

Introducció:

En aquesta pràctica hem implementat l'algoritme Seam Carving, per reduir automàticament el mida d'una imatge o augmentar-lo sense perdre contingut ni congruència visual. La pràctica esta dividida en tres blogs principals:

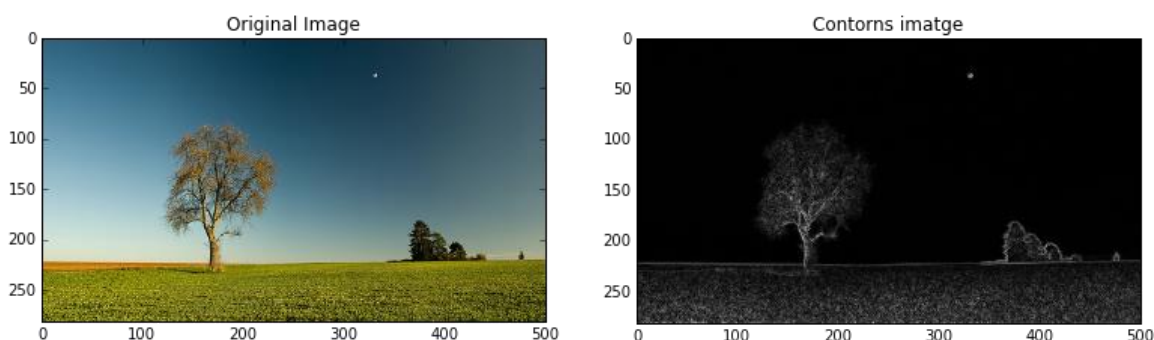
- Reducció: Donada una imatge, aplicarem l'algoritme un numero d'iteracions per tal de reduir-la extraient els pixels amb menys importància o menys pes a la imatge.
- Eliminació: Aquí després de seleccionar manualment el contorn de l'objecte a eliminar, l'algoritme eliminarà les columnes de pixels passant sempre pel contingut d'aquesta macarà, així s'aconsegueix eliminar l'objecte mantenint la coherència visual.
- Augmentar tamany: En aquesta part el que fem es fer una cerca dels seams amb valor mínim, i aquests seams copiar-los, per a que faci l'efecte de que l'imatge s'ha ampliat.

Procediment:

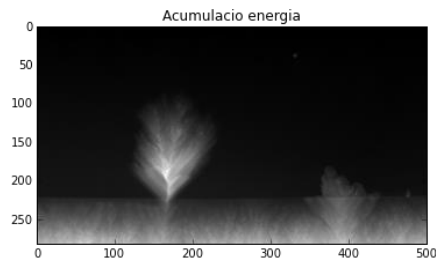
1 - Reducció:

En aquesta primera part el que volem fer com s'ha dit a la introducció es reduir la imatge traient parts no importants.

Primer ens centrem a calcular un gradient de la imatge que ens donarà els contorns de la imatge per tal de poder extreure la zona de mes pes.



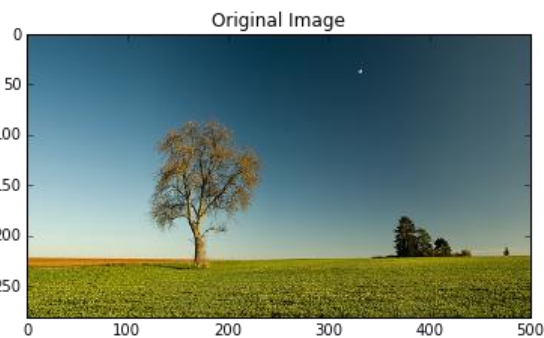
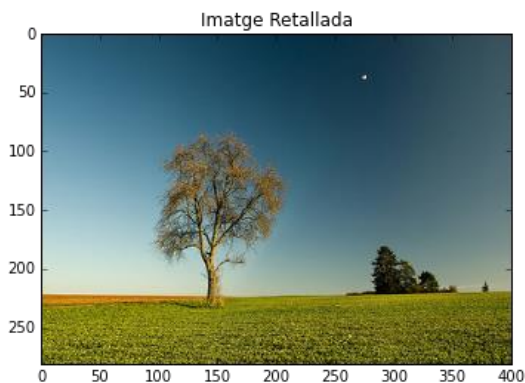
Ara calculem la matriu imatge d'energia, on tenim valors mes alts on tenim mes informació per exemple on tenim l'arbre i l'arbust, tal com es pot veure a la següent imatge. Funció *energia(imatge)*. Aquesta es fa reomplint una matriu que tenim inicialitzada amb 0 i cada pixel de la matriu es calcula com el mínim dels veïns mes un cost, que serà el valor a la matriu d'energia en aquella coordenada.



Ara ens toca identificar on tenim el valor mínim i trobar el camí amb menys pes des del píxel mínim fins a la part superior de la imatge. Això ho fem amb un backtraking on mirarem els píxels de dalt a la dreta, esquerra i centre i seguirem pel de menys pes. Amb això obtenim les coordenades (path) del camí mínim, calculades per la funció *tall_vertical(grad,cost)*. Aquestes coordenades son les que puc eliminar de la imatge.

Per borrar, creo una funció *borrar_cami (img, path)*, que te com a paràmetres d'entrada la matriu imatge i les coordenades a borrar, i vaig recorrent i copiant els píxels bons a una nova matriu que serà la que retornaré. Si el píxel esta dintre el path no el copio.

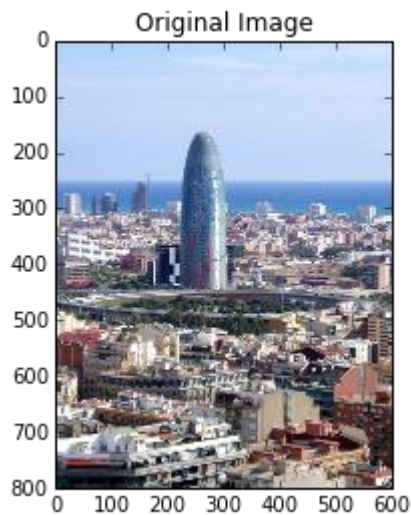
Un cop tinc les funcions necessàries aplico el Seam Carving *carving(imatge,iterations)*, on el que faig es durant les iteracions especificades vaig cridant les diferents funcions de càlcul energia, gradient, buscar camí i borrar, aconseguint així reduir la imatge.



Finalment guardem una copia de la imatge reduïda amb el nom *Reduccion_final.jpg*. Per altra vanda, aprofitant el codi facilitat pel professor generem l'animació dels camins que es van eliminant pintant aquests de color vermell. Es important per tal de poder veure aquesta animació comentar la línia de codi *%matplotlib inline*, així ens sortirà una finestra emergent amb l'animació. (Aquesta línia la deixem per defecte comentada a la pràctica.

2 - Eliminació:

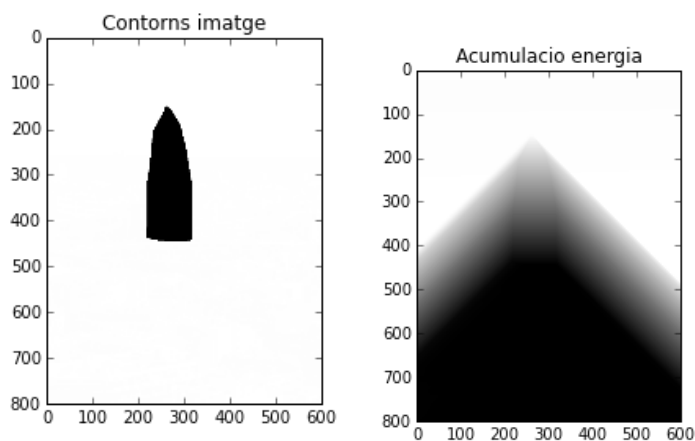
En aquesta part com s'ha comentat a la introducció la finalitat es eliminar un objecte i en aquest cas en concret el que volem eliminar es la torre Agbar de la imatge de Barcelona.



En aquesta part reaprofitem les funcions programades per la primera part i les adaptem per al nou procediment. També aprofitem el codi facilitat pel professor que ens permet seleccionar les coordenades de l'objecte a eliminar.

La funció *tall_vertical(grad,cost)*, la aprofitem tal qual ha estat implementada en la primera part. En el cas de borrar el camí afegim una modificació per tal de poder eliminar tant si tenim una imatge amb color com en blanc i negre, ja que si es amb color necessitem una

matriu amb 3ds i si es en blanc i negre no. Per altra part, en aquesta ocasió el nostre gradient el que obté es el contorn d'on tenim l'objecte amb valors molt baixos i alts a la resta de la imatge. Al calcular l'energia l'obtenim segons aquest contorn seleccionat.



Al final aconseguim eliminar l'objecte mantenint el valor de la imatge com es pot veure en la següent comparació aconseguida.



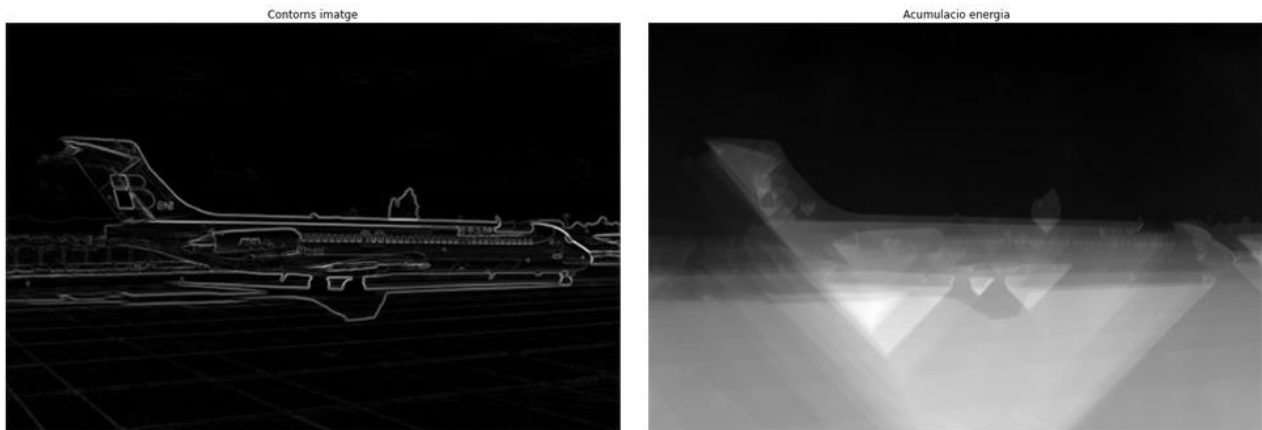
3 - Augmentar tamany

En aquesta part de la practica la finalitat es augmentar de tamany la imatge, el que augmentarem seran les parts menys rellevants de la imatge, que son les parts que si les augmenten no tenen importància visual en aquesta, com el fons de la imatge.

Comencem per agafar la imatge 'iberia.jpg', aquesta serà la imatge a augmentar, el que fem amb la imatge per a que funcioni correctament es re-escala amb un factor, en aquest cas 2, i la passem a escala de grisos per a poder treballar millor amb ella.

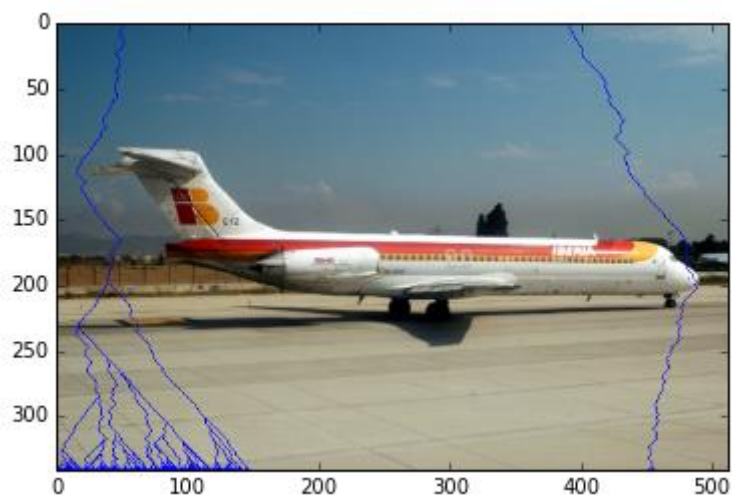


Seguidament tindrem dues funcions que les hem reciclat de les anteriors parts de la practica, una es la funció de gradient que el que farà serà fer una cerca dels contorns de la imatge i la funció energia que ens mostrarà la matriu acumulada.



Ara el que farem serà trucar a la funció carving, passant-li com a paràmetre el nombre d'iteracions que volem fer i una copia de la imatge RGB, aquesta funció s'encarregarà de cercar totes les línies verticals (seams) que tinguin el valor mínim, es a dir, les menys rellevants en la imatge, trucant a la funció `tall_vertical`. Aquesta funció ens retornarà una llista amb aquests seams menys rellevants,

segons les iteracions que nosaltres li hagem passat, aquesta funció funciona amb el algorisme de programació dinàmica backtracking, per acabar nosaltres tenim que marcar les línies per a poder observar que realment son les línies menys rellevants de la imatge, això o fem amb la funció `mark_seam` que s'encarregarà de pintar aquestes línies de color blau. Una vegada fet això trucarem a la funció `afegir_seam`, que l'únic que farà serà afegir els seams a la imatge per a augmentar-la de tamany



Una vegada hem augmentat de tamany la imatge ja podem mostrar el resultat obtingut. Es pot veure que hem augmentat les parts menys rellevants de la imatge es a dir, el fons, sense tocar l'avió.

Seams afegits



Conclusions:

Pràctica molt interessant i on anem comprovant els coneixements adquirits. Hem aconseguit obtenir els resultats demanats tot i que al final les hores dedicades acaben sent molt elevades. Al principi havíem implementat sense l'animació ja que aquesta ens donava alguns problemes però finalment s'ha aconseguit solucionar. Hem provat amb altres imatges i el resultat es igualment satisfactori, creiem que es una practica que podrà ser aprofitada per la realització del projecte final