

Práctica 2 – Extracción, descripción y similitud de características locales

Fernández Moreno, Jose Luis–Ramasco Gorria, Pedro

PREGUNTAS TAREAS OPCIONALES

El objetivo de esta práctica es el de detectar las correspondencias entre los puntos de interés de dos imágenes ligeramente diferentes. Para ello se implementan por separado 3 tareas, la primera implementa el descriptor de puntos de interés de Harris que nos devuelve las coordenadas de las esquinas detectadas. La segunda tarea es la de implementar el descriptor en donde puede ser un histograma de niveles de gris alrededor de un punto o un histograma de gradientes orientados también entorno a un punto. Por último, se implementa en la tercera tarea el código que calcula las correspondencias entre los descriptores calculados anteriormente mediante el método de la distancia mínima umbralizada.

2.1 Aplicando los descriptores de tipo 'hist' de la tarea 2 sobre la imagen camera () de Skimage (paquete skimage.data, <https://bit.ly/2ZsXfN7>), analice como cambian los valores del descriptor calculado si utiliza:

- Tamaño de vecindario con valores 8 y 16
- Número de bins con valores 16 y 32

Para la realización de esta pregunta hemos leído la imagen camera de Skimage para luego pasarla por la función `detectar_puntos_interés_harris` la cual nos devuelve las coordenadas de las esquinas de la imagen.

Posteriormente con la información extraída hacemos uso de la función implementada en la tarea dos, `descripción_puntos_interes`.

A esta la hemos llamado 4 veces, una con los valores de `vtam` igual a 8 y `nbins` igual a 16, otra con `vtam` igual a 8 y `nbins` igual a 32, otra con `vtam` igual a 16 y `nbins` igual a 16 y por último otra con `vtam` igual a 16 y `nbins` igual a 32.

A continuación, presentamos los histogramas de los descriptores para cada uno de los casos descritos:

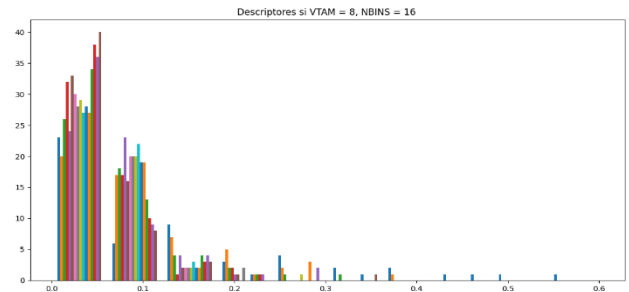


Figura 1: Descriptores si $V_{tam}=8, N_{bin}=16$

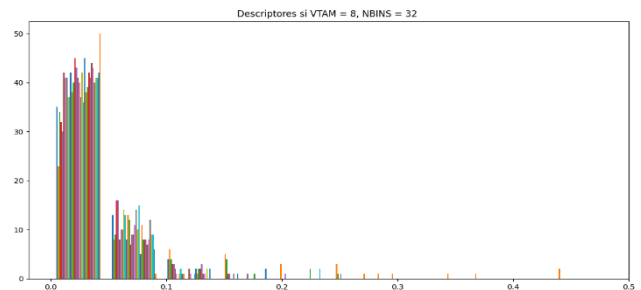


Figura 2: Descriptores si $V_{tam}=8, N_{bin}=32$

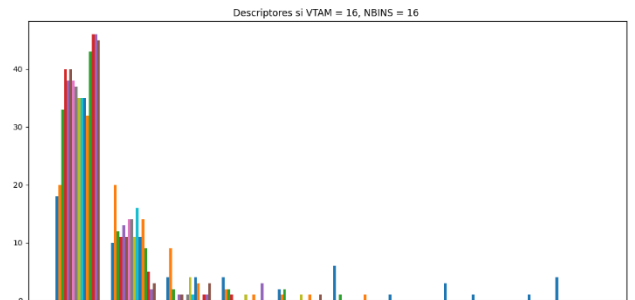


Figura 3: Descriptores si $V_{tam}=16, N_{bin}=16$

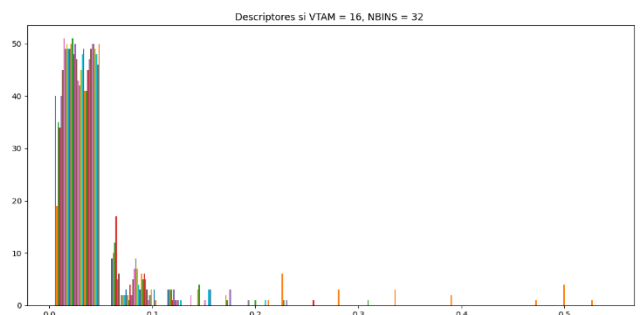


Figura 4: Descriptores si $V_{tam}=16, N_{bin}=32$

Se puede apreciar que el histograma se distribuye de manera semejante en todos los casos salvo en el de vtam igual a 16 y nbins igual a 32.

Vemos como en los tres primeros histogramas los descriptores se agrupan en diferentes intervalos mientras que en último caso la mayor parte de los descriptores se concentran cerca del 0.0.

La razón de que esto ocurra es que para el caso de vtam es 16 y nbins igual a 32 estamos usando un número de bins que es justamente el doble del tamaño de la ventana que se emplea en la tarea dos para calcular los descriptores.

A su vez, cabe mencionar, que cada uno de los descriptores calculados va a tener una dimensión de número de esquinas detectadas por número de bins empleado:

```

✓ coords_esquinas: ar
> special variables
> [0:52] : [array([4
> dtype: dtype('int
> max: 470
> min: 109
> shape: (52, 2)

```

Figura 5: coordenadas esquinas

```

✓ descriptores_vtam8_bins16:
> special variables
> [0:52] : [array([0.0987654
> dtype: dtype('float64')
max: 0.6049382716049383
min: 0.0
> shape: (52, 16)

```

Figura 6: descriptores vtam=8, bins=16

```

✓ descriptores_vtam16_bins16:
> special variables
> [0:52] : [array([0.15916955
> dtype: dtype('float64')
max: 0.657439446366782
min: 0.0
> shape: (52, 16)

```

Figura 7: descriptores vtam=16, bins=16

```

✓ descriptores_vtam8_bins32:
> special variables
> [0:52] : [array([0.024691
> dtype: dtype('float64')
max: 0.48148148148148145
min: 0.0
> shape: (52, 32)

```

Figura 8: descriptores vtam=8, bin=32

```

✓ descriptores_vtam16_bins32:
> special variables
> [0:52] : [array([0.0415224
> dtype: dtype('float64')
max: 0.5467128027681661
min: 0.0
> shape: (52, 32)

```

Figura 9: descriptores vtam=16, bins=32

De esta información se puede extraer la conclusión de que si variamos nbins el tamaño de los descriptores también lo cambia, pero siempre el número de filas se mantiene fijo ya que es el número de esquinas detectado, y este es el que es y no lo podemos cambiar (pero nbins sí ya que somos nosotros los que lo seleccionamos al pasárselo a la función que nos calcula los descriptores)

2.2 Analice las correspondencias obtenidas en la tarea 3 con los descriptores 'hist' y 'magori' sobre la primera imagen de test. Visualice y razone porque los cambios que observe en los experimentos que realice.

A continuación, vamos a presentar los resultados de la primera imagen de test y posteriormente sacaremos conclusiones de lo observado en las imágenes.

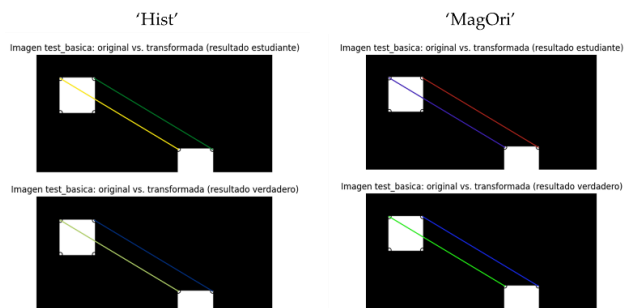


Figura 10: Imagen_test_basica_original_vs_transformada

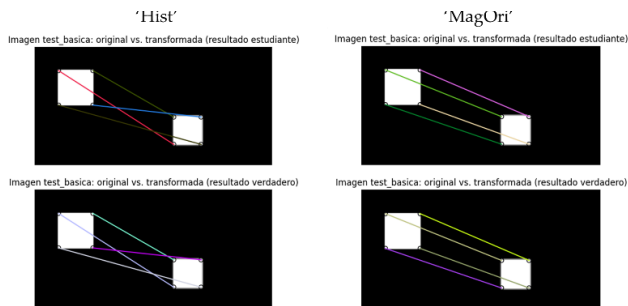


Figura 11: Imagen_test_basica_original_vs_transformada

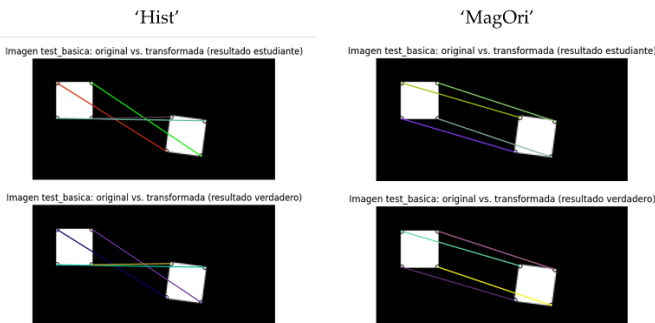


Figura 12: Imagen_test_basica_original_vs_transformada

La principal diferencia que se puede apreciar es que cuando usamos 'magori' la detección es correcta en cuanto a asociación de esquina a esquina del cuadrado se refiere. 'Hist' a diferencia de 'magori' no se basa en la distribución de orientaciones mediante el gradiente,

solo en la creación de un histograma de niveles de gris. Esta es la razón por la cual en las imágenes en las que aplicamos 'hist' vemos como las correspondencias no están ligadas a las esquinas correctas y vemos que las líneas se cortan entre sí. Sin embargo, en la primera de las imágenes los resultados entre ambos métodos son idénticos ya que en la segunda imagen solo se pueden ver las esquinas superiores del cuadrado.

2.3 Aplique las funciones de las tareas 1, 2 y 3 sobre los pares de imágenes proporcionados en la carpeta 'img'. Para cada par de imágenes, discuta las diferencias en las esquinas devueltas (si existieran) y qué fenómenos pueden explicar las diferencias observadas. Incluya ejemplos visuales de los experimentos que realice.

Para esta pregunta hemos ido aplicando las funciones que hemos implementado en las tareas y que han pasado los tests a cada uno de los pares de imágenes que se proporcionan en el guion de la práctica. Primero detectamos las esquinas con la función `detectar_puntos_interes_harris`.

Luego sacamos los descriptores con `descripción_puntos_interes` primero empleando 'hist' y después con 'magori'.

Posteriormente, calculamos las correspondencias entre los descriptores que emplean el mismo método con la función `correspondencias_puntos_interes`. Esta última nos devuelve el array que indica que coordenada de la imagen 1 se corresponde con la de la imagen 2.

El primer par de imágenes, 'Egaudi'; al calcular las esquinas vemos como en la imagen 2 se pueden detectar muchas más que en la imagen 1 (en la imagen 2 se detecta 1135 y en la imagen 1 64).

Que la imagen 2 detecte más que la 1 se debe a que EGaudi_1 es una imagen menos contrastada en cuanto a niveles de intensidad se refiere al compararla con EGaudi_2. Debido al mayor contraste vemos como se aprecian más detalles en EGaudi_2 lo que incrementa el número de puntos de interés detectados.



Figura 13: Imágenes Egaudi 1 y 2

En estas imágenes vemos como los bordes y el cielo es más homogéneo en la de la izquierda (EGaudi1) y en la imagen de la derecha (EGaudi2) los bordes destacan más y hay una mejor iluminación global en la imagen.

A su vez cabe destacar que en EGaudi1 faltan objetos que sí que aparecen en EGaudi2 (personas diferentes caminando en diferente lugar de la imagen y parte de un edificio a la izquierda de la 2), pudiendo ser otro motivo por el cual se detectan más puntos de interés en EGaudi2 que en EGaudi1.

Tras contrastar todos los procesos que hemos mencionado pasamos a realizar las imágenes de correspondencia, tanto para el modelo "hist" y el "mag_ori", los resultados son los siguientes:

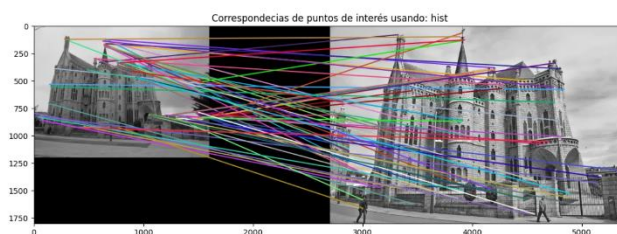


Figura 14: Correspondencia de puntos usando hist

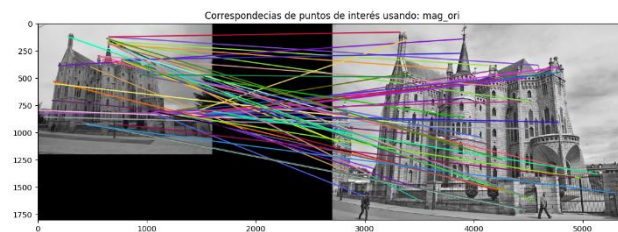


Figura 15: Correspondencia de puntos usando mag_ori

Si observamos los resultados vemos como las correspondencias no son del todo correctas.

Esto puede ser debido a que la imagen está muy contrastada y con alta iluminación, lo que provoca que el algoritmo falle y no detecte de las esquinas de la manera correcta. Como ya hemos dicho, hay ciertos objetos que no salen en ambas imágenes, lo que puede ser otro de los motivos por el que las correspondencias no sean tan correctas.

El segundo par de imágenes, 'Mount_Rushmore', pasa algo muy similar que con el primer par de imágenes. Para Mount_Rushmore1 se detectan 23 puntos de interés y para Mount_Rushmore 2 se detectan 56.

La diferencia no es tanta como con las EGaudi1, y esto puede ser porque la intensidad global de ambas Mount_Rushmore es similar y a simple vista no tan abrupta como con las EGaudi. Los bordes se perciben

mejor en Mount_Rushmore2 y esta puede ser la razón por la que se detectan más puntos de interés.

También se aprecia como en la Mount_Rushmore2 hay una ladera rocosa que predomina más que en Mount_Rushmore1, y puede ser otra razón por la que detecta más puntos de interés.



Figura 16: Imagenes Mount Rushmore

De nuevo procedemos a sacar los resultados de las correspondencias para "hist" y "mag_ori":

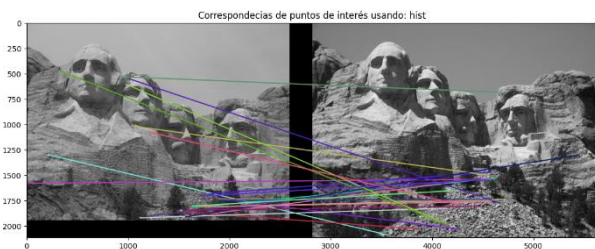


Figura 17: Correspondencia de puntos de interés usando hist



Figura 18: Correspondencia de puntos de interés usando mag_ori

Observando ambos resultados podemos ver como la mayoría de las correspondencias se centran en la zona inferior de la imagen que corresponde con la zona rocosa. Ambas imágenes son muy similares, sin embargo, el hecho de que la mayoría de las flechas se focalicen en la zona rocosa puede ser debido a que en esa zona encontramos un efecto semejante al ruido. En este caso las correspondencias siguen sin ser del todo correctas.



Figura 19: Imagen NotreDame

El tercer par de imágenes, 'Notredame', detectamos para Notredame1 72 puntos de interés y para Notredame2 109. En este caso, si observamos la iluminación global de la imagen es la misma o muy similar, entonces la diferencia en cuanto el número de puntos detectados reside sobre todo en la diferencia de objetos entre las dos imágenes.

Vemos en la imagen NotreDame2 como hay más personas, se aprecia un edificio a la derecha de la imagen, y aparece un elevador en escena. Esto es lo que puede estar haciendo que en Notredame1 se estén detectando menos puntos de interés.



Figura 20: Imagen NotreDame

Finalmente mostraremos los resultados para "hist" y "mag_ori" de este último par de imágenes:

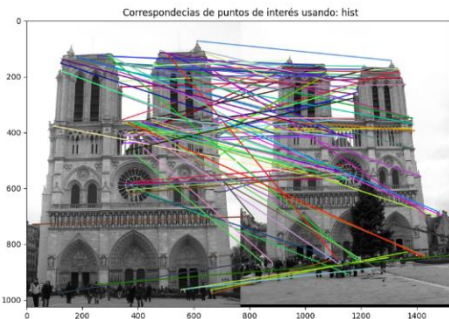


Figura 21: Correspondencia de puntos de interés usando hist

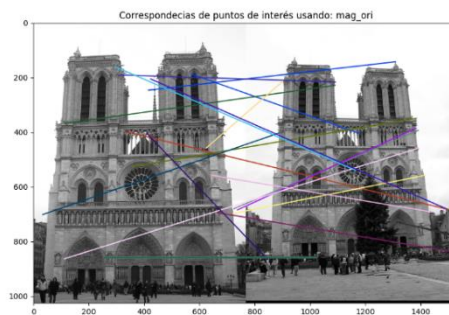


Figura 22: Correspondencia de puntos de interés usando mag_ori

Observando las correspondencias en este par de imágenes podemos observar cómo hay una mayor coincidencia. El motivo de esta coincidencia se puede deber a la gran similitud de las imágenes, únicamente se observan cambios en ciertos objetos de las imágenes (ej: personas). Además poseen una iluminación muy similar, traduciéndose en una mejor detección de correspondencias. Como último cabe destacar que en este par de imágenes el proceso "hist" funciona mucho mejor que el "mag_ori"

Como conclusión a este análisis de correspondencias podemos decir que este último par de imágenes será en el que se detecten más. En general la detección de las correspondencias no es del todo efectiva, por los motivos que ya hemos mencionado en cada par de imágenes

Nota: hemos variado los parámetros de la función \rightarrow `detectar_puntos_interes_harris` y el tamaño de las ventanas. Todo esto con la intención de obtener unos resultados las más robustos posibles. Esta modificación la hemos realizado debido a que vemos como le cuesta llegar a unos valores que nos proporcionen una correlación correcta. Por último, queremos comentar como ambos procesos de análisis de imágenes no cumplen con la correspondencia esperada entre ellas.

2.4 EXTIENDA LA FUNCIÓN DESARROLLADA EN LA TAREA 3 PARA CONSIDERAR LA CORRESPONDENCIA DE PUNTOS DE INTERÉS CON DISTANCIA MÍNIMA UMBRALIZADA ALREDEDOR DEL PUNTO DE INTERÉS (I.E. TAREA 3) JUNTO CON EL CRITERIO NEAREST NEIGHBOR DISTANCE RATIO

En este ejercicio hemos intentado implementar la función `nndr` (Nearest Neighbor Distance Ratio) dentro de la tarea 3. Para esta implementación nos hemos basado en los pasos de las diapositivas de teoría y hemos aplicado el nuevo tipo de correspondencia al par de imágenes Notre-Dame de forma análoga a como lo hicimos en la tarea 2.3. En este caso hemos indicado en la función \rightarrow `correspondencia_puntos_interes` que el valor `tipoCorr='nndr'`

Tras ejecutar nuestro código sobre el par de imágenes obtenemos:

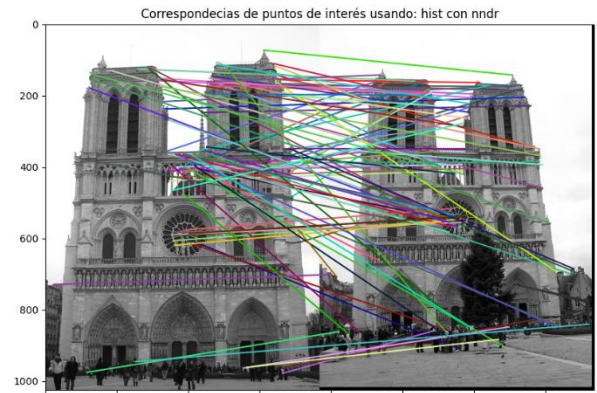


Figura 23: Correspondencia de puntos de interés usando: hist con nndr

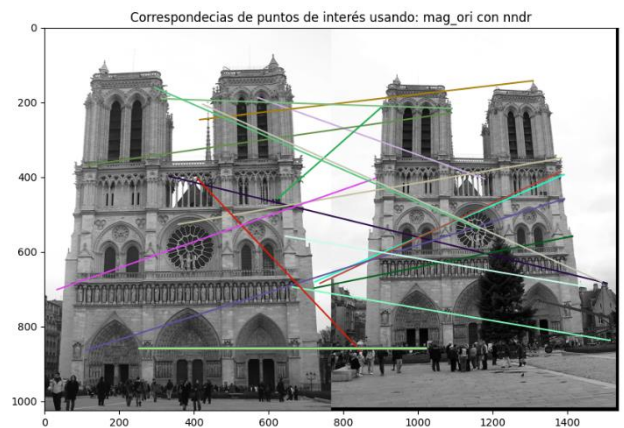


Figura 24: Correspondencia de puntos de interés usando mag_ori con nndr

Como conclusión cabe destacar que los resultados obtenidos no son los esperados, debido a que creemos que la función está mal implementada, pues no se observa prácticamente variación.

A su vez, creemos que la implementación es errónea debido a que no apreciamos grandes diferencias respecto a la forma "minDist"

En esta imagen podemos observar cómo hemos llegado a un acierto del 96.15%

4 CARGA DE TRABAJO

Indique brevemente la carga (en horas) de cada tarea de esta práctica. Puede utilizar como ejemplo la Tabla 1.

Tarea	Horas dedicadas (Jose Luis Fernandez Moreno)	Horas dedicadas (Pedro Ramasco Gorria)
Tarea 1	4 horas	4 horas
Tarea 2	4 horas	4 horas
Tarea 3	4 horas	4 horas
Tarea 4	4 horas	4 horas
Memoria	4 horas	4 horas

5 COMENTARIOS ADICIONALES

En la tarea 2 hemos estado encontrado dificultades para poder pasar el test de descriptores, debido a motivos que no hemos sido capaces de encontrar explicación en el código, pensamos que este código está bien. Creemos que estan relacionado con la precisión numérica, pero aun asi no llegamos a pasar el test al 100% y no se cumplen las especificaciones.

```

- Descriptor verdadero del test: [[25.57 18.52 1.99 4.6 6.63 3.03 13.66 12.6 30.9 18.77 5.23 4.72 5.77 5.64 8.94 14.72]]
- Descriptor estudiante: [[25.57 18.52 1.99 4.6 6.63 3.03 13.66 12.6 30.9 18.77 5.23 4.72 5.77 5.64 8.94 14.62]]

```

Figura 25: Fallo en precisión

Podemos apreciar que el unico valor que cambia es el último descriptor

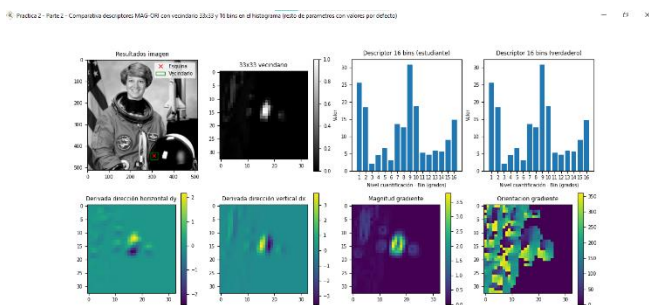


Figura 26: Comparación de descriptores

Al aplicar el test podemos ver como el histograma de descriptores el nuestro es prácticamente igual al test de la practica

```

* Finalizado en 4.431 secs
* RESULTADO FINAL: 200/208 DESCRIPTORES CORRECTOS ( 96.15% )

```

Figura 27: Porcentaje de Acierto

REFERENCIAS

Incluya en esta sección todas las referencias utilizadas para desarrollar el prototipo. Estas referencias deben seguir un formato estructurado como los ejemplos.

- [1] J.S. Bridle, "Probabilistic Interpretation of Feedforward Classification Network Outputs, with Relationships to Statistical Pattern Recognition," *Neurocomputing – Algorithms, Architectures and Applications*, F. Fogelman-Soulie and J. Herault, eds., NATO ASI Series F68, Berlin: Springer-Verlag, pp. 227-236, 1989. (Book style with paper title and editor)
- [2] W.-K. Chen, *Linear Networks and Systems*. Belmont, Calif.: Wadsworth, pp. 123-135, 1993. (Book style)
- [3] H. Poor, "A Hypertext History of Multiuser Dimensions," *MUD History*, <http://www.ccs.neu.edu/home/pb/mud-history.html>. 1986. (URL link *include year)
- [4] K. Elissa, "An Overview of Decision Theory," unpublished. (Unpublished manuscript)
- [5] 2001_TCSVT_Color_and_texture_descriptors
- [6] A Comparative Study of Image Descriptors in Recognizing Human Faces Supported by Distributed Platforms
- [7] harris-88-023
- [8] Lowe2004_ijcv04
- [9] Szeliski_Book_sec4.1
- [10] Material de Moodle