Based, Visual Non-OCR-Based and Non Visual. OCR stands for Optical Character Recognition and it is an artificial intelligence program that is used for automatically reading scanned images of handwritten, typewritten or printed text. Normally, they are calibrated to recognise some specific character fonts and have difficulties when the image has low resolution. The recognition rate drastically drops at recognising cursive text, with recognition rates even lower than those of hand-printed text. The disadvantages of the OCR systems can be used as an advantage if applied to CAPTCHAs, so only human beings can recognise the text [39].

| Methods | Difficulty | Friendliness | Popularity | Uses Human Psychology |
|---|---|---|---|---|
| OCR | High/Average | Low | Low | High/Average |
| Visual Non-OCR | Average | High/Average | High | Average |
| Non Visual | High | Average/Low | Low | Average |

**Table 2.1:** Categorisation of the CAPTCHA methods according to: (a) Difficulty, (b) Friendliness, (c) Popularity, and (d) Human Psychology

The first type, the OCR-Based CAPTCHAs, is based on creating an image containing a word or a set of random characters which the user has to recognise and type. As a rule, the characters appear distorted and with different image effects. Due to these image manipulations OCR systems fail when recognising, allowing the CAPTCHA to protect the web application. To increase the level of robustness of these tests, sometimes the distortions and effects applied make it difficult for the user to recognise the characters and consequently they fail the test. As a result, the users find these CAPTCHAs annoying and time-consuming.

The second type, the Visual Non OCR-Based CAPTCHAs, is based on diverse sets of images that do not imply character recognition. These tests range from recognising faces or objects, to trivia or math questions, etc. They are usually more entertaining for the user and faster than the OCR-Based ones, but at the same time the distortions and effects applied should be more restricted, since the user's knowledge about the content in the image can be quite limited.

The third and final type, the Non Visual CAPTCHAs, is based on audio or semantic tests. For an audio test, the program chooses a word or a random sequence of numbers, renders them into a sound clip and applies a specific distortion. To pass the test, the user has to type its contents. For the semantic test, the user has to extract the content of what they are reading or seeing. The robustness of these programs relies on the difference in skills to recognise spoken or semantic language between humans and machines. The audio tests are an alternative for those users who are visually impaired, however they require the presence of speakers or earphones, which can be an obstacle depending on the situation.

Regardless of the type of method used in the CAPTCHAs, they share common characteristics that define them; First, the generation of the tests should be completely automated by a machine. Only human intervention should be required to pass the test. Second, the code, the data, and the algorithm should ideally be public since CAPTCHAs benefit from peer review, which is normally successful at identifying weaknesses [131]. Finally, a robust CAPTCHA should rely on a completely random generation system for choosing the corresponding characters, images or other files. The solutions should not be contained in databases because they could be cracked. Also, the machine generating the tests should not be able to solve them. The aim is to create a CAPTCHA that is immune to imminent attacks.

Notwithstanding, creating programs to break through CAPTCHA tests has become an important area in research, since it would mean a significant advance in machine learning. Also, the necessity in designing CAPTCHAs which are robust, secure and usable has become a priority due to the inefficiency of some methods to resist attacks [131, 205, 206]. Breaking a CAPTCHA involves developing an automated program to solve a CAPTCHA challenge. For OCR-based methods this would consist of a three-stage approach consisting of preprocessing, segmentation and classification stages recognition [102].

One issue to take into account when developing a CAPTCHA method is its usability, which is a measure of the effectiveness, efficiency and satisfaction with which specified users can achieve specified goals in a particular environment [23]. The usability of a CAPTCHA test is determined by the accuracy, response time and perceived difficulty of the user. To make a CAPTCHA desirable for the users it needs to have high accuracy, low response time and low perceived difficulty [182] (see Table 2.1)

Another important matter when developing a CAPTCHA is the so called human friendliness that refers to how easy the test can be solved from the point of view of the human users. For this, CAPTCHA systems exploit the findings of cognitive psychology. Cognitive psychology is a field of psychology that explores internal mental processes. It deals with how humans think, remember, perceive, make decisions, and solve problems. These include: pattern and object

recognition, semantic memory, mental imagery, grammar and phonetics, language acquisition, logic and problem solving. Studying how the brain works can have a positive effect in the creation of new HIP methods that rely on the strong points of humans. One example is the use of Gestalt psychology that states that humans experience things that are not part of our simple sensations. What we see in certain occasions is believed to have an effect on the whole event, that is not contained in the sum of the parts. This can happen when completing objects or words when they are not finished or when imagining objects close together as a whole [101].

## 2.2 OCR-Based CAPTCHA Methods

CAPTCHA tests based on letters and number recognition are the most widespread of HIP methods. The process involves characters rendered into an image and distorted before presenting them to the user.To build a reading CAPTCHA test, there are several choices to take into account that can affect the complexity and user-friendliness of the CAPTCHA [9]:

1. Character data set: Numbers and letters selected to use in the tests.
2. Affine transformations: Translation, rotation, shearing and scaling applied to the characters.
3. Perspective transformations and image warping: elastic transformations of the image - global warping (a character) or local warping (at pixel level).
4. Adversarial clutter: Random lines, dots, or geometric shapes that intersect with the characters.
5. Background and foreground textures: Textures are used to generate a coloured image from bi-level or gray-scale masks generated using the preceding steps.

6. Language: The language set used determines the conditional and joint probabilities of character occurrence and recognition. These tests can use: random characters, words from a dictionary or a phonetic generator.

To pass the test, the user has to identify all the characters in the correct order and type them. The following reasons are the basis for the expansion and acceptance of these tests [9]:

- Optical character recognition is a well researched field and has been extensively developed in the last two decades.
- Characters were created by humans and learnt since childhood.
- Each character has a corresponding key in the keyboard and a corresponding ASCII code. A word of 8 characters can have over a 1000 billions permutations of characters.
- Localization and recognition issues are minimal when using western characters and numbers.
- OCR-Based tests can be quickly generated.

The remaining of this section discusses the characteristics and the categorisation of the available OCR-methods, (see Figure 2.1). As a general reference guide, Table 2.2 shows a summary of the following methods and in Figure 2.2 there are some samples of CAPTCHAs.

**Dictionary word based methods:** The images contain words extracted from specific language dictionary. The amount of existing words is limited so the amount of solutions is very narrow. They use different sets of distortions and rotations.The most prominent works on these methods can be found in [20], [44], [115], and [156].

**Pseudo-random word based methods:** The images contain words that make sense phonetically but not grammatically. The number of results they can produce is increased but is re-

| Classification | Name | Author | Summary | Input to solve the test |
|---|---|---|---|---|
| Dictionary word | GIMPY | Blum et al., 2000 [20] | Seven dictionary words with distortions, occlusion and cluttering | Three words typed correctly |
| | EZ-GIMPY | Blum et al., 2000 [20] | One dictionary word with distortions and cluttering | One word typed correctly |
| | Pessimal Print | Coates et al., 2001 [44] | English dictionary words combined with a typeface and a set of image-degration parameters | One word type correctly |
| | Dynamic visual patterns | Liao and Chang 2004 [115] | Images containing information-embedding visual patterns of words, using foregrounds and backgrounds of dots | One word typed correctly |
| | Handwritten CAPTCHA | Rusu and Govindaraju 2007 [156] | Repository of handwritten words not recognised by OCR programs | One word typed correctly |
| Pseudo-random word | BaffleText | Chew and Baird 2003 [41] | Non-English but pronounceable word with mask degradation | One word typed correctly |
| | ScatterType | Baird and Riopka 2005 [10] | Non-English but pronounceable word with characters fragmented with horizontal and vertical cuts. The fragments are scattered by horizontal and vertical displacements | One word typed correctly |
| | reCAPTCHA | Von Ahn et al., 2008 [20] | One English dictionary word and one scanned word from a book non recognised by OCR programs | The dictionary word is the control word, so if the user types it correctly, they pass the test |
| Random characters | TGC CAPTCHA | Dailey and Namprempre 2004 [50, 134] | Sequence of k distorted characters, one at a time, with basic distortions and row sliding | *k* correct responses |
| | Kanizsa CAPTCHA | Saalo 2010 [158] | Random background overlaid with white text | One word typed correctly |
| | 3D Visual word-based CAPTCHA | Romero Macias and Izquierdo 2009 [150] | Shadows created by 3D characters, with distortions applied after rendering the models | One word typed correctly |

**Table 2.2:** OCR-Based methods

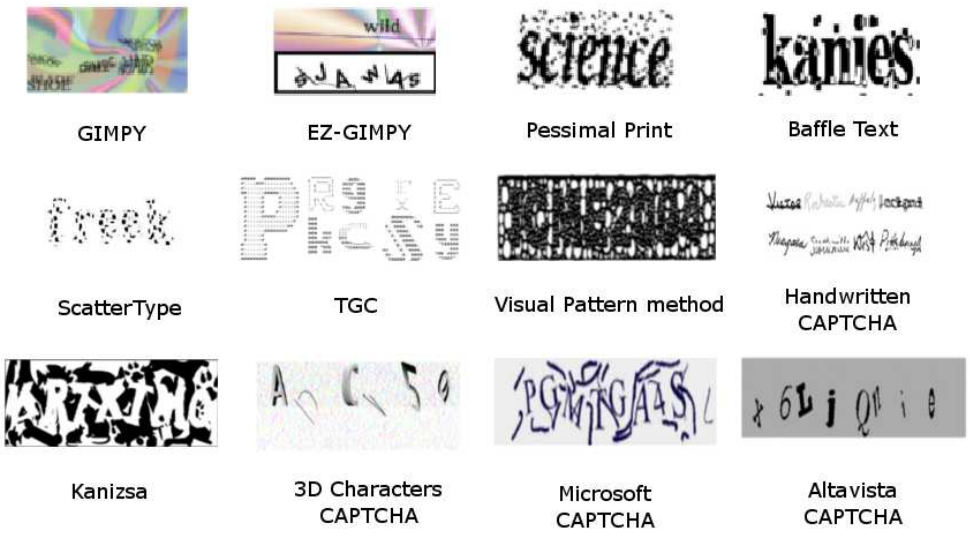


**Figure 2.2:** Samples of CAPTCHAs generated by OCR-based methods: First row: dictionary words methods; Second row: pseudo random; Third row: random methods.

stricted to the diversity of syllables of the specific language. They also use different sets of distortions and rotations.The most prominent works on these methods can be found in [41], [10], and [20].

**Random characters based methods:** The images contain words with random letters and numbers. The number of results they can produce is increased exponentially. Their efficiency resides in the random function to determine the characters. They also use different sets of distortions and rotations. The most prominent works on these methods can be found in [50], [134], [158], and [150].

### 2.2.1 Reliability of Visual-OCR methods

One of the two most important factors when developing a new CAPTCHA method is the robustness against machine attacks. The strength of a method is based on the accumulative effects applied on it. Also, the choices made when developing the effects can increase the difficulty

both for the machines and the human users to pass the test, so creating a balance is necessary [31].

When it comes to OCR-based CAPTCHAs, the larger the character set and the longer the word the stronger the test is. Using words from a dictionary can make the test easier to break. In the absence of a language model, the strength of the CAPTCHA improves exponentially with the length of the CAPTCHA and polynomially with the character set size. Distortions can also increase the security of the CAPTCHA but not dramatically. Background and foreground textures usually bring a minimal improvement in security [39].

To test the robustness of the different approaches the best way is to build automated programs to break CAPTCHAs and assess their success in solving particular tests [73]. There are few attacks on the GIMPY and EZ-GIMPY methods; Mori and Malik [130, 131] have successfully broken the EZ-GIMPY (92% success) and GIMPY (33% success) methods. Thayananthan et al [183] have also been successful at breaking EZ-GIMPY. Recently, Moy et al [133]have broken the purely distortion based HIP GIMPY-r with a success rate of 78%. Also the authors of [40, 73] have proved to break six particular methods: EZ-GIMPY/Yahoo, Yahoo v2, Mailblocks, Register, Ticketmaster, and Google. A study made by Kumar Chellapilla's group (http://research.microsoft.com/ kumarc/) at Microsoft Research focused on CAPTCHA letters with distortion and noise and found that a neural network could recognise a single character much easier than a human could.

The same attack as the one considered in [131] was used by Chew and Baird to defeat Pessimal Print and Baffletext [41] with a success rate of 40% and 11%, respectively. Also, if we consider that Pessimal Print uses 70 possible words, there is a 1/70 chance of a machine guessing it right. Recently, Jonathan Wilkins published an study on the reCAPTCHA program stating a failure rate of 17.5% [13, 196].

The second most important factor in the reliability of a CAPTCHA method is the human friendliness. It encompasses two important aspects: (a) the visual appeal and the annoyance of

the method and (b) how good the algorithm separates both human users and machines [9]. For OCR-based methods, the friendliness factor depends mostly in the design of the algorithm. In this case, it is related to the length of the word, if the characters form a dictionary word, and if there is a phonetic generator to make the recognition task easier. Normally, eight letters are used to create the test. Most tests also use different kinds of background and textures that can be quite intrusive at recognising.

The authors in [39] presented three studies on human users to test the friendliness of diverse available OCR-based methods. The first study was about human accuracy under rotation, scaling, local warping, and global warping separately. The second and the third studies were about human accuracy in the presence of background and foreground arc clutter. The results obtained in the first study showed that users were correct at 99% or higher with plain, translated, rotated or scaled characters. For global warping, local warping and a combination of local warping and other distortions, the accuracy significantly decreases. The results obtained in the other studies showed that adding clutter do not affect human accuracy whilst machines are poor at it, which can be used to design segmentation-based methods.

## 2.3 Visual Non OCR-Based CAPTCHA Methods

The second type of CAPTCHAs use other visual methods that do not rely on recognising characters. These tests rely on the visual capacity of the users to identify different kinds of objects, faces or images. They have become more popular recently due to their simplicity and readiness.

Image-based techniques involve the use of diverse concepts or patterns which the human user needs to identify correctly. The size and dimensions of each generated CAPTCHA image, the dimensions of the images in the databases, and the level of difficulty may vary considerably from one CAPTCHA to another. The images used in the creation of the test can come from diverse sources such as a specific server or Internet. Also, different sets of distortions and

noise can be added to the final image to increase the level of difficulty. The aim is to create a composite CAPTCHA image and present it to the user in which the solution may be a click, a text string, a rotation, etc [182].

The remaining of this section discusses the various available Visual non OCR-methods, describing their characteristics and the categorisation shown in Figure 2.1.

**Quiz CAPTCHAs:** Quiz CAPTCHAs are tests based on puzzles, quizzes, or a trivia question. The premise of these programs is the assumption of a common base of factual knowledge that most humans already know and most computers do not know and cannot learn. The most prominent works on these methods can be found in [20], [189], [42], [172], [103], [75], [132], [146], [124] and [203]. As a general reference guide, Table 2.3 shows a summary of the following methods and in Figure 2.3 there are some samples of the CAPTCHAs.

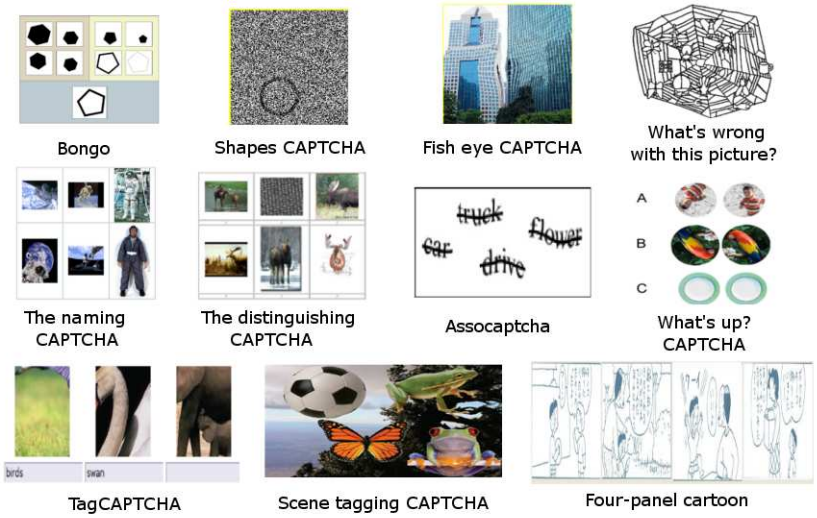


**Figure 2.3:** Samples of Visual non OCR-based methods based on quiz CAPTCHAs.

**Match CAPTCHAs:** Match CAPTCHAs are tests based on selecting an option from the ones available. The choices can come from a list or from hash visualisation. These test are also called image labelling CAPTCHAs. The most prominent works on these methods can be found in [20], [54], [57], [51], [114], [191], [163], [168], [121], [170], [165], [53], [55], [218],

and [95]. As a general reference guide, Table 2.4 shows a summary of the following methods and in Figure 2.4 there are some samples of CAPTCHAs.


**Figure 2.4:** Samples of Non OCR-based methods based on match CAPTCHAs.

**Spatial CAPTCHAs:** Spatial CAPTCHAs are based on 3D models or rendered animations. They might present a heavy burden because the servers on the web are not typically equipped with powerful graphics cards. The most prominent works on these methods can be found in [82], [91], [152], [180], and [37]. As a general reference guide, Table 2.5 shows a summary of the following methods and in Figure 2.5 there are some samples of CAPTCHAs.

**Implicit CAPTCHAs:** They were proposed in [8] and [167]. Their main purpose is to make CAPTCHAs less disrespectful to human users and more effective against machine attacks. Implicit CAPTCHAs are shown as normal links, where the user has to click only once. They are based on the user's experience of the context of a website. As a general reference guide, Table 2.6 shows a summary of the following methods and in Figure 2.6 there are some samples of CAPTCHAs.

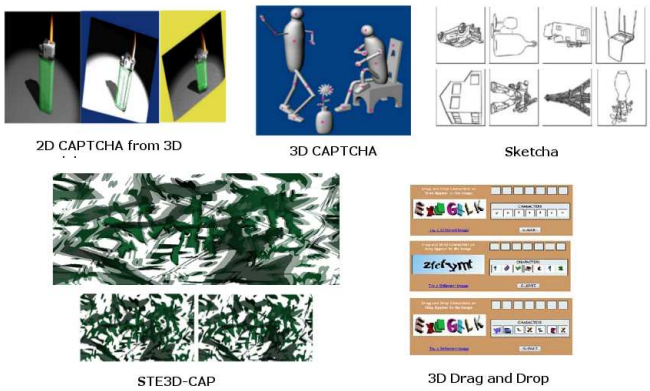


**Figure 2.5:** Samples of Visual non OCR-based methods based on spatial CAPTCHAs.


**Figure 2.6:** Samples of Visual non OCR-based methods based on implicit CAPTCHAs.

**Face Recognition CAPTCHAs:** There are some CAPTCHA tests available based upon face recognition or face detection and the most prominent works on these methods can be found in [155], [127], [151], and [61]. As a general reference guide, Table 2.7 shows a summary of the following methods and in Figure 2.7 there are some samples of CAPTCHAs.

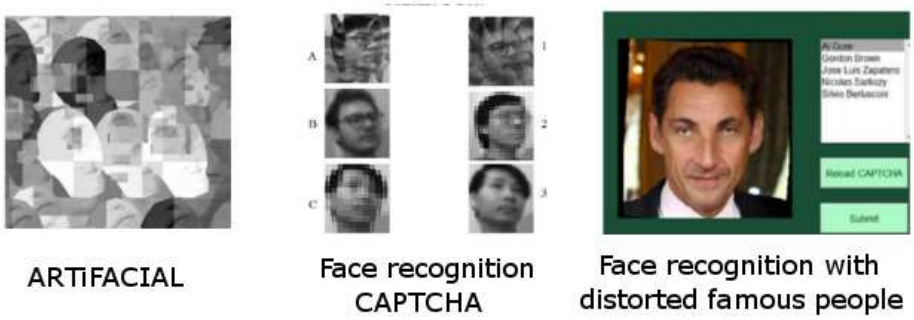


**Figure 2.7:** Samples of Visual non OCR methods based on face recognition CAPTCHAs.

**Video CAPTCHAs:** Video CAPTCHAs are methods in which the tests are presented as videos to the users. To pass the test, it is first necessary to watch the video and then input what is asked. The videos used can come from different sources, e.g. Youtube, Flickr, or, on the other hand, they can be produced in real time, to avoid malicious attacks through a database. Works with video CAPTCHAs can be found in [173], [98], [86], and [48]. As a general reference guide, Table 2.8 shows a summary of the following methods and in Figure 2.8 there are some samples of CAPTCHAs.


**Figure 2.8:** Samples of Visual non OCR-based methods based on video CATPCHAs.

**Natural CAPTCHAs:** They were developed as an alternative to the ”synthetic CAPTCHAs” generated by machines. Natural CAPTCHAs present scans or photos of real documents. This method was proposed by [118] in an effort to strengthen the robustness of CAPTCHAs. The idea is that with a large enough external supply of tests and graders, the machine does not need to generate and grade the test for it to be practical. Graders can be gathered in a process called ”Collaborative Filtering” attributed to [43]. In this process, the subjects are asked to solve more than one CAPTCHA. While the computer knows the answer for the first CAPTCHA and uses it to validate the user, the rest of the answers are used to discover the answers of the remaining CAPTCHAs.

| Name | Author | Summary | Input to solve the test |
|---|---|---|---|
| Bongo | Blum et al., 2002 [20] | Puzzle test based on the Mensa tests that displays two series of blocks with different characteristics | Determining which series belong to each answer |
| Shapes CAPTCHA | Wagner 2005 [189] | Five shapes distorted and randomly chosen | Recognising a specific shape |
| Fisheye CAPTCHA | Wagner 2005 [189] | Picture with a fisheye distortion in it | Clicking in the centre of the distortion |
| What's Wrong With this Picture? CAPTCHA | Wagner 2005 [189] | Picture of a scene with different objects | Clicking on the objects that do not belong to the scene |
| The naming CAPTCHA | Chew and Tygar 2004 [42] | Six images with a common term | Typing the correct term |
| The distinguishing CAPTCHA | Chew and Tygar 2004 [42] | Two sets of images with three images each with equal probability of having a common subject in both sets | Distinguishing if the common subject is there or not |
| The anomaly CAPTCHA | Chew and Tygar 2004 [42] | Five images of a common subject and one different | Recognising the image that does not belong in the set |
| Question based CAPTCHA | Shirali-Shahreza 2007 [172] | Mathematical question presented as an image | Multiple choice answer |
| Assocaptcha | Kulkarni 2008 [103] | Three words with a common subject and one word unrelated, distorted and cluttered | Distinguishing the unrelated word |
| What's Up CAPTCHA? | Gossweiler 2009 [75] | Image containing an object with a different orientation | Selecting the orientation |
| TagCAPTCHA | Morrison et al., 2009 [132] | Set of images | Describing the images with an English free-text word |
| SPC CAPTCHA | Jain et al., 2009 [146] | Set of images with or without a tag | Finding the correct order sequence of the images |
| Scene tagging CAPTCHA | Matthews and Zou 2010 [124] | Image with multiple individually distorted objects | Recognising the relationship between the objects |
| Four-Panel Cartoon CAPTCHA | Yamamoto et al., 2010 [203] | Four different panels with a common content | Identifying the correct order in the sequence |

**Table 2.3:** Visual non OCR-based methods based on quiz CAPTCHAs.

| Name | Author | Summary | Input to solve the test |
|---|---|---|---|
| PIX | Blum et al., 2002 [20] | Four to six random images of an object | Choosing the keyword that describes the objects |
| ESP-PIX | Blum et al., 2004 [20] | Four different images | Choosing the keyword that relates all the objects |
| SQ-PIX | Blum et al., 2009 [20] | Three different random images | Tracing the required image |
| Déjà Vu | Dhamija and Perrig 2000 [54] | Set of images including the ones previously selected by the user and some decoy images | Recognising the ones he selected |
| Asirra | Douceur et al., 2007 [57] | Set of 12 images of cats and dogs | Identifying the cats |
| IMAGINATION | Datta et al., 2005 [51] | Collage of different distorted images | Choosing one image and annotate it from a word list |
| CAPTCHA using exchanging blocks | Liao 2006 [114] | Image presented with regions of it exchanged | Recognising the exchanged blocks |
| KittenAuth | Warner 2006 [191] | Set of nine images of little animals | Identifying three kittens |
| HumanAuth | Schmalfeldt and Kramlich 2007 [163] | Set of images with natural and non-natural source | Identifying the ones with a natural source |
| The Drawing Captcha | Shirali-Shahreza 2006 [168] | Set of dots filling a background, and some of them with different shapes | Connecting the dots that are different |
| The Collage Captcha | Shirali-Shahreza 2007 [170] | Set of different pictures distorted and cluttered | Identifying the image requested by the program |
| SemCAPTCHA | Lupkowski and Urbanski 2008 [121] | An image with three words distorted, two correlated semantically | Clicking in the non-correlated word |
| Drag and Drop | Desai and Patadia 2008 [165] | A composite image with different objects | Identifying two objects and drag and drop them in the required place |
| DnD | Desai and Patadia 2009 [53] | Similar to drag and drop, a composite image with blocks of characters | Identifying the required blocks and drag and drop them |
| JACI | Doyle 2009 [55] | Set of different images correlated by content | Matching the images based on the content |
| Cortcha | Zhu et al., 2010 [218] | A collage of different objects | Identifying a computer-segmented object, cropped and detached from its original image |
| Image Flip CAPTCHA | ISeCure 2010 [95] | Image with different objects, some of them flipped | Identifying the non-flipped objects |

**Table 2.4:** Visual non OCR-based methods based on match CAPTCHAs.

| Name | Author | Summary | Input to solve the test |
|---|---|---|---|
| 2D CAPTCHA from 3D models | [82] | Image with a 3D object rendered into 2D with distortions and lightning effects | Labelling the object choosing a keyword from a list |
| 3-D CAPTCHA | [91] | Image of 3D objects with character labels | Identifying the labels required by the system |
| Sketcha | [152] | Image of 3D line drawings with upright orientation | Turning the drawings to the right orientation |
| STE3D-CAP | [180] | Images created by a stereo pair of cameras | Recognising the image and own the appropriate hardware |
| 3D Drag-n-Drop CAPTCHA | [37] | 3D characters | To drag and drop the characters into the boxes |

**Table 2.5:** Visual non OCR-based methods based on spatial CAPTCHAs.

| Name | Author | Summary | Input to solve the test |
|---|---|---|---|
| Implicit CAPTCHA | Baird and Bentley 2005 [8] | Images with an specific context | Clicking a link related to the context |
| CAPTCHA for Children | Shirali-Shahreza 2008 [167] | Image with random objects downloaded from Internet and distorted | Answering a question made using Text-to-Speech system |

**Table 2.6:** Visual non OCR-based methods based on implicit CAPTCHAs.

| Name | Author | Summary | Input to solve the test |
|---|---|---|---|
| ARTiFACIAL | Rui and Liu 2004 [155] | Image with a distorted face embedded in a cluttered background | Clicking in six specific points of a face |
| Face recognition CAPTCHA | Misra and Gaj 2006 [127] | Two distorted faces of a human face | Matching two images as being of the same person |
| Face recognition captcha | Romero Macias and Izquierdo 2011 [151] | Image of a celebrity face with morphing | Recognising three faces in a row, selecting the answer from a list |
| Social CAPTCHA | Facebook Inc. 2010 [61] | Photos of the user's friends | Clicking in the friend's name |

**Table 2.7:** Visual non OCR-based methods based on face recognition CAPTCHAs.

| Name | Author | Summary | Input to solve the test |
|---|---|---|---|
| Motion CAPTCHA | Sajad Shirali-Shahreza 2008 [173], Mohammad Shirali-Shahreza 2008 | Video of a person's action | Describing the movement by selecting from a list |
| Content-based video labelling CAPTCHA | Kurt Alfred Kluever, Richard Zanibbi 2009 [98] | Videos downloaded from Youtube with tags written by the owner of the video | Writing three tags |
| NuCAPTCHA | NuCAPTCHA Inc. 2010 [86] | Video contains a sentence with words in different colours | Typing the word in red |
| Animation CAPTCHA | [48] | Video of a moving object with a complex texture background | Identifying the object |

**Table 2.8:** Visual non OCR-based methods based on video CAPTCHAs.

| Name | Author | Main idea | Input to solve the test |
|---|---|---|---|
| Natural CAPTCHA | Daniel Lopresti 2005 [118] | Scans or photos of real documents | Identifying the scene |
| Collaborative Filtering | Monica Chew and J. D. Tygar 2005 [43] | Images coming from an external supply of tests and graders | Answering more than one test, one answer is known, the others serve to answer other captchas |

**Table 2.9:** Visual non OCR-based methods based on natural CAPTCHAs.

### 2.3.1 Reliability and Usability of Non-OCR based CAPTCHA methods

Image CAPTCHAs were first developed to overcome the shortcomings of OCR-based methods using image recognition or image classification. The main advantage of these techniques is the improved human friendliness and in consequence, a higher success rate compared to OCR-based CAPTCHAs. On the other hand, the main issue is that they require large databases of preclassified images. Furthermore, the databases need to be large and updated frequently with new images to avoid malicious attacks. Each type of image CAPTCHA has certain advantages and disadvantages which could be useful in deciding which test is better for a specific web service.

Quiz CAPTCHAs are hardly easy enough for humans to solve reliably or strong enough to stop machines to break through them. If the test presents a trivia question, it is being assumed a common base of factual knowledge that most humans already know but computers do not. This could manifest a problem for those users who do not have the base knowledge. Another issue is that the strength of these tests can often be compromised by how often a random guess can be right, e.g., the Bongo method can be broken 50% of the time.

Match CAPTCHAs are quite human friendly but normally require huge store space for tagged media files and the users have too many choices to answer. If the media database is too small and the images can be differentiated easily, the CAPTCHA can be easily broken. Some possible solutions will be the use of distorted and scrambled letters of an unique solution to avoid long lists. Asirra's method has been beaten by [70] with a probability of 82.7% of distinguish the cats and is able to solve the challenge in 12.2% of the time.

Spatial CAPTCHAs are theoretically the strongest ones to resist machine attacks due to the difficulty in recognising 3D text, but the amount of resources required to render the images could be too high. Also, web servers are not usually equipped with powerful graphics cards, so the process of creating an image out of 3D objects could be impractical.

Face recognition CAPTCHAs are in terms of ease the most friendly ones due to the fact that human faces are the most recognisable object to humans, regardless of culture and social background and users can recognise them even if they appear distorted, occluded or with lighting changes. On the other hand, face recognition is a very wide and well studied field, and machines can be trained to recognise facial features as well as faces, so with sufficiently large databases, computers might be able to achieve a certain percentage of success.

Video CAPTCHAs presents an innovative alternative to images or text. The reliability of these methods depends on the encryption of the video as well as the content. If the video contains characters to be recognised, the text inside it should not be accessible by any means. Also, tagging a video could manifest a problem, since it requires knowledge that users may not have.

Natural CAPTCHAs represent a more human-friendly version of image methods. In this way, the content can be recognised easily but the databases used should be big in comparison with others, since the images can be found on the internet.

## 2.4 Non Visual CAPTCHA Methods

In the last category, the most prominent methods are based on audio or semantics. Generally, CAPTCHA audio algorithms render some numbers, letters or words into a sound file, and then distort it and present it to the user. They also have been developed as an alternative for disabled people or people with special needs [169, 174, 175].

**Sound based methods:** In the first methods created [34, 99], a word is read using Text-to-Speech technology and the user needs to type the word. Google audio CAPTCHA consists of one human speaker saying random digits in the range of 0-9. Similar works are the Diggs CAPTCHA, Google's audio CAPTCHA and the audio reCAPTCHA [20]. In [65], the user

| Class | Name | Author | Main idea | Input to solve the test |
|---|---|---|---|---|
| Audio | A reverse Turing test using speech | Greg Kochanski, Daniel Lopresti, Chilin Shih 2002 [99] | Ten randomly chosen digit sequences with noise distortion | Typing the digit sequence |
| | Using a Text-to-Speech Synthesizer to Generate a RTT | Tsz-Yan Chan 2003 [34] | Sound clip with a digit sequence in the foreground and some English words in the background, both overlapped | Recognising the words |
| | Audio reCAPTCHA | Manuel Blum, Luis A. von Ahn, John Langford 2008 [20] | Sound clip with several speakers who speak random digits plus background noise | Identifying a sequence of eight digits |
| | Google audio CAPTCHA | Google Inc. | Sound clip of one speaker saying random digits 0-9, the phrase once again, followed by the same sequence plus background noise | Recognising the sequence of digits |
| | Digg's CAPTCHA | Digg.com Inc. | Sound clip of one speaker saying digits and characters with background noise | Recognising the sequence of digits and characters |
| | Audio CAPTCHA | Haichang Gao, Honggang Liu, Dan Yao, Xiyang Liu, Uwe Aickelin 2010 [65] | The programs presents a sentence for the user to read and the system will detect if it is human or not | Reading out loud a given sentence |
| | HIPUU | Graig Sauer, Jonathan Holman, Jonathan Lazar, Harry Hochheiser, Jinjuan Feng 2010 [162] | The system presents sounds related to an image | Selecting the corresponding label |
| Text Domain | Text-based CAPTCHAs | Philip Brighten Godfrey 2002 [68] | English text extracted from literature and with one word replaced for another | Detecting the bogus word |
| | Towards HIP in the text-domain | Richard Bergmair, Stefan Katzenbeisser 2004 [16] | A text containing several natural English sentences | Identifying which sentences are meaningful replacements of each other in each test-instance |
| | A CAPTCHA in the Text Domain | Pablo Ximenes, Andr dos Santos, Marcial Fernandez, Joaquim Celestino 2006 [202] | A structure with several "knock knock" jokes | Identifying the one that makes sense |

**Table 2.10:** Visual non OCR-based methods based on Non Visual CAPTCHAs.

is required to read aloud a given sentence to prove he is human. The latest version of audio methods is called HIPUU and its main focus resides in its usability [162].

**Text domain methods:** The second kind of non visual CAPTCHA methods are the Text Domain CAPTCHAs. These methods are based on the ability every human posses to distinguish different linguistic contexts and find the word that does not belong to the context. Unlike OCR-based methods, these algorithms do not require recognition of characters but semantic analysis. There are few of these tests due to the complexity in creating reliable algorithms [16, 68, 202].

### 2.4.1 Reliability of Non Visual CAPTCHA Methods

One of the most well known works on breaking audio CAPTCHAs in the one developed by [181]. In the tests, they used three different ASR techniques on segments of words or noise of the CAPTCHA. They successfully broke through the Google Audio CAPTCHA, the Diggs CAPTCHA and the audio reCAPTCHA.

Apart from the robustness against attacks, in the case of an audio/sound CAPTCHA the human friendliness encompasses three aspects: (a) the level of the noise in the audio, (b) the language of the audio and (c) the human ability when memorising what it is said [161, 207]. Typically, the distortion applied in these tests is background noise, and it affects highly the outcome of the CAPTCHA. Also, some characters and numbers are difficult to decipher because the user can confuse them with other ones that are similar. A study deployed at Microsoft's Hotmail service showed that none of the subjects were able to pass the test due to the distortions [63].

The content we can find in audio CAPTCHAs is usually language specific. Characters and digits are read in a specific language and are often not understandable to people who do not speak that particular language. Therefore, the big issue in this case is the localisation of the CAPTCHA.

The authors of [161] did an experiment to test the usability of the audio CAPTCHAs. The results obtained say that these tests fell well short of the 90% success rate necessary for a good CAPTCHA, and one possible explanation is that the audio CAPTCHA imposes more cognitive load than an average human user can handle. Another study made by [30] shows that the audio CAPTCHAs are the hardest amongst all the different kinds of CAPTCHAs. To leverage the difficulty in audio CAPTCHAs some studies have been carried out to study how to improve the usability of these kinds of methods [19].

## 2.5 DeCAPTCHAs

Breaking a CAPTCHA has not only become a topic of interest in research, but also a commercial software product and the main reason is the increasing use of internet marketing. Internet marketing, also known as online marketing, web marketing or e-marketing, is associated to the advertising and the marketing of products or services over the web including web pages, e-mail and wireless media. The objective of the companies is to increase the awareness of their company's goods and services. The common branches of internet marketing go from display advertising to search engine marketing and optimization, social media marketing, email marketing and referral or affiliate marketing [119].

Since internet marketing has become a daily routine every time a user navigates through the net, the amount of times a CAPTCHA is required to be solved has increased proportionally and so the need for automated tools to avoid repetitive tasks. A DeCAPTCHA is a software tool that automatically solves CAPTCHAs to allow users to save hours of manual work and reduce cost when monotonous tasks are automated. Another factor that has increased the popularity of DeCAPTCHAs is the fact that CAPTCHAs have become increasingly annoying since most of the time it requires more than one try to solve them.

The DeCAPTCHA service has become a hit in a small amount of time since it is capable of solving most of the CAPTCHAs displayed on sites like Ebay, CNN, Megaupload, Yahoo, Live, etc. DeCAPTCHA tools can be divided into two types as follows:

**Automated DeCAPTCHAs** This service is the most expanded type on DeCAPTCHA since it does not require human intervention to pass the tests. The DeCAPTCHAs are based on software tools and so their capabilities vary depending on the type of test they are trying to solve. Since the tools available are far from perfect they do not have a 100% success rate.

- Optical Character Recognition DeCAPTCHAs: They mostly use OCR software to recognise the characters in the test. New techniques include segmentation and clustering of each character.
- Audio DeCAPTCHAs: They capture the audio file of the CAPTCHA test and decode it after nullifying the background music or audio.

OCR DeCAPTCHAs are a one-type of service decoder. Their principal use is solving the OCR CAPTCHAs by inserting the path of where the CAPTCHA image is located and in result, they will return the decoded text answer. These services usually have a response time between 10-30 seconds or if well engineered less than 1 second. Their correctness rate is inherent to the complexity of the code and its flexibility to support different character shapes. The solving speed is one of the most important factors to take into account when buying this kind of service since most CAPTCHAs have an expiry time of 3-5 minutes [52].

These services typically charge expensive 1,000 DeCAPTCHA packet answers. Prices range from 2 dollars to around 4,000 dollars per packet. Once the user has made the payment, they get an amount of CAPTCHA credits to use. In case the DeCAPTCHA service fails solving the test, the user can notify the provider and get back the credits. Some examples of DeCAPTCHA providers are Death by CAPTCHA, DeCaptcher, Bypass CAPTCHA, etc. Also,


**Figure 2.9:** Logos of some commercial automatic DeCAPTCHAs.

these services are a high risk of investment, since a simple change in the CAPTCHAs can make the OCR unusable [52].

On the other hand, audio DeCAPTCHAs focus only on solving audio CAPTCHAs. One of the pioneers in this service is the Stanford audio DeCAPTCHA created by [28]. This program could listen to and correctly decipher commercial audio captchas used by Digg, eBay, Microsoft, Yahoo and reCAPTCHA. It managed to decipher Microsoft's audio CAPTCHA about 50% of the time or reCAPTCHA's codes 1% of the time, the most difficult ones of those tested, but even this small success rate is considered trouble for websites such as YouTube and Facebook that get hundreds of millions of visitors each day [29].

**Manual DeCAPTCHAs** The constant demand for programs to solve CAPTCHAs produced a sudden decrease of OCR software availability and a higher demand for DeCAPTCHAs. The result was the creation of human DeCAPTCHA services. This service implies hundreds of human teams working on solving the CAPTCHAs. These teams normally come from developing countries such as India or China. People involved in the teams go from virtual assistants, designers, project managers, illustrators, copywriters to data entry specialists. In the beginning, their tasks was the digitisation of books and company documents. Later, it evolved to decode CAPTCHAs and offer it as a service. Since the tests are solved by humans, these services usually have a response time between 10-30 seconds and can have a correctness rate of around 94%-99% [52].

The basic functionality of the human DeCAPTCHAs is as follows:

| | OCR DeCAPTCHAs | Human DeCAPTCHA services |
|---|---|---|
| Initial cost | Very high | Very low |
| Setup time | Very long: 2-3 months | Short: hours |
| Speed | Very fast | Slow |
| Correctness rate | 40% average | over 90% average |

**Table 2.11:** Comparison of OCR DeCAPTCHAs and Human DeCAPTCHAs services.

- Save locally the image of the CAPTCHA
- Send the image through a HTTP interface or a API client
- Wait for the response and insert in the CAPTCHA answer box.

These services typically charge a fixed price per 1,000 DeCAPTCHA packet answers. Prices range from one dollar to around 8 dollars per packet. Once the user has made the payment, they get an amount of CAPTCHA credits to use. In case the DeCAPTCHA service fails solving the test, the user can notify to the provider and get back the credits.

Manual DeCAPTCHAs are intended for CAPTCHA tests that require a human brain to find the answer since there are not yet robots capable of solving the questions:

- Quiz, Trivial or Math CAPTCHAs
- Video CAPTCHAs
- Face Recognition or Image CAPTCHAs
- Semantic CAPTCHAs or OCR CAPTCHAs

Both automatic and manual DeCAPTCHA services have a high demand on the market. Since different users have different requirements, a comparison is necessary before buying any service to acquire the most appropriate one (see table 6.2) [52].

In this chapter, the concept of CAPTCHA has been introduced. Along with it, its evolution from being a branch of HIP programs to being used as the primary method to prevent security

breaches in web services. A categorisation for CAPTCHAs has been presented as a novel contribution. Also, every type of CAPTCHA has been analysed depending on factors like reliability, robustness, efficiency and human friendliness. Finally, a new type of software called DeCAPTCHA has been introduced, whose functionality precludes normal CAPTCHAs.

# Chapter 3

# Digital Image Manipulation

The efficiency and robustness of the new CAPTCHA approaches that are presented in this thesis depend on a number of factors related to the warping distortions, morphing techniques, and 3D computer graphics software applied to the consequential images that are used in each method. To increase the robustness against machine recognition, it is necessary to apply a significant amount of distortions in the images, but at the same time, preserving the quality of the characters or the objects presented so human users are capable of recognising them without difficulty. In the CAPTCHA approaches developed, the balance between distortion and quality is one of the most important factors to take into account. Another important factor is the efficiency of the algorithms, since the creation of the CAPTCHA test and the processing of the user's input is done in real-time. For this, the manipulation of the databases, 2D characters, 3D objects, and face pictures must be done in the least amount of time possible and with minimum CPU charge.

In the following sections, digital warping and morphing techniques are described. The techniques used for the creation of the distortions in each approach are described, giving special attention to the specific algorithms used. In addition, it provides an overview of 3D computer

graphics techniques, specifically in the creation of 3D models, since they are used for the creation of characters in one of the approaches introduced in this thesis.

## 3.1 Digital Image Warping and Morphing

The main objective of CAPTCHAs is protecting web applications from malicious software so the tests that users have to pass must be more complex than only random words or images. For this reason, the final image or characters are digitally distorted. The applied distortions should make more difficult the recognition for the bots but not for the humans.

Digital image warping and morphing are the foundations of basic operations in computer graphics and they are evolving along with the field development, incorporating the latest research advances and trends. These operations include camera transformation, interactive modeling techniques, computer animation, image-based rendering, etc. Whilst warping focuses on deforming a single graphical object, morphing focuses on the interpolation between two objects.

When working with warping and morphing techniques, an image can be expressed as a graphical object that is described as a function: $f : U \subset \mathbb{R}^2 \rightarrow \mathbb{R}^n$, where U is the set that corresponds to the image shape. The function $f$ is the attribute function of the image. The most common attribute is colour, but another ones can be useful in different applications such as opacity, scene depth, etc.

To study and create different warping and morphing techniques, it is necessary to understand transformations between graphical objects. The first type of transformation is the one related to the Euclidean space and is the key concept to study this subject. Considering a subset $U \subset \mathbb{R}^m$ of the euclidean space, the set of transformations $T : U \rightarrow \mathbb{R}^n$ are called domain and $T(U)$ is the image of $T$. This transformation is commonly known as *mapping*. When $U = \mathbb{R}^n$, $T$ is a *global transformation* and it affects all the points of the space. However, when the trans-

formation occurs only on selected points of $U$, the result is an *intrinsic transformation*. A *local transformation* is a half way transformation between a global and a intrinsic transformation and it affects the selected points $p$ and the points in the neighbourhood. To construct this type of transformation the object is subdivided into smaller pieces, then the transformation between each piece is obtained and put together to transform the whole object. Triangles and quadrilaterals are the preferred shapes to represent the blocks of the graphical objects. Also, to define the transformations between blocks there are two possible ways: first, defining a parametrisation and second, defining a local coordinate system. Interpolation solves the problem of scattered data.

The second type of transformation is between two graphical objects that takes into account both the shape and attributes of the objects. The *shape transformation* is defined by a spatial transformation $T : U_1 \rightarrow U_2$. The *attributes transformation* has two options: the first option is to compute the attributes from information about the transformed object; the second option is to transform the attributes from the original object. If the attribute function $f$ depends on the shape geometry, it can be recomputed at each point of the transformed objects. If the attributes are the material type of an object or the colour, they can be computed by applying the transformation in the original object. The strategies to compute the values are forward mapping and inverse mapping.

A very important matter when computing transformations is the specification of the transformation, since it has a great influence in the design of the user interface and the final result. It needs a finite number of parameters that vary depending on the type of transformation. Warping transformations require a finite points specification that normally concur with the vertices of the image. Metamorphosis or morphing transformations require more complex functions; parametric specification, algorithmic specification or specification by parts.

In the rest of the chapter, several classes of transformations will be presented both for warping and morphing image manipulations.

### 3.1.1 Digital Image Warping

Digital image warping [200] is a specific branch of image processing that consists of geometric transformations of digital images. A geometric transformation is a vector function that maps the spatial correspondence between points from an input image to an output image. Image warping includes many applications that go from scale, translation or rotation to distortion compensation, decalibration for image registration, geometrical normalisation, map projection or texture mapping.

An image is a graphical object that can be expressed as $O = (U, f), U \subset \mathbb{R}^m$, so when a continuous $k$-parameter family of transformations is applied, the result is a continuous deformation of the image called warping. For each fixed vector $v \in \mathbb{R}^k$, the instantiation of the warping is expressed as $F_v(O)$.

#### Specification methods

To create a successful warping it is necessary to specify the transformation. This implies the consequential description and representation of the transformation. The basic notion for a transformation is that there is a finite number of parameters that can be represented. Therefore, the aim is to create different techniques that are adequate to different problems. The most common representations are as follows:

- Parametric representation. This representation uses a finite dimensional parametrisation of the space. It is used for linear transformations of the space $\mathbb{R}^n$, affine transformations and projective transformations of the projective space $\mathbb{RP}^n$.
- Representation by sampling. It considers a transformation with a finite set of points that are going to be sampled.
- Representation by parts. It describes a set of transformations and their restrictions.

- Representation by transformation decomposition. It uses a combination of simple transformations to create more complex ones. It is used for isometries, plane projective transformations, twist transformations or separable transformations.

**Warping computation**

The basic formulations for the transformations can be expressed in terms of a general homogeneous transformation matrix. They include simple planar mappings: affine and perspective transformations. For non-planar mappings, bilinear transformations are applied. Warping is generally done by polynomial transformations.

For 2D image projections, the general mapping function is $[x,y] = [X(u,v),Y(u,v)]$, where $[u,v]$ are the input image coordinates and $[x,y]$ refers to output pixel coordinates. X and Y are the mapping functions that show the spatial relationship between the two images. The spatial transformation used for the forward mapping can be expressed in terms of a 3x3 transformation matrix:

$$T_1 = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \tag{3.1}$$

This 3x3 matrix specifies 2-D coordinate transformations and operates in a homogeneous coordinate system [200]. The general representation of a transformation is:

$$\left[x',y',w'\right] = [u,v,w]\,T_1 \tag{3.2}$$

where *x'*, *y'*, and *w'* are the destination coordinates and *u*, *v*, and *w* are the source coordinates.

*Affine transformations* are used for planar mapping that corresponds to an orthographic projection (also called orthogonal projection that represents a three-dimensional object in two dimensions) or parallel plane projection from the input image onto the output image. Affine mappings preserve collinearity but they do not necessarily preserve angles or lengths. It has six degrees of freedom thus facilitates only triangle-to-triangle mappings. The forward mapping function can be expressed as:

$$\left[x', y', 1\right] = [u, v, 1] \begin{bmatrix} a_{11} & a_{12} & 0 \\ a_{21} & a_{22} & 0 \\ a_{31} & a_{32} & 1 \end{bmatrix} \tag{3.3}$$

*Affine transformations* are a composition of translations, rotations, shears and dilations. As a result, geometric contraction, expansion, dilation, reflection, rotation, shear, similarity transformation, spiral similarities and translations can be done with the affine matrix [194].

*Perspective transformations* are planar mappings that preserve parallel lines only when they are parallel to the projection plane. In other respects, the lines converge to a vanishing point (see Figure 3.1). It preserves lines in all orientations and it has nine degrees of freedom, facilitating quadrilateral-to quadrilateral mappings. The forward mapping function can be expressed as:

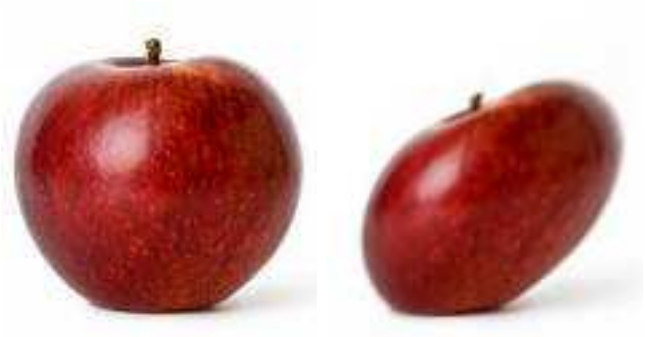

**Figure 3.1:** Example of a perspective transformation

$$\left[x',y',w'\right] = [u,v,w] \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \tag{3.4}$$

*Bilinear transformation* is a four-corner mapping for nonplanar quadrilaterals. It uses forward mapping to map rectangles onto nonplanar quadrilaterals. Bilinear mappings preserve lines that are vertical or horizontal in the input image and the points remain equispaced. Lines outside these two directions are not preserved as lines. Also, bilinear interpolation is necessary to evaluate the X and Y mapping functions. The forward mapping function can be expressed as:

$$[x,y] = [uv,u,v,1] \begin{bmatrix} a_3 & b_3 \\ a_2 & b_2 \\ a_1 & b_1 \\ a_0 & b_0 \end{bmatrix} \tag{3.5}$$

*Polynomial transformations* are used to invert an unknown distortion function when geometric correction is required. The 1st to 5th order equations describe how the distorted image has to be warped to be geometrically corrected. Up to the first order, polynomial transformations can perform rotation, scaling and translation on the entire image with a low computational complexity. In higher orders, more complex warpings can be achieved but the computational complexity is also higher. An interpolation grid is introduced to reduce the computational cost rather than applying the mapping on the entire image. The forward mapping functions can be expressed as:

$$u = \sum_{i=0}^{N} \sum_{j=0}^{N-i} a_{ij} x^i y^j \tag{3.6}$$

$$v = \sum_{i=0}^{N} \sum_{j=0}^{N-i} b_{ij} x^i y^j \tag{3.7}$$

### 3.1.2 Digital Image Morphing

Digital image morphing is an image processing technique to transform one shape into another. This process is also called *metamorphosis*. The idea is to create an animation with a sequence of intermediate images which when put together with the original images would represent the metamorphosis from one image to the other [72]. Nowadays, morphing techniques are commonly used by many different fields and not only computer vision, such as TV and movies for animations, art, medical image processing, etc. Image morphing combines image warping with diverse methods to control the colour transition in the intermediate images created in the process (colour interpolation). Face morphing is a specific application that morphs one face into

another. In our research, we have focused in this application to create distorted face images that serve to develop face recognition CAPTCHAs.

Considering two graphical objects such as $O_1 = (U_1, f_1)$ and $O_2 = (U_2, f_2)$ with $U_1, U_2 \subset \mathbb{R}^m$, a morphing between $O_1$ and $O_2$ is a $k$-parameter continuous family of transformations, $F : O_1 \times \mathbb{R}^k \to \mathbb{R}^n$, such that there are the consequent parameter values $v_0$ and $v_1$ that satisfy the function $F_{v_0}(O_1) = O_1$ and $F_{v_1}(O_1) = O_2$. For each vector $v \in \mathbb{R}^k$, a new graphical object is obtained $O_v = F_v(O_1)$. The family of graphical objects obtained perform a transition from one object to another, as $v$ varies on the parametrical space. Basically, a metamorphosis or morphing is a continuous deformation from source object $O_1$ to the destination object $O_2$, transforming both the shape and the attributes.

A morphing path or an animation path φ is the spatial interpolation between the points on the source object to the destination object, so considering a morphing $F : O_1 \times \mathbb{R}^k \to O_2$ with a parameter path $c : [0,1] \to \mathbb{R}^k$, for each point $p \in U_1$, the restriction $F \restriction p \times [0,1] : p \times [0,1] \to O_2$ defines a path φ on $\mathbb{R}^n$ that connects p to the point $F(p) \in U_2$.

The algorithms to control the animated process should calculate the new positions and colour transition rates for the pixels in each image. During the morphing process there are three stages [199]:

**Specification methods**

A morphing technique consists of two warpings and a blending operation. Therefore, to specify the parameters, it follows the same procedure as a warping. The morphing transformation tends to focus more on features than parameters since the main objective is to create an animation of one object transforming into another.

The specification methods define the control points in the image that are going to be used for the warping. Normally, this is a difficult process since it is necessary to specify the object or the boundaries. In most cases this is performed manually. Most morphing techniques use

specification by parts where the image is subdivided in a collection $U = U_i, i = 1, ..., n$ of subsets of the set $U$, so there is a restriction $T_i = T|U_i$ of the transformation $T$ for each subset. Accordingly, the final transformation is referred as a collection of pairs $(U_i, T_i)$. So for each local set of parts in the source graphical object $U_i$, there is the corresponding set of parts in the destination image $V_i$, and the transformations occur between these sets. Normally, there is a one-to-one correspondence between the parts, but it is not necessary. Finally, the final global transformation $T$ is gathered through a reconstruction method.

Inside this class of specification, it is possible to define subclasses of specification:

- specification by partition
- specification by features

**Specification by partition** In this subclass of specification, the subsets of the source object $U_i$ and the destination object $V_j$ constitute a partition of the whole object, since

$$\begin{gathered} \bigcup U_i = U, \quad \bigcup V_i = V \\ U_i \bigcap U_j = \varnothing \quad V_i \bigcap V_j = \varnothing, \quad if \quad i \neq j, \end{gathered} \tag{3.8}$$

where there is a one-to-one correspondence between both subsets. The partitions used are triangulations or cellular decompositions called meshes. Usually a mesh is a collection of vertices, edges and faces that defines the shape of a polyhedral object [200]. For morphing, it is necessary to define the meshes in the source image and in the destination. Usually the number of meshes are the same and the only thing that changes is the location of the vertices in the meshes [108].

**Specification by features** In this subclass of specification, the source and the destination object have common distinguished features that are used to specify the transformation. This type of specification does not define partitions, but to allow a smooth and good morphing,

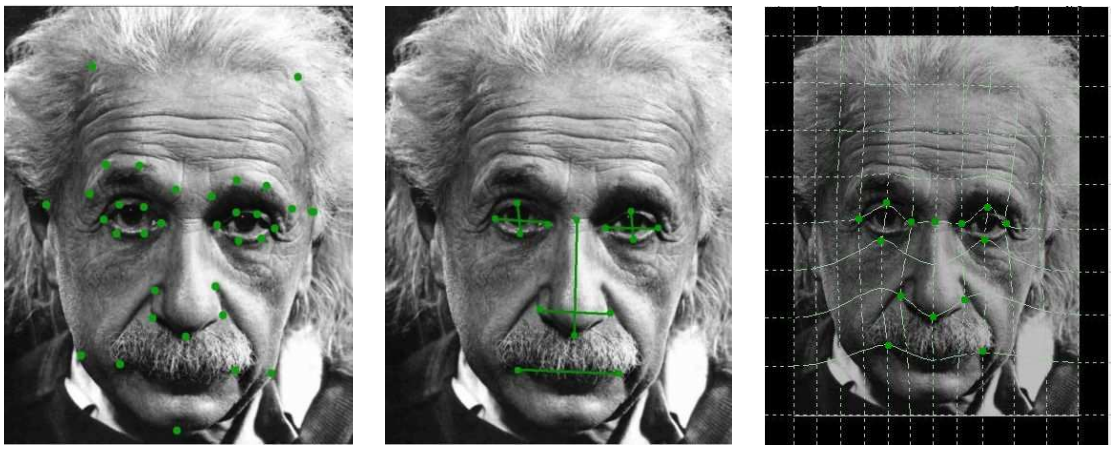

**Figure 3.2:** Feature specification types: (a) points, (b)lines and (c) meshes.

there are some conditions that must be followed; adjacency relationships (i.e, pixels in an image) must be the same in the source and destination objects. Also, feature specifications are normally non-overlapping to avoid multiple definitions when the features are intersecting. In actual morphing algorithms, the most common techniques used are as follows [199]:

- Point-based specification: The number and the location of the points determine how accurately two images can be morphed. A high number of points will lead to a much more accurate warp, however it will also increase the computational cost the system requires to perform the method. In our research, we have focused on face morphing, therefore control points are selected within the face region [6].

- Higher-dimensional features: In this case, the specifications are 1D features that correspond with oriented line segments. For the transformation, there are two possibilities; one pair of lines or multiple pairs of lines. In the first case, only one line in each image is placed and the warp will consist of moving appropriately the pixels to maintain their relative position from the specified line to the destination line. In the second case, there are multiple lines, so each line in the source has its corresponding line in the destination [14].

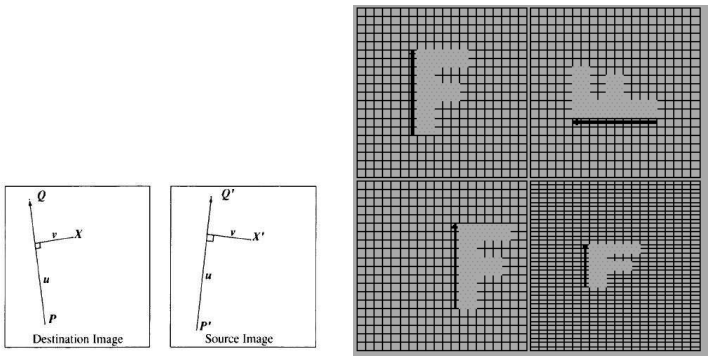


**Figure 3.3:** Feature-Based Morphing process with one pair of lines

### Morphing computation

Warp generation defines the algorithm used to calculate and transform the pixels in one image to the new mapped pixels in the other image. The most popular techniques are as follows [199]:

- Feature-Based (Field) morphing: This technique is based upon fields of influence surrounding two-dimensional control primitives. It uses lines to relate features in the source image to features in the destination image. For the warping, it performs the reverse mapping in which every pixel in the destination image is run through and sampled to the correct pixel in the source image. This method can morph images with both one pair of lines or multiple pairs of lines [14].

  For the first approach, a pair of corresponding lines in the source and destination images define a coordinate mapping from the destination image pixel coordinate $X$ to the source image pixel coordinate $X'$ such that for a line $PQ$ in the destination image and $P'Q'$ in the source image.

$$X' = P' + u \cdot (Q' - P') + \frac{v \cdot Perpendicular(Q' - P')}{||Q' - P'||} \tag{3.9}$$

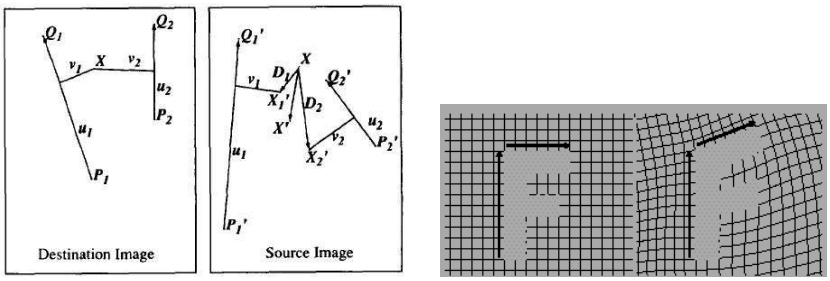


**Figure 3.4:** Feature-Based Morphing process extracted with multiple pair of lines

For the second approach, a weighting of the coordinate transformations for each line is performed. A position $X_i'$ is calculated for each pair of lines. The displacement $D_i = X_i' - X$ is the difference between the pixel location in the source and destination images, and a weighted average of those displacements is calculated. The weight is determined by the distance from X to the line. This average displacement is added to the current pixel location X to determine the position $X'$ to sample in the source image. The single line case is a special case of the multiple lines case, assuming the weight never goes to zero anywhere in the image. The weight assigned to each line should be strongest when the pixel is exactly on the line, and weaker the further the pixel is from it.

$$weight = (\frac{length^p}{a + dist})^b \tag{3.10}$$

where *length* is the length of a line, *dist* is the distance from the pixel to the line, and *a*, *b*, and *p* are constants that can be used to change the relative effect of the lines.

- Mesh Morphing: In this method, the input and the output images are partitioned into a mesh of patches. Each patch delimits an image region within the image which a continuous mapping function applies. Thus, the warping consists of transforming each patch onto its counterpart in the second image. This algorithm uses the source and destination images with two 2D arrays of coordinates with the same dimensions which impose a rectangular

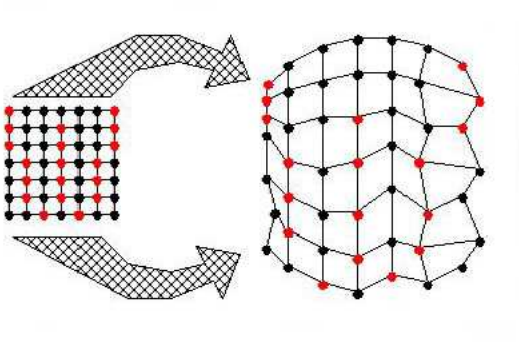

**Figure 3.5:** Mesh morphing process extract from http://davis.wpi.edu/

topology to the mesh [200]. The two images are processed through 2-pass warping with 2 output intermediate images I1 and I2. The first pass is responsible for resampling each row independently. It maps all initial image points coordinates $(u,v)$ to their $(x,v)$ co-ordinates in the intermediate image , thereby positioning each input point into its proper output column.The second pass then resamples each column in intermediate image, mapping every $(x,v)$ point to its final $(x,y)$ position in $I1/I2$. An interpolated mesh M is used by each frame in the transformation and it is computed by performing linear interpolation between respective points in the source and destination image [200].

**Blending Techniques**

The blending process is the last stage and it begins once the pixels are located in order to preserve the features. This stage defines the range of warping and the colour blending between the two images. Defining a morphing sequence $O_\lambda$ , $0 \leq \lambda \leq 1$, from the graphical object $O_1$ to the graphical object $O_2$, for each $\lambda$ we obtain a graphical object $O_\lambda$ with intermediate shape and attributes. If $\lambda$ is close to 0, the final object will be more similar to $O_1$, and when $\lambda$ is close to 1, it will resemble $O_2$. To compute the blending process, the general procedure is as follows:

- Forward warping: Warping of the source object $O_1$.
- Inverse warping: Warping of the destination object $O_2$.
- Blending of the two warped object following the $\lambda$ weight.

The forward and inverse warpings are computed to bring together the source and the destination targets while preserving the feature properties of each one. The warpings and the blending process constitute the morphing algorithms. The blending process is both the geometry and the attributes blending and it requires a *geometry alignment* before computing the features. The geometry alignment blends geometrically the distinctive features in both the source and destination images to create a smoother transition.

The generic approach to create a blending technique is to perform a weighted interpolation of the function values. Given two functions $f$ and $g$, the function $h$ produces the blending as follows:

$$h(p,t) = w(p,t)f(p) + (1 - w(p,t))g(p), \quad p \in \mathbb{R}^3, t \in [0,1] \tag{3.11}$$

where $w(p,0) = 1$, and $w(p,1) = 0$, $\forall p$

**Cross-dissolve** It is a simpler version of the blending technique by weighted average. It was initially the colour blending method of choice, but it produces undesirable artifacts referred to as ``ghosts´´due to the computed warp function [93].

Cross-dissolve is a linear interpolation method, renamed like that by the cinematography industry. Linear interpolation is a first-degree method that passes a straight line through every two consecutive points of the input signal [145]. The fundamental equation of linear image blending is:

$$h(p,t) = (1-t)f(p) + tg(p), \quad p \in \mathbb{R}^3, t \in [0,1] \tag{3.12}$$

This method reduces variation, sharpness and contrast.

**Scheduled Blending** It uses a scheduling function that produces a prearranged blending of the attribute properties [72].

## 3.2 3D Computer Graphics Approach

Attempts have been made to use 3D models [91] to generate CAPTCHAs. The use of 3D models helps in overcoming the limitations of traditional CAPTCHA tests. Besides, using computer graphics algorithms is not computational expensive anymore.

Approaches dealing with the manipulation of 3D objects fall in the area of computer graphics. Computer graphics is a field of computer science that studies the management of visual content and geometric information as well as how to digitally synthesise them. In computer graphics, the input information is usually non-visual and the output is an image. Computer graphics has two main areas of research: modelling and visualisation [187]. Visualisation allows the visual representation of stored data in the computer. Modelling is the discipline that uses geometric modelling as well as computer vision to create models.

3D computer graphics is based upon virtual 3D models in a 3D space using geometric data with the aim of either performing calculations or rendering 2D images for real-time viewing or for displaying [83]. 3D computer graphics is divided sequentially into three phases: modelling, layout and animation and rendering.

The first stage of 3D computer graphics is modelling. This consists of developing a mathematical representation of a 3D object through graphics software. A 3D model is a representation of an object of the virtual world that contains information about its geometry and surface properties [166]. Once the model is created, it is necessary to place lights in the virtual world. The sources, the locations and the characteristics must be defined. Objects in the 3D space are constructed with elementary geometric shapes such as triangles, polygons, lines, curved surfaces, etc.

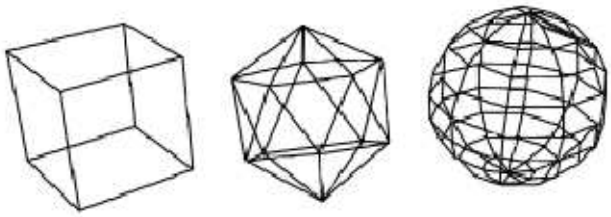

**Figure 3.6:** Wireframe 3D models

There are three main categories for 3D models:

1. Solid models: The models are represented as solid objects and based on constructive solid geometry (CSG) or Boundary representation (B-REP) methods to define solid shapes.

2. Surface models: The models are represented by a surface or the boundary of the characterised object. The surfaces are defined, trimmed and merged.

3. Wireframe models: The models are represented by specifying each edge of the characterised object where two mathematically continuous homogeneous surfaces encounter or by connecting the vertices of the object. It is suited for real-time systems and complex 3D models due to its simplicity for rendering (see Figure 3.6).

One of the most popular methods for representing an object is the polygon mesh modelling technique. An object is constructed from a number of surfaces each one of them represented by a mesh of polygons. Each polygon is constructed at the same time by a set of vertices. The vertices are three-dimensional points in the world coordinate space and the mesh is referred to the group of polygons connected together by a common vertex [192]. Amongst the most used methods of constructing meshes, we can find:

1. Primitives: A primitive is a predefined polygonal mesh created by specific modelling software. The most commons ones are: cubes, pyramids, cylinders, 2D primitives and spheres.

2. Box modelling: This technique is used to create the model by modifying primitive shapes in a way to create a draft of the final model. First, the subdivide tool splits edges and

surfaces into smaller parts by adding new vertices. Second, the extrude tool creates a new mesh by connecting different faces.

3. Inflation modelling: The user creates the 3D model using 2D images with different perspectives and angles of the same object.

Once the model is created and positioned in the 3D world, the next step is rendering to create the actual 2D image from the model. The rendering process will contain information about geometry, viewpoint, texture, lighting and shading. The most important features when rendering are:

- Shading: controls the variation in colour and brightness with lighting.
- Reflection/Scattering: controls the interaction with the surface at a given point.
- Texture-mapping: controls the application of details to surfaces. ‘
- Shadow: a method to obstruct the light.
- Indirect illumination: controls the global illumination of the scene.
- Refraction and diffraction: controls the bending of the light.
- Transport: a method to describe how illumination in a scene gets from one place to another. Visibility is a major component of light transport.
- 3D projection: a method to map three-dimensional points to a two-dimensional plane.

Rendering algorithms use a number of different techniques to obtain a final image but it is computational complex to trace every ray of light in a scene. Thus, more efficient light transport techniques have been developed:

- Scanline rendering and rasterisation: It is a process for rendering 3D models from primitives. It works on polygon-by polygon basis in the case of rasterisation and row-by row basis with scanline rendering [22].

- Ray tracing: It describes a method for generating an image by tracing the path of light through pixels in an image plane. It works by tracing a path from an imaginary eye through each pixel in a virtual screen, and calculating the colour of the object visible through it [22].
- Ray casting: This algorithm follows rays of light from the eye of the observer to a light source. The main difference with ray tracing is that ray casting does not compute the new tangent of a ray of light after intersecting a surface [153].
- Radiosity: It is a method which attempts to simulate the way in which directly illuminated surfaces act as indirect light sources that illuminate other surfaces [74].

3D models have recently been used in creating new types of CAPTCHAs, mainly spatial and object recognition ones. One of the most recent approaches [91] uses a manually designed library of three-dimensional objects. The attributes of each object are described by a label. The user has to enter the selected attributes to pass the test. The issues with this type of CAPTCHA are, firstly, that objects are selected from a database but they always remain the same. Secondly, the user has to recognise attributes from objects and not every user is going to be able to do it.

In this chapter, the main methods to digitally manipulate images used in the development of the CAPTCHAs has been described. Firstly, digital warping and morphing techniques are presented, giving special attention to the specific algorithms used to create the CAPTCHA approaches. And secondly, this chapter provides an overview of 3D computer graphics techniques, specifically, the creation of 3D models, since they are used for the creation of characters in the 3D models visual-based approach introduced in this thesis.

# Chapter 4

# Digital Image Recognition

This chapter takes a look on digital image recognition techniques with the aim of testing the reliability of the proposed CAPTCHA methods, against machine attacks. The main goal of researching these techniques is to find adequate methods to recognise characters, objects and faces. For character recognition, the usual software used is the OCR software, but the research done in this area has shown that there are more advance techniques capable of segmentation and clustering that are capable of recognising characters. Face recognition is a very important and wide field of research and every year many technological advances are proposed.

In the following sections, SIFT technique and face recognition methods are described. The SIFT technique is used to recognise objects and characters and a general overview of its functionality is presented. In addition, a general state-of-the-art review of the most used and important face recognition techniques is presented since the experiments for the second approach are based on face image recognition.

## 4.1 Scale Invariant Features Transform (SIFT)

Image recognition is a task that can be solved almost without any effort by a human but it cannot be solved straightforwardly by a computer vision technique. In computer vision, image features are characteristic points of an object in an image. Image features are used to describe and identify the object when attempting to locate that object in a random image that can contain many other objects. Image features need to be robust against changes in image scale, noise, local geometric distortion and illumination.

SIFT [120] is an approach for detecting and extracting local features descriptors which allows to find an object only based on the location of its keypoints and its appearance. The main applications include object recognition, video tracking, match moving, 3D modelling and image stitching. SIFT presents the following advantages:

- SIFT features are invariant and robust against image noise, uniform scaling, rotation, minor changes in illumination and in 3D camera viewpoint.
- Highly distinctive features: correct object matching with low probability of mismatch.
- Ease of extraction even having a large database of features.
- Object and scene recognition.

This approach transforms image data into scale-invariant coordinates related to local features. It generates a large number of features so the entire image is fully covered with scales and locations. The procedure extracts keypoints of objects from a set of reference images and stores them in a database. An object is recognised in a new image by independently comparing each feature from the new image to the features database and finding candidate matching features based on the Euclidean distance of their feature vectors. From the full set of matches, subsets of keypoints that agree on the object and its location, scale, and orientation in the new image are identified to filter out good matches. The determination of consistent clusters is performed rapidly by using an efficient hash table implementation of the generalized Hough transform.

Each cluster of 3 or more features that agree on an object and its pose is then subject to further detailed model verification and subsequently outliers are discarded. Finally the probability that a particular set of features indicates the presence of an object is computed, given the accuracy of fit and number of probable false matches. Matches that pass through all these tests can be identified as correct with high confidence. To minimise the cost of the feature extraction, the process is divided into four stages cascade filtering.

### 4.1.1 Scale-space extrema detection

It is the first stage of the cascade filtering algorithm which identifies the potential interest keypoints over scales and locations that can be recognised under different views of the same object. The process is done by constructing a Gaussian pyramid and searching for local keypoints in a series of difference-of-Gaussians (DoG) images.

Detecting locations that are invariant to scale change of the image can be achieved by searching for stable features across all possible scales, using a continuous function of scale. Koenderink [100] and Lindeberg [117] showed that the only possible scale-space kernel is the Gaussian function. Thus, the scale space of an image is defined as a function, $L(x,y,\sigma)$, that is produced from the convolution of a variable-scale Gaussian, $G(x,y,\sigma)$, with an input image, $I(x,y)$:

$$L(x,y,\sigma) = G(x,y,\sigma) * I(x,y) \tag{4.1}$$

In order to detect efficiently the stable keypoint locations in scale space, David Lowe proposed to use scale-space extrema in the difference-of-Gaussians function convolved with the

image, $D(x,y,\sigma)$, which can be computed from the result of:

$$D(x,y,\sigma) = (G(x,y,k\sigma) - G(x,y,\sigma)) * I(x,y) = L(x,y,k\sigma) - L(x,y,\sigma) \tag{4.2}$$

where k is a constant multiplicative factor. There are several reason for choosing this specific function; first, it is particularly efficient to compute. Second, the maxima and the minima of $D(x,y,\sigma)$ provides the most stable image features. Finally, a close approximation to the scale-normalised Laplacian of Gaussians is provided.

The remaining step consists on calculating the maxima and minima of $D(x,y,\sigma)$. The process is done by comparing each pixel of the Gaussian pyramid in the DoG images to its eight neighbours at the same scale and nine corresponding neighbouring pixels in each of the neighbouring scales. If the pixel value is the maximum or minimum among all compared pixels, it is selected as a candidate keypoint.

### 4.1.2 Accurate keypoint localization

Following the first step where all the candidate keypoints were identified, in the second stage the candidate keypoints are localised to sub-pixel accuracy and eliminated if found to be unstable. The algorithm performs a detailed fit to the nearby data for location, scale, and ratio of principal curvatures. The points with low contrast or poorly localised along an edge are rejected.

The location of the points is based on Brown and Lowe [24] approach for fitting 3D quadratic function to the local sample points in order to determine the interpolated location of the maximum. His approach uses the Taylor expansion of scale-space function, $D(x,y,\sigma)$ shifted so that the origin is at the sample point:

$$D(x) = D + \frac{\partial(D^T)}{\partial(x)}x + \frac{1}{2}x^T\frac{\partial(D^2)}{\partial(x^2)}x \tag{4.3}$$

where $x = (x, y, \sigma)^T$ is the offset of the sample point. The location of the extremum $\hat{x}$ is determined by:

$$\hat{x} = -\frac{\partial^2(D^{-1})}{\partial(x^2)}\frac{\partial(D)}{\partial(x)} \tag{4.4}$$

As proposed by Brown, the Hessian and derivative of D are approximated by using differences of neighbouring sample points, and such solution have a minimal cost. Once the offset $\hat{x}$ is calculated, the extrema location can be interpolated adding the offset and the location of its sample point. In order to reject unstable extrema with low contrast, the function value at the extremum, $D(\hat{x})$, is calculated:

$$D(\hat{x}) = D + \frac{1}{2}\frac{\partial(D^T)}{\partial(x)}\hat{x} \tag{4.5}$$

The DoG function will have strong responses along edges, even if the candidate keypoint is unstable to small amounts of noise. Therefore, in order to increase stability, we need to eliminate the keypoints that are located on a frame's border:

1. The Hessian matrix of each candidate keypoint is calculated for each candidate keypoint.
2. Determinant and trace of each candidate keypoint are estimated.
3. The main curvature of a candidate keypoint is determined by using the trace and determinant.

Therefore, it is only necessary to check that the ratio of principal curvatures is below some threshold, r. Thus, we check that the following inequation is true:

$$\frac{Tr(H)^2}{Det(H)} < \frac{(r+1)^2}{r} \tag{4.6}$$

### 4.1.3 Orientation Assignment

The third stage identifies the dominant orientations for each keypoint based on its local image properties. The assigned orientation, scale and location for each keypoint enables the keypoint to be invariant to image rotation. However, it is also necessary to keep the already obtained scale-invariance. For this reason, the previously selected scale of the candidate keypoint is used to choose the Gaussian smoothed image, $L(x,y,\sigma)$, with the closest scale, so all computations are performed in a scale-invariant manner. For each image sample, $L(x,y)$, at this scale, the gradient magnitude, $m(x,y)$, and orientation, $\theta(x,y)$, is precalculated using pixel differences:

$$m(x,y) = \sqrt{(L(x+1,y) - L(x-1,y))^2 + (L(x,y+1) - L(x,y-1))^2} \tag{4.7}$$

$$\theta(x,y) = \tan^{-1}\left(\frac{L(x,y+1) - L(x,y-1)}{L(x+1,y) - L(x-1,y)}\right) \tag{4.8}$$

Subsequently, a 36 bins orientation histogram is formed from the gradient orientations of sample points within a region around the keypoint. Peaks in the orientation histogram correspond to dominant directions of local gradients. Once the highest peak in the histogram is detected, any other local peak that is within 80% of the highest peak is used to also create a keypoint with that orientation. Thus, for locations with multiple peaks of similar magnitude, there

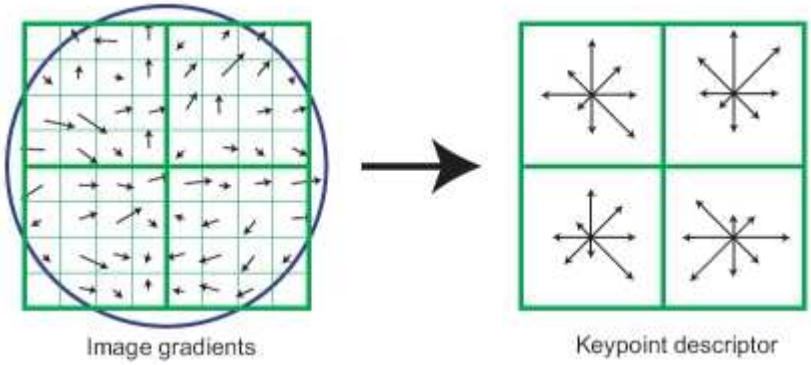


**Figure 4.1:** Procedure to calculate the local image descriptor [120]

will be multiple keypoints with the same location and scale but different orientations. Even though only about 15% of points are assigned multiple orientations, the stability of matching is highly increased. For better accuracy, each keypoint gradient magnitude value consists on the interpolation of the three closest peak's gradient magnitudes.

### 4.1.4 Local image descriptor

The final stage builds a local image descriptor for each keypoint, based upon the image gradients in its local neighbourhood that is highly distinctive yet is as invariant as possible to remaining variations, such as change in illumination or 3D viewpoint.

The keypoint descriptor used by SIFT is created by sampling the magnitudes and orientations of the image gradient in the patch of pixels around the keypoint, and building smoothed orientation histograms to capture the important aspects of the patch (see Figure 4.1). A 4x4 array of histograms, each with 8 orientation bins, captures the rough spatial structure of the patch. This 128-element vector is then normalised to unit length and thresholded to remove elements with small values. The keypoint descriptor representation is noteworthy in several respects:

1. The representation is cautiously designed to avoid boundary effects problems. Smooth changes in location, orientation and scale for not causing radical changes in the feature vector.
2. It is compact, expressing the patch of pixels using a 128 element vector.
3. The representation is resilient to perspective deformations even though it is not explicitly invariant to affine transformations.

## 4.2 Face Recognition

### 4.2.1 Face Recognition Systems

Face recognition is a task that humans perform effortless and in a natural way. Because of that, one of the main challenges in artificial intelligence is to understand how people recognise faces and develop a reliable and robust face recognition system, as recognising faces is a major step in building intelligent machines. Research in this area is not only a matter of artificial intelligence, but also the subject of diverse applications such as biometric authentication, surveillance and multimedia management [193].

A face recognition system is a computer application that allows automatic identification of faces in both images and videos [76]. It can have one or both modes: face verification (or authentication) or face identification (or recognition). Face verification (or authentication) consists in confirming or rejecting the identity of a subject by matching a given face with a stored face template whose identity is being claimed [185]. Face identification (or recognition), on the other hand, consists in comparing an unknown input image with the templates of known individuals to identify the identity of the input image [185]. The whole system can be divided into three main steps: Face detection, feature extraction and face recognition (see Figure 4.2).

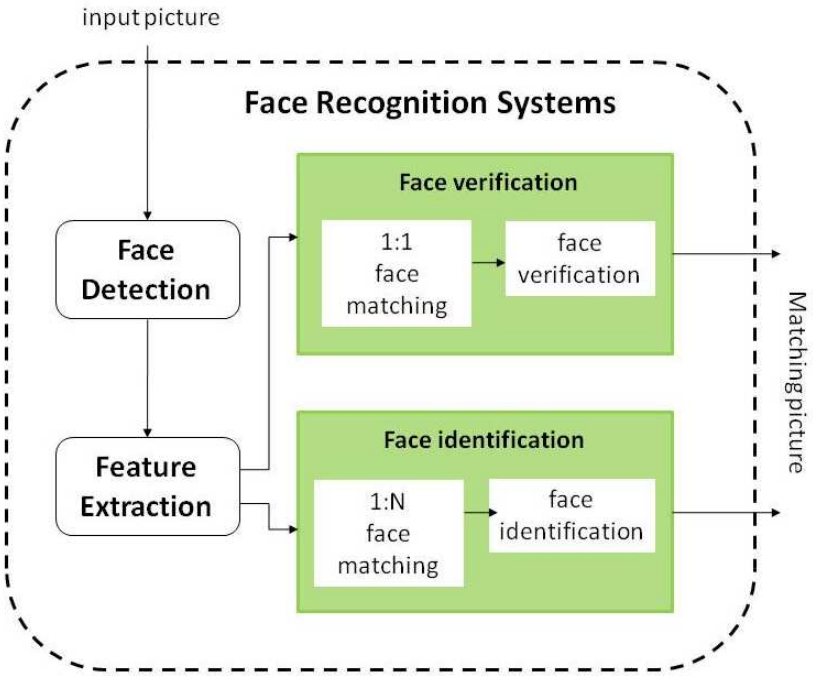


**Figure 4.2:** Flow diagram of the two approaches for face recognition: (a) Face verification system and (b) Face identification system

Despite the extensive research carried out addressing this challenge, 2D automatic face recognition systems are still far away from being capable of recognising faces in every kind of situation, primarily because of the diverse ambient conditions (view point, illumination) and also because of temporal effects such as occlusion, disguise, face ageing, facial expressions, etc [62]. Different approaches of face recognition for 2D images can be categorised into tree main groups as follows [216]:

- Holistic Approach: In this approach the whole face contour is taken into account in the face detection system. Amongst the most widely used algorithms we can find: eigenfaces [186], probabilistic eigenfaces [129], fisherfaces [15], support vector machines [139], nearest feature lines (NFL) [113] and independent-component analysis approaches [12]. All these methods are based upon principal component analysis (PCA) algorithms.

- Feature-based Approach: In this approach, local features on face such as eyes and nose, are segmented and used as input data to construct a structural classifier. In this cate-

gory are also included pure geometry [123], dynamic link architecture [198] and hidden Markov model methods [136].

- Hybrid Approach: This approach is developed following neurophysiologic research. These studies show that human beings perceive both local features and whole face. The most significant features are eyes, mouth and nose, and when represented, it is natural for these features to describe distinguishing characteristics not present in other facial features. The main aim in this approach is to develop models for the eyes and mouth as ravines on the image surface. The most common techniques are modular eigenfaces [142], hybrid local feature [141], shape normalised [45], and component-based methods [81].

### 4.2.2 Face Detection

The recent explosion of research activities in face recognition is due to the necessity of information about a user's identity, state and intent. This information can be extracted from images so computers can react accordingly. In the last ten years, many research demonstrations and commercial applications have been developed to reach that aim [210]. Face detection is the first step in a face recognition system. A face detector identifies and locates the face despite the position, scale, rotation, illumination, age and expression. It should work regardless of what it is been analysed is an image or a video. A face detector determines whether or not there are any faces in the given arbitrary image and, if present, returns the image location and the extent of each face [210]. The performance of these techniques depends on the quality of face localisation and their ability to normalise, since classification tends to be highly nonlinear and non-convex [112].

The challenges associated with face detection are related to the following factors [210]:

- Pose: The face in an image may vary due to the relative camera-face pose (frontal, 45 degree, profile, upside down)

- Presence or absence of structural components: Facial features such as moustaches, beards and glasses may or may not be present. They have a great deal of variability including shape, color and size.
- Facial expression: A person's facial expression can highly change the appearance of the face.
- Occlusion: Faces may be partially occluded by some objects or by other faces in an image with a group of people.
- Image orientation: Face images directly vary for different rotations about the camera's optical axis.
- Imaging conditions: Face appearance varies due to factors such as lighting (spectra, source distribution and intensity) and camera characteristics (sensor response, lenses).

Face detection is usually performed following several cues: skin color, motion, facial/head shape, facial appearance, or a combination of them. The most efficient methods nowadays to detect faces from a single intensity or color image are classified into four categories [112]:

1. Knowledge-based methods: These methods are based on the rules derived from human knowledge of what constitutes a typical face. The rules capture the relationships between facial features and can be represented by their relative distances and positions. Facial features are extracted first in the input image, and the candidates are identified following the coded rules. Also, a verification process is applied to reduce false detections. The principal problem with these methods is the difficulty in translating human knowledge into well-defined rules. These methods are designed mainly for face location [112].
2. Feature invariant approaches: These approaches aim to find invariant features of face for detection even when the pose, viewpoint, or lighting conditions vary. Eyebrows, eyes, nose, mouth and hair-line are commonly extracted by using edge detector to build a statistical model describing their relationships. The principal localization methods applied

nowadays for a effective segmentation apart from facial features are textures and skin color or combination of all of them. The principal problem with these approaches is that the facial features can be severely corrupted due to occlusion, illumination or noise. These methods are designed mainly for face location [112].

3. Template matching methods: In these methods, a standard face pattern is stored or parameterized to describe the face as a whole or the facial features separately. The correlations between the input image and the stored patterns are computed for detection using face contour, eyes, nose, and mouth independently. These methods are highly inadequate for face detection since they are very sensitive to variation in scale, pose, and shape. Techniques such as multiresolution, multiscale, subtemplates and deformable templates have been proposed to achieve scale and shape invariance. These methods have been used for both face localization and detection [112].

4. Appearance-based methods: These methods develop templates from training the system with faces. They rely on techniques from statistical analysis and machine learning to find the relevant characteristics of face and nonface images. They classify the images into two sub-images (face and non-face). A face/non-face classifier is constructed training the system with a face database to distinguish the regions based on pixels. Pixels in the face area are high correlated meanwhile in the non-face area present less regularity. These methods are sensitive to changes in facial appearance, lighting and facial expressions since the boundaries may become very complex to differentiate. Nonlinear classifiers are created for such cases. Amongst the most common methods we can find: eigenfaces, distribution-based methods, neural networks, support vector machines (SVMs), sparse network of Winnows(SNoW), Naive Bayes Classifier, Hidden Mark and AdaBoost algorithms [112].

Statistical learning-based methods [112] classify the faces by learning from training data to extract good features and build classification engines. The data used for the training is appearance images or features extracted from appearance. Both, prior knowledge about faces and

| Classification Methods | Representative Approaches |
|---|---|
| Knowledge-based methods | |
| –Hierarchical | Multiresolution rule-based approach [208] |
| Feature-invariant methods | |
| –Facial Features | Grouping edges [213] [110] |
| –Texture | Space Grey-Level Dependence Matrix (SGLD) of face pattern [49] |
| –Skin color | Mixture of Gaussian [209] [125] |
| –Multiple Features | Integration of skin color, size and shape [97] |
| Template Matching Methods | |
| –Predefined Face Templates | Shape Template [47] |
| –Deformable Face Templates | Active Shape Model (ASM) [106] |
| Appearance-based Methods | |
| –Eigenface | Eigenvector decomposition and clustering [186] |
| –Distribution-based | Gaussian distribution and Multilayer perceptron [179] |
| –Neural Network | Ensemble of neural networks and arbitration schemes [154] [211] |
| –Support Vector Machine (SVM) | SVM with polynomial kernel [139] |
| –Naive Bayes Classifier | Joint Statistics of local appearance and position [164] |
| –Hidden Markov Model | Higher order statistics with HMM [147] |
| –Information Theorical Approach | Kullback relative information [111] |

**Table 4.1:** Classification of Methods for Face Detection

variations in the training data are taken into account for the learning. The principal mathematical methods to divide the image through statistical analysis are:

- Principal Component Analysis (PCA): It is a mathematical procedure derived from Karhunen - Loeve's transformation [201]. It transforms a given s-dimensional vector representation of correlated variables of each face in a training set of images, into a smaller number of uncorrelated variables called principal components. These principal components are a t-dimensional subspace whose basis vectors correspond to the maximum variance direction in the original image space. If the elements in an image are considered random variables, the PCA principal components are defined as eigenvectors of the scatter matrix [186].

- Linear Discriminant Analysis (LDA): It is closely related to PCA since both look for linear combinations of variables which best explain the data [126]. However, LDA attempts to model the difference between the classes of data while PCA, on the other hand, does

not take into account any difference in class, and the factor analysis builds the feature combinations based on differences rather than similarities. LDA discerns the vectors in the underlying space that best discriminate among classes. For all samples of all classes the between-class scatter matrix SB and the within-class scatter matrix SW are defined. The goal is to maximise SB while minimizing SW, in other words, maximise the ratio $det|SB|/det|SW|$ . This ratio is maximised when the column vectors of the projection matrix are the eigenvectors of $(SW^{-1} * SB)$ [58].

- Independent Component Analysis (ICA): It is a computational method for separating multivariate signals into additive subcomponents. It minimises both second-order and higher-order dependencies in the input image and attempts to find the basis along which the data is statistically independent [12].

### 4.2.3 Feature Extraction

Once the face detection step has extracted the facial components, the image is normalised based on the located points regarding the geometrical properties, following geometric transformations and morphing. Often, the image is further normalised following photometric transformations such as grey scale and illumination. With the image normalised, the second step is applied. Feature extraction is performed to extract distinctive characteristics of each face to distinguish faces of different people [84].

In this step, facial features hold an important relevance. Many recognition systems need facial features in addition to the face detection system. Facial human features can be of different types: region [157], key point (landmark) [197] and contour [45].

Three main kind of facial feature extraction methods can be distinguished as follows [7]:

1. Geometric, Feature-Based methods: Generic methods based on edges, lines, and curves. Faces can be recognised even when the details of the facial features are no longer resolved.

The information left is geometrical, and the methods extract relative position and other parameters of distinctive features such as eyes, mouth, nose, and chin [88] [18] [69].

2. Feature Template-based methods: The face in the image is compared using a suitable metric (typically the euclidean distance) with a single template representing the whole face. These methods are used to detect facial features such as eyes and they have difficulty when the appearances of the features change significantly (closed eyes, eyes with glasses, open mouth, etc) since each feature is characterised with a fixed template. Thus this technique suffers of a strong scale and poses dependency [214] [27].

3. Structural template matching methods: They take into consideration geometrical constraints on the features. A novel model is the Active Shape Model which is much more robust in terms of handling variations in image intensity and feature shape [45] [94].

4. Color Segmentation techniques: These methods use the skin color to isolate the face. Any non-skin color is a candidate for a facial feature such as eyes, mouth, etc [36].

5. Appearance-based approaches: They extract characteristics from the image which are not simply eyes or mouth. Amongst these methods, we can find PCA, ICA or Gabor-wavelets [186] [12] [184].

Regarding facial feature extraction based on templates, the eyes are the most important feature in the face, due to several reasons [7]:

- Eyes are crucial to establish the state of human beings.
- Eye appearance is less affected by temporal changes such as facial hair, transparent spectacles, ageing, etc.
- Eyes position allows to locate and identify the face scale and its in-plane rotation.
- Accurate eye location enables to identify other face features of interest for face recognition.

Due to this evidences, normally a feature extraction method is evaluated following its performance in terms of error measures only when extracting eyes location [32].

### 4.2.4 Face Recognition Techniques

The last step in a face recognition system is the face matching. In this step, the extracted feature vector of the initial image is matched against the faces in the database [112]. The result will be an identified face is the matches are confident enough or an unknown face on the other hand. Face recognition techniques can be classified into two broad categories [27]: analytic or feature-based approaches and holistic or appearance-based methods. The analytic approaches [69] extract a set of geometrical features from the face such as eyes, nose, mouth, etc., and together with the face outline form a feature vector. This vector, which includes properties and relations such as areas,distances and angles between feature points, is used to find a candidate from a face database [87] [46] [123].

The holistic methods take into consideration the global properties of the human face pattern. The face is recognised as a whole without using only certain points obtained from different face features. In general, these methods operate directly on pixel level without the facial features detection step. Holistic methods use techniques to transform the image into a low-dimensional feature space with enhanced discriminatory power [177]. Basically, they project an image into a subspace and find the closest pattern. These methods present very good results in standard, well-illuminated frontal face images. The major appearance-based algorithms are the following:

- Eigenfaces: It is one of the most used and investigated approaches. It uses principal component analysis (PCA) to represent the faces [186]. Eigenfaces are the principal components of the distribution of faces. The eigenvectors are the covariance matrix of the set of faces and they are ordered to represent different amounts of variation among the faces. Each face is represented by a linear combination of the eigenfaces. Illumination

normalisation is normally necessary for this approach. The most prominent method are the one developed by [177] and [186].

- Neural Networks: The advantage of using this method is due to its nonlinearity in the network. Therefore, the feature extraction may be more efficient that linear methods such as eigenfaces. The main disadvantage appears when the number of individuals increases because the computational cost will increase. They also need multiple model faces per person in order to train the system [216] [96].
- Geometrical Feature Matching: They are based on a set of geometrical features of a face. They use a vector to represent the location and the size of the main facial features. One of the pioneering works was done by [87].
- Graph Matching: M. Lades et al. [105] presented a dynamic link structure for distortion invariant object recognition, which employed elastic graph matching to find the closest stored graph. This method is an extension to classical artificial neural networks.
- Fisherface: This method was proposed by Belhumeur et al. [15], which uses PCA and Fishers linear discriminant analysis to produce subspace projection matrix that is very similar to that of the eigenfaces method. However, this approach minimises the variation within each class, yet maximising class separation, so the problem with variations in the same images such as different lighting conditions can be overcome.

Face Recognition has always been a very popular topic in the research field because its applications may be very useful for personal verification and recognition. On the other hand, it also has been a very challenging task because of the innate difficulties to implement a reliable system due to all different situations that human faces can be found in. In the last twenty years, many diverse techniques have been developed in an attempt to create a steady and robust system that could be used in a huge number of situations.

| **Approach** | Summary | Performance |
|---|---|---|
| Bayesian Eigenfaces [142] [128] | Generalisation of the Principal Component Analysis (PCA) approach by examining the probability distribution of intra-personal variations in appearance of the same individual and extra-personal variations in appearance due to difference in identity. Going from eigenfaces to modular eigenfeatures that correspond to face components, such as eyes, nose, and mouth (referred to as eigeneyes, eigennose, and eigenmouth). | Performance with FERET database:<br>Pose: Frontal training view well up to 45 degrees rotation<br>Illumination: Handled well<br>Expression: Problems with screams, deformation of the mouth and eye narrowing<br>Occlusion: Less sensitive to upper face occlusion<br>Recognition rate of 95% on the FERET database. |
| Discriminant Analysis [58] [215] | Method based on combining PCA and LDA to distinguish the different roles of within- and between-class scatter by applying discriminant analysis. The method consists of two steps: first the projection of the face image from the original vector space to a face subspace via PCA, second, via LDA, obtaining the best linear classifier. | Performance with FERET database:<br>Pose: Frontal training view well up to 45 degrees rotation<br>Illumination: Handled well<br>Expression: Problems with screams, deformation of the mouth and eye narrowing<br>Occlusion: Less sensitive to upper face occlusion<br>Recognition rate of 95% on the FERET database. |
| Discriminant Eigenfeatures [215] | Face recognition system which uses automatic selection of features from an image training set using the theories of multidimensional discriminant analysis and the associated optimal linear projection. This MDF space discounts factors unrelated to classification, such as lighting direction and facial expression when such variations are present in the training data. | |
| Dynamic link architecture [105] | This method represents individual faces by a rectangular graph, each node labeled with a set of complex Gabor wavelet coefficients, called a jet. A jet is used to represent the local features of the face images based on the Gabor wavelet transforms. Only the magnitudes of the coefficients are used for matching and recognition. For the recognition of a new face, each graph in the database is matched to the constructed graph of the new image separately and the best match indicates the recognised person. Rotation in depth is compensated for by elastic deformation of the graph. | It performs very well in terms of invariance to rotation. However, the matching process is computationally expensive. |

| **Approach** | Summary | Performance |
|---|---|---|
| Elastic bunch graph matching (EGM) [197] | This method extends the dynamic link architecture method in order to increase the matching accuracy for bigger databases and handle larger variations in poses. EGM uses the phase of the complex Gabor wavelet coefficients to achieve a more accurate location of the nodes and to disambiguate patterns. The goal of EGM on a test image is to find the fiducial points and thus extract from the image a graph that maximises the similarity. The morphological elastic graph matching has been proposed for improvement. | Performance with FERET database:<br>For frontal against frontal images, the recognition rate was very high (98% for the first rank and 99% for the first 10 ranks) compared with matching of profile images.<br>EGM-based systems have good performance in general. However, they require a large-size image, which restricts the application to video-based surveillance. |
| Fisherfaces [15] | This method improves the performance of direct PCA approach by applying first PCA for dimensionality reduction and then Fisher's linear discriminant analysis. | Fisherface algorithms are believed to outperform eigenface methods, since LDA extracts features more suitable for classification purposes. On the other hand, Fisherfaces require multiple images for training for each person, which is not always available for some applications. It is weak against different illumination and head pose. It has a rate of recognition of 96% in perfect conditions. |
| Independent component analysis [12] | This method seeks non-orthogonal basis that are statistically independent while PCA finds a set of orthogonal basis for face images of which the transformed features are uncorrelated. The basis images developed by PCA depend only on second-order image statistics. ICA generalizes the concept of PCA to higherorder image statistics relationships. | ICA method, when carried out in the properly compressed and whitened space, performs better than the Eigenfaces and Fisherfaces methods, but its performance deteriorates significantly when augmented by an additional discriminant criteria such as the FLD. |
| Line edge map [66] | The line edge map (LEM) approach extracts lines from a face edge map as features, based on a combination of template matching and geometrical feature matching. Similarity of face images can be measured by a face feature representation scheme based on the LEM.The faces are encoded into binary edge maps using the Sobel edge detection algorithm. The Hausdorff distance was chosen and is calculated without an explicit pairing of points in their respective data sets. | The LEM method possesses the advantages of a feature-based approach, which is invariant to illumination, has low memory requirements, and shows high recognition performance using template matching. |
| Hybrid Neural Networks [116] [107] | These methods combine local image sampling, a self-organizing map (SOM) neural network and a convolutional neural network. SOM provides a quantization of the image as well as partial invariance to translation, rotation, scale and deformation in the image sample. | These methods perform extremely well when it comes to recognise one single person. However, when the number of people increases, the computation expenses become more demanding.<br>Multiple model images per person are necessary for training the system. |

| **Approach** | Summary | Performance |
|---|---|---|
| Support Vector Machines [80] [79] | SVMs are an effective pattern classifier. They find the optimal separating hyperplane that maximises the margin of separation in order to minimise the risk of misclassification not only for the training samples, but also the unseen data in the test set. A SVM classifier is a linear classifier where the class separating hyperplane is chosen to minimise the expected classification defined by a weighted combination of a small subset of the training vectors, called support vectors. | SVMs operate in an induction principle called structural risk minimisation. Structural risk minimisation aims at minimising an upper bound on the expected generalisation error. Compared with a standard PCA method, the SVM verification system was found to be significantly better. |
| Embedded Bayesian Network(EBN) [135] [137] | EBN is a generalisation of the embedded hidden Markov models that are a set of statistical models used to characterise the statistical properties of a signal. An EBN is defined recursively as a hierarchical structure where the ”parent” node is a Bayesian network (BN) that conditions the EBNs or the observation sequence that describes the nodes of the ”child” layer. | This method has more flexibility in terms of natural face variations, scaling, and rotations, while significantly reducing the complexity of the fully connected 2D HMM. |
| Volterra Faces [104] | Face images are spatially arranged as image patches. This method has a smooth non-linear functional mapping for the corresponding patches such that in the range space, patches of the same face are close to one another, while patches from different faces are far apart, in L2 sense. For this, Volterra kernels are used. It tries to minimise intraclass distances while maximising interclass distances. | The Volterra kernel computation reduces the generalized eigenvalue problem which translates to a very efficient computation of kernels for any order of approximation of the functional. |

**Table 4.2:** Comparison of major algorithms for face recognition

In this chapter, the different techniques used to test the reliability of the proposed CAPTCHA methods against machine attacks are described. Firstly, the SIFT technique used to recognise objects and characters is presented, along with a general overview of its functionality. And secondly, a general state-of-the-art review of the most used and important face recognition techniques is presented since the experiments for the image CAPTCHA approach are based on face image recognition.

# Chapter 5

# Human Perception and Recognition

The human friendliness of the novel CAPTCHA approaches that are presented in this thesis depends on a number of factors related to how human beings perceive and understand characters, images, and objects. It is essential that the CAPTCHA tests can be easily passed by the human users to avoid wasting their time, being boring, or what is worse, getting the users to avoid using web services that require human verification. For the CAPTCHA approaches developed, an exhaustive study on human perception and recognition has been done.

In the following sections, Gestalt psychology theory and human face perception and recognition are described. The Gestalt psychology is used for the creation of the shadow character CAPTCHA approach since it explains how human beings are capable of recognising shapes, objects and characters as a whole even if their outline is incomplete. The human face perception and recognition theory is used for the creation of the second approach based on a face recognition CAPTCHA method and it explains how human beings recognise faces as a natural social and cultural process and the brain capabilities for doing so are more developed than for recognising objects.

## 5.1 Gestalt Psychology

Understanding human psychology is an important factor to create successful CAPTCHAs. In this research, we have focused in a particular branch called Gestalt psychology. The concept of gestalt, German word for ``essence or shape of an entity's complete form´´, was first introduced in contemporary psychology by Christian von Ehrenfels following the theories of David Hume, Johann Wolfgang von Goethe, Immanuel Kant, David Hartley, Ernst Mach and Max Wertheimer. Later on, Fritz Perls, Laura Perls and Paul Goodman created the gestalt therapy by bringing together the diverse European and American theories and backgrounds to synthesise a new psychotherapy and social theory [21].

The main principle of Gestalt psychology focuses on the concept that the brain is holistic, parallel, and analog, with self-organising tendencies. This principle maintains that ”The whole is other than the sum of the parts” [101], and the human eye sees objects in their entirety before perceiving their individual parts. The gestalt effect stipulates that perception is the product of complex interactions among various stimuli and it depends on the form-generating capability of our senses to perceive whole forms instead of just a collection of simple lines and curves [85].

At the beginning of the 20th century, the school of Gestalt practised a series of theoretical and methodological principles that attempted to represent the subjective experience of perception such as follows [109]:

- Principle of totality states that the conscious experience should be viewed holistically, as a totality of the dynamic interactions of components of the brain.
- Principle of psychophysical isomorphism states that there is a correlation between the perceptual phenomena and the activity in the brain.
- Phenomenon experimental analysis states that any psychological experiment should have as a starting point a phenomena and not sensory qualities.

- Biotic experiment states the need of conducting real experiments on natural situations and real conditions to reproduce with higher fidelity the habitual situations of the subjects.

The perceptual process exhibits four key gestalt properties such as emergence, reification, multistability, and invariance. the ubiquity of these properties in every aspect of perception suggests that gestalt phenomena are fundamental to the nature of the perceptual mechanism [109].

**Emergence** Emergence is the process of complex pattern formation from simpler rules. The main characteristic is that the final global form is not computed in a single pass but continuously. The best example can be observe in Figure 5.1. In the picture, the local regions of the image do not contain enough information to distinguish significant form contours from insignificant noisy edges, but as soon as the subject is recognised, the perception of a dalmatian dog is very vivid despite the fact that much of its perimeter is missing. The Gestalt theory does not offer any specific computational mechanism to explain this property in visual perception [109].

**Figure 5.1:** Example of the Gestalt property emergence.

**Reification** Reification is the constructive or generative principle of perceptual processing, by which the final form is perceived by filling-in of a more complete and explicit perceptual entity based on a less complete visual input. The Kanizsa figure shown in Figure 5.2 is one of the most famous illusions produced by the Gestalt theory. In the figure, a triangle can be

recognise by filling-in perceptually, and producing visual edges in places where there are none in the input [109].

**Figure 5.2:** Example of the Gestalt property reification showing the Kanizsa triangle and a snake.

**Multistability** Multistability refers to the visual process of perception. Perception must involve some kind of dynamic process whose stable states represent the final percept. One famous example is the Necker cube, shown in Figure 5.3, where a prolonged viewing of the picture produces spontaneous reversals, and the final percept depth is inverted [109].

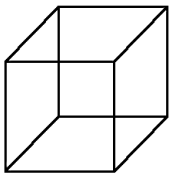

**Figure 5.3:** Example of the Gestalt property multistability.

**Invariance** Invariance is the property in which an object can be recognised regardless of its rotation, translation, scale, change of lighting or background, or texture and motion [109](see Figure 5.4).

The Gestalt principles of perception come from the law of prägnanz (german for language), and describe the organization of perceptual scenes. The law of prägnanz says that when we look at the world we usually perceive complex scenes composed of many groups of objects on some background, with the objects themselves consisting of parts, which may be composed of smaller parts, etc., and we tend to order our experience in a manner that is regular, orderly,

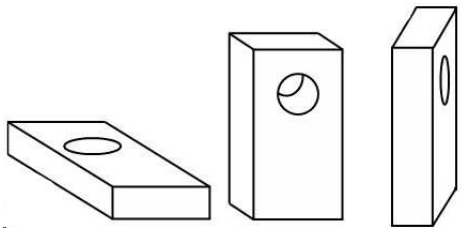

**Figure 5.4:** Example of the Gestalt property invariance.

symmetric, and simple. The interpretation of sensation that comes from the perception are called the ``Gestalt laws´´ [56]. They are as follows:

**Law of Proximity** The law of proximity states that elements tend to be perceived as aggregated into groups if they are near each other. For example, in Figure 5.5, the first row is perceived as a sextuplet while the second and third row, due to the change of distance between some of the components, the patches are perceived not just collectively as a sextuplet, but also as being subdivided into groups.

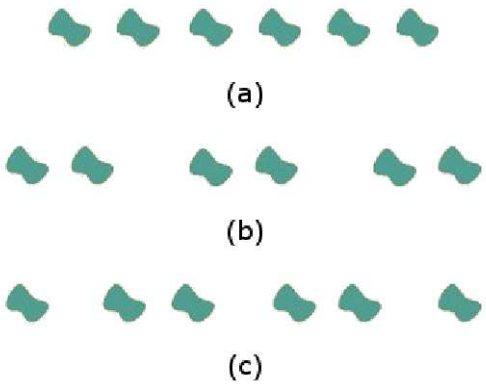


**Figure 5.5:** Example of the Gestalt law of proximity.

**Law of Similarity** The law of similarity states that elements tend to be integrated into groups if they are similar to each other. This similarity can occur in the form of shape, colour, shading or other qualities. For example, as shown in Figure 5.6, the shapes have a constant distance between them, but they are perceptually partitioned into three adjacent pairs, due to the sim-

ilarity of visual attributes such as lightness (first row), colour (second row), size (third row), orientation (forth row), or shape (fifth row).

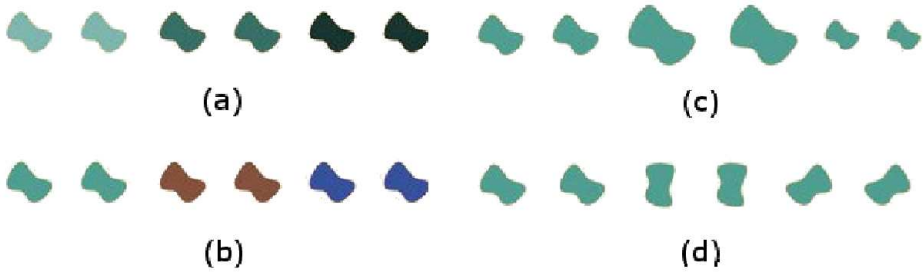


**Figure 5.6:** Example of the Gestalt law of similarity.

**Law of Symmetry** The law of symmetry states that the mind perceives objects as being symmetrical and forming around a centre point. When two symmetrical elements are unconnected the mind perceptually connects them to form a coherent shape. Similarities between symmetrical objects increase the likelihood that objects will be grouped to form a combined symmetrical object.

**Law of Common Fate** The law of common fate states that elements tend to be perceived as grouped together if they move together. We perceive elements of objects to have trends of motion, which indicate the path that the object is on. For example, if there are an line of dots and half the dots are moving upward while the other half are moving downward, we would perceive the upward moving dots and the downward moving dots as two distinct units.

**Law of Continuity** The law of continuity states that oriented elements or groups tend to be integrated into perceptual wholes if they are aligned with each other. In cases where there is an intersection between objects, individuals tend to perceive the two objects as two single uninterrupted entities. Stimuli remain distinct even with overlap. We are less likely to group elements with sharp abrupt directional changes as being one object. For example, as shown in

Figure 5.7, the principle is applied in the same way for elements arranged along lines, as well as for patterns built from corresponding lines themselves.

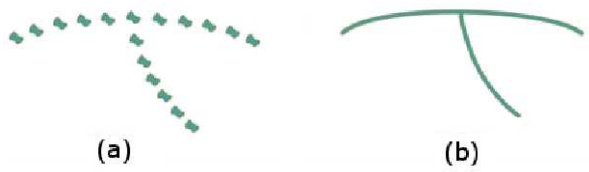


**Figure 5.7:** Example of the Gestalt law of continuity.

**Law of Closure** The law of closure states that elements tend to be grouped together if they are parts of a closed figure. Specifically, when parts of a whole picture are missing, our perception fills in the visual gap. For example, as shown in Figure 5.8, we perceive the circle and the square as a whole even thought there are gaps missing.

**Figure 5.8:** Example of the Gestalt law of closure.

**Law of Good Gestalt** The law of good gestalt explains that elements tend to be grouped together if they are parts of a pattern which is a good Gestalt, meaning as simple, orderly, balanced, unified, coherent, regular, etc as possible, given the input. This law implies that as individuals perceive the world, they eliminate complexity and unfamiliarity in order to observe a reality in its most simplistic form. The law of good gestalt focuses on the idea of conciseness which is what all of gestalt theory is based on.

**Law of Past Experience** The law of past experience implies that elements tend to be grouped together if they were together often in the past experience of the observer. If two objects tend to be observed within close proximity, or small temporal intervals, the objects are more likely to be perceived together. For example, as shown in Figure 5.9, we perceive the stroke as roman characters, even when there is change in colour, distance, or distortions.


**Figure 5.9:** Example of the Gestalt law of past experience.

## 5.2 Human Face Perception and Recognition

Face Perception is a cognitive process that humans perform easily and with high rate of success. This process uses the brain and the mind [1] to comprehend and interpret the human face. The theoretical starting point for human face recognition is that we have image-based templates in our memories and the process of recognition activates the corresponding template for the face/object that wants to be recognised. However, it is almost impossible for the brain to cover all available faces in the world and convert them into templates, so breaking them into features makes the problem easier to solve [25].

Face perception involves diverse processes that occur in distinct areas of the brain. The result of evolution and functional differentiation have place the areas in the temporal lobe known

[1]The brain is the part of the central nervous system situated within the skull, meanwhile, the mind is the term used to describe the higher functions of the human brain, which are subjectively conscious.

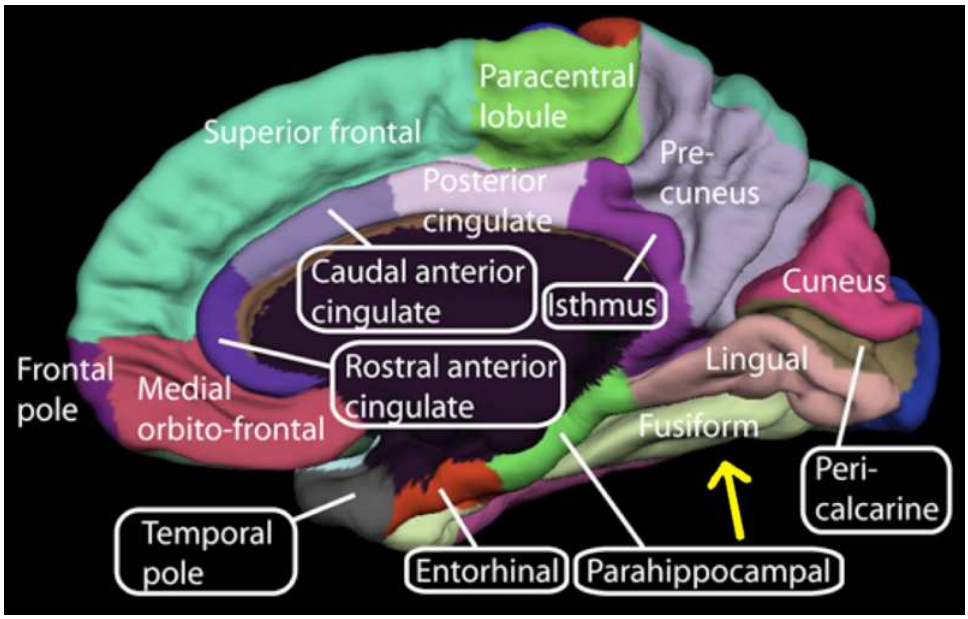


**Figure 5.10:** Medial surface of cerebral cortex by Hagmann P, Cammoun L, Gigandet X, Meuli R, Honey CJ, et al.

as fusiform gyrus and as well as another cortical areas such as the inferior temporal gyri [90]. These areas are more active during face viewing than object viewing and they are profoundly related to the social, emotional and associative tasks that the brain performs. This may be related to the need of nurturing social and emotional skills since birth. An example of how the brain is divided can be seen in this Figure 5.10.

Sex gender and race are also important matters when it comes to face recognition. Studies have demonstrated [60]that male subjects used the right hemisphere activation system, while the female subjects used the left one. Furthermore, face recognition performance in female individuals was not associated to estimated intelligence or several basic cognitive processes, which may suggest a role for sex hormones. Women tend to recognise more women faces than men, albeit there are no difference with male faces [148].

The importance of other-race experience in own-versus other-race face processing has been further researched by [140], [178], [190]. They showed the importance of the relationship between amount and type of other-race contact and the ability to perceptually differentiate other-

race faces. Individuals with greater experience with other-races demonstrate more accurate answers when discriminating other-race faces than individuals with less experience.

As mentioned in the first paragraphs of this section, face perception and recognition is an experience-expectant process, which begins within the first six months of life. This field is enthusiastically studied by psychologists and neurologists in an attempt to understand the mechanism of brain activity. Faces provide an early channel of communication through discrete neural circuits. Also, it is seen as a special ability that has been selected for thorough evolutionary processes, and conserved across species. This is possible since face is the main tool for communicating in groups of increasing size. Only recently,in the scale of human evolution, human language and abstract reasoning have emerged, thus face language, to some extent, is universal [138].

Although, despite of the studies in this area, little is known about the differences between object recognition and face recognition [89], [67]. Experimental results with adult beings have suggested that faces are perceived as a special class of stimuli, separate from other patterned objects. Considering face perception ”engages a domain-specific system for processing both configural and part-based information about faces” [212], we conclude that this is needed to accommodate geometry, illumination, occlusion and disguise and temporal changes.

One of the most widely accepted theories of face perception [26] argues that understanding faces involves several stages: pictorial, structural, visually derived semantic, identity-specific semantic, name, expression and facial speech codes. This model implies that face perception involves different independent sub-processes that work in parallel. A proved method to help understand the complex functionality of the brain is the study of brain-injured or neurological ill people. Prosopagnosia [78] is the impairment in the ability to recognise familiar faces, and its commonly accompanied by brain damage. As individuals with such an impairment may have different abilities to understand faces, the investigation of this illness has helped to develop the theories of stage-face perception models [26].

Face perception is a natural ability developed by every single human being and it is used in a multitude of social functions. For that very same reason we have decided to create a new CAPTCHA test based upon face recognition, to allow people from diverse cultural backgrounds and societies take the test and pass it successfully.

In this chapter, Gestalt psychology and human face perception and recognition theories are described for the purpose of making the CAPTCHA approaches more human friendly when solving them. The Gestalt psychology section describes how human beings are capable of recognising shapes, objects and characters as a whole even if their outline is incomplete. The human face perception and recognition theory section describes how human beings recognise faces as a natural social and cultural process and the brain capabilities for doing so are more developed than for recognising objects. These theories are later used to create new types of CAPTCHAs that are easy for human to solve but difficult for machines.

# Chapter 6

# Visual Word-Based CAPTCHA

In the traditional approaches to OCR-based CAPTCHAs, affine transformations are used, i.e., rotation, scale, or shear transformations, due to its good performance against OCR software [11]. However, it is a well known fact that with specific segmentation and clustering techniques, these methods can be broken with a high rate of success [40, 73, 73, 130, 131], which can then lead to an increase of spam or malicious software in web services. To ensure the robustness and efficiency of the OCR-based CAPTCHAs there are generally two issues to be addressed; The first is the quality of the image warping techniques applied, where the distortions must avoid being too simplistic such that they can be recognised straightforwardly by a computer vision technique. The second issue is that the warping effects applied can sometimes make it harder for humans to differentiate the letters in the word. Additionally, these types of tests are often found annoying and time-consuming for most people. The aim is to create a human-friendly test that humans can pass easily.

In this chapter, a visual word-based CAPTCHA novel approach is presented. Along with the developed shadow characters, the distortion techniques applied allow an increase in robustness against machine recognition whilst preserving a fairly low difficulty in terms of human recognition. Human and machine performance are compared with state-of-the-art character

recognition software and human users. The results obtained in the novel CAPTCHA [150] are summarised. Exhaustive experiments were performed to ensure the efficiency of the approach, where one hundred human users with different social backgrounds and technological knowledge evaluated the test. For each experiment, a random CAPTCHA was designed for both the human user and the machine software.

## 6.1 Properties of the Visual Word-Based CAPTCHA based on Shadow Characters

As the work presented in this chapter primarily targets the OCR-based CATPCHA's methods, in this section the basic properties of the framework are described. The structure of the algorithms is primarily the result of trying to minimise the risk of machine recognition to decipher the test's solution or preventing an unwanted subtraction of the data from the server, and increase human usability. The former corresponds to one of the main concerns when developing a new CAPTCHA, since the purpose of the tests is to prevent the expansion of spam through web services, and the latter corresponds to the level of human skills necessary to solve the tests, such as accuracy, response time, and perceived difficulty of the user. To increase security, there are many strategies that can be used that involve distortions, background noise, and diverse types of character fonts [9]. There exists several methods that focus on occlusion and distortions [20, 44, 50] or character fonts and outlines [156, 158]. The majority of these methods can prevent OCR software recognition, but are weak against segmentation and cluttering techniques [40, 73, 73, 130, 131]. These methods were created with the only purpose of security, and due to this fact, their human usability is very low, so most users find them very annoying since they require more than one attempt to solve the test [9].

The approaches presented here target a new type of characters that cannot be recognised by computer software. The characters are developed by emulating their shadows that are created

by a frontal lighting source. So instead of using a font to represent them, they are outlined by their shadows producing only discontinuous lines that the human eye can put together and recognise as a word or a set of random letters and numbers, whilst machines cannot. The main goal of using this brand new characters is their robustness against machine attacks and their human usability since the level of distortions needed is low. In the following paragraphs, the main stages of the algorithms are explained. In the CAPTCHA methods presented in this thesis, it was important to develop a secure and effective method since the CAPTCHAs should be created in real-time and in an automated way. The framework is composed as follows:

1. CAPTCHA development: CAPTCHA programs are made with the purpose of avoiding machines to break through web systems but at the same time, allow people to access them easily. Humans have the capacity of recognising a multitude of objects in images with little effort, despite the fact that every object in the image may vary in diverse ways: point of view, scaling, translations, small rotations, etc. Computer vision algorithms are robust in recognising features and objects even if they are rotated, translated or distorted. However, they are weak when the pattern of the object or the feature is partially obstructed or it is not subdued to a boundary. The aim of the CAPTCHAs is increase the gap between what a human can recognise and a computer vision program cannot. To increase this gap, we developed characters defined only by their shadows. This new type of letters and numbers are recognised with little effort by humans but in terms of machines it becomes a serious issue (see Figure 6.1).

2. Shadows characteristics: The shadows are created by positioning, in an imaginary 3D space, a frontal lighting source that creates a 45°shadow, from top left to bottom right. The shadow is always the same. In terms of human recognition, that is the easiest recognisable kind of shadow, since human beings that read latin alphabet, read from left to right, and from top to bottom.

3. Distortion techniques: Many computer vision techniques can solve the test by comparing features extracted from a database. To avoid this problem we introduce distortions to our

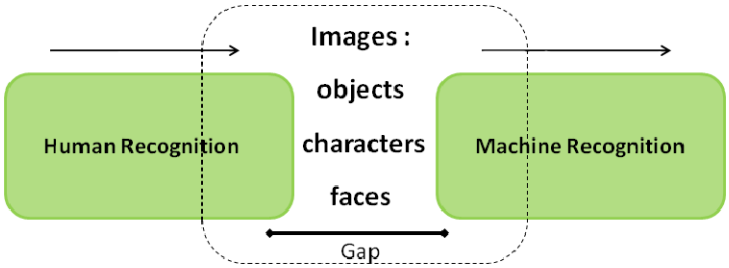


**Figure 6.1:** Scheme of the recognition gap between humans and machines. The aim is to increase human recognition, decrease machine recognition and increase the gap between the two of them.

images in real-time. In this process, characters are distorted randomly following geometric transformations. In our approaches we used affine and perspective transformations and warping. The characters are distorted separately following a uniform probability function. The effects created are limited to prevent excessive distortions, and the range of the distortions are limited between 0.2 to 0.6, being the complete range between 0 and 1. The upper limit was defined by the results obtained by human users, since the main purpose of this research is to create more human friendly CAPTCHA approaches, and the bottom limit was defined to avoid creating effects easy recognisable by machines.

4. Rendering into a 2D image: In our approach we have used light functions in the coding to make the process of rendering easy and with low computational cost. The creation of the shadows is done by variating the camera view points and the lighting sources for the 3D objects, whilst for the 2D characters is fixed.

## 6.2 Visual Word-Based CAPTCHA based on 2D Shadow Characters

The concept of shadows characters is introduced here. The first step for the creation of the new letters and numbers was to discover what kind of characters could be developed to improve the

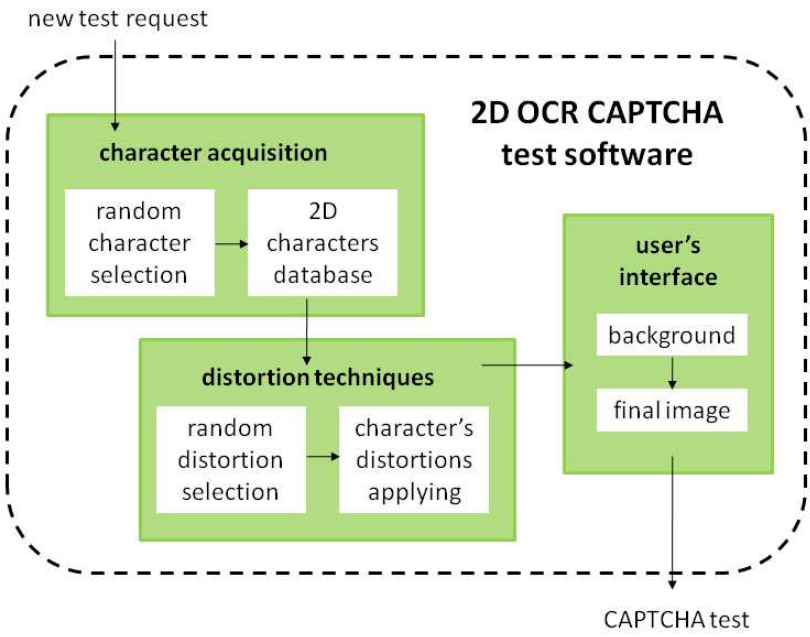


**Figure 6.2:** Block diagram of the CAPTCHA based on 2D shadow characters.

currently available programs. The idea came from analysing human psychology and perception and seeing how human beings can perceive disconnected parts as a whole [101]. Aiming to increase the gap between human recognition and machine recognition, the shadow characters were developed. This first algorithm was aiming to overcome the weaknesses of the available OCR tests and produce an increment in human usability.

The visual word-based CAPTCHA based on 2D shadow characters method is based upon two dimensional characters defined by their shadows. Figure 6.2 shows a block of the approach. The following is a detailed explanation of the whole procedure:

1. An image database was created, composed by letters and numbers (see Figure 6.3). The images were designed with a white background and the characters were drawn by projecting the shadows to define the contours. The main advantage of doing so is that the character boundary is not entirely defined as a whole. Thus, the character recognition process gets more difficult for the machines but remains easy for humans [101].

**Figure 6.3:** Examples of 2D shadow characters.

2. Once the database was finished, the software of the CAPTCHA test was developed. The program randomly selects six letters or numbers from the database following a uniform distribution function. After the characters are selected, the program randomly distorts each character separately. When the six characters are distorted, they are put together in the final image and the background is added.

3. The final step is the user's interface development. This interface is divided into two parts. The upper section contains the image with the characters along with a button to change the displayed image, in case it is too difficult to solve. The bottom section contains a text box to introduce manually the characters displayed and a button to submit the final answer. If all the characters submitted match perfectly the characters in the image, the user passes the test if not, the user fails it.

## 6.3 Visual Word-Based CAPTCHA based on 3D Character models

The visual word-based CAPTCHA based on 3D characters method goes one step further in the development of the shadows characters. In this algorithm, instead of using plain characters, a database with 3D models was developed [150]. The models represent the letters and numbers in the English alphabet and they are used to create the shadow characters by applying lighting effects and different camera view points. To make independent shadows, in each character, lighting effects are applied randomly for each model to provide different perspectives of the shadows. These effects are done by positioning the camera focus and the light source in differ-

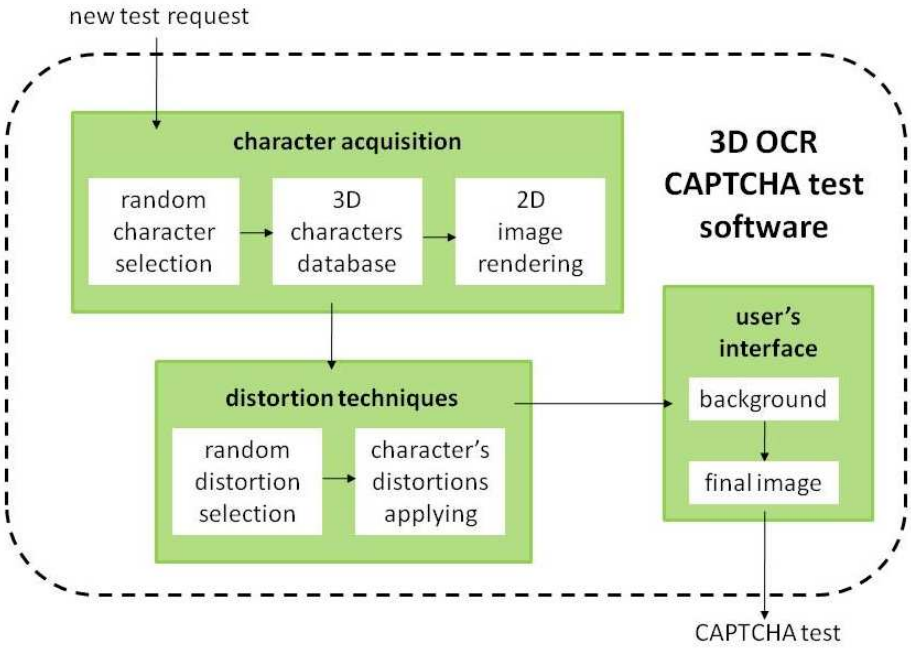


**Figure 6.4:** Block diagram of the CAPTCHA based on 3D characters.

ent points in the 3D world. To prevent excessive distortions and unusual light effects, the space range is limited in both cases.

The lightning source and the camera focus are always frontal and positioned within a limited range that never exceeds the dimensions of the models. These positions were obtained through diverse type of experiments to see the kind of shadows produced by the approach. These experiments resulted in weird shapes once the lightning source and the camera focus were not frontal and in the same plane in the 3D space. The range of movement is between 0.2 and 0.8, for both the camera and the lightning source, being the complete range between 0 and 1 along the x and y axis. The maximum space separation allowed between both of them is set to be 0.5, yet again, to limit the creation of non recognisable shapes. As with the previous approach, the distortions applied are also constricted between 0.2 to 0.6, being the complete range between 0 and 1. The upper limit was defined by the results obtained by human users, since the main purpose of this research is to create more human friendly CAPTCHA approaches, and the low limit was defined to avoid creating effects easily recognisable by machines.

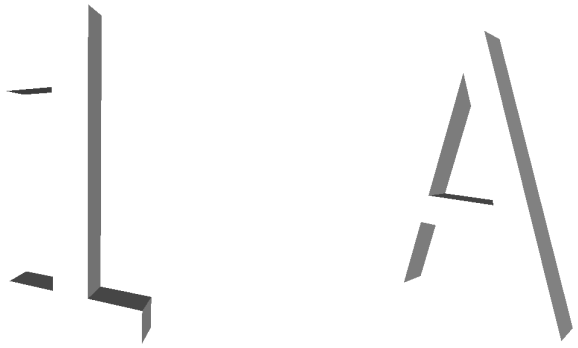

**Figure 6.5:** Examples of 3D model characters.

Figure 6.4 shows a block of the approach. The steps taken for the development of the program are as follows:

1. A database of 3D models of letters and numbers was created. The models have a white surface, no texture, and a white background in the 3D world to allow creating the shadow effects by placing the lighting source and the camera viewpoint in different positions (see Figure 6.5).

2. To use the models for the CAPTCHA tests, it is necessary to render the 3D characters into 2D images using a low computational rendering process. In the rendering process, the lighting effects and the viewpoint are defined, as well as the position of the 3D models in the 3D world.

3. The characters are again randomly selected using an uniform distribution function. Once the 2D image is obtained, the random distortions are applied separately to each image. Finally, all the characters are put together, adding also the final background.

4. The user's interface works following the same guidelines as the one created for the visual word-based CAPTCHA based on 2D characters method.

Figure 6.5 shows two examples of the 3D characters rendered into 2D images. The image representing the number "1" is defined by five lines. A person can, with barely no trouble, differentiate and connect those lines and form mentally the meaning within it [101]. When it

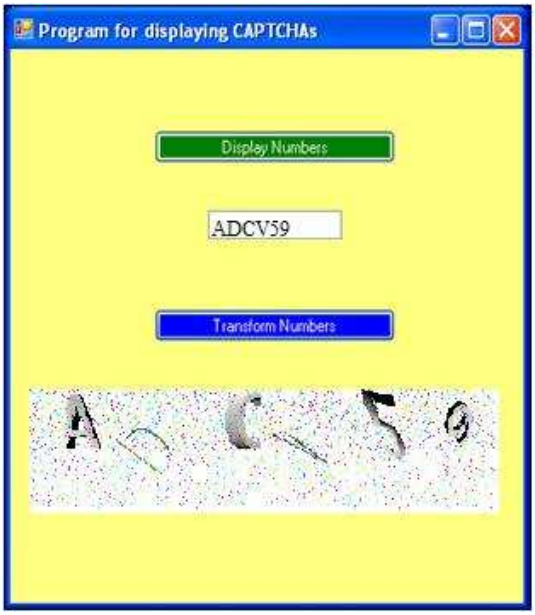


**Figure 6.6:** User's interface for Visual Word-Based CAPTCHA based on 3D Character models

comes to visual perception, most people can form a mental image of what they are seeing or at least the corresponding shape [195]. Also, the perception depends on the person's experience. If there is no grounding, that person may literally not perceive it. As not everybody perceives every object the same way, the focus is to create images that are based on common knowledge and easy to recognise. On the other way, a machine will not recognise what those five lines mean using the current algorithms, such as feature recognition and pattern recognition software.

This method is very flexible because the lighting effects may vary in lot of different ways. Also, depending on the camera viewpoint, one light source can cause different types of shadows. Albeit, this does not affect human recognition because of the restrictions made to the light and camera sources.

## 6.4 Performance analysis of the Visual word-based CAPTCHA approaches

In this section, the performance of the proposed approaches to increase security in web applications is presented. The developed methods allow an increase in Internet security and human usability. The evaluation is done by using optical recognition programs as well as the computer vision technique SIFT. The robustness and effectiveness of the approaches are shown through a comparison with the results obtained by the computer vision techniques and the results obtained from the human visual recognition [150]. Exhaustive experiments were performed under different setups, with the main purpose of comparing the results obtained with the available CAPTCHA methods. The experiments are performed using one hundred different images for both methods. The same images are used for the programs and the people.

The evidence presented demonstrates that the visual word-based CAPTCHAs are capable of handling the web applications' security requirements, and improve the results obtained by the existing OCR-CAPTCHAs that can be found on Internet. In this thesis, the principal objective is to increase the efficiency in terms of human recognition while improving the robustness in order to prevent computers from solving the tests.

### 6.4.1 Visual Word-Based CAPTCHA based on 2D Shadow Characters

For these experiments, different kind of OCR programs over images without distortions has been used. The images had different types of backgrounds to make it more difficult to the recognition programs. The images consist of six random characters and the backgrounds consists on three types: blank background, random points background and lines background. The points and the lines that appear in the images were created in a random manner to avoid the creation of recognisable patterns. The OCR programs are: Microsoft Image OCR, FreeOCR.net and Open OCR cuneiform. Performance of the visual word-based CAPTCHA based on 2D

**Figure 6.7:** Example of a visual word-based CAPTCHA based on 2D shadow characters with a blank background.

shadow characters has been evaluated for all these OCR softwares and here the results are presented. The results are summarised by calculating an average recognition percentage obtained by the different OCR software, since the results are very similar amongst all the OCR software. This set of experiments targets only the robustness against machine attacks, whereas human usability will be studied in following sections.

The first set of images are those with blank background. For this test a simple white background has been used as it gives more facility for the OCR software when recognising. An example of an image used is visualised in Figure 6.7. The results obtained show that the boundaries made by the shadows are no so easily recognised by the OCR programs. Most of the time, the characters are not recognised and the programs return garbage values. Characters as ”M” or ”W” can be recognised since their shadows represent over 90% of their surface(see table 6.1).

In the second set of experiments, backgrounds with points were introduced. An example of an image used is visualised in Figure 6.8. The aim of introducing a different background was to increase the difficulty for OCR software when recognising characters, due to the interference with the points at the boundaries of the characters. The results obtained show that the recognition percentage obtained by the OCR programs is reduced in the images with points in the background. An average of 99% of the time, the guess made by the programs returns zero character recognition(see table 6.2).

| Number of characters recognised | Average recognition percentage |
|---|---|
| 0 | $\geq 98\%$ |
| 1 | 1% |
| 2 | $\leq 1\%$ |
| 3 | 0 |
| 4 | 0 |
| 5 | 0 |
| 6 | 0 |

**Table 6.1:** Results obtained with visual word-based CAPTCHA based on 2D shadow characters with blank background. Most of the time the characters are not recognised and the programs return garbage values.

**Figure 6.8:** Example of a visual word-based CAPTCHA based on 2D shadow characters with a background with points.

In the third and final set of experiments, backgrounds with lines were introduced. An example of an image used is visualised in Figure 6.9). In this method, the main focus is to increase the difficulty for OCR software when recognising characters and compare results with the ones obtained with images with points in the background. The results obtained show that the recognition percentage obtained by the OCR programs is also reduced in comparison with the blank background, but show no improvement in comparison with the results obtained from the second set of experiments. The fact that the lines are horizontal and interfere with the boundaries causes the programs to make wrong guesses when it comes to distinguish characters such as ”F” and ”T” (see table 6.3).

According to the results, the presented approach performed well as it does not allow OCR programs to recognise the characters in our tests. The results showed that even though there

| Number of characters recognised | Average recognition percentage |
|---|---|
| 0 | $\geq 99\%$ |
| 1 | $\leq 1\%$ |
| 2 | 0% |
| 3 | 0 |
| 4 | 0 |
| 5 | 0 |
| 6 | 0 |

**Table 6.2:** Results obtained with visual word-based CAPTCHA based on 2D shadow characters with background with points. The recognition percentage obtained by the OCR programs is reduced due to the interference with the points at the boundaries of the characters.

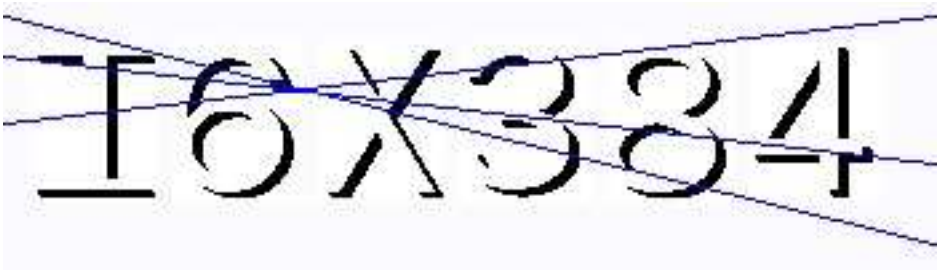

**Figure 6.9:** Example of a visual word-based CAPTCHA based on 2D shadow characters with a background with lines.

were cases that the OCR software could recognise one character, in none of the experiments, the computer software could pass a single test. As presented in chapter one, there have been many attempts to break the visual word-based CAPTCHAs by using optical character recognition. They were successful due to the simplicity of the characters and the distortions. By creating the shadow characters, we have managed to avoid OCR machine recognition, but to avoid recognition done by segmenting and cluttering it is necessary to include distortions and evaluate the results with a more complex object recognition software.

Once the distorting techniques were applied to create different examples, SIFT method was used to evaluate the effectiveness and robustness of the final approach. The recognition process uses matching between the image features and a database features that have known objects. To evaluate the images with SIFT, the database created for the method was used to extract the

| Number of characters recognised | Average recognition percentage |
|---|---|
| 0 | $\geq 99\%$ |
| 1 | $\leq 1\%$ |
| 2 | 0% |
| 3 | 0% |
| 4 | 0% |
| 5 | 0% |
| 6 | 0% |

**Table 6.3:** Results obtained with visual word-based CAPTCHA based on 2D shadow characters with background with lines. The recognition percentage obtained by the OCR programs is also reduced in comparison with the blank background, but show no improvement in comparison with the results obtained from the second set of experiments

features of the shadow characters. Once the features are extracted, a comparison was carried out character by character to check if there was a match between them and the characters in the image. In the Figure 6.10 can be appreciated the matching between the database features and the features in the CAPTCHA.

The experiments done show that most of the time, if a character is clearly recognised, the matches are between 2 and 3 vectors. If the match is no clear enough, one vector is obtained, and a decision is needed to decide if the match is valid or not. If there are not matches at all, there are no pointing vectors. We tested the matching algorithm with 100 images. In order to pass, the algorithm has to recognise the six characters in the CAPTCHA.

During the experiments, also the weakness of SIFT character recognition system was evaluated. To do that, the recognition rate for the different distortions applied was recorded to create statistics to see which distortions are the most efficient against recognition programs and specifically, SIFT. The valid matches obtained show that characters rotated 45 degrees or sheared vertically are recognised with 45.84% accuracy. Vertical mirror has a 60% of correct answers but 66.7% of times we have obtained only one matching vector. Horizontal shearing has an accuracy of 17.64% of correct answers as opposed to perspective distortions that have 28.57% of correct answers (see Figure 6.11).

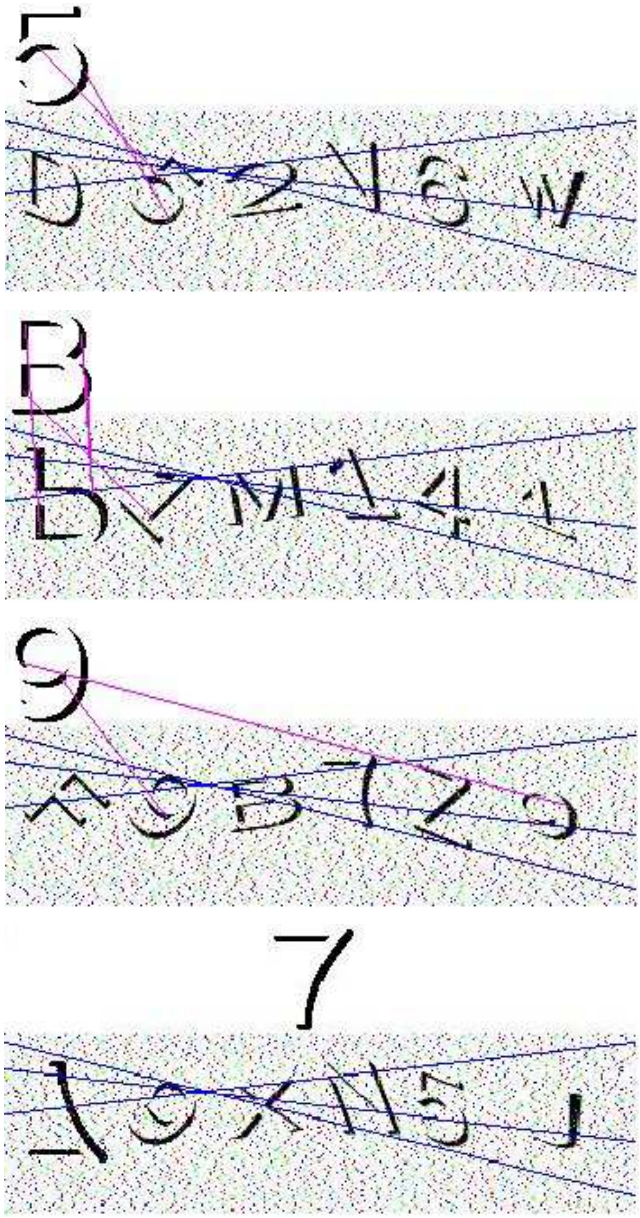

**Figure 6.10:** Matching results from SIFT with 2D shadow characters

According to the results, the inclusion of distortions have proved the robustness of the method (see table 6.4). The rate of tests passed successfully by SIFT software is 0%. However, it also shows that some distortions and characters allow better recognition that others. If we make a comparision of our results with the results obtained by other researchers with the available CAPTCHAs [40, 73, 73, 130, 131], the improvements can be clearly seen since the percentage of correctly guessed test goes from 30% up to 70%.

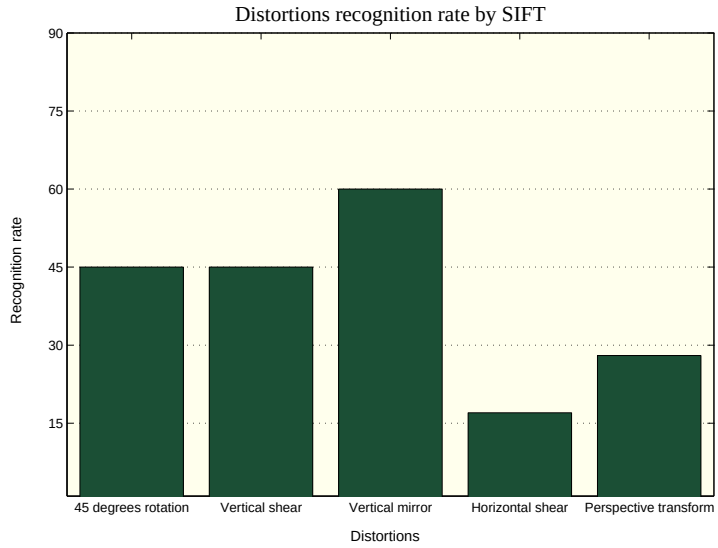


**Figure 6.11:** Statistics obtained from evaluating the capacity of SIFT software to recognise diverse distortions applied to the 2D characters approach. The distortions include: 45 degree rotations, vertical mirror, vertical shearing, horizontal shearing and perspective transformations.

| Number of characters recognised | Recognition percentage |
|---|---|
| 0 | 20% |
| 1 | 10% |
| 2 | 20% |
| 3 | 50% |
| 4 | 10% |
| 5 | $\leq 1\%$ |
| 6 | 0% |

**Table 6.4:** SIFT results on the images tested for the 2D characters approach. It shows that the rate of tests passed successfully by SIFT software is 0%. However, it also shows that some distortions and characters allow better recognition that others.

To test the performance of the CAPTCHA tests it is also necessary an evaluation of human usability, and its results hold the same importance as the machine recognition ones. To evaluate the images with human beings, one hundred different people with different levels of Internet knowledge and visual capacity were selected. The images were presented to the users with one requirement; fill the box with the characters you see on the image. In order to pass the test; the people have to recognise the six characters in every image. During the tests, 70 % of the

| Number of characters recognised | Percentage of Test |
|---|---|
| 0 | 0% |
| 1 | 0% |
| 2 | 0% |
| 3 | 0% |
| 4 | 0% |
| 5 | $\leq 3\%$ |
| 6 | 97% |

**Table 6.5:** Results obtained in the tests solved by humans for the 2D shadow characters approach. Although not all the times the six characters are fully recognised, 90% of the times, the person using this CAPTCHA in a web application can pass the test without having issues

people failed in recognising the character ”P”. They identified it as a ”b”. Also, the character ”X” has been recognised as a ”t” and as the symbol +. In terms of human recognition, this approach is also robust and efficient. Although not all the times the six characters are fully recognised, 90% of the times, the person using this CAPTCHA in a web application can pass the test without having issues. To make it simpler, some directives can be given to help solve the tests (see table 6.5).

### 6.4.2 Visual Word-Based CAPTCHA based on 3D Character models

For these experiments, the SIFT software over images with distortions has been used. The images consist in six random characters rendered and distorted. Performance of the visual word-based CAPTCHA based on 3D characters models has been evaluated and here the results are presented. The results are summarised by calculating a recognition percentage obtained by SIFT software. This set of experiments targets only the robustness against machine attacks, whereas human usability will be studied in following sections.

Image recognition is a task that can be solved almost without any effort by a human but it cannot be solved straightforwardly by a computer vision technique. Thus, to check the ef-

ficiency and the robustness of the approach we had used the computer vision technique called Scale Invariant Feature Transform (SIFT) [24]. This method extracts distinctive invariant features from images. The features are used to perform matching between different views of an object or a scene. We had chosen this approach because of the following advantages: 1. The features are invariant to image scale and rotation and they are robust across a substantial range of affine distortion, change in 3D viewpoint, addition of noise and change in illumination. 2. This algorithm has highly distinctive features that allows correct object matching with low probability of mismatch and also it is robust in identifying clustered and occluded objects. 3. Ease of extraction even having a large database of features. The recognition uses matching between the image features and a database features that have known objects.

To evaluate the images with SIFT, a 2D image database was created by rendering every model in a fixed position and light effect (see Figure 6.5). The database was then used to extract the features of the shadow characters. Once the features were extracted, a set of experiments was performed by comparing each random image test created with every character in the database. In Figure 6.12 can be appreciated the matching between the database features and the features in the CAPTCHA. Most of the time, if a character is clearly recognised, the matches are between 2 and 3 vectors. If the match is not clear enough, we can obtain one vector. But we have to decide if the match is valid or not. If there are no matches at all, there are no pointing vectors. Figure 6.12 shows results for matching between features in our database and features in the CAPTCHA. The first image shows a clear match between a known feature and the corresponding feature in the image. An incorrect match is showed in the second image. Because SIFT is based upon vectors, if the features are similar enough, there can be matching errors. The last image shows no match with this technique.

During the experiments, also the weakness of SIFT character recognition system was evaluated. To do that, the recognition rate for the different distortions applied was recorded to create statistics to see which distortions are the most efficient against recognition programs and specifically, SIFT. The matching algorithm was tested within 100 images. In order to pass, the

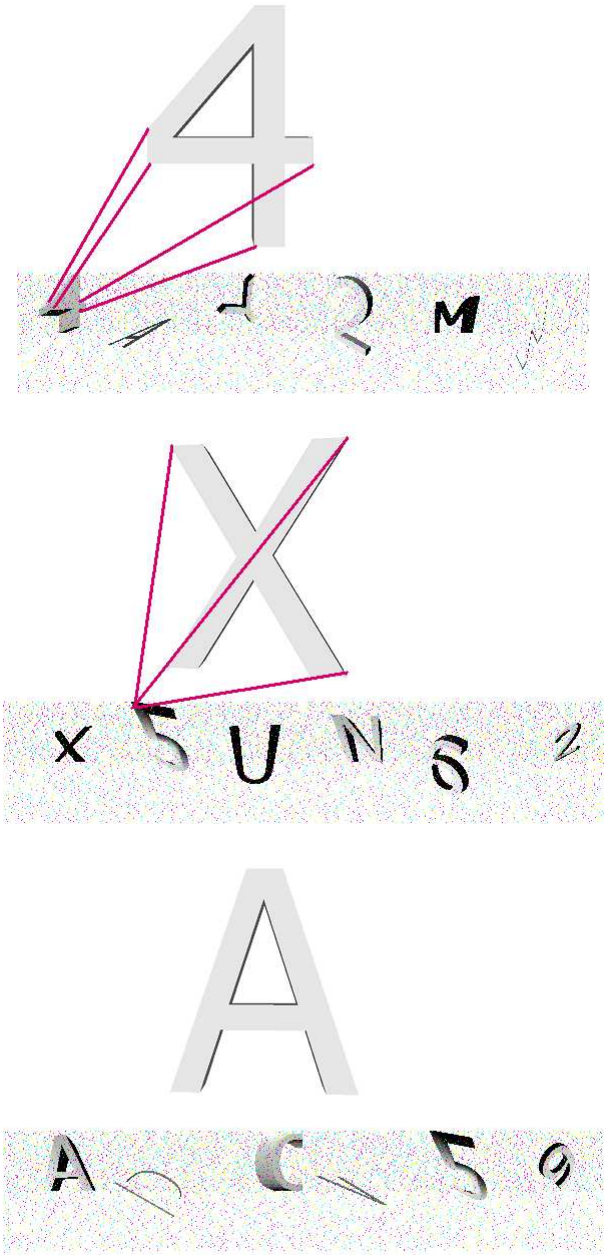

**Figure 6.12:** Matching results from SIFT with 3D models

algorithm has to recognise the six characters in the CAPTCHA. In 65% of cases, 3D characters only rotated around x or y label are recognised. Also, translations and scaling are easily recognised by SIFT. Nevertheless, shearing, perspective transformation and warping cannot be recognised by SIFT (see Figure 6.13). As scaling and translation are always accompanied

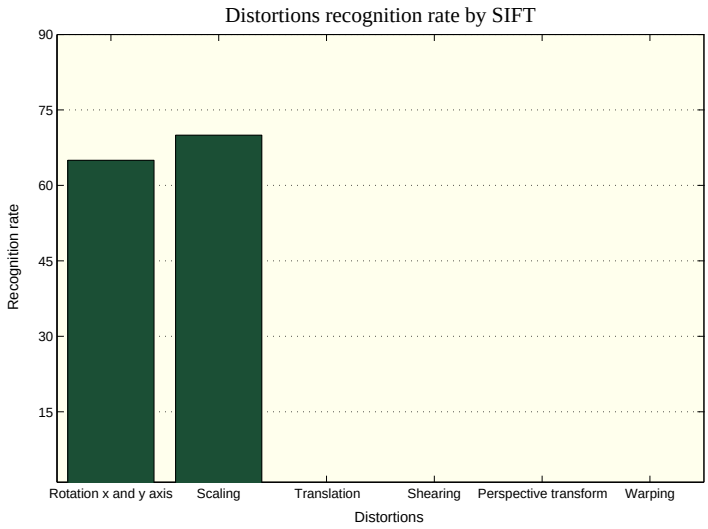


**Figure 6.13:** Statistics obtained from evaluating the capacity of SIFT software to recognise diverse distortions applied to the 3D characters approach. The distortions include: rotation around x and y labels, translations, scaling, shearing, perspective transformation and warping.

| Number of characters recognised | Percentage of Test |
|---|---|
| 0 | 90% |
| 1 | 5% |
| 2 | 5% |
| 3 | 0% |
| 4 | 0% |
| 5 | 0% |
| 6 | 0% |

**Table 6.6:** SIFT results on the images tested for the 3D models approach. In 65% of cases, 3D characters only rotated around x or y label are recognised. Also, translations and scaling are easily recognised by SIFT. Nevertheless, shearing, perspective transformation and warping cannot be recognised by SIFT

by other distortions, characters are hardly recognised by the computer vision technique(see table 6.6).

There are two main factors that make this algorithm robust and efficient; first, the fact that the characters are randomly chosen in real time make the recognition much harder for

| Number of characters recognised | Percentage of Test |
|---|---|
| 0 | 0% |
| 1 | 0% |
| 2 | 0% |
| 3 | 0% |
| 4 | 0% |
| 5 | $\leq 3\%$ |
| 6 | 97% |

**Table 6.7:** Results obtained in the tests solved by humans for the 3D models approach. Although not all the times the six characters are fully recognised, 90% of the times, the person using this CAPTCHA in a web application can pass the test without having issues.

computer vision techniques based upon words matching as well as OCR programs. Second, the distortions applied are complicated enough to make the shape matching more difficult.

To evaluate the images with human beings, one hundred different people with different levels of Internet knowledge and different visual capacity were selected. The images were presented to the human users by giving them only one direction: fill the box with the characters you see on the image. In order to pass the test, a person had to recognise the six characters in every image. During the tests, 30% of the people failed recognising the ”7”. They identified it as a ”1”. Also, the character ”6” was recognised as ”8” and as the symbol. Most of the issues recognising letters or numbers during a test can be produced by the lighting effects since the boundaries between the background and the model can become blurred. Therefore, during the lighting process, we had to be careful to place the light and the point of view according to some specific parameters. In terms of human recognition, we can say that this approach is also robust and efficient. Although not all the times the six characters are fully recognised, 90% of the times, the person using this CAPTCHA in a web application can pass the test without having issues. To make it simpler, some directives can be given to help recognise the characters (see table 6.7).

In this chapter, a visual word-based CAPTCHA novel approach is described. The creation of shadow characters along with the distortions applied are explained. Human and machine performances are compared with state-of-the-art character recognition software and human users. The results obtained in the novel CAPTCHA [150] are also summarised. Exhaustive experiments were performed to ensure the efficiency of the approach, where one hundred human users with different social backgrounds and technological knowledge evaluated the test.

# Chapter 7

# Image Based CAPTCHA

In the current approaches to image-based CAPTCHAs, the main idea is to use images of objects, faces, or shapes to present the user with a quiz test, a matching test, or different kinds of recognition tests. In addition, background noise or distortions are applied to the images to avoid machine recognition. Due to the simplicity of the tests, or because of the insufficient amount of distortions, many of these approaches have become obsolete since they can be passed, with a high rate of success, by machines (see chapter one) [70, 118]. To increase the robustness and efficiency of the image-based CAPTCHAs there are generally three issues to be addressed; the first is the general concept in which the test is based, since it will imply an uncomplicated approach easily deciphered by machines. The second issue is the quality of the image warping techniques applied, where the distortions must avoid being too simplistic that can be recognised straightforwardly by a computer vision technique. The third and final issue is that even though the aim is to develop a test that requires a certain level of reasoning skills and distortions, these modifications can sometimes make it harder for humans to solve the test. Furthermore, these types of tests are often found annoying and time-consuming for most people. The aim is to create a human-friendly test that humans can pass easily.

In this chapter an image-based CAPTCHA novel approach is presented. Along with the developed face cartoons, the distortion techniques applied allow an increase on robustness against machine recognition whilst preserving a fairly low difficulty in terms of human recognition. Human and machine performance are compared to state-of-the-art face recognition software and human users. In the following, the results of the novel CAPTCHA presented in [151] are summarised. Exhaustive experiments were performed to ensure the efficiency of the approach, where one hundred human users with different social backgrounds and technology knowledge evaluated the test. For each experiment, a random CAPTCHA was designed for both the human user and the machine software.

## 7.1 Properties of the Image CAPTCHA based on Face Recognition

As the work presented in this chapter primarily targets the visual non OCR-based CATPCHA's methods, in this section the basic properties of the framework are described. The structure of the algorithms is primarily the result of trying to create a new concept for a visual CAPTCHA, minimising the risk of machine recognition to decipher the test's solution, and increasing human usability. The former corresponds to one of the main concerns when developing an image CAPTCHA, since most of the available methods are very simplistic and the concept is easily understood by a machine. The latter two corresponds to web security and the level of human skills necessary to solve the tests, such as accuracy, response time, and perceived difficulty of the user. To increase security, there are many strategies that can be used that involve distortions, background noise, diverse types of objects, and the use of different ways to solve the tests such as rotating an object, clicking, texting, etc [182]. There exists several methods that focuses on clicking a button [20, 42, 189] or texting the solution of a problem [57, 121]. The majority of these methods can be easily broken by a machine, since some of them require users to choose between two options or the distortions applied are not enough. These methods were created

with the main purpose of increasing human usability, consequently their robustness against machine attacks decreased.

The approach presented here consists of creating a brand new type of CAPTCHA test. This test uses face images[1] instead of characters. The faces used in this approach are faces from well known people which belong to the industry of the cinema, arts, politicians or sports. The approach consists of morphing the face images into cartoons or animals and show them to the users. To pass this test, the user has to recognise three faces and choose the proper name amongst the ones provided (see Figure 7.1). The framework is as follows:

1. CAPTCHA development: We have chosen to continue our research in this field because of two reasons; firstly, humans are very perceptive when it comes to faces. They can easily recognise a face they have seen before if that face is associated with a famous person. As said in the state of the art, the human brain starts learning face perception within the first six months of life and keeps on doing so following a process in which they gain more and more experience within the same-race faces or other-race faces, gender, etc. Secondly, thanks to the developing technologies and the media, the world is reaching a global state where information is readily accessible to everyone, especially when it involves people equated with entertainment.

2. Morphing techniques: Many computer vision techniques can solve the test by comparing features extracted from a database. To avoid this problem we introduce a morphing process to our images in real-time. In this process, faces are morphed into a random cartoon or animal. The faces are randomly morphed by varying the variables in the process.

[1]Face images are images containing only frontal faces.

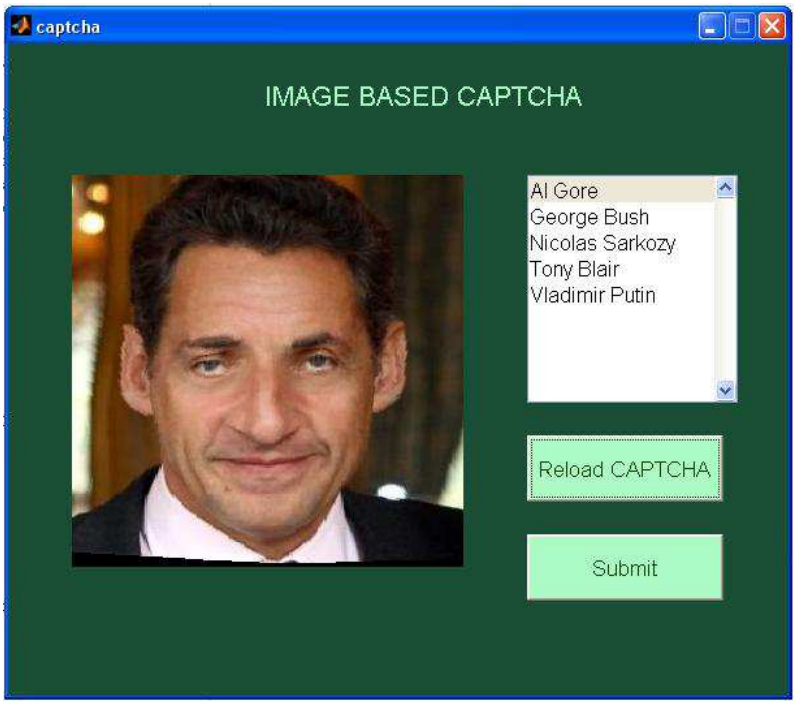


**Figure 7.1:** User's interface for Image-Based CAPTCHA.

## 7.2 Image-Based CAPTCHA based on Face Recognition

The concept of distorted faces is introduced here. The first step was the creation of a database of face images of well known people. This database consists of a set of faces images of different people, such as Albert Einstein, Queen Elizabeth II or David Beckham. All the images have a prefixed size of 280x309 pixels in jpeg format. Once the face images are selected and changed to the desired format, the following step was to select and create another set of images for the distortions. For this, different face images of animals and cartoons were selected. The format and the size of the distortion set of images have to be exactly the same as the human face images.

The next step was the introduction of a morphing algorithm to create the distortion between the images. The morphing technique applied to the images is a feature-based morphing process with multiple pairs of lines. This technique transforms one digital image into another. Multiple pairs of lines define the mapping from one image to the other. Each line in the source image has a corresponding line in the destination image such that for a line PQ in the destination image

**Figure 7.2:** Multiple pairs of lines sued for the morphing technique. For each source image's line, there are corresponding lines in the destination image.

and P'Q' in the source image. After mapping each source line into the corresponding destination line, the algorithm performs the blending between the images [14]. For this CAPTCHA approach, the blend-factor to determine the level of morphing is randomly chosen to allow an increase in uncertainty for security reasons. Finally, instead of interpolating both images, the source image is kept with only the distortion.

For each source image that contains the face of a well-known person there is a database of lists that contain names. In each list, the names selected for each famous personality are related to them by their age, gender or profession. For instance, if the face selected belongs to a middle aged brunette actress, the names in the corresponding database should belong to middle aged brunette actresses with similar face features. In our CAPTCHA interface, four names are randomly chosen from the database, and they appear together with the real name. In order not to confuse the users, the names of the famous people whose faces are distorted are not used in the database of other personalities. Figure 7.3 shows a block diagram of the approach. The following is a detailed explanation of the whole procedure:

1. Two face images' databases and one names' database were created. The well-known personalities database that contains the source images, the animal and cartoons database that contains the destination images, and the names' database contain names from different celebrities than the ones in the face images.

2. Once the databases were finished, the software of the CAPTCHA test was developed. The program randomly selects a face image and a cartoon image from the databases, following a uniform distribution function.

3. The morphing is then applied from the source image to the destination image. The variables that define the morphings are given a random value within a range that allows a morphing difficult enough for the machines and easy for the human users. The range was calculated testing the resulting images with human users.

4. When the source image is selected, also a random selection of four names is extracted from the names database. Together with the real name, the names are put in a list for the user to select one.

5. The final step is the user's interface development. This interface is divided into two parts. The left section contains the image with the morphed face. The right section contains a vertical text box with the names. To pass this test, the user needs to recognise three faces in a row. If one of the faces is unknown, there is always the possibility of reload the CAPTCHA and start anew.

Nowadays, face recognition has become a popular research field not only in computer vision but also for neuroscientists and psychologists. Mainly because advances in computer vision and machine learning which provide useful insights of how the human brain works to other fields and vice-versa. In this thesis, it is a problem of artificial intelligence and machine learning. The aim was to expand the correlation between what a human can recognise and a machine cannot. By doing so, an advanced and efficient software is created to allow people to login to web applications quickly meanwhile machines are not able to automatically do the same.

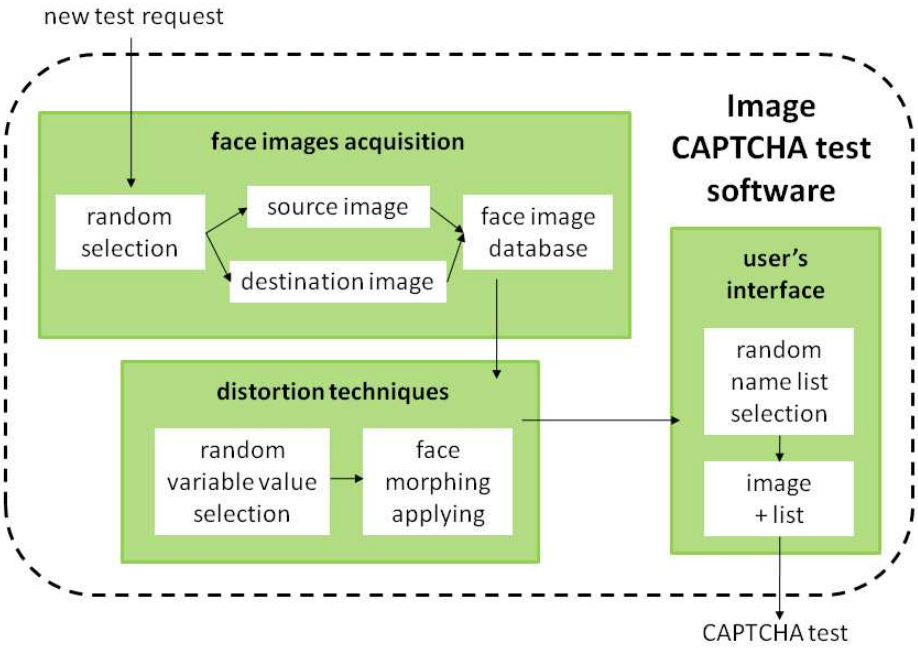


**Figure 7.3:** Block diagram of the image-based CAPTCHA based on face recognition.

## 7.3 Performance analysis of the Image CAPTCHA approach

In this section, the performance of the proposed approach using distorted faces is presented. The developed method allows an increase in Internet security and human usability. The evaluation is done by using different face recognition systems. A face recognition system is an algorithm that given an image, can identify or verify the person that appears in the image using a stored database of faces. To check the efficiency and robustness of the morphed images, three different face recognition systems were used that are well known for their accuracy. The first two face recognition softwares chosen were the ones generated by Colorado State University called CSU Face Identification Evaluation System [17] which includes the PCA algorithm and the LDA algorithm. For the third algorithm, the matlab code generated by Ritwik Kumar that contains the Volterrafaces face recognition system was used [104].

For the experiments with the CSU system, a face database was created to train the face recognition systems and to measure the recognition rate. The database consists of a set of distorted faces and a set of non-distorted faces. For the PCA and LDA algorithms, the output was the distances between the distorted images and the training face database.

To validate the results, two sets of experiments were carried out; The first one is an experiment with the distorted faces, and the second one is an experiment with non-distorted faces. In both cases the same set of non-distorted faces was kept for the training phase. Finally, the distances were plotted into the standard FERET cumulative match curves(CMS) [143]. The results are given in recognition rate per rank. The rank is computed comparing each probe image with the closest gallery image of the same subject. Finally, a rank curve is generated by summing the number of correct matches for each rank.

The evidence presented demonstrates that the visual image-based CAPTCHA is capable of handling the web applications security requirements, and improve the results obtained by the actual image-based CAPTCHAs that can be found on the Internet. In this thesis, the principal objective is to increase the efficiency in terms of human recognition while improving the robustness in order to avoid computers to solve the tests.

### 7.3.1 Principle Components Analysis System

The first face recognition system used to test the image based CAPTCHA approach was the Principle Components Analysis (PCA) [186]. This system works with linear transformations in the feature space. The feature vectors are formed by concatenating the pixel values from the images. These raw feature vectors are very large and are highly correlated. PCA rotates feature vectors from this large, highly correlated subspace to a small subspace which has no sample covariance between features. PCA has two valuable properties when it is used for face recognition:

1. It can be used to reduce the dimensionality of the feature vectors in either a lossy or lossless manor.

2. PCA eliminates all of the statistical covariance in the transformed feature vectors, resulting in a diagonal covariance matrix for the transformed (training) feature vectors.

The PCA interface generates distance files. The algorithm projects the feature vectors onto the basis. It then computes the distance between pairs of images in the list. The output is a set of distance files containing the distance from each image to all other images in the list. Figure 7.4 shows the results obtained with the PCA algorithm and the euclidean and mahcosine distance metrics. For PCA algorithm, two different distance metrics are provided:

Euclidean (L2):

$$D_{Euclidean}(u,v) = \sqrt{\Sigma(u_i - v_i)^2} \tag{7.1}$$

Mahalanobis Cosine:

$$S_{MahCosine}(u,v) = \cos(\theta_{mn}), D_{M}ahCosine(u,v) = -S_{MahCosine}(u,v) \tag{7.2}$$

Examining the results obtained in both graphics (see Figure 7.4), it can be appreciated a decrease in terms of recognition when the distortions are applied to the face images. Although the amount of reduction is not very high, the results are quite good when considering that the system has not got a high recognition factor even with the results obtained in the non-distorted images.

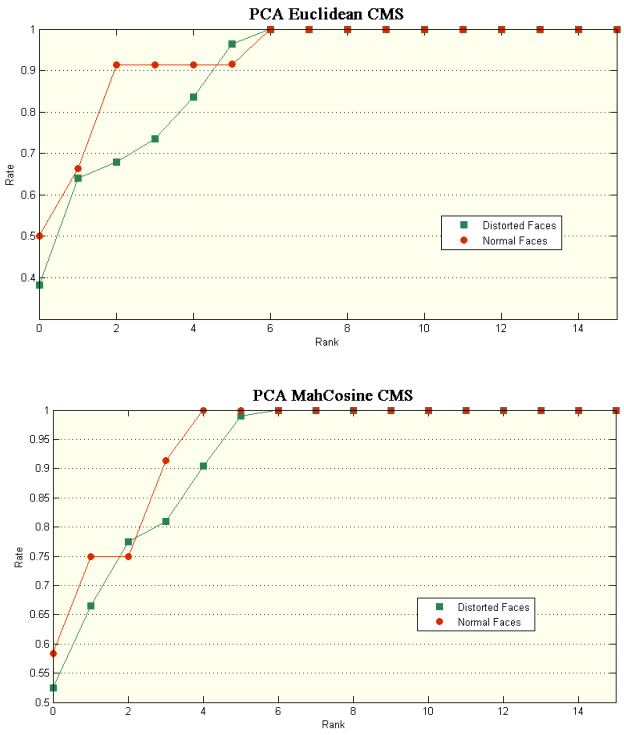


**Figure 7.4:** Results extracted from the PCA algorithm: (a) Euclidean metric and (b) Mahcosine metric. There is a decrease in terms of recognition when the distortions are applied to the face images.

### 7.3.2 Linear Discriminant Analysis

The second algorithm used was Linear Discriminant Analysis (PCA+LDA) with Fisher's Linear Discriminants [215]. The LDA training system attempts to produce a linear transformation that emphasise differences between classes while reducing differences within classes. The goal is to form a subspace that is linearly separable between classes. Each individual is taken as a class and the training system requires multiple images per subject. LDA training is performed firstly, by using PCA to reduce the dimensionality of the feature vectors. And secondly, by performing LDA on the training data to further reduce the dimensionality in such a way that class distinguishing features are preserved. A final transformation matrix is produced by multiplying the PCA and LDA basis vectors to produce a full input image to LDA space transformation matrix. The algorithm produces a set of LDA basis vectors. These basis vectors produce a

transformation of the feature vectors. The output is a set of distance files containing the distance from each image to all other images in the list.

For LDA algorithm, a PCA+LDA specific distance measure is provided. It was proposed by [217]:
LDASoft:

$$D_{LDASoft}(u,v) = \Sigma\lambda_i^{0.2}(u_i - v_i)^2 \tag{7.3}$$

Figure 7.5 shows the results obtained with the LDA algorithm and the LDAsoft distance metric.

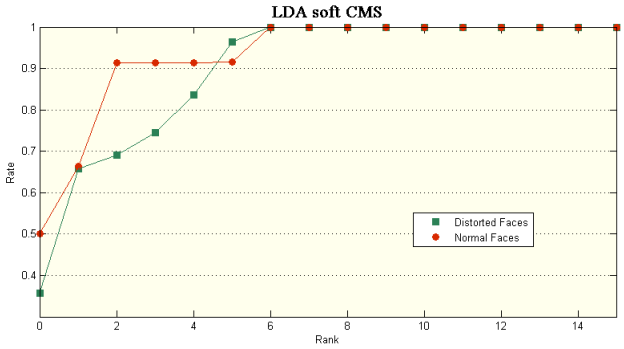


**Figure 7.5:** Results extracted from the LDA algorithm with LDAsoft metric. There is a decrease in terms of recognition when the distortions are applied to the face images.

Examining the results obtained with this algorithm (see Figure 7.5), it can also be appreciated a decrease in terms of recognition when the distortions are applied to the face images. The same observation as the previous algorithm is valid, and also the values obtained are very similar to the PCA system.

### 7.3.3 Volterrafaces Face Recognition System

The third algorithm used to test the image-based CAPTCHA approach was Volterrafaces. This approach represents face images as a spatial arrangement of image patches, and seek a smooth

| | Yale A | Extended Yale B | Image-based CAPTCHA |
|---|---|---|---|
| Volterra Linear | 6.11 | 6.35 | 20.92 |
| Volterra Quad | 10.19 | 13.0 | 19.80 |

**Table 7.1:** Results obtained with the Volterrafaces system in average recognition error rates. There is a clear increase in the average error rate recognition when compared to other results obtained from face images' databases.

non-linear functional mapping for the corresponding patches such that in the range space, patches of the same face are close to one another, while patches from different faces are far apart, using L2 as the distance measure. Volterra kernels are used to generate successively better approximations to any smooth non-linear functional. During the testing phase, each patch from the test image is classified independently and casts a vote towards image classification. The class with the maximum votes is chosen as the winner [104].

In order to create the image patches, this algorithm uses thumbnails. Accordingly, a database of thumbnails was created. This database contains a set of distorted images created by the captcha interface and a set of non-distorted images to train the system. In this thesis, the results are presented comparing the ones obtained through the experiments ran by [104] with the Yale A[2] and Yale B face database[3] and the results obtained with the CAPTCHA's images. For the experiments, a linear kernel size of 5 *x* 5, and the Quadratic kernel size of 3 *x* 3 were used. The results, measured in average recognition error rate, are presented in Table 7.1:

The results obtained with this algorithm (see Table 7.1) shows a higher average recognition error when using the distorted faces in comparison with other face images' databases. It also shows that the CAPTCHA algorithm works even with more recent face recognition systems.

[2]http://cvc.yale.edu/projects/yalefaces/yalefaces.html
[3]http://cvc.yale.edu/projects/yalefacesB/yalefacesB.html


**Figure 7.6:** Results in the Image-based CAPTCHA tests solved by humans.

### 7.3.4 Human Recognition

To evaluate the images with human beings, we have selected fifty different people with different levels of cultural knowledge and different visual capacity. As mentioned in human perception section, human recognition depends on diverse factors such as sex gender, race and the possibility of having prosopagnosia. For these reasons, we selected the famous people images in our database based upon their popularity, sex gender and race. Also, to run the experiments, we took into account the different capabilities of individuals by splitting them into two groups; females and males. We presented the images by giving them only one direction: select the corresponding name of the person you think they are. In order to pass the test, a person had to recognise three different people in a row. Figure 7.6 show the results obtained.

Looking at Figures 7.4 - 7.5, three observations can be made on the performances of the first two face recognition systems:

- The maximum rank feasible with the provided database is four, since for the training phase, only four face images were provided. Once that rank is exceed the recognition rate goes to one because it recognises all the images as valid.
- Due to the small amount of images provided to the experiments, a general poor performance is obtained with the non-distorted faces database as compared with the FERET results [62].
- From the comparison curves, it can be concluded that the images created by the CAPTCHA system cannot be easily recognised by the face recognition systems due to the poor recognition rates obtained.

Looking at the Table 7.1, the image-based CAPTCHA images have a higher average error recognition rate as compared with frontal face images from Yale A and Yale B face databases. It can be generally concluded that the results from the previous algorithms together with the Volterrafaces results highlight the fact that image-based approach produced significant improvements increasing the gap of what a human can recognised and a machine cannot.

In terms of human recognition,it can affirmed that the image-based CAPTCHA is a robust and efficient system. Although not all the times all the face images are recognised, in average, 96% of the times, the person using our test can pass the test without having issues. Further developments can include a study to evaluate which famous people are recognised more easily.

In this chapter, an image-based CAPTCHA novel approach is described. The creations of the face cartoons, along with the morphing techniques applied are explained. Human and machine performance are compared to state-of-the-art face recognition software and human users. The results obtained in the novel CAPTCHA [151] are also summarised. Exhaustive experiments were performed to ensure the efficiency of the approach, where one hundred human users with different social backgrounds and technological knowledge evaluated the test.

# Chapter 8

# Conclusions and Future Work

## 8.1 Conclusions

The aim of the research presented in this thesis was to increase the gap of what humans can recognise and machines cannot. Additionally, the creation of more robust and efficient novel methods was targeted. The main focus was centred on creating CAPTCHA tests using human psychology and universal common knowledge. The first step towards the developed methods was to analyse the current approaches and distinguish their weaknesses and possible ways to improve them. This includes a research of computer vision software that allows machines to break through the tests.

The research on the current methods available uncovered the necessity of a classification to categorise the algorithms by the computer vision techniques used and by human aptitudes. For the classification, three main categories have been considered: OCR-based methods, Visual non OCR-Based methods and non Visual methods. These categories have been divided into subcategories for a more accurate classification. Along with the sub-categorisation, an extensive analysis of the available methods and their reliability was presented in the thesis, reaching the following conclusions:

OCR-Based methods were the first CAPTCHAs to emerge. They had a quick expansion to many different web applications as well as many prototypes. Along with the expansion, several different programs to break through them arose which provoked an increase in difficulty in the tests. Nowadays, most users find the annoying and time consuming.

- Visual non OCR-Based methods emerged to explore diverse sides of HIP methods. At the beginning they focused on solving quizzes or matching problems but rapidly expanded to many other areas. Also, their reliability increased with time, going for easy to break to more secure that OCR-Based methods. Their diversity make them more human friendly and less time consuming.

-Non visual methods arose as an alternative to visual methods due to some visual impairments users may have. They weren't as successful as the others due to their difficulty and language restrictions.

The second step in this research was the development of two novel methods to prevent spam and malicious software to break through web applications and increase security when login in. The first method uses shadows to represent characters. The shadow boundaries were chosen to develop the fact that humans can easily recognise objects and characters only by the shadows but machines cannot. The distortions applied to the images are based upon geometric transformations that include affine and perspective transformations. The approach based on 2D shadow characters shows an improvement in efficiency and robustness over the actual CAPTCHAs. The visual word-based CAPTCHA using 3D models is based upon lighting effects to create 3D shadows boundaries. The performance of this algorithm highlights that using 3D models yields better results in terms of efficiency and robustness. These tests are more difficult to solve for computer vision techniques but they still remain easy for humans. In this method, one of the challenges faced was that people visually impaired or with mental illness as dyslexia should be able to recognise the characters. However, it is also necessary to make the tests difficult enough for the machines not to break through them.

Humans can easily recognise cartoons or sketches from famous people, even if they are rotated or manipulated. A machine cannot recognise this type of image because it does the matching by pattern or feature extraction and the original one is very different. The second method uses distorted faces of world famous people to create a test to secure web applications. The main basis for the development of this method was the innate ability of human beings to recognise faces. The distortions applied to the face images are based upon a feature-based morphing process with multiple pairs of lines. The performance of this algorithm highlights two facts; firstly, using distorted faces as a test increments the efficiency and robustness of the previous approaches and secondly, it increments the difficulty for face recognition techniques to break through our system.

## 8.2 Future Work

In addition to the developed work, there are some challenges that have appeared while developing the second approach. The main focus addresses the level of distortions applied to the faces. The main reason is that a high distortion factor can make the faces indistinguishable and a low rate can make it to easy for the face recognition system to break through the test. To measure the appropriate levels of morphing, different variables and factors were taken into account; cross-dissolve factor range, human recognition capabilities and the cartoon or animal used in the destination image. Another important factor to take into account was that people with prosopagnosia have more difficulties when recognising and distinguishing human faces, and even though there is nothing much that can be changed in this approach, the only alternative to help the human users with this problem is which kind of faces can be used.

Practical realisations of methods presented in this thesis have enabled a high efficiency and robustness in the OCR-based CAPTCHA approach and the Image-based CAPTCHA approach. On the other hand, these realisations have also uncovered several interesting topics for future research, as well as some issues that have not been yet adequately resolved. These include:

-Since human and machine recognition depends on the diverse distortions applied, it is necessary an optimisation of the warping and morphing techniques by improving the algorithms and creating smoother transitions for the original image to the distorted one. New morphing techniques should also be taken into consideration.

-Evaluation study of face recognition by human users depending on geographical locations. Knowing the cultural background and social knowledge is an important factor to increase the success rate by users. Also, it will be necessary to update the database depending on the latest celebrities or personalities that are famous at that moment.

-Although the developed methods can prevent machines to successfully pass the current CAPTCHAs, as the computer vision techniques research advances similarly the CAPTCHAs should improve. Therefore, the techniques applied and the human psychology used should be further studied.

# Publications


[1] E.Izquierdo K.Vaiapury, C.Romero Macias. A new integrated methodology for 3d discrepancy checking and measuremenbt in assenbly. This is a article for 3DTVCON 2012., 2012.

[2] C. Romero Macias and E. Izquierdo. Visual word-based captcha using 3d characters. In *3rd International Conference on Crime Detection and Prevention (ICDP 2009)*, pages 1–5. IET, 2009.

[3] C. Romero Macias and E. Izquierdo. Image captcha based on distorted faces. In *4th International Conference on Imaging for Crime Detection and Prevention (ICDP 2011)*, pages 1–6. IET, 2011.

[4] C. Romero Macias and E. Izquierdo. A survey of captchas: Are computers getting the better of us? This is a journal for ACM Surveys., 2011.


# References


[5] AltaVista. Altavista's ``add-url´´site, protected by the earliest known captcha. `http://altavista.com/sites/addurl/newurl`, 1997.

[6] N. Arad, N. Dyn, D. Reisfeld, and Y. Yeshurun. Image warping by radial basis functions: Application to facial expressions. *CVGIP-Graphical Models and Image Processing*, 56(2):161–172, 1994.

[7] E. Bagherian and R.W.O.K. Rahmat. Facial feature extraction for face recognition: a review. In *International Symposium on Information Technology, ITSim 2008*, volume 2, pages 1–9. IEEE, 2008.

[8] H.S. BAIRD and J.L. BENTLEY. Implicit captchas. In *SPIE proceedings series*, pages 191–196. Society of Photo-Optical Instrumentation Engineers, 2005.

[9] H.S. Baird and D.P. Lopresti. *Human interactive proofs: second international workshop, HIP 2005, Bethlehem, PA, USA, May 19-20, 2005: proceedings*, volume 3517. Springer Verlag, May 2005.

[10] H.S. Baird, M.A. Moll, and S.Y. Wang. Scattertype: A legible but hard-to-segment captcha. In *Eighth International Conference on Document Analysis and Recognition*, pages 935–939. IEEE, 2005.

[11] H.S. Baird and K. Popat. Human interactive proofs and document image analysis. In *Proceedings of the 5th International Workshop on Document Analysis Systems V*, pages 507–518. Springer-Verlag, 2002.

[12] M.S. Bartlett, J.R. Movellan, and T.J. Sejnowski. Face recognition by independent component analysis. *IEEE Transactions on Neural Networks*, 13(6):1450–1464, 2002.

[13] R. Beede. Analysis of recaptcha effectiveness. `http://www.rodneybeede.com/reCAPTCHA_weakened.html`, 2010.

[14] T. Beier and S. Neely. Feature-based image metamorphosis. *Computer Graphics*, 26(2):35–42, 1992.

[15] P.N. Belhumeur, J.P. Hespanha, and D.J. Kriegman. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):711–720, 1997.

[16] R. Bergmair and S. Katzenbeisser. Towards human interactive proofs in the text-domain. *Information Security*, pages 257–267, 2004.

[17] R. Beveridge and B. Draper. CSU Face Identification Evaluation System. `http://www.cs.colostate.edu/evalfacerec/index10.php`, 2010.

[18] M. Bichsel. *Strategies of robust object recognition for the automatic identification of human faces*. PhD thesis, ETH, Zurich, 1991.

[19] J.P. Bigham and A.C. Cavender. Evaluating existing audio captchas and an interface optimized for non-visual use. In *Proceedings of the 27th international conference on Human factors in computing systems*, pages 1829–1838. ACM, 2009.

[20] M. Blum, LA Von Ahn, J. Langford, and N. Hopper. The captcha project, ``completely automatic public turing test to tell computers and humans apart´´. *School of Computer Science, Carnegie-Mellon University, http://www. captcha. net*, 2000.

[21] B. Bocian. *Fritz Perls in Berlin, 1893-1933: Expressionismus, Psychoanalyse, Judentum*. EHP Verlag Andreas Kohlhage,Bergisch Gladbach, 2010.

[22] W.J. Bouknight. A procedure for generation of three-dimensional half-toned computer graphics presentations. *Communications of the ACM*, 13(9):527–536, 1970.

[23] J. Brooke, N. Bevan, F. Brigham, S. Harker, and D. Youmans. Usability statements and standardisation: Work in progress in iso. In *Proceedings of the IFIP TC13 Third Interational Conference on Human-Computer Interaction*, pages 357–361. North-Holland

Publishing Co., 1990.

[24] M. Brown and D.G. Lowe. Invariant features from interest point groups. In *British Machine Vision Conference*, pages 656–665. British Machine Vision Association, 2002.

[25] V. Bruce, P.R. Green, and M.A. Georgeson. *Visual perception*. Psychology Press, 1996.

[26] V. Bruce and A. Young. Understanding face recognition. *British journal of psychology*, 77(3):305–327, 1986.

[27] R. Brunelli and T. Poggio. Face recognition: Features versus templates. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 15(10):1042–1052, 2002.

[28] E. Bursztein, R. Beauxis, H. Paskov, D. Perito, C. Fabry, and J. Mitchell. The failure of noise-based non-continuous audio captchas. In *IEEE Symposium on Security and Privacy (SP)*, pages 19–31. IEEE, 2011.

[29] E. Bursztein and S. Bethard. Decaptcha: breaking 75% of ebay audio captchas. In *Proceedings of the 3rd USENIX conference on Offensive technologies*, pages 8–8. USENIX Association, 2009.

[30] E. Bursztein, S. Bethard, C. Fabry, J.C. Mitchell, and D. Jurafsky. How good are humans at solving captchas? a large scale evaluation. In *2010 IEEE Symposium on Security and Privacy*, pages 399–413. IEEE, 2010.

[31] E. Bursztein, M. Martin, and J. Mitchell. Text-based captcha strengths and weaknesses. In *Proceedings of the 18th ACM conference on Computer and communications security*, pages 125–138. ACM, 2011.

[32] P. Campadelli, R. Lanzarotti, and G. Lipori. Automatic facial feature extraction for face recognition. *Face Recognition*, pages 31–58, 2007.

[33] S. Chakrabarti and M. Singbal. Password-based authentication: Preventing dictionary attacks. *IEEE Computer*, 40(6):68–74, 2007.

[34] T.Y. Chan. Using a test-to-speech synthesizer to generate a reverse turing test. In *Tools with Artificial Intelligence, 2003. Proceedings. 15th IEEE International Conference on*, pages 226–232. IEEE, 2003.

[35] M. Chandrasekaran, R. Chinchani, and S. Upadhyaya. Phoney: Mimicking user response to detect phishing attacks. In *Proceedings of the 2006 International Symposium on a World of Wireless, Mobile and Multimedia Networks*, pages 668–672. IEEE, 2006.

[36] T.C. Chang, T.S. Huang, and C. Novak. Facial feature extraction from color images. In *Pattern Recognition, 1994. Vol. 2-Conference B: Computer Vision & Image Processing., Proceedings of the 12th IAPR International. Conference on*, volume 2, pages 39–43. IEEE, 1994.

[37] SK Chaudhari, AR Deshpande, SB Bendale, and RV Kotian. 3d drag-n-drop captcha enhanced security through captcha. In *Proceedings of the International Conference & Workshop on Emerging Trends in Technology*, pages 598–601. ACM, 2011.

[38] K. Chellapilla, K. Larson, P. Simard, and M. Czerwinski. Computers beat humans at single character recognition in reading based human interaction proofs (hips). In *Proceedings of the Second Conference on Email and Anti-Spam*, pages 21–22, 2005.

[39] K. Chellapilla, K. Larson, P. Simard, and M. Czerwinski. Designing human friendly human interaction proofs (hips). In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 711–720. ACM, 2005.

[40] K. Chellapilla and P. Simard. Using machine learning to break visual human interaction proofs (hips). In *In Proceedings of the Advances in Neural Information Processing Systems (NIPS) Conference*, volume 17. MIT Press, 2004.

[41] M. Chew and H.S. Baird. Baffletext: a human interactive proof. In *In Proc., 10th IS&T/SPIE Document Recognition & Retrieval Conference*, pages 305–316. Citeseer, 2003.

[42] M. Chew and J. Tygar. Image recognition captchas. *Information Security*, pages 268–279, 2004.

[43] M. Chew and J. Tygar. Collaborative filtering captchas. *Human Interactive Proofs*, pages 95–110, 2005.

[44] A.L. Coates, H.S. Baird, and RJ Faternan. Pessimal print: a reverse Turing test. In *Sixth International Conference on Document Analysis and Recognition. Proceedings.*, pages 1154–1158. IEEE, 2001.

[45] T. Cootes, G. Edwards, and C. Taylor. Active appearance models. *Computer VisionECCV98*, pages 484–498, 1998.

[46] I.J. Cox, J. Ghosn, and P.N. Yianilos. Feature-based face recognition using mixture-distance. In *Computer Vision and Pattern Recognition, 1996. Proceedings CVPR'96, 1996 IEEE Computer Society Conference on*, pages 209–216. IEEE, 1996.

[47] I. Craw, D. Tock, and A. Bennett. Finding face features. In *Computer VisionECCV'92*, pages 92–96. Springer, 1992.

[48] J.S. Cui, L.J. Wang, J.T. Mei, D. Zhang, X. Wang, Y. Peng, and W.Z. Zhang. Captcha design based on moving object recognition problem. In *3rd International Conference on Information Sciences and Interaction Sciences (ICIS)*, pages 158–162. IEEE, 2010.

[49] Y. Dai and Y. Nakano. Face-texture model based on sgld and its application in face detection in a color scene. *Pattern recognition*, 29(6):1007–1017, 1996.

[50] M. Dailey and C. Namprempre. A text graphics character captcha for password authentication. In *TENCON 2004. 2004 IEEE Region 10 Conference*, pages 45–48. IEEE, 2004.

[51] R. Datta, J. Li, and J.Z. Wang. Imagination: a robust image-based captcha generation system. In *Proceedings of the 13th annual ACM international conference on Multimedia*, pages 331–334. ACM, 2005.

[52] decaptchablog.com. List of best decaptcher (or decaptcha) services. `http://decaptchablog.com/`, March 2011.

[53] A. Desai and P. Patadia. Drag and drop: A better approach to captcha. In *2009 Annual IEEE India Conference (INDICON)*, pages 1–4. IEEE, 2009.

[54] R. Dhamija and A. Perrig. Dj vu: A user study using images for authentication. In *Proceedings of the 9th conference on USENIX Security Symposium-Volume 9*, pages 4–4. USENIX Association, 2000.

[55] R. Doyle. Image-based captcha with jaci. `http://ryandoyle.net/devel/jaci/`, 2009.

[56] W Ellis. A source book of gestalt psychology. *A source book of Gestalt psychology*, pages 71–88, 1938.

[57] J. Elson, J.R. Douceur, J. Howell, and J. Saul. Asirra: a captcha that exploits interest-aligned manual image categorization. In *In Proceedings of ACM CCS*, pages 331–334. ACM, 2007.

[58] K. Etemad and R. Chellappa. Discriminant analysis for recognition of human face images. *Journal of the Optical Society of America A, JOSA A*, 14(8):1724–1733, 1997.

[59] Europa.eu. Data protection: "junk" e-mail costs internet users 10 billion a year worldwide - commission study. `http://europa.eu/rapid/`, 2001.

[60] D.E. Everhart, J.L. Shucard, T. Quatrin, and D.W. Shucard. Sex-related differences in event-related potentials, face recognition, and facial affect processing in prepubertal children. *Neuropsychology*, 15(3):329–341, 2001.

[61] Facebook. Facebook social captcha. `http://www.facebook.com`, 2011.

[62] FERET. Face recognition vendor test. `http://www.frvt.org/`, 2006.

[63] Paul Festa. spam-bot tests flunk the blind, cnet news.com. `http://www.news.com/2100-1032-1022814.html`, July 2, 2003.

[64] I. Fischer and T. Herfet. Visual captchas for document authentication. In *IEEE 8th Workshop on Multimedia Signal Processing*, pages 471–474. IEEE, 2006.

[65] H. Gao, H. Liu, D. Yao, X. Liu, and U. Aickelin. An audio captcha to distinguish humans from computers. In *Third International Symposium on Electronic Commerce and Security (ISECS)*, pages 265–269. IEEE, 2010.

[66] Y. Gao and M.K.H. Leung. Face recognition using line edge map. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(6):764–779, 2002.

[67] I. Gauthier and M.J. Tarr. Becoming a" Greeble" expert: exploring mechanisms for face recognition. *Vision Research*, 37(12):1673–1682, 1997.

[68] P.B. Godfrey. Text-based captcha algorithms. 2002.

[69] A.J. Goldstein, L.D. Harmon, and A.B. Lesk. Identification of human faces. *Proceedings of the IEEE*, 59(5):748–760, 2005.

[70] P. Golle. Machine learning attacks against the asirra captcha. In *Proceedings of the 15th ACM conference on Computer and communications security*, pages 535–542. ACM, 2008.

[71] P. Golle and N. Ducheneaut. Preventing bots from playing online games. *Computers in Entertainment (CIE)*, 3(3):3–3, 2005.

[72] J. Gomes. *Warping and morphing of graphical objects*, volume 1. Morgan Kaufmann, 1999.

[73] J.T. Goodman and R. Rounthwaite. Stopping outgoing spam. In *Proceedings of the 5th ACM conference on Electronic Commerce*, pages 30–39. ACM, 2004.

[74] C.M. Goral, K.E. Torrance, D.P. Greenberg, and B. Battaile. Modeling the interaction of light between diffuse surfaces. In *ACM SIGGRAPH Computer Graphics*, volume 18,

pages 213–222. ACM, 1984.

[75] R. Gossweiler, M. Kamvar, and S. Baluja. What's up captcha?: a captcha based on image orientation. In *Proceedings of the 18th international conference on World wide web*, pages 841–850. ACM, 2009.

[76] R. Gross, J. Shi, and J.F. Cohn. *Quo vadis face recognition?* Citeseer, 2001.

[77] Internet World Stats. Miniwatts Marketing Group. World internet users and population stats. `http://www.internetworldstats.com/stats.htm`, June 2011.

[78] T. Gruter, M. Gruter, and C.C. Carbon. Neural and genetic foundations of face recognition and prosopagnosia. *Journal of Neuropsychology*, 2(1):79–97, 2008.

[79] G. Guo, S.Z. Li, and K. Chan. Face recognition by support vector machines. In *Fourth IEEE International Conference on Automatic Face and Gesture Recognition*, pages 196–201. IEEE, 2000.

[80] B. Heisele, P. Ho, and T. Poggio. Face recognition with support vector machines: Global versus component-based approach. In *Eighth IEEE International Conference on Computer Vision, ICCV 2001*, volume 2, pages 688–694. IEEE, 2001.

[81] B. Heisele, P. Ho, J. Wu, and T. Poggio. Face recognition: component-based versus global approaches. *Computer Vision and Image Understanding*, 91(1):6–21, 2003.

[82] M.E. Hoque, D.J. Russomanno, and M. Yeasin. 2d captchas from 3d models. In *SoutheastCon, 2006. Proceedings of the IEEE*, pages 165–170. IEEE, 2005.

[83] D.H. House. Overview of three-dimensional computer graphics. *ACM Computing Surveys (CSUR)*, 28(1):145–148, 1996.

[84] C.L. Huang and C.W. Chen. Human facial feature extraction for face interpretation and recognition. *Pattern recognition*, 25(12):1435–1444, 1992.

[85] G. Humphrey. The psychology of the gestalt. *Journal of Educational Psychology*, 15(7):401, 1924.

[86] NuCaptcha Inc. Nucaptcha security platform. `http://www.nucaptcha.com/`, 2010.

[87] T. Kanade. *Picture processing system by computer complex and recognition of human faces*. PhD thesis, Dept. of Information Science, Kyoto University, 1973.

[88] T. Kanade. *Computer recognition of human faces*, volume 47. Birkhäuser, 1977.

[89] N. Kanwisher. The ventral visual object pathway in humans: evidence from fmri. *The visual neurosciences*, pages 1179–1189, 2003.

[90] N. Kanwisher, J. McDermott, and M.M. Chun. The fusiform face area: a module in human extrastriate cortex specialized for face perception. *The Journal of Neuroscience*, 17(11):4302–4311, 1997.

[91] Michael G. Kaplan. The 3-d captcha. `http://spamfizzle.com/CAPTCHA.aspx`, 2009.

[92] E.J. Kartaltepe and S. Xi. Towards blocking outgoing malicious impostor emails. In *Proceedings of the 2006 International Symposium on a World of Wireless, Mobile and Multimedia Networks*, pages 5–pp. IEEE, 2006.

[93] S. Karungaru, M. Fukumi, N. Akamatsu, and A. Takuya. Automatic human faces morphing using genetic algorithms based control points selection. *International Journal of Innovative Computing, Information and Control*, 3(2):1–6, 2007.

[94] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. *International journal of computer vision*, 1(4):321–331, 1988.

[95] J.W. Kim, W.K. Chung, and H.G. Cho. A new image-based captcha using the orientation of the polygonally cropped sub-images. *The Visual Computer*, 26(6):1135–1143, 2010.

[96] M. Kirby and L. Sirovich. Application of the karhunen-loeve procedure for the characterization of human faces. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 12(1):103–108, 1990.

[97] R. Kjeldsen and J. Kender. Finding skin in color images. In *Proceedings of the Second International Conference on Automatic Face and Gesture Recognition*, pages 312–317. IEEE, 1996.

[98] K.A. Kluever and R. Zanibbi. Balancing usability and security in a video captcha. In *Proceedings of the 5th Symposium on Usable Privacy and Security*, page 14. ACM, 2009.

[99] G. Kochanski, D. Lopresti, and C. Shih. A reverse turing test using speech. In *Seventh International Conference on Spoken Language Processing, ICSLP*, pages 1357–1360, 2002.

[100] J.J. Koenderink. The structure of images. *Biological cybernetics*, 50(5):363–370, 1984.

[101] K. Koffka. *Principles of Gestalt psychology*. New York, 1955.

[102] A. Kolupaev and J. Ogijenko. Captchas: Humans vs. bots. *Security & Privacy, IEEE*, 6(1):68–70, 2008.

[103] C.E. Kulkarni. Assocaptcha: designing human-friendly secure captchas using word associations. In *CHI '08, CHI Conference on Human Factors in Computing Systems*, pages 3705–3710. ACM, 2008.

[104] R. Kumar, A. Banerjee, and B.C. Vemuri. Volterrafaces: Discriminant analysis using volterra kernels. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2009*, pages 150–155. IEEE, 2009.

[105] M. Lades, J.C. Vorbruggen, J. Buhmann, J. Lange, C. von der Malsburg, R.P. Wurtz, and W. Konen. Distortion invariant object recognition in the dynamic link architecture. *IEEE transactions on Computers*, 42(3):300–311, 1993.

[106] A. Lanitis, C.J. Taylor, and T.F. Cootes. Automatic face identification system using flexible appearance models. *Image and Vision Computing*, 13(5):393–401, 1995.

[107] S. Lawrence, C.L. Giles, A.C. Tsoi, and A.D. Back. Face recognition: A convolutional neural-network approach. *IEEE Transactions on Neural Networks*, 8(1):98–113, 1997.

[108] A.W.F. Lee, D. Dobkin, W. Sweldens, and P. Schröder. Multiresolution mesh morphing. In *Proceedings of SIGGRAPH*, volume 99, pages 343–350, 1999.

[109] S. Lehar. Gestalt isomorphism and the primacy of subjective conscious experience: A gestalt bubble model. *Behavioral and Brain Sciences*, 26(4):375–408, 2003.

[110] TK Leung, MC Burl, and P. Perona. Finding faces in cluttered scenes using random labeled graph matching. In *Computer Vision, 1995. Proceedings., Fifth International Conference on*, pages 637–644. IEEE, 1995.

[111] M.S. Lew. Information theoretic view-based and modular face detection. In *Proceedings of the Second International Conference on Automatic Face and Gesture Recognition*, pages 198–203. IEEE, 1996.

[112] S.Z. Li and A.K. Jain. *Handbook of face recognition*. Citeseer, 2005.

[113] S.Z. Li and J. Lu. Face recognition using the nearest feature line method. *IEEE Transactions on Neural Networks*, 10(2):439–443, 1999.

[114] W.H. Liao. A captcha mechanism by exchange image blocks. In *18th International Conference on Pattern Recognition, ICPR 2006*, volume 1, pages 1179–1183. IEEE, 2006.

[115] W.H. Liao and C.C. Chang. Embedding information within dynamic visual patterns. In *International Conference on Multimedia and Expo (ICME)*, volume 2, pages 895–898. IEEE, 2004.

[116] S.H. Lin, S.Y. Kung, and L.J. Lin. Face recognition/detection by probabilistic decision-based neural network. *IEEE Transactions on Neural Networks*, 8(1):114–132, 1997.

[117] T. Lindeberg. Scale-space theory: A basic tool for analyzing structures at different scales. *Journal of applied statistics*, 21(1-2):225–270, 1994.

[118] D. Lopresti. Leveraging the captcha problem. *Human Interactive Proofs*, pages 31–45, 2005.

[119] LoveToKnow. Define online marketing. `http://reference.yourdictionary.com/word-definitions/define-online-marketing.html`, April 2012.

[120] D.G. Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

[121] P. Lupkowski and M. Urbanski. Semcaptchauser-friendly alternative for ocr-based captcha systems. In *International Multiconference on Computer Science and Information Technology, IMCSIT 2008*, pages 325–329. IEEE, 2008.

[122] MAAWG. Metrics report. `http://www.maawg.org/email_metrics_report`, 2011.

[123] BS Manjunath, R. Chellappa, and C. von der Malsburg. A feature based approach to face recognition. In *Computer Vision and Pattern Recognition, 1992. Proceedings CVPR'92., 1992 IEEE Computer Society Conference on*, pages 373–378. IEEE, 1992.

[124] P. Matthews and C.C. Zou. Scene tagging: image-based captcha using image composition and object relationships. In *Proceedings of the 5th ACM Symposium on Information, Computer and Communications Security*, pages 345–350. ACM, 2010.

[125] S.J. McKenna, S. Gong, and Y. Raja. Modelling facial colour and identity with gaussian mixtures. *Pattern recognition*, 31(12):1883–1892, 1998.

[126] G.J. McLachlan. Discriminant analysis and statistical pattern recognition. *Wiley Series in Probability and Mathematical Statistics*, 1, 1992.

[127] D. Misra and K. Gaj. Face recognition captchas. In *International Conference on Internet and Web Applications and Services/Advanced, AICT-ICIW'06*, pages 122–122. IEEE, 2006.

[128] B. Moghaddam, C. Nastar, and A. Pentland. Bayesian face recognition using deformable intensity surfaces. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR'96*, pages 638–645. IEEE, 1996.

[129] B. Moghaddam, W. Wahid, and A. Pentland. Beyond eigenfaces: Probabilistic matching for face recognition. In *Automatic Face and Gesture Recognition, 1998. Proceedings. Third IEEE International Conference on*, pages 30–35. IEEE, 2002.

[130] G. Mori, S. Belongie, and J. Malik. Efficient shape matching using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(11):1832–1837, 2005.

[131] G. Mori and J. Malik. Recognizing objects in adversarial clutter: Breaking a visual captcha. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 1, pages I–134. IEEE, 2003.

[132] D. Morrison, S. Marchand-Maillet, and É. Bruno. Tagcaptcha: annotating images with captchas. In *Proceedings of the ACM SIGKDD Workshop on Human Computation*, pages 44–45. ACM, 2009.

[133] G. Moy, N. Jones, C. Harkless, and R. Potter. Distortion estimation techniques in solving visual captchas. In *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2004*, volume 2, pages II–23. IEEE, 2004.

[134] C. Nampempre and M.N. Dailey. Mitigating dictionary attacks with text-graphics character captchas. *IEICE TRANSACTIONS on Fundamentals of Electronics, Communications and Computer Sciences*, 90(1):179–186, 2007.

[135] A.V. Nefian. Embedded bayesian networks for face recognition. In *IEEE International Conference on Multimedia and Expo, ICME'02*, volume 2, pages 133–136. IEEE, 2002.

[136] A.V. Nefian and M.H. Hayes III. Hidden markov models for face recognition. In *Acoustics, Speech and Signal Processing, 1998. Proceedings of the 1998 IEEE International Conference on*, volume 5, pages 2721–2724. IEEE, 1998.

[137] A.V. Nefian and M.H. Hayes III. Maximum likelihood training of the embedded hmm for face detection and recognition. In *International Conference on Image Processing*, volume 1, pages 33–36. IEEE, 2000.

[138] C.A. Nelson. The development and neural bases of face recognition. *Infant and Child Development*, 10(1-2):3–18, 2001.

[139] E. Osuna, R. Freund, and F. Girosit. Training support vector machines: an application to face detection. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 130–136. IEEE, 1997.

[140] A.J. OToole, K.A. Deffenbacher, D. Valentin, and H. Abdi. Structural aspects of face recognition and the other-race effect. *Memory & Cognition*, 22(2):208–224, 1994.

[141] P.S. Penev and J.J. Atick. Local feature analysis: A general statistical theory for object representation. *Network: computation in neural systems*, 7(3):477–500, 1996.

[142] A. Pentland, B. Moghaddam, and T. Starner. View-based and modular eigenspaces for face recognition. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR'94*, pages 84–91. IEEE, 1994.

[143] P.J. Phillips, H. Moon, S.A. Rizvi, and P.J. Rauss. The FERET evaluation methodology for face-recognition algorithms. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(10):1090–1104, 2002.

[144] A. Pinar Saygin, I. Cicekli, and V. Akman. Turing test: 50 years later. *Minds and Machines*, 10(4):463–518, 2000.

[145] T. Porter and T. Duff. Compositing digital images. *ACM Siggraph Computer Graphics*, 18(3):253–259, 1984.

[146] A. Raj, A. Jain, T. Pahwa, and A. Jain. Picture captchas with sequencing: Their types and analysis. *International Journal for Digital Society (IJDS)*, 1(3):208 – 220, 2010.

[147] AN Rajagopalan, K.S. Kumar, J. Karlekar, R. Manivasakan, M.M. Patil, U.B. Desai, PG Poonacha, and S. Chaudhuri. Finding faces in photographs. In *Sixth International Conference on Computer Vision*, pages 640–645. IEEE, 1998.

[148] J. Rehnman and A. Herlitz. Higher face recognition ability in girls: Magnified by own-sex and own-ethnicity bias. *Memory*, 14(3):289–296, 2006.

[149] Security MVP Robert Moir. Defining malware: Faq. `http://technet.microsoft.com`, October 2003.

[150] C. Romero Macias and E. Izquierdo. Visual word-based captcha using 3d characters. In *3rd International Conference on Crime Detection and Prevention (ICDP 2009)*, pages 1–5. IET, 2009.

[151] C. Romero Macias and E. Izquierdo. Image captcha based on distorted faces. In *4th International Conference on Imaging for Crime Detection and Prevention (ICDP 2011)*,

pages 1–6. IET, 2011.

[152] S.A. Ross, J.A. Halderman, and A. Finkelstein. Sketcha: a captcha based on line drawings of 3d models. In *Proceedings of the 19th international conference on World wide web*, pages 821–830. ACM, 2010.

[153] S.D. Roth. Ray casting for modeling solids. *Computer Graphics and Image Processing*, 18(2):109–144, 1982.

[154] H.A. Rowley, S. Baluja, and T. Kanade. Neural network-based face detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 20(1):23–38, 2002.

[155] Yong Rui and Zicheg Liu. Artifacial: automated reverse turing test using facial features. In *Proceedings of the eleventh ACM international conference on Multimedia*, pages 295–298. ACM, 2003.

[156] A. Rusu and V. Govindaraju. Handwritten captcha: Using the difference in the abilities of humans and machines in reading handwritten words. In *Ninth International Workshop on Frontiers in Handwriting Recognition (IWFHR-9)*, pages 226–231. IEEE, 2004.

[157] Y.S. Ryu and S.Y. Oh. Automatic extraction of eye and mouth fields from a face image using eigenfeatures and multilayer perceptrons. *Pattern recognition*, 34(12):2459–2466, 2001.

[158] V. Saalo. Novel captcha schemes. `http://www.cse.hut.fi/en/publications/B/11/`, 2010.

[159] Microsoft Safety and Security Center. How to recognize phishing email messages, links, or phone calls. `http://www.microsoft.com/security/online-privacy/phishing-symptoms.aspx`, 2012.

[160] S. Saklikar and S. Saha. Public key-embedded graphic captchas. In *5th IEEE Consumer Communications and Networking Conference (CCNC)*, pages 262–266. IEEE, 2008.

[161] G. Sauer, H. Hochheiser, J. Feng, and J. Lazar. Towards a universally usable captcha. In *Proceedings of the Symposium on Accessible Privacy and Security, ACM Symposium On Usable Privacy and Security (SOUPS'08)*. ACM, 2008.

[162] G. Sauer, J. Holman, J. Lazar, H. Hochheiser, and J. Feng. Accessible privacy and security: a universally usable human-interaction proof tool. *Universal Access in the Information Society*, 9(3):239–248, 2010.

[163] P. Schmalfeldt and J.Kramlich. Humanauth. `http://sourceforge.net/projects/humanauth/`, 2007.

[164] H. Schneiderman and T. Kanade. Probabilistic modeling of local appearance and spatial relationships for object recognition. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 45–51. IEEE, 1998.

[165] N.A. Shah and M.T. Banday. Drag and drop image captcha. *Sprouts: Working Papers on Information Systems*, 8(46), 2008.

[166] Y. Shirai. *Three-dimensional computer vision*. Springer-Verlag New York, Inc. New York, NY, USA, 1987.

[167] M. Shirali-Shahreza. Captcha for children. In *IEEE SMC Third International Conference on System of Systems Engineering*, pages 1–6. Ieee, 2008.

[168] M. Shirali-Shahreza and S. Shirali-Shahreza. Drawing captcha. In *28th International Conference on Information Technology Interfaces*, pages 475–480. IEEE, 2006.

[169] M. Shirali-Shahreza and S. Shirali-Shahreza. Captcha for blind people. In *IEEE International Symposium on Signal Processing and Information Technology*, pages 995–998. IEEE, 2007.

[170] M. Shirali-Shahreza and S. Shirali-Shahreza. Collage captcha. In *9th International Symposium on Signal Processing and Its Applications, ISSPA 2007*, pages 1–4. IEEE, 2007.

[171] M. Shirali-Shahreza and S. Shirali-Shahreza. Passwordless login system for mobile phones using captcha. In *49th International Symposium ELMAR*, pages 243–246. IEEE, 2007.

[172] M. Shirali-Shahreza and S. Shirali-Shahreza. Question-based captcha. In *International Conference on Computational Intelligence and Multimedia Applications*, volume 4, pages 54–58. IEEE, 2007.

[173] M. Shirali-Shahreza and S. Shirali-Shahreza. Motion captcha. In *2008 Conference on Human System Interactions*, pages 1042–1044. IEEE, 2008.

[174] M.H. Shirali-Shahreza and M. Shirali-Shahreza. Localized captcha for illiterate people. In *International Conference on Intelligent and Advanced Systems, ICIAS*, pages 675–679. IEEE, 2007.

[175] S. Shirali-Shahreza and A. Movaghar. A new anti-spam protocol using captcha. In *IEEE International Conference on Networking, Sensing and Control*, pages 234–238. IEEE, 2007.

[176] S. Shirali-Shahreza, M. Shirali-Shahreza, and A. Movaghar. Restricted access to exam grades on the web by hip. In *6th IEEE/ACIS International Conference on Computer and Information Science (ICIS)*, pages 967–971. IEEE, 2007.

[177] L. Sirovich and M. Kirby. Low-dimensional procedure for the characterization of human faces. *Journal of the Optical Society of America A*, 4(3):519–524, 1987.

[178] AE Slone, JB Brigham, and CA Meissner. Social and cognitive factors affecting the own-race advantage in white. *Basic and Applied Social Psychology*, 22:71–84, 2000.

[179] K.K. Sung and T. Poggio. Example-based learning for view-based human face detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 20(1):39–51, 2002.

[180] W. Susilo, Y.W. Chow, and H.Y. Zhou. Ste3d-cap: Stereoscopic 3d captcha. *Cryptology and Network Security*, pages 221–240, 2010.

[181] J. Tam, J. Simsa, S. Hyde, and L. Von Ahn. Breaking audio captchas. *Advances in Neural Information Processing Systems*, 1(4), 2008.

[182] M. Tariq Banday and N.A. Shah. A study of captchas for securing web services. *International Journal of Secure Digital Information Age IJSDIA*, 2011.

[183] A. Thayananthan, B. Stenger, P.H.S. Torr, and R. Cipolla. Shape context and chamfer matching in cluttered scenes. In *Proceedubgs of the 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2003*, volume 1, pages I–127. IEEE, 2003.

[184] Y. Tian, T. Kanade, and J.F. Cohn. Evaluation of gabor-wavelet-based facial action unit recognition in image sequences of increasing complexity. In *Automatic Face and Gesture Recognition, 2002. Proceedings. Fifth IEEE International Conference on*, pages 229–234. IEEE, 2002.

[185] L. Torres. Is there any hope for face recognition? In *Proceedings of the 5th International Workshop on Image Analysis for Multimedia Interactive Services, WIAMIS*, volume 21, page 23, 2004.

[186] M. Turk and A. Pentland. Eigenfaces for recognition. *Journal of cognitive neuroscience*, 3(1):71–86, 1991.

[187] L. Velho, A.C. Frery, and J. Gomes. *Image processing for computer graphics and vision*. Springer-Verlag New York Inc, 2008.

[188] L. Von Ahn, M. Blum, and J. Langford. Telling humans and computers apart automatically. *Communications of the ACM*, 47(2):56–60, 2004.

[189] N.R. Wagner. Captchas and information hiding. http://www.cs.utsa.edu/ wagner/captcha/, 2003.

[190] P.M. Walker and J.W. Tanaka. An encoding advantage for own-race versus other-race faces. *Perception*, 32(9):1117–1126, 2003.

[191] O. Warner. The cutest human-test: Kittenauth. `http://www.thepcspy.com/kittenauth`, 2006.

[192] A. Watt. *Fundamentals of three-dimensional computer graphics*. Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA, 1990.

[193] H. Wechsler. *Reliable face recognition methods: system design, implementation and evaluation*. Springer-Verlag New York Inc, 2007.

[194] Eric W. Weisstein. Affine transformation. MathWorld–A Wolfram Web Resource,`http://mathworld.wolfram.com/AffineTransformation.html`, 1999.

[195] A. Wettlaufer. *In the mind's eye: the visual impulse in Diderot, Baudelaire and Ruskin*. Rodopi, 2003.

[196] J. Wilkins. Strong captcha guidelines v1. 2. *Retrieved Nov*, 10:2010, 2009.

[197] L. Wiskott, J.M. Fellous, N. Kuiger, and C. von der Malsburg. Face recognition by elastic bunch graph matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):775–779, 1997.

[198] L. Wiskott and C. Malsburg. Recognizing faces by dynamic link matching. *Neuroimage*, 4(3):S14–S18, 1996.

[199] G. Wolberg. Image morphing: a survey. *The Visual Computer*, 14(8):360–372, 1998.

[200] G. Wolberg, Institute of Electrical, and Electronics Engineers. Computer Society. *Digital image warping*, volume 10662. IEEE computer society press California, 1990.

[201] S. Wold, K. Esbensen, and P. Geladi. Principal component analysis. *Chemometrics and intelligent laboratory systems*, 2(1):37–52, 1987.

[202] P. Ximenes, A. dos Santos, M. Fernandez, and J. Celestino. A captcha in the text domain. In *On the Move to Meaningful Internet Systems 2006: OTM 2006 Workshops*, pages 605–615. Springer, 2006.

[203] T. Yamamoto, T. Suzuki, and M. Nishigaki. A proposal of four-panel cartoon captcha: The concept. In *13th International Conference on Network-Based Information Systems (NBiS)*, pages 575–578. IEEE, 2010.

[204] R.V. Yampolskiy. Detecting and controlling cheating in online poker. In *5th IEEE Consumer Communications and Networking Conference (CCNC)*, pages 848–853. IEEE, 2008.

[205] J. Yan and A.S. El Ahmad. Breaking visual captchas with naive pattern recognition algorithms. In *Twenty-Third Annual Computer Security Applications Conference (ACSAC)*, pages 279–291. IEEE, 2007.

[206] J. Yan and A.S. El Ahmad. A low-cost attack on a microsoft captcha. In *Proceedings of the 15th ACM conference on Computer and Communications Security*, pages 543–554. ACM, 2008.

[207] J. Yan and A.S. El Ahmad. Usability of captchas or usability issues in captcha design. In *Proceedings of the 4th symposium on Usable Privacy and Security*, pages 44–52. ACM, 2008.

[208] G. Yang and T.S. Huang. Human face detection in a complex background. *Pattern recognition*, 27(1):53–63, 1994.

[209] J. Yang and A. Waibel. A real-time face tracker. In *Proceedings 3rd IEEE Workshop on Applications of Computer Vision, WACV'96.*, pages 142–147. IEEE, 1996.

[210] M.H. Yang, D.J. Kriegman, and N. Ahuja. Detecting faces in images: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(1):34–58, 2002.

[211] M.H. Yang, D. Roth, and N. Ahuja. A snow-based face detector. In *Advances in Neural Information Processing Systems 12*. Citeseer, 2000.

[212] G. Yovel and N. Kanwisher. Face perception:: Domain specific, not process specific. *Neuron*, 44(5):889–898, 2004.

[213] K.C. Yow and R. Cipolla. Feature-based human face detection. *Image and Vision Computing*, 15(9):713–735, 1997.

[214] A.L. Yuille, P.W. Hallinan, and D.S. Cohen. Feature extraction from faces using deformable templates. *International journal of computer vision*, 8(2):99–111, 1992.

[215] W. Zhao, R. Chellappa, and A. Krishnaswamy. Discriminant analysis of principal components for face recognition. In *Third IEEE International Conference on Automatic Face and Gesture Recognition*, pages 336–341. IEEE, 1998.

[216] W. Zhao, R. Chellappa, P.J. Phillips, and A. Rosenfeld. Face recognition: A literature survey. *Acm Computing Surveys (CSUR)*, 35(4):399–458, 2003.

[217] W. Zhao and Md.). Computer Vision Laboratory University of Maryland (College Park. *Robust image based 3D face recognition*. Computer Vision Laboratory, Center for Automation Research, University of Maryland, 2000.

[218] B.B. Zhu, J. Yan, Q. Li, C. Yang, J. Liu, N. Xu, M. Yi, and K. Cai. Attacks and design of image recognition captchas. In *Proceedings of the 17th ACM conference on Computer and communications security*, pages 187–200. ACM, 2010.