International Journal of Engineering & Technology, 7 (1.1) (2018) 43-46

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

Optimization of biometric recognition using cuckoo search algorithm: a preliminary version for minutia based fingerprint identification

Dhileepan Thangamanimaran[1*], M. Sharat Chandar[2], S. Chandia[3]

[1]II Year M.Sc (Software Systems), Department of Computing, Coimbatore Institute of Technology, Coimbatore, India.
[2]Department of Computing, Coimbatore Institute of Technology, Coimbatore, India
[3]Assistant Professor, Department of Computing, Coimbatore Institute of Technology, Coimbatore, India
**Corresponding author E-mail:dhileepan123@gmail.com*

Abstract

Currently Behavioural Biometrics is the most widely used means of security. Though Behavioural Biometrics is highly reliable and secure, the data handling process is quite complex. This Problem can be solved by optimizing the process using cuckoo search algorithm. This Paper seeks to optimize the process of fingerprint matching by using an optimal algorithm. The Minutiae in the form of a matrix is extracted from a fingerprint. The Matrix is then split into smaller matrices with increasing dimension and then compared. The matrix with least dimension it is matched. If the Match is true then the verification of next generation bigger matrix is done. If the Match tends to be false then the matrix is skipped and the process is continued for the next matrix in the database. The Process is repeated until accurate match is obtained.
Though the time reduced by the optimization of the finger print matching algorithm is insignificant for a smaller data set such as finger print data, it can be a key factor when a larger set of Behavioural biometrics data is considered.

Keywords: *Minutiae, Cuckoo Search, Optimization, Fingerprint.*

1. Introduction

We are always at a risk of losing everything we hold due to loss of personal data. This is where the security of data becomes an issue. The existing methods of authenticating a person can be easily exploited. Biometric methods such as finger print recognition, facial recognition and iris recognition can also be exploited using a photograph to gain access.
This era of technology and innovation has paved way for a new breed of criminals, "The Identity Thieves". Criminals can steal your passwords, pin numbers and even your basic biometric features. The Only thing they can't steal or replicate from you is your behaviour. Every person is unique and has a distinct lifestyle. Certain features like our style of walking, reacting to events etc. do mark us apart. [6][7]
Behavioural biometrics can verify how the people are familiar with the data and the application they are using by assessing how they engage with it. It works on the behavioural pattern of an individual and hence does not require a modification in the user experience. Also it doesn't require any new kind of hardware; it can be done with existing hardware but including a software analysis. They can be collected non-obtrusively or even without the knowledge of the user. [8]
Behavioural biometrics can also be used in diagnosis of diseases by analysing the anomality in the data pattern. For Example, let us consider an individual's writing pattern, when you have Alzheimer's disease, your mental faculties deteriorate as does your handwriting. If letters in the same sentence are frequently slanting in different directions, this could be a sign of schizophrenia. Some researchers conclude that Parkinson's disease can also be detected in handwriting. Very small or cramped writing, to the point where the letters are so small that even the writer can't read them, is an indicator. [3][4][5]
There are two fundamental processes involved in biometric systems: identification and verification. Both the processes require storage of large amounts of data. One of the most important factors that renders behavioural biometrics impossible to implement is the complexity of data. Many Factors are considered in analysing a person's behavioural biometric pattern. This type of data analysis requires lot of processor time and it is inefficient.
One of the ways of solving this problem is by optimization of data used for analysis. To overcome the limits of traditional data analysis techniques, we propose a cuckoo-search based algorithm for behavioural biometric verification and thereby increasing the overall performance of the system.
Cuckoo-search is a metaheuristic optimization algorithm developed by Xin-she Yang and Suash Deb. It was inspired by the obligate brood parasitism of some cuckoo species by laying their eggs in the nests of other host birds of other species. Some host birds can engage direct conflict with the intruding cuckoos. Some cuckoo species have evolved in such a way that female parasitic cuckoos are often very specialized in the mimicry in colours and pattern of the eggs of a few chosen host species. Cuckoo search idealized such breeding behaviour, and thus can be applied for behavioural biometrics analysis.
We have put forth a fingerprint recognition algorithm, a modified version of minutiae fingerprint recognition algorithm in which cuckoo search is used for optimisation. We find this as an analysis of using optimisation algorithms in biometric recognition and identification to overcome drawbacks of traditional data analysis techniques in complex data.

2. General cuckoo-search algorithm

Cuckoo search uses the following representations:
Each egg in a nest represents a solution, and a cuckoo egg represents a new solution. The aim is to use the new and potentially better solutions (cuckoos) to replace a not-so-good solution in the nests. In the simplest form, each nest has one egg. The algorithm can be extended to more complicated cases in which each nest has multiple eggs representing a set of solutions.
CS is based on three idealized rules:

1. Each cuckoo lays one egg at a time, and dumps its egg in a randomly chosen nest;
2. The best nests with high quality of eggs will carry over to the next generation;
3. The number of available hosts nests is fixed, and the egg laid by a cuckoo is discovered by the host bird with a probability Pa ∈ (0,1). [1]

begin
Objective function f(x), x = (x1, ..., xd) ^T
Generate initial population of n host nests xi (i = 1, 2, ..., n).
while (t <MaxGeneration) or (stop criterion)
Get a cuckoo randomly by L´evy flights evaluate its quality/fitness Fi.
Choose a nest among n (say, j) randomly
if (Fi > Fj)
replace j by the new solution;
end
A fraction (pa) of worse nests are abandoned and new ones are built;
Keep the best solutions (or nests with quality solutions);
Rank the solutions and find the current best
end while
Postprocess results and visualization
End

3. Fingerprint analysis

Fingerprint recognition

Human Fingerprints are unique, difficult to alter and durable. Fingertips contain ridges and valleys which forms distinctive patterns. Fingerprints are distinguished by certain features called Minutiae. Among the various types of minutia, the following are most significantly used:

- Ridge ending-the abrupt end of a ridge
- Ridge bifurcation- a single ridge that divides into two

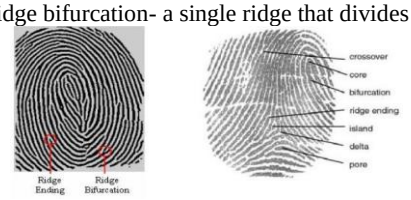

Fig. 1: Types of minutiae

A fingerprint recognition system consists of fingerprint acquiring device, minutia extractor and minutia matcher.

Fingerprint acquisition

A Fingerprint sensor is used to capture a digital image of the fingerprint. The image processed to create a biometric template which is stored and used for matching with previously stored templates. Commonly used Fingerprint sensors include optical, capacitive, thermal, piezo resistive and ultrasonic.
Fingerprint image is enhanced to obtain a clear Image. Various Image enhancement techniques are employed to reduce the noise and enhance the definition of ridges against valleys. Then the Minutia is extracted and compared with a template stored.

Histogram equalization

Histogram equalization is a technique for adjusting image intensities to enhance contrast by adjusting the intensity distribution on a histogram. This allows areas of lower local contrast to gain a higher contrast without affecting the global contrast. This method is useful in images with backgrounds and foregrounds that are both bright or both dark. [2]

Binarization

Fingerprint-Image-Binarization transforms a gray image to a 1-bit binarized image. Most minutiae extraction algorithms operate on binary images where there are only two levels of interest: the black pixels which represents ridges and the white pixels which represents the valleys. This Improves the Contract between the ridges and valleys. [3]

Thinning

Thinning is a morphological operation that successively erodes away foreground pixels until they are one-pixel wide. Thinning is normally only applied to binary images, and produces another binary image as output. It is the final step prior to minutiae extraction. [3]

Minutiae extraction

This method extracts the ridge endings and bifurcations from the skeleton image by examining the local neighbourhood of each ridge pixel using a 3×3 window. The ridge can be divided into bifurcation, ridge ending and non-minutiae point based on it.

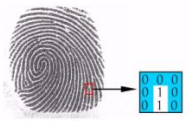

Fig. 2: Binarized ridge ending

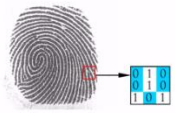

Fig. 3: Binarized Ridge bifurcation

Minutiae matching

Minutiae Obtained are plotted in a matrix form and a cuckoo search based matching algorithm is employed. The eggs are taken as sub matrix of the fingerprint 1-bit grey scale image. The eggs which survives are taken as the next generation cuckoo which again lays eggs. They new eggs laid are of greater dimension matrix in which the previous generation elements are also a subset. The cuckoo will be killed at an instance when the anomality is detected i.e. when the required level of significance is obtained the cuckoo survives else the vice versa.
The cuckoos will lay eggs till a most significant match of fingerprint is obtained.

4. Optimized fingerprint recognition algorithm

Algorithm to obtain sub-matrices from a single 1 – bit greyscale fingerprint image matrix.
Step 1: Start
Step 2: Get matrix dimension.
Step 3: if matrix_dimension is even
Set i as 3
else
Set i as 2
Step 4: Initialise generation as 1

Step 5: Calculate offset,
offset = (matrix_dimension – i) / 2
Step 6: initialise row_start and column_start as Offset
Step 7: initialise row_end and column_end as Offset + i
Step 8: Initialise sub-matrix as
Matrix[row_start : row_end] [column_start : column_end]
Step 9: increment i by 2
Step 10: incement generation by 1
Step 11: if i < matrix_dimension
Repeat steps 5 to 11
Step 12: Stop

Cuckoo search algorithm in fingerprint recognition

begin
y = fingerprint to be matched
Generate initial population of n images xi (i = 1, 2, ..., n)
while (till n images)
while (CurrentGeneration < MaxGeneration)
xi = submatrix corresponding to current generation
y = submatrix corresponding to current generation
if (xi and y matches)
image is taken to next generation
else
image is not taken to next generation
break while loop
end if
next generation
end while
next image
end while
Postprocess results
end

5. Simulation and results

The 1-bit grayscale image of the finger print is represented as a square matrix with 1's and 0's. The sub-matrix identification algorithm is simulated using c language taking the 1-bit gray scale image as a 2-dimensional array. The simulation displays the element in the sub-matrix as the pair of subscript (ie. Row subscript followed by Column subscript) and the subscript ranges from 0 to n-1 , where n is the dimension of the matrix.

Simulating sub-matrix algorithm
Enter Matrix dimension: 10
Even dimension
Taking the following as 1-bit fingerprint image matrix dimension

[0][0] [0][1] [0][2] [0][3] [0][4] [0][5] [0][6] [0][7] [0][8] [0][9]

[1][0] [1][1] [1][2] [1][3] [1][4] [1][5] [1][6] [1][7] [1][8] [1][9]

[2][0] [2][1] [2][2] [2][3] [2][4] [2][5] [2][6] [2][7] [2][8] [2][9]

[3][0] [3][1] [3][2] [3][3] [3][4] [3][5] [3][6] [3][7] [3][8] [3][9]

[4][0] [4][1] [4][2] [4][3] [4][4] [4][5] [4][6] [4][7] [4][8] [4][9]

[5][0] [5][1] [5][2] [5][3] [5][4] [5][5] [5][6] [5][7] [5][8] [5][9]

[6][0] [6][1] [6][2] [6][3] [6][4] [6][5] [6][6] [6][7] [6][8] [6][9]

[7][0] [7][1] [7][2] [7][3] [7][4] [7][5] [7][6] [7][7] [7][8] [7][9]

[8][0] [8][1] [8][2] [8][3] [8][4] [8][5] [8][6] [8][7] [8][8] [8][9]

[9][0] [9][1] [9][2] [9][3] [9][4] [9][5] [9][6] [9][7] [9][8] [9][9]

For i = 3 Generation:1

[3][3] [3][4] [3][5] [3][6]

[4][3] [4][4] [4][5] [4][6]

[5][3] [5][4] [5][5] [5][6]

[6][3] [6][4] [6][5] [6][6]

For i = 5 Generation:2

[2][2] [2][3] [2][4] [2][5] [2][6] [2][7]

[3][2] [3][3] [3][4] [3][5] [3][6] [3][7]

[4][2] [4][3] [4][4] [4][5] [4][6] [4][7]

[5][2] [5][3] [5][4] [5][5] [5][6] [5][7]

[6][2] [6][3] [6][4] [6][5] [6][6] [6][7]

[7][2] [7][3] [7][4] [7][5] [7][6] [7][7]

For i = 7 Generation:3

[1][1] [1][2] [1][3] [1][4] [1][5] [1][6] [1][7] [1][8]

[2][1] [2][2] [2][3] [2][4] [2][5] [2][6] [2][7] [2][8]

[3][1] [3][2] [3][3] [3][4] [3][5] [3][6] [3][7] [3][8]

[4][1] [4][2] [4][3] [4][4] [4][5] [4][6] [4][7] [4][8]

[5][1] [5][2] [5][3] [5][4] [5][5] [5][6] [5][7] [5][8]

[6][1] [6][2] [6][3] [6][4] [6][5] [6][6] [6][7] [6][8]

[7][1] [7][2] [7][3] [7][4] [7][5] [7][6] [7][7] [7][8]

[8][1] [8][2] [8][3] [8][4] [8][5] [8][6] [8][7] [8][8]

Enter Matrix dimension: 9
Odd dimension
Taking the following as 1-bit fingerprint image matrix dimension

[0][0] [0][1] [0][2] [0][3] [0][4] [0][5] [0][6] [0][7] [0][8]

[1][0] [1][1] [1][2] [1][3] [1][4] [1][5] [1][6] [1][7] [1][8]

[2][0] [2][1] [2][2] [2][3] [2][4] [2][5] [2][6] [2][7] [2][8]

[3][0] [3][1] [3][2] [3][3] [3][4] [3][5] [3][6] [3][7] [3][8]

[4][0] [4][1] [4][2] [4][3] [4][4] [4][5] [4][6] [4][7] [4][8]

[5][0] [5][1] [5][2] [5][3] [5][4] [5][5] [5][6] [5][7] [5][8]

[6][0] [6][1] [6][2] [6][3] [6][4] [6][5] [6][6] [6][7] [6][8]

[7][0] [7][1] [7][2] [7][3] [7][4] [7][5] [7][6] [7][7] [7][8]

[8][0] [8][1] [8][2] [8][3] [8][4] [8][5] [8][6] [8][7] [8][8]

For i = 2 Generation:1

[3][3] [3][4] [3][5]

[4][3] [4][4] [4][5]

[5][3] [5][4] [5][5]

For i = 4 Generation:2

[2][2] [2][3] [2][4] [2][5] [2][6]

[3][2] [3][3] [3][4] [3][5] [3][6]

[4][2] [4][3] [4][4] [4][5] [4][6]

[5][2] [5][3] [5][4] [5][5] [5][6]

[6][2] [6][3] [6][4] [6][5] [6][6]

For i = 8 Generation:4

[0][0] [0][1] [0][2] [0][3] [0][4] [0][5] [0][6] [0][7] [0][8]

[1][0] [1][1] [1][2] [1][3] [1][4] [1][5] [1][6] [1][7] [1][8]

[2][0] [2][1] [2][2] [2][3] [2][4] [2][5] [2][6] [2][7] [2][8]

[3][0] [3][1] [3][2] [3][3] [3][4] [3][5] [3][6] [3][7] [3][8]

[4][0] [4][1] [4][2] [4][3] [4][4] [4][5] [4][6] [4][7] [4][8]

[5][0] [5][1] [5][2] [5][3] [5][4] [5][5] [5][6] [5][7] [5][8]

[6][0] [6][1] [6][2] [6][3] [6][4] [6][5] [6][6] [6][7] [6][8]

[7][0] [7][1] [7][2] [7][3] [7][4] [7][5] [7][6] [7][7] [7][8]

[8][0] [8][1] [8][2] [8][3] [8][4] [8][5] [8][6] [8][7] [8][8]

6. Conclusion and future work

The recognition of behavioral biometrics includes more complex data and that is something which requires optimization than that of physical biometrics. If the behavioral biometric recognition gets optimized it can help in various fields which include medical and security. We would continue to work on using cuckoo search to optimize behavioral biometrics such as walking pattern, writing pattern.

Fingerprint Search Algorithm using cuckoo search is efficient since the prints compared the prints in an incremental way starting from the center. Thus, an optimized algorithm is produced using Cuckoo Search. Since Fingerprints are represented as a 2d matrix the effectiveness of an optimized algorithm is insignificant.

While in case of More Complex Biometric Features like Walking pattern, Writing Style, etc. Many Variables and Environmental factors are involved. Then an Optimized algorithm becomes an absolute necessity. Complex Biometric Algorithms can also be optimized using cuckoo search. By Optimizing using Cuckoo Search, the time taken to find a complete match from the database is reduced drastically and thus increasing the overall efficiency of the system.

References

[1] Yang XS & Deb S, "Cuckoo search via Lévy flights", *World Congress on Nature & Biologically Inspired Computing*, (2009), pp.210-214.

[2] Dorothy R, Joany RM, Joseph Rathish R, Santhana Prabha S & Rajendran S, "Image enhancement by Histogram equalization", *Advances in Recent Trends in Communication and Networks*, (2010).

[3] Erbilek M & Fairhurst M, "A methodological framework for investigating age factors on the performance of biometric systems", *Proceedings of the on Multimedia and security*, (2012), pp.115-122.

[4] Mondal S, Bours P & Idrus SZ, "Complexity measurement of a password for keystroke dynamics: Preliminary study", *Proceedings of the 6th International Conference on Security of Information and Networks*, (2013), pp.301-305.

[5] Mishra A, Bharadi V & Kekre H, "Multimodal biometrics", *Proceedings of the International Conference and Workshop on Emerging Trends in Technology*, (2010), pp.1002-1003.

[6] Hong F, Wei M, You S, Feng Y & Guo Z, "Waving authentication: your smartphone authenticate you on motion gesture", *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems*, (2015), pp. 263-266.

[7] Morales A & Fierrez J, "Keystroke Biometrics for Student Authentication:A Case Study", *Proceedings of the ACM Conference on Innovation and Technology in Computer Science Education*, (2015), pp. 337-337.

[8] Eberz S, Rasmussen KB, Lenders V & Martinovic I, "Evaluating Behavioral Biometrics for Continuous Authentication: Challenges and Metrics", *Proceedings of the ACM on Asia Conference on Computer and Communications Security*, (2017), pp.386-399.