Hierarchical Kernel Fitting for Fingerprint Classification and Alignment

Abstract

Fingerprint classification consists of labeling a fingerprint impression as one of several major types of fingerprints: arch, left loop, right loop, whorl, etc. The problem of fingerprint matching amounts to deciding whether or not two impressions were produced by the same finger. We propose a model based method for fingerprint classification which only uses the flow field, avoiding the non-trivial computation of the thinned ridges and minutia points. For each class, a fingerprint kernel is defined, which models the shape of fingerprints in that class. The classification is then achieved by finding the kernel that best fits to the flow field of the given fingerprint. We obtain a classification accuracy of 91.25% on the NIST 4 database. We also show how the kernel fitting procedure can be used for an initial fingerprint alignment.

1 Introduction

Fingerprints are biometric characteristics of humans, that consist of ridges and furrows at the tips of the fingers. There is evidence [5] that human awareness of these patterns predates Christianity. Since the last century fingerprints have been used systematically for authentication, particularly in criminal investigation. Based on empirical evidence [5, 6] it is widely accepted that fingerprints uniquely identify an individual. Unlike other authentication methods (such as face or voice recognition, which may not distinguish identical twins), fingerprint authentication is unequivocal. Moreover, fingerprints do not change over time, which gives them an important advantage over other biometric measurements. Lastly, in comparison to other biometrics techniques (such as retinal scan and iris scan), fingerprint acquisition is relatively simple and inexpensive. All these characteristics make the use of fingerprints one of the most pervasive methods of authentication, both in the forensic sector as well as in the commercial sector (e.g. access control).

Given the astronomical size[1] of fingerprint databases, it became apparent that an indexing methodology which would assign fingerprints into a small number of categories was in order. The system developed at the end of the 19-th century by Sir Edward Henry is the basis of modern automatic fingerprint identification systems (AFIS) in the majority of English-speaking countries, and serves the purpose of what is usually termed as (primary) classification of fingerprints [2, 8, 11, 10]. The system adopted by the FBI contains eight basic fingerprint patterns [5, 6]: arch, tented arch, left loop, right loop, whorl, central pocket loop, double loop and accidental. Throughout this paper, we shall confine ourselves to only four major classes of fingerprints [2]: arch (A), left loop (L), right loop (R) and whorl (W). See Figure 1.

There have been a number of methods proposed for fingerprint classification, including (but not limited to) syntactic methods [14], singularity based methods [11, 13], and neural network methods [1]. For a more detailed literature review see for instance [2, 8, 10] and the references therein.

In this paper we address the problem of automatic fingerprint classification into one of the 4 types (A,L,R,W) using a novel method which we term *kernel fitting*. Detailed classification results are shown in Section 3. In addition, we demonstrate how our method can be used for fingerprint alignment, as a pre-processing step for fingerprint matching.

2 Mathematical model

2.1 Flow field

We model the ridges of a fingerprint as curves in the plane, and define the flow field as the direction of the tangent to these curves at each point [3]. There have been several methods proposed for the extraction of the flow field [3, 12, 13]. We employ the method developed by Hong in [9], which averages a function of the gradient of the intensity over a 16×16 window. The window moves in increments of 12 pixels in each

[1] There were 810,188 records (containing 10 fingerprints each) upon the formation of the Identification Division of the FBI in 1924 [5]. Currently there are 226 million cards on file, representing about 79 million individuals [4].

[2] However, the method we propose can be readily extended to more classes of fingerprints.

[3] Except at a small number of points, where the curve is not smooth (e.g. bifurcations).

(a) Arch (b) Left loop

(c) Right loop (d) Whorl

Figure 1: Four major classes of fingerprints.

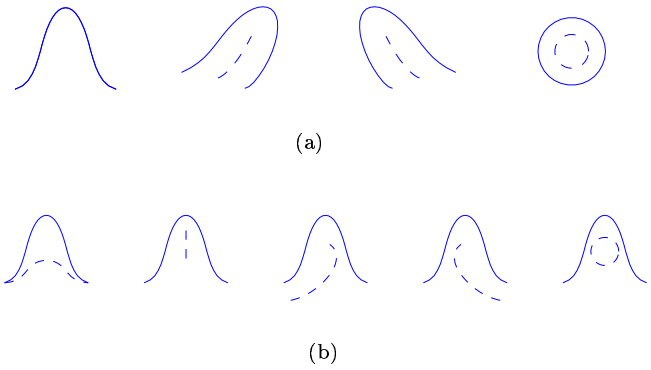
direction, so for fingerprint images of size 512×512 the flow field is a relatively small 43×43 array. See Figure 5. Thus, conceptually, we think of the flow field as a unit vector field in the plane, or, equivalently, as its argument $\beta \in [0, \pi)$ [4] (the angle between the direction of the vector and the x-axis).

2.2 Fingerprint kernels

We define class-specific *kernel curves* as follows. The kernel of the whorl is the unit circle. The kernels of the other classes are defined empirically using splines (see Figure 2). Given points $p_1, \ldots, p_{n+1}$ and (equally spaced) scalars $t_1 = 0, t_2 \ldots, t_{n+1} = 1$, we can find a curve $\sigma : [0, 1] \to \mathbf{R}^2$ such that $\sigma(t_k) = p_k$, and such that both coordinate functions $(x(t), y(t))$ are polynomials (necessarily of degree n). This can be done by solving two Vandermonde systems, where the unknowns are the coefficients of the polynomials $x(t)$ and $y(t)$.

2.3 Kernel hierarchy

Above any of the kernel curves, there is always an arch. This makes the arch category inherently ambiguous. Put differently, if an arch is found in an unknown fingerprint, one cannot simply label that fingerprint as "arch"; we must "descend" inside the arch. Consequently, to dis-ambiguate the classification, we use splines again to define sub-kernels at a fixed position inside the top level kernels. We defined 5 sub-kernels for the arch alone, and one sub-kernel for each of the other kernels. All the sub-kernels bear the same label as the parent kernel. See Figure 2



(a)

(b)

Figure 2: Kernel hierarchy. (a) The continuous curves are the high level kernels of A,L,R,W, respectively; the dotted curves are the low-level kernels of L,R,W. (b): the dotted curves are the low-level curves relative to A.

2.4 Kernel fitting

Suppose we have a smooth vector field V defined over some region in the plane $\mathbf{R}^2$. Let $\gamma(t) = (x(t), y(t))$ be a parametric curve in $\mathbf{R}^2$, let $\dot{\gamma}$ be the tangent to γ, and α its argument. As before, let β be the argument of V. See Figure 3.

Define an energy functional which captures the difference between the direction of $\dot{\gamma}$ and that of the vector field V at the point $\gamma(t)$ by

$$E(\gamma) = \frac{\int_\gamma \sin^2(\alpha - \beta(\gamma)) d\gamma}{\int_\gamma d\gamma} \qquad (1)$$

The denominator in (1) is a normalization factor and rules out the trivial solution (when γ degenerates to a point).

From a mathematical standpoint, finding the extrema of a functional over a space of functions is in general highly non trivial, because one has to address the issues of existence and regularity of the solutions. Indeed, a functional over, say, C^0 (the space of continuous functions) may not have an extremum in C^0, for the simple reason that C^0 is not closed under any topology. It is beyond the scope of this paper to address the existence and regularity of extrema of the above functional.

[4] There is no canonical orientation of the ridges, so $\beta \in [0, \pi)$ rather than $[0, 2\pi)$.

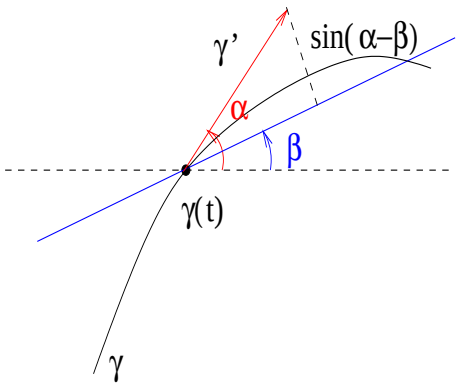


Figure 3: Red vector: unit tangent vector to γ. Blue line: direction of the flow field at the point $\gamma(t)$.

For fingerprint classification, the problem of minimizing the energy (Eq. (1)) can be cast as the problem of finding minima over a (5-dimensional) space of transformations of curves in the plane, rather than over the space of all curves. Loosely speaking, given a fingerprint, we do not fit an arbitrary curve to its flow field, but rather a curve with a predefined shape that models the core pattern of the fingerprint.

Specifically, we consider only the transformations:

$$T(x) = R_\phi \cdot D \cdot x + \tau \qquad \forall x \in \mathbf{R}^2 \qquad (2)$$

In this equation R_ϕ is a rotation of angle ϕ about the origin. $D = \begin{pmatrix} a & 0 \\ 0 & b \end{pmatrix}$ is a dilation in the horizontal and vertical direction and τ is a translation vector.

2.5 Algorithm

Let A, L, R, W be the (high level) kernels of the "arch", "left loop", "right loop" and "whorl" respectively. Let also T be a training set. In our experiments T had 1000 fingerprints.

Training: For a fixed fingerprint class, split T into T_+ and T_-, the positive and negative samples in that class. Plot the distributions of the energy (Eq. (1)) over T_+ and T_- and find a threshold which indicates when the fit is sufficiently good. Obtain separation thresholds for all L,R and W.

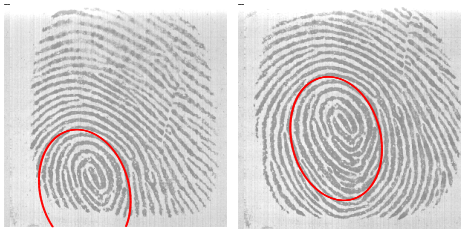
Classification: Fit each of the kernels of L, R, and W (but not A) to a test sample, with scores E_l, E_r and E_w, respectively.

- **Case 1.** Precisely one of E_l, E_r or E_w is below its own threshold. Label the sample as the class of best fit.
- **Case 2.** More than one of E_l, E_r and E_w is below its own threshold. Compare the sub-kernels of each of the candidate curves, and label the sample as the sub-kernel of best fit.
- **Case 3.** None of E_l, E_r and E_w is below its own threshold. Then fit the arch A and check the sub-kernels of A. The label is then given by the smallest score inside the arch.

3 Experimental results

3.1 Kernel fitting

Figure 4 shows two impressions of the same finger and the whorl kernel fit to them. The fit is consistent, in the sense that an ellipse of the same size is fit to both impressions, and it is positioned at about the same region in the fingerprints. Note also, that a very good fit is found in (a), even though the kernel curve is not entirely contained in the image. This is the effect of the denominator in the definition of the energy function (Eq. (1)).



(a) (b)

Figure 4: Elliptic kernel fit to two impressions of the same finger.

3.2 Fingerprint classification

We used the NIST 4 database [7] to test the proposed method. The database consists of 4000 fingerprint impressions, of size 512 × 512 pixels. Each impression was labeled as one of the "arch", "tented arch", "left loop", "right loop" and "whorl" by a human expert. Some of the images were ambiguous and had more than one label. We considered a fingerprint to be correctly classified if the label assigned by our method was among the labels assigned by the human expert. Furthermore, since in the *natural* distribution of fingerprints the arch and the tented arch combined make up for approximately 5% of all fingerprints [4, 8], we classified both "arch" and "tented arch" simply as "arch". However, we did not alter the distribution of the classes in the NIST database, which is uniform. It takes under 0.3 seconds to classify a single

image on a Pentium III 800 MHz processor, which yields a total time of under 20 minutes for the entire database of 4000 images. Under these conditions, with no rejection option, we obtained a classification rate of 91.25%. Table 1 presents our result in comparison with other classification methods. In the "4 classes" column the arch and tented arch were placed in the same category, whereas in the "5 classes" column, the arch and tented arch were considered distinct.

Algorithm	4 classes	5 classes
Kernel Fitting	91.25%	-
MASK [2]	-	87.1%
KARU & JAIN [11]	91.1 %	85.4%
JAIN et. al.(*) [10]	94.8%	90%

Table 1: Comparison between various fingerprint classification methods. (*) On 2000 fingerprints, with a rejection rate of 1.75%.

Table 2 presents the confusion matrix. Recall that in the NIST database some of the fingerprints had two true labels. In building the confusion matrix we counted these images once for each of the true labels. The labels in the leftmost column represent the true labels.

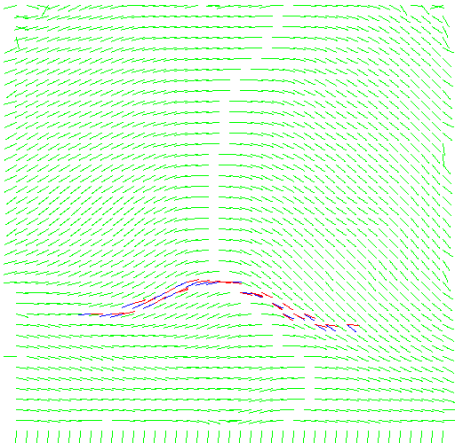
	L	R	W	A
L	730	2	18	54
R	5	754	7	66
W	32	35	718	11
A	74	39	7	1448

Table 2: The confusion matrix.

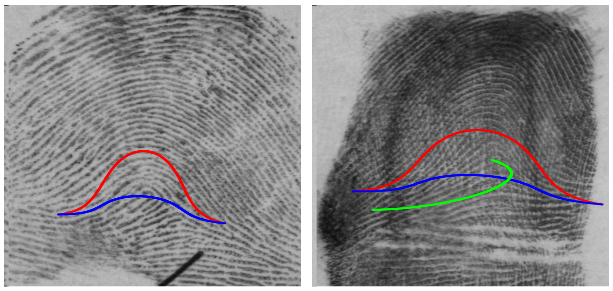
We have identified two main reasons for the classification errors that our approach makes. The first type of error is due to the flow field. In some cases, the fit is good, yet the test sample is misclassified due to the fact that the flow field does not provide a faithful representation of the fingerprint (Figures 5 (a) and (b)). This, in turn, is due to the loss of information incurred when the fingerprint is represented as a flow field. The second type of error is due to the curve fitting itself: a wrong kernel may have a slightly better score than the true kernel (Figure 5 (c)).

3.3 Fingerprint alignment

Once the kernel has been fit to the fingerprints, it is then a trivial matter to find a transformation that overlaps one impression onto the other. A priori, fingerprints belonging to different classes cannot be aligned. Assume then that the two fingerprints are in the same class. Let $\gamma_j = T_j(\gamma_0)$, $j = 1, 2$ be



(a)



(b) (c)

Figure 5: Two types of misclassifications. (a) Best (sub) kernel fit to the flow field. (b) Overlay of the kernel (red) and best fitting sub-kernel (blue) onto the fingerprint impression associated with (a). The true fingerprint class is L and was classified as A. (c) Left loop fingerprint. Blue (arch) and green (left loop): best two sub-kernels fit to the flow field, overlaid on the fingerprint impression. The blue curve has a slightly better score than the green curve and the fingerprint is misclassified as A.

the curves fit to the two fingerprints, both obtained from the same kernel γ_0. Then the transformation that aligns the two fingerprints is $T = T_1 \circ T_2^{-1}$. Figure 6 shows the impressions from Figure 4 (a) and 4 (b) aligned by overlapping the kernels of best fit.

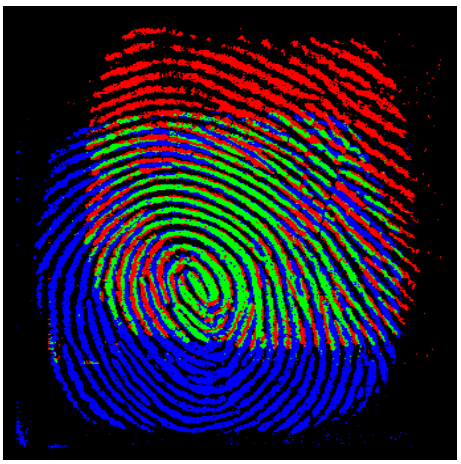


Figure 6: Alignment of impressions in Figure 4 (a) and (b). The green pixels represent the exact overlap.

4 Summary and conclusion

We have presented a novel method for fingerprint classification into one of four general classes: arch, left loop, right loop and whorl, by fitting a kernel of a predefined shape and size to a vector field. Except for the whorl, which is modeled by a circle, the other kernel curves are constructed empirically, by using polynomial splines. The performance of our method is influenced by the amount of the kernel curve that is *a priori* visible in the fingerprint impression. Classification is inherently ambiguous if only a very small part of the kernel is present. However, on "reasonable" impressions, the method proves to be reliable and fast, as it avoids thinning the ridges and extracting the minutia points.

Our long term goal is to devise an algorithm for fingerprint matching that avoids as much as possible the use of minutia points as landmarks, by using the energy defined in this paper (Eq. (1)) as a measure of similarity between two impressions.

References

[1] G.T. Candela, P.J. Grother, C.I. Watson, R.A. Wilkinson, and C.L Wilson. Pcasys - a pattern-level clasification automation system for fingerprints. Technical Report NISTIR 5647, NIST, 1995.

[2] R. Capelli, A Lumini, D Maio, and D. Maltoni. Fingerprint classification by directional image partitioning. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 21(5):402–421, 1999.

[3] T. Chang. Texture analysis of digitized fingerprints for singularity detection. In *Proc. 5th IPCR*, pages 478–480, 1980.

[4] FBI. http://www.fbi.gov.

[5] FBI. *Advanced Latent Fingerprint School.* U.S. Department of Justice, Federal Bureau of Investigation, 1983.

[6] FBI. *The Science of Fingerprints.* U.S. Department of Justice, Federal Bureau of Investigation, 1985.

[7] U.S. National Institute for Standards and Technology. Nist 4. http://www.itl.nist.gov.

[8] L. Hong. *Automatic Personal Identification Using Fingerprints.* PhD thesis, Michigan State University, 1998.

[9] A.K. Jain, L. Hong, and R. Bolle. On-line fingerprint verificatoin. *IEEE Trans. of Pattern Recognition and Machine Intelligence*, 19(4):302–314, 1997.

[10] Anil K. Jain, S. Prabhakar, and Lin Hong. A multichannel approach to fingerprint classification. *IEEE Trans. of Pattern Recognition and Machine Intelligence*, 21(4):348–359, 1999.

[11] K. Karu and A.K. Jain. Fingerprint classification. *Pattern Recognition*, 29(3):389–404, 1996.

[12] M. Kass and A. Witkin. Analyzing oriented patterns. *Comput. Vision Graphics Image Process.*, 37(4):362–385, 1987.

[13] M. Kawagoe and A. Tojo. Fingerprint pattern classification. *Pattern Recognition*, 17(3):295–303, 1984.

[14] K Rao and K Balk. Type classification of fingerprints: A syntactic approach. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 2(3):223–231, 1980.