

# Об автоматическом распознавании лиц

М. П. Волченков, И. Ю. Самоненко

Процесс идентификации личности по изображению лица состоит из двух задач: первая — это выделение лица на изображении, вторая — распознавание. Сканирование исходного изображения окном меньшего размера — задача вычислительно трудная, как правило, решаемая перебором окон с разным расположением и размером. Вторая задача может быть решена различными методами: статистическими, нейросетевыми, эластичных графов, характерных точек и др. В статье приводится обзор методов решения второй задачи. Приведен оригинальный алгоритм и результаты его работы для задачи слежения за лицами в видеопотоке.

## 1. Введение

Автоматическая идентификация личности человека по его изображению на фотографии или в видеопотоке имеет широкое коммерческое и научное применение. Данная тематика появилась в начале 80-х годов, однако, ее бурное развитие началось в 90-х, после создания новых технологий в сфере обработки изображений и вычислительных машин. Данная технология имеет особый интерес в связи с тем, что может происходить бесконтактно.

В общем случае, задача идентификации личности состоит из двух этапов. Первый — это определение местоположения лица на изображении. Для этого исходное изображение сканируют окном меньшего размера, и каждый раз определяют некоторую степень схожести изображения в окне с человеческим лицом. Этот этап наиболее вычислительно трудоемкий, так как требуется проводить полное сканирование для разных размеров окна, а так же каждый раз находить

степень схожести изображения в окне с лицом. Формально, изображение лица может задаваться структурно (лицо — это овал, на котором в середине расположен нос, симметрично глаза, и т. д.), по цвету кожи (если фон имеет цвет, отличный от цвета кожи), статистически и списком примеров изображений лиц.

После того, как выбрано окно, про которое с большой достоверностью можно сказать, что оно содержит только лицо человека, начинается идентификация лица. Для этого применяется набор алгоритмов: статистических, нейросетевых, марковских цепей, эластичных графов, вейвлет анализ, анализ характерных точек и др. В современных системах, как правило, эти методы используются в совокупности.

## 2. Методы идентификации

Пусть дано изображение, про которое априори известно, что оно содержит только лицо человека.

### 2.1. Метод главных компонент

Метод главных компонент (Principal Component Analysis — PCA) [1] является статистическим методом. По сути, в нем нигде не используется, что объектом обработки являются изображения лиц. Он оперирует с векторами в некотором линейном пространстве.

Пусть  $I$  — черно-белое изображение лица  $Ver$  на  $Hor$  пикселей, то есть

$$I \in \text{Mat}(Ver \times Hor).$$

Будем считать, что  $I \in \mathbb{R}^N$ , где  $N = Ver \cdot Hor$ . Требуется определить некоторое подпространство  $U \subset \mathbb{R}^N$ , размерности много меньшей  $N$ , при проекции на которое потеря информации изображения  $I \in \mathbb{R}^N$  будет минимальной. Более точно. Пусть обучающая выборка лиц  $\Omega = \{I_1, \dots, I_n\}$ ,  $I_i \in \mathbb{R}^N$ . Математическое ожидание  $E = E(\Omega) = \frac{1}{n} \sum_{i=1}^n I_i$ . Определим матрицу

$$A = [I_1 - E, \dots, I_n - E] \in \text{Mat}(N \times n),$$

и корреляционную матрицу

$$C = \frac{1}{n} \sum_{i=1}^n (I_i - E)(I_i - E)^t = \frac{1}{n} AA^t \in \text{Mat}(N \times N).$$

Требуется найти правые собственные вектора матрицы  $C$  (они те же, что и у матрицы  $D = nC$ ). Это сложно сделать напрямую при больших значениях  $N$ , поэтому используется другой путь.

Пусть  $v$  — собственный вектор и  $\lambda$  — соответствующее ему собственное значение, тогда

$$Dv = \lambda v \Rightarrow AA^t v = \lambda v \Rightarrow A^t AA^t v = \lambda A^t v,$$

следовательно,  $y = A^t v$  — собственный вектор с собственным значением  $\lambda$  для матрицы  $A^t A \in \text{Mat}(n \times n)$ , где  $n$  всего лишь число лиц в обучающей выборке  $\Omega$ , а не размер изображения  $N$ .

Пусть  $\{r_1, \dots, r_k\}$  — первые  $k$  векторов ( $k \leq n$ ), отвечающих наибольшим различным собственным значениям матрицы  $A^t A$ , то есть

$$A^t A r_i = \lambda_i r_i \Rightarrow AA^t A r_i = \lambda_i A r_i \Leftrightarrow AA^t v_i = \lambda_i v_i \Leftrightarrow Dv_i = \lambda_i v_i,$$

где  $v_i = A r_i$ . Следовательно,  $v_i$  являются собственными векторами матрицы  $D$ . Заметим, что каждый  $v_i$  есть линейная комбинация векторов  $I_1 - E, \dots, I_n - E$ , то есть линейной комбинацией исходных лиц  $I_1, \dots, I_n$ . В связи с чем,  $v_i$  имеют лицеподобный вид, их часто называют собственными лицами (eigenfaces), (см. рис. 1).

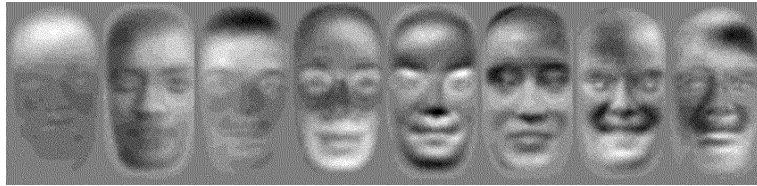


Рис. 1.

Произвольность обучающей выборки  $\Omega$ , как правило, позволяет считать, что  $rk(A) = n$  (то есть максимальный ранг) и все вектора  $r_1, \dots, r_k$  — линейно независимы. Следовательно, вектора

$v_i = Ar_i$  ( $i = 1, \dots, k$ ) тоже линейно независимы и образуют подпространство  $U = \text{Lin}\langle v_1, \dots, v_k \rangle$  размерности  $k$ . Более того, вектора  $v_i$  — ортогональны. В самом деле: так как  $\lambda_i(v_i, v_j) = (AA^t v_i, v_j) = (v_i, AA^t v_j) = (v_i, AA^t v_j) = \lambda_j(v_i, v_j)$ , где  $(\cdot, \cdot)$  — скалярное произведение. В таком случае, если  $\lambda_i \neq \lambda_j$ , то  $(v_i, v_j) = 0$ .

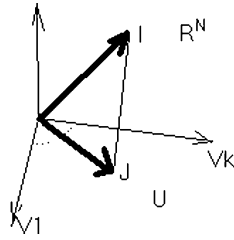


Рис. 2.

Любой вектор  $I = (i_1, \dots, i_N)^t$  ортогонально проецируется в подпространство  $U$ .  $J = \text{Pr}_U(I) = (j_1, \dots, j_k)^t \in \mathbb{R}^k$ ,  $j_i = (I, v_i)$  (см. рис. 2). В силу максимальности собственных значений, отвечающих  $v_i$ , набор чисел  $j_1, \dots, j_k$  характеризуют случайную величину  $I$  по наиболее значимым параметрам, и в силу ортогональности  $v_i$  ( $i = 1, \dots, k$ ), эти параметры независимы. Теперь изображение любого лица представляется вектором в  $k$ -мерном пространстве, где  $k$  много меньше  $N$ .

Степень различия между двумя лицами определяется как  $\rho(A, B) = \sum a_i b_i$ , где  $A = (a_1, \dots, a_k)$ ,  $B = (b_1, \dots, b_k)$  — проекция изображений двух лиц. Если два изображения похожи, то различие между их проекциями мало, поэтому, изображения одного человека определяют некоторую область в  $U$  (см. рис. 3).

Для функционирования алгоритма распознавания требуется определить подпространство  $U$ . При этом,  $\Omega$  должна содержать по возможности наибольшую выборку различных изображений лиц. Следует отметить, что все лица должны быть в одном положении, например в фас, а также изображения должны быть одинакового размера —  $Ver$  на  $Hor$  пикселей.

Пусть  $\{\Omega_i, (i = 1, \dots, K)\}$ .  $\Omega_i = \{I_1^i, \dots, I_{n(i)}^i\}$  множество изображений  $i$ -ого человека. Находим  $\Lambda_i = \text{Pr}_U(\Omega_i) = \{J_1^i, \dots, J_{n(i)}^i\}$ , где

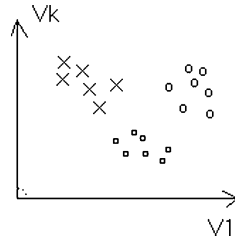


Рис. 3.

$J_t^i = \text{Pr}_U(I_t^i - E)$ . Если требуется распознать неизвестное изображение  $A$ , то находим  $B = \text{Pr}_U(A - E)$ . Затем, среди векторов  $\{J_1^i, \dots, J_{n(i)}^i\}$  ( $i = 1, \dots, K$ ) находим  $J_m^t$  ближайший к  $B$ . Если расстояние между  $J_m^i$  и  $B$  не превосходит некоторого порогового значения  $D_{\min}$ , то считаем, что вектор  $B$  принадлежит классу  $t$ , и  $A$  — это изображение лица с номером  $t$ . Если расстояние между  $J_m^i$  и  $B$  больше  $D_{\min}$ , то считаем, что человека с изображением  $A$  нет. Пороговое значение  $D_{\min}$  устанавливается эмпирически.

Эффективность распознавания зависит от размерности  $U$ , то есть от количества собственных лиц, от качества фотографий: разрешения, освещенности, расположения лица на фотографии, и т. д. Необходимо иметь одинаковое освещение и положение головы на всех фотографиях, хотя допустимы небольшие отклонения: очки, небольшие повороты головы, улыбки и т. п.

## 2.2. Скрытые марковские модели

Скрытые Марковские модели (Hidden Markov Models — HMM) моделируют случайные процессы, в частности эффективно применяются в распознавании речи. Впервые идея использования скрытой одномерной Марковской модели (1D-HMM) была предложена Samaria [4].

Скрытой Марковской моделью называется тройка  $\lambda = (A, B, \pi)$ .

Здесь  $S = \{s_1, \dots, s_N\}$  — множество состояний модели,  $V = \{v_1, \dots, v_K\}$  — выходной алфавит,  $\pi = (p_i) \in \text{Mat}(1 \times N)$  вектор начальных вероятностей, здесь  $p_i$  — вероятность того, что система в начальный момент времени  $t = 1$  находится в состоянии

$s(1) = s_i \cdot \sum_{i=1}^N p_i = 1$ .  $A = (a_{ij}) \in \text{Mat}(N \times N)$  — матрица перехода состояний,  $a_{ij} = P(s(t) = s_j | s(t-1) = s_i)$ ,  $\sum_{i=1}^N a_{ij} = 1$  для всех  $i = 1, \dots, N$ ,  $B = (b_i(k)) \in \text{Mat}(N \times K)$  — матрица появлений символов,  $b_i(k) = P(o(t) = v_k | s(t) = s_i)$ ,  $o(t)$  — выходной символ в момент  $t$ ,  $\sum_{k=1}^K b_i(k) = 1$  для всех  $i = 1, \dots, N$ .

В начальный момент времени  $t = 1$ , модель находится в некотором состоянии  $s(1)$ ,  $s(1)$  выбирается случайно по распределению  $\pi$ , то есть  $P(s(1) = s_i) = p_i$ . На выходе модели появляется символ  $o(1)$  из алфавита  $V$  согласно распределению  $B$ , то есть

$$P(o(1) = v_k) = b_i(k).$$

Если в момент  $t-1$  модель находится в состоянии  $s_i$ , и ее выходная последовательность это  $O(t-1) = \{o(1), \dots, o(t-1)\}$ , то в момент  $t$  модель перейдет в состояние  $s_j = s(t)$  с вероятностью

$$P(s(t) = s_j | s(t-1) = s_i) = a_{ij}.$$

На выходе модели появится символ  $v_k = o(t)$  с вероятностью

$$P(o(t) = v_k | s(t) = s_i) = b_i(k).$$

Получим выходную последовательность  $O(t) = \{o(1), \dots, o(t)\}$ . Процесс останавливается, когда модель переходит в некоторое финальное состояние  $s^* \in S$ , либо, когда текущее время  $t$  становится больше некоторой константы  $T$ .

Аналогично определяется Марковская модель в случае, когда выходные символы принадлежат некоторому бесконечному непрерывному множеству. Тогда функция распределения выходных символов непрерывна. В этом случае в качестве функции распределения часто берут Гауссовую смесь, то есть

$$b_i(O) = \sum_{k=1}^K c_{ik} N(O, m_{ik}, U_{ik}),$$

где  $N(O, m_{ik}, U_{ik})$  — гауссиан с математическим ожиданием  $m_{ik}$ , и ковариационной матрицей  $U_{ik}$ .

В задаче распознавания лиц используются одномерные и псевдо-двумерные линейные модели. Суть одномерной линейной модели заключается в том, что в ней невозможны переходы в состояния с меньшим номером (см. рис. 4).

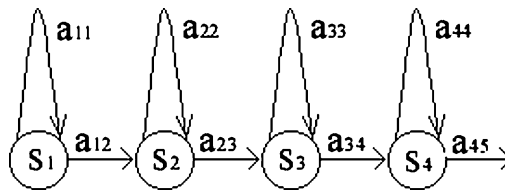


Рис. 4.

Будем рассматривать изображения, на которых лица только в «фас», и строго заключены в границы изображения. У модели будет 5 состояний: волосы, лоб, глаза, нос и рот. Изображения просматривается сверху вниз, и модель последовательно переходит из одного состояния в другое.

Пусть изображение имеет размеры (в пикселях)  $Ver$  по вертикали и  $Hor$  по горизонтали. Пусть размер окна просмотра  $L$  по вертикали. Это окно будет пробегать изображение сверху вниз, накрывая предыдущее окно на  $P$  пикселей (см. рис. 5). Тогда число блоков, на которое разбивается изображение равно  $T$ .

$$T = \frac{Ver - L}{L - P} + 1.$$

Следует отметить важность выбора параметров  $L$  и  $P$ . Большое значение  $P$  улучшает качество распознавания. Но выбор  $L$  требует осторожности. При большом значении  $L$  возможен захват изображений из разных состояний, при малых значениях, возможно, что окно будет содержать слишком мало информации. В [5] использовалась база лиц Olivetty Research Ltd (ORL). Эта база содержит 400 черно-белых фотографий, 40 человек по 10 фотографий. Изображения размера 92 по горизонтали и 112 по вертикали пикселей. Значения  $L$  и  $P$  предлагалось  $L = 10$ ,  $P = 9$ .

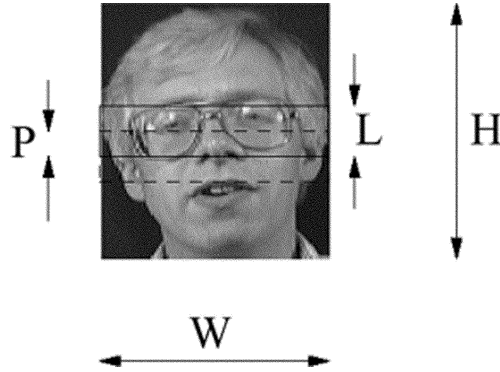


Рис. 5.

Другая проблема состоит в том, что вектор окна  $V \in \mathbb{R}^{L \cdot H_{or}}$  очень чувствителен к слабым изменениям исходного изображения. В связи с этим, используются различные преобразования  $V$ . В [5] предложено двумерное дискретное косинус преобразование (2D-DCT). При 2D-DCT изображение рассматривается как двумерный сигнал. В общем виде 2D-DCT  $C(x, y)$  функции  $F(x, y)$  на области  $[0, M] \times [0, N]$  представляется следующим образом:

$$C(x, y) = A(x)A(y) \sum_{i=1}^M \sum_{j=1}^N F(i, j) \cos\left(\frac{(2i-1)x\pi}{2M}\right) \cos\left(\frac{(2j-1)y\pi}{2N}\right),$$

где  $A(x)$  — нормирующий множитель.

Значения  $C(x, y)$  при малых значениях  $|x| + |y|$  отвечают низким частотам, при больших значениях — высоким. Как показано [5] вся высокочастотная составляющая всех пяти участков лица почти равна нулю, поэтому в [5] от преобразования лишь 39 коэффициентов, лежащих в окне  $3 \times 13$ . Они отвечают низким частотам. Эта операция позволяет сократить размерность и оставить только существенную информацию. Но для преобразования вектора  $V$  можно использовать не только 2D-DCT, см., например, [6].

Итак, для каждого изображения из базы надо построить свою Марковскую модель. Для этого изображение разбивается на состояния, и инициализируется матрица  $B$ . Затем используется алгоритм



Баума-Уэлша (Baum-Welsh Algorithm) для настройки параметров  $A$ ,  $B$  и  $\pi$ . После того, как для каждого изображения  $I_k$  из базы данных настроена Марковская модель  $\lambda_k$ , распознавание неизвестного изображения  $I$  проводится следующим образом: на каждую модель  $\lambda_k$  подается изображение  $I$ , и с помощью алгоритма прямого-обратного хода (Forward-backward Algorithm) вычисляется значение  $P(I|\lambda_k)$  — вероятности появления  $I$  в модели  $\lambda_k$ . Изображение  $I$  максимально похоже на  $I_t$ , где  $t = \operatorname{argmax}_k P(I|\lambda_k)$ .

Более эффективное распознавание обеспечивают псевдо-двумерные линейные Марковские модели (P2D-НММ). Такая модель состоит из линейной модели с суперсостояниями. Каждое суперсостояние является отдельной линейной Марковской моделью (см. рис. 6). Переход в новое суперсостояние возможен только тогда, когда система находится в финальном состоянии модели из предыдущего суперсостояния. По изображению пробегает окно размером  $L \times R$  пикселей. Суперсостояния отвечают разбиению изображения на столбцы, а последовательные переходы по состояниям внутри суперсостояния — проходу окна сверху вниз по данному столбцу.

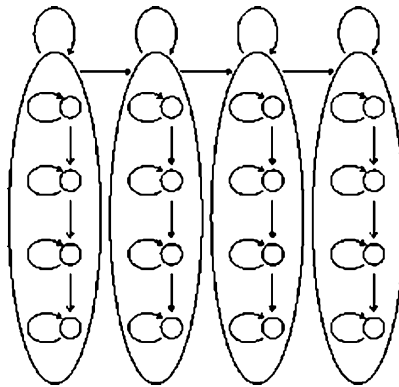


Рис. 6.

В [6] для базы (ORL) значения  $L$  и  $R$  предлагались  $L = 12$ ,  $R = 8$ . Окна идут с перекрытием 75% (см. рис. 7). На выходе появляется вектор  $O \in \mathbb{R}^{L \cdot R}$ .

Как и в случае 1D-НММ, вектор  $O$  сильно реагирует на слабые

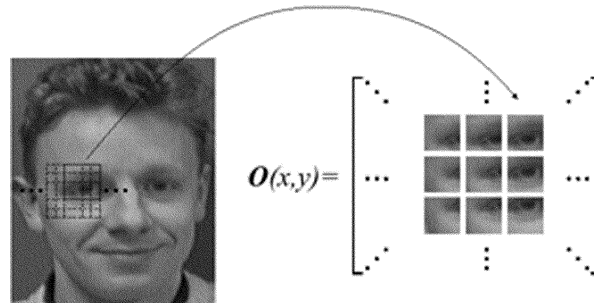


Рис. 7.

изменения изображения. Поэтому, в [6] предлагается использование 2D-DCT вектора  $O$ . Обучение и распознавание происходит по аналогии с 1D-НММ.

В алгоритме может варьироваться структура P2D-НММ: число суперсостояний и число состояний в каждом суперсостоянии. В [6] наилучшее распознавание было получено при использовании модели  $6 \times 6$ . Так же может меняться число гауссианов в Гауссовой смеси.

Заметим, что НММ позволяет сегментировать изображение лица. Для этого можно использовать алгоритм Витерби (Viterbi Algorithm). Он позволяет, при фиксированных  $U = o_1 \dots o_T$  — последовательности наблюдений и Марковской модели  $\lambda$ , найти последовательность  $Q = q_1, \dots, q_T$ , такую, что  $Q = \underset{W}{\operatorname{argmax}}(P(U|W, \lambda))P(U|Q, \lambda)$  — вероятность появления в  $\lambda$  последовательности  $U$  при проходе по состояниям  $W = w_1, \dots, w_T$ . На рис. 8 приведен пример сегментации, полученный с помощью P2D-НММ в [6].

Как показывают тесты, P2D-НММ являются эффективным средством для распознавания лиц. В [7] было предложено использование псевдо-трехмерных Марковских моделей для распознавания выражения лица.

### 2.3. Вейвлеты Габора (Gabor Wavelet)

Вейвлет (всплеск) анализ появился в функциональном анализе для исследования локального поведения функций [10]. Он широко

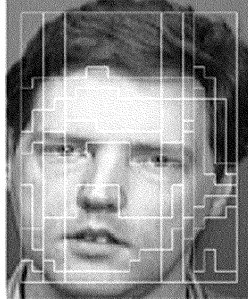


Рис. 8.

применяется при анализе различных сигналов. В [11] рассмотрены применения вейвлетов в компьютерной графике. Вейвлет преобразование — это свертка исходного изображения с некоторой функцией  $\Psi$  — вейвлетом. Двумерный вейвлет Габора  $\Psi_k(x)$  это:

$$\Psi_k(x) = \left( \frac{\|k\|^2}{\sigma^2} \right) \cdot \exp\left( \frac{-\|k\|^2 \cdot \|x\|^2}{2\sigma^2} \right) \cdot \left( \exp(ikx) - \exp\left( \frac{-\sigma^2}{2} \right) \right),$$

где  $k = (\omega \cos \theta, \omega \sin \theta)$ ,  $i$  — мнимая единица,  $x \in \mathbb{R}^2$ .

Обычно рассматривают вещественную (четную) компоненту  $R_k(x) = \text{Re}(\Psi_k(x))$  (см. рис. 9.1) и мнимую (нечетную) компоненту  $S_k(x) = \text{Im}(\Psi_k(x))$  (см. рис. 9.2) вейвлета Габора [8].

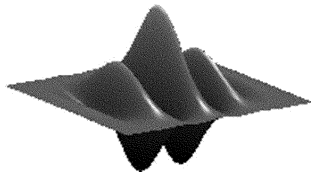


Рис. 9.1.

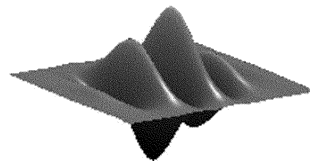


Рис. 9.2.

Вектор  $k$  определяет частоту и направление функций  $R_k(x)$  и  $S_k(x)$ .  $\omega$  — отвечает частоте, а  $\theta$  — повороту. На рисунках 10.1 и 10.2 изображены функции  $R_k(x)$  и  $S_k(x)$  соответственно, при разных значениях  $\omega$  и  $\theta$ .

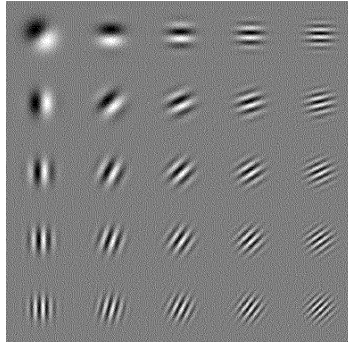


Рис. 10.1.

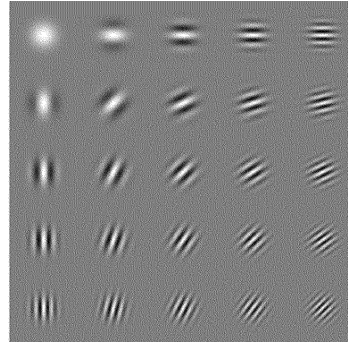


Рис. 10.2.

Вейвлет преобразование  $T_k(x, y)$  функции (изображения)  $I(x, y)$  представляется в виде:

$$T_k(x, y) = \iint I(u, v) \Psi_k(x - u, y - v) du dv. \quad (*)$$

В [8] предложен метод использования четного вейвлет преобразования Габора для распознавания лиц. Рассмотрим функцию  $R_k(x, y)$  при различных значениях  $k(\omega, \theta)$  (см. рис. 11) и изображение лица  $I(x, y)$  (см. рис. 12). Для каждого значения  $k(\omega, \theta)$  вычислим  $T_k(x, y)$  (см. рис. 13).

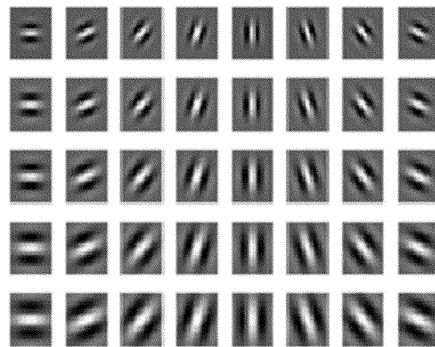


Рис. 11.



Рис. 12.

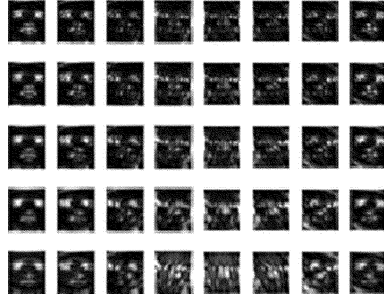


Рис. 13.



Рис. 14.

С помощью полученного семейства изображений  $T_k(x, y)$  можно найти на лице характерные точки. Точку  $(x_0, y_0)$  назовем характерной, если

$$T_k(x_0, y_0) = \max_{(x,y) \in P} (T_k(x, y)), \quad \text{и}$$

$$T_k(x_0, y_0) > \frac{1}{WH} \sum_{x=1}^W \sum_{y=1}^H T_k(x, y), \quad k = 1, \dots, 40,$$

где  $P$  — прямоугольное окно с центром в точке  $(x_0, y_0)$ ,  $W$  — ширина изображения,  $H$  — высота изображения. Первое уравнение говорит о том, что значение  $T_k(x_0, y_0)$  максимально в  $P$ -окрестности  $(x_0, y_0)$ . Второе — о том, что это максимум не локальный. Здесь важен размер окна  $P$ , в [8] размер предлагался  $9 \times 9$  пикселей. Если рассмотреть полученные характерные точки на исходном изображении  $I(x, y)$ , то они сконцентрируются около глаз, носа, рта и т. д. (см. рис. 14).

Для каждой характерной точки  $(x_k, y_k)$  каждого изображения  $i$  определим характерные вектора (вектора черт лица)  $v_{i,k} = (x_k, y_k, T_{i,j}(x_k, y_k), j = 1, \dots, 40)$ . Такой вектор состоит из координат особой точки и 40 вейвлет коэффициентов, отвечающих этой точке. С помощью характерных векторов  $v_{i,k}$  и  $v_{j,t}$  определим степень схожести изображений  $i$  и  $j$ .

$$S_{k,r}(i, j) = \frac{\sum_{m=3}^{42} |v_{i,k}(m)| \cdot |v_{j,r}(m)|}{\sqrt{\sum_{m=3}^{42} |v_{i,k}(m)|^2 \sum_{l=3}^{42} |v_{j,r}(l)|^2}},$$

где  $v_{i,k}(m)$  —  $m$ -я компонента вектора  $v_{i,k}$ .

Пусть даны база данных лиц и лицо  $X$ , которое требуется распознать. Алгоритм распознавания следующий: для каждого характерного вектора  $v_{x,k}$  и лица  $i$  из базы данных составим множество  $N_i^k$  ( $k = 1, \dots, 40$ ), состоящее из векторов  $v_{i,r}$  со следующими свойствами:

1.  $\sqrt{(x_k - x_r)^2 + (y_k - y_r)^2} < C_1$ ,

где  $C_1$  — радиус окружности примерно содержащий область глаз, носа и т. д.

2.  $S_{k,r}(i, X) > C_2$ ,

где  $C_2$  — стандартное отклонение функции  $S_{k,r}(i, X)$  по всем  $i$  из базы данных.

Во всех непустых множествах

$$S_i^k = \max_{r \in N_i^k} (S_{k,r}(i, X)).$$

Тогда степень схожести изображений  $i$  и  $X$  можно характеризовать величиной  $S_i = E(S_i^k)$  — средним значением  $S_i^k$  по всем  $k$ .

В этом алгоритме следует отметить следующие моменты:

1. Выбор константы  $C_1$  следует из условий съемки изображений, размера изображения в пикселях и т. д.  $C_1$  подбирается эмпирически.
2. Для двух разных изображений, даже одного человека, количества характерных векторов могут быть различными.

### 3. Тестирование результатов

Эффективность описанных методов существенно зависит от свойств изображений. Необходимо, чтобы все фотографии снимались

с одинаковым ракурсом и освещением. Для всех методов требуется, чтобы размеры изображения и лица были одинаковы. Вследствие чего, для сравнения эффективности различных методов, требуется их тестирование на одинаковых базах данных. Существует несколько общепринятых распространенных баз данных:

*Olivetty Research Ltd.* ORL (AT&T). Содержит 40 черно-белых изображений. 40 человек по 10 фотографий. Фотографии в фас с небольшими поворотами головы. Рис. 15. Адрес: <http://www.uk.research.att.com>.



Рис. 15.

*Yale Face Database*. Содержит 165 черно-белых изображений. 15 человек с различными выражениями лица (радость, грусть, удивление, и т. д.) Фотографии в фас. Различная освещенность. Рис. 16. Адрес: <http://cvc.yale.edu>.



Рис. 16.

*MIT (Massachusetts Institute of Technology) Database.* Содержит 432 изображения. Различная освещенность, поворот головы и дальность до объектива. Адрес: <ftp://whitechapel.media.mit.edu/pub/images>.

*FERET.* Адрес: <http://www.nist.gov/humanid/feret>.

База ORL является наиболее распространенной, и описанные алгоритмы дали на ней следующие результаты.

Алгоритм	Процент распознавания
Метод главных компонент	80%
Линейный дискриминант Фишера	91%
Одномерная Марковская Модель	84%
Двумерная Марковская Модель	99.5%
Вейвлеты Габора	95.5%

Комбинирование этих методов дает высокий процент распознавания, однако, на практике получившиеся изображения требуют предварительной обработки. Основная проблема предварительной обработки состоит в определении местонахождения лица на изображении. Далее требуется изменить размер изображения лица до эталонного, после чего провести выравнивание гистограммы освещенности. При этом требуется не потерять качество изображения. Например, если требуется получить качество распознавания сходное с качеством на базе ORL, то разрешающая способность фотографирующего устройства должна в итоге давать изображение лица примерно  $100 \times 100$  пикселей.

#### 4. Методы слежения за динамическими объектами

Пусть дано изображение в виде матрицы  $A$  размера  $m \times n$ . В окне  $B$  размером  $k \times k$ ,  $k \leq \min(n, m)$  приблизим его полиномом от двух переменных. Элемент  $B(i, j)$  заменяется на  $\tilde{B}(i, j) = ai^2 + bj^2 + cij + di + ej + f$ , и необходимо минимизировать функционал

$$F(a, b, c, d, e, f) = \sum_{i,j=1}^k (B(i, j) - (ai^2 + bj^2 + cij + di + ej + f))^2 \rightarrow \min,$$



что сводится к решению системы линейных уравнений

$$\frac{dF}{da} = 0, \quad \frac{dF}{db} = 0, \quad \dots, \quad \frac{dF}{df} = 0.$$

Представляет интерес квадратичная часть такого приближения — матрица

$$C = \begin{pmatrix} a & \frac{c}{2} \\ \frac{c}{2} & b \end{pmatrix}.$$

Пусть  $\lambda_1^t, \lambda_2^t > 0$  — собственные значения матрицы  $C^t$  в  $t$ -ом случае,  $t = 1, \dots, M$ , где  $M$  — число случаев, когда оба собственных значения положительны. Пусть

$$\alpha = \frac{1}{M} \sum_{t=1}^M \frac{\lambda_1^t + \lambda_2^t}{2}, \quad \beta = q\alpha, \quad \text{где } 0 < q \leq 1,$$

Сопоставим матрице  $B^t$  точку  $r(t)$  которую назовем *предварительной характерной*, если  $\min(\lambda_1^t, \lambda_2^t) > \beta$ , если при этом  $\lambda_1^t \lambda_2^t > \sigma$ , то назовем ее *характерной*. Здесь  $\sigma$  — некоторый порог. Методом статистических испытаний было получено, что для матриц, элементы которых отличаются не больше чем на  $\varepsilon$ , вероятность отличия собственных чисел более чем на  $5\varepsilon$ , не превосходит 0.02.

Далее можно перейти от матрицы изображения  $A$  к списку координат ее характерных точек. На рис. 17 представлен описанный выше метод, выделяющий на изображении предварительные характерные точки (серого цвета) и характерные точки (белого цвета).

Устойчивость описанного метода определения характерных точек позволяет работать с последовательностью кадров видеопотока. Небольшие изменения изображения успешно учитываются рассмотренным алгоритмом, что иллюстрируется на рис. 18 и 19.

На рис. 18 совмещены характерные точки с соседних кадров (точки первого кадра — серого цвета, второго — белого).

Изменения структуры характерных точек от кадра к кадру несущественны, что позволяет решать задачу слежения за характерными точками по форме точек и их взаимному расположению с течением времени. На рисунке 19 показано применение алгоритма выделения характерных точек для случая последовательных снимков облаков.

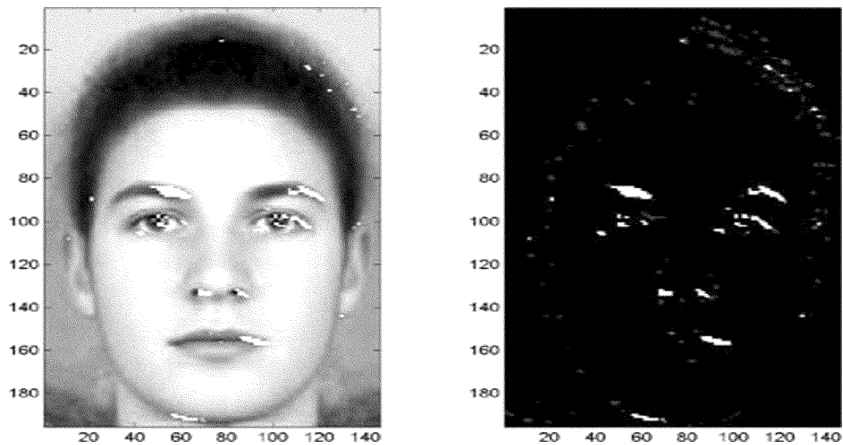


Рис. 17.

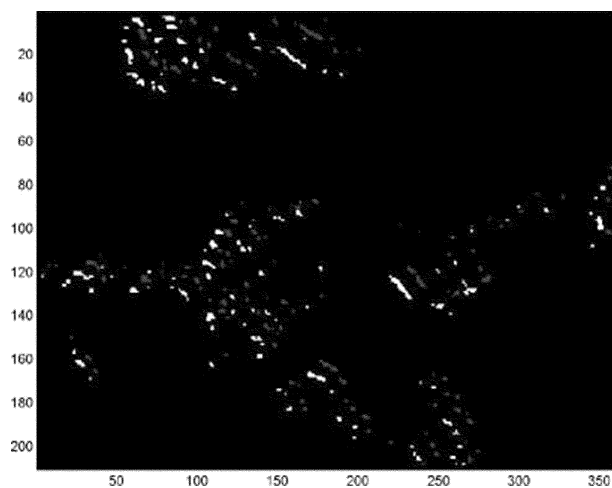


Рис. 18.

Для нахождения соответствия характерных точек последовательных кадров необходимо решить обобщенную задачу интерполяции: по множеству точек  $(x_i, y_i)$ ,  $i = 1, 2, \dots, nN$  найти семейство функций  $f_1, \dots, f_n$  таких, что для каждого  $i$  найдется  $l$  такое, что  $f_l(x_i) = y_i$ . Уточним постановку задачи.

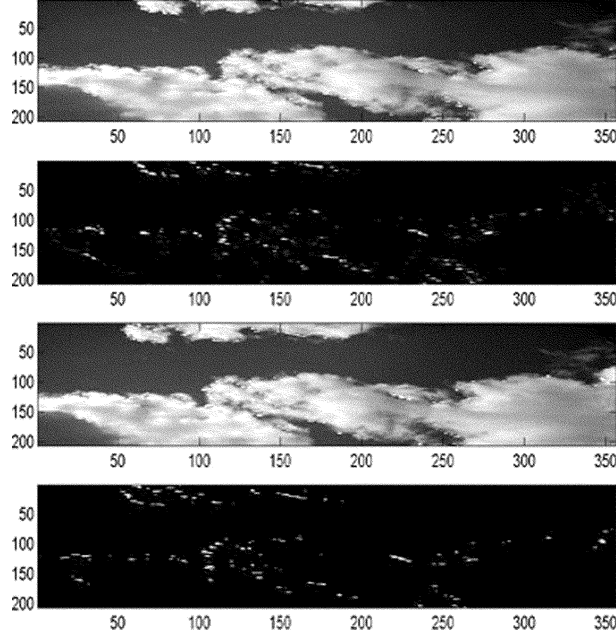


Рис. 19.

**Задача 1.** Пусть дано множество точек  $Z_i = \{(x_i^{(l)}, y_i^{(l)}), l = \overline{1, n}, i = \overline{1, N}; x_i^{(l)}, y_i^{(l)} \in \mathbb{R}\}$ , и известно, что существует семейство функций  $f_l, l = \overline{1, n}$ , таких, что при всех  $i, l$   $|y_i^{(l)} - f_l(x_i^{(l)})| < \delta$ . Требуется эти функции найти.

Для решения этой задачи для случая семейства линейных и квадратичных функций был разработан рекурсивный алгоритм  $\mathcal{A}$  накопления коэффициентов при линейном и квадратичном членах. В линейном случае при аппроксимации зависимости прямыми вида  $y = kx + b$  он состоит в следующем. Если обозначить

$$S_N = \sum_{i=1}^N (x_i^{(l(i))})^2, \quad U_N = \sum_{i=1}^N x_i^{(l(i))}, \quad V_N = \sum_{i=1}^N 1 = N,$$

$$R_N = \sum_{i=1}^N y_i^{(l(i))} \cdot x_i^{(l(i))}, \quad T_N = \sum_{i=1}^N y_i^{(l(i))},$$

то коэффициенты семейства прямых для  $N$  последовательных кадров будут определяться следующими соотношениями

$$k^N = \frac{T_N U_N - R_N V_N}{U_N^2 - S_N V_N}, \quad b^N = \frac{R_N U_N - T_N S_N}{U_N^2 - S_N V_N}.$$

При динамическом появлении новых изображений (то есть при увеличении  $N$ ,  $\tilde{N} = N + 1$ ) пересчет коэффициентов будет происходить лишь путем добавления одного нового члена в  $S_{\tilde{N}}$ ,  $U_{\tilde{N}}$ ,  $V_{\tilde{N}}$ ,  $R_{\tilde{N}}$  и  $T_{\tilde{N}}$ . Имеет место

**Теорема 1.** *Если  $N \geq n$ , то алгоритм  $\mathcal{A}$  решает задачу 1 (о восстановлении траекторий) со сложностью  $n^2 N$ .*

Алгоритм допускает модификацию, когда учитывается не только расположение характерных точек, но и их интенсивность.

Из теоремы следует, что точную картину движения можно отследить с определенной точностью при рассмотрении числа кадров, сопоставимого с количеством точек. Поэтому, для того, чтобы не рассматривать слишком большого количества кадров, целесообразно разбивать каждый снимок на области с небольшим количеством характерных точек на них.

Еще один рассмотренный в работе метод представляет собой некоторое изменение задачи о назначениях. Требуется решить задачу о назначении характерных точек из последовательных кадров, где точек на разных кадрах, вообще говоря, различное количество. Математически ее можно сформулировать следующим образом:

$$\begin{aligned} \sum_{i=1}^m \sum_{j=1}^n x_{ij} d_{ij} &\rightarrow \min, \\ \sum_{i=1}^m x_{ij} &\geq 1 \quad \forall j \in \overline{1, n}, \\ \sum_{j=1}^n x_{ij} &\geq 1 \quad \forall i \in \overline{1, m}, \\ x_{ij} &\in \{0, 1\}. \end{aligned}$$

Здесь  $d_{ij}$  — расстояние между  $i$ -ой точкой одного кадра и  $j$ -ой из второго кадра,  $x_{ij} = 1$  — назначать,  $x_{ij} = 0$  — не назначать  $i$ -ую точку первого кадра  $j$ -ой точке второго кадра (матрица назначений). Под *минимальным паросочетанием* понимается случай, когда достигается минимальная сумма расстояний между выбираемыми парами точек. Анализируя задачу, получаем утверждение:

**Утверждение 1.** *Если паросочетание  $\Sigma$  минимально, то отрезки, соединяющие соответствующие точки разных кадров, не пересекаются.*

Можно использовать метод нахождения минимального паросочетания для решения задачи соответствия характерных точек последовательных кадров.

**Задача 2.** Даны точки  $(x_1^l, y_1^l)$ ,  $(x_2^l, y_2^l)$ ,  $l = 1, \dots, n$  первого и второго кадров соответственно, известно, что существует подстановка  $T : \{1, \dots, l\} \rightarrow \{1, \dots, l\}$ , такая что

$$K_1 < (y_2^{T(l)} - y_1^{(l)}) / (x_2^{T(l)} - x_1^{(l)}) < K_2,$$

$$c_1 < \sqrt{(x_1^{(l)} - x_2^{T(l)})^2 + (y_1^{(l)} - y_2^{T(l)})^2} < c_2.$$

Восстановить подстановку  $T$ .

**Теорема 2.** *Решение задачи минимального паросочетания дает решение задачи 2. При этом*

$$nc_1 \sqrt{1 + K_1^2} \leq \Sigma \leq nc_2 \sqrt{1 + K_2^2}.$$

В результате на основе предложенных алгоритмов была написана программа, которая находит минимальные паросочетания и проверяет выполнение теоремы 2. Компьютерные эксперименты показывают, что существенные всплески величины  $\Sigma$  возникают при появлении точек, движение которых не описывается ранее найденным семейством параллельных прямых.

Работа выполнена на кафедре математической теории интеллектуальных систем механико-математического факультета МГУ им. М. В. Ломоносова под руководством проф. Бабина Д. Н.

## Список литературы

- [1] Turk M., Pentland A. Eigenfaces for recognition // Journal of Cognitive Science. 1991. P. 71–86.
- [2] Belhumer Peter N., Hespanha Joao P., Kriegman David J. Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection. 1996 European Conference on Computer Vision.
- [3] Duda R., Hart P. Patten Clasification and Scene Analysis. New York: Wiley, 1973.
- [4] Samaria F. Face recognition using Hidden Markov Models. PhD thesis. Engineering department, Cambrige University. Oct. 1994.
- [5] Nefian Ara V., Hayes III Monson H. Hidden Markov Models For Face Recognition. Center for Signal and Image Processing School of Electrical and Computer Engineering Georgia Institute of Technology, Atlanta.
- [6] Eickeler Stefan, Müller Stefan, Rigoll Gerhard. High Performance Face Recognition Using Pseudo 2-D Hidden. Gerhard-Mercator-University Duisburg, Department of Computer Science, Faculty of Electrical Engineering.
- [7] Müller Stefan, Wallhoff Frank, Hülsken Frank. Facial Expression Recognition Using Pseudo 3-D Hidden Markov Models. Dep. of Computer Science, Faculty of Electrical Engineering.
- [8] Kepenekci Burcu. Face Recognition Using Gabor Wavelet Transform. A thesis submitted to the graduate school of natural sciences of the Middle East technical university.
- [9] Yang G., Huang T. S. Human Face Detection in Complex Background // Pattern Recognition. Vol. 27. No. 1. 1994. P. 53–63.
- [10] Добеши И. Десять лекций по вейвлетам. РХД Москва–Ижевск, 2001.
- [11] Столниц Э., ДеРоуз Т., Салезин Д. Вейвлеты в компьютерной графике. РХД Москва–Ижевск, 2002.