Cryptographically Secure Identity Certificates

EMO PROJE YARIŞMASI

Hazırlayan : Aslı Deniz YİĞİT
Boğaziçi Üniversitesi

Supervisor: Prof.Dr. Emin ANARIM

Haziran 2007

ABSTRACT

Forgery is one of the techniques of fraud, including identity theft [1]. Due to the increase on the number of the forgery in the digital platform, an irresistible need for the new authorization techniques have occurred.
This project has been prepared to prevent such forgery attempts especially to validate the identity cards. The validation section consists of four main parts. These are scanning the identity card, (if needed) rotating the image of the card to make it horizontal, grabbing the owner's image including the text, compare the image and the text with the image and text which are embedded in the smart card, also their hash bits. By observing the results of the whole algorithm, the validation of the identity card can be made successfully.

These are scanning the identity card, (if needed) rotating the image of the card to make it horizontal, grabbing the owner's image including the text, creating the hash bits based upon to the grabbed image and the text and then comparing the hash bits with the bits taken from the barcode or the smart card.

INTRODUCTION:

"A typical identity certification such as a driver's license, passport, or visa, consists of a personal portrait photo, an arbitrary message, and one or more features whose purpose is to guarantee authenticity" [1]. Majority of the today's ID cards contain a picture of the authorized user, a simple and effective form of biometric identification. Due to advance in the computer technology, high-resolution scanners and printers and photo editing software have been produced. The usage of these products makes the forgery simpler than ever. Creating fake ID cards can be made even by using a simple inkjet or laser printer. These print-outs are integrated with similar card designs and plastic structures which resemble the original ones. The technical details of this forgery has been widely spread out over the internet forums and websites. Unfortunately, this illegal method has found a large community over the world. Furthermore, ready-to-use templates for many types of ID cards can be easily found on the Internet.
To prevent these kinds of fraud attempts, various secure printing techniques have been advanced to make the ID cards more secure. For instance, many modern cards contain holograms, that are difficult to clone unless expensive equipments are used. Another example of this is the common usage of barcodes. In the U.S. most of the driving licences include a 2-dimensional barcode, which contains the same information as on the front of the license. Barcodes keep the information of the involved owner in its digital structure. They also make the frequent control of credentials for low-security applications possible [2].

The examples can be incremented. For example, some cards carry a magnetic strip which will also contain the same information on their back sides. By means of this, the information embedded on the barcode and the magnetic strip can be compared to validate the cards authentication. The magnetic strips may also contain different identifying information. Although magnetic strips can also be cracked, they provide extra security to the card against an intermediate forger [2].
Other hidden security devices can also be implemented to the card, including embedded secure chips which are designed to be very difficult to forge: the two technologies may also be combined, in the case of smart cards.
In this project by inspiring these advanced methods, barcode and smart card based authentication method has been discussed.

MODULE I-Scanning and Rotating the Image

In this part of the project; first of all, an image on an identity card is scanned. Some criteria should be satisfied for a successful image processing. For example, the image which will be parallel to the sides of the working space. For this reason, the image must be rotated with the needed angle which can be calculated by an algorithm. At the further parts of this section, this algorithm will be explained.
In the first part of the first module, the alignment process of the card's view is done. "kobay.m" is responsible from this job. It first gets a copy of the original scanned image, then binaries it and fills image regions and holes. Then morphological open and close processes are done to prevent salt and pepper noise respectively. After these, a search algorithms starts to find two white pixels on upper side of the card. In order to find the correct slope of the card, these points should not be taken from the rounded corners of the card. Therefore, the algorithms takes two pixels from the middle of the upper side of the card. After calculating the slope and the angle corresponding to that slope, the program rotates the image with the angle calculated. So the card view at the scanned image becomes zero angle with the horizontal angle.

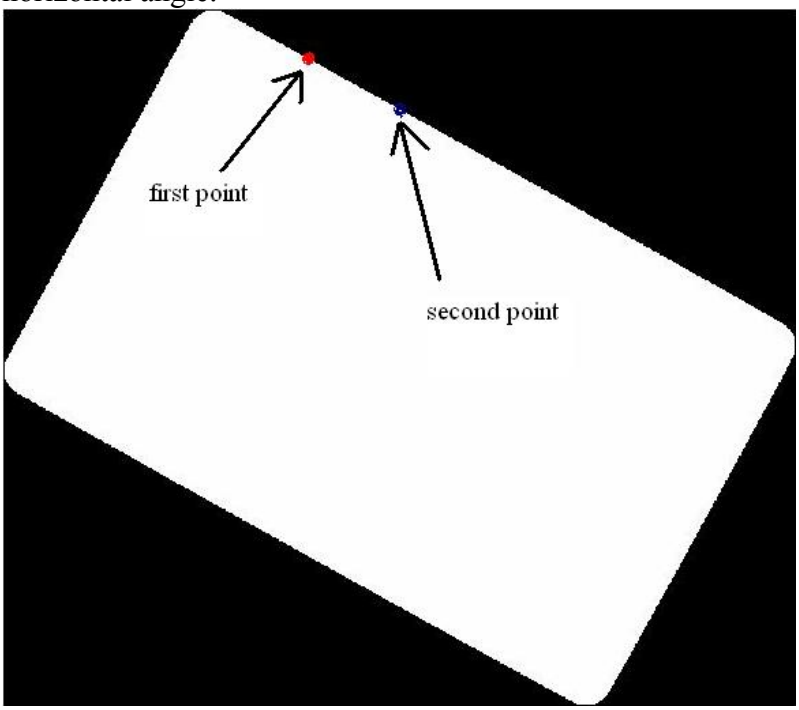

Figure 1: Example of finding the points to calculate the slope

In the second part of the first module, grabbing the photograph at the card is made. "border.m" is responsible from this job. It first gets a copy of the original scanned image, then binerazes it and fills image regions and holes. Then morphological open and close processes are done to prevent salt and pepper noise respectively. The algorithms first calculates the maximum and the minimum row and column values of the white pixels at the binerazed image and depending on these values the scanned and rotated image is cropped depending only these coordinates. Therefore, only the cards view is obtained. However, there are still problems due to rounded corners of the card. In order to get rid of them, for 200dpi scanned image 50 pixels are cut form upper, lower and the left side and 25 pixels are cut from right side of the card. So, a search algorithm can be started to find the location of the photograph and to crop only it. The algorithm searching for the photograph works as follows:

- A point (p_left) somewhere in the middle of the text and the photograph is found.
- Starting from that point and going to the right side, algorithm searches for black pixel. When it finds the first black pixel (the column value of that point will be a left border line for the photograph), it gets the coordinates of it.
- Also three points are searched, starting from top, bottom and the right most of the card to find such points which will be used for clipping the photograph.
- After finding four points, clipping is done.

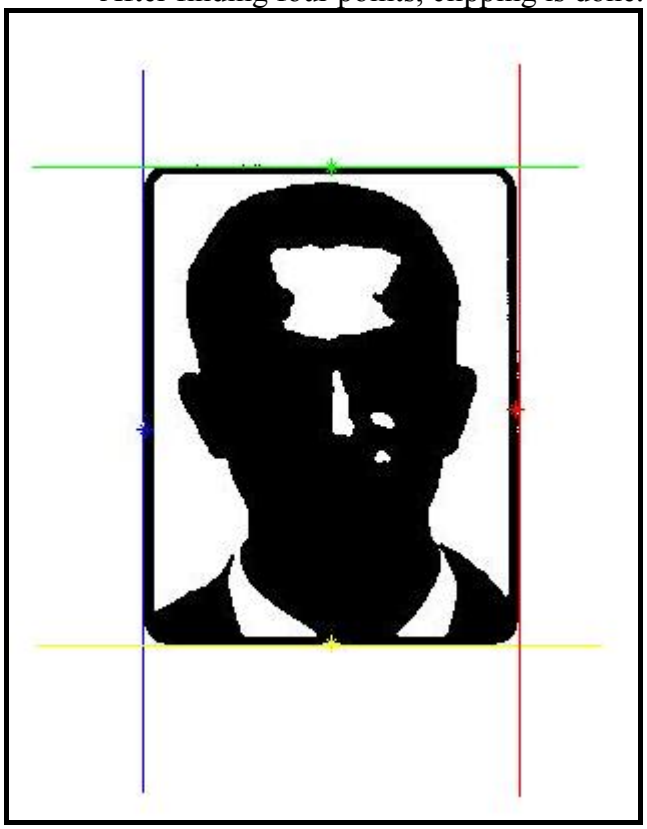

Figure 2: Four points which will be used for clipping

MODULE II-Grabbing Text from Image

This module is created for being able to convert an image of text, such as a scanned paper, into computer-editable text. The text in an image is not editable: the letters are made of tiny dots (pixels) that together form a picture of text. During Optical Character Recognition (OCR), the software analyzes an image and converts the pictures of the characters to editable

text based on the patterns of the pixels in the image. After OCR, you can export the converted text and use it with a variety of word-processing [4].

Specifications

Before implementing OCR, specifications must have be determined. Those are:

- The characters in the image should be written by English 'Arial' type with capital letter whose font size is 16.
- For high performance punctuation marks such as 'j, :, ?, %, !, ;' are not considered.

Implementation of OCR, mainly depends on the theory of correlation. OCR decided the letters by looking at the correlation outputs. So it is useful to mention the correlation definition here.

Theory

The objective in computing the correlation between the two signals is to measure the degree to which the two signals are similar and thus to extract some information that depends to a large extent on the application. Correlation of signals is often encountered in radar, sonar, digital communications, geology, and other areas in science and engineering [5]. The correlation between signals x(n) and y(n) is:

$r_{xy}(l)=Ó\ x(n)y(n-l) \quad l=….-1, 0, 1, 2, …..$ (1)

If any one or both of the signals involved in the cross-correlation are scaled, the shape of the cross correlation sequence does not change; only the amplitudes of the cross-correlation sequence are scaled accordingly. Since scaling is unimportant, it is often desirable, in practice, to normalize the autocorrelation and cross correlation sequences to the range from -1 to 1. Thus normalized cross-correlation sequence is:

$ñ_{xy}(l) = r_{xy}(l) / (r_{xx}(0)*r_{yy}(0))^{1/2}$ (2)

In OCR the correlation between two images are computed. These images are represented by 2 dimensional matrices so we used the algorithm of 2-D cross-correlation and by looking at the normalized cross-correlation values a decision upon the letter is made. Normalization is important since each letter image is 2-D matrices with different dimensions and contents.

Method

The path followed during the project is given in the figure below:

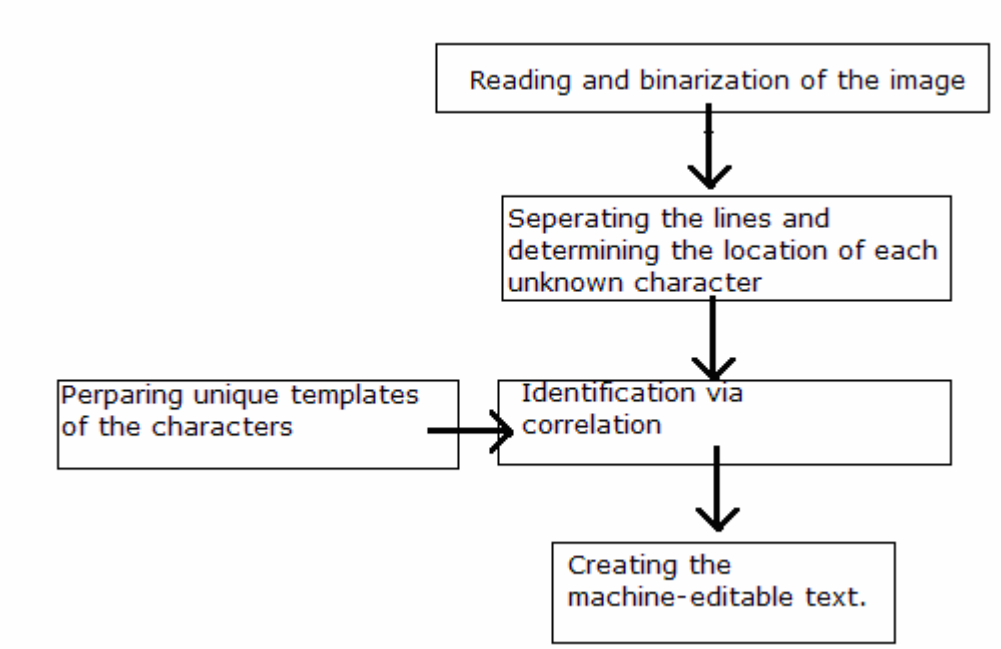

Figure 3: Paths followed during OCR

It is necessary to explain the figure briefly. At first an image composed of capital letters, whose font is Arial and font size is16, created.. Then the image is loaded into MATLAB for applying image processing applications. After loading the image, a binerization is applied on it which changes the value of the pixels pure white and pure black according to a specific threshold value. Now it is time to separate the lines and characters hence determine the location of character images. Now each character images obtained individually and therefore A decision upon corresponding character ASCII values can be made by using previously prepared template. Template consists of 86 character images and corresponding character ASCII values. Identification of character images can be done via 2-D cross-correlation. Among all the elements in the template, the one having the maximum cross correlation result with the unknown character of the image is decided to be the unknown. This process is applied for all the unknown character of the images. This procedure gives the machine-editable text. The codes for OCR can be found at the Appendix B.

MODULE III - Hashing Algorithms

A hash function is a reproducible method of turning some kind of data into a (relatively) small number that may serve as a digital "fingerprint" of the data.
To create such fingerprints which are also known as hash codes,hash values or simply hashes, algorithm hashes, in other words “chops and mixes” the data. A well designed cryptographic hash function is a "one-way" operation, that means one can not obtain the input data from its hash codes,therefore forgery is also very difficult. In order to distinguish whether the data has been disturbed for example during the transmission, functions for error detection and correction are used.As seen, hash functions are so suitable for authentication process[6].

Hash functions can be used for various applications such as determining data at the input file is changed or not. If a hash value is calculated for a piece of data, and then one bit of that data is changed, a hash function with a strong mixing property usually produces a completely different hash value [6]. However, in this project a hashing function which is robust to some

imperfections that result in the practical usage, such as rotation or “wear-and-tear” effects of the photo on the ID-card. Therefore, PRSQ Hashing [8] algorithms is implemented.

CONCLUSION:

This project is composed of three modules. First and the third module works without any problem. However second module has a problem. The problem is based on the template of OCR algorithm. Template is prepared at the paint with font Arial and font size of 16. At the paint, for example, For character of “A” it gives 16*16 pixels. However, Ocr gets an input with a high resolution for such an input same character can be represented 130*100 pixels. Because the OCR algorithm looks one-to-on correlation between the template and the input, It was not able to find any correct character. However, if the input image of the OCR is resized by a factor which should be found by experimentally for a constant dpi value, the algorithm can give correct results with approximately 90%.But, since hash values are used to verify the cards authentication, any wrong output taken from the OCR makes the hashes to change a lot, and this will cause a false alarm. In order to prevent these things, an OCR algorithm which is suitable for the specifications of this project must be implemented. Implementation of a new and suitable OCR algorithm and some different robust hashing algorithms can be future work for this project.