

УДК 004.932

Алгоритм классификации отпечатков пальцев

Ломов Д.С., студент

*Россия, 105005, г. Москва, МГТУ им. Н.Э. Баумана,
кафедра «Программное обеспечение ЭВМ и информационные технологии»*

*Научный руководитель: Деон А.Ф., к.т.н, доцент
Россия, 105005, г. Москва, МГТУ им. Н.Э. Баумана
bauman@bmstu.ru*

Идентификация по отпечаткам пальцев используется во многих сферах деятельности, таких как защита информации, системы управления доступом, медицина, криминалистика и т.д. К настоящему времени разработано много программ данного профиля от систем для небольших компаний до сложных исследовательских программных комплексов. Процесс программной идентификации по отпечаткам пальцев требует больших временных и ресурсных затрат, однако позволяет помочь эксперту в данной области или полностью заменить его. Для ускорения его работы можно использовать различные методы и улучшения, в частности классификацию.

Целью данной работы является описание возможности использования программной оценки в системах обработки дерматоглифической информации с учетом классов отпечатков пальцев, характеризующихся узорами в виде левых петель, правых петель, завитков, дуг и полусфер.

Основными данными, несущими информацию, являются расположение и строение папиллярных линий узора отпечатка.

Процесс классификации отпечатка пальца состоит из двух основных этапов: предварительной обработки изображения и сравнения с базой данных на основе классификатора.

Предварительная обработка изображения

Для корректности проведения эксперимента необходимо произвести предварительную обработку изображения, полученного со сканера отпечатков пальцев. Этот процесс состоит из следующих этапов:

- 1) улучшение качества исходного изображения отпечатка (Нормализация изображения, применение фильтра Габора)
- 2) получение ориентационного изображения и выделение регистрационной точки;
- 3) разбиение изображения на секторы. Получение областной маски изображения.

Далее каждый из этапов будет рассмотрен подробнее.

Нормализация изображения

Пусть $I(i, j)$ - уровень интенсивности пикселя (i, j) монохромного изображения, M и V обозначают оценочное среднее значение и изменение изображения I соответственно, а $G(i, j)$ обозначает нормализованное значение пикселя (i, j) . M и V рассчитываются следующим образом:

$$M = \frac{1}{N_1 \cdot N_2} \sum_{i=1}^{N_1} \sum_{j=1}^{N_2} I(i, j) \quad (1.1)$$

$$V = \frac{1}{N_1 \cdot N_2} \sum_{i=1}^{N_1} \sum_{j=1}^{N_2} (I(i, j) - M)^2 \quad (1.2)$$

Нормализованное изображение определяется как:

$$G(i, j) = \begin{cases} M_0 + \sqrt{\frac{V_0 \times (I(i, j) - M)^2}{V}}, & \text{если } I(i, j) > M \\ M_0 - \sqrt{\frac{V_0 \times (I(i, j) - M)^2}{V}}, & \text{иначе} \end{cases} \quad (1.3)$$

где M_0 и V_0 заданное среднее значение и значение изменения соответственно.[3]

Главное назначение нормализации - уменьшить изменения значений интенсивности вдоль выступов и впадин для облечения последующей обработки изображения. Рисунок 1 показывает пример нормализации изображения.



а

б

Рис. 1. Пример нормализации изображения (а – сканированный отпечаток, б – нормализованное изображение)

Применение фильтров Габора

Ровный симметрический фильтр Габора имеет следующую общую форму в пространственной области:

$$G(x, y; f, \gamma) = \exp \left\{ \frac{-1}{2} \left[\frac{x'^2}{\delta_x} + \frac{y'^2}{\delta_y} \right] \right\} * \cos(2\pi f x'), \quad (1.4)$$

$$x' = x \sin \gamma + y \cos \gamma, \quad (1.5)$$

$$y' = x \cos \gamma - y \sin \gamma, \quad (1.6)$$

где f - частота синусоидальной волны плоскости по направлению γ от оси X , и δ_x , и δ_y определяют Гауссовскую огибающую по осям X и Y соответственно, которые определяют полосу пропускания фильтра Габора.

Изображение отпечатка пальца разделено на четыре составляющих изображения, соответствующие четырем различным значениям $\theta (0^\circ, 45^\circ, 90^\circ, 135^\circ)$ относительно оси X . Изображение отпечатка пальца свернуто с каждым из четырех фильтров Габора, чтобы произвести четыре составляющих изображения.

В результате фильтрации исходного изображения четырьмя фильтрами Габора, различающимися значением параметра θ , получаются четыре изображения, в которых в зависимости от направления выделяются те или иные выступы.

Результат работы фильтра Габора показан на рисунке 2.

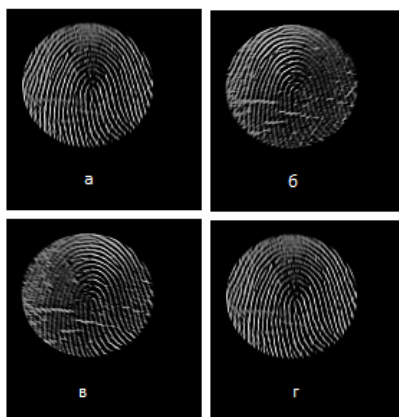


Рис. 2. Изображение после обработки фильтрами Габора (а – $\theta = 0^\circ$, б – $\theta = 45^\circ$, в – $\theta = 90^\circ$, г – $\theta = 135^\circ$)

Получение ориентационного изображения

Основной целью этапа является программное получение ориентации папиллярных линий для возможности последующего определения регистрационной точки.

Ориентационное изображение определяет локальное направление и координаты папиллярных линий. Для представления изображения отпечатка пальца в виде ориентированного изображения можно применить метод оценки ориентации с помощью «метода наименьших квадратов».[3]

Процесс получения ориентационного изображения из нормализованного состоит из следующих этапов:

1. Разбиение нормализованного изображения на блоки размером $W \times W$, где $W = 14$;
2. Вычисление градиентов d_x и d_y в каждом пикселе (i, j) с помощью оператора Собеля;

Оператор Собеля использует матрицы размерностью 3×3 , с помощью которых происходит преобразование изображения для вычисления приближенных значений производных по горизонтали и вертикали. Для вычисления изображений d_x и d_y , используются следующие преобразования:

$$d_y = \begin{pmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{pmatrix} * G \quad (1.7)$$

$$d_x = \begin{pmatrix} -1 & -2 & 1 \\ -2 & 0 & 2 \\ -1 & 2 & 1 \end{pmatrix} * G, \quad (1.8)$$

Далее производится вычисление градиента по формуле:

$$H = \sqrt{d_x^2 + d_y^2} \quad (1.9)$$

3. Оценка локальной ориентации изображения в блоке. Границы блока расположены симметрично относительно пикселя (i, j) . Для вычисления локальной ориентации применяются следующие соотношения:

$$V_x(i, j) = \sum_{u=i-\frac{w}{2}}^{i+\frac{w}{2}} \sum_{v=j-\frac{w}{2}}^{j+\frac{w}{2}} 2\delta_x(u, v)\delta_y(u, v), \quad (1.10)$$

$$V_y(i, j) = \sum_{u=i-\frac{w}{2}}^{i+\frac{w}{2}} \sum_{v=j-\frac{w}{2}}^{j+\frac{w}{2}} \delta_x^2(u, v) - \delta_y^2(u, v), \quad (1.11)$$

$$\theta(i, j) = \frac{1}{2} \tan^{-1} \left(\frac{V_y(i, j)}{V_x(i, j)} \right), \quad (1.12)$$

где $\theta(i, j)$ - оценка методом наименьших квадратов локальной ориентации выступа в блоке в центре с пикселем, имеющим координаты (i, j) .

Для построения локальных ориентационных линий используются следующие тригонометрические преобразования:

$$x_{1,2} = i \mp W \cdot \cos(\theta(i, j)), \quad (1.13)$$

$$y_{1,2} = j \pm W \cdot \sin(\theta(i, j)), \quad (1.14)$$

где x_1, y_1 – начальные координаты ориентационной линии, x_2, y_2 – конечные координаты.



Рис. 3. Результат получения ориентационного изображения (а – Нормализованное изображение, б – Ориентационное изображения)

Нахождение регистрационной точки

Регистрационная точка – центр предполагаемой области распознавания, т.е. области, несущей основную информацию об отпечатке при классификации. Самым простым способом нахождения регистрационной точки может являться поиск наибольшего «излома» папиллярных линий. Он требует высокого качества изображения, однако может повысить скорость нахождения точки.

Разбиение изображения на секторы

Сектором изображения отпечатка пальца, предоставленного для классификации, называется часть изображения, несущая информацию для сравнения с базой данных. Для получения результата необходимо разложить изображение на совокупность секторов S_i , где i -й сектор задан в полярной системе координат (r, θ) следующим образом:

$$S_i = \{(x, y) | b(T_i + 1) \leq r \leq b(T_i + 2), \theta_i \leq \theta \leq \theta_{i+1}, 1 \leq x \leq N, 1 \leq y \leq M\}$$

где

$$T_i = i \operatorname{div} k, \quad (1.19)$$

$$\theta_i = (i \operatorname{mod} k) \left(\frac{2\pi}{k} \right), \quad (1.20)$$

$$r = \sqrt{(x - x_c)^2 + (y - y_c)^2}, \quad (1.21)$$

$$\theta = \tan^{-1} \left(\frac{y - y_c}{x - x_c} \right), \quad (1.22)$$

где (x_c, y_c) — точка центра, b — ширина каждой полосы, и k — число

секторов, которые рассматривают в каждой полосе.[3]

В рамках проведения эксперимента изображение с центром в регистрационной точке делилось на 96 секторов. Каждая полоса шириной $b = 20$ была сегментирована в шестнадцать секторов ($k = 16$).

Получение областной маски

Областная маска представляет собой изображение, разделенное на секторы с вычисленной для каждого сектора средней интенсивностью в полученной области. Принадлежность пикселя к определенному сектору определяется системой:

$$\begin{cases} (x - x_c) \geq \mp \tan(\theta_1) \cdot (y - y_c) \\ (x - x_c) \leq \mp \tan(\theta_2) \cdot (y - y_c) \\ (x - x_c)^2 + (y - y_c)^2 \geq r_1^2 \\ (x - x_c)^2 + (y - y_c)^2 < r_2^2 \end{cases}, \quad (1.23)$$

где (x, y) — координаты проверяемого пикселя, (x_c, y_c) — координаты регистрационной точки, θ_1, θ_2 — начальный и конечный углы сектора, r_1, r_2 — внутренний и внешний радиусы сектора. Далее вычисляется среднюю интенсивность каждого сектора по формуле:

$$I_{mid}(S_k) = \frac{1}{N+M} \sum_{i=1}^N \sum_{j=1}^M I(i, j), \quad (1.24)$$

где S_k – k -й сектор изображения, i, j – относительные координаты пикселей, N, M – количество пикселей в секторе, I_{mid} – средняя интенсивность, $I(i, j)$ – интенсивность пикселя. Результат получения областной маски представлен на рисунке 4:

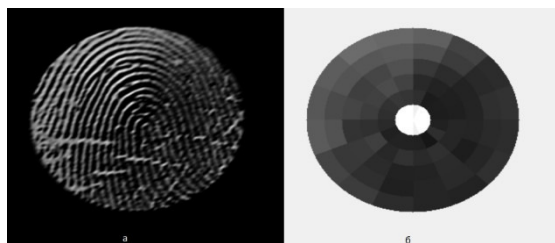


Рис. 4. Результат получения областной маски (а – исходное изображение, полученное с помощью фильтра Габора; б – областная маска изображения)

Работа классификатора

Финальным этапом алгоритма является сравнение полученной области изображения с данными, имеющимися в базе по параметрам интенсивностей секторов. Одним из способов может являться кластеризация методом «К»-ближайших соседей. [2] Принцип метода заключается в том, что вначале находятся «К» ближайших соседей для тестового образца в пространстве признаков отпечатка. После этого тестовый образец относится к классу, который наиболее часто представлен среди «К» ближайших соседей. Алгоритм работы классификатора можно разделить на несколько основных этапов:

1. Для каждого образца из базы данных вычисляется разностный вектор по формуле:

$$V_i = |M(S_{base[k]}) - M(S_{scan[k]})|, \quad (1.25)$$

где V_i – разностный вектор, $M(S_{base[k]})$ – средняя интенсивность сектора изображения из базы данных, $M(S_{scan[k]})$ – средняя интенсивность сектора изображения сканируемого отпечатка пальца.

2. Получение векторного расстояния для каждого образца из БД, которое вычисляется следующим образом:

$$D_m = \sqrt{\sum_{k=1}^{N_s \cdot N_{im}} V_{k.m}^2}, \quad (1.26)$$

где D_m – расстояние для заданного образца, N_s – количество секторов, M_{im} – количество фильтров, V – разностный вектор.

3. Сортировка массива векторных расстояний;
4. Определение веса каждого класса в базе данных. Вес класса – степень принадлежности исходного отпечатка к определенному классу. Наиболее высокий вес соответствует найденному классу обрабатываемого отпечатка. Его определение происходит по формуле:

$$W_i = \sum_{k=1}^{N_c} \frac{1}{D_k}, \quad (1.27)$$

где W_i – вес класса, N_c – количество классов ($N_c = 5$), D_k – векторные расстояния для образцов, принадлежащих данному классу.

5. Класс с наибольшим весом выбирается в качестве наиболее вероятного.

Представленная последовательность действий позволяет реализовать классификацию отпечатков пальцев по типу узоров.

Список литературы

1. Maltoni D. Handbook of fingerprint recognition. New York: Springer, 2003. 496 p.
2. Журавлев Ю.И., Рязанов В.В., Сенько О.В. Распознавание. Математические методы. Программная система. Практические применения. М.: Фазис, 2006. 159 с.
3. Prabhakar S. Fingerprint Classification and Matching Using a Filterbank. East Lansing: Submitted to Michigan State University, 2001. 240 p.
4. Bishop C. Pattern Recognition and Machine Learning. New York: Springer, 2006. 738 p.