

Ćwiczenie 3: Biometria tęczówki

Zadania do wykonania:

1. Rejestracja zdjęć tęczówki z wykorzystaniem kamery *IrisGuard AD100* (trzy zdjęcia dla każdego oka, łącznie 6 zdjęć).
(Pobrać dane - przycisk **capture**, nazwę plików zaczynać od **prefixu** - *bit* i podać **numer stanowiska**, np. *'bit8'*. Pliki powinny mieć nazwę: **bit1_Ca_L0.bmp**.)
2. Wykorzystać skrypt **BIO_IRIS_seg** w celu ręcznej segmentacji tęczówki dla wszystkich zdjęć własnych tęczówek.

np. `BIO_IRIS_seg('bit2',1,'L','C')` - kolejne argumenty: nazwa użytkownika, numer pomiaru (0..2), oznaczenie oka ('L','R'), typ czytnika: ('C' - CrossMatch ISCAN 2, 'I' - IrisGuard) - przeczytać opis!

Segmentacja:

- ustawienie kursora na środku źrenicy i wciśnięcie przycisku **'p'**, ustawienie kursora na okręgu źrenicy i przyciśnięcie **'p'**, naciśnięcie lewego przycisku myszy poza obszarem zdjęcia oka
- ustawienie kursora na środku tęczówki (obiekty zazwyczaj nie są współśrodkowe) i wciśnięcie przycisku **'i'**, ustawienie kursora na okręgu tęczówki i przyciśnięcie **'i'**, naciśnięcie lewego przycisku myszy poza obszarem zdjęcia oka. - zaznaczenie zakłóceń - **'s'**, czyszczenie zakłóceń - **'c'**, zakończenie działania programu - **'x'**

W wyniku działania tego skryptu otrzymamy:

- wynik segmentacji - zaznaczone obszary istotne z punktu rozpoznawania tożsamości - **bit1_Cm_L0_m.bmp**
- obrazy tęczówek w układzie biegunowym - **bit1_Cm_L0_p.bmp**
- maski obrazów tęczówek w układzie biegunowym wskazującymi zlokalizowane zakłócenia - **bit1_Cm_L0_pm.bmp**

Po zakończeniu segmentacji dla wszystkich obrazów wywołać skrypt **BIO_IRIS_scale**

```
BIO_IRIS_scale('bit2'),
```

który na podstawie surowych obrazów oraz wyników automatycznej i ręcznej segmentacji wygeneruje przeskalowane obrazy tęczówki wykorzystywane w kodowaniu w następnych etapach ćwiczenia.

Umieścić dane na *Dysku Sieciowym*.

3. Wykorzystać skrypt `iriscompare` w celu dobrania najlepszego filtra (oraz jego parametrów) tak, aby **zminimalizować** odległość Hamminga pomiędzy obrazami **tej samej tęczówki** jednocześnie **maksymalizując** odległość Hamminga pomiędzy obrazami **różnych tęczówek**. Po otrzymaniu zadowalających rezultatów zanotować:

- rodzaj filtra:, oraz
- parametry filtra:

W pierwszej kolejności wybrać pliki *'doKodowania'* zakończone na *'_s'*, a następnie z maskami *'_sm'*

4. Wykorzystać skrypt `BIO_IRIS_match` w celu wyznaczenia średnich wyników porównań tych samych *g* i różnych *i* tęczówek dla filtra wybranego w poprzednim kroku.

```
[genScores impScores] = BIO_IRIS_match(F, sigma, lambda, phi, R, M)
F - typ wykorzystywanego filtra ('LoG', 'Haar', 'Gabor', 'DoG'), sigma, lambda, phi
- wybrane parametry filtra, R - specyfikacja czytnika ('C' - CrossMatch ISCAN2, 'I'
- IrisGuard), M - tryb segmentacji ('m' - ręczna, 'a' automatyczna),
np. [genScores impScores] = BIO_IRIS_match('Gabor', 64, 32, 45, 'C', 'm');
```

Skorzystać ze skryptu `BIO_IRIS_EER(genScores, impScores)` w celu wyznaczenia średnich wyników porównań wewnątrzklasowych (*g*) i międzyklasowych (*i*) oraz błędu zrównoważonego *EER*.

Zanotować wyniki dla:

- segmentacji automatycznej:
 $g_a = \dots\dots\dots$, $i_a = \dots\dots\dots$,
 $EER_a = \dots\dots\dots$
- segmentacji ręcznej:
 $g_m = \dots\dots\dots$, $i_m = \dots\dots\dots$,
 $EER_m = \dots\dots\dots$

Komentarz do otrzymanych wyników: