

ПОСТРОЕНИЕ МАТЕМАТИЧЕСКОЙ МОДЕЛИ ПРОЦЕССА РАСПОЗНАВАНИЯ ОТПЕЧАТКОВ ПАЛЬЦЕВ ДЛЯ АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ КОНТРОЛЯ ДОСТУПА

Евстюничева А.В., группа АСУ-01а

Руководитель доц. каф. АСУ Привалов М.В.

Отпечатки пальцев являются одними из неизменных на протяжении всей жизни, биометрических характеристик человека. Как следствие — широкое применение систем автоматического распознавания пальцев (от систем управления доступа и паспортного контроля, проведения электронных платежей и аутентификации на Web-ресурсах до проектов гражданской идентификации).

Существуют определенные трудности в создании таких систем. Это точное распознавание изображений отпечатков пальцев. При использовании двухэтапной классификации (отнесение отпечатка пальца сначала в один из шести классов (рис.1) на первом этапе, а после внутриклассовой идентификации на втором этапе) задача сводится не только к правильной идентификации, но и в правильном выборе класса отпечатка пальца.

В данной работе предлагается изменить классы отпечатков пальцев. И вместо шести классов использовать пять, объединяя классы «дуга» и «полусфера» вследствие больших трудностей их различения между собой. При этом следует отказаться от объединения классов «завитушка» и «двойная петля», хотя это часто используется в различных методах, но приводит зачастую к ошибочной классификации двойной петли.

На данный момент наиболее удачным, с высокой точностью классификации является многоканальный подход к распознаванию отпечатков пальцев [1].

Схема алгоритма многоканального подхода представлена на рис. 2.

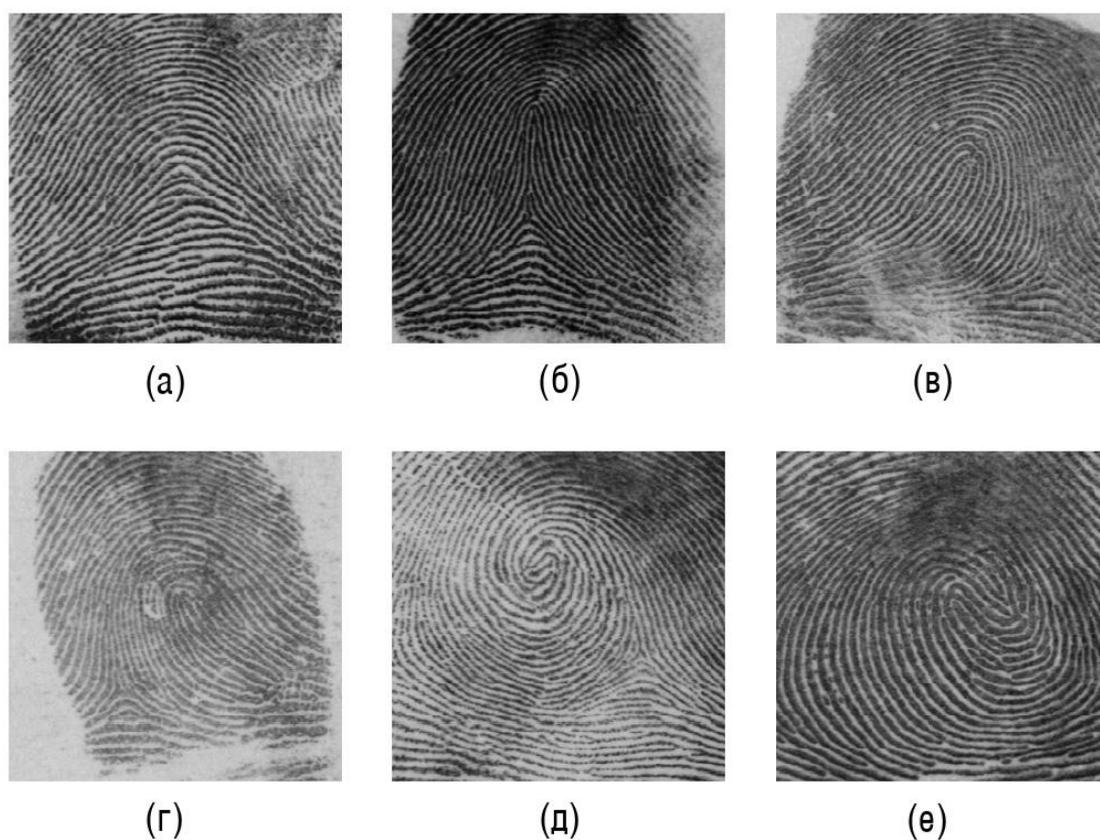


Рисунок 1 — Основные классы отпечатков пальцев:

(а) — дуга (A), (б) — полусфера (Т), (в) — левая петля (L),
 (г) — правая петля (R), (д) — завитушка (W), и
 (е) — двойная петля (W)

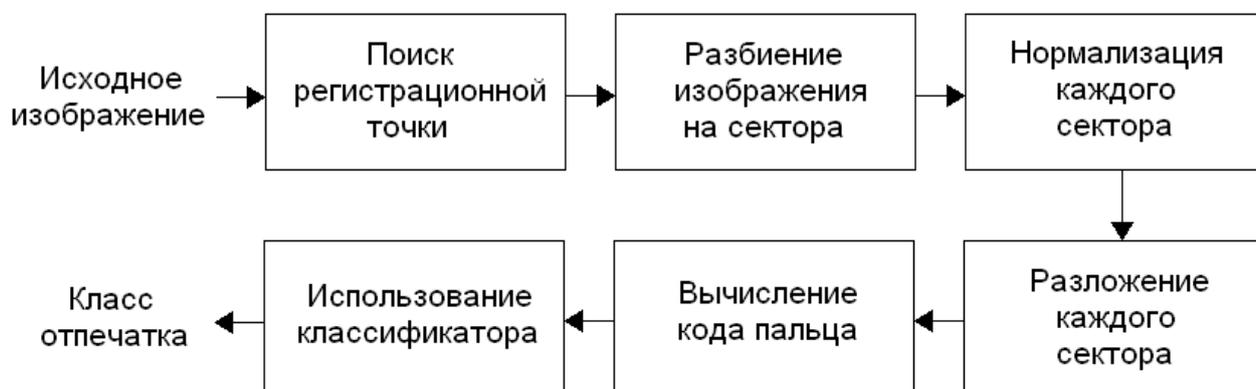


Рисунок 2 — Схема алгоритма многоканального подхода

Для нахождения регистрационной точки используется следующий алгоритм [2]:

Пусть $I(u, v)$ обозначают серый уровень в пикселе (u, v) в $M \times N$ изображении отпечатка пальца. Используем изображения размерностью 500×500 .

- 1) Разделить входное изображение на блоки размером $w \times w$ (16×16).
- 2) Вычислить G_x и G_y градиент каждого пикселя (u, v) .

Компоненты вектора градиента аппроксимируются конечными разностями первого порядка.

3) Вычислить оценку локального размещения каждого блока расположенного симметрично относительно пикселя (i, j) , для чего используются следующие уравнения [3]:

$$\theta(i, j) = \frac{1}{2} \tan^{-1} \left(\frac{\sum_{u=i-\frac{w}{2}}^{i+\frac{w}{2}} \sum_{v=j-\frac{w}{2}}^{j+\frac{w}{2}} (G_x^2(u, v) G_y^2(u, v))}{\sum_{u=i-\frac{w}{2}}^{i+\frac{w}{2}} \sum_{v=j-\frac{w}{2}}^{j+\frac{w}{2}} 2G_x(u, v) G_y(u, v)} \right), \quad (1)$$

где $\theta(i, j)$ — наименьшая квадратная оценка локального выступа в блоке, расположенном симметрично относительно пикселя (i, j) . Математически она представляет направление, которое является ортогональным к доминантному направлению спектра Фурье $w \times w$ окна.

- 4) Сгладить область ориентации в местном соседстве.

Чтобы выполнить низкочастотную фильтрацию, изображение нужно конвертировать в непрерывное векторное поле, которое определяется следующим образом:

$$\Phi_x(i, j) = \cos(2 \theta(i, j)), \quad (2)$$

$$\Phi_y(i, j) = \sin(2 \theta(i, j)), \quad (3)$$

где Φ_x и Φ_y — компоненты x и y векторного поля, соответственно. Затем выполняется свертка с фильтром:

$$\Phi'_x(i, j) = \sum_{u=-q/2}^{q/2} \sum_{v=-q/2}^{q/2} W(u, v) \Phi_x(i-u, j-v), \quad (4)$$

$$\Phi'_y(i, j) = \sum_{u=-q/2}^{q/2} \sum_{v=-q/2}^{q/2} W(u, v) \Phi_y(i-u, j-v), \quad (5)$$

где W — двумерный низкочастотный фильтр с модельным интегралом и размером фильтра $q \times q$. По умолчанию размер фильтра 5×5 , а в качестве фильтра используем Гауссовский фильтр, в котором веса соседних пикселей соответствуют пространственному Гауссовскому распределению:

$$c(r, r_0) = 4e^{-\left(\frac{|r-r_0|}{t}\right)^2}, \quad (6)$$

где r_0 — центральная точка (та, для которой рассчитывается новое значение); r — текущая точка; $c(r, r_0)$ — вес, придаваемый точке r при расчете значения точки r_0 ; $|r-r_0|$ — расстояние от точки r до точки r_0 ; t — ширина Гауссовского распределения.

Таким образом, сглаживающий оператор выполняется на блоковом уровне.

5) Вычислить локальное рубцовое размещение на (i, j) :

$$O(i, j) = \frac{1}{2} \tan^{-1} \left(\frac{\Phi'_y(i, j)}{\Phi'_x(i, j)} \right), \quad (7)$$

Таким образом, может быть получена довольно ровная (сглаженная) размещенная полевая оценка.

6) Инициализировать изображение меток A , где будет указана точка ядра.

7) Для каждого пикселя (i, j) в O вычислить индекс Пуанкаре [4], и назначить соответствующие пиксели в изображении меток A значением одного из индекса Пуанкаре — $(1/2)$. Индекс Пуанкаре в пикселе (i, j) , приложенный к

пальцевидной кривой, которая состоит из последовательности пикселей, которые на расстоянии одного пикселя от соответствующей кривой, вычислен следующим образом:

$$\text{Пуанкаре } (i, j) = \frac{1}{2\pi} \sum_{k=0}^{N_{\psi}-1} \Delta(k), \quad (8)$$

$$\Delta(k) = \begin{cases} \delta(k), & \text{если } |\delta(k)| < \pi/2 \\ \pi + \delta(k), & \text{если } \delta(k) \leq -\pi/2 \\ \pi - \delta(k), & \text{иначе,} \end{cases} \quad (9)$$

$$\delta(k) = O(\psi_x(k'), \psi_y(k')) - O(\psi_x(k), \psi_y(k)), \quad (10)$$

$$k' = (k+1) \bmod N_{\psi}, \quad (11)$$

где $\psi_x(\cdot)$ и $\psi_y(\cdot)$ — координаты x и y закрытой пальцевидной кривой с N_{ψ} пикселями.

8) Найти связанные компоненты в A [5]. Если область связанного компонента больше семи, ядро обнаружено в средней точке связанного компонента. Если область связанного компонента больше 20-ти, два ядра обнаружены в средней точке связанного компонента.

9) Если обнаружено больше, чем два ядра, возвратитесь к Шагу 4).

10) Если два ядра обнаружены, центру назначаются координаты основной точки с нижним значением y (верхнее ядро). Если только одно ядро обнаружено, центр имеет координаты основного центра этого ядра.

11) Если ни одна основная точка не найдена, вычислите матрицу ковариации векторной области в местном соседстве ($w \times w$) каждой точки в области ориентации. Определите изображение признаков I с наибольшим собственным значением матрицы ковариации для каждого элемента в изображении ориентации. Ядро найдено в средней точке наибольшего связанного компонента в пороговом изображении F и центр назначен с координатами этого ядра.

Матрица ковариации вычисляется следующим образом:

$$M(i, j) = \frac{\sum_{x=1, y=1}^{i, j} (x - x_c)(y - y_c)^T}{w^2}. \quad (12)$$

Найденный центр перемещен на 40 пикселей вниз для дальнейшей обработки; это связано с тем, что большинство информации об отпечатке пальца находится в нижней части отпечатка пальца. Это значение было определено опытным путем.

Далее раскладываем изображение на совокупность секторов S_i , где i -сектор S_i вычислен в терминах параметров (r, θ) следующим образом:

$$S_i = \{(x, y) | b(T_i + 1) \leq r < b(T_i + 2), \theta_i \leq \theta < \theta_{i+1}, 1 \leq x \leq N, 1 \leq y \leq M\}, \quad (13)$$

где

$$T_i = i \operatorname{div} k, \quad (14)$$

$$\theta_i = (i \operatorname{mod} k) \left(\frac{2\pi}{k} \right), \quad (15)$$

$$r = \sqrt{(x - x_c)^2 + (y - y_c)^2}, \quad (16)$$

$$\theta = \tan^{-1} \left(\frac{y - y_c}{x - x_c} \right), \quad (17)$$

где (x_c, y_c) — точка центра, b — ширина каждой полосы, и k — число секторов, которые рассматривают в каждой полосе. Мы используем шесть концентрических полос вокруг найденного центра. Каждая полоса с 20 пикселями шириной ($b=20$) сегментирована в восемь секторов ($k=8$). Самая внутренняя полоса не используется, так как содержит мало пикселей. Таким образом, в общей сложности имеем $8 \times 6 = 48$ секторов (с S_0 до S_{47}).

Перед разложением изображения отпечатка пальца, мы нормализуем область в каждом секторе отдельно к постоянному среднему и разнице. Нормализация сделана, чтобы удалить эффекты шумов датчика и различных давления пальца. Для всех пикселей в секторе S_i , нормализованное изображение определено как:

$$N_i(x, y) = \begin{cases} M_o + \sqrt{\frac{(v_o) \times (I(x, y) - M_i)^2}{v_i}}, & \text{если } I(x, y) > M_i \\ M_o - \sqrt{\frac{(v_o) \times (I(x, y) - M_i)^2}{v_i}}, & \text{иначе} \end{cases} \quad (18)$$

где M_o и V_o — требуемое среднее и значение разницы, соответственно.

К полученному изображению применяем ровный симметрический фильтр Габора, который имеет следующую общую форму:

$$G(x, y; f, \gamma) = \exp\left\{\frac{-1}{2}\left[\frac{x'^2}{\delta_x^2} + \frac{y'^2}{\delta_y^2}\right]\right\} \cdot \cos(2\pi f x'), \quad (19)$$

$$x' = x \cdot \sin\gamma + y \cdot \cos\gamma, \quad (20)$$

$$y' = x \cdot \cos\gamma - y \cdot \sin\gamma, \quad (21)$$

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.

В нашем алгоритме, частота фильтра f — это средняя частота выступов ($1/K$), где K — расстояние между выступами. Среднее расстояние между выступами — приблизительно 10 пикселей в изображении отпечатка пальца 500 точек на дюйм. Опытным путем определены значения δ_x и δ_y и оба берутся равными 4.0.

Пусть $C_{i\varphi}(x, y)$ будет составляющим изображением, соответствующим φ для сектора S_i . Для i ($i=[0, 1, \dots, 47]$) и $\varphi \in [0^0, 45^0, 90^0, 135^0]$ стандартное отклонение $F_{i\varphi}$, определенное как:

$$F_{i\varphi} = \sqrt{\sum_{K_i} (C_{i\varphi}(x, y) - M_{i\varphi})^2}, \quad (22)$$

где K_i — число пикселей в S_i , и $M_{i\varphi}$ — средние значения пикселя в $C_{i\varphi}(x, y)$.

Далее этот код пальца подается на классификатор для дальнейшей классификации. В качестве классификатора можно выбрать один из следующих:

- 1) Классификатор нейронная сеть;
- 2) Классификатор К-ближайших соседей;
- 3) Евклидовы расстояния.

Результатом классификации на данном этапе является отнесение отпечатка пальца в один из пяти указанных классов.

Следующим этапом является сопоставление отпечатка с подмножеством отпечатков пальцев в выбранном классе. В качестве алгоритма сопоставления можно выбрать один из следующих классификаторов:

- Классификатор Нейронная сеть [1];
- Классификатор К-ближайших соседей [1];
- Классификатор «отклонение выбора» [1];
- Классификатор скрытой модели Маркова [6];
- Классификатор дерева решения [6];
- Комбинированный классификатор [6].

Таким образом, построена математическая модель, позволяющая описать процесс распознавания пальцев в автоматизированной системе. Направлением дальнейших исследований является экспериментальное исследование построенной модели с целью определения набора признаков и классификатора, обеспечивающих наиболее высокую точность классификации.

Перечень ссылок

1. Anil Jain, Salil Prabhakar, and Lin Hong. A Multichannel Approach to Fingerprint Classification. *IEEE Trans. on PAMI*, Vol. 21, No. 4, pp. 348–359, Apr. 1999.
2. Lin Hong, Yifei Wan, and Anil Jain. Fingerprint Image Enhancement: Algorithm and Performance Evaluation. *IEEE Trans. on PAMI*, Vol. 20, No. 8, pp. 777–789, Aug. 1998.
3. Anil Jain, Lin Hong, and Ruud Bolle. On-Line Fingerprint Verification. *IEEE Trans. on PAMI*, Vol. 19, No. 4, pp. 302–314, Apr. 1997.
4. Davide Maltoni, Dario Maio, Anil K. Jain, Salil Prabhakar. *Handbook of Fingerprint Recognition*: Springer-Verlag New York, Inc., 2003. — P. 348.
5. Александр Вежнев. Выделение связанных областей в цветных и полутоновых изображениях / Электронный ресурс. Способ доступа: URL: www.ict.edu.ru/ft/002397/num4confld.pdf.
6. Andrew Senior. A Combination Fingerprint Classifier. *IEEE Trans. on PAMI*, Vol. 23, No. 10, pp. 1165–1174, Oct. 2001.