

Министерство науки и высшего образования Российской Федерации
Московский Физико-технический институт
(Государственный Университет)
Факультет управления и прикладной математики
Кафедра Интеллектуальных систем
при Вычислительном Центре им. А.А.Дородницына РАН

Выпускная квалификационная работа
"Разработка метода выделения века как
параметрической кривой на изображении глаза"

Студента 4-го курса Ефремова Сергея Владимировича

Научный руководитель
д-р техн. наук Матвеев И. А.

Москва, 2020

Аннотация

Рассматривается задача выделения века как параметрической кривой на изображении глаза. Исследованы предложенные ранее схемы решения этой проблемы, на основе изученных материалов разработан подход по его выделению как параболы с помощью поиска максимума на множестве проекций. Предложен, реализован и протестирован алгоритм, основанный на параболическом преобразовании и преобразовании Радона.

Содержание

1	Введение	4
1.1	Цели и задачи работы	6
2	Постановка задачи	7
2.1	Задача выделения века	7
2.2	Формальная постановка задачи	8
2.3	Оценка качества	9
3	Обзор существующих методов	10
3.1	Модели аппроксимации века	10
3.1.1	Эллиптический контур века	10
3.1.2	Параболический контур века	10
3.2	Методы выделения контуров	11
3.2.1	Производные 1-го и 2-го порядков	12
3.2.2	Оператор Робертса	13
3.2.3	Оператор Превитта	14
3.2.4	Оператор Собеля	14
3.2.5	Преобразование Хафа	15
3.2.6	Преобразование Радона	17
3.2.7	Интегро-дифференциальный оператор (IDO)	18
3.2.8	Оператор Кэнни	19
4	Описание модели	20
4.1	Этап предобработки изображения	20
4.2	Преобразование Радона	24
5	Результаты работы алгоритма	25
5.1	Пример полученных кривых	25
5.2	Распределение ошибки	25
5.3	Сравнение результатов с другими методами	26
6	Заключение	27

1 Введение

На сегодняшний день можно с уверенностью говорить о том, что биометрические технологии идентификации вошли в нашу жизнь и используются во всех сферах человеческой деятельности, в том числе и в системах контроля. Существует целый ряд биометрических модальностей, позволяющих идентифицировать человека: рисунок папиллярных линий пальца, радужка глаза, походка, голос, почерк. Такое разнообразие методов объясняется тем, что при богатстве потенциальных сценариев использования биометрических модальностей, каждая из них может быть использована по-разному. Применение биометрических технологий уже активно осуществляется при проведении платежных операций в банковской сфере, особо нуждающейся в защите данных. Здесь они заменяют уже ставшие традиционными методы идентификации пользователей: ПИН-коды, пароли и т.д. Биометрические технологии имеют широкую перспективу использования, так как обладают целым рядом неоспоримых преимуществ, среди которых можно выделить неотчуждаемость. Например, пароль может быть передан или украден третьим лицом, и система все равно идентифицирует его как исходного пользователя, тогда как биометрическими модальностями труднее воспользоваться другому человеку или украсть без ведома обладателя.

Использование биометрических систем настолько повсеместно, что обыватель даже не замечает, как часто и повсеместно ими пользуется. Так, для разблокировки мобильных устройств, применяется отпечаток пальца, снимок лица или радужки. Еще одним преимуществом биометрических систем считается удобство взаимодействия с ними пользователя. Их эффективность работы основывается на способности обработки поступающих данных с учетом изменчивости состояния и условий окружающей среды. То есть, даже в случае значительных искажений параметров входных данных, в связи с изменчивостью состояний (например, ухудшение условий освещенности, затемнение ресницами, отвод взгляда), системы биометрической идентификации, использующие изображения глаза, должны выдавать корректный результат и аутентифицировать пользователя адекватно. Также неоспоримо важным фактором для эффективности систем такого рода, является необходимость работы в условиях ограниченности не только вычислительных ресурсов, но и времени, требующегося на распознавание.

Благодаря стремительному развитию технологий, с каждым днём все больше расширяется сфера практического применения технических средств цифровой обработки изображений. Одним из направлений стало распознавание человека на основе изображения глаза (рис. 1). Эта идея не нова, ещё в 1936 году офтальмологи [1] предложили использовать радужную оболочку глаза для распознавания. Далее исследователями

всего мира разрабатывались различные методы распознавания человека по его радужной оболочке, но по-настоящему сильный рост интереса произошел только в начале XXI века. Это объясняется качественными изменениями условий жизни общества: эволюция средств регистрирования и обработки изображений, необходимость распознавания личности в связи с террористической угрозой [14, 18, 19]. Наряду с идентификацией обработка изображения радужки применяется в медицинских целях, где также важно выделение области интереса.

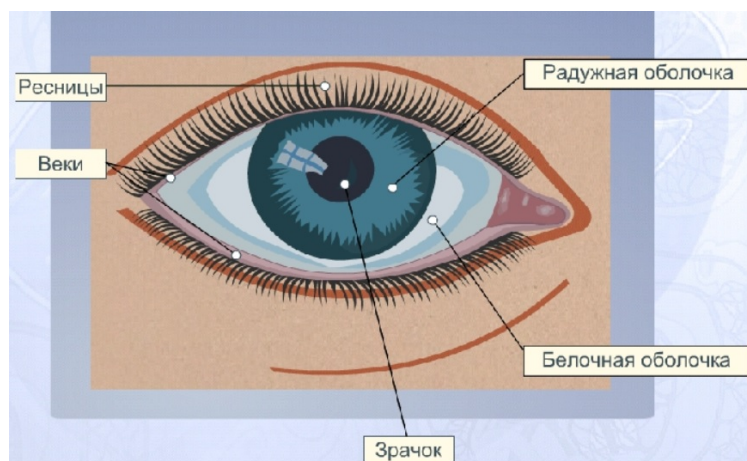


Рис. 1: Внешнее строение глаза.

Однако получить качественное и обособленное изображение радужки на практике не так просто, и важной задачей оказывается необходимость корректно определять границы глаза, задаваемые веками. Исследования в этом направлении породили различные модели, которые также используют в биометрических системах распознавания, поскольку уточнение границ и исключение нерелевантных областей оказывает значительное влияние на точность результата.

В ходе данного исследования разработан метод выделения века как параметрической кривой на изображении глаза. Он состоит в определении параметров кривой максимизацией суммы вертикального градиента яркости, рассматриваемого в окрестности века.

1.1 Цели и задачи работы

Цель и задачи исследования. Целью исследования является построение модели, позволяющей определять коэффициенты параболы, задающей положение верхнего века и улучшить качество выделения по сравнению с существующими моделями. Для реализации этой цели были поставлены следующие задачи:

- изучить существующие подходы к решению задачи выделения века на изображении глаза;
- изучить существующие алгоритмы решения этой задачи;
- создать метод предобработки данных;
- реализовать алгоритмическую часть модели, построенной на идее оптимизации и градиентного анализа;
- сравнить полученные результаты с результатами существующих алгоритмов.

Научная новизна. Используется новый подход к определению коэффициентов параболы, с помощью поиска максимума на множестве проекций.

Методы исследования. Алгоритмы реализованы на языке программирования C++ с использованием библиотеки OpenCV 3.4.

Практическая ценность. Полученная модель может быть использована в качестве встраиваемого модуля. Например, с её помощью можно:

- определять границы зрачка глаза для повышения точности идентификации по радужке;
- дополнять существующие биометрические системы с помощью признаков не только радужки, но и линии века, получая более высокое качество распознавания.

2 Постановка задачи

2.1 Задача выделения века

В алгоритмах идентификации, использующих радужку, чаще всего начинают с определения границы радужка-склера [2, 7, 12] или нормализуют радужку (рис. 2) [17], а лишь только потом определяют границу века. Это приводит к необходимости выполнять большое количество дополнительных вычислений. Рассмотрим только те методы, в которых для детектирования века необходимо заранее определить не более чем положение зрачка или радужки.

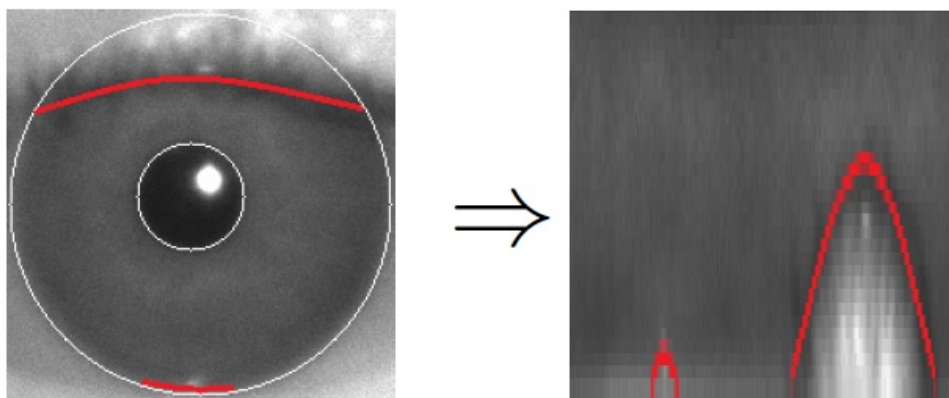


Рис. 2: Определение положения века.

Можно определить два ключевых этапа в произвольном алгоритме детектирования века на изображении глаза, один из которых — предобработка изображения, а второй — локализация века. Как правило, исследователи работают только над одним из этапов, в то время как второй заимствуют из уже ранее написанных работ, предлагая новые комбинации предобработка/локализация, которые ранее не были описаны в научной литературе. Предложенный метод сравнивается с наиболее успешными комбинациями, показывающими наиболее точный результат при тестировании.

Для нахождения границы века в [20] предложено использовать в качестве первого шага выделение границ на изображении, а затем для локализации века — преобразование Хафа.

В [12] используется сглаживание изображения с применением фильтра Гаусса и интегродифференциальный оператор (IDO) для того, чтобы найти контуры на изображении, в том числе нижнее и верхнее веко.

Ещё один метод выделения века предложен в [21], для предобработки используется одномерный фильтр пиковой формы, хорошо удаляющий шум от ресниц, а для локализации IDO.

Интуитивно понятный метод изложен в [16]. Его идея заключается в разделении области радужки и века горизонтальными прямыми, их положение вдоль вертикальной оси определялось значением максимального отклика после осуществления свёртки изображения с использованием фильтра Собела. Эта же свёртка применяется в [15] в сочетании с IDO.

В исследованиях [6, 7] используется анизотропная диффузия для погашения шумов при предобработке, фильтр Собела для выделения границ и преобразование Хафа для параболических кривых.

Для выделения границ в [22] предложен асимметричный оператор Кэнни и для локилизации — аппроксимация параболической кривой с помощью метода наименьших квадратов.

В качестве одного из методов определения границ века также можно выделить метод адаптивных контуров, предложенный в [3]. Его основная идея — задать модель контура последовательностью точек и оптимизировать функционал на этом контуре, таким образом добиваясь его эволюции.

Почти все изложенные выше методы базируются на необходимости наличия дополнительных внешних условий или требований к качеству изображения, поэтому не всегда отличаются устойчивостью в условиях постоянно изменяющегося окружения.

2.2 Формальная постановка задачи

Пусть $f(x, y)$ — множество точек, задающих линию века на изображении глаза и $y = a(x - x_0)^2 + y_0$ — уравнение аппроксимирующей параболы. Также пусть площадь круга радужки со зрачком — S . Обозначим площадь области в круге радужки, в которой парабола выше линиии века как E_2 , и площадь области, в которой выше веко — E_1 (рис. 3). Тогда суть задачи — найти такие коэффициенты (a, x_0, y_0) :

$$(a, x_0, y_0) = \arg \min_{(a, x_0, y_0)} \left\langle \max \left\{ \frac{E_1}{S}, \frac{E_2}{S} \right\} \right\rangle. \quad (2.1)$$

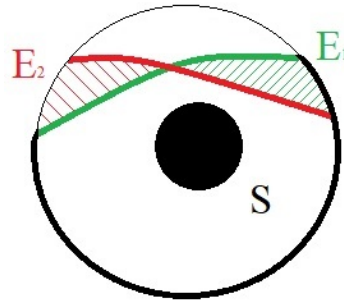


Рис. 3: Веко — зеленая линия и аппроксимирующая парабола — красная.

2.3 Оценка качества

В качестве еще одного метода оценки полученного результата предлагается рассмотреть отклонение вершины параболы от эталона. Ожидается, что это будет самая высокая точка кривой века (рис. 4).

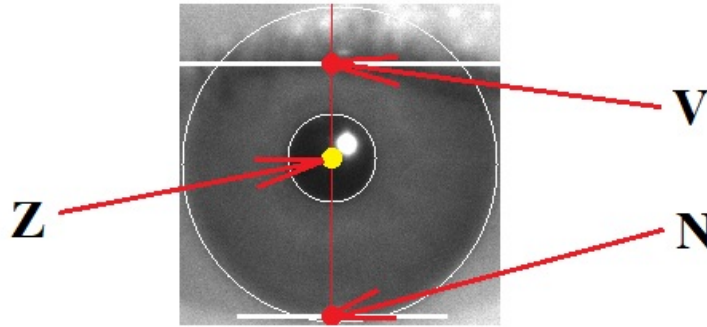


Рис. 4: Определение положения века: V и N — точки, соответствующие положениям верхнего и нижнего века; Z — центр зрачка.

Тогда анализируемая метрика примет вид:

$$dist = \sqrt{\frac{(y - y_0)^2}{y_0^2} + \frac{(x - x_0)^2}{x_0^2}}, \quad (2.2)$$

где (x_0, y_0) — экспертная разметка вершины, а (x, y) — экспериментальная.

3 Обзор существующих методов

3.1 Модели аппроксимации века

3.1.1 Эллиптический контур века

Один из возможных способов приблизить линию века — описать её с помощью эллиптической кривой. Эта модель основана на предположении, что глазное яблоко имеет сферическую форму:

$$x^2 + y^2 + z^2 = R^2. \quad (3.1)$$

Такое предположение приводит к модели, основанной на открытости глаза. Степень открытости — угловое положение века относительно центра сферы. Тогда кривую века можно получить как пересечение сферы глазного яблока с плоскостью проходящей через ось x с углом φ относительно оси z (рис. 5). Уравнение этой плоскости имеет вид:

$$\tan \varphi = \frac{y}{z}. \quad (3.2)$$

А кривая пересечения, моделирующая веко, задается эллиптической кривой:

$$x^2 + \frac{y^2}{\sin^2 \varphi} = R^2. \quad (3.3)$$

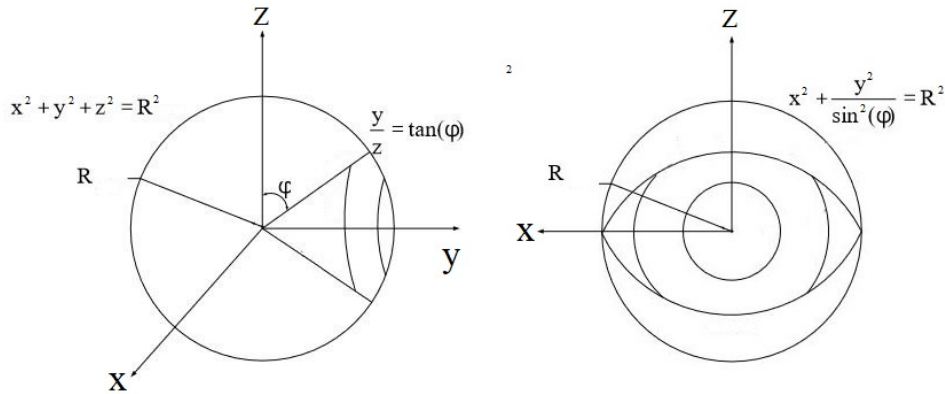


Рис. 5: Вид сбоку и спереди эллиптического контура века.

3.1.2 Параболический контур века

На практике оказывается, что определять кривую века как пересечение плоскости и сферы довольно грубо. Эмпирическим путем было установлено, что форма века глаза человека гораздо лучше аппроксимируется квадратичной параболой (рис. 6). Тогда уравнение параметрической кривой, задающее контур века принимает вид:

$$y = a(x - x_0)^2 + y_0. \quad (3.4)$$

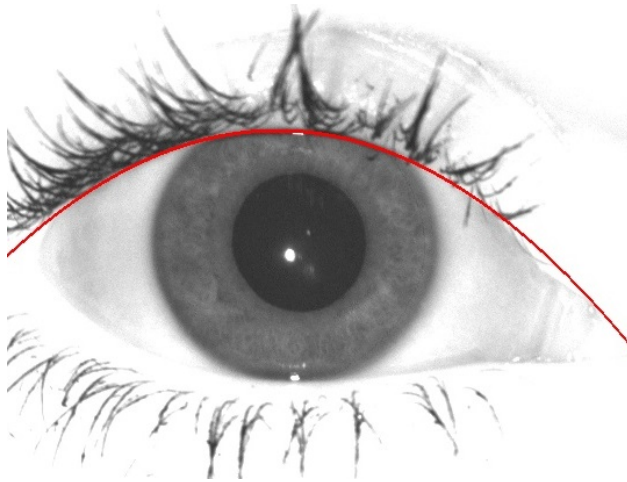


Рис. 6: Пример параболического контура века на изображении глаза.

3.2 Методы выделения контуров

Практически все методы выделения контуров основываются на перепадах сигнала яркости. Наиболее общим способом их поиска является обработка изображения методом скользящего окна, называемого в разной литературе также маской, фильтром, ядром или шаблоном. Оно представляет из себя некую прямоугольную матрицу, накладываемую на группу пикселей исходного изображения. Элементы такой матрицы называют коэффициентами, а разного рода преобразования над ней — фильтрацией или пространственной фильтрацией (рис. 7).

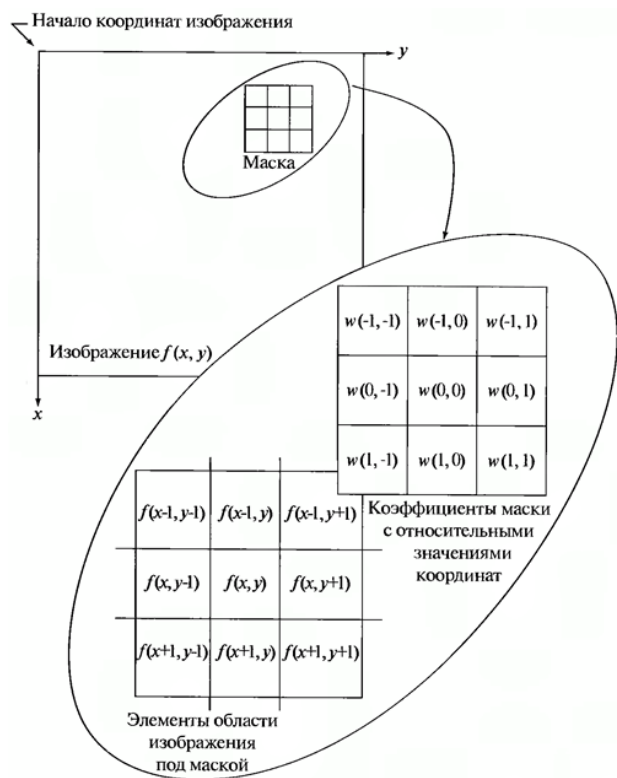


Рис. 7: Схема пространственной фильтрации.

Метод основан на перемещении фильтра от точки к точке, вычисляется отклик для каждой из них. Для случая линейной пространственной фильтрации значение задается суммой произведения коэффициентов маски на соответствующие значения яркостей в точках, покрытых маской. Так, для квадратной маски размерности 3×3 отклик R линейной фильтрации в точке (x, y) изображения на рис. 7 определится как сумма произведений значений маски на значения пикселей под маской:

$$R = \sum_{s=-1}^1 \sum_{t=-1}^1 w(s, t) f(x + s, y + t). \quad (3.5)$$

3.2.1 Производные 1-го и 2-го порядков

Для определения направления и величины перепада яркости чаще всего применяют градиент изображения ∇f , определяемый в точке (x, y) как вектор:

$$\nabla f \equiv \text{grad}(f) \equiv \begin{bmatrix} g_x \\ g_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}. \quad (3.6)$$

Важным геометрическим свойством градиента является тот факт, что его направление совпадает с направлением максимальной скорости изменения функции f в точке (x, y) . Направление вектора градиента задаётся углом между направлением вектора ∇f и осью x :

$$\alpha(x, y) = \text{arctg} \left(\frac{g_y}{g_x} \right). \quad (3.7)$$

Направление контура в произвольной точке (x, y) таким образом получается перпендикулярно направлению $\alpha(x, y)$ вектора градиента в этой точке (рис. 8).



Рис. 8: Применение градиента для нахождения границ контуров на изображении.

При работе с изображениями функция яркости имеет дискретную природу, поэтому вместо производных первого и второго порядка используются их дискретные аналоги.

Первая производная одномерной функции может быть задана как разность значений соседних элементов:

$$\frac{\partial f}{\partial x} = f(x + 1) - f(x). \quad (3.8)$$

Аналогично вторая производная определяется как разность соседних значений первой производной:

$$\frac{\partial^2 f}{\partial^2 x} = f(x+1) + f(x-1) - 2f(x). \quad (3.9)$$

Нетрудно заметить, что по факту эти преобразования задают маски фильтрации для первой и второй производной вдоль вертикальной координатной оси (рис. 9).

-1	1
1	-2
	1

Рис. 9: Маски фильтрации для первой и второй производной вдоль вертикальной координатной оси.

Однако способ дискретно приблизить производную, описанный выше, не единственный. Можно, например, использовать маски размером 3×3 . Далее описаны другие вариации градиентных операторов.

3.2.2 Оператор Робертса

Обозначим область изображения размером 3×3 и значения яркостей каждой из ее точек в соответствии с рис. 10.

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

Рис. 10: Яркости в области 3×3 на изображении.

Один из возможных способов нахождения первых частных производных в точке z_5 состоит в применении перекрестного оператора Робертса:

$$g_x = \frac{\partial f}{\partial x} = (z_9 - z_5), \quad (3.10)$$

$$g_y = \frac{\partial f}{\partial y} = (z_8 - z_6). \quad (3.11)$$

Такое преобразование задает маски фильтрации на рис. 11 соответственно.

-1	0	0	-1
0	1	1	0

Рис. 11: Маски оператора Робертса.

Основным преимуществом данного градиентного оператора является высокая скорость обработки изображения, но зачастую устойчивость оставляет желать лучшего.

3.2.3 Оператор Превитта

Используем обозначения рис. 10. Тогда оператор Превитта может быть задан выражениями:

$$g_x = \frac{\partial f}{\partial x} = (z_7 + z_8 + z_9) - (z_1 + z_2 + z_3), \quad (3.12)$$

$$g_y = \frac{\partial f}{\partial y} = (z_3 + z_6 + z_9) - (z_1 + z_4 + z_7). \quad (3.13)$$

Маска этого градиентного оператора имеет вид, изображённый на рис. 12.

-1	-1	-1	-1	0	1
0	0	0	-1	0	1
1	1	1	-1	0	1

Рис. 12: Маски оператора Превитта.

3.2.4 Оператор Собеля

Небольшое изменение формул (3.12) в виде использования весового коэффициента 2 для средних элементов, позволяет уменьшить эффект сглаживания для функции градиента:

$$g_x = \frac{\partial f}{\partial x} = (z_7 + 2z_8 + z_9) - (z_1 + 2z_2 + z_3), \quad (3.14)$$

$$g_y = \frac{\partial f}{\partial y} = (z_3 + 2z_6 + z_9) - (z_1 + 2z_4 + z_7). \quad (3.15)$$

Маски, используемые оператором Собеля, отображены на рис. 13.

-1	-2	-1	-1	0	1
0	0	0	-2	0	2
1	2	1	-1	0	1

Рис. 13: Маски оператора Собеля.

Стоит отметить, что у каждой из градиентных масок, описанных выше, сумма весовых коэффициентов равна нулю, то есть эти операторы будут давать нулевой отклик на областях постоянной яркости, как и следовало ожидать от дифференциального оператора.

3.2.5 Преобразование Хафа

Рассмотренные выше методы применимы, когда имеется дополнительная информация о положении исследуемых объектов на изображении. Зачастую таких данных нет и приходится исходить из заранее заданных глобальных свойств. Ниже изложен метод, основанный на выяснении того, лежат ли множества пикселей на кривой заранее заданной формы. Рассмотрим случай, когда искомая фигура изначально имеет форму прямой, и воспользуемся преобразованием Хафа для её поиска. Рассмотрим точку (x_i, y_i) исходного изображения и общее уравнение прямой на плоскости: $y = ax + b$. Через точку (x_i, y_i) проходит бесконечное множество прямых, удовлетворяющих уравнению: $y_i = ax_i + b$. Перепишем это уравнение в виде: $b = x_i a - y_i$ и перейдём систему координат Oab , называемую пространством параметров. Очевидно, что в нём для заданной пары (x_i, y_i) существует уравнение единственной прямой. Более того для каждой пары $(x_i, y_i), (x_j, y_j)$ точек в исходном пространстве, в пространстве параметров существует единственная точка (a', b') , находящаяся на пересечении соответствующих для точек прямых пространства параметров и задающая прямую, на которой лежат эти точки в исходном пространстве (рис. 14).

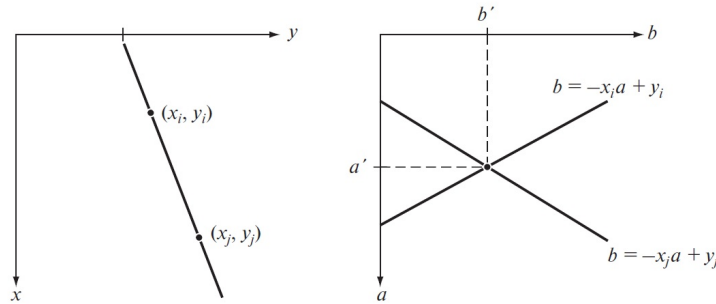


Рис. 14: Исходная плоскость (xy) , пространство параметров (ab) .

Проблема использования такого перехода проявляется при попытке отобразить прямую близкую к вертикальной, так как ее угловой коэффициент a стремится к бесконечности. Чтобы обойти данную трудность прямую представляют с помощью нормали:

$$x \cos \Theta + y \sin \Theta = \rho. \quad (3.16)$$

Геометрическую интерпретацию параметров ρ и Θ можно увидеть на рис. 15-а. Каждая синусоида на рис. 15-б представляет семейство прямых, проходящих через конкретную точку (x_k, y_k) на исходной плоскости xy . Точка пересечения (ρ', Θ') на этом рисунке соответствует прямой, проходящей через точки (x_i, y_i) и x_j, y_j на рис. 15-а.

Осуществив преобразование Хафа, далее предлагается разбить пространство параметров $\rho\Theta$ на так называемые ячейки накопления, как показано на рис. 15-в. Затем

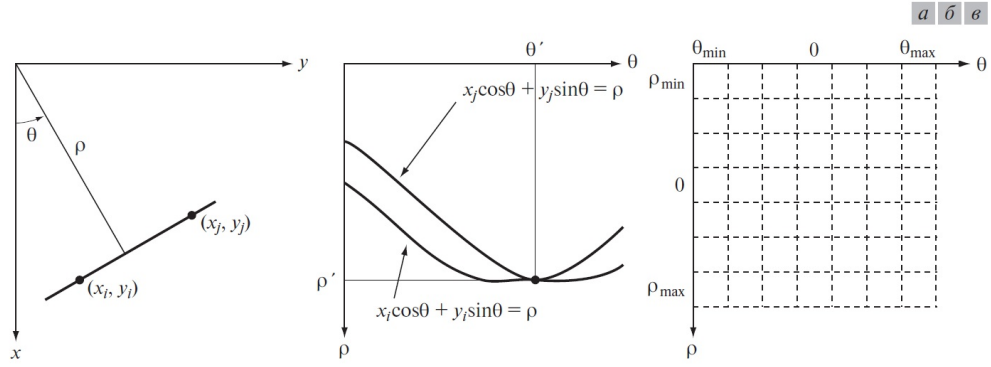


Рис. 15: (а) Представление прямой на исходной плоскости с помощью параметров нормали (ρ, Θ) . (б) Синусоиды на плоскости $\rho\Theta$. (в) Разбиение плоскости $\rho\Theta$ на ячейки накопления.

для каждой точки (x_k, y_k) исходной плоскости перебираем возможные значения для параметра Θ и находим соответствующее ему значение $\rho = x_k \cos \Theta + y_k \sin \Theta$. Если выбор значения Θ_p приводит к решению ρ_q , то увеличиваем значение ячейки $A(p, q)$ на 1. После выполнения этого алгоритма значение $A(p, q) = Q$ означает, что на исходной плоскости находится Q точек лежащих на прямой $\rho_p = x_k \cos \Theta_q + y_k \sin \Theta_q$. Отсюда и находят прямые контуры поиском локальных максимумов.

Аналогичным образом можно применять преобразование Хафа к любой функции вида $g(v, c) = 0$, где v — вектор координат, а c — вектор коэффициентов. Так, любое коническое сечение, в частности парабола, может быть задано уравнением в полярных координатах:

$$\rho = \frac{de}{1 \pm e \cos \Theta} \quad (3.17)$$

или

$$\rho = \frac{de}{1 \pm e \sin \Theta}, \quad (3.18)$$

где e — эксцентриситет и d — кратчайшее расстояние между фокусом и директрисой. Для конических сечений эксцентриситет имеет разные значения в зависимости от формы кривой. Для эллипса он лежит в диапазоне: $0 < e < 1$, для гиперболы — $e > 1$, а для параболы — $e = 1$, её далее и рассмотрим.

Парабола может быть определена как геометрическое множество точек, равноудалённых от неподвижной точки, называемой фокусом, и прямой, называемой директрисой (рис. 16). Эта парабола описывается уравнением:

$$(y - y_0)^2 = 4a(x - x_0) \quad (3.19)$$

или

$$Ay^2 + By + C = x \quad (3.20)$$

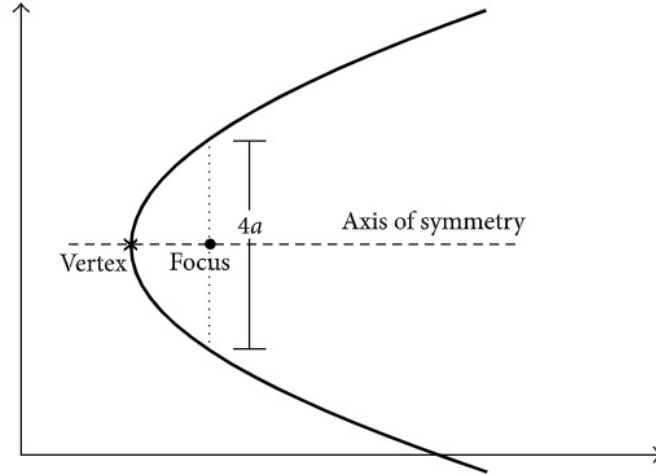


Рис. 16: Парабола.

Коэффициенты можно вычислить по формулам [8]:

$$A = \frac{y_k(x_j - x_i) + y_j(x_i - x_k) + y_i(x_k - x_j)}{(y_i - y_j)(y_i - y_k)(y_j - y_k)}, \quad (3.21)$$

$$B = \frac{y_k^2(x_j - x_i) + y_j^2(x_i - x_k) + y_i^2(x_k - x_j)}{(y_i - y_j)(y_i - y_k)(y_j - y_k)}, \quad (3.22)$$

$$C = \frac{y_j y_k (y_j - y_k) x_i + y_k y_i (y_k - y_i) x_j + y_i y_j (y_i - y_j) x_k}{(y_i - y_j)(y_i - y_k)(y_j - y_k)}. \quad (3.23)$$

Таким образом, перебрав все тройки точек, переходим в пространство коэффициентов, в котором, аналогично пространству для прямых, считаем значения в ячейках накопления, но уже в трехмерных.

Наконец, можно получить значения координат вершины и апертуры параболы:

$$x_0 = -\frac{B}{2A}, \quad (3.24)$$

$$y_0 = C - \frac{B^2}{4A}, \quad (3.25)$$

$$4a = \frac{1}{A}. \quad (3.26)$$

3.2.6 Преобразование Радона

Рассмотрим простейший случай преобразования Радона, а именно случай двух переменных. Исходим из предположения, что $f(x, y)$ определена на всей плоскости и достаточно быстро убывает на бесконечности. Последнее условие необходимо для сходимости несобственных интегралов. Тогда преобразованием Радона $f(x, y)$ называется функция [13]:

$$Rf(s, \alpha) = \int_{-\infty}^{\infty} f(x(z), y(z)) dz = \int_{-\infty}^{\infty} f(s \cos \alpha + z \sin \alpha, s \sin \alpha - z \cos \alpha) dz \quad (3.27)$$

Геометрический смысл этого преобразования — это интеграл от функции вдоль прямой, перпендикулярной вектору $\vec{n} = (\cos \alpha, \sin \alpha)$ и проходящей на расстоянии s от начала координат вдоль этого вектора (рис. 17).

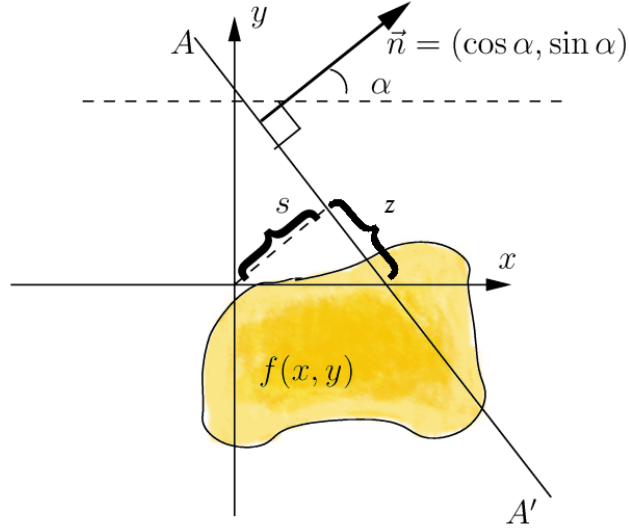


Рис. 17: Преобразование Радона.

3.2.7 Интегро-дифференциальный оператор (IDO)

Оператор, предложенный в [10], а затем развитый в [11, 12], используется в первую очередь для детектирования объектов круглой формы: в оригинале зрачка и радужки. Также предложено детектировать верхние и нижние веки путём настройки поиска контура с кругового на дугообразный [12].

Интегро-дифференциальный оператор определяется как:

$$\max_{r, x_0, y_0} \left| G_\sigma(r) * \left(\frac{\delta}{\delta r} \right) \int_{r, x_0, y_0} \frac{I(x, y)}{2\pi r} ds \right|. \quad (3.28)$$

При непрерывном применении оператора на изображении максимизируется исследуемый функционал по трём параметрам (x_0, y_0) — координаты центра и радиус r . Ожидается, что границы зрачка и радужки будут максимизировать производную интеграла по круговым границам, так как на них значения интенсивности резко меняются. $G_\sigma(r)$ — сглаживающая функция. Для век алгоритм аналогичный, только форма искоемых контуров будет иметь другой вид. Существенная проблема для данного метода — излишняя зашумлённость изображения, которая может породить внезапные перепады у интегрируемого функционала.

3.2.8 Оператор Кэнни

Детектор контуров, предложенный в [9], является многоступенчатым алгоритмом обнаружения контуров различной формы и природы. Последовательность действий может быть описана таким образом:

1. Применить гауссовский фильтр для сглаживания изображения и удаления шума;
2. Найти градиенты интенсивности изображения;
3. Подавить немаксимумы для корректного определения границ;
4. Применить пороговую фильтрацию для определения потенциальных контуров;
5. Выделение наиболее весомых контуров.

Фильтр Гаусса

Уравнение, задающее ядро Гауссовского фильтра размера $(2k + 1) \times (2k + 1)$:

$$H_{ij} = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{(i - (k + 1))^2 + (j - (k + 1))^2}{2\sigma^2}\right); \quad 1 \leq i, j \leq (2k + 1). \quad (3.29)$$

Очевидно, выбор ядра сглаживания влияет на производительность детектора. Чем больше размер фильтра, тем ниже чувствительность к шуму, но тем больше шансы потерять важную информацию.

4 Описание модели

Предложенный в данной работе метод определения положения века основан на применении вертикального градиента с последующими параболическим преобразованием и преобразованием Радона (рис. 18)).

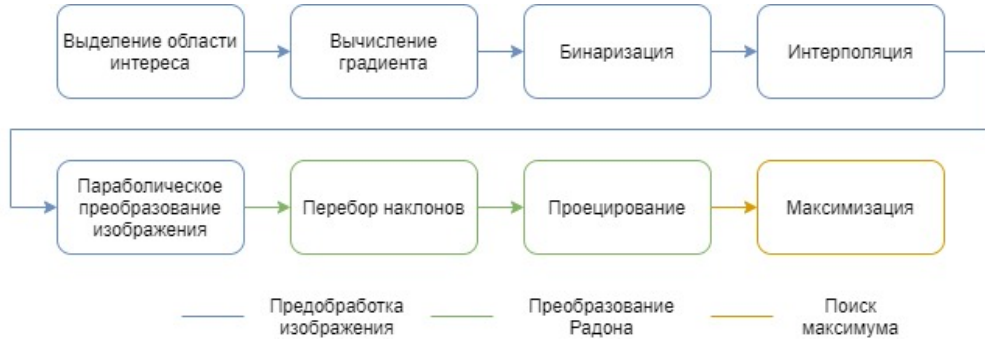


Рис. 18: Алгоритм детектирования века на изображении.

В качестве модели кривой, задающей линию века, выбрана параболическая. Как было описано выше, она позволяет довольно чётко определить границу века и на практике оказывается выигрышной. Таким образом, основной задачей алгоритма становится определить три ключевых параметра: a , x_0 и y_0 в (4.1), однозначно задающих положение века.

$$y = a(x - x_0)^2 + y_0. \quad (4.1)$$

4.1 Этап предобработки изображения

В качестве входных данных алгоритма, помимо самого изображения глаза, используются координаты центра и радиус зрачка. Нужны эти данные для того, чтобы локализовать веко и ускорить обработку изображения. Высоту рамки области, в которой ищется веко, устанавливаем равной $4P_{up}R$, где $P_{up}R$ — радиус зрачка. Область поиска смещаем на 15% от вертикальной координаты центра зрачка. Также оставляем отступ в 5% от горизонтальной координаты центра зрачка слева и справа. Полученный результат локализации изображен на рис. 19. Все коэффициенты подобраны опытным путём.

Заметим, что линия века на фотографии чаще всего сопровождается резким перепадом яркости по вертикали. Соответственно легко можно отследить линию века, продифференцировав изображение вдоль вертикальной оси координат (рис. 20).

Помимо непосредственного дифференцирования изображения, также применяем

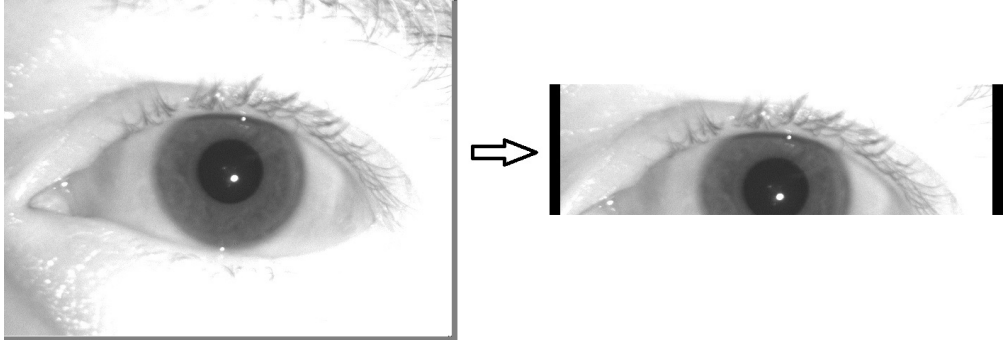


Рис. 19: Сужение области поиска.

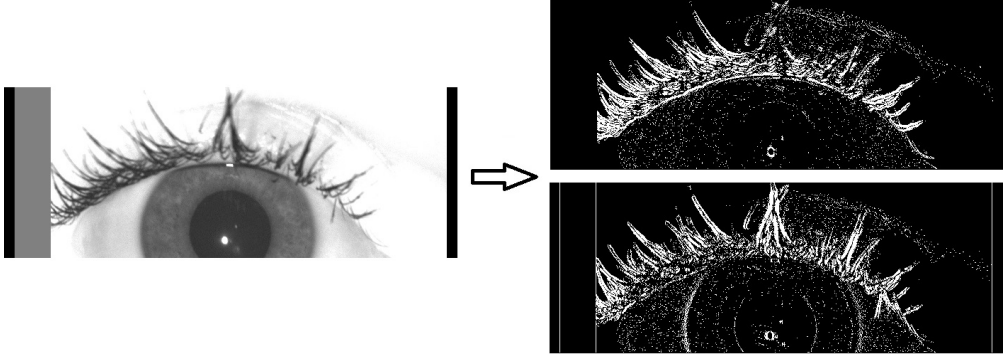


Рис. 20: Применение к изображению вертикального и горизонтального градиента с пороговой фильтрацией.

пороговую фильтрацию к модулю разницы значений яркостей в соседних пикселях:

$$I_d(x, y) = \begin{cases} 255, & \text{если } |I(x, y + 1) - I(x, y)| > \text{por}, \\ 0, & \text{иначе.} \end{cases} \quad (4.2)$$

Величина порогового значения выбрана равной $\text{por} = 10$ эмпирически.

Заметим, что изображение получилось достаточно зашумлённым. Для того, чтобы это исправить, воспользуемся линейной дискретной интерполяцией. Линейное сглаживание по трём точкам [5] — это операция усреднения с помощью интерполяционных многочленов, обеспечивающая возможность получения усредненного значения в точке по заданным соседним. Горизонтальное сглаживание задаётся формулами (4.3), где X — число пикселей в строке.

$$I_a(x, y) = \begin{cases} \frac{5I(0,y)+2I(1,y)-I(2,y)}{6}, & \text{если } x = 0 \\ \frac{I(x-1,y)+I(x,y)+I(x+1,y)}{3}, & \text{если } 1 \leq x \leq X - 1 \\ \frac{5I(X,y)+2I(X-1,y)-I(X-2,y)}{6}, & \text{если } x = X \end{cases} \quad (4.3)$$

Однако, так как мы работаем в бинаризованном пространстве, формула (4.3) при-

обретает вид:

$$I_{a-bin}(x, y) = \begin{cases} I_a, & \text{если } I_a = 255, \\ 0, & \text{иначе.} \end{cases} \quad (4.4)$$

После наложения интерполяционного фильтра изображение приобретает вид рис. 21.

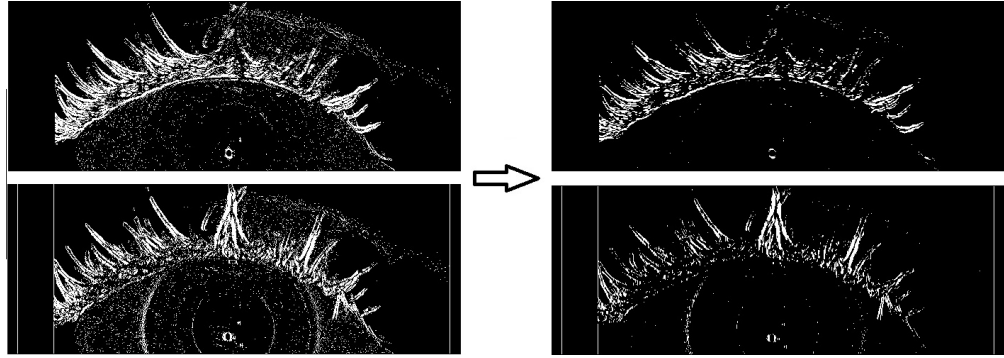


Рис. 21: Применение к изображениям вертикального и горизонтального градиента горизонтальной и вертикальной интерполяции соответственно.

Заметим, что помимо линии века, очень большой вклад в изображение вертикального градиента всё ещё дают ресницы, однако, на изображении горизонтального градиента веко практически отсутствует, а ресницы всё так же хорошо видны. Воспользуемся этим моментом для уменьшения шума ресниц (рис. 22).

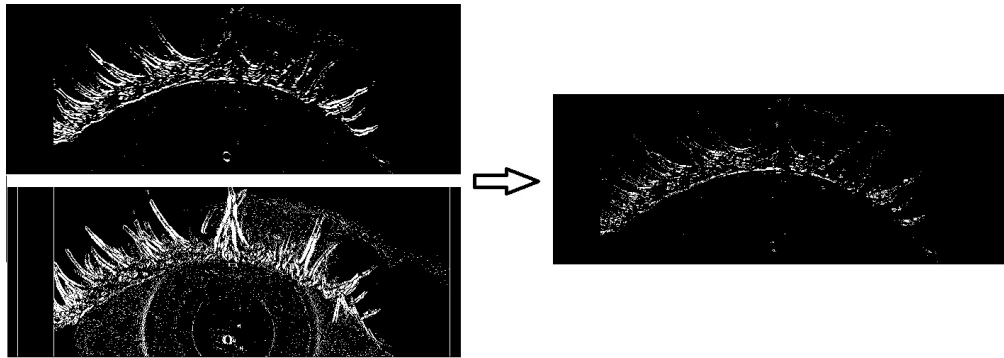


Рис. 22: Уменьшение шума ресниц.

Последним этапом предобработки изображения является параболическое преобразование. Его можно представить таким образом:

$$\begin{cases} x' = x, \\ y' = y - ax^2, \end{cases} \quad (4.5)$$

где параметр a определяет это преобразование. Заметим, что если в исходном простран-

стве была задана кривая вида:

$$y = a(x - x_0)^2 + y_0, \quad (4.6)$$

то в пространстве (y', x') она примет вид:

$$y = -2ax_0x + (ax_0^2 + y_0) = bx + c, \quad (4.7)$$

то есть станет прямой.

Перебирая возможные коэффициенты, мы можем добиться того, чтобы веко приняло форму прямой в конечном пространстве. Пример удачного преобразования на рис. 23.

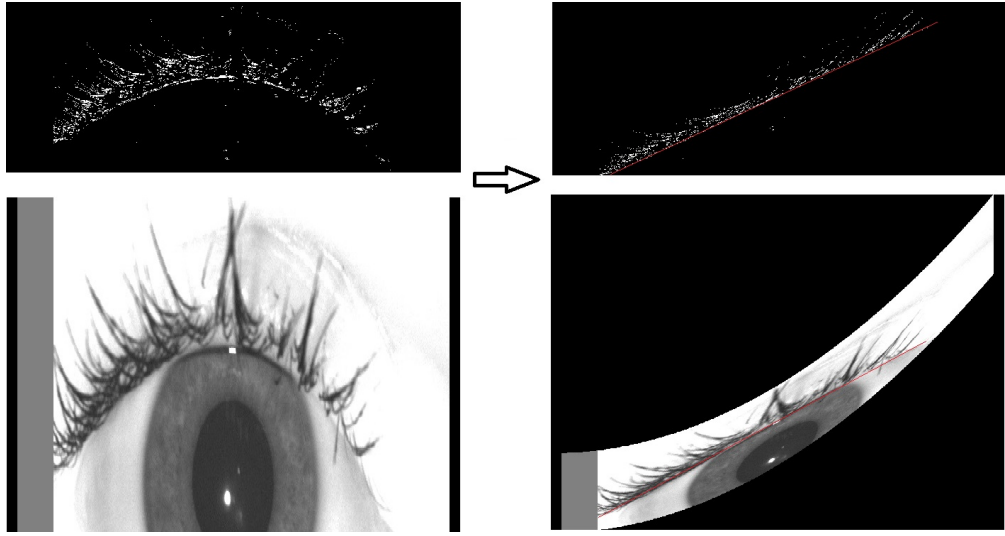


Рис. 23: Параболическое преобразование изображения.

Теперь необходимо найти прямую и определить её наклон для корректного определения коэффициентов x_0 и y_0 для века из уравнения (4.6).

4.2 Преобразование Радона

Перейдём к дискретной интерпретации этого метода. Для того, чтобы применить такое преобразование к изображению, можно разделить его на два этапа: выполнить преобразование поворота и просуммировать построчно значения яркостей. Тогда изменение наклона может быть задано формулой:

$$\begin{cases} x' = x, \\ y' = y - bx. \end{cases} \quad (4.8)$$

Тогда, если прямая изначально имела вид $y = bx + c$, то после преобразования она примет вид $y = c$. Так как шум на изображении всё равно присутствует, далее нужно максимизировать просуммированное значение яркостей вдоль всех строк изображения. Полученный максимум ожидаем в положении последнего искомого коэффициента c . Пример преобразования наклона на рис. 24.

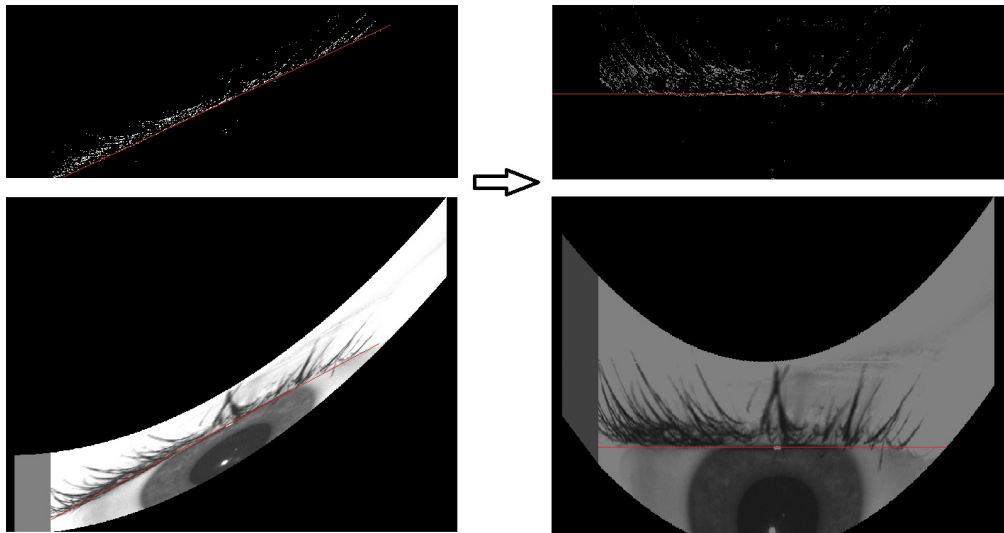


Рис. 24: Поворот изображения.

5 Результаты работы алгоритма

5.1 Пример полученных кривых

Пример параболических кривых, построенных с помощью описанного выше алгоритма, на рис. 25.

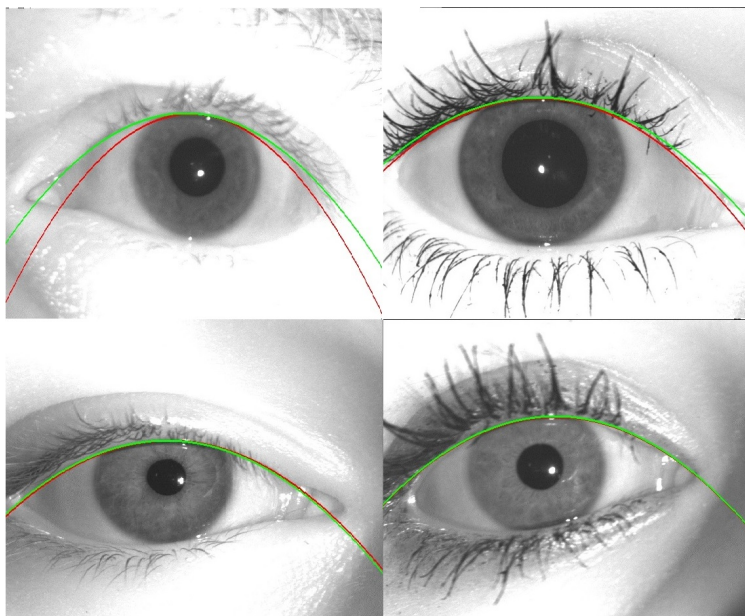


Рис. 25: Сравнение результата с эталоном. Красная линия — эталон, зеленая — результат алгоритма.

5.2 Распределение ошибки

Распределение величины ошибки на всей базе изображений *LG4000* (2864 изображения) на рис. 26.

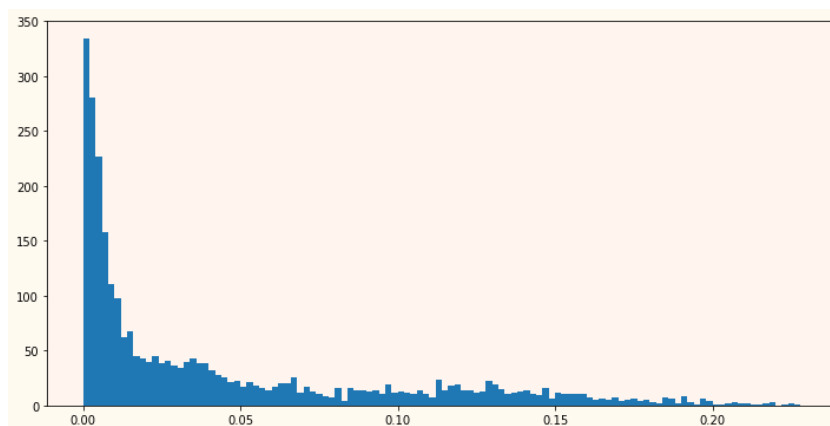


Рис. 26: Распределение величины ошибки на базе *LG4000*.

Средняя величина ошибки на всей выборке — 4,8%, медианная ошибка — 2,1%.

5.3 Сравнение результатов с другими методами

Также рассчитаем величину ошибки $dist$ (2.2) и ее распределение (рис. 27).

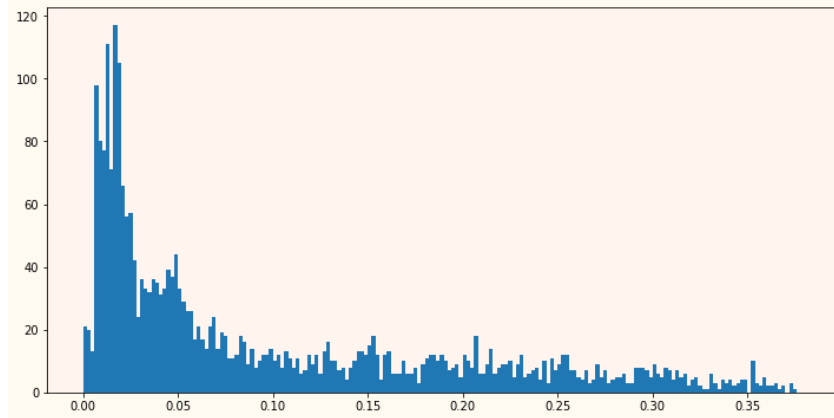


Рис. 27: Распределение величины ошибки $dist$ на базе $LG4000$.

Средняя величина ошибки $dist$ на всей базе изображений — 12,2%, медианная ошибка — 5,8%.

В Таблице 1 дано сравнение результатов работы алгоритма с другими методами, упомянутыми в разделе 2.1.

Модель	Ошибка, %
Пр. Хафа [20]	19,4
Ф. Собела + IDO [15]	7,7
Ф. Собела + пр. Хафа [7]	11,5
Ф. Габора + IDO [21]	14,2
Исследуемая модель	12,2

Таблица 1: Сравнение точности определения крайней верхней точки верхнего века существующими современными алгоритмами с исследуемой моделью

Нетрудно заметить, что даже эта оценка качества даёт сопоставимый результат.

6 Заключение

Предложен метод, разработан и протестирован алгоритм выделения века как параболической кривой. Сравнение с разметкой эксперта показывает состоятельность подхода и хорошее качество результата. Аналогичные выводы можно сделать из сравнения результатов работы предложенного метода с существующими алгоритмами. Возможными направлениями развития могут быть: сужение потенциальной области поиска искомых параметров, уменьшение шума ресниц и оптимизация работы алгоритма для неожиданных углов наклона изображения.

Список литературы

- [1] Вельховер Е.С., Шульпина Н.Б., Алиева З.А. и др. Иридодиагностика // М.: Медицина. — 1988. — Р. 240.
- [2] Соломатин И.А., Матвеев И.А., Новик В.П. Определение видимой области радужки классификатором текстур с опорным множеством // Автоматика и телемеханика. — 2018. — Рр. 127–143.
- [3] Смирнов Д.А., Матвеев И.А. Определение границ век на изображении глаза методом активных контуров // Труды ИСА РАН, Динамика неоднородных систем. — 2006. — Vol. 25. — Рр. 200–207.
- [4] Гонсалес Р., Вудс Р. Цифровая обработка изображений // Техносфера. — 2012.
- [5] Дьяконов В.П. Справочник по алгоритмам и программам на языке бейсик для персональных ЭВМ // М.: Наука. — 1987.
- [6] Adam M., Rossant F. et al. Eyelid localization for iris identification // Radioengineering. — 2008. — Vol. 17, no. 4. — Рр. 82–85.
- [7] Adam M., Rossant F. et al. Reliable eyelid localization for iris identification // Advanced Concepts for Intelligent Vision Systems Conference Proceedings. — 2008. — Рр. 1062–1070.
- [8] Ballard D. Generalizing the hough transform to detect arbitrary shapes // Pattern Recognition. — 1981. — Vol. 13, no. 2. — Рр. 111–122.
- [9] Canny J. A computational approach to edge detection // IEEE Trans. Pattern Anal. Mach. Intell. — 1986. — Vol. 8, no. 6. — Рр. 679–698.
- [10] Daugman J. High confidence visual recognition of persons by a test of statistical independence // IEEE Transactions on Pattern Analysis and Machine Intelligence. — 1993. — Vol. 15, no. 11. — Рр. 1148–1161.
- [11] Daugman J. The importance of being random: statistical principles of iris recognition // Pattern Recognit. — 2003. — Vol. 36. — Рр. 279–291.
- [12] Daugman J. How iris recognition works // IEEE Trans. Circuits Syst. Video Techn. — 2004. — Vol. 14. — Рр. 21–30.
- [13] Deans S. The radon transform and some of its applications // New York: John Wiley and Sons. — 1983.

- [14] Future challenges based on the multiple biometric grand challenge // NIST information access division: Multiple biometric grand challenge. — 2010.
- [15] *Kang B., Park K.* A robust eyelash detection based on iris focus assessment // *Pattern Recognition Letters*. — 2007. — Vol. 28, no. 13. — Pp. 1630–1639.
- [16] *Masek L.* Recognition of human iris patterns for biometric identification // *Measurement*. — 2003. — Vol. 32, no. 8. — Pp. 1502–1516.
- [17] *Min T., Park R.* Comparison of eyelid and eyelash detection algorithms for performance improvement of iris recognition // *Conference (International) on Image Processings Proceedings*. — 2008. — Pp. 257–260.
- [18] Multiple biometric evaluation // <http://www.nist.gov/itl/iad/ig/mbe.cfm>. — 2009.
- [19] Multiple biometric grand challenge // <http://www.nist.gov/itl/iad/ig/mbgc.cfm>. — 2007.
- [20] *Wildes R.* Iris recognition: an emerging biometric technology // *Proceedings of the IEEE*. — 1997. — Vol. 85, no. 9. — Pp. 1348–1363.
- [21] *Xiangde Z., Qi W. et al.* Noise detection of iris image based on texture analysis // *Chinese Control and Decision Conference Proceedings*. — 2009. — Pp. 2366–2370.
- [22] *Yang L., Wu T. et al.* Eyelid localization using asymmetric canny operator // *Conference (International) on Computer Design and Applications Proceedings*. — 2010. — Pp. 533–535.