Computer Algebra for Fingerprint Matching

Stefano Bistarelli[1,2], Giandomenico Boffi[1], and Fabio Rossi[3*]

[1] Università "G. d'Annunzio", Dipartimento di Scienze,
Viale Pindaro 42, 65127 Pescara, Italy
`bista@sci.unich.it, gboffi@unich.it`
[2] CNR, Istituto di Informatica e Telematica,
Via G. Moruzzi 1, 56124 Pisa, Italy
`Stefano.Bistarelli@iit.cnr.it`
[3] Università di Trieste, Dipartimento di Scienze Matematiche
Via A. Valerio 12/1, 34127 Trieste, Italy
`rossif@univ.trieste.it`

Abstract. We show in this paper how some algebraic methods can be used for fingerprint matching. The described technique is able to compute the score of a match also when the template and test fingerprints have been not correctly acquired. In particular, the match is independent of translations, rotations and scaling transformations of the template. The technique is also able to compute a match score when part of the fingerprint image is incorrect or missed. The algorithm is being implemented in CoCoA, a computer algebra system for doing computations in Commutative Algebra.

Keywords: fingerprint matching, point pattern matching, zero-dimensional ideals, Gröbner basis, non-linear polynomial systems, Sturm sequences, affine transformations.

1 Introduction

Biometrics deals with automated methods of identifying a person or verifying the identity of a person based on physiological or behavioral characteristics [16]. Examples of physiological characteristics include hand or finger images, facial or vocal characteristics. Behavioral characteristics are traits that are learned or acquired. Dynamic signature verification is related to a behavioral characteristic.

Biometric technologies are becoming the foundation of an extensive array of highly secure identification and personal verification solutions. As the level of security breaches and transaction fraud increases, the need for highly secure identification and personal verification technologies is becoming apparent. Biometrics-based solutions are able to provide for confidential financial transactions and personal data privacy.

Biometric authentication requires comparing a registered or enrolled biometric sample (biometric template or identifier) against a newly captured biometric sample (for example, the one captured during a login). This is a three-step process (*Capture*, *Process*, *Enroll*) followed by a *Verification* or *Identification* process.

- *Capture:* A raw biometric sample is captured by a sensing device, such as a fingerprint scanner or video camera.
- *Process:* The distinguishing characteristics are extracted from the raw biometric sample and converted into a processed biometric identifier record (sometimes called biometric sample or biometric template).
- *Enroll:* The biometric template is stored/registered in a storage medium for later comparison during an authentication phase. Notice that the original biometric sample cannot be reconstructed from this identifier.
- *Verification:* In this mode ("1 to 1 matching"), a newly captured/processed biometric sample taken for instance during a login, is compared against a previously enrolled sample to address the question *"Are you the person you claim to be?"*.
- *Identification:* In this mode ("1 to N matching"), the individual does not claim an identity. The individual presents a biometric sample and the system tries to identify the individual from a database of stored biometric samples. This process intends to answer the question *"Who are you?"*[1].

The most common biometric techniques are: signature verification, retinal analysis, facial analysis, fingerprint verification, hand geometry and voice verification. Among all of the above, fingerprint verification is one of the most affordable and easy to use and we focus on it.

In the following sections, we first give some basic information about fingerprints (§2) and about minutiae (§3). Then our approach is described (§4) and some conclusions (§5) are put forward.

2 Fingerprints

Fingerprints are graphical flow-like ridges present on human fingers (see Fig. 1). Their formations depend on the initial conditions of the embryonic mesoderm from which they develop.

Humans have used fingerprints for a very long time [12]. Modern fingerprint techniques were initiated in the late 16th century [9, 14, 10]. In 1684, English plant morphologist Nehemiah Grew published a paper reporting his systematic study on the ridge, furrow, and pore structure in fingerprints, which is believed to be the first scientific paper on fingerprints [12, 5, 10]. Since then, a number of researchers have invested huge amounts of effort studying fingerprints. In 1788,

---

[1] This mode is sometimes associated with law enforcement applications but can also be used for other applications where the user voluntarily presents its biometric sample and expects to be recognized by the system.

Fig. 1. An example of fingerprint.

a detailed description of the anatomical formations of fingerprints was given by Mayer [8], in which a number of fingerprint ridge characteristics were identified. Starting from 1809, Thomas Bewick began to use one of his fingerprints as his trademark, which is believed to be one of the most important contributions in the early scientific study of fingerprint identification [12]. Purkinje proposed the first fingerprint classification scheme in 1823, which classified fingerprints into nine categories according to the ridge configurations [12]. Henry Fauld, in 1880, first scientifically suggested the individuality and uniqueness of fingerprints. At the same time, Herschel asserted that he had practiced fingerprint identification for about 20 years [12]. The discovery of the uniqueness of fingerprints established the foundation of modern fingerprint identification. In the late 19th century, Sir Francis Galton conducted an extensive study of fingerprints [12]. He introduced the minutiae features for single fingerprint classification in 1888.

The biological principles of fingerprints are: the individual epidermal ridges and furrows (valleys) have different characteristics for different fingers, the configuration types only vary within limits which allow for systematic classification, and the configurations and minute details of individual ridges and furrows are *permanent* and *unchanging* for each finger.

3 Minutiae

Typically, automatic fingerprint identification and authentication systems rely on representing the two most prominent structures: ridge endings and ridge bifurcations [18, 10, 11].

Fig. 2 shows examples of ridge endings (b) and ridge bifurcations (a). These two structures are background-foreground duals of each other and pressure variations could convert one type of structure into the other. Therefore, many common representation schemes do not distinguish between ridge endings and bifurcations. Both the structures are treated equivalently and are collectively called minutiae. The simplest of the minutiae-based representations constitute a list of points defined by their coordinates (many of the other ridge structures could be described as a combination of ridge endings and bifurcations [12, 10]). Typically,
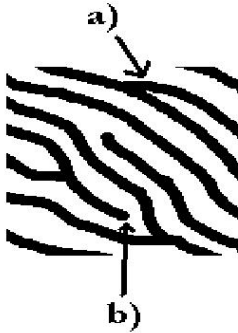
Fig. 2. An example of ridge endings b) and bifurcations a).

though, these minimal minutiae-based representations are further enhanced by tagging each minutia (or each combination of minutiae, e.g., pairs, triplets) with additional features. For instance, each minutia could be associated with the orientation of the ridge at that minutia; or each pair of minutiae could be associated with the ridge count: the number of ridges visited during the linear traversal between the two minutiae. The ANSI-NIST standard representation of a fingerprint is based on minutiae and includes minutiae location and orientation [2]. A minutiae-based representation might also include one or more global attributes like orientation of the finger, fingerprint class, etc.

4 Our Matching Algorithm

Our representation is minutiae-based and each minutia is described by its location (x, y coordinates). In this case, the fingerprint verification problem may be reduced to a point pattern matching (minutiae pattern matching) problem. In the ideal case, if

- the correspondence between the template minutiae pattern and input minutiae pattern is known,
- there are no deformations such as translations, rotations and others, and
- each minutia present in a fingerprint image is exactly localized,

then fingerprint verification is only a trivial task of counting the number of spatially matching pairs between the two images. Determining whether two representations of a finger extracted from two impressions of its, possibly separated by a long duration of time, are indeed representing the same finger, is an extremely difficult problem.

Fig. 3 illustrates the difficulty with an example of two images of the same finger. In particular,

a) The finger may be placed at different locations on the glass platen resulting in a translation of the minutiae from the test representation with respect to those in the reference representation.

Fig. 3. Two different fingerprint impressions of the same finger [10]. In order to know the correspondence between the minutiae of these two fingerprint images, all the minutiae must be precisely localized and the deformations must be recovered.

b) The finger may be placed with different orientations on the glass platen resulting in a rotation of the minutiae from the test representation with respect to those of the reference representation.

c) The finger may exert a different (average) downward normal pressure on the glass platen resulting in a spatial scaling of the minutiae from the test representation with respect to those in the reference representation.

d) Spurious minutiae may be present in both the reference as well as the test representations. (Manual work, accidents etc. inflict injuries to the finger, thereby changing the ridge structure of the finger, either permanently or semi-permanently.)

e) Genuine minutiae may be absent in the reference or test representations. (Skin disease, sweat, dirt, humidity in the air all confound the situation resulting in a non-ideal contact situation.)

A matcher may rely on one or more of the above, resulting in a wide spectrum of behavior. At the one end of the spectrum, we have the "Euclidean" matcher, who allows only rigid transformations among the test and reference representations. At the other extreme, we have the "topological" matcher, who may allow the most general transformations.

A good matcher needs to stay somewhere in the middle. And this is the real difficulty. The technique we have developed takes easy care of traslations, rotations and scaling transformations. With some shrewdness, also the appearing/disappearing minutiae problem is dealt with.

4.1 The basic algorithm

Using minutia coordinates as templates, reduce the problem of fingerprint matching to the problem of point pattern matching. We have to solve a *decisional* problem: "Does there exist a mapping between the template and the test sets of points?".

Notice that, in our application, we need only to know if such a transformation exists; the problem of *finding* the mapping is, in general, much harder, also with our techniques (cf. e.g. [13]).

Let us consider two sets $S_1 := \{P_1, \ldots, P_n\} \subseteq \mathbb{Q}^2$ and $S_2 := \{Q_1, \ldots, Q_m\} \subseteq \mathbb{Q}^2$, of n and m points, representing two different fingerprint impressions. Let us also assume $m \leq n$ (we can always exchange S_1 with S_2 in order to have $m \leq n$).

The following procedure decides whether there exists an affine change of coordinates of $\mathbb{R}^2$ that maps the points of S_2 to some subset of S_1.

The procedure is made of five steps:

Step 1: Using the algorithm in [1], we compute a (finite) set $\{f_1 \ldots f_k\} \subseteq \mathbb{Q}[x, y]$ of generators of the defining ideal $I \subseteq \mathbb{C}[x, y]$ of S_1.

Step 2: If

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} x' \\ y' \end{pmatrix} + \begin{pmatrix} e \\ f \end{pmatrix}, ad - bc \neq 0$$

is the generic affine change of coordinates of $\mathbb{R}^2$, we consider the polynomials

$$F_{ij}(a, \ldots, f) := f_i(ax'_j + by'_j + e, cx'_j + dy'_j + f) \in \mathbb{Q}[a, b, c, d, e, f, g],$$

$$1 \leq i \leq k, 1 \leq j \leq m,$$

where (x'_j, y'_j) are the coordinates of the points of S_2.
We must study the real compatibility of the following system of non-linear polynomials

$$\begin{cases} F_{ij}(a, \ldots, f) = 0 \\ ad - bc - g = 0 \end{cases} \tag{1}$$

under the unique constraint $g \neq 0$.

Step 3: Let

$$J := (\{F_{ij}(a, \ldots, f), ad - bc - g\}) \subseteq \mathbb{C}[a, b, c, d, e, f, g]$$

be the ideal of the system. First of all, we decide whether J is a proper ideal, by computing the reduced Gröbner basis G of J with respect to a chosen term order $<$ (e.g. grevlex; cf. [6, Chapter 2, §7]). If $G = \{1\}$, then no change of coordinates can exist; otherwise, J is a zero-dimensional ideal, because every solution of the algebraic system (1) gives, in particular, a map from S_2 to S_1.

Using the same G, and using the algorithms in [7, Chapter 2, §2], we compute
- $L := Rad(J) \subseteq \mathbb{C}[a, b, c, d, e, f, g]$ (Rad = radical),
- the unique monic generator $m_g(g)$ of $L \cap \mathbb{C}[g]$.

Moreover, by means of Hilbert function computation of $\mathbb{C}[a, \ldots, g]/In_<(L)$ ($In_<$ = initial ideal with respect to a chosen term order $<$), we find
- $D := dim_{\mathbb{C}}(\mathbb{C}[a, b, c, d, e, f, g]/L)$.

It is well known that $deg(m_g(g)) \leq D$.(Cf. e.g. [7, Chapter 2], [4, Chapter 8, §8.6].)

Step 4: If $deg(m_g(g)) = D$, then, using the well known technique of Sturm sequences (cf. e.g. [7, Chapter 2], [4, Chapter 8, §8.8]), we compute the number r of real roots of $m_g(g)$. Observe that $m_g(g) \in \mathbb{Q}[g]$, because all computations can be performed in the ground field $\mathbb{Q}$, and that $m_g(g)$ has no multiple roots in $\mathbb{C}$ because L is a radical ideal.

It is easy to see that r is also the number of the real solutions of the system (1), because L is in normal position with respect to g and then the reduced Gröbner basis of L with respect to any elimination term order of $\{a, b, c, d, e, f\}$, takes the form $\{m_g(g), a - n_a(g), \ldots, f - n_f(g)\}$, where $\{n_a(g), \ldots, n_f(g)\} \subseteq \mathbb{Q}[g]$ (cf. e.g. [4, Chapter 8, Proposition 8.77]).

Then:

- If $r \geq 2$, there exists at least an affine change of coordinates.

- If $r = 1$, there exists an affine change of coordinates, if $m(g)$ has a constant coefficient.

- If $r = 0$, no affine change of coordinates exists.

Step 5: If $deg(m_g(g)) < D$, then the arguments in [4, Chapter 8, §8.6] show that we can always assume that there exists at least a variable $v \in \{a, b, c, d, e, f\}$ such that $deg(m_v(v)) = D$, where $m_v(v)$ is the unique monic generator of $L \cap \mathbb{C}[v]$.

As in Step 4, we compute the number of real solutions of (1), counting the real roots of $m_v(v)$.

Consider now the system

$$\begin{cases} F_{ij}(a, \ldots, f) = 0 \\ ad - bc - g = 0 \\ g = 0 \end{cases} \qquad (2)$$

and, using the previous procedure, compute the number r'' of real solutions of (2).

- If $r'' < r'$, then there exists an affine change of coordinates.

- If $r'' = r'$, no affine change of coordinates exists.

4.2 A refinement

In case the algorithm in 4.1 gives negative answer, replace S_2 by all subsets $S_2 - \{Q\}$ where $Q \in S_2$, until the first success. If the answers are negative for all $Q \in S_2$, try $S_2 - \{Q, Q'\}$ where $Q, Q' \in S_2 (Q \neq Q')$, until the first success ($\binom{m}{2}$ possible cases), and so on.

We can define a notion of "deficiency" by: $Def(S_1, S_2) := Card(S_1) - Card(S_2')$, where S_2' is the first subset of S_2 for which the algorithm has a positive answer.

The lower $Def(S_1, S_2)$ turns out to be, the more likely it is that S_1 and S_2 correspond to the same fingerprint.

An acceptable level of nonzero deficiency shall be determined empirically, in view of the applications one has in mind.

4.3 Implementation

The algorithm has been implemented in CoCoA [20], a computer algebra system for doing computations in Commutative Algebra. In our opinion, other computer algebra systems do not make all the necessary ingredients immediately available to us as CoCoA does. The functions of CoCoA which have proved particulary useful are:

- **ReducedGBasis**; SYNTAX ReducedGBasis(M:IDEAL, MODULE, or TAGGED("Quotient")):LIST; this function computes a reduced Gröbner basis of M.

- **IdealOfPoints**; SYNTAX IdealOfPoints(Points:LIST):IDEAL; this function computes the reduced Gröbner basis for the ideal of all polynomials which vanish at the given set of points.

- **Radical**; SYNTAX Radical(I:IDEAL):IDEAL; this function computes the radical of I.

- **Elim**; SYNTAX Elim(X:INDETS,M:IDEAL):IDEAL; this function returns the ideal obtained by eliminating the indeterminates X from M.

- **Hilbert**; SYNTAX Hilbert(R:RING or TAGGED("Quotient")):TAGGED; this function computes the Hilbert function for R.

- **Der**; SYNTAX Der(F,X:INDET):POLY; this function returns the derivative of F with respect to the indeterminate X.

- **DivAlg**; SYNTAX DivAlg(X:POLY,L:LIST of POLY):RECORD; this function performs the division algorithm on X with respect to L. It returns a record with two fields: 'Quotients' holding a list of polynomials, and 'Remainder' holding the remainder of X upon division by L.

5 Conclusions

We develop a new algorithm for fingerprint matching able to take care of traslations, rotations and other affine transformations. The algorithm is also able to match fingerprints when some minutiae are missed or when some unreal minutia is detected.

A number of point pattern matching algorithms related to our problem have been proposed in the literature [3, 17, 19, 21]. However, the use of an algebraic technique is, to the best of our knowledge, totally new.

An accurate analysis of the efficiency of the algorithm is planned for the near future.

We remark that, in order to count the number of real solutions of the system (1) in 4.1, one can also resort to the methods of [15]. Our approach looks simpler, but it is still unclear which of the two ways is more convenient.

References

[1] Abbott, J., Bigatti, A., Kreuzer, M., Robbiano, L.: Computing ideals of points. J. Symbolic Comput. **30** (2000) 341–356

[2] American National Standard for Information Systems: Data format for the interchange of fingerprint information. Doc# ANSI/NIST-CSL 1-1993 (1993)

[3] Ansari, N., Chen, M.H., Hou, E.S.H.: 13 - A Genetic Algorithm for Point Pattern Matching. In: Dynamic, Genetic, and Chaotic Programming. John Wiley & Sons (1992)

[4] Becker, T., Weispfenning, V.: Gröbner Bases. Springer-Verlag, New York (1993)

[5] Castle, T.: (Online authentication using combined smart card and fingerprint recognition)

[6] Cox, D., Little, J., O'Shea, D.: Ideals, Varieties, and Algorithms. 2nd edn. Springer-Verlag, New York (1996)

[7] Cox, D., Little, J., O'Shea, D.: Using Algebraic Geometry. Springer-Verlag, New York (1998)

[8] Cummins, H., Midlo, C.: Finger Prints, Palms and Soles. Dover Publications Inc., New York (1961)

[9] Federal Bureau of Investigation: The science of fingerprints: Classification and uses (1984)

[10] Jain, K., Hong, L., Pankanti, S., Bolle, R.: An identity-authentication system using fingerprints. In: Proc. of IEEE. Volume 85. (1997) 1365–1388

[11] Jain, A.K., Prabhakar, S., Hong, L., Pankanti, S.: Filterbank-based fingerprint matching. IEEE Transactions on Systems, Man, and Cybernetics **9** (2000) 846–859

[12] Lee, H.C., Gaensslen, R.E.: Advances in Fingerprint Technology. Elsevier, New York (1991)

[13] Manocha, D.: Numerical methods for solving polynomial equations. In: Proceedings of Symposia in Applied Mathematics. Volume 53. (1998) 41–66

[14] Newham, E.: The biometric report (1995)

[15] Pedersen, P., Roy, M.F., Szpirglas, A.: Counting real zeros in the multivariate case. In: Computational Algebraic Geometry. Volume 53., Birkäuser, Boston (1993) 203–224

[16] Podio, F.L.: Biometrics technologies for highly secure personal authentication. (http://www.itl.nist.gov/lab/bulletns/bltnmay01.htm)

[17] Ranade, A., Rosenfeld, A.: Point pattern matching by relaxation. Pattern Recognition **12** (1993) 269–275

[18] Ratha, N., Karu, K., Chen, S., Jain, A.K.: A real-time matching system for large fingerprint database. IEEE Trans. on PAMI **18** (1996) 799–813

[19] Starink, J.P.P., Backer, E.: Finding point correspondence using simulated annealing. Pattern Recognition **28** (1995) 231–240

[20] The CoCoA Team: CoCoA system: Computations in commutative algebra. (http://cocoa.dima.unige.it)

[21] Ton, J., Jain, A.K.: Registering landsat images by point matching. IEEE Transactions on Geoscience and Remote Sensing **27** (1989) 642–651