



Consignas

- El trabajo es individual.
- Seleccionar una opción.
- Será requisito para la presentación final del trabajo haber completado la entrega de todas las prácticas de la materia.
- Preparar un reporte de análisis del artículo utilizando el formato *Lecture Notes*¹, incluyendo comentarios de trabajos anteriores (estado del arte), detalle de la implementación realizada, los resultados obtenidos, conclusiones y bibliografía.
- La presentación final será de 10 minutos por expositor, exponiendo en (hasta) 15 slides el trabajo y análisis de resultados. Será requisito grabar previamente la presentación en un video de 10 minutos, la cual será cargada previo a la clase final en un repositorio compartido.

Opciones de Trabajos finales

Opción 1 **SkinTone**.

Paper: “Analysis of Manual and Automated Skin Tone Assignments for Face Recognition Applications”. (<https://arxiv.org/abs/2104.14685>). En el link² se encuentra una base de imágenes de personas, y las etiquetas asociadas a los SkinTone para cada persona están en un Jason. Las capturas de cada persona se encuentra dentro de un folder. Existen varias fotografías por persona, capturadas bajo distintos tipos de iluminación. En el archivo de las etiquetas se indican cuales capturas se hicieron con muy poca luz (“dark_files”). Basado en este artículo se pide en la implementación:

- Detectar el rostro (se pueden usar herramientas de OpenCV).
- Extraer la ROI.
- Calcular el SkinTone usando el método Individual Topology Angle (ITA) explicado en el paper.
- Hacer un análisis de la diferencia obtenida bajo distintas iluminaciones.
- Realizar también un análisis de la corrección de color que se le realizó a imágenes del dataset MORPH a partir de fijar el color gris del fondo. Proponer como sería la función de transformación del color.
- Entregar el código en un zip.

Opción 2 **Descriptores basados en HoG**.

Paper: “Face Recognition Using Co-Occurrence Histograms of Oriented Gradients”, Autores: Thanh-Toan DO, Ewa KIJAK. En él se utilizan las bases de caras YALE y ORL respectivamente. Se pide una implementación del artículo que:

¹<https://www.overleaf.com/latex/templates/springer-lecture-notes-in-computer-science/kzwwpvhwnvfj#.WuA4JS5uZpi>

²https://drive.google.com/file/d/1XTsWAJgekV_S1aNH0312BBwjrt0DeUy8/view?usp=sharing

- Detecte una cara entrenando con features HOG y COHOG.
- Muestre el efecto de modificar los parámetros de los features: por ejemplo en el cálculo de gradiente, el número de bins, la dimensión de la ventana.
- Calcule tasa de aciertos.
- Permita sacar conclusiones
- Puede utilizar herramientas de librerías, github es necesario que muestre resultados con propuestas de cambio de parámetros.
- Recursos
 1. <https://hal.archives-ouvertes.fr/hal-00766960/document>
 2. http://vision.ucsd.edu/datasets/yale_face_dataset_original/yalefaces.zip
 3. <https://www.kaggle.com/tavarez/the-orl-database-for-training-and-testing>

Opción 3 **Descriptor FREAK**

Paper: A. Alahi, R. Ortiz and P. Vandergheynst, “FREAK: Fast Retina Keypoint,” IEEE Conference on Computer Vision and Pattern Recognition, 2012, pp. 510-517, [Link](#)

En la presentación se pide:

- Presentar de forma detallada el descriptor FREAK y comparar con el LBP tradicional.
- Comparar FREAK con el descriptor MFREAK (“MREAK : Morphological Retina Keypoint Descriptor”, Himanshu Vaghela, Manan Oza, Sudhir Bagul. <https://arxiv.org/abs/1901.08213>). Analizar las similitudes, y ventajas de cada uno.
- Analizar la implementación en OpenCV del descriptor FREAK y experimentar con los parámetros.