

Яремко Олег Ігорович, Ткачук Василь Геннадійович

– Розробка та дослідження автоматизованої системи для підвищення якості ідентифікації при дактилоскопії (комплексна тема).

Керівник: кандидат технічних наук, зав. кафедри КТ Микитишин Андрій Григорович.

АНОТАЦІЯ

Дипломна робота складається з пояснювальної записки та графічної частини (ілюстративний матеріал – слайди).

Об'єм графічної частини дипломної роботи становить ___ слайдів.

Об'єм пояснювальної записки складає ___ друкованих сторінок формату А4 (210×297), об'єм додатків – ___ друкованих сторінок формату А4.

Дипломна робота складається з шести розділів, в яких нараховується ___ рисунків та ___ таблиць з даними.

В роботі використано ___ літературних джерел.

В результаті проведеної роботи проаналізовано основні параметри та фактори, що впливають на якість ідентифікації під час дактилоскопії. Метою було створити прототип системи розпізнавання біо-відбитків пальців на основі нечіткої логіки та вивчити якість цього алгоритму порівняно з іншими методами в цій галузі.

Ключові слова: ідентифікація, біометрична система, нечітка логіка, відбитки пальців.

ЗМІСТ

ВСТУП	6
1. АНАЛІТИЧНА ЧАСТИНА	10
1.1. Біометричні системи	10
1.2 Нечітка логіка	12
2. ТЕХНОЛОГІЧНА ЧАСТИНА	15
2.1 Система розпізнавання відбитків пальців	15
2.2 Вимір якості відбитків пальців	16
2.3 Оцінка якості відбитку	17
2.4 Оцінка сходження двох відбитків	18
3 КОНСТРУКТОРСЬКА ЧАСТИНА	20
3.1. Система підтримки прийняття рішень з використанням нечіткої логіки	20
3.2 Застосування правил	24
3.3 Аналіз і порівняння	26
4 НАУКОВО-ДОСЛІДНИЦЬКА	29
4.1 Детектор кордонів Кенні	29
4.2 Подвійна фільтрація	31
4.3 Метод Віюли-Джонса	33
4.3 Ознаки Хаара	36
5. СПЕЦІАЛЬНА ЧАСТИНА	41
5.1. Алгоритм NFIQ	41
5.2 Помилки FMR, FNMR та EER.	42
5.3 Метод визначення якісних областей відбитків пальців	44
5.4 Зіставлення відбитків пальців за допомогою MCC SDK	49
6 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ	53
6.1 Організація охорони праці при роботі з системою управління	53
6.2 Електробезпека	55
6.3 Розрахунок заземлення	58
ОСНОВНІ ВИСНОВКИ ДИПЛОМНОЇ РОБОТИ	62
БІБЛІОГРАФІЯ	63
Додаток	65

ВСТУП

У сучасному середовищі не обійтися без інформаційних технологій, тому при користуванні за обладнанням багатьох користувачів виникає потреба у захисті інформації шляхом ідентифікації особи.

У більшості випадків доступ до системи відбувається за допомогою логіна на пароля. Але перевірка даних користувача тільки даним методом не є гарантією, що ця людина є власником інформації. Пароль міг бути здобутий не законним методом.

Цю проблему можна вирішити за допомогою біометричного методу. сертифікація. Підтвердьте особу користувача на основі фізіологічних даних. Особистість кожної людини - це райдужний відбиток пальця, голос і сітківка.

Загальний алгоритм роботи стандартної біометричної системи включає вилучення біометричних ознак, наприклад, зняття відбитків пальців або запис голосу. Після цього проводиться їх обробка та побудова шаблонів. Таким чином, із ознаки виділяються відмінні риси. Залежно від типу біометричної системи, верифікації чи ідентифікації, вибирається еталонна ознака. У разі верифікації він єдиний, а у разі ідентифікації як зразок береться кожна ознака з бази даних. Для шаблону та зразка обчислюється чисельна метрика схожості, яка порівнюється з пороговим значенням. Так приймається рішення про збіг людини, якій належать одержані ознаки, з тим людиною, кому належать еталонні [7]. Така стандартна біометрична система має низку особливостей.

На кожному етапі роботи системи відбувається значна втрата інформації і саме рішення ґрунтується лише на чисельній метриці схожості.

Дані, отримані з сенсорів, залежить від шуму, викликаного довкіллям і фізіологічними особливостями людини (чистота сенсора, освітленість, температура тіла). Це призводить до того, що ухвалення рішення про ідентичність ознак має ймовірну природу.

Таким чином, більшість досліджень у галузі біометрії спрямовані на збільшення точності та надійності систем.

Для покращення та збільшення точності прийнятого рішення потрібно використовувати біометричні системи які враховують декілька факторів. Дані фактори притаманні для систем що містять більше однієї біометричної ознаки, або більше одного алгоритму для розрахунку оцінки схожості. Також можна враховувати вихідну якість біометричної ознаки. Ці підходи мають на меті використання значної кількості факторів для ухвалення рішення. Такий підхід ускладнює стандартну систему та не дає ідеального результату.

Можна зробити висновок, що визначення на основі нечіткої логіки є одним із підходів до вирішення цієї проблеми. Нечітка логіка - розділ математики, що є узагальненням класичної логіки та теорії множин. Загальний висновок у нечіткій логіці ґрунтується на змінних, котрим визначено не бінарна, а безперервна приналежність. Цей напрямок роботи забезпечує більш надійну обробку інформації. Також він дає можливість більш якісного налагодження компонентів за даними зміни параметрів нечіткого виведення. І також визначати параметри та важливість інших факторів, що в подальшому робить заключення.

Ідентифікація за відбитками пальців є найпоширенішим біометричним способом визначення особистості. Відбитки пальців все ширше застосовують у криміналістиці, в пропускних системах аеропортів тощо. завдяки своїй унікальності та незмінності.

Однак завжди існує можливість отримати неякісне зображення відбитка. Причини можуть бути різні: не нормальний епітелій шкірного покриття, рани на шкірі або несамовитий рух пальця під час зняття відбитка (рис. 1).



Рис. 1: Приклади відбитків пальців

Таким чином, після розпізнавання відбитка може виявитися неуспішним зіставлення ще з одним відбитком цього ж пальця. Або навпаки такий відбиток може збігтися з відбитком іншої людини. Також використання неякісних зображень відбитків знижує продуктивність біометричних систем.

Щоб уникнути подібних проблем заздалегідь, розроблено метод опису якості. Так, перевірте якість перед використанням відбитка пальця, щоб ідентифікувати особу. Наприклад, в алгоритмі NFIQ (NIST Fingerprint Image Quality) кожне зображення відбитка пальця характеризується значенням від 1 (висока якість) до 5 (низька якість) [1]. Існує також більш специфічний алгоритм LFIQ (Latent Fingerprint Image Quality), який спеціалізується на визначенні якості друкованої продукції. Частіше використовується в судовій медицині [2].

Загальні ознаки включають поля орієнтації, які представляють зовнішній вигляд відбитка пальця, орієнтацію зображення, кривизну та загальне розташування ліній, схожих на сосочки. Місцеві особливості або деталі – це місцеві особливості шкірного сосочка, які є унікальними для кожної точки відбитка. На практиці використовується понад 100 хвилин, найвідомішими з яких є дві, які розривають або розривають лінії капілярів [3]. Далі розглянемо підхід, заснований на місцевих особливостях. Це пояснюється тим, що сьогодні він є найпоширенішим і надійнішим, ніж глобальний підхід на основі ознак.

Для порівняння відбитків за мінущіями в роботі буде використаний алгоритм MCC (Minutia Cylinder-Code) [4], як найточніший і найшвидший з практично застосовних. У ньому кожної мінущії зіставляється циліндр, в якому враховуються положення та напрямок найближчих сусідів мінущії.

Для того щоб оцінити надійність дактилоскопічних систем, зазвичай використовують ймовірності помилок 1-го та 2-го роду. В даному випадку це помилковий збіг і помилкове розбіжність відбитків пальців. Їх визначено ймовірність помилкового збігу (False Match Rate, FMR) і можливість помилкового розбіжності відбитків (False Non-Match Rate, FNMR) [3]. У випадку алгоритм, використовуваний для порівняння, приймає рішення виходячи з деякого порогового значення t . У зв'язку з

цим важливо пам'ятати, що помилки FMR і FNMR залежать один від одного, і при збільшенні чутливості системи, тобто при збільшенні порога (і відповідно зниження FMR), також підвищується FNMR. Для наочного представлення даних можливостей використовують ROC-криві (Receiver Operating Characteristic, операційна характеристика приймаючого пристрою) або DET-криві (Detection-Error Tradeoff, компроміс помилок виявлення). ROC-криві є залежністю FMR(t) від (1-FNMR(t)) при варіюванні порогового значення, а DET-криві – залежність FMR(t) від FNMR(t).

Рівна частота помилок (EER) іноді використовується для вказівки збігу між помилками FMR і FNMR. Чим нижчий рівень EER, тим надійніше система.

Постановка проблеми

Метою даної роботи є створення аналогу системи розпізнавання відбитків пальців на основі методу нечітких правил мислення та дослідження якості цього алгоритму в порівнянні з іншими методами в цій галузі. Для досягнення цих цілей були розроблені такі вимоги:

- Вивчати технологію систем розпізнавання відбитків пальців та використання нечіткої логіки.
- Алгоритми та прототипування програмної частини мультимодальної системи розпізнавання відбитків пальців.
- Створити підсистему прийняття рішень що володіє нечіткою логікою.
- Порівняйте результати з відомими методами.
- Отримайте картку якості відбитків пальців відповідно до алгоритму NFIQ.
- Запропонувати критерії визначити якість межі відбитку пальців.
- Використовуйте MCCSDK [5], щоб виконати серію порівнянь відбитків пальців з різною кількістю зразків за хвилину залежно від якості.
- Обчисліть похибки FMR, FNMR та EER та порівняйте результати порівняння

1. АНАЛІТИЧНА ЧАСТИНА

1.1 Біометричні системи.

Біометричні системи діляться на два класи: мономодальні та мультимодальні.

Приклад мономодальної системи (Рис.1.1).

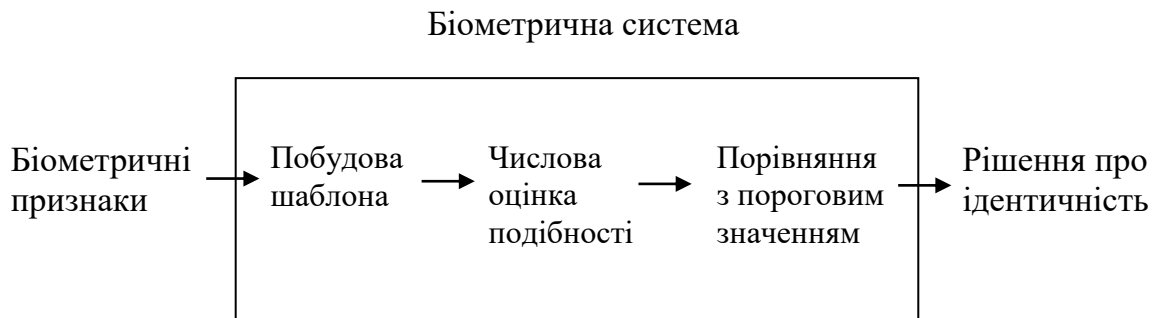


Рис. 1.1 Послідовність виконання мономодальної біометричної системи

На вхід їй надходить одна ознака, отримана зі сканера. Далі вважатимемо, що як ознаки виступають відбитки пальців, хоча ця схема застосовна до всіх біометричних ознак. Спочатку слід обробляти відбиток, за допомогою якого підвищується якість зображення. Далі по ньому будується шаблон, що зазвичай є компактною структурою даних, яка відображає відмітні ознаки відбитка, та компактна для порівняння. Еталонний шаблон друку вибирається з бази даних і порівнюється з дослідженим. Отримані результати порівнюються з пороговим значенням, щоб визначити ідентичність отриманих відбитків і посилань.

Для підвищення точності використовуються мультимодальні системи [11]. У мультимодальних системах множинна модальність, один датчик (рисунок 1.1) та/або алгоритм (рисунок 1.2), використовуються в тій чи іншій комбінації для прийняття біометричних або верифікаційних рішень. Функції, які використовуються в мультимодальній системі, можуть бути однаковими

(кілька відбитків пальців) або різними (вказівні та проміжні відбитки пальців, або фотографія людини та запис голосу).

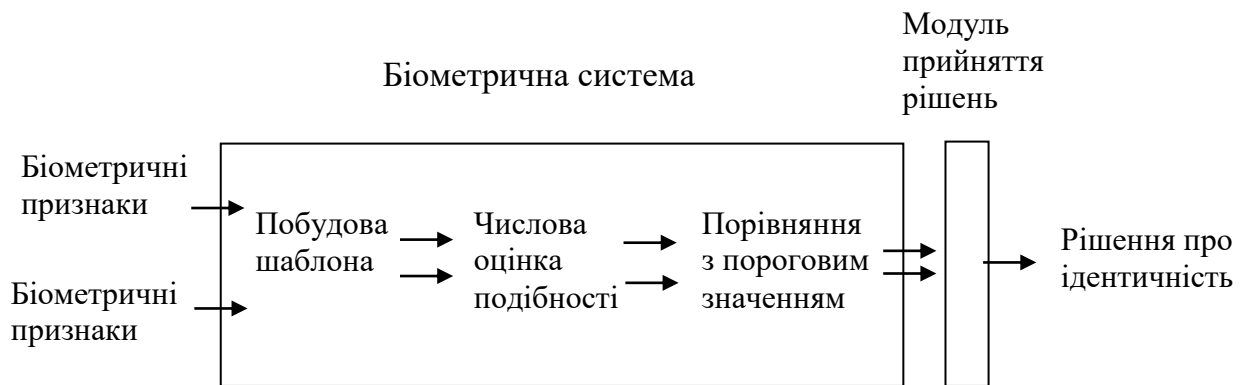


Рис. 1.2 Схема мультимодальної системи з використанням двох біометричних функцій

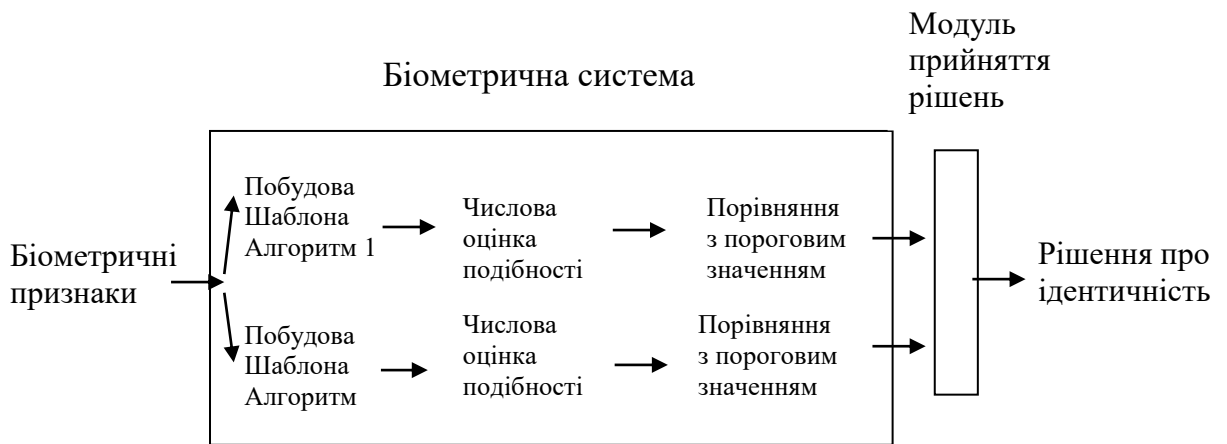


Рис. 1.3 Схема для мультимодальної системи з використанням двох алгоритмів для побудови шаблону.

Як видно з малюнків 1.2 і 1.3, кількість модальностей має бути оптимальною, оскільки при прийнятті остаточного рішення можуть виникнути непорозуміння. Для деяких модулів деякі відповіді – це наступні стратегії прийняття рішень.

- Звірка характеристик і рішень, коли всі модулі дають позитивні результати. Цей приклад більш точно відображає результати.
- Порівняння з пороговим і більшістю рішень. У цьому прикладі, навпаки, ймовірність помилкових допусків висока, а відмова від помилкових допусків зменшується.
- Виведення розв'язку за допомогою нечіткої логіки. Таким чином, ви можете приймати детальні рішення щодо процесу прийняття рішень і враховувати кілька факторів.

1.2 Нечітка логіка

Засоби застосування з теорії множин та загальної логіки використовуються у нечіткій логіці. У літературі опис нечіткої логіки започатковано Лотфі Заде в 1965 [1]. Нечітка множина A – є сукупністю впорядкованих пар, що складені з елементів x узагальненої множини X та відповідних ступенів приналежності.

Функція власності вказує, наскільки елемент x міститься у A . Безліч значень цієї функції – інтервал. Коли багато результатів, неякісна змінна буде мати характеристики звичайної. При використанні нечітких множин дії об'єднання та перетину можна зазначити таким чином:

- Максимінні
- Алгебраїчні
- Обмежені

Висновок нечіткої логіки є мовною змінною [2]. Це змінні, значення яких є мовною фразою. Взевши для прикладу змінну «Якість», виходить

«Високий», «Середній», «Низький». Отримане значення мовної змінної є нечітким набором. Параметр характеристики якості можна змінити від 0,75 до 1,0.

Можна зробити заключення у три етапи.

- Фаззифікація (Fuzzification)

Числові характеристики перетворюються на значення змінних людської суб'єктивної оцінки за при використанні обчислювальних відповідних формул.

- Використання правил

Вони повинні бути стандартизовані до виду "якщо А, то В" і містять також лінгвістичні змінні. Насамперед потрібно визначити ступінь належності посилки. Для цього використовується одну із різних видів обчислення множин. Надалі результати представляються та надається обчислена міра власності.

- Дефаззифікація (defuzzification)

З отриманих даних набувають числові результати.

Метод нечітка логіки в біосистемах.

Використання нечіткої логіки в біометричних системах зустрічається рідко в сучасних дослідженнях.

В публікації «Fuzzy Fusion in Multimodal Biometric Systems» [4] професор В. Конті проаналізовано біометричний план з багатьма ознаками, яка виконує розпізнавання відбитків пальців. Використовували вказівний палець і відбиток посередині. Якість відбитків пальців також використовується для додаткової оцінки при визначенні особи. Однак результати неможливо перевірити, оскільки не існує правил для виведення нечіткої логіки та пояснення того, як оцінити якість.

В іншій статті, Fuzzy Logic Decision Fusion в мультимодальних біометричних системах [3], описується мультимодальна біометрична

система, яка використовує відбитки пальців, фотографії обличчя та голосові записи. Метод нечіткої логіки в даному випадку використовується не безпосередньо для розв'язування задачі, а для визначення коефіцієнтів порівняння відбитка та збігу зображення.

цьому завданні ми розробили мультимодальну біометричну систему з нечітким висновком за допомогою трьох відбитків пальців. Якість відбитків пальців оцінювалася кількома різними алгоритмами, включаючи стандартний NFIQ, який враховував недоліки [3] та [4].

2. ТЕХНОЛОГІЧНА ЧАСТИНА

2.1 Система розпізнавання відбитків пальців

Для реалізації мультибіометричної системи була обрана система, яка аналізує три відбитки пальців. Для аналізу властивостей біометричних даних було вирішено використовувати відбитки пальців. Зі збільшенням кількості проаналізованих відбитків якість не покращується. В результаті накопичується інформація, а якість таким чином не може бути досягнута. Я вирішив зупинитися на 3 відбитках.

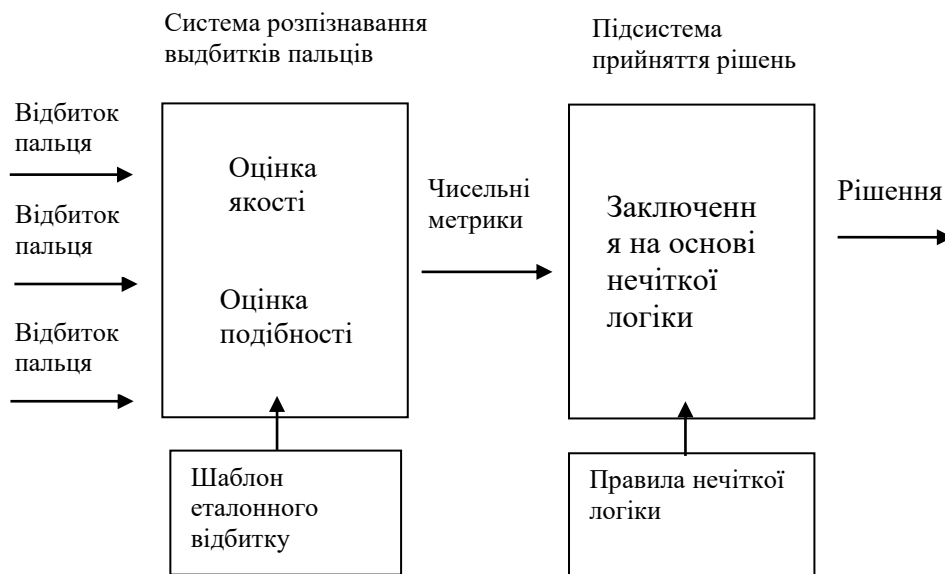


Рис. 2.1 Архітектура мультимодальної біометричної системи

Дану систему можна розділити на такі складові: перша складова розпізнавання відбитків пальців та друга складова - прийняття рішення. При проведенні розпізнавання спочатку потрібно проаналізувати схожість двох відбитків в подальшому виникає потреба проаналізувати також і якість. Система була реалізована мовою C#, платформа .NET Framework 4.5.

2.2 Вимір чіткості відбитків пальців

Щоб краще оцінити якість, ви можете використовувати три алгоритми для вимірювання якості відбитків пальців за різними критеріями.

Алгоритм NFIQ

Стандартний і широко використовуваний метод якості малюнку відбитків пальців NIST [5] корисний для оцінки якості відбитків пальців. Цей алгоритм був винайдений Національним інститутом стандартів і технологій. Орієнтація та кривизна капілярної лінії, контрастність відбитків враховують цей метод при оцінці якості. Малюнок ділиться на частинки і якість оцінюється окремо для кожної частинки. В результаті роботи алгоритму отримуємо карту якості, що містить оцінку від 1 до 5 за кожну частинку друкованої продукції. Звідти ви можете отримати дані про відсоток низькоякісних блоків і середню якість всього друку. На малюнку 2.2 показано високоякісний і низький друк для оцінки алгоритму NFIQ. Для реалізації цього алгоритму була використана бібліотека NFIQ. [10]



Рис. 2.2 Відбитки пальця високої та низької якості за оцінкою алгоритму NFIQ

В процесі виникає потреба відобразити фон при цьому сховавши капілярні лінії. На малюнку 2.3 показаний відбиток із великою частиною фону. Цей тип аналізу заснований на порівнянні чіткості орієнтації окремої частинки (ступінь ясності капілярної лінії) і середньої величини всього відбитка. Результатом роботи є метрика, яка у процентному відношенні характеризує фон об'єкту.

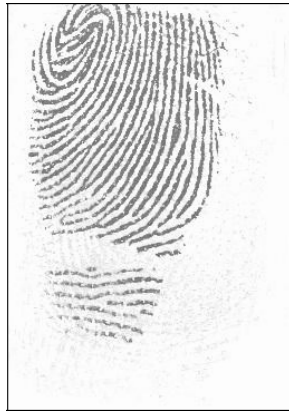


Рис. 2.3 Показник фону, який є надмірним.

2.3 Оцінка якості відбитку

Якщо палець був занадто вологим, який сканують то відбитки будуть з низькою яскравістю. Це дає негативний результат бо капілярні лінії зливаються разом (Рис. 2.4).

Аналізуючи відбитки, можна зробити висновок, що яскраві відбитки гірше темних, а лінії капілярів нечіткі. Остаточний аналіз базується на кількості пікселів, яскравість яких нижче певного порогу. Відсоток отриманих піксельних даних для всіх пікселів друку визначає показник

якості. Тому для покращення аналізу фон і яскравість розглядаються для оцінки якості NFIQ. Це тому, що ці параметри не враховуються.



Рис. 2.4 Малюнок пальця з капілярними лініями, що об'єднуються.

В подальшому є потреба визначення чіткості малюнку, тому використано емпіричний метод. В процесі роботи обчислено поріг яскравості пікселів, ті що вважалися темними, також враховано напрямок при оцінці фону та інші параметри.

2.4 Оцінка сходження двох відбитків

При використанні наступного модуля системи розпізнавання відбитків вдалося оцінити ймовірність, що при порівнянні двох відбитків пальця вони належать тій самій людині.

В подальшому проводиться покращення якості відбитка різними методами для його зручного вигляду, щоб робота не зупинялась і давала позитивний результат. Воно проходить у три етапи.

У першому етапі – сегментації – відбувається визначення фону відбитка та її країв на зображенні. Далі ці області нічого очікувати включатися до розгляду, оскільки алгоритми, заточені працювати з відбитками – зображеннями певного виду і структури – дають велику помилку інших видах зображень.

На другому етапі покращення зображення проводиться його бінаризація: зображення у градаціях сірого наводиться до чорно-білого.

Нарешті, на третьому етапі відбувається витончення папілярних ліній так, щоб вони були завширшки один піксель.

Після покращення зображення, потрібно було визначитися із розміщенням мінуцій. Мінуція – являє собою місце на відбитку де сходяться або розходяться капілярні лінії. Де розташовані мінуції та їх вигляд є унікальним кожної людини [7]. Для реалізації виділення мінуцій були використані бібліотеки CUDA Fingerprint [6].

У роботі зберігаються дані де розміщені мінуції, в подальшому дана база даних зручна для порівнянь. Алгоритм побудови шаблонів Minutia Cylinder Code (MCC) [8] кожної мінуції будує "циліндр", який визначає положення найближчих мінуцій щодо даної. Мірою схожості двох відбитків загальна відстань Хеммінга між "циліндрами". MCC розроблено в Лабораторії біометричних систем Болонського університету. Цей алгоритм стійкий до обертання, зміщення та розтягування пальців. Для його реалізації було використано бібліотеку MCC SDK [9].

Параметри підрахунку схожості в алгоритмі MCC та інші параметри системи було визначено емпірично.

3 КОНСТРУКТОРСЬКА ЧАСТИНА

3.1 Система підтримки прийняття рішень з використанням нечіткої логіки.

Дослідження проводять на 3-х відбитках, далі будують метрику рішень. При використанні 5-ти метрик, 4-ри з них є якісними і 1-на схожості з еталоном.

Фазифікація

Як говорилося вище, нечіткий висновок проводиться в три етапи і перший з них – фазифікація. Переглядаються результати чисельних метрик, які потім перетворюються у лінгвістичні змінні, використовуючи різні функції приналежності.

Фазифікація подібності

Наступним етапом є фазифікація. Отримані лінгвістичні змінні фіксуються у даних Identity з результатами Same та Different. Вказані обчислення є нечіткими множинами. Графічне представлення діяльності приналежності зображено на графіку (Рис. 3.1)

Метрика подібності, що отримується від алгоритму МСС, коливається від 0 до 1, де 1 позначає абсолютно однакові відбитки, 0 – вказує на різні отримані показники. Дані отримані з метрики будуть являтися як аргументом функції приналежності. В подальшому отримані результати підрахунку виходять характеристики приналежності для розраховуваних змінних.

Скажімо, якщо результат МСС дорівнює 0.6, то результат для змінної Identity Same рівний 0.88, а для змінної Identity Different – 0.12.

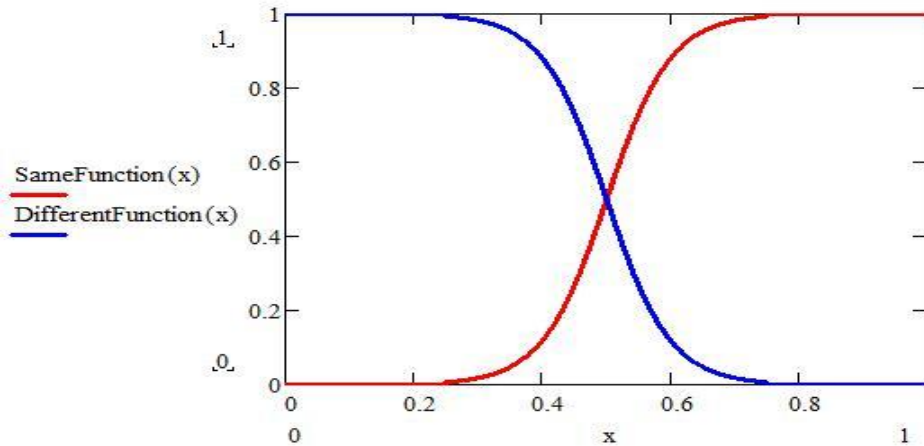


Рис. 3.1 Змінна функція членства Identity

У традиційних методах визначення цей зсув графіка (рисунок 3.1) суттєво реагує на зміни порогу порівняння

Фазифікація якості.

Конкретне значення числової змінної виводиться для метрики якості, подібно до того, як був здійснений перехід від явного значення параметра до нечіткого значення якості. У майбутньому нам потрібно окреслити діапазон допусків, результуючі змінні та функцію належності для цього значення.

Середній показник якості.

Середній показник якості при використанні алгоритму NFIQ може коливатися від 0 до 4. При цьому, чим менше значення, тим нижча якість і вища якість.

Отримувана перемінна - QualityNfiq має результат Low, High.

Функції приналежності:

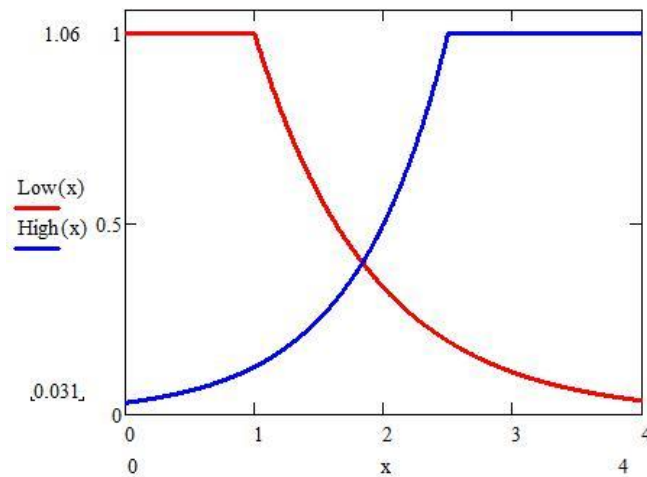


Рис. 3.2 Змінна схема членства QualityNfiq

Показник відсотка блоків низької якості.

Відсоткові показники для блоків низької якості з використанням алгоритму NFIQ діапазон від 0 до 100. Одиниці низької якості – це одиниці з оцінкою NFIQ 0 або 1. Блок змінних низької якості зі значенням Little і Many.

Функції приналежності:

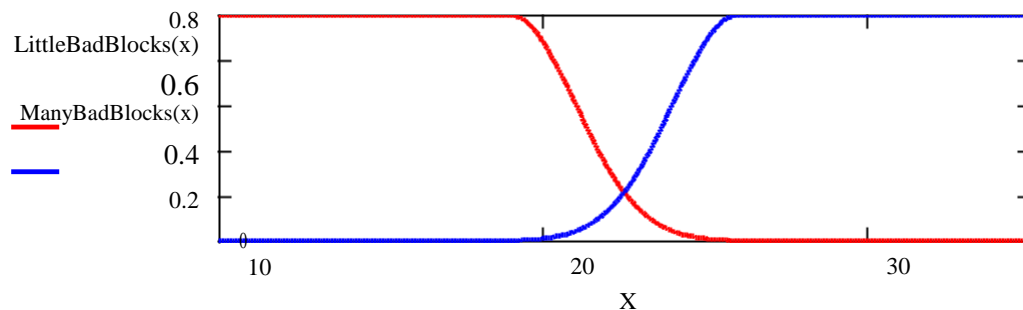


Рис. 3.3 Змінна схема членства LowQualityBlocks

Функціональні графіки створюються для значень від 10 до 40. За межами діапазону отриманих даних функцій не підлягає зміні.

Показник яскравості

Яскравість відбитка при розрахунку значення знаходиться в діапазоні 0-100.

Якщо результат змінюється навколо 0, ви отримаєте звичайний друк, а якщо значення велике, значення темне.

Обчислений результат – Brightness отримало результат High та low

Функції приналежності:

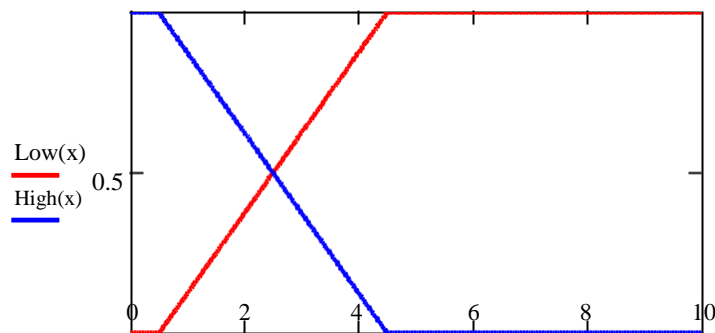


Рис. 3.4 Змінна схема членства Brightness

Креслення функції членства коливається від 0 до 10. Якщо значення виходить за межі діапазону, значення функції не змінюється.

Відсоток метричного фону.

Показники метрик відсотка фону є у межах між 0 та 100.

Отримувана перемінна Background з результатом Large та Normal.

Функції приналежності:

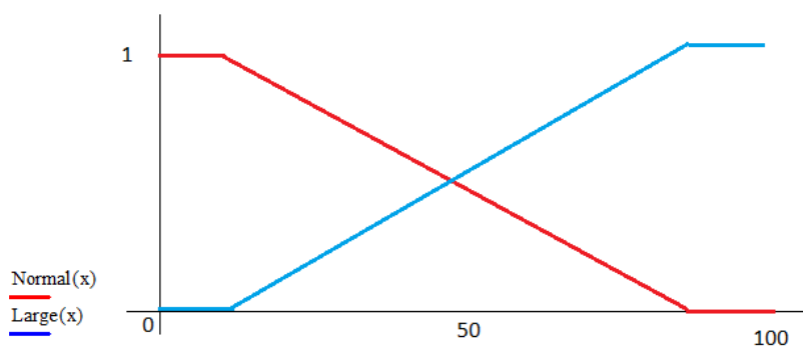


Рис. 3.5 Змінна схема членства Background

Вигляд цих функцій власності було визначено експериментальним методом.

3.2 Застосування правил

Наступним кроком у невизначеному висновку є застосування правил. Перед цим кроком правило буде прочитано з файлу. Змінні мови, отримані під час фасилітації, надходять у вхідні дані.

Висновок відповідає всім правилам, які зустрічаються в посиланні, оскільки всі значення мовних змінних присутні одночасно. Максимальна операція використовується для розрахунку ступеня власності ділянки. Крім того, якщо змінна вартість тюрми ще недоступна, вона буде створена з пакетом володіння, рівним володінню посилкою. Якщо є, ступінь атрибуції визначається як максимальна існуюча та отримана.

Висновок змінної якості.

На ранніх стадіях змінні аналізуються Quality, що має дані Low, Middle та High. Перемінна характеризує общє значення чіткості для друку. Оскільки правил досить багато, буде описано принцип їхньої побудови. Повний перелік правил можна побачити у додатку. Послідовність встановлення змінної якості йде таким чином.

- При отриманні високого результату відсилки то і якість велика.
- При отриманні низького результату відсилки то і якість середня.
- Якщо ще більш низький результат то і якість ще нижча.

Зразок для значень:

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

Кома в цьому випадку означає логічне «І». Отже, в даному випадку це правило означає: Якщо зображення яскраве і має невеликий фон, воно велике при оцінці алгоритму NFIQ, а якщо частинок низької якості мало, то якість хороша.

Висновок для одного відбитку.

У цій межі висновку задіяна змінні чіткості та подібності, а правила базуються на таких принципах:

- Якщо чіткість велика, відповідь відповідає схожості змінних
- Якщо чіткість середня, відклик з коефіцієнтом $\frac{1}{2}$ відповідає подібності змінних, а коефіцієнт $\frac{1}{2}$ невідомий.
- Якщо отримуємо якість низьку, то визначити відповідь важко

Приклади правил:

quality high, identity different => fingerprintAnswer no

quality low, identity same => fingerprintAnswer idk (idk = I don't know)

Висновок загальної характеристики для 3-х відбитків.

У заключному розділі правил використовуємо результат для 3-х відбитків з пальців. За результатами досліджень приймаються до уваги значення які мають більшість . Отже якщо результат з 3-х має значення Yes, No і Idk числові дані будуть такі : , -1 і 0. Остаточне значення буде округлене у бік збільшення модуля.

Приклади 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

Мавши більш узагальний результат, ми дефазифікуємо його і, як наслідок, отримуємо відклик з максимальною відповідальністю.

3.3 Аналіз і порівняння

Види помилок розпізнавання.

Через стохастичний характер рішення про ідентифікацію, відповідь біометричної системи на один і той же відбиток завжди позитивна, різно-негативна. Це призведе до помилки класифікації. Це ймовірність помилкового відмову (FRR) і частота помилкового прийняття (FAR). Під час порівняння порогових значень вказівка високого порогу не тільки зменшує частота помилкового надання, але й збільшує ймовірність випадкової відказу у доступі. І навпаки, встановлення низького порогу збільшує ймовірність помилкового введення та зменшує ймовірність помилкової відмови у доступі. Враховуючи помилку, можна оцінити близькість значення системи (Equal Error Rate, EER), тобто точку $FAR = FRR$ (рисунок 3.6).

В ідеалі графіки FAR і FRR не перетинаються, а значення EER дорівнює нулю [7].

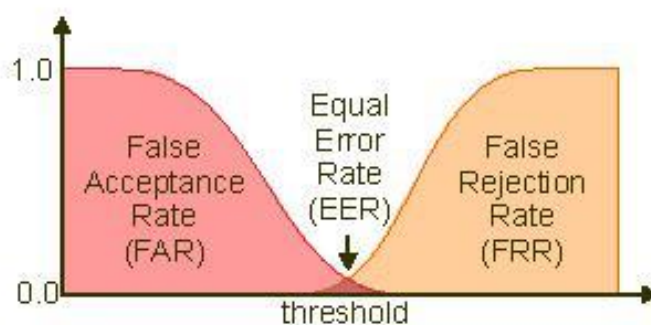


Рис. 3.6 Помилки FAR, FRR та EER

Порівняння систем.

Мною було порівняно результат визначення системи прийняття рішень на основі нечіткої логіки з результатом розпізнавання подібної системи, яка відрізняється лише частина, що визначає рішення. Це було зроблено для забезпечення стабільності та запобігання впливу інших факторів на результати.

Вони описані нижче.

1. Одноmodalьна функція порівняння порогів.
2. Багатомодальна система порівняння порогових значень. Якщо отриманий результат кожного з трьох відбитків хорошій, система дає плюсову відповідь.
3. Мультиmodalьна частина порівняння порогових значень. Відповідь системи може мати значення при звичайній більшості голосів.

Таблиця 3.1 Порівняння ERR та порогів систем

	Система з нечіткою логікою	Одноmodalьна система	Багатомодальна система	Мультиmodalьна система з відкликом три з трьох
ERR	17.81%	30.82%	29.33%	28.21%
Поріг	0.525	0.545	0.545	0.53

При використанні нечіткої логіки межі системи беруться на перетині функцій значень, яким вони належать Identity Same та Identity Different.

Тестова множина.

Щоб переконатися, що вам потрібно використовувати відкриту базу даних FVC2000, DB2 Set A, ця база даних містить 800 відбитків пальців по 8 відбитків пальців на один. Існуючі відбитки в базі даних відображаються у форматі .tif з розміром 364 x 256 пікселів і роздільною здатністю 500 dpi. Зверніть увагу, що база даних FVC2000 має велику кількість значень з низькою якістю друку і побудована для усереднення результатів. З цих восьми відбитків пальців еталонний вважався NFIQ найвищої якості. Під час розрахунку ймовірності неправильної відмови у користуванні з решти семи відбитків було обрано три відбитки. Їх подали до входу системи. Три відбитки пальців, за крім того відбитків пальців, були випадковим чином відібрані з усієї бази даних для обчислення ймовірності неправильного доступу.

Значення чіткості відбитків і шаблони MCC були кешовані, щоб скоротити час праці системи.

4 НАУКОВО-ДОСЛІДНА ЧАСТИНА

4.1 Детектор кордонів Кенні

У 1986 році професор Массачусетського технологічного інституту Джон Кенні (John F. Canny; 1953) розробив метод визначення меж зображення – детектор кордонів Кенні або алгоритм Кенні [13]. Цей метод складається з п'яти окремих кроків:

1. Згладжування
2. Пошук градієнтів
3. Зменшення немаксимумів
4. Подвійна порогова фільтрація
5. Трасування області неоднозначності

Згладжування

Усі знімки, що отримуються за допомогою сканування, неминуче містять деяку кількість шуму, який усувається за допомогою розмиття зображення. У алгоритмі для розмиття використовується фільтр Гаусса з ядром $N \times N$. Даний фільтр є матрицею згортки, заповнену за законом нормального розподілу.

Пошук градієнтів

На цьому етапі відбувається пошук напрямків зміни яскравості на зображенні за допомогою оператора Собеля. Спочатку вихідне зображення A згортається по осі X і по осі Y за допомогою наступних фільтрів:

$$K_{GX} = \begin{pmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{pmatrix}, \quad K_{GY} = \begin{pmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{pmatrix}.$$

Тобто $G_x = K_{G_x} * A$, а $G_y = K_{G_y} * A$

де * - операція згортки.

Після чого обчислюється наближене значення градієнта і його напрямок:

$$G = \sqrt{G_x^2 + G_y^2}, \quad \theta = \arctg\left(\frac{G_y}{G_x}\right).$$

Потенційними межами на зображенні A вибираються точки, у яких градієнт має найбільше значення.

Зменшення немаксимумів

Коли немаксимальне значення зменшується, здійснюється пошук пікселів, які досягають максимального значення градієнта в напрямку вектора градієнта. Розглянемо приклад, зображений малюнку 4.1:

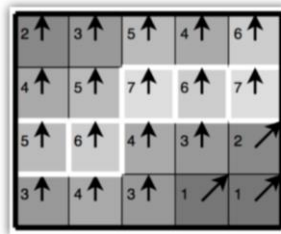


Рис.4.1 Вибір локального максимуму

Більшість пікселів з прикладу мають напрямок на північ, тому значення градієнта в цих точках порівнюються зі значеннями градієнта нижчерозташованих і вищерозташованих пікселів. Тому пікселі з білими контурами залишаються в отриманому зображенні, а решта пригнічується.

4.2 Подвійна порогова фільтрація

Більшість пікселів, що залишилися після попереднього кроку визначають межі зображення, проте в алгоритмі Кенні застосовується порогова фільтрація, яка дозволяє виключити пікселі, викликані шумом вихідного зображення. Межовий детектор Кенні використовує два порогових значення фільтра в діапазоні 0-255. Якщо значення пікселя вище верхньої межі, то це буде максимальне значення (ліміт вважається дійсним), а якщо воно нижче, піксель буде придушено. Пікселі, значення яких знаходяться в діапазоні між порогами, отримують фіксоване середнє значення.

Трасування області неоднозначності

На останньому етапі аналізуються групи пікселів, що потрапили в середню область. Пікселі, що лежать поруч із кордоном, приєднуються до неї, інші – відкидаються.

Перетворення Хафа

Перетворення Хафа [14] — це алгоритм знаходження ліній, кіл та інших не складних фігур на малюнку. Ідея алгоритму полягає в тому, що дані мають об'єкт, що цікавить, у вигляді параметричних рівнянь, які утворюють фазовий (батарейний) простір. Потім витягується двійкове зображення (в результаті дії детектора кордонів Кенні), сортуються всі межі пікселів, і вважається, що кожен захоплений піксель належить до бажаної кривої. Так, на піксель зображення

Розраховуються необхідні рівняння та отримані необхідні параметри.

Він зберігається в топологічному просторі Хафи. Після цього проходить обхід простору, щоб вибрати максимальне значення, яке дозволить вибрати

найбільшу кількість пікселів у проголосованому зображенні. Це дасть вам параметри рівняння для об'єкта, який вас цікавить. Для цього завдання потрібне перетворення Хафа, щоб виявити круглу райдужну оболонку. Тому розглянемо роботу з цим алгоритмом, щоб знайти коло.

Як відомо точки кола можна, можливо уявити наступною формулою:

$$(x-a)^2 + (y-b)^2 = R^2$$

де (a, b) – координати центру кола, а R – її радіус. Рівняння сімейства кіл має вигляд:

$$F(a,b,R,x,y) = (x-a)^2 + (y-b)^2 - R^2 \quad (4.1)$$

Як видно з рівняння (4.1), для пошуку кіл потрібно задавати три параметри: дві координати центру і радіус. Оскільки розмірність фазового простору позначається ефективності використання перетворення Хафа, бажано будь-яким чином мінімізувати кількість параметрів (4.1). Наприклад, можна визначити можливі значення радіусів шуканого кола $R_1 \dots R_2$, тим самим зменшивши простір Хафа на одиницю. Так, кожною точці (X, Y) зображення ставиться в відповідність набір точок фазового простору (a, b) . Для кожної точки (a_0, b_0) фазового простору (a, b) створюється лічильник, який відповідає за кількість точок (x, y) , що лежать на колі з параметрами (a, b) . Безперервне фазовий простір переводиться в дискретний, шляхом накладання сітки на простір (a, b) , одному осередку якого відповідає набір кривих з близькими значеннями a і b . Параметри a і b комірки характеризуються як середні значення точок a_i і b_i , що потрапили в цей осередок. Для кожної білої точки зображення шукаються осередки у фазовому просторі (a, b) і збільшується лічильник. Після обходу всіх точок бінарного зображення, фазове простір заповниться. Далі будуть обрані кола, які містять найбільшу кількість точок зображення. Такий результат досягається шляхом введення межі: осередки, значення лічильників яких вище за цю

межу оголошуються справжніми колами — решта відкидається. Проте, такий підхід загрожує появою кіл, які мають “майже однаковий” центр. Такі кола слід виключити так:

1. Знайти комірку з максимальним значенням лічильника. Якщо це значення менше за межу, вихід.
2. Округність, що відповідає знайденому осередку (a' , b') прийняти за справжню.
3. Знайти білі точки на вихідному бінарному зображенні, що лежать на знайденому колі.
4. Знайдені точки перефарбувати в чорний колір, і для кожного кола, що проходить через ці точки, знайти відповідну точку у фазовому просторі (a , b) і зменшити лічильник відповідного осередку на одиницю.
5. Перейти до кроку 1.

4.3 Метод Віоли-Джонса

Метод Віоли-Джонса [15] - метод, що дає можливість детектувати об'єкти в реальному часі. Цей принцип був розроблений у 2001 році зусиллями Пола Віоли та Майкла Джонса і одразу став проривом в області виявлення об'єктів. Незважаючи на те, що основною метою розробки методу Віоли-Джонса було виявлення осіб, алгоритм одно добре працює і розпізнає різні класи зображень, у тому числі і риси особи за різних умов освітленості під кутом до 30 градусів. Як одне з найкращих співвідношення ефективності розпізнавання до швидкості, метод Віоли-Джонса має низький шанс помилкових спрацьовувань. Даний метод використовує принцип ковзного вікна, суть якого полягає у виборі деякого прямокутного вікна на зображенні, наприклад, 24x24 пікселів. Після чого проводиться обхід досліджуваного зображення з

переміщенням вікна деякий позитивний крок. По досягненні кінця зображення ковзне вікно переноситься на початок зображення і процес пошуку продовжується. Це може статися, наприклад, доти, доки розміри вікна не співпадуть з досліджуваним зображенням. Тобто для кожного вікна вирішується завдання класифікації – потрібно вирішити, чим є або не є об'єкт у ковзному вікні.

Також метод Віоли-Джонса базується на наступних правилах:

1. Подання досліджуваного зображення в інтегральному вигляді. Це дозволяє швидко проводити обчислення.
2. Здатність цифрових зображень використовувати функцію Хаара [16], тобто розпізнавати зображення, які використовуються для пошуку об'єктів, що цікавлять.
3. Використовуйте Boost, щоб вибрати найбільш відповідну функцію для об'єкта, знайденого в цій частині зображення.
4. Використовуйте каскад функцій, щоб негайно відхилити вікно, де об'єкт не знайдено. Розглянемо кожен із принципів докладніше.

Інтегроване представлення зображення.

Уявлення, що розглядається, є одним з основних етапів у багатьох методах розпізнавання об'єктів, наприклад, у вейвлет-перетвореннях і SURF(Speeded Up Robust Features) методі. Інтегроване уявлення зображення описується матрицею, розмірність якої збігаються з розмірами досліджуваного зображення. Формула для розрахунку елементів матриці виглядає наступним чином:

$$I(x, y) = \sum_{x' \leq x, y' \leq y} i(x', y'), (1.1)$$

де $i(x', y')$ – показує чіткість пікселя вихідного елемента картинки.

Рекурентний аналог формули (4.1)

$$I(x,y) = i(x,y) - I(x-1, y-1) + I(x, y-1) + I(x-1, y) \quad (4.2)$$

З (4.1) видно, кожен елемент матриці є суму пікселів у прямокутнику від 0,0 до x,y. Тобто в усіх частинках міститься сума яскравостей всіх пікселів, що знаходяться ліворуч і вище за цей елемент. Розрахунок матриці проводиться за лінійний час, пропорційний числу пікселів на зображенні. Таким чином, інтегральне уявлення дозволяє розраховувати сумарну яскравість будь-якого прямокутника на зображенні всього за чотири звернення до пам'яті. Примітно, що час розрахунків залежить від площі прямокутника. Для більшої наочності розглянемо рисунок 4.2.

У верхньому прямокутнику кожна комірка є яскравістю пікселя вихідного зображення. Нижній прямокутник є ніщо інше, як інтегральне уявлення зображення. Припустимо, нам потрібно обчислити суму пікселів S червоного прямокутника. Очевидно, що всередині червоного прямокутника можна виразити через суми і різниці суміжних прямокутників в інтегральному зображенні за наступною формулою (4.3).

3	2	4	1	
5	1	5	6	
2	3	1	8	
9	1	2	5	
3	5	9	10	
8	A	11	B	27
10		14		41
19	D	26	C	58

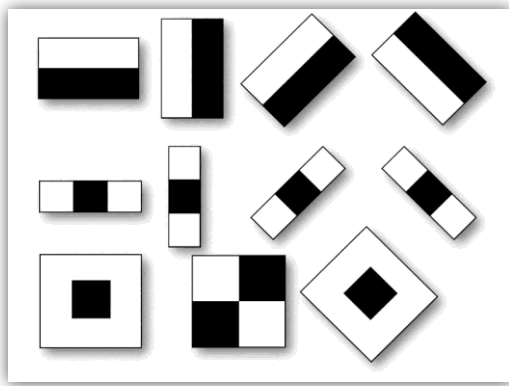
Рис. 4.2.Інтегральне уявлення зображення.

$$S = A + C - B - D \quad (4.3)$$

4.4 Ознаки Хаара

Ознакою f об'єкта a називається відображення $f: A \rightarrow D_f$,

де D_f безліч допустимих значень ознаки. Якщо заданий набір ознак $f_1 \dots f_n$, то вектор $x = (f_1(a) \dots f_n(a))$ прийнято називати відповідною характеристикою об'єкта $a \in A$. Допускається не проводити відмінностей між ознаковим описом і самим об'єктом. При цьому безліч $A = D_{f_1} \times \dots \times D_{f_n}$ називають визначеним простором. Оригінальний метод Віоли-Джонса використовує прямокутний елемент, який називається примітивом



Хаара.

Рис. 4.3 Стандартні примітиви Хаара.

Але в розширеному методі, зокрема представленому в бібліотеці OpenCV використовують і додаткові примітиви.

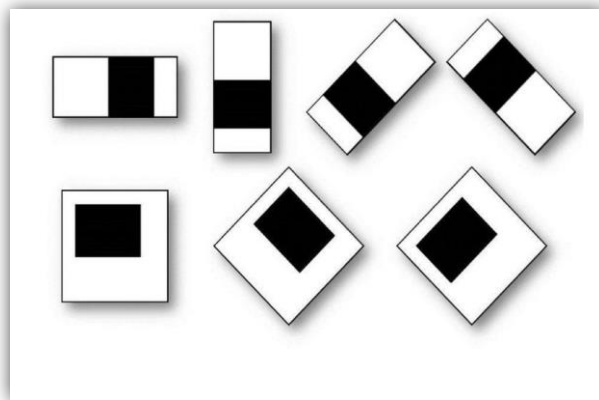


Рис. 4.4 Додаткові примітиви Хаара.

Чисельним значенням цієї ознаки називається:

$$F = U - V,$$

де U — сума значень яскравості точок, закритих ясною межею примітиву Хаара, а V — сума значень яскравості точок, закритих темною частиною Хаара. Для обчислення U та V використовуйте інтегральне представлення вихідного зображення. Розглянемо приклад обчислення значення наступного примітиву Хаара на основі (4.3).

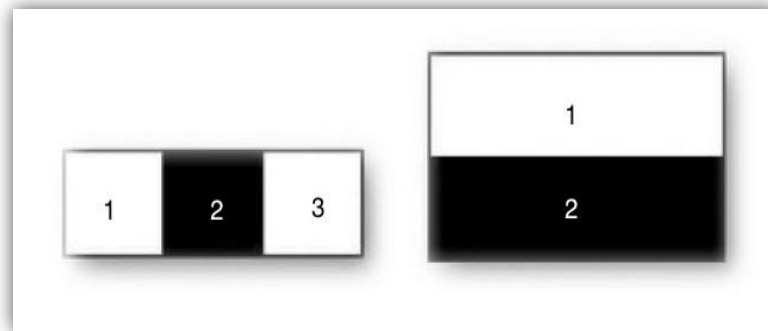


Рис. 4.5 Примітиви Хаара.

Для трьох суміжних прямокутників $F = S_1 + S_3 - S_2$.

Для двох суміжних прямокутників $F = S_1 - S_2$,

де S_1, S_2, S_3 — суми яскравостей пікселів зображення, що закривається прямокутниками 1, 2, 3 відповідно.

Виявляється, що для кожного вікна розміром 24×24 пікселя існує 160000 комбінацій ознак Хаара. Очевидно, що для побудови класифікатора нам знадобляться не всі. Для вибору необхідних ознак буде використано алгоритм посилення класифікаторів — AdaBoost[17].

AdaBoost

Бустинг (англ. boosting – покращення) – це метод ітеративної побудови вигляду алгоритмів машинного навчання, в якому кожен слідує алгоритм хоче перекрити недоліки композиції всіх попередні алгоритми. Бустинг є діючим алгоритмом, тобто алгоритм приймає локально оптимальні рішення на кожному етапі, припускаючи, що кінцеве рішення також виявиться оптимальним. AdaBoost є алгоритмом адаптивного бустингу. Це означає, що кожен наступний класифікатор побудований на об'єкті, який погано класифікований попереднім класифікатором. У результаті AdaBoost буде строгий класифікатор (допускає мало помилок), що є лінійною комбінацією зважених слабких класифікаторів (допускають велику кількість помилок, не дозволяючи надійно визначити клас досліджуваного об'єкта).

$$h(x) = \text{sign} \left(\sum_{j=1}^M \alpha_j h_j(x) \right), \quad h_j(x) = \begin{cases} 1, & s_j f_j < \theta_j \\ -1, & s_j f_j \geq \theta_j \end{cases}$$

де Q_j – поріг класифікації, f_i – значення примітиву даного об'єкта, $S_j \in \pm 1$ – полярність класифікатора, а \square – об'єкт, що класифікується.

Розглянемо алгоритм AdaBoost:

Дано: $X^n = ((x_1, y_1), \dots, (x_i, y_i))$ – навчальна вибірка, де y_i – номер класу для зразка x_i ($y_i \in \pm 1$);

1. Ініціалізуються ваги $w_{1,i} = \frac{1}{2m}, \frac{1}{2l}$,

де m, l – кількість негатив

та позитивних зразків відповідно.

2. Для кожного $t = 1, \dots, T$:

а) нормуємо ваги $w_{t,i} = \frac{w_{t,i}}{\sum_{j=1}^n w_{t,j}}$

б) вибираємо слабкіший класифікатор з найменшою похибкою $\epsilon_t = \min_{f,p,\theta} \sum_i^n w_i |h(x_i, f, p, \theta) - y_i|$

в) визначаємо класифікатор $h_t(x) = h(x, f_t, p_t, Q_t)$, де f_t, p_t, Q_t – значення на яких досягається ϵ_t .

г) оновлюються ваги $w_{t+1,i} = w_{t,i} \beta_t^{1-e_i}$,

де функція втрат $\beta_t = \frac{\epsilon_t}{1-\epsilon_t}$,

якщо $e_i=0$ тоді об'єкт x_i визначений коректно, коли $e_i=1$ навпаки.

3. Отримуємо класифікатор

$$c(x) = \begin{cases} 1, & \sum_{t=1}^T \alpha_t h_t(x) \geq \frac{1}{2} \sum_{t=1}^T \alpha_t \\ 0, & \sum_{t=1}^T \alpha_t h_t(x) < \frac{1}{2} \sum_{t=1}^T \alpha_t \end{cases},$$

де

$$\alpha_t = \log \frac{1}{\beta_t}.$$

Таким чином, відкидаються всі примітиви, які обнулюють значення функції класифікатора.

Виділимо переваги та недоліки розглянутого алгоритму.

Переваги:

1. У деяких завданнях узагальнююча здатність може покращуватися у міру збільшення числа базових алгоритмів.
2. Легкість реалізації
3. Час побудови композиції майже повністю визначається часом навчання базового алгоритму, тому накладні витрати на підвищення є невеликими.
4. Можна ідентифікувати об'єкти, які є шумовими (об'єктами з

найбільшою вагою)

Недоліки:

1. Якщо буде спостерігатися значна кількість шуму в даних, то буде потрібно перенавчання. Справа в тому, що β_t сильно збільшує ваги об'єктів, на яких помиляються багато базових алгоритмів. Однак ймовірно, що ці об'єкти виявляться шумовими викидами. Як результат, AdaBoost починає пристосовуватися до шуму, що змушує перенавчати. Ця проблема вирішується шляхом видалення викидів або застосування менш агресивних функцій втрат. Зокрема, застосовується алгоритм GentleBoost.
2. Для AdaBoost потрібна досить довга навчальна вибірка.
3. Занадто впевнена стратегія AdaBoost може призводити до побудови неоптимального набору базових алгоритмів. Для покращення композиції слід періодично повертатися до раніше побудованих алгоритмів та перенавчати їх.
4. Бустинг може призводити до побудови композицій, що складаються із сотень алгоритмів. Такі величезні композиції вимагають великих обсягів пам'яті для зберігання базових алгоритмів та суттєвих тимчасових витрат на обчислення класифікацій.

Так чи інакше на сьогодні підхід посилення слабких класифікаторів є одним з найефективніших методів класифікації через свою високу швидкість, ефективність роботи та відносну простоту реалізації.

5. СПЕЦІАЛЬНА ЧАСТИНА

Проблема якості відбитків пальців.

Щоб визначити особистість при ідентифікації людини часто використовують відбитки пальців. Так чи інше для алгоритмів розпізнавання та зіставлення відбитків важлива чіткість картинки зображень, та зменшення по сторонніх негативних факторів. Під негативним фактором (шум) розуміються не чіткі, розмиті показники відбитка, області без вираженого напряму капілярних ліній і т.д., з яких важко отримати інформацію, що описує відбиток.

Неякісні зображення відбитка є наслідком поганого епітелію шкіри, великих пор або ненавмисний рух пальця під час зняття відбитка.

Погані картинки відбитків пальців зазвичай характеризуються відсутністю чіткої орієнтації шкірного сосочка і розмитим зображенням.

В даний час для визначення якості друку використовуються такі методи: NFIQ і LFIQ. Останній дуже особливий, оскільки задіяно для відбитків пальців, що отримано на місці скоєння злочину. Тому алгоритм NFIQ потребує більш детального дослідження.

5.1. Алгоритм NFIQ

NFIQ (NIST Fingerprint Image Quality) – дані послідовності встановлення чіткості відбитків пальців, були винайдені у Національному Інституті стандартів та технологій (NIST). Даний метод бере до уваги інформація про напрямки та кривизну капілярних ліній. Також важливою характеристикою є чіткість зображення.

На початковому етапі тестування беруться до уваги зображення сірого крльору, що далі ділиться на блоки, що не перетинаються. Далі частинки –

блоки обмежуються своєю траєкторією. Контур аналізується та оцінюється по своїй якості. Суміжні контури можуть між собою перетинатися.

Продовжується аналіз, який визначає кривизну капілярних ліній, чіткість зображення та ін. Проаналізувавши кривизну можна кількісно оцінити отриману якість. Можна стверджувати про наявність п'яти рівнів якості: погане (5), середнє (4), хороше (3), дуже хороше (2) та відмінне (1). При оцінці якості даних частинок потрібно створити карту якості. Отримані результати визначають вектори якості та кількісні показники відбиткових хвилин. Тут якість хвилин означає якість блоку, в який він розміщений. Потім орисовий вектор задіюється як вхід для класифікатора багатосарового перцептрона (MLP).

На виході алгоритм надає показники чіткості малюнку відбитків пальців, як і у випадку окремих блоків.

5.2 Помилки FMR, FNMR та EER

При порівнянні двох відбитків можуть бути висловлені 2 гіпотези:

- H_0 : дані відбитки належать різним людям (незбіг)
- H_1 : дані відбитки належать одній людині (збіг).

Результатом зіставлення є певне значення подібності, як було зазначено раніше. Залежно від обраного порогу це значення інтерпретується як збіг чи розбіжність відбитків, тобто підтверджується одне із пред'явлених гіпотез.

Для таких гіпотез існує два типи помилок:

- 1-го роду: помилковий збіг відбитків та
- 2-го роду: хибне розбіжність відбитків.

Відповідно, визначено як часто виходить помилковий збігу (False Match Rate, FMR) та ймовірність помилкового збігу відбитків (False Non-Match Rate, FNMR) (мал. 5.1).

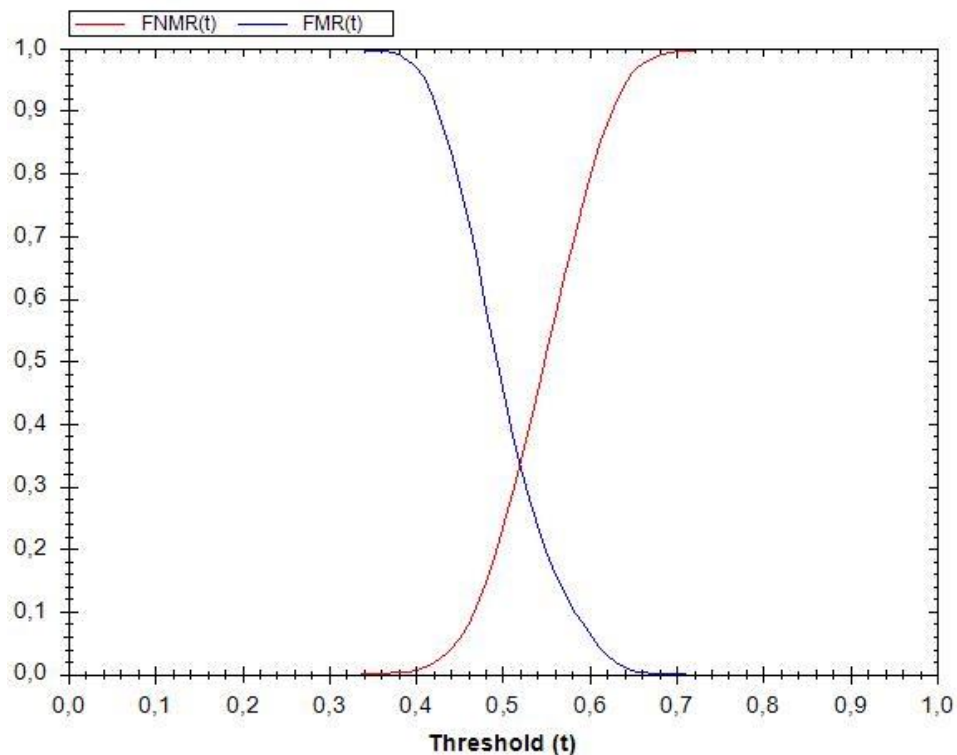


Рис. 5.1 Приклад залежності помилок FMR та FNMR

Як було зазначено раніше дані помилки пов'язані. Якщо зменшити поріг, для отримання більш терпимої системи по відношенню до вхідних даних, то FMR зростає. З іншого боку, якщо збільшити поріг для поліпшення надійності системи, збільшиться FNMR.

Наприклад, у криміналістиці потрібно забезпечити мінімальне значення FNMR. Висока ймовірність помилкової ідентифікації при цьому не є критичною, оскільки метою є пошук злочинця, навіть якщо потрібно організувати додаткову експертизу для певної кількості подібних відбитків. З іншого боку, для систем безпеки з високим рівнем захисту важливо, щоб значення FMR було якнайменше. У такому разі високе значення FNMR є

допустимим. Як правило, більшість біометричних систем вимагають деякого компромісу між значеннями FMR та FNMR.

Для визначення оптимального порогу використовують ROC-криві або DET-криві (мал. 5.2). Ділянка збігу помилок FMR та FNMR називається коефіцієнтом рівної ймовірності похибок 1-го та 2-го роду (Equal Error Rate, EER). Чим нижчий рівень EER, тим система вважається надійнішою. У практичних системах значення EER близько 0,05.

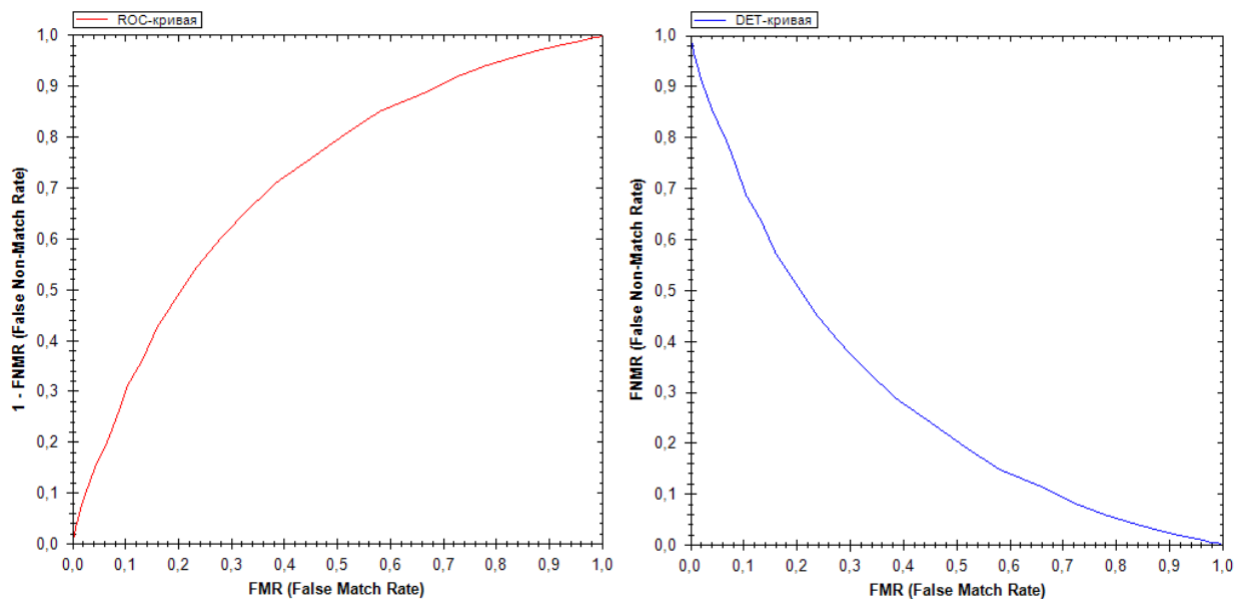


Рис. 5.2 ROCкрива та DETкрива

5.3 Метод визначення якісних областей відбитків пальців

Межа розміщення якості

Алгоритм NFIQ визначає якість друку на основі контрасту зображення, орієнтації капілярної лінії та областей високої кривизни. Тому для кожного відбитка розраховується:

- Карта напрямків.
- Межа розміщення низької різноманітності.
- Карта в невизначених напрямках.
- Межа знаходження високої кривизни.

Чатине де знаходиться якість відбитків пальців розраховується шляхом об'єднання всіх результатів. Кожному блоку в зображенні присвоюється значення від 5 (невисока якість) до 1 (висока якість). На малюнку 5.3 наведено приклад карти якості. Якість 1 деталей позначена білим, а якість 5 деталей позначена чорним.



Рис. 5.3 Відбиток пальця та його картка чіткості

Карта напрямків

Мета створення цієї карти – показати частинки з великою кількістю смуг та вказати їх основний напрямок. На початковому етапі картинка поділяється на квадратні блоки, що не перетинаються, зі стороною $M=8$ (рис. 5.4). Щоб отримати основний напрям блоку потрібно розглянути деяку околицю – вікно зі стороною $L=24$ зсув блоку щодо вікна $N=8$). При невідповідності об'єму картинки розміру блоку, виділена ділянка може бути за межами картинки. Тому картинку узгоджують додатковим сірим кольором зі значенням біля 112.

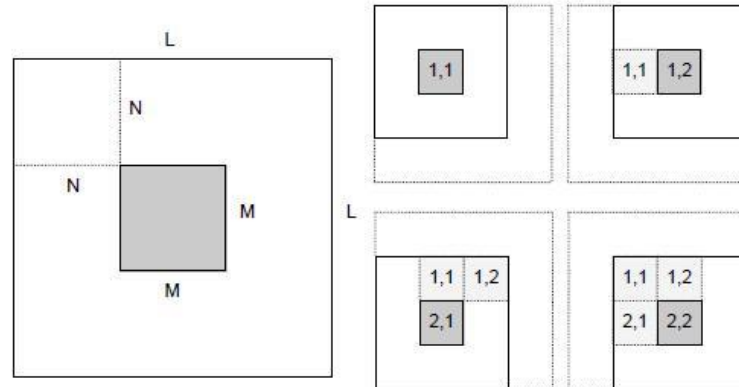


Рис. 5.4 Блоки зображення та вікна, що перекриваються

Далі отримані частинки зображення обертаються поетапно в 16 напрямках (значення обертання $11,25^\circ$), а потім у кожному напрямку виконується дискретне перетворення Фур'є (DFT), а точніше, виконується потрібний поворот. Числа обертання коливаються від 0 до 15, а блоки без певного напрямку відповідають -1.

Приклад певних напрямів представлений на малюнку 5.5, а на малюнку 5.6 – карта напрямків.

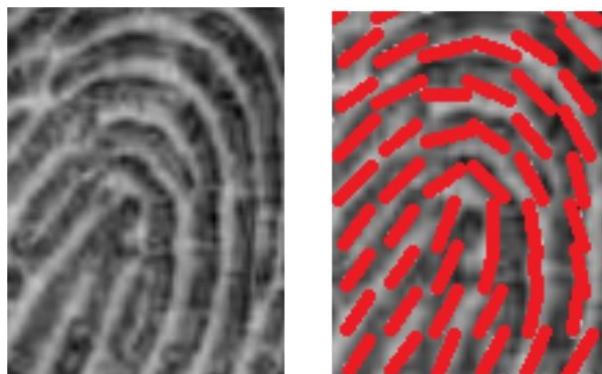


Рис. 5.5 Частина вихідного зображення відбитка пальця і це показ з виділеною орієнтацією.

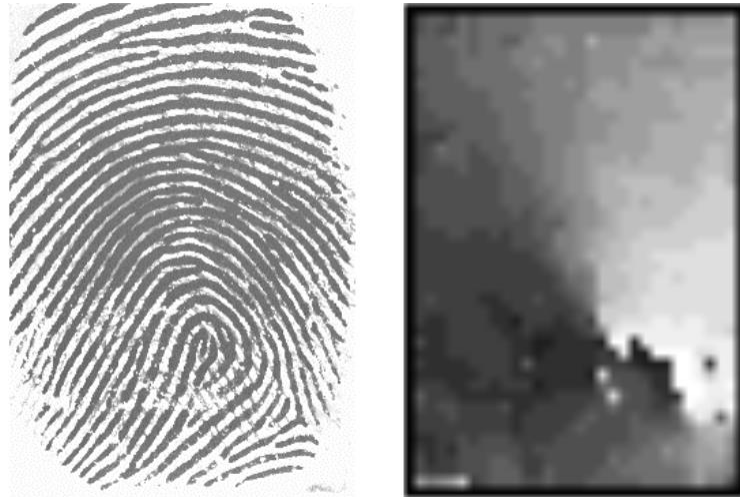


Рис. 5.6 Вигляд відбитка пальця та його межа розміщення напрямків

Критерії оприділення якісних областей відбитків

В послідовності дій NFIQ визначалося якість всього малюнку відбитка. Однак, як правило, більшість частинок картинки мають якісь чіткі зображення, якими і індефікуємо наш відбиток. Якщо відсоток таких зображень не достатньо великий, то оцінка якості буде відємна .

Щоб мати можливість використовувати малюнок відбитка в цій ситуації, ми вирішили використовувати тільки область високої якості друку, а саме блоки якості 1, 2 і 3 для порівняння. Тому ви можете отримати надійні результати навіть під час друку. Його можна вважати бідним.

Експерименти

В рамках даної роботи було проведено ряд експериментів за схемами, представленими на рис. 5.7 та 5.8. Для кожного експерименту якісні блоки визначалися по-різному.

Для експериментів було взято 800 відбитків із відкритої бази даних відбитків пальців FVC 2000 [12]. Алгоритми обчислення карт якості та виділення мінущій реалізовані на С# для платформи .NET 4.5.



Рис. 5.7 Обчислення шаблонів відбитків пальців



Рис. 5.8 Зіставлення шаблонів та обчислення помилок FMR, FNMR та EER

5.4 Зіставлення відбитків пальців за допомогою MCC SDK

Для порівнянь використовувалася .NET бібліотека MCC SDK. Можна виділити кілька етапів роботи:

- створення шаблонів відбитків (метод "MccSdk.CreateMccTemplate");
- установка параметрів зіставлення (метод "MccSdk.SetMccMatchParameters");
- зіставлення шаблонів відбитків (метод "MccSdk.MatchMccTemplates").

Шаблони створюються з урахуванням списку мінущій відбитка. Таким чином, для кожного випадку вибору якісних областей, бралися мінущії з цих областей.

Як параметри зіставлення було взято стандартні значення, надані бібліотекою, крім методу визначення локального подібності. Було використано метод LSS, як найшвидший.

Як було сказано раніше, схожість відбитків виражається числом від 0 (відбитки належать різним людям) до 1 (відбитки належать одній людині). Підсумкове значення ідентифікації або неідентифікації відбитків залежить від порогу.

Помилки FMR, FNMR та EER та порівняння результатів

Для кожної вибірки мінущій було обчислено помилки FMR, FNMR залежно від порогового значення. Також були обчислені EER, як точки збігу помилок FMR та FNMR, та були побудовані графіки DET-кривих (рис. 5.9).

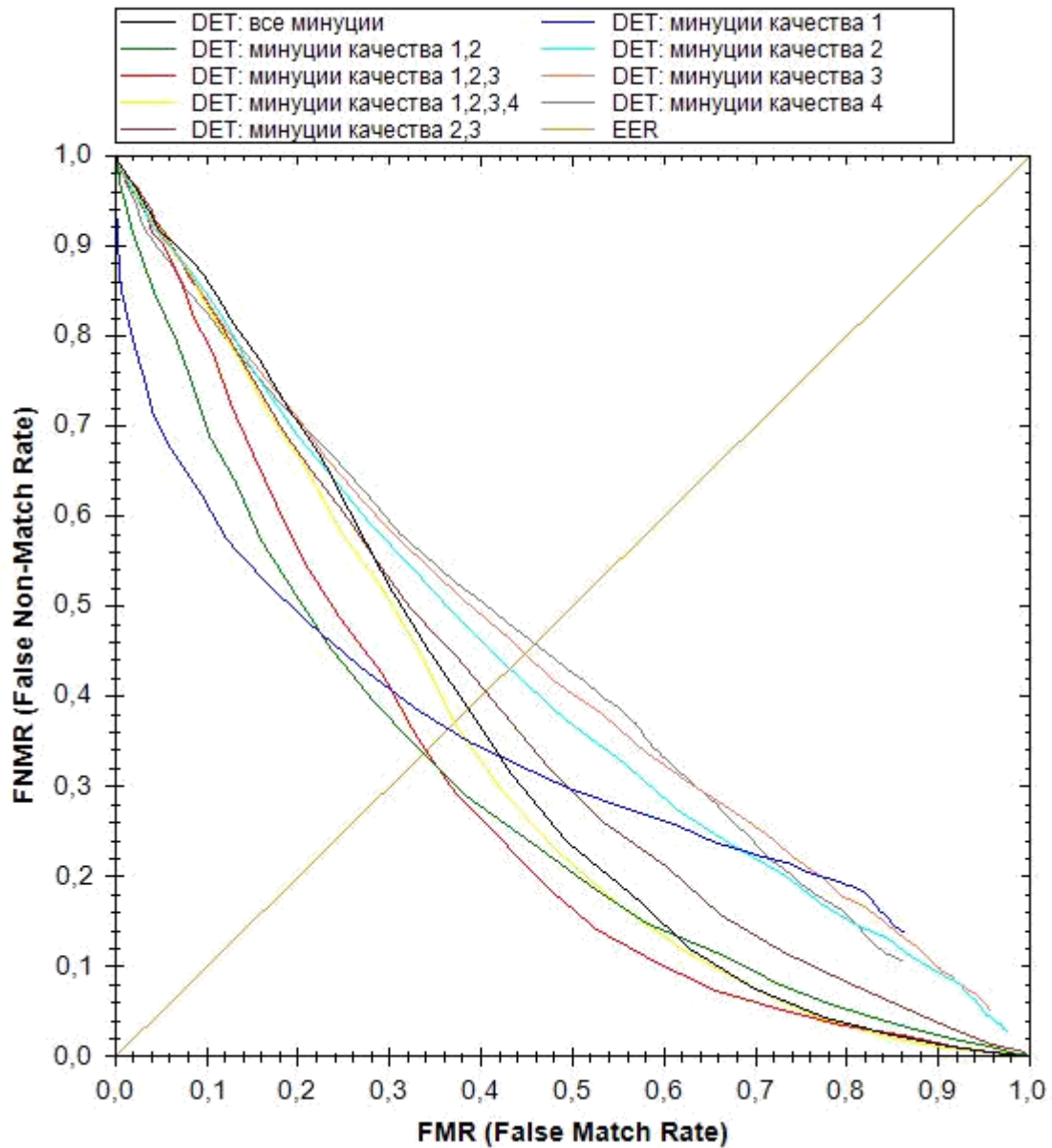


Рис. 5.9 DET-криві для різних вибірок мінуцій відповідно до їх якості

За результатами видно, що для невеликих вибірок мінуцій з блоків однієї якості 2 або 3 або 4, DET графік стає більше схожий на лінійну залежність $y = x$. Якщо дотримуватися такого вибору мінуцій, рішення системи ідентифікації схожі на випадкове ворожіння.

Також варто відзначити той факт, що у кожного з цих графіків максимальне значення ймовірності помилкового збігу FMR не дорівнює 1. Це

означає, що яким би малим не був поріг, все одно деякі відбитки, що належать різним людям, не співпадуть.

Особливо варто відзначити графік для мінущій якості 1. Видно, що в даному випадку максимальне значення обох ймовірностей не досягає 1, тому до описаного вище варто додати, що яким би високим був поріг, все одно деякі відбитки однієї людини будуть розпізнані як однакові. Причина виникнення подібних особливостей прихована у недостатній кількості даних.

Розглядаючи решту випадків видно два найбільш підходящі графіки: при вибірці мінущій якості 1, 2 і при вибірці 1, 2, 3. Видно, що значення EER у них досить близькі, більш точні результати можна подивитися в Таблиці 1. Можна сказати, що вибір мінущій якості 1, 2, 3 застосовні там, де помилкова ідентифікація не так критична, як помилкова неідентифікація. Нижче наведено результати обчислення EER для всіх випадків у порядку зменшення. Відразу варто зазначити, що значення дещо вищі, ніж у реальних біометричних систем, через точність алгоритму виділення мінущій. Видно, що найменше значення EER досягається при виборі мінущій якості 1, 2, однак, оскільки різниця між ними досить мала (різниця дорівнює 0.00337), то її можна інтерпретувати як статистичний шум при даній кількості відбитків, що обробляються.

Таблиця 5.1 Значення EER для різних вибірок мінущій

Якість мінущій	Порог	EER
Мінущія якості 1, 2	0,51894	0,33665
Мінущія якості 1, 2, 3	0,56711	0,34002
Мінущія якості 1	0,43436	0,36398
Мінущія якості 1, 2, 3, 4	0,59010	0,37073
Всі мінущії	0,59996	0,38696
Мінущія якості 2, 3	0,52336	0,40525
Мінущія якості 2	0,43923	0,43107

Мінуція якості 3	0,42647	0,44775
Мінуція якості 4	0,41888	0,45867

6 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ

6.1 Організація охорони праці при роботі з системою управління

Охорона праці розглядає проблеми забезпечення здорових і безпечних умов праці. Виявляє і вивчає можливі причини нещасних випадків, професійних захворювань, аварій, вибухів, пожеж і розробляє систему заходів і вимог з метою виключення цих причин і створення безпечних і сприятливих для людини умов праці.

Завдання охорони праці є зведення до мінімуму імовірності пошкодження або захворювання працівників з одночасним забезпеченням комфорту при максимальній продуктивності праці.

Навчання працівників безпеці праці проводять відповідно до вимог ГОСТ 12. 0.004 - 79, який встановлює порядок і види навчання. На всіх підприємствах і в організаціях незалежно від характеру і ступеню небезпеки виробництва навчання працівників проводять при підготовці нових робітників, проведенні різноманітних видів інструктажів і підвищенні кваліфікації.

Контроль за своєчасним і якісним навчанням виконує відділ охорони праці чи інженер з охорони праці, або ІТП, на якого наказом керівника підприємства покладено ці обов'язки. Ті, що вперше поступають на роботу, навчання проходять згідно з "Типовим положенням про підготовку і підвищення кваліфікації робітників". В журналі обліку навчальної роботи реєструють навчальну тему, за якою проводилось навчання.

Інструктаж працюючих поділяють на вступний, початковий, на робочому місці, повторний, позаплановий і початковий.

Вступний інструктаж з усіма, хто поступає на роботу незалежно від їх освіти і стажу роботи по даній професії, проводить інженер з охорони праці за програмою, затвердженою головним інженером підприємства, про проведення вступного інструктажу з обов'язковим підписом того, хто проводив інструктаж і того, хто його отримував.

Початковий інструктаж на робочому місці, повторний, позаплановий і поточний проводить керівник робіт.

Початковий інструктаж на робочому місці проводять при прийомі на роботу нових робітників за інструкцією з охорони праці, розробленою для окремих професій або видів робіт. Всі робітники після цього інструктажу і перевірки знань 2-5 змін (залежно від навичок і стажу роботи) працюють під наглядом бригадира чи майстра, потім оформляється допуск до їх самостійної праці.

Повторний інструктаж проходять всі працівники незалежно від кваліфікації, освіти і стажу роботи через три місяці. Його проводять з метою перевірки знання робітниками правил і норм з охорони праці.

Позаплановий інструктаж проводять коли змінилися правила охорони праці або технологічний процес, обладнання, інструмент та інші фактори, що впливають на безпеку праці; коли працівники порушують правила охорони праці, що можуть призвести чи призвели до травм, аварій чи пожежі, вибуху. Його проводять індивідуально чи з групою робітників однієї професії за програмою початкового інструктажу на робочому місці. При його реєстрації вказують причину, яка спричинила його проведення.

Умови праці мають велике значення практично для всіх виробничих показників - продуктивності праці, якості робіт, безпеки працівників та інше.

Санітарно-гігієнічні умови праці характеризуються показниками виробничого середовища - рівнем освітлення, мікрокліматичними параметрами, загазованістю і запиленістю повітряного середовища, рівнем шуму і вібрації, наявністю іонізуючого випромінювання та інше.

6.2 Електробезпека

Електричні установки, з якими доводиться мати справу практично всім працюючим по встановленню та налагодженню засобів автоматизації, виявляють для людини велику потенційну небезпеку, яка збільшується у зв'язку з тим, що органи чуття людини не можуть на відстані виявити присутність електричної напруги на обладнанні.

Степінь ураження електричним струмом залежить від цілого ряду факторів: значення сили струму, електричного опору тіла людини та тривалості протікання через неї струму, виду та частоти струму, індивідуальних властивостей людини та умов навколишнього середовища.

Конструкція електроустановок має відповідати умовам їх експлуатації та забезпечувати захист персоналу від дотику з струмоведучими та рухомими частинами, а обладнання - від попадання всередину посторонніх твердих тіл та води.

Конструкція, вид виконання, спосіб встановлення, клас ізоляції застосовуваних провідників, кабелів, пристроїв та іншого електрообладнання відповідають вимогам електробезпеки. За ступенем ураження людей електричним струмом котельня відноситься згідно ПУЕ 1.1.13 до категорії приміщень з підвищеною небезпекою (висока температура, можливість одночасного дотику до металевих елементів технологічного обладнання або металоконструкцій будинку та металевих корпусів електрообладнання).

У нормальному режимі роботи обладнання - можливість ураження працівників електричним струмом виключена. Але на випадок аварії для запобігання ураження струмом людей передбачене захисне заземлення. Згідно ПУЕ 1.7.65 допустимий опір заземлення повинен бути не більшим 10 Ом.

При виконанні монтажних робіт використовуються переносні електроінструменти (електродрилі, електрошліфувальні установки, тощо). Для забезпечення безпечної праці корпуси однофазних електроприймачів повинні занулюватись.

Захист людини від ураження електричним струмом в мережах з зануленням здійснюється тим, що при замиканні одної з фаз на занулений корпус в ланці цієї фази виникає струм короткого замикання, що діє на струмовий захист (плавкий запобіжник, автомат), в результаті чого відбувається відключення аварійної ділянки від мережі. Крім того, ще до спрацювання захисту струм короткого викликає перерозподіл напруги в мережі, що приводить до зниження напруги корпусу відносно землі. Таким чином, занулення зменшує напругу дотику та обмежує час, на протязі якого людина, що доторкнулася до корпусу, може потрапити під дію напруги.

Для того, щоб забезпечити швидке (на протязі декількох секунд) відключення аварійної ділянки, струм короткого замикання повинен бути достатньо великим. Відповідно до вимог ПУЕ струм короткого замикання повинен не менше ніж в три рази перевищувати номінальний струм плавкої вставки найближчого запобіжника або номінальний струм нерегульованого розчеплювача автоматичного вимикача. При використанні автоматичних вимикачів, що мають тільки електромагнітний розчіплювач (відсічку), струм короткого замикання повинен перевищувати значення струму встановлення миттєвого спрацювання в 1,25-1,4 рази в залежності від номінального струму.

В однофазних електроприймачів, що включені між фазним та нульовим робочим проводами, занулення корпусів слід виконувати з допомогою окремого (третього) провідника, який повинен з'єднувати корпус електроприймача з нульовим захисним проводом. В таких випадках під'єднувати корпуси електроприймачів для забезпечення електробезпеки до нульового робочого проводу недопустимо, оскільки при його розриві

(перегоранні запобіжника) всі під'єднані до нього корпуси виявляться під фазною напругою відносно землі.

В мережі з зануленням недопустимо використовувати заземлення окремих електроприймачів, не під'єднавши їх перед цим до нульового захисного провідника. В цьому випадку при замиканні фази на заземлений, але не приєднаний до нульового захисного провідника корпус створюється коло струму через заземлення цього корпусу та заземлення нейтралі джерела струму. Такий випадок небезпечний, оскільки засоби захисту не зможуть відключити такий електроприймач через мале значення струму і тому небезпечна напруга на всіх корпусах може зберігатися тривалий період, поки заземлений приймач не буде відключений вручну.

Важливо відмітити, що якщо занулений корпус одночасно заземлений, то це тільки покращує умови безпеки, оскільки забезпечує додаткове заземлення нульового захисного проводу.

Для ізоляції людини від частин електроустановок, що знаходяться під напругою, використовуються основні та допоміжні ізолюючі засоби, а саме слюсарно-монтажний інструмент з ізольованими ручками, коврики, ізолюючі підставки, тощо.

У приміщеннях, де знаходяться вимірювальні прилади, необхідно забезпечити виконання заходів по боротьбі з статичною електрикою (тобто прилади повинні бути заземлені). Найпростішим засобом є підтримка відносної вологості повітря на рівні 50 - 60 % за допомогою побутового електрозволожувача.

Підлогу слід виконувати відповідно до ГОСТ 12.4.124-83, використовуючи антистатичне покриття на проходах і біля робочих місць.

Робітникам рекомендовано носити одягу з природних матеріалів або з комбінованих - природних і штучних волокон. Для зняття електростатичних зарядів з одягу слід використовувати антистатика побутового призначення.

Оскільки корпуси приладів виконані з металу, то для усунення небезпеки ураження людини електричним струмом (можливий пробій на корпус приладу) використовується захисне заземлення.

6.3 Розрахунок заземлення

Розрахуємо систему заземлення для електроустаткування, яке працює від напруги 220 В.

$$R_{\text{ззз}} \leq \frac{U}{I_p} = \frac{220}{66} = 3.3 \leq 4 \text{ Ом}$$

Визначаємо опір ґрунту: $\rho = k_n * \rho_n = 2 * 200 = 400 \text{ Ом м}$,

де k_n - коефіцієнт підсилення;

ρ_n — питомий опір ґрунту (вибирається з довідкової літератури).

Визначаємо опір одиночного вертикального заземлювача:

$$R_B = \frac{\rho}{2\pi} \left(\ln \frac{2l}{d} + \frac{1}{2} * \frac{4t+1}{4t-1} \right)$$

де t - відстань від середини заземлювача до поверхні ґрунту, м;

l, d - довжина і діаметр стержня заземлювача, м;

$$R_B = 96 \text{ Ом.}$$

Визначаємо опір сталевій полосі, що з'єднує стержневі заземлювачі:

$$R_{II} = (\rho / 2\pi b) * \ln(l^2 / dt) = 61 \text{ Ом.}$$

Визначаємо орієнтовне число стержневих заземлювачів:

$$n = R_B / [r_B] \eta_B = 96 / 4 * 1 = 24 \text{ шт.};$$

r_B - допустимий по нормам опір заземлюючого пристрою,

η_B - коефіцієнт використання вертикальних заземлювачів (для орієнтовного розрахунку приймається рівним 1).

Приймаємо розміщення вертикальних заземлювачів по контуру з відстанню між сталевими заземлювачами рівним 2l. З довідкової літератури визначаємо $\eta_B = 0,66$ і $\eta_r = 0,39$.

Визначаємо необхідну кількість вертикальних заземлювачів

$$n = R_B / [r_B] \eta_B = 96 / (4 * 0.66) = 36$$

Розраховуємо загальний розрахунковий опір аземлюючого пристрою R з врахуванням з'єднувальної полоси

$$R = R_B R_{II} / (R_B \eta_r + R_{II} \eta_B n) = 3.9 \text{ Ом.}$$

Розрахунок проведено правильно, оскільки виконується умова $R \leq [r_B]$.

Розрахунок штучного заземлення:

Приймаємо, що опір захисного заземлення не повинен перевищувати 4

Ом:

$$R_{33} = \frac{R_c R_n}{R_c + R_n} \leq 4 \text{ Ом}$$

де R_{33} – опір захисного заземлення;

R_c – опір стержневих заземлювачів;

R_{II} – опір поперечних заземлювачів.

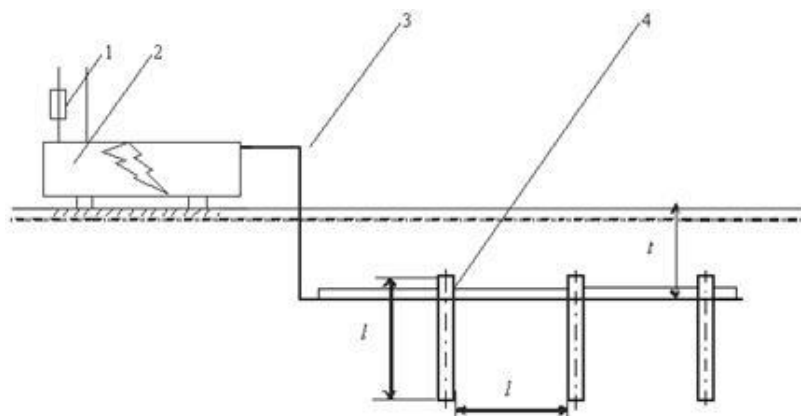


Рисунок 6.1 - Пристрій заземлення

4 – плавка вставка; 2 – електроустановка; 3 – з'єднувальна штаба; 4 – трубчастий заземлювач

Опір одиночного стержневого заземлювача розтіканню електричного струму:

$$R_{oc} = \frac{\rho_{\text{г}}}{2\pi l} \left(\ln \frac{2l}{d} + \ln \frac{4h' + l}{4h' - l} \right)$$

де h – відстань від поверхні ґрунту до заземлювача і становить 0,8 м;

l – довжина стержневого заземлювача 3 м;

d – діаметр стержневого заземлювача 50 мм.

$$R_{oc} = \frac{750}{2 \cdot 3,14 \cdot 3} \left(\ln \frac{2 \cdot 3}{0,05} + \ln \frac{4 \cdot 0,8 + 3}{4 \cdot 0,8 - 3} \right) = 39,8 \cdot (0,18 + 3,43) = 143,8 \text{ Ом}$$

Опір одиночного поперечного заземлювача:

$$R_{ок} = \frac{\rho_{\text{г}}}{2\pi l} \ln \frac{2l^2}{bh'}$$

де l – довжина поперечного заземлювача 2,5 м;

b – ширина полоси заземлювача 30 мм;

$\rho_{\text{г}}$ – розрахунковий опір ґрунту: для поперечних електродів 1000 Ом·м, для стержневих електродів 750 Ом·м.

$$R_{ок} = \frac{1000}{2 \cdot 3,14 \cdot 2,5} \ln \frac{2 \cdot 2,5^2}{0,03 \cdot 0,8} = 63,7 \cdot 6,25 = 398,1 \text{ Ом}$$

В наслідок взаємовпливу вводимо коефіцієнт використання заземлювачів:

$$\eta = \frac{R_0}{nR_0}$$

де R_d – допустимий опір заземлення, що становить 4 Ом;

R_0 – опір одиночного заземлювача.

З цієї формули методом ітерацій підбирають n , при якому $\eta = 1$:

n	R_n	R_c	R_o	η
1	398,1	143,8	105,6	26,1
5	398,1	143,8	105,6	5,2
10	398,1	143,8	105,6	2,6
15	398,1	143,8	105,6	1,7
20	398,1	143,8	105,6	1,3
25	398,1	143,8	105,6	1,1
26	398,1	143,8	105,6	1,0
27	398,1	143,8	105,6	0,9

Отже приймаємо кількість одиночних заземлюючих електродів рівною 26.

ОСНОВНІ ВИСНОВКИ ДИПЛОМНОЇ РОБОТИ

В результаті проведеної роботи було проаналізовано основні параметри та чинники, які впливають на забезпечення підвищення якості ідентифікації при дактилоскопії. Створено прототип біометричної системи для розпізнавання відбитків пальців, заснованої на методах нечіткої логіки, дослідження якості даного алгоритму в порівнянні з іншими методами в даній області.

У ході виконання роботи було досягнуто наступних результатів.

- Вивчено предметну область систем розпізнавання відбитків пальців, а також основні принципи нечіткої логіки.
- Створено прототип платформи для прийняття рішень на основі нечіткої логіки (Трьохфазний алгоритм, що включає облік якості та підходить для мультибіометричної системи).
- Створено алгоритм та прототип програмної частини мультимодальної системи для розпізнавання відбитків пальців, що використовує систему прийняття рішень на основі нечіткої логіки, алгоритм розпізнавання мінуцій МСС та алгоритм оцінки якості відбитків NFIQ.
- Проведено порівняння результатів з альтернативними стратегіями прийняття рішень, що показав переваги алгоритму, що базується на нечіткій логіці порівняно з іншими.

БІБЛІОГРАФІЯ

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,
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||8 1&Req=&>
9. Minutia Cylinder-Code SDK
<http://biolab.csr.unibo.it/research.asp?organize=Activities&select=&selObj=82&pathSubj=111||8||8 2&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
12. Maio D., Maltoni D., Cappelli R., Wayman J. L., Jain A.K. FVC2000: Fingerprint verification competition // 2002.
13. Canny J. F. A computational approach to edge detection // Pattern Analysis and Machine Intelligence, IEEE Transactions on. 1986. №6. P. 679-698.
14. Duda, R. O., P. E. Hart Use of the Hough Transformation to Detect Lines and Curves in Pictures // Comm. ACM, 1972, Vol. 15, P. 11–15;

15. Viola P., Jones M. Rapid object detection using a boosted cascade of simple features // *Computer Vision and Pattern Recognition* 2001
16. Viola P., Jones M. Robust Real-time Object Detection // *International Journal of Computer Vision* 2001
17. Freund Y., Schapire R.E. A decision-theoretic generalization of on-line learning and an application to boosting. In *Computational Learning Theory: Eurocolt '95*, pages 23–37. Springer-Verlag, 1995.

Додаток

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