

# **VYSOKÉ UČENIE TECHNICKÉ V BRNE**

Fakulta informačných technológií



**Biometrické systémy**

**2016/2017**

**Rozpoznávanie dúhovky**

Filip Gulán (xgulan00)

30.11.2016

Marek Marušic (xmarus05)

# Obsah

|                               |   |
|-------------------------------|---|
| 1 Úvod                        | 2 |
| 2 Popis biometriky            | 2 |
| 3 Popis biometrickej databáze | 2 |
| 4 Popis vlastného riešenia    | 3 |
| 4.1 Lokalizácia dúhovky       | 3 |
| 4.2 Extrakcia dúhovky         | 4 |
| 4.3 Normalizácia dúhovky      | 4 |
| 4.4 Získanie kódu dúhovky     | 4 |
| 4.5 Porovnanie kódov          | 5 |
| 5 Výsledky                    | 5 |
| 6 Záver                       | 7 |
| 7 Spustenie projektu          | 7 |

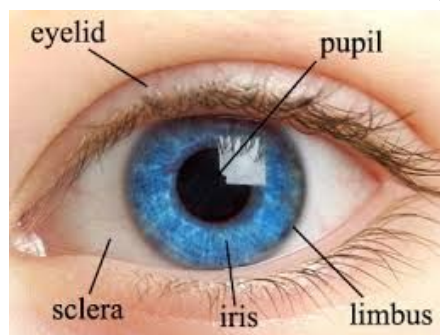
# 1 Úvod

Úlohou projektu bolo vytvoriť biometrický systém, ktorý rozpoznáva dúhovky oka a následne overiť jeho funkčnosť pomocou zvolenej biometrickej databázy. Systém je tvorený v C# pomocou BIOframework<sup>1</sup>, ktorý v sebe obsahuje Emgu CV, čo je vlastne wrapper nad OpenCV knižnicou pre spracovanie obrázkov.

Pre overovanie funkčnosti biometrického systému bola zvolená biometrická databáza CASIA Iris V1. Systém po vyhodnotení vizualizuje svoje výsledky formou grafu, kde je možné zobrazit' ROC a DET krivky atď.

## 2 Popis biometriky

Dúhovka je farebná časť oka, ktorá je umiestnená medzi rohovkou a čočkou a obklopuje pupilu. Farba dúhovky je u jednotlivých ľudí odlišná a je dedičná. Dúhovka je pre každého človeka jedinečná, pretože obsahuje rôzne textúry a vzory čo činí dúhovku skvelou biometrickou vlastnosťou. Má v sebe kruhový sval zvierach, ktorý ovláda veľkosť pupily v strede oka. Zmenou veľkosti pupily riadi dúhovka množstvo svetla, ktoré vstupuje do oka a teda ho chráni pred príliš intenzívnym svetlom, alebo naopak umožňuje vidieť aj v tme. Na obrázku 1. je možné vidieť jednoduchú skladbu ľudského oka spolu s dúhovkou a pupilou.[1]



Obrázok 1: Jednoduché zloženie oka. Zdroj [1]

## 3 Popis biometrickej databáze

Pre overenie funkčnosti bola zvolená databáza Chinese Academy of Sciences Institute of Automation verzia 1 (CASIA v1), ktorá je pre vedecké a vzdelávacie účely voľne k stiahnutiu na stránke <http://biometrics.idealtest.org/>. Jednotlivé snímky databázy boli nasnímané pomocou vlastnoručne vytvorenej kamery na snímanie dúhoviek. Osem 850nm NIR osvetľovačov bolo rozmiestnených okolo snímacieho senzoru, aby bolo zaistené, že dúhovka bude adekvátne a rovnomerne osvetlená. Kvôli ochrane výrobného tajomstva kamery, boli

---

<sup>1</sup> <http://www.stud.fit.vutbr.cz/~xvanaj00/bio/>

pupily nasnímaných dúhoviek automaticky detekované a nahradené kruhom konštantnej intenzity, aby sa zamedzilo zrkadleniu obrazu a tak odhaleniu kamery a osvetľovačov.

Databáza obsahuje dokopy 756 obrázkov zo 108 očí. Pre každé oko bolo nasnímaných 7 snímok, ktoré boli získané počas 2 sedení (3 snímky z prvého sedenia, 4 snímky z druhého). Všetky snímky boli uložené vo formáte BMP pri rozlíšení 320x280 pixelov. Meno snímku je vo formáte XXX\_S\_Y.bmp kde XXX je unikátny identifikátor oka (0-108), S je poradie sedenia, na ktorom bola snímka nasnímaná a Y poradie snímky počas daného sedenia.

## 4 Popis vlastného riešenia

Pre implementáciu biometrického systému sme použili nasledovné fázy aplikované postupne:

1. Lokalizácia dúhovky
2. Extrakcia dúhovky
3. Normalizácia dúhovky
4. Získanie kódu dúhovky
5. Porovnávanie kódov

### 4.1 Lokalizácia dúhovky

Pred samotnou lokalizáciou dúhovky je na obrázok aplikovaný algoritmus *EqualizeHist*, ktorý normalizuje jas a zvýši kontrast obrázka. Následne sa na obrázok striedavo aplikujú filtre *cvThreshold* a *cvSmooth*. Filter *cvThreshold* zabezpečuje funkciu prahovania a *cvSmooth* funkciu rozostrenia obrázku. Týmto striedaním spomenutých filtrov je docielené toho, že sa odstráni takmer všetko, okrem čiernej pupily. Po invertovaní farieb získame masku, v ktorej je plocha pupily bielej farby a zvyšok obrázka je čiernej farby. V tejto maske je potom jednoduchšie detekovať plochu a polohu pupily. Pomocou funkcie *FindContours()* (zvolená na základe presnejších výsledkov, ako pri tiež skúšanej metóde, kde bola pupila detekovaná pomocou získavania Houghových kruhov metódou *HoughCircles*) získame kontúry pomocou, z ktorých neskôr získame plochu a stred pupily. Následným vytvorením kruhu a jeho zväčšením o vhodne zvolenú konštantu detekujeme dúhovku.

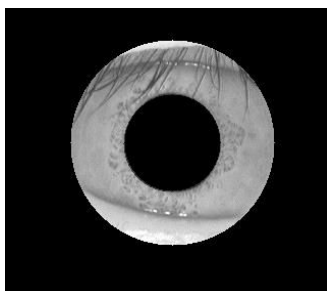


Obrázok 2:

Výsledok po striedavom aplikovaní *cvThreshold* a *cvSmooth*.

## 4.2 Extrakcia dúhovky

Pomocou kruhov, ktoré ohraničujú dúhovku a pupilu získaných v predchádzajúcom kroku vytvoríme masky, ktoré postupne aplikujeme na originálny obrázok. Jednou maskou docielime, že začerníme okolie dúhovky a druhým docielime, že začerníme pupilu, ktorá síce má čiernu farbu, ale na každej snímke má rôznu intenzitu. Po aplikovaní masiek získame teda iba detail dúhovky.



Obrázok 3: Výsledok po aplikovaní získaných masiek na originálny obrázok

## 4.3 Normalizácia dúhovky

Normalizácia je zabezpečená pomocou konverzie obrázku z kartézskeho súradnicového systému na polárny. Tento obrázok je neskôr rotovaný o 270 stupňov. Z rotovaného obrázku postupne vyberieme len riadky, ktoré v celom riadku obsahujú iba dáta dúhovky čím získame obrázok o veľkosti 280x32 pixelov. Na tento obrázok je aplikovaná metóda *EqualizeHist()*.



Obrázok 4: Normalizovaná dúhovka

## 4.4 Získanie kódu dúhovky

Za účelom získania kódu dúhovky sme na normalizovaný obrázok z kapitoly 4.3 aplikovali *SmoothGaussian* filter s hodnotami šírky kernelu = 3, výšky kernelu = 9,  $\sigma_1 = 10$ ,  $\sigma_2 = 10$ . Tieto hodnoty sme získali postupným skúšaním rôznych vstupných hodnôt pre *SmoothGaussian* metódu a vybrali sme tie, po ktorých aplikácii systém dosahoval najlepšie výsledky.



Obrázok 5: Gausov filter

Na takto získaný obrázok sme použili metódu Local Binary Pattern[4], ktorá pre každý pixel nastaví prah na hodnotu jeho farby a skúma jeho susedné pixely. Ak je hodnota pixelu v okolí menšia získame 0 a naopak ak je hodnota vyššia získame 1. Po preskúmaní okolia sme získali novú 8 bitovú hodnotu pixelu. Pomocou tejto metódy sme získali lokálny popis textúry. Na

koniec sme na obrázok aplikovali *ThresholdBinary*, čím sme odfiltrovali hodnoty nižšie ako priemerná hodnota obrázku a nastavili tieto hodnoty na 0 a vyššie hodnoty ako priemerná hodnota boli nastavené na 1. Takýto čiernobiely obrázok bol použitý ako kód dúhovky.



Obrázok 6: Kód dúhovky

Tento kód dúhovky sme zvolili po experimentovaní s rôznymi postupmi získavania kódu na základe najlepších dosiahnutých výsledkov. V iných prípadoch sme skúšali kód dúhovky generovať z normalizovaného obrázku pomocou vertikálneho rozdelenia obrázku na 16 rovnako veľkých častí (testovaná bola efektivita rôznych počtov rozdelenia). Z týchto častí bola vypočítaná priemerná hodnota. Každá časť bola potom porovnávaná s vypočítaným priemerom a ak bola jej priemerná hodnota vyššia do kódu dúhovky bola pridaná 1 inak bola pridaná 0. V ďalšom pokuse sme na normalizovaný obrázok aplikovali disktrétnu furierovu transformáciu a pre každý pixel sme zapísali 2 bity do kódu dúhovky. Tieto bity mali hodnotu podľa toho či v reálnom a imaginárnom obrázku mal pixel hodnotu väčšiu alebo menšiu ako 0.

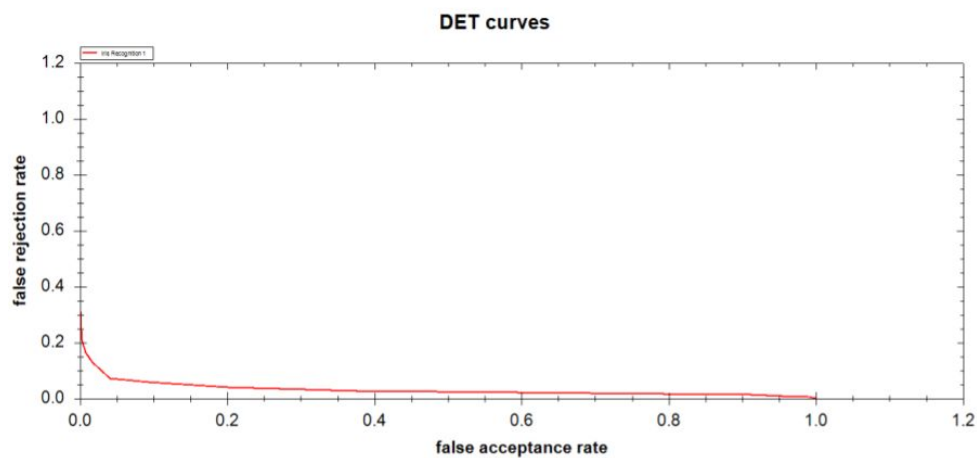
## 4.5 Porovnávanie kódov

Pre porovnávanie kódov dúhoviek sme použili výpočet hammingovej vzdialenosti[3] medzi dvoma porovnávanými kódmi a shiftovanie kódu dúhovky. Tento spôsob porovnávania sa nám osvedčil ako najúčinnější zo skúšaných postupov. Ďalej sme sa pokúsili porovnávať vektorovú similaritu medzi kódmi dúhovky pomocou rovnice 1, kde A a B sú kódy dúhoviek.[4]

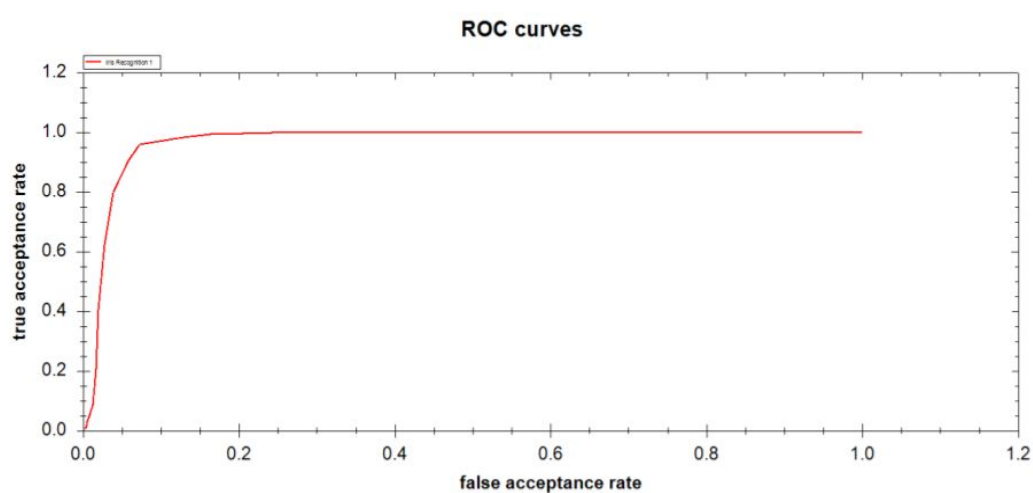
$$Similarity = \frac{\langle A, B \rangle}{(|A||B|)} \quad (1)$$

## 5 Výsledky

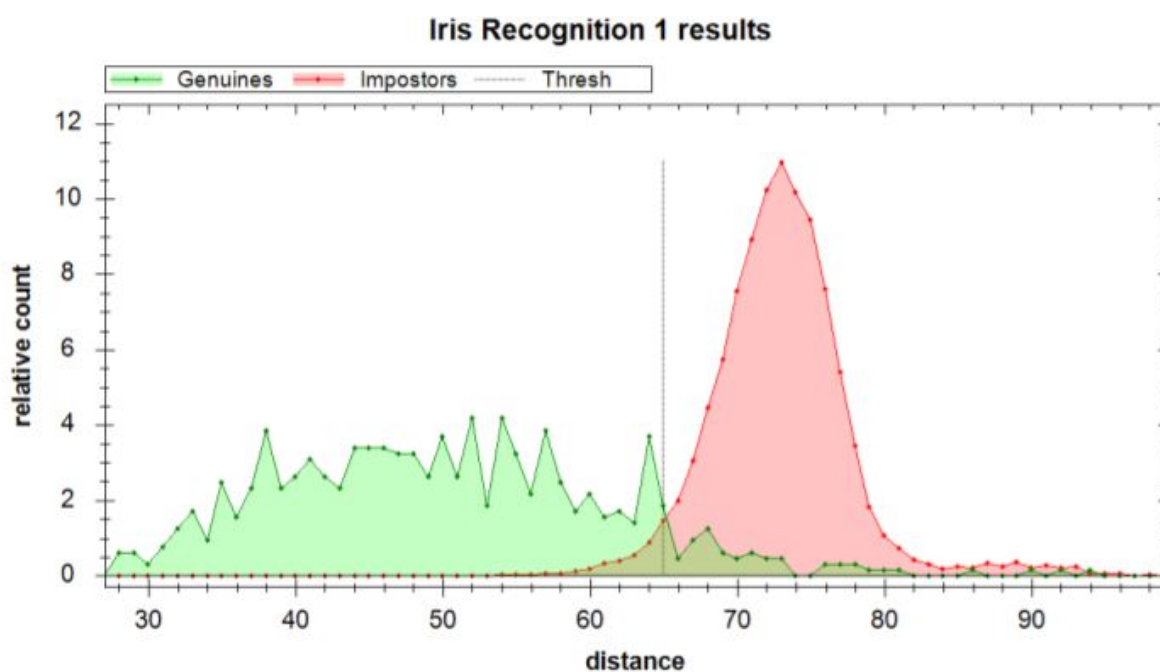
Po implementácii vyššie spomínaných fáz sme po zvolení prahu s hodnotou 27242.624 dosiahli hodnoty FAR 3.3% a FRR 8.5%. Počet True Acceptance bol 593, False Acceptance 2318, True Rejection 67018 a False Rejection 55.



Obrázok 7: DET krivka



Obrázok 8: ROC krivka



Obrázok 9: Výsledný graf vyhodnotenia algoritmu

## 6 Záver

Výsledkom tohoto projektu je jednoduchý Biometrický systém, ktorý rozpoznáva podľa dúhovky oka. Jeho funkčnosť bola otestovaná na databáze CASIAv1, ktorá obsahuje 756 snímkov očí pochádzajúcich zo 108 očí. Na tejto zvolenej databáze náš systém dosahoval hodnoty FAR 3.3% a FRR 8.5% pri zvolení najoptimálnejšieho prahu s hodnotou 27242.624. Systém bol implementovaný v jazyku C# pomocou BIOframeworku.

## 7 Spustenie projektu

Projekt je nutné nakopírovať do priečinku BIO.Framework a spustiť vo Visual Studio BIO.Framework projekt. V menu je nutné kliknúť na pridať existujúci projekt a vybrať projekt BIO.Framework.Project.IrisRecognition. Následne je nutné v súbore BIO.Framework.Project.IrisRecognition/ProjectSettings.cs nastaviť cestu k databáze obrázkov. Obrázky musia mať príponu .bmp.

## Referencie

- [1] LANEY, Heidi. *Iris Function*. In: *Med Health Daily* [online]. 2016 [cit. 2016-11-28]. Dostupné z: <http://www.medhealthdaily.com/iris-function/>
- [2] GITE, H.R. a C.N. MAHENDER. IRIS CODE GENERATION AND RECOGNITION. *International Journal of Machine Intelligence*. 2011, 3(3), 103-107. ISSN 0975–2927. Dostupné z: <http://www.bioinfo.in/contents.php?id=31>
- [3] DRAHANSKY, Martin, SEMERÁD, Lukáš. Rozpoznávání podle duhovky a sítnice [online]. [cit. 20151203]. Dostupné z: [https://www.fit.vutbr.cz/study/courses/BIO/private/07\\_BIO\\_Prednaska\\_2016.pdf](https://www.fit.vutbr.cz/study/courses/BIO/private/07_BIO_Prednaska_2016.pdf)
- [4] TIAN, Qichuan, Hua QU, Lanfang ZHANG a Ruishan ZONG. Personal Identity Recognition Approach Based on Iris Pattern. *State of the art in Biometrics*. InTech, 2011. DOI: 10.5772/17110. ISBN 978-953-307-489-4. ISSN 0975–2927. Dostupné z: <http://www.intechopen.com/books/state-of-the-art-in-biometrics/personal-identity-recognition-approach-based-on-iris-pattern>