ИЗВЕСТИЯ РАН. ТЕОРИЯ И СИСТЕМЫ УПРАВЛЕНИЯ, 2022, № 3, с. 122–129

ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ

УДК 004.93'11, 004.852

ОБНОВЛЕНИЕ БИОМЕТРИЧЕСКОГО ШАБЛОНА ПРИ ПОМОЩИ ОЦЕНКИ КАЧЕСТВА ИСХОДНЫХ ДАННЫХ[1]

© 2022 г. С. Б. Кубентаева[a,b], И. А. Матвеев[c,*], И. А. Соломатин[a,b,]**

[a] *МФТИ, Долгопрудный, МО, Россия*

[b] *Samsung R&D Institute Russia, Москва, Россия*

[c] *ФИЦ ИУ РАН, Москва, Россия*

**e-mail: matveev@ccas.ru*

***e-mail: ivan.solomatin@phystech.edu*

Поступила в редакцию 10.12.2021 г.
После доработки 18.12.2021 г.
Принята к публикации 31.01.2022 г.

Биометрические признаки, используемые в системах распознавания, подвержены старению. Кроме того, возможны вариации условий работы, не учтенные при регистрации пользователя. Поэтому одна из задач, которые необходимо решать при построении универсальной и длительно функционирующей биометрической системы, – обновление биометрического шаблона. Предлагается алгоритм обновления биометрического шаблона, использующий оценку расстояния от предъявленных биометрических признаков до признаков пользователя, зарегистрированных в идеальных условиях. Такая оценка рассчитывается нейросетью, обученной на базе данных с большой вариабельностью условий регистрации. Алгоритм протестирован на системе распознавания по лицу.

DOI: 10.31857/S0002338822030088

Введение. Биометрическое распознавание человека все шире используется в современном мире. Существует множество *биометрических модальностей* – характерных особенностей человека, по которым можно проводить распознавание, например, форма лица, рисунок радужной оболочки глаза, отпечатки пальцев, динамика походки, тембр голоса и т.д. [1]. *Биометрический шаблон* – совокупность числовых признаков, определяемых биометрической системой по данным в одной или нескольких модальностях. Получение данных биометрической системой называется *регистрацией биометрии*. Внесение шаблона и идентификатора пользователя в базу данных называется *регистрацией пользователя*. В дальнейшем при идентификации человека производится регистрация биометрии, создается шаблон и сравнивается с записанным в базе данных. Однако из-за различия условий регистрации, а также, возможно, изменений биометрических характеристик человека распознавание может быть нестабильным [2]. Это в той или иной степени присуще любой биометрической модальности: лицу [3], радужке [4], отпечатку пальца [5] и т.д. Таким образом, одна из задач, которые необходимо решать при построении биометрической системы, – обновление шаблона [6]. Общая схема действий, производимых биометрической системой распознавания, показана на рис. 1. Это позволяет системе подстраиваться под особенности регистрации в различных условиях, а также отслеживать возможные изменения в характеристиках пользователя, что в итоге улучшает точность распознавания [7].

1. Алгоритмы обновления шаблонов. Существует множество различных алгоритмов обновления биометрического шаблона [8]. Простейший вариант – добавлять признаки всех новых регистраций, принятых системой. Однако размер шаблона ограничен объемом используемой памяти и временем, отводимым для сравнения. Предпринимались попытки создавать комбинированные шаблоны путем усреднения [9]. Однако они не получили большого развития, поскольку пространство биометрических признаков не является линейным и даже может не быть замкнутым относительно операции сложения или усреднения. В общем случае в пространстве биометрических признаков доступна только операция определения расстояния $\rho(\vec{a},\vec{b})$ между его

[1] Работа выполнена при частичной финансовой поддержке РФФИ (гранты № 19-31-90171, 20-01-00609).

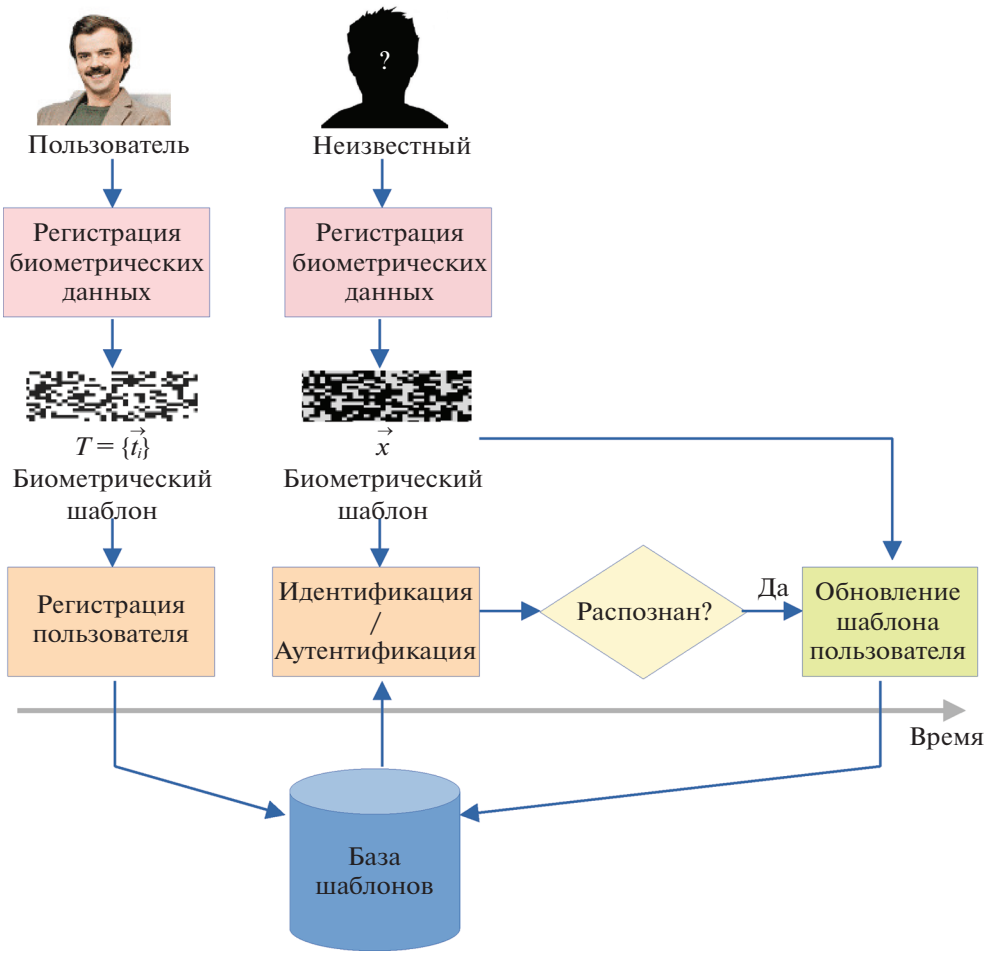

Рис. 1. Схема действий биометрической системы

элементами $\vec{a}$ и $\vec{b}$. Далее, в соответствии с установившейся терминологией, будем называть наборы биометрических признаков (элементы пространства биометрических признаков) *векторами*, хотя они не образуют векторного пространства.

Обновление шаблонов обычно происходит с помощью замены старых векторов на полученные в последующих регистрациях. Выбор заменяемого вектора может осуществляться различными способами [10]: Random, Naive, FIFO, LFU. Эти алгоритмы обладают значимым недостатком: они неустойчивы к так называемому “отравлению шаблона” [11] – процедуре, с помощью которой злоумышленник может испортить шаблон таким образом, чтобы система принимала его вместо настоящего пользователя. Данная процедура воспроизводима только в случае, если у злоумышленника есть большое число попыток доступа и информация о внутреннем устройстве системы. Однако даже такая угроза компрометации считается недопустимой.

Существуют методы, которые выбирают для добавления наиболее подходящие векторы по некоторому критерию, и по этому же критерию выбирают удаляемые [12]. Примером такого алгоритма является MDIST. Изначально MDIST был описан как алгоритм генерации шаблона, т.е. выбора подмножества векторов, полученных при регистрации, которые включаются в шаблон [13]. В пространстве биометрических признаков векторы, соответствующие одному человеку, лежат в области относительно небольшого диаметра. Векторы, полученные системой при регистрации, составляют кластер внутри этой области. Предполагается, что центр области образуют признаки, полученные при хороших условиях регистрации, а центроид кластера при достаточно большом числе элементов близок к центру области. Метод MDIST выбирает N век-

торов из общего числа M таким образом, чтобы минимизировать среднее взаимное расстояние между ними:

$$(n_1,\ldots,n_N) = \arg \min_{i_1,\ldots,i_N \in 1,M} \sum_{i,j\in(i_1,\ldots,i_N)} \rho(\vec{t}_i,\vec{t}_j), \tag{1.1}$$

где $\vec{t}_i$ – вектор признаков, входящий в шаблон. Если изначальный набор обладает большой вариабельностью, выбранное подмножество будет находиться в окрестности центроида кластера, соответствующего пользователю.

В [10] предложено использовать MDIST как алгоритм обновления шаблона. Высказано предположение о том, что в длительной перспективе (при большом количестве полученных системой признаков) шаблон сведется к множеству векторов, которое имеет малый диаметр и расположено в окрестности центра кластера, соответствующего пользователю. Однако если признаки, полученные при начальной регистрации, недостаточно вариабельны и находятся в удалении от центра области, такой процесс займет длительное время. Возможно даже, что шаблон сойдется в удалении от истинного центра кластера, что приведет к снижению точности распознавания.

Другим алгоритмом, выбирающим элементы шаблона, является DEND [13], который проводит кластеризацию всех имеющихся векторов признаков и внутри каждого кластера выбирает центроид с помощью алгоритма MDIST. Данный метод также подвержен опасности “отравления шаблона”.

Существуют графовые методы отбора признаков. Например, в [14] предлагается строить по шаблону взвешенный граф, в котором вес ребра равен расстоянию между векторами $\vec{t}_i$. Для выбора элементов, которые будут добавлены в шаблон, производится разделение графа на две части минимальным разрезом.

В данной работе предлагается алгоритм, который аналогично MDIST [10] при регистрации нового вектора и выполнении некоторых условий заменяет им один из старых векторов шаблона. Однако решение о замене и выбор заменяемого вектора осуществляется не процедурой (1.1), а при помощи оценки качества исходных данных. При сделанных предположениях качество исходных данных может служить оценкой расстояния до центра области пространства признаков, принадлежащей пользователю. Алгоритм протестирован в применении к распознаванию по лицу.

2. Постановка задачи. Будем описывать алгоритм применительно к аутентификации по лицу. Допустим, что у системы имеется один *Пользователь*. Задача системы – на каждом предъявленном изображении проверять, представлен на нем Пользователь или другой человек. Имеется фиксированый алгоритм извлечения признаков (FE – feature extraction), который переводит изображение в вектор признаков таким образом, что расстояние ρ между векторами, соответствующими одному человеку, мало, а разным людям – велико. При регистрации Пользователя создается его шаблон T – набор векторов признаков, полученных при регистрации биометрии:

$$T = \{\vec{t}_i\}_{i=1}^{N}. \tag{2.1}$$

Расстоянием от вектора признаков $\vec{x}$ до шаблона считается расстояние от вектора до ближайшего элемента шаблона:

$$\rho(\vec{x},T) = \min_i \rho(\vec{x},\vec{t}_i). \tag{2.2}$$

При верификации входное изображение I сначала переводится в вектор признаков: $\vec{x} = FE(I)$. Решение о том, Пользователь на изображении или другой человек, выносится на основании расстояния от $\vec{x}$ до шаблона в сравнении с некоторым порогом θ:

$$\begin{aligned} &\rho(\vec{x},T) \le \theta \rightarrow \text{ на изображении Пользователь,} \\ &\rho(\vec{x},T) > \theta \rightarrow \text{ на изображении другой человек.} \end{aligned} \tag{2.3}$$

Точность распознавания системы обычно оценивается варьированием порога θ и оценкой долей ложноположительных (FPR) и ложноотрицательных (FNR) результатов на наборе данных при различных значениях порога. В работе используются следующие меры качества алгоритма:

1) E_{-4} – величина FNR при $FPR = 10^{-4}$, т.е. доля ложноотрицательных результатов при фиксированном пороге, обеспечивающем долю ложноположительных результатов, равную 10^{-4}. В литературе распространено обозначение $FNR@FPR = 10^{-4}$;

2) $E_{-5} = FNR@FPR = 10^{-5}$;

3) $E_{-6} = FNR@FPR = 10^{-6}$;

4) $E_{-7} = FNR@FPR = 10^{-7}$.

Обновление шаблона методом типа MDIST заключается в добавлении нового вектора признаков $\vec{x}$ вместо одного из старых ($\vec{t}_j$), если выполняется некоторое условие U:

$$\begin{gathered} P_0 = \sum_{i,j=1}^{N} \rho(\vec{t}_i, \vec{t}_j), \\ P_k = \sum_{i,j=1}^{N} \rho(\vec{t}_i^{(k)}, \vec{t}_j^{(k)}), \\ \vec{t}_i^{(k)} = \begin{cases} \vec{x}, & \text{если } k = i, \\ \vec{t}_i & \text{иначе}, \end{cases} \\ q = \arg\max_k P_k, \\ T' = \begin{cases} \{\vec{x}\} \cup T \backslash \{\vec{t}_q\}, & U - \text{истинно}, \\ T & \text{иначе}. \end{cases} \end{gathered} \tag{2.4}$$

В качестве U в работе используется условие $\rho(\vec{x}, T) < \theta_{update}$, где порог соответствует распознаванию с ошибкой E_{-7}. Таким образом, один из старых векторов шаблона заменяется на новый, только если тот дает распознавание Пользователя с высокой уверенностью.

3. Предлагаемый метод. Расстояние $d = \rho(FE(I), \vec{c})$ от вектора признаков лица на изображении до центра кластера не может быть рассчитано непосредственно, но при этом является функцией изображения I. В данной работе предлагается обучить нейронную сеть, предсказывающую это расстояние по изображению. Используется предположение, что алгоритмы FE переводят в центроид кластера изображения высокого качества, которые лучше всего поддаются распознаванию. Для лиц – это центрированные изображения с хорошим освещением. По краям кластера располагаются низкокачественные изображения – в экстремальных позах, с плохим освещением, неточно сегментированные. Данное предположение подтверждается при анализе набора данных [15] (*Extended Yale Face Database B*). Таким образом можно воспринимать расстояние d как показатель качества исходных данных, значение ноль соответствует наивысшему качеству (некоторому идеальному изображению Пользователя). Примеры изображений из данного набора с различными d приведены на рис. 2.

3.1. Алгоритм обновления шаблона. Рассчитав для каждого полученного системой изображения расстояние d, его можно хранить вместе с соответствующим шаблоном, формально записав как $d(\vec{t}_i)$. Обновление шаблона предлагается проводить так, чтобы минимизировать суммарное d в шаблоне:

$$\sum_{\vec{t}_i \in T} d(\vec{t}_i) \to \min. \tag{3.1}$$

Процедура (2.4) становится формально более простой:

$$\begin{gathered} q = \arg\max_k d_k, \\ T' = \begin{cases} \{\vec{x}\} \cup T \backslash \{\vec{t}_q\}, & U - \text{истинно}, \\ T & \text{иначе}. \end{cases} \end{gathered} \tag{3.2}$$

Таким образом, проблема сводится к построению наилучшего метода оценки d по изображению. Предлагается использовать для этого нейросеть, обученную на корпусе данных.

3.2. Подготовка данных для обучения нейронной сети. В качестве архитектуры применяется модификация популярной сети LeNet [16]. Архитектура приведена в табл. 1. На вход сети подаются те же изображения, что подаются на вход алгоритму FE. Результат последнего полносвязного слоя после функции активации – предсказание искомого параметра d.

Для обучения и тестирования взята база изображений [15]. Эта база выбрана, поскольку она предоставляет большое количество изображений (до 585) на каждого пользователя, а изображения имеют высокую вариабельность условий освещения и углов поворота лиц.

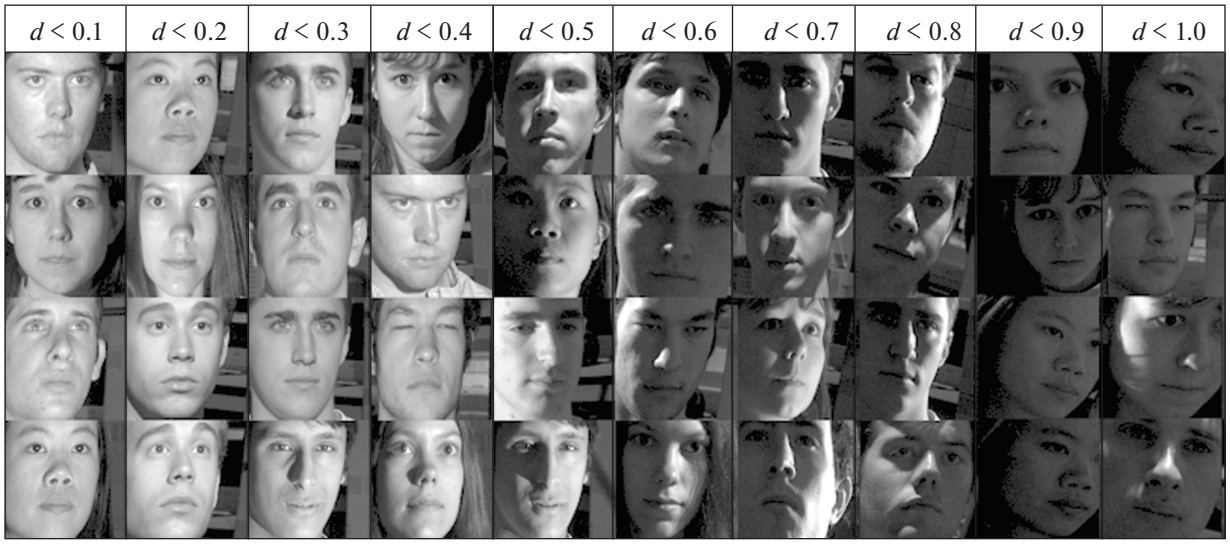

Рис. 2. Примеры изображений с различными d из базы данных

Для подготовки данных необходим набор изображений $J = \{I_i\}$, подающихся на вход алгоритму FE, а также идентификатор пользователя для каждого изображения $\{\nu_i\}_{i=1}^{|J|}$. По всему набору изображений J рассчитываются векторы признаков:

$$X = \{\vec{x}_i\}_{i=1}^{|J|}, \quad \vec{x}_i = \mathrm{FE}(I_i). \tag{3.3}$$

Далее признаки группируются по идентификаторам пользователей на соответствующих изображениях; для пользователя с номером k

$$X_k = \{\vec{x}_i \mid \nu_i = k\}. \tag{3.4}$$

Таблица 1. Архитектура нейронной сети для оценки расстояния d

Слой	Параметр	Размер выхода
Input		$1 \times 142 \times 142$
Conv 3 × 3	Stride = 2	$4 \times 70 \times 70$
ReLu		$4 \times 70 \times 70$
Conv 3 × 3	Stride = 2	$8 \times 34 \times 34$
ReLu		$8 \times 34 \times 34$
Conv 3 × 3	Stride = 2	$16 \times 7 \times 7$
ReLu		$16 \times 7 \times 7$
Conv 3 × 3	Stride = 2	$16 \times 3 \times 3$
ReLu		$16 \times 3 \times 3$
Conv 3 × 3	Stride = 2	$16 \times 1 \times 1$
ReLu		$16 \times 1 \times 1$
DropOut	Rate = 0.7	$16 \times 1 \times 1$
FullyConnected		$32 \times 1 \times 1$
ReLu		$32 \times 1 \times 1$
DropOut	Rate = 0.7	$32 \times 1 \times 1$
FullyConnected		$1 \times 1 \times 1$
Sigmoid		$1 \times 1 \times 1$

Таблица 2. Результаты тестирования сети на тестовом и валидационном наборе данных

	Выборка	
Мера	валидационная	тестовая
MAE (mean average error)	0.09	0.08
Доля изображений, на которых d предсказано с ошибкой ≤0.1	0.68	0.70
Доля изображений, на которых d предсказано с ошибкой ≤0.2	0.98	0.96
Доля изображений, на которых d предсказано с ошибкой ≤0.4	0.999	0.998

Таблица 3. Ошибка алгоритма распознавания по лицу с использованием различных алгоритмов обновления шаблона

Мера	Без обновления	MDIST [13]	**Предлагаемый**
E_{-4}	0.017	0.020	0.015
E_{-5}	0.076	0.025	0.018
E_{-6}	0.260	0.029	0.019
E_{-7}	0.300	0.033	0.019

Затем для каждого пользователя определяется центроид кластера:

$$\vec{c}_k = \arg\min_{\vec{c}\in X_k} \sum_{\vec{x}\in X_k, \vec{x}\neq\vec{c}} \rho(\vec{c},\vec{x}). \tag{3.5}$$

С помощью центроида можно оценить целевое значение d для каждого изображения. Для того, чтобы исключить возможную разницу в размерах кластеров для разных пользователей, полученное d нормируется на максимальное значение для каждого кластера:

$$d_i = \frac{\rho(x_i, c_{v_i})}{\max\limits_{x\in X_{v_i}} \rho(x, c_{v_i})}. \tag{3.6}$$

4. Численные эксперименты. В качестве алгоритма FE для оценки предлагаемого алгоритма применена реализация [17], находящаяся в открытом доступе [18] (модель “buffalo_s”). Набор данных взят из открытой базы [15].

Обучающий, валидационный и тестовый наборы данных содержат соответственно 5017, 3281 и 8433 изображений. Изображения каждого пользователя входят лишь в один из наборов, чтобы исключить возможность переобучения на конкретные лица. Обучение проводилось с помощью алгоритма градиентного спуска Adam [19] в течение 20 эпох.

Меры качества на тестовом и валидационном наборах по результатам обучения приведены в табл. 2.

4.1. Алгоритм обновления шаблона. Для тестирования работы алгоритма обновления шаблона на тестовом наборе данных рассчитана точность работы системы распознавания при различных алгоритмах обновления шаблона. Так как используемый алгоритм FE обладает крайне высокой точностью на данных [15], для регистрации каждого пользователя не задействованы вектора, которые лежат в окрестности центра кластера:

$$d_i < 0.4. \tag{4.1}$$

Таким образом имитируется ситуация, в которой на этапе регистрации изображение лица может быть недостаточно качественным.

Из изображений, не удовлетворяющих условию (4.1), для каждого пользователя u случайным образом выбиралось 10 изображений, которые составляли изначальный шаблон для распознавания. Далее системе на вход подавались остальные изображения пользователя u, затем – все

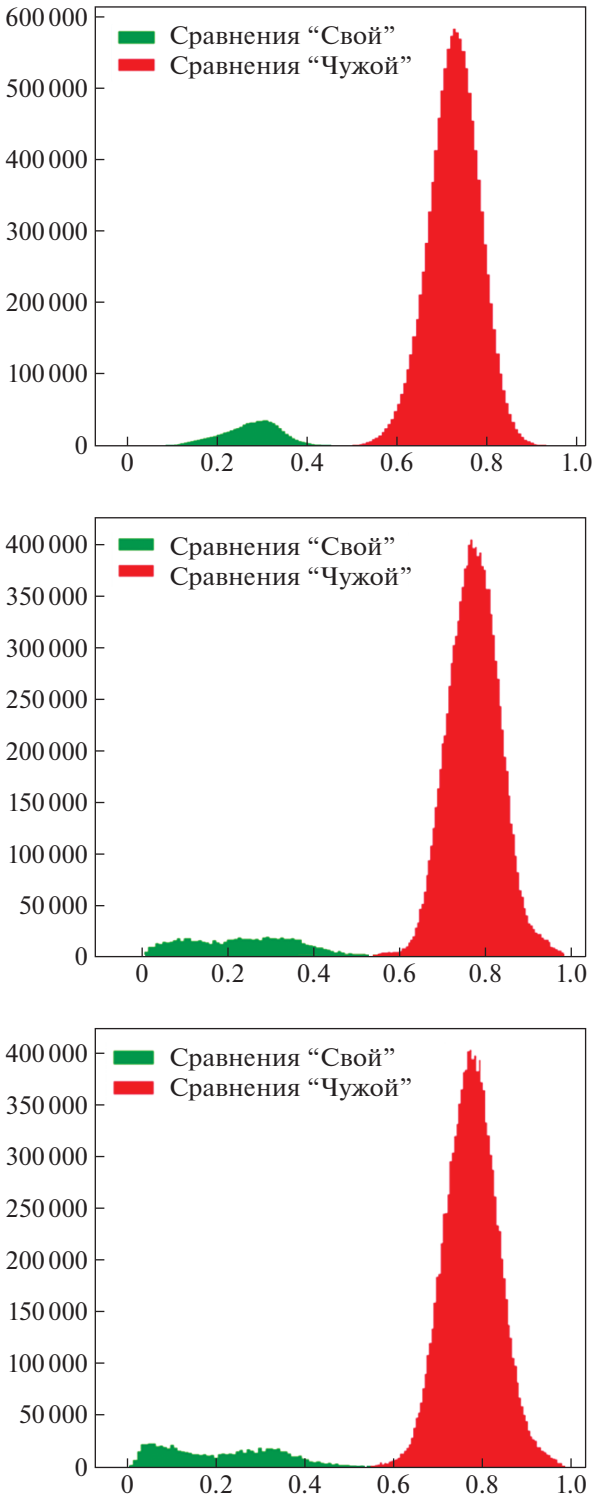

Рис. 3. Распределения попарных расстояний при распознавании шаблона, полученного: *а* – без обновления, *б* – алгоритмом MDIST [13], *в* – предлагаемым алгоритмом

изображения других пользователей. Для каждого изображения, поданного на вход, также выполнялась процедура (3.2). Эта процедура выполнялась 100 раз для каждого пользователя, чтобы оценить работу алгоритма при различных выборах начального шаблона. По распределению полученных в результате процедуры попарных расстояний оценивались *FNR* при различных *FPR* в соответствии с мерами 1–4.

В табл. 3 приведены значения мер качества, полученных с помощью вышеописанной процедуры на основе модели “buffalo_s” реализации [18] алгоритма [17] для трех случаев: без обновления шаблона, с обновлением алгоритмом MDIST и с обновлением предлагаемым методом. На рис. 3 приведены соответствующие гистограммы распределений попарных расстояний, полученных в ходе тестов. Расстояния, нормированные на диапазон [0; 1], задают ось абсцисс. По оси ординат отложены количества сравнений с таким расстоянием.

Заключение. Предложен алгоритм обновления биометрического шаблона, развивающий идею MDIST. Однако для определения заменяемого элемента шаблона используется метрика – расстояние до признаков, полученных в идеальных условиях, которая вычисляется с помощью нейронной сети. Этот подход показал высокое качество на валидационном множестве, применение данного алгоритма в системе распознавания по лицу дало снижение ошибки распознавания.

СПИСОК ЛИТЕРАТУРЫ

1. *Jain A.K., Ross A.* Introduction to Biometrics / Eds A.K. Jain, P. Flynn, A. Ross. Handbook of Biometrics. Springer, 2008. P. 1–22. ISBN 978-0-387-71040-2.
2. *Lantinis A.* A Survey of the Effects of Aging on Biometric Identity Verification // Intern. J. Biometrics. 2010. V. 2. № 1. P. 34–52.
3. *Ramanathan N., Chellappa R.* Face Verification Across Age Progression // IEEE Trans. Image Processing. 2006. V. 15. № 11. P. 3349–3361.
4. *Czajka A.* Template Ageing in Iris Recognition // Proc. Intern. Conf. Bio-inspired Systems and Signal Processing. Barcelona, Spain, 2013. P. 70–78.
5. *Kirchgasser S., Uhl A.* Template Ageing in Non-minutiae Fingerprint Recognition // Proc. 5th Intern. Workshop Biometrics and Forensics. Coventry, UK, 2017. P. 1–5.
6. *Hasse G., Wolf A.* Data Quality, Interoperability, Biometrics Fusion, and Template Ageing: Challenges for ePassports // Proc. Biometric Consortium Conf. Arlington, VA, USA, 2005.
7. *Carls J., Raines R., Grimaila M., Rogers S.* Biometric Security Enhancements Through Template Aging Matching Score Analysis // Proc. 3rd Intern. Conf. Information Warfare and Security. Omaha, NE, USA, 2008.
8. *Гнеушев А.Н., Ковков Д.В., Матвеев И.А., Новик В.П.* Оптимизация выбора биометрического эталона из последовательности // Изв. РАН. ТиСУ. 2015. № 3. С. 72–78.
9. *Hollingsworth K., Bowyer K., Flynn P.* Image Averaging for Improved Iris Recognition / Eds M. Tistarelli, M. Nixon. Lecture Notes in Computer Science. Advances in Biometrics. V. 5558. P. 1112–1121. Berlin, Heidelberg: Springer, 2009. P. 1112–121.
10. *Freni B., Marcialis G.L., Roli F.* Replacement Algorithms for Fingerprint Template Update // Intern. Conf. Image Analysis and Recognition. Berlin, Heidelberg: Springer, 2008. P. 884–893.
11. *Lovisotto G., Eberz S., Martinovic I.* Biometric Backdoors: A Poisoning Attack Against Unsupervised Template Updating // Proc. IEEE European Sympos. Security and Privacy. Genova, Italy, 2020. P. 184–197.
12. *Marcialis G.L., Rattani A., Roli F.* Biometric Template Update: An Experimental Investigation on the Relationship between Update Errors and Performance Degradation in Face Verification / Eds N. da Vitoria Lobo, T. Kasparis, F. Roli et al. Structural, Syntactic, and Statistical Pattern Recognition. Lecture Notes in Computer Science. V. 5342. Berlin, Heidelberg: Springer, 2008.
13. *Uludag U., Ross A., Jain A.* Biometric Template Selection and Update: a Case Study in Fingerprints // Pattern Recognition. 2004. V. 37. № 7. P. 1533–1542.
14. *Rattani A., Marcialis G.L., Roli F.* Biometric Template Update Using the Graph Mincut Algorithm: A Case Study in Face Verification // Proc. IEEE Biometrics Sympos. Tampa, FL, USA, 2008. P. 23–28.
15. *Georghiades A.S., Belhumeur P.N., Kriegman D.J.* From Few to Many: Illumination Cone Models for Face Recognition under Variable Lighting and Pose // IEEE TPAMI. 2001. V. 23. № 6. P. 643–660.
16. *LeCun Y., Boser B., Denker J.S. et al.* Handwritten Digit Recognition with a Back-propagation Network // Advances in Neural Information Processing Systems. 1989. V. 2.
17. *Chen S., Liu Y., Gao X., Han Zh.* Mobilefacenets: Efficient CNNs for Accurate Real-time Face Verification on Mobile Devices // Proc. Chinese Conf. Biometric Recognition. Cham: Springer, 2018. P. 428–438.
18. InsightFace: 2D and 3D Face Analysis Project. URL: https://github.com/deepinsight/insightface (дата обращения: 08.11.2021).
19. *Kingma D.P., Ba J. Adam* A Method for Stochastic Optimization // arXiv preprint arXiv:1412.6980. 2014.