Repte 4

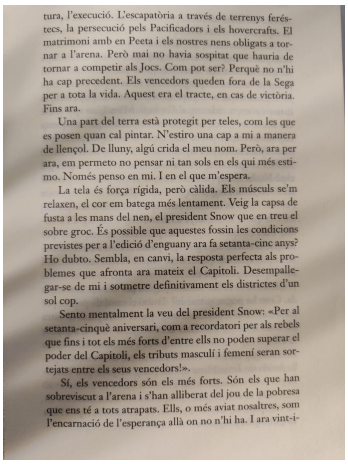
Alumne: Martí Caixal Joaniquet
NIU: 1563587, data: 15/03/2022

1 Introducció

Al repte 4 es demana fer una fotografia d'una pàgina d'un llibre que contingui molt de text. És important que hi hagi mala il·luminació o ombres per tal de simular un cas real. L'objectiu, doncs, és poder trobar el nombre de lletres, paraules i línies.

2 Imatge original

A contiuació es mostra la imatge original amb la que es farà l'activitat.



tura, l'execució. L'escapatòria a través de terrenys feréstecs, la persecució pels Pacificadors i els hovercrafts. El matrimoni amb en Peeta i els nostres nens obligats a tornar a l'arena. Però mai no havia sospitat que hauria de tornar a competir als Jocs. Com pot ser? Perquè no n'hi ha cap precedent. Els vencedors queden fora de la Sega per a tota la vida. Aquest era el tracte, en cas de victòria. Fins ara.

Una part del terra està protegit per teles, com les que es posen quan cal pintar. N'estiro una cap a mi a manera de llençol. De lluny, algú crida el meu nom. Però, ara per ara, em permeto no pensar ni tan sols en els qui més estimo. Només penso en mi. I en el que m'espera.

La tela és força rígida, però càlida. Els músculs se'm relaxen, el cor em batega més lentament. Veig la capsa de fusta a les mans del nen, el president Snow que en treu el sobre groc. És possible que aquestes fossin les condicions previstes per a l'edició d'enguany ara fa setanta-cinc anys? Ho dubto. Sembla, en canvi, la resposta perfecta als problemes que afronta ara mateix el Capitoli. Desempallegar-se de mi i sotmetre definitivament els districtes d'un sol cop.

Sento mentalment la veu del president Snow: «Per al setanta-cinquè aniversari, com a recordatori per als rebels que fins i tot els més forts d'entre ells no poden superar el poder del Capitoli, els tributs masculí i femení seran sortejats entre els seus vencedors!».

Sí, els vencedors són els més forts. Són els que han sobreviscut a l'arena i s'han alliberat del jou de la pobresa que ens té a tots atrapats. Ells, o més aviat nosaltres, som l'encarnació de l'esperança allà on no n'hi ha. I ara vint-i-

Figura 1: Imatge original

3 Binarització

El primer pas és binaritzar la imatge amb la que es treballa. És a dir, passar de tenir múltiples tonalitats de colors, a tenir només 2 colors. La imatge resultant (figura 2) permet realitzar operacions morfològiques amb més facilitat.

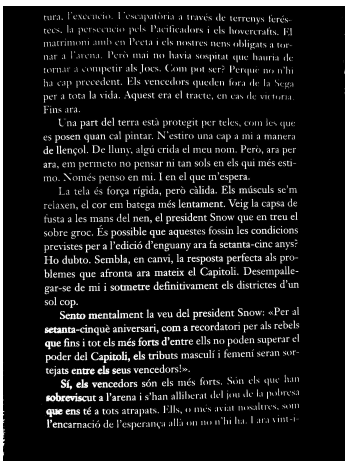


Figura 2: Imatge binaritzada

És molt important trobar una bona binarització. A continuació es mostren dos intents on el resultat no era satisfactori (figura 3). El primer només és capaç de binaritzar la zona més fosca de la foto. El segon, en canvi, està agafant zones sense lletres degut a la ombra projectada.

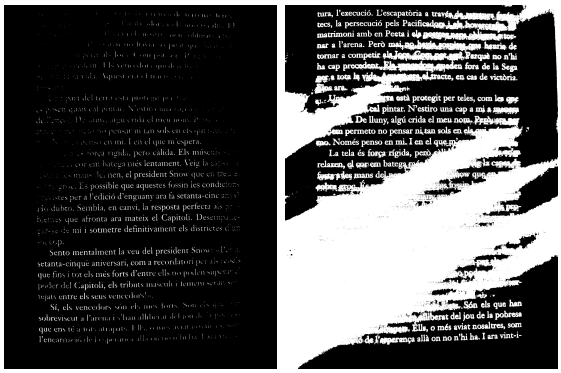


Figura 3: Binaritzacions incorrectes

Com es veu, la il·luminació dificulta molt la feina i una bona binarització és el que marca la diferència entre poder completar l'objectiu o no.

Tot i que sembla que la binarització és correcte, les lletres presenten diferents comportaments depenent de la zona on apareguin de la imatge (figura 4). No només hi ha aquest problema, sinó que les puntuacions que no estan connectades amb la lletra poden donar peu a càlculs erronis.

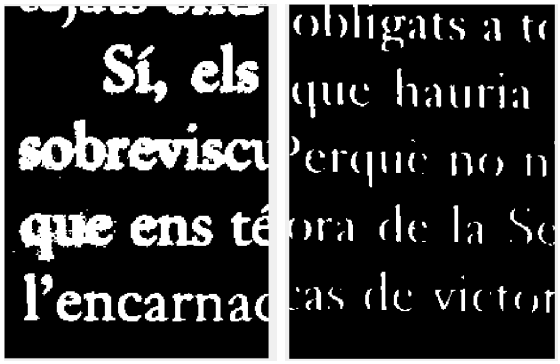


Figura 4: lletres mal definides

4 Eliminant variació de la imatge

Com ja s'ha vist, una simple binarització no és òptima quan hi ha ombres per en mig. Una alternativa és buscar la variació del fons i restar-li la imatge original. D'aquest manera s'aconsegueix tenir només el text sense gaires variacions.

Per trobar la variació del fons es pot fer amb la següent operació morfològica:

```
kernelFons = strel('disk', 60);
fons = imclose(im, kernelFons);
```

Com es pot veure, les zones amb més ombra ara són les que estan més fosques. Quan es resti al fons la imatge original, les lletres d'aquella zona quedaran "equilibrades"amb les de les zones amb més llum. De la mateixa manera, les

zones amb excès de llum també estan equilibrades amb la resta de la imatge.

Figura 5: Variació del fons

La figura 6 mostra com ara ja no hi variació al text. Es pot veure la millora si es compara amb les figures 2 i 4

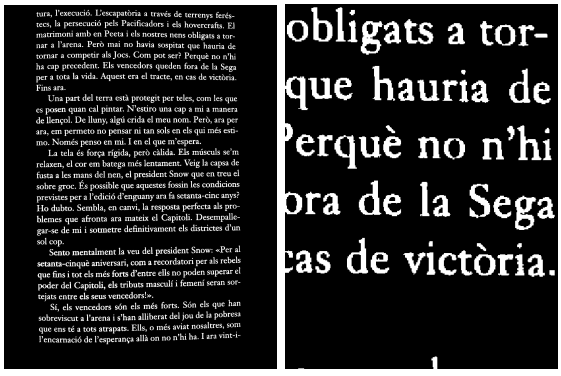


Figura 6: Lletres sense variacions

5 Detecció de lletres

Per aconseguir una bona definició de les lletres, es fa ús de dos operacions morfològiques:

1. Close: s'aconsegueix tancar (ajuntar) lletres que estan separades en dos, però a la vegada es manté relativament la mida exterior de la lletra. El kernel utilitzat té una forma de rectangle vertical, permetent així ajuntar verticalment els accents i altres puntuacions per molt altes que siguin. La mida horitzontal és més reduida per tal de no ajuntar dos lletres en una de sola.

```
tancar = strel('rectangle', [10 5]);
lletres = imclose(im, tancar);
```

2. Close + Erode: s'erosiona els contorns, aconseguint separar la gran majoria de lletres que a la primera operació havíen quedat ajuntades. Amb el kernel en forma de línia vertical, s'aconsegueix eliminar la majoria de línies que anteriorment han fusionat dues lletres.

```
erosionar = strel('rectangle', [3 1]);
lletres = imerode(lletres, erosionar);
```

La figura 7 mostra que la primera operació, tot i resoldre el problema amb les puntuacions, ajunta algunes lletres. En canvi, amb la segona s'aconsegueixen separar.

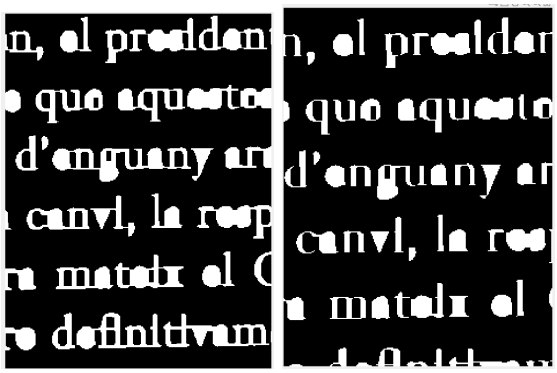


Figura 7: Operacions morfològiques 1 i 2

A continuació es mostra el resultat final de la imatge sencera. Si bé el resultat ha millorat, ha resultat impossible poder separar les lletres de la zona més fosca alhora que ajuntar les de la zona més clara.

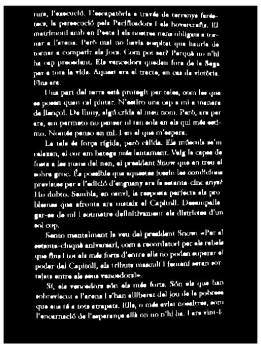


Figura 8: Lletres separades

Mitjançant la funció de Matlab "*bwlabel*", es pot assignar una etiqueta a cada regió continua blanca i així calcular quin és el total que hi ha. La imatge sense tractar dona resultats molt alts ja que moltes lletres amb accent, punts de la *I* i similars estan separats i conten com a dos. El resultat d'aplicar només closing, com ja s'ha vist, ajunta lletres i fa que només contin com a una. Amb el resultat final, que manté les puntuacions juntes però separa les lletres, s'aconsegueix un valor de 1678 lletres.

Original	Close	Close + Erode
2054	1151	1678

Taula 1: Total de lletres

6 Detecció de paraules

El segon objectiu és poder determinar el nombre de paraules que hi ha la fotografia. De nou, es fa ús de dos operacions morfològiques:

1. Dilate: Permet ampliar els contorns. Amb el kernel utilitzat de 10x10, el resultat és ja bastant satisfactori. Verticalment permet ajuntar les puntuacions de les lletres, però és suficientment petit per no interferir amb altres files. Horitzontalment és suficientment ample com per ajuntar lletres entre elles, però no el suficient com per sobrepassar l'espai que separa paraules entre elles.

```
rectangleParaula = strel('rectangle',
    [10 10]);
paraules = imdilate(im,
    rectangleParaula);
```

2. Dilate + Close: Amb el closing s'aconsegueix tancar els espais interiors que hagin pogut quedar dins de les paraules.

```
square = strel('square', 5);
paraules = imclose(paraules, square);
```

La figura 9 mostra el resultat d'ambdues operacions morfològiques.

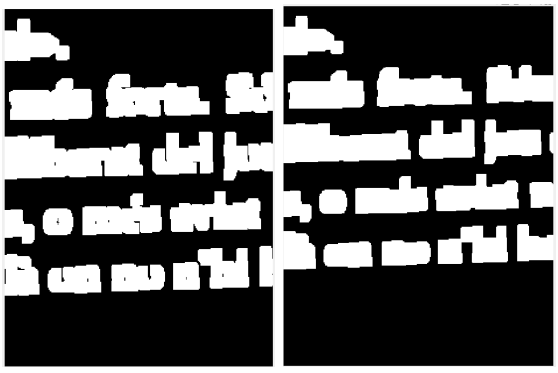


Figura 9: Operacions morfològiques 1 i 2

La imatge resultant és bastant satisfactòria (veure figura 10). Comparant-la amb la imatge original, sembla que aconsegueix separar pràcticament totes les paraules correctament.

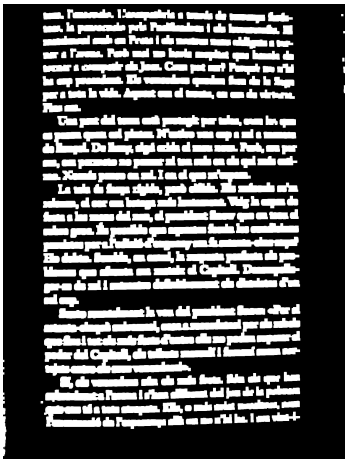


Figura 10: Paraules separades

De la mateixa manera que l'apartat anterior, s'ha fet un recompte de quantes paraules hi ha a la pàgina. El resultat final és 288 paraules. La diferència entre la primera operació i la segona és que, molt probablement, la primera encara té alguna paraula separada en dos. Fent el closing s'aconsegueix ajuntar aquestes restants.

Original	Dilate	Dilate + Close
2054	330	288

Taula 2: Total de paraules

7 Detecció de línies

El tercer i últim objectiu és detectar i comptar el nombre de línies que hi ha a la pàgina. També s'ha fet ús de dos passos per arribar al resultat:

1. Dilate: l'objectiu d'aquest primer pas és ajuntar totes les paraules d'una línia en una sola regió. Al ser totes de la mateixa llargada i alçada, es pot aconseguir fàcilment amb un kernel rectangular 10 x 50.

```
rectangleLinia = strel('rectangle',
    [10 50]);
linies = imdilate(im, rectangleLinia);
```

2. Dilate + Erode: aquesta segona operació permet crear una separació entre les línies. Així les regions que ja estan correctament definides no es modifiquen, però a la vegada permet separar les regions que s'estan ajuntant per una fina línia.

```
rectangleSeparar = strel('rectangle',
    [14 70]);
linies = imerode(linies,
    rectangleSeparar);
```

La figura 11 mostra el resultat d'ambdues operacions morfològiques. Es pot apreciar perfectament la separació de les dos últimes línies. Per altra banda, al tractar

amb regions més grans, és molt fàcil eliminar el soroll que hi ha als laterals de la imatge original.

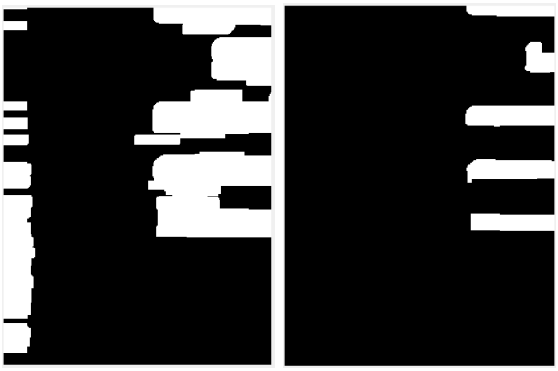


Figura 11: Operacions morfològiques 1 i 2

La imatge resultant és molt satisfactoria (veure figura 12).

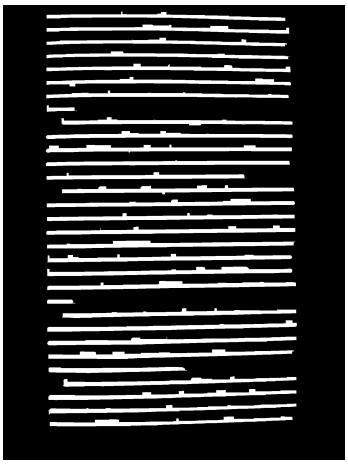


Figura 12: Línies separades

Igual que s'ha realitzat als anteiors apartats, es procedeix a fer un recompte de quantes línies hi ha a la pàgina. El resultat final és 31 línies, fàcilment comprovable que en aquest cas ha encertat el resultat. També es veu com únicament amb la primera operació hi ha línies que estan juntes i només compten com a una de sola.

Original	Dilate	Dilate + Erode
2054	45	31

Taula 3: Total de línies