

## Lab 4.

Nathan M. Marín\*

*Departamento de Física, Universidad de los Andes.*

(Dated: 20-02-2018)

### I. INTRODUCCIÓN

En este informe se practican dos métodos de formar imágenes Híbridas: uno consiste en poner en una sola imagen las bajas frecuencias de una imagen junto con las altas de otra, de forma tal que al ver de cerca se vea una imagen y de lejos se vea otra. El otro consiste en unir de forma suave por la mitad dos imágenes distintas haciendo uso de la pirámides Gaussianas y Laplacianas de las imágenes.

Para conseguir las frecuencias bajas de una imagen se aplica a esta un filtro Gaussiano, entre mayor sea la desviación estándar del filtro más suaviza la imagen, es decir, más se deshace de las frecuencias altas. Para conseguir las frecuencias altas de una imagen se resta la imagen original con la imagen suavizada con el filtro Gaussiano.

La pirámide Gaussiana de una imagen consiste de representaciones de la imagen a diferentes resoluciones, lo que se hace para generar el siguiente nivel de esta pirámide es agarrar el nivel actual y suavizarlo con un filtro Gaussiano, posteriormente se recorta su resolución a la mitad (se guarda de a un píxel de por medio).

La pirámide Laplaciana guarda la información que la Gaussiana elimina al subir de nivel en ella. Entonces, para construir el nivel  $i$  de la pirámide Laplaciana se toma el nivel  $i+1$  de la Gaussiana y se rescala a la resolución del nivel  $i$ , luego se resta la imagen del nivel  $i$  con la imagen rescalada del nivel  $i+1$ . Esta resta es el nivel  $i$  de la pirámide Laplaciana.

### II. MÉTODOS

Para hacer el código se utilizó el lenguaje de programación python, en particular se usan las siguientes librerías:

- numpy para poder trabajar las imágenes como arrays.
- matplotlib.pyplot para poder visualizar los resultados.
- Image from PIL para cargar las imágenes, rescalarlas, reflejarlas y cortarlas.
- scipy.ndimage para aplicar el filtro Gaussiano.

- cv2 para restar las imágenes, sumarlas, hacer cross-correlación con filtros propios (en particular para correr la imagen hacia la izquierda) y para crear las pirámides Gaussianas y Laplacianas.

### III. RESULTADOS

Esta sección presenta los resultados y una guía de los procesos efectuados para alcanzar el objetivo así como la lógica detrás de estos procesos. Lo que no presenta es el código que se escribió para resolver el problema pues sería poco práctico llenar esto de imágenes de código, para ver el código el notebook de python está en el mismo repositorio que este pdf.

#### A. Bajas y altas frecuencias

Las imágenes que se unen en este caso son de un lobo y de un tigre. Del lobo se tomarán las altas frecuencias y del tigre las bajas. Las imágenes originales son las siguientes:



FIG. 1. Imagen original del lobo tomada de internet.



FIG. 2. Imagen original del tigre tomada de internet.

---

\* nm.marin10@uniandes.edu.co

Lo primero es dejar las dos imágenes con la misma resolución, en este caso se decide transformarlas a 177 X 266. Luego se obtiene las bajas frecuencias del tigre y las altas del lobo. A continuación se muestran las altas frecuencias del lobo y las bajas del tigre



FIG. 3. Imagen de las altas frecuencias del lobo.

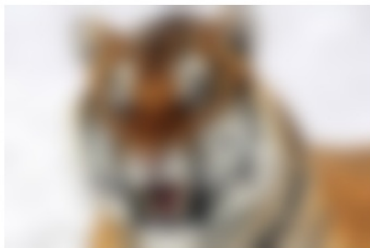


FIG. 4. Imagen de las bajas frecuencias del Tigre.

Uno podría sentirse tentado a simplemente sumar ahora las imágenes pero el resultado no queda muy bien pues el lobo y el tigre no están del todo alineados. Así, que antes de sumarlas se debe reflejar el lobo y luego correrlo un poco a la izquierda (5 píxeles para ser exacto). Una vez se han hecho estos cambios se suman las imágenes y se cortan para dejar por fuera zonas que no se ven tan bien. A continuación se muestra la imagen final.

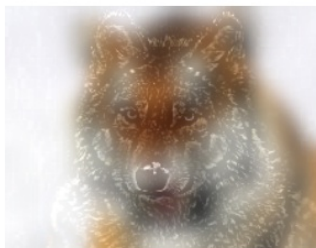


FIG. 5. Imagen final combinada del lobo y el tigre.

Para apreciar el efecto de alejarse de la imagen se hizo

una pirámide Gaussiana de la imagen, a continuación están los 3 primeros niveles.



FIG. 6. Primer nivel de la pirámide Gaussina.

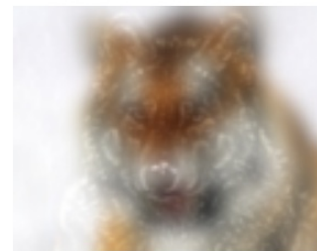


FIG. 7. Segundo nivel de la pirámide Gaussina.

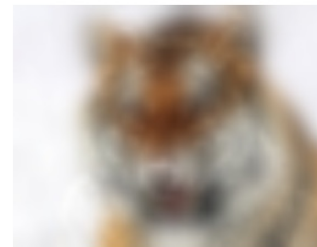


FIG. 8. Tercer nivel de la pirámide Gaussina.

## B. Blended Image

Entonces tenemos dos imágenes, una de Batman y una más. Lo primero es recortarlas para que de las dos sólo se vean las caras. Una vez están recortadas se rescalan a una resolución de 160 X 96. Se eligen estos números para no tener problemas al reducir a la mitad la resolución, pues si eventualmente la resolución llega a tener un número impar a la hora de hacer pyrUp para subir de nivel los tamaños de las imágenes no van a concordar y no se podrán sumar.

Se hacen tanto las pirámides Gaussianas como las Laplacianas de las dos imágenes. Luego en la resolución más baja de la pirámide Gaussina se unen por la mitad las dos imágenes, esta imagen unida se va subiendo de resolución con pyrUp y a cada paso se le suma el nivel de la pirámide Laplaciana correspondiente para recuperar los detalles de alta frecuencia.

---