

Определение видимой области радужки классификатором локальных текстурных признаков*

И. А. Соломатин, И. А. Матвеев

ivan.solomatin@phystech.edu

Московский физико-технический институт, Россия, г. Долгопрудный, Институтский пер., 9
Вычислительный центр РАН им. А.А.Дородницына, Россия, г. Москва, ул. Вавилова, 40

Распознавание человека по изображению радужной оболочки — актуальная задача в биометрических системах. Помимо выделения радужки как кольцевой области для повышения точности распознавания определяют области затенения (блики, веки, ресницы и т. д.). Задача выделения затенений радужки может быть поставлена как классификация пикселей кольцевой области на два класса: «радужка» и «затенение». В кольцевой области определяется сектор с минимальной дисперсией яркости, который, как правило, не содержит затенений (в данной работе этот сектор не вычисляется, а рассматривается как часть входных данных алгоритма). Далее строится классификатор на основе многомерного гауссиана, который обучается на выборке, задаваемой по пикселям этого сектора. Параметры классификатора были оптимизированы с помощью генетического алгоритма. Проблема шума и ошибок с классификацией некоторых участков изображения решается с помощью применения морфологической постобработки. Был проведен вычислительный эксперимент, и получено распределение функционала качества алгоритма.

Ключевые слова: *обработка изображений; распознавание образов; биометрическая идентификация по радужке*

DOI: 10.21469/22233792.1.14.03

1 Введение

Необходимый этап биометрической идентификации по радужной оболочке — сегментация радужки, т. е. выделение области изображения, содержащей только радужку. В сегментацию помимо выделения окружностей, аппроксимирующих внешнюю и внутреннюю границы радужной оболочки, входит выделение областей затенения. Последняя задача достаточно важна для распознавания человека по радужке, так как ее решение задает конечную эффективную область для распознавания. В англоязычной литературе эта задача называется EES (Eyelids, Eyelashes, Shadows) localization. Существуют различные подходы к ее решению. В [1] осуществляется выделение области век с использованием интегрально-дифференциальных операторов, граница века аппроксимируется некоторой кривой. В [2] для локализации век используется преобразование Хафа. В [3] граница века аппроксимируется параболой. В [4] для обнаружения затенений используется мера качества радужки, вводится вероятностная мера качества радужки, основанная на смеси многомерных гауссианов, которая сравнивается с мерой, основанной на преобразовании Фурье. Для распознавания ресниц в [5] применяется статистическая оценка. В [6] описан метод распознавания век, использующий горизонтальный медианный фильтр и распознавание границ. В [7] для распознавания ресниц используется оператор Собеля, затем ресницы удаляются с изображения с помощью медианного фильтра. Существуют решения, выделяющие все затенения сразу (без разделения на веки, ресницы, блики и т. п.). Одно из таких решений описано в [8], в нем применяется классификатор, основанный на смесях

*Работа выполнена при финансовой поддержке РФФИ, проект № 15-01-05552.

гауссианов, разделяющий точки на два класса: «радужка» и «затенение». Для обучения классификатора используются изображения, размеченные вручную.

Недостатками метода [8] являются необходимость в ручной разметке и использование одного и того же классификатора для всех изображений. Единый классификатор для всех изображений является сомнительным решением вследствие того, что изображения разных радужных оболочек могут иметь значительные вариации текстурных признаков (разная структура, разное освещение и т. д.), что при обучении на большой выборке порождает очень широкий класс «радужка», обладающий малой разделяющей способностью, а при обучении на малой выборке — узкий класс, не пригодный для многих изображений.

В данной работе предлагается использовать классификатор на основе многомерного гауссиана, обучающийся на незатененном секторе S . Использование сектора S для обучения позволяет избежать использования обучающей выборки, размеченной вручную, т. е. построить полностью автоматический метод. Таким образом, содержание работы — реализация алгоритма поиска областей затенения с использованием сектора S вместо ручной разметки. В качестве сектора S используется сектор с минимальной дисперсией яркости с фиксированным углом раствора. Важно заметить, что в данной работе прежде всего ставился вопрос о принципиальной работоспособности метода распознавания затенений на основе классификатора, обучающегося на элементе того же изображения. Вопрос максимальной оптимизации пока не ставился. В дальнейшем, в частности, будет подробнее изучен вопрос нахождения сектора S и вопрос выбора признаков, по которым строится классификатор.

2 Постановка задачи

На вход подается черно-белое изображение I , которое является прямоугольной матрицей $W \times H$ из целых беззнаковых однобайтовых чисел $I(i, j) \in [0; 255]$, задающих яркость каждого пикселя изображения. Также на вход подаются координаты центров и радиусы двух окружностей, аппроксимирующих внешнюю и внутреннюю границу радужки (вычисленные, например, методами [9]), и незатененный сектор S , задаваемый серединным углом α и раствором $\Delta\alpha$.

Требуется выделить области затенения, т. е. области изображения, на которых изображена не радужка, а веко, ресница, блик и т. п. Строго говоря, нужно классифицировать все точки изображения, лежащие внутри кольца, аппроксимирующего радужку по двум классам — «радужка» и «затенение», т. е. получить бинарное изображение J :

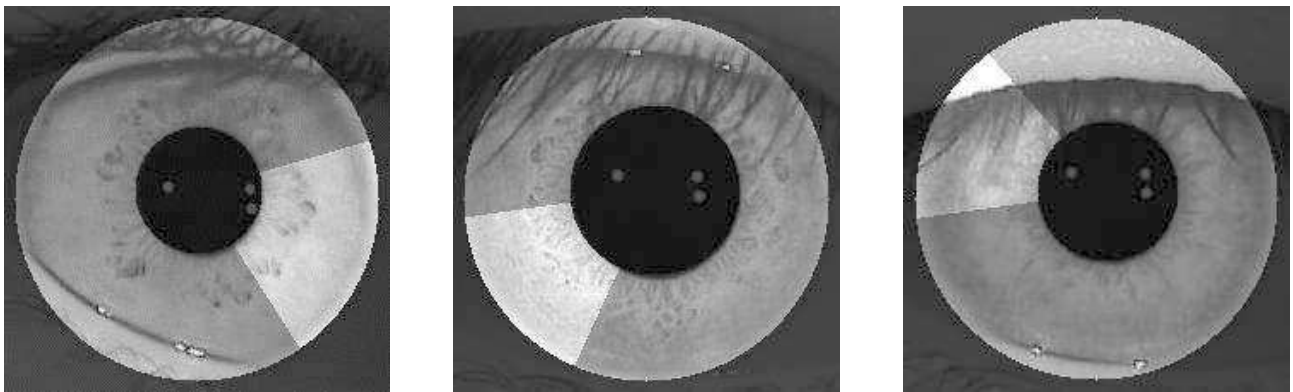
$$J(i, j) \in \{0, 1\}; \quad i \in [1; H]; \quad j \in [1, W], \quad (1)$$

где $J(i, j) = 1 \Leftrightarrow$ точка $(i, j)^T$ классифицирована как затенение.

3 Метод решения

Рассмотрим точки, находящиеся внутри сектора $S = \{\mathbf{x}_n\}_{n=1}^N$, где N — число точек, принадлежащих S . Локальные текстурные признаки этих точек будут являться обучающей выборкой для классификатора. На рис. 1 приведены примеры сектора S для различных изображений радужки.

Стоит заметить, что в качестве сектора S выбирается сектор с фиксированным раствором $\Delta\alpha$ такой, что его дисперсия минимальна, и, строго говоря, этот сектор может содержать затенения. В таких случаях точность распознавания не может быть высокой. Пример того, как наличие затенений в секторе S влияет на точность распознавания, рассмотрен в разд. 5 (пример 3, рис. 8). В данном примере сектор S содержит ресницы; таким



(а) S не содержит затенений

(б) S не содержит затенений

(в) S содержит затенения

Рис. 1 Примеры секторов с минимальной дисперсией яркости (выделены светлым)

образом, классификатор, обученный по этому сектору, не считает ресницы затенениями, что уменьшает точность. В дальнейшем планируется написать более точный алгоритм поиска незатененного сектора радужки.

В качестве локальных текстурных признаков выбираются следующие $K = 8$ признаков:

1. $B(\mathbf{x}_n)$ — яркость в точке $\mathbf{x}_n = (x, y)^T$:

$$B(\mathbf{x}_n) = I(x, y).$$

2. $\bar{B}(\mathbf{x}_n)$ — средняя яркость в окрестности точки \mathbf{x}_n (в окне размера a).

В качестве окна размера a будем использовать квадрат со стороной $2a + 1$. Тогда количество точек в окне размера a равно $(2a + 1)^2$.

Обозначим момент яркости в окрестности размера a точки \mathbf{x}_n :

$$M_a^{(z)} = \frac{1}{(2a + 1)^2} \sum_{i=-a}^a \sum_{j=-a}^a I^z(x + i, y + j).$$

Тогда

$$\bar{B}(\mathbf{x}_n) = M_a^{(1)}. \tag{2}$$

3. $\sigma(\mathbf{x}_n)$ — среднеквадратичное отклонение яркости в окрестности точки \mathbf{x}_n (в окне размера b):

$$\sigma(\mathbf{x}_n) = \sqrt{M_b^{(2)} - \left(M_b^{(1)}\right)^2}. \tag{3}$$

4. $\mathbf{C}(\mathbf{x}_n)$ — вектор из пяти компонент дискретного косинусного преобразования в окрестности точки \mathbf{x}_n .

Как правило, дискретное косинусное преобразование считается в окне $w \times h$, где $w = h = 8$, но так как в данном случае, из соображений симметрии, желательно, чтобы окно было симметрично центрированным, используется окно 7×7 ($w = h = 7$). Двумерное дискретное косинусное преобразование в окне $w \times h$ при $w = h = 7$ с центром в точке $(x, y)^T$:

$$\hat{C}_{p,q}(x, y) = \alpha_p \alpha_q \sum_{i=-3}^3 \sum_{j=-3}^3 I(x + i, y + j) \cos \frac{\pi(2i + 7)p}{2h} \cos \frac{\pi(2j + 7)q}{2w},$$

$$0 \leq p \leq h - 1; \quad 0 \leq q \leq w - 1,$$

где

$$\alpha_p = \begin{cases} \frac{1}{\sqrt{w}}, p = 0; \\ \frac{\sqrt{2}}{\sqrt{w}}, 1 \leq p \leq h - 1; \end{cases} \quad \alpha_q = \begin{cases} \frac{1}{\sqrt{h}}, q = 0; \\ \frac{\sqrt{2}}{\sqrt{h}}, 1 \leq q \leq w - 1. \end{cases}$$

Из полученной матрицы коэффициентов 7×7 берутся первые 5 коэффициентов за исключением $\hat{C}_{0,0}$ (рис. 2), так как коэффициент $\hat{C}_{0,0}$ равен средней яркости в окрестности, которая уже используется в качестве признака:

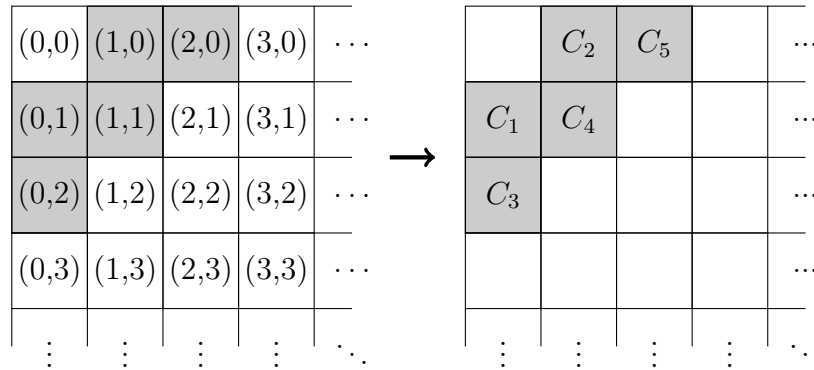


Рис. 2 Используемые коэффициенты дискретного косинусного преобразования

$$C(x_n) = (\hat{C}_{0,1}, \hat{C}_{1,0}, \hat{C}_{0,2}, \hat{C}_{1,1}, \hat{C}_{2,0})^T.$$

В формулах (2) и (3) размеры окрестностей a и b выступают в качестве параметров алгоритма.

Каждой точке сектора $S : \mathbf{x}_n = (x, y)^T$ сопоставляем вектор из признаков:

$$\mathbf{p}_n = (B(\mathbf{x}_n), \bar{B}(\mathbf{x}_n), \sigma(\mathbf{x}_n), (C(\mathbf{x}_n))^T)^T.$$

Находим средний вектор:

$$\boldsymbol{\mu} = \frac{1}{N} \sum_{n=1}^N \mathbf{p}_n; \quad \hat{\mathbf{p}}_n = \mathbf{p}_n - \boldsymbol{\mu}.$$

Из векторов $\hat{\mathbf{p}}_n$ составляется матрица объект-признак M размерности $K \times N$:

$$M = (\hat{\mathbf{p}}_1 \quad \hat{\mathbf{p}}_2 \quad \cdots \quad \hat{\mathbf{p}}_n).$$

Предполагаем, что объекты в пространстве, задаваемом данными параметрами, распределены по многомерному гауссиану. Гипотезу о принадлежности каждого пикселя классу «радужка» принимаем или отвергаем, основываясь на расстоянии Махаланобиса.

Матрица ковариаций, нормированная на средний вектор, задается следующей формулой:

$$C = MM^T.$$

Расстояние Махаланобиса от \mathbf{x} до $\boldsymbol{\mu}$:

$$D(\mathbf{x}) = \sqrt{(\mathbf{x} - \boldsymbol{\mu})^T C^{-1} (\mathbf{x} - \boldsymbol{\mu})}.$$

Считаем, что пиксель \mathbf{x} лежит в классе «радужка», если

$$\exp(-D^2(\mathbf{x})) > P_{\text{порог}}. \tag{4}$$

Таким образом, алгоритм задается следующими параметрами: $a, b, P_{\text{порог}}$, где a и b — размеры окон (окрестностей) для расчета 1-го и 2-го признаков (2) и (3), а $P_{\text{порог}}$ — пороговое значение в формуле (4). Для определения оптимальных значений указанных параметров был использован генетический алгоритм.

4 Подбор параметров

Важно заметить, что применение генетического алгоритма для подбора параметров является не обучением, а настройкой алгоритма, т.е. параметры выбираются единожды и фиксируются.

Для применения генетического алгоритма нужно ввести функцию ошибки алгоритма для оценки качества решения для конкретных параметров. Пусть результат работы алгоритма равен J , а экспертная разметка областей затенения равна \hat{J} , где J и \hat{J} — бинарные изображения вида (1).

Пусть N_0 — общее количество классифицируемых точек, лежащих внутри заданной границы радужки, а \hat{N}_0 — количество точек, классифицируемых как затенения в экспертной разметке. Обозначим через $\delta_{\text{Iris}}(i, j)$ индикаторную функцию множества точек, лежащих внутри кольца, аппроксимирующего область радужки. Тогда

$$N_0 = |\{(i, j) \mid \delta_{\text{Iris}}(i, j) = 1\}|; \quad \hat{N}_0 = |\{(i, j) \mid \delta_{\text{Iris}}(i, j)\hat{J}(i, j) = 1\}|.$$

Тогда относительная ошибка первого рода равна

$$E_1 = \frac{1}{\hat{N}_0} \sum_{i=1}^H \sum_{j=1}^W \hat{J}(i, j)(1 - J(i, j))\delta_{\text{Iris}}(i, j),$$

а относительная ошибка второго рода равна

$$E_2 = \frac{1}{N_0 - \hat{N}_0} \sum_{i=1}^H \sum_{j=1}^W J(i, j)(1 - \hat{J}(i, j))\delta_{\text{Iris}}(i, j).$$

Функция ошибки определяется как сумма относительных ошибок первого и второго рода:

$$E = E_1 + E_2.$$

В данном генетическом алгоритме особь — это вектор, состоящий из трех параметров алгоритма: $v = (a, b, P_{\text{порог}})^T$. Оценка качества особи производилась на наборе из $M = 10$ изображений с экспертной разметкой затенений: $\mathcal{I} = \{I_m\}_{m=1}^M$, т.е. функция качества особи определялась как средняя точность алгоритма на изображениях из \mathcal{I} :

$$f(v) = f \left(\begin{pmatrix} a \\ b \\ P_{\text{порог}} \end{pmatrix} \right) = \frac{1}{M} \sum_{m=1}^M (1 - E(I_m, a, b, P_{\text{порог}})),$$

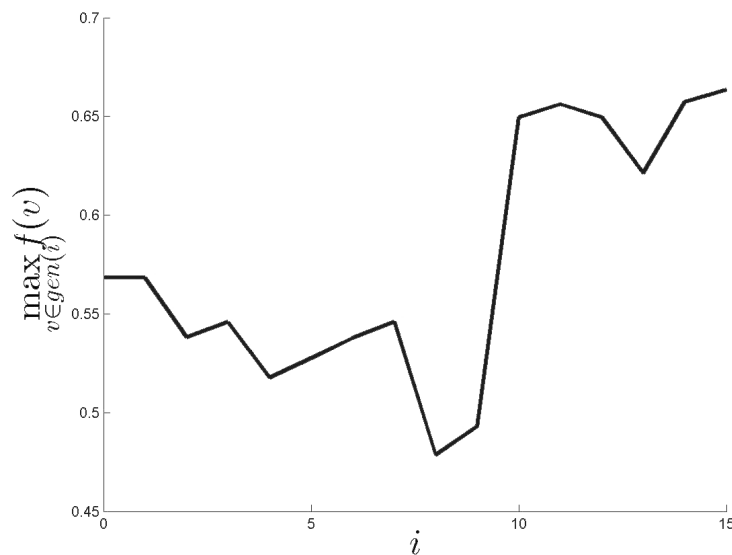


Рис. 3 Зависимость максимальной функции качества в популяции от номера поколения

где $E(I, a, b, P)$ — значение функции ошибки алгоритма с параметрами (a, b, P) на изображении I . Селекция выполнялась методом рулетки, т.е. вероятность выбора особи v в качестве родителя пропорциональна $f(v)$.

В генетическом алгоритме использовались следующие генетические операторы:

- скрещивание:

$$v \otimes w = \begin{pmatrix} a_1 \\ b_1 \\ P_1 \end{pmatrix} \otimes \begin{pmatrix} a_2 \\ b_2 \\ P_2 \end{pmatrix} = \begin{pmatrix} r(a_1, a_2) \\ r(b_1, b_2) \\ r(P_1, P_2) \end{pmatrix},$$

где $r(x, y)$ — случайная величина (распределенная по Бернулли):

$$r(x, y) = \begin{cases} x & \text{с вероятностью } p = \frac{1}{3}; \\ y & \text{с вероятностью } p = \frac{1}{3}; \\ \frac{x+y}{2} & \text{с вероятностью } p = \frac{1}{3}; \end{cases}$$

- мутация:

$$M(v) = M \begin{pmatrix} a \\ b \\ P \end{pmatrix} = \begin{cases} v & \text{с вероятностью } p = \frac{2}{3}; \\ (r_1, b, P)^T & \text{с вероятностью } p = \frac{1}{9}; \\ (a, r_1, P)^T & \text{с вероятностью } p = \frac{1}{9}; \\ (a, b, r_2)^T & \text{с вероятностью } p = \frac{1}{9}, \end{cases}$$

где r_1 — дискретная случайная величина, равномерно распределенная на $[1, 15]$; r_2 — непрерывная случайная величина, равномерно распределенная на $[0,6, 1]$. Было проведено вычисление 15 поколений при размере популяции в 10 особей. На рис. 3 приведен

график зависимости максимальной функции качества в популяции от номера поколения в процессе выполнения генетического алгоритма. Лучшей особью в начальной популяции является вектор $v_0 = (5, 5, 0,7)^T$.

В результате работы генетического алгоритма получены следующие параметры:

$$v^* = \begin{pmatrix} a \\ b \\ P \end{pmatrix} = \begin{pmatrix} 10 \\ 7 \\ 0,85 \end{pmatrix}; \quad f(v^*) = 0,664.$$

Строго говоря, утверждать, что данные параметры являются оптимальными, нельзя, однако на рис. 3 видно, что $f(v^*)$ больше, чем $f(v_0)$, т. е. подбор параметров генетическим алгоритмом улучшил результат. Это также видно на гистограммах на рис. 5 (см. разд. 5).

5 Вычислительный эксперимент

Метод был реализован в системе Matlab, код реализации находится в общем доступе [10]. К сожалению, возможности сравнить результаты с результатами других методов пока нет, ввиду отсутствия размеченных баз, на которых получены результаты других методов. Данная реализация метода была протестирована на базе изображений CASIA [11] с экспертной разметкой границ радужки, незатененного сектора и областей затенения (для оценки ошибки). Алгоритм запускался с параметрами: $v_0 = (5, 5, 0,7)^T$ и v^* .

Примеры вывода алгоритмов с параметрами v_0 и v^* на одном и том же изображении приведены на рис. 4. На рис. 4 показаны две проблемы алгоритма:

- 1) на изображениях виден шум (некоторые отдельные точки радужки классифицируются как затенения и наоборот);
- 2) существует проблема с распознаванием век — часть верхнего века мало отличается от радужки.

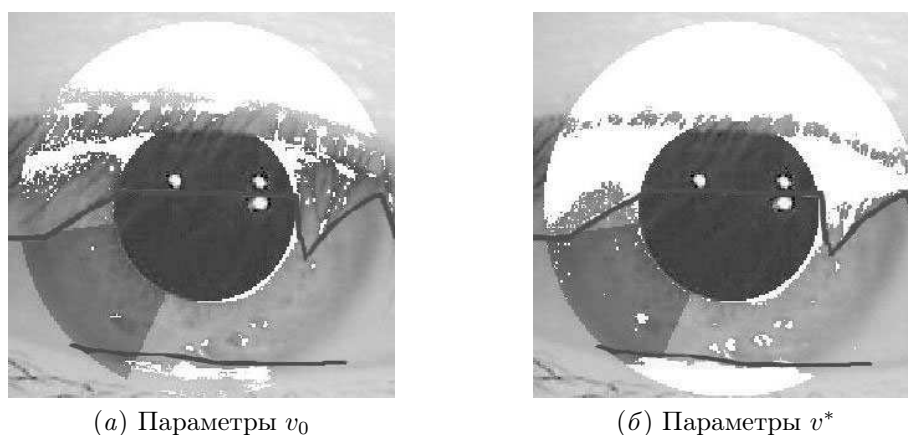


Рис. 4 Результат работы алгоритма на различных наборах параметров

Для компенсации данных недостатков алгоритма применялась морфологическая пост-обработка. К логическому изображению вывода алгоритма применялись фильтры замыкания и размыкания. Фильтры считались в окнах 7×7 . Точность алгоритма увеличилась после применения фильтра, это видно на рис. 5–8.

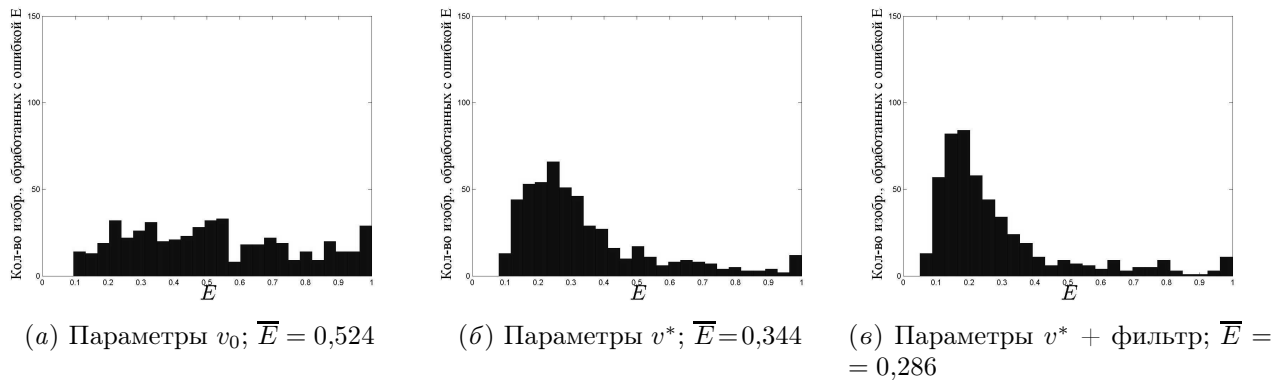


Рис. 5 Гистограммы ошибок для различных модификаций алгоритма

На рис. 5 представлены гистограммы распределения функционалов качества E для каждой из исследуемых модификаций алгоритмов на 500 изображениях из базы CASIA [11].

Стоит также заметить, что в экспертной разметке затенений не учитываются блики, в то время как данный алгоритм классифицирует блики как затенения. На некоторых изображениях это приводит к заведомо заниженной оценке точности распознавания.

Далее приведены три примера работы алгоритма с параметрами v^* и применением фильтра.

Пример 1. На рис. 6 показаны результаты работы алгоритмов на файле 2003R05.bmp из базы CASIA, изображенном на рис. 6, а. Результат до применения фильтра изображен на рис. 6, б. Результат после применения фильтра изображен на рис. 6, в. Время работы: $t = 12,7$ с. Функция ошибки: $E = 0,102$.

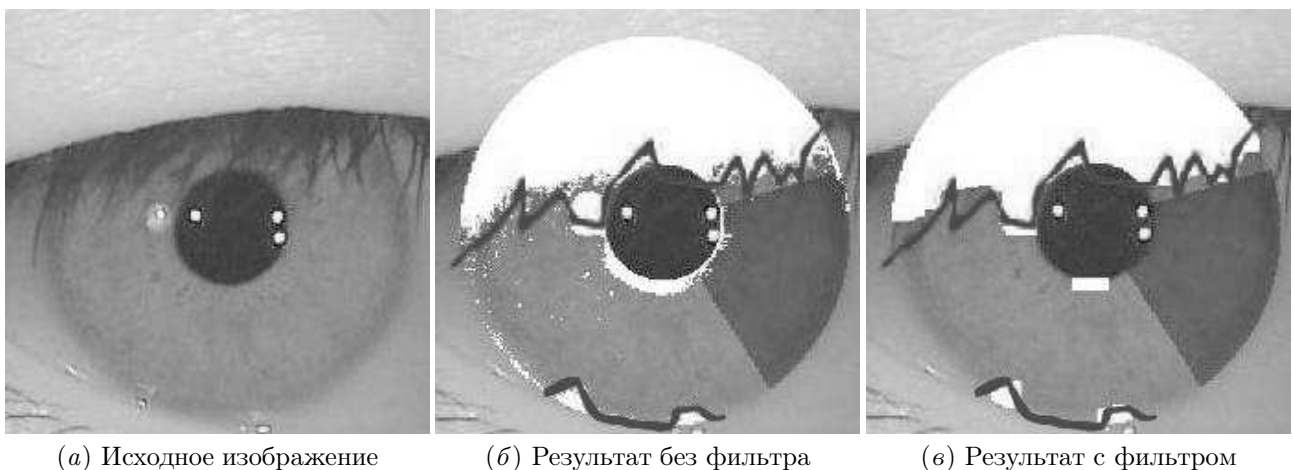


Рис. 6 Результаты работы на изображении 2003R05.bmp из базы CASIA [11]

Пример 2. На рис. 7 показаны результаты работы алгоритмов на файле 2007R01.bmp из базы CASIA, изображенном на рис. 7, а. Результат до применения фильтра изображен на рис. 7, б. Результат после применения фильтра изображен на рис. 7, в. Время работы: $t = 10,3$ с. Функция ошибки: $E = 0,156$.

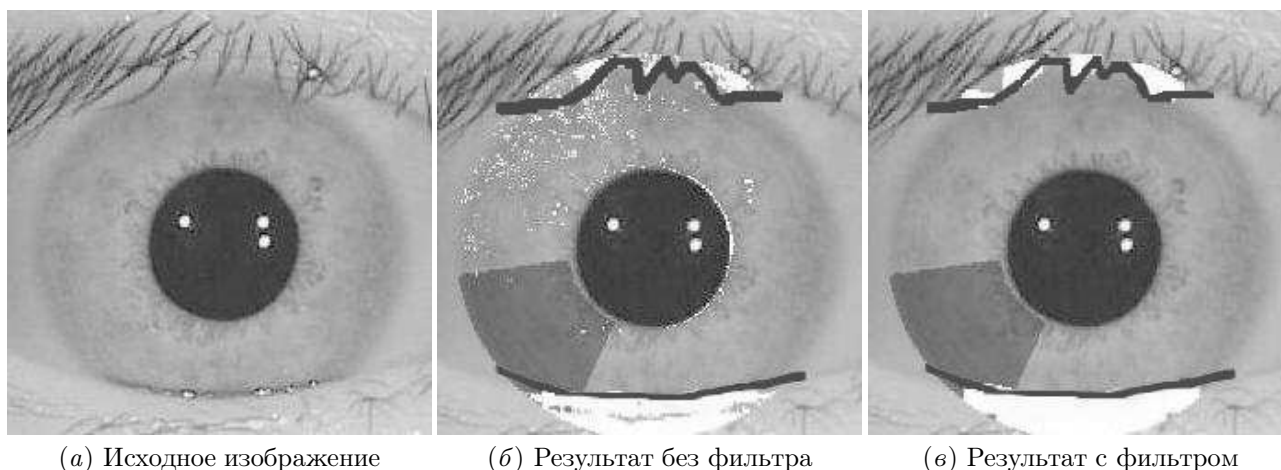


Рис. 7 Результаты работы на изображении 2007R01.bmp из базы CASIA [11]

Пример 3. На рис. 8 показаны результаты работы алгоритмов на файле 2015R11.bmp из базы CASIA, изображенном на рис. 8, а. Результат до применения фильтра изображен на рис. 8, б. Результат после применения фильтра изображен на рис. 8, в. Время работы: $t = 10,7$ с. Функция ошибки: $E = 0,484$. В этом примере сектор S содержит затенения и, как следствие, алгоритм не распознал ресницы, однако тем не менее распознал веки.

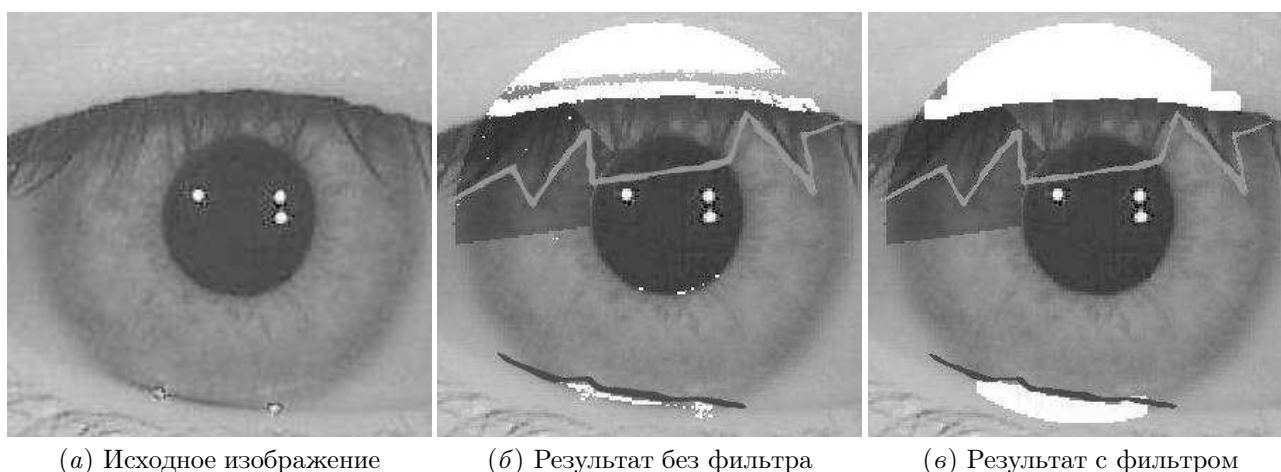


Рис. 8 Результаты работы на изображении 2015R11.bmp из базы CASIA [11]

6 Заключение

Представлен метод, позволяющий с высокой точностью выделять области затенения радужки. Основное преимущество метода состоит в том, что он является полностью автоматическим и не требует ручной разметки для обучающей выборки. Также отличительной чертой является тот факт, что метод строит новый классификатор для каждого изображения, что обеспечивает стабильность работы на изображениях с разной структурой радужки и с разной освещенностью. Скорость работы метода невелика (среднее время работы на одном изображении на выборке CASIA [11] составляет 10 с), однако это время можно уменьшить, используя более быстрые языки программирования.

Литература

- [1] *Daugman J.* How iris recognition works // Conference (International) on Image Processing Proceedings. IEEE, 2002. Vol. 1. P. 33–36. doi: <http://dx.doi.org/10.1109/ICIP.2002.1037952>
- [2] *Wildes R. P.* Iris recognition: An emerging biometric technology // Proc. IEEE, 1997. Vol. 85. No. 9. P. 1348–1363. doi: <http://dx.doi.org/10.1109/5.628669>
- [3] *He Z. F., Tan T. N., Sun Z. A., Qiu X. C.* Robust eyelid, eyelash and shadow localization for iris recognition // 15th IEEE Conference (International) on Image Processing Proceedings. IEEE, 2008. P. 265–268.
- [4] *Krichen E., Garcia-Salicetti S., Dorizzi B.* A new probabilistic iris quality measure for comprehensive noise detection // 1st IEEE Conference (International) on Biometrics: Theory, Applications, and Systems. IEEE, 2007. P. 1–6. doi: <http://dx.doi.org/10.1109/BTAS.2007.4401906>
- [5] *Daugman J.* New methods in iris recognition // IEEE Trans. Syst. Man Cyb., 2007. Vol. 37. No. 5. P. 1167–1175. doi: <http://dx.doi.org/10.1109/TSMCB.2007.903540>
- [6] *He Z. F., Tan T. N., Sun Z. A., Qiu X. C.* Toward accurate and fast iris segmentation for iris biometrics // IEEE Trans. Pattern Anal., 2009. Vol. 31. No. 9. P. 1670–1684.
- [7] *Zhang D., Monro D. M., Rakshit S.* Eyelash removal method for human iris recognition // Conference (International) on Image Processing Proceedings. IEEE, 2006. P. 285–288. doi: <http://dx.doi.org/10.1109/ICIP.2006.313181>
- [8] *Li Y., Savvides M.* A pixel-wise, learning-based approach for occlusion estimation of iris images in polar domain // ICASSP. IEEE, 2009. P. 1357–1360. doi: <http://dx.doi.org/10.1109/ICASSP.2009.4959844>
- [9] *Ганькин К. А., Гнеушев А. Н., Матвеев И. А.* Сегментация изображения радужки глаза, основанная на приближенных методах с последующими уточнениями // Известия РАН. Теория и системы управления, 2014. №2. С. 80–94. doi: <http://dx.doi.org/10.1134/S1064230714020099>
- [10] *Соломатин И. А.* Реализация алгоритма выделения областей затенения радужки классификатором локальных текстурных признаков. 2015. <http://svn.code.sf.net/p/mlalgorithms/code/Group274/Solomatin2015EESLocalization/code/>.
- [11] Chinese Academy of Sciences Institute of Automation. CASIA-IrisV3: CASIA-Iris-Lamp image database. <http://www.cbsr.ia.ac.cn/IrisDatabase.htm>.

Поступила в редакцию 14.06.15

Detecting visible areas of iris by qualifier of local textural features*

I. A. Solomatin and I. A. Matveev

`ivan.solomatin@phystech.edu`

Moscow Institute of Physics and Technology, 9 Institutskiy per., Dolgoprudny, Russia

Dorodnicyn Computer Centre of the Russian Academy of Sciences, 40 Vavilova st., Moscow, Russia

A person recognition by the image of the iris is an actual problem. To increase the accuracy of recognition, usually areas of occlusion are detected in addition to locating of the iris as an annular region. The problem of occlusion detection can be set as the classification of pixels from annular region into two classes: “iris” and “occlusion.” In the annular region, the segment with minimum dispersion of brightness is selected, which usually contains no occlusion (in this article, this segment is not calculated and supposed to be a part of the input data).

*This work was done under financial support of the Russian Foundation for Basic Research (grant 15-01-05552)

Then, a classifier based on multivariate Gaussian is built and after that, it is trained on the training set, which is set by local textural features of the pixels from this sector. The parameters of the classifier were optimized using genetic algorithm. The problems with noise and errors of classification in particular areas of the image are solved by applying morphological postprocessing. A computational experiment was carried out and it allowed to obtain the distribution of the functional of quality of the algorithm.

Keywords: *image processing; pattern recognition; biometric identification by iris*

DOI: 10.21469/22233792.1.14.03

References

- [1] Daugman, J. 2002. How iris recognition works. *Conference (International) on Image Processing Proceedings*. IEEE. 1:33–36. doi: <http://dx.doi.org/10.1109/ICIP.2002.1037952>
- [2] Wildes, R. P. 1997. Iris recognition: An emerging biometric technology. *IEEE Proc.* 85(9):1348–1363. doi: <http://dx.doi.org/10.1109/5.628669>
- [3] He, Z. F., T. N. Tan, Z. A. Sun, and X. C. Qiu. 2008. Robust eyelid, eyelash and shadow localization for iris recognition. *15th IEEE Conference (International) on Image Processing Proceedings*. IEEE. 265–268.
- [4] Krichen, E., S. Garcia-Salicetti, and B. Dorizzi. 2007. A new probabilistic iris quality measure for comprehensive noise detection. *1st IEEE Conference (International) on Biometrics: Theory, Applications, and Systems*. 1–6. doi: <http://dx.doi.org/10.1109/BTAS.2007.4401906>
- [5] Daugman, J. 2007. New methods in iris recognition. *IEEE Trans. Syst. Man Cyb.* 37(5):1167–1175. doi: <http://dx.doi.org/10.1109/TSMCB.2007.903540>
- [6] He, Z. F., T. N. Tan, Z. A. Sun, and X. C. Qiu. 2009. Toward accurate and fast iris segmentation for iris biometrics. *IEEE Trans. Pattern Anal.* 31(9):1670–1684.
- [7] Zhang, D., D. M. Monro, and S. Rakshit. 2006. Eyelash removal method for human iris recognition. *Conference (International) on Image Processing Proceedings*. 285–288. doi: <http://dx.doi.org/10.1109/ICIP.2006.313181>
- [8] Li, Y., and M. Savvides. 2009. A pixel-wise, learning-based approach for occlusion estimation of iris images in polar domain. *ICASSP*. IEEE. 1357–1360. doi: <http://dx.doi.org/10.1109/ICASSP.2009.4959844>
- [9] Gankin, K., A. Gneushev, and I. Matveev. 2014. Iris image segmentation based on approximate methods with subsequent refinements. *J. Comput. Sys. Sc. Int.* 53(2):224–238. doi: <http://dx.doi.org/10.1134/S1064230714020099>
- [10] Solomatin, I. A. 2015. Algorithm, detecting visible areas of iris by qualifier of local textural features. Available at: <http://svn.code.sf.net/p/mlalgorithms/code/Group274/Solomatin2015EESLocalization/code/> (accessed January 11, 2016).
- [11] CASIA. Chinese Academy of Sciences Institute of Automation. CASIA-IrisV3: CASIA-Iris-Lamp image database. Available at: <http://www.cbsr.ia.ac.cn/IrisDatabase.htm> (accessed January 11, 2016).

Received June 14, 2015