

---

## Práctica 7:

### Morfología I

---

#### 1. Objetivos

- Comprender la Reconstrucción Morfológica binaria y en escalas de grises.
- Comprender la etiquetación de componentes conexos.
- Aplicar los conceptos de morfología matemática en un problema biomédico específico.

#### 2. Parámetros de entrega

##### Sicua

- Adjuntar un único archivo (a menos que se especifique lo contrario) con los códigos de la siguiente forma: *main\_CódigoEstudiante#1\_CódigoEstudiante#2.m*.
- Adjuntar **TODO** el código que se entregó en el ítem anterior, en formato *.txt*. Llámelo de igual manera: *CódigoEstudiante#1\_CódigoEstudiante#2.txt*. Esto con el fin de poder evaluar en Sicua automáticamente cualquier intento de copia o similitud entre los algoritmos. Cualquier intento de copia será tratado de acuerdo al reglamento de la Universidad. Aquel grupo que no incluya este ítem en su entrega tendrá una nota de 0 en el laboratorio.
- Adjuntar un único archivo con el informe en PDF con el mismo nombre del primer ítem: *CódigoEstudiante#1\_CódigoEstudiante#2.pdf*. El informe debe presentarse tipo artículo con formato CVPR.
- En el caso de que no tenga compañero, **DEBE** utilizar un cero como el código de su compañero, e.i. *main\_CódigoEstudiante\_0.m* para el archivo de matlab, *CódigoEstudiante\_0.pdf* para el informe y *CódigoEstudiante\_0.txt* para el archivo de texto.
- **NO** se permiten archivos comprimidos tales como zip, rar, 7z, tar, gz, xz, iso, cbz, etc. Aquel grupo que envíe su informe como un archivo comprimido no tendrá calificación.
- No se recibirá ningún archivo por algún medio diferente a Sicuaplus.

### 3. Algunas reglas

- La asistencia a la sección de laboratorio inscrita es obligatoria. Acorde con el Reglamento General de Estudiantes de Pregrado de la Universidad de los Andes, la inasistencia a más del 20 % de las clases resultará en la reprobación de la materia.
- Cada práctica (Informe, código, etc.) tendrá plazo de entrega hasta las 11:59 pm del domingo de la semana de la sesión. El vínculo para el envío del laboratorio dejará de estar disponible luego de esta hora. Así mismo, se les entregará la guía de la práctica el sábado anterior a la sesión. Por tanto, dispondrán de una semana completa para la realización del mismo.
- Los informes deben realizarse **únicamente** con la pareja. Esto quiere decir que aunque es válido discutir los problemas con sus compañeros, la solución y el código, deben ser de su completa autoría. Está prohibido copiar literalmente el algoritmo y/o procedimiento desarrollado por otro grupo. Si llega a obtener un código de internet, asociado al problema a resolver, este debe estar debidamente referenciado y usted debe entenderlo por completo.

### 4. Parámetros de calificación

#### Resultados

- Todos los códigos deben mantener orden y coherencia en la ejecución de comandos, es decir, cada vez que se muestre una figura, el programa debe esperar para que se presione una tecla, para así continuar con la siguiente y así sucesivamente (para esto utilice *pause*). Tenga en cuenta que si se quieren contrastar dos imágenes use *subplot* o *imshowpair*.
- Toda figura debe llevar su título y descripción en el informe.
- El código debe estar debidamente comentado.
- NUNCA utilice rutas absolutas para leer o guardar archivos. Este es el error más común en la ejecución de los códigos.
- Para generar rutas utilice *fullfile* ya que puede que corra los laboratorios usando Linux o Windows y los separadores de archivos cambian dependiendo del sistema operativo.
- Asuma que dentro de la carpeta de ejecución del código se encuentran los archivos necesarios para el laboratorio.

Ejemplo: dentro del código principal el estudiante quiere leer la imagen *im.png*.

- Forma incorrecta: `imread('C:/Estudiante/MisDocumentos/ElLab/imagenes/im.png')`.
- Forma correcta: `imread(fullfile('imagenes','im.png'))`.

## Informe

Todos los laboratorios deben realizarse en formato CVPR. Para aquellos que desean realizarlo en  $\text{\LaTeX}$ , pueden obtener una plantilla del formato en el siguiente link. Cabe resaltar que los informes del laboratorio no deben contener ninguna sección de artículo científico, esto significa que no deben incluir ninguna división como resultados, abstract o conclusiones. Por consiguiente, deben responder únicamente a las preguntas del informe. Recuerden incluir imágenes de sus resultados y documentarlas debidamente. Por último, el informe tiene una longitud máxima de 2 páginas. Se pueden incluir imágenes como anexos pero las imágenes principales deben ser parte del informe.

## Bonos

Cada grupo ganará puntos que le suben la nota por cada una de las siguientes características que se cumpla:

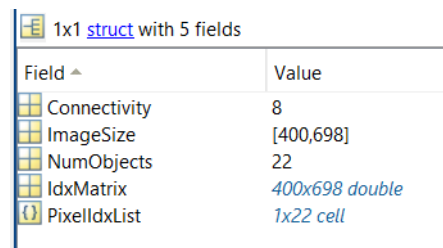
1. Parametrizar el código, es decir, no incluir cantidades fijas en el código y en caso tal de ser inevitable definir las como parámetros de entrada.
2. Desarrollar el informe en  $\text{\LaTeX}$ . (Aquellas personas que lo desarrollen en  $\text{\LaTeX}$ , deben escribir al final del informe "**Realizado en  $\text{\LaTeX}$** ". De lo contrario no se contará el bono. Los grupos que intenten reproducir la frase en un informe realizado en Word tendrán 0 en la nota de dicho laboratorio. Para poder escribir el logo utilice el comando " $\text{\LaTeX}$ " en su informe de latex.

Estos puntos se asignarán de acuerdo al criterio de las profesoras.

## 5. Informe

### 5.1. Función de componentes conexos (40 %)

Familiarícese lo que más pueda con las funciones `bwlabel()` y `bwconncomp()` de Matlab. Para esto, realice unos ejemplos usando imágenes indexadas para que entienda muy bien el funcionamiento de las funciones, así como los parámetros de entrada. Realice una función `FUNCTION [Estructura] = MyConnComp_Código#1_Código#2 (bw)` que permita contar componentes conexos de una imagen. Esta debe tener por parámetro de entrada una imagen binaria (bw) y retornar una estructura como la de `bwconncomp`. Se debe incluir dentro de la estructura una matriz con componentes indexados como la resultante de `bwlabel` (Ver Figura 1). Para la matriz con componentes indexados (`IdxMatrix`) debe establecer criterios de error con respecto a `bwlabel`. Con respecto a la lista de pixeles indexados (`PixelIdxList`) el resultado debe ser comparable con el de `bwconncomp`.



Field	Value
Connectivity	8
ImageSize	[400,698]
NumObjects	22
IdxMatrix	400x698 double
PixelIdxList	1x22 cell

Figura 1: Ejemplo de la estructura resultante de la función `MyConnComp`.

*Tip:* Para crear la lista de pixeles indexados puede hacer uso de la función `sub2ind`.

1. Realice un diagrama de flujo con los pasos que va a seguir para desarrollar la función.
2. Implemente el código en Matlab.

### 5.2. Segmentación de imágenes mediante componentes conexos (20 %)

Pruebe su función en 3 imágenes distintas una de la otra, extraiga los objetos conexos dentro de estas. Descargue sus imágenes usando el comando `websave`. Las imágenes deben tener un número diferente de objetos, es decir, no se puede tener más de una imagen en la que reconozcan 3 objetos. Por otro lado, deben reconocer al menos 2 objetos en cada imagen.

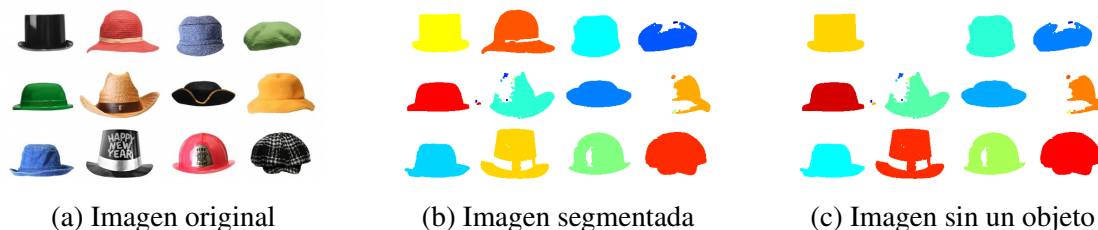


Figura 2: Ejemplo del resultado de segmentación de los objetos en la imagen original.

1. ¿Qué característica deberían tener las imágenes para que no se dificulte la tarea de extraer componentes conexos?
2. Para cada una de las 3 imágenes, muestre la imagen original (Figura 2a), la imagen segmentada como la de la Figura 2b y una tercera imagen con alguno de los elementos eliminados como en la Figura 2c (*Tip*: usar `PixelIdxList`). En el código debe aparecer el paso a paso, de la imagen original al resultado, es decir, utilice pausas y subplots.

### 5.3. Problema Biomédico (40 %)

La identificación de núcleos celulares en imágenes microscópicas es de gran importancia para extraer características morfológicas de los tejidos y realizar análisis de patologías. Usted tiene una base de datos de núcleos celulares de diferentes tejidos biológicos y ha sido contratado para desarrollar un algoritmo que cuente la cantidad de núcleos que hay en cada una de las imágenes. Las imágenes las puede encontrar en el siguiente *link*. Los datos se extrajeron de la base de datos *MoNuSeg*. *Tip*: utilice los métodos de morfología matemática aprendidos en clase para eliminar las estructuras que no desea. También puede hacer uso de técnicas aprendidas en laboratorios pasados.

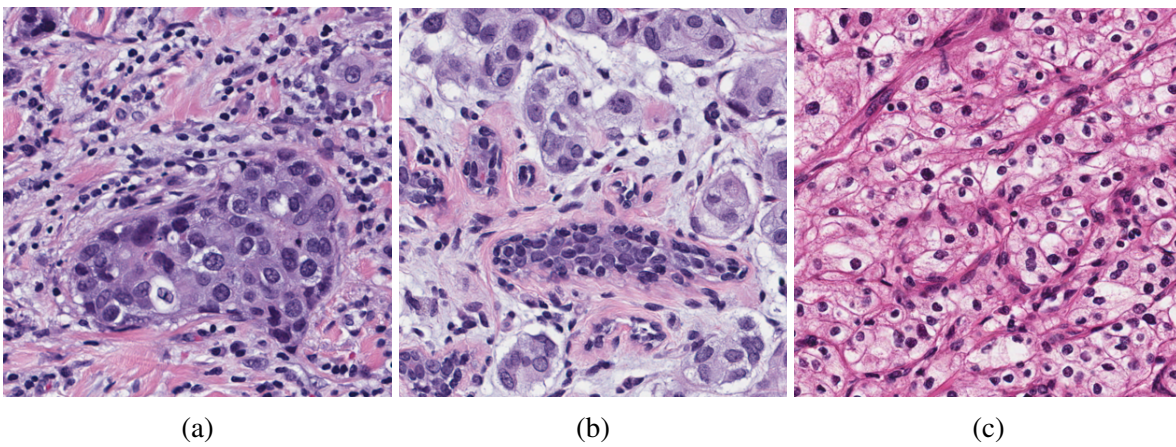


Figura 3: Ejemplo de imágenes presentes en la base de datos *MoNuSeg*.

1. Explique brevemente por qué es posible usar morfología matemática para resolver este tipo de problemas.
2. Desarrolle un algoritmo que cuente la cantidad de núcleos que hay en cada una de las imágenes. Describa su algoritmo, ¿qué métodos incorporó en el desarrollo de la solución?
3. Reporte sus resultados y el número promedio de núcleos por imagen.