

SAINT-PETERSBURG STATE UNIVERSITY

Mathematics&Mechanics Faculty

Department of Software Engineering

Borisova Albina

Fingerprint recognition by fuzzy logic approach

Bachelor's Thesis

Admitted for defence

Head of the chair:
professor A. Terehov

Scientific supervisor:
Senior Lecturer S. Sartasov

Reviewer:
D. Mordvinov

Saint-Petersburg

2014

САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

Математико-механический факультет
Кафедра системного программирования

Борисова Альбина Михайловна

Распознавание отпечатков пальцев методами нечеткой логики

Бакалаврская выпускная работа

Допущен к защите
зав. кафедрой
д.ф.-м.н., профессор Терехов А.Н.

Научный руководитель:
ст. преподаватель Сартасов С.Ю.

Рецензент:
Мордвинов Д.А.

Санкт-Петербург
2014

ОГЛАВЛЕНИЕ

Введение	4
Постановка задачи	6
Глава 1. Обзор существующих подходов	7
Биометрические системы	7
Нечеткая логика.....	9
Нечеткая логика в биометрических системах	10
Глава 2. Система для распознавания отпечатков пальцев	11
Измерение качества отпечатков пальцев	11
Алгоритм NFIQ	11
Выделение фона.....	12
Оценка яркости отпечатка	12
Оценка схожести двух отпечатков	13
Глава 3. Система принятия решения с помощью нечеткой логики	15
Фаззификация.....	15
Фаззификация схожести.....	15
Фаззификация качества.....	16
Применение правил.....	17
Вывод переменной качества	18
Вывод ответа для одного отпечатка	18
Вывод общего ответа для трех отпечатков	19
Глава 4. Анализ и сравнение.....	20
Виды ошибок распознавания	20
Сравнение систем.....	20
Тестовое множество	21
Заключение	22
Список литературы	23
Приложение	24

ВВЕДЕНИЕ

В настоящий момент информационные технологии проникли почти во все сферы жизни, поэтому проблема безопасности доступа к информационным системам и частной информации стала крайне актуальной. Классическая аутентификация пользователя зависит от паролей и ключей, которые могут быть легко утеряны или забыты. Также она не гарантирует, что паролем пользуется именно тот человек, которому предоставлен доступ. Эта проблема может быть решена путем использования биометрической аутентификации, т.е. аутентификации, которая проверяет личность пользователя на основе его физиологических или поведенческих характеристик, таких как черты лица, голос, отпечатки пальцев.

Общий алгоритм работы стандартной биометрической системы включает в себя извлечение биометрических признаков, например, снятие отпечатков пальцев или запись голоса. После этого производится их обработка и построение шаблонов. Таким образом, из признака выделяются отличительные черты. В зависимости от типа биометрической системы, верификации или идентификации, выбирается эталонный признак. В случае верификации он единственный, а в случае идентификации в качестве эталона берётся каждый признак из базы данных. Для шаблона и эталона вычисляется численная метрика схожести, которая уже сравнивается с пороговым значением. Так принимается решение о совпадении человека, которому принадлежат извлеченные признаки, с тем человеком, кому принадлежат эталонные [7]. Такая стандартная биометрическая система имеет ряд особенностей.

На каждом этапе работы системы происходит значительная потеря информации и само решение основывается только на численной метрике схожести.

Данные, полученные с сенсоров, зависят от шума, вызванного окружающей средой и физиологическими особенностями человека (чистота сенсора, освещенность, температура тела). Это приводит к тому, что принятие решения об идентичности признаков имеет вероятностную природу.

Таким образом, большая часть исследований в области биометрии направлены на увеличение точности и надежности систем.

Наиболее часто для повышения качества принятия решения применяются мультимодальные биометрические системы. Такие системы используют более одного биометрического признака, либо более одного алгоритма для расчета оценки схожести. Также можно учитывать исходное качество биометрического признака. Эти подходы

позволяют использовать большее количество факторов для принятия решения. При таком подходе из-за усложнения системы классическая бинарная логика и сравнение по порогу не дают идеального результата.

Вывод на основе нечеткой логики является одним из подходов к решению этой проблемы. Нечеткая логика – раздел математики, являющийся обобщением классической логики и теории множеств. Вывод в нечеткой логике основан на переменных, для которых определена не бинарная, а непрерывная принадлежность. Этот подход обеспечивает меньшую потерю информации при работе системы. Также он дает возможность тонкой и детальной настройки системы за счет изменения параметров нечеткого вывода. Это позволяет менять приоритет того или иного фактора, влияющего на решение.

ПОСТАНОВКА ЗАДАЧИ

Целью данной работы является создание прототипа биометрической системы для распознавания отпечатков пальцев, основанной на методах нечеткой логики, исследование качества данного алгоритма в сравнении с другими методами в данной области. Для достижения этих целей были сформулированы следующие задачи.

- Изучить предметную область систем распознавания отпечатков пальцев, а также основные принципы и понятия нечеткой логики.
- Создать алгоритм и прототип программной части мультимодальной системы для распознавания отпечатков пальцев.
- Создать подсистему принятия решений на основе нечеткой логики.
- Сравнить результаты распознавания с альтернативными стратегиями принятия решений.

ГЛАВА 1. ОБЗОР СУЩЕСТВУЮЩИХ ПОДХОДОВ

БИОМЕТРИЧЕСКИЕ СИСТЕМЫ

Биометрические системы делятся на два класса: мономодальные и мультимодальные. Стандартная мономодальная система строится по следующей схеме (Рис.1).

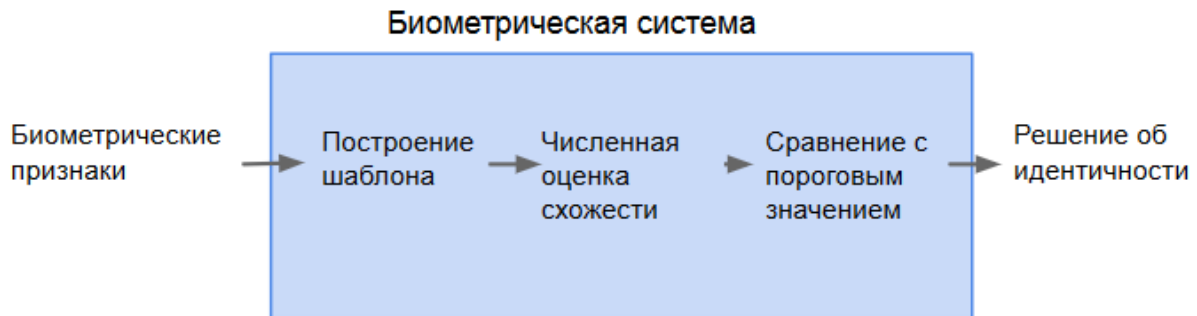


Рисунок 1. Схема работы мономодальной биометрической системы

На вход ей поступает один признак, полученный со сканера. Далее будем считать, что в качестве признаков выступают отпечатки пальцев, хотя эта схема применима ко всем биометрическим признакам. Сначала следует обработка отпечатка, с помощью которой повышается качество изображения. Далее по нему строится шаблон, обычно представляющий собой компактную структуру данных, которая отражает отличительные признаки отпечатка, и удобная для сравнения. Из базы данных выбирается шаблон эталонного отпечатка, и между ним и данным отпечатком вычисляется оценка схожести. Данная оценка сравнивается с некоторым пороговым значением, и принимается решение об идентичности полученного отпечатка и эталонного.

Для повышения точности применяются мультимодальные системы [11]. В мультимодальных системах применяется более одной модальности, то есть одного датчика (Рис. 2) и/или алгоритма (Рис. 3), в той или иной комбинации для принятия определенного решения в отношении биометрической идентификации или верификации. Признаки, используемые в мультимодальной системе, могут быть одинаковыми (несколько отпечатков одного пальца) или разными (отпечатки указательного и среднего пальца или же фотография лица и запись голоса).



Рисунок 2. Схема мультимодальной системы, использующей 2 биометрических признака

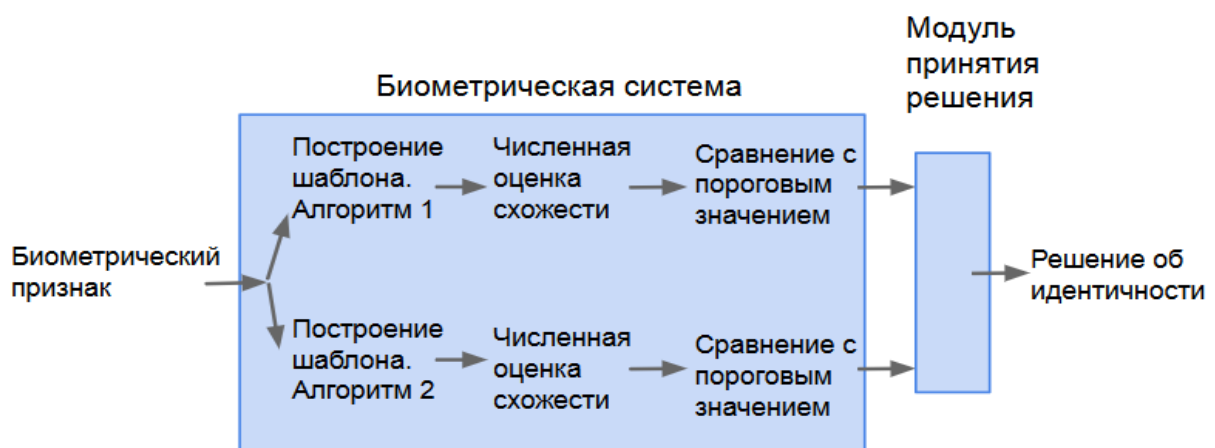


Рисунок 3. Схема мультимодальной системы, использующей 2 алгоритма построения шаблонов

Как видно из Рисунков 2 и 3, при увеличении количества модальностей возникает проблема принятия решения. В случае с несколькими модулями и, соответственно, несколькими ответами есть следующие стратегии принятия решений.

- Сравнение по порогу и вынесение положительного решения, если все модули ответили положительно. Этот случай характеризуется меньшей вероятностью ложного допуска и большей – ложного отказа в допуске.
- Сравнение по порогу и голосование простым большинством. Этот случай, напротив, повышает вероятность ложного допуска и понижает – ложного отказа в допуске.
- Вывод решения с помощью нечеткой логики. Этот способ позволяет детально настроить процесс принятия решения и дает возможность учитывать большее количество факторов.

НЕЧЕТКАЯ ЛОГИКА

Нечеткая логика обобщает теорию множеств и классическую логику. Понятие нечеткого множества введено впервые Лотфи Заде в 1965 году [1]. Нечёткое множество A – совокупность упорядоченных пар, составленных из элементов x универсального множества X и соответствующих степеней принадлежности $\mu_A(x)$.

$$A = \{(x, \mu_A(x)) | x \in X\}$$

Функция принадлежности $\mu_A(x)$ указывает, в какой степени элемент x принадлежит множеству A . Множество значений этой функции – интервал $[0,1]$. В случае множества значений $\{0,1\}$, нечеткое множество является обычным множеством.

В отличие от классических множеств над нечеткими множествами операции объединения и пересечения определяются тремя разными способами.

- Максимальные

$$\mu_{A \cup B}(x) = \min(\mu_A(x), \mu_B(x))$$

$$\mu_{A \cap B}(x) = \max(\mu_A(x), \mu_B(x))$$

- Алгебраические

$$\mu_{A \cup B}(x) = \mu_A(x) + \mu_B(x) - \mu_A(x) * \mu_B(x)$$

$$\mu_{A \cap B}(x) = \mu_A(x) * \mu_B(x)$$

- Ограниченные

$$\mu_{A \cup B}(x) = \min(1, \mu_A(x) + \mu_B(x))$$

$$\mu_{A \cap B}(x) = \min(0, \mu_A(x) + \mu_B(x) - 1)$$

Вывод в нечеткой логике основывается на лингвистических переменных [2]. Это переменные, значениями которых являются фразы некоторого языка. Например, переменная "Качество" может принимать значения "Высокое", "Среднее", "Низкое". Каждое значение лингвистической переменной является нечетким множеством. Например, Качество может быть Высоким на 0,8 и Низким на 0,1.

Вывод проходит в три этапа.

- Фаззификация (fuzzification)

Численные значения переводятся в значения лингвистических переменных с помощью применения соответствующих функций принадлежности.

- Применение правил

Правила имеют стандартный вид “если A , то B ” и используют лингвистические переменные. Сначала вычисляется степень принадлежности посылки. Для этого используется один из трех видов операций над множествами. После этого значению переменной в заключении присваивается вычисленная степень принадлежности.

- Дефаззификация (defuzzification)

Из выведенных переменных получают численные значения и результаты.

НЕЧЕТКАЯ ЛОГИКА В БИОМЕТРИЧЕСКИХ СИСТЕМАХ

К сожалению, применение нечеткой логики в биометрических системах слабо представлена в текущих исследованиях.

В статье Fuzzy Fusion in Multimodal Biometric Systems [4] профессор V.Conti с соавторами описывает мультимодальную биометрическую систему для распознавания отпечатков пальцев, использующую отпечатки указательного и среднего пальца. В качестве дополнительных факторов при принятии решения об идентичности используется качество отпечатка пальца. Однако отсутствие правил вывода нечеткой логики и описания методов оценки качества делают результаты не верифицируемыми.

Статья Fuzzy Logic Decision Fusion in a Multimodal Biometric System [3] описывает мультимодальную биометрическую систему, в которой используется запись голоса, фотография лица и отпечаток пальца. Нечеткая логика в ней используется не непосредственно для принятия решения, а для вычисления коэффициентов, с которыми используется мера схожести отпечатков и фотографий.

В данной работе была разработана мультимодальная биометрическая система с нечетким выводом, использующая три отпечатка пальца, где качество отпечатка оценивалось несколькими разными алгоритмами, в том числе и стандартным NFIQ, в которой были учтены недостатки работ [3] и [4].

ГЛАВА 2. СИСТЕМА ДЛЯ РАСПОЗНАВАНИЯ ОТПЕЧАТКОВ ПАЛЬЦЕВ

Для реализации мультибиометрической системы была выбрана система, работающая с тремя отпечатками пальцев. Из биометрических признаков выбраны отпечатки пальцев, поскольку они просты в использовании и дают надежные результаты. В системе используется именно три отпечатка, так как меньшее количество биометрических признаков не даёт в полной мере возможность испытывать разные стратегии принятия решения, большее же количество замедляет работу системы, но не даёт содержательного отличия от трех отпечатков.

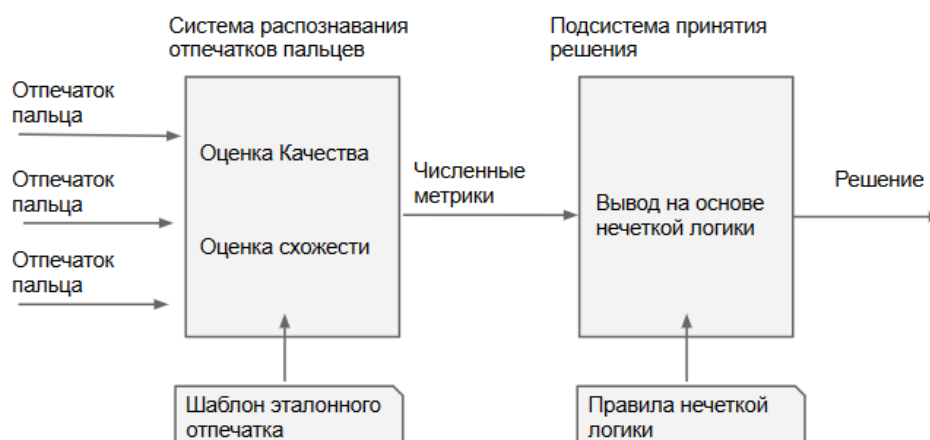


Рисунок 4. Архитектура мультимодальной биометрической системы

Вся система состоит из двух основных частей: системы распознавания отпечатков пальцев и подсистема принятия решения. Система распознавания состоит из независимых модулей: модуля оценки схожести двух отпечатков и модуля оценки качества изначального отпечатка. Система была реализована на языке C#, платформа .NET Framework 4.5.

ИЗМЕРЕНИЕ КАЧЕСТВА ОТПЕЧАТКОВ ПАЛЬЦЕВ

Для получения более точной оценки качества применяются три алгоритма, которые измеряют качество отпечатка пальца по разным критериям.

АЛГОРИТМ NFIQ

NIST Fingerprint Image Quality [5] является стандартным и широко используемым алгоритмом для оценки качества отпечатков пальцев. Он разработан в Национальном Институте Стандартов и Технологий. При оценке качества он учитывает направление папиллярных линий, контраст отпечатка и кривизну папиллярных линий. Изображение разбивается на блоки и качество оценивается независимо в каждом блоке. На выходе

алгоритма получается карта качества, которая содержит оценку от 1 до 5 для каждого блока отпечатка. Из нее можно получить две численные оценки – процент блоков низкого качества и среднее качество всего отпечатка. На рисунке 5 представлены отпечатки высокого и низкого качества по оценке алгоритма NFIQ. Для реализации данного алгоритма была использована библиотека NFIQ. [10]



Рисунок 5 Отпечатки пальца высокого и низкого качества по оценке алгоритма NFIQ

ВЫДЕЛЕНИЕ ФОНА

Также проводится выделение фона – части изображения без папиллярных линий. На рисунке 6 показан отпечаток с большой частью фона. Данный алгоритм основан на сравнении определенности направления (того, насколько выражены папиллярные линии) в данном блоке и в среднем по всему отпечатку. На выходе алгоритма получается метрика, показывающая процент фона на всем изображении.

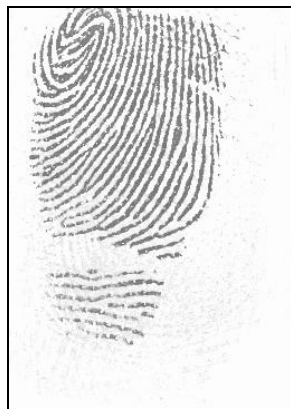


Рисунок 6 Отпечаток пальца с большим количеством фона

ОЦЕНКА ЯРКОСТИ ОТПЕЧАТКА

Отпечатки с низкой яркостью получаются если, палец, который сканируют, был слишком влажным. Это приводит к тому что папиллярные линии сливаются вместе (Рис. 7). В отличие от темных, светлые отпечатки обычно не являются некачественными, и на них читаются папиллярные линии. Оценка строится подсчетом тех пикселей, чья яркость ниже определенного порога. Процентное соотношение таких пикселей ко всем пикселям отпечатка

определяет метрику качества.

Определение яркости и фона дополняют оценку качества NFIQ, так как в нем эти параметры не учитываются.



Рисунок 7 Отпечаток пальца со сливающимися папиллярными линиями

Параметры методов определения качества отпечатков были подобраны эмпирически. Таким образом были определены порог яркости пикселей, которые считались темными, коэффициент сравнения определенности направления при оценке фона и другие параметры.

ОЦЕНКА СХОЖЕСТИ ДВУХ ОТПЕЧАТКОВ

Второй модуль системы распознавания отпечатков оценивает то, с какой вероятностью два отпечатка пальца являются отпечатками одного и того же пальца человека.

Перед определением схожести проводится улучшение качества изображения отпечатка пальца и приведение его к виду, удобному для последующей работы.

Оно проходит в три этапа.

На первом этапе – сегментации – происходит определение фона отпечатка и его краев на изображении. Далее эти области не будут включаться в рассмотрение, так как алгоритмы, заточенные на работу с отпечатками – изображениями определенного вида и структуры – дают большую ошибку на других видах изображений.

На втором этапе улучшения изображения проводится его бинаризация: изображение в градациях серого приводится к черно-белому.

Наконец, на третьем этапе происходит утончение папиллярных линий так, чтобы они были шириной в один пиксель.

После улучшения изображения проводится выделение минуций. Минуция – это особая точка на отпечатке пальца, на которой папиллярная линия раздваивается или заканчивается. Именно взаимное расположение минуций и их вид является уникальным для каждого человека [7]. Для реализации выделения минуций были использованы библиотеки CUDAFingerprint [6].

Далее по набору минуций строится шаблон – компактная структура данных, удобная для сравнения. Алгоритм построения шаблонов Minutia Cylinder Code (MCC) [8] для каждой минуции строит "цилиндр", который описывает положение ближайших минуций относительно данной. Мерой схожести двух отпечатков выступает сумма расстояний Хемминга между "цилиндрами". MCC разработан в Лаборатории Биометрических Систем Болонского университета. Данный алгоритм устойчив к поворотам отпечатка, его сдвигам и растяжению. Для его реализации была использована библиотека MCC SDK[9]. Параметры подсчета схожести в алгоритме MCC и другие параметры системы были определены эмпирически.

ГЛАВА 3. СИСТЕМА ПРИНЯТИЯ РЕШЕНИЯ С ПОМОЩЬЮ НЕЧЕТКОЙ ЛОГИКИ

Подсистема принятия решений принимает на вход пять численных метрик для каждого из трех отпечатков: четыре метрики качества и одна метрика схожести с эталоном.

ФАЗЗИФИКАЦИЯ

Как говорилось выше, нечеткий вывод проводится в три этапа и первый из них – фаззификация. Приведение численных метрик в значения лингвистических переменных проводится с помощью соответствующих функций принадлежности.

ФАЗЗИФИКАЦИЯ СХОЖЕСТИ

При фаззификации, то есть получении лингвистических переменных, метрики схожести результатом является переменная Identity со значениями Same и Different. Оба эти значения являются нечеткими множествами. Их функции принадлежности представлены на графике (Рис. 8)

Метрика схожести, получаемая от алгоритма МСС, колеблется от 0 до 1, где 1 обозначает абсолютно одинаковые отпечатки, а значения, близкие к 0, – разные. Эта метрика берется в качестве аргумента функций принадлежности, и в результате вычисления получают степени принадлежности для соответствующих переменных.

Например, если ответ МСС равен 0.6, то степень вхождения Identity Same будет 0.88, а Identity Different – 0.12.

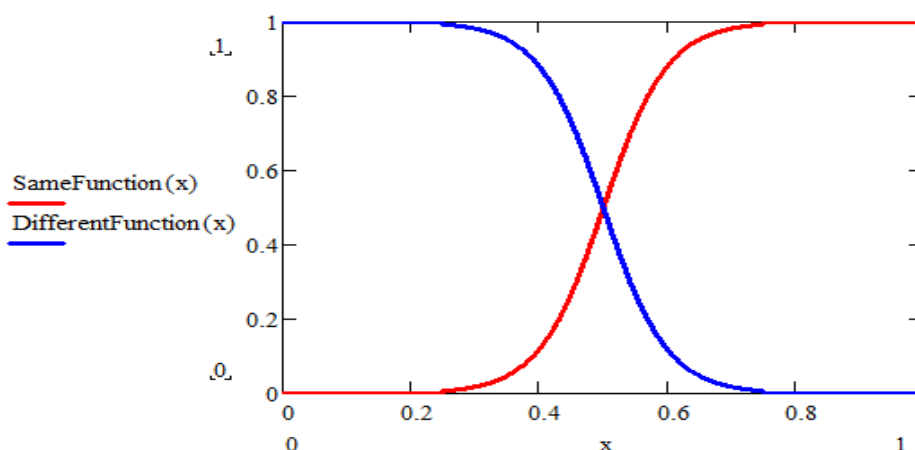


Рисунок 8 Функции принадлежности переменной Identity

Сдвиг данного графика (Рис. 8) семантически соответствует изменению порога сравнения в классических системах распознавания.

ФАЗЗИФИКАЦИЯ КАЧЕСТВА

Аналогично тому, как проводилась фаззификация качества, лингвистические переменные выводятся для метрик качества. Далее для каждой из них будет описана область допустимых значений, получаемая переменная и функции принадлежности ее значений.

Метрика среднего качества

Метрика среднего качества алгоритма NFIQ меняется в пределах от 0 до 4, где 0 – низкое качество, 4 – самое высокое.

Получаемая переменная – QualityNfiq со значениями Low, High.

Функции принадлежности:

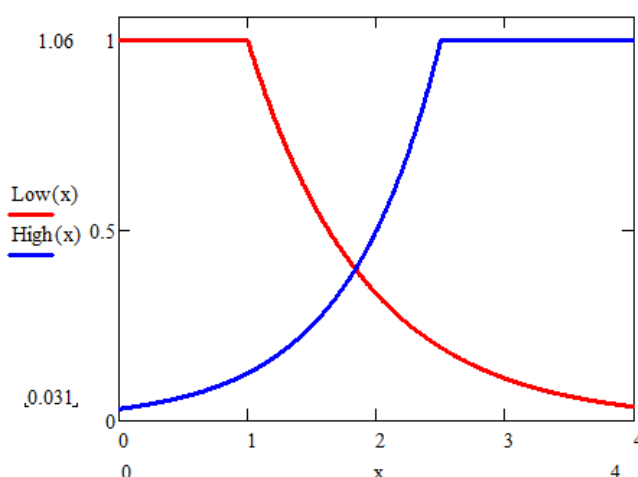


Рисунок 9 Функции принадлежности переменной QualityNfiq

Метрика процента блоков низкого качества

Метрика количества блоков низкого качества по алгоритму NFIQ колеблется от 0 до 100. Блоками низкого качества считаются те, у которых оценка NFIQ 0 или 1.

Получаемая переменная – LowQualityBlocks со значениями Little и Many.

Функции принадлежности:

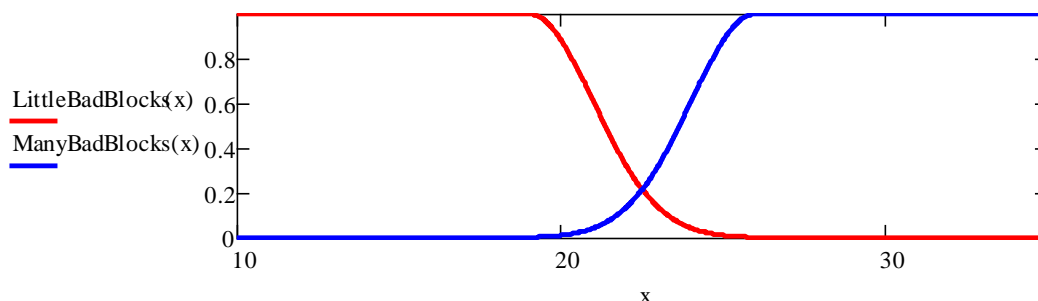


Рисунок 10 Функции принадлежности переменной LowQualityBlocks

График функций построен для значений от 10 до 40. Вне рассматриваемого диапазона значения функций не меняются.

Метрика яркости

Метрика яркости отпечатка также меняется в пределах от 0 до 100.

При этом значения, близкие к 0, характеризуют нормальный отпечаток, а большие – темный.

Получаемая переменная – Brightness со значениями High и low

Функции принадлежности:

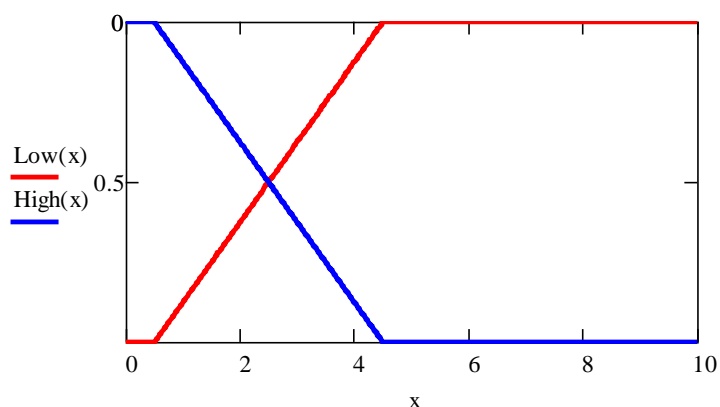


Рисунок 11 Функции принадлежности переменной Brightness

График функций построен для значений от 0 до 10. Вне рассматриваемого диапазона значения функций не меняются.

Метрика процента фона

Метрика процента фона колеблется между 0 и 100.

Получаемая переменная Background со значениями Large и Normal.

Функции принадлежности:

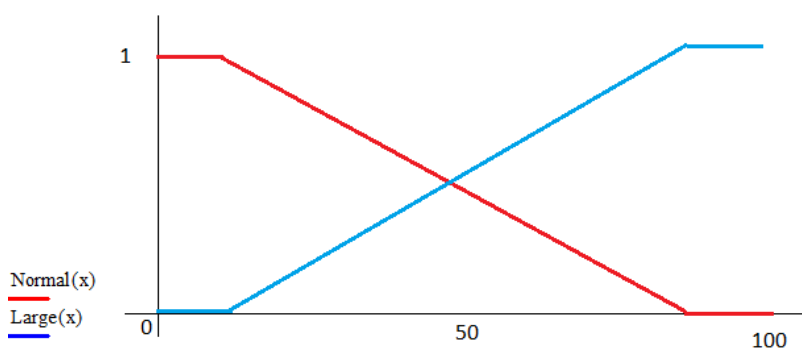


Рисунок 12 Функции принадлежности переменной Background

Вид данных функций принадлежности был определен эмпирически.

ПРИМЕНЕНИЕ ПРАВИЛ

Следующий этап нечеткого вывода – применение правил. Перед началом этого этапа правила считываются из файла. На вход подаются лингвистические переменные, полученные

при фаззификации.

Так как одновременно существуют все значения лингвистической переменной, вывод идет по всем правилам, где она встречается в посылке. Для вычисления степени принадлежности посылки используются максимумные операции. Далее, если значение переменной в заключении еще не существует, оно создается со степенью принадлежности равной, степени принадлежности посылки. Если оно существует, то степень принадлежности определяется как максимум из существующей и выведенной.

ВЫВОД ПЕРЕМЕННОЙ КАЧЕСТВА

Сначала выводится переменная Quality со значениями Low, Middle и High. Она показывает общее значение качества для отпечатка. Так как правил достаточно много, будет описан принцип их построения. Полный перечень правил можно увидеть в приложении.

Правила вывода переменной качества строились по следующему принципу.

- Если все значения в посылке высокие, то значение качества высокое.
- Если одно из значений в посылке низкое, то качество среднее.
- Если более одного низкого значения – качество низкое.

Пример правила:

Brightness high, Background normal, QualityNfiq high, LowQualityBlocks little => Quality high

Запятая в данном случае означает логическое "И". То есть в данном случае это правило означает следующее. Если у изображения высокая яркость, небольшое количество фона, качество по оценке алгоритма NFIQ высокое и блоков низкого качества мало, то его качество высокое.

ВЫВОД ОТВЕТА ДЛЯ ОДНОГО ОТПЕЧАТКА

В этой части вывода используются переменная качества и переменная схожести и правила строятся по следующему принципу.

- Если качество высокое, то ответ совпадает с переменной схожести
- Если качество среднее, ответ с коэффициентом $\frac{1}{2}$ совпадает с переменной схожести и с коэффициентом $\frac{1}{2}$ неизвестен
- Если качество низкое, ответ неизвестен

Примеры правил:

quality high, identity different => fingerprintAnswer no

quality low, identity same => fingerprintAnswer idk

(idk = I don't know)

Вывод общего ответа для трех отпечатков

В последней части правила применяются к ответам от трех отпечатков пальцев. Принцип построения правил можно описать как обобщение голосования простым большинством на три ответа. То есть, если ответы Yes, No и Idk представить соответственно как 1, -1 и 0, то общий ответ будет получаться усреднением ответов от трех отпечатков и округлением в сторону увеличения модуля.

Примеры правил:

f1 yes, f2 yes, f3 idk => answer yes

f1 yes, f2 yes, f3 no => answer yes

f1 yes, f2 no, f3 idk => answer idk

Получив общий ответ, мы дефаззифицируем его и в результате получаем ответ с наибольшей степенью принадлежности.

ГЛАВА 4. АНАЛИЗ И СРАВНЕНИЕ

ВИДЫ ОШИБОК РАСПОЗНАВАНИЯ

В связи с вероятностной природой принятия решения об идентичности, не всегда ответ биометрической системы для идентичных отпечатков бывает положительным, а для различных – отрицательным. Из-за этого возникают ошибки классификации: соответственно, вероятность ложного отказа в доступе (False Rejection Rate, FRR), и вероятность ложного допуска (False Acceptance Rate, FAR). При сравнении по порогу, назначение высокого порога приводит не только к снижению вероятности ложного допуска, но и к возрастанию вероятности ложного отказа в доступе. Если же выставлен низкий порог, наоборот, растет вероятность ложного допуска и уменьшается ложного отказа в доступе. Поэтому точность системы оценивается коэффициентом равной вероятности ошибок (Equal Error Rate, EER) – точка, где $FAR=FRR$ (Рис. 13). В идеальной ситуации графики FAR и FRR не пересекаются, и значение ERR равно нулю [7].

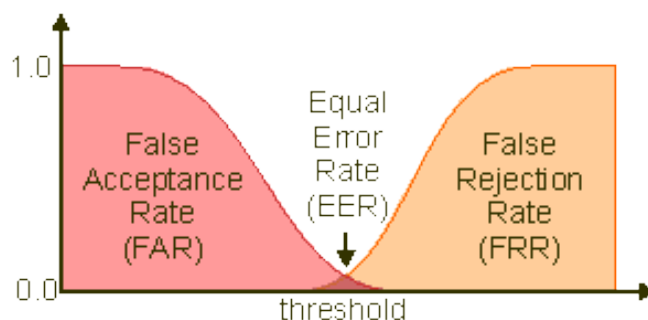


Рисунок 13: Ошибки FAR, FRR и EER

СРАВНЕНИЕ СИСТЕМ

Результаты распознавания системы с принятием решения на основе нечеткой логики сравнивались с результатами распознавания схожих систем, каждая из которых отличалась от данной только блоком принятия решения. Это было сделано для того, чтобы исключить влияние других факторов, не связанных с системой принятия решения. Они описаны ниже.

1. Мономодальная система со сравнением по порогу.
2. Мультиформальная система со сравнением по порогу. Положительный ответ системы выносится, если ответ для каждого из трех отпечатков положительный.

3. Мультимодальная система со сравнением по порогу. Ответ системы определяется голосованием простым большинством.

Таблица 1 Сравнение ERR и порогов систем

	Система с нечеткой логикой	Мономодальная система	Мультимодальная система с голосованием	Мультимодальная система с ответом три из трех
ERR	17.81%	30.82%	29.33%	28.21%
Порог	0.525	0.545	0.545	0.53

В качестве порога системы с распознаванием на основе нечеткой логики была взята точка пересечения функций принадлежности значений Identity Same и Identity Different.

ТЕСТОВОЕ МНОЖЕСТВО

Для тестирования использовалась открытая база данных FVC2000, DB2 набор A. Она состоит из 800 отпечатков, по 8 отпечатков на один палец. Каждый отпечаток представлен в виде изображения в формате .tif с размером 364 x 256 пикселей и разрешением 500 dpi. Стоит отметить, что FVC2000 была построена для тестирования систем распознавания отпечатков пальцев и содержит много отпечатков низкого качества. Для каждого пальца эталонным считался отпечаток лучшего качества по NFIQ из данных восьми. При вычислении вероятности ложного отказа в доступе из оставшихся семи отпечатков выбирались тройки отпечатков. Они и подавались на вход системе. Для вычисления вероятности ложного доступа тройки отпечатков выбирались случайно из всей базы данных, исключая отпечатки данного пальца.

Значения качества отпечатков и шаблоны MCC кэшировались для уменьшения времени работы системы.

ЗАКЛЮЧЕНИЕ

В ходе выполнения работы были достигнуты следующие результаты.

- Изучена предметная область систем распознавания отпечатков пальцев, а также основные принципы нечеткой логики.
- Создан прототип платформы для принятия решений на основе нечеткой логики (Трехфазный алгоритм, включающий учет качества и подходящий для мультибиометрической системы).
- Создан алгоритм и прототип программной части мультимодальной системы для распознавания отпечатков пальцев, использующий систему принятия решений на основе нечеткой логики, алгоритм распознавания минуций МСС и алгоритм оценки качества отпечатков NFIQ.
- Проведено сравнение результатов с альтернативными стратегиями принятия решений, показавший преимущества алгоритма, основанного на нечеткой логике в сравнении с другими.

Исходный код системы можно найти на сайте <https://code.google.com/p/fingerprint-fuzzy-logic/>

СПИСОК ЛИТЕРАТУРЫ

- [1] *Заде Л.* Понятие лингвистической переменной и его применение к принятию приближенных решений, 1976.
- [2] *Штовба С.Д.* Введение в теорию нечетких множеств и нечеткую логику. 2003.
- [3] *Chun Wai Lau, Bin Ma, Helen M. Meng, Y.S. Moon and Yeung Yam,* Fuzzy Logic Decision Fusion in a Multimodal Biometric System, 2004
- [4] *Conti V., Milici G., Ribino P., Sorbello F. and Vitabile S.,* Fuzzy Fusion in Multimodal Biometric Systems
- [5] *Craig I. Watson, Michael D. Garris, Elham Tabassi, Charles L. Wilson, R. Michael McCabe,* User's Guide to NIST Biometric Image Software, 2004
- [6] Cuda-fingerprinting. A PhD research on two-stage fingerprint matching and CUDA usage
<https://code.google.com/p/cuda-fingerprinting/>
- [7] *Maltoni D., Maio D., Jain A. K., Prabhakar S.* Handbook of Fingerprint Recognition, second edition, 2009.
- [8] Minutia Cylinder-Code
<http://biolab.csr.unibo.it/research.asp?organize=Activities&select=&selObj=81&pathSubj=111||8||81&Req=&>
- [9] Minutia Cylinder-Code SDK
<http://biolab.csr.unibo.it/research.asp?organize=Activities&select=&selObj=82&pathSubj=111||8||82&Req=&>
- [10] NIST BIOMETRIC IMAGE SOFTWARE
<http://www.nist.gov/itl/iad/ig/nbis.cfm>
- [11] *Ross, Arun A., Nandakumar, Karthik, Jain, Anil K.,* Handbook of Multibiometrics, 2006

ПРИЛОЖЕНИЕ

Quality rules

brightness high, background normal, QualityNfiq high, LowQualityBlocks little = quality high

brightness low, background normal, QualityNfiq high, LowQualityBlocks little = quality middle

brightness high, background large, QualityNfiq high, LowQualityBlocks little = quality middle

brightness high, background normal, QualityNfiq low, LowQualityBlocks little = quality middle

brightness high, background normal, QualityNfiq high, LowQualityBlocks many = quality middle

brightness low, background large = quality low

background large, QualityNfiq low = quality low

QualityNfiq low, LowQualityBlocks many = quality low

brightness low, QualityNfiq low = quality low

brightness low, LowQualityBlocks many = quality low

background large, LowQualityBlocks many = quality low

QualityNfiq low, LowQualityBlocks many = quality low

Fingerprint rules

quality high, identity same = fingerprintAnswer yes

quality high, identity different = fingerprintAnswer no

quality middle, identity same = fingerprintAnswer yes * 0,5

quality middle, identity same = fingerprintAnswer idk * 0,5

quality middle, identity different = fingerprintAnswer no * 0,5

quality middle, identity different = fingerprintAnswer idk * 0,5

quality low, identity same = fingerprintAnswer idk

quality low, identity different = fingerprintAnswer idk

Answer rules

fingerprintAnswer1 yes, fingerprintAnswer2 yes, fingerprintAnswer3 yes = answer yes

fingerprintAnswer1 idk, fingerprintAnswer2 idk, fingerprintAnswer3 idk = answer idk

fingerprintAnswer1 no, fingerprintAnswer2 no, fingerprintAnswer3 no = answer no

fingerprintAnswer1 yes, fingerprintAnswer2 yes, fingerprintAnswer3 idk = answer yes

fingerprintAnswer1 yes, fingerprintAnswer2 idk, fingerprintAnswer3 yes = answer yes

fingerprintAnswer1 idk, fingerprintAnswer2 yes, fingerprintAnswer3 yes = answer yes

fingerprintAnswer1 no, fingerprintAnswer2 no, fingerprintAnswer3 idk = answer no

fingerprintAnswer1 no, fingerprintAnswer2 idk, fingerprintAnswer3 no = answer no

fingerprintAnswer1 idk, fingerprintAnswer2 no, fingerprintAnswer3 no = answer no

fingerprintAnswer1 idk, fingerprintAnswer2 idk, fingerprintAnswer3 no = answer no

fingerprintAnswer1 idk, fingerprintAnswer2 no, fingerprintAnswer3 idk = answer no

fingerprintAnswer1 no, fingerprintAnswer2 idk, fingerprintAnswer3 idk = answer no

fingerprintAnswer1 idk, fingerprintAnswer2 idk, fingerprintAnswer3 yes = answer yes

fingerprintAnswer1 idk, fingerprintAnswer2 yes, fingerprintAnswer3 idk = answer yes

fingerprintAnswer1 yes, fingerprintAnswer2 idk, fingerprintAnswer3 idk = answer yes

fingerprintAnswer1 no, fingerprintAnswer2 no, fingerprintAnswer3 yes = answer no

fingerprintAnswer1 no, fingerprintAnswer2 yes, fingerprintAnswer3 no = answer no

fingerprintAnswer1 yes, fingerprintAnswer2 no, fingerprintAnswer3 no = answer no

fingerprintAnswer1 yes, fingerprintAnswer2 yes, fingerprintAnswer3 no = answer yes

fingerprintAnswer1 yes, fingerprintAnswer2 no, fingerprintAnswer3 yes = answer yes

fingerprintAnswer1 no, fingerprintAnswer2 yes, fingerprintAnswer3 yes = answer yes

fingerprintAnswer1 yes, fingerprintAnswer2 no, fingerprintAnswer3 idk = answer idk

fingerprintAnswer1 yes, fingerprintAnswer2 idk, fingerprintAnswer3 no = answer idk

fingerprintAnswer1 no, fingerprintAnswer2 yes, fingerprintAnswer3 idk = answer idk

fingerprintAnswer1 no, fingerprintAnswer2 idk, fingerprintAnswer3 yes = answer idk

fingerprintAnswer1 idk, fingerprintAnswer2 no, fingerprintAnswer3 yes = answer idk

fingerprintAnswer1 idk, fingerprintAnswer2 yes, fingerprintAnswer3 no = answer idk