**SESSIO 3**:

**Filtrat d'imatges:**

* J = imnoise(image, nomfiltre, mitjana, variança) -> aixo modifica el soroll de la imatge.
* Q = imfilter(I, H); -> aplica el filtre H sobre la imatge I. (per no tenir problemes la imatge I ha de ser de doubles)

Per pasar una imatge a doubles -> I = double(I)

Una imatge de doubles, per veurela be -> imshow(Q, [ ])

**Filtres de suavitzat**:

* H = fspecial(tipus de filtre, paràmetres) -> per crear filtres de suavitzat.
  + tipus de filtre:
    - *'average'* [f c] -> normalitza
    - *'disk'*-> li dones un radi y posa tot el cercle a 1 i els voltants a 0
    - *'gaussian'* [size, sigma] -> size es [f c] i el sigma es la corba diferencial.

**Filtres de contrast**:

* Laplacia: serveix per detectar canvis bruscos a la imatge (aplicant el filtre de la matriu H)

H = [-1 -1 -1; -1 8 -1; -1 -1 -1];

* Sobel: detectar contorns: (aplicant el filtre de la matriu H)

H = [-1 -2 -1; 0 0 0; 1 2 1];

Per tal de que trobi els contorns verticals i horitzontals.

GX = imfilter(I, H);

GY = imfilter(I, H’);

S = abs(GX) + abs(GY);

Aplicar una funcio a una imatge: J = arrayfun(@myfunction, I);

Per obtener els pixels veins: J = colorfilt (I, [f c], ‘sliding’, @myfunction);

La variable X de myfunction son tots els veins del