### **IIC3724 - Reconocimiento de Patrones**

### **Proyecto: Reconocimiento de iris**

### **Primer semestre 2023**

### **Grupo 27**

### Tomás Couso, Sol Covacich, Fernanda Guzmán, Benjamín Lillo, Agustín Urrutia

### 

### **Introducción**

La biometría consiste en técnicas de medición de características físicas para la identificación de un individuo. En la actualidad interactuamos con sistemas biométricos a diario, tales como el reconocimiento facial, la identificación de voz, huellas dactilares y el método explorado en este proyecto, el reconocimiento de iris. Estos identificadores son de gran utilidad en sistemas de seguridad o autenticación gracias a su eficiencia y efectividad. En particular, el iris humano se destaca entre otros descriptores biométricos porque tiene un patrón aleatorio y lleno de información, está compuesto por ligamentos arqueados, surcos, crestas, criptas, pecas, entre otros. Además, exceptuando los cambios en el color de los ojos, los patrones del iris son estables a lo largo del tiempo (Rathgeb y Busch, 2017).

El proyecto documentado en este informe tiene como objetivo desarrollar y comparar el desempeño de 3 modelos de reconocimiento de iris. Para ello se experimentó con distintos descriptores, algoritmos de selección o transformación y clasificadores, además de diferentes métodos para la medición de desempeño. En las siguientes secciones se detalla el proceso de obtención y procesamiento de las imágenes, la extracción, selección y transformación de características utilizadas y finalmente los modelos de clasificación junto a su desempeño.

### **Estado del arte**

La obtención de la base de datos corresponde a la primera etapa en el reconocimiento de iris. Estas bases de datos deben cumplir con varios requisitos importantes: contar con una amplia cantidad de muestras intraclase, tener un tamaño lo suficientemente grande para respaldar el accuracy obtenido, ser específica, proporcionar información neta, abarcar un largo período en la captura de las imágenes y, por último, no haber sido editada. Entre las bases de datos públicas se encuentran Bath, que ofrece imágenes de alta resolución (1280 x 960 px.) en JPEG2000. También se encuentra la base de datos del Institute of Automation, Chinese Academy of Sciences (CASIA), que tiene muestras ideales en biometría y posee 4 versiones con distintas configuraciones en cada versión. El listado de base de datos sigue con MBGC, ND-CrossSensor, ND- Iris-0405, UBIRIS, UPOL, ICE, WVU, entre otros (Rathgeb et al., 2013). Las imágenes de estas bases de datos públicas se pueden encontrar en longitudes de ondas cercanas al infrarrojo (NIR), en el espectro visible, con alta resolución, sintéticas o capturadas a distancia y en condiciones no restringidas (Rathgeb y Busch, 2017).

A partir de la base de datos, se realiza el preprocesamiento de las imágenes. Los pasos genéricos del preprocesamiento incluyen la segmentación del iris, la normalización de textura mediante la transformación a un sistema de coordenadas doblemente adimensionales y no concéntricas, y la aplicación de una máscara para eliminar el ruido (Rathgeb y Busch, 2017).

Se pueden encontrar muchos métodos de segmentación de iris, cuyo objetivo es encontrar los límites pupilar y límbico. Existen métodos de entrenar una red neuronal que clasifica los pixeles como pertenecientes o no al iris, obteniendo mejores resultados que los métodos que asumen límites circulares. También existe la segmentación mediante elipses para los límites. Por otro lado, se utilizaron contornos activos geométricos, además de eliminar pestañas por medio de operadores de apertura (Bower y Burge, 2016).

Las complicaciones principales en la segmentación es que muchos iris no son circulares, sino levemente elípticos, además de sufrir oclusiones causadas por las pestañas o párpados. También es común encontrar reflejos especulares y una falta de alineación entre el centro de la pupila y el centro del iris (Broussard e Ives, 2009).

A partir del iris segmentado, es importante realizar un análisis de su textura. Por lo que muchas publicaciones se centran en esta área. Investigadores realizaron un reconocimiento de iris a partir de log-Gabor en un procesador de señales digitales y utilizaron CASIA v1. Otro grupo de investigación se enfocó en realizar “eigen-iris” que replica el reconocimiento de “eigen-faces” en iris, mediante el uso de PCA. Por otro lado, se tienen las propuestas basadas en filtros de texturas. Investigadores proponen un vector de texturas basado en el análisis de ondas en CASIA v2. y, en otra publicación, utilizaron SIFT. También se han usado transformadas de ondas separables orientadas. Al igual que filtros log-Gabor cuaterniónicos para texturas en la base de color UBIRIS. Otros investigadores usaron las matrices de coocurrencia de niveles de gris (GLCM). Además, se ha propuesto el uso GLCM modificado que analiza tríos de píxeles, en lugar de pares. Varios investigadores han propuesto enfoques que analizan la región del iris en múltiples partes, utilizando PCA y LDA en las bases CASIA, UPOL y UBIRIS (Bower y Burge, 2016).

Muchos algoritmos propuestos entregan buenos resultados. Por ejemplo, el grupo de investigación que utilizó log-Gabor obtuvo una alta tasa de verificación de hasta el 98.62% en CASIA v1 (Fatt et al., 2009). No obstante, se encuentra que esta base de datos no es significativamente grande, tiene regiones artificiales de iris, haciéndolas circulares y de intensidad constante. Por lo que muchas publicaciones que se basan en esta base de datos tienen rendimientos muy altos (Bower y Burge, 2016). A partir de esto se vuelve a destacar la importancia de la base de datos. Muchas propuestas pueden lograr un accuracy alto, pero se debe analizar el tipo de imágenes que utilizaron y cuestionar si factores externos a sus modelos permiten lograr los resultados obtenidos.

En la extracción de características se puede destacar que los reflejos del iris pueden variar los canales de color. También existen otras fuentes de información, como la región peri-ocular que mejora significativamente los resultados del reconocimiento (Bower y Burge, 2016). De hecho, según Miller et al. (2010) la zona periocular es una de las regiones más discriminativas y tiene un alto potencial como rasgo biométrico, y en su propuesta utilizaron LBP para el reconocimiento de la textura de la piel en esta zona.

### **Métodos propuestos**

Obtención y procesamiento de imágenes

Para todos los modelos, se trabajó con imágenes a color de 800 x 800 píxeles centradas del ojo izquierdo de 90 personas, donde el diámetro del iris ocupó 570 píxeles aproximadamente. El set de datos de entrenamiento consistió en 12 imágenes por persona, con un total de 1080 imágenes. El desempeño final de los modelos se midió con un set de testing de 4 imágenes por persona, con un total de 360 imágenes. Para aislar el iris del resto del ojo, se normalizaron todas las imágenes a través de una transformación de coordenadas polares a rectangulares y luego se hizo un recorte del área fuera del iris (esclerótica y párpado), bajo la suposición de imágenes centradas con igual tamaño de iris.



**Fig. 1:** Procesamiento de la imagen previo a la extracción de características

**3.1 Modelo 1**

Extracción de características: se extrajo LBP con 4x4 particiones cuadradas de cada imagen. Esto se aplicó a cada canal RGB por separado y se concatenaron los descriptores, obteniendo así 2831 características extraídas.

Limpieza y normalización: se aplicó el algoritmo de clean para deshacerse de características redundantes o constantes. Se obtuvieron 2813 feats. No se aplicaron otros algoritmos de selección o transformación. Finalmente, se normalizaron las características mediante un escalamiento min-max.

Clasificación: el clasificador utilizado fue un modelo de Logistic Regression.

Evaluación: se utilizó Shuffle Split Cross Validation con 4 folds, cuyos parámetros óptimos (C=2.1 y solver=‘newton-cg’) fueron encontrados mediante Grid Search.

**3.2 Modelo 2**

Extracción de características: se extrajo LBP con 4x4 particiones para la imagen en escala de grises y se extrajo DAISY. Este último es propuesto por Tola et al (1) como un descriptor inspirado en SIFT, localiza *keypoints*, pero es capaz de ser computado de manera más eficiente, mediante el reemplazo de las sumas ponderadas de las normas de los gradientes por convoluciones de los gradientes en direcciones específicas con varios filtros Gaussianos. Este paso permite hacer los histogramas solo una vez por región, y re-utilizarlos para los píxeles vecinos. Se concatenaron los descriptores para obtener 1184 características en total.

Limpieza y normalización: Se utilizó el algoritmo clean, pero este no descartó features (se mantuvieron las 1184). Se aplicó una normalización previa a la clasificación.

Clasificación:el clasificador utilizado fue un modelo de Support Vector Machine, en específico uno de orden C (SVC).

Evaluación: se utilizó Cross Validation con 4 folds, cuyos parámetros óptimos (C=10 y gamma=0.01) fueron encontrados mediante Grid Search.

**3.2 Modelo 3**

Extracción de características: se extrajo LBP con 4x4 particiones en escala de grises y se extrajo TAS (Threshold Adjacency Statistics), planteadas por Hamilton et al (3). Este proceso consiste en primero aplicar un umbral que se deshace de los píxeles más oscuros, calcular el promedio de los píxeles restantes (μ) y aplicar un nuevo umbral binario definiendo un rango cuyo centro es μ. A esta imagen binaria se le calculan 9 estadísticos: el primero es la cantidad de píxeles blancos con 0 vecinos blancos, el segundo la cantidad con 1 vecino blanco, etc., hasta llegar a 8 vecinos blancos. Estos estadísticos están diseñados para aprovechar la disimilitud en las imágenes binarizadas. Además de estos 9 estadísticos, se extrajeron 9 más sobre la imagen inversa binarizada (equivalentes a los descritos pero para pixeles negros en vez de blancos). Esto se repitió para cada canal RGB, quedando con 54 (9x2x3) features. Se concatenaron los descriptores para obtener 1106 características en total.

Limpieza: se aplicó clean y se normalizaron las features, obteniendo 1102 features.

Transformación y selección: se aplicó el algoritmo de PCA para 300 features seguido de un SBS de 200 features.

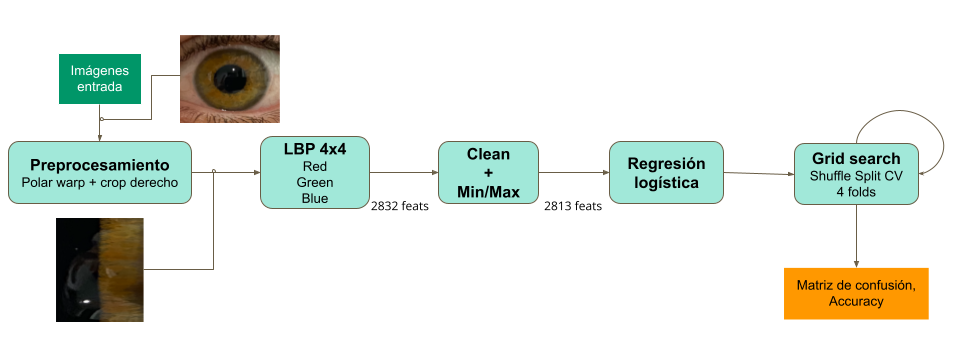
Clasificación: El clasificador utilizado fue una red neuronal (MLPClassifier) de 2 capas ocultas de tamaño 360 y 3600 respectivamente, con optimizador “adam”, learning rate adaptativo de 0.001, y función de activación identidad.

Evaluación: se utilizó una estrategia hold-out con 8 fotos por persona para el entrenamiento, y 4 fotos para el conjunto de validación.

La **Tabla 1** resume el procedimiento seguido para definir los 3 modelos descritos anteriormente, junto a la cantidad de características (features) utilizadas para clasificar.

|  | Descriptores | Clean | Selección y transformación | Clasificador | Parámetros |
| --- | --- | --- | --- | --- | --- |
| **Modelo 1** | LBP 4x4 RGB (2832 feats) | 2813 feats | - | Logistic Regression | C = 2.1 solver = newton-cg |
| **Modelo 2** | LBP 4x4 + DAISY, grayscale (1184 feats) | 1184 feats | - | SVM | C=1 gamma=0.1 kernel=linear |
| **Modelo 3** | LBP 4x4 + TAS RGB  (1106 feats) | 1102 feats | PCA + SBS (200 feats) | 2 Layer NN | learning rate\_init = 0.001  learning\_rate=adaptative  layer\_sizes = (360, 3600)  solver = adam  activation=identity |

**Tabla 1**: Resumen de los 3 modelos propuestos



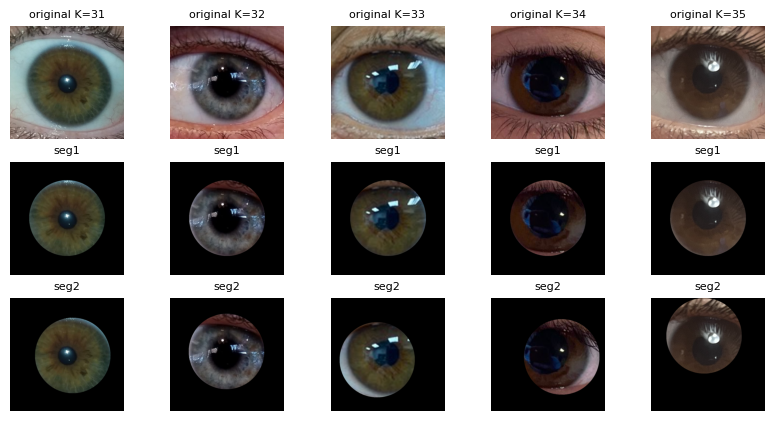
**Figura 2:** Diagrama del modelo 1, que obtuvo el mejor desempeño en el testing final.

### **Experimentos y resultados**

Procesamiento y segmentación

Para la segmentación del iris se probó la detección de círculos usando Hough Transform (Hough Circles en OpenCV), pero se descartó el uso de este método por el más sencillo de un crop igual para todas las imágenes. La **Figura 3** muestra ejemplos de las imágenes resultantes utilizando ambos métodos.

Finalmente, se tomó la decisión de hacer la normalización (transformación a representación rectangular del iris y la pupila) y luego el crop, ya que este era constante para todas las imágenes y no requería detección de círculos.



**Figura 3:** Métodos de segmentación probados. La primera fila muestra imágenes originales de 800 x 800 px, la segunda (seg1) es un crop de un círculo centrado de diámetro 570, la tercera (seg2) utiliza la detección de Hough Circles para definir el área cortada (varía según imagen).

Extracción

Inicialmente, se extrajeron múltiples descriptores que no fueron utilizados en los modelos finales. Además de los que ya se mencionaron (LBP, DAISY y TAS), se extrajo también: descriptor de Gabor; descriptores para la GLCM de Haralick de distancia (específicamente *contraste, disimilitud, homogeneidad, energía y correlación*)*;* y Dominant Colors, planteado por Yang et al (5). Dominant Color utiliza Kmeans para obtener K clusters de colores representativos (en RGB) y sus respectivos porcentajes en la imagen. Dichos colores y porcentajes posteriormente se concatenan y normalizan para obtener el vector de features.

Como primer acercamiento a la efectividad de cada descriptor, cada uno por separado fue utilizado en un clasificador KNN de 1 vecino. El desempeño de cada descriptor en este primer clasificador se puede ver en la **Tabla Anexa 1.** Cada set de datos fue normalizado antes de entrar al clasificador. También se probó concatenando descriptores e ingresándolos al clasificador KNN. Los resultados se encuentran en la **Tabla Anexa 2**.

A partir de estos resultados, se tomó la decisión de seguir trabajando con 3 descriptores diferentes:

1. LBP 4x4 en RGB
2. LBP Gray + DAISY gray
3. LBP Gray + TAS en RGB

Selección y transformación

Se probaron los siguientes métodos de selección: SBS, KBEST, NMF, Truncated SVD y selección mediante Extra Trees Classifier.

KBEST de la librería Scikit learn escoge las mejores K características mediante una función de puntuación. Esta función debe variar dependiendo si el objetivo es discreto/continuo y si las características son negativas/no-negativas. En este caso se usó una función f-regression, que entrega como resultado f-value entre las características.

NMF (Non-Negative Matrix Factorization) es una función de transformación de características de Scikit learn que entrega una representación de las características basada en partes de una matriz de entrada no negativas (Quintero et al., 2020). Su algoritmo busca 2 matrices no negativas cuyo producto se aproxime a la entrada.

Truncated SVD (Truncated Singular Value Decomposition) es una función de transformación de características de Scikit learn, SVD es una selección no supervisada que descompone una matriz rectangular en valores singulares. Estos valores singulares tienen la propiedad de que los primeros k sean mayores a los valores descartados. Existe una relación directa entre lo que informa el valor singular i y la i-ésima nueva dimensión (Fallucchi y Zanzotto, 2009).

Extra Trees Classifier es una función de clasificación Scikit learn que se puede utilizar como seleccionador. Se basa en un conjunto de árboles de decisión, cuyos nodos contienen la importancia de cada característica calculada con el método Gini. Cada nodo divide los datos en otros conjuntos según una función de impureza, por lo que los valores similares se agrupan juntos y los diferentes se separan. A mayor impureza las características son más importantes (Kharwar y Thakor, 2022).

Clasificadores

Se estudiaron cuatro clasificadores sobre las selecciones de características con mejor desempeño. En base a las pruebas realizadas, se llegó a los modelos detallados en la **Tabla 2:**

|  | Descriptores | Clean | Selección y Transformación | Evaluación de Rendimiento | | Accuracy (Test G01) | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Modelo 1 | LBP 4x4 RGB (2832 feats) | 2813 feats | - | Shuffle Split Cross Validation con 4 folds | | **89.44%** | | |
| Modelo 2 | LBP 4x4 + DAISY, grayscale (1184 feats) | 1184 feats | - | Cross Validation con 4 folds | | 83.61% | | |
| Modelo 3 | LBP 4x4 + TAS RGB  (1106 feats) | 1102 feats | PCA + SBS (200 feats) | Hold-out (8 fotos entrenamiento, 4 testeo por clase) | | 70.31 | | |

**Tabla 2:** Características, esquema de evaluación de desempeño y accuracy de los modelos seleccionados.

Se incluye un Ablation study del modelo con mejor desempeño en la **Tabla 3:**

|  | Accuracy KNN-1 (Test G01) | | Accuracy Logistic Regression  (Shuffle CV 4 folds) | | | Accuracy Logistic Regression  (Test G01) | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Sin selección | PCA + SFS | Sin selección | KBEST | PCA + SFS | Sin selección | KBEST | PCA + SFS |
| # Features | 2832 | 50 | 2832 | 540 | 50 | 2832 | 540 | 50 |
| LBP RGB | **73.88%** | 73.33% | **83.22%** | 79.5% | 74.19% | **89.44%** | 83.89% | 79.72% |

**Tabla 3:** Ablation study del modelo 1 con y sin algoritmos de selección y transformación. Se observa mejor desempeño al utilizar todas las features originales.

### **Análisis y discusión**

El aspecto del que más se conversó como equipo, fue sobre el trade off entre tener un modelo robusto para predecir y eficiente al momento de ser ejecutado. Se buscó que este no dependiese de demasiadas variables, para así evitar la “maldición de la dimensionalidad”. Al momento de tomar la decisión, quien cumplía con estas propiedades fue LBP de tamaño 4x4, ya que la función de extracción era rápida y los atributos extraídos de ella permiten discriminar entre clases, sumado con que la dimensionalidad se mantenía dentro de un número razonable.

Utilizando las características extraídas con LBP en escala RGB y sin seleccionar features, se obtuvo una accuracy del 89.44% en el conjunto de test, el mayor dentro de todos los modelos seleccionados. Se probó reducir la cantidad de atributos realizando selección o transformación de ellos, pero como se vio en el apartado de experimentos y resultados, la accuracy disminuía bastante. Inicialmente, se había probado con LBP 2x2, el cual tenía un buen resultado, considerando que solo fue como un primer acercamiento, por lo que para futuros modelos, se consideraría la opción de usar LBP con más particiones, pensando en que podría mejorar los resultados y seguir siendo un modelo rápido. Para evitar alta dimensionalidad de atributos, convendría hacer selección y/o transformación en esta oportunidad.

Para los modelos dos y tres, se optó por usar LBP en escala de grises, ya que la cantidad de atributos disminuía a un tercio de lo que es la escala RGB y la accuracy disminuía poco, lo que se buscó compensar complementando el modelo con otras funciones extractoras de características. El modelo dos fue la concatenación de LBP gris junto a DAISY en escala de grises. En este modelo se pudo ver un accuracy de 83.63% en el conjunto de test. El tercer modelo consistió en la concatenación de LBP gris junto a TAS, donde el accuracy resultó ser de 70.31%. Dado que dicha selección es comparable a las otras dos en la fase de selección y transformación de características, se considera que tales resultados hacen referencia más al método utilizado que a la naturaleza de los atributos.

Resumiendo, se observa una prevalencia de la textura por sobre los colores en el desempeño de los modelos implementados, de manera consistente con la literatura (Rathgeb y Busch, 2017). En esta línea, posiblemente se observarían mejoras al trabajar con el canal gris y una mayor cantidad de particiones de la imagen, con el reparo de manejar el aumento de dimensionalidad adecuadamente mediante transformaciones y selecciones con buen desempeño en los otros modelos, como PCA o KBest.

### **Conclusiones**

Si bien los resultados obtenidos se consideran razonables, existen algunas posibles mejoras a este que se podrían realizar en una segunda iteración. Primero, el método propuesto en este informe no profundiza en el procesamiento de las imágenes antes de la extracción de características. Se realizó una segmentación del ojo basada en una característica específica de la base de datos. Para trabajos futuros sería conveniente realizar un proceso más robusto que puede incluir la disminución de ruido, eliminación parcial de los reflejos del ojo, una segmentación más precisa del iris, sin incluir la pupila, la aplicación de filtros, entre otras opciones.

A su vez, se puede mejorar el rendimiento cambiando el sensor utilizado en la adquisición de imágenes. Si se utilizan imágenes en longitudes de onda cercanas al infrarrojo (NIR), se podrían identificar mejor las texturas del iris, pero se perdería la información del color de ojos. No obstante, se pudo evidenciar que el uso de características basadas en los colores no tuvieron gran desempeño y la diferencia entre usar los canales RGB en LBP en comparación a una escala de grises es, *a priori*, de alrededor de 1%. Además, que, si se desea implementar esta propuesta a futuro, el color de ojos de una persona es variable, en cambio, su textura no.

También se puede concluir que es importante utilizar características de texturas para reconocimiento de iris, también que las características de local-regionpueden ser útiles en el modelo. En específico, LBP permite extraer mucha información de la textura del iris, convirtiéndose en la característica con más peso en los modelos, por lo que a futuro se pueden utilizar más particiones con el objetivo de mejorar el rendimiento.

La aplicación de selección y transformación de características es un paso que se puede investigar en la propuesta. A pesar de que se contó con diversas estrategias para reducir la dimensionalidad de características, no lograron ser un aporte significativo. De hecho, no se pudo aplicarlas en el mejor modelo, debido a que las características que se descartaron poseían mucha relevancia, por lo que al quitarlas se perdía un porcentaje de rendimiento significativo.

### 

### **Bibliografía**

E. Tola, V. Lepetit and P. Fua, "DAISY: An Efficient Dense Descriptor Applied to Wide-Baseline Stereo," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 32, no. 5, pp. 815-830, May 2010, doi: 10.1109/TPAMI.2009.77.

Hamilton, N.A., Pantelic, R.S., Hanson, K. et al. Fast automated cell phenotype image classification. BMC Bioinformatics 8, 110 (2007). <https://doi.org/10.1186/1471-2105-8-110>

López, J. González, J. (2005). Estado del arte: reconocimiento automático del iris humano. Scientia et Technica 3(29), 77-78. <https://dialnet.unirioja.es/descarga/articulo/4834209.pdf>

Yang, Nai-Chung & Chang, Wei-Han & Kuo, Chung-Ming & Li, Tsia-Hsing. (2008). A fast MPEG-7 dominant color extraction with new similarity measure for image retrieval. Journal of Visual Communication and Image Representation. 19. 92-105. 10.1016/j.jvcir.2007.05.003.

Bowyer, K. W., & Burge, M. J. (2016). Handbook of Iris Recognition (K. W. Bowyer & M. J. Burge, Eds.; 2nd ed. 2016.). Springer London. https://doi.org/10.1007/978-1-4471-6784-6

Rathgeb, C., Uhl, A., & Wild, P. (2013). Iris Biometrics From Segmentation to Template Security (1st ed. 2013.). Springer New York. https://doi.org/10.1007/978-1-4614-5571-4

Rathgeb, C., & Busch, C. (2017). Iris and Periocular Biometric Recognition - 1.1 Introduction. Institution of Engineering and Technology (The IET). Retrieved from: https://app.knovel.com/hotlink/pdf/id:kt011DWPO5/iris-periocular-biometric/introduction

Broussard, R. P., & Ives, R. W. (2009, March). Using artificial neural networks and feature saliency to identify iris measurements that contain the most discriminatory information for iris segmentation. In 2009 IEEE Workshop on Computational Intelligence in Biometrics: Theory, Algorithms, and Applications (pp. 46-51). IEEE.

Fatt, R. N. Y., Haur, T. Y., & Ming, M. K. (2009, April). Iris verification algorithm based on texture analysis and its implementation on DSP. In 2009 International Conference on Signal Acquisition and Processing (pp. 198-202). IEEE.

Miller, P. E., Lyle, J. R., Pundlik, S. J., & Woodard, D. L. (2010, September). Performance evaluation of local appearance based periocular recognition. In 2010 Fourth IEEE International Conference on Biometrics: Theory, Applications and Systems (BTAS) (pp. 1-6). IEEE.

Quintero, A., Hübschmann, D., Kurzawa, N., Steinhauser, S., Rentzsch, P., Krämer, S., Andresen, C., Park, J., Eils, R., Scheslener, M. & Herrmann, C. (2020). ShinyButchR: interactive NMF-based decomposition workflow of genome-scale datasets. Biology Methods and Protocols, 5(1), bpaa022.

Fallucchi, F., & Zanzotto, F. M. (2009, September). Singular value decomposition for feature selection in taxonomy learning. In Proceedings of the International Conference RANLP-2009 (pp. 82-87).

Kharwar, A. R., & Thakor, D. V. (2022). An ensemble approach for feature selection and classification in intrusion detection using extra-tree algorithm. International Journal of Information Security and Privacy (IJISP), 16(1), 1-21.

### 

### **Anexos**

| **Descriptor** | **Parámetros** | **Accuracy KNN-1** |
| --- | --- | --- |
| LBP Gray | 4x4 particiones | 61.95% |
| **LBP RGB** | **4x4 particiones  3 canales, RGB** | **62.50%** |
| Gabor Gray | 8 rotaciones  8 dilataciones | 18.06% |
| Haralick Gray | distancia 1 pixel  4 ángulos (0, 45, 90 y 135) | 11.95% |
| Dominant Colors | 3 clusters | 16.95% |
| TAS Gray |  | 32.94% |
| TAS RGB |  | 46.11% |
| DAISY Gray |  | 59.17% |
| DAISY RGB |  | 60.28% |

**Tabla Anexa 1:** Accuracy preliminar obtenida en un clasificador KNN de 1 vecino para los descriptores extraídos

| **Descriptores** | **Cantidad de feats** | **Accuracy KNN-1** |
| --- | --- | --- |
| LBP Gray | 944 | 72,22% |
| LBP Gray + Haralick | 964 | 72,5% |
| LBP Gray + Gabor | 1011 | 73.88% |
| LBP Gray + Dominant Colors | 956 | 70.55% |
| **LBP Gray + TAS RGB** | **1106** | **75.83%** |
| LBP Gray + TAS Gray | 998 | 74.72% |
| LBP Gray + DAISY RGB | 1664 | 71,94% |
| **LBP Gray + DAISY Gray** | **1184** | **73,05%** |

**Tabla Anexa 2:** Accuracy preliminar de concatenación de descriptores extraídos en un clasificador KNN de 1 vecino.