Fuzzy Hashing

Lea Grieder (328216)
Leila Sidjanski (330810)

February 2024

EPFL

Contents

1 Introduction

1.1 Background

1. Intoduce the systems that are used nowadays for authentication and explain why it is preferable to use biometric authentication systems and more precisely finger-vein authentication.

Authentication entails confirming the validity of a claimed identity seeking access to a system or resource. Over decades, authentication mechanisms have progressed from basic password systems in the 1960s to advanced methods such as multifactor authentication by the late 2010s, driven by a persistent commitment to combat evolving security risks while enhancing user convenience[1]. Various authentication methods are employed, including password-based authentication, certificate-based authentication, one-time passwords multifactor authentication, and biometric authentication[2]. Biometric authentication involves extracting, measuring, and statistically analyzing unique physical characteristics of individuals. This method is often deemed more secure than traditional authentication methods due to the difficulty in duplicating biometric traits. Biometric authentication encompasses technologies such as facial recognition, fingerprint recognition, eye recognition, and voice recognition[3]. However, despite its security advantages, biometric authentication isn't impervious to exploitation by malicious actors. For instance, fingerprints left on surfaces could be replciated, or hackers might acquire images of individuals online to trick systems into granting access.
Given that most advanced biometric authentication systems primarily rely on externally visible physical attributes, it appears that finger-vein authentication could address this concern or potentially increase the difficulty of theft or impersonation. This approach introduces a unique layer of security by focusing on internal, anatomical features that are less susceptible to replication or theft compared to external characteristics.

Need to explain multiple things: Introduce the systems that are used nowadays for authentication and explain why it is preferrable to use biometric authentication systems and more precisely finger-vein authentication. Explain how the scanner works on a high level (2 cameras). Explain the state of the current project (where is it left, what has been achieved already, by both Burcu and Simon) and how the authentication system works (Simon has already explained this in his introduction, explain a bit differently) Briefly introduce the concept of fuzzy hashing (security reasons and usability)

The escalation of digital identity verification demands has led to a surge in biometric authentication systems, with finger-vein recognition emerging as a uniquely secure modality due to its internal and non-trace-leaving nature.

This project extends the work on optimizing a finger-vein recognition pipeline that has demonstrated the lowest Equal Error Rate (EER) Equal Error Rate (EER) by incorporating a novel hashing step to process the output. The purpose of integrating hash functions within this context is twofold: to bolster the security of biometric data by converting it into a hash value, hence safeguarding against unauthorized reconstruction of the biometric template, and to enhance the system's efficiency by facilitating rapid comparison of hashed values in place of actual biometric data.

Simon Sommerhalder and Burcu Yildiz have both made significant contributions to our system. Simon has innovated a method to align finger-vein images independently, enhancing security

[1] https://cybersecurity.asee.io/blog/history-of-authentication/
[2] https://www.microsoft.com/en-us/security/business/security-101/what-is-authentication.
[3] Problem with url

by eliminating the need to compare the model and probe images side by side. He organized the process into six clear steps: masking, pre-alignment, histogram equalization, feature extraction, post-alignment, and measuring distance. This organization allows each part of the process to be individually assessed and improved. Additionally, Simon has developed various functions to boost the system's efficiency and achieve the optimal balance between false acceptances and rejections.

1.2 Presentation of the Project

Explain what we will be testing/implementing in this project without getting into too many details yet (we will get more in detail in the theoretical framework where we explain what is in the fuzzy hash sections 1-5 and how we will test what is in there) Explain how we will use the previous work done by Simon and Burcu (?)

1.3 Structure of the Report

Explain how our report will be structured -> Explain how we needed to start by understanding and retesting (as asked from Serge) what had been done before us (the software part as we have already covered what has been done on a high level in the introduction). We also needed to assess again the efficiency of the algorithms developped by Simon and Burcu. Then, explain that we will start by going through the theoretical concepts that we need to understand in order to implement everything (sections 1-5 from fuzzyhash). Then we will move onto the actual software implementation, 1:N matching (etc...) -> on doit update ceci à la fin et expliquer ce qu'on a actually réussi à faire

Additional comments and definitions (may or may not be useful) Fuzzy Hashing: Fuzzy hashing is a technique used to generate a hash value that remains consistent even when the input data has minor variations. This is particularly useful in biometrix, when the data captured(like finger-vein patterns) may have slight differences each time due to changes in the environment or the way the biometric trait is presented.

Purpose of hashing: By storing a hash of the extracted biometric feature rather than the extracted feature itself, the privacy of the user is enhanced. Even if the hash data is compromised, it should not reveal any personal biometric information. Hashed values have fixed sizes which makes storage requirements predictable and efficient.

Fuzzy Extractors: takes the concept of fuzzy hashing further by enabling secure error-tolerant biometric authentication. It consists of two main algorithms, Gen (generate) and Rep (reproduce). It enables the secure extraction and reproduction of a key from noisy input data, like biometric data.

2 Theoretical Framework

Here we explain the theoretical concepts and maths that are explained in our document fuzzy-text

2.1 Biometric Setting

Explain section 1 of the document (parameterization and representation of biometric data). Explain what biometric data is and its importance in security applications.Explain the transformation of finger pictures into compressed formats for vein pixel extraction (Simon's pipeline).

Explain the probability models of section 1: introduce p and the concept of matching biometric captures with an optimal offset, denoted by sigma. to minimize the mismatch probability (p(Xi!=Yi))

2.2 Concept of Fuzzy Hashing

Explain what fuzzy hashing is and how it is different from standard hashing in security systems. Describe the PreHash functionality on biometric data, using random permutation to select specific indices corresponding to feature points. Explain the formula for comparing two pre-hashed biometric captures using the hamming weight and the bitwise AND operation to assess the match probability. Explain the calculation of the matching score, using the ratio of the hamming weight of the conjunction of two bit strings to their combined hamming weights. Explain what mu is.

2.3 Compressed Fuzzy Hashing

Explain how post-hash functions compress the pre-hash bit strings into a more compacted form, maintaining near uniform distribution when inputs are random. Discuss the implications of tis compression on matching probability and introduce the concept of D=2 puissance d to understand the compression ratio and its effects on hash collision probabilities. Detail the formula calculating the probability of hash matches after compression and the role of mu in determining these probabilities, particularly how it adjusts based on the distribution of biometric captures (same, different, independent)

2.4 Pre-Hash and Post-Hash Implementation and Analysis

Explain the algorithm and how we have implemented it

2.5 Applications in Biometric Macthing

Illustrate the use of fuzzy hashing in biometric matching by employing the hamming distance to compare hashes, reducing the data required for storing biometric templates and enhancing privacy. Explain how a threhold is defined for determining a match and how false negative rates (FNR) and false positive rates (FPR) are calculated using the cumulative distirbution function (CDF) of the normal distribution, emphasizing the statistical approach to balancing match accuracy and security.

2.6 1:N Matching

Discuss the challenges and strategies for matching a single biometric sample against a large database, focusing on the computational and memory complexities involved. Explain how parameters are adjusted to manage trade offs between time complexity, space complexity, and match probability. Provide insight into the statistical models used for evaluating the performance of 1:N natching systems, including the derviation of FNR and the optimization of parameters for efficient and accurate matching across large datasets.

3 Experimental Verifications

3.1 Methodology for Assessing Theoretical Values (p, delta, mu)

3.2 Analyzing FPR and FNR (Disussion of Results)

4 Optimization Strategies

4.1 Data Compression Techniques for m=1, d=4

4.2 Efficiency Improvement in Hashing Process

5 [1:N] Matching and System Evaluation

5.1 Implementation of [1:N] Matching

5.2 System Performance Evaluation

6 Conclusion

6.1 Future Directions and Enhancement

7 Definitions

Equal Error Rate (EER) Metric used to evaluate the performance of a system. It represents the point at which the system's false acceptance rate (FAR) equals its false rejection rate (FRR). A lower EER indicates a more accurate and reliable system as it signifies a balanced trade-off between security (minimizing FAR) and usability (minimizing FRR)

False Acceptance Rate (FAR) This is the probability of incorrectly accepting an unauthorized user

False rejection Rate (FRR) This is the the probability of incorrectly rejecting an authorized user

Hash Function A hash function is an algorithm that converts input data of any size to a fixed-size string of characters, which typically acts as a data fingerprint. The output, known as a hash, is unique for different inputs in ideal cases, making hash functions crucial for cryptography, data integrity, and indexing in databases.