

# Evaluación Introducción a la Visión Computacional

Profesora: Daniela Opitz

Entrega: 25/11/2018

En grupos de máximo tres personas deberán resolver los dos problemas descritos abajo. Deben entregar el notebook con el código para resolver cada problema explicado en detalle.

**Cada pregunta tiene un puntaje máximo de 6.0 puntos y será evaluada con una nota entre 1.0 y 7.0.**

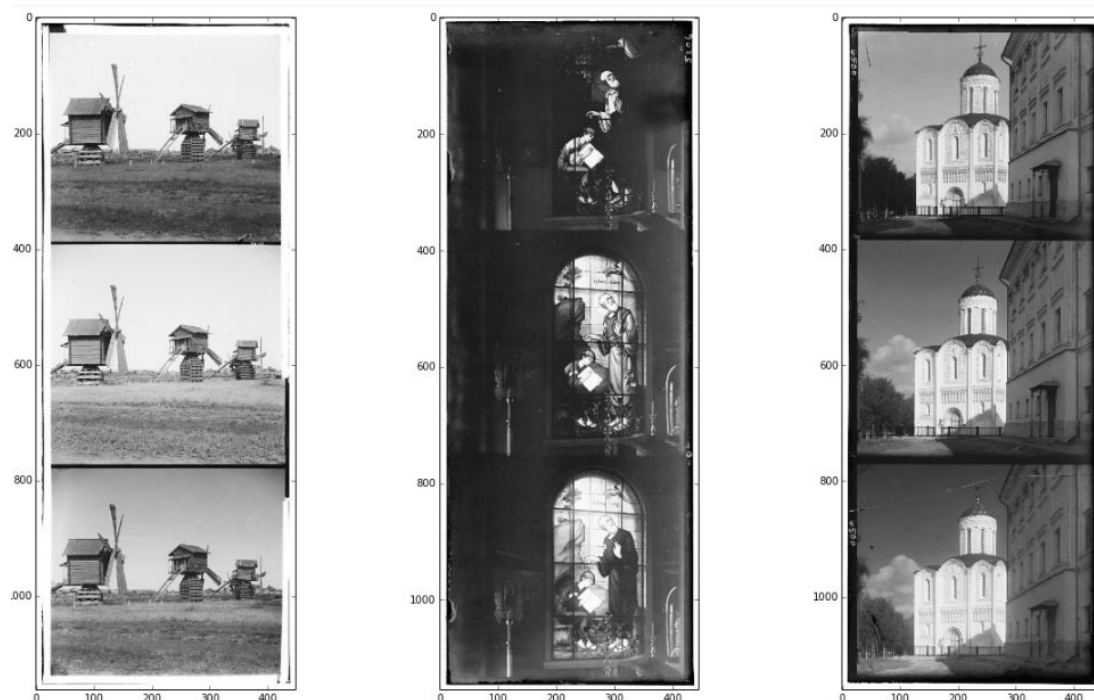
**Si el código funciona pero no está explicado en detalle se descontará hasta un máximo de 0.5 puntos por pregunta.**

**Por cada día de atraso se descontarán 0.5 pts.**

1. El archivo imagenes1.zip contiene seis imágenes con los canales R, G, B separados y desalineados. Elija tres de ellas para recortar y separar los canales R, G, B y alinearlos para obtener una imagen a color lo más nítida posible. Para ello siga los siguientes pasos:

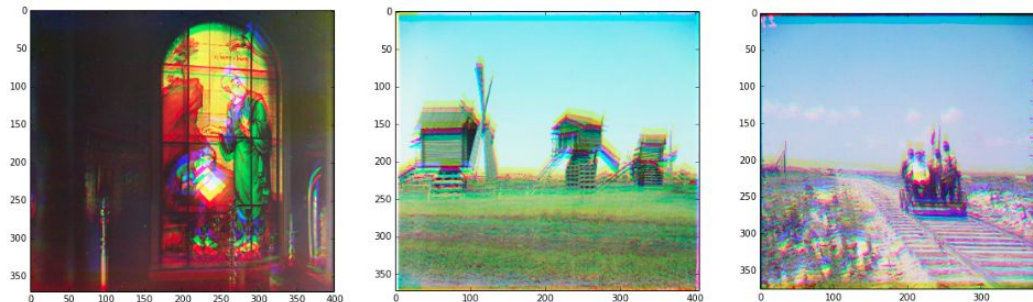
- 1.1. Cree una función que reciba una lista de imágenes y las visualice sin ningún procesamiento previo y utilícela para explorar las imágenes a recortar y alinear. Todas las imágenes deben ser desplegadas en el mismo gráfico. **Puntaje: 1.0 pt.**

Ejemplo del output:



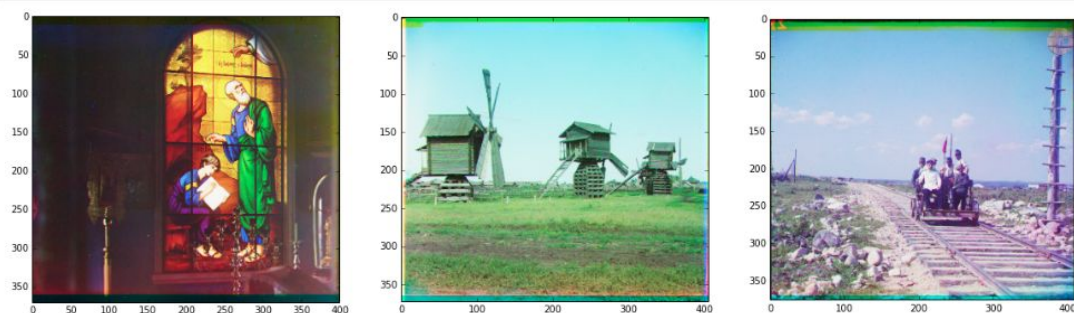
- 1.2. Cree una función que corte cada imagen en tres para obtener los canales R, G y B y los combine para obtener una imagen a color sin utilizar técnicas para alinear los canales. **Puntaje: 2.0 pts.**

Ejemplo del output:



- 1.3. Cree una función que retorne las imágenes a color alineadas lo mejor posible. Si remueve los bordes, alinear las imágenes será más sencillo. **Puntaje: 3.0 pts.**

Ejemplo del output:



**Puntaje Total Pregunta 1: 6.0 pts.**

2. El archivo imagenes2.zip contiene 129 imágenes de futbolistas famosos. Hay N futbolistas diferentes donde N es desconocido y N y menor a 129 . Esto significa que hay varias imágenes para el mismo futbolista. Clusterice el set de imágenes y asígnele un ID siguiendo los siguientes pasos.

- 2.1. Use la librería **face\_recognition**, detecte las caras en cada imagen y genere un vector de características para cada cara. Entregue un archivo que describa el nombre de la imagen, la localización de la cara y el vector de características. **Puntaje: 1.0 pt.**

- 2.2. Utilice los vectores de características para agrupar las caras de los futbolistas utilizando el algoritmo DBSCAN. Cada grupo corresponderá a un futbolista . Si usted conoce a los futbolistas pueden reemplazar los típicos IDS generados (0, 1, 2, 3 ...) por sus nombres. **Puntaje: 1.0 pt.**
- 2.3. Indique cuantos futbolistas distintos (numero de clústeres) hay en el data set. **Puntaje: 0.5 pt.**
- 2.4. Genere una forma de visualizar los clústeres. Para esto extraiga cada cara de cada imagen y despléguela en grupo. Despliegue hasta un máximo de 5 caras por grupo, ignore el resto. Cambie el tamaño de las imágenes para visualizar inteligentemente. Recuerde que el archivo generado en 2.1 existe la información necesaria. **Puntaje: 2.5 pt.**
- 2.5. En el set de imágenes hay un futbolista que no podrá ser identificado porque no pertenecerá a ningún clúster (outlier). Despliegue su cara. **Puntaje: 1.0 pt.**
- 2.6. Usted puede generar su propio set de datos. Si en vez de utilizar el set de datos entregado en clases genera su propio set de imágenes remplazando la categoría de futbolista con políticos, actores, integrantes de su familia o fotografías de los integrantes del grupo en diferentes circunstancias recibirá puntuación extra. **Puntaje: 2.0 pts.**

**Puntaje Total Pregunta 2: 6.0 pts (8.0 pts. considerando el ejercicio adicional).**