

Parcial Nº 1 - Módulo 1			
Asignatura	Tópicos Especiales - Visión Artificial	Código	0756
Profesor	José Carlos Rangel Ortiz		
Entrega	Miércoles 18 de Septiembre de 2024. Hasta las 11:59 pm	Valor	60 puntos

Indicaciones

En este documento se presentan 3 enunciados para desarrollar programas utilizando OpenCV con Python.

Debe resolver los 3 problemas en un mismo notebook el cual se debe guardar y mostrar los resultados utilizando Matplotlib.

Se deberán entregar dos archivos de manera separada como evidencia de realización de esta sección del parcial.

1. Se entregara un documento **PDF** que contendrá las imágenes generadas por cada sección del parcial y las cuales se indican en cada una de estas. Este archivo se subirá en un espacio reservado para ello en Moodle. Este documento tendrá un valor de 6 puntos en su parcial. Recuerde incluir en este archivo un encabezado corto con sus datos personales (nombre, cédula, e-mail y grupo). Nombre este archivo como *VA_Parcial_1_NombreApellido.pdf*
2. Archivo de Notebook de Jupyter (*.ipynb*). Este archivo se subirá en un espacio reservado para ello en Moodle. El mismo debe estar listo para ser ejecutado sin que se requiera de ningún tipo de cambio por parte del docente, las únicas modificaciones válidas serán las relativas a la ubicación de la imagen o vídeo a procesar. Este código tendrá un valor de 54 puntos en su parcial. Recuerde incluir en este código un bloque de comentarios con sus datos personales (nombre, cédula, e-mail y grupo). Nombre este archivo como *VA_Parcial_1_NombreApellido.ipynb*

0.1. Código para agregar texto a una imagen

En algunas secciones del parcial se pedirá que se agregue un texto utilizando una función propia de OpenCV, para esto se puede utilizar el siguiente código. De igual manera cada estudiante tiene libertad de agregar el texto usando un método distinto.

```
cv2.putText(img=canvas1,  
            text="Texto indicado",  
            org=(15,100),  
            fontFace=3,  
            fontScale=1,  
            color=(0,0,255),  
            thickness=5)
```

1. Problema 1: Filtros y Parches

En esta sección se utilizará el archivo *Problema1.jpg*. Cargue la imagen indicada en su programa. Deberá aplicar a esta imagen una serie de filtros de la siguiente manera.

1. Modifique la imagen a un tamaño de 640×480 pixeles.
2. Debe crear 3 parches que contendrán los cuadrantes $Q1$, $Q2$ y $Q4$ del esquema mostrado en la Figura 1.

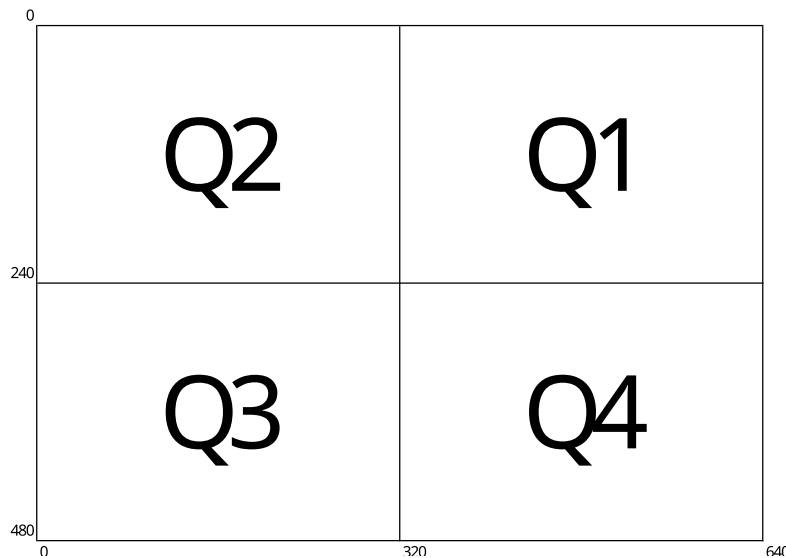


Figura 1: Esquema de Cuadrantes.

3. Usando el parche de $Q1$ aplique un filtro *Median Blur* con radio de 15.
4. Usando el parche de $Q2$ aplique un filtro *Bilateral* con diámetro de 27.
5. Usando el parche de $Q4$ aplique un filtro *Gaussiano* con radio de 29×29 .
6. Cree una copia de la imagen original y reemplace en esta imagen los parches con los filtros aplicados a su respectiva zona dentro de la imagen.
7. Agregue a esta imagen un rotulo de texto que incluya su correo electrónico en la esquina inferior izquierda de la imagen.
8. Presente una matriz utilizando `matplotlib` con 4 imágenes, según se indica a continuación:
 - Muestre la imagen original, genere y muestre el histograma de esta imagen a su lado.
 - Muestre la imagen con parches y de igual manera su histograma al lado de la imagen.

9. Agregue el código para que se almacene esta matriz en su computadora, con el nombre "**Nombre_Apellido_P1.png**", adjunte esta imagen en su PDF de evidencias de resultados.
10. Agregue el código para que se visualice el resultado en tiempo de ejecución.

Rubrica Problema 1	
Elemento	Valoración
Aplicación de Filtros	6
Parches	6
Colocación del texto	2
Presentación de Resultados	4
Total	18

2. Problema 2: Composición en Tiempo Real

En esta sección se utilizará una cámara como entrada de imágenes a la computadora, recuerde que en ausencia de la cámara puede utilizar un vídeo como fuente de imágenes.

1. Configure su código para utilizar la cámara web/vídeo como fuente de datos para este problema.
2. Modifique la imagen de entrada a un tamaño de 640×480 píxeles.
3. Tomando la imagen de entrada como base, genere 4 imágenes rotadas 45, 135, 225, 315 grados, respectivamente, de la posición de la imagen original, tome como pivote el centro de la imagen original.
4. Cada imagen rotada debe contener el ángulo de rotación escrito en la esquina superior izquierda de la imagen.
5. Utilizando estas 4 nuevas imágenes genere una composición visual donde se unan las 4 imágenes en una sola y que se despliegue en pantalla usando CV2, como se muestra en la Figura 2.
6. Despliegue sus resultados en tiempo real.
7. Configure su código para que se pueda almacenar la imagen en el disco duro, al pulsar una tecla, con el nombre "**Nombre_Apellido_P2.png**". Recuerde agregar esta imagen a su archivo PDF de evidencias del parcial.

Rúbrica Problema 2	
Elemento	Valoración
Colocación del texto	2
Rotación de Imagen	6
Composición	6
Presentación de Resultados	4
Total	18



Figura 2: Ejemplo de salida de composición de imágenes rotadas.

3. Problema 3: Histograma en Tiempo Real

En esta sección se utilizará una cámara como entrada de imágenes a la computadora, recuerde que en ausencia de la cámara puede utilizar un vídeo como fuente de imágenes.

1. Configure su código para utilizar la cámara web/vídeo como fuente de datos para este problema.
2. Modifique la imagen de entrada a un tamaño de 640×480 píxeles.
3. Calcule el histograma con *matplotlib* para cada imagen de entrada en el dispositivo de captura.
4. Despliegue en tiempo real y una sola ventana de *CV2* la imagen de entrada y el histograma de dicha imagen. Ver figura 3. Puede consultar el uso de la función `fig.canvas.draw()` de *matplotlib* para generar imágenes a partir de un plot.

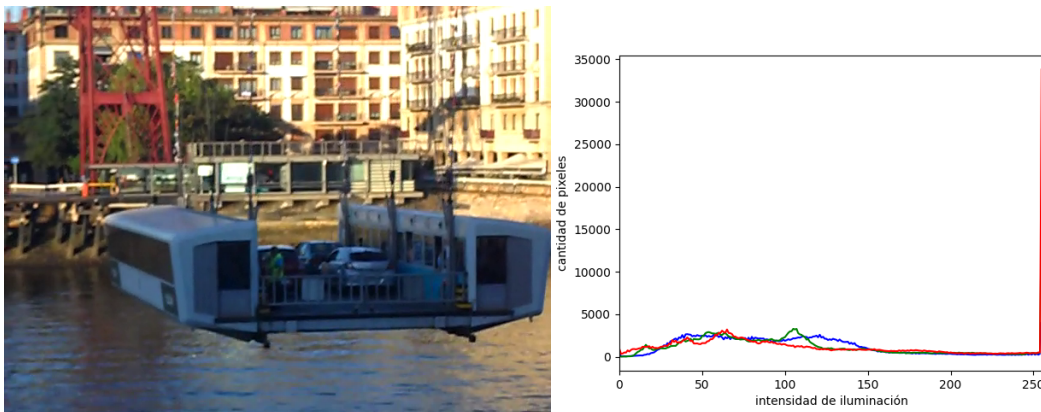


Figura 3: Ejemplo de Salida de Imagen e Histograma.

5. Despliegue sus resultados en tiempo real.
6. Configure su código para que se pueda almacenar la imagen en el disco duro, al pulsar una tecla, con el nombre "**Nombre_Apellido_P3.png**". Recuerde agregar esta imagen a su archivo PDF de evidencias del parcial.

Rúbrica Problema 3	
Elemento	Valoración
Configuración adecuada de la cámara	6
Calculo de Histograma	6
Presentación de Resultados	6
Total	18

Consejos Generales para la realización

- Ubique la imagen en una carpeta llamada **img** en la misma ubicación de su código fuente, de tal manera que la ruta al archivo sea por ejemplo la siguiente: *img/P1.jpg*.
- De igual manera almacene todos sus resultados en una carpeta llamada **out**.
- Recuerde que puede utilizar funciones (`def()`) para realizar procesos repetitivos de los códigos que se solicitan.
- Recuerde que puede consultar la documentación de OpenCV y Python para conocer sobre el uso de las diversas funciones y opciones disponibles.
- Puede ser creativo, respetando los puntos exigidos.
- Sea claro y ordenado en los documentos que se solicitan. Mantenga el orden y coherencia al programar.
- Evite el plagio entre los programas de sus compañeros y personalice su código.
- ¡Buena Suerte!