электронный журнал
МОЛОДЕЖНЫЙ НАУЧНО-ТЕХНИЧЕСКИЙ ВЕСТНИК
Издатель ФГБОУ ВПО "МГТУ им. Н.Э. Баумана". Эл No. ФС77-51038.

УДК 004.93

Разработка программной системы для идентификации личности на контрольно-пропускном пункте

Кузнецов В.В., *студент*
Россия, 105005, г. Москва, МГТУ им. Н.Э. Баумана,
кафедра «Системы обработки информации и управления»

Научный руководитель: Филиппов М.В.,к.т.н., доцент
Россия, 105005, г. Москва, МГТУ им. Н.Э. Баумана,
irudakov@bmstu.ru

Введение

Практически все современные системы контроля и управления доступом, используемые на российских предприятиях, регистрируют сам факт прохода человека, владеющего электронным пропуском, без идентификации личности самого человека. Таким образом, по пропуску может пройти любой человек, что серьезно ставит под угрозу безопасность.

Проблема состоит в том, что на данный момент нет универсального набора алгоритмов качественно решающих задачу определения личности. Связано это с тем, что при съёмке камерой видеорегистрации поступающий видеопоток подвергается различным искажениям, например, перспективным преобразованиям, связанным с перемещением камеры либо объектов в кадре, изменениям освещённости и зашумлением. Также меняется сам человек, его прическа, одежда, аксессуары.

Для решения данного вопроса предлагается создать программный продукт, идентифицирующий личность человека на КПП, используя видеопоток от камеры видеорегистрации.

Выбор алгоритмов для решения задачи

Существующие алгоритмы обнаружения лиц можно разбить на две широкие категории. К первой категории относятся методы, отталкивающиеся от опыта человека в распознавании лиц и делающие попытку формализовать и алгоритмизовать этот опыт, построив на его основе автоматическую систему распознавания. Вторая категория опирается на инструментарий распознавания образов, рассматривая задачу обнаружения лица, как частный случай задачи распознавания.

Основа методов первой категории - эмпирика, является одновременно их сильной и слабой стороной. Большая изменчивость объекта распознавания, зависимость вида лица на изображении от условий съемки и освещения позволяют без колебаний отнести обнаружение лица на изображении к задачам высокой сложности. Применение эмпирических правил позволяет построить некоторую модель изображения лица и свести задачу к выполнению некоторого количества относительно простых проверок. Однако, несмотря на безусловно разумную посылку - попытаться использовать и повторить уже успешно функционирующий инструмент распознавания - человеческое зрение, методы первой категории пока далеки по эффективности от своего прообраза, поскольку исследователи, решившие избрать этот путь, сталкиваются с рядом серьезных трудностей. Во-первых, процессы, происходящие в мозгу во время решения задачи распознавания изображений изучены далеко не полностью, и тот набор эмпирических знаний о человеческом лице, которые доступны исследователям на "сознательном уровне", далеко не исчерпывает инструментарий, используемый мозгом "подсознательно". Во-вторых, трудно эффективно перевести неформальный человеческий опыт и знания в набор формальных правил, поскольку чересчур жесткие рамки правил приведут к тому, что в ряде случаев лица не будут обнаружены, и напротив, слишком общие правила приведут к большому количеству случаев ложного обнаружения.

Можно перечислить следующие проблемы, общие для методов второй категории:

1. Зависимость от ориентации и масштаба лица. Большинство классификаторов не являются инвариантными к повороту лица в плоскости изображения и изменению его размера. Поэтому для успешного обнаружения лица, отличного по размеру или ориентации от лиц в тренировочном наборе, требуется дополнительная обработка входного изображения (масштабирование, поворот). Проблему изменения масштаба решают, обычно, путем полного перебора всех возможных прямоугольных фрагментов изображения всех возможных размеров. Попытка же рассматривать еще и все возможные углы поворота лиц в плоскости изображения приведет к тому, что время выполнения и без того долгой процедуры перебора фрагментов превысит все мыслимые пределы. Если говорить о повороте головы вне плоскости изображения, то это является проблемой для многих методов из обоих категорий, поскольку при значительном повороте лицо на изображении изменяется настолько сильно, что многие признаки и правила (заданные разработчиком или полученные неявно) распознавания фронтального изображения лица становятся совершенно непригодными.

2. Неявный способ определения признаков для распознавания лица таит в себе потенциальную опасность: классификатор, обладающий недостаточно репрезентативным набором изображений лиц, теоретически может выделить вторичные или ложные признаки в качестве важных. Одно из следствий - потенциальная зависимость от освещения, которое преобладало в тренировочном наборе. В ряде случаев применяется дополнительная предобработка изображения для компенсации влияния освещения.

3. Высокая вычислительная сложность. Во-первых, сами классификаторы часто включают в себя большое количество достаточно сложных вычислений; во-вторых, полный перебор всех возможных прямоугольных фрагментов изображения сам по себе занимает большое количество времени. Это затрудняет использование некоторых методов в системах реального времени (например - отслеживании перемещения лица в видеопотоке).

Сравнивать между собой качество распознавания методов разных категорий достаточно тяжело, поскольку в большинстве случаев, опираться можно только на данные испытаний, предоставляемые самими авторами, поскольку провести крупномасштабное исследование по реализации большинства известных методов и сравнения их между собой на едином наборе изображений не представляется возможным по причине невообразимой трудоемкости этой задачи.

На основе информации, предоставляемой авторами методов, также сложно провести корректное сравнение, поскольку проверка методов часто производится на разных наборах изображений, с разной формулировкой условий успешного и неуспешного обнаружения. К тому же проверка для многих методов первой категории производилась на значительно меньших наборах изображений.

Заметное различие между первой и второй категорией описанных методов заключается еще и в том, что эмпирические методы часто довольно просты в реализации (особенно относительно методов второй категории), и предоставляют возможность гибкой настройки под конкретную задачу путем модификации интуитивно понятных параметров. Методы, опирающиеся на инструментарий распознавания образов, требуют значительных усилий по формированию тренировочных наборов изображений и обучению классификатора. Влияние параметров, контролирующих классификатор, на его поведение часто далеко неочевидно. Однако трудоемкость создания работающих прототипов методов второй категории частично компенсируется высокими заявленными показателями качества распознавания на больших коллекциях изображений.

Что касается рекомендаций по выбору метода для решения задачи обнаружения лиц, то можно сказать, что выбирать подходящий метод, исходя из цифровых показателей

качества распознавания вряд ли целесообразно. Скорее, все зависит от конкретной задачи и условий в которых должен функционировать разрабатываемый алгоритм. Построение универсального метода, обеспечивающего высокий уровень распознавания при отсутствии ограничений на исходные изображения в настоящее время не представляется возможным, однако для большинства конкретных задач можно создать методы, предоставляющие достаточный уровень распознавания.

В качестве условий, влияющих на выбор метода решения задачи, можно перечислить следующие:

– предполагаемое разнообразие лиц: ограниченный набор людей, ограничения на возможный тип лица (раса, присутствие растительности на лице, очков и т.д.), отсутствие ограничений;

– ориентация лиц на изображении: строго вертикальная (или наклон под известным углом), в определенных границах вблизи известного угла наклона, любая;

– цветное или черно-белое изображение;

– масштаб лиц, разрешение и качество изображения (зашумленность, степень сжатия);

– предполагаемое количество лиц, присутствующих на изображении: известно, примерно известно, неизвестно;

– условия освещения: фиксированные известные, приблизительно известные, любые;

– фон: фиксированный, контрастный однотонный, слабоконтрастный зашумленный, неизвестный;

– что важнее - не пропустить ни одного лица или минимизировать количество случаев ложного обнаружения?

Взвесив все за и против было принято решение использовать метод Виолы-Джонса для обнаружения лица на кадре видео потока

Разработанный метод идентификации

Разработанный в данной статье метод идентификации лица содержит следующие шаги:

1. поиск лица (Виола-Джонс);
2. выделение особых точек (SIFT);
3. поиск соответствий между особыми точками (kd-дерево);
4. оценка модели (RANSAC).

1. Поиск лица

На первом этапе применяется метод Виола-Джонса для поиска лица на кадре с камеры видеорегистрации. Он является самым лучшим по соотношению качества обнаружения и скорости.

Данный метод основывается на 3-ёх ключевых принципах. Во-первых, он использует специальное представление изображения, называемое «интегральным изображением», которое позволяет быстро вычислять характеристики его элементов. Во-вторых, применяется специальный обучающий алгоритм, который позволяет выделить небольшое количество ключевых характеристик среди всех возможных. В-третьих, применяется метод, позволяющий использовать сложные классификаторы каскадно, тем самым давая возможность быстро отбрасывать объекты фона и заострять внимание на те объекты, которые наиболее похожи на искомые.

2. Определение особых точек

На втором этапе определяются инвариантные характеристики двух изображений лиц путём выделения особых точек и их дескрипторов. Особая точка $\boldsymbol{m}$– это точка изображения, окрестность которой $\boldsymbol{o(m)}$ можно отличить от окрестности любой другой точки изображения $\boldsymbol{o(n)}$ в некоторой другой окрестности особой точки $\boldsymbol{o_2(m)}$. Дескриптор – это идентификатор особой точки, выделяющий её из остального множества особых точек.

Для обнаружения особых точек используется модифицированный алгоритм преобразования масштабно-инвариантных характеристик (англ. ***Scale Invariant Feature Transform***, ***SIFT***). Данный алгоритм позволяет определять особые точки в виде капель (англ. ***blobs***), так как они инвариантны ко всем преобразованиям (к аффинным преобразованиям, изменениям освещённости, положению камеры и к шуму). Капли – это структуры, описываемые координатами центра, масштабом и направлением. Подобного вида структуры являются самыми сложными, высокоуровневыми среди всех видов форм особых точек, и поэтому обеспечивают устойчивое их обнаружение. Алгоритм ***SIFT*** показывает лучшие результаты обнаружения особых точек по качеству чёткости и контрастности изображения, чем другие алгоритмы обнаружения капель, и подходит к предметной области. Недостатком ***SIFT*** является его вычислительная сложность, что ограничивает его применение в режиме постобработки.

Алгоритм ***SIFT*** разделяется на 5 частей:

1. построение пирамиды гауссиан и их разностей. На этом шаге обеспечивается инвариантность к масштабированию;

2. определение экстремумов (рис. 1 б);

3. уточнение особых точек (рис. 1 в);

4. определение ориентаций особых точек (обеспечивается инвариантность к повороту) (рис. г);

5. построение дескрипторов (обеспечивается инвариантность к освещению, шуму, изменению положения камеры).

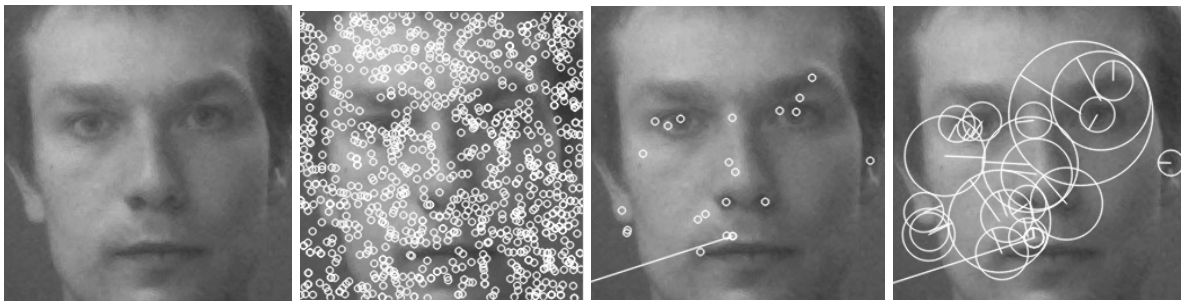

Рис. 1 а, б, в, г. Этапы выделения особых точек

2.1. Построение пирамиды гауссиан и их разностей

На первом шаге алгоритма ***SIFT*** строится масштабируемое пространство изображения – набор изображений, сглаженных фильтром Гаусса

2.2. Определение экстремумов

По ним строится разность гауссиан $\boldsymbol{D(x,y,\sigma)}$– попиксельное вычитание изображений в одной октаве с разным коэффициентом размытия. Октава – изображение в одном масштабе, размытое фильтром Гаусса (4 изображения в одной октаве). На этом шаге обеспечивается инвариантность к масштабированию. Затем определяются экстремумы, которые заносятся в список потенциальных особых точек.

2.3. Уточнение особых точек

Далее происходит уточнение особых точек, которое состоит из двух составляющих:

1. исключаются точки с малой контрастностью с помощью вычисления экстремума разности гауссиан. Разность гауссиан раскладывается многочленом Тейлора второго порядка, взятого в точке вычисленного экстремума;

2. исключаются граничные точки (точки, имеющие большой локальный изгиб вдоль границы и малый в перпендикулярном направлении).

2.4. Определение ориентаций особых точек

Этот шаг направлен на назначение соответствия ориентации особым точкам на основе локальных свойств изображения. Шаг обеспечивает инвариантность к повороту. Выполняются следующие действия:

1. Направление особой точки вычисляется исходя из направлений градиентов соседних точек. Вычисления градиентов производятся на изображении в пирамиде гауссианов, с масштабом наиболее близким к масштабу особой точки. Величина m и направление θ градиента в точке (x, y) вычисляются, используя разности пикселей:

$$m(x,y) = \sqrt{\left(L(x+1,y) - L(x-1,y)\right)^2 + (L(x,y+1) - L(x,y-1))^2},$$

$$\theta(x,y) = \arctan\left(\frac{L(x,y+1) - L(x,y-1)}{L(x+1,y) - L(x-1,y)}\right),$$

где $L(x, y)$ – значение яркости в точке (x, y).

2. Определить окно (окрестность) особой точки, в котором будут рассмотрены градиенты. Окно является круглым, поскольку формируется свёрткой гауссова ядра с радиусом размытия $\sigma = 1{,}5$ масштаба особой точки. Для гауссова ядра действует правило «трёх сигм»: значение гауссова ядра очень близко к нулю на расстоянии, превышающем 3σ. Таким образом, радиус окна определяется как 3σ.

3. Найти направление особой точки из гистограммы направлений. Гистограмма состоит из 36 компонент, которые равномерно покрывают промежуток в 360 градусов: каждая точка окна (x, y) вносит вклад, равный $m * G(x, y, \sigma)$, в ту компоненту гистограммы, которая покрывает промежуток, содержащий направление градиента $\theta(x, y)$.

4. Найти направление особой точки. Направление находится в промежутке, покрываемом максимальной компонентой гистограммы. Значения максимальной (max) и двух соседних компоненты интерполируются параболой. Точка максимума параболы берётся в качестве направления особой точки. Если в гистограмме есть ещё компоненты с величинами не меньше $0{,}8 * \max$, то они аналогично интерполируются и создаются дополнительные особые точки с найденными направлениями.

2.5. Построение дескрипторов

На конечном шаге алгоритма ***SIFT*** для окрестности особой точки вычисляются изменения яркостей точек, по которым строится дескриптор. Дескриптор – это вектор из 64 чисел, позволяет получить инвариантность относительно положения камеры. Затем дескриптор нормализуется, за счёт чего достигается инвариантность относительно изменения освещения.

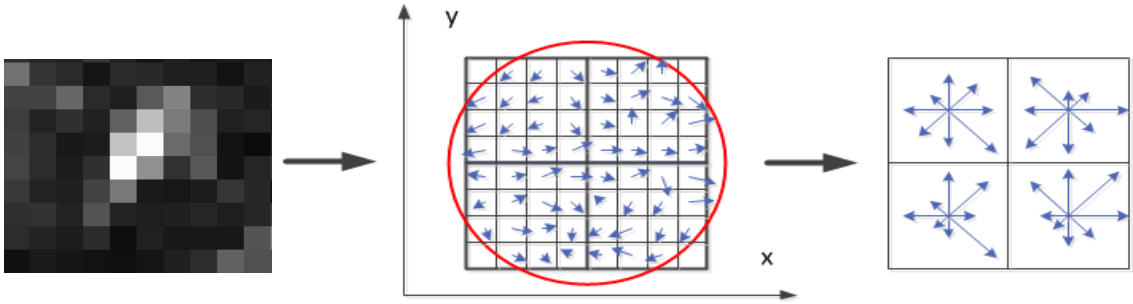

Рис. 2. Построение дескриптора

На рисунке 5 показана часть изображения (слева), схематично показаны направления градиентов (в середине) и полученный на их основе дескриптор (справа). На изображении в середине показаны пиксели, обозначенные квадратами. Эти пиксели берутся из квадратного окна дескриптора, которое в свою очередь поделено на четыре равных части (регионы). Стрелка в центре каждого пикселя обозначает градиент этого пикселя. Центр окна находится между пикселями. Центр выбирается как можно ближе к точным координатам особой точки. Круг обозначает окно свертки с гауссовым ядром (аналогично окну для вычисления направления особой точки). Для ядра определяется значение σ, равное половине ширины окна дескриптора. Значение каждой точки окна дескриптора умножается на значение гауссова ядра в этой точке, как на весовой коэффициент. Использование гауссова ядра позволяет избежать скачков в дескрипторе с малыми изменениями положения окна и меньшее увеличить значения градиентов, расположенных на краях окна, так как на них больше всего влияют ошибки неточного совмещения. При расчёте дескриптора изображение не поворачивается, а поворачиваются координаты градиента на угол, соответствующий ориентации квадрата.

Справа изображен дескриптор особой точки размерностью 2x2x8. Первые две цифры в значении размерности обозначают количество областей по горизонтали и вертикали. Те квадраты, которые охватывали пиксели на левом изображении, справа охватывают гистограммы, построенные на пикселях этих областей. Третья цифра в размерности дескриптора означает количество компонент гистограммы. Гистограммы в

областях вычисляются так же, как и гистограмма направлений. Учитываются следующие особенности:

1. каждая гистограмма покрывает участок в 360 градусов и делит его на 8 частей;
2. в качестве весового коэффициента берется значение гауссова ядра, общего для всего дескриптора;
3. в качестве весовых коэффициентов берутся коэффициенты трилинейной интерполяции, чтобы распределить значение каждого выбранного значения градиента в прилегающие компоненты гистограммы.

Каждому градиенту в окне дескриптора соответствуют три координаты (x, y, n), где x – расстояние до градиента по горизонтали, y – расстояние по вертикали, n – расстояние до направления градиента в гистограмме дескриптора, в которую вносит вклад этот градиент. Точка отсчёта – левый нижний угол окна дескриптора и начальное значение гистограммы. Единичный отрезок – размеры областей по горизонтали и вертикали, для x и y соответственно, и количество градусов в компоненте гистограммы для n. Коэффициент трилинейной интерполяции определяется для каждой координаты (x, y, n) градиента как $1 - d$, где d равно расстоянию от координаты градиента до середины единичного отрезка, в котором находится данная координата. Каждое вхождение градиента в гистограмму умножается на три весовых коэффициента трилинейной интерполяции.

Дескриптор особой точки состоит из всех полученных гистограмм. Показано, что лучшие результаты получаются при использовании дескриптора размерности 128 компонент (4x4x8). Для получения инвариантности относительно освещения (яркости и контрастности) полученный дескриптор нормализуется к единичной длине. Этим приёмом обеспечивается инвариантность линейного изменения. Устойчивость к нелинейным изменениям освещения (например, при резком изменении погодных условий) определяется относительными величинами градиентов и в меньшей степени их направлениями, поэтому все компоненты дескриптора, значение которых больше 0,2, сокращаются до этого значения, после чего дескриптор нормализуется к единичной длине ещё раз (Lowe, D.G..2004. Distinctive Image Features from Scale-Invariant Keypoints. January 5, 2004). Затем вещественные компоненты дескриптора преобразуются в целочисленные в диапазон [0; 255].

3. Нахождение соответствий

В результате формирования множества особых точек на двух изображениях возникает проблема выявления соответствий между этими множествами. Необходимо найти соответствия между особыми точками, которые лежат в перекрывающихся областях изображений. Поиск соответствий между изображениями ведётся на основе дескрипторов особых точек. Обозначим два множества особых точек V_1 и V_2 соответственно, с дескрипторами в 128-мерном пространстве (точками). Расстояние двумя точками в декартовых координатах в n-мерном пространстве определяется как:

$$d(u,v) = \sqrt{\sum_{i=1}^{n}(u_i - v_i)^2},$$

где u_i, v_i – компоненты точек u и v соответственно.

Пусть дана исходная точка поиска $p \in V_1$. Ближайшим соседом для точки p является такая точка $q \in V_2$, что для любой другой точки $q' \in V_2$ выполняется

$$d(p,q) < d(p,q'),$$

где $d(p,q)$ – расстояние между двумя дескрипторами особых точек.

Задача поиска ближайшего соседа (англ. nearest neighbor search, NNS) состоит в нахождении ближайшего соседа для произвольной исходной точки поиска p.

Для выполнения поиска используется kd-дерево (от англ. k-dimensional tree, k-мерное дерево) – это структура, предназначенная для хранения конечного множества точек k-мерного пространства, в данном случае 128-мерного. Kd-дерево является разновидностью бинарного дерева, каждый узел которого содержит следующие поля:

- массив особых точек;
- признак, является ли узел дерева листом;
- разделяющую размерность, по которой происходит разделение (индекс массива дескрипторов);
- левое и правое kd-поддеревья, содержащие точки, находящиеся по разные стороны разделяющей медианы.

Необходимо разбить всё пространство на блоки (гиперпространства) и искать в этих блоках в порядке возрастания расстояния до точки p. Блок – это область пространства, которая может разбиваться на более мелкие блоки, в зависимости от количества точек, попавших в данный блок. Блоки получаются разделением пространства на два подпространства, перпендикулярно делящей гиперплоскости. Более формально: пусть $d_{min}(p,B)$ обозначает минимальное расстояние от точки p до любой точки блока B.

$d_{min}(p, B) = 0$, если точка лежит внутри блока B. Функция $d_{min}(p, B)$ считается как расстояние от точки p до блока B.

Деление массива особых точек относительно дескрипторов производится следующим образом:

1. Покомпонентно находится дисперсия (от 1 до 128).
2. Выбирается компонента с максимальной дисперсией.
3. По всем дескрипторам по этой компоненте находится медиана, относительно которой производится деление. Медиана играет роль гиперплоскости, делящей гиперпространство на 2 гиперподпространства – блока.
4. Узел дерева делит массив особых точек относительно дескрипторов таким образом, что в левое поддерево попадают точки, у которых значения компоненты меньше или равны медианы.

Использование медианы позволяет получить сбалансированное kd-дерево.

Построение kd-дерева сводится к следующей рекурсивной подпрограмме, блок-схема алгоритма которой показана на рисунке 6. Деление производится в каждом листе относительно полученных множеств дескрипторов до тех пор, пока не останется одного дескриптора.

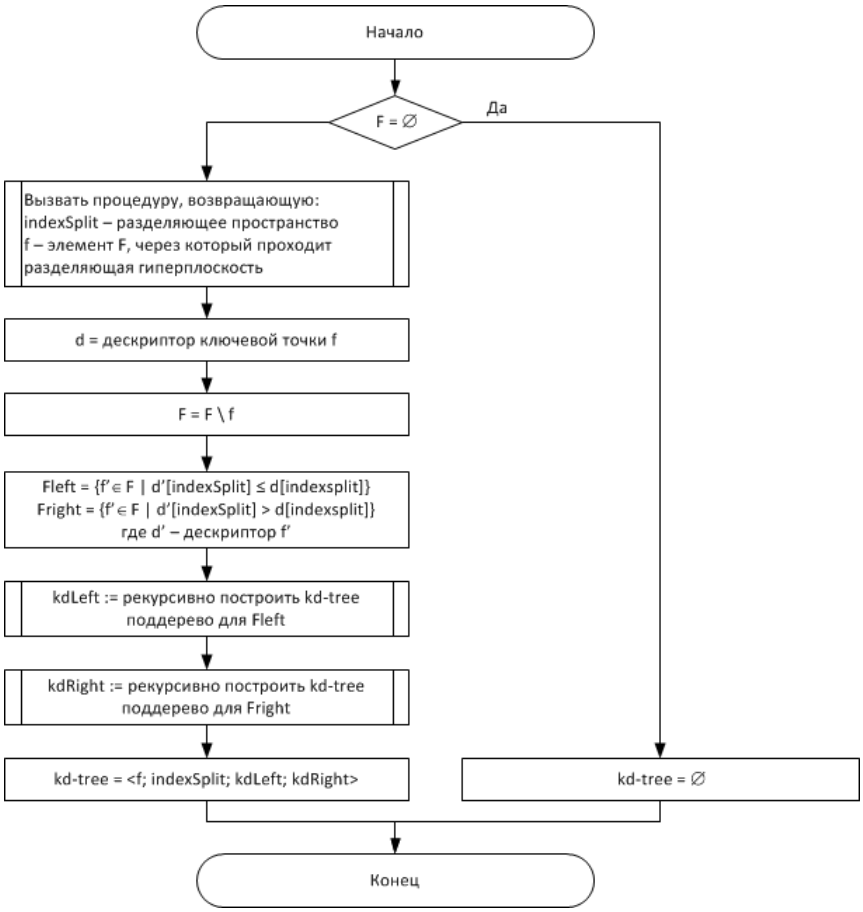

Рис. 3. Блок-схема алгоритма построения kd-дерева

По построенному дереву происходит поиск ближайшего дескриптора из множества V_2. Алгоритм ищет в иерархической структуре точек блок, который содержит точку p.

Находится расстояние R_{min} до единственного дескриптора, содержащегося в найденном блоке. Потенциально в гиперсферу с радиусом R_{min} могут попасть другие точки, если расстояния от искомой точки до ближайшего блока меньше, чем расстояние до соседней (данная ситуация отражена на рисунке 6 для двумерного пространства). В этом случае все соседние с данным блоком являются те, которые лежат выше по дереву. Поднимаясь вверх по дереву, блоки, встречающиеся на пути поиска, помещаются в очередь с приоритетом. В очереди блоки сортируются в соответствии с функцией $d_{min}(p, B)$. Алгоритм посещает блоки в порядке уменьшения расстояния от дочки p. Если блок соответствует внутреннему узлу дерева поиска, тогда все соответствующие ему блоки помещаются в очередь. Если блок является терминальным (листом kd-дерева), то вычисляется расстояние от p до каждой точки блока. Расстояние до ближайшей точки хранится в переменной l_{min}. Алгоритм заканчивает свою работу, когда расстояние до очередного блока, извлеченного из очереди, станет больше, чем l_{min}. Алгоритм NNS является оптимальным в том смысле, что он посещает только те блоки, которые пересекаются с гиперсферой радиуса R_{min} (расстояние до ближайшего соседа) и центром в исходной точке p.

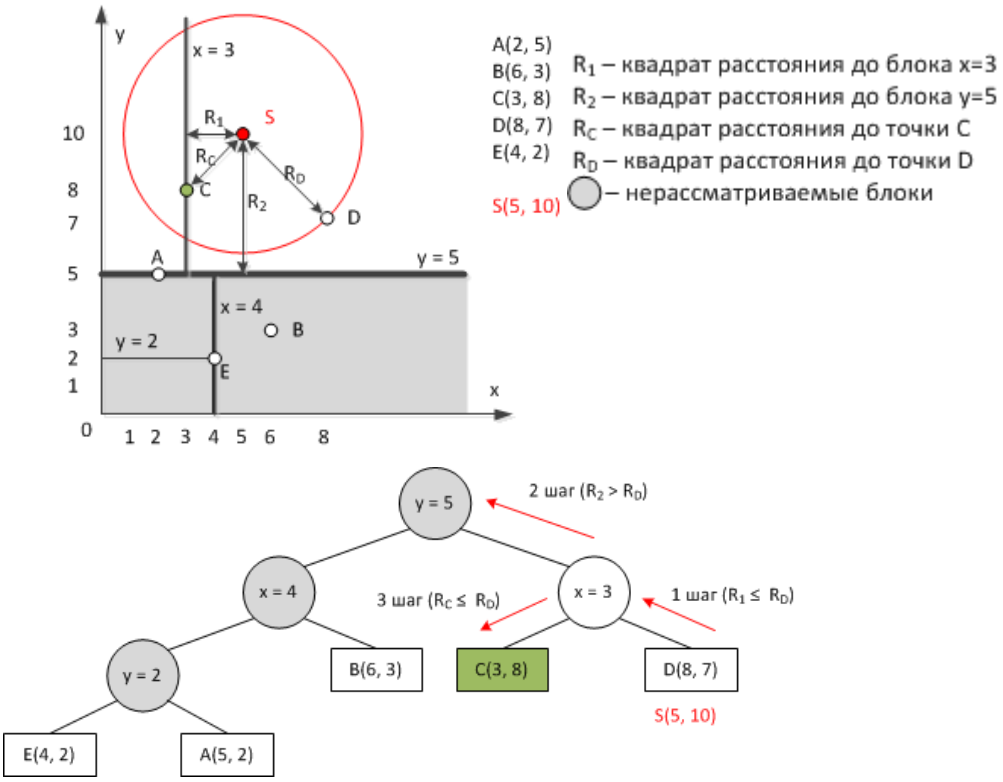

Рис. 4. Построение kd-дерева и поиск для двумерного пространства

Для двумерного пространства построение kd-дерева для пяти точек A(2, 5), B(6, 3), C(3, 8), D(8, 7), E(4, 2) и поиск ближайшей точки для S(5,10) отображён на рисунке 7. В нижней части рисунка 7 показано построенное kd-дерево и последовательность обхода при поиске ближайшей точки. В результате поиска ближайшей точкой является точка C(3, 4). Графическая иллюстрация расположения блоков и точек показана сверху на рисунке 7. В двумерном пространстве гиперплоскостью является прямая, а гиперсферой – окружность.

Оценка модели

Полученная последовательность может иметь ложные соответствия. Для удаления ложных соответствий применяется метод согласования случайных выборок ***RANSAC*** (англ. ***Random Sample Consensus***) – итерационный, вероятностный метод, в котором определяется минимальная симметрическая погрешность между парами точек, после чего вычисляются коэффициенты матрицы перспективного преобразования ***H*** размерности ***3x3*** (матрица гомографии) методом прямого линейного преобразования ***DLT*** (англ. ***Direct Linear Transformation***). Решается система линейных алгебраических уравнений, в результате определяются 8 коэффициентов (параметров ***DLT***), показывающих связь между системами координат плоскостей двух изображений.

После выбора лучшей модели отображения как результат мы имеем лучшее соответствие между входящим кадром с изображением лица и изображением лица сотрудника из базы данных.

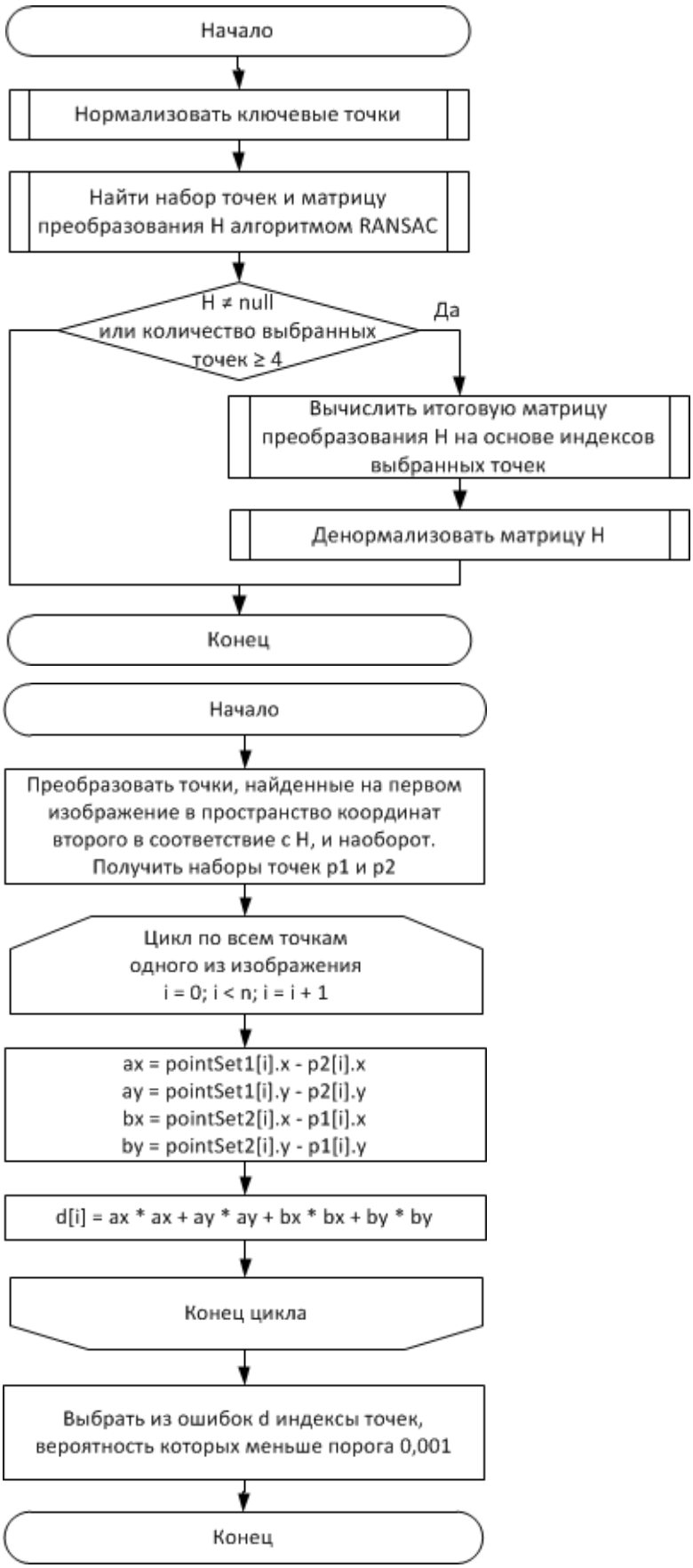

Рис. 5. Блок-схема алгоритма нахождения гомоморфизма и определения точек с минимальной симметрической погрешностью

Заключение

В представленной работе был произведён:

- анализ предметной области,
- анализ и выбор алгоритмов идентификации лица;
- анализ алгоритмов выделения особых точек;
- разработан метод идентификации лица и его программная реализация (с ограничением на одно лицо одновременно присутствующее в кадре);
- произведено тестирование разработанного ПО.

К основным направлениям развития следует отнести:

- модификация алгоритмов для ускорения работы и более тесного взаимодействия между компонентами;
- интегрирование программного продукта в систему контроля и управления доступом на основе электронных карт;
- увеличение числа лиц одновременно присутствующих в кадре.

Список литературы

1. Brown M., Lowe D.G. Automatic Panoramic Image Stitching using Invariant Features // International Journal of Computer Vision. 2007. Vol. 74. No. 1. PP. 59-73. DOI: 10.1007/s11263-006-0002-3;
2. Конушин А.С. Слежение за точечными особенностями сцены (Pointfeaturetracking) // Компьютерная графика и мультимедиа. Электрон. журн. 2003. № 1(5). Режим доступа: http://cgm.computergraphics.ru/content/view/54 (дата обращения 07.06.2012);
3. Juan L., Gwun O. A Comparison of SIFT, PCA-SIFT and SURF // International Journal of Image Processing (IJIP). 2009. Vol. 3. No. 4. P. 143-152;
4. Tran P. Pyramid Matching Using SURF and SIFT Descriptors for SVM Classification. UniversityofKansas. Режим доступа: http://www.ittc.ku.edu/~potetz/EECS_841_Fall09/Lectures/Final%20Projects/Paper_04.pdf (дата обращения 12.06.2012);
5. Valgren C., Lilienthal A.J. SIFT, SURF & Seasons: Appearance-based Long-term Localization in Outdoor Environments // Robotics and Autonomous Systems. 2010. Vol. 58. No. 2. PP. 149–156. DOI: http://dx.doi.org/10.1016/j.robot.2009.09.010;
6. Meng Y., Tiddeman B. Implementing the Scale Invariant Feature Transform (SIFT) Method. Citeseer. Режим доступа: http://www.cs.st-andrews.ac.uk/~yumeng/yumeng-SIFTreport-5.18_bpt.pdf (дата обращения 10.06.2012);

7. Гонсалес Р., Вудс Р. Цифровая обработка изображений. Пер. с англ. под ред. П.А. Чочиа. М.: Техносфера, 2005. 1072 с.
8. Kybic J., Vnucko I. Approximate Best Bin First k-d Tree All Nearest Neighbor Search with Incremental Updates. Czech Technical University in Prague. Режим доступа: ftp://cmp.felk.cvut.cz/pub/cmp/articles/kybic/Kybic-TR-2010-10.pdf (дата обращения 10.06.2012).
9. Hartley R.I., Zisserman A. Multiple View Geometry in Computer Vision. 2nd ed. Cambridge: Cambridge University Press, 2003. 672 p.