

Albert Folch
Xavi Moreno

Morphing

Introducción

Morphing es una técnica para la transformación gradual de una imagen origen a una imagen destino. Para ello se realizan una serie de técnicas que explicaremos aquí.

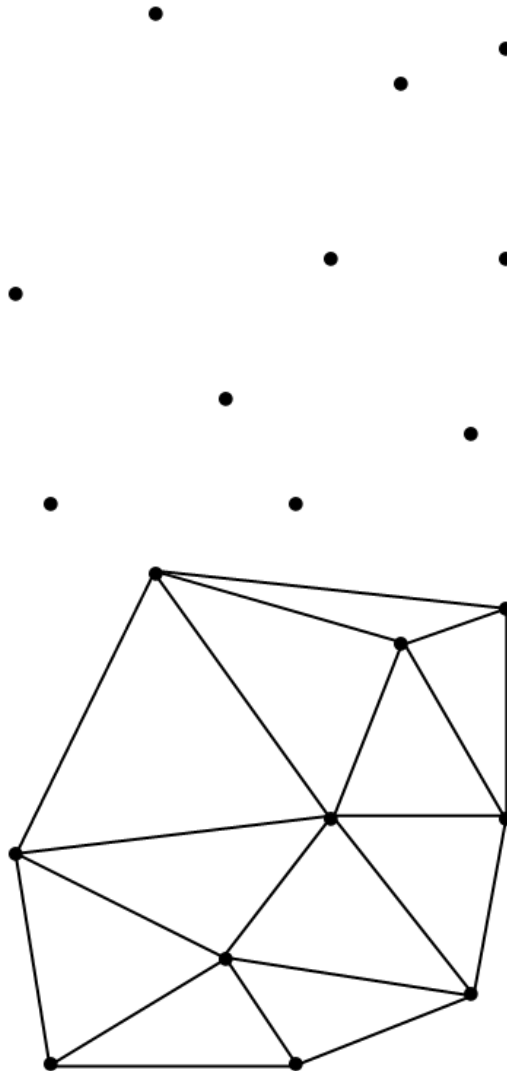


Imagen origen (izq.). Imagen destino (der.)

Técnicas

Triangulación

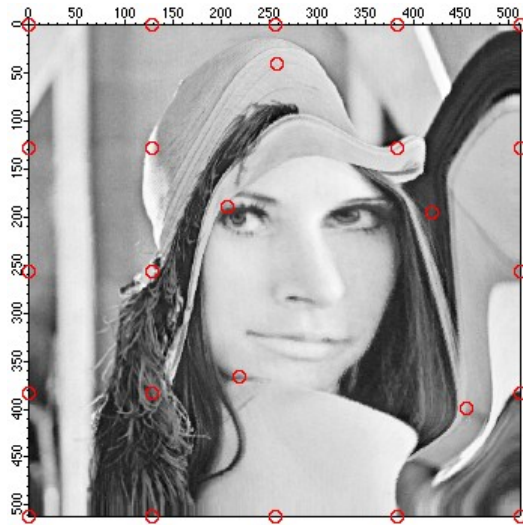
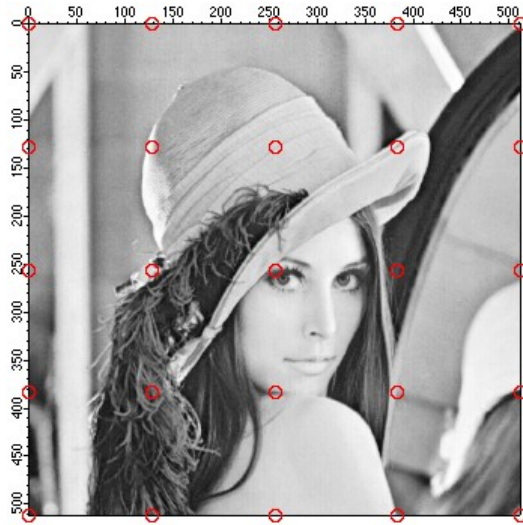
A partir de **una nube de puntos** se trata de formar **triángulos sin que no cruce ninguna línea con las demás**. Una vez realizado este proceso se forma una red (*mesh*).



Fuente: http://es.wikipedia.org/wiki/Triangulaci%C3%B3n_de_Delaunay

Warping

Una vez se tiene una triangulación de la imagen lo que se quiere es poder mover los puntos y deformar la imagen. Para ello utilizamos warping, una técnica que a partir de **una imagen**, **una mesh original** y **una mesh transformada**, nos devuelve la **imagen transformada**.



Fuente:

<http://www.wavemetrics.com/products/igorpro/imageprocessing/imagetransforms/imageinterpolation.htm>

Cross Dissolve

Esta técnica se basa en la transición gradual entre imágenes. Donde a partir de **dos imágenes** y **una variable que refleje el tiempo**, nos devuelve **una imagen resultante**. La fórmula utilizada es:

$$I = I_1 * (1 - t) + I_2 * t$$

Algoritmo

Para hacer morphing, una vez explicadas las técnicas, tenemos que seguir estos pasos:

1. Obtener los puntos característicos de cada imagen
2. Obtener los puntos intermedios de cada frame (interpolación)
3. Para cada interpolación:
 1. Transformamos (triangulación + warping) la imagen origen a la interpolación i-ésima
 2. Transformamos (triangulación + warping) la imagen destino a la interpolación i-ésima
 3. A partir de las dos imágenes alineadas, aplicamos un cross dissolve que varíe con el tiempo
 4. Mostramos resultante de cross dissolve

Este sería el pseudocódigo:

```
p1,p2 = getpoints(img1,img2)
frames = interpolation(p1,p2,numframes)
for f in frames:
    f_img1 = warp(img1, p1, f)
    f_img2 = warp(img2, p2, f)
    frame = crossdissolve(f_img1, f_img2)
    show(frame)
```

Programa

Todo el código se ha hecho con python (Ipython Notebook) y este el enlace. Todo el documento está documentado de arriba a bajo.

Dependencias:

- numpy
- matplotlib
- scipy
- skimg
- pylab
- PIL

Funcionalidad del programa:

- Limpia el notebook (Kernel > Restart)
- Ejecuta todas las celdas (Cell > Run All)
- Se abrirá una ventana con la primera imagen. Selecciona un punto
- Ahora se abrirá la segunda imagen, selecciona el punto equivalente al primero
- Estos dos procesos se repetiran hasta que pulses botón derecho sobre la imagen
- Ahora veremos los puntos seleccionados. Cerramos la ventana.
- Si en la función de morphing está puesta la opción de 'stream' se abrirá una ventana donde veremos una transición de imágenes
- Si en la función de morphing está puesta la opción de 'saveVideo' se guardaran las imágenes en una carpeta output en el directorio del Notebook

Hay que tener en cuenta que las imágenes (tanto de origen como destino) tienen que tener el **mismo tamaño** de anchura y altura.

Resultado

El resultado puede verse en la página web:

<https://sites.google.com/site/pimub2014/home/project1>