КЫРГЫЗСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ИМЕНИ И.РАЗЗАКОВА

На правах рукописи

УДК007+681.322+681.5.015.42

АЛИМСЕИТОВА ЖУЛДЫЗ КЕНЕСХАНОВНА

РАЗРАБОТКА ИНТЕЛЛЕКТУАЛЬНОЙ АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ РАСПОЗНАВАНИЯ БИОМЕТРИЧЕСКИХ ОБРАЗОВ

05.13.16 – Применение вычислительной техники, математического моделирования и математических методов в научных исследованиях

Диссертация на соискание ученой степени
кандидата технических наук

Научный руководитель:
канд. техн. наук, доцент
Боскебеев К.Д.

Бишкек - 2019

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ

Актуальность темы диссертации. В настоящее время для усиления защиты информационных ресурсов разрабатываются технологии биометрической аутентификации личности, где биометрические данные человека преобразуются в криптографический ключ или пароль. Для их реализации применяются интеллектуальные методы и модели, базирующиеся на теории нейронных сетей (НС).

Вопросы обеспечения безопасности, в том числе аутентификации пользователей при обработке информации, были обозначены в Государственной программе «Цифровой Казахстан» и нашли отражение в Концепции кибербезопасности «Киберщит Казахстана». Разработка указанной концепции и анализ научно-практических исследований в данной области показывают, что мало исследованными остаются вопросы повышения эффективности распознавания биометрических образов в информационных системах, что и обуславливает актуальность настоящего исследования.

Связь темы диссертации с крупными научными программами, основными научно-исследовательскими работами, проводимыми научными учреждениями. Диссертационная работа проводилась в рамках научно-исследовательских проектов КазНИТУ имени К.И. Сатпаева №753МОН.ГФ.13.13 «Исследование вариантов реализации и разработка действующего лабораторного образца ON-LINE системы биометрического обезличивания электронных историй болезней для медицинского учреждения», № 757.МОН.ГФ.15.ИИТ.6 «Исследование, гармонизация, модификация и постановка на учет группы стандартов по биометрической поддержке информационной безопасности», в рамках научно-исследовательского проекта КГТУ им. И. Раззакова «Моделирование и анализ безопасности граждан по базе данных биометрики Кыргызской Республики».

Цель и задачи исследования. Целью работы является повышение эффективности распознавания биометрических образов за счет применения биометрико-нейросетевых методов.

Задачами исследования являются:

– разработка нейросетевой модели, позволяющей эффективно распознавать пользователей на основе анализа геометрических параметров фрагментов рукописного текста изменяемого размера;

– снижение размерности входной выборки за счет учета корреляционных связей между выходными сигналами НС;

– разработка метода синтеза критерия хи-квадрат распределений зависимых данных;

– разработка автоматизированной интеллектуальной системы распознавания биометрических образов в информационных системах;

– разработка методики формирования биометрических баз рукописных образов и отпечатков пальцев;

– экспериментальное исследование автоматизированной интеллектуальной системы распознавания биометрических образов в информационных системах.

Научная новизна полученных результатов заключается в следующем:

– разработана композитная нейросетевая модель, которая за счет использования в сверточной НС модулей долгой краткосрочной памяти, а также за счет адаптации параметров модели к условиям системы биометрической аутентификации, обеспечивает эффективное распознавание пользователей на основе анализа геометрических параметров фрагментов рукописного текста изменяемого размера;

– предложен метод снижения входной выборки НС за счет учета корреляционных связей между выходными сигналами НС;

– предложен метод синтеза критерия хи-квадрат распределений зависимых данных, позволяющий существенно увеличить достоверность оценок проверки статистических гипотез;

– разработана архитектура автоматизированной интеллектуальной системы распознавания биометрических образов с использованием нейросетевых технологий;

– разработана методика формирования баз рукописных образов и отпечатков пальцев.

Практическая значимость полученных результатов. Предложенные нейросетевые модели и методы позволили разработать интеллектуальную автоматизированную систему распознавания биометрических образов, которая позволяет с достаточной точностью распознавать отпечатки пальцев и рукописный почерк, а также может быть использована для создания инструментальных средств.

Экономическая значимость полученных результатов достигается внедрением разработанной в диссертационной работе системы, использованием представленных методов и технологий в процессе автоматизации управленческих и инженерных задач.

Основные положения диссертации, выносимые на защиту.

– композитная нейросетевая модель, которая обеспечивает эффективное распознавание пользователей на основе анализа геометрических параметров фрагментов рукописного текста изменяемого размера;

– влияние корреляционных связей между выходными сигналами НС при оценке энтропии преобразователей биометрия-код;

– метод снижения входной выборки НС за счет учета корреляционных связей между выходными сигналами НС;

– архитектура автоматизированной интеллектуальной системы распознавания биометрических образов с использованием нейросетевых технологий;

– методика формирования баз рукописных образов и отпечатков пальцев.

Личный вклад соискателя. Основные положения и результаты диссертационной работы, выносимые на защиту, получены автором самостоятельно. В работах, написанных в соавторстве, личный вклад

соискателя заключается в следующем: [13] – проведены исследования по выявлению аномального состояния СОВ, [2, 25, 26, 56, 57] – проведен анализ технологий распознавания биометрических образов; [36, 53] – проведен анализ НС и выбор алгоритмов обучения; [51, 58, 59] – предложен метод снижения входной выборки НС за счет учета корреляционных связей между выходными сигналами НС; [72, 73] – предложен синтез хи-квадрат критерия для зависимых данных; [32, 54, 79, 85, 86] – проведено формирование баз рукописных образов и отпечатков пальцев.

Апробации результатов диссертации. Результаты исследований докладывались и обсуждались на семинарах кафедры «Информационная безопасность» КазНИТУ имени К.И. Сатпаева, кафедры «Информационные системы и технологии» КГТУ имени И.Раззакова, а также международных конференциях «Innovation Challenges In Multidisciplinary Research&Practice» (Kuala Lumpur, 2014), «Информационные и телекоммуникационные технологии: образование, наука, практика» (Алматы, 2015); «Інформаціина безпека та компютерні технологи» (Кировоград, 2016); Интеллектуальные информационные и коммуникационные технологии – средство осуществления третьей индустриальной революции в свете стратегии «Казахстан-2050» (Астана, 2017); «Актуальні питання забезпечення кібербезпеки та захисту інформації» (Киев, 2018).

Полнота отражения результатов диссертации в публикациях в журналах: «Известия КГТУ» (4 статьи); «Вестник НАН РК» (2 статьи); «Доклады НАН РК» (1 статья); «Journal of Theoretical and Applied Information Technology» (1 статья); «Захист информації» (1 статья); «Известия НАН РК» (1 статья), «Intelligent Systems in Cybernetics and Automation Control Theory» (1 статья), а также подтверждены 2 авторскими свидетельствами на программы.

Структура и объем диссертации. Диссертация состоит из введения, трех глав и приложений. Полный объем диссертации составляет 164 страницы, в том числе основного текста 136 страниц, включая 56 рисунков и 12 таблиц. Список использованной литературы состоит из 87 наименований.

ГЛАВА 1. ИНТЕЛЛЕКТУАЛЬНЫЕ МЕТОДЫ РАСПОЗНАВАНИЯ ОБРАЗОВ В ИНФОРМАЦИОННЫХ СИСТЕМАХ

1.1. Структура и требования к разработке систем распознавания биометрических образов

В основе разработки биометрических распознающих систем и систем распознавания образов лежат одинаковые принципы [1]. Модульная архитектура распознавания образов представлена на (рис. 1.1).

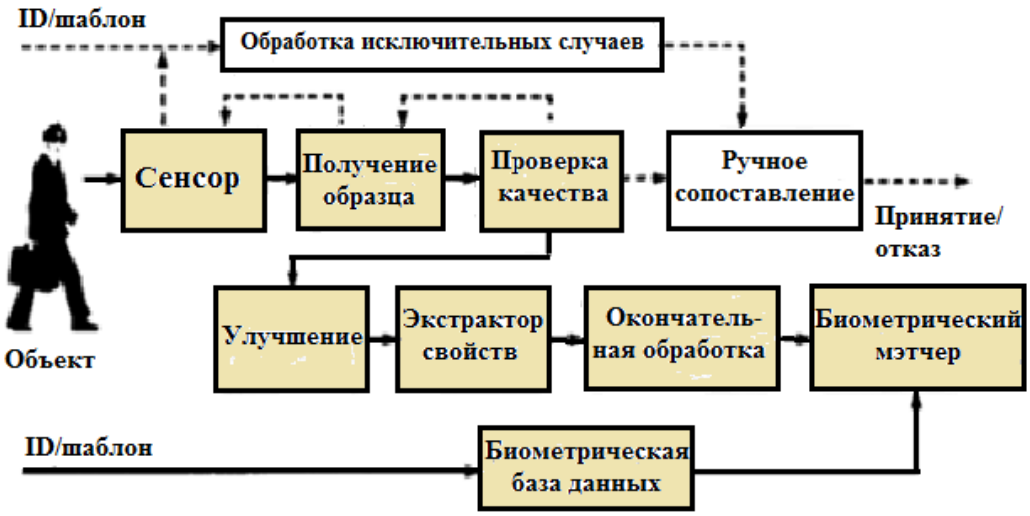


Рис. 1.1 - Модульная архитектура распознавания образов: закрашенные блоки - автоматизированные процессы, белые блоки - процессы, выполняемые вручную

К системам распознавания образов, в том числе к биометрическим предъявляют следующие требования:

- Точность. Оценивается двумя параметрами. Возможностью получения доступа злоумышленником (ошибка второго рода) и возможность отказа в доступе легитимному пользователю (ошибка первого рода). Эти вероятности подсчитываются для пользователей, не причастных к процессу обработки исключительных случаев [1].

- Скорость вычисления. В биометрической системе скорость вычисления

является очень значимым параметром. Кроме того необходимо знать масштабируемость системы (способность поддерживать различные по масштабу базы данных пользователей) [2].

- Обработка исключительных случаев. Это ручное сравнение образцов, изображенных на рис. 1.1 в белых блоках. Она является частью любой биометрической системы, так как у объекта могут быть случаи невозможности использования или/и регистрации или невозможности получения биометрических параметров. Важным показателем для биометрической аутентификации является количество исключительных случаев, приемлемое число которых, а также их обработка должны быть предусмотрены при разработке систем.

- Стоимость системы. В стоимость входит затраты на компоненты системы, расходы на техническое обслуживание, стоимость обучения персонала и пользователей, затраты на обработку одной операции. Кроме того, необходимо заранее рассчитать стоимость обработки исключительных случаев.

Учитывая все вышеперечисленные требования, было принято решение интеллектуализировать систему распознавания образов, чтобы исключить необходимость обработки исключительных случаев, поддержать высокую масштабируемость системы и снизить ее стоимость. Для этого в данной диссертации используются нейронные сети.

1.2. Нейронные сети: основные понятия и характеристики

Человеческий мозг обрабатывает информацию с помощью сети нейронов. К примеру, для распознавания лица компьютеру требуются минуты, а иногда часы, а человеческий мозг на это тратит 100-120 мс. Это говорит о том, что человеческий мозг распознает образы намного быстрее современных компьютеров. Из этого делаем вывод, что человеческий мозг это сеть нейронов, обрабатывающая информацию высокоэффективно и надежно. [3]. Этот факт

стимулирует ученых вести работы по воссозданию и исследованию нейронных сетей (НС).

Фактически «нейронная сеть» это биологический термин, и поэтому НС, построенные человеком должны называться искусственными нейронными сетями (ИНС).

Под термином нейронная сеть (neural network) понимают сеть элементов (искусственных нейронов), связанных синаптическими связями. НС похожа на человеческий мозг в двух аспектах:

– приобретение знаний происходит в процессе обучения;

– для сохранения знаний используются межнейронные соединения, называемые синаптическими весами.

В теории НС есть три основных понятия: нейрон, архитектура сети и обучение.

Модель нейрона представлена на рис. 1.2 [3].

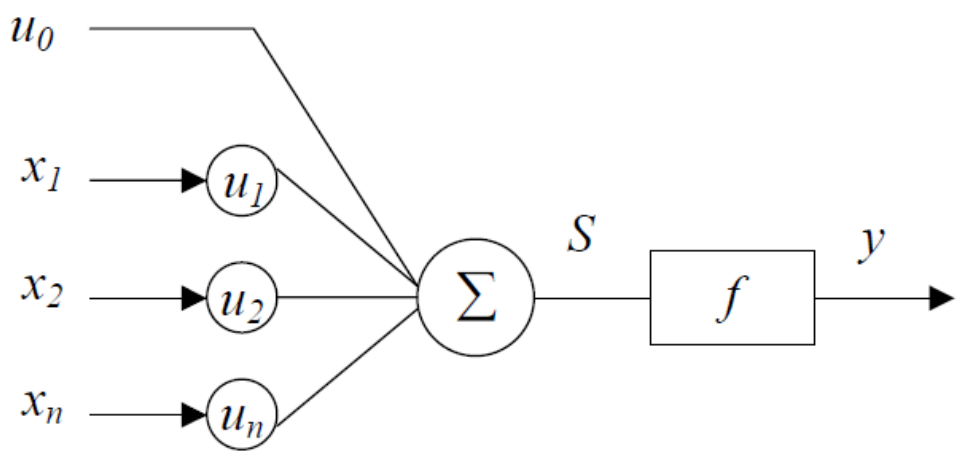


Рис. 1.2. Модель нейрона

Модель нейрона (neuron) первыми предложили МакКаллок и Питтс. Они предложили использовать его в качестве бинарного порогового элемента, который на выходе дает 1, если взвешенная сумма n входных сигналов xi, i = 1, 2...n больше определенного уровня u, и 0 – в остальных случаях. В большинстве случаях u связывают с входом x0=1 и принимают за весовой коэффициент, если

u>0, то связи возбуждающие, если u<0, то тормозные. Математическое выражение представлено формулой (1.1):

$$S = \sum_{i=1}^{n} u_i x_i + u_0, \tag{1.1}$$

где u_0 - величина смещения.

Функция активации нейрона имеет ступенчатый вид, то есть если суммарное входное воздействие превышает некоторое значение, то на выходе нейрона y появляется сигнал [3]:

$$f(s) = \begin{cases} a, \text{если } s > a \\ s, \text{если } -a \leq s \leq a \\ -a, \text{если } s < -a \end{cases} \text{– линейная.}$$

МакКаллок и Питтс доказали, что сеть нейронов такого вида, при правильно подобранных весах может выполнять универсальные вычисления.

Архитектуру НС можно рассматривать в виде графа с искусственными нейронами в узлах, соединенными взвешенными связями. Архитектуру НС определяют следующие параметры: количество входных, скрытых и выходных нейронов, структура связей (топология сети), правила распространения и комбинирования сигналов, правила исчисления выходного сигнала нейрона и правила обучения, корректирующие связи в сети. Возможности сети в значительной степени зависят от вида НС. Количество входных, скрытых и выходных нейронов зависит от вида НС и от решаемой задачи. НС по архитектуре связей можно разделить на два вида: сети с прямым распространением сигнала, где имеются только последовательные связи, и рекуррентные сети, где имеются обратные связи [3].

На рис. 1.3 показаны сети каждого вида. Сети с прямым распространением сигнала являются статическими, а рекуррентные сети динамическими. В сетях прямого распространения сигнала для входов нейронов вырабатываются

выходные значения, которые не зависят от предыдущего состояния сети. В рекуррентных сетях входы нейронов модифицируются, так как имеются обратные связи, которые приводят к изменению состояния сети.

Рис. 1.3. Архитектура связей НС

К основополагающим свойствам мозга относится способность к обучению. В НС обучение – это настройка архитектуры сети и весов связей для эффективного выполнения заданной задачи [3].

Веса связей НС настраиваются по обучающей выборке. По мере настройки весовых коэффициентов работа сети улучшается. По сравнению с другими системами свойства НС обучаться на примерах делает их привлекательными. Так как другие системы строятся по определенным правилам, заложенным человеком. Для обучения НС необходимы данные о модели внешней среды функционирования и понимание того как модифицировать весовые параметры сети. То есть алгоритм обучения это правила для настройки весов. НС

анализирует входную информацию как человеческий мозг. По результатам анализа НС выдает выходную информацию, которая формируется в процессе обучения. Для успешного использования пользователями НС необходимо уметь выбирать архитектуру сети, данные и интерпретировать результаты [3].

Многослойный персептрон состоит из последовательно соединенных между собой слоев искусственных нейронов. При обучении многослойного персептрона весовые коэффициенты меняются так, чтобы минимизировать среднеквадратичный функционал ошибки нейронной сети. Количество входных примеров должно быть ограничено, они могут быть дискретными и непрерывными. К преимуществам многослойных персептронов относят возможность обучения на коррелированных и зашумленных обучающих данных. Процесс обучения многоитерационный и длительный. Сферой применения многослойных персептронов являются системы распознавания образов.

Сверточные НС являются модификацией многослойного персептрона, структура которого приспособлена для распознавания двухмерных изображений с высоким уровнем шума. Их основной областью применения являются системы распознавания рукописного текста.

Сеть с радиальными базисными функциями состоит из входного, скрытого и выходного нейронных слоев. Каждый их скрытых нейронов предназначен для хранения отдельного эталонного образа, который соответствует отдельному классу. Входные параметры могут быть как дискретные, так и непрерывные. Имеет низкое качество обучения на коррелированных и зашумленных обучающих данных. Их основной областью применения являются системы распознавания образов.

Топографическая карта Кохонена состоит из входного и выходного (топографического) нейронных слоев. Количество нейронов входного слоя равно количеству компонент входных образов. Каждый входной нейрон связан с каждым топографическим нейроном, который соответствует определенному классу образов. Учебные данные могут иметь как дискретный, так и

непрерывный характер. В процессе многоитерационного обучения рассчитываются весовые коэффициенты топографических нейронов и происходит распределение библиотечных образов на классы (кластеры), количество которых определяется требуемой точностью распознавания. По сравнению с многослойным персептроном имеет значительно меньший срок обучения, что делает их эффективными для целей разведывательного анализа данных. Традиционной сферой применения является визуализация классифицированных данных в системах распознавания образов и анализа текстовой информации.

Основным преимуществом сети Хопфилда, относительно НС с прямым распространением сигнала является динамичность и итерационность обработки данных, что должно положительно влиять на вычислительные возможности. Обучение происходит путем непосредственной обработки учебных данных. Учебные образы должны быть слабо коррелированны между собой. Традиционной сферой использования сети Хопфилда является решение задач классификации зашумленных данных и выделения прототипов.

Основной особенностью сети ART является возможность динамического запоминания новых образов без полного переобучения и потери информации об образах, которые уже были в ней сохранены. Для этого в НС используется специфический по отношению к классическим НС двухуровневый алгоритм сравнения входного образа с содержимым памяти. ART может использоваться для разведывательного анализа данных и способна учиться на коррелированных учебных примерах с фиксированным количеством входных параметров. Традиционно ART используется для классификации образов и распознавания изображений.

НС являются нелинейными. Это дает возможность использовать их самостоятельно или с другими традиционными методами. Они позволяют ослабить или вообще снять проблему «проклятия размерности», которая при большом количестве переменных не может смоделировать линейные зависимости. При увеличении размерности задачи накопление погрешностей

вычислений, плохая обусловленность приводит к ухудшению решений. «Проклятие размерности» является причиной того, что во многих приложениях невозможно использовать линейную алгебру и классическую многомерную статистику.

В случаях, когда необходимо сделать предварительный анализ данных и найти зависимости между переменными НС показывают высокую эффективность. Анализируемые данные могут быть неполными, противоречивыми и даже искаженными. При существовании между входными и выходными данными какой-либо даже не обнаруживаемой корреляционными методами связи НС может на нее настроиться. Кроме того, современные НС обладают следующими возможностями: позволяют уменьшить объем входной информации, сравнивая ее и сохраняя только существенные данные, прогнозировать на основе входных данных различные ситуации, в том числе и критические и т.д.

Резюмируя вышесказанное, можно сказать, что НС – это системы, которые на основе анализа входных данных, обучения сети дают результаты обработки, даже если на входе была плохо структурированная информация. Главным отличием НС от других систем является то, что им не нужна заранее известная модель. НС строят модель на основе входной информации. Поэтому их широко используют для решения задач классификации, прогнозирования, управления и, по сути, они являются базой для организации интеллектуальных систем защиты информации [4, 5].

1.3. Методы распознавания атак с использованием нейронных сетей

Существует различные примеры применения экспертных систем и систем поддержки принятия решений (интеллектуальных систем) для обеспечения информационной безопасности и кибербезопасности компьютерных систем [6, 7]. Одним из примеров интеллектуальных систем, активно использующих НС,

являются системы обнаружения вторжений (СОВ), которые позволяют анализировать, контролировать, прогнозировать и блокировать атаки. Впервые концепция СОВ была предложена в работе [8], где предусматривался анализ системных журналов событий на предмет различных нарушений. Также в работе [9] была описана экспертная СОВ, где основное внимание уделялось профилям нормальной поведения системы, статистическому анализу данных и т.д.

Задачи современных СОВ в основном направлены на обработку информационных потоков в режиме реального времени, обеспечение адаптивности, а также отчасти на выполнение некоторых функций систем предотвращения вторжений, то есть осуществление оперативных ответных действий согласно обнаруженной атаке [10]. Такие системы обычно содержат два базовых компонента: модули слежения, представляющие собой различные датчики, детекторы, сенсоры, реализованные в виде программного или программно-аппаратного обеспечения и направленные на сбор исходных данных; управляющие модули в виде консолей, менеджеров, отвечающие за обработку собранной информации и конфигурирование модулей слежения. Также в их структуру может входить система управления базами данных для хранения данных, связанных с работой системы [11].

По методу обнаружения атак разделяют СОВ, базирующиеся на сигнатурном (шаблонном) и аномальном принципах обнаружения вторжений. Сигнатурный подход на сегодняшний день является наиболее распространенным [12]. Основная его идея заключается в описании атаки в виде шаблона (сигнатуры) и его поиска в контролируемом пространстве (например, в сетевом трафике, протоколах, журнале регистрации и др.).

Системы второго типа ориентированы на обнаружение аномального поведения. Они содержат профиль нормального (ненормального) поведения системы и обнаруживают отклонения от него [13, 14]. Эти СОВ основаны на том, что отклонение от нормального поведения системы считается аномальным состоянием. Примером такого состояния может служить большое число

соединений за короткий промежуток времени, высокая загрузка центрального процессора или использование оборудования, которое обычно не задействуется пользователем и другое.

Как отмечалось выше, средства обнаружения аномального типа направлены на построение образа нормальной активности в системе с конкретным диапазоном значений переменных и состояний, а для аномального случая необходимо учитывать более значительный диапазон изменения значений переменных и состояний. Это особенно ощутимо, если параметры, отражающие среду окружения, являются нечеткими и слабо формализованными. Более эффективными в этом отношении являются подходы, основанные на формализации суждений экспертов и их использование в процессе принятия решений о возможности осуществления атакующих действий на информационные системы и сети [10, 15].

Так как любое отклонение от заложенной нормальной активности вызовет срабатывание системы, то одним из ее существенных недостатков является высокий уровень ложных срабатываний. Во-вторых, это значительное время для получения статистических данных при использовании статистических методов, а для случая использования нейросетевых технологий – для реализации процесса обучения. И еще два обстоятельства – процесс создания соответствующего профиля нормального состояния системы довольно длительный процесс, и какие-либо изменения приводит к тому, что профиль становится неактуальным.

1.4. Биометрические методы аутентификации личности

Другим ярким примером применения НС является защита информационных ресурсов автоматизированных систем от несанкционированного доступа. Самой распространенной является парольная защита доступа к информационным ресурсам, которая обладает существенной уязвимостью, так как пользователи не всегда в состоянии запоминать

случайные пароли. У владельца информационного ресурса могут быть сомнения в том, что доступ был получен санкционированным пользователем [2]. Существуют такие угрозы парольной защиты как перехват, подмена, разглашение и другие.

В настоящее время для усиления защиты от несанкционированного доступа злоумышленников к информационным ресурсам разрабатываются технологии биометрической аутентификации личности путем преобразования личных биометрических данных человека в его криптографический ключ или длинный пароль доступа.

Криптографическая поддержка биометрических технологий является важнейшим моментом, обеспечивающим доверие к биометрическим данным. Форма, в которой осуществляется криптографическая поддержка биометрических технологий, может быть разной, однако без криптографической поддержки современная биометрия обойтись не может.

На рис. 1.4 [16] показан нейросетевой вариант связывания открытого ключа асимметричной криптографии с геометрией лица человека и нейросетевой вариант связывания рукописного пароля человека с его личным криптографическим ключом. Если охватить и тот и другой варианты связывания биометрии с ключами электронной подписью, то получим два варианта открытого и закрытого электронного удостоверения личности, которыми можно пользоваться в Интернете и иных открытых информационных пространствах. Задача создания для среды Интернет безопасных биометрических паспортов и иных удостоверений личности стоит весьма и весьма актуально.

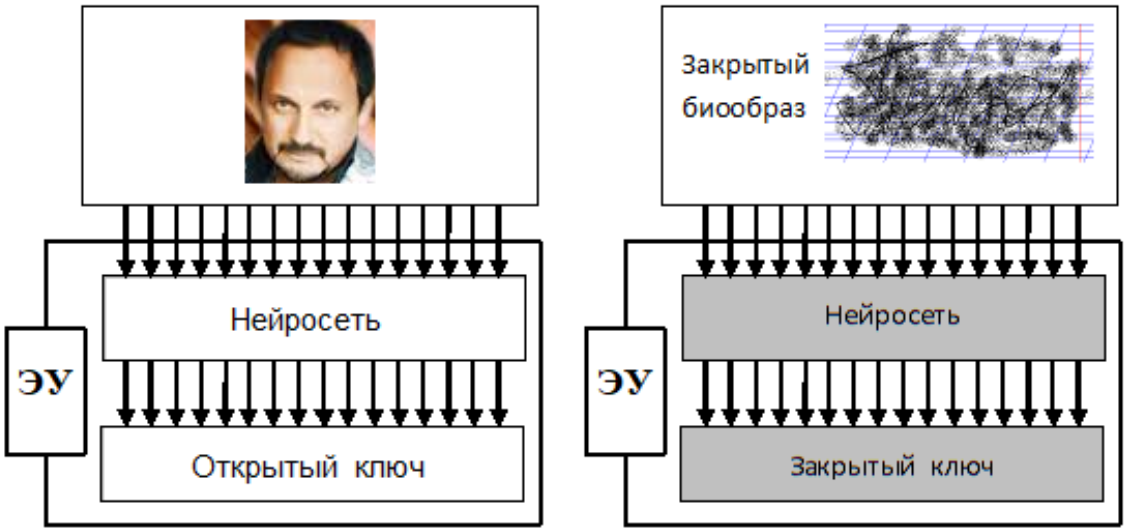


Рис. 1.4. Структура открытого и закрытого электронного удостоверения для биометрической авторизации личности пользователя

На сегодняшний день биометрию можно условно разделить на две ветви. Первая ветвь применяет открытые биометрические образы человека. Чаще всего, это биометрия паспортно-визовых документов и баз биометрических данных о преступниках. Применение открытых биометрических образов не может обеспечить человеку анонимность [16].

Вторая ветвь строится на тайне биометрического образа, которая может обеспечиваться различными путями. Важным является тот момент, что биометрия тайных образов человека намного сильнее биометрии открытых образов. Причиной этого является то, что открытые биометрические образы – это просто биометрия, тайные биометрические образы – это биометрия, умноженная на «тайну» рукописного пароля, голосового пароля, отпечатка пальца неизвестного лица [16]. Для получения доступа к биометрии тайных образов злоумышленник вынужден создавать огромные базы из случайных биометрических образов, принадлежащих разным людям и далее пытаться их предъявлять биометрическому программному роботу, хранящему личный ключ своего хозяина.

1.5. Анализ биометрических технологий распознавания образов

Все биометрические технологии распознавания образов делятся на три группы. К первой группе относятся технологии, основанные на анализе статических характеристик человека, а ко второй группе относятся технологии, основанные на анализе динамических характеристик человека. В третьей группе можно сочетать технологии первой и второй группы, то есть использовать мультибиометрические технологии [1, 2].

Статическими характеристиками человека называются характеристики неотделимые от человека и данные от рождения. Наиболее распространенными являются отпечаток пальца, радужная оболочка глаза, сетчатка глаза, геометрия руки или лица и другие. Статические характеристики человека практически не зависят от психофизиологического состояния пользователей, не требуют много времени на регистрацию, а стойкость подбора высокая (от 10^2 до 10^{13} попыток) [1, 17].

К динамическим характеристикам человека относятся характеристики, приобретённые со временем или же способные меняться с возрастом или внешним воздействием. Наиболее распространенными являются распознавание личности по голосу и рукописному почерку [17].

В работе были проанализированы наиболее распространенные технологии распознавания биометрических образов: отпечаток пальца, геометрия лица, радужная оболочка глаза, геометрия руки, рукописная подпись и голосовая фраза. При проведении анализа рассматривалась стойкость технологии, которая оценивается ошибками первого рода (ОПР) и ошибками второго рода (ОВР), стоимости реализации системы и информативности биометрического образа. Информативность биометрического параметра – v вычисляется по формуле [16]:

$$I(v) = -\log_2(P_2(v)) .$$

При нулевой вероятности ОПР, чем выше информативность, тем ниже

вероятность ОВР.

Результаты анализа биометрических технологий приведены в таблице 1.1 и на рисунке 1.5.

Таблица 1.1 - Сравнительный анализ биометрических технологий с точки зрения стоимости ее реализации, стойкости защиты и информативности биометрического образа

Технология	Стоимость	Стойкость	Информативность
1. Анализ рисунка отпечатка пальца	от 60 до 300 долларов	ОПР ≈ 2%. ОВР на уровне 10^{-5}	от 10 до 20 бит
2. Анализ геометрии лица	от 40 до 200 долларов	Вероятность ОВР удается снизить до величины 10^{-5}	до 16,6 бит
3. Анализ радужной оболочки глаза	от 1500 до 6500 долларов	Вероятность ОВР составляет от 10^{-5} до 10^{-11}	от 16 до 35 бит
4. Анализ геометрии руки	от 150 до 480 долларов	Вероятность ОВР составляет от 10^{-2} до 10^{-4}	не менее 6,6 бит
5. Анализ голоса	от 60 до 120 долларов	ОПР и ОВР составляют от 2 до 1%	около 3 бит
6. Анализ динамики рукописного почерка	от 40 до 70 долларов	Вероятности ОВР на уровне от 10^{-2} до 10^{-4} для открытых рукописных образов в форме автографов и	Зависит от длины рукописного парольного слова и может составлять для имени человека 6,5-13 бит, а

		рукописного воспроизведения имени человека. При использовании тайного рукописного парольного слова из пяти букв вероятность ОВР от 10^{-4} до 10^{-8}	для парольного слова из пяти букв 13-26,5 бит. При использовании нескольких рукописных слов информативность растет пропорционально числу используемых слов.

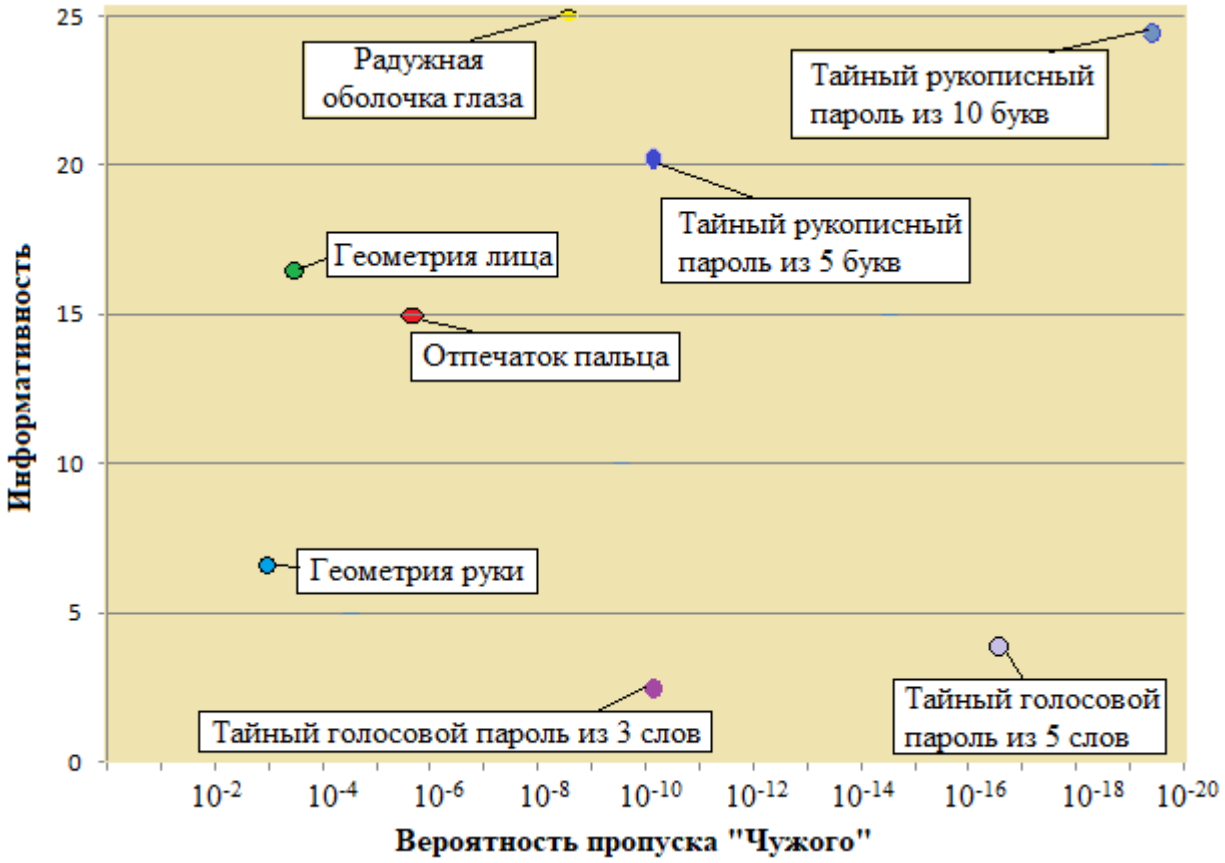


Рис. 1.5. Сравнительный анализ биометрических образов распознавания личности по стойкости и информативности

Сравнительный анализ показал, что средства статической биометрии стоят дороже, чем динамические. Во-первых, самая дешевая технология распознавания биометрических образов по отпечаткам пальцев стоит около $100, тогда как затраты на средства биометрические аутентификации по

голосовой фразе и рукописному почерку гораздо ниже. Это обуславливается наличием аппаратных устройств ввода голосовой и рукописной информации в карманных компьютерах и смартфонах. Во-вторых, вероятность ОВР в статических метода высока по сравнению с динамическими. Например, для упомянутой выше технологии распознавания биометрических образов по отпечаткам пальцев она составляет порядка 10^{-6}. Стойкость же тайного пятибуквенного рукописного пароля биометрико-нейросетевой защиты для атаки среднестатистического пользователя находится на уровне 10^{-9}. Понятно, что стойкость защиты зависит от длины используемого пароля. Так, если рукописная парольная фраза будет состоять из двух пятибуквенных слов, тогда стойкость защиты от подбора возрастет до 10^{-18} возможностей. Для подбора неизвестной биометрии комбинациями из 10^{-18} возможностей для компьютера средней производительности потребуется несколько столетий непрерывной работы.

1.6. Технологии распознавания образов на основе модели нечетких экстракторов

На сегодняшний день США и страны Евросоюза биометрические параметры человека связывают с криптографическим ключом с помощью «нечетких экстракторов» [16, 18-24]. «Нечеткие экстракторы» [25] это специальные обогатители (алгоритмы) превращающие бедную неоднозначную размытую биометрическую информацию в сильный личный ключ пользователя (последовательности битов). Так как нет возможности вводить абсолютно одинаково биометрические параметры, «нечеткие экстракторы» могут компенсировать возникающие из-за этого ошибки. Выделенные последовательности битов квантуются, а затем его ошибки корректируются путем применения классических избыточных кодов с обнаружением ошибок [26].

Любой контролируемый биометрический параметр можно представить как функцию, изменяющуюся во времени. Например, в виде функции можно представить изменение яркости участков радужной оболочки глаза при радиальном сканировании, изменение амплитуды колебаний звуковой волны при сканировании голоса или координаты колебаний пера при воспроизведении рукописного пароля. Пример одного из таких колебаний приведен на рис. 1.6 [16].

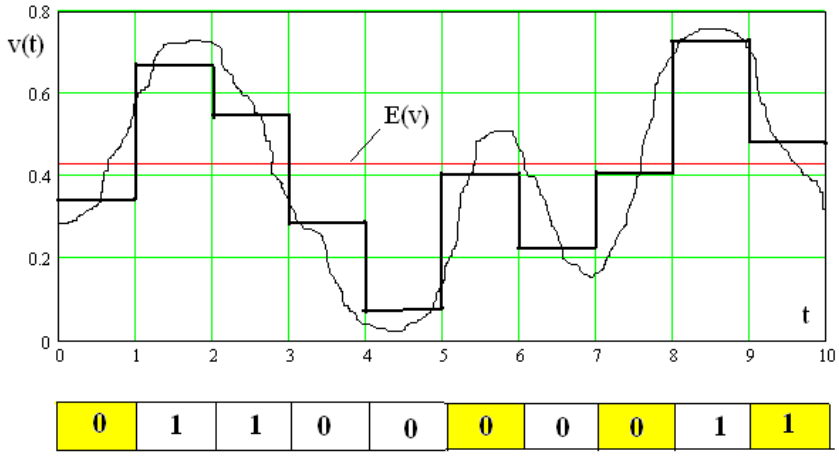


Рис. 1.6. Квантование кривой изменения биометрического параметра v(t) относительно среднего значения

Для простейшего квантования кривой изменения биометрических данных разобьем время на заданное число интервалов (например, на рисунке 1.6. 10 интервалов). Затем в каждом интервале квантования посчитаем среднее значение параметра и сравним со средним значением по всем интервалам наблюдения E(v). Если на интервале среднее значение параметра v(t) больше E(v), то этому разряду биометрического кода присвоим «1». Если на интервале среднее значение параметра v(t) меньше E(v), то этому разряду биометрического кода присвоим «0» и таким образом получим биометрический код (нижняя часть рисунка 1.6). Разряды биометрического кода, которые попали на участки квантования с примерно одинаковыми отрицательными и положительными фрагментами кривой v(t) называются нестабильными.

Таковыми являются примерно половина разрядов биометрического кода. На рисунке они отмечены темной заливкой. На практике до 80% наиболее нестабильных разрядов биокода маскируют, а оставшиеся 20 % исправляют с применением кодов, способных обнаруживать и корректировать ошибки.

Биометрический код человека можно защитить различными способами, одним из которых является применение криптографии. В криптографии очень много алгоритмов и для реализации «нечетких экстракторов» было выбрано гаммирование [16, 27-30]. Схема работы «нечетких экстракторов» с гаммированием биометрического кода личности представлена на рис. 1.7 [16].

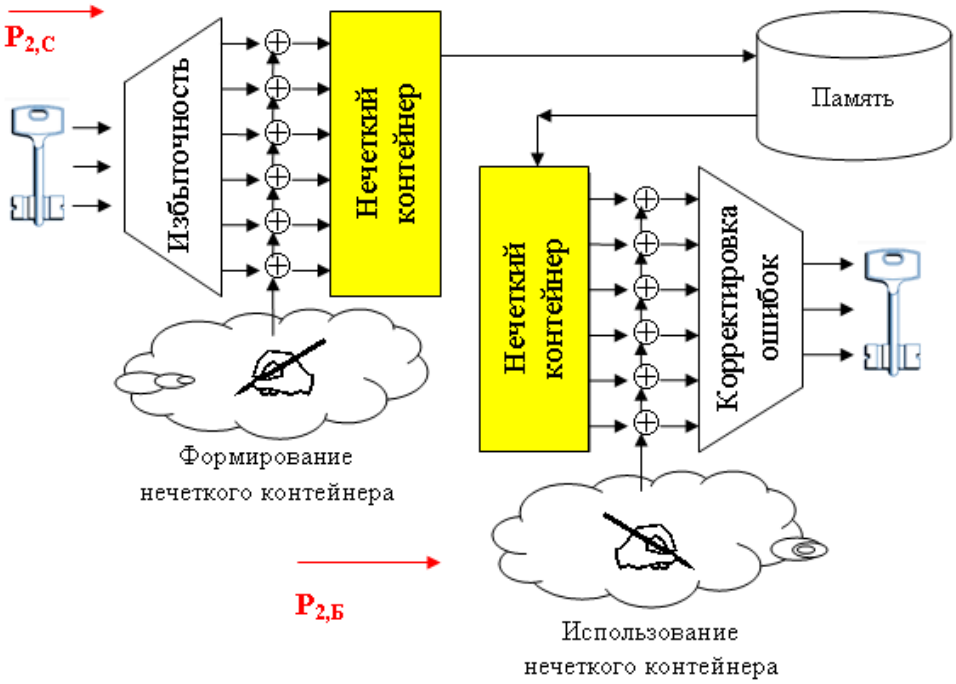


Рис. 1.7. Схема работы «нечетких экстракторов» с гаммированием биометрического кода личности

Для защиты биометрических кодов используют личный секретный криптографический ключ человека, к которому применяют избыточный самокорректирующийся код. Обычно применяют коды БЧХ (Боуза-Чоухуры-Хоквингема) и поэтому получают гамму в 10 раз длиннее секретного ключа. Далее к биометрическому коду применяют гамму для получения «нечеткого

контейнера», который сохраняют для проведения процесса аутентификации [16, 31].

При проведении аутентификации вводят биометрический образ, оцифровывают его и гаммируют с сохраненным «нечетким контейнером». В результате получают избыточный самокорректирующийся код криптографического ключа. В полученном коде содержатся ошибки, возникшие при формировании «нечеткого контейнера» и биометрического кода аутентификации. Эти ошибки исправляются, если не превышают исправляющую способность самокорректирующегося кода [16, 31].

«Нечеткие экстракторы» интересны тем, что позволяют в явной форме разделить стойкость криптографической защиты биометрического кода от наблюдения и стойкость биометрической защиты от атак подстановки случайных образов «Чужой». Стойкость криптографической защиты будет пропорциональна длине личного ключа, а биометрическая стойкость защиты пропорциональна длине биокода и показателям стабильности его разрядов.

1.7. Выводы к главе 1

1. Рассмотрены методы распознавания атак на ресурсы информационных систем и методы распознавания биометрических образов для защиты информационных ресурсов от несанкционированного доступа. Показано, что для их эффективной реализации наиболее перспективным является использование возможностей НС.

2. Проведено исследование наиболее распространенных технологий распознавания биометрических образов: отпечаток пальца, геометрия лица, радужная оболочка глаза, геометрия руки, рукописная подпись и голосовая фраза. Был дан их сравнительный анализ с точки зрения стоимости реализации, стойкости защиты и информативности образа. Из статических методов бесспорным было преимущество технологии распознавания по отпечатку пальца, из динамических – по рукописному почерку.

ГЛАВА 2. МОДЕЛИ И МЕТОДЫ РАСПОЗНАВАНИЯ БИОМЕТРИЧЕСКИХ ОБРАЗОВ С ИСПОЛЬЗОВАНИЕМ НЕЙРОННЫХ СЕТЕЙ

Для распознавания биометрических образов используем модель биометрия-код (рис. 2.1.), которая позволяет преобразовать биометрический образ в криптографический код длиной 256 бит. Биометрические образы делятся на образы «Свой» и «Чужой». К образу «Свой» относится биометрический образ легитимного пользователя, а биометрический образ злоумышленника относится к образу «Чужой» [32, 33].

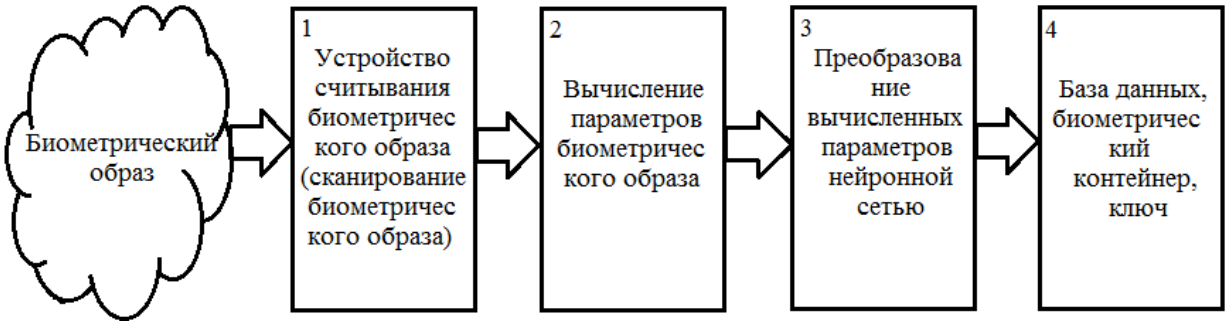


Рис. 2.1. Преобразование биометрических параметров на основе модели биометрия-код

Согласно рис. 2.1. биометрический образ проходит три этапа и преобразуется в криптографический ключ для образа «Свой» или «белый шум» для образа «Чужой», кроме этого создается база данных «Чужой» и биометрический контейнер, который заносится в базу данных «Свой» [33].

В качестве биометрических образов будут использоваться отпечатки пальцев и рукописный почерк.

В следующем разделе рассмотрим выбранные для сканирования биометрических образов устройства считывания.

2.1. Сканирование биометрического образа

Первым этапом преобразования биометрических параметров является сканирование биометрического образа с помощью устройства считывания.

В качестве устройства считывания для сканирования папиллярных рисунков отпечатков пальцев будем использовать оптический сканер отпечатков пальцев Futronic FS-80 (рис. 2.2.). Критериями, по которым была выбрана данная модель, являются: хорошее качество формируемого изображения, диапазон рабочих температур, надежность устройства, приемлемая цена [17, 20].

Рис. 2.2. Сканер Futronic FS-80

Futronic FS-80 сканирует отпечаток пальца через 14 миллиметровое стеклянное окно, которое имеет защитное напыление, предотвращающее механические повреждения и быстрый износ устройства, и позволяет получать качественный рисунок отпечатка пальца без его протяжки [17, 20].

Данная модель сканера отпечатка пальцев использует разработанную и запатентованную компанией Futronic технологию «Live Finger Detection (LFD)», что означает «Распознавание живого пальца», которая предотвращает использование злоумышленниками муляжей отпечатков пальцев, сделанных из резины, силикона и другого материала для доступа к важной информации [17, 20].

В качестве преобразователя графического начертания естественных рукописных образов применяется графический планшет GenusWizardPen 5x4

(рис. 2.3.), состоящий из поля размером 14x10 см с разрешающей способностью до 4064 линий/дюйм, беспроводного пера, позволяющего оцифровывать до 1024 градации силы нажатия. Графический планшет соединяется с персональным компьютером при помощи USB порта. Используется трех координатная система, позволяющая учитывать перемещения пера по x(t), y(t) и силу его нажатия на планшет z(t).

Рис. 2.3. Графический планшет Genius G-Pen F350

Кроме того, для ввода рукописного почерка может быть использовано любое устройства, где есть приложение для ввода рукописных образов.

2.2. Обоснование выбора наиболее эффективного вида модели для распознавания рукописного почерка

Как показывают результаты исследований, проведенные в первом разделе диссертации, одной из основных тенденций развития современных нейросетевых систем биометрической аутентификации пользователей, базирующихся на распознавании рукописных символов, является необходимость реализации анализа рукописных текстовых фрагментов неопределенной длины. Другими словами нейросетевая система должна иметь возможность динамически настраиваться на анализ различных объемов рукописных символов. Это в первую очередь позволяет изменять длину

парольных данных, что в соответствии с [34, 35], положительно сказывается на стойкости системы аутентификации в целом. Отметим, что в известных нейросетевых системах биометрической аутентификации возможность в широких пределах изменять длину рукописного пароля отсутствует. Это в первую очередь можно объяснить одним из принципиальных ограничений классических нейросетевых моделей лежащих в основе блока распознавания таких систем - фиксированное количество входных параметров.

Таким образом, для добавления в нейросетевую систему биометрической аутентификации указанной возможности нам необходимо разработать новую нейросетевую модель, которая с одной стороны должна быть адаптирована к неопределенному количеству входных параметров, а с другой должна быть адаптирована к особенностям ожидаемых условий эксплуатации.

В качестве отправной точки построения нейросетевой модели нами использована хорошо апрбированная и научно обоснованная методология разработки эффективных нейросетевых систем защиты информации, изложенная в работе [34]. Укрупнено эта методология предполагает двухэтапное построение нейросетевой модели:

- обоснование выбора наиболее эффективного вида модели;
- определение параметров модели эффективного вида.

При этом для обоснования выбора наиболее эффективного вида модели нами использован **принцип** – наиболее эффективным является тот вид нейросетевой модели, характеристики которого наиболее полно соответствуют значимым условиям поставленной задачи защиты информации [36]. В базовом варианте предлагается разделить множество значимых условий на категории, характеризующие учебные данные, ограничения процесса обучения, вычислительные мощности, выходную информацию, техническую реализацию и сферу применения нейросетевых средств. Таким образом, задача выбора наиболее эффективного вида нейростевой модели сводится к задаче многофакторной оптимизации, описываемой выражением:

$$E_{\Sigma}(a_i) = \sum_{k=1}^{K} E_k(a_i) \to \max,\ a_i \in \mathrm{A},\ i = 1,2,...,I \qquad (2.1)$$

где E_{Σ} - интегральный критерий оптимизации вида нейросетевой модели, a_i - i-ий вид нейросетевой модели, А, I - множество и количество допустимых видов нейросетевых моделей, E_k - k-ый критерий эффективности, K – количество критериев эффективности.

Также в [34] приведен универсальный перечень базовых критериев эффективности, используемых в выражении (2.1). В этих же работах сформулирована рекомендация адаптировать универсальный перечень к поставленной задаче защиты информации. Отметим, что необходимость модификации вызвана также достаточно большим прогрессом в развитии теории НС, который произошел после публикации [34]. Например, за счет функции активации типа softmax возможность представления выходного сигнала в вероятностном виде в настоящее время может быть реализована практически для любого вида нейросетевых моделей. Следовательно, соответствующий базовый критерий оценки эффективности использовать нецелесообразно.

Модификация указанного перечня, проведенная нами с учетом современного состояния теории нейронных сетей и учитывающая специфику задачи биометрической аутентификации на основании анализа рукописного текста неопределенной длины, приведена в таблице 2.1.

Таблица 2.1- Критерии эффективности нейросетевых моделей

№	Категория	Объяснение критерия
E_1	Учебные данные	Возможность использовать учебные примеры с различным количеством входных параметров
E_2		Возможность обучения на ограниченной выборке
E_3		Допустимость шума в учебных данных

E_4		Допустимость корреляции учебных примеров
E_5		Необходимость пропорционального представления примеров, каждого из распознаваемых классов
E_6		Возможность использования дискретных входных параметров
E_7		Возможность использования маркированных учебных примеров
E_8		Возможность использования не маркированных учебных примеров
E_9	Процесс обучения	Короткий срок обучения
E_{10}		Автоматизация обучения
E_{11}	Процесс обучения	Возможность дообучения
E_{12}		Качество обучения
E_{13}		Возможность обучения на экспертных данных
E_{14}		Возможность преднастройки весовых коэффициентов
E_{15}		Возможность параллелизации процесса обучения
E_{16}		Возможность выделения сетью иерархических признаков, распознаваемых объектов
E_{17}	Вычислительная мощность	Объем памяти (отношение количества правильно запомненных примеров к количеству весовых коэффициентов)
E_{18}		Экстраполяция результатов обучения за пределы обучающей выборки
E_{19}	Техническая реализация	Скорость принятия решения
E_{20}		Объем программной реализации
E_{21}		Объем используемой памяти
E_{22}	Апробированная сфера применения	Моделирование временных рядов
E_{23}		Анализ изображений
E_{24}		Распознавание рукописного текста

Дальнейший ход процесса обоснования выбора наиболее эффективного вида нейросетевой модели обусловлен результатами анализа современных нейросетевых систем биометрической аутентификации и нейросетевых систем распознавания рукописного текста. Эти результаты показывают, что в таких системах в основном применяются следующие виды среди нейросетевых моделей: глубокие НС на базе автоэндкодера, глубокие НС с функцией активации ReLU, двухслойный персептрон, сверточные НС и рекуррентные НС долгой краткосрочной памяти. Соответственно приведенным данным выражение (2.1) возможно модифицировать так:

$$E_{\Sigma}(a_i) = \sum_{k=1}^{24} E_k(a_i) \to \max,\ a_i \in \mathrm{A},\ i = 1,2,...,5, \tag{2.2}$$

$$\mathrm{A} = \{\mathrm{DNN_a}, \mathrm{DNN_r}, \mathrm{MLP_2}, \mathrm{CNN}, \mathrm{LSTM}\}, \tag{2.3}$$

где $\mathrm{DNN_a}$ - глубокая НС на базе автоэндкодера, $\mathrm{DNN_r}$ - глубокая НС с функцией активации ReLU, $\mathrm{MLP_2}$ - двухслойный персептрон, CNN - сверточная НС, LSTM - рекуррентная НС долгой краткосрочной памяти.

Также в соответствии с рекомендациями [34] значимость критериев для поставленной задачи биометрической аутентификации учтена нами за счет ввода в (2.2) соответствующих весовых коэффициентов:

$$E_{\Sigma}(a_i) = \sum_{k=1}^{24} (r_k \times E_k(a_i)),\ a_i \in \mathrm{A},\ i = 1,2,...,5, \tag{2.4}$$

$$E_{\Sigma}(a_i) \to \max,\ a_i \in \mathrm{A},\ i = 1,2,...,5, \tag{2.5}$$

где r_k – весовой коэффициент k-го критерия оптимизации.

Результаты [34, 35, 37] позволили в первом приближении выставить оценки соответствия основных видов нейросетевых моделей принадлежащих множеству А, критериям эффективности перечисленным в таблице 2.1.

Указанные оценки выставлены по трехбалльной шкале и приведены в таблице 2.2. Критерий E_i=1, если i-тая характеристика задачи биометрической аутентификации полностью обеспечивается в данном виде нейросетевой модели, E_i=0,5 – если характеристика может быть обеспечена за счет оптимизации параметров нейросетевой модели данного вида и E_i=0 – если не обеспечивается.

Таблица 2.2 - Величины критериев эффективности

№	**Вид нейросетевой модели**				
	DNN_a	DNN_r	MLP_2	CNN	LSTM
E_1	0	0	0	0	1
E_2	0	0	0	0,5	0,5
E_3	0,5	0,5	0,5	0,5	0,5
E_4	0,5	0,5	0,5	0,5	0,5
E_5	0,5	0,5	0,5	0,5	0,5
E_6	1	1	1	1	1
E_7	1	1	1	1	1
E_8	1	1	1	1	1
E_9	0,5	1	0,5	0,5	0,5
E_{10}	0,5	0,5	1	1	1
E_{11}	1	0	0	0	0,5
E_{12}	0,5	1	1	1	1
E_{13}	0	0	0,5	0	0
E_{14}	0,5	0	0	0	0
E_{15}	1	0,5	0,5	0,5	0,5
E_{16}	0,5	0,5	0	0,5	0,5
E_{17}	0,5	0,5	0,5	1	1
E_{18}	1	1	0,5	1	1
E_{19}	1	1	1	1	0,5

E_{20}	0,5	1	1	1	1
E_{21}	1	1	0,5	1	1
E_{22}	0	0	0	0,5	1
E_{23}	0,5	0,5	0,5	1	0,5
E_{24}	0,5	0,5	0,5	1	0,5

Также базируясь на данных [34, 35, 37] для каждого из показанных в таблице 2.2 критериев эффективности, нами определены величины весовых коэффициентов. Указанные величины весовых коэффициентов приведены в таблице 2.3.

Таблица 2.3 - Величины весовых коэффициентов для критериев эффективности

r_1	r_2	r_3	r_4	r_5	r_6	r_7	r_8
1	0,5	0,3	0,3	0,3	1	1	0,2
r_9	r_{10}	r_{11}	r_{12}	r_{13}	r_{14}	r_{15}	r_{16}
0,5	0,5	0,3	0,9	0,3	0,3	0,3	0,5
r_{17}	r_{18}	r_{19}	r_{20}	r_{21}	r_{22}	r_{23}	r_{24}
0,5	0,5	0,3	0,3	0,3	0,5	1	1

В результате для каждого исследуемого вида нейросетевых моделей рассчитана величина интегрального критерия эффективности. Суть расчетов сводилась к подстановке данных таблицы 2.2 и таблицы 2.3 в выражение (2.4). Полученные величины представлены в таблице 2.4.

Таблица 2.4 - Величины интегрального критерия эффективности для апробированных видов нейросетевых моделей

Величина интегрального критерия	Вид нейросетевой модели				
	DNN_a	DNN_r	MLP_2	CNN	LSTM
E_Σ	6,7	6,75	6,35	7,35	9,35

Использовав данные таблицы 2.4 и выражение (2.5) нами определено, что наиболее эффективными видами нейросетевых моделей являются сверточная НС и рекурентная НС долгой краткосрочной памяти. Таким образом, можно предположить, что наибольшие перспективы имеет нейросетевая модель интегрирующая в себе возможности указанных типов сетей. Указанное предположение подтверждается подобными выводами, обоснованными в работах [34, 35, 37-39].

2.3. Композитная нейросетевая модель распознавания пользователей на основе анализа геометрических параметров фрагментов рукописного текста изменяемого размера

Определение эффективного вида нейросетевой модели позволило нам перейти к следующему этапу разработки, направленному на конкретизацию параметров модели. При этом в доступной литературе целостная методология оптимизации параметров сверточной НС, использующей LSTM-модули, отсутствует. Поэтому вначале нами проведены исследования с целью определения перечня возможных оптимизируемых параметров такой сети. При этом в качестве прототипа нами использована разработанная в [40] нейросетевая модель вида CNN-LSTM, структура которой показана на рис. 2.4.

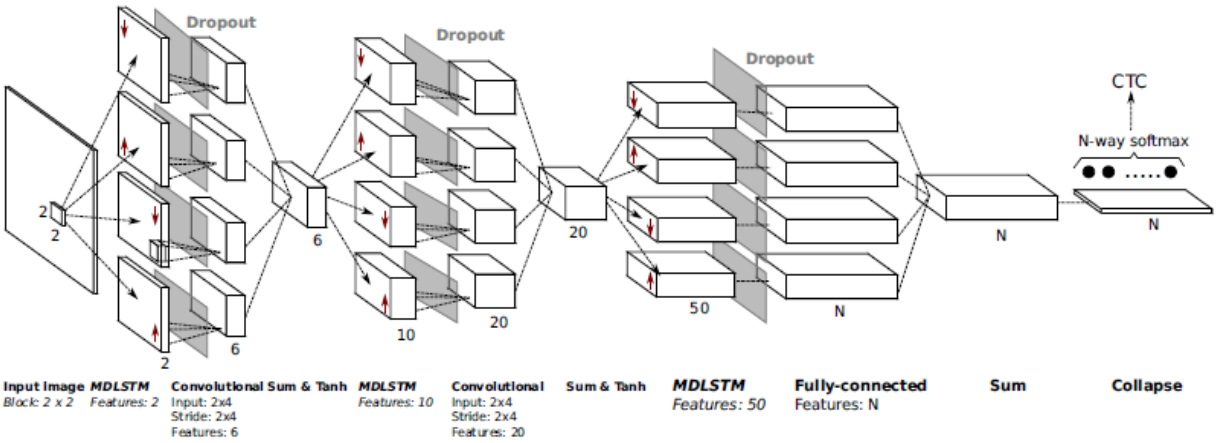


Рис. 2.4. Структура нейросетевой модели CNN-LSTM

По сути, структура этой сети повторяет, показанную на рис. 2.5, структуру классической сверточной НС. Основной особенностью является то, что каждый из сверточных нейронных слоев представляют собой отдельный LSTM-модуль, структура которого показана на рис. 2.6. Таким образом, в первом приближении оптимизацию параметров нейросетевой модели CNN-LSTM мы рассмотрели с точки зрения оптимизации сверточной НС с учетом указанной особенности.

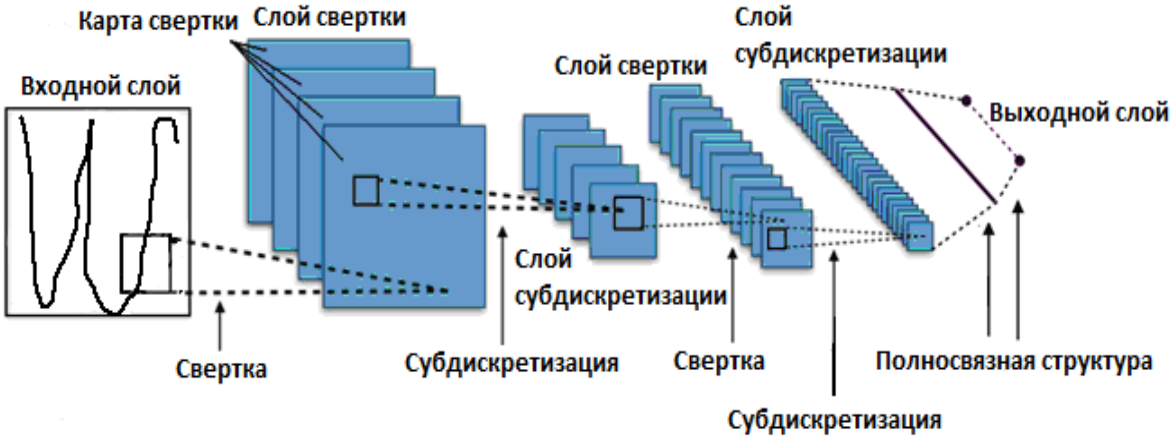


Рис. 2.5. Структура сверточной НС

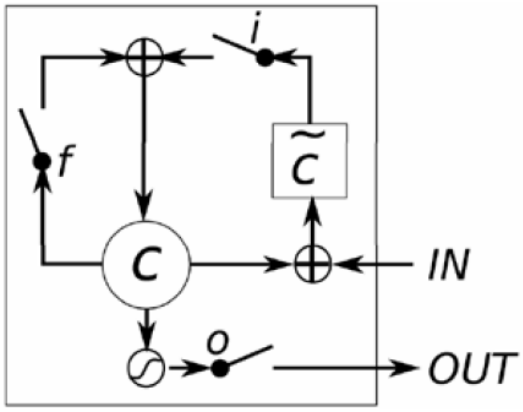


Рис. 2.6. Структура LSTM-модуля

Функционирование LSTM-модуля подобно функционированию на классической рекурентной НС, для которой значение выходного сигнала в момент времени *t* вычисляется следующим образом:

$$y^{(t)} = f\left(Wx^{(t)} + Uy^{(t-1)} + b^{y}\right), \tag{2.6}$$

где $x^{(t)}$, $y^{(t)}$- входной и выходной вектор сети в момент времени *t*, W, U, b^{y} - параметры, изменяемые в процессе обучения (обучаемые параметры), f - функция нелинейного преобразования.

В качестве нелинейного преобразования часто используют сигмоид (2.7), гиперболический тангенс (2.8) или ReLU (2.9):

$$f(x) = \frac{1}{1+e^{-x}}, \tag{2.7}$$

$$f(x) = \tanh(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}}, \tag{2.8}$$

$$f(x) = \max(0, x). \tag{2.9}$$

Отличительной особенностью LSTM-модуля является механизм вычисления выходного сигнала, в котором используются входные значения, предыдущее состояние сети и фильтры (gates). Фильтры определяют как

информация будет использоваться для вычисления выходных значений на текущем слое, значений скрытого слоя на следующем шаге. Для этого используется входной фильтр, фильтр забывания и выходной фильтр. Также в LSTM-модуле предусмотрен запоминающий блок (memory cell), в котором вычисляется состояние сети *h*. Для проведения вычислений необходимы текущее входное значение $x^{(t)}$ и значение блока на предыдущем шаге $c^{(t-1)}$.

Назначением входного фильтра (input gate) $i^{(t)}$ является определение того как на текущем шаге значение блока памяти должно влиять на результат. Выходной сигнал фильтра рассчитывается так:

$$i^{(t)} = \frac{1}{1+e^{-\left(W^i x^{(t)} + U^i h^{(t-1)}\right)}}, \tag{2.10}$$

Выход входного фильтра $i^{(t)}$ может изменяться от 0 до 1. При нуле входные значения вообще не учитываются, при единице учитываются полностью. «Фильтр забывания» (forget gate) дает возможность исключить при вычислениях значения памяти предыдущего шага:

$$f^{(t)} = \frac{1}{1+e^{-\left(W^f x^{(t)} + U^f h^{(t-1)}\right)}}, \tag{2.11}$$

Полученные с помощью выражений (2.10, 2.11) значения $i^{(t)}$, $f^{(t)}$ используются для вычисления состояние блока памяти на текущем шаге. Для этого применяются выражения (2.12, 2.13)

$$\tilde{c}^{(t)} = \tanh\left(W^c x^{(t)} + U^c h^{(t-1)}\right), \tag{2.12}$$

$$c^{(t)} = f^{(t)} \circ c^{(t-1)} + i^{(t)} \circ \tilde{c}^{(t)}, \tag{2.13}$$

Расчет выходного сигнала выходного фильтра (output gate) реализуется так:

$$o^{(t)} = \frac{1}{1+e^{-\left(W^o x^{(t)} + U^o h^{(t-1)}\right)}} \ , \tag{2.14}$$

Результирующий выходной сигнал LSTM-модуля вычисляется с помощью выражения:

$$h^{(t)} = o^{(t)} \circ \tanh\left(c^{(t)}\right) . \tag{2.15}$$

Следует отметить, что значения весовых коэффициентов $W^i, U^i, W^f, U^f, W^c, U^c, W^o, U^o$ рассчитываются в процессе обучения LSTM-модуля, который реализуется на базе общеизвестного **алгоритма обратного распространения ошибки** [37-40]. На сегодня известны некоторые модификации LSTM-сетей, которые хотя и имеют несколько меньшую ресурсоемкость, но проигрывают классическому варианту сети в гибкости обработки. Также анализ структурных особенностей и математического аппарата LSTM-модуля позволяет нам утверждать, что он является атомарным объектом, а следовательно адаптировать его к особенностям поставленной задачи не требуется. Поэтому к поставленной задаче распознавания рукописного текста неопределенной длины в композитной нейросетевой модели CNN-LSTM целесообразно в первую очередь адаптировать структурные параметры, унаследованные ею от сети сверточной НС.

Входное поле сверточной НС соответствуют размеру распознаваемого изображения. Поэтому количество входных параметров равно размеру такого изображения. Количество выходных нейронов равно количеству распознаваемых образов. Структура скрытых нейронных слоев подбирается эмпирически.

Суммарный входной сигнал некоторого нейрона сверточного слоя рассчитывается так:

$$x_k^{(i,j)} = f\left(x_{0,k} + \sum_{s=1}^{K}\sum_{t=1}^{K} w_{k,s,t} x^{((i-1)+s,(j+t))} \right), \tag{2.16}$$

где $x_k^{(i,j)}$ - входной сигнал (i,j)-го нейрона k-ой карты признаков, $x_{0,k}$ - смещение нейронов k-ой карты признаков, K - размер рецептивной области нейрона (размер ядра свертки), $w_{k,s,t}$ - весовой коэффициент (s,t)-ой синаптической связи нейрона k-ой карты признаков, x - выход нейрона предыдущего слоя.

Выходной сигнал нейрона карты признаков рассчитывается путем подстановки входного сигнала в функцию активации. В основном применяются функции вида (2.7, 2.8, 2.9). В нейронах выходного слоя как правило используется функция активации типа Softmax:

$$y_i = \frac{\exp(q_i)}{\sum_{k=1}^{L_{out}} \exp(q_k)}, \tag{2.17}$$

где y_i - выход i-го нейрона выходного слоя, q_k - суммарный входной сигнал для k-го нейрона выходного слоя, L_{out} - количество выходных нейронов.

На основании теоретических работ, посвященным сверточным НС мы можем утверждать, что их основными структурными параметрами являются:

- Размер входного поля - a_0.

- Количество входных нейронов - L_{in}.

- Количество выходных нейронов - L_{out}.

- Количество нейронов в полносвязном слое - L_f.

- Количество сверточных слоев - K_{ls}.

- Количество карт признаков в каждом сверточном слое - $L_{h,k}$.

- Количество слоев подвыборки (субдискретизации) - K_{ld}.

- Масштабный коэффициент для каждого слоя подвыборки - m_l. При этом размер l-го слоя подвыборки рассчитывается так:

$$c_l = a_k / m_l \,, \tag{2.18}$$

где a_k - размер сверточного слоя, который предшествует l-му слою подвыборки.

- Размер ядра свертки для каждого k-го сверточного слоя $(b \times b)_k$.

- Смещение рецептивного поля при выполнении каждой k-ой процедуры свертки $d_k, k \in [1, K_{ls}]$.

- Размер карты признаков для каждого k-го сверточного слоя - $(a \times a)_k$. При этом:

$$a_k = (a_{k-1} - b_k + 2r_k)/d_k + 1. \tag{2.19}$$

где r_k - количество дополняющих нулей для k-го сверточного слоя.

- Структура связей между соседними слоями свертки/подвыборки.

Очевидно, что именно эти параметры нами могут быть адаптированы к условиям применения сверточной НС.

С учетом необходимости минимизации ошибки распознавания модель оптимизации структурных параметров сверточной НС мы можем записать с помощью выражения

$$\begin{cases} \Delta\left(L_{in}, L_{ls}, L_{out}, K_{h,k}, b_k, K_{ls}, \left|\mathbf{Q}_{\mathbf{i,i+1}}\right|_{K_{ls}}\right) \to \min \\ R \le R_{\max} \end{cases}, \tag{2.20}$$

где Δ - ошибка распознавания, $\left|\mathbf{Q_{i,i+1}}\right|_{K_{ls}}$ - вектор, состоящий из матриц которые определяют связи между соседними скрытыми слоями нейронов, R-ресурсоемкость сети, $R_{\max}$ - максимально допустимая ресурсоемкость сети.

Также известны четыре принципа адаптации структуры сверточной НС к задаче биометрической аутентификации пользователя на основании анализа двухмерной геометрии их биометрических образов, отображаемых в окне фиксированного размера. Учитывая поставленную задачу распознавания рукописного текста эти принципы можно сформулировать так:

Принцип 1. Количество сверточных слоев должно быть равно количеству уровней распознавания двухмерного изображения рукописных символов среднестатистическим пользователем.

Принцип 2. Количество карт признаков в n-ом сверточном слое должно быть равно количеству признаков на n-ом уровне распознавания.

Принцип 3. Размер ядра свертки для n-го сверточного слоя должен быть равен размеру распознаваемых признаков на n-ом иерархическом уровне.

Кроме этого, исходя из практического опыта применения сверточных НС [34, 35] можно сделать вывод о том, что использование сверточных слоев может привести к затенению распознаваемых признаков. Поэтому нами предложен еще один принцип адаптации.

Принцип 4. Использование сверточных слоев не должно искажать геометрические параметры признаков, используемых для распознавания рукописных символов.

Также в известных принципах адаптации не учтена возможность анализа изображения произвольного размера. Отметим, что в разрабатываемой нейросетевой модели CNN-LSTM этот недостаток нивелируется за счет рекурентно-последовательной подачи на вход сети фрагментов анализируемого изображения. Очевидно, что сеть должна быть адаптирована к размеру указанного фрагмента. Ведь именно этот размер определяет атомарный блок

входных параметров нейросетевой модели, что и учитывается следующим принципом адаптации:

Принцип 5. Размер фрагмента анализируемого изображения, используемого в качества атомарного блока входных данных сети, равен минимально допустимому размеру символа рукописного текста.

Формирование математического обеспечения (2.6-2.20), а также указанных принципов адаптации позволило нам перейти к следующему этапу разработки нейросетевой модели. На этом этапе реализована адаптация значений параметров CNN-LSTM к задаче распознавания пользователей на основании геометрии принадлежащего им фрагментов рукописного текста. На рис. 2.7 показана укрупненная блок-схема алгоритма адаптации значений параметров CNN-LSTM.

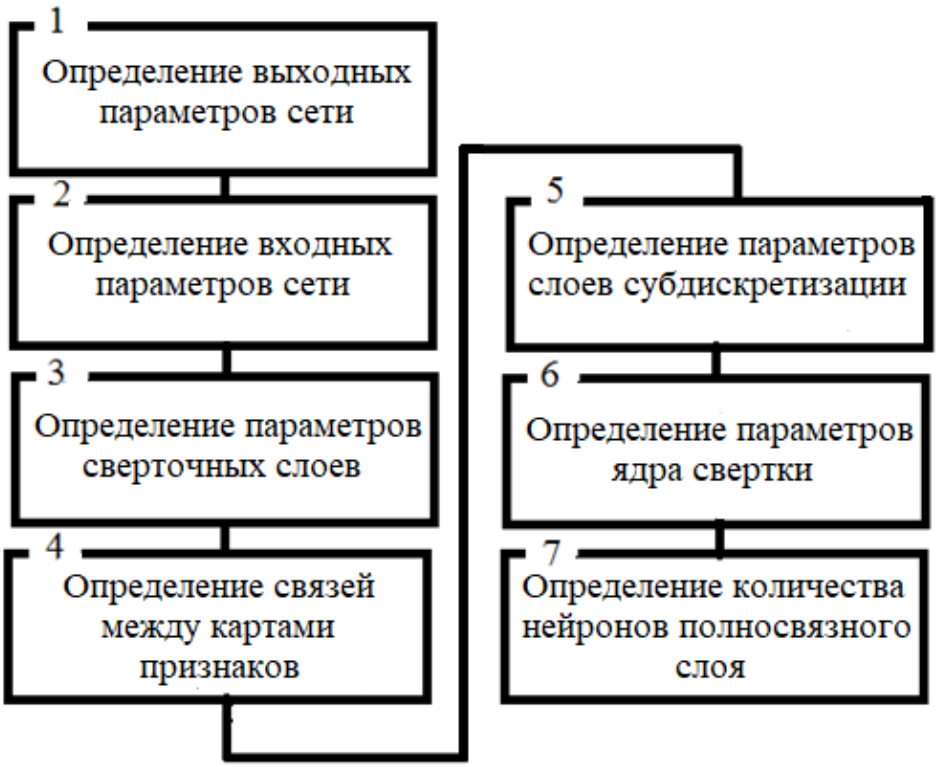


Рис. 2.7 – Укрупненная блок-схема алгоритма адаптации значений параметров CNN-LSTM

Блок-схема предполагает реализацию следующих семи этапов:

Этап 1. Базируясь на методологии построения нейросетевых систем защиты информации [34], нами определено, что отправным пунктом разработки нейросетевой модели CNN-LSTM является определение количества входных и выходных нейронов. Нами принято, что количество выходных нейронов сети должно быть равным количеству распознаваемых пользователей.

Этап 2. Количество входных нейронов, в соответствии с пятым принципом принято равным количеству пикселей изображения соответствующего одному символу рукописного текста. Учтено, что анализируемое изображение символа имеет черно-белый формат, а перед подачей в нейросеть это изображение скелетизируется.

В соответствии с [35, 40] при использовании распространенного программно-аппаратного обеспечения биометрической аутентификации размер такого изображения составляет 33x33 пикселей. Обозначив количество легитимных пользователей символом K_l, получим: размер входного поля $a_0 = 33$, количество входных нейронов $L_{in} = 33 \times 33 = 1089$, а количество выходных нейронов:

$$L_{out} = K_l + 1. \tag{2.21}$$

При этом учтено, что количество распознаваемых пользователей должно быть на единицу больше количества легитимных пользователей.

Этап 3. Использовав первый принцип адаптации путем экспертного оценивания нами определена достаточность трехуровневого распознавания. Таким образом, количество слоев свертки $K_{ls} = 2$. Отметим, что небольшое количество скрытых слоев нейрсетевой модели позволяет использовать для скрытых нейронов функцию активации типа гиперболического тангенса, что в свою очередь предопределяет возможность отказа от использования механизма Droput.

Этап 4. Так же в соответствии со вторым принципом адаптации

базируясь на результатах [35, 37-40] нами принято использовать:

- 6 карт признаков на первом уровне распознавания;
- 24 карт признаков на втором уровне распознавания.

Предполагается, что на первом уровне распознавания НС должна выделить элементарные признаки представляющие собой четыре различным образом ориентированных отрезка, а также пустую и заполненную области. Поскольку сверточная НС устойчива только к небольшим аффинным преобразованиям, то различным образом ориентированные и масштабированные фигуры одинаковой формы можно рассматривать как разные признаки. Для всех шести элементарных фигур (признаков на первом уровне распознавания) количество возможных преобразований принято равным 4. Поэтому количество значащих признаков на следующем уровне распознавания равно 24, что соответствует количеству карт признаков во втором сверточном слое. В результате $L_{h,1} = 6, L_{h,2} = 24$.

Этап 5. Исходя из позиций минимизации вычислительных ресурсов и упрощения структуры сети нами принято отказаться от использования слоев подвыбоки (субдискретизации). Следовательно $K_{ld} = 0$.

Этап 6. Отталкиваясь от первого и четвертого принципов нами определено, что размер ядра свертки для первого и второго сверточного слоя равен $(b \times b)_1 = (b \times b)_2 = (5 \times 5)$. Смещение рецептивного поля для первого и второго сверточных слоев принято равным $d_1 = d_2 = 2$. Также принято $r_1 = r_2 = 0$. Подставив эти данные в выражение (2.17) определен размер карт признаков для первого и второго сверточных слоев $a_1 = 15$, $a_2 = 6$.

Этап 7. Расчет минимального количества нейронов полносвязного слоя может быть произведен исходя из позиций минимальной достаточности определенной теоремой Хехт-Нильсена:

$$L_{f,\min} = 2L_{h,2} + L_{out}. \tag{2.22}$$

Подставив в выражение (2.20) известные значения получим:

$$L_{f,\min} = 2L_{h,2} + L_{out} = 2 \times 24 + K_l + 1 = 49 + K_l . \qquad (2.23)$$

Таким образом, рассчитанные нами значения параметров нейросетевой модели CNN-LSTM равны: $a_0 = 33$, $L_{in} = 1089$, $K_{ls} = 2$, $L_{h,1} = 6$, $L_{h,2} = 24$, $(b \times b)_1 = (5 \times 5)$, $(b \times b)_2 = (5 \times 5)$, $d_1 = d_2 = 2$, $r_1 = r_2 = 0$, $a_1 = 15$, $a_2 = 6$. Отметим, что функционирование модели CNN-LSTM определено выражениями (2.6-2.23). Для расчета значений количества выходных параметров и количества нейронов в полносвязном слоя следует воспользоваться выражениями (2.21, 2.23), указав в них количество легитимных пользователей, которых должна распознать конкретная система биометрической аутентификации.

Структура адаптированной нейросетевой модели CNN-LSTM представлена на рис. 2.8.

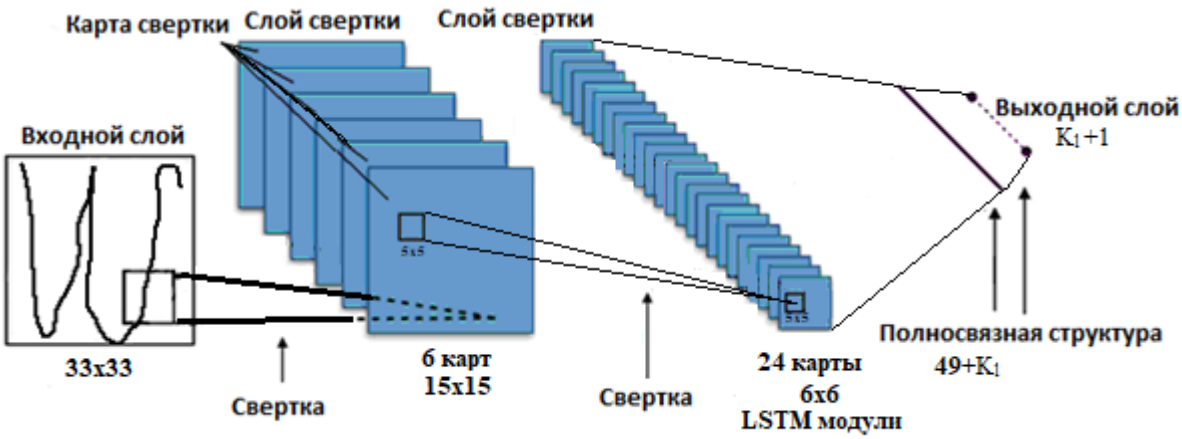


Рис. 2.8. Структура адаптированной нейросетевой модели CNN-LSTM

2.4. Обучение разработанной нейронной сети

Разработанная НС должна быть обучена преобразовывать тайный биометрический образ «Свой» в личный ключ пользователя. Обучение должно

осуществляться автоматически, то есть в процесс подбора параметров ИНС не должен вмешиваться человек. А пользователь должен быть уверен, что его ключ, участвующий в обучении, не будет скомпрометирован [26].

В процессе обучения при предъявлении на вход ИНС элементов вектора «Свой» сеть должна выдавать личный криптографический ключ, а при предъявлении элементов вектора «Чужой», случайные состояния – «белый шум». Это достигается путем правильного подбора автоматом обучения весовых коэффициентов ИНС. Для обучения образы «Свой» и «Чужие» поочередно подаются на вход ИНС, а автомат обучения в промежутке подбирает весовые коэффициенты [17].

Рассмотрим процесс обучения на примере вырожденного нейросетевого преобразователя биометрия-код с одним выходом [41], показанного на рис. 2.9.

Понятно, что обучение одного нейрона особой трудности не представляет и для этой цели может быть использован любой из известных на сегодня алгоритмов обучения [42-49].

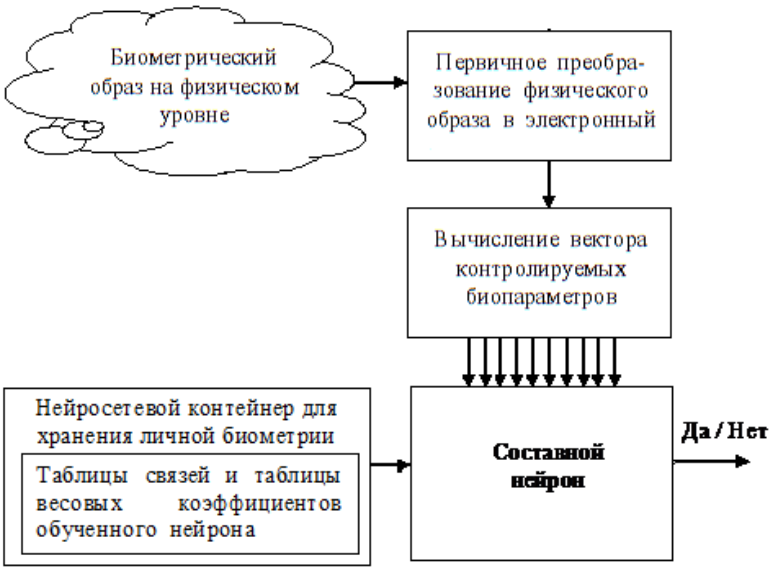


Рис. 2.9. Вырожденный нейросетевой преобразователь биометрия-код с единственным составным нейроном

Для иллюстрации процесса обучения мы будем использовать среду моделирования «НейроПреподаватель» [50]. В качестве биометрического образа нами выбрана динамика рукописного почерка, обучающая выборка будет формироваться из нескольких примеров «Свой» и нескольких примеров «Чужой» (рис. 2.10) [51].

В качестве примеров образа «Свой» будем использовать тринадцать рукописных образов слова «Алматы», воспроизведенных одним и тем же человеком.

В качестве образов «Чужие» нами были использованы слова «Мәди», «Айнұр», «Тараз» и другие (рис. 2.11) [51], воспроизведенные рукописно разными людьми. То есть, при обучении нейрона нужно подобрать его весовые коэффициенты таким образом, чтобы множество образов «Чужие» и множество примеров одного образа «Свой» были, как можно более сильно, разнесены на выходе сумматора нейрона. Как правило, в качестве нейрона рассматривают сумматор с некоторым нелинейным элементом на выходе [51].

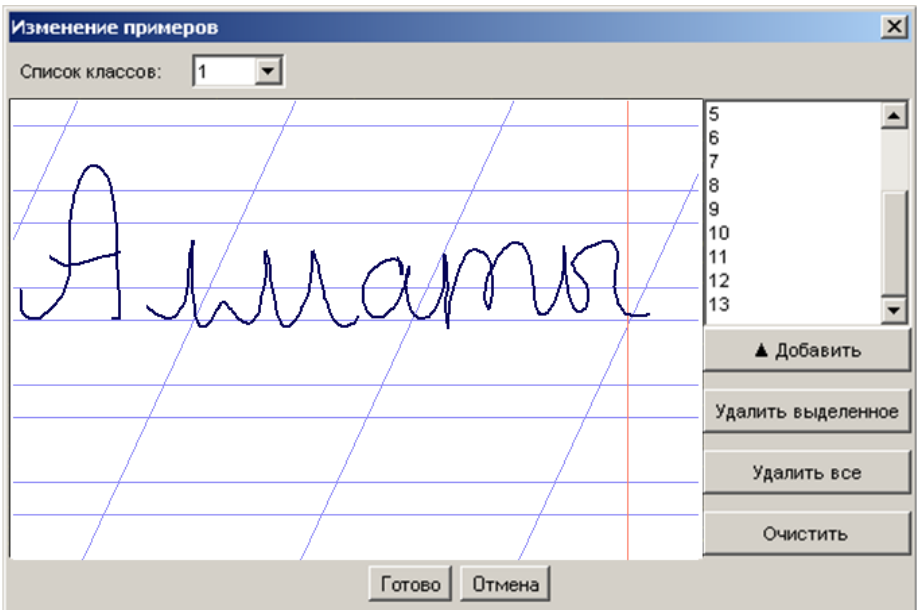


Рис.2.10. Экранная форма ввода примеров рукописного пароля «Свой» в среде моделирования «НейроПреподаватель»

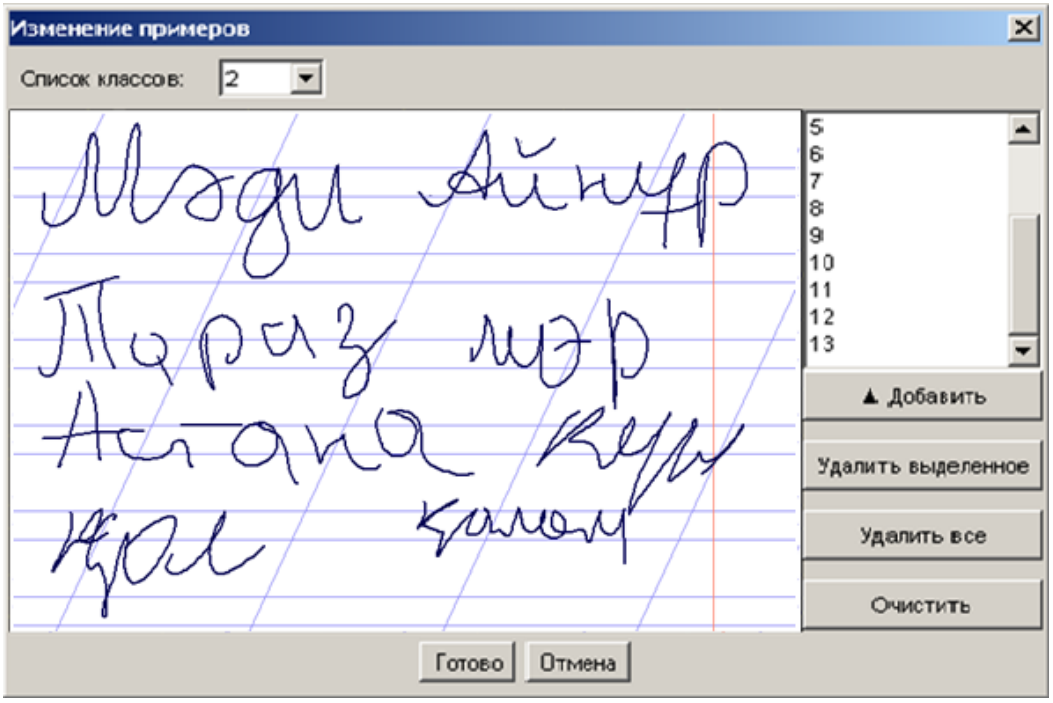


Рис. 2.11. Экранная форма ввода примеров рукописных образов «Чужие» в среде моделирования «НейроПреподаватель»

В итоге нам необходимо при обучении нейрона итерационно подобрать его весовые коэффициенты – μ_i всех входов сумматора. Если обозначить через $z(\bar{v})$ выходные отклики сумматора на образы «Свой», то получится следующая запись:

$$z(\bar{v}) = \mu_0 + \sum_{i=1}^{96} \mu_i v_i \qquad (2.24).$$

Так как необходимо, чтобы все примеры рукописного слова «Алматы» давали отклик на выходе нейрона «1», то настройка единственного нелинейного элемента должна осуществляться по следующему правилу:

$$\begin{cases} y(\xi_i) = \text{"0"} \quad \text{если} \quad \xi_i \langle \min(z(\bar{v}_i)); \\ y(\xi_i) = \text{"1"} \quad \text{если} \quad \xi_i \geq \min(z(\bar{v}_i)) \end{cases} \qquad (2.25).$$

В этой ситуации число примеров недостаточно для устойчивого обучения. В связи с этим сумматор нейрона следует сделать составным, как это показано на рис. 2.12 [52].

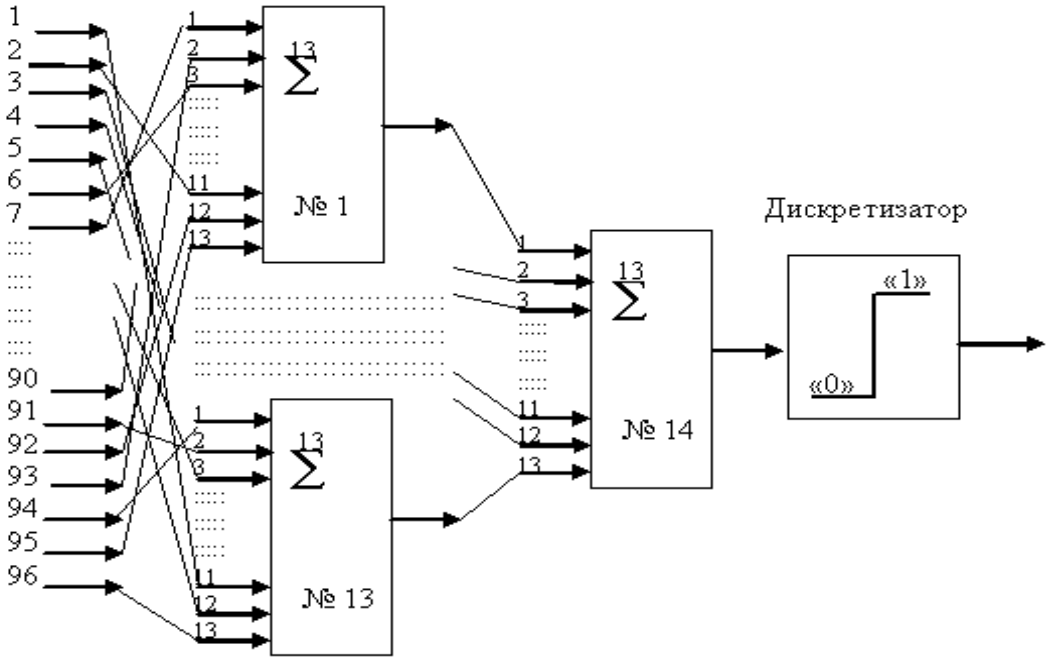


Рис. 2.12. Искусственный нейрон с составными сумматорами

Для обучения каждого из 14 сумматоров составного нейрона вполне достаточно обучающей выборки из 13 примеров, так как каждый сумматор имеет по 13 входов (число входов не превышает числа примеров). При организации составного сумматора каждый из сумматоров первого слоя необходимо подключать случайно к входам нейрона. Очевидно, что необходимо проследить факт использования всех входов [51].

Каждый из сумматоров первого слоя обучается самостоятельно на примерах образов «Свой» и «Чужие». После того как сумматоры первого слоя обучены, данные примеров с их входов транслируются на выходы сумматоров и далее используются при обучении единственного сумматора второго слоя. Нелинейные элементы на выходах сумматоров первого слоя отсутствуют, что позволяет нам рассматривать все сумматоры как один сумматор с большим числом входов. Декомпозиция полной задачи обучения на 14 более простых

подзадач необходима только для режима автоматического обучения. В режиме ручного обучения нейрона (такой режим в среде моделирования «НейроПреподаватель» имеется) можно добиться удовлетворительных результатов.

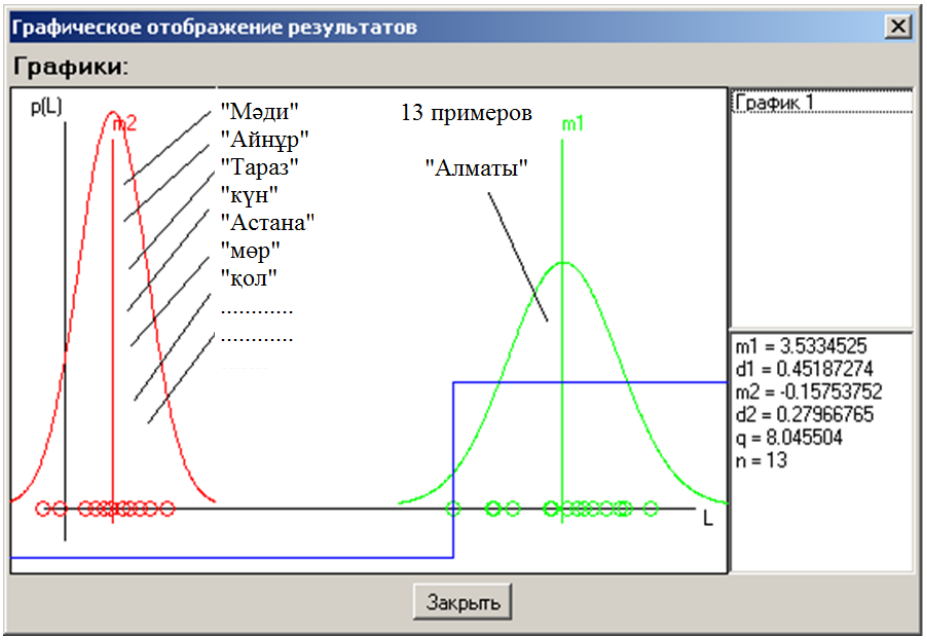


Рис. 2.13. Экранная форма ввода результатов обучения нейрона в среде моделирования «НейроПреподаватель»

Результаты автоматического обучения последнего сумматора составного нейрона отображены на рис. 2.13 [51], где m1 – это центр (математическое ожидание) рукописных образов "Алматы", а m2 – это центр (математическое ожидание) рукописных образов «Мәди», «Айнұр», «Тараз» и так далее.

Следует отметить тот положительный факт, что информацией о параметрах обученного нейрона намного труднее воспользоваться, чем информацией в биометрическом шаблоне. Условия формирования биометрического шаблона прямо указывают на положение контролируемых биометрических параметров.

Совершенно иначе обстоит дело с весовыми коэффициентами и связями обученного составного нейрона. По ним нельзя точно указать, где следует ожидать появления биометрических параметров «Свой». Пример таблиц связей

составного нейрона и весовых коэффициентов обученного составного нейрона приведен на рис. 2.14.

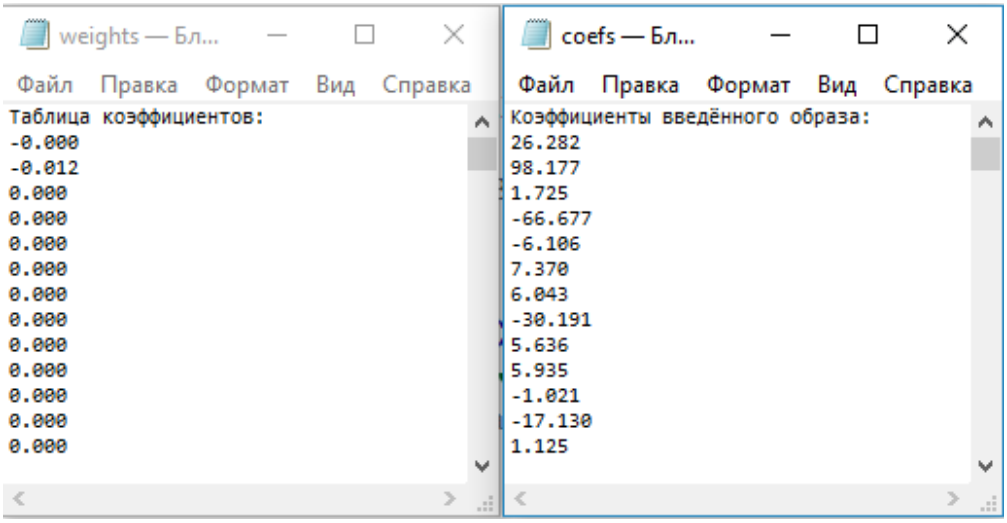


Рис. 2.14. Пример таблиц связей составного нейрона и весовых коэффициентов обученного составного нейрона

Из рисунка видно, что информация, размещаемая в так называемый биометрический контейнер, очень похожа на шифротекст. Ее нельзя интерпретировать так же просто как информацию биометрического шаблона. В этом отношении использование ИНС следует рассматривать как один из методов сокрытия персональной биометрической информации пользователя [51].

2.5. Нейросетевая модель распознавания отпечатков пальцев

При работе с отпечатками пальцев нам необходимо получить биометрические параметры, которые будут инвариантны к размерности вектора контрольных точек и смещению пальца. Для этого сначала откорректируем координаты контрольных точек, затем используем двумерное дискретное ортогональное преобразование [17, 31]. Принцип работы вышеописанного преобразования показан на рисунке 2.15.

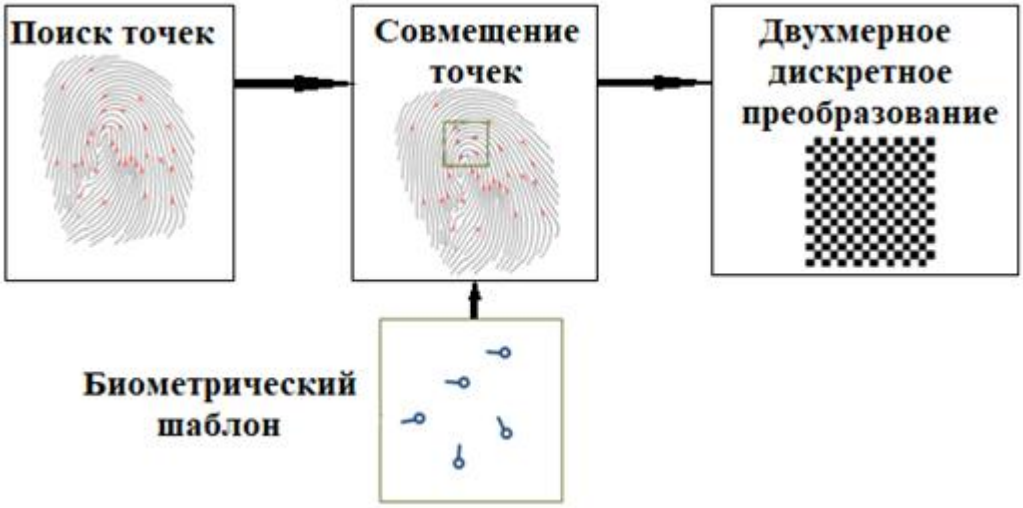


Рис. 2.15. Алгоритм преобразования биометрических параметров отпечатков пальцев

На первом этапе, на полученном со сканера отпечатков пальцев изображении папиллярного рисунка найдем контрольные точки. Данные о расположении контрольных точек и их направлении занесем в вектора.

На втором этапе полученные вектора преобразуем согласно шаблону. Шаблон будет формироваться во время обучения ИНС сети по нескольким изображениям папиллярных рисунков отпечатков пальцев. Из вектора с наиболее вероятными к извлечению контрольными точками сформируем шаблон. По входному вектору и сформированному шаблону алгоритм совмещения определит значения $\Delta X, \Delta Y, \Delta R$. Здесь ΔX - означает смещение двухмерной поверхности по оси OX, ΔY - смещение по двухмерной поверхности оси OY, а ΔR - поворот относительно начала координат. В соответствии со значениями $\Delta X, \Delta Y, \Delta R$ преобразуем входной вектор (осуществляется поворот эвклидовой плоскости). Затем к полученному вектору применим двумерное дискретное преобразование Уолша-Адамара [17, 31]:

$$K(u,v) = \sum_{x=0}^{N-1}\sum_{y=0}^{N-1} P_{xy}\, g(x,y,u,v) \qquad (2.26)$$

где $K(u,v)$ - это коэффициенты результатов преобразования; P_{xy} - входная матрица размерности N^2; $g(x,y,u,v)$ - ядра двумерного прямого преобразования Уолша-Адамара размерности N^2. Они были получены рекурсивным методом начиная с $H_1 = 1$ с помощью построения блочных матриц по формуле [17, 31]:

$$H_{2^n} = \begin{bmatrix} H_{2n-1} & H_{2n-1} \\ H_{2n-1} & -H_{2n-1} \end{bmatrix}, \qquad (2.27).$$

На этапе реализации преобразователя «биометрия-код» важным вопросом является выбор структуры НС. Анализ литературы по ИНС [26, 53, 54] показал, сети делят на одно-, двух-, трехслойные и с большим числом нейронов. Для биометрии ГОСТ Р 52633.5–2011 [55] рекомендует однослойные или двухслойные НС. Для двухслойных НС функции каждого слоя разделены. К функциям первого слоя относятся обогащение биометрических данных и квантование обогащенных данных. Нейроны второго слоя при недостаточном обогащении исправляют ошибки биометрического кода нейронов первого слоя [16, 26, 56].

Второй слой нейронов можно заменить классическим кодом, обнаруживающим и исправляющим ошибки, но нейросетевое корректирование ошибок выгоднее по причине того, что при обучении второго слоя на примерах биометрических кодов «Свой» для каждого из его разрядов оценивается реальный показатель стабильности [26].

Практические исследования показали, что большинство разрядов биометрического кода имеет высокую стабильность. Только некоторые разряды кода оказываются нестабильными и их положение известно [26, 56].

При обучении второй слой нейронов корректирует нестабильные разряды и одновременно хэширует (перемешивает) все разряды биометрического кода. Классические самокорректирующиеся коды, строящиеся в рамках предположения о том, что ошибки между разрядами кода распределены

равновероятно, проигрывают нейросетевым корректорам ошибок, так как не учитывают реальное распределение показателей стабильности биометрических кодов «Свой» [16, 26].

После выбора количества слоев сети нужно выбрать количество входов каждого нейрона и задать связи входов с номерами входов сети. Например, если у НС 480 входов со средней информативностью примерно 0.3 бита, то потребуются нейроны с количеством входов от 1 до 18. Качество биометрических параметров и их корреляционных связей влияет на число входов нейрона. Нужное количество входов определяется непосредственно во время обучения нейрона. Поэтому сначала случайно задаем малое количество входов, потом если качество решения не соответствует заданному, увеличиваем количество входов нейрона. В итоге получаем однослойную сеть, где у каждого нейрона будет свое количество входов, случайно подключенных к входам всей сети. После обучения для каждого нейрона дополнительно получаем таблицу весовых коэффициентов входных связей [26].

Таблица весовых коэффициентов и таблица связей нейронов формально описывают обученную сеть. В двухслойных сетях эти таблицы создаются для каждого слоя нейронов. Слои нейронов обучаются последовательно. После обучения первого слоя нейронов примеры образов «Свой» и «Все Чужие» транслируются с входа НС на выходы нейронов. Таким образом, получают примеры биометрических кодов, на которых обучаются нейроны второго слоя [16, 26].

В биометрических системах процесс обучения нейронов должен проходить без участия человека, поэтому нужно иметь абсолютно устойчивый автомат обучения. ГОСТ Р 52633.5–2011 [55] рекомендует использовать не итерационные алгоритмы обучения. Эти алгоритмы строятся на подборе значений весовых коэффициентов и их вероятных знаков для подбора ожидаемого решения. Это несколько снижает качество обучения, но ведет к появлению эффекта высокой устойчивости вычислений. Чем больше входов у обучаемого нейрона, тем выше устойчивость вычислений будет тем выше.

Для обучения НС для распознавания отпечатков пальцев выбран абсолютно устойчивый не итерационный алгоритм обучения.

2.6. Проверка качества обучения преобразователей биометрия-код

После обучения системы нам необходимо оценить качество обучения, то есть провести тестирование преобразователей биометрия-код. Для тестирования применяют N_1 векторов образов «Свой» и N_2 векторов образов «Чужой». На рис. 2.16. приведена схема тестирования системы.

Любая биометрическая защита должна быть способна хорошо распознавать образ «Свой» и надежно выделять множество образов «Чужие» («Все Чужие»). Очевидно, что средство биометрической защиты может ошибаться. Основной и первой задачей для биометрии является обеспечение доступа донору биометрического образа «Свой». Ошибка при выполнении этой задачи называется ошибкой первого рода (ОПР). Вероятность появления ОПР P_1 является основной характеристикой эффективности работы системы.

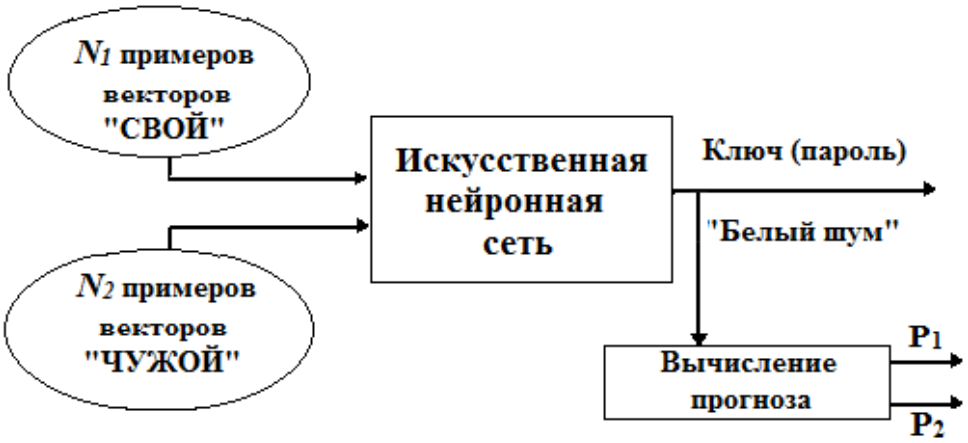


Рис. 2.16. Схема тестирования системы

Для оценки высокой вероятности ОПР (например, $P_1 \approx 0.1$) необходимо $N_1=20$ тестовых примеров. Если вероятности ОПР большие, то проблемы

тестирования их значений не существует. В настоящее время эта ситуация характерна для большинства биометрических приложений. Большинство современных приложений биометрической аутентификации проблему высокой доступности решают за счет увеличения числа разрешенных попыток предъявления образа «Свой» во время одного сеанса. Например, если по политикам информационной безопасности пользователю разрешено использовать три попытки предъявления биометрического образа «Свой», то вероятности получить доступ с 1, 2, 3 попыток составят $P_{1,1}$=0.1, $P_{1,2}$=0.01, $P_{1,3}$=0.001. Предоставление пользователю трех попыток в место одной поднимает вероятность доступа с 0.9 до 0.999.

Второй задачей средства биометрической аутентификации является препятствовать доступу донору образа «Чужой». Второй важнейшей характеристикой биометрических средств является вероятность появления ошибок второго рода (ОВР) P_2 из-за возможных коллизий образов «Свой» и «Чужой» на рассматриваемом множестве признаков (биометрических параметров) [16, 57].

Если придерживаться гипотезы нормальности закона распределения значений показателей критерия Хэмминга, то можно вычислить математическое ожидание распределения данных p(h(x)) и его среднеквадратическое отклонение. Приближенную оценку вероятности ОВР можно вычислить по следующей формуле:

$$P_2 \approx \frac{1}{\sigma(h(x)) \cdot \sqrt{2\pi}} \int_0^{\max(h(c))} \exp\left\{ \frac{-(E(h(x)) - u)^2}{2 \cdot \sigma^2(h(x))} \right\} \cdot du \ , \qquad (2.28)$$

где σ(h(x)) – среднеквадратическое отклонение расстояний кодов «Чужие»; max(h(c)) – максимально возможное значение расстояние Хэмминга кодов «Свой»; E(h(x)) – математическое ожидание расстояний Хэмминга кодов «Чужие».

Чем больше биометрических параметров принимает в расчет средство биометрической аутентификации, тем меньше будет вероятность ОВР P_2. Если система может анализировать сотни или тысячи биометрических параметров, то ее можно считать высоконадежной. На сегодняшний день лучшие средства высоконадежной биометрической аутентификации обеспечивают вероятность ОВР на уровне одной миллиардной и меньше, то есть злоумышленник, пытающийся преодолеть биометрическую защиту, должен предъявить миллиард разных биометрических образов (например, воспроизвести своей рукой миллиард рукописных паролей) [16]. Если считать, что на воспроизведение одного рукописного пароля уходит 10 секунд, то злоумышленнику потребуется 10 млрд. секунд, а это составит 321 год непрерывных усилий. Это на много больше времени жизни одного человека.

На практике часто пользуются понятием стойкости преобразователя биометрия-код, которая обратно пропорциональна его вероятности ОВР.

2.7. Проблемы размерности задач распознавания образов и пути их решения

Входных биометрических данных N у преобразователя биометрия-код всегда больше, чем число выходов n (чем длина ключа). Из-за низкой информативности биометрических данных условие N>n всегда выполняется. Одного биометрического параметра не достаточно, чтобы получить один бит криптографического ключа. Обязательно необходимо использовать избыточное число биометрических параметров. Они используются для «обогащения» входных биометрических данных НС преобразователя.

2.7.1. Применение малого количества примеров для представления многомерных непрерывных биометрических образов

Сложность распознавания биометрических образов вызвана многими причинами, одной которых является высокая размерность задачи. То есть, приходится учитывать множество «плохих» биометрических параметров, поэтому невозможно воспользоваться классической линейной алгеброй и многомерной статистикой. Сложность положения заключается в том, что обучение преобразователей биометрия-код приходится делать на малом количестве примеров в обучающей выборке [16, 59].

Пользователи для обучения готовы предъявить до 20 примеров биометрического образа «Свой». Если попросить предъявить 100 или 200 примеров, то эта воспринимается ими негативно. Пользователи не хотят прилагать большие усилия для обучения преобразователя биометрия-код. Это означает, что 512-мерные распределения непрерывных параметров образа «Свой» вынуждено представляются всего 20 примерами по каждому из параметров.

Используя 20 примеров, невозможно точно вычислить математическое ожидание биометрических параметров $E(v_i)$, их дисперсию $\sigma(v_i)$ и коэффициенты корреляции между параметрами $r_{i,j}$. При этом относительная ошибка в расчете математического ожидания может доходить до ±25 %, относительная ошибка дисперсии может доходить до ±50 %, относительная ошибка расчета коэффициентов корреляции может доходить до ±100 %. При таких данных для распределения образа «Свой» построить многомерную аналитическую модель невозможно [16, 54].

Работа в пределах гипотезы нормального нормированного распределения значений биометрических параметров предполагает построение для них 512-мерной корреляционной матрицы.

Эта ситуация получила название «проклятия размерности». Учет дополнительных биометрических параметров не приводит к более точным результатам. Попытки увеличивать размерность решаемой задачи приводит к ухудшению ее решения. Происходит накопление погрешностей вычислений. «Проклятие размерности» делает невозможным использование линейной алгебры и классической многомерной статистики в биометрии [16, 54].

Ослабить проблему «проклятия размерности» или вообще снять позволяет применение ИНС. Для обучения ИНС с 512 входами и 256 выходами в работе [55] предложен алгоритм автоматического с использованием всего 20 примеров образа «Свой». Когда непрерывная функция (континуум) представляется малым числом примеров, возникают большие погрешности дискретизации. Они способны накапливаться при обучении НС. Интегрирование 512 отсчетов измерений во времени повышает точность вычислений. Чтобы подавить естественные ошибки вычислений данные можно усреднять по 512 разным пространствам, а не интегрировать по времени. Как следствие чем выше размерность решаемой задачи, тем лучше работает алгоритм. Возникает эффект, обратный проблеме «проклятия размерности» [16, 59].

2.7.2. Применение энтропии для снижения размерности задачи распознавания образов

Сложность (размерность) решаемой задачи определяет возможности интеллекта (искусственного и естественного). С точки зрения математики, любую задачу можно представить как многомерную функцию $F(x_1, x_2, x_3, ..., x_n)$, где n – число переменных [16, 59].

Размерность решаемой задачи распознавания образов связана со значением выходной энтропии кодовых состояний преобразователя биометрия-код, полученная воздействием на него образами «Все Чужие».

Покажем на примере шифрования текстов важность этого показателя. Если текст не был зашифрован, то любой носитель того или иного языка его прочтет.

У любого естественного языка между его знаками имеются корреляционные связи. Признаком всех биометрических кодов являются большие коэффициенты парной корреляции между разрядами кодов. Коэффициент парной корреляции (коэффициент Пирсона) показывает тесноту линейной корреляционной связи между зависимой и независимой случайной величиной. Он представляется некоторым числом от -1 до 1. Шифрование разрушает естественные корреляционные связи. После шифрования разряды кодов становятся независимыми, парных корреляций не будет ($r_{i,j}$= 0.0 для всех $i \neq j$) [16, 59]. То есть код будет «белым шумом».

«Белый шум» имеет следующую характеристику: энтропия каждого его разряда будет равна одному биту, то есть он будет случайным и его длина будет равна его энтропии. У кода длиной 256 бит энтропия H будет равна 256 битам. Энтропия будет падать, если код будет не случайным [16, 59]:

$$H(256) \langle\ 256 \text{ при } \left|r_{i,j}\right| \rangle\ 0.0 \text{ хотя бы для одной пары } i \neq j\,. \qquad (2.29)$$

Энтропия связана с вероятностью ОВР. Если рассмотреть симметричное шифрование с 256 битным ключом, то при использовании случайного ключа вероятность расшифровывания текста с первой попытки составит

$$P_2 = 2^{256} \text{ или } H(256) = 256 = -\log_2(P_2). \qquad (2.30)$$

В общем виде:

$$H(n) = -\log_2(P_2)\,, \qquad (2.31)$$

где n – длина биометрического кода; P_2 – вероятность ОВР преобразователя биометрия-код.

Согласно ГОСТ Р 52633.0–2006 [60] среднее значение модуля корреляционных связей в разрядах биометрических кодов «Чужой» не должно быть больше 0.15. Если $E(|r_{i,j}|) = 0.15$, вероятность ОВР будет порядка $P_2=2^{25.6}$, что меньше на порядок. При увеличении модулей коэффициентов парной корреляции стойкость преобразователя биометрия-код к атакам подбора будет падать [59].

Важным моментом при этом является вопрос вычисления энтропии выходных биометрических кодов. Понятно, что классический метод вычисления многомерной энтропии с использованием формулы Шеннона

$$H(256) = -\sum_{i=1}^{2^{256}} P_i \cdot \log_2(P_i), \qquad (2.32)$$

где P_i – вероятность появления i-го состояния биометрического кода, требует больших вычислительных затрат и размеров исходных биометрических данных.

Так как для преобразователей биометрия-код с 256 выходами число состояний будет равно 2^{256}, то вычисление энтропии по формуле (2.32) технически невозможно, поэтому нужно выбирать или создавать более экономичные варианты вычислений. Одним из таких вариантов является переход из поля обычных кодов в поле кодов расстояний Хэмминга [59, 61].

2.7.3. Применение меры Хэмминга

Преобразователь биометрия-код – это отображение многомерных континуумов в код длиной 256 бит. На вход преобразователя подаются континуальные (непрерывные) биометрические данные, а на выходе получают дискретные коды. Число возможных состояний кода всегда конечно. Поэтому для исследования выходных кодов преобразователя можно использовать расстояние (меру) Хэмминга. Для вычисления меры Хэмминга двух кодов

одинаковой длины их сравнивают поразрядно, а затем подсчитывают число не совпавших разрядов. Если выходной код представить в виде двоичного вектора $\bar{x}$, то мера Хэмминга до кода «Свой» $\bar{c}$ вычисляется по формуле [59]:

$$h = \sum_{i=1}^{256} x_i \oplus c_i \,, \tag{2.33}$$

где 256 – длина сравниваемых кодов; i – номер сравниваемых разрядов; $\oplus$ – операция сложения по модулю два.

На рис. 2.17 [16, 53, 54] показано вычисленное распределение расстояний Хэмминга между выходными кодами «Чужие» и выходным кодом «Свой».

Для кодов «Свой» мера Хэмминга сжимается (прижимается к оси системы координат рис. 2.17), так как применение нейросетевых преобразователей дает возможность улучшать их качество и нестабильность на выходе преобразователя доходит до 7 бит.

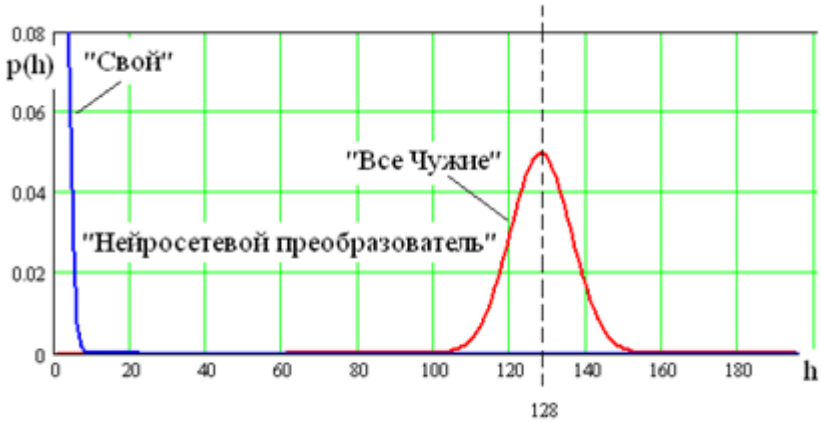


Рис. 2.17. Распределение расстояний Хэмминга для кодов длиной 256 бит

Распределения Хэмминга кодов «Все Чужие» очень хорошо описывается нормальным законом распределения значений. Далее, вычисляя энтропию выходных кодов, рассматриваем только разницу между длинами кода [41, 50, 52, 62, 63].

Если длина выходного кода мала и разряды слабо коррелированны, то размерность решаемой биометрической задачи при вышеописанном методе оценки может быть очень высокой. Размерность решенной задачи биометрической защиты будет падать с ростом корреляционных связей между разрядами кода. Если разряды кода будут полностью коррелированы при любой длине, то его энтропия будет 1 бит и размерность задачи также составит 1 бит.

Если энтропия выходных кодов преобразователя не растет, то на рост размерности решаемой задачи не влияют увеличение количества связей, слоев нейронов, входов преобразователя и алгоритм обучения [52, 63-65].

2.8. Особенности преобразования биометрия-код при использовании малых тестовых баз

2.8.1. Оценка вероятности возникновения ошибок первого рода

Проведенные нами исследования показали, что биномиальный закон распределения значений биометрических кодов для образов «Свой» работать не будет, так как разряды выходного кода сильно коррелированны (практически равны единице), а энтропия нулю [16, 64, 66-70], то есть:

$$\begin{cases} E(|r_{i,j}|) \approx 1.0; \\ H_{256}(c) \approx 0.0. \end{cases} \tag{2.34}$$

В связи с этим нам нужно знать при выполнении (2.34), к какому предельному распределению стремится биномиальный закон [16].

Результаты наших исследований показывают, что предельным распределением биномиального закона будет распределение хи-квадрат с почти нулевыми величинами степеней свободы [16, 17]:

$$P(256, h, P_1 \to 0.0, |r| \to 1.0) \to \chi^2(\omega \approx 0), \qquad (2.35)$$

где ω – величина степеней свободы.

Величина степеней свободы ω описывает меру Хэмминга при первом отказе «Своему» в k-опытах:

$$\omega = \frac{h}{k}.$$

Если при проведении испытаний из k-опытов было несколько ошибок с ненулевыми величинами меры Хэмминга, то ω вычисляется как среднее меры Хэмминга [16, 17, 31]

$$\omega = E(h) \approx \frac{1}{k}\sum_{i=1}^{k} h_i.$$

Если в проведенных испытаниях отказов в доступе не было, то чтобы рассчитать ω будем считать, что в следующем (k + 1) испытании будет минимальная ошибка в один бит:

$$\omega = \frac{1}{k+1}.$$

Чтобы от 3 до 10 раз уменьшить объем тестовой выборки и продолжительность тестирования, для расчета вероятности ОПР используем формулу [16, 17, 31]:

$$P_1 \approx \int_1^{\infty} \frac{1}{2^{\frac{\omega}{2}} \cdot \Gamma\left(\frac{\omega}{2}\right)} \cdot x^{\frac{\omega}{2}-1} \cdot e^{-\frac{x}{2}} \cdot dx, \qquad (2.36)$$

где $\Gamma(\,.\,)$ – гамма функция.

Среднюю корреляцию между разрядами кодов «Свой» будем вычислять по формуле:

$$r \approx (1 - \frac{2 \cdot E(h)}{n}). \tag{2.37}$$

Описанное приближение нужно применять только когда «Своему» нужно представлять доступ при первой попытке с высокой вероятностью. Только в этом случае возникает проблема тестирования вероятности ОПР. Например, если задана вероятность доступа с первой попытки 0.999, то для ее правильной оценки требуется 10 000 тестовых примеров. Вот в этом случае нужно использовать приближение распределения меры Хэмминга «Свой» χ^2-распределением с почти нулевым показателем величины степеней свободы [16].

2.8.2. Оценка вероятности возникновения ошибок второго рода

При тестировании приближенная оценка вероятности ОВР P_2 тестируемого преобразователя биометрия-код оценивалась в рамках гипотезы нормальности закона распределения значений показателей критерия Хэмминга [61, 62]:

$$P_2 \approx \frac{1}{2} - \frac{1}{\sqrt{2\pi}} \cdot \int_0^{E(h)/\sigma(h)} \exp(-t^2/2)dt \tag{2.38}$$

Однако результаты экспериментов в ходе выполнения комплексного проекта по тестированию с различными размерами баз естественных биометрических образов «Чужой» показали, что реальное распределение величин показателей меры Хэмминга различается от нормального закона распределения значений. В связи с этим вычисленный по формуле (2.38)

прогноз, является приблизительным и должен быть скорректирован с учетом средней величины модуля парных корреляций, средней величины уникальности, стабильности, информативности биометрических параметров тестируемого биометрического образа [62].

В связи с этим кратко остановимся на этих показателях.

Для i-го биометрического параметра *показателем средней стабильности* v_i, является характеристика, вычисляемая по формуле:

$$c(v_i) = \frac{\sigma_{\text{Чужой}}(v_i)}{\sigma_{\text{Свой}}(v_i)}$$

где $\sigma_{\text{Чужой}}(v_i)$ – стандартное отклонение i-го биометрического параметра множества образов «Чужой» на i-том входе преобразователя биометрия-код; $\sigma_{\text{Свой}}(v_i)$ – стандартное отклонение i-го биометрического параметра множества образов «Свой» на i-том входе преобразователя биометрия-код.

Биометрические параметры образа «Свой» считаются стабильными, если динамический диапазон их изменения занимает менее 30% динамического диапазона образов все «Чужие». Биометрические параметры средней стабильности имеют динамический диапазон от 30% до 60% от возможного динамического диапазона все «Чужие». Нестабильные биометрические параметры имеют динамический диапазон, составляющий от 60% до 90% от возможного динамического диапазона образов все «Чужие». Также ГОСТ Р 52633.1-2009 [71] определяет показатель стабильности разряда исходного ключа – показатель, изменяющийся в пределах от 0.0 (разряд абсолютно нестабилен) до 1.0 (разряд полностью стабилен), вычисляемый по следующей формуле:

$$\omega_i = 2 \cdot \left|0{,}5 - P_{0,i}\right| = 2 \cdot \left|0{,}5 - P_{1,i}\right|$$

где $P_{0,i}$ – вероятность появления состояния «0» в контролируемом i-том разряде; $P_{1,i}$ – вероятность появления состояния «1» в контролируемом i-том разряде.

Для образов «Свой» разряды выходного кода обычно стабильны, то есть для них выполняется условие $0{,}5 \le \omega_i \le 1{,}0$. Для образов «Чужой» большинство разрядов выходного кода нестабильны, то есть для них выполняется условие $0 \le \omega_i \le 0{,}5$.

Кроме того, ГОСТ Р 52633.1-2009 [71] дополнительно определяет *показатель уникальности биометрических параметров:*

$$u_i = \frac{|E(\xi_i) - E(\nu_i)|}{\sigma(\xi_i)},$$

где $E(\xi_i)$ - математическое ожидание i-го биометрического параметра образов все «Чужие» на i-том входе преобразователя биометрия-код; $E(\nu_i)$ - математическое ожидание i-го биометрического параметра образа «Свой» на i-том входе преобразователя биометрия-код; $\sigma(\xi_i)$ – дисперсия параметра ξ образов «Все Чужие».

Показатель уникальности u_i позволяет численно оценить, насколько центр распределения $p(\nu_i)$ того или иного биометрического параметра отличается от центра так называемого среднестатистического распределения $p(\xi_i)$. Показатель уникальности для биометрических образов обычно меняется в интервале от 0 до 4. Как правило, уникальных параметров крайне мало. Например, для среднестатистического образа «Свой» менее 25% параметров имеют показатель уникальности более 1.

Для биометрических параметров образа «Свой» чем выше величины средней стабильности $E(s_i)$ и средней уникальности $E(u_i)$, тем легче решить

задачу преобразования биометрического образа «Свой» в однозначный выходной код.

Следующим важным параметром, влияющим на качество преобразования «биометрия-код», является *информативность*. Это именно тот качественный показатель, который ранее оценивался лингвистическими переменными «хороший» и «плохой». Для формализации этого понятия упростим ситуацию и будем рассматривать простейшее правило принятия решения «Свой» – «1» и «Чужой» – «0». При этом пороги квантования «Свой» разнесем на расстояние $\pm 3\ \sigma(v)$ от математического ожидания $E(v)$ распределения значений «Свой». В этом случае вероятностью ОПР можно пренебречь, а вот вероятность ОВР всегда будет достаточно велика [16]:

$$P_2(v) \approx \frac{1}{\sigma(\xi)\sqrt{2\pi}} \int_{\min(v)}^{\max(v)} \exp\left\{\frac{-(E(\xi)-u)^2}{2\sigma^2(\xi)}\right\} du\,,$$

где $E(\xi)$ – математическое ожидание биометрического параметра ξ образов «Все Чужие»; $\sigma(\xi)$ – среднеквадратическое отклонение параметра ξ образов «Все Чужие»; $\min(v)$ – минимальное значение параметра – v образа «Свой»; $\max(v)$ – максимальное значение параметра – v образа «Свой».

При нулевой вероятности ОПР, чем меньше вероятность ОВР, тем выше информативность биометрического параметра v:

$$I(v) = -\log_2(P_2(v))\,.$$

Информативность биометрического параметра будет расти по мере стабилизации биометрического параметра, те есть по мере сжатия распределения «Свой». Кроме того, на информативность влияет положение распределения данных образа «Свой» по отношению к распределению образов «Все Чужие». Эта ситуация иллюстрируется рис. 2.18 [16].

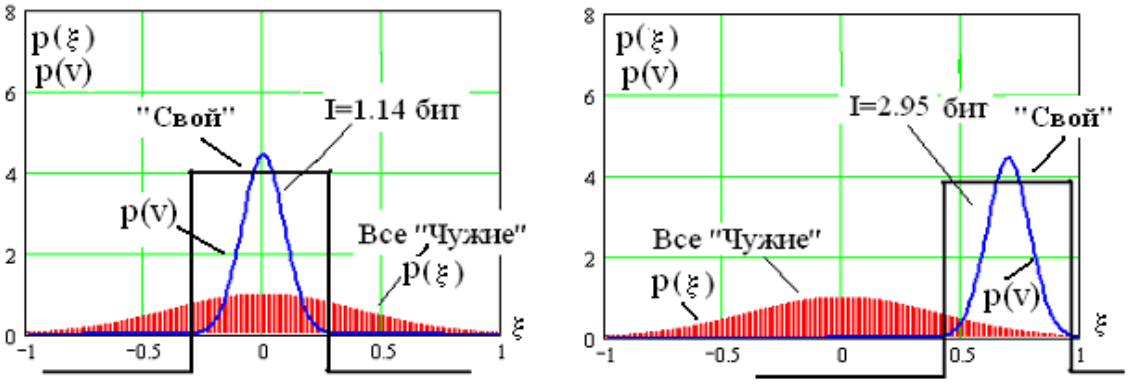


Рис. 2.18. Влияние положения распределения образа «Свой» p(v) на информативность параметра v

Из рисунка видно, что смещение одного и того же распределения данных «Свой» на периферию распределения данных образа «Все Чужие» приводит к значительному росту информативности параметра. Биометрический параметр можно считать «хорошим», если его информативность больше одного бита. И соответственно «удовлетворительными» будут биометрические параметры с информативностью менее одного бита.

Для ИНС «плохими» являются биометрические параметры, информативность которых менее 0.05 бита [16]. Эффективность нейросетевой обработки биометрических данных обусловлена тем, что она позволяет использовать низко информативные данные.

Кроме того, следует отметить, что широко применявшаяся в прошлом веке практика выделения и использования наиболее информативных параметров для высоконадежной биометрии неприемлема по следующим причинам. Во-первых, использование только наиболее информативных параметров существенно компрометирует биометрический образ «Свой», так как у каждого уникального биометрического образа наиболее информативные биометрические параметры имеют свое уникальное расположение. Во-вторых, использование только наиболее информативных параметров значительно снижает качество конечного решения.

Практика показала, что использование только 16 наиболее информативных параметров дает вероятность ОВР (вероятность ошибочного пропуска «Чужого») на уровне 10^{-3} при аутентификации личности на динамике воспроизведения рукописного слова *«Алматы»*. Если же мы отказываемся от практики применения только наиболее информативных параметров и увеличиваем число учитываемых параметров до 416, то вероятность ОВР снижается до величины 10^{-12}. Происходит снижение вероятности ошибок в миллиард раз при увеличении размерности решаемой задачи с 16 анализируемых параметров до 416 биометрических параметров.

Информация, содержащаяся в 16 наиболее информативных параметрах рукописного слова *«Алматы»,* оставляет примерно 10 бит ($P_2 \approx 10^{-3} \approx 2^{-10}$). Информация, содержащаяся в 416 параметрах рукописного слова *«Алматы»,* оставляет примерно 40 бит ($P_2 \approx 10^{-12} \approx 2^{-40}$). То есть, в 400 относительно плохих в информационном отношении параметрах размещается 30 бит информации. Большая часть информации (3/4) содержится именно в так называемых плохих данных. В хороших данных содержится меньшая часть (1/4) информации биометрического образа.

Как следствие, эффективные преобразователи биометрия-код не могут быть построены простой оцифровкой биометрических параметров, а также на классических процедурах распознавания биометрических образов. При простой оцифровке данных выходные коды образа «Свой» оказываются очень нестабильными и их не удается скорректировать классическими методами обнаружения и исправления ошибок. При попытках предварительной обработки данных (до оцифровывания) классическими процедурами искусственного интеллекта происходят значительные информационные потери, выбрасывается подавляющая часть информации.

2.8.3. Метод синтеза критерия хи-квадрат распределений зависимых данных

В предыдущих разделах нами было показано, что гипотеза независимости для биометрических данных не работает. Даже сформировав вводом случайных рукописных паролей случайные биометрические образы «Чужой», мы получаем коррелированные (зависимые) биометрические данные [66, 72-74].

Для проверки статистических гипотез во многих областях исследований применяют хи-квадрат критерий, и биометрия не является исключением. Использование классических гистограмм для проверки гипотезы нормальности потребует выборку данных от 50 до 200 [66, 70]. Для этого можно применить базы биометрических образов «Чужой» и/или «Свой» [55, 72, 73].

Использование минимальной выборки из 50 примеров даст нам возможность в динамическом интервале наблюдаемого параметра разместить 10 столбиков гистограммы. Среднее количество попаданий в каждый интервал будет равен пяти. Оценка математического ожидания и среднеквадратического отклонения по этой же выборке даст возможность применения хи-квадрат критерия с 8 степенями свободы. С ростом числа столбиков гистограммы (числа степеней свободы) растет мощность критерия хи-квадрат. Возникает вопрос: может ли использование этой выборки данных увеличить количество степеней свободы критерия хи-квадрат или снизить требования к размерам исходной выборки.

Применение с помощью цифрового фильтра дополнительной (не традиционной) дискретизации данных и сглаживания результирующей гистограммы дает теоретическую возможность повысить достоверность статистических оценок, не увеличивая размер исходной выборки [72]. Но при наступлении некоторого предела это приводит к росту ошибки (к уменьшению получаемой информации).

Зависимость данных является причиной появления ошибок «сглаживания». Увеличение окна усредняющего цифрового фильтра, ведет к

увеличению коррелированности его выходных данных. Для оценки коррелированности отсчетов применяют формулу [72, 73]:

$$r \approx \left\{ 1 - \left\{ \frac{E(\Delta_{ВЫХ})}{E(\Delta_{ВХ})} \right\}^2 \right\}, \qquad (2.39)$$

где $E(\Delta_{ВЫХ})$ - математическое ожидание скачков столбцов выходной «сглаженной» гистограммы; $E(\Delta_{ВЫХ})$ - математическое ожидание скачков столбцов входной классической гистограммы [72].

При высокой коррелированности данных (2.39) использовать критерий хи-квадрат нельзя, по причине того, что он работает только в рамках предположения независимости. Чтобы решить этот вопрос была разработана методика синтеза хи-квадрат распределений зависимых данных [73, 75, 76]. Для этого с помощью разработанной программы имитации зависимых данных создадим *n* генераторов нормального белого шума - ξ_k. Для связки полученных данных перемножим их на матрицу, состоящую из единичной диагонали и одинаковых элементов (2.40) [72, 73]:

$$\begin{bmatrix} 1 & a & \cdots & a \\ a & 1 & \cdots & a \\ \cdots & \cdots & \cdots & \cdots \\ a & a & \cdots & 1 \end{bmatrix} \times \begin{bmatrix} \xi_{1,i} \\ \xi_{2,i} \\ \cdots \\ \xi_{n,i} \end{bmatrix} = \begin{bmatrix} y_{1,i} \\ y_{2,i} \\ \cdots \\ y_{n,i} \end{bmatrix} \quad (2.40), \qquad R = \begin{bmatrix} 1 & r & \cdots & r \\ r & 1 & \cdots & r \\ \cdots & \cdots & \cdots & \cdots \\ r & r & \cdots & 1 \end{bmatrix} \quad (2.41).$$

Результатом будут равнокоррелированные к друг другу случайные выходные данные (2.41). Величиной равной коррелированности будет монотонная функция *a*, которая является единственным регулируемым параметром. Затем возведением в квадрат и суммированием центрированных и номерованных случайных данных получим случайную величину, которая будет распределена по закону Пирсона для зависимых данных - $\chi^2(m, r)$ [72, 73].

При проведении эксперимента по проверке методики синтеза хи-квадрат распределений зависимых данных, используя описанную выше методику, мы рассчитали значения хи-квадрат распределений для степеней свободы m=3 и m=4. Результаты отражены в таблицах 2.5. и 2.6.

Таблица 2.5. Значения хи-квадрат распределений при разных значениях коррелированности данных при числе степеней свободы m=3

m=3		0	2	4	6	8
R	0,0	0,0	0,21	0,1	0,27	0,18
	0,5	0,18	0,2	0,08	0,27	0,17
	0,9	0,51	0,22	0,05	0,28	0,17
	0,99	0,87	0,21	0,05	0,27	0,17

Таблица 2.6. Значения хи-квадрат распределений при разных значениях коррелированности данных при числе степеней свободы m=4

m=4		0	2	4	6	8	10
R	0,0	0,0	0,18	0,12	0,08	0,04	0,02
	0,5	0,05	0,2	0,1	0,06	0,05	0,02
	0,9	0,3	0,1	0,05	0,06	0,05	0,02
	0,99	0,7	0,09	0,05	0,04	0,03	0,02

По полученным таблицам построены кривые, которые показывают изменения формы хи-квадрат распределения для зависимых данных для степеней свободы m=3 и m=4 (рис. 2.19) [72, 73].

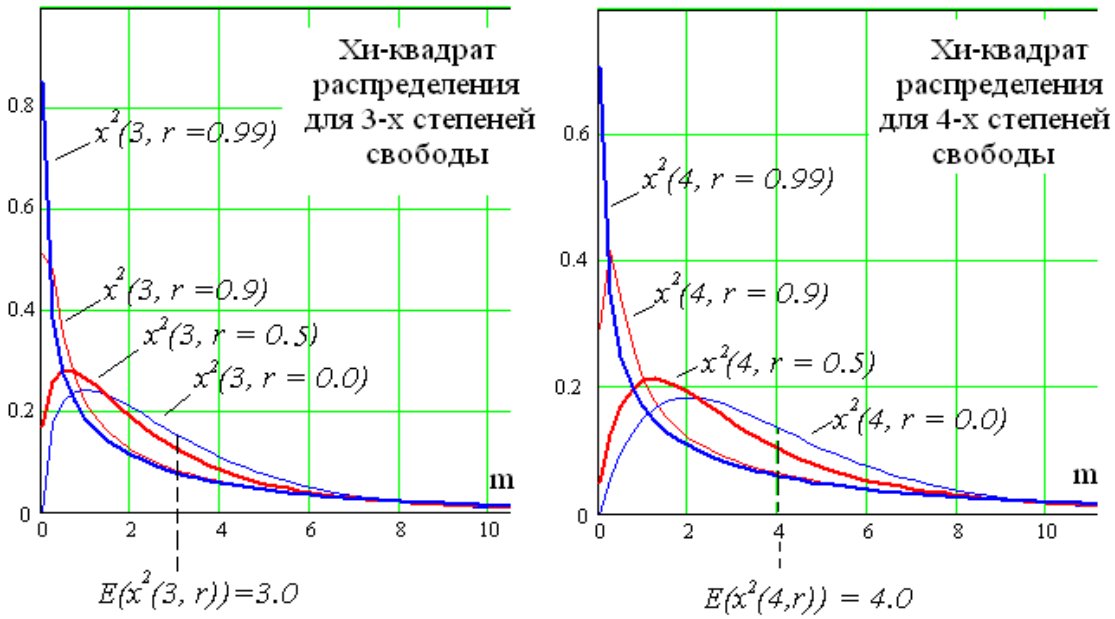


Рис. 2.19. Графики хи-квадрат распределений для 3 и 4 степеней свободы

Рисунок 2.19 показывает, что число степеней свободы хи-квадрат распределения зависимых данных и их математические ожидания имеют точное совпадение:

$$E(\chi^2(m,r)) = m \quad . \tag{2.42}$$

Свойство (2.42) работает для любых коэффициентов равной коррелированности. С повышением коррелированности данных происходит частичная утрата количества степеней свободы [72, 73].

Зная число степеней свободы и соответствующий ей коэффициент равной коррелированности данных можно увеличить достоверность оценок проверки статистических гипотез.

2.9. Выводы к главе 2

1. Показано, что принципиально важным преимуществом нейросетевых корректоров является учет неравномерности распределения ошибок в разрядах

биометрического кода, вследствие чего НС всегда дают более низкие значения ошибок первого и второго рода.

2. Впервые разработана композитная нейросетевая модель CNN-LSTM, обеспечивающая возможность эффективного распознавания пользователей на основе анализа геометрических параметров фрагментов рукописного текста изменяемого размера. В отличие от известных, разработанная модель базируется на адаптированной к условиям задачи биометрической аутентификации сверточной НС в которой каждая из карт свертки представляет собой отдельный LSTM-модуль.

3. Описывается влияние размерности НС на качество распознавания образов. Рассматривается применение энтропии для снижения размерности задачи. Показано, что можно достичь снижения размерности входной выборки за счет учета корреляционных связей между выходными сигналами НС.

4. Предлагается метод синтеза критерия хи-квадрат распределений зависимых данных. Показано, что число степеней свободы хи-квадрат распределения зависимых данных и их математические ожидания имеют точное совпадение. Для любых коэффициентов равной коррелированности это свойство имеет силу. Увеличение коррелированности данных приводит к частичной утрате числа степеней свободы. Зная число степеней свободы и соответствующий ей коэффициент равной коррелированности данных можно увеличить достоверность оценок проверки статистических гипотез.

ГЛАВА 3. ИНТЕЛЛЕКТУАЛЬНАЯ АВТОМАТИЗИРОВАННАЯ СИСТЕМА РАСПОЗНАВАНИЯ БИОМЕТРИЧЕСКИХ ОБРАЗОВ

3.1. Классификация баз данных и пользователей разработанной интеллектуальной автоматизированной системы распознавания биометрических образов

Диссертационная работа проводилась в рамках научно-исследовательских проектов КазНИТУ имени К.И. Сатпаева. В процессе реализации проектов исполнителями решались разные задачи. В результате были защищены две диссертации на соискание степени доктора PhD, где авторами были отражены полученные ими результаты. В рамках данных исследований одной из моих задач была классификации баз данных и пользователей системы распознавания образов.

Для достаточно полного отражения естественного статистического распределения биометрических признаков людей нам необходимо создавать базы естественных биометрических образов. Для подтверждения характеристик тестируемых средств необходимы достаточного размера базы биометрических образов «Свой» и «Чужой» [77, 78].

Чтобы провести достоверное тестирование требуются небольшие базы биометрических образов «Свой» и большие базы биометрических образов «Чужой» (свыше 10^{12} образов) [32, 77, 79, 80]. Если формирование баз «Свой» не вызывает затруднений, то формирование баз «Чужой» требует больших затрат времени и труда. Выходом из данного положения является использование усеченных баз «Чужой», содержащих 10^3–10^5 образов, полученных в рамках работы системы [28, 32, 77].

Имеющиеся базы можно дополнять с помощью специальных технологий, например, путем случайного выбора людей и получением от них случайных образов, что вполне возможно для сбора рукописных паролей. Чтобы

гарантировать равномерное заполнение полученного таким образом многомерного пространства «Все Чужие», необходимо заполнить имеющиеся в нем пустоты, используя, например, алгоритм скрещивания биометрических образов [16].

Нестабильность присуща всем биометрическим образам. При преобразовании биометрического образа в криптографический ключ основной проблемой является их неоднозначность, нестабильность и размытость [18, 60]. Поэтому нам необходимо держать под контролем нестабильность ввода биометрического образа и классифицировать пользователей по показателю нестабильности [79].

Стабильность ввода биометрических образов определяется вектором контролируемых биометрических параметров и зависит от умения пользователем вводить биометрический образ. Поэтому при вводе рукописного образа необходимо выбрать такое слово, которое наиболее стабильно для конкретного почерка, а затем тренироваться его стабильно вводить.

Для всех пользователей системы проводится классификация по стабильности ввода биометрических образов. Для этого необходимо оценить стабильность пользователей и построить нормированное распределение показателя стабильности (рис. 3.1).

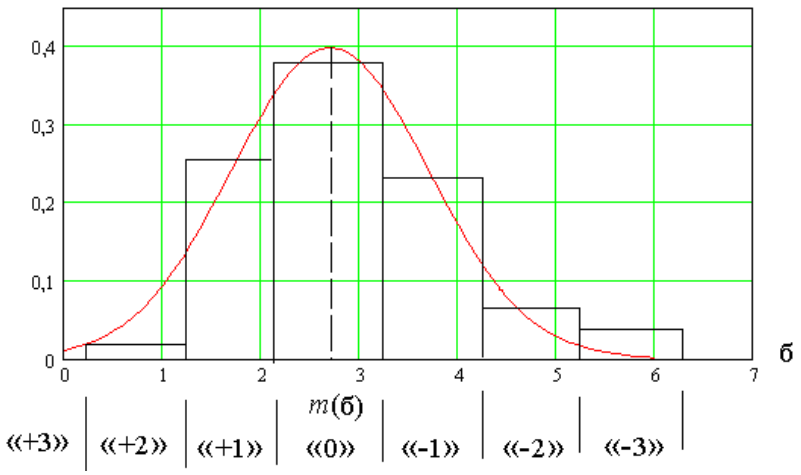


Рис. 3.1. Распределение пользователей по стабильности их биометрических образов

Столбцы гистограммы соответствуют классу стабильности пользователей. Для проведения классификации распределение «Все Свои» было разделено на интервалы, которые равны дисперсии, а центром интервала было математическое ожидание. В итоге была получена классификация стабильности «+3», «+2», «+1», «0», «–1», «–2», «–3». Классификация должна строиться только для дееспособных людей.

Стабильность введения биометрических образов это важный статистический показатель, поэтому формируемые базы биометрических образов должны хорошо отражать реальные показатели по процентному содержанию биометрических образов, которые принадлежат к разным классам стабильности.

Еще одним важным параметром для биометрических систем является степень уникальности биометрического образа «Свой» [79].

Чтобы оценить уникальность биометрического параметра необходимо вычислить вероятность попадания в интервал «Свой» при вводе образов «Все Чужие» по формуле:

$$P_{\text{c}} = \frac{1}{\sqrt{2\pi}\sigma_{\text{B}}} \int_{m_{\text{c}}-3\sigma_{\text{c}}}^{m_{\text{c}}+3\sigma_{\text{c}}} \exp\left(\frac{(m_{\text{B}} - x)^2}{2\sigma_{\text{B}}^2}\right) \mathrm{d}x\,, \tag{3.1}$$

где m_{c}, σ_{c} – математическое ожидание и дисперсия распределения параметра «Свой»; m_{B}, σ_{B} – математическое ожидание и дисперсия распределения параметра «Все чужие».

Уникальностью конкретного биометрического параметра будет являться величина обратная вероятности (3.1). Вычислить среднюю уникальность биометрических параметров или уникальность биометрического образа можно с помощью

$$U = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{P_{C,i}}\,. \tag{3.2}$$

Уникальность может быть применена для классификации биометрических образов, которая проводится с помощью статистического исследования большого количества биометрических образов.

В первой главе диссертации в сравнительном анализе биометрических технологий мы показали, что у различных биометрических образов разная информативность (сложность).

При использовании статических биометрических образов человек не может их менять (усложнять) по своему усмотрению. Поэтому, у статического образа низкая информативность и он легко компрометируется [50, 81].

При использовании динамических биометрических образов человек их может изменять. Например, можно изменить вводимый рукописный образ сменив слово-пароль или голосовой образ изменив фразу [50, 82]. Чтобы повысить стойкость динамических биометрических образов к атакам подбора можно их усложнять и хранить в тайне.

Стойкость биометрических образов зависит от их информативности (сложности), поэтому при формировании баз биометрических образов необходимо выбирать длину слов–паролей не менее чем из 5 букв, а при использовании отпечатков пальцев использовать не менее 22 особенностей.

3.2. Методика формирования биометрических баз рукописных образов и отпечатков пальцев

Следующей моей задачей в рамках проектов была разработка методики формирования биометрических баз рукописных образов и отпечатков пальцев. Для создания методики формирования биометрической базы рукописных образов нами проводился анализ работ [50, 61, 78, 83, 84].

Целями разработанной методики являются:

− формирование базы естественных рукописных образов большого размера;

− исследование распределений статистических моментов естественных рукописных образов;

− проведение ускоренного тестирования распознавания биометрических образов из базы естественных рукописных образов большого размера;

− проведение полного тестирования системы распознавания образов по рукописному слову-паролю с использованием биометрических образов базы;

− формирование баз реальных биометрических образов «Чужой» по рукописному слову-паролю, которые можно встроить в средства аутентификации.

Данной методикой нами определен порядок формирования биометрической базы. Сбор биометрических образов осуществляется в три этапа.

Первый этап – отбор доноров биометрии. Биометрические образы должны быть получены от доноров, которые могут пользоваться биометрическими средствами, поэтому этот этап является обязательным [85].

Второй этап – обучение донора и создание базы биометрических образов «Свой». К обучению доноров относиться освоение донором программы, устройства ввода, выработка навыков стабильного написания рекомендуемого слова. При создании базы биометрических образов «Свой» донор должен ввести 40 образов предложенного слова. На данном этапе контролирующее лицо должен следить за действиями донора, вести протокол и при необходимости требовать от донора повторного ввода биометрического образа.

Третий этап – формирование базы биометрических образов «Чужой». На этом этапе донор должен воспроизводить на устройстве считывания, задаваемые программным модулем слова. Программный модуль задает разные слова длиной от пяти до семи букв, которые необходимо писать один раз.

Рекомендуемое нами время на формирование баз «Свой» и «Чужой» по разработанной методике не должно быть больше 80 минут [86]. Увеличение времени приводит к ухудшению качества вводимых рукописных образов.

Работа доноров биометрии является затратной частью формирования баз биометрических образов, поэтому для сокращения затрат мы рекомендуем, чтобы один донор был привлечен к формированию нескольких видов баз динамических биометрических образов [85].

Формирование биометрической базы отпечатков пальцев также осуществляется в три этапа.

Первый этап – отбор доноров биометрии. Биометрические образы должны быть получены от дееспособных людей (людей, способных пользоваться биометрической защитой), поэтому этот этап является обязательным.

Второй этап – обучение донора и создание базы биометрических образов «Свой». К обучению доноров относиться освоение им программы, устройства ввода, выработка навыков стабильного ввода отпечатков пальцев. При создании базы биометрических образов «Свой» донор должен внести 20 образов. На данном этапе контролирующее лицо должен следить за действиями донора, вести протокол и при необходимости заставляет его перемещать палец для правильного ввода биометрического образа.

Чтобы отпечаток пальца меньше деформировался его необходимо отрывать и снова прикладывать к сканеру. Перекатывание пальца сильно деформирует рисунок папиллярных линий и это влияет на качество базы.

Для создания полной базы биометрических данных сканирование необходимо провести для каждого пальца каждой руки пользователя.

3.3. Архитектура интеллектуальной автоматизированной системы распознавания биометрических образов

В разработанной нами интеллектуальной автоматизированной системе легитимному пользователю системы выдается ключ длиной 256 бит. Данный ключ невозможно передать другому пользователю. Биометрические образы не хранятся в виде образов или их кодов. В формируемом системой нейросетевом контейнере хранятся только параметры обученной НС. Обучение НС

проводится на 7-20 биометрических образах. Для считывания биометрических образов используются устройства, описанные в разделе 2.1 диссертации.

На рис. 3.2 представлена архитектура автоматизированной системы распознавания образов, реализованной на языке СИ.

В состав системы входят следующие подсистемы: инициализации приложения (ИП); поддержки графического интерфейса и выполнения основного функционала (ПГИВОФ); поддержки графического интерфейса диалога тестирования (ПГИДТ); поддержка графического интерфейса диалога генерации ключа (ПГИДГК).

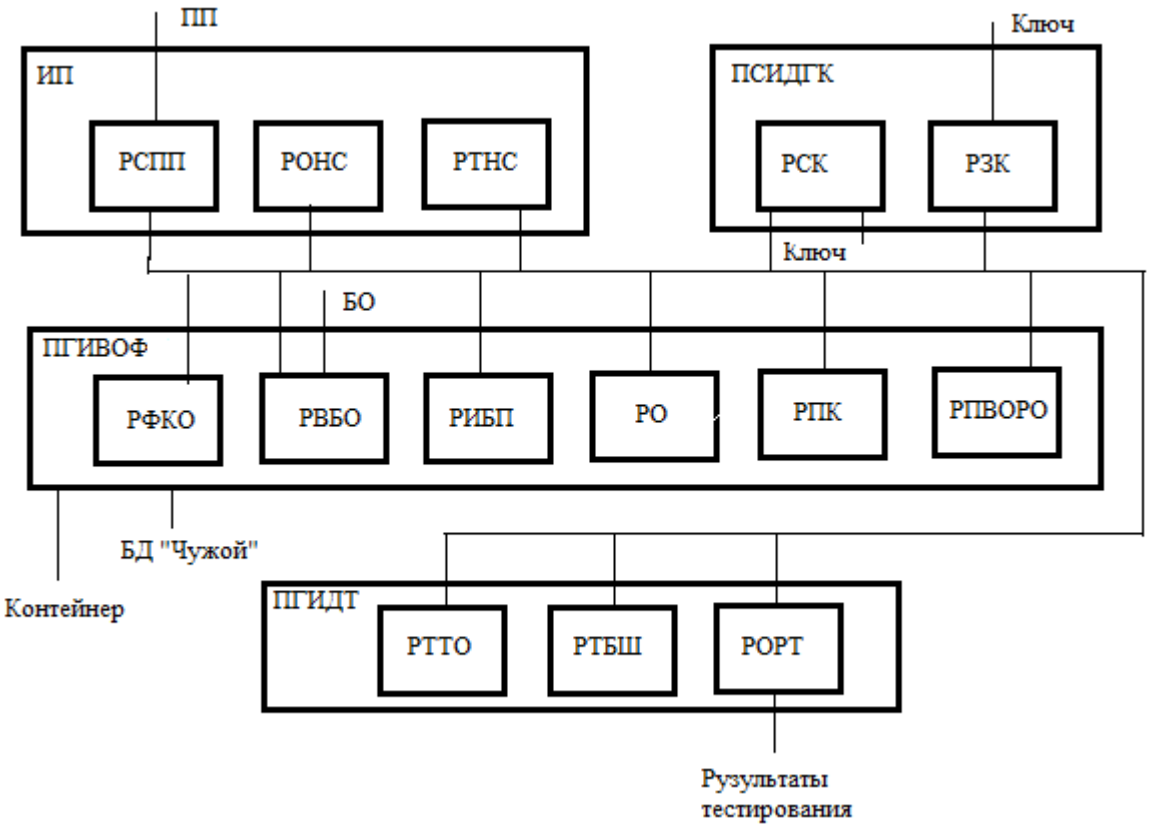


Рис. 3.2. Архитектура автоматизированной системы распознавания образов

Подсистема инициализации приложения (ИП) – поддержка интерфейса программы, создание пользовательского пароля, инициализация режима обучения НС и режима тестирования НС.

Подсистема поддержки графического интерфейса и выполнения основного функционала (ПГИВОФ) – формируется клиентская область для ввода биометрических образов, ввод биометрических образов, извлечение биометрических параметров, обучение НС, получение ключа, его хэширование, проверка введенного образа в режиме обучения, формирование биометрического контейнера, таблиц весовых коэффициентов.

Подсистема поддержки графического интерфейса диалога тестирования (ПГИДТ) – позволяет проводить тестирование ручное и автоматическое на тестовых выборках и «белом шуме», делает обработку результатов тестирования и выводит посчитанные результаты.

Подсистема поддержка графического интерфейса диалога генерации ключа (ПГИДГК) – позволяет сохранить сгенерированный ключ, загрузить ранее сохраненный ключ.

Каждая подсистема состоит из режимов работы. Перечень приведен в таблице 3.1.

Таблица 3.1 – Режимы работы подсистем

Название режима	Функция
РСПП	Режим создания пользовательского пароля. Позволяет создавать пользователю 256 битный (32 символьный) пароль или сгенерировать его автоматически и применять его для обучения НС.
РОНС	Режим обучения НС.
РТНС	Режим тестирования НС.
РСК	Режим сохранения ключа.
РЗК	Режим загрузки ключи.
РФКО	Режим формирования клиентской области.
РВБО	Режим ввода биометрических образов.

РИБП	Режим извлечения биометрических параметров.
РО	Режим обучения.
РПК	Режим получения ключа.
РВВОВО	Режим проверки введенного образа в режиме обучения.
РТТО	Режим тестирования на тестовых образах.
РТБШ	Режим тестирования на «белом шуме».
РОРТ	Режим обработки результатов тестирования.

После ввода биометрических образов, производится извлечение биометрических параметров. Далее в режиме обучения осуществляется обработка полученных биометрических параметров. Блок-схема режима обучения НС представлена на рис. 3.3.

Входными данными системы являются:

1) биометрические образы, вводимые с устройства считывания;
2) пароль легитимного пользователя;
3) база биометрических параметров «Все Чужие».

Из биометрических образов извлекаются биометрические параметры, из которых формируется матрица биометрических параметров. В матрице строками являются вектора биометрических параметров, которые имеют различное число элементов, формат и тип. Объединив столбцы матрицы можно сформировать набор векторов нескольких биометрических параметров. Матрица биометрических параметров нами определена согласно требованиям ГОСТ Р 52633.0–2006 [60].

База естественных биометрических образов «Все Чужие» предназначена для тестирования качества обучения системы. Обучение считается качественным, когда в результате тестирования получаем требуемые значения вероятностей ОПР (P_1) и ОВР (P_2).

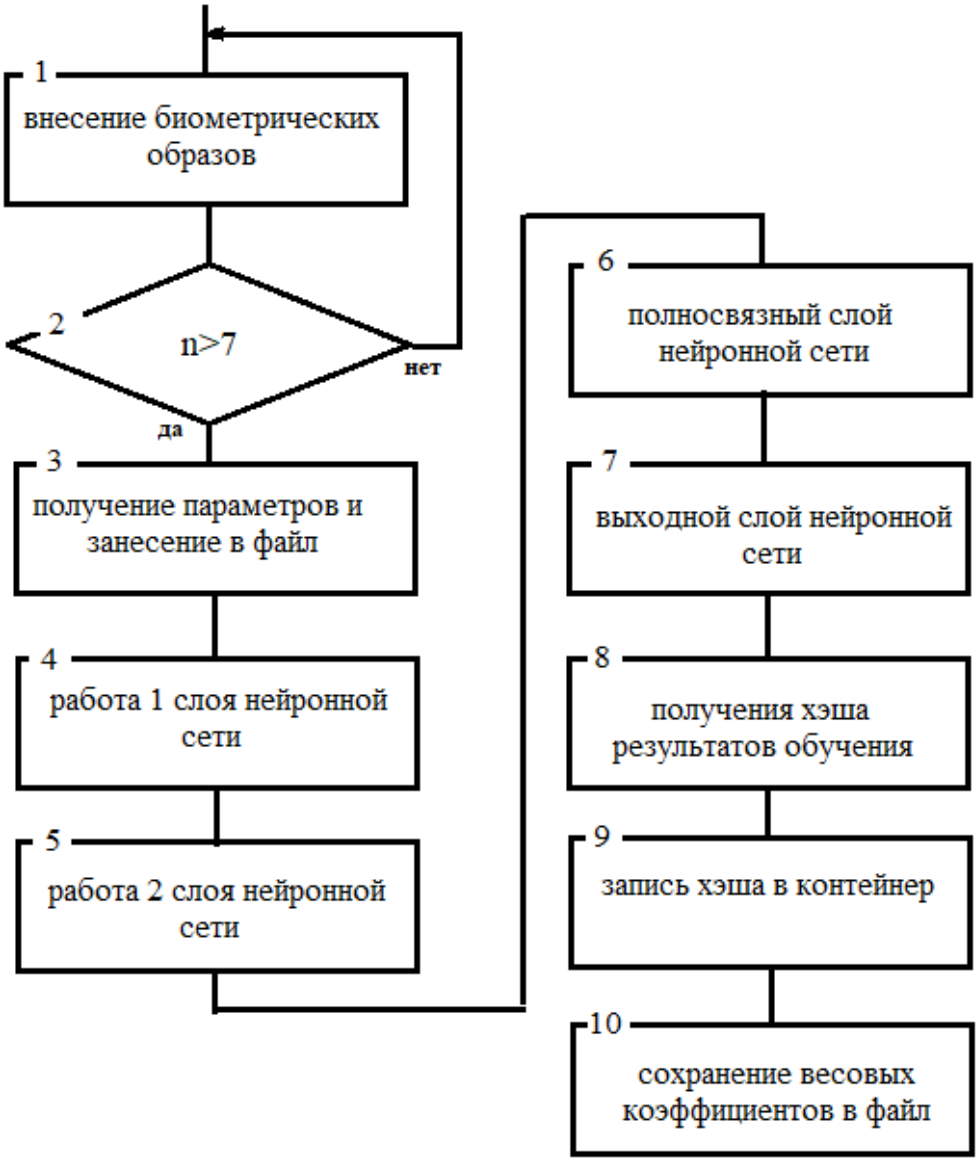


Рис. 3.3. Блок-схема режима обучения НС

Биометрический контейнер служит для связывания ключа/пароля пользователя с его личной биометрией. Общие требования к структуре биометрического контейнера определяются в соответствии с ГОСТ Р 52633.4–2011 [33].

3.4. Система распознавания рукописных образов

Согласно второму этапу формирования баз биометрических образов донорам биометрии необходимо выработать навыки стабильного написания

слова-пароля и освоить разработанную автоматизированную систему распознавания биометрических образов «Нейро-Тест 1.2» [87].

В программе используется эмулятор ИНС, разработанной в главе 2.

На рис. 3.4 показано главное окно программы [57, 87].

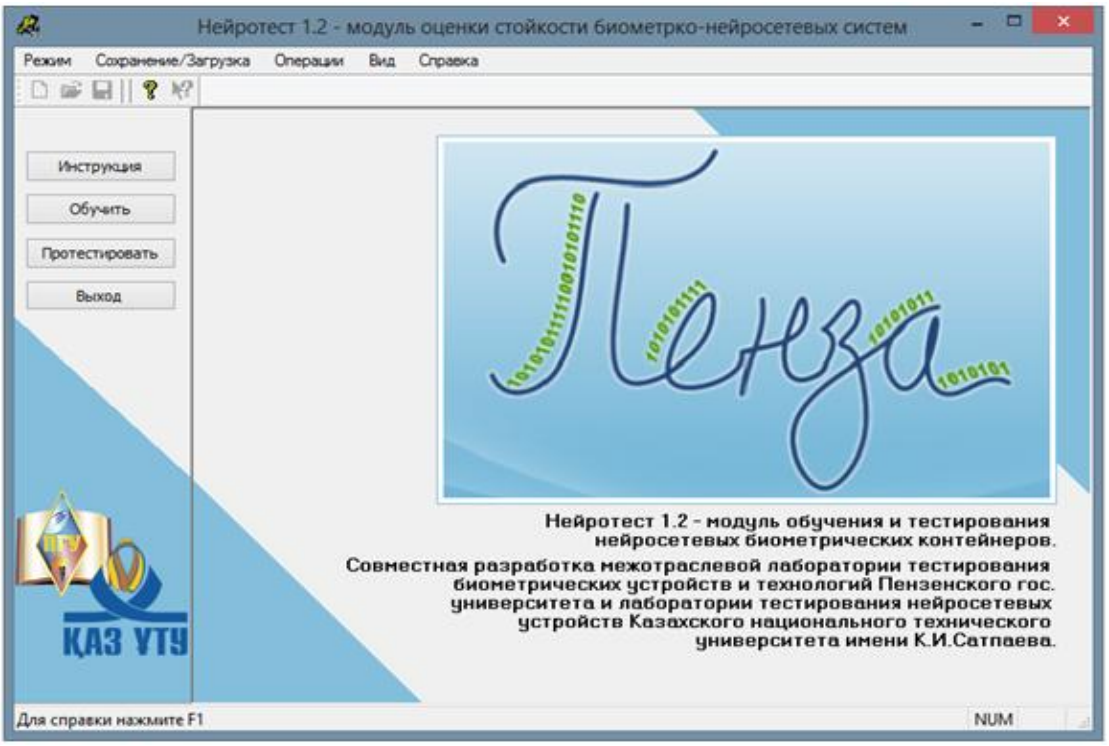


Рис. 3.4. Главное окно программы «НейроТест 1.2»

Работа программы в режиме обучения. Для инициализации режима обучения необходимо нажать кнопку «Обучить» или выбрать в пункте «Режим» главного меню пункт «Обучение системы». Для обучения системе нужно задать пароль (ключ) донора, для этого необходимо в пункте «Режим» главного меню выбрать пункт «Задать пароль» (рис. 3.5) [87].

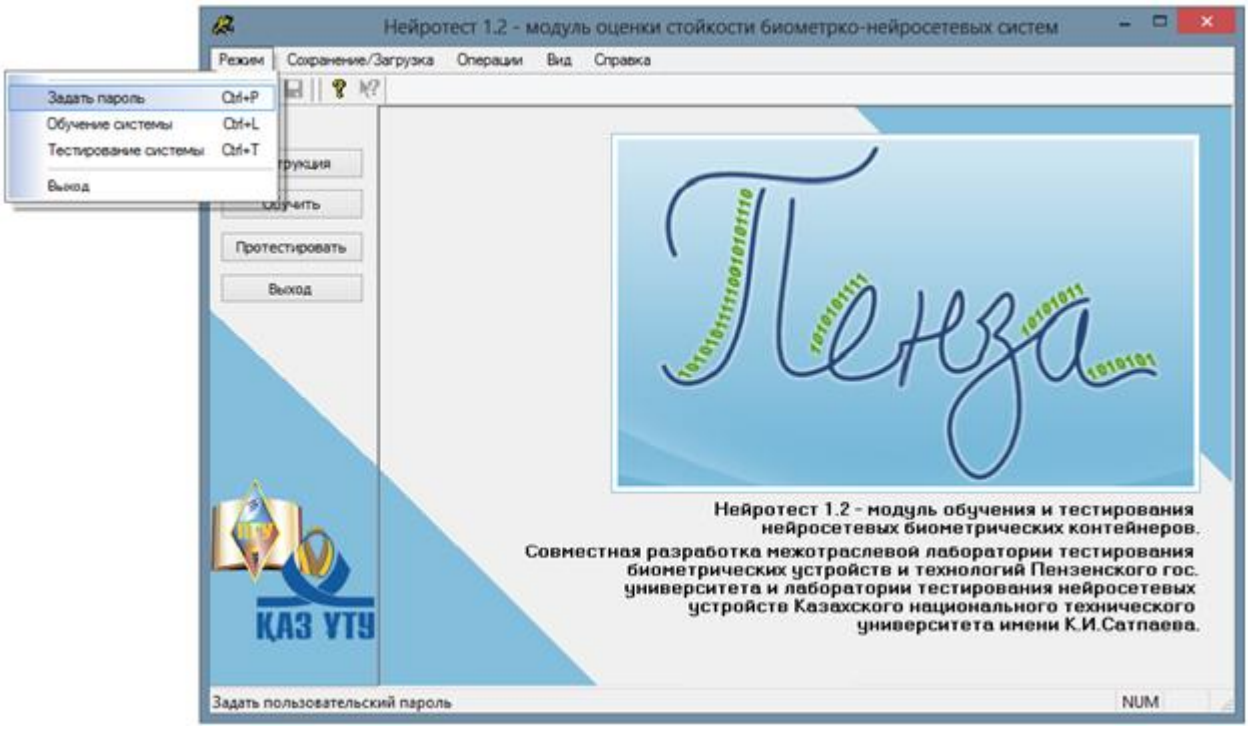


Рис. 3.5. Регистрация нового пользователя

В появившемся окне (рис. 3.6) необходимо внести пользователя и пароль. Также можно нажатием на кнопку «Автоматически сгенерировать новый пароль» задать пароль для пользователя. Максимальная длина пароля 32 символа (256 бит). Если пользователь уже зарегистрирован в системе, то необходимо внести пользователя и нажатием на кнопку загрузить ранее созданный пароль [87]. Затем нажать кнопку «ОК».

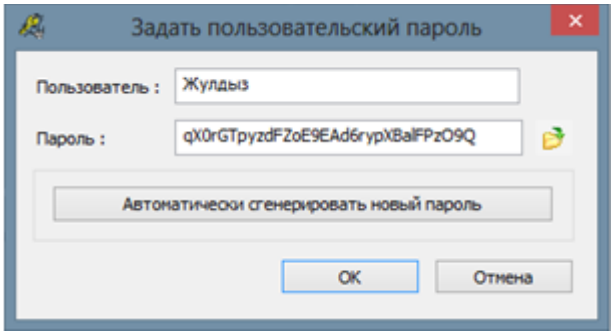


Рис. 3.6. Окно для задания пользовательского пароля

В результате появиться окно с запросом на использование введенного пароля при обучении. Нажатие на кнопку «Да» является началом обучения системы.

Далее необходимо воспроизвести на графическом планшете или другом устройстве заранее заданное или выбранное слово (рис. 3.7.) [87].

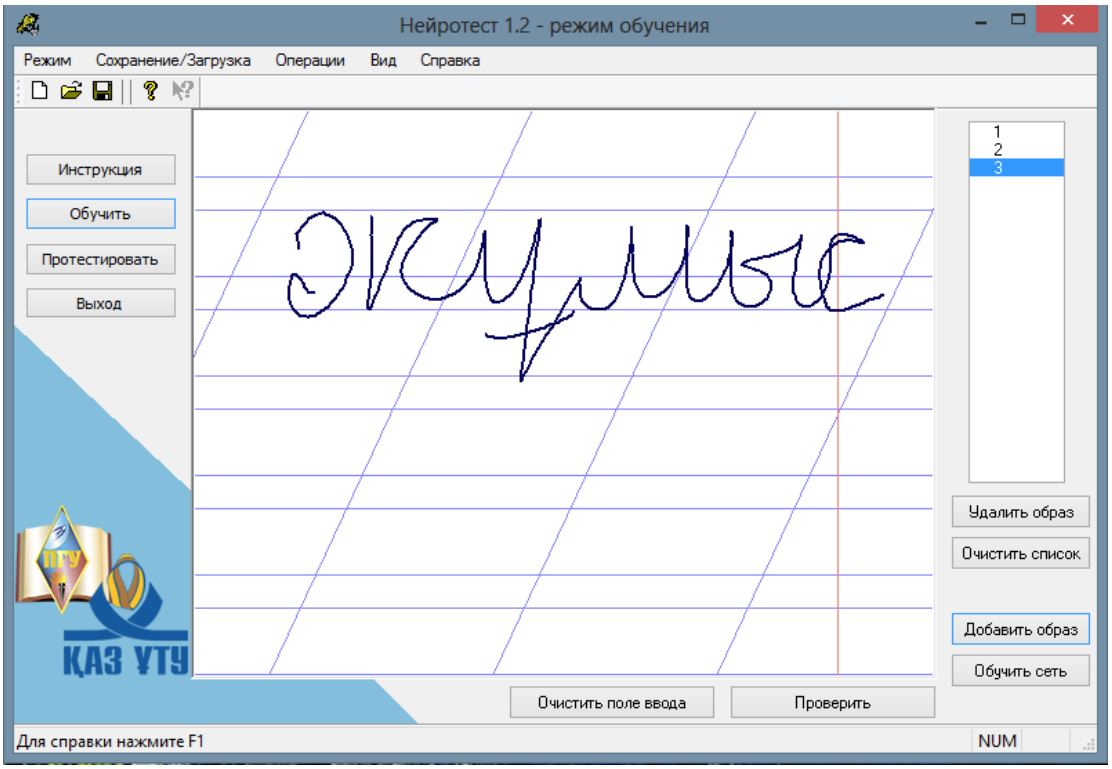


Рис. 3.7. Введение и просмотр рукописных образов

Добавление введенного образа в базу примеров осуществляется нажатием на кнопку «Добавить образ». После этого поле очищается для ввода следующего образа, а в правой части окна увеличивается количество введенных примеров. Кнопка «Очистить поле ввода» позволяет удалить введенное слово без занесения в базу примеров. Таким образом, необходимо воспроизвести не менее 7 слов [53].

Для просмотра сохраненных примеров нужно щелкнуть мышкой на интересующем номере в правой части окна. Не понравившийся пример можно удалить нажатием на кнопку «Удалить образ» [87].

Для удаления всех образов применяется кнопка «Очистить список».

Обучение НС системы осуществляется нажатием кнопки «Обучить сеть». На экране появиться окно с результатами обучения НС (рис. 3.8). В результатах отражаются группа стабильности пользователя и вероятности ОПР и ОВР [53, 86, 87]. Как производится классификация пользователей по стабильности, было описано в разделе 3.1.1. данной диссертации.

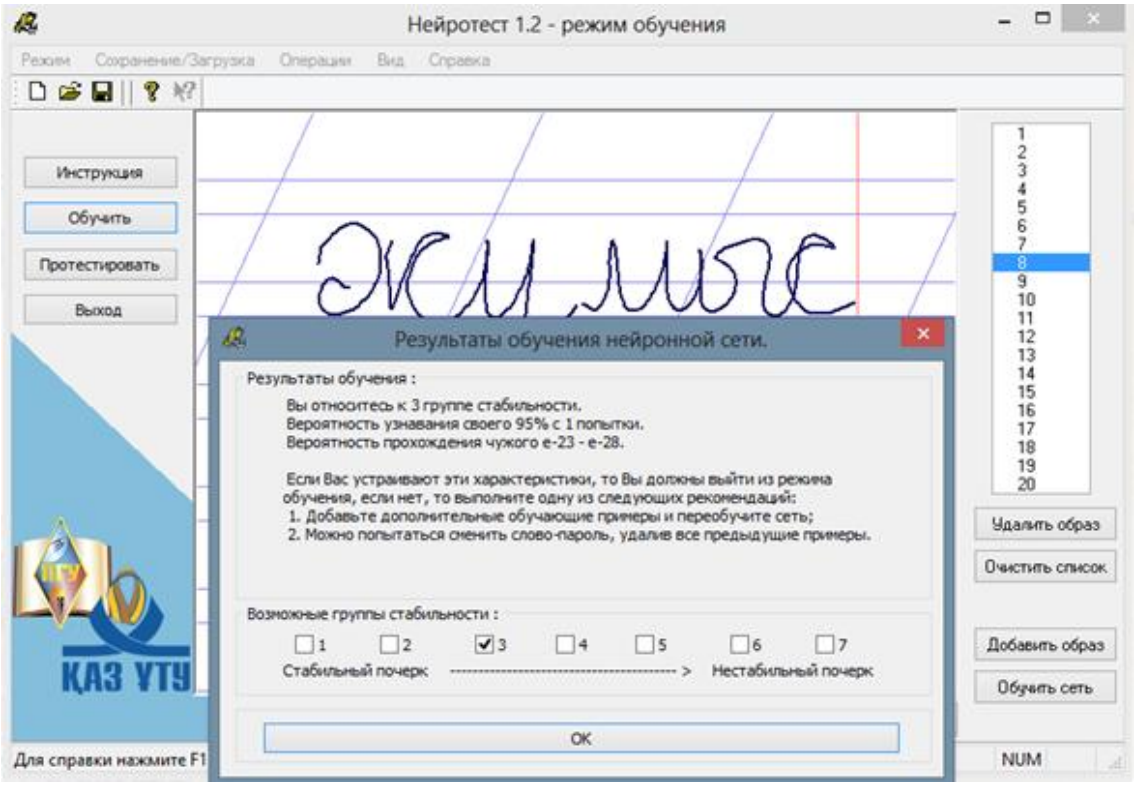


Рис. 3.8. Результаты обучения нейронной сети

В случае если донора не устраивают полученные результаты обучения, то можно следовать выданным рекомендациям, самостоятельно изменить номер группы стабильности и переучить НС. Мы рекомендуем изменять номер группы стабильности не более чем на одну или две, то есть лучше переходить в соседние группы [86, 87].

Например, пусть пользователь оказался в пятой группе стабильности, у которой не совсем хорошие показатели вероятностных характеристик. Тогда можно изменить условия обучения, одним из следующих действий:

а) удалить наиболее непохожие примеры;

б) удалить все внесенные образы и заново их написать;

в) добавить нескольких дополнительных обучающих примеров. После этого заново обучить систему, нажав кнопку «Обучить сеть». Если после нескольких попыток обучить сеть попасть в желаемую группу стабильности не удается, то необходимо сменить слово-пароль [87].

Режим контроля распознавания «Своего». После обучения НС нужно проверить качество распознавания системой биометрического образа «Свой». Для этого необходимо воспроизвести рукописное слово и нажать кнопку «Проверить». При этом появляется окно (рис. 3.9), показывающее сгенерированный НС ключ в двоичной кодировке. Биты, которые не совпали с исходным ключом, будут отмечены звездочками. Для отображения ключа в шестнадцатеричной кодировке необходимо нажать на кнопку «Символьное представление» [87].

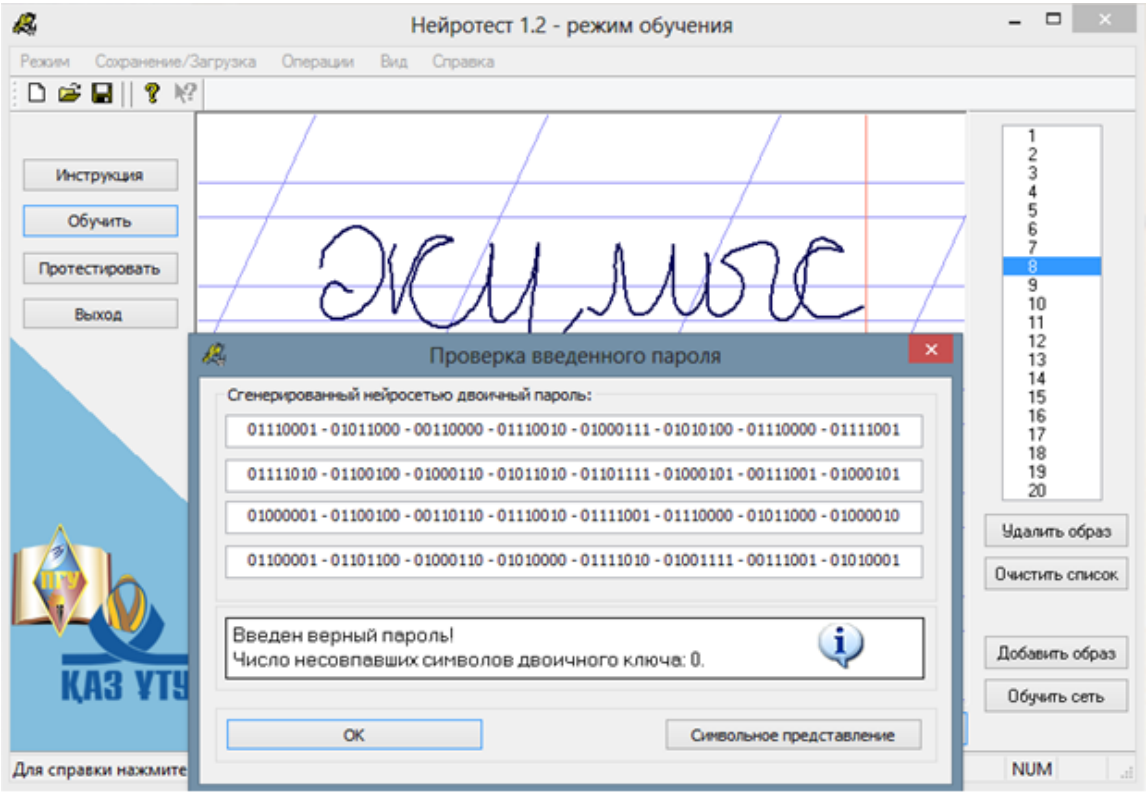


Рис. 3.9. Проверка введенного образа «Свой»

Полученные данные можно использовать, чтобы самостоятельно протестировать систему.

Если образ «Свой» плохо распознается системой, то необходимо добавить в базу несколько новых примеров и заново обучить сеть. Сеть будет лучше узнавать рукописный образ донора, но могут ухудшиться некоторые характеристики системы.

Если донора устраивают результаты обучения, то данные нужно сохранить на диск. Для этого в пункте главного меню «Сохранение/загрузка» нужно выбрать пункт «Сохранить образы на диск». Для каждого донора создается папка, название которой соответствует пользователю или номеру персонального идентификатора донора биометрии [53, 54, 87].

3.5. Формирование обезличенной тестовой рукописной базы образов

Процесс и порядок формирования биометрической базы тестовых биометрических образов определяется разработанной нами в разделе 3.2 методикой и включает этапы отбора доноров биометрии, освоения донором биометрии тестовой программы и собственно формирования базы биометрических образов.

Этап отбора доноров биометрии обязателен по причине того, что биометрические образы должны быть получены от доноров биометрии, способных пользоваться средствами биометрии. Проводится специалистами, входящими в группы профессионально-психологического отбора.

Перед каждым новым сеансом необходима проверка психофизиологического состояния доноров – оно должно быть нормальным.

Каждый донор должен уметь работать с программой сбора биометрических образов. С этой целью перед сбором биометрических образов необходимо в течение 40 минут ознакомить доноров с программой, дать им

возможность потренироваться и закрепить полученные навыки, провести контрольную тест-проверку.

Для обезличенности персональных биометрических данных каждый донор должен иметь персональный идентификационный номер, который должен знать только он один.

Каждый донор должен знать цели и задачи сбора биометрических рукописных образов. Поэтому эта информация должна быть доведена до них до начала сбора данных.

Доноры биометрии – люди, средний возраст которых колеблется от 18 до 23 лет. Все доноры являются уверенными пользователями ПЭВМ.

Интеллектуальная автоматизированная система распознавания биометрических образов по особенностям его рукописного почерка «Нейро-Тест 1.2» включает в себя персональный компьютер, графический планшет, программный модуль «Нейрокриптон-формирователь биометрических баз» с прилагаемыми к нему словарями слов.

Персональный компьютер предназначен для установки программного модуля со словарями, хранения и обработки сформированных электронных биометрических данных.

В качестве персональных компьютеров использовались стационарные и переносные компьютеры.

Системные требования: Windows® 2000/NT/XP, 800 КБ на жестком диске для самого программного модуля и 6 словарей.

В качестве преобразователя графического начертания естественных рукописных образов в электронные образы применяется графический планшет GenusWizardPen.

После этапа обучения и тестирования качества ввода рукописных образов донор биометрии под контролем инструктора приступает к формированию обезличенной рукописной базы биометрических образов. Для этой цели используется программно-аппаратный модуль «Нейрокриптон - формирователь биометрических баз», который включает следующие модули:

BioImgDBCreator.exe – модуль предназначен для сбора тестовых рукописных образов «Свой» и «Чужой».

Каждому новому пользователю предлагается ввести 40 одинаковых образов, которые в дальнейшем будут использоваться как образы «Свой», и 100 различных случайных слов «Чужой», которые затем будут добавлены в общую обезличенную базу тестовых образов.

AlmatyTurkestanPenza.exe – модуль предназначен для сбора баз близких образов. Собранные образы в дальнейшем будут использоваться для оценки искусственно ослабленных систем биометрической аутентификации.

bnc32.dll – биометрико-нейросетевая библиотека, поддерживающая работу НС (моделирование, обучение, тестирование). Также осуществляет преобразование полученных координат рукописных образов в функционалы.

Программа автоматически предлагает донору биометрии слова для воспроизведения на графическом планшете из словарей: *diction_full.txt* и *Almaty_Turkestan_Penza.txt.*

Программа позволяет получать координаты X и Y вводимых пользователем рукописных образов. Также фиксируются время и давление каждой новой точки рукописного образа. Основные характеристики снимаемых с помощью планшета фирмы Genius данных представлены в табл. 3.2.

Таблица 3.2 – Снимаемые данные (для планшета и реально получаемые)

	теоретический		Практический	
	минимум	максимум	минимум	максимум
Координата X	0	9268	1	9267
Координата Y	0	6268	1	6266
Давление P	0	1024	1	1023
Время T (в сек.)	-	-	1	20
Количество точек	-	-	100	3565

Для примера приведем характеристики слова *«Кеңсе»:*

Количество точек в образе: 577.

Координаты X изменяются от 1570 до 8058.

Координаты Y изменяются от 1974 до 4433.

Давление (нажим) пера меняется от 35 до 618.

На написание образа затрачено 4 секунды.

Вводимые пользователем образы сохраняются в папке Data. Каждому новому пользователю присваивается свой уникальный идентификатор, например, *0A12E317B986A712*. После завершения сеанса одного пользователя программа сохраняет в отдельные файлы полученные образы «Своего» (*.mimg), образы «Чужого» (*.simg) и обученный биометрико-нейросетевой контейнер (*.bnc), который в дальнейшем используется при аутентификации данного пользователя.

Словарь *diction_full* представляет из себя текстовый документ, содержащий 10 000 слов длиной от 4 до 6 букв на казахском языке (рис. 3.10). Аналогичные словари для этого примера и в дальнейшем существуют для русского и кыргызского языков.

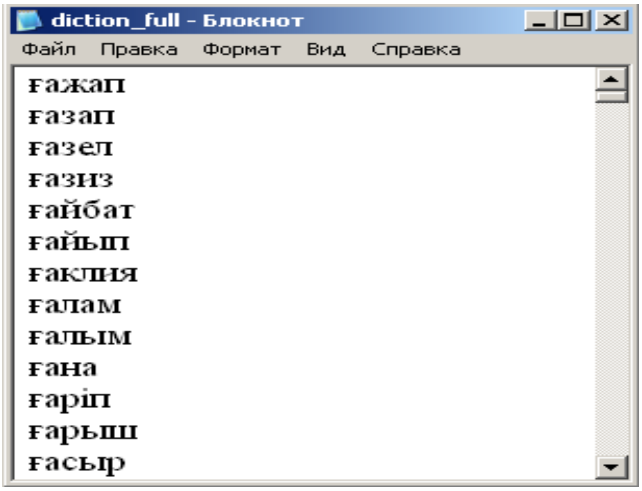


Рис. 3.10. Словарь diction_full

В дальнейшем программа случайным образом выбирает слова из данного словаря и предлагает пользователю для ввода. Это основной словарь модуля формирования базы тестовых образов. При необходимости данный словарь

можно редактировать (например, можно добавлять новые слова). Данный словарь используется модулем *BioImgDBCreator.exe.*

Словарь *Almaty_Turkestan_Penza.txt* представляет из себя текстовый документ, содержащий 597 слов (рис. 3.11).

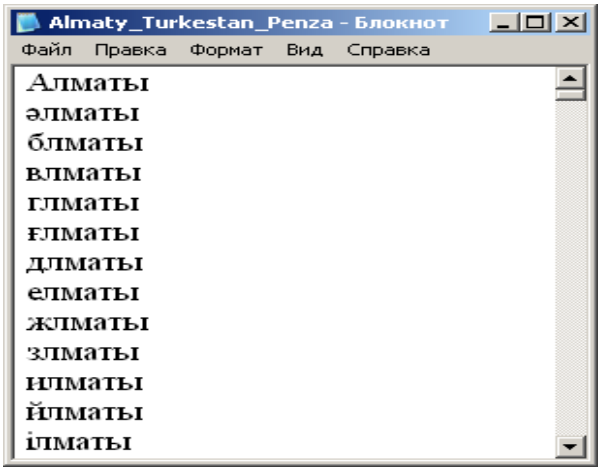


Рис. 3.11. Словарь Almaty_Turkestan_Penza.txt

В словаре содержатся слова «Алматы», «Түркістан» и «Пенза», с одной измененной буквой, например *Абматы*, *Шлматы, Алматя* и т.п. В дальнейшем собранные с использованием данного словаря тестовые образы используются при тестировании биометрических систем при условии частично скомпрометированного пароля легального пользователя. Данный словарь используется модулем *AlmatyTurkestanPenza.exe.*

При формировании обезличенных баз естественных биометрических образов необходимо учитывать, что в среднем образы от каждого нового донора занимают 800 КБ: 40 образов «Свой», 100 образов «Чужой» и обученный биометрический контейнер.

Процесс сбора биометрических данных происходит следующим образом.

После запуска программы сбора биометрических образов появляется окно аутентификации пользователей, представленное на рис. 3.12.

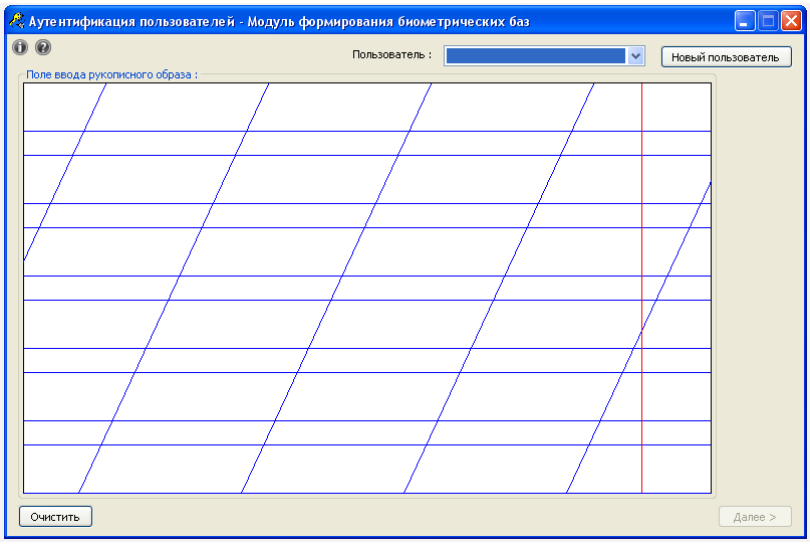


Рис. 3.12. Главное окно программы (окно аутентификации пользователей)

Если пользователь впервые работает с программой, то необходимо создать новую учетную запись данного пользователя. Для этого необходимо нажать кнопку «Новый пользователь». После этого появится окно ввода рукописных образов, представленное на рис. 3.13.

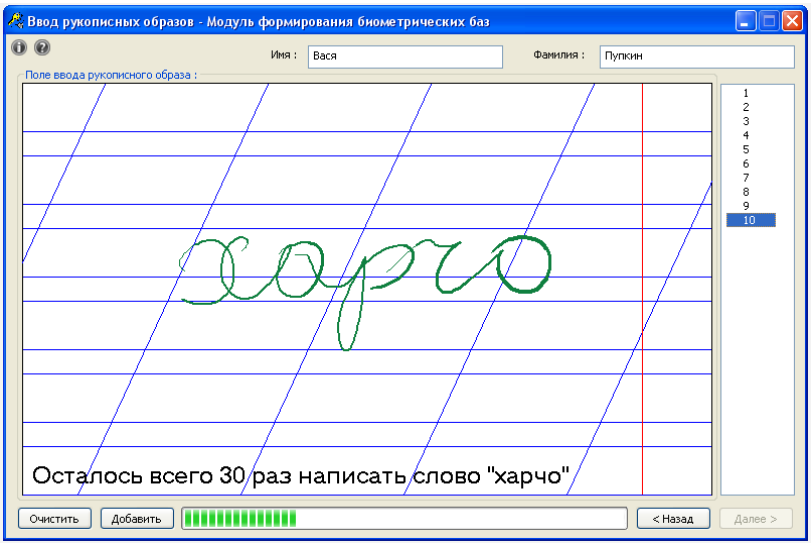


Рис. 3.13. Формирование базы образов «Свой»

В поле ввода необходимо ввести условные кодовые данные пользователя (назначаются администратором или контролирующим лицом с целью обезличивания личности и его персональных биометрических данных в базе

естественных рукописных данных) работающего с системой пользователя (в дальнейшем эти данные будут использоваться при аутентификации пользователя при повторной работе с системой). После выполнения указанной выше процедуры необходимо на поле ввода рукописных паролей 40 раз написать предложенное системой слово. Система предлагает для написания случайно выбранное слово из словаря. После написания одного слова пользователю следует нажать кнопку «Добавить», после чего введенный образ добавится в список образов и счетчик оставшихся образов уменьшится на единицу. Если введенное слово не соответствует обычному почерку пользователя, то следует нажать кнопку «Очистить». После очистки поля ввода можно приступать к повторному написанию. Правильность ввода проверяет контролирующее лицо.

Для формирования базы образов «Чужой» нужно нажать кнопку «Далее», которая активизируется после ввода всех образов «Свой».

Для прекращения формирования баз или отмены сеанса пользователя нужно нажать на кнопку «Назад». При этом вся введенная ранее информация будет удалена.

Введенные данные сохраняются только при нажатии на кнопку «Далее».

При формировании базы образов «Чужой» пользователь будет вводить случайным образом выдаваемые системой слова.

Инструктор и донор должны внимательно следить за тем, чтобы вводимое слово точно соответствовало предложенному системой слову.

После ввода 100 рукописных образов становится активной кнопка «Далее». Нажатие данной кнопки приводит к сохранению введенных пользователем образов и завершению сеанса работы текущего пользователя. После чего следующий донор биометрии может приступать к работе с системой.

3.6. Тестирование системы распознавания биометрических образов с использованием сформированных тестовых рукописных баз

В программе «Нейро-Тест 1.2» тестирование сформированной базы рукописных образов может проводиться в режиме обучения и в режиме тестирования. Предусмотрено ручное и автоматизированное тестирование.

Для инициализации режима тестирования необходимо нажать на кнопку «Протестировать» (рис. 3.5) или в пункте главного меню «Режим» выбрать пункт «Тестирование сети» [87].

Тестирование может проводиться только для обученной НС. Режим тестирования позволяет проверять вводимые рукописные образы, добавлять «удачный» образ в базу обучающих примеров.

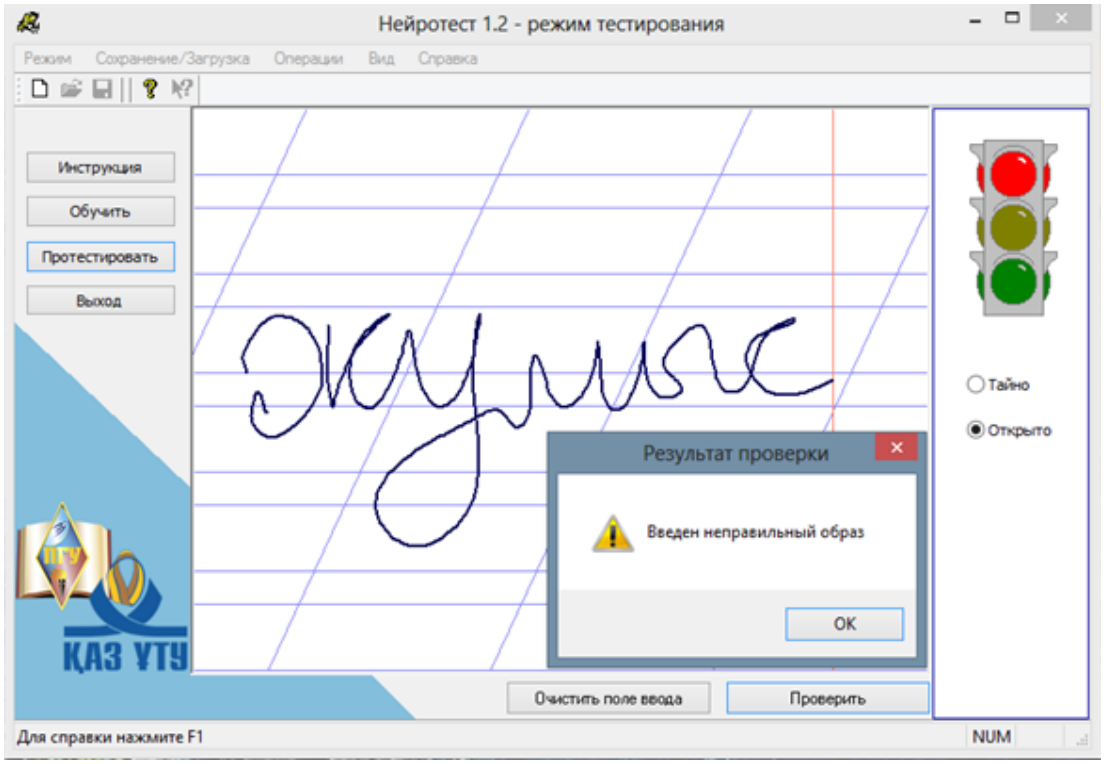


Рис. 3.14. Ввод неправильного рукописного образа

Для тестирования системы воспроизводиться рукописное слово-пароль и нажимается кнопка «Проверить». Расположенный в верхнем правом углу светофор показывает результат проверки и выводится сообщение с результатом

проверки. Красный свет светофора, говорит об очень большом расхождении между сохраненным в базе и вновь введенным проверочным словом (рис. 3.14) [31, 87].

При предъявлении образа «Своего» частое загорание красного света светофора говорит о плохой узнаваемости. При предъявлении образа близкого к сохраненному в базе, когда не совпадают лишь несколько бит ключа загорается желтый свет светофора. При полном совпадении предъявленного образа с сохраненным в базе загорается зеленый свет светофора. Значит, ключ воспроизводится НС без ошибок (рис. 3.15) [87].

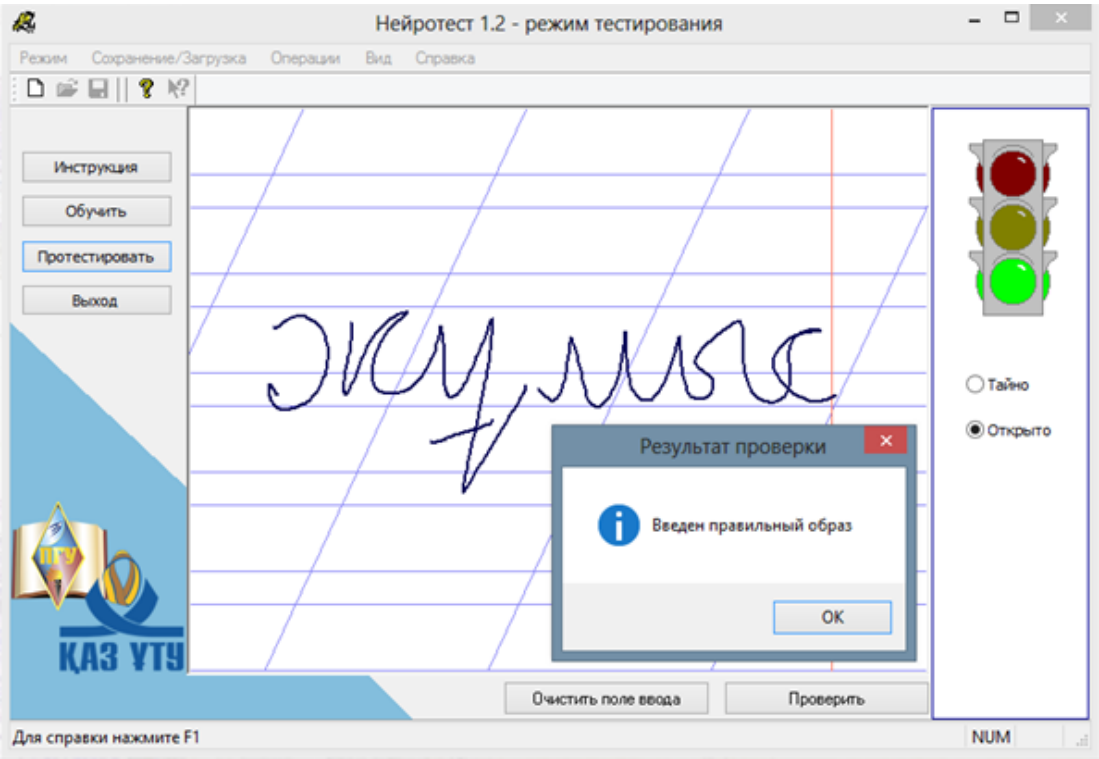


Рис. 3.15. Ввод правильного рукописного образа

Самостоятельное статистическое тестирование системы. При обучении системы программа выдает данные по ОПР и ОВР, которые являются результатом нейросетевого прогнозирования (рис. 3.8). Можно самостоятельно для своего почерка и выбранного слова-пароля оценить стойкость программы. Для этого нужно оценить вероятности ОПР и ОВР как описано ниже.

Оценка вероятности ОПР (отказ «Своему»). Необходимо вводить выбранное первоначально слово в режиме тестирования и фиксировать загорающиеся цвета светофора. Затем посчитать количество загораний светофора различных цветов, сложить количество загораний светофора красным и желтым цветами и разделить на общее число попыток [87]. Это и будет вероятностью ОПР для конкретного почерка и слова.

Оценка вероятности ОВР (пропуска «Чужого»). В режиме тестирования необходимо вводить случайные слова и нажимать на кнопку «Проверить». Для каждого введенного слова нужно фиксировать число не совпавших бит ключа. Среднее количество ошибок должно быть примерно 128. Затем нужно сделать вычисление по формуле [86, 87]:

$$\sigma = \sqrt{\frac{\sum_{i=1}^{N}(128-x_i)^2}{N}} \quad (3.3),$$

где N – число испытаний, x_i – число не совпавших бит в каждом испытании.

Результатом вычисления будет среднеквадратическое отклонение (дисперсия). Чтобы оценить стойкость можно использовать табл. 3.3 [86]. Таблица рассчитана для рукописного слова из 5 букв. Уменьшение длины слова ведет к падению стойкости системы.

Таблица 3.3 - Взаимосвязь дисперсии со стойкостью системы [87]

Σ	P_2	Σ	P_2
5-10	10^{-120}	25-30	10^{-6}
10-15	10^{-32}	30-35	10^{-4}
15-20	10^{-15}	35-45	10^{-3}
20-25	10^{-9}	45-60	10^{-2}

Автоматизированное тестирование. В программе реализовано два вида автоматизированного тестирования [88]:

– на тестовых образах – тестирование, основанное хранимых в базе тестовых примеров реальных рукописных образах. Здесь моделируется ситуация взлома защиты когда злоумышленник располагает базой образов. Этот вид тестирования дает возможность оценки стойкости системы распознавания при атаках с помощью больших баз биометрических образов.

– на белом шуме – тестирование, основанное на искусственно синтезированных образах. Здесь моделируется ситуация, при которой злоумышленник подбирает параметры рукописного пароля конкретного донора, располагая основными показателями биометрических коэффициентов. Этот вид тестирования дает возможность оценить стойкость системы к атакам компьютерного перебора.

Для тестирования на тестовых образах нужно выбрать в пункте главного меню «Операции» пункт «Тестировать на тестовых образах» и в появившемся окне указать базу тестовых образов. Результаты отразятся в всплывающем окне (рис. 3.16).

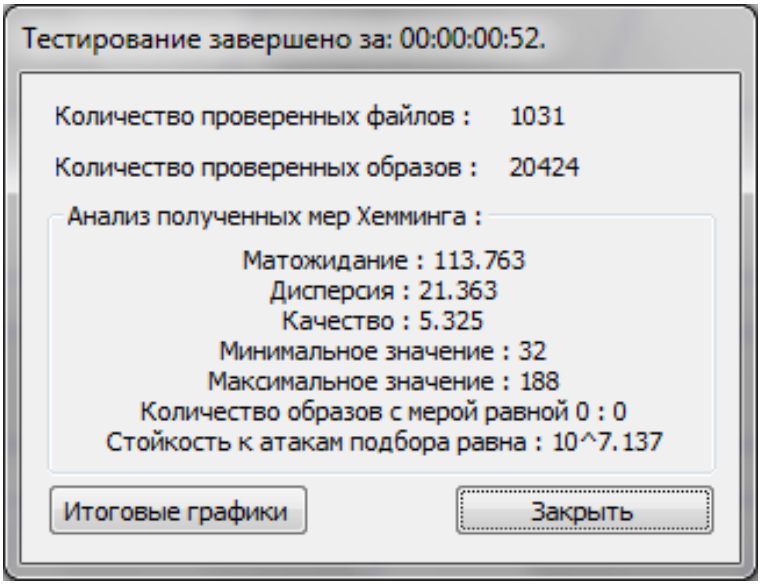


Рис. 3.16. Результат тестирования на тестовых образах

Для инициализации тестирования на белом шуме нужно выбрать в пункте главного меню «Операции» пункт «Тестировать на белом шуме». Результат тестирования представлен на рис. 3.17 [87].

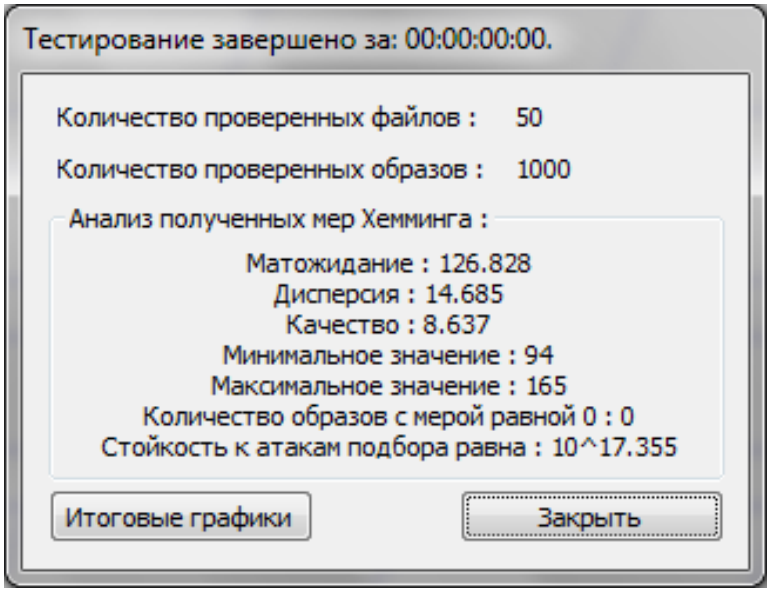


Рис. 3.17. Результат тестирования на белом шуме

Время тестирования зависит от объема базы биометрических образов. В результатах тестирований выдается анализ полученных мер Хемминга и реальная стойкость только что обученной системы. Система будет идеально обученной, если математическое ожидание меры Хемминга будет равно 128, дисперсия равна 16, а качество – 8.

Если стойкость системы ниже 10^{12} мы рекомендуем переучить систему. Для этого нужно удалить неудачные примеры или сменить используемое при обучении слово-пароль [87].

3.7. Система распознавания рисунков отпечатков пальцев

Согласно второму этапу формирования баз биометрических образов донорам биометрии необходимо выработать навыки стабильного ввода

отпечатков пальцев и освоить разработанную автоматизированную систему распознавания биометрических образов отпечатков пальцев «FINGER».

В программе используется эмулятор ИНС, разработанной в главе 2.

Главное окно программы представлено на рис. 3.18.

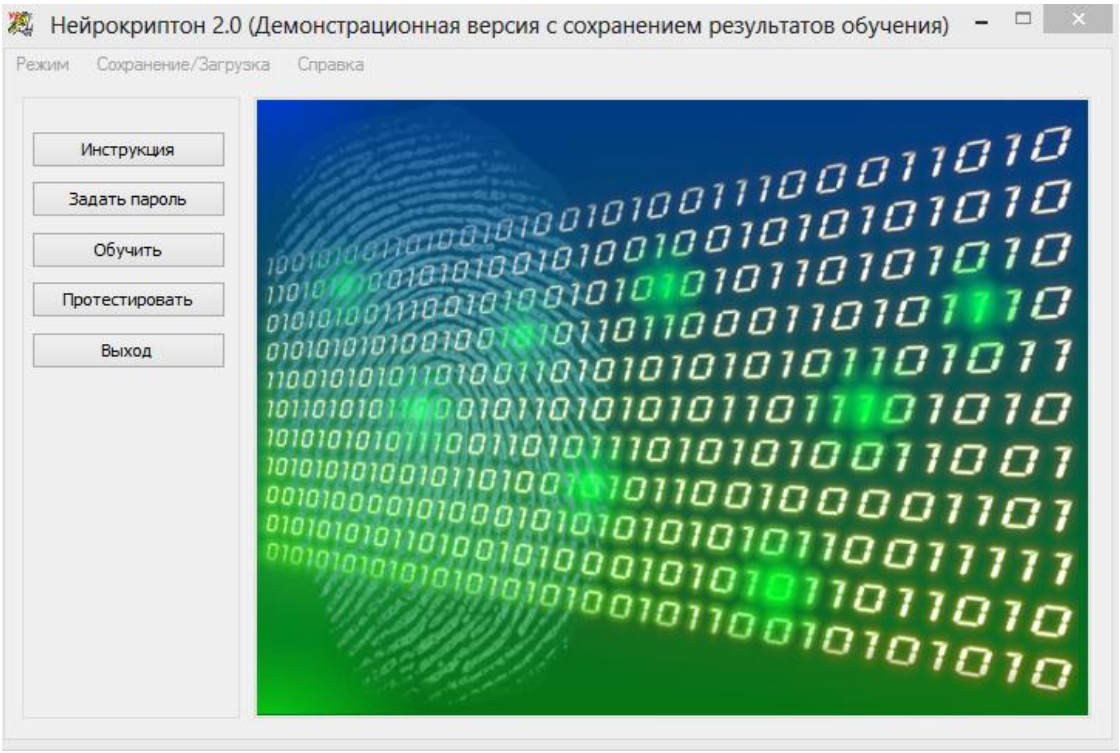


Рис. 3.18. Главное окно программы

Чтобы начать работу новый пользователь должен нажать кнопку «Задать пароль». В появившемся окне нужно внести новый пароль или нажать на кнопку «Автоматически сгенерировать новый пароль» для генерации пароля программой. При выборе пункта «Скрыть пароль» сгенерированный или введенный пароль будет показан в виде звездочек. При выборе пункта «Показать пароль» пароль отобразиться в явном виде (рис. 3.19).

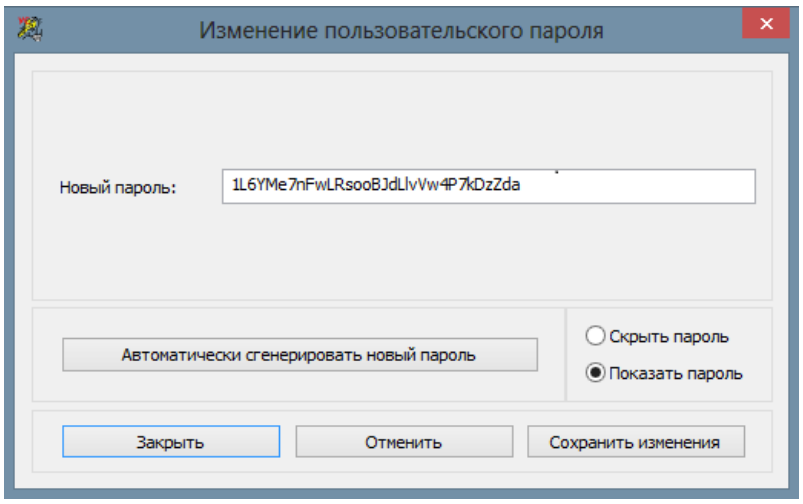


Рис. 3.19. Генерация пароля

Нажатием на кнопку «Сохранить изменения» нужно сохранить пароль нового пользователя.

Если пользователь уже работал с системой и генерировал ключ, то необходимо в пункте главного меню «Сохранение/загрузка» выбрать пункт «Загрузить ключ с диска». Откроется окно, где необходимо выбрать нужный файл и нажать кнопку «Открыть». Программа выдаст сообщение «Ваш пароль прочитан!».

Кроме загрузки ключа с диска пользователи могут сохранить ключ на диск, удалить ключ из программы, сохранить и загрузить обученную сеть.

Для обучения НС, необходимо в главном окне программы нажать на кнопку «Обучить». Откроется окно обучения НС (рис. 3.20). После нажатия на кнопку «Захват» программа предложит приложить палец к сканеру отпечатков пальцев для формирования обучающей выборки.

Для качественного ввода образа в программе предусмотрен анализ положения пальца на сканере. Для этого в окне выводятся стрелки вправо и влево для направления передвижения пальца и соответствующие сообщения. Необходимо ввести 15-20 образов.

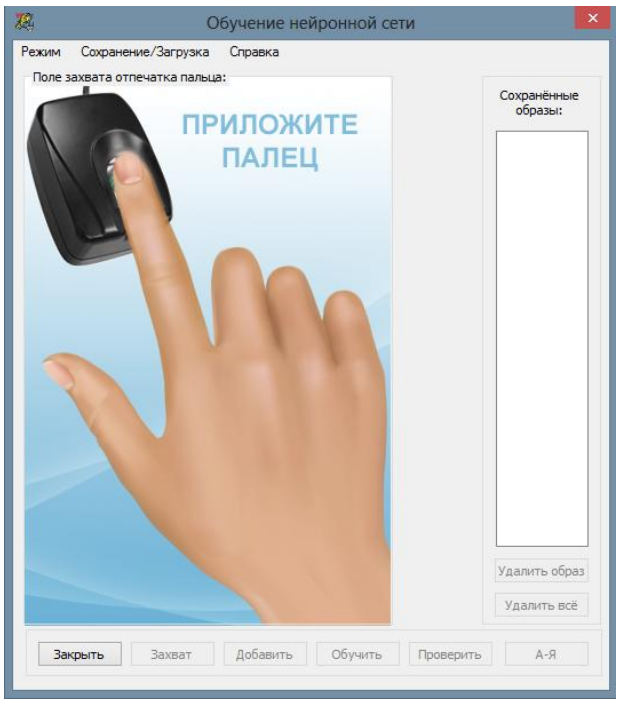


Рис. 3.20. Обучение НС

Введя отпечаток пальца для его добавления в обучающую выборку необходимо нажать на кнопку «Добавить». Номер сохраненного образа появиться в окне справа. Добавленный образ можно удалить, щелкнув на его номер и нажав на кнопку «Удалить образ». Также есть возможность удалить все образы, нажав на одноименную кнопку.

Если введено необходимое число примеров отпечатков пальцев, для обучения НС необходимо нажать на кнопку «Обучить». В результате формируется файл, содержащий весовые коэффициенты и биометрический контейнер.

3.8. Формирование обезличенной базы рисунков отпечатков пальцев

После этапа обучения и тестирования качества ввода отпечатка пальца, донор биометрии приступает к формированию обезличенной базы биометрических образов.

Система предназначена для формирования базы естественных биометрических образов папиллярных рисунков отпечатков пальцев обеспечивает выполнение следующих функций:

– обучение пользователя и контроль корректного ввода образов;

– загрузку и визуальную проверку ранее созданной базы.

ПО функционирует под управлением ОС Windows®2000/NT/XP/Vista/7 и требует следующую минимальную конфигурацию: процессор типа Intel с тактовой частотой не менее 300 МГц; ОЗУ – 128 Мбайт; объем свободного дискового пространства на жестком диске – 1,5 Гбайт; объем видеопамяти – 2 Мбайт.

Для сбора отпечатков пальцев к ПЭВМ через USB-порт необходимо подключить сканер отпечатков пальцев Futronic FS80 и установить соответствующие драйверы.

Для вызова и загрузки программы необходимо перейти в папку с программой на жестком диске и запустить файл *FingerProj.exe*. В результате выполнения программы на мониторе отобразится главное окно модуля формирования баз отпечатков (рис. 3.21).

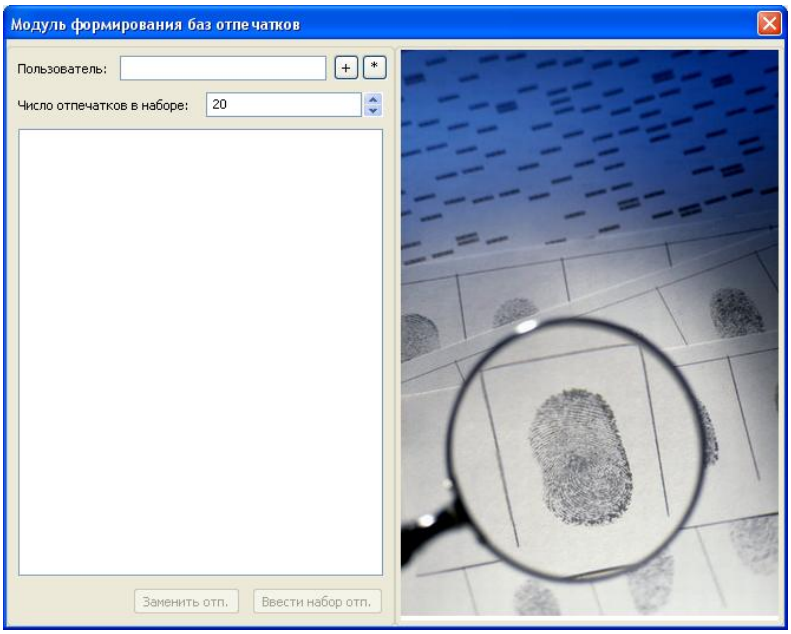


Рис. 3.21. Общий вид окна модуля формирования баз отпечатков пальцев

В левой части главного окна программы располагается область, в которую вводится информация о пользователе и его образах. В правой части содержится окно для просмотра сохраненных образов баз отпечатков.

Программа позволяет создать новую базу биометрических данных или редактировать уже существующую. Формирование базы осуществляется добавлением пользователя и захвата изображений его отпечатков пальцев со сканера. Файлы базы сохраняются в рабочей папке с программой.

Для добавления нового пользователя в поле «Пользователь» укажите «Идентификационный номер» пользователя.

В поле «Число образов» введите количество вводимых образов одного пальца. По умолчанию поле «Число образов» принимает значение 20. Минимальное число вводимых образов одного пальца равно 8. Максимальное число вводимых образов одного пальца равно 40.

После нажатия «+» новый пользователь добавляется в дерево каталогов. Дерево каталогов служит для повышения навигации пользователей в среде и содержит списки образов пальцев пользователей, которые необходимо ввести или заменить (рис. 3.22).

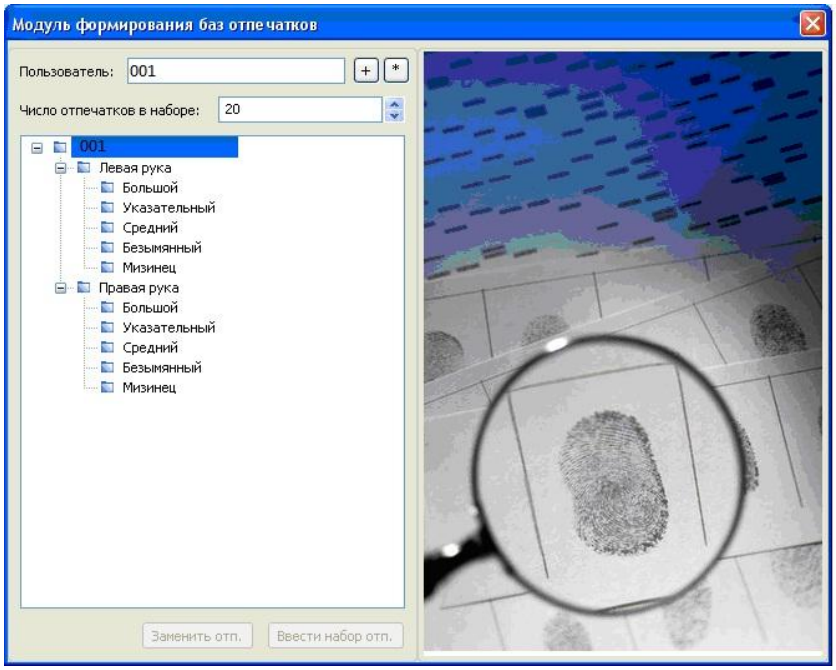


Рис. 3.22. Общий вид окна регистрации нового пользователя

После добавление нового пользователя в дерево каталогов баз отпечатков выберите подкаталог «Левая рука» или подкаталог «Правая рука». Укажите подкаталог соответствующий названию пальца выбранной руки. При нажатии «Ввести набор отп.» запустится процедура захвата набора образов выбранного пальца (рис. 3.23).

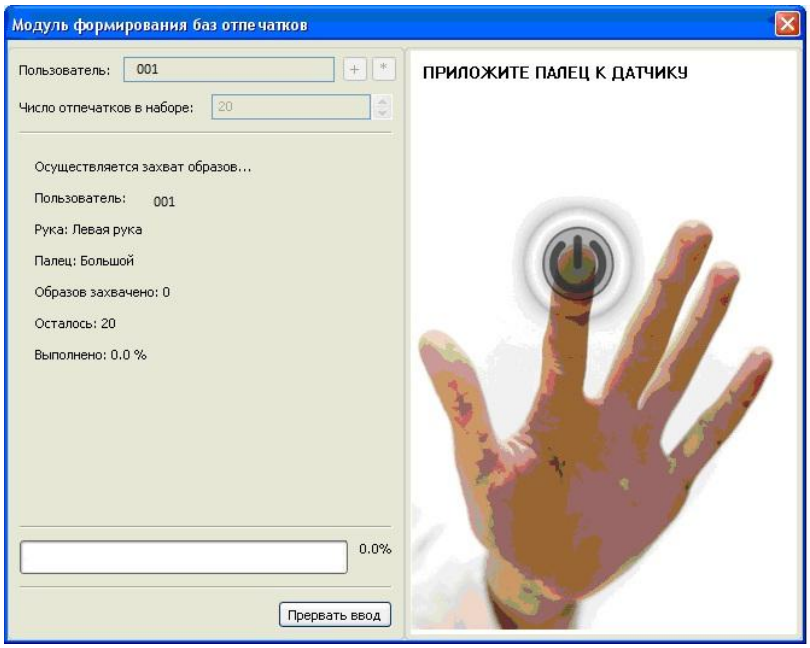


Рис. 3.23. Общий вид окна снятия отпечатка пальцев

В левой части окна программы располагается область, в которой выводится информация о пользователе и состояние ввода его образов. Индикатор прогресса показывает ход процесса формирования базы. В правой части окна программы находится область вывода захваченного изображения со сканера отпечатков пальцев, а также вывод графических сообщений. Сообщения уведомляют пользователя о необходимости корректного расположения пальца на датчике.

После появления сообщения «ПРИЛОЖИТЕ ПАЛЕЦ К ДАТЧИКУ» необходимо расположить палец на сканере и прижать его с небольшим усилием. Захваченное изображение отпечатка должно находиться во внутренней рамке и преимущественно по центру. Если захват отпечатка пальца не происходит, то следует:

– прижать сильнее палец при появлении сообщения «ПРИЛОЖИТЕ ПАЛЕЦ СИЛЬНЕЕ»;

– сместить расположение пальца при появлении сообщений «СМЕСТИТЕ ПАЛЕЦ ВЛЕВО» или «СМЕСТИТЕ ПАЛЕЦ ВПРАВО» в направлении указанной стрелки (рис. 3.24);

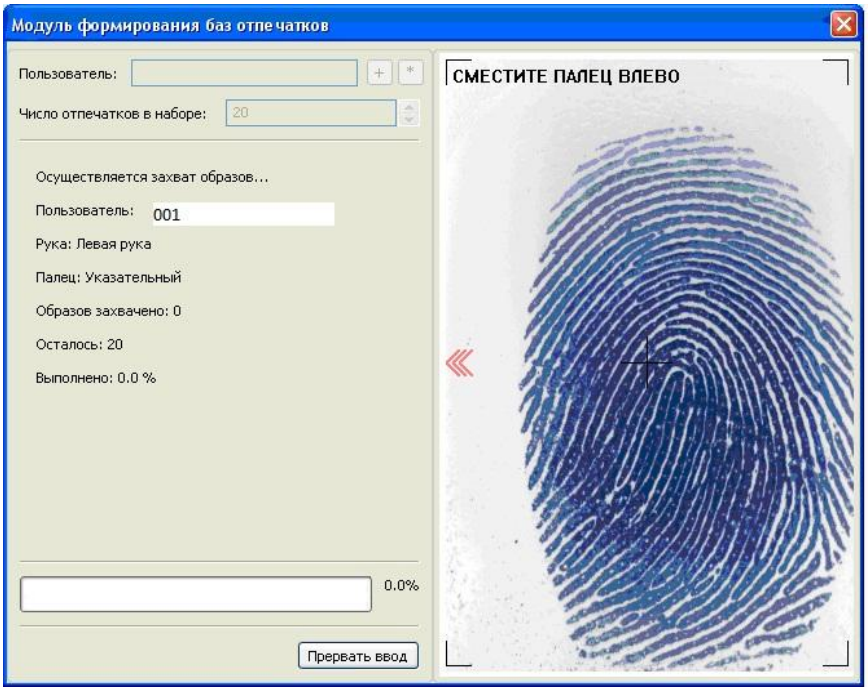


Рис. 3.24. Ввод отпечатка пальца

– повернуть палец вертикально при появлении сообщения «ПОВЕРНИТЕ ПАЛЕЦ ВЕРТИКАЛЬНО», так чтобы края отпечатка были условно параллельны рамке окна и симметричны относительно друг друга (рис. 3.25).

Для меньшей деформации отпечатка пальца при перемещении рекомендуется его последовательно отрывать и прикладывать к сканеру. Перекатывание сильно деформирует рисунок папиллярных линий, что может повлиять на качество базы.

Рис. 3.25. Корректировка ввода отпечатка пальца

Если местонахождение пальца на сканере удовлетворяет требованиям расположения, образ будет захвачен. При завершении сканирования установленного числа образов отпечатка пальца появится информационное сообщение «Отпечатки сохранены».

Список захваченных образов отпечатков пальцев будет сформирован в подкаталоге соответствующего пальца. Для создания полной базы биометрических данных сканирование необходимо провести для каждого пальца пользователя.

При изменении (обновлении или замене образов) существующих баз их следует загрузить, нажав на символ «*». После нажатия появится предупреждающее сообщение (рис. 3.26).

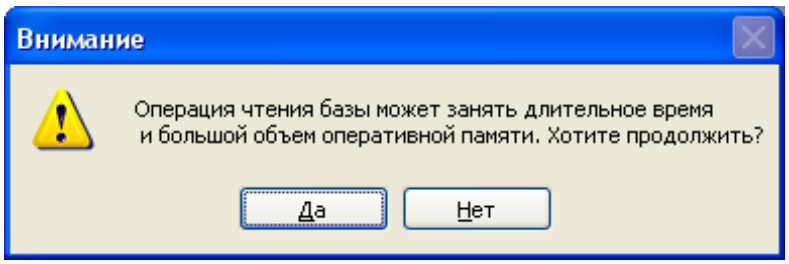


Рис. 3.26. Предупреждающее сообщение

Чтение большой базы требует значительного времени и ресурсов ПЭВМ, поэтому операцию можно отменить, нажав «Нет». При нажатии «Да» запуститься процедура загрузки базы.

После окончания загрузки дерево каталогов заполнится ранее созданными базами пользователей и содержащимися в них записями (образы). Для просмотра образа отпечатка пальца следует выбрать каталог пользователя и соответствующий подкаталог пальца руки (рис. 3.27).

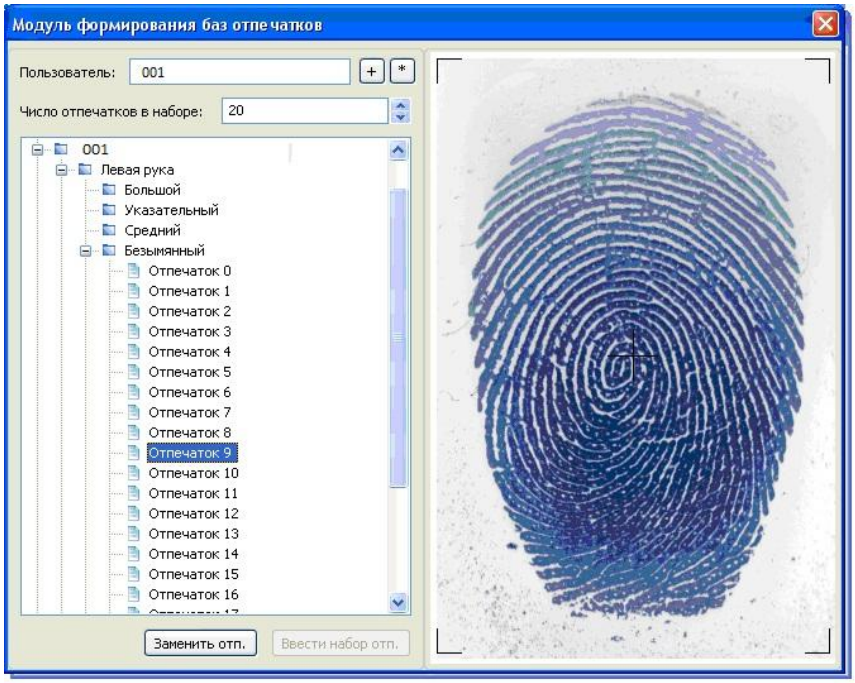


Рис. 3.27. Чтение базы отпечатков пальцев

При нажатии «Ввести набор отп.» запустится процедура захвата набора образов выбранного пальца.

Для замены образа отпечатка пальца следует выбрать в дереве каталогов необходимый палец и нажать «Заменить отп.». После чего запуститься процедура захвата соответствующего образа.

3.9. Тестирование системы распознавания биометрических образов с использованием сформированных баз рисунков отпечатков пальцев

В программе «FINGER» предусмотрена возможность тестирования результатов обучения. Для инициализации режима тестирования необходимо нажать на кнопку «Протестировать» в главном окне программы. В результате загрузиться окно тестирования обученной НС (рис. 3.28).

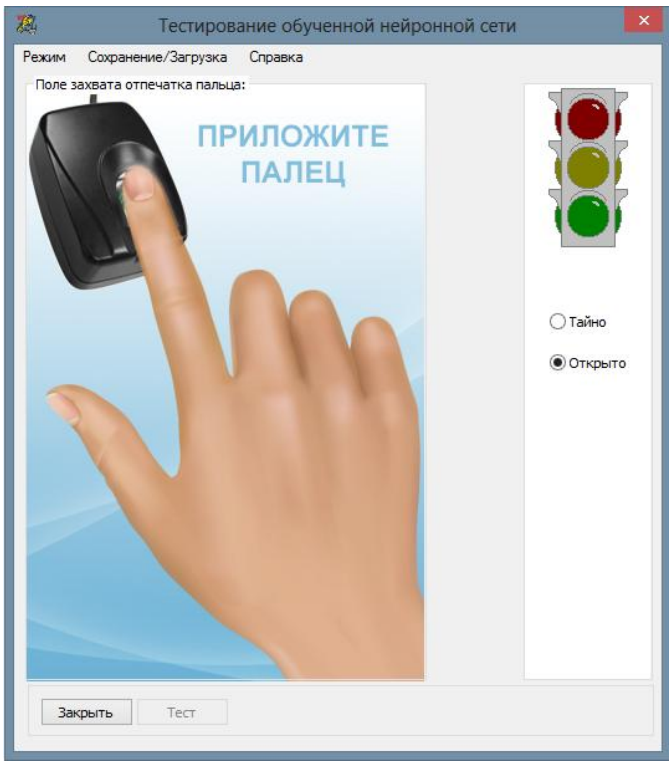


Рис. 3.28. Инициализация тестирования

Для проведения тестирования пользователю нужно ввести отпечаток пальца, который ранее использовался для обучения НС, и нажать на кнопку «Тест». При предъявления биометрического образа «Свой» на светофоре в правом верхнем углу окна загорится зеленый свет и выйдет окно с результатом тестирования. Результатом будет сгенерированный НС двоичный ключ и сообщение о правильности введенного пароля (см. рис. 3.29). Для просмотра

ключа в шестнадцатеричной кодировке необходимо нажать на кнопку «Символьное представление».

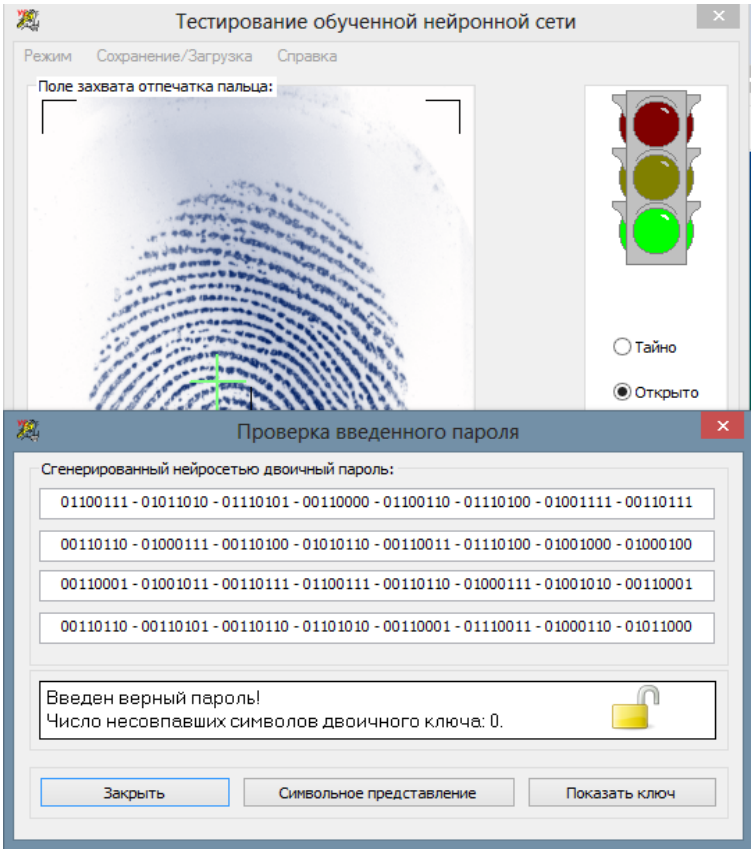


Рис. 3.29. Предъявление биометрического образа «Свой»

При предъявлении биометрического образа «Чужой» на светофоре загорится красный свет и выйдет окно с результатом тестирования. Результатом будет сгенерированный НС двоичный ключ, где не совпавшие символы будут отмечены звездочкой, сообщение о неправильности введенного пароля и число не совпавших символов двоичного ключа (см. рис. 3.30).

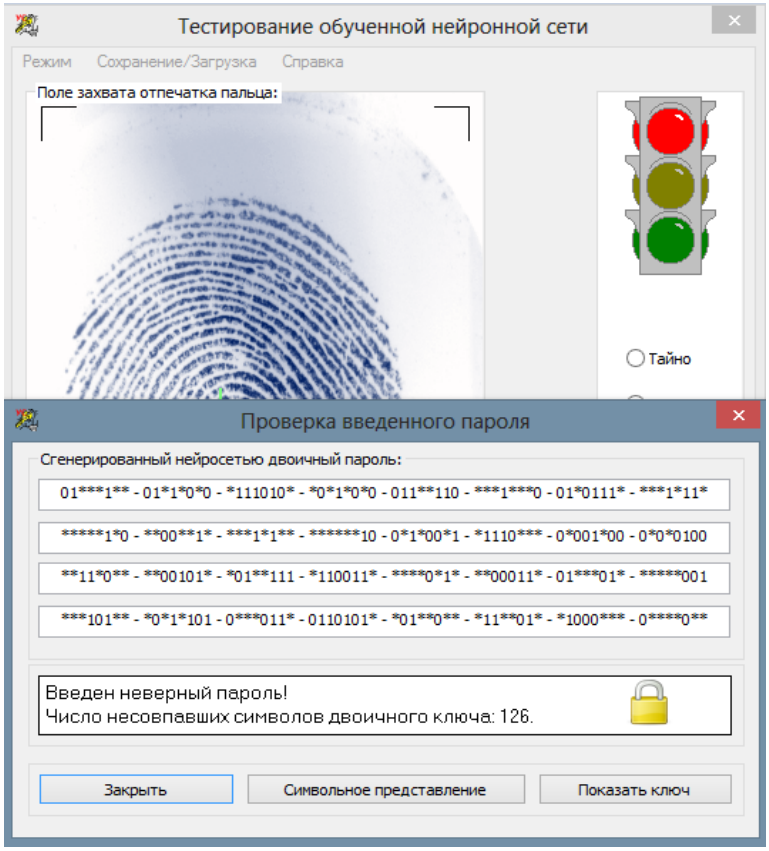


Рис. 3.30. Предъявление биометрического образа «Чужой»

3.10. Сравнительный анализ разработанной системы распознавания образов с аналогами

Для оценки качества разработанной системы распознавания образов по рукописному почерку были рассмотрены и проанализированы системы Mnist_brain-master, Нейротест 1.1, Нейротест 1.2.

Для анализа использовались программные средства, приведенные в таблице 3.4.

Таблица 3.4 – Основные характеристики программных средств практического исследования

Названия	Место распространения	Назначение
MNIST	Свободное распространение	Предназначен для распознавания биометрических образов по

	(http://yann.lecun.com/exdb/mnist/)	особенностям рукописного почерка на основе геометрического анализа. Источник данных необходимый для обучения нейросетевой модели.
Tensorflow	Свободное распространение (https://www.tensorflow.org/ install)	Программные библиотеки для нейросетевых моделей на основе сверточных НС.
Mnist_brain-master	Свободное распространение https://github.com/ApelSYN/mnist_brain	Предназначен для распознавания введенных рукописных биометрических образов по графическим показателям. Источник необходимый для нейросетевых моделей на основе сверточных НС.
NeuroPro	Свободное распространение (http://www.neuropro.ru)	Использование нейросетевых моделей в виде двухслойного персептрона

Mnist_brain-master – система, реализованная на нейросетевой модели на основе сверточных НС. Позволяет распознавать рукописный текст не более 5 символов.

Нейротест 1.1 – система, реализованная нами на нейросетевой модели на основе двухслойного персептрона. Рекомендует выбирать рукописный текст от 3 до 5 символов.

Нейротест 1.2 – система, реализованная нами на нейросетевой модели на CNN-LSTM. Ориентирован на рукописный текст переменной длины.

Сравнительный анализ проводился по скорости тестирования и стойкости (величине ОВР) и приведен в таблице 3.5.

Таблица 3.5 - Сравнительный анализ систем распознавания образов

Количество образов	Mnist_brain-master		Нейротест 1.1		Нейротест 1.2	
	Скорость тестирования (сек)	Точность (P_2)	Скорость тестирования (сек)	Точность (P_2)	Скорость тестирования (сек)	Точность (P_2)
1000	0,0	10^{-13}	0,0	10^{-17}	0,0	10^{-17}
10000	0,58	10^{-13}	0,36	10^{-15}	0,24	10^{-16}
25000	3,13	10^{-12}	1,27	10^{-15}	0,59	10^{-15}
50000	13,06	10^{-10}	5,19	10^{-13}	3,02	10^{-14}
100000	37,41	10^{-8}	12,53	10^{-11}	7,46	10^{-12}

Сравнительный анализ показал, что разработанная нами система выигрывает по времени тестирования и по величине ОВР. Все три системы имеют высокую масштабируемость, так как имеют высокую скорость распознавания.

3.11. Выводы к главе 3

1. Дается классификация биометрических образов пользователей по их стабильности, уникальности биометрических параметров.

2. Дается классификация баз естественных биометрических образов по области их применения.

3. Сформированы требования к техническому и программному обеспечению автоматизированного формирования баз биометрических тестовых образов.

4. Разработаны требования к формированию баз естественных биометрических образов «Свой» и «Чужой», предназначенных для тестирования средств биометрической аутентификации. Обосновывается допустимость использования неполных баз естественных биометрических образов «Чужой».

5. На основе сформированных требований разработаны методики формирования биометрической базы естественных рукописных образов и папиллярных рисунков отпечатков пальцев, в соответствии с которыми определен порядок формирования биометрических баз.

6. Разработана архитектура интеллектуальной автоматизированной системы распознавания образов.

7. Описан специализированный программно-аппаратный комплекс «Нейро-Тест 1.2», предназначенный для обучения и тестирования НС по рукописному слову-паролю.

8. Сформирована тестовая рукописная база образов с использованием разработанного программного модуля «Нейрокриптон – формирователь биометрических баз» с прилагаемыми к нему словарями слов.

9. Проведено тестирование системы распознавания биометрических образов с использованием сформированных баз рукописных почерков. Результаты экспериментов показали величину ОВР в пределах 10^{-15}.

10. Описан специализированный программно-аппаратный модуль, предназначенный для проведения процедур обучения и тестирования по рисункам отпечатков пальцев.

11. Сформирована тестовая база папиллярных рисунков отпечатков пальцев с использованием специального программно-аппаратного модуля.

12. Проведено тестирование системы распознавания биометрических образов с использованием сформированных баз папиллярных рисунков отпечатков пальцев. Результаты экспериментов показали величину ОВР в пределах 10^{-4}.

ВЫВОДЫ

1. Рассмотрены методы распознавания атак на ресурсы информационных систем и методы распознавания биометрических образов для защиты информационных ресурсов от несанкционированного доступа. Показано, что для их эффективной реализации наиболее перспективным является использование возможностей НС.

2. Дан сравнительный анализ биометрических технологий с точки зрения стоимости реализации, стойкости защиты и информативности образа.

3. Разработана композитная нейросетевая модель, которая за счет использования в сверточной НС модулей долгой краткосрочной памяти, а также за счет адаптации параметров модели к условиям системы биометрической аутентификации, позволяет с точностью 10^{-15} реализовать распознавание пользователей на основе анализа геометрических параметров фрагментов рукописного текста изменяемого размера.

4. Излагаются общие принципы обучения нейросетевых преобразователей на примере вырожденного НСП биометрия-код с одним выходом. Показано, что абсолютно устойчивый не итерационный алгоритм обучения является более эффективным.

5. Показано, что учитывая корреляционные связи между выходными сигналами НС можно уменьшить размерность входной выборки.

6. Предлагается метод синтеза критерия хи-квадрат распределений зависимых данных.

7. На основе сформированных требований разработаны методики формирования биометрической базы естественных рукописных образов и папиллярных рисунков отпечатков пальцев, в соответствии с которыми определен порядок формирования биометрических баз.

8. Описан специализированный программно-аппаратный комплекс «Нейро-Тест 1.2», предназначенный для обучения и тестирования НС по рукописному слову-паролю.

9. Сформирована тестовая рукописная база образов с использованием разработанного программного модуля «Нейрокриптон – формирователь биометрических баз» с прилагаемыми к нему словарями слов.

10. Проведено тестирование системы распознавания биометрических образов с использованием сформированных баз рукописных почерков. Результаты экспериментов показали величину ОВР в пределах 10^{-15}.

11. Описан специализированный программно-аппаратный модуль, предназначенный для проведения процедур обучения и тестирования по рисункам отпечатков пальцев.

12. Сформирована тестовая база папиллярных рисунков отпечатков пальцев с использованием специального программно-аппаратного модуля.

13. Проведено тестирование системы распознавания биометрических образов с использованием сформированных баз папиллярных рисунков отпечатков пальцев. Результаты экспериментов показали величину ОВР в пределах 10^{-4}.

ПРАКТИЧЕСКИЕ РЕКОМЕНДАЦИИ

Результаты диссертации использовались:

− предложенный метод снижения входной выборки НС, разработанная методика формирования баз рукописных образов и отпечатков пальцев, программно-аппаратные комплексы использовались в Научно-исследовательском центре «Тезис» КПИ им. И. Сикорского (Украина, Киев) для обеспечения и разграничения доступа к ресурсам ИС, что подтверждается актом внедрения (акт внедрения от 11.09.2017 в приложении);

− разработанная на основе композитной нейросетевой модели интеллектуальная автоматизированная система распознавания рукописного почерка использовалась в качестве средства биометрической аутентификации личности для защиты ресурсов ИС в ТОО «QUARES» (Алматы) и показала достаточно высокую эффективность распознавания биометрических образов, что подтверждается актом внедрения (акт внедрения от 09.10.2017 в приложении);

− разработанная интеллектуальная автоматизированная система распознавания образов, реализующая предложенные модели и методы, внедрены в учебный процесс на кафедре «Информационная безопасность» КазНИТУ имени К.И. Сатпаева (Алматы, Казахстан) (акт внедрения от 20.12.2016 в приложении) и на кафедре «Безопасность информационных технологий» Национального авиационного университета (Киев, Украина) (акт внедрения от 24.07.2017 в приложении).

Результаты диссертации могут использоваться:

- для распознавания других биометрических образов как статических, так и динамических;

- для внедрения в готовые системы, как модуль высоконадежной биометрико-нейросетевой аутентификации.

На разработанную интеллектуальную автоматизированную систему распознавания рукописных образов «Нейротест 1.2» было получено авторское свидетельство № 1240 от 08.01.2019 года.

На разработанный программно-аппаратный комплекс биометрико-нейросетевой аутентификации личности по отпечаткам пальцев FINGER было получено авторское свидетельство № 1287 от 11.01.2019 года.

Материалы исследований были использованы в учебнике «Қолданбалы криптология: шифрлау әдістері», рекомендованным Министерством образования и науки РК (приложение).

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

1. **Болл, Р. М.** Руководство по биометрии [Текст] / [Р. М. Болл, Дж. Х. Коннел, Ш. Панканти и др.]. – М.: Техносфера, 2007. – 368 с.

2. **Алимсеитова, Ж.** Технологии распознавания образов с использованием биометрии личности [Текст] / Ж. Алимсеитова, К. Дж. Боскебеев // Известия Кыргызского государственного технического университета им. И. Раззакова. – 2017. – № 1(41) часть 2. – С. 11–17.

3. Теория нейронных сетей [Электронный ресурс]. – Режим доступа: evloevoleg.narod.ru/Neural/Par_2.pdf

4. **Вороненко, Д. И.** Нейросети – за и против [Текст] / Д. И. Вороненко. – Харьков, 2004.

5. **Данько, Т. П.** Системы искусственного интеллекта в разработке корпоративных маркетинговых стратегий [Текст] / Т. П. Данько, М. А. Ходимчук // Маркетинг в России и за рубежом. – 2000. – № 5. – С. 26-36.

6. **Lakhno, V.** Development of a decision support system based on expert evaluation for the Situation Center of Transport Cybersecurity [Текст] / V. Lakhno, B. Akhmetov, A. Korchenko, Z. Alimseitova, V. Grebenuk // Journal of Theoretical and Applied Information Technology. – 2018. – Vol. 96. – № 14. – P. 4530-4540.

7. **B. Akhmetov.** Development of Sectoral Intellectualized Expert Systems and Decision Making Support Systems in Cybersecurity [Текст] / B. Akhmetov, V. Lakhno, B. Akhmetov, Z. Alimseitova // Intelligent Systems in Cybernetics and Automation Control Theory. CoMeSySo 2018. Advances in Intelligent Systems and Computing. – Vol 860. – P.162-171.

8. **Новак, Дж.** Как обнаружить вторжение в сеть. Настольная книга специалиста по системному анализу = Network Intrusion Detection. An Analyst's Handbook [Текст] / Дж. Новак, С. Норткатт, Д. Маклахен, перевод И. Дранишникова. – М.: Лори, 2012. – 384 с.

9. **Russell, J.** Intrusion detection system [Текст] / J. Russell, R. Cohn. – Stoughton, WI, USA: Book on Demand Ltd., 2012. – 158 p. – ISBN 9785512319819.

10. **Лукацкий, А.** Системы обнаружения атак. Взгляд изнутри [Электронный ресурс]. – М.: Техносфера, 1999. – Режим доступа: World Wide Web. – URL: http:// www.electronics.ru/journal/article/1714. – Загл. с экрана.

11. **Шаньгин, В. Ф.** Защита информации в компьютерных системах и сетях [Текст] / В. Ф. Шаньгин. – М.: ДМК–Пресс, 2012. – 592 с.

12. **Callegari, C.** A new statistical approach to network anomaly detection [Текст] / C. Callegari, S. Vaton, M. Pagano // Proc. of Performance Evaluation of Computer and Tele–communication Systems (SPECTS). – 2008. – P. 441–447.

13. **Ахметов, Б.** Система выявления аномального состояния в информационных системах [Текст] / Б. Ахметов, А. Корченко, Ж. Алимсеитова, Н. Жумангалиева // Доклады НАН РК. – 2017. –№5. – С. 28–37.

14. **Callegari, C.** Application of wavelet packet transform to network anomaly detection [Текст] / C. Callegari, S. Giordano, M. Pagano // Proc. of Int. Conf. on Next Generation Teletraffic and Wired/Wireless Advanced Networking (NEW2AN). – 2008. – P. 246–257.

15. **Котов, В. Д.** Современное состояние проблемы обнаружения сетевых вторжений [Текст] / В. Д. Котов, В. И. Васильев // Вестник УГАТУ. – 2012. – Т. 16. – №3(48). – С. 198–204.

16. **Ахметов, Б. С.** Основы биометрической аутентификации личности. [Текст]: *у*чебное пособие / Б. С. Ахметов, А. И. Иванов, В. А. Фунтиков, А. Ю. Малыгин. – Алматы: КазНТУ им. К.И. Сатпаева, 2014. – 151 с.

17. Исследование вариантов реализации и разработка действующего лабораторного образца ON–LINE системы биометрического обезличивания электронных историй болезней для медицинского учреждения [Текст]: отчет НИРС 0868/ГФ3. – Алматы, 2014. – 65 с.

18. **Dodis, Y.** Fuzzy Extractors: How to Generate Strong Keys from Biometrics and Other Noisy Data [Текст] / Y. Dodis, L. Reyzin, A. Smith // EUROCRYPT 2004: Advances in Cryptology – EUROCRYPT, 2004. – P. 523-540.

19. **Sahai, A.** Fuzzy identity–based encryption [Текст] / A. Sahai, B. Waters // Proceedings of EUROCRYPT 2005, LNCS 3494. Springer – Verlag, 2005. – P. 457–473.

20. **Baek, J.** New construction of fuzzy identity–based encryption [Текст] / J. Baek, W. Susilo, J. Zhou // Proceedings of the 2nd ACM Symposium on Information, Computer and Communications Security. – New York, USA, 2007. – P. 368–370.

21. **Dodis, Y.** Robust fuzzy extractors and authenticated key agreement from close secrets [Текст] / Y. Dodis, J. Katz, L. Reyzin, A. Smith // Advances in Cryptology – CRYPTO 2006. Cynthia Dwork, ed. Vol. 4117 of Lecture Notes in Computer Science. Springer – Verlag, 20 – 24 August 2006. – P. 232–250.

22. **Kanukurthi, B.** Key Agreement from Close Secrets over Unsecured Channels [Текст] / B. Kanukurthi, L.Reyzin // EUROCRYPT 2009: Advances in Cryptology – EUROCRYPT, 2009. – P. 206–223.

23. **Boyen, X.** Reusable cryptographic fuzzy extractors [Текст] / X. Boyen // Eleventh ACM Conference on Computer and Communication Security. ACM, 2004. – P. 82–91.

24. **Boyen X.** Secure remote authentication using biometric data [Текст] / X. Boyen, Y. Dodis, J. Katz, R. Ostrovsky, A. Smith // Advances in Cryptology – EUROCRYPT 2005. R. Cramer, ed. Springer – Verlag, 2005. – Vol. 3494 of LNCS. – P. 147–163.

25. **Боскебеев, К. Дж.** Информационная система обработки цифровой информации для идентификации объекта [Текст] / К. Дж. Боскебеев, Ж. К. Алимсеитова // Известия Кыргызского государственного технического университета им. И. Раззакова. – 2016. – № 1(37). – С. 8–12.

26. **Алимсеитова, Ж.** Анализ использования технологий распознавания биометрических образов [Текст] / Ж. Алимсеитова, Ж. З. Акматалиева, К. Дж. Боскебеев // Известия Кыргызского государственного технического университета им. И. Раззакова – 2017. – № 2(42). – С. 14–19.

27. **Daugman, J.** Probing the Uniqueness and Randomness of Iris Codes: Results From 200 Billion Iris Pair Comparisons [Текст] / J. Daugman // Proceedings of the IEEE. – 2006. – Vol.94, №11. – P. 1927-1935.

28. Описание изобретения к патенту RU 2316120 C2 Российская федерация, МПК H04L 9/28, H04L 9/32, G06F 12/14. Биометрическая система аутентификации [Текст] / А. Л. Чморра, А. В. Уривский; Опубл. 27.01.2008, Бюл. № 3. Патентообладатель «Самсунг Электроникс» (KR).

29. **Ушмаев, О. С.** Получение устойчивого криптографического ключа из биометрической характеристики изображения отпечатков пальцев [Текст] / О. С. Ушмаев, В. В. Кузнецов // The 22nd International Conference on Computer Graphics and Vision. –Russia, Moscow, 2012. – P. 125–127.

30. **Морелос–Сарагоса, Р.** Искусство помехоустойчивого кодирования [Текст] / Р. Морелос–Сарагоса. – М.: Техносфера, 2007. – 320 с.

31. **Досжанова, А. А.** Модели и алгоритмы обработки информации в биометрико–нейросетевых системах обезличивания электронных историй болезней [Текст]: дис. Ph.D доктор: 6D070400 / А. А. Досжанова – Алматы, 2014. – 115 с.

32. **Ахметов, Б. С.** Методика формирования баз биометрических образов [Текст] / Б. С. Ахметов, Н. А. Сейлова, Ж. К. Алимсеитова, А. Балтабай // Сборник материалов Всероссийской научно–практической конференции «Информационно–телекоммуникационные системы и технологии». – Кемерово, 2015. – С. 1–3.

33. **ГОСТ Р 52633.4–2011**. Защита информации. Техника защиты информации. Интерфейсы взаимодействия с нейросетевыми преобразователями биометрия–код [Текст]. – Введ. 2011-12-01. – М.: Стандартинформ, 2012. – 46с.

34. **Корченко, А.** Нейросетевые модели, методы и средства оценки параметров безопасности интернет–ориентированных информационных систем. [Текст]: монография / А. Корченко, И. Терейковский, Н. Карпинский, С. Тынымбаев. – Киев: НАУ, 2016. – 276 с.

35. **Терейковский, И. А.** Нейросетевое распознавание рукописных символов в системе биометрической аутентификации [Текст] / И. А. Терейковский, Л. А. Терейковская, А. О. Корченко, Ж. М. Алибиева // Інформаційні технології в економіці і природокористуванні. – 2017. –№ 2. – С. 29–42.

36. **Ахметов, Б.** Определение оптимального типа нейросетевой модели для биометрической аутентификации [Текст] / Б. Ахметов, Л. Терейковская, И. Терейковский, Ж. Алимсеитова // Сборник трудов IV Международной научно–практической конференции «Интеллектуальные информационные и коммуникационные технологии – средство осуществления третьей индустриальной революции в свете Стратегии «Казахстан–2050»» посвященной 70–летию профессора М. Бейсенби. – Астана, 2017 – С. 155–157.

37. **Будыльский, Д. В.** GRU и LSTM: современные рекуррентные нейронные сети [Текст] / Д. В. Будыльский // Young Scientist. – 2015. – №15 (95). – С. 51–53.

38. **Graves, A.** A novel connectionist system for unconstrained handwriting recognition [Текст] / A. Graves [et al.] // Pattern Analysis and Machine Intelligence, IEEE Transactions on. — 2009. — Vol. 31, no. 5. — P. 855–868.

39. **Doetsch, P.** Fast and robust training of recurrent neural networks for offline handwriting recognition [Текст] / P. Doetsch, M. Kozielski, H. Ney // 14th International Conference Frontiers in Handwriting Recognition (ICFHR), 2014. – P. 279–284.

40. **Pham, V.** Dropout improves recurrent neural networks for handwriting recognition [Текст] / V. Pham [et al.] // 14th International Conference Frontiers in Handwriting Recognition (ICFHR), 2014 – P. 285–290.

41. **Ахметов, Б. С.** Синхронизация процедур нейросетевого обучения особенностям рукописного почерка по точкам смены направления движения пера [Текст] / Б. С. Ахметов, А. В. Елфимов, А. И. Иванов, А. Ю. Малыгин // Труды II Международной научно-практической конференции

«Информационно-инновационные технологии: интеграция науки, образования». – Алматы, 2011. – 2 том. – С. 118-123

42. **Волчихин, В. И.** Быстрые алгоритмы обучения нейросетевых механизмов биометрико-криптографической защиты информации [Текст] / В. И. Волчихин, А. И. Иванов, В. А. Фунтиков. – Пенза: Изд-во Пенз. гос. ун-та, 2005. – 276с.

43. **Волчихин, В. И.** Основы обучения искусственных нейронных сетей [Текст]: учебное пособие / В. И. Волчихин, А. И. Иванов. – Пенза: Изд-во Пенз. гос. ун-та, 2004. – 116 с.

44. **Оганезов, А. Л.** Применение нейронных сетей в задачах распознавания образов. [Текст]: дисс. кан-та физ.-мат. наук: 05.13.11 / А. Л. Оганезов. – Тбилиси. 2006. – 149 с.

45. **Keun-Rong, Hsieh** A Neural Network Model which Combines Unsupervised and Supervised Learning [Текст] / Keun-Rong Hsieh, Wen-Tsuen Chen // IEEE Trans. on Neural Networks. –1993. – vol.4, No.2. – P. 569-574.

46. **Короткий, С.** Нейронные сети: обучение без учителя. [Электронный ресурс]. 1996. Режим доступа: http://www.gotai.net/documents/doc-nn-004.aspx.

47. **Nilsson, N.** Introduction to Machine Learning *[Текст]* / N. Nilsson // Unpublished draft, Stanford University. –1996. – P. 39-68.

48. **Riedmiller, M.** RPROP - a fast adaptive learning algorithms. *[Текст]* / M. Riedmiller, H. Brawn // Technacal Report – Karlsruhe: University Karlsruhe, 1992.

49. **Федорова, Н. Н.** Параллельная реализация алгоритмов обучения нейронных сетей прямого распространения с использованием стандарта MPI. [Текст] / Н. Н. Федорова, С. А. Терехов // Режим доступа: http://www.aconts.com/pub/archive/ijcnn99_p423_rus.pdf.

50. **Ахметов, Б. С.** Алгоритмы тестирования биометрико–нейросетевых механизмов защиты информации [Текст]: монография / Б. С. Ахметов, В. И. Волчихин, А. И. Иванов, А. Ю. Малыгин. – Алматы: Издательство LEM, 2013. – 152 с.

51. **Akhmetov, B. S.** Training of neural network biometry–code converters *[Текст]* / B. S. Akhmetov, A. I. Ivanov, Zh. K. Alimseitova // Известия НАН РК. Серия геология и технические науки. – 2018. –№1. – С. 61–68.

52. **Ахметов, Б. С.** Энтропийно-корреляционный подход к расчету вероятности совместного появления большого числа зависимых событий. [Текст] / Б. С. Ахметов, А. И. Иванов, Т. С. Картбаев, Д. Н. Надеев, А. Ю. Малыгин, И. В. Огнев // Вестник КБТУ. – 2013. – №2(25). – С. 54-58.

53. **Malygin, A.** Application of artificial neural networks forhandwritten biometric images recognition [Текст] / A. Malygin, N. Seilova, K. Boskebeev, Zh. Alimseitova // Journal Computer modeling and Technologies. – 2017. – volume 21, №1. – P. 31–38.

54. **Ахметов, Б.** Применение искусственных нейронных сетей для распознавания биометрических образов [Текст] / Б. Ахметов, Н. Сейлова, К. Боскебеев, Ж. Алимсеитова // Вестник НАН РК. – 2017. –№6. – С. 75–84.

55. **ГОСТ Р 52633.5–2011.** Защита информации. Техника защиты информации. Автоматическое обучение нейросетевых преобразователей биометрия–код доступа [Текст]. – Введ. 2011-12-01. – М.: Стандартинформ, 2012. – 20с.

56. **Алимсеитова, Ж. К.** Система распознавания биометрических образов [Текст] / Ж. К. Алимсеитова, К. Боскебеев // Тези доповідей участників iv міжнародної науково–практичної конференції «Актуальні питання забезпечення кібербезпеки та захисту інформації». – Киев, 2018. – С. 23–24.

57. **Алимсеитова, Ж. К.** Идентификация личности по рукописному почерку [Текст] / Ж. К. Алимсеитова, Н. А. Сейлова // Сборник материалов III Международной научно–практической конференции "Фундаментальные научные исследования: теоретические и практические аспекты". – Кемерово, 2017. – С.188–190.

58. **Иванов, А. И.** Вычисление энтропии слабо коррелированных и сильно коррелированных длинных биометрических кодов на малых тестовых

выборках. [Текст] / А. И. Иванов, Б. Б. Ахметов, А. В. Безяев, К. А. Перфилов, Ж. К. Алимсеитова // Вестник НАН РК. – 2015. – №3. – С. 64–70.

59. **Алимсеитова, Ж.** Проблемы размерности задач распознавания образов и пути их решения [Текст] / Ж. Алимсеитова, Н. Сейлова, С. Гнатюк // Захист інформації. – 2017. – том 19, №4. – С. 310–316.

60. **ГОСТ Р 52633.0–2006.** Защита информации. Техника защиты информации. Требования к высоконадежным средствам биометрической аутентификации [Текст]. – Введ. 2006-12-27. – М.: Стандартинформ, 2006. – 28с.

61. **Ахметов, Б. С.** Технология использования больших нейронных сетей для преобразования нечетких биометрических данных в код ключа доступа [Текст]: монография / [Б. С. Ахметов, А. И. Иванов, В. А. Фунтиков и др.]. – Алматы: Издательство LEM, 2014. – 144 с.

62. **ГОСТ Р 52633.3–11**. Защита информации. Техника защиты информации. Тестирование стойкости средств высоконадежной биометрической защиты к атакам подбора [Текст]. – Введ. 2011-12-01. – М.: Стандартинформ, 2012. – 16с.

63. **Ахметов, Б. С.** Биометрическая аутентификация граждан в открытых информационных пространствах [Текст] / Б. С. Ахметов, А. И. Иванов, Т. С. Картбаев, А. Ю. Малыгин, А. А. Досжанова // Труды I международной научно-практической конференции «Интеллектуальные информационные и коммуникационные технологии-средство осуществления третьей индустриальной революции в свете стратегии "Казахстан-2050"». – Астана, 2013. – С. 458-459.

64. **Ахметов, Б. С.** Учет влияния корреляционных связей на результаты тестирования преобразователей биометрия-код [Текст] / Б. С. Ахметов, В. И. Волчихин, А. Ю. Малыгин, И. В. Урнев // Информационные и телекоммуникационные технологии: образование, наука, практика: Сборник трудов Международной научно-практической конференции – Алматы, 2012. – С. 34–37.

65. **Ахметов, Б. С.** О связи высокоразмерной энтропии и высокоразмерной корреляции с математическим ожиданием модулей коэффициентов парной корреляции [Текст] / Б. С. Ахметов, Т. С. Картбаев, А. Ю. Малыгин, Д. Н. Надеев // Вестник ВКГТУ им. Д. Серикбаева. Серия Информационные и телекоммуникационные технологии. – 2013. – №3-4. – С. 240-244.

66. **Волчихин, В. И.** Нейросетевая защита персональных биометрических данных [Текст]: монография / В. И. Волчихин, А. И. Иванов, И. Г. Назаров, В. А. Фунтиков, Ю. К. Язов. – М.: Радиотехника, 2012. – 160 с.

67. **Надеев, Д. Н.** Синтез таблиц вероятности ошибок первого и второго рода для неидеальных биометрико-нейросетевых преобразователей с 256 выходами [Текст] / Д. Н. Надеев // Нейрокомпьютеры: разработка, применение. – 2007. – № 12. – С. 30–31.

68. **Иванов, А. И.** Энтропийно-корреляционная оценка хэширующих свойств нейросетевого преобразователя биометрия-код доступа [Текст] / А. И. Иванов, А. В. Майоров, Ю. К. Язов // Нейрокомпьютеры: разработка, применение. – 2012. – № 3. – С. 35–40.

69. **Ахметов, Б. С.** Моделирование длинных биометрических кодов, воспроизводящих корреляционные связи выходных данных нейросетевого преобразователя [Текст] / Б. С. Ахметов, В. И. Волчихин, В. И. Куликов, Е. А. Малыгина // Нейрокомпьютеры: разработка, применение. – 2012. – №3. – С. 40-43.

70. **Кобзарь, А. И.** Прикладная математическая статистика. Для инженеров и научных работников [Текст] / А. И. Кобзарь – М.:ФИЗМАТЛИТ, 2006. – 816 с.

71. **ГОСТ Р 52633.1–2009**. Защита информации. Техника защиты информации. Требования к формированию баз естественных биометрических образов, предназначенных для тестирования средств высоконадежной биометрической аутентификации [Текст]. – Введ. 2006-12-15. – М.: Стандартинформ, 2010. – 24 с.

72. **Akhmetov, B. S.** Criterion synthesis the chi–square for dependent data [Текст] / B. S. Akhmetov, Zh. K. Alimseitova, N. I. Serikova, A. I. Ivanov, Yu. V.

Foontikova // 2nd International Conference on Innovation Challenges In Multidisciplinary Research & Practice. – Kuala Lumpur, Malaysia, 2014. – P. 64–70.

73. **Ахметов, Б. С.** Синтез критерия хи-квадат для зависимых данных [Текст] / Ж. К. Алимсеитова, Н. И. Серикова, А. И. Иванов, Ю. В. Фунтикова // Труды международного форума «Инженерное образование и наука в XXI веке: проблемы и перспективы», посвященной 80-летию КазНТУ имени К.И. Сатпаева – Алматы, 2014. – Том II. – С. 368-372.

74. **Akhmetov, B.** Biometric technology in securing the Internet using large neural network technology [Текст] / B. Akhmetov, T. Kartbayev, A. Doszhanova, A. Ivanov, A. Malygin // World Academy of science, Engineering and technology. – Singapore, 2013. – Iss.79. – P.129–138.

75. **Ахметов, Б. С.** Аппроксимация биномиального зависимого закона композициями нормального, равномерного, арксинусного распределения значений [Текст] / Б. С. Ахметов, Д. Н. Надеев, И. В. Урнев // Нейрокомпьютеры: разработка, применение. – 2012. – №3. – С. 17-20.

76. **Фунтикова, Ю. В.** Гипотеза χ^2 распределения расстояний Хэмминга для кодов биометрической аутентификации примеров образа «Свой» [Текст] / Ю. В. Фунтикова, А. И. Иванов, О. С. Захаров // Труды научно-технической конференции пензенских предприятий, обеспечивающих БЕЗОПАСНОСТЬ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ – Пенза, 2014. – Том 9. – С. 7-8.

77. Исследование, гармонизация, модификация и постановка на учет группы стандартов по биометрической поддержке информационной безопасности [Текст]: отчет НИРС 1022/ГФ4 – 15–ОТ. – Алматы, 2015. – 119 с.

78. **Kanade, S**. Multi–biometrics based cryptographic key regeneration scheme [Текст] / S. Kanade, D. Petrovska–Delacretaz, B. Dorizzi // IEEE 3rd International Conference on Biometrics: Theory, Applications, and Systems (BTAS’09). – Washington, 2009. – P. 333–339.

79. **Ахметов, Б.С.** Высоконадежная аутентификация: требования к базам биометрических образов [Текст] / Б. С. Ахметов, Ж. К. Алимсеитова, А. Картаев // Збірник тез доповідей міжнародна науково–практична конференція

"Інформаціина безпека та компютерні технологи" InfoSec&CompTech. – Кіровоград, 2016. – С. 11–12.

80. **Чморра, А. Л.** Маскировка ключа с помощью биометрии [Текст] / А. Л. Чморра // Проблемы передачи информации. – 2011. – №2(47). – С. 128–143.

81. Обзор методов измерения параметров статических биометрических образов личности [Текст] / А. Ю. Малыгин, Ю. И. Олейник, Е. А. Малыгина [и др.]. – М.: ЦВНИ МО РФ, 2006. – Деп. в центральном справочно–информационном фонде МО РФ, справка № 14504. Сер. Б. Вып. № 75, инв. № Б5838

82. Обзор методов измерения параметров динамических биометрических образов человека [Текст] / А. Ю. Малыгин, Ю. И. Олейник, Е. А. Малыгина [и др.]. – М.: ЦВНИ МО РФ, 2006. – Деп. в центральном справочно–информационном фонде МО РФ, справка № 14503. Сер. Б. Вып. № 75, инв. № Б5837

83. **Иванов, А.И.** Прогнозирование уровня защищенности, обеспечиваемого папиллярным рисунком отпечатка пальца [Текст] / А. И. Иванов, Д. А. Фунтиков, С. Л. Агафонов // Современные технологии безопасности. – 2005. – №3 (14). – С. 36–37.

84. **Wayman, J. L.** Testing and Evaluating Biometric Technologies: What the ustomer Needs to Know [Текст] / J. L. Wayman // Proc. CTST'97. – P. 329–348 (www.engr.sisu.edu/biometrics/fhwa.htm).

85. **Ахметов, Б. С.** Формирование биометрической базы рукописных образов на казахском языке для программ биометрической аутентификации личностей. [Текст] / Б. С. Ахметов, Ж. К. Алимсеитова, А. И. Малыгин, Х. И. Юбузова // Труды II Международной научно–практической конференции «Информационные и телекоммуникационные технологии: образование, наука, практика». – Алматы, 2015. – Том II. – С. 32–35.

86. **Akhmetov, B. S.** Methodology of biometric image databases formation [Текст] / B. S. Akhmetov, N. A. Seilova, Zh. K. Alimseitova, A. Baltabay // First

International Conference ISPIT 2015: Conference proceedings information security and protection of information technology. – St. Petersburg: Russia, 2015. – С. 48–52.

87. **Алимсеитова, Ж.** Программно–аппаратный модуль распознавания рукописных образов [Текст] / Ж. Алимсеитова // Известия Кыргызского государственного технического университета им. И. Раззакова. – 2018. – № 1(45). – С. 11-19.

ПРИЛОЖЕНИЯ

1. АКТЫ ВНЕДРЕНИЯ

- КазНИТУ имени К.И.Сатпаева (Казахстан, Алматы);

- Национальный авиационный университет (Украина, Киев);

- Научно-исследовательский центр «Тезис» КПИ имени Игоря Сикорского (Украина, Киев);

- ТОО «QUARES» (Казахстан, Алматы).

2. УЧЕБНИК

3. Свидетельства об авторском праве на программу

- «Нейротест 1.2»;

- Программно-аппаратный комплекс биометрико-нейросетевой аутентификации личности по отпечаткам пальцев FINGER

4. ПРОГРАММНЫЕ КОДЫ (МОДУЛИ)

Акты внедрения

Қ.И. СӘТБАЕВ атындағы ҚАЗАҚ ҰЛТТЫҚ ТЕХНИКАЛЫҚ ЗЕРТТЕУ УНИВЕРСИТЕТІ
КАЗАХСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
имени К.И. САТПАЕВА
KAZAKH NATIONAL RESEARCH TECHNICAL UNIVERSITY after K.I. SATPAYEV

Ақпараттық және телекоммуникациялық технологиялар институты

Институт информационных и телекоммуникационных технологий

Institute of information and telecommunication technologies

Қазақстан Республикасы,
050013, Алматы қаласы, Сәтбаев көшесі 22
№ _____

Республика Казахстан
050013, Алматы, ул. Сатпаева 22
Тел. 8 (727) 257-70-44, 257-71-34

Акт внедрения
результатов НИР в учебный процесс

Настоящим подтверждаем, что в ходе выполнения диссертационного исследования на тему: "Разработка интеллектуальной автоматизированной системы распознавания биометрических образов" разработан курс "Биометрия и нейронные сети" в объеме 3 кредита (30 лекционных и 15 лабораторных часов), который внедрен в учебный процесс на кафедре "Информационная безопасность" по специальности 5В100200 - Системы информационной безопасности.

Разработчики курса: лектор Алимсеитова Ж.К.

Директор ИИиТТ **Б.С. Ахметов**

Зав. кафедрой ИБ **Н.А. Сейлова**

ЗАТВЕРДЖУЮ:

Проректор з навчальної
методичної роботи
Національного авіаційного
університету

Т. Іванова

"07" 07 2017 р.

АКТ

впровадження у навчальний процес результатів дисертаційної роботи
Алімсеітової Жулдиз Кенесхановни
«Розробка інтелектуальної автоматизованої системи розпізнавання біометричних образів»
на здобуття наукового ступеня кандидата технічних наук

Комісія у складі: голова – завідувач кафедри безпеки інформаційних технологій (БІТ) Корченко О.Г., доцент кафедри БІТ Казмірчук С.В., доцент кафедри БІТ Гнатюк С.О. склали даний акт про те, що результати дисертаційного дослідженнАлімсеітової Жулдиз Кенесхановни «Розробка інтелектуальної автоматизованої системи розпізнавання біометричних образів» впроваджені в навчальний процес і використовуються на кафедрі БІТ при викладанні дисципліни «Моделювання систем захисту інформації» ", що входить до навчального плану підготовки магістрів за спеціальністю "Кібербезпека".

№ з/п	Назва роботи, що впроваджується	Форма впровадження	Ефективність від впровадження
	1	2	3
1.	Метод інтелектуального розпізнавання кіберзагроз біометричних образів	Лекція	Систематизація навчального матеріалу та надання студентам знань щодо інтелектуалізованого розпізнавання образів при моделюванні систем захисту інформації.
2.	Імітаційні моделі розпізнавання біометричних образів для систем захисту інформації.	Лабораторна робота	Ознайомлення та навчання студентів використовувати метод інтелектуального ідентифікації та аутентифікації розпізнавання біометричних образів для реалізації процедур.

Голова комісії,
завідувач кафедри БІТ,
лауреат Державної премії України
в галузі науки і техніки, д.т.н., проф. О. Корченко

Члени комісії:
доцент кафедри БІТ,
к.т.н., доц. С. Казмірчук

доцент кафедри БІТ,
к.т.н., доц. С. Гнатюк

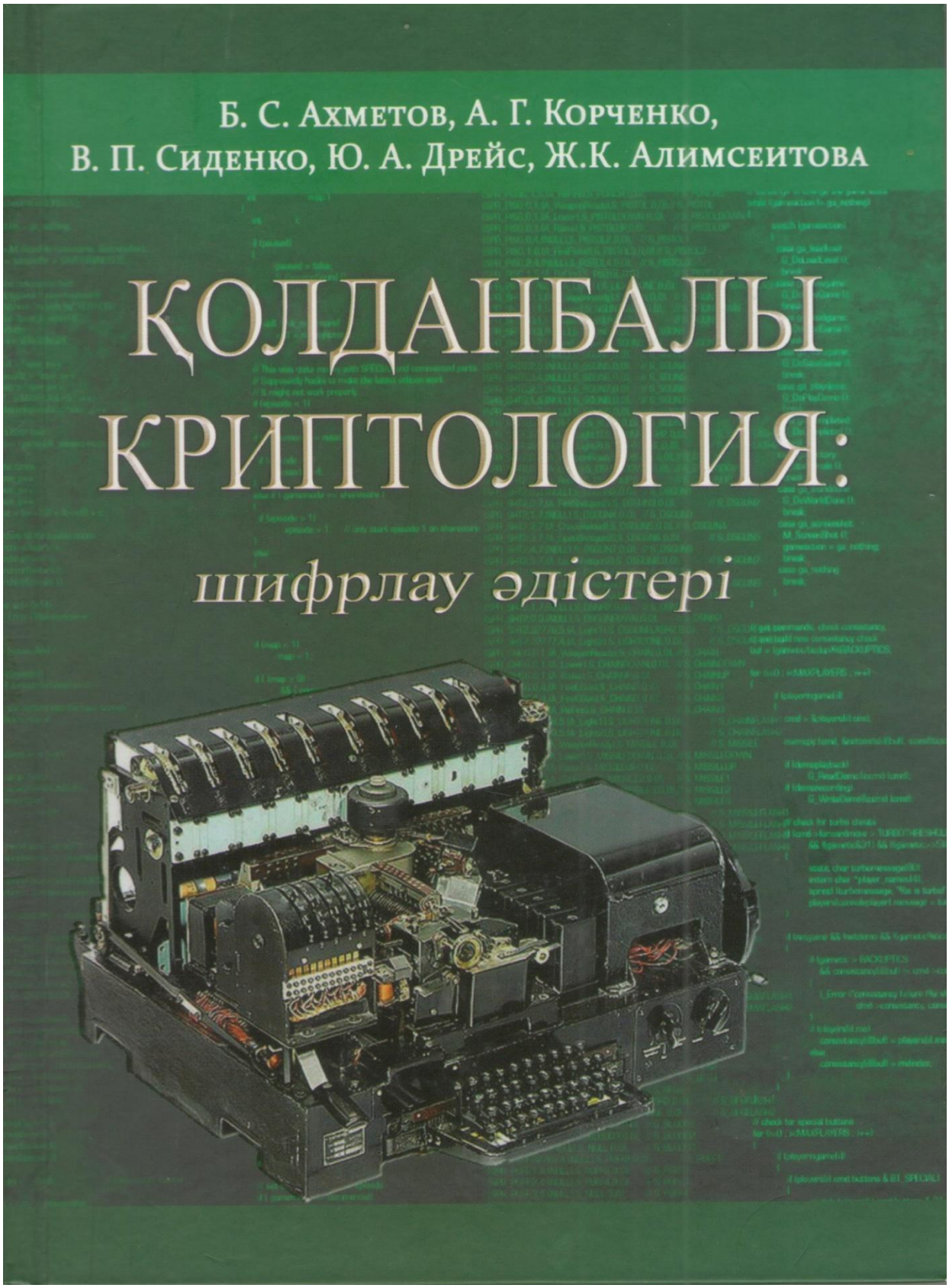
АКТ

о внедрении результатов диссертационной работы
Алимсеитовой Жулдыз Кенесхановны
«Разработка интеллектуальной автоматизированной системы распознавания биометрических образов»
в Научно-исследовательском центре «ТЕЗИС» КПИ им. Игоря Сикорского

Настоящий акт составлен в подтверждение того, что теоретические и экспериментальные исследования, программно-аппаратные комплексы, приведенные в диссертационной работе Алимсеитовой Ж.К. «Разработка интеллектуальной автоматизированной системы распознавания биометрических образов» применялись для формирования баз и распознавания биометрических идентификаторов для эффективного обеспечения и разграничения доступа к ресурсам информационных систем Научно-исследовательского центра «ТЕЗИС».

Директор НИЦ «ТЕЗИС» М.И. Прокофьев

Учебник


Б. С. Ахметов, А. Г. Корченко,
В. П. Сиденко, Ю. А. Дрейс, Ж.К. Алимсеитова
ҚОЛДАНБАЛЫ
КРИПТОЛОГИЯ:
шифрлау әдістері

Свидетельства об авторском праве на программы

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ

РЕСПУБЛИКА КАЗАХСТАН

АВТОРЛЫҚ ҚҰҚЫҚПЕН ҚОРҒАЛАТЫН ОБЪЕКТІЛЕРГЕ ҚҰҚЫҚТАРДЫҢ МЕМЛЕКЕТТІК ТІЗІЛІМГЕ МӘЛІМЕТТЕРДІ ЕНГІЗУ ТУРАЛЫ

КУӘЛІК

2019 жылғы « 8 » қаңтар № 1240

Автордың (лардың) жеке , аты, әкесінің аты (егер ол жеке басын куәландыратын құжатта көрсетілсе):

АЛИМСЕЙТОВА ЖУЛДЫЗ КЕНЕСХАНОВНА; КАРТБАЕВ ТИМУР СААТДИНОВИЧ; АХМЕТОВ БАХЫТЖАН СРАЖАТДИНОВИЧ; ДОСКАНОВА АЛИЯ АМАНТАЕВНА

Авторлық құқық объектісі: ЭЕМ-ге арналған бағдарлама

Объектінің атауы: НейроТест 1.2

Объектіні жасаған күні: 26.10.2018

СВИДЕТЕЛЬСТВО

О ВНЕСЕНИИ СВЕДЕНИЙ В ГОСУДАРСТВЕННЫЙ РЕЕСТР ПРАВ НА ОБЪЕКТЫ, ОХРАНЯЕМЫЕ АВТОРСКИМ ПРАВОМ

№ 1240 от « 8 » января 2019 года

Фамилия, имя, отчество, (если оно указано в документе, удостоверяющем личность) автора (ов):

АЛИМСЕЙТОВА ЖУЛДЫЗ КЕНЕСХАНОВНА; КАРТБАЕВ ТИМУР СААТДИНОВИЧ; АХМЕТОВ БАХЫТЖАН СРАЖАТДИНОВИЧ; ДОСКАНОВА АЛИЯ АМАНТАЕВНА

Вид объекта авторского права: программа для ЭВМ

Название объекта: НейроТест 1.2

Дата создания объекта: 26.10.2018

Құжат түпнұсқалығын http://www.kazpatent.kz/ru сайтының "Авторлық құқық" бөлімінде тексеруге болады https://copyright.kazpatent.kz

Подлинность документа возможно проверить на сайте kazpatent.kz в разделе «Авторское право» https://copyright.kazpatent.kz

Подписано ЭЦП

Оспанов Е. К.

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ

РЕСПУБЛИКА КАЗАХСТАН

АВТОРЛЫҚ ҚҰҚЫҚПЕН ҚОРҒАЛАТЫН ОБЪЕКТІЛЕРГЕ ҚҰҚЫҚТАРДЫҢ МЕМЛЕКЕТТІК ТІЗІЛІМГЕ МӘЛІМЕТТЕРДІ ЕНГІЗУ ТУРАЛЫ

КУӘЛІК

2019 жылғы « 11 » қаңтар № 1287

Автордың (лардың) жеке , аты, әкесінің аты (егер ол жеке басын куәландыратын құжатта көрсетілсе):

АЛИМБЕКОВА ЖҰЛДЫЗ КЕНЕСХАНОВНА; АХМЕТОВ БАХЫТЖАН ОРАЖАТДИНОВИЧ; КАРТБАЕВ ТИМУР САЛТАНОВИЧ; ДОСЖАНОВА АЛИЯ АМАНТАЕВНА; ТОЛЫБАЕВ ШАРАПАТДИН ДОСЖАНУРАЕВИЧ

Авторлық құқық объектісі: ЭЕМ-ге арналған бағдарлама
Программно-аппаратный комплекс биометрико-нейросетевой аутентификации личности

Объектінің атауы: по отпечаткам пальцев FINGER

Объектіні жасаған күні: 25.10.2018

СВИДЕТЕЛЬСТВО

О ВНЕСЕНИИ СВЕДЕНИЙ В ГОСУДАРСТВЕННЫЙ РЕЕСТР ПРАВ НА ОБЪЕКТЫ, ОХРАНЯЕМЫЕ АВТОРСКИМ ПРАВОМ

№ 1287 от « 11 » января 2019 года

Фамилия, имя, отчество, (если оно указано в документе, удостоверяющем личность) автора (ов):

АЛИМБЕКОВА ЖҰЛДЫЗ КЕНЕСХАНОВНА; АХМЕТОВ БАХЫТЖАН ОРАЖАТДИНОВИЧ; КАРТБАЕВ ТИМУР САЛТАНОВИЧ; ДОСЖАНОВА АЛИЯ АМАНТАЕВНА; ТОЛЫБАЕВ ШАРАПАТДИН ДОСЖАНУРАЕВИЧ

Вид объекта авторского права: программа для ЭВМ
Программно-аппаратный комплекс биометрико-нейросетевой аутентификации личности

Название объекта: по отпечаткам пальцев FINGER

Дата создания объекта: 25.10.2018

Құжат түпнұсқалығын http://www.kazpatent.kz/ru сайтының "Авторлық құқық" бөлімінде тексеруге болады https://copyright.kazpatent.kz

Подлинность документа возможно проверить на сайте kazpatent.kz в разделе «Авторское право» https://copyright.kazpatent.kz

Подписано ЭЦП

Осланов Е. К.

Программные коды

```
/////////////////////////////////////////////////////////////////////////////
//файл:LearnDlg.h:файл заголовка
//описание:     /////////////////////////////////////////////////////////////////////////////

#pragma once
#include "afxwin.h"
#include "../CFP/CCorrectImg.h"
#include "../CFP/CFPSegment.h"
#include "../CFP/CFPOrientation.h"
#include "../CFP/CFPEnhance.h"
#include "../CFP/CFPThinning.h"
#include "../CFP/CFPMinutiae.h"
#include "../CFP/CFunctional.h"
#include "../CFP/CFPTemplate.h"

//

class CLearnDlg : public CDialog
{
	DECLARE_DYNAMIC(CLearnDlg)

public:
	CLearnDlg(CWnd* pParent = NULL);   // стандартныйконструктор
	CLearnDlg(int mode, CWnd* pParent = NULL);
	virtual ~CLearnDlg();

	intmode_dlg;  // Режимдиалоговогоокна
	int	IFLAG; // Флаг, указывающийнато, чтоОПзахвачен
	int iLight;
	int	correctPosX;   // ПозицияОП
	float correctAngle;
	BYTE *scanImg;	// ИзображениеОП
	int *Img;
	int	*sImg;
	CWinThread* CaptureThread;// ПотокзахватаОП
	int binKeyArr[NEURON_NUMBER];   // Обучающийключ
	int imageCount;// Счётчикобразов
	FILE*  stream;

	//КлассыобработкиизображенияОП
	CCorrectImgcorrectobj;
	CFPSegment sobject;
	CFPOrientation orobject;
	CFPEnhance eobject;
	CFPThinning tobject;
	CFPMinutiae mobject;
	CFunctional fobject;
	CFPTemplate templobject;
```

```
// Данныедиалоговогоокна
	enum { IDD = IDD_LEARN_DIALOG };

protected:
	virtual void DoDataExchange(CDataExchange* pDX);	// поддержка DDX/DDV
	virtual BOOL OnInitDialog();

	DECLARE_MESSAGE_MAP()
public:
	CButton m_add_button;
	CButton m_clear_button;
	CButton m_test_button;
	CStatic m_static_img_count;
	CStatic m_static_fin_img;
	afx_msg void OnBnClickedAddButton();
	afx_msg void OnBnClickedClearButton();
	afx_msg void OnBnClickedTestButton();
	afx_msg void OnPaint();
	void AddImage();
	void LearnNet();
	void DeleteAllImages();
	void AuthUser();
	void getMinutiaeFromImage(int &numberMinutiae, FPPoint *sMinutiae);

	CButton m_learn_button;
	afx_msg void OnBnClickedLearnButton();
	afx_msg void OnClose();
	afx_msg void OnBnClickedCancel();
	CStatic m_static_label;
	afx_msg BOOL OnEraseBkgnd(CDC* pDC);
	CStatic m_static_btn;
};

/////////////////////////////////////////////////////////////////////////////
//файл:LearnDlg.cpp:файлреализации
//описание:	/////////////////////////////////////////////////////////////////////////////

#include "stdafx.h"
#include "NeuroFinger.h"
#include "LearnDlg.h"
#include "ProgressLearnDlg.h"
#include "../Tools/FileManipulations.h"
#include "../Tools/ConversionSFN.h"
#include "../Tools/NetSimulation.h"
#include "../Tools/HashingKey.h"

#include "../Futronic/ftrapi.h"
#include "../CFP/FPDefine.h"
#include "atlimage.h"

bool CancelCaptureOperation;
```

```
bool GoodCaptureOperation;
int countImageSuc;    // счетчикзахвата
UINT CaptureLearning (LPVOID pParam); //
ПотокзахватаОПиформированияобучающейвыборки
UINT AuthenticateUser (LPVOID pParam); // Потокаутентификации
UINT CaptureTesting (LPVOID pParam); // Потоктестирования
void FTR_CBAPI cbControl(FTR_USER_CTX Context, FTR_STATE StateMask,
FTR_RESPONSE *pResponse,
 FTR_SIGNAL Signal, FTR_BITMAP_PTR pBitmap );
// диалоговоеокно CLearnDlg
IMPLEMENT_DYNAMIC(CLearnDlg, CDialog)

CLearnDlg::CLearnDlg(CWnd* pParent /*=NULL*/)
	: CDialog(CLearnDlg::IDD, pParent)
{

}

CLearnDlg::CLearnDlg(int mode, CWnd* pParent /*=NULL*/)
: CDialog(CLearnDlg::IDD, pParent)
{
	mode_dlg = mode;
}

CLearnDlg::~CLearnDlg()
{
	delete [] scanImg;
	delete [] Img;
	delete [] sImg;
}

void CLearnDlg::DoDataExchange(CDataExchange* pDX)
{
	CDialog::DoDataExchange(pDX);
	DDX_Control(pDX, IDC_ADD_BUTTON, m_add_button);
	DDX_Control(pDX, IDC_CLEAR_BUTTON, m_clear_button);
	DDX_Control(pDX, IDC_TEST_BUTTON, m_test_button);
	DDX_Control(pDX, IDC_STATIC_IMG_COUNT, m_static_img_count);
	DDX_Control(pDX, IDC_STATIC_FIN_IMG, m_static_fin_img);
	DDX_Control(pDX, IDC_LEARN_BUTTON, m_learn_button);
	DDX_Control(pDX, IDC_STATIC_LABEL, m_static_label);
	DDX_Control(pDX, IDC_STATIC_BTN, m_static_btn);
}

BEGIN_MESSAGE_MAP(CLearnDlg, CDialog)
	ON_BN_CLICKED(IDC_ADD_BUTTON, &CLearnDlg::OnBnClickedAddButton)
	ON_BN_CLICKED(IDC_CLEAR_BUTTON, &CLearnDlg::OnBnClickedClearButton)
	ON_BN_CLICKED(IDC_TEST_BUTTON, &CLearnDlg::OnBnClickedTestButton)
	ON_WM_PAINT()
	ON_BN_CLICKED(IDC_LEARN_BUTTON, &CLearnDlg::OnBnClickedLearnButton)
	ON_WM_CLOSE()
//	ON_BN_CLICKED(IDCANCEL, &CLearnDlg::OnBnClickedCancel)
```

```
ON_BN_CLICKED(IDCANCEL, &CLearnDlg::OnBnClickedCancel)
ON_WM_ERASEBKGND()
END_MESSAGE_MAP()

BOOL CLearnDlg::OnInitDialog()
{
	CDialog::OnInitDialog();

	// Проверка связи со сканером ОП
	CaptureThread=NULL;
	FTRAPI_RESULT StatusDevice;
	FTRInitialize();
	StatusDevice=FTRSetParam(FTR_PARAM_CB_FRAME_SOURCE,
(FTR_PARAM_VALUE)FSD_FUTRONIC_USB);


	while (StatusDevice==FTR_RETCODE_DEVICE_NOT_CONNECTED)
	{
		if (MessageBox("Сканеротпечатковпальцев Futronic FS-80
неподключенккомпьютеру!\n" "Подключитесканеринажмите "Повтор" ","Макет",
MB_RETRYCANCEL | MB_ICONSTOP )==IDCANCEL)

		{
			PostMessage(WM_CLOSE);
			return FALSE;
		}
		else {
			FTRInitialize();
			StatusDevice=FTRSetParam(FTR_PARAM_CB_FRAME_SOURCE,
(FTR_PARAM_VALUE)FSD_FUTRONIC_USB);
		}

	}

	scanImg=new BYTE [320*480];
	Img=new int [320*480];
	sImg= new int [30*20];

	memset(Img,0,320*480*sizeof(int));
	memset(scanImg,0,320*480*sizeof(BYTE));
	memset(sImg,0,20*30*sizeof(int));

	IFLAG=3;
	correctPosX=0;
	CancelCaptureOperation=false;
	iLight = 1;

	// Задаемключ
	CHashingKey objHash;
	char userPsw[33]={"AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA0"};
	objHash.Psw2BinKey(userPsw, binKeyArr);
```

```
	switch (mode_dlg){
		caseTRAIN_MODE: {
			// Определяем количество сохранённых образов
			imageCount = 0;
			while(true)
			{
				objCFileManip.ChangeImageName(imageCount+1,
objCFileManip.tofileFullPath);
				if ( !(fopen_s(&stream,objCFileManip.tofileFullPath,"r")) )
				{
					int ch = fgetc( stream );
					fclose(stream);
					if( ch != -1) imageCount++;
					else break;
				}
				else break;
			}

			SetWindowText("Регистрация пациента и обучение нейронной сети");

			if (imageCount == 0){
				m_clear_button.EnableWindow(0);
			}

			if (imageCount < 2){
				m_learn_button.EnableWindow(0);
			}

			m_test_button.EnableWindow(0);
			GetDlgItem(IDC_TEST_BUTTON)->ShowWindow(SW_HIDE);

			// Вывод числа примеров
			CString str;
			str.Format(_T(" %d"),imageCount);
			m_static_img_count.SetWindowText(str);

			break;
		}
		case TEST_MODE: {

			SetWindowText("Тестированиеобученнойнейронноймсети");

			GetDlgItem(IDC_ADD_BUTTON)->ShowWindow(SW_HIDE);
			GetDlgItem(IDC_CLEAR_BUTTON)->ShowWindow(SW_HIDE);
			GetDlgItem(IDC_LEARN_BUTTON)->ShowWindow(SW_HIDE);
			GetDlgItem(IDC_STATIC_USER_LABEL)->ShowWindow(SW_HIDE);
			GetDlgItem(IDC_STATIC_USER_NAME)->ShowWindow(SW_HIDE);
			m_static_label.SetWindowText("МераХэмминга:");

			break;
		}
		case AUT_MODE: {
```

```
                SetWindowText("Аутентификацияпациента");

                GetDlgItem(IDC_ADD_BUTTON)->ShowWindow(SW_HIDE);
                GetDlgItem(IDC_CLEAR_BUTTON)->ShowWindow(SW_HIDE);
                GetDlgItem(IDC_LEARN_BUTTON)->ShowWindow(SW_HIDE);
                GetDlgItem(IDC_TEST_BUTTON)->ShowWindow(SW_HIDE);
                GetDlgItem(IDC_STATIC_IMG_COUNT)->ShowWindow(SW_HIDE);
                GetDlgItem(IDC_STATIC_LABEL)->ShowWindow(SW_HIDE);

                AuthUser();

                break;
            }
            default: break;
        }

        return TRUE;
}

void CLearnDlg::OnPaint()
{
        CPaintDC dc(this); // device context for painting

        CBitmap bitmap, *pBitmap;
        char *BMPCAP = new char[320*480*4];
        memset(BMPCAP, 0, 320*480*4);

        // Создание совместимого контекста устройства
        CDC dcMem;
        dcMem.CreateCompatibleDC(&dc);

        if (IFLAG == 1) // ОПринят
        {
            for(int i=0; i<320*480; i++){
                BMPCAP[i*4]   =  (255-(Img[i]))*0.6f+100;  //синий
                BMPCAP[i*4+1] =  (255-(Img[i]))*0.87+32;   //зеленый
                BMPCAP[i*4+2] =  (255-(Img[i])); //красный
            }

            bitmap.CreateBitmap(320, 480, 1, 32, BMPCAP);
            pBitmap=dcMem.SelectObject(&bitmap);

            // Вывод границ рамки и центрального перекрестия
            CPen pen1(PS_SOLID ,1,RGB(0,0,0));
            dcMem.SelectObject(&pen1);
            // Рисование разметки области ввода
            dcMem.MoveTo(160,220);
            dcMem.LineTo(160,260);
            dcMem.MoveTo(140,240);
            dcMem.LineTo(180,240);

            dcMem.MoveTo(10,10);
```

```
dcMem.LineTo(35,10);
dcMem.MoveTo(10,10);
dcMem.LineTo(10,35);

dcMem.MoveTo(310,10);
dcMem.LineTo(285,10);
dcMem.MoveTo(310,10);
dcMem.LineTo(310,35);

dcMem.MoveTo(10,470);
dcMem.LineTo(35,470);
dcMem.MoveTo(10,470);
dcMem.LineTo(10,445);

dcMem.MoveTo(310,470);
dcMem.LineTo(310,445);
dcMem.MoveTo(310,470);
dcMem.LineTo(285,470);

// Вывод указателей перемещений
CPenpen2(PS_SOLID ,2,RGB(255,125,125));// Беремкарандаш
dcMem.SelectObject(&pen2);

if (correctPosX<-10){
	// Вывод стрелки влево
	dcMem.MoveTo(0,240);
	dcMem.LineTo(10,230);
	dcMem.MoveTo(0,240);
	dcMem.LineTo(10,250);

	dcMem.MoveTo(10,240);
	dcMem.LineTo(20,230);
	dcMem.MoveTo(10,240);
	dcMem.LineTo(20,250);

	dcMem.MoveTo(20,240);
	dcMem.LineTo(30,230);
	dcMem.MoveTo(20,240);
	dcMem.LineTo(30,250);
}

if (correctPosX>10){
	// Вывод стрелки вправо
	dcMem.MoveTo(320,240);
	dcMem.LineTo(310,230);
	dcMem.MoveTo(320,240);
	dcMem.LineTo(310,250);

	dcMem.MoveTo(310,240);
	dcMem.LineTo(300,230);
	dcMem.MoveTo(310,240);
	dcMem.LineTo(300,250);
```

```
                dcMem.MoveTo(300,240);
                dcMem.LineTo(290,230);
                dcMem.MoveTo(300,240);
                dcMem.LineTo(290,250);
            }

            // Копирование содержимого одного контекста в другой
            CRect rc1,rc2;
            GetClientRect(rc1);
            m_static_fin_img.GetClientRect(rc2);
            CRect xy;
            xy.left = rc1.right*(11)/(473);
            xy.top  = rc1.bottom*(11)/(504);
            dc.BitBlt(xy.left,xy.top,320,480,&dcMem,0,0,SRCCOPY);
        }

        if (IFLAG == 0) { // ОПнепринят

            bitmap.LoadBitmap(IDB_BITMAP3);
            pBitmap=dcMem.SelectObject(&bitmap);
            CRect rc1,rc2;
            GetClientRect(rc1);
            m_static_fin_img.GetClientRect(rc2);
            CRect xy;
            xy.left = rc1.right*(11)/(473);
            xy.top  = rc1.bottom*(11)/(504);
            dc.BitBlt(xy.left,xy.top,320,480,&dcMem,0,0,SRCCOPY);
        }

        if (IFLAG == 3) { // Режимневыбран

            bitmap.LoadBitmap(IDB_BITMAP1);
            pBitmap=dcMem.SelectObject(&bitmap);
            CRect rc1,rc2;
            GetClientRect(rc1);
            m_static_fin_img.GetClientRect(rc2);
            CRect xy;
            xy.left = rc1.right*(11)/(473);
            xy.top  = rc1.bottom*(11)/(504);
            dc.BitBlt(xy.left,xy.top,320,480,&dcMem,0,0,SRCCOPY);
        }

        // Восстановление первоначально созданного битового массива
        if(pBitmap) dcMem.SelectObject(pBitmap);

        // Удаление совместимого контекста устройства
        dcMem.DeleteDC();
        delete (BMPCAP);

    }
```

```
// обработчики сообщений CLearnDlg

void CLearnDlg::OnBnClickedAddButton()
{
	m_add_button.EnableWindow(0);

	if((CaptureThread!=NULL) && (GoodCaptureOperation==false)){

		CancelCaptureOperation=true;
		WaitForSingleObject(CaptureThread->m_hThread,INFINITE);
		delete(CaptureThread);
		CaptureThread=NULL;
	}

	FTRInitialize();
	FTRSetParam(FTR_PARAM_CB_FRAME_SOURCE,
(FTR_PARAM_VALUE)FSD_FUTRONIC_USB);
	FTRSetParam( FTR_PARAM_CB_CONTROL, (FTR_PARAM_VALUE)cbControl);

	memset(scanImg,0,320*480*sizeof(BYTE));

	GoodCaptureOperation=false;
	CancelCaptureOperation=false;
	CaptureThread=AfxBeginThread(CaptureLearning,this,THREAD_PRIORITY_NORMAL,0
,CREATE_SUSPENDED);
	CaptureThread->m_bAutoDelete=false;
	CaptureThread->ResumeThread();
}

void CLearnDlg::OnBnClickedClearButton()
{
	CancelCaptureOperation=true;

	if(CaptureThread!=NULL){
		WaitForSingleObject(CaptureThread->m_hThread,INFINITE);
		delete(CaptureThread);
		CaptureThread=NULL;
	}

	DeleteAllImages();

	m_add_button.EnableWindow(1);
	m_clear_button.EnableWindow(0);
	m_learn_button.EnableWindow(0);
}

void CLearnDlg::OnBnClickedTestButton()
{
	if(CaptureThread!=NULL) {
		CancelCaptureOperation=true;
		WaitForSingleObject(CaptureThread->m_hThread,INFINITE);
		delete(CaptureThread);
```

```
		CaptureThread=NULL;
	}

	FTRInitialize();
	FTRSetParam(FTR_PARAM_CB_FRAME_SOURCE,
(FTR_PARAM_VALUE)FSD_FUTRONIC_USB);
	FTRSetParam( FTR_PARAM_CB_CONTROL, (FTR_PARAM_VALUE)cbControl);

	memset(scanImg,0,320*480*sizeof(BYTE));

	GoodCaptureOperation=false;
	CancelCaptureOperation=false;
	CaptureThread=AfxBeginThread(CaptureTesting,this,THREAD_PRIORITY_NORMAL,0,
CREATE_SUSPENDED);
	CaptureThread->m_bAutoDelete=false;
	CaptureThread->ResumeThread();
}

void CLearnDlg::OnBnClickedLearnButton()
{
	m_add_button.EnableWindow(0);
	m_learn_button.EnableWindow(0);
	m_clear_button.EnableWindow(0);

	CancelCaptureOperation=true;

	if(CaptureThread!=NULL){
		WaitForSingleObject(CaptureThread->m_hThread,INFINITE);
		delete(CaptureThread);
		CaptureThread=NULL;
	}

	this->LearnNet();

	m_add_button.EnableWindow(1);
	m_learn_button.EnableWindow(1);
	m_clear_button.EnableWindow(1);
}

void CLearnDlg::LearnNet()
{

	this->IFLAG=3;
	this->Invalidate(0);
	if (this->imageCount >= 2){
		//Обучениенейроннойсети
		CProgressLearnDlg objProgressDlg(imageCount, binKeyArr, NULL);
		objProgressDlg.DoModal();

		if (objProgressDlg.rezultLearn) MessageBox("Нейроннаясетьнеобучена! \n
Необходимо заменить примеры с ошибкой.","Макет", MB_ICONWARNING);
	}
```

```
	else {MessageBox("Количество примеров не достаточно для обучения! \n Необходимо
более одного примера.","Макет", MB_ICONWARNING);}

}

// Добавление нового образа
void CLearnDlg::AddImage()
{
	if  ( imageCount < 100)
	{
		//Увеличениесчётчикаобразов
		imageCount++;
		CString fileName;
		objCFileManip.ChangeImageName(imageCount,fileName);
		//ЗаписьизображенияОП
		objCFileManip.WriteImageFile(fileName,this->Img);

		//Отображение списка сохранённых образов
		CString str;
		str.Format(_T("     %d"), imageCount);
		m_clear_button.EnableWindow(1);

		if (imageCount >= 2){
			m_learn_button.EnableWindow(1);
		}

		str.Format(_T(" %d"),imageCount);
		m_static_img_count.SetWindowText(str);
	}
	else MessageBox("Добавление примера невозможно!\n Достигнуто максимальное
количество\n обрабатываемых системой образов!", "Макет", MB_ICONWARNING);
}

// Удалитьвсеобразы
voidCLearnDlg::DeleteAllImages()
{
	if ( MessageBox (" Вы уверены, что хотите удалить все сохраненные примеры?",
"Макет", MB_ICONQUESTION|MB_YESNO) == IDYES )
	{
		objCFileManip.DeleteAllIMFiles(imageCount);
		MessageBox (" Все образы успешно удалены.", "Макет", MB_ICONASTERISK);
		//Обнуление списка сохраненных образов
		imageCount = 0;
		this->IFLAG = 3;
		CString str;
		str.Format(_T(" %d"),imageCount);
		m_static_img_count.SetWindowText(str);
		this->Invalidate(0);
	}

}
```

```
void CLearnDlg::AuthUser(){

	if((CaptureThread!=NULL) && (GoodCaptureOperation==false)){

		CancelCaptureOperation=true;
		WaitForSingleObject(CaptureThread->m_hThread,INFINITE);
		delete(CaptureThread);
		CaptureThread=NULL;
	}

	FTRInitialize();
	FTRSetParam(FTR_PARAM_CB_FRAME_SOURCE,
(FTR_PARAM_VALUE)FSD_FUTRONIC_USB);
	FTRSetParam( FTR_PARAM_CB_CONTROL, (FTR_PARAM_VALUE)cbControl);

	memset(scanImg,0,320*480*sizeof(BYTE));

	GoodCaptureOperation=false;
	CancelCaptureOperation=false;
	CaptureThread=AfxBeginThread(AuthenticateUser,this,THREAD_PRIORITY_NORMAL,
0,CREATE_SUSPENDED);
	CaptureThread->m_bAutoDelete=false;
	CaptureThread->ResumeThread();
}

// Функцияполученияминуцийпоизображению
void CLearnDlg::getMinutiaeFromImage(int &numberMinutiae, FPPoint *sMinutiae){

	float *oImg;
	int *EnhImg;
	int *BinImg;
	int *tImg;
	int sizeoImg;

	sizeoImg=int(FpSizeImgHt*FpSizeImgWt/(BLKSZ*BLKSZ));
	oImg=new float [sizeoImg];

	EnhImg=new int[FpSizeImgHt*FpSizeImgWt];
	BinImg= new int [FpSizeImgHt*FpSizeImgWt];
	tImg= new int [FpSizeImgHt*FpSizeImgWt];

	memset(oImg,0,sizeoImg*sizeof(float));
	memset(EnhImg,0,FpSizeImgWt*FpSizeImgHt*sizeof(int));
	memset(BinImg,0,FpSizeImgWt*FpSizeImgHt*sizeof(int));
	memset(tImg,0,FpSizeImgWt*FpSizeImgHt*sizeof(int));
	memset(sMinutiae,0,300*sizeof(FPPoint));

	this->sobject.getSegment(this->Img,this->sImg,FpSizeImgWt,FpSizeImgHt);
	this->orobject.getOrientation(this->Img,FpSizeImgWt,FpSizeImgHt, oImg);
// Определениеполяориентаций
	//this->eobject.EnhanceGauss(this->Img,EnhImg,FpSizeImgWt,FpSizeImgHt);
```

```
// ПреобразованиеГаусса
	this->eobject.EnhanceMfs8600(Img,oImg,EnhImg,FpSizeImgWt,FpSizeImgHt);
// Улучшениекачестваизображения
	this->eobject.ExtractRidges(EnhImg,oImg,BinImg,FpSizeImgWt,FpSizeImgHt);
// Извлечениегребнейилибинаризация
	this->tobject.thinning(BinImg,tImg,FpSizeImgWt,FpSizeImgHt);
// Утончениепапиллярныхлиний
	numberMinutiae=this->mobject.getMinutiae(tImg,sImg,oImg,FpSizeImgWt,FpSizeImgHt,
sMinutiae); // Поискособыхточекилиминуций

	delete [] oImg;
	delete [] EnhImg;
	delete [] tImg;
	delete [] BinImg;

}

void FTR_CBAPI cbControl( FTR_USER_CTX Context, FTR_STATE StateMask,
FTR_RESPONSE *pResponse,
						FTR_SIGNAL Signal, FTR_BITMAP_PTR pBitmap
)
{

	if (CancelCaptureOperation==true)
		*pResponse = FTR_CANCEL;

	if (Signal==2){
		*pResponse = FTR_CONTINUE;
	}

	if (Signal==1){
		*pResponse = FTR_CANCEL;
		countImageSuc = 0;
	}

	if (Signal==0){
		//	countImageSuc++; // счетчикзахвата
	}

}

// Поток захвата ОП и формирования обучающей выборки
UINT CaptureLearning (LPVOID pParam){

	CLearnDlg*		pObject=(CLearnDlg*)pParam;
	float percent;
	FTRAPI_RESULT Rlt;

	pObject->IFLAG=0;	// вывести изображение "Приложите палец"

	pObject->Invalidate(0);
	int rescorrect=0;
```

```
countImageSuc=0; // счетчикзахвата
bool captureFlag = false; // флагзахватаизобр. отп.

while (1) // пока не выполнится условие коррекции
{
	Rlt=FTRCaptureFrame( NULL, pObject->scanImg);

	if ((Rlt==FTR_RETCODE_OK)) {

		if (captureFlag)pObject->IFLAG=0;
		elsepObject->IFLAG=1; // вывестиизображениеОП

		for (int i=0; i<320*480; i++)
			pObject->Img[i]=(int)pObject->scanImg[i];

		// Проверкакорректногорасположения
		percent=pObject->sobject.getSegment(pObject->Img,pObject-
>sImg,FpSizeImgWt,FpSizeImgHt);

		if (percent > 30)  {
			rescorrect=pObject->correctobj.IsCorrectImg(pObject->sImg);
			pObject->correctobj.getPositionX(pObject->correctPosX);
			pObject->correctobj.getPositionAngle(pObject->correctAngle);

		}

		pObject->Invalidate(0);

		if((rescorrect == 1) && (!captureFlag))  {
			pObject->AddImage(); captureFlag = true; }
	}

	// Закрытиеокнаобучения
	if ((Rlt==FTR_RETCODE_CANCELED_BY_USER)
&& (CancelCaptureOperation==TRUE)) {
		break;
	}

	// Выводкартинки "приложитепалец"
	if ((Rlt==FTR_RETCODE_CANCELED_BY_USER)
&& (CancelCaptureOperation==FALSE)) {
		if (pObject->IFLAG!=0){
			pObject->IFLAG=0;
			pObject->Invalidate(0);
		}

		captureFlag = false;
	}
}

FTRTerminate();
```

```
	GoodCaptureOperation=true;

	pObject->correctPosX=0;

	return 0;

}

// Поток аутентификации пользователя
// Поток завершается при удачной аутентификации
UINTAuthenticateUser (LPVOIDpParam){

	CLearnDlg*		pObject=(CLearnDlg*)pParam;
	FTRAPI_RESULT Rlt;
	float firstWeightsArr [ALL_WEIGHTS_LAYER_1];		// векторвесов 1 слоя
	float secondWeightsArr[ALL_WEIGHTS_LAYER_2];	// векторвесов 2 слоя
	float xorLink[ALL_XOR_INPUTS_1];		// векторсвязейвходовэлементов XOR
	CString sMessage;
	float percent = 0;
	CRect rc1;
	CRect xy;
	pObject->GetClientRect(rc1);
	xy.left = rc1.right*(255)/(315);
	xy.top  = rc1.bottom*(60)/(310);
	xy.right = rc1.right;
	xy.bottom = rc1.bottom;

	// Заполнениемассивоввесовыхкоэффициентов
	objCFileManip.tofileFullPath = objCFileManip.fullPath + "Data\\WeightTable.dat";
	objCFileManip.ReadWeightsFromFile(objCFileManip.tofileFullPath, firstWeightsArr,
secondWeightsArr, xorLink, sMessage);

	if (sMessage != "")	MessageBox(pObject->m_hWnd,sMessage,"Макет",
MB_ICONWARNING);
	else {
		int noError=256;
		int sNum = 0;
		int tNum=TEMPLPOINT;
		FPPoint *tMinutiae = new FPPoint [TEMPLPOINT];
		objCFileManip.ChangeImageTemplateMinutiae(objCFileManip.tofileFullPath);
		objCFileManip.ReadImageMinutiaeFile(objCFileManip.tofileFullPath,tMinutiae,
tNum);

		pObject->IFLAG=0;	// вывести изображение "Приложите палец"
		pObject->m_test_button.EnableWindow(0);
		pObject->Invalidate(0);

		while (noError==256) // поканевыполнитсяусловие
		{
			Rlt=FTRCaptureFrame( NULL, pObject->scanImg);

			if ((Rlt==FTR_RETCODE_OK)) {
```

```
pObject->IFLAG=1; // вывестиизображениеОП
pObject->Invalidate(0);
for (int i=0; i<320*480; i++)
	pObject->Img[i]=(int)pObject->scanImg[i];

// Проверкакорректногорасположения
percent=pObject->sobject.getSegment(pObject->Img,pObject-
>sImg,FpSizeImgWt,FpSizeImgHt);
if (percent > 30) {
	pObject->correctobj.IsCorrectImg(pObject->sImg);
	pObject->correctobj.getPositionX(pObject->correctPosX);
	pObject->correctobj.getPositionAngle(pObject-
>correctAngle);
}

FPPoint *sMinutiae = new FPPoint [300];
pObject->getMinutiaeFromImage(sNum, sMinutiae);

float Res=0;
float	dX=0;
float	dY=0;
float	dR=0;

// Определение смещения и преобразование
int h = pObject-
>templobject.MatchingSpeedTemplate(tMinutiae,TEMPLPOINT,sMinutiae,sNum,&Res,&dX,&d
Y,&dR);

pObject->templobject.AffineTransform(sMinutiae, sNum, dX*(-1),
dY*(-1), 0);

float *wacoeffs = new float [NCFOURIER_COEF];
pObject->fobject.WelshAdamar(sMinutiae, sNum, wacoeffs);

delete [] sMinutiae;
//Моделирование сети на полученных данных, вычисление количества несовпадений
int checkKeyArr[NEURON_NUMBER];
CSimNet objSimNet;
int	errorCount; // количествонесовпадений

errorCount = objSimNet.ImageRecognition (
	wacoeffs, firstWeightsArr,
	pObject->binKeyArr, checkKeyArr, xorLink
	);

delete [] wacoeffs;

// Считаемчислоошибок
errorCount=0;
for (int ii=0; ii<NEURON_NUMBER; ii++){
	if (checkKeyArr[ii]!=pObject->binKeyArr[ii])errorCount++;
}
```

```
                        // Вывести зеленый свет

                        if(errorCount == 0){
                                noError=0;
                                pObject->iLight = 0;
                                pObject->RedrawWindow(&xy);
                        }

                }

                else{

                        // Вывести красный свет
                        pObject->iLight = 1;
                        pObject->RedrawWindow(&xy);

                }

                // Закрытие окна аутентификации
                if ((Rlt==FTR_RETCODE_CANCELED_BY_USER)
&& (CancelCaptureOperation==TRUE))
                        break;

                // Выводкартинки "приложитепалец"
                if ((Rlt==FTR_RETCODE_CANCELED_BY_USER)
&& (CancelCaptureOperation==FALSE)) {
                        if (pObject->IFLAG!=0){
                                pObject->IFLAG=0;
                                pObject->Invalidate(0);
                        }
                }

            }
            FTRTerminate();
            delete [] tMinutiae;
        }

        GoodCaptureOperation=true;
        if(!CancelCaptureOperation) {
            pObject->m_test_button.EnableWindow(1);

        }

        return 0;
}

// Потоктестирования
UINT CaptureTesting (LPVOID pParam){

        CLearnDlg*          pObject=(CLearnDlg*)pParam;
        FTRAPI_RESULT Rlt;
```

```
	float firstWeightsArr [ALL_WEIGHTS_LAYER_1];		// векторвесов 1 слоя
	float secondWeightsArr[ALL_WEIGHTS_LAYER_2];	// векторвесов 2 слоя
	float xorLink[ALL_XOR_INPUTS_1];	// векторсвязейвходовэлементов XOR
	CString sMessage;
	float percent = 0;

	// Заполнение массивов весовых коэффициентов
	objCFileManip.tofileFullPath = objCFileManip.fullPath + "Data\\WeightTable.dat";
	objCFileManip.ReadWeightsFromFile(objCFileManip.tofileFullPath,firstWeightsArr,
secondWeightsArr, xorLink, sMessage);

	if (sMessage != "")	MessageBox(pObject->m_hWnd,sMessage,"Макет",
MB_ICONWARNING);
	else {
		int noError=256;
		int sNum = 0;
		int tNum=TEMPLPOINT;
		FPPoint *tMinutiae = new FPPoint [TEMPLPOINT];
		objCFileManip.ChangeImageTemplateMinutiae(objCFileManip.tofileFullPath);
		objCFileManip.ReadImageMinutiaeFile(objCFileManip.tofileFullPath,tMinutiae,
tNum);

		pObject->IFLAG=0;	// вывести изображение "Приложите палец"
		pObject->m_test_button.EnableWindow(0);
		pObject->Invalidate(0);

		while (noError==256) // поканевыполнитсяусловие
		{
			Rlt=FTRCaptureFrame( NULL, pObject->scanImg);

			if ((Rlt==FTR_RETCODE_OK)) {
				pObject->IFLAG=1; // вывестиизображениеОП
				pObject->Invalidate(0);
				for (int i=0; i<320*480; i++)
					pObject->Img[i]=(int)pObject->scanImg[i];

				// Проверкакорректногорасположения
				percent=pObject->sobject.getSegment(pObject->Img,pObject-
>sImg,FpSizeImgWt,FpSizeImgHt);
				if (percent > 30) {
					pObject->correctobj.IsCorrectImg(pObject->sImg);
					pObject->correctobj.getPositionX(pObject->correctPosX);
					pObject->correctobj.getPositionAngle(pObject-
>correctAngle);
				}

				FPPoint *sMinutiae = new FPPoint [300];
				pObject->getMinutiaeFromImage(sNum, sMinutiae);

				float Res=0;
				float	dX=0;
				float	dY=0;
```

```
                    float    dR=0;

                    // Определение смещения и преобразование
                    int h = pObject-
>templobject.MatchingSpeedTemplate(tMinutiae,TEMPLPOINT,sMinutiae,sNum,&Res,&dX,&d
Y,&dR);

                    pObject->templobject.AffineTransform(sMinutiae, sNum, dX*(-1),
dY*(-1), 0);

                    float *wacoeffs = new float [NCFOURIER_COEF];
                    pObject->fobject.WelshAdamar(sMinutiae, sNum, wacoeffs);

                    delete [] sMinutiae;
//Моделирование сети на полученных данных, вычисление количества несовпадений
                    int checkKeyArr[NEURON_NUMBER];
                    CSimNet objSimNet;
                    int   errorCount; // количествонесовпадений

                    errorCount = objSimNet.ImageRecognition (
                        wacoeffs, firstWeightsArr,
                        pObject->binKeyArr, checkKeyArr, xorLink
                        );

                    delete [] wacoeffs;

                    // Считаемчислоошибок
                    errorCount=0;
                    for (int ii=0; ii<NEURON_NUMBER; ii++){
                        if (checkKeyArr[ii]!=pObject->binKeyArr[ii])errorCount++;
                    }

                    // Вывести зеленый свет
                    // Вывод числа примеров
                    CString str;
                    str.Format(_T(" %d"),errorCount);
                    pObject->m_static_img_count.SetWindowText(str);

                    noError=0;

                }

                else{

                    // Вывести красный свет

                }

                // Закрытиеокнааутентификации
                if ((Rlt==FTR_RETCODE_CANCELED_BY_USER)
&& (CancelCaptureOperation==TRUE))
                    break;
```

```
                        // Выводкартинки "приложитепалец"
                        if ((Rlt==FTR_RETCODE_CANCELED_BY_USER)
&& (CancelCaptureOperation==FALSE)) {
                                if (pObject->IFLAG!=0){
                                        pObject->IFLAG=0;
                                        pObject->Invalidate(0);
                                }
                        }

                }
                FTRTerminate();
                delete [] tMinutiae;
        }

        GoodCaptureOperation=true;
        if(!CancelCaptureOperation) {
                pObject->m_test_button.EnableWindow(1);

        }

        return 0;
}

void CLearnDlg::OnClose()
{
        CancelCaptureOperation=true;
        if(CaptureThread!=NULL){
                WaitForSingleObject(CaptureThread->m_hThread,INFINITE);
                delete(CaptureThread);
                CaptureThread=NULL;}

        CDialog::OnClose();
        }

void CLearnDlg::OnBnClickedCancel()
{
        CancelCaptureOperation=true;
        if(CaptureThread!=NULL){
                WaitForSingleObject(CaptureThread->m_hThread,INFINITE);
                delete(CaptureThread);
                CaptureThread=NULL;}

        OnCancel();
}
BOOL CLearnDlg::OnEraseBkgnd(CDC* pDC)
{
        CDialog::OnEraseBkgnd(pDC);

        if (mode_dlg == AUT_MODE){
                CDC dcMem;
                CBitmap bmp1;
```

```
		CBitmap *pBmp1;
		BITMAP  bi;
		CClientDC dc(this);

		switch(iLight){
			case 0: bmp1.LoadBitmapA(IDB_BITMAP4); break;
			case 1: bmp1.LoadBitmapA(IDB_BITMAP5); break;
		}

		dcMem.CreateCompatibleDC(&dc);
		pBmp1= dcMem.SelectObject(&bmp1);
		bmp1.GetBitmap(&bi);

		CRect rc1;
		CRect xy;
		GetClientRect(rc1);
		xy.left = rc1.right*(255)/(315);
		xy.top  = rc1.bottom*(60)/(310);
		dc.TransparentBlt(xy.left, xy.top, bi.bmWidth, bi.bmHeight, &dcMem, 0, 0,
bi.bmWidth, bi.bmHeight, RGB(255,255,255));
		dcMem.SelectObject(pBmp1);
	}

	return TRUE;
}
```