

МЕТОД УСКОРЕННОЙ ИДЕНТИФИКАЦИИ ОТПЕЧАТКОВ ПАЛЬЦЕВ

© 2021 В.Ю. Гудков

Южно-Уральский государственный университет

(454080 Челябинск, пр. им. В.И. Ленина, д. 76)

E-mail: diana@sonda.ru

Поступила в редакцию: 11.08.2020

В статье излагается метод ускоренной идентификации изображений отпечатков пальцев на основе шаблонов, которые формируются в результате автоматической обработки изображений. Метод опирается на свойства ближайших окрестностей контрольных точек в виде окончаний и разветвлений линий узоров пальцев и состоит из двух этапов. На первом этапе каждая контрольная точка запросного шаблона сравнивается с каждой контрольной точкой ссылочного шаблона из базы данных и оценивается степень схожести таких пар контрольных точек. Для ускорения вычислительных операций вводятся классы, которые позволяют быстро аккумулировать степень схожести контрольных точек из этих двух шаблонов в гистограмме. Оценивается качество такой гистограммы. Гистограммы строятся для всех ссылочных шаблонов из базы данных и одного запросного шаблона. На втором этапе на основе оценок гистограмм отбираются наиболее похожие шаблоны, число которых значительно меньше объема базы данных. Эти шаблоны сравниваются дополнительно с учетом консолидации контрольных точек и оценивается компактность расположения соответствующих пар контрольных точек из двух сравниваемых шаблонов. Значительное ускорение алгоритма идентификации достигается за счет отбрасывания непохожих пар контрольных точек на первом этапе и пар шаблонов с плохими оценками гистограмм на втором этапе. Приводятся результаты экспериментов, опубликованные в интернете.

Ключевые слова: отпечаток пальца, идентификация, контрольная точка, гистограмма.

ОБРАЗЕЦ ЦИТИРОВАНИЯ

Гудков В.Ю. Метод ускоренной идентификации отпечатков пальцев // Вестник ЮУрГУ. Серия: Вычислительная математика и информатика. 2021. Т. 10, № 1. С. 32–48. DOI: 10.14529/cmse210103.

Введение

Для биометрической идентификации личности человека успешно применяются отпечатки пальцев [1]. Задача идентификации является основной для биометрических систем, таких как автоматизированные дактилоскопические системы, системы контроля доступа, паспортно-визовые системы и др. Работа этих систем опирается на специфические физиологические характеристики человека [2], которые сохраняют в базе данных (БД). В ее состав входят изображения отпечатков пальцев и их шаблоны — модели изображений [3]. Для ускорения вычислительных операций идентифицируют шаблоны, а не оригинальные изображения [3]. Применение шаблонов позволяет обеспечить приемлемый уровень качества распознавания [4, 5], однако производительность процедуры идентификации при этом остается ограниченной [6]. Это явление особенно становится заметным с ростом объема БД. Компенсировать его можно за счет повышения стоимости материальной части биометрической системы. Однако такой подход не всегда приемлем [5, 7]. Как существенно сократить время задачи идентификации без увеличения стоимости системы, обеспечивая при этом приемлемый уровень снижения качества распознавания? Эта задача чрезвычайно актуальна и может быть решена с помощью новых математических моделей изображений, новых парадигм, методов и алгоритмов распознавания изображений [8, 9].

Целью исследования является разработка метода и алгоритма ускоренной идентификации изображений отпечатков пальцев по их шаблонам.

Статья состоит из трех разделов и заключения. В разделе 1 коротко освещаются предшествующие работы, указываются недостатки опубликованных методов и нерешенные проблемы. В разделе 2 излагается основная идея по улучшению качества и повышению производительности задачи идентификации, предлагаются классификаторы на основе топологических и геометрических свойств изображений, описывается способ сравнения контрольных точек (КТ) [3] для запросного и ссылочного шаблонов, а также способ построения гистограммы и оценки ее качества. Здесь же освещается новый алгоритм сравнения шаблонов отпечатков пальцев для больших БД и раскрываются особенности ускоренной идентификации. В разделе 3 представлены результаты экспериментов, опубликованные в интернете. В заключении приводится резюме, выводы и направление дальнейших исследований.

1. Обзор предшествующих работ

Для улучшения качества и повышения производительности задачи идентификации предложено много методов [1, 2, 4–9]. Корректный выбор пары КТ (одна из запросного, а другая из ссылочного шаблонов), образующих пару предполагаемых похожих признаков, в основном определяет качество и производительность идентификации [5, 6]. Однако множество КТ, а также отношений между ними, может быть сохранено в шаблоне различными способами. Наиболее часто для сравнения нескольких пар КТ применяется гребневый счет, разность расстояний между ними и разность их векторов направлений [7–10]. Под гребневым счетом понимают число линий на изображении, которые пересекаются отрезком, проведенным от одной КТ до другой [3, 11].

Локальная топологическая структура для КТ (LMTS — local minutiae topological structure) предложена в работе [12]. Структура устойчива к появлению шумов на изображении и деформации узора, потому что отражает его топологические свойства. Применение LMTS снижает ошибки идентификации. Несколько позже был предложен метод, объединяющий структуры LMTS в пары [13], что позволило авторам улучшить предшествующие результаты исследований за счет увеличения числа сравниваемых геометрических и топологических характеристик изображений. Следующим этапом развития этого подхода стало объединение LMTS в созвездия по 3 и более элементов [14]. Интересно, что аналогичный LMTS, но несколько упрощенный подход для оценки подобия локальных структур был предложен ранее в [15], однако он не обеспечил сравнимые с созвездием LMTS результаты. Авторам указанных работ удалось повысить производительность и улучшить результаты идентификации отпечатков пальцев.

В работе [16] специальные структуры, называемые k -plet, формируются для каждой КТ изображения. На основе этих структур предложен метод двойного поиска в ширину (CBFS — Coupled Breadth First Search). Многошаговый алгоритм CBFS обрабатывает изображение, фильтрует, строит скелет линий [1, 2, 7] и, прослеживая скелет линий, детектирует структуру k -plet для каждой КТ. По сути эти структуры являются небольшими подграфами, описывающими скелет узора в некоторой области КТ. Подграфы объединяются в граф. При идентификации изображений такие графы сравниваются и вычисляется степень их подобия.

В работе [17] метод CBFS был развит с помощью преобразования графа в ориентированный граф. Каждая КТ снабжается ориентированным графом смежности (MAG — Minutiae Adjacency Graph) на основе выбора другой ближайшей КТ и заданных ограничений. Тогда исследуемая КТ как вершина графа представляется в виде $\{i, j, d_{ij}, r_{ij}, \varphi_{ij}\}$,

где i, j — номера КТ, а значения d_{ij} , r_{ij} и φ_{ij} — расстояние, гребневый счет и разность векторов направлений между этими КТ.

В работе [18] идея графа MAG приобретает интересную модификацию. В ней предлагаются графы для петель, дельт и завитков узора так же, как и для КТ, а граф рассматривается как иерархически организованное дерево. При сравнении шаблонов все графы используются консолидировано с учетом признаков различного уровня [3].

В работе [19] предлагается использовать хэш-коды для ускорения идентификации в больших БД. Для каждой КТ строится вектор цилиндрического кода (МСС — Minutia Cylinder Code), инвариантный к положению и ориентации КТ. Новизна заключается в оригинальном численном расчете степени похожести векторов МСС на основе классификации и взвешенного сглаживания элементов векторов. Показана их хорошая эффективность. Развитие идеи МСС предложено в работе [20] на основе компактного представления признаков в векторе сферического кода и модификации метрического пространства (MSCC — Minutia Spherical Coordinate Code). В ней авторы вводят представление признаков на сбалансированной решетке, что позволяет лучше отбирать идентичные КТ.

В целом, эти методы и алгоритмы демонстрируют развитие задачи идентификации. Но они ориентированы на относительно хорошие изображения. Корректный отбор идентичных КТ нарушается на изображениях с дефектами. Качество изображения можно оценить по критерию NIST (National Institute of Standards and Technology) [21]. Идентификация состоит из ряда подзадач классификации, которые выполняются при решении самой задачи идентификации, а при обработке изображений — лишь частично. Ясно, что предварительная классификация разнообразных структур и подграфов, а также их упорядочение, снижает объем вычислений при выполнении основной задачи.

Подведем итог. При идентификации отбор наиболее похожих пар КТ необходим, но недостаточен. Проблемы это не решает. Решение видится в том, чтобы дополнительно накопить оценки похожести отобранных пар КТ заранее и аккумулировать их. Такие оценки позволяют отфильтровать непохожие шаблоны при минимуме вычислений и выполнить более детальное их сравнение для небольшой доли БД. Такое решение не просто реализовать в алгоритме.

Сформулированное направление исследований оказывается не без подводных камней. Они скрываются в изменчивости и широкой вариабельности изображений отпечатков пальцев, в фрагментарности изображений и поражающих изображение деформациях узоров, в неочевидных искажениях узоров в процессе слепообразования отпечатков на воспринимающей поверхности. Это влияет на состав шаблонов и классификацию структур, подграфов и графов. Для компенсации нежелательного эффекта необходимо разработать устойчивые классификаторы различного уровня и механизм классификации накопленных оценок похожести КТ.

2. Метод ускоренной идентификации

2.1. Классификаторы первого уровня

Рассмотрим поэтапно метод и алгоритм идентификации отпечатков пальцев. На первом этапе определяются классификаторы. Их большая часть рассчитывается при обработке изображений. Основной состав шаблона изображения — это список КТ. Каждая КТ m_i определяется следующим образом:

$$m_i = \{(x_i, y_i), \alpha_i, t_i\}, i \in 1..n, \quad (1)$$

где (x_i, y_i) — координаты m_i ; α_i — вектор направления m_i в диапазоне $[0; 2\pi)$; t_i — тип m_i , $t_i \in \{0, 1\}$, $t_i = 0$ для КТ типа разветвления и $t_i = 1$ для КТ типа окончания [7]; n — количество КТ на изображении. Вектор направления α_i указывает в сторону увеличения

числа линий по касательной к линии, на которой располагается КТ m_i [22]. Дополнительно каждая КТ снабжается описанием некоторой компактной окрестности, которое назовем гнездом.

Рассмотрим следующую последовательность операций.

Выполним обработку и скелетизацию изображения [22]. Этот шаг можно выполнить по-разному [1, 2], но от качества его выполнения зависят ошибки идентификации. На скелете изображения детектируем КТ, часть из которых отмечена точками (см. рис. 1).



Рис. 1. Скелет изображения с группой КТ, детектированных в гнезде, которое построено на основе сечения, показанного толстой линией

От каждой КТ отметим проекции на соседних скелетных линиях. Этих линий две. Проекция перпендикулярна вектору направления КТ (см. рис. 2).

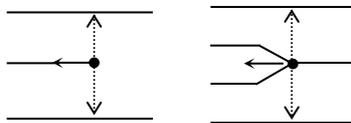


Рис. 2. Проекция от окончания и разветвления

От каждой КТ проведем сечение на несколько линий в обе стороны. Для обеспечения устойчивости сечения оно проводится перпендикулярно линиям (см. рис. 1). Сечением каждая линия разрезается на две части, которые назовем связями. Связи можно нумеровать по линейному порядку, но лучше нумеровать их по часовой стрелке, как показано на рис. 3. Множество связей и признаки, детектированные на связях, образуют гнездо. Таким образом, каждая КТ снабжается гнездом. Два гнезда представлены на рис. 3.

Для КТ типа окончания номер 19 (показано слева на рис. 3) нумерация связей начинается с линии, заканчивающейся окончанием (связь номер 0). Для КТ типа разветвления номер 19 (показано справа) нумерация связей начинается с раздваивающейся линии (связь номер 0). Сечение для КТ типа окончания генерирует 17 связей (0..16), для КТ типа разветвления — тоже 17 связей (0..16). Глубину сечения задают от 2 до 8 линий.

Указанный метод нумерации связей по спирали позволяет легко менять глубину сечения без нарушения структуры гнезда. Это важное свойство.

При конструировании гнезда проследим ход каждой связи, начиная от сечения. Это задает ориентацию связи. Проследивая связь, детектируем КТ, располагаемую на связи, или отмеченную проекцию от другой КТ (КТ номер 20..27 на рис. 3).

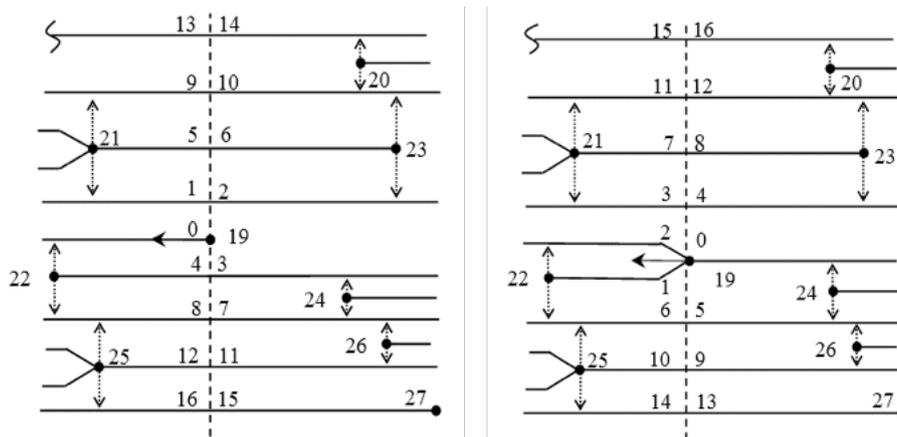


Рис. 3. Сечения для окончания и разветвления с пронумерованными связями

Объем шаблона в основном зависит от количества связей, определяемых по формуле

$$w_i = 4x + 2 + (-1)^{t_i}, \tag{2}$$

где t_i — тип m_i (1); x — число разрезаемых линий. Поэтому число линий, разрезаемых сечением, необходимо задавать аккуратно. С одной стороны, с ростом числа x появляется возможность обогащения гнезда данными. С другой стороны, увеличивается требуемый объем памяти для представления гнезда в алгоритме идентификации. При $x = 7$ объем памяти для размещения одного шаблона прокатного отпечатка пальца колеблется около 30 Кб. Тогда для работы с 1 млн. дактилокарт, а каждая дактилокарта содержит 10 отпечатков пальцев, требуется около 300 Гб памяти. А объем БД современных дактилоскопических систем иногда достигает 100 млн. дактилокарт.

При построении гнезда и прослеживании связей на скелете встречаются КТ или проекции от них, которые формируют события (см. рис. 4). Каждое событие ассоциируется с номером КТ, сформировавшей это событие, и с номером связи [7, 22]. В составе события отражается совпадение или несовпадение ориентации вектора направления КТ с ориентацией связи с точностью до $\pi/2$, наличие на связи КТ или проекции от нее, причем справа или слева от связи, обрыв связи. События 1100 и 0000 не снабжаются номером КТ, потому что связь замыкается, не встречая событие, или обрывается на краю узора. Совокупность событий и ассоциированных с ними номеров КТ, упорядоченных по номерам связей, образует гнездо. Его можно трактовать как ориентированных граф. Достоинства конструкций в виде СВФС и МАГ в гнезде учитываются в полной мере.

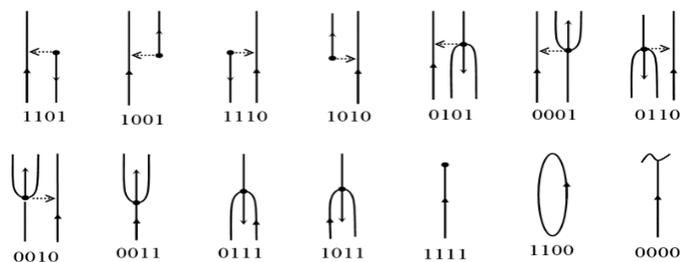


Рис. 4. События, детектируемые на связях при прослеживании скелетных линий

Математически множество связей в гнезде для КТ m_i записывается в виде

$$N_i = \{m_k, d_{ik}, r_{ik}, \gamma_{ik}, v_{ik}\}, m_k = \{(x_k, y_k), \alpha_k, t_k\}, k \in K, \tag{3}$$

где d_{ik} , — расстояние между КТ m_i и m_k ; r_{ik} — гребневый счет между ними; γ_{ik} — азимут как угол поворота вектора направления α_i до луча, ориентированного от m_i на

m_k (см. рис. 5), α_k — вектор направления (1); v_{ik} — событие, сформированное m_k , в гнезде для m_i (см. рис. 4); K — количество связей в гнезде. Ясно, что ассоциированные со связями данные устойчивы: они не зависят от ориентации и положения системы координат. Они же инвариантны к масштабу изображения, кроме d_{ik} .

Рассмотрим КТ m_i . Она и КТ на связи m_k , $k \in K$, образуют диполь (см. рис. 5). Диполь снабжается длиной d_{ik} , азимутом γ_{ik} , гребневым счетом r_{ik} (3), а также углом φ_{ik} как разностью направлений векторов α_i и α_k (1). Диполь как вектор выражается в виде

$$D_{ik} = (\gamma_{ik}, d_{ik}, r_{ik}, \varphi_{ik}). \quad (4)$$

Для КТ m_i на основе гнезда можно построить множество диполей. Назовем это множеством диполей гнезда (MND — Multiplicity of Nest Dipoles). На рис. 5 показаны два диполя из запросного и ссылочного шаблонов. В ссылочном шаблоне рассматривается гнездо для КТ m_j .

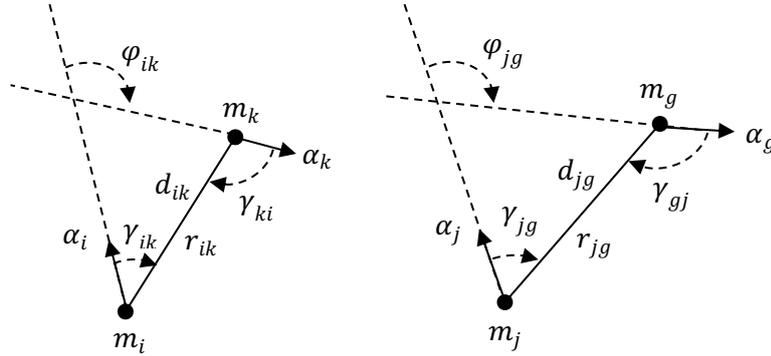


Рис. 5. Параметры диполей для запросного и ссылочного шаблонов

Все построенные диполи MND сохраняются в шаблоне. Сравним каждый диполь $\omega(m_i, m_k)$ из запросного шаблона с каждым диполем $\omega(m_j, m_g)$ из ссылочного шаблона. Оценим степень их похожести по формуле:

$$\psi(\omega(m_i, m_k), \omega(m_j, m_g)) = f(\Delta d, \Delta \gamma_1, \Delta \gamma_2, \Delta \gamma_3). \quad (5)$$

Здесь $\Delta d = \|d_{ik} - d_{jg}\|$ — норма разности длин диполей (см. рис. 5); $\Delta \gamma_1$ — норма разности углов азимутов γ_{ik} и γ_{jg} ; $\Delta \gamma_2$ — норма разности углов азимутов γ_{ki} и γ_{gj} ; $\Delta \gamma_3$ — норма разности углов φ_{ik} и φ_{jg} , см. (4); функция f является полиномом первого порядка. Нормируем значение функции $0 \leq \psi \leq 1$ [23]. Если $\psi = 1$, то диполи идентичны.

Степень похожести диполей по топологии зависит от события v_{ik} запросного шаблона и события v_{jg} ссылочного шаблона (3). Зависимость определяется составом бит в событии, см. рис. 4. Рассматривая состав бит, определим числа τ_{ik} и τ_{jg} как состояние связей. Состояние связи равно нулю, если вектор направления КТ ориентирован вдоль связи, иначе единица [7, 22]. Оно кодируется соответствующим событием с помощью функции как таблицы $T_1 = (\sim, 0, 0, 0, \sim, 1, 1, 1, \sim, 0, 0, 1, \sim, 1, 1, 1)$ в виде:

$$\tau_{ik} = T_1(v_{ik}),$$

$$\tau_{jg} = T_1(v_{jg}).$$

В таблице T_1 символ \sim означает отсутствие состояния связи. Такая связь не используется.

Для оценки степени похожести диполей по топологии применим операцию «исключающее или»:

$$\lambda(\omega(m_i, m_k), \omega(m_j, m_g)) = \tau_{ik} \oplus \tau_{jg}. \quad (6)$$

Ясно, что $\lambda \in \{0, 1\}$.

Для расчета финальной степени похожести двух диполей умножим оценки (5) и (6):

$$s_{ik,jg} = \psi(\omega(m_i, m_k), \omega(m_j, m_g)) * \lambda(\omega(m_i, m_k), \omega(m_j, m_g)). \quad (7)$$

Очевидно, что $0 \leq s_{ik,jg} \leq 1$. Для ускорения операций по вычислению различных норм необходимо задать длину диполей и величин углов в целочисленном диапазоне, например, с помощью квантования, а затем применить сложение, вычитание и вычисление абсолютных значений многобайтовых величин [24].

Для каждой КТ в ее гнезде выберем по крайней мере две ближайшие КТ. Например, в гнезде КТ m_i на рис. 6 выбраны две КТ m_l и m_r . Чем богаче выбор, тем выше надежность идентификации и ниже производительность алгоритма. Соответствующий гребневый счет равен r_{il} и r_{ir} . Угол азимута $\gamma_{ik} \in [0; 2\pi), k \in \{l, r\}$, с помощью квантования преобразуется к целочисленному углу $\gamma_{ik} \in 0..63$, который рассматривается как один из 64 возможных классов.

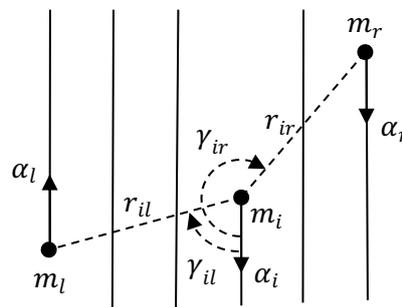


Рис. 6. Два диполя для образующей гнездо КТ m_i

Такого числа классов недостаточно: высока вероятность ложного сравнения двух различных диполей. Дополнительный бит в описании диполя позволяет удвоить число классов. Это бит состояния связи $\tau_{ik} = T_1(v_{ik})$, где $k \in \{l, r\}$, $\tau_{ik} \in \{0,1\}$. Таким образом, число классов становится равным 128.

Введем еще один реверсивный бит состояния связи τ_{ki} , который вычисляется в гнезде для КТ $m_k, k \in \{l, r\}$ аналогичным способом. Тогда классы $c_{ik}, k \in \{l, r\}$, для диполей на рис. 6, определяются в виде одного байта (см. рис. 7).

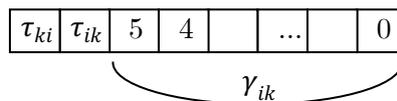


Рис. 7. Структура класса c_{ik}

Таким образом, число классов расширяется до 256. Перепишем структуру диполей MND как вектора (4), считая класс диполя (см. рис. 7) его первым компонентом:

$$D_{ik} = \{c_{ik}, d_{ik}, r_{ik}, \varphi_{ik}\}, k \in \{l, r\}. \quad (8)$$

При обработке изображения упорядочим диполи по номеру классов и сформируем список диполей для каждого шаблона. Эта предварительная операция упорядочения существенна. Вычислим пересечение списков запросного и ссылочного шаблонов [23]. Если классы совпадают, вычислим разность расстояний d_{ik} и d_{jg} , разность гребневого счета r_{ik} и r_{jg} , разность углов φ_{ik} и φ_{jg} (см. рис. 5). Если разности в пределах заданных допусков, то КТ m_i и m_j рассматриваются как похожие и сохраняются в списке L_0 . Список L_0 оформляется в виде пар индексов

$$L_0 = \{(i, j)\}. \quad (9)$$

Величины допусков определяются на этапе обучения алгоритма. Сравнение классов диполей выполняется столь же быстро, как и сравнение целых значений. Остальные вычисления выполняются строго после сравнения классов. Таким образом, список L_0 формируется очень быстро. Для узора, содержащего 30..100 КТ, обычно строят 60..200 диполей. С учетом сравнения классов как пересечения списков [23], контроля допусков для разностей параметров диполей и алгоритмических трюков [24] при сопоставлении двух шаблонов на основе классификации всех возможных пар КТ на практике требуется менее 10^3 вычислительных операций.

2.2. Классификаторы второго уровня

В списке L_0 согласно (9) сохраняется много предполагаемых похожих пар КТ, причем одной и той же КТ из запросного шаблона может быть поставлено в соответствие несколько КТ из ссылочного шаблона. Ясно, что избирательность классификаторов первого уровня недостаточна. Поэтому введем классификаторы на основе топологии гнезда.

В результате воздействия помех тип КТ может мутировать. На рис. 3 КТ типа окончания 19 мутирует влево и замыкается в КТ типа разветвления 19. Видно, что связи номер 0 для КТ типа окончания соответствует связь номер 2 для КТ типа разветвления, связи номер 1 — связь номер 3 и т.д. Мутации δ распознают по несовпадающим типам КТ согласно (1).

Если мутации нет, то связи не перенумеровывают, а используют как есть по формуле

$$H_0(j) = \{0,1,2, 3,4,5,6, 7,8,9,10, 11,12,13,14\}, \quad \delta = 0.$$

При замыкании КТ типа окончания влево связи перенумеровывают по формуле

$$H_1(j) = \{2,3,4, 0,1,7,8, (5,6,11,12), (9,10,15,16)\}, \quad \delta = 1.$$

При замыкании КТ типа окончания вправо связи перенумеровывают по формуле

$$H_2(j) = \{1,2,0, 5,6,3,4, (9,10,7,8), (13,14,11,12)\}, \quad \delta = 2.$$

При разрыве КТ типа разветвления влево связи перенумеровывают по формуле

$$H_3(j) = \{2,0,1, 5,6,3,4, (9,10,7,8), (13,14,11,12)\}, \quad \delta = 3.$$

При разрыве КТ типа разветвления вправо связи перенумеровывают по формуле

$$H_4(j) = \{3,4,0, 1,2,7,8, (5,6,11,12), (9,10,15,16)\}, \quad \delta = 4.$$

Каждая формула — вектор, в котором индекс элемента (счет от 0) есть номер связи для КТ из запросного шаблона. Значение элемента — номер связи для КТ из ссылочного шаблона. Так связи номер 7 (элемент номер 7) КТ типа окончания соответствует связь номер 5 КТ типа разветвления, см. $H_1(j)$. А связи номер 11 (элемент номер 11) — связь номер 9 так же по $H_1(j)$. Круглыми скобками выделены элементы, значения которых подчиняются закону периодичности. Последующие значения вычисляются добавлением числа 4 ($9=5+4$) (см. рис. 3). Поэтому легко строить произвольную глубину сечения.

Выберем пару КТ m_i и m_j из списка L_0 (см. рис. 5). Гнезда каждой КТ образуют набор состояний связей $\sigma_i = \{\tau_{ik1}, \tau_{ik2}, \dots\}$ и $\sigma_j = \{\tau_{jg1}, \tau_{jg2} \dots\}$ согласно формулам (3) и (6). Для 32 связей образуется два 32-битных слова σ_i и σ_j . Номер связи (см. рис. 3) определяет позицию соответствующего бита состояния этой связи. Таким образом, 32-битное слово отражает топологические свойства некоторой окрестности изображения. Сравним слова $\sigma_{ij} = \sigma_i \oplus \sigma_j$ побитовой операцией «исключающее или» и инвертируем значения бит в слове σ_{ij} , чтобы биты для совпадающих по состоянию связей стали равны единице. Введем

таблицу $T_2 = (0,1,1,1,0,1,1,1,0,1,1,1,0,1,1,1)$ для маскирования событий 1100 и 0000 (см. рис. 4) и на основе событий согласно (3) определим маскирующие биты в виде

$$w_{ik} = T_2(v_{ik}),$$

$$w_{jg} = T_2(v_{jg}).$$

Два набора маскирующих бит $\mu_i = \{w_{ik1}, w_{ik2}, \dots\}$ и $\mu_j = \{w_{jg1}, w_{jg2} \dots\}$ формируют две 32-битные маски. Вычислим обобщенную маску. $\mu_{ij} = \mu_i \& \mu_j$, применяя операцию «побитовое и», и функцией f подсчитаем число единичных бит $b_{ij} = f(\sigma_{ij} \& \mu_{ij})$ в слове σ_{ij} [24].

Для учета возможных мутаций δ типов КТ воспользуемся таблицами перенумерации связей $H_\delta(j)$, определим максимум числа подобных связей в гнездах по топологии в виде

$$b_{ij}^\delta = \max_{\delta} b_{iH_\delta(j)} \quad (10)$$

и сформируем список L_1 для лучших оценок b_{ij}^δ пар КТ m_i и m_j :

$$L_1 = \{(i, j, b_{ij}^\delta)\}, i \in \{1, n_1\}, j \in \{1, n_2\}, \quad (11)$$

где n_1 — количество КТ на запросном шаблоне, n_2 — на ссылочном. Упорядочим список L_1 по оценкам b_{ij}^δ . Список L_1 , построенный на классификаторах второго уровня, отражает значительно лучшую избирательность похожих КТ и формируется очень быстро, так как слова и маски рассчитываются на этапе обработки узора. Для оценки классификаторов второго уровня (вызов функции f [24]) при совпадении типов двух КТ требуется менее 25 одноканальных команд, а при несовпадении — менее 50. Классификаторы первого уровня пропускают лишь долю от числа $n_1 n_2$ сравниваемых КТ, равную, как правило, 20..200. Поэтому классификаторы второго уровня оцениваются менее чем за $50 * 200 = 10000$ вычислительных операций, а с учетом накладных расходов число операций удваивается.

2.3. Классификаторы третьего уровня

Классификаторы второго уровня отражают подобие окрестностей гнезд КТ по топологии. Следующий шаг в повышении избирательности шаблонов заключается в аккумуляции наиболее подобных по топологии гнезд. Для этого из упорядоченного списка L_1 по (11) выберем M лучших пар КТ по оценке (10), причем $M \leq \min(n_1, n_2)$, и оценим степень похожести двух шаблонов в виде

$$s = \sum_{m=1}^M b_{i_m j_m}^\delta, \quad (12)$$

где индексы i_m, j_m уникальны и не дублированы. Значение M определяется при обучении алгоритма. Его увеличение улучшает отбор шаблонов, но замедляет алгоритм.

Сравнивая запросный шаблон с каждым ссылочным шаблоном из БД, построим гистограмму похожести шаблонов $H(s)$ для запросного шаблона, где s определяется по критерию (12). Таким образом, классификаторы третьего уровня аккумуляруются в гистограмме. Опираясь на уже вычисленные оценки (10), оценка по (12) выполняется за M команд суммирования.

На этом первый этап идентификации шаблонов отпечатков пальцев заканчивается. Он обеспечивает основное ускорение метода идентификации.

2.4. Алгоритм идентификации шаблонов

Второй этап идентификации опирается на гистограммные оценки.

Построим функцию распределения вероятностей $F(s)$ для гистограммы $H(s)$ и определим квантиль уровня p для лучших оценок по критерию (12). Он определит пороговую величину для доли ссылочных шаблонов БД, которую обозначим в виде $F^{-1}(p)$. Для ускорения идентификации список L_1 можно сохранить на первом этапе.

Для выбранной пары запросного и одного ссылочного шаблона с оценкой $b_{ij}^\delta > F^{-1}(p)$ рассчитаем финальную степень подобия.

Из списка L_1 выберем $K = \sqrt{n_1 n_2}$ наиболее похожих пар КТ по оценке (10). Рассмотрим пару КТ m_i и m_j (см. рис. 3). Обозначим числами K_i и K_j количество связей в гнезде для КТ m_i и m_j соответственно. Воспользуемся геометрическими и топологическими характеристиками изображения [25, 26] и оценим подобие гнезд по формуле

$$Q_{ij} = \sum_{k,g=1}^{\min(K_i, K_j)} \max_{\delta} \psi \left(\omega(m_i, m_k), \omega(m_j, m_{H\delta(g)}) \right), \quad (13)$$

где ψ вычисляется по оценке (5) с учетом возможных мутаций δ для КТ из ссылочного шаблона. Диполи сравниваются на идентичных или перенумерованных связях.

На основе списка L_1 по (11) и оценки подобия по критерию (13) сконструируем список L_2 в виде

$$L_2 = \{(i, j, Q_{ij})\}, i \in \{1, n_1\}, j \in \{1, n_2\}, \quad (14)$$

где n_1 и n_2 — количество КТ как в (11). Упорядочим список L_2 по подобию (13).

Список L_2 сохраняет множество индексов пар КТ. Выберем лучшую пару КТ m_{i_1} и m_{j_1} из списка L_2 по (14), $i_1 \in \{1, n_1\}, j_1 \in \{1, n_2\}$. Положим число $t \leftarrow 1$ пар КТ выбранных из списка L_2 и зададим пороговое значение thr . Построим стек L_3 и очистим его $L_3 \leftarrow \{\emptyset\}$. Обнулیم переменные $V_1 \leftarrow 0, V_2 \leftarrow 0, N \leftarrow 0$. Зададим длину маршрута len , функцию сортировки $sort$, функцию чтения вершины стека pop .

Алгоритм развивается от начальной пары КТ m_{i_1} и m_{j_1} . Он представлен на рис. 8 и состоит из следующих шагов.

```

1:  $L_3 \leftarrow L_3 + (i_1, j_1); (i_1, j_1) \in L_2; Q_{i_1 j_1} \leftarrow 0$ 
2:  $V_1 \leftarrow 0; N \leftarrow 0$ 
3: for all  $(i, j) \in L_2$  do
    if  $Q_{ij} = 0$  then оценим  $Q_{ij}$  по (13)
4:  $L_3 \leftarrow sort L_3$ ; на вершине стека располагается лучшая пара КТ
5: Читаем пару КТ  $(i, j) \leftarrow pop L_3$ ; обновляем  $N \leftarrow N + 1$ 
6: Оценим  $Q_{ij}$  по (13) и аккумулируем оценки  $V_1 \leftarrow V_1 + Q_{ij}$ 
7: Удалим пары КТ с теми же индексами  $i$  или  $j$  из  $L_3$ 
    for all  $(a, b) \in L_3$  do
        if  $a = i$  or  $b = j$  then
             $L_3 \leftarrow L_3 - (a, b)$ 
8: Сравним диполи  $D_{ik}$  и  $D_{jg}$  и рассчитаем степень похожести  $s_{ik, jg}$ 
    согласно формулам (4)–(7) в зависимости от равенства классов,
    разностей углов  $\varphi_{ik}$  и  $\varphi_{jg}$ , углов азимутов, расстояний  $d_{ik}$  и  $d_{jg}$ 
    и гребневого счета  $r_{ik}$  и  $r_{jg}$  (см. рис. 5);
    если разности в пределах заданных допусков, то  $L_3 \leftarrow L_3 + (k, g)$ 
    и  $Q_{kg} \leftarrow 0$ ;
    пара КТ  $m_k$  и  $m_g$  рассматривается как кандидат идентичных КТ;
    согласно формуле (8) может сформироваться множество пар
     $(m_{k_1}, m_{g_1}), (m_{k_2}, m_{g_2})$  и т.д.;
    все они сохраняются в  $L_3$ 
9: if  $L_3 \neq \{\emptyset\}$  and  $N < len$  then go to 3
10: if  $V_2 < V_1$  then  $V_2 \leftarrow V_1$ 
11: Очистим стек  $L_3 \leftarrow \{\emptyset\}$ 
    if  $t < thr$  and  $L_2 \neq \{\emptyset\}$  then
         $L_3 \leftarrow L_3 + (i_{t+1}, j_{t+1}), (i_{t+1}, j_{t+1}) \in L_2; Q_{i_{t+1} j_{t+1}} \leftarrow 0$ 
12: if  $L_3 \leftarrow \{\emptyset\}$  then go to 2
13: return  $V_2$ 

```

Рис. 8. Алгоритм идентификации шаблонов отпечатков пальцев

От начальных пар КТ маршруты проходят по другим КТ как по вершинам графов. Лучшая оценка одного из маршрутов есть степень подобия шаблонов.

Согласно алгоритму, маршруты тяготеют к вершинам графов с наиболее похожими гнездами, которые консолидируются. Это близко к работам [16, 19]. Финальная оценка шаблонов находится в виде

$$MS = V_2 / \min(n_1, n_2, len), \quad (15)$$

где n_1 — количество КТ запросного шаблона, n_2 — ссылочного шаблона, len — заданная длина маршрута. Для шаблонов вне доли БД, определенной величиной $F^{-1}(p)$, положим $MS = 0$. На этом заканчивается второй этап идентификации.

3. Вычислительные эксперименты

Для тестирования алгоритма было набрано 8788 изображений из архивов FVC 2000, 2002, 2004 и 2006 годов (только оптические сенсоры) [27]. Из этих БД дубли изображений были обнаружены и удалены. В полученной тестовой БД сохранены изображения хорошего, среднего, плохого качества и с малым пересечением фрагментов изображений.

На рис. 9 показана зависимость вероятности ошибок распознавания p (вероятность ошибки первого рода, FRR — false rejected rate) от числа n (млн. сравнений пар шаблонов в секунду) на процессоре Intel i7-4770@3,4 ГГц на тестовой БД. Ошибка p измеряется в зависимости от вероятности ошибки второго рода (FMR — false match rate) [2, 13, 19]. Верхний график соответствует ошибке первого рода при FMR равной 10^{-5} (далее FMR₁₀₀₀₀), средний — при FMR равной 10^{-3} (далее FMR₁₀₀₀), нижний — равным ошибкам первого и второго рода (EER — equal error rate).

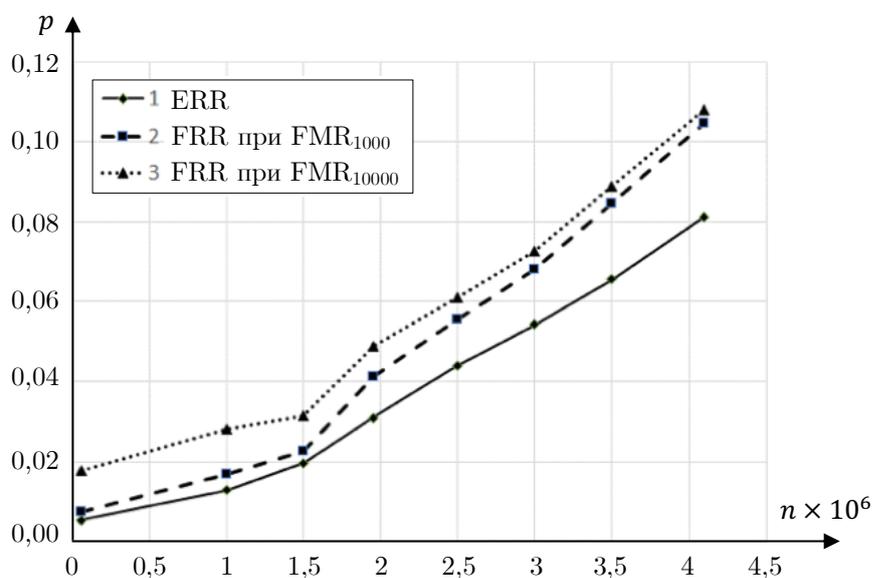


Рис. 9. Графики ошибок первого рода при идентификации

Данные результатов измерений ошибок распознавания, по которым построены графики на рис. 9, приведены в табл. 1.

Производительность алгоритма регулировалась величинами допусков, определяющих работу классификаторов всех трех уровней, а также долей ссылочных шаблонов. Минимальные ошибки распознавания получены без применения ускоряющих классификаторов и по всей тестовой БД (всего 55 тыс. сравнений пар шаблонов в секунду, см. табл. 1). Следует обратить внимание на наклон ломанных, который показывает, во сколько раз рост производительности алгоритма опережает рост ошибок распознавания. В экспериментах это отношение варьируется от 4 до 8.

Таблица 1

Экспериментальные данные результатов идентификации

Сравнений млн/с	EER	FMR ₁₀₀₀	FMR ₁₀₀₀₀
0,055	0,0053	0,0073	0,0177
1	0,0128	0,0167	0,0279
1,5	0,0196	0,0225	0,0315
1,95	0,0309	0,0412	0,0491
2,5	0,0441	0,0555	0,0611
3	0,0543	0,0681	0,0727
3,5	0,0655	0,0845	0,0887
4,1	0,0813	0,1045	0,1078

Для сравнения минимальных ошибок распознавания (без ускорителей) с ошибками распознавания других разработчиков из различных стран метод, реализованный в алгоритме FPM, был проверен международным тестом верификации отпечатков пальцев на сайте FVC-ongoing [25]. Наиболее интересные результаты, опубликованные для базы изображений FV-HARD-1.0, представлены в табл. 2. Обработка изображений, на которую опирается алгоритм, изложена в [26]. Таким образом, минимальные ошибки распознавания на рис. 9 соответствуют лидирующим результатам независимого международного тестирования. Необходимо заметить, что для малых ошибок значимый наклон ломанных линий обеспечить очень трудно.

Таблица 2

Результаты независимого международного тестирования на сайте FVC-ongoing

Алгоритм	Время обработки, мс	Время сравнения, мс	FMR ₁₀₀₀ , %	FMR ₁₀₀₀₀ , %
MM_FV	855	51	0,342	0,626
FPM	323	26	0,952	1,227
EMB9300	82	10	1,092	1,542
TigerAFIS	464	20	1,077	1,781
HXKJ	1413	33	0,797	1,879

Заключение

В статье предлагается метод и алгоритм ускорения идентификации изображений отпечатков пальцев. Он состоит из двух этапов. На первом этапе применяются классификаторы трех уровней, позволяющие ускорить отбор идентичных пар КТ и сократить долю ссылочных шаблонов для второго этапа идентификации. Для этого используются геометрические и топологические характеристики узоров, а также гистограммные оценки. На втором этапе выполняется детальное сравнение шаблонов, но только для малой доли шаблонов из БД. Применение многоуровневых классификаторов и гистограммных оценок позволяет значительно увеличить отношение величины роста производительности алгоритма к величинам роста ошибок идентификации и создать программу коммерческого уровня.

Предложенный метод опирается на топологические события и состояния связей, определяемые для каждой КТ на этапе обработки. Эти признаки сохраняются в шаблоне.

Размер шаблона увеличивается незначительно. Большая часть классификаторов рассчитывается на этапе обработки изображения и используется при идентификации шаблонов в готовом виде. Это позволяет применить однократные вычислительные операции при построении списков L_0 и L_1 и значительно уменьшить время идентификации.

Дальнейшее направление исследований видится в развитии числа уровней классификаторов, проверке алгоритма на отдельной БД FVC-2004-DB1 [27], участии в тестировании на сайте NIST USA, применении набора команд микроархитектуры Intel Core SSE4, во внедрении нейротехнологий в состав действующего алгоритма.

Литература

1. Bolle R.M., Connel J.Y., Pankanti S., et al. Guide to Biometrics. New York, Springer-Verlag, 2004. 364 p. DOI: 10.1007/978-1-4757-4036-3.
2. Maltoni D., Maio D., Jain A.K., et al. Handbook of Fingerprint Recognition. London, Springer-Verlag, 2009. 494 p. DOI: 10.1007/978-1-84882-254-2.
3. ISO/IEC 19794-2:2011. Information technology – Biometric data interchange formats – Part 2: Finger minutiae data (дата обращения: 23.07.2020).
4. Bae G., Lee H., Hwang S.D., et al. Secure and Robust User Authentication Using Partial Fingerprint Matching // Proceedings of 2018 IEEE International Conference on Consumer Electronics, ICCE. 2018. P. 1–6. DOI: 10.1109/icce.2018.8326078.
5. Hidayat R., Souvanlit K., Bejo A. An Improvement of Minutiae-based Fingerprint Matching: Two Level of Scoring System // Proceedings of 2016 International Symposium on Electronics and Smart Devices, ISESD. 2016. P. 264–267. DOI: 10.1109/ISESD.2016.7886730.
6. Singh P., Kaur L. Fingerprint Feature Extraction Using Morphological Operations // Proceedings of 2015 International Conference on Advances in Computer Engineering and Applications. 2015. P. 764–767. DOI: 10.1109/ICACEA.2015.7164805.
7. Гудков В.Ю. Способ математического описания и идентификации отпечатков пальцев // под ред. член-корр. РАН В.Л. Арлазарова и д.т.н. проф. Н.Е. Емельянова // Обработка изображений и анализ данных: Труды ИСА РАН. М.: ЛИБРОКОМ. 2008. Т. 38. С. 336–356.
8. Liao C.C., Chiu C.T. Fingerprint Recognition with Ridge Features and Minutiae on Distortion // Proceedings of 2016 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP. 2016. P. 2109–2113. DOI: 10.1109/ICASSP.2016.7472049.
9. Barman S., Chattopadhyay S., Samanta D., et al. An Efficient Fingerprint Matching Approach Based on Minutiae to Minutiae Distance Using Indexing With Effectively Lower Time Complexity // Proceedings of 2014 International Conference on Information Technology. 2014. P. 179–183. DOI: 10.1109/ICIT.2014.46.
10. Tran M.H., Duong T.N., Nguyen D.M., et al. A Local Feature Vector for an Adaptive Hybrid Fingerprint Matcher // 2017 International Conference on Information and Communications, ICIC. 2017. P. 249–253. DOI: 10.1109/INFOC.2017.8001668.
11. Гудков В.Ю. Модель гребневого счета на основе топологии дактилоскопического изображения // Вестник ЧелГУ. 2011. № 13. С. 99–108.
12. Jiang X., Yau W.Y., Fingerprint Minutiae Matching Based on the Local and Global Structures // Proceedings of the 15th International Conference on Pattern Recognition, ICPR-2000. 2000. Vol. 2. P. 1038–1041. DOI: 10.1109/ICPR.2000.906252.
13. Feng Y., Feng J., Chen X., et al. A Novel Fingerprint Matching Scheme Based on Local Structure Compatibility // Proceedings of the 18th International Conference on Pattern Recognition, ICPR'06. 2006. P. 374–377. DOI: 10.1109/ICPR.2006.137.

14. Cao J., Feng J. A Robust Fingerprint Matching Algorithm Based on Compatibility of Star Structures // Proceedings of the Sixth International Symposium on Multispectral Image Processing and Pattern Recognition, MIPPR 2009. 2009. Vol. 7498. Remote Sensing and GIS Data Processing and Other Applications, 74983X. DOI: 10.1117/12.832357.
15. Ratha N.K., Pandit V.D., Bolle R.M., et al. Robust Fingerprint Authentication Using Local Structure Similarity // Workshop on Applications of Computer Vision. 2000. P. 29–34. DOI: 10.1109/WACV.2000.895399.
16. Chikkerur S., Cartwright A., Govindaraju V. K-plet and CBFS: A graph Based Fingerprint Representation // Proceedings of the International Conference on Biometrics, ICB 2006: Advances in Biometrics. 2006. P. 309–315. DOI: 10.1007/11608288_42.
17. Chen X., Wang L., Li M. An Efficient Graph-Based Algorithm for Fingerprint Representation and Matching // Proceedings of the 3rd International Conference on Multimedia Technology, ICMT 2013. 2013. P. 1019–1029. DOI: 10.2991/icmt-13.2013.125.
18. Leslie S., Sumathi C.P. A Robust Hierarchical Approach to Fingerprint Matching Based on Global and Local Structures // International Journal of Applied Engineering Research. 2018. Vol. 13, no. 7. P. 4730–4739.
19. Capelli R., Ferrara M., Maltoni D. Fingerprint Indexing Based on Minutia Cylinder-Code // IEEE transactions on pattern analysis and machine intelligence. 2011. Vol. 33, no. 5. P. 1051–1057. DOI: 10.1109/TPAMI.2010.228.
20. Zheng F., Yang C. Latent Fingerprint Match using Minutia Spherical Coordinate Code // International Conference on Biometrics, ICB 2015 (Phuket, Thailand, May, 19-22, 2015). 2015. P. 357–362. DOI: 10.1109/ICB.2015.7139061.
21. Tabassi E., Wilson C., Watson C. Fingerprint Image Quality. NIST Internal Report 7151, National Institute for Standards and Technology, 2004. URL: <https://www.nist.gov/publications/fingerprint-image-quality> (дата обращения: 23.07.2020).
22. Гудков В.Ю., Аркабаев Д.И. Способ сравнения отпечатков папиллярных узоров. РФ Пат. 2331108, МПК G 06 K 9/62, 2008. Бюл. 22.
23. Новиков Ф.А. Дискретная математика для программистов: Учебник для вузов, 3-е изд. СПб: Питер, 2009. 384 с.
24. Warren H.S. Hacker’s Delight, 2nd ed. Addison-Wesley Professional, 2018. 512 p.
25. Dorizzi B., Cappelli R., Ferrara M., et al. Fingerprint and On-Line Signature Verification Competitions at ICB 2009 // Proceedings of the International Conference on Biometrics, ICB 2009. 2009. P. 725–732. DOI: 10.1007/978-3-642-01793-3_74.
26. Гудков В.Ю. Методы первой и второй обработки дактилоскопических изображений. Миасс: Геотур, 2009. 237 с.
27. FVC — ongoing: on-line evaluation of fingerprint recognition algorithms. URL: <https://biolab.csr.unibo.it/FVCOnGoing/UI/Form/Home.aspx> (дата обращения: 23.07.2020).

Гудков Владимир Юльевич, д.ф.-м.н., профессор, кафедра ЭВМ, Южно-Уральский государственный университет (национальный исследовательский университет) (Челябинск, Российская Федерация)

ACCELERATED FINGERPRINT IDENTIFICATION METHOD

© 2021 V.J. Gudkov

*South Ural State University (pr. Lenina 76, Chelyabinsk, 454080 Russia)**E-mail: diana@sonda.ru*

Received: 11.08.2020

The article outlines a method for accelerated identification of fingerprint images based on templates as image models. They are formed as a result of automatic processing of images. The method is based on the properties of the nearest neighborhoods of minutiae in the form of endings and bifurcations and consists of two stages. At the first stage, each minutia of the query template is compared with each minutia of the reference template from the database and the similarity of such pairs of minutiae are estimated. To speed up computational operations, classes are introduced that allow you quickly accumulate the similarity of minutiae from these two templates in a histogram. Histograms are built for all reference templates from the database and one query template. At the second stage, based on histogram estimates, the most similar templates are selected, the number of which is much less than the size of the database. These templates are compared additionally taking into account the consolidation of minutiae and the compactness of the location of the corresponding pairs of minutiae. Significant acceleration of the identification algorithm is achieved by discarding dissimilar pairs of minutiae at the first stage and pairs of patterns with poor histogram estimates at the second stage. The results of experiments are presented, which are published on the Internet.

Keywords: fingerprint, identification, minutia, histogram.

FOR CITATION

Gudkov V.J. Accelerated Fingerprint Identification Method. *Bulletin of the South Ural State University. Series: Computational Mathematics and Software Engineering*. 2021. Vol. 10, no. 1. P. 32–48. (in Russian) DOI: 10.14529/cmse210103.

This paper is distributed under the terms of the Creative Commons Attribution-Non Commercial 3.0 License which permits non-commercial use, reproduction and distribution of the work without further permission provided the original work is properly cited.

References

1. Bolle R.M., Connel J.Y., Pankanti S., et al. Guide to Biometrics. New York, Springer-Verlag, 2004. 364 p. DOI: 10.1007/978-1-4757-4036-3.
2. Maltoni D., Maio D., Jain A.K., et al. Handbook of Fingerprint Recognition. London, Springer-Verlag, 2009. 494 p. DOI: 10.1007/978-1-84882-254-2.
3. ISO/IEC 19794-2:2011. Information technology – Biometric data interchange formats – Part 2: Finger minutiae data (accessed: 23.07.2020).
4. Bae G., Lee H., Hwang S.D., et al. Secure and Robust User Authentication Using Partial Fingerprint Matching. Proceedings of 2018 IEEE International Conference on Consumer Electronics, ICCE. 2018. P. 1–6. DOI: 10.1109/icce.2018.8326078.
5. Hidayat R., Souvanlit K., Bejo A. An Improvement of Minutiae-based Fingerprint Matching: Two Level of Scoring System. Proceedings of 2016 International Symposium on Electronics and Smart Devices, ISESD. 2016. P. 264–267. DOI: 10.1109/ISESD.2016.7886730.
6. Singh P., Kaur L. Fingerprint Feature Extraction Using Morphological Operations. Proceedings of 2015 International Conference on Advances in Computer Engineering and Applications. 2015. P. 764–767. DOI: 10.1109/ICACEA.2015.7164805.

7. Gudkov V.J. Methods for Mathematical Description and Identification of Fingerprints. Editor Arlazarov V.L., Emeljanov N.E. Image Processing and Data Analysis: Proceedings of ISA RSA. LIBROKOM. 2008. Vol. 38. P. 336–356. (in Russian)
8. Liao C.C., Chiu C.T. Fingerprint Recognition with Ridge Features and Minutiae on Distortion. Proceedings of 2016 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP. 2016. P. 2109–2113. DOI: 10.1109/ICASSP.2016.7472049.
9. Barman S., Chattopadhyay S., Samanta D., et al. An Efficient Fingerprint Matching Approach Based on Minutiae to Minutiae Distance Using Indexing With Effectively Lower Time Complexity. Proceedings of 2014 International Conference on Information Technology. 2014. P. 179–183. DOI: 10.1109/ICIT.2014.46.
10. Tran M.H., Duong T.N., Nguyen D.M., et al. A Local Feature Vector for an Adaptive Hybrid Fingerprint Matcher. 2017 International Conference on Information and Communications, ICIC. 2017. P. 249–253. DOI: 10.1109/INFOC.2017.8001668.
11. Gudkov V.J. Ridge Counting Model Based on the Topology of a Fingerprint Image. Bulletin of Chelyabinsk State University. 2011. Vol. 13. P. 99–108. (in Russian)
12. Jiang X., Yau W.Y. Fingerprint Minutiae Matching Based on the Local and Global Structures. Proceedings of the 15th International Conference on Pattern Recognition, ICPR-2000. 2000. Vol. 2. P. 1038–1041. DOI: 10.1109/ICPR.2000.906252.
13. Feng Y., Feng J., Chen X., et al. A Novel Fingerprint Matching Scheme Based on Local Structure Compatibility. Proceedings of the 18th International Conference on Pattern Recognition, ICPR'06. 2006. P. 374–377. DOI: 10.1109/ICPR.2006.137.
14. Cao J., Feng J. A Robust Fingerprint Matching Algorithm Based on Compatibility of Star Structures. Proceedings of the Sixth International Symposium on Multispectral Image Processing and Pattern Recognition, MIPPR 2009. 2009. Vol. 7498. Remote Sensing and GIS Data Processing and Other Applications, 74983X. DOI: 10.1117/12.832357.
15. Ratha N.K., Pandit V.D., Bolle R.M., et al. Robust Fingerprints Authentication Using Local Structure Similarity. Workshop on Applications of Computer Vision. 2000. P. 29–34. DOI: 10.1109/WACV.2000.895399.
16. Chikkerur S., Cartwright A., Govindaraju V. K-plet and CBFS: A graph Based Fingerprint Representation. Proceedings of the International Conference on Biometrics, ICB 2006: Advances in Biometrics. 2006. P. 309–315. DOI: 10.1007/11608288_42.
17. Chen X., Wang L., Li M. An Efficient Graph-Based Algorithm for Fingerprint Representation and Matching. Proceedings of the 3-rd International Conference on Multimedia Technology, ICMT 2013. 2013. P. 1019–1029. DOI: 10.2991/icmt-13.2013.125.
18. Leslie S., Sumathi C.P. A Robust Hierarchical Approach to Fingerprint Matching Based on Global and Local Structures. International Journal of Applied Engineering Research. 2018. Vol. 13, no. 7. P. 4730–4739.
19. Capelli R., Ferrara M., Maltoni D. Fingerprint Indexing Based on Minutia Cylinder-Code. IEEE transactions on pattern analysis and machine intelligence. 2011. Vol. 33, no. 5. P. 1051–1057. DOI: 10.1109/TPAMI.2010.228.
20. Zheng F., Yang C. Latent Fingerprint Match using Minutia Spherical Coordinate Code, International Conference on Biometrics, ICB 2015 (Phuket, Thailand, May, 19-22, 2015). 2015. P. 357–362. DOI: 10.1109/ICB.2015.7139061.
21. Tabassi E., Wilson C., Watson C. Fingerprint Image Quality. NIST Internal Report 7151, National Institute for Standards and Technology, 2004. URL: <https://www.nist.gov/publications/fingerprint-image-quality> (accessed: 23.07.2020).
22. Gudkov V.J., Arkabaev D.I. Method for Comparing Fingerprints of Papillary Patterns. Patent RF, no. 2331108, G 06 K 9/62, 2008. Vol. 22. (in Russian)

23. Novikov F.A. Discrete Mathematics for Programmers: Textbook for Higher Schools. St. Petersburg, Piter, 2009. 384 p. (in Russian)
24. Warren H.S. Hacker's Delight, 2nd ed., Addison-Wesley Professional, 2018. 512 p.
25. Dorizzi B., Cappelli R., Ferrara M., et al. Fingerprint and On-Line Signature Verification Competitions at ICB 2009. Proceedings of the International Conference on Biometrics, ICB 2009. 2009. P. 725–732. DOI: 10.1007/978-3-642-01793-3_74.
26. Gudkov V.J. Methods of the First and Second Processing of Fingerprint Images. Miass, Publishing of the Geotour, 2009. 237 p. (in Russian)
27. FVC – ongoing: on-line evaluation of fingerprint recognition algorithms. URL: <https://biolab.csr.unibo.it/FVConGoing/UI/Form/Home.aspx> (accessed: 23.07.2020).