

Uniwersytet Rzeszowski
Wydział Nauk Ścisłych i Technicznych
Instytut Informatyki



BIOMETRYCZNE SYSTEMY ZABEZPIECZEŃ

INSTRUKCJA DO ĆWICZEŃ LABORATORYJNYCH

TREŚCI KSZTAŁCENIA: OBRAZY DAKTYLOSKOPIJNE TOKENIZACJA- METODA PROSTA I GRAFOWA , TOKENIZACJA
TĘCZÓWKI OKA ALGORYTM JOHNA DAUGMANA

Spis treści

1.	Cele laboratorium	2
2.	Wprowadzenie.....	2
3.	Zadania do samodzielnego rozwiązania	10

1. Cele laboratorium

Celem laboratorium jest zrozumienie procesu tokenizacji w kontekście danych biometrycznych oraz praktyczna implementacja algorytmów do ekstrakcji cech z obrazów daktyloskopijnych (minucie) oraz porównanie prostych i grafowych metod dopasowania tokenów. Zastosowanie algorytmu Johna Daugmana do tokenizacji pola tęczówki oka oraz wyznaczanie miary podobieństwa poprzez obliczenie dystansu Hamminga.

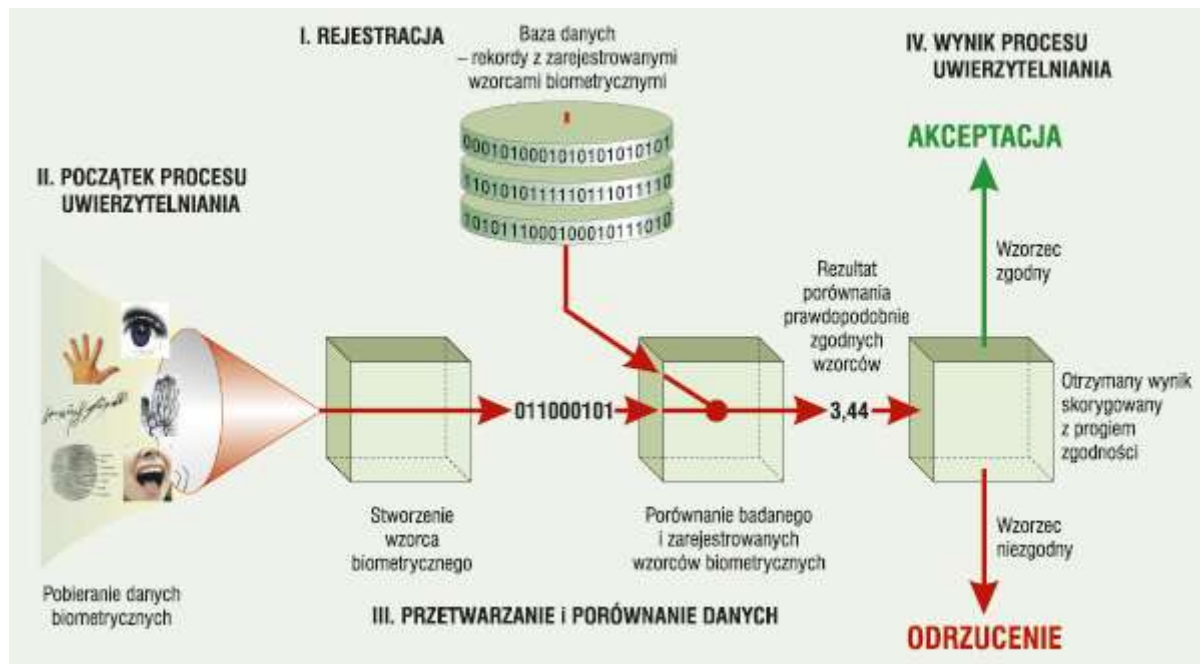
2. Wprowadzenie

Obrazy daktyloskopijne (ang. fingerprint images) to obrazy przedstawiające wzór linii papilarnych palca, składający się z charakterystycznych grzbietów (ridge) i dolin (valley). Te lokalnie periodyczne struktury są unikalne dla każdej osoby i stanowią podstawę systemów biometrycznych opartych na rozpoznawaniu odcisków palców.

Etapy przetwarzania obrazu daktyloskopijnego dotyczą następujących etapów, które można zestawić w poniższej sekwencji działań algorytmicznych.

1. Pozyskiwanie obrazu (Image Acquisition) - rejestracja obrazu odcisku palca przy użyciu skanera optycznego, pojemnościowego, ultradźwiękowego itp.
2. Wstępne przetwarzanie (Preprocessing) - korekcja kontrastu, usuwanie szumu, poprawa jakości obrazu.
3. Wzmocnienie i filtrowanie (Enhancement & Filtering) - zastosowanie filtrów (np. Gabora) w celu uwydatnienia struktury linii papilarnych.
4. Binarizacja i ścienianie (Binarization & Thinning) - przekształcenie obrazu do formy binarnej (0/1) i redukcja grubości linii do 1 piksela.
5. Ekstrakcja minucji (Minutiae Extraction) - wykrywanie punktów charakterystycznych (np. zakończenia linii – ending, rozwidlenia – bifurcation).

Na Rys. 1 przedstawiono schemat ogólny przetwarzania i autentykacji materiału biometrycznego.



Rys. 1. Etapy przetwarzania obrazu daktyloskopijnego/tęczówki oka.

Modele matematyczne i techniki dopasowywania

Parametry techniczne obrazu linii papilarnych:

Obraz odcisku palca można matematycznie opisać jako funkcję dwuwymiarową:

$$I(x, y): Z^2 \rightarrow [0, 255] \quad (1)$$

gdzie $I(x, y)$ – wartość jasności piksela o współrzędnych (x, y) , zakres $[0, 255]$ oznacza skalę szarości (8-bitowy obraz), obraz jest zwykle rozmiaru $W \times H$, np. 512×512 pikseli. Typowe obrazy daktyloskopijne mają rozdzielczość 500 dpi, co odpowiada około $19,7 \mu\text{m}$ na piksel. Taka dokładność pozwala uchwycić detale niezbędne do wykrycia minucji (np. rozwidleń i zakończeń linii papilarnych).

Obrazy odcisków palców wykazują lokalnie okresowe wzorce - struktury grzbietów i dolin. Filtrowanie oparte na częstotliwości służy do zwiększenia wyrazistości linii papilarnych, tłumienia szumów oraz poprawy niezawodności wykrywania minucji.

Modele matematyczne używane w przetwarzaniu materiałów daktyloskopijnych:

Filtracja przestrzenna – filtr Gabora, który łączy funkcję Gaussa (lokalizacja przestrzenna) z falą sinusoidalną (analiza częstotliwości). Jest pasmowo-przepustowym filtrem kierunkowym, często używanym w analizie obrazów biometrycznych (daktyloskopia, tęczówka), tekstur oraz detekcji krawędzi. Filtr wykorzystywany jest w przetwarzaniu obrazów daktyloskopijnych, ponieważ wzmacnia lokalne struktury grzbietów (ridge), tłumi szum oraz działa selektywnie względem częstotliwości i kierunku. Matematycznie filtr Gabora w dwóch wymiarach definiuje się jako:

$$G(x, y) = \exp\left(-\frac{x'^2 + \gamma^2 y'^2}{2\sigma^2}\right) \cdot \cos\left(2\pi \frac{x'}{\lambda} + \phi\right) \quad (2)$$

ze współrzędnymi podlegającymi możliwej rotacji maski filtru:

$$\begin{cases} x' = x\cos\theta + y\sin\theta \\ y' = -x\sin\theta + y\cos\theta \end{cases} \quad (3)$$

gdzie:

λ - długość fali (częstotliwość grzbietu),

θ kąt orientacji lokalnych grzbietów,

ϕ przesunięcie fazowe (zazwyczaj 0 lub $\pi/2$),

σ - szerokość obwiedni gaussowskiej (skala),

γ współczynnik kształtu przestrzennego (eliptyczność).

Pierwszy człon (funkcja Gaussa): ogranicza analizę do lokalnego obszaru obrazu. Drugi człon (część zespolona): odpowiada za analizę częstotliwości linii i ich lokalnego kierunku. Całość: filtr przestrzenny czuły na określoną częstotliwość i orientację – doskonały do wzorców takich jak linie papilarne.

Zastosowanie w przetwarzaniu odcisków palców:

$$I'(x, y) = I(x, y) * G_{\theta(x, y), \lambda(x, y)}(x, y) \quad (4)$$

gdzie $*$ oznacza operator splotu. W wyniku otrzymuje się poprawiony kontrast grzbietów linii papilarnych, redukuje się szumy i rozmycia tła. Zabieg ten pozwala na bardziej skuteczną ekstrakcję minucji i szacowanie przepływu linii papilarnych.

Transformacje w tym binaryzacja (np. metodą Otsu) – zamiana obrazu na postać czarno-białą oraz ścienianie/szkieletyzacja (thinning) – sprowadzenie linii grzbietów do jednego piksela szerokości.

3. Tokenizacja obrazów daktyloskopijnych

Model punktów minucyjnych - zbiór punktów reprezentujących cechy linii (zakończenia, rozwidlenia) jako:

$$M = [x, y, \theta] \quad (5)$$

Tworzą tzw. token biometryczny, który może zostać użyty do dopasowywania odcisków palców.

Wśród technik dopasowania można wyróżnić metodę:

- Prostą, która porównuje pozycję i orientację minucji. Jest to szybka i prosta metoda, ale czuła na przesunięcia i rotacje obrazu wejściowego.
- Grafowa, która porównuje relację między minucjami (odległość oraz kąty). Metoda ta odporna jest na deformacje i częściowe rotacje lecz jej złożoność obliczeniowa jest większa.

False Minutiae Removal – usuwanie fałszywych minucji

W procesie przetwarzania obrazu odcisku palca (zwłaszcza po szkieletyzacji) mogą pojawiać się fałszywe minucje (false minutiae), które nie odpowiadają rzeczywistym cechom linii papilarnych.

Typowe źródła fałszywych minucji:

- Zakończenia (endings) przy krawędzi obrazu – powstają, gdy linia jest obcięta przez granicę obrazu linii papilarnych.
- Zbyt bliskie rozwidlenia (bifurcations) – mogą być artefaktem powstałym na etapie przetwarzania morfologicznego obrazu.
- Zakłócenia spowodowane szumem, niepełnym odciskiem, zniekształceniem, niską jakością obrazu.

Metody usuwania fałszywych minucji

1. Heurystyki geometryczne (regułowe) - stosuje się zestaw reguł (ang. rule-based filtering):

- Odległość między minucjami: jeśli dwa punkty minucji znajdują się bardzo blisko siebie (np. <10 pikseli), mogą być artefaktem.
- Kąt orientacji: jeśli orientacje są zbyt podobne lub całkowicie rozbieżne, a punkty są blisko – można je odrzucić.
- Pozycja względem granic obrazu: zakończenia w pobliżu krawędzi często są fałszywe.
- Sąsiedztwo topologiczne: analiza liczby linii w otoczeniu (np. w 3×3).

2. Ocena jakości minucji

Dla każdej minucji można przypisać miarę jakości $Q_m(i)$, uwzględniającą lokalny kontrast i wyrazistość grzbietów, gładkość orientacji, stabilność po przefiltrowaniu, odległość od innych minucji. Minucje o $Q_m(i) < threshold$ można odrzucić.

3. Metody uczenia maszynowego (ML-based filtering) - coraz częściej wykorzystuje się modele ML do klasyfikacji minucji jako prawdziwa (true) i fałszywa (false). Warunkiem koniecznym jest skonstruowanie właściwego ciągu uczącego i poprowadzenie nadzorowanego procesu uczenia.

Przykład 1. Proste dopasowywanie minucji (porównanie x,y, kąta orientacji minucji).

Token = list of minutiae: (x, y, θ)

Matching = compare each minutia from print A to print B using distance + orientation threshold.

```
% Simple Minutiae Matching Tokenization and Comparison
% -----
% Token format: M = [x, y, theta] for each minutia
% Example minutiae tokens for two fingerprints
M1 = [100 150 pi/4;
      200 250 pi/2;
      300 350 pi/6];

M2 = [102 152 pi/4.1;
      199 248 pi/2.05;
      400 450 pi/3];

% Thresholds
pos_thresh = 10; % Max distance in pixels
angle_thresh = deg2rad(10); % Max angular difference

% Matching procedure
matches = 0;
for i = 1:size(M1,1)
    for j = 1:size(M2,1)
        d = norm(M1(i,1:2) - M2(j,1:2)); % Euclidean distance
        a = abs(M1(i,3) - M2(j,3)); % Orientation difference
        if d < pos_thresh && a < angle_thresh
            matches = matches + 1;
            break; % One match per minutia
        end
    end
end
% Similarity score
score = matches / min(size(M1,1), size(M2,1));
fprintf('Simple Matching Score: %.2f\n', score);
```

Przykład 2. Grafowe dopasowanie minucji – reprezentacja minucji jako wierzchołków grafu i analiza relacji między nimi:

Token = graph: nodes = minutiae, edges = distances & relative angles

Matching = compare edge structures

```
function edges = build_edges(M)
    edges = [];
    N = size(M,1);
    for i = 1:N
        for j = i+1:N
            dist = norm(M(i,1:2) - M(j,1:2));
            angle = atan2(M(j,2) - M(i,2), M(j,1) - M(i,1));
            edges(end+1, :) = [i, j, dist, angle];
        end
    end
end

% Example minutiae positions for two fingerprints
M1 = [100 150 pi/4;
      200 250 pi/2;
      300 350 pi/6];
M2 = [102 152 pi/4.1;
      199 248 pi/2.05;
      400 450 pi/3];
% Build edge lists
edges1 = build_edges(M1);
edges2 = build_edges(M2);
% Graph matching: count matching edges
dist_thresh = 15;
angle_thresh = deg2rad(15);
```

```

matched_edges = 0;
for i = 1:size(edges1,1)
    for j = 1:size(edges2,1)
        d_diff = abs(edges1(i,3) - edges2(j,3));
        a_diff = abs(edges1(i,4) - edges2(j,4));
        if d_diff < dist_thresh && a_diff < angle_thresh
            matched_edges = matched_edges + 1;
            break;
        end
    end
end
end
% Normalize score
score_graph = matched_edges / min(size(edges1,1), size(edges2,1));
fprintf('Graph Matching Score: %.2f\n', score_graph);

```

4. Algorytm Johna Daugmana – tokenizacja pola tęczówki

Tokenizacja to proces przekształcenia obrazu tęczówki oka w unikalny, cyfrowy kod biometryczny (tzw. iris code), który może być później użyty do jednoznacznej identyfikacji osoby. Algorytm Johna Daugmana jest jednym z najbardziej znanych i skutecznych podejść do tego zadania.

Główne etapy algorytmu Daugmana

- Segmentacja - lokalizacja tęczówki i źrenicy oraz usunięcie powiek, rzęs i refleksów świetlnych. W tym etapie otrzymuje się wyizolowany pierścień tęczówki- zwykle aplikowana jest transformata Paula Hougha w wersji kołowej.
- Normalizacja (Daugman Rubber Sheet Plate Model) - przekształcenie współrzędnych biegunowych (okrągła tęczówka) do prostokątnej siatki o stałym rozmiarze, np. 64×512. Etap ten pozwala na porównanie tęczówek różnych rozmiarów i położenia. Kolejno down sampling do standardu 8×126 pikseli.
- Kodowanie (Feature Encoding) – operacja splotu z wykorzystaniem zespolonego filtru Gabora i enkrypcja fazy odpowiedzi filtru jako kodu binarnego (np. 2-bitowego). Wynikiem tego etapu jest Iris code – zwarty wektor danych (np. 2048 bitów).
- Porównywanie (Matching) - obliczanie odległości Hamminga między kodami tęczówek, z uwzględnieniem masek wskazujących nieczytelne fragmenty (refleksy pola rogówki) lub części niewidoczne ze względu na położenie powiek lub rzęs.

Model obrazu tęczówki

Tęczówka jest przechwytywana jako obraz w skali szarości:

$$I(x, y): Z^2 \rightarrow [0, 255] \quad (6)$$

Typowy rozmiar: ~320×240, 8-bitowa skala szarości. Celem jest wyodrębnienie wyraźnych, stabilnych cech tekstury z pierścieniowego obszaru tęczówki między źrenicą a twardówką. Matematycznie pole pierścienia wyznaczają okręgi wykryte z wykorzystaniem kołowej transformaty Hougha:

Pupil: (x_p, y_p, r_p)

Iris: (x_i, y_i, r_i)

Iris Normalization, czyli tzw. Model Kauczukowej Płyty Daugmana (Daugman's Rubber Sheet Plate Model), który jest kluczowym krokiem w procesie rozpoznawania tęczówki. Tęczówka u każdej osoby ma unikalny wzór tekstury, ale obrazy tęczówek różnią się:

- rozmiarem (zmienna odległość od kamery),
- pozycją (niedoskonała rejestracja),
- kształtem (naturalne zniekształcenia, np. powieki, źrenica).

Aby umożliwić porównanie różnych obrazów tęczówek, należy je przekształcić do wspólnego, normalizowanego układu współrzędnych – niezależnego od rozmiaru i pozycji oka.

Model zakłada przekształcenie pierścienia tęczówki z układu biegunowego (polarny) na kartezjański na czyli na prostokątną siatkę o ustalonej rozdzielczości (np. 64×512 piksele). Rozmiar w potęgę 2 ze względu na szybkość przekształceń częstotliwościowych.

Przekształcenie współrzędnych wykonywane jest dla każdego punktu w znormalizowanym obrazie, o współrzędnych (r, θ) . Wyznaczane są współrzędne w obrazie źródłowym:

$$x(r, \theta) = (1 - r)x_p(\theta) + rx_i(\theta) \quad (7)$$

$$y(r, \theta) = (1 - r)y_p(\theta) + ry_i(\theta) \quad (8)$$

gdzie: $r \in [0,1]$ odległość radialna między źrenicą a zewnętrzną krawędzią tęczówki, $\theta \in [0,2\pi]$ kąt obrotu wokół środka źrenicy, (x_p, y_p) współrzędne punktu na brzegu źrenicy (pupil boundary), (x_i, y_i) współrzędne punktu na brzegu tęczówki (iris boundary)

Tęczówkę można wyobrazić sobie jako gumowy pierścień. Model Daugmana „rozciąga” ją na prostokątną siatkę, zachowując rozkład tekstury włókien tworzących tęczówkę. Normalizacja umożliwia porównanie tęczówek różnych osób w tym samym układzie, ogranicza wpływ zmienności źrenicy (np. pod wpływem światła) oraz przygotowuje dane do późniejszego etapu kodowania (np. przez filtry Gabora).

Model filtru Gabora w algorytmie Daugmana stosowany jest do wyodrębnienia fazy sygnału lokalnego → wykorzystywany do binarnego kodowania.

$$G(x, y) = \exp\left(-\frac{x'^2 + y'^2}{2\sigma^2}\right) \cdot \exp(j2\pi f x') \quad (9)$$

Przykład 3. Kodowanie Gabora 2D

```
function iris_feature_encoding_demo(iris_name)
% Simulated normalized iris (e.g., 64x512)
%iris = rand(64, 512); % Replace with real normalized iris data
iris = imread(iris_name)';
% Gabor filter parameters
lambda = 10; % Wavelength
theta = 0; % Orientation (radians)
sigma = 4.0; % Gaussian envelope
gamma = 0.5; % Spatial aspect ratio
psi = 0; % Phase offset
% Create Gabor filter
gaborSize = 21;
[x, y] = meshgrid(-floor(gaborSize/2):floor(gaborSize/2), -
floor(gaborSize/2):floor(gaborSize/2));
x_theta = x * cos(theta) + y * sin(theta);
y_theta = -x * sin(theta) + y * cos(theta);
G = exp(-(x_theta.^2 + (gamma^2) * y_theta.^2) / (2 * sigma^2)) .* exp(1i * (2 * pi *
x_theta / lambda + psi));

% Apply Gabor filter
filtered = conv2(iris, G, 'same');
% Phase quantization (2-bit)
phase = angle(filtered);
iris_code = zeros(size(phase));
iris_code(phase >= -pi & phase < -pi/2) = 0; % 00
iris_code(phase >= -pi/2 & phase < 0) = 1; % 01
iris_code(phase >= 0 & phase < pi/2) = 2; % 10
iris_code(phase >= pi/2 & phase <= pi) = 3; % 11
% Display results
figure;
subplot(1,3,1); imagesc(iris); title('Normalized Iris'); colormap gray;
subplot(1,3,2); imagesc(real(filtered)); title('Filtered (Real Part)'); colormap gray;
subplot(1,3,3); imagesc(iris_code); title('Encoded Iris Code (2-bit)'); colormap gray;
end
```


Dystans Hamminga – wyznaczanie dopasowania pary tokenów

Obliczany jest tylko dla ważnych (niezamaskowanych) bitów poprzez porównanie dwóch kodów tęczówki C_1 i C_2 przy użyciu:

$$HD = \frac{1}{N} \sum_{i=1}^N (C_1(i) \oplus C_2(i)) \cdot M(i) \quad (10)$$

Gdzie: \oplus : XOR, $M(i)$ - maska wskazująca prawidłowe (niewykluczone) bity, N - liczba prawidłowych bitów

Przykład 4.

```
function hd = iris_hamming_distance(code1, code2, mask1, mask2)
% Inputs:
% code1, code2: matrices of same size with 2-bit iris codes (0-3)
% mask1, mask2: binary masks (1 = valid, 0 = occluded)
% Ensure input sizes match
if any(size(code1) ~= size(code2)) || any(size(mask1) ~= size(mask2))
    error('Code and mask sizes must match.');
```

```
end
% Combine masks to exclude occluded bits
valid_mask = mask1 & mask2;
% Count total valid bits
N = sum(valid_mask(:));
if N == 0
    hd = NaN; % No valid comparison possible
    return;
end
% XOR difference
diff = bitxor(uint8(code1), uint8(code2));
% Only count where valid
hd = sum(diff(valid_mask) > 0) / N;
end
```

5. Zadania do samodzielnego rozwiązania

Zadanie 1.

Wykonaj tokenizację wybranej pary obrazów daktyloskopijnych i wyznacz podobieństwa zestawów wyodrębnionych minucji z wykorzystaniem metody prostej oraz grafowej. Porównaj skuteczność obu metod.

Zadanie 2.

Wykonaj tokenizację wybranej podanego zestawu obrazów tęczówki oka i wyznacz dla każdej pary dystans Hamminga HD. Oceń otrzymane wyniki ze względu na podobieństwo i poziom okluzji pola tęczówki.