



**INF34515-TU - PROJET EN INFORMATIQUE I**

UNIVERSITÉ DU QUÉBEC À RIMOUSKI

DÉPARTEMENT DE MATHÉMATIQUES, D'INFORMATIQUE ET DE GÉNIE

---

# **Développement d'un système biométrique pour l'authentification basé sur l'iris des yeux**

---

*Étudiant :*

François Beaulieu

*Professeur :*

Yacine Yaddaden, Ph. D.

Date : 26 mars 2022

## Table des matières

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>Système biométrique basé sur l’iris de l’œil</b>	<b>3</b>
2.1	Base de données d’iris . . . . .	3
2.2	Pré-traitement . . . . .	3
2.3	Segmentation . . . . .	3
2.4	Normalisation . . . . .	4
2.5	Extraction des caractéristiques . . . . .	4
2.5.1	GLCM . . . . .	4
2.5.2	PCA . . . . .	6
2.6	Processus de classification . . . . .	6
<b>3</b>	<b>Conclusion</b>	<b>6</b>

## 1 Introduction

[6]

## 2 Système biométrique basé sur l'iris de l'œil

### 2.1 Base de données d'iris

Le choix de la base de données d'iris est très important. La qualité des images peut influencer l'analyse des images. Il existe plusieurs bases de données disponibles sur internet.

Les éléments pouvant influencer la qualité des images sont les suivants :

- Les sources de lumière
- Lentilles
- Capteur
- Unité de contrôle

[6][3]

La base de données utilisée pour ce projet est « MMU iris dataset ». La base de données de « Multimedia University » (MMU1) est une base de données publique composée d'images d'oeil pour l'entraînement de modèles de système biométrique basé sur l'iris de l'oeil. Cet ensemble de données se compose de 5 images de l'iris gauche et droit de 46 personnes, pour un total de 460 images avec quelques fichiers vides.[1]

### 2.2 Pré-traitement

Ce processus consiste à enlever certains artefacts contenus dans les images d'origine, afin d'augmenter la précision de la reconnaissance de l'iris. Parmi les traitements pour retirer ces artefacts, on retrouve la suppression du bruit, la suppression du flou, la suppression des reflets, l'étirement du contraste, la suppression de l'occlusion des cils et la correction de l'angle du regard. Dans le pré-traitement, on doit aussi convertir les images couleurs en gris.[6][3]

### 2.3 Segmentation

La segmentation de l'iris consiste à la détection de toutes les frontières de l'iris et de la pupille. C'est l'extraction de la région d'intérêt. Les techniques « Edge detection » et « Hough Transform » sont très populaires pour la segmentation de l'iris.

Pour ce projet, j'ai utilisé la technique « Hough Transform » de « scikit-image ». « Hough

Transform » dans sa forme la plus simple est une méthode pour détecter des lignes droites, mais elle peut également être utilisée pour détecter des cercles ou des ellipses. Pour détecter un cercle, on fournit un intervalle de rayons plausibles. Pour chaque rayon, des cercles sont extraits et on garde seulement les meilleurs candidats.[5]

## 2.4 Normalisation

La normalisation consiste à transformer l'iris circulaire en rectangle pour faciliter l'extraction des caractéristiques. « Daugman's Rubber Sheet Model » est une méthode de normalisation utilisée pour la reconnaissance de l'iris.[6][3]

Pour ce projet, j'ai utilisé la méthode « Warp Polar » de « scikit-image » pour transformer l'iris détecté à l'étape précédente en rectangle. Ensuite, j'ai retiré la partie noire (la pupille) de l'image rectangulaire. Pour ce faire, j'ai retiré toutes les lignes de l'image qui contenait 50% ou plus de pixel avec des valeurs inférieures à (45, 45, 45).

## 2.5 Extraction des caractéristiques

L'extraction des caractéristiques consiste à représenter les caractéristiques de texture, afin de différencier les iris durant le processus de classification.[6][3]

### 2.5.1 GLCM

Pour l'extraction des caractéristiques de l'iris, on peut utiliser la méthode GLCM (Gray-Level Co-occurrence Matrix). GLCM est une méthode statistique de second ordre pour l'analyse de texture. Les caractéristiques qui résultent des statistiques au premier ordre renseignent sur la répartition des niveaux de gris de l'image. Cependant, ils n'incluent aucune information sur les positions relatives des différents gris au sein de l'image. Les statistiques de second ordre sont utilisées pour fournir ces informations, où les pixels sont pris en compte par paires. Les statistiques du deuxième ordre et des ordres supérieurs prédisent deux valeurs de pixel ou plus l'une de l'autre à des emplacements particuliers. Les matrices de cooccurrence de niveau de gris sont des exemples de caractéristiques de second ordre de texture statique. La matrice décrit le nombre d'occurrences d'une paire (I, J) où I est le niveau de gris d'un pixel et J est le niveau de gris d'un pixel distants ou décalés de l'autre pixel (situé à 0°, 45°, 90° ou 135° du premier). Le nombre d'occurrences sera situé à la case (I, J) de la matrice GLCM.

De nombreuses caractéristiques de texture peuvent être identifiées à partir de la matrice GLCM. En voici quelques-unes :

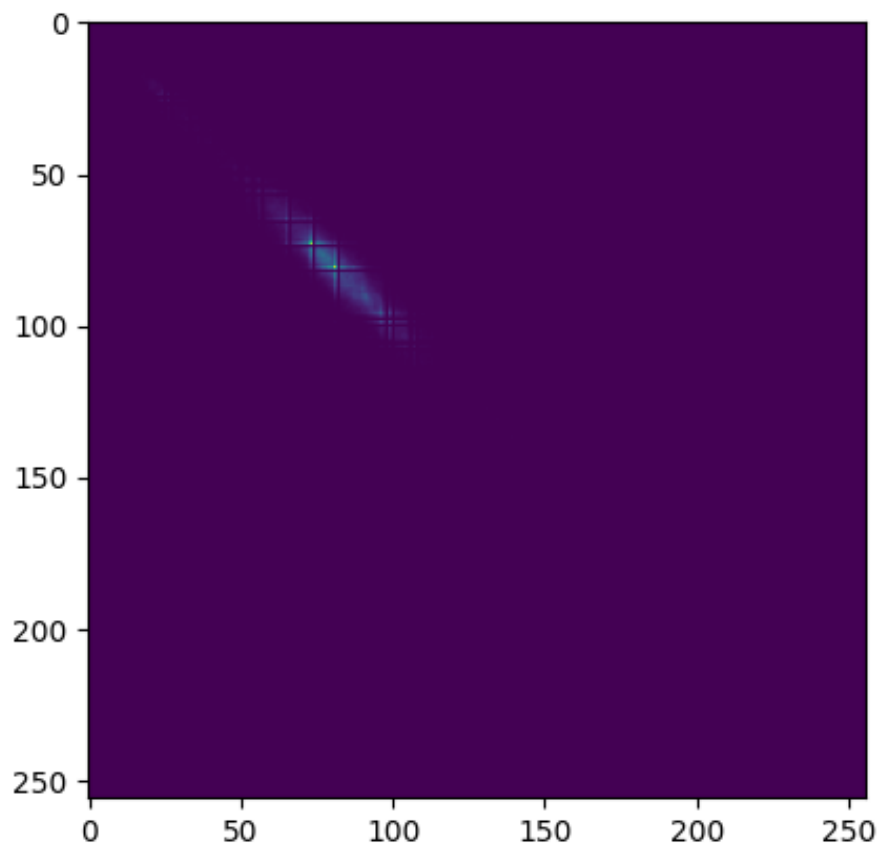
- **Contraste** : Évalue la variation locale du niveau de gris dans les valeurs d'intensité des pixels

- **Énergie** : Le nombre de paires répétées est calculé. L'énergie doit être élevée si les paires de pixels répétées sont élevées.
- **Entropie** : Décrit le degré de hasard nécessaire à la compression d'image.
- **Homogénéité** : Teste si les paires de pixels sont homogènes. L'homogénéité est supposée être élevée si les valeurs des pixels de chaque paire de pixels sont identiques.
- **Corrélation** : Fournit une connexion dans la paire de pixels entre les deux pixels. La corrélation est supposée être vitale si les niveaux de gris des paires de pixels sont fortement corrélés.

[4]

Pour ce projet, j'ai utilisé la méthode « GLCM » de « scikit-image » pour créer la matrice GLCM à partir de l'image de l'iris.

Voici un exemple de résultat de la matrice GLCM de l'iris d'un oeil :



### 2.5.2 PCA

La réduction de la dimensionnalité consiste à réduire considérablement le nombre de caractéristiques pour faciliter le processus d'entraînement. L'analyse en composantes principales (PCA) est de loin l'algorithme de réduction de dimensionnalité le plus populaire. Cet algorithme identifie d'abord l'hyperplan le plus proche des données, puis projette les données sur celui-ci. Avant de pouvoir projeter les données sur l'hyperplan de dimension inférieure, on doit choisir le bon hyperplan. On doit donc choisir le bon axe avant de faire la projection des données. Dépendamment de l'axe choisi sur l'hyperplan, on peut conserver le plus possible la variance ou seulement la conserver faiblement. En conservant le plus la variance, on perdra le moins d'informations possible. L'objectif du PCA sera donc de conserver le plus de variance pour garder le plus d'informations possible. Le PCA identifiera l'axe qui représente la plus grande quantité de variance dans l'ensemble d'apprentissage. Ensuite, il trouvera l'axe qui représente la plus grande quantité de variance restante et ainsi de suite. Il trouvera autant d'axes que le nombre de dimensions de l'ensemble de données. Donc, pour deux dimensions, il identifiera 2 axes. [2]

Pour ce projet, j'ai transformé chaque matrice 2D GLCM de chaque iris en un vecteur 1D grâce à la méthode « flatten » de « numpy ». Par la suite, j'ai ajouté tous les vecteurs 1D à un tableau. Ensuite, j'ai appliqué la méthode « PCA » de « scikit-learn » sur le tableau de vecteurs.

## 2.6 **Processus de classification**

Le processus de classification est la dernière étape du système de reconnaissance. Il consiste à trouver le degré de similarité entre une image test et une image dans la base de données. Il permet de détecter si la personne est dans la base de données ou si c'est un imposteur.[6][3]

Il existe de nombreuses techniques pour classifier les données. Une de ces techniques est la validation croisée. La validation croisée consiste à séparer les données en, par exemple, 10 sous-groupes aléatoires pour l'entraînement. Ensuite, on entraîne et évalue les données 10 fois en choisissant 9 des sous-groupes pour l'entraînement et le dernier pour l'évaluation.[2] Il y a aussi la méthode des k voisins les plus proches (k-NN). Elle consiste à trouver les k échantillons dont l'entrée est la plus proche d'une nouvelle entrée.

## 3 **Conclusion**

## Références

- [1] 193070017. Mmu iris dataset. <https://www.kaggle.com/naureenmohammad/mmu-iris-dataset>, Jul 2020. pages 3
- [2] Aurélien Géron. *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow : Concepts, tools, and techniques to build intelligent systems.* " O'Reilly Media, Inc.", 2019. pages 6
- [3] Alfi Zuhriya Khoirunnisaa, Lutfi Hakim, and Adhi Dharma Wibawa. The biometrics system based on iris image processing : A review. In *2019 2nd International Conference of Computer and Informatics Engineering (IC2IE)*, pages 164–169. IEEE, 2019. pages 3, 4, 6
- [4] M Taha and H Ahmed. Second-order statistical methods glcm for authentication systems. *Iraqi Journal for Electrical and Electronic Engineering*, 17(1) :1–6, 2021. pages 5
- [5] Stéfan van der Walt, Johannes L. Schönberger, Juan Nunez-Iglesias, François Boulogne, Joshua D. Warner, Neil Yager, Emmanuelle Gouillart, Tony Yu, and the scikit-image contributors. scikit-image : image processing in Python. *PeerJ*, 2 :e453, 6 2014. pages 4
- [6] J Jenkin Winston and D Jude Hemanth. A comprehensive review on iris image-based biometric system. *Soft Computing*, 23(19) :9361–9384, 2019. pages 3, 4, 6