Автономная некоммерческая общеобразовательная организация"Физтех-лицей"имени П.Л. Капицы

XX научно-практическая конференция «Старт в инновации»

Современная методология идентификации по сетчатке глаза: анализ, сравнение, реализация

Выполнили: Садчиков Андрей
Лодыгин Роман
Ментий Артемий

Руководитель: Поляков Георгий Владимирович

г. Долгопрудный
2021

Содержание

1 Введение **2**
1.1 Роль биометрии в современной жизни 2
1.2 Цели и задачи 2

2 Строение глазного дна **3**
2.1 С точки зрения биологии 3
2.2 Проблематика снятия изображения 3

3 Способы идентификации **3**
3.1 Классические алгоритмы 3
3.1.1 Применение фазовой корелляции 3
3.1.2 Алгоритм, основанный на углах Харисса 5
3.1.3 Алгоритм, основанный на точках разветвления 6
3.2 Нейросетевые алгоритмы 8

4 Реализация алгоритмов **10**
4.1 Алгоритм, основанный на методе фазовой корреляции 10
4.1.1 Извлечение рисунка сосудов 10
4.1.2 Оценка сдвига изображения 11
4.1.3 Скорость выполнения 12
4.2 Нейросетевой алгоритм 12
4.2.1 Алгоритм работы сверточных сетей 12
4.2.2 Обучение сети 13
4.2.3 Переобучение 14

5 Оценка точности **15**
5.1 Точность корреляции 15
5.1.1 Зависимость точности от искажения изображения . . . 15
5.1.2 Способы увеличения точности 16
5.2 Точность нейронной сети 17
5.2.1 Зависимость точности от количества эпох 17

6 Вывод **17**

7 Приложение **18**

1 Введение

1.1 Роль биометрии в современной жизни

В современном мире всё чаще используются различные способы идентификации человека. От простых – в виде пароля или кода, до индивидуального ключа, который может использовать различные биометрические характеристики человека. Идентификация человека становится всё более важным аспектом современной жизни. Поэтому всё важнее становятся способы защиты от потери данных. Биометрия — это система распознавания людей по поведенческим, психологическим, анатомическим и физиологическим особенностям. На данный момент существует множество способов идентификации личности человека. Важно отметить то, что по данным специалистов в области биологии и медицины, человек имеет множество индивидуальных черт, в которые входят: узор пальцев, почерк, рисунок сетчатки глаза, голос и многое другое.

1.2 Цели и задачи

Создать оптимальное решение, способное выполнять аутентификацию по изображению сетчатки глаза. Для этого:

1. Реализовать один из классических алгоритмов идентификации
2. Создать и обучить модель нейронной сети
3. Описать достоинства и недостатки каждого из двух полученных решений, сравнить полученные результаты и определить, если это возможно, наилучший путь решения проблемы.

2 Строение глазного дна

2.1 С точки зрения биологии

Сетчатка – это внутренняя оболочка глаза, являющаяся высокодифференцированной нервной тканью, играющей важнейшую роль в обеспечении зрения. Уникальность строения сетчатки обеспечивает функционирование зрительного анализатора.

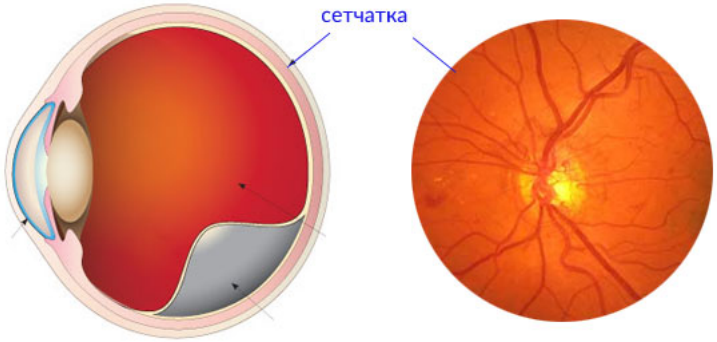


Рис. 1: Схематичное изображение сетчатки.

Сетчатка имеет две основные функции: центральное и периферическое зрение. Благодаря периферическому зрению человек ориентируется в пространстве.

2.2 Проблематика снятия изображения

Так как сетчатка расположена относительно глубоко, снятие изображения сетчатки не является тривиальной задачей. В отличие от лица или радужной оболочки, практически невозможно использовать камеру смартофна или ноутбука для идентификации по сетчатке. Для получения изображений высокого качества используют офтальмоскоп.

3 Способы идентификации

3.1 Классические алгоритмы

3.1.1 Применение фазовой корелляции

Фазовая корреляция - это подход к оценке относительного поступательного смещения между двумя похожими наборами данных, представимых в виде двумерной матрицы.

Следующее изображение(рис.2) демонстрирует использование фазовой корреляции для определения относительного поступательного движения

между двумя изображениями, поврежденными независимым гауссовым шумом. Второе изображение было переведено на (30,33) пикселей по координатам x, y. Соответственно, можно ясно увидеть пик в представлении фазовой корреляции примерно на тех же координатах (30,33).

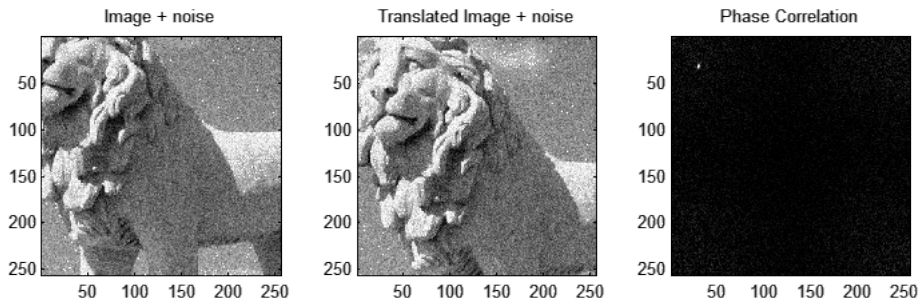


Рис. 2: Демонстрация результата фазовой корелляции двух изображений.

Расчет фазовой корреляции двух изображений представляет собой сложную многоэтапную задачу и основывается на преобразованиях Фурье, однако отличается скоростью относительно других корреляционых алгоритмов. Помимо того, метод фазовой корреляции устойчив к шумам и многим другим дефектам.

Определим функцию фазовой корреляции двух матриц как

$$F(A, B) = (x_0, y_0)$$

где x_0, y_0 - сдвиг между матрицами A и B по соответсвующим координатам.

Пусть нам даны изображения карт сосудов, сегментированных из двух соответсвующих им сетчаток, представленных в виде двоичных матриц A и B, где 1 соответствует сосуду, а 0 - его отсутствию.

Для оценки сдвига двух матриц, требуется нивелировать относительное вращения между ними. Пусть $P(M)$ - функция преобразования изображения, представленного в виде матрицы M из декартовых координат в полярные. Тогда

$$F(P(A), P(B)) = (r, \theta),$$

где θ - относительное вращение.

Выполнив разворот матрицы B на θ градусов получим матрицу B_{rot} и вычислим относительное смещение

$$(x_0, y_0) = F(A, B_{rot})$$

После перемещения матрицы B_{rot} на x_0, y_0 по осям x, y соответственно относительно ее изначального положения, получим матрицу B_{trans}. Показатель схожести между двумя матрицами определим как

$$similarity = \frac{nonZero(A\&B_{trans})}{nonZero(A|B_{trans})}$$

3.1.2 Алгоритм, основанный на углах Харисса

Детектор углов Харисса - это метод обнаружения точек резкого изменения контрастности изображения, который применяется компьютерного зрения для определения характеристик изображения.

Для ускорения работы детектора Харисса изображение из трехмерной матрицы RGB - формата переводят в двумерную, где цветом пикселя будет являться градация серого.

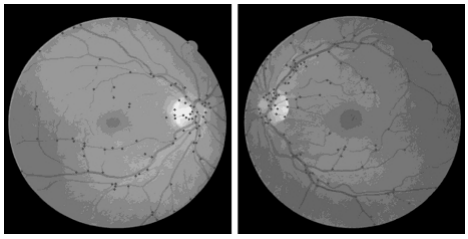


Рис. 3: Найденные для сетчатки углы Харисса

Перейдем к определению схожести двух матриц, для которых найдены их углы Харисса.
Пусть найдена N+1 точка, тогда для каждой j-й точки её декартовы координаты (x_i, y_i) преобразуются в полярные (r_i, θ_i) и определяется вектор признаков матрицы $v_j = (r_j, \theta_j)$, в котором

$$r_j = (r_{j1}, ...r_{jN})$$

$$\theta_j = (\theta_{j1}, ...\theta_{jN})$$

Модель подобия между неизвестным вектором v и вектором признаков точки u_i имеющих размер M в некоторой точке i определяется уравнением:

$$S_{vu_i} = \sum_{k=1}^{M} max(exp(-\alpha(v_s - u_i - u_k)^2))$$

где $v_s \in v$, и α – некоторая заданная постоянная.

Пусть вектор $(v_{1a}, ...v_{ka})$ – вектор признаков первого изображения, где v_{ja} имеет размер K–1, а вектор $(v_{1b}, ...v_{kb})$ – вектор признаков второго изображения, где v_{jb} имеет размер J–1. Определим функцию схожести векторов

$$\gamma(i) = max_{1 \leq j \leq min(J,K)} S(v_i, v_j)$$

тогда коэффициентом схожести данных изображений будет

$$similarity = \sum_{i=1}^{min(K,J)} \gamma(i)$$

3.1.3 Алгоритм, основанный на точках разветвления

Данный алгоритм, в отличии от предыдущего, более устойчив к шумам, но применяется для бинарных изображений. Первым этапом преобразования исходного изображения является поиск точек разветвления(бифуркации) и пересечения.

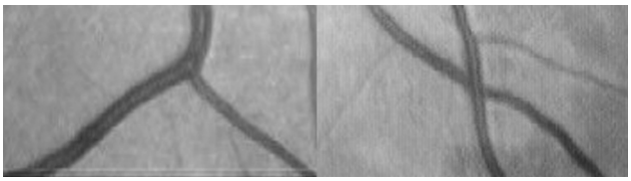


Рис. 4: Пример точек бифуркации и пересечения

Поиск точек разветвления начинается с нахождения нулевого участка любого сосуда на изображении.

Следующим шагом происходит анализ пикселей, которые ещё не были рассмотрены. Это приводит к 16 возможным вариантам (рис. 5). Если пиксель в середине окна не имеет соседей серого цвета, как показано на рис. 5 (a), то он отбрасывается и ищется другой пиксель кровеносных сосудов. В других случаях это либо конечная точка, либо внутренняя (не включая точки бифуркации и пересечения).

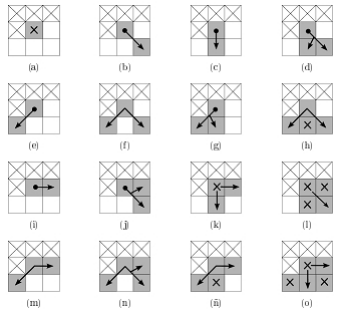


Рис. 5: 16 возможных конфигураций четырёх соседних пикселей (белые точки – фон, серые – сосуды). Три пикселя сверху и один слева уже были проанализированы, поэтому игнорируются. Также игнорируются серые с крестиком пиксели. Точки со стрелочкой внутри могут стать следующим центральным пикселем. Пиксели с чёрной точкой внутри – это конечные точки.

На каждом шаге сосед серого цвета последнего пикселя помечается как пройденный и выбирается следующим центральным пикселем в окошке 3 x 3. Выбор такого соседа определяется следующим критерием: наилучший сосед, имеющий наибольшее количество непомеченных серых соседей. Это связано со стремлением поддерживать сосуд толщиной в один пиксель.

Для 4 точек показателем схожести определяют 4 параметра - x, y, θ, S, где x, y - смещение, а θ, S - угол поворота и изменение масштаба, соответственно. Показатели определяются 2 векторами (x_a, y_a, θ_a) и (x_b, y_b, θ_b) образованными 2 точками A и B. Из них находим

$$\theta = \theta_a - \theta_b$$

$$S = \frac{v_i}{v_j}$$

$$x_0 = Sy_a sin\theta - Sx_a cos\theta + x_b$$

$$y_0 = Sx_a sin\theta - Sy_a cos\theta + y_b$$

Тогда показатель схожести между парами 4 точек β можно рассчитать как расстояния между векторами $u_1 = (x_{01}, y_{01}, \theta_1, S_1)$ и $u_2 = (x_{02}, y_{02}, \theta_2, S_2)$

$$\beta = distance(u_1, u_2)$$

Как я уже упомянул, для определения подобия используются пары контрольных точек(контрольных векторов). Оптимизация поиска контрольных точек является сложной с алгоритмической точки зрения задачей, поэтому, как правило, используется постой перебор. Пусть общее число контрольных точек на первом изображения равняется A, а на втором B. Тогда количество пар контрольных точек(контрольных векторов) на первом изображении равно $\frac{A^2-A}{2}$, а на втором $\frac{B^2-B}{2}$. Перебирая для каждого набора точек

возможное совпадение, получаем общее число вариантов выбора контрольных векторов

$$M = \frac{(A^2 - A)(B^2 - B)}{4}$$

Для каждого из M вариантов выбора набора векторов размером C $((u_1, v_1), ...(u_C, v_C))$, будем определить функцию схожести

$$Sim_m = \sum_{n=1}^{C} distance(u_n, v_n)$$

Итак, метрика схожести двух сетчаток определяется как

$$min(Sim_1, ...Sim_M)$$

В отличии от двух других алгоритмов, даже при оптимизации и больших допущениях данный алгоритм отличается высокой вычислительной сложностью. Приняв $A = B = 50$, получим $M = 1,562 * 10^6$. Такое число итераций даже для современных процессоров является сложной задачей и делает данный алгоритм применимым только с учетом оптимизации в виде намеренного отбрасывания некоторого числа точек разветвления.

3.2 Нейросетевые алгоритмы

Нейронная сеть в простейшем случае — математическая модель, состоящая их нескольких слоёв элементов, выполняющих параллельные вычисления.

Ключевыми понятиями, необходимыми для понимания прицнипа устройства нейронной сети являются синапсы и нейроны.

Базовые элементы искусственной нейронной сети также принято называть нейронами. Нейрон — это вычислительная единица, которая получает информацию, производит над ней простые вычисления и передает ее дальше.

Синапс - это связь между двумя нейронами. Каждый синапс обладает единственным параметром веса(рис. 6). Благодаря весу, входная информация изменяется, когда передается от одного нейрона к другому. Допустим, есть 3 нейрона, которые передают информацию следующему. Тогда у нас есть 3 веса, соответствующие каждому из этих нейронов. у нейрона, имеющего больший вес, выходная информация будет доминирующей в системе. Совокупность весов нейронной сети или матрица весов — это своеобразный мозг всей системы. Именно благодаря этим весам, входная информация обрабатывается и превращается в результат.

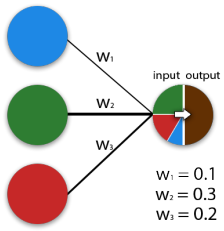
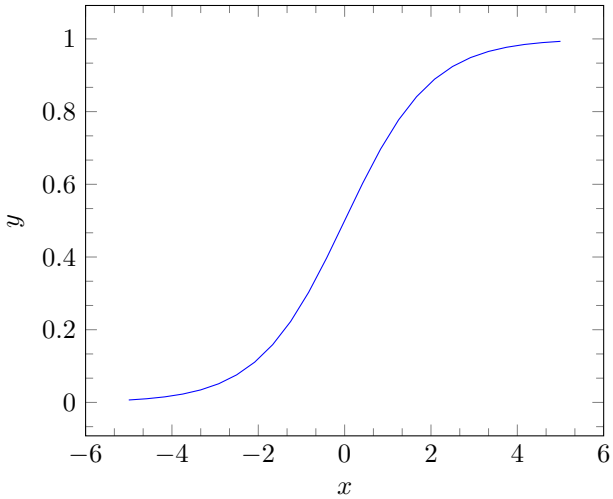


Рис. 6: Пример зависимости выходных данных от весов синапса.

Для нормализации выходных данных в нейросетях используются функции активации, повзоляющие получить из некоторых входных данных приемлемый ответ. Классическим примером функции активации можно считать функцию сигмоида $f(x) = \frac{1}{1+e^{-x}}$, использующуюся в задачах бинарной классификации.

Сигмоид



Область значений сигмоида $E = (0, 1)$, что позволяет представить некие входные данные в виде числа 0, 1 или неопределнности.

Нейронные сети, как правило, состоят из нескольких слоёв, которые можно разбить на внешний(входной), скрытые слои и выходной.

Одной их ключевых особенностей нейронной сети является её свойство обучаться. Обучением нейронной сети называется итеративный процесс, уменьшающий отклонение выхода сети от эталона. Обучение состоит из шагов, называемых эпохами обучения, на каждом из которых происходит оптимизация «весов» нейронной сети — параметров скрытых слоёв сети. Сеть обучается примерах, и их количество пропорционально правильности ее выхода. В задачах распознавания объектов примером будет являться пара вход-выход, состоящая из входного изображения и соответствующего

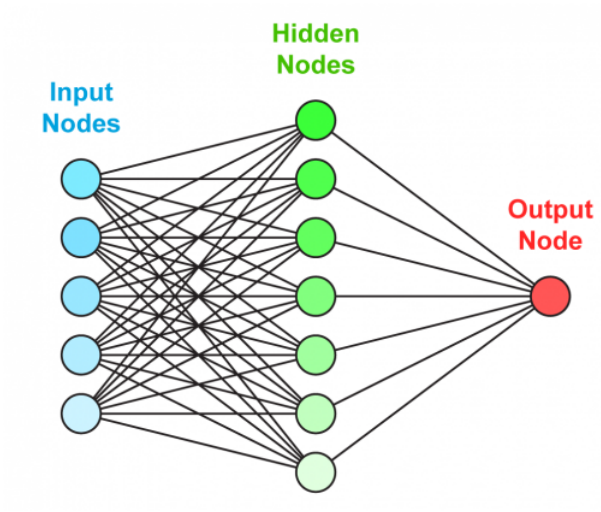


Рис. 7: Схематичный пример усройства простой сети

ему ярлыка — названия.

Глубокой нейронной сетью называют нейронную сеть с несколькими внутренними слоями. В вопросах идентификации предпочтение отдается именно глубоким, в особенности - сверточным нейросетям.

Сверточные сети используют в своей основе принцип фильтра - свёртки для выделения ключевых параметров изображения и преобразования их в более компактный вид для последующей оценки.

4 Реализация алгоритмов

4.1 Алгоритм, основанный на методе фазовой корреляции

Как наиболее понятный и точный, согласно данным исследования [2], для реализации был выбран именно этот алгоритм.

4.1.1 Извлечение рисунка сосудов

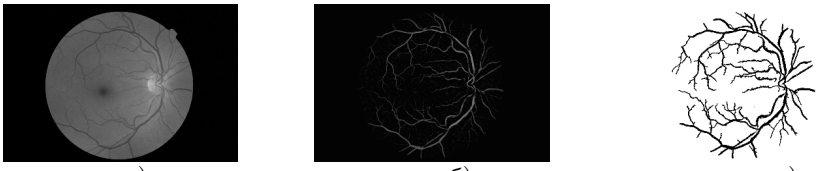
Для увеличения точности фазовой корреляции, применяется выделение ключевых свойств изображения и удаление лишних шумов. Уникальным элементом сетчатки является именно расположение кровенсных сосудов, поэтому перед непосредственным применением алгоритмики, описанной в пункте 3.1.1, исходное изображение проходит несколько этапов предобработки.

В ходе работы использовалась библиотека компьютерного зрения Opencv, а так же дистрибутив свободного научного и инженерного программного обеспечения Numpy

Первым этапом является перевод изображения из RGB-формата в градации серого и увеличение контрастности изображения путем выравнивания его гистограммы. Для этого применяется метод Clahe библиотеки Opencv.

На втором этапе происходит многоступенчатое удаление шумов и тела сетчатки с помощью морфологических преобразований. Далее, с помощью метода FindContours находятся границы перехода между сосудами и глазным дном, внутренние области сосудов закрашиваются в черный. После этого изображение переводится в двоичный формат путём отсеивания - так пиксели, чье значение в ходе преобразований стало ниже определнного порога принимают значение нуля, остальные принимают значение единицы.

Рис. 8: Оригинальное изображение



а) б) в)

Рис. 9: Процесс сегментации: а) Преобразавание в градации серого, б) удаление лишних элементов, в) повышение контрастности и выделение контуров сосудов.

Таким образом из изначального цветного изображения удается получить практически точную карту сегментированных кровеносных сосудов сетчатки.

4.1.2 Оценка сдвига изображения

Для оптимизации алгоритма, проводится многоэтапное преобразование изображений. Так, на первом этапе несколько раз выполняется поворот изображения на полученный в ходе оценки (п. 3.1.1) угол с последующим сдвигом на полученное относительное смещение. На втором этапе применяется обратный порядок действий: сначала сдвиг, и только потом поворот. На каждой итерации каждого этапа подсчитывается мера схожести, и среди

всех полученных показателей для данной пары изображений выбирается наибольший.

Такие меры предприняты для предотвращения возможных ситуаций 'ложного отказа', связанных с высокой чувствительностью метрик схожести. Так, для двух одинаковых изображений, имеющих сдвиг менее чем в 5% относительного общего числа пискелей по осям x и y, $similarity \approx 0,7$, что много ниже оцененного оптимальным в ходе экспериментов порога в 0,9 (см. приложение, рис.16).

4.1.3 Скорость выполнения

Очевидно, что итерационный процесс многократно увеличивает время выполнения программы, однако решением данной проблемы стало сжатие изображения в несколько раз по соответсвующим осям. Данный ход влияет на точность, но, если оценить сложность вычисления фазовой корелляции изначального изображения

$$O(xy)$$

и сложность вычисления фазовой корелляции сжатого в k раз изображения,

$$O(\frac{xy}{k^2})$$

становится явным выигрыш в k^2 раз. Скорость выполнения всех этапов алгоритма для изображения размером 960 на 720 пикселей $= 2 \pm 1,5$ секунд, что является приемелемым, но довольно долгим результатом. В ходе экспериментов было получено оптимальное значение $k = 3$, позволяющее сохранить относительню точность, но значительно ускорить время работы программы.

Решение можно протестировать и ознакомиться с его исходным кодом, перейдя по ссылке в приложениях.

4.2 Нейросетевой алгоритм

4.2.1 Алгоритм работы сверточных сетей

Как уже упоминалось в пункте 3.2, наиболее распространнеными в вопросе классификации и сравнения изображений являются именно сверточные нейронные сети. Свертка изображения помогает выделить характерные для изображения свойства. На рисунках 10 и 11 схематично демонстрируется принцип свертки выского уровня.

Для построения и обучения сети была выбрана библиотека Keras. Keras — открытая нейросетевая библиотека, написанная на языке Python. Она представляет собой надстройку над фреймворками Deeplearning4j, TensorFlow и Theano. Итоговая архитекутра нейронной сети состоит 6 внутренних слоев (3 слоя свертки и 3 слоя пулинга - подвыбора, см.приложение рис.13).

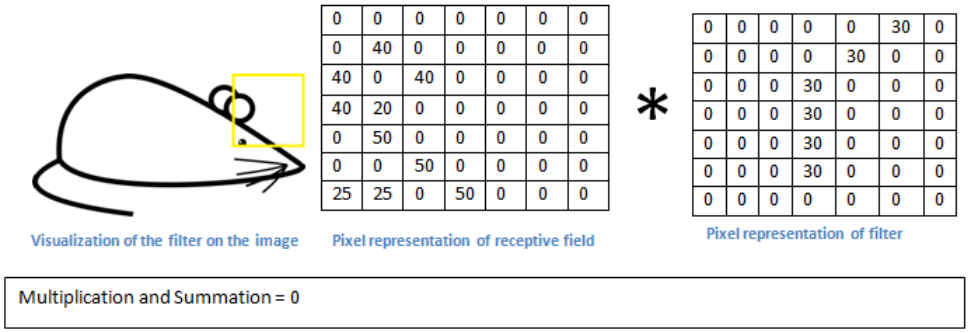


Рис. 10: Окно фильтра проходит по всему изображению и значения фильтра умножаются на значения пикселей области, выделенной окном

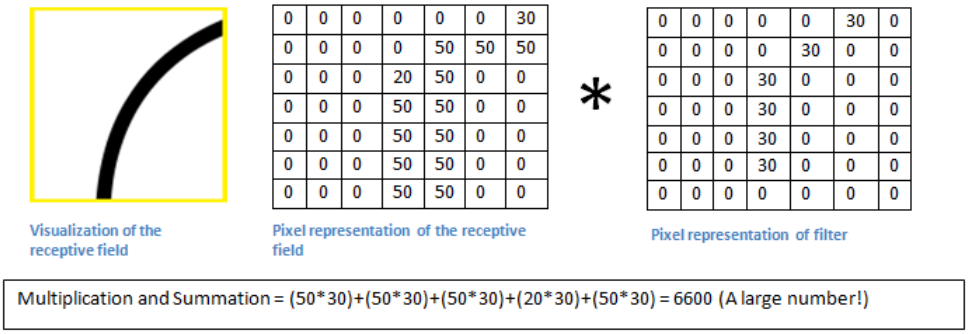


Рис. 11: В случае удачного нахождения кривой, фильтр действительно выдает высокие значения.

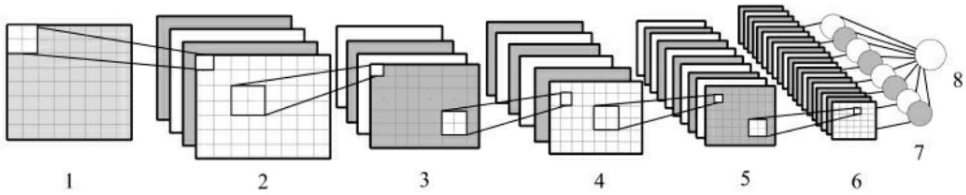


Рис. 12: Наглядное представление слоёв сети: 1 - вход; 2, 4, 6 - свёрточные слои; 3, 5 - подвыборные слои(пулинга); 7 - слой обычных нейронов; выход.

4.2.2 Обучение сети

Для обучения нейронной сети классификации даже простых обьектов, требуется зачастую несколько тысяч или даже больше изображений для каж-

дого примера. Стоит заметить, что даже человек может опознать на глаз далеко не каждую сетчатку, и в ходе исследования так и не было найдено достаточно большой базы данных для полноценного обучения, поэтому для расширения имеющихся наборов данных была применена аугментация - намеренная деформация и искажение данных как попытка предсказать и описать возмонжые входные данные.

Для повышения точности изображения предварительно были заменены соответсвующими им сегментированными картами сосудов.

Для обучения модели на начальном этапе было выбрано и дополнено с помощью аугментации несколько наборов изображений. Количество примеров для каждого человека составило 1000 бинарных изображений сосудов глазного дна размера 128 на 128 пикселей.

4.2.3 Переобучение

Из-за относительно малого входного набора в ходе обучения возник крайне сильный эффект переобучения.

Переобучение в машинном обучении и статистике — явление, когда построенная модель качественно оперирует с обучавшим её набором, но теряет способность к обобщению относительно плохо работает на примерах, не участвовавших в обучении.

Для борьбы с переобучением был использован слой Dropout библиотеки Keras.

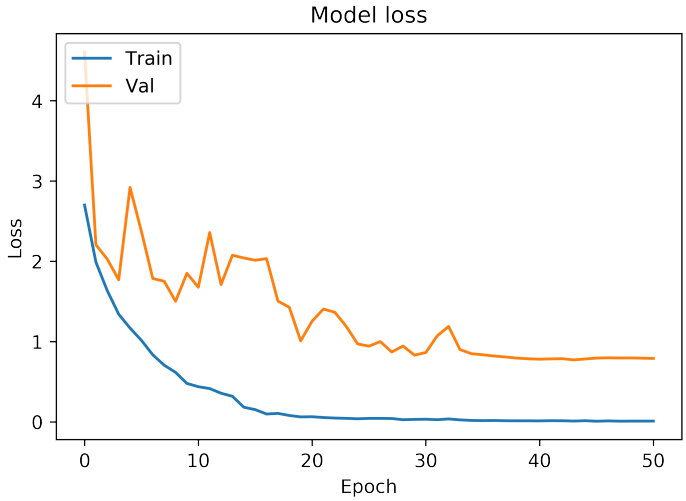


Рис. 13: Здесь Val loss - доля ошибки в тестирующей выборке, Train loss - на обучающей. Видно, что уже на первых итерациях точность работы на тренирующей выборке становится больше точности на тестовой, так как модель запоминает все возможные комбинации входных данных.

5 Оценка точности

5.1 Точность корреляции

5.1.1 Зависимость точности от искажения изображения

В ходе тестирования были использованы открытые данных MESSIDOR и RIDB. Для повышения разнообразия входных данных была использована намеренная деформация входных данных. Итоговый рамзер тестовых данных составил 2000 изображений. Для наглядности был введен коэффициент деформации

$$K = \sqrt{\frac{1}{2}}\sqrt{(\frac{shift_{max}}{max(img_x, img_y)})^2 + (\frac{rot_{max}}{45})^2}$$

где $shift_{max}$ - маскимальное отклонение по осям x и y от изначального изображения по соответствующим осям, (img_x, img_y) - размеры исходного изображения по осям x, y, а rot_{max} - маскимально возможный угол поворота. Значения же сдвига и угол поворота вычислялись как случайное число в интервалах $(-shift_{max}, shift_{max})$ и $(-rot_{max}, rot_{max})$ соответственно.

Угол 45 градусов был приянт как наибольший возможный ввиду практической невозожности допущения подобного отклонения в реальной жизни.

Полученные графики точности наглядно демонстрируют особенности алгоритма(см. приложение).

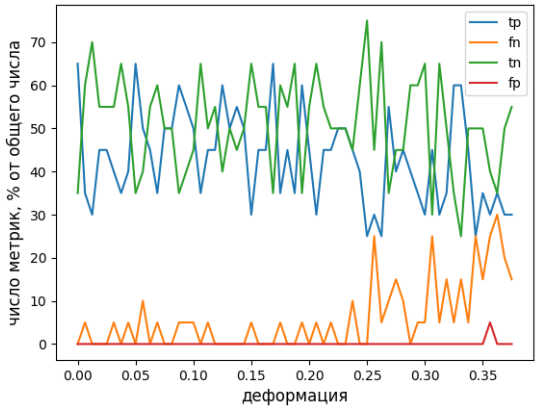


Рис. 14: Зависимость точности от деформации изображения

Здесь метрики tp, tn, fp, fn отражают правильно распознанное совпадение(True Positive), правильно распознанное несовпадение(True Negative), ложный отказ(False Negative) и ложный допуск(False Positive) соответственно.

Важными для оценки работы алгоритмов идентификации метриками также являются accuracy(общая доля верных ответов), precision(точность отсеивания неправильных) и recall (точность распознавания правильных).

$$accuracy = \frac{tp + tn}{tp + tn + fp + fn}$$

$$precision = \frac{tp}{tp + fp}$$

$$recall = \frac{tp}{tp + fn}$$

Как пример, для графика изображенного на рис. 14 получаем:

$$tp = 50,9\%; tn = 43,4\%; fn = 5,6\%; fn = 0,1\%$$

$$accuracy = 0,943; precison = 0,998; recall = 0,901$$

5.1.2 Способы увеличения точности

Одним из действенных способов увлечения точности алгоритма является последовательный перебор всех возможных углов поворота от и до определенного значения и оценка сходства между первым и вторым повернутым на данный угол изображениями.

Пусть N - количество итераций. Тогда время работы программы увеличится в N раз. Обращаясь к пункту 4.1.3, видно, что при N = 20, время выполнения составит около 4 секунд. Разумеется, это довольно много, зато такой способ гарантирует практически абсолютную точность.

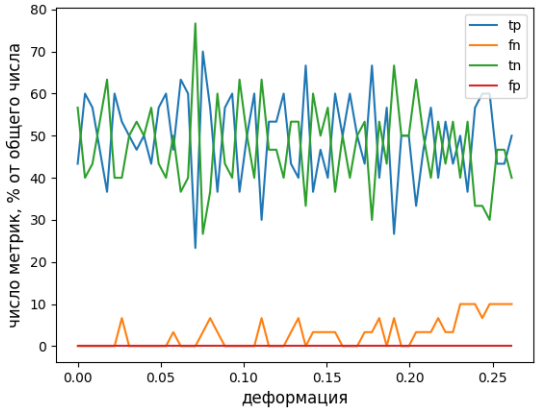


Рис. 15: Метрики полученные с помощью переборного алгоритма.

$$tp = 47,5\%; tn = 49,7\%; fn = 2,8\%; fn = 0\%$$

$$accuracy = 0,972; precison = 1; recall = 0,945$$

5.2 Точность нейронной сети

5.2.1 Зависимость точности от количества эпох

При обучении нейронной сети было выявлено оптимальное количество эпох $E = 30$, после которого достигается практически абсолютная точность, что видно из графика рис.13.

Полностью обученная модель на наборе данных с коэффициентом деформации $K = 0,4$ (см. п. 5.1.1) при классификации сетчаток 3 различных людей демонстрирует следующие показатели:

$$accuracy = 0.995; precison = 1.; recall = 0.991$$

6 Вывод

Исследование и оценка двух разных походов дали свои результаты. Даже не имея в распоряжении первоклассных баз данных и выичслительных мощностей оказалось возможным создать рабочую и довольно точную систему идентификации.

Оба решения продемонстрировали свою эффективность и являются жизнеспособными, пусть и не сравнятся со всемирно признанными промышленными алгоритами. И всё же, на взгляд авторов, именно классические алгоритмы являются более приоритетеными для дальнейшего развития. Этому способствует как отсутствие необходимости обучения модели решения, так и относительная легковесность и простота корректировки. С другой стороны, для опытного человека создание грамотной нейросетевой модели - тривиальная задача, более простая, чем реализация, например, алгоритма, основаной на фазовой корелляции.

7 Приложение

Ссылка на репозиторий с готовым к применению консольным приложением:

`https://github.com/kxlll/Retinal_identification`

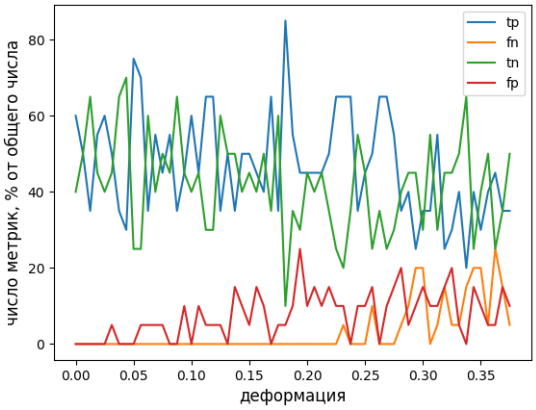


Рис. 16: Результат снижения допустимого порога схожести с 0,9 до 0,85. Заметно серъезное увеличение количества метрик fp и уменьшение метрик fn.

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 128, 128, 32)	896
activation (Activation)	(None, 128, 128, 32)	0
batch_normalization (BatchNo	(None, 128, 128, 32)	128
max_pooling2d (MaxPooling2D)	(None, 64, 64, 32)	0
dropout (Dropout)	(None, 64, 64, 32)	0
conv2d_1 (Conv2D)	(None, 64, 64, 64)	18496
activation_1 (Activation)	(None, 64, 64, 64)	0
batch_normalization_1 (Batch	(None, 64, 64, 64)	256
conv2d_2 (Conv2D)	(None, 64, 64, 64)	36928
activation_2 (Activation)	(None, 64, 64, 64)	0
batch_normalization_2 (Batch	(None, 64, 64, 64)	256
max_pooling2d_1 (MaxPooling2	(None, 32, 32, 64)	0
dropout_1 (Dropout)	(None, 32, 32, 64)	0
flatten (Flatten)	(None, 65536)	0
dense (Dense)	(None, 512)	33554944
activation_3 (Activation)	(None, 512)	0
batch_normalization_3 (Batch	(None, 512)	2048
dropout_2 (Dropout)	(None, 512)	0
dense_1 (Dense)	(None, 3)	1539
activation_4 (Activation)	(None, 3)	0

Total params: 33,615,491
Trainable params: 33,614,147
Non-trainable params: 1,344

Рис. 17: Точные характеристики каждого из слоёв сети.

Список литературы

[1] Biometric retina identification based on neural network / Fahreddin Sadikoglu, Selin Uzelaltinbulat // 12th International Conference on Application of Fuzzy Systems and Soft Computing, ICAFS 2016, 29-30 August 2016, Vienna, Austria. 2013.

[2] A Novel Retinal Identification System / Mohammad-Shahram Moin // Iran Telecommunication Research Center. 2008

[3] Методы аутентификации по сетчатке глаза / Максим Нафиков // habr.com/ru/post/261309 . 2015

[4] Retina Identification Database RIDB/ data.mendeley.com/datasets/tjw3zwntv6/1

[5] MESSIDOR Databse / www.adcis.net/en/third-party/messidor

[6] DRIVE: Digital Retinal Images for Vessel Extraction // drive.grand-challenge.org

[7] InceptionV3 для распознавания изображений / Максим Титков // habr.com/ru/post/321834

[8] A Mathematical Theory of Deep Convolutional Neural Networks/ Thomas Wiatowski and Helmut Bolcskei // Fellow, IEEE

[9] Официальная документация Keras // keras.io