VISIÓ PER

COMPUTADOR

Informe de laboratori 6

Marc Cervilla Rovira

07/04/2021

**Introducció:**

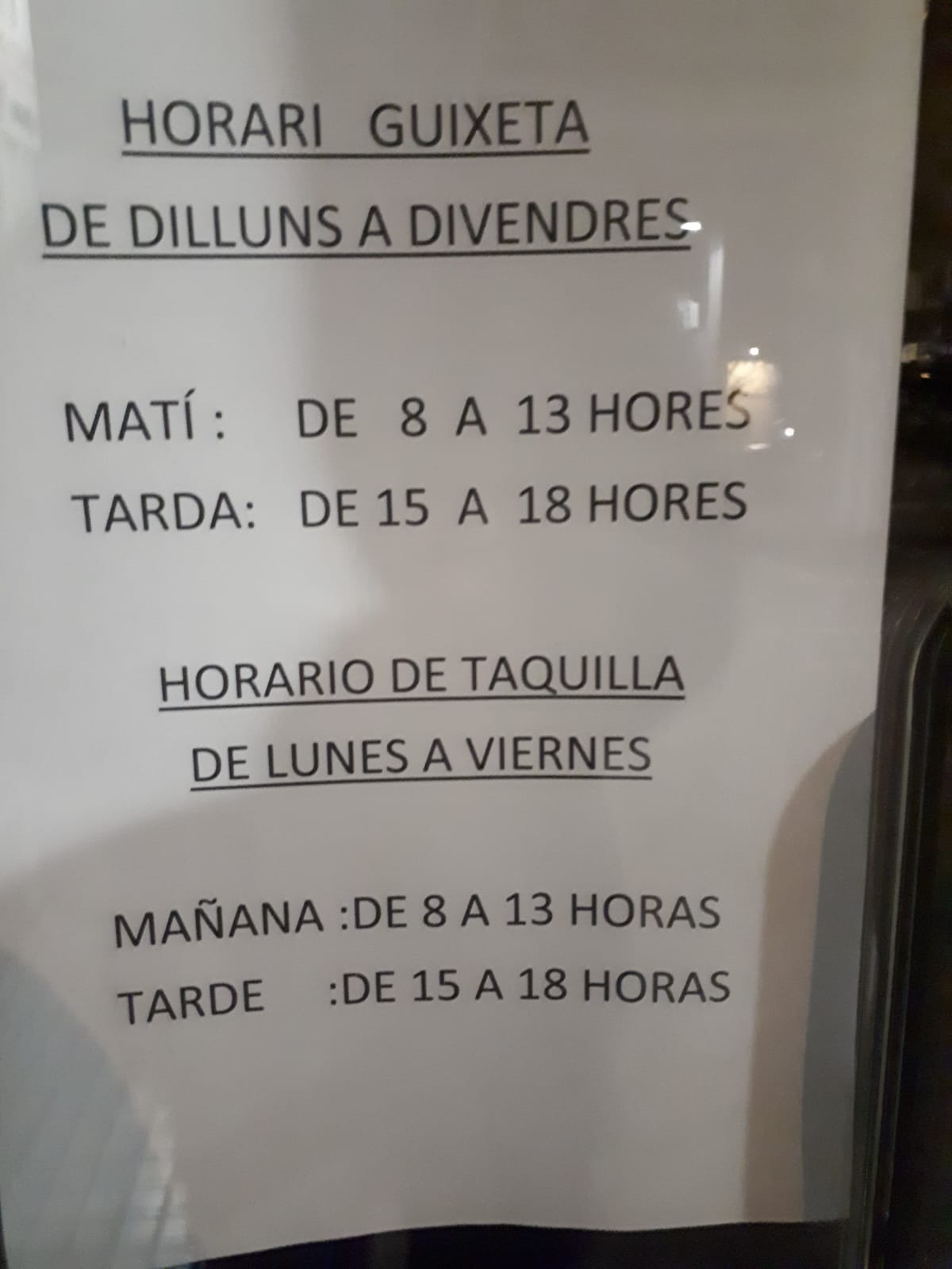
En aquesta sessió farem una petita aplicació per a introduir-nos en les tècniques bàsiques de binarització i segmentació d’imatges, sobretot treballarem amb els següents conceptes:

* Binaritzacions globals.
* Binarització local mitjançant la funció *colfilt*.
* Segmentació per agrupament de píxels en imatges binaritzades.

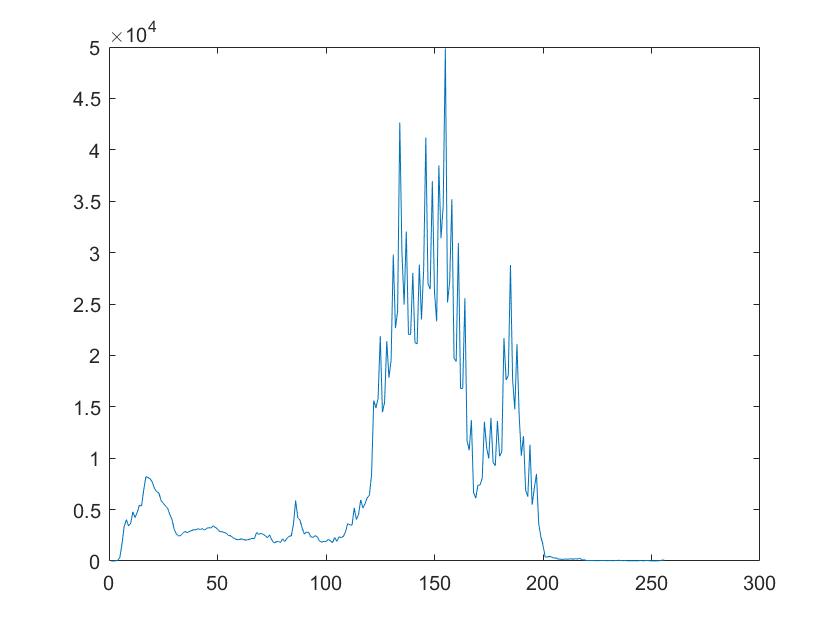
L’aplicació consistirà en fer una captura d’un document de text i obtenir el contrast del text a partir d’una imatge realitzada amb el mòbil, ja que normalment la iluminació d’aquestes imatges no està controlada.

**Implementació:**

En primer lloc, obtenim la imatge amb la càmera del mòbil. En el meu cas, he utilitzat la següent imatge, on la iluminació no és gens bona:



A continuació, he passat la imatge a nivells de gris i he realitzat una binarització global per tal que ens quedi, per una banda el full de paper en blanc i, per altra banda, el fons en negre. Si fem l’histograma de la imatge podem veure el següent:



Per tant, si fem una binarització del que sería el 80% de la imatge, podem estimar a ull que estarà pel voltant de 50, així doncs, utilitzant un threshold en 50 podem binaritzar la imatge.

I = rgb2gray(imread('mala\_ilum.jpeg'));

J = I>50;

Però com és normal, també ens algunes lletres dins de la fulla, per tant, per treure-les he utilitzat un close de la següent manera:

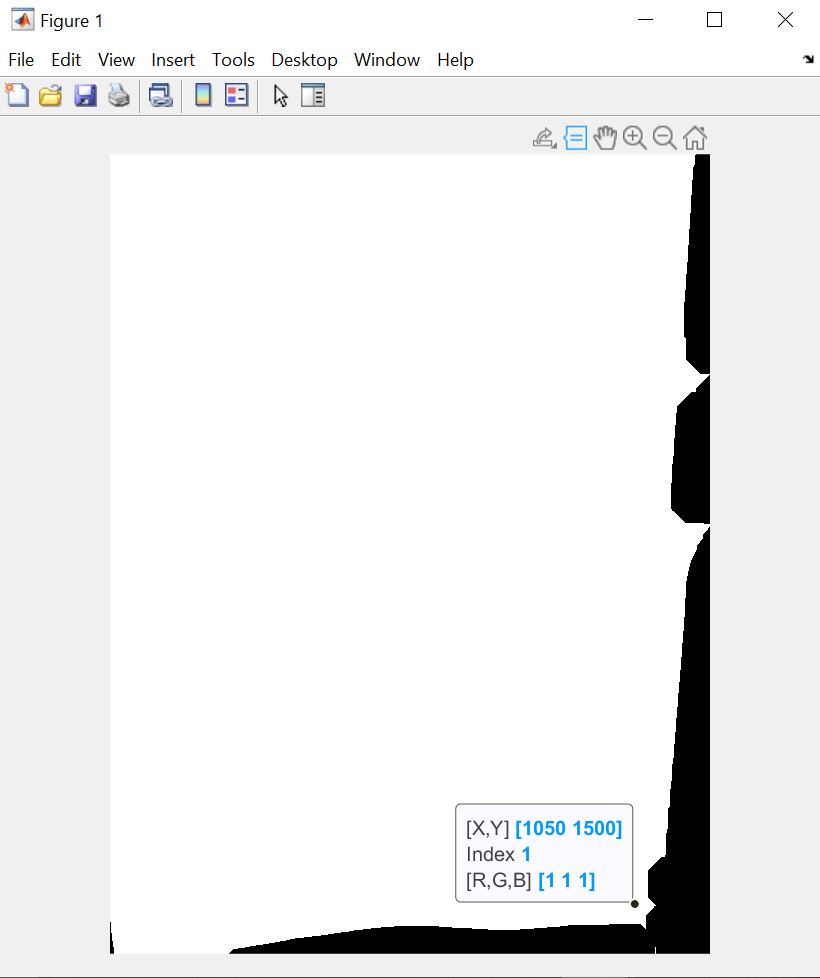
disk = strel('disk',50);

J = imclose(J,disk);

El següent pas sería retallar la imatge per tal que només hi hagi el full (sense el fons), això ho podem aconseguir utilitzant la funció de matlab ‘imcrop’.

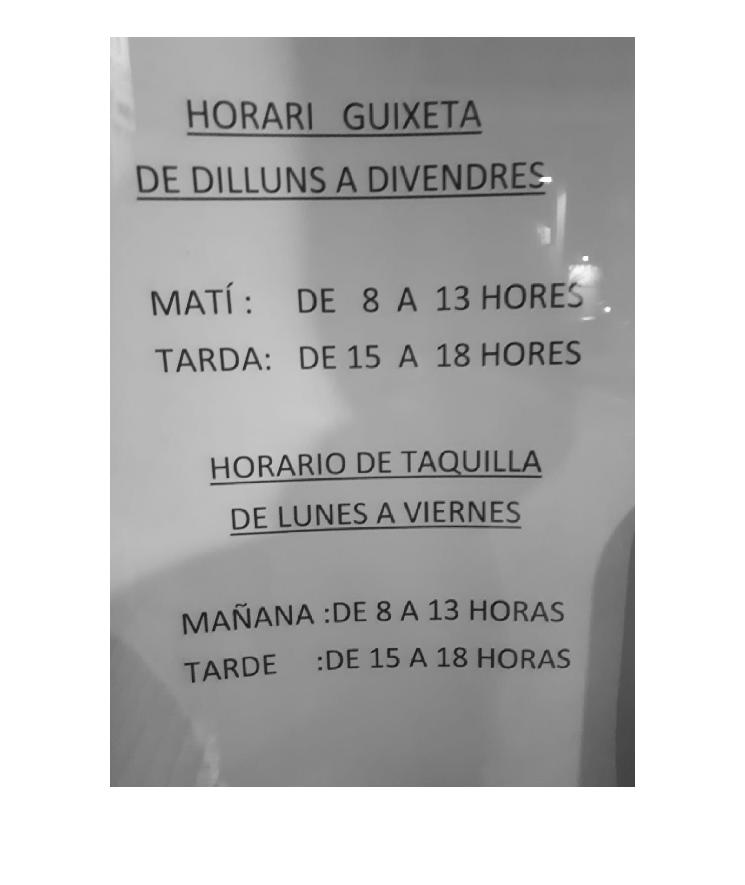
En el cas que obtinguessim la imatge binaria del full perfectament recuadrada podriem aplicar la funció ‘imcrop’ a la imatge inicial en escala de grisos utilitzant la imatge binaria com a mapa. Però aquest no és el nostre cas, ja que la iluminació no és gens bona i no queda ben recuadrat com veurem a continuació, per tant, podem mirar la imatge que ens queda, mirar en quin punt acabaria la pàgina manualment i aplicar la funció ‘imcrop’ utilitzant els punts que haguem trobat.

En el meu cas, la meva imatge binaritzada i els punts trobats manualment es veuen a continuació:



Com ja he esmentat, utilitzant la funció de la següent manera aconseguim la imatge retallada que voliem.

R = imcrop(I,[0,0,1050,1500]);



Un cop tenim la imatge retallada, volem aplicar una binarització local per tal de contrastar les lletres del full i, per exemple, contar quants caracters tenim.

Per fer-ho utilitzarem la funció ‘colfilt’, la qual aplica, per cada pixel, la funció que nosaltres volguem a una finestra lliscant dels pixels del voltant.

La crida a la funció seria la següent:

F = colfilt(R,[M N], 'sliding', @myfunction);

i la funció myfunction:

function [y] = myfunction(vec\_x)

%UNTITLED3 Summary of this function goes here

% Detailed explanation goes here

[f c] = size(vec\_x);

mean = sum(vec\_x)/f;

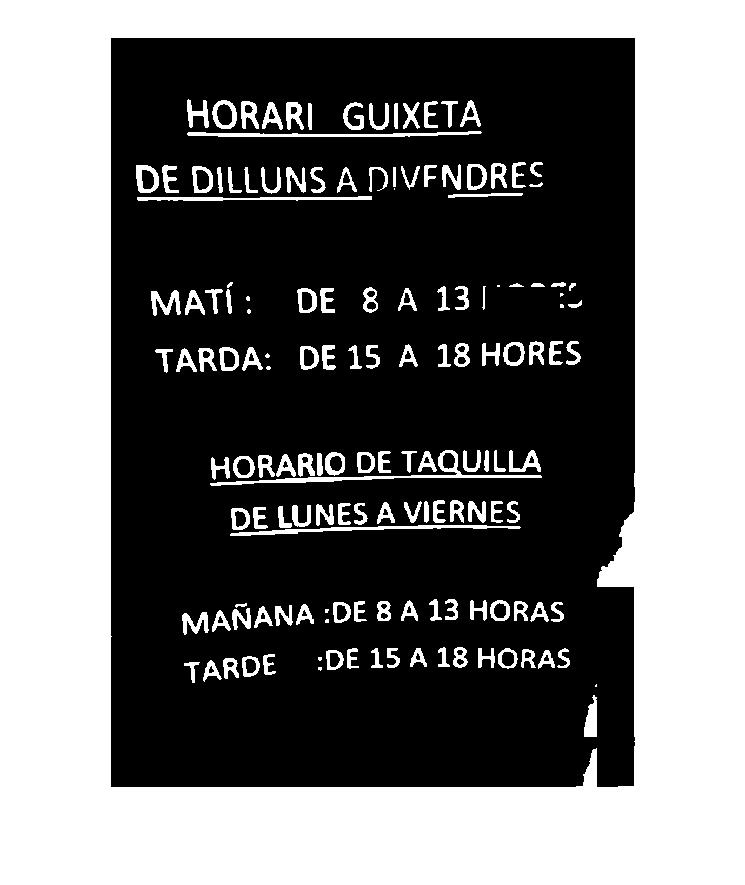
y = vec\_x(f/2+0.5)>mean+K;

end

El que fem és aplicar per tota la imatge una finestra lliscant de tamany [M N], on un pixel es binaritzarà segons el threshold resultant de la mitjana d’aquella finesta més un valor K, on M, N i K són variables lliures que dependràn de la iluminació de cada imatge i de la grandaria del text.

He probat d’utilitzar una finestra més gran del tamany d’un caràcter, però els resultats no són gens bons i no binaritza bé la imatge.

En el meu cas, he utilitzat una finestra de [5 5] i una K de 40 nivells de gris per sobre la mitja i el resultat, tot i els canvis d’iluminació per tota la imatge, és prou correcte com es mostra a continuació:



Podem observar que en algun punt concret dona resultats erronis degut al reflexe que tenia la imatge original, però en general obtenim el text binaritzat a blanc separat del que seria el fons de la pàgina.

Per últim, podem contar el nombre de caràcters que té el full utilitzant la funció següent:

CC = bwconncomp(F);

La qual ens retorna un struct dels diferents components connexes de la imatge i podem observar que ens apareixen 153 objectes diferents, que si ens poséssim a contar els caràcters que conté la nostre imatge veuriem que aproximadament és un valor correcte.