

UNIVERSITAT POLITÈCNICA DE CATALUNYA

---

## Sessió 4

---

### INFORME

### VISIÓ PER COMPUTADOR

#### *Autors*

DAVID LATORRE  
ADRIÀ AUMATELL



UNIVERSITAT POLITÈCNICA  
DE CATALUNYA  
BARCELONATECH

## PATRÓ 1:

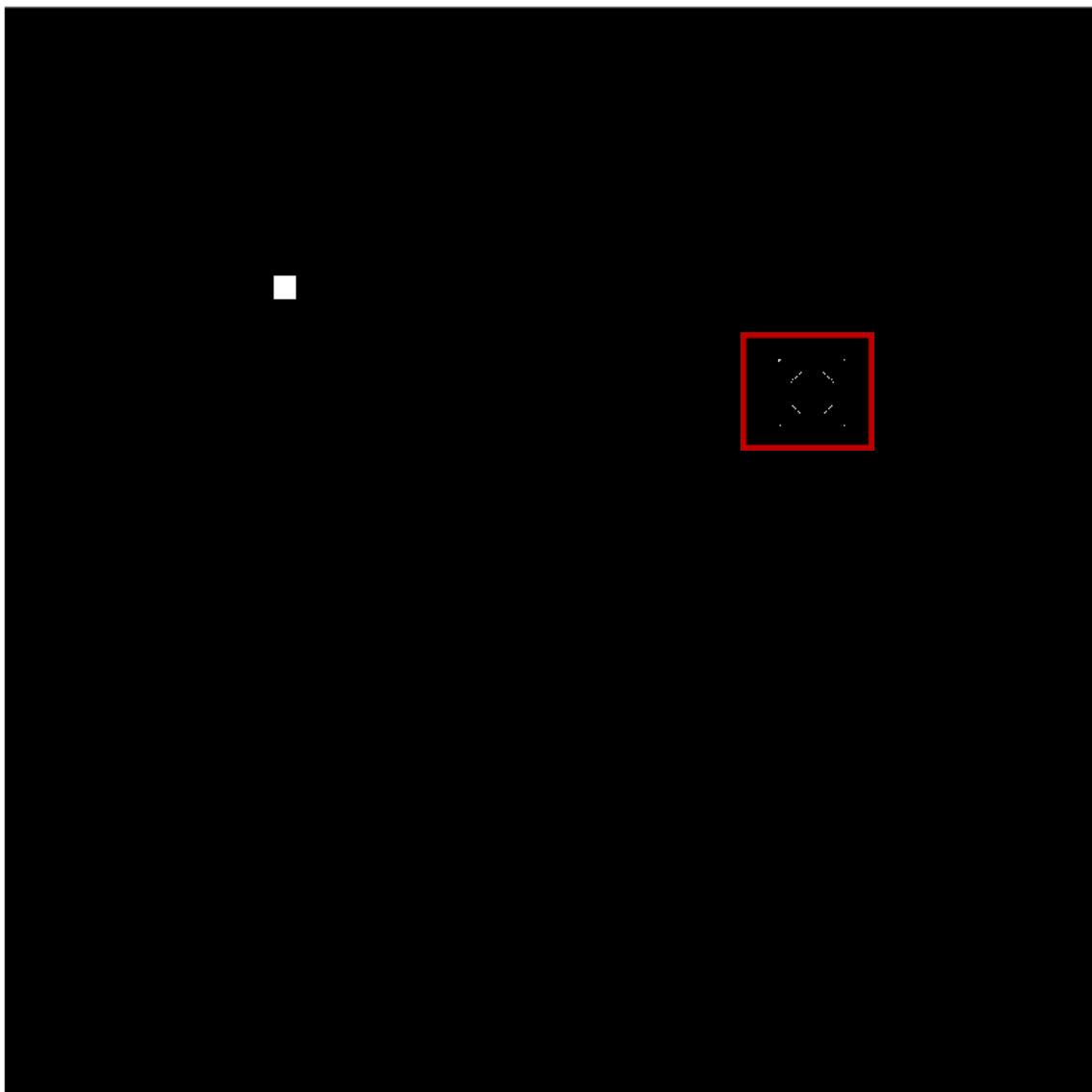
El primer patró que hem escollit és el següent:



Per poder obtenir el patró, realitzarem tres passos. Obtindrem per separat els còrners, el rombe central i la resta de la imatge.

Per començar, aillarem el cos del patró. Aplicarem dues erosions a la imatge amb una línia com a element estructurant amb angle 45 i 135 graus respectivament. El resultat és

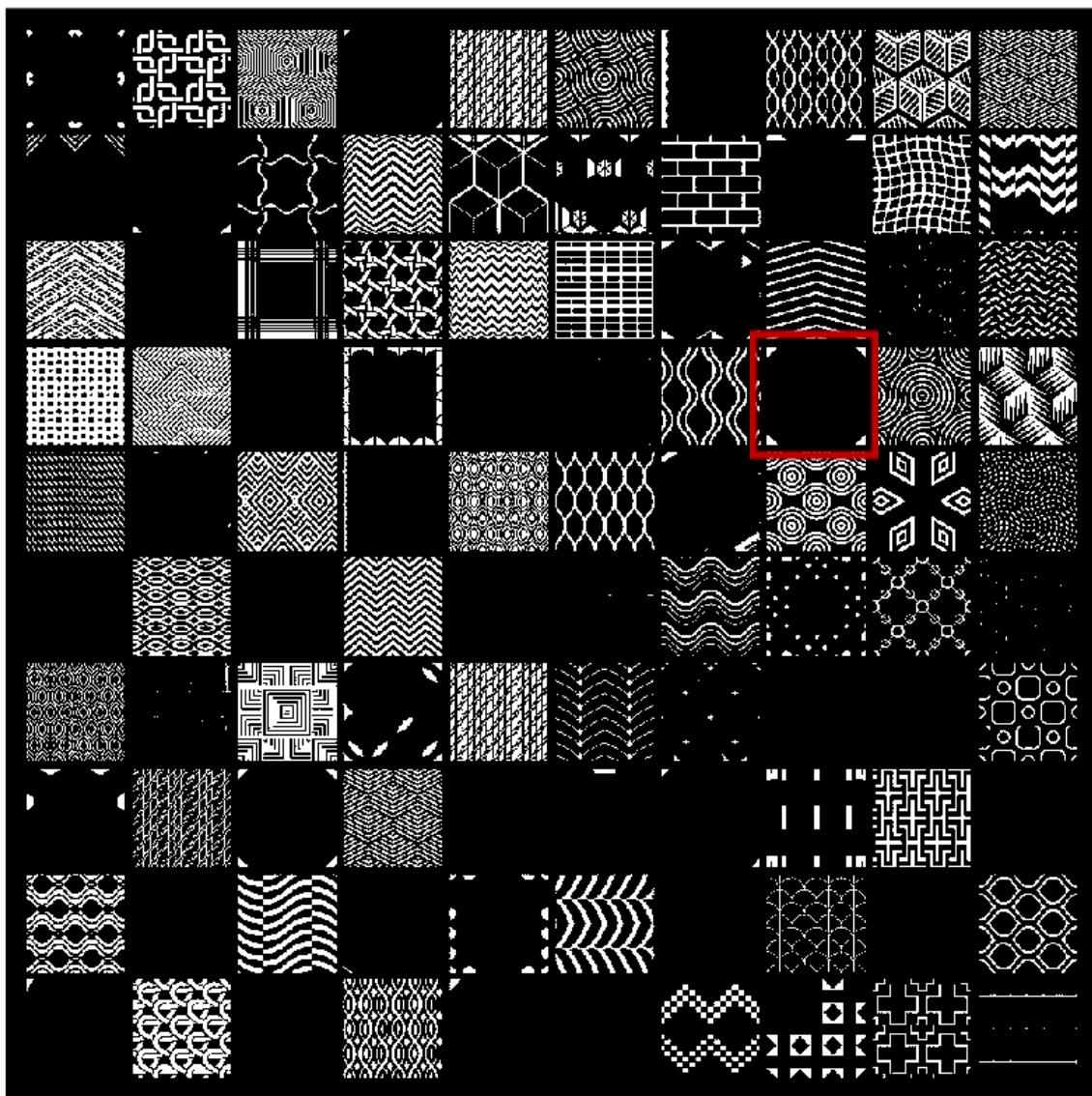




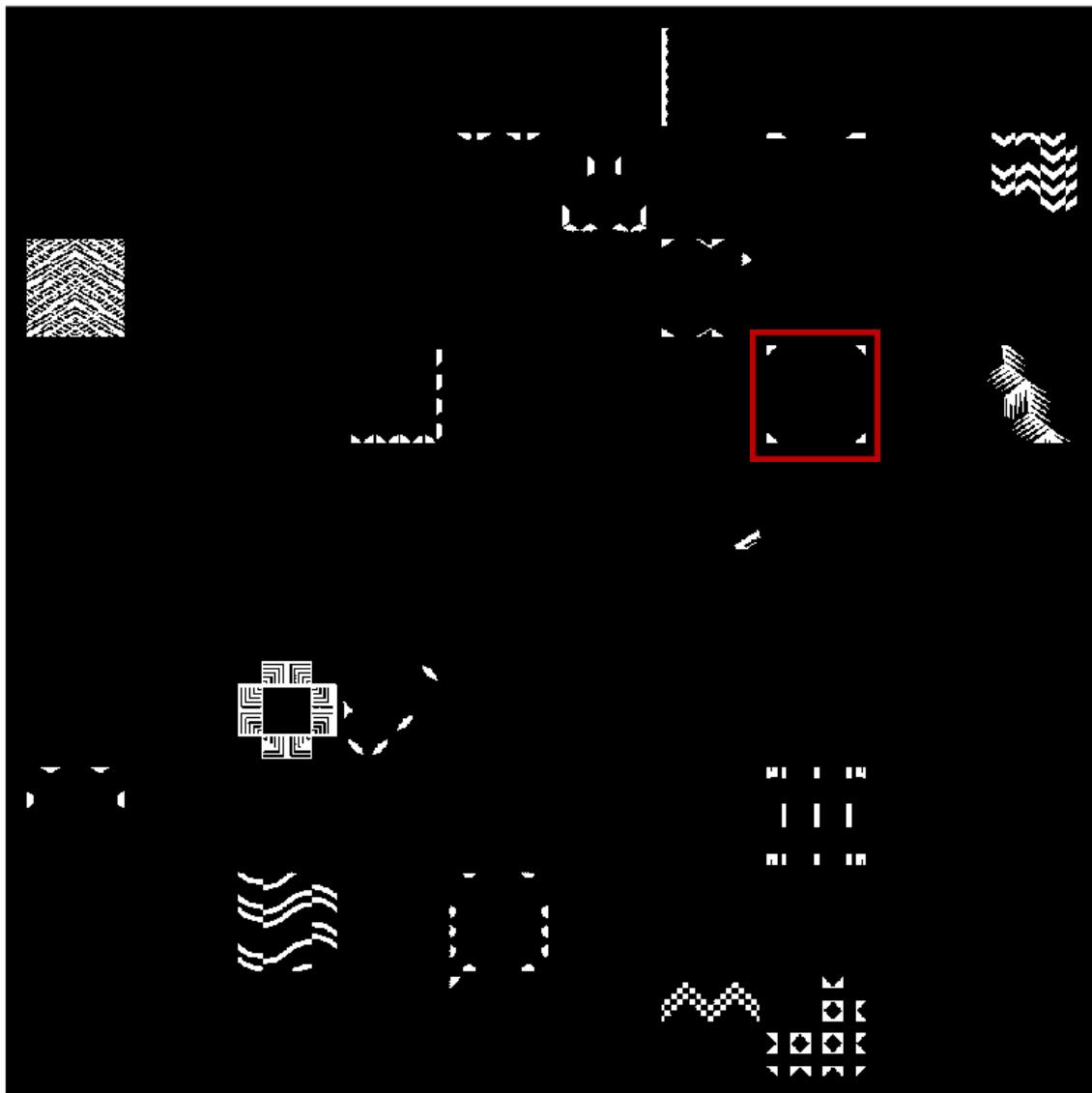
Per acabar aquest pas, només ens falta eliminar el quadrat de la imatge. És tan fàcil com realitzar una obertura amb un element estructurant quadrat de costat 20. Una vegada fet això, només cal reconstruir la imatge per obtenir el cos del patró:



El següent pas serà obtenir els còrners del patró. Com que els còrners son zones petites, eliminarem els components amb més àrea mitjançant un open per suprimir els components de menor àrea i una operació AND per “invertir” la imatge. El resultat és el següent:

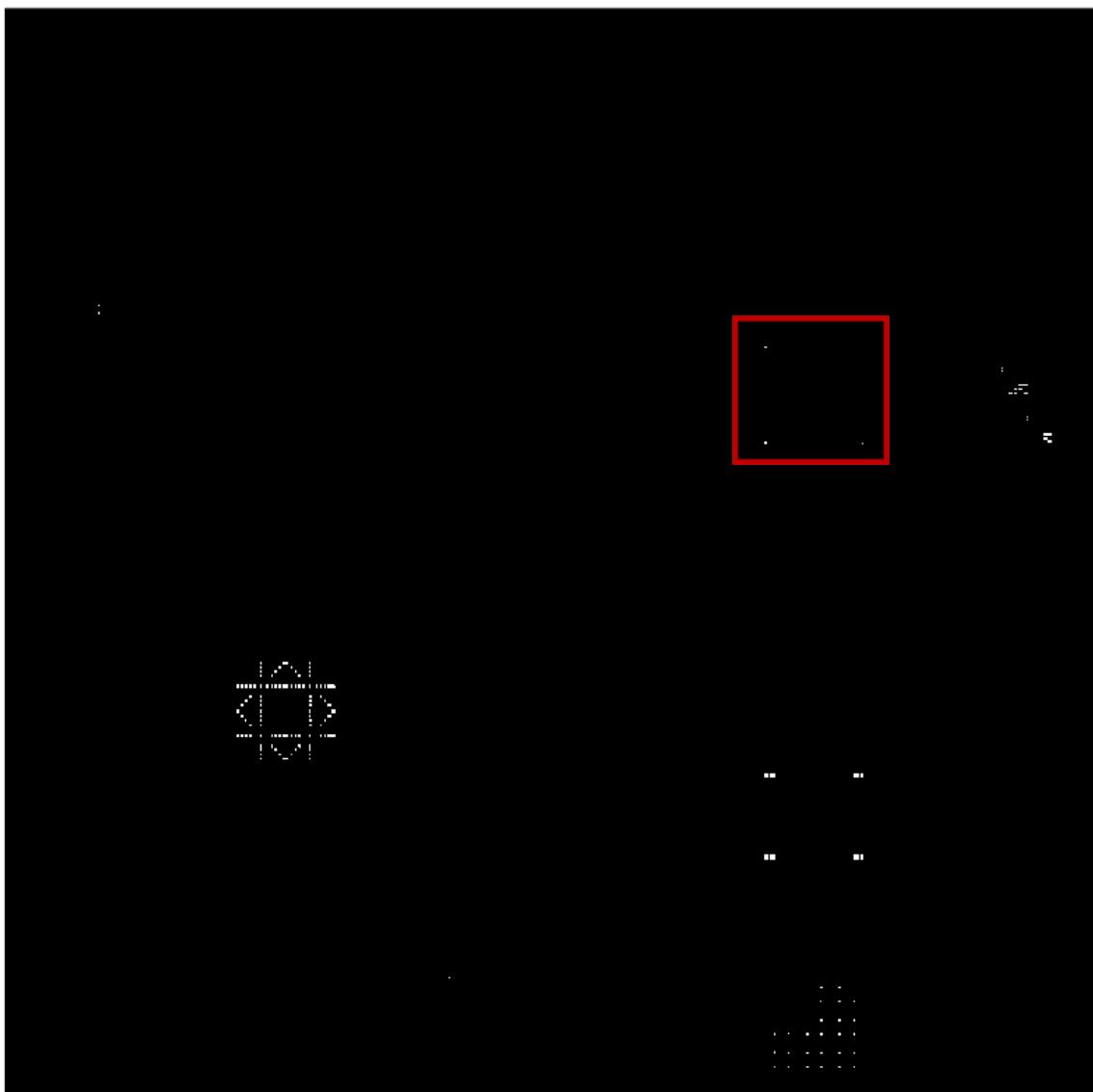
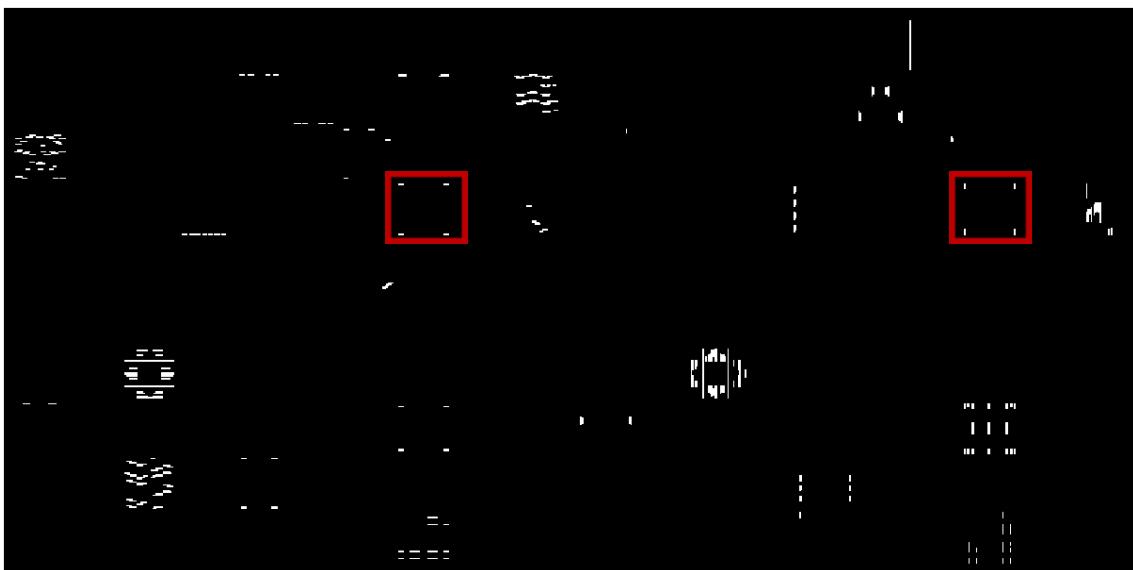


Per eliminar els patrons que no volem, intentarem eliminar totes aquelles formes que tinguin una mida diferent a la dels còrners de patró. Apliquem dues erosió, una amb element estructurant 5 i l'altra amb element estructurant 6, i n'obtenim el residu i el reconstruïm. El resultat és el següent:



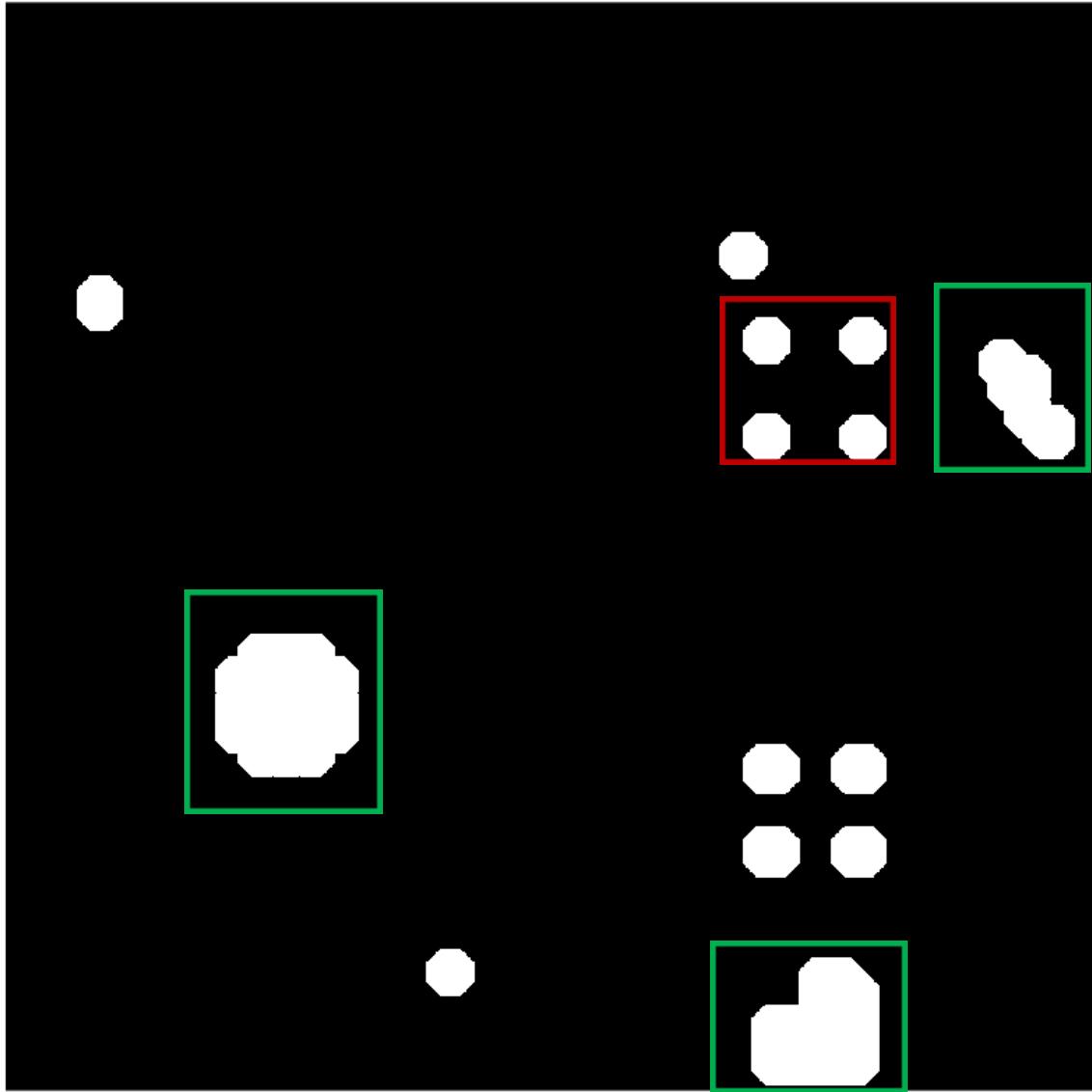
La imatge és més neta que l'anterior, però encara queda feina per fer. Aprofitant que els còrner tenen forma triangular, intentarem quedar-nos amb aquells píxels que sobrevisquin tan a una erosió amb EE línia amb angle o graus com a una erosió amb EE línia amb angle 90 graus.

A continuació mostrem les imatges obtingudes a partir de les dues erosions i la imatge resultant de fer una AND amb les dues anteriors:

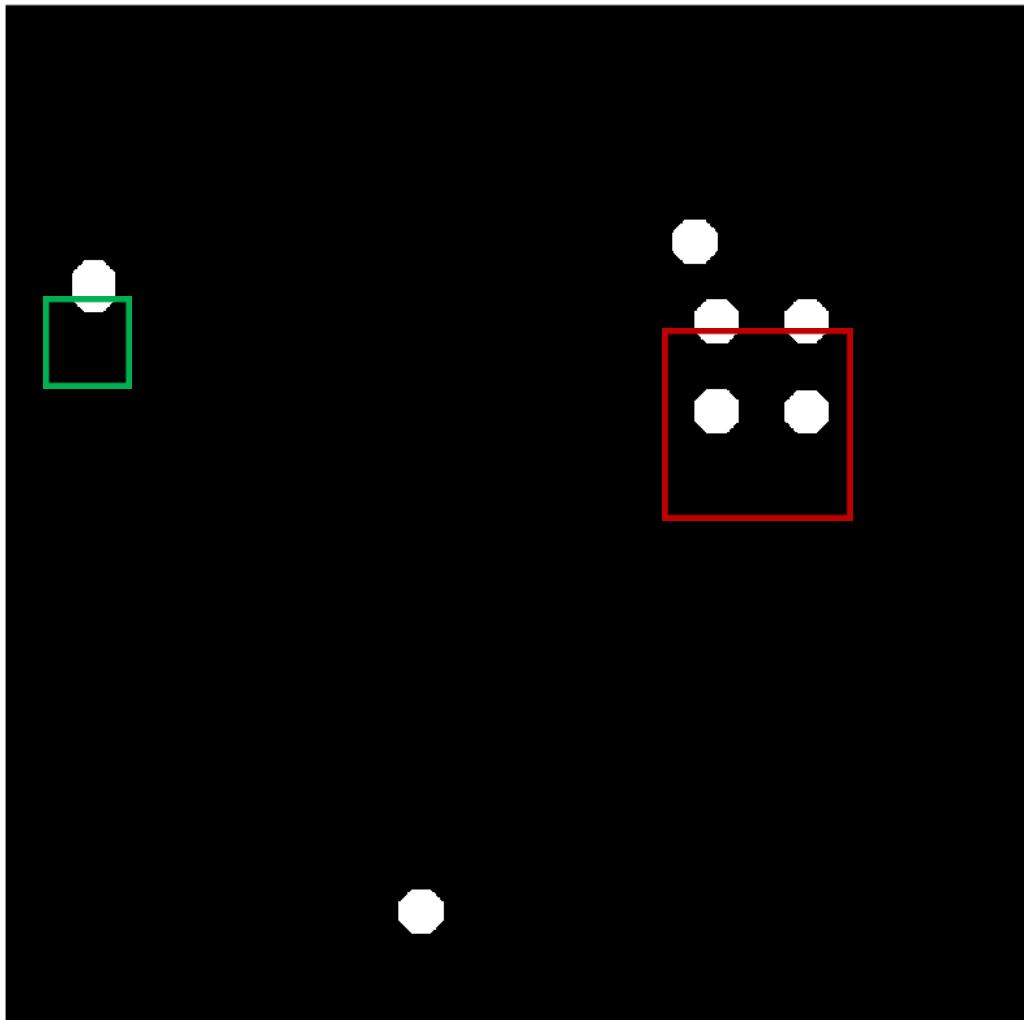


(Nota: Dins del requadre vermell hi ha un píxel per cada còrner, a la imatge no ho sembla)

Com que els diferents components que queden a la imatge tenen mida semblant, intentarem eliminar aquells que estiguin molt propers entre ells, traient avantatge del fet que els còrnars del patró a aïllar estan força separats. Això es pot aconseguir fàcilment mitjançant una dilatació:

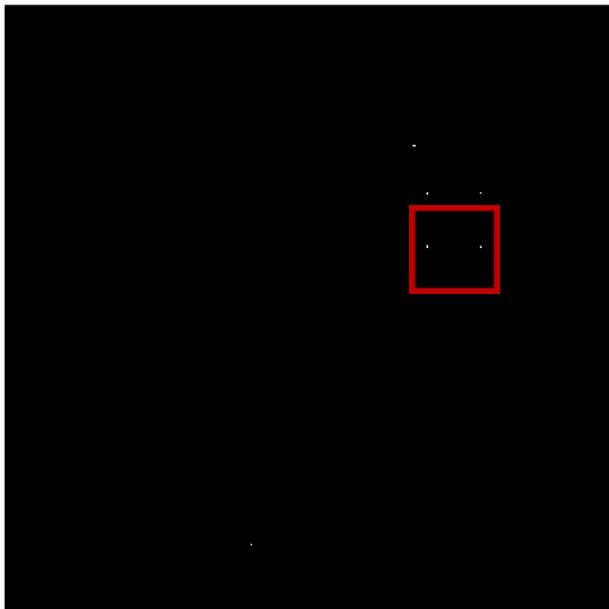


Tal com es veu a la imatge, han aparegut 3 grans estructures (marcades amb verd), que son fàcilment eliminables mitjançant el residu de dues obertures amb uns EE ben escollits. Després d'aquesta operació, realitzarem una obertura per eliminar-ne les restes que puguin quedar. Ara tenim això:

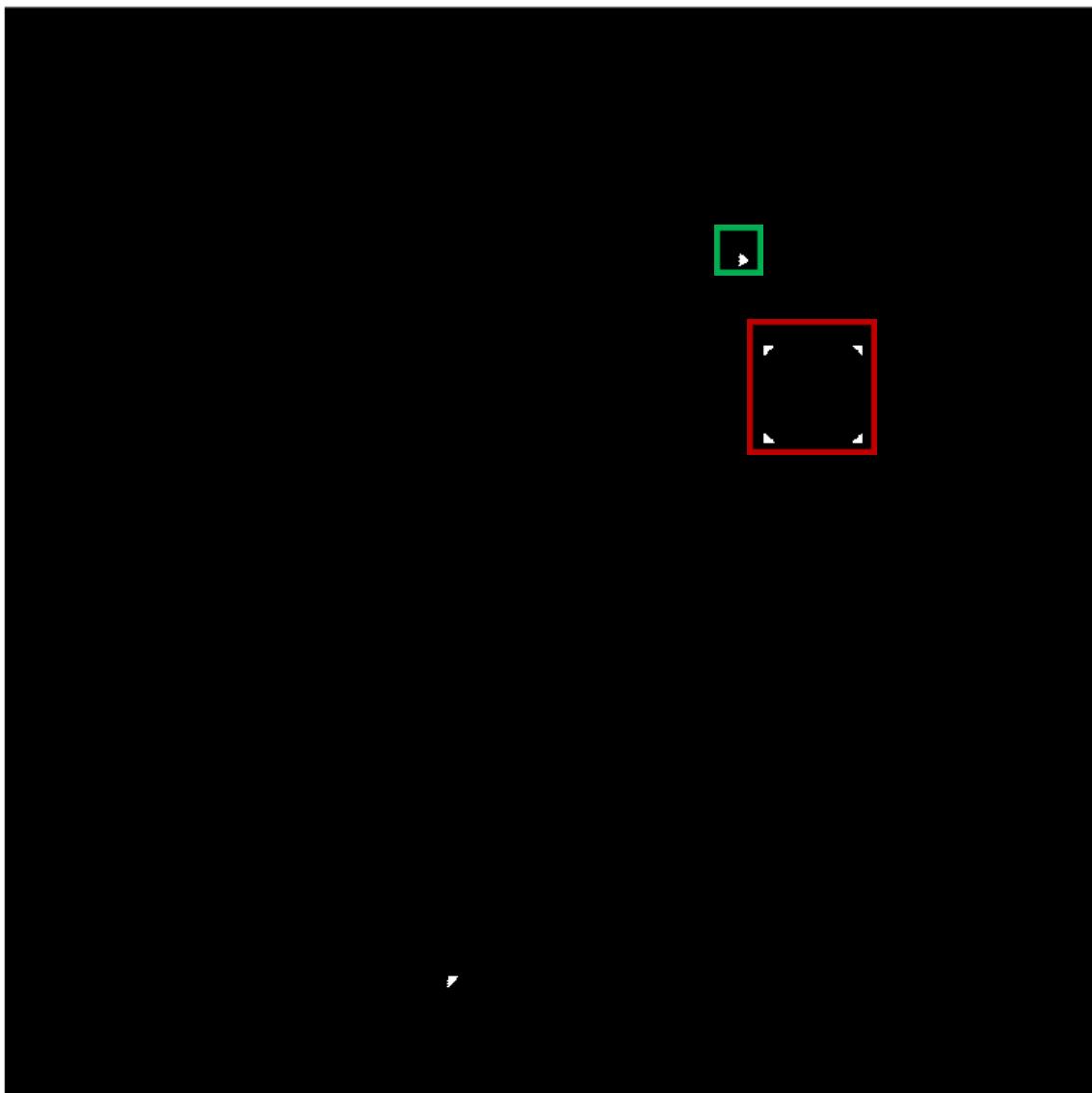


El nostre pròxim objectiu serà el component ressaltat amb el quadre verd. Donat que té una alçada lleugerament major que la resta, podem eliminar-lo obtenint el residu

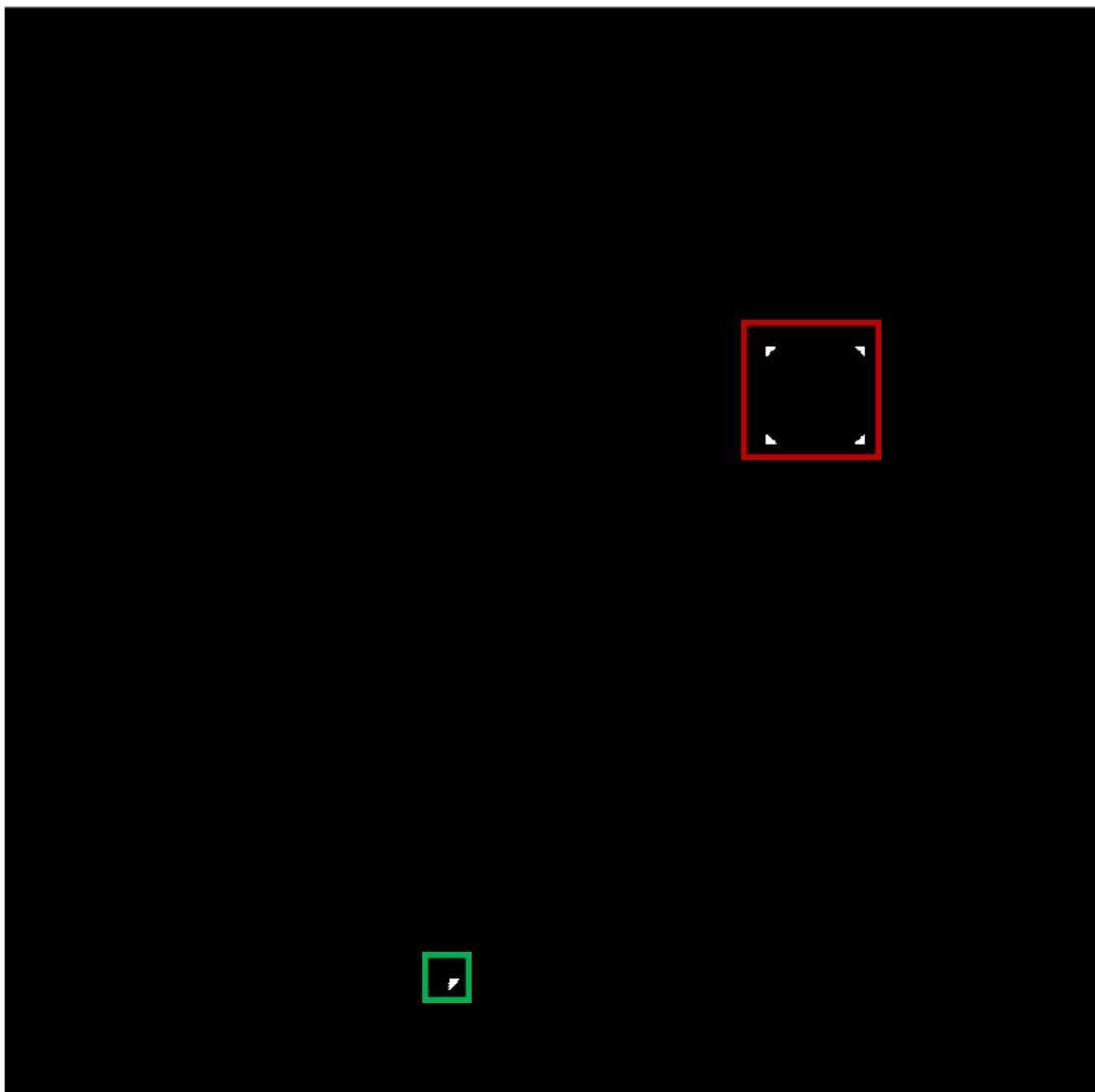
entre dues obertures amb EE línia amb angle 90 graus. Després d'eliminar les restes amb una erosió, obtenim:



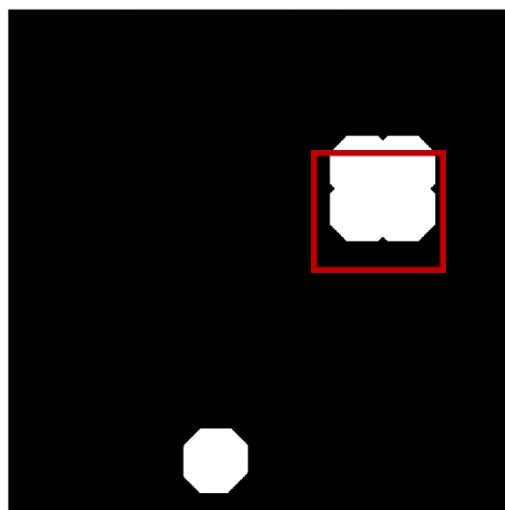
Per continuar, ens cal reconstruir la imatge:



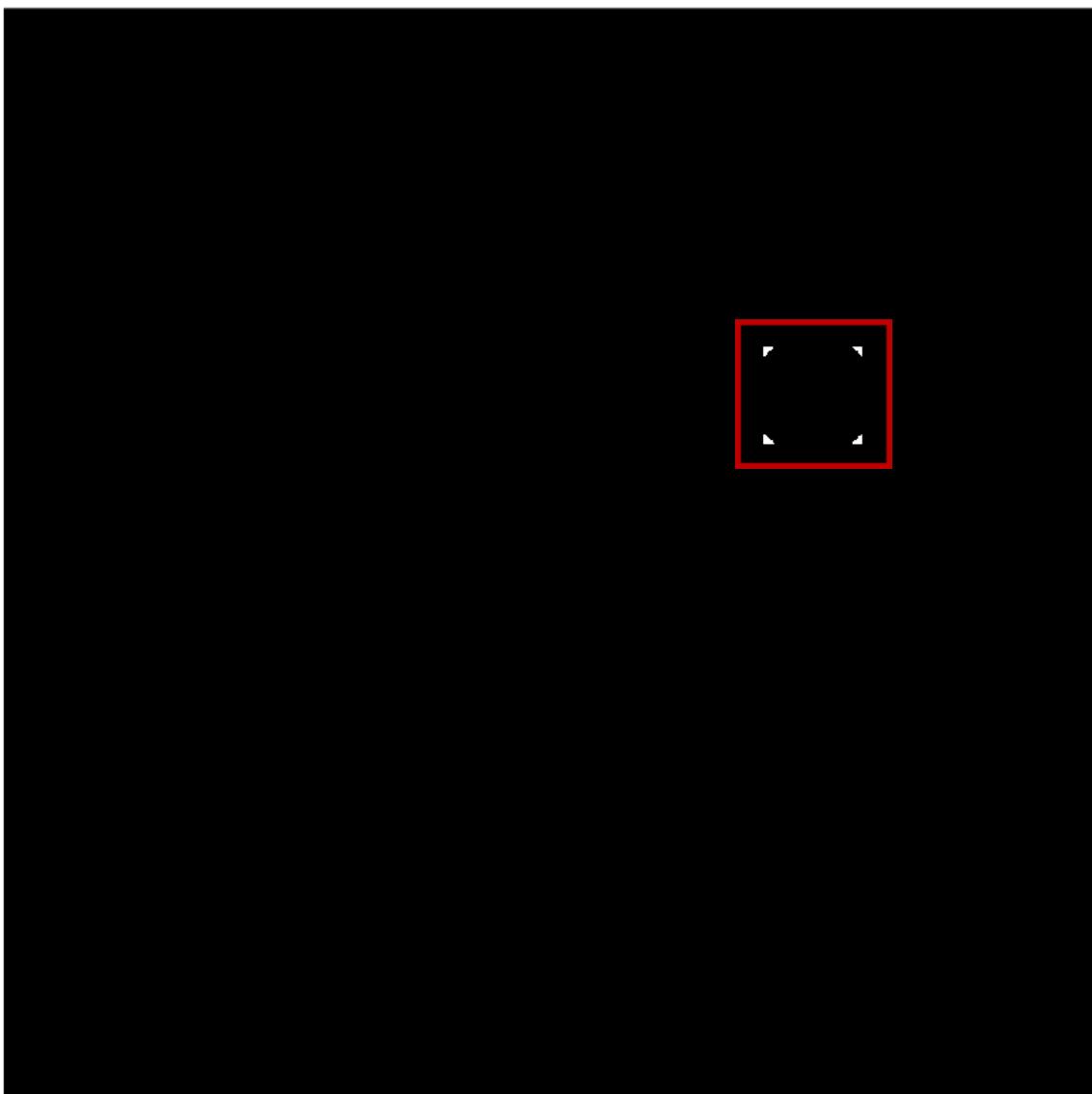
S'observa que només ens falta eliminar dos components per a obtenir els còrners desitjats. Malauradament, tots dos components son triangles de mida idèntica als còrners. Eliminarem el remarcat amb un requadre verd quedant-nos amb aquells píxels supervivents de dues obertures amb EE línia de direccions 45 i 135 graus i llargada 13 i 11 respectivament. Seguidament, reconstruirem la imatge resultant:



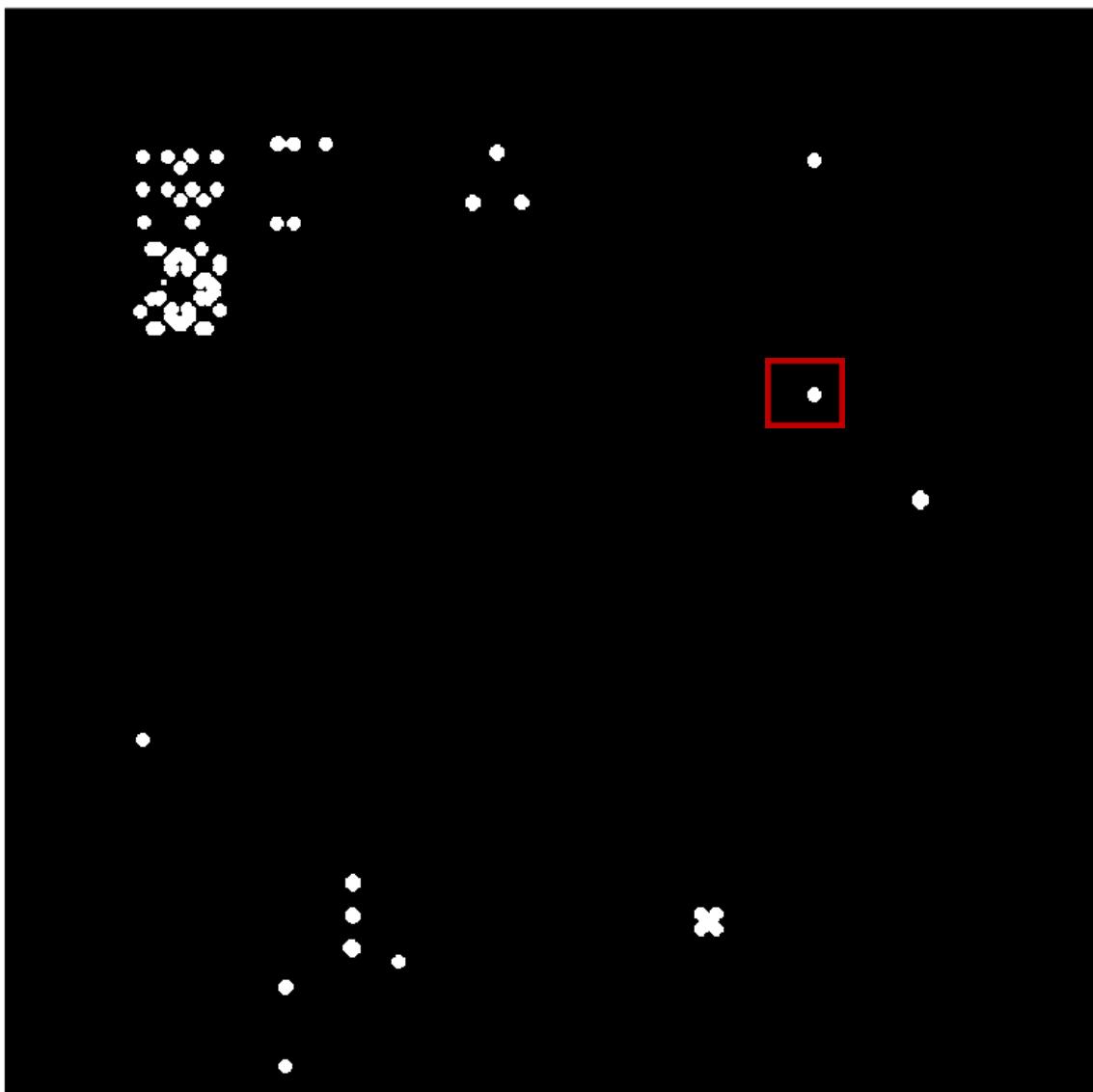
Per eliminar l'últim component (requadre verd), farem ús de la distància entre els còrners. Aquesta vegada, però, ens quedarem amb aquelles components que siguin properes. Per assolir l'objectiu, fem una dilatació de la imatge i obtenim això:



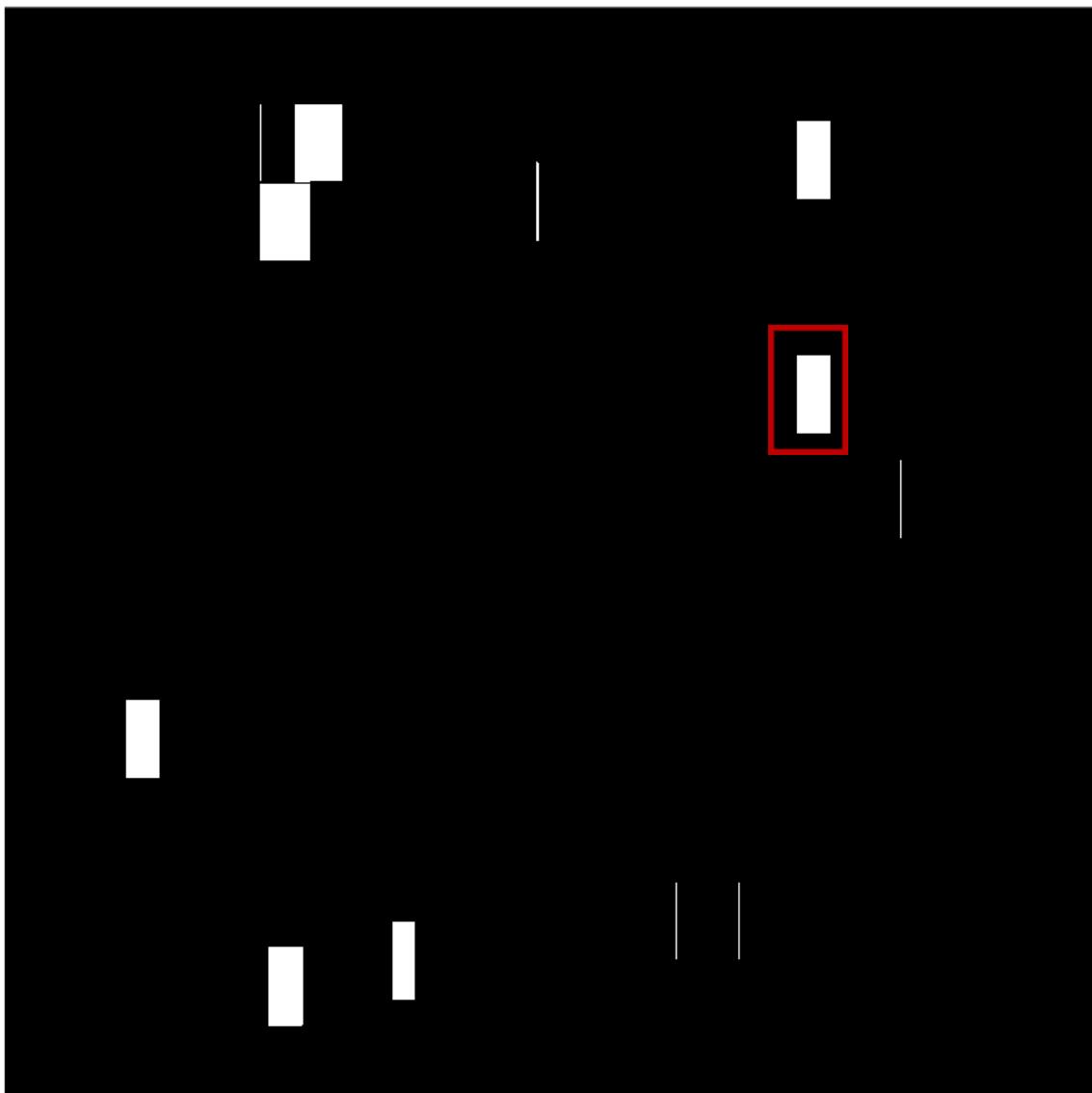
Ara només falta fer una obertura amb un EE quadrat suficientment gros perquè no càpiga al component que volem eliminar. Com que no podem fer una reconstrucció de la imatge per culpa d'haver unit els còrners, fem una operació AND entre la imatge dilatada i l'última reconstrucció que hem fet. El resultat és el següent:



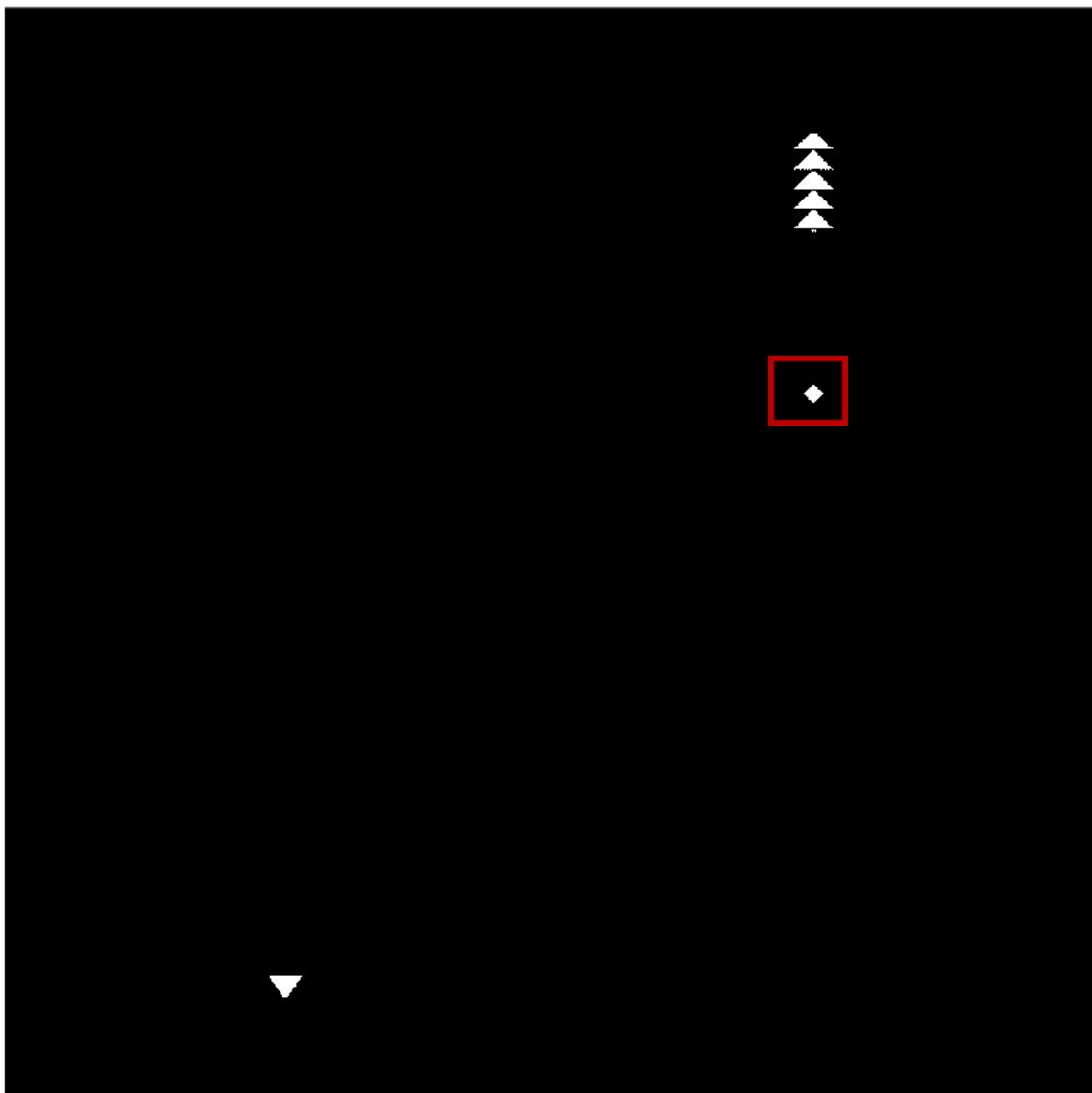
Per últim, hem d'obtenir el centre del patró. Començarem recuperant la imatge original i aplicant-li dues obertures amb EE diamant i obtenint-ne el residu. Així ens només ens quedem amb aquells element que tinguin una forma i mida molt similars al centre del patró. Després de netejar-ho amb una obertura, obtenim el següent:



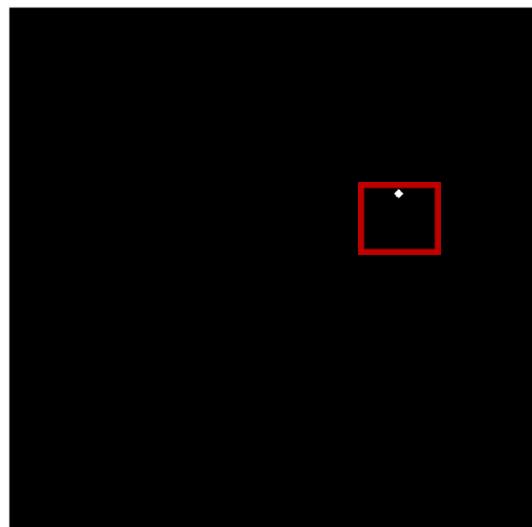
El pròxim pas serà eliminar aquells element que siguin molt propers uns dels altres, aprofitant que el centre del patró està força allunyat de qualsevol altre component. Per fer-ho, apliquem una dilatació i posteriorment filtrem els resultats calculant el residu entre dues obertures amb EE línia direcció 90 graus i llargada 71 i 72. Realitzats aquests passos, tenim el següent:



Com que tots els elements ara tenen forma de rectangle, els podem erosionar amb un EE rectangle de mides semblants al del centre del patró(requadre vermell). Després apliquem una reconstrucció. El resultat es mostra a continuació.



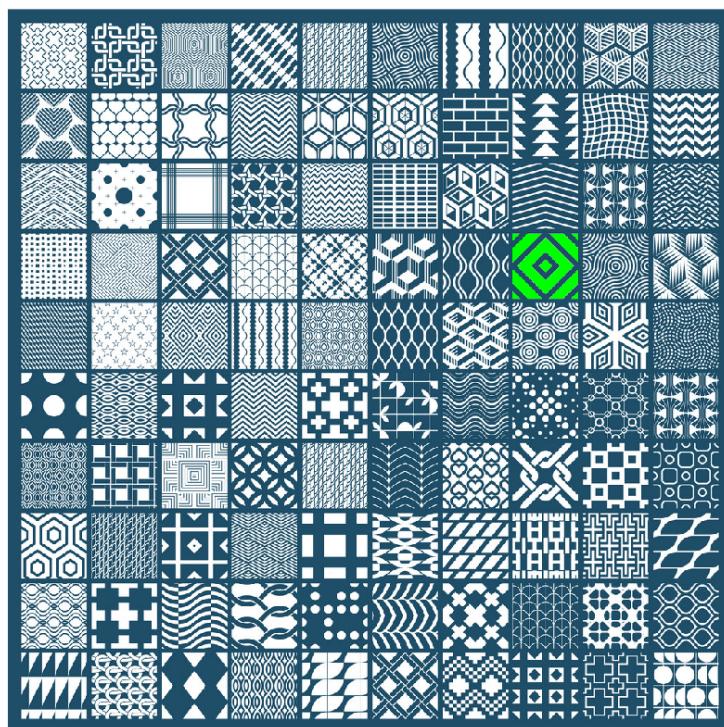
Finalment, calculem el residu de dues obertures amb EE disc i radis 6 i 7 i reconstruïm la imatge.



Per acabar, només queda ajuntar les tres parts per obtenir el patró complert:



Si ho sobreposem amb la imatge original, obtenim:



Codi patró 1:

```
%% Llegir imatge
I = imread('24486978.bmp');
I = rgb2gray(I);
figure
imshow(I);

%% Decisió de llindar
h = imhist(I);
figure
plot(h);

BW = I > 200;
figure
imshow(BW);

%% Patró 1

%% Marges
EE = strel('line', 30, 45);
OI = imerode(BW,EE);
EE2 = strel('square', 10);
OI2 = imerode(OI,EE2);

EE = strel('line', 30, 135);
OI4 = imerode(BW,EE);
EE2 = strel('square', 10);
OI3 = imerode(OI4,EE2);

MIX = or(OI2,OI3);
figure
montage({OI,OI4});
figure
imshow(MIX);

%%
EE3 = strel('square', 20);
OI4 = imopen(MIX, EE3);

MIX2 = and(MIX,not(OI4));
RC = imreconstruct(MIX2,BW);
figure
imshow(RC);

%% Còrnars

EE = strel('disk', 4);
OI = imopen(BW,EE);
OI = imreconstruct(OI,BW);

OI2 = and(BW, not(OI));
figure
imshow(OI2);
```

```

EE = strel('square', 6);
OI3 = imopen(OI2,EE);

EE = strel('square', 5);
OI4 = imopen(OI2,EE);

OI5 = and(OI4, not(OI3));
OI5 = imreconstruct(OI5, BW);

imshow(OI5);

%%
EE = strel('line', 9, 0);
OI6 = imopen(OI5,EE);

EE2 = strel('line', 9, 90);
OI7 = imopen(OI5,EE2);
figure
montage({OI6,OI7});

OI8 = and(OI6,OI7);
figure
imshow(OI8);

%%
EE3 = strel('disk', 22);
OI9 = imdilate(OI8,EE3);
figure
imshow(OI9);
%%
EE3 = strel('disk', 23);
OI10 = imopen(OI9,EE3);

EE4 = strel('disk', 22);
OI11 = imopen(OI9,EE4);

OI12=and(not(OI10),OI11);
figure

EE4 = strel('disk', 22);
OI13 = imopen(OI12,EE4);

figure
imshow(OI13);

%%
EE5 = strel('line', 43, 90);
OI14 = imopen(OI13,EE5);

EE6 = strel('line', 44, 90);
OI15 = imopen(OI13,EE6);

OI16 = and(not(OI15),OI14);

```

```

EE7 = strel('rectangle',[41 18]);
OI16 = imerode(OI16,EE7);

figure
imshow(OI16);

OI16 = imreconstruct(OI16,BW);

figure
imshow(OI16);

%%

EE8 = strel('line',11 , 135);
OI17 = imerode(OI16,EE8);

EE9 = strel('line',13 , 45);
OI18 = imerode(OI16,EE9);
OI19 = or(OI18,OI17);
OI19 = imreconstruct(OI19,BW);

figure
imshow(OI19);

%%

EE10 = strel('disk' , 60);
OI20 = imdilate(OI19,EE10);

figure
imshow(OI20);

EE11 = strel('square' , 100);
OI20 = imopen(OI20,EE11);

OI21 = and(OI20, OI19);
figure
imshow(OI21);

%%

RC2 = or(RC,OI21);
figure
imshow(RC2);

%%

EE12 = strel('diamond' , 8);
OI22 = imopen(BW,EE12);

EE13 = strel('diamond' , 9);
OI23 = imopen(BW,EE13);

OI24 = and(not(OI23),OI22);

figure
imshow(OI24);

%%

```

```

EE14 = strel('disk', 3);
OI25 = imopen(OI24,EE14);

figure
imshow(OI25);

EE15 = strel('disk', 30);
OI26 = imdilate(OI25,EE15);

EE16 = strel('line', 72, 90);
OI27 = imopen(OI26,EE16);

EE17 = strel('line', 71, 90);
OI28 = imopen(OI26,EE17);

OI28 = and(not(OI27),OI28);

figure
imshow(OI28);
%%
EE18 = strel('rectangle',[72 31]);
OI29 = imerode(OI28,EE18);
OI29 = imreconstruct(OI29,BW);

figure
imshow(OI29);
%%
EE19 = strel('disk', 7);
OI30 = imopen(OI29,EE19);

EE20 = strel('disk', 6);
OI31 = imopen(OI29,EE20);
OI31 = and(not(OI30), OI31);

OI31 = imopen(OI31,EE20);

%%
RC3 = imreconstruct(OI31,BW);

figure
imshow(RC3);

%%
FINAL = or(RC,RC2);
FINAL = or(FINAL,RC3);

figure
imshow(FINAL);

%%
IMG = imread('24486978.bmp');
IMG(:,:,1) = IMG(:,:,1) .* uint8(not(FINAL));
IMG(:,:,3) = IMG(:,:,3).* uint8(not(FINAL));
IMG(:,:,2) = IMG(:,:,2) + uint8(FINAL)*255;

```

```
figure  
imshow(IMG);
```

## PATRÓ 2:

A continuació fem el mateix amb un patró diferent. Tots els patrons, tot i que a primera vista semblin fàcils o difícils, tenen la seva dificultat, ja que reconeixer un patró només amb les estructures bàsiques de la comanda ‘strel’ és bastant complicat.

Hem acabat escollint el següent patró:

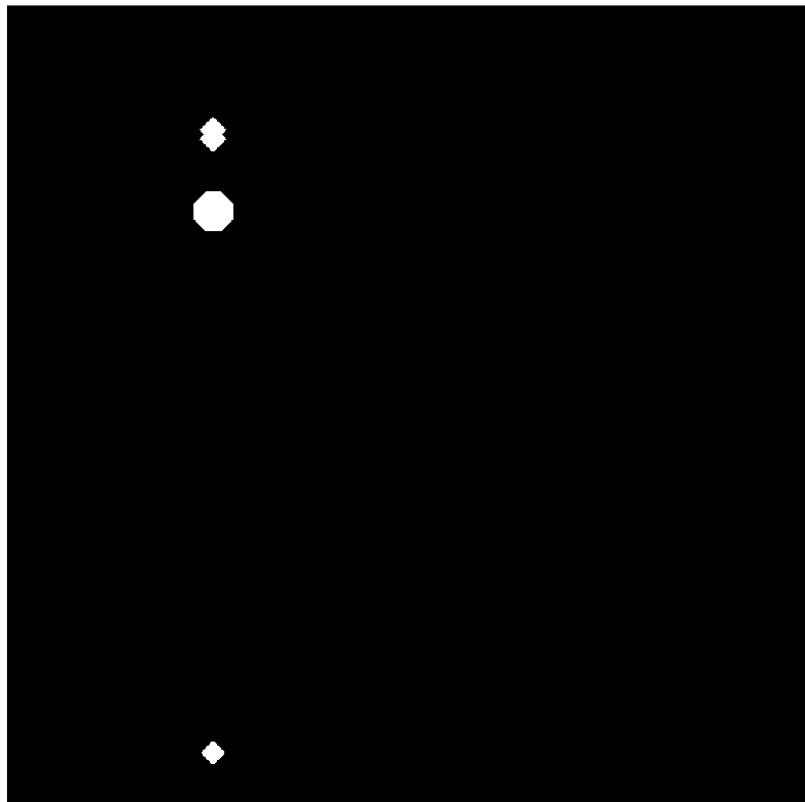


Primer de tot, aprofitem la binarització que hem fet anteriorment. Per tant, la variable BW representa la imatge de forma binaritzada que conté tots els patrons de la imatge.

El procés que realitzarem per identificar aquest patró serà similar al que hem seguit per identificar l'anterior.

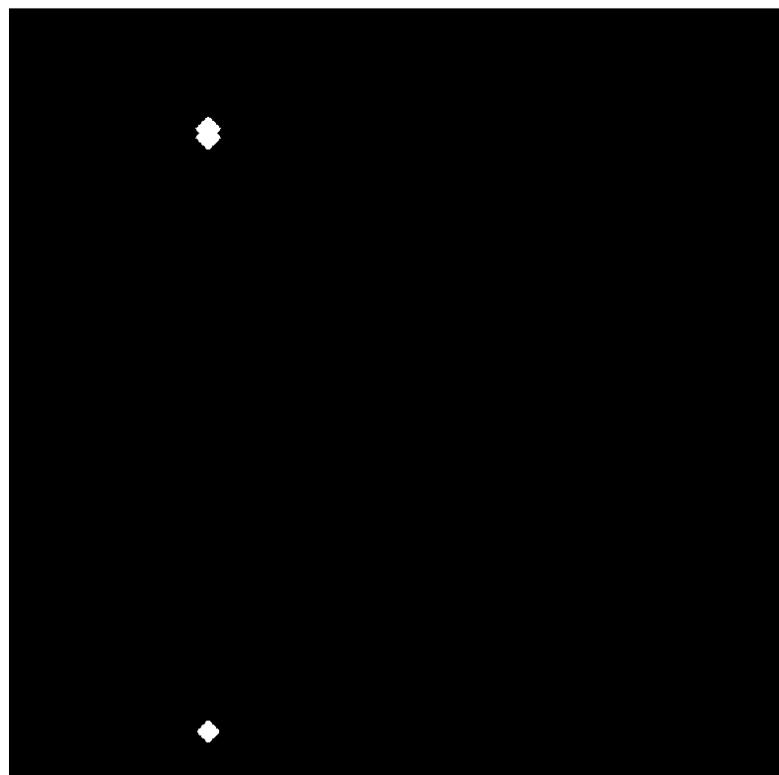
Observem que el patró està format per tres principals tipus de figures: el rombe central, quatre triangles grans (els que se situen a les parts centrals superior, esquerre, inferior, dreta), i quatre triangles petits (situats als exteriors). Per tant, podem identificar per separat aquests tipus de figures.

Comencem identificant primer el rombe central. Aquesta és la figura més fàcil d'identificar de tot el patró, ja que és la més gran de tot el patró. A més, la seva aparença és similar a la d'un diamant, per tant, podem aplicar l'operador d'obertura amb un component estructurant de diamant per eliminar tots aquells objectes petits de la imatge. Si fem això, ens queda:



Ens queda per tant, una imatge ja sorprendentment neta. Ara ens falta eliminar els espuris, és a dir, les dues figures superior de la imatge.

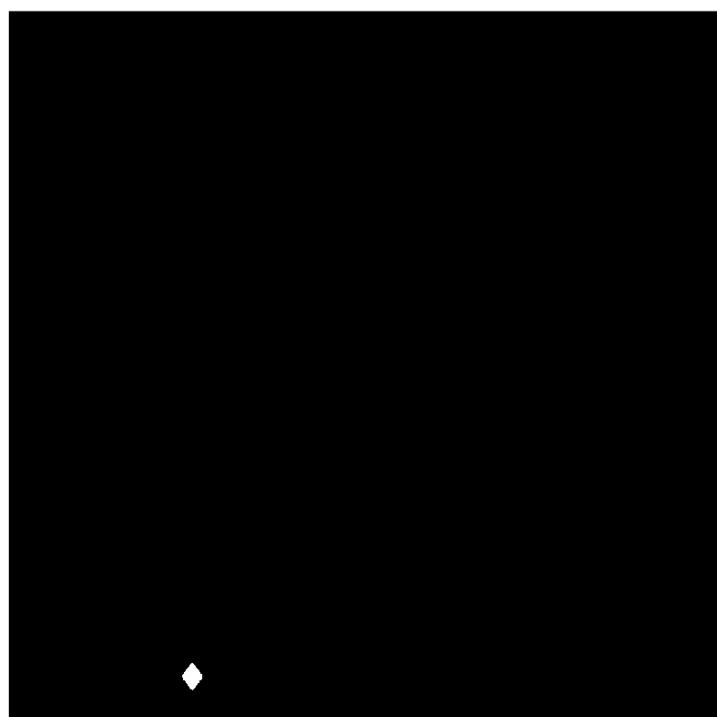
La figura més gran la podem eliminar fàcilment quedant-nos amb la imatge diferència entre la que tenim respecte aplicar-li l'operador d'obertura amb un component estructurant suficientment gran:



Per eliminar l'altre figura, o podem fer de forma similar amb un element estructurant adient:



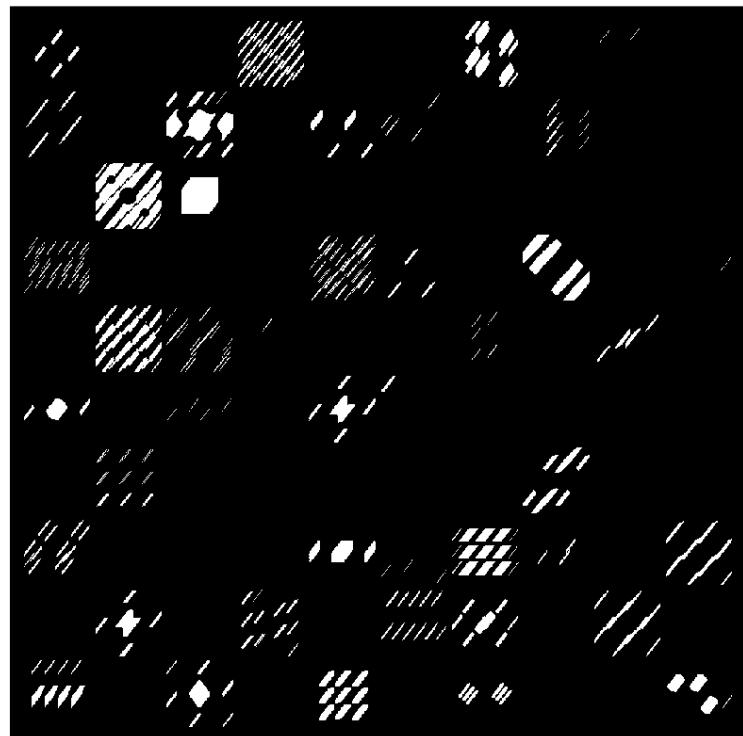
Ara ja hi podem aplicar la reconstrucció respecte la imatge original, de forma que ja haurem reconegut la figura central del patró:



Ara anem a identificar els triangles grans de la imatge.

Si ens fixem, en cadascun d'aquests quatre triangles hi podem identificar línies de 53 i 125 graus.

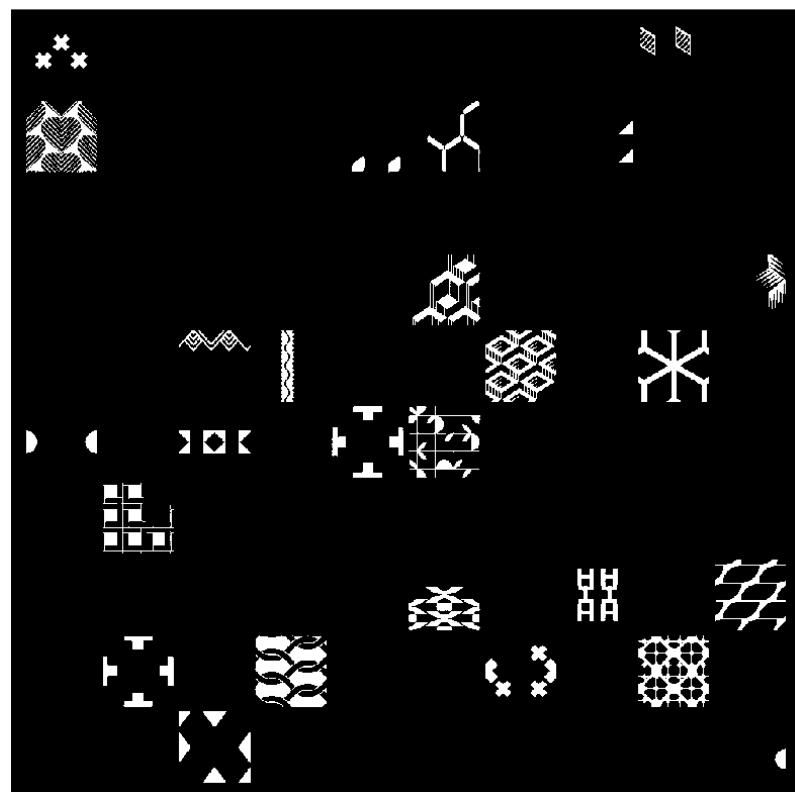
Per tant, el primer que farem serà filtrar figures en la imatge que siguin similars o més grans a una línia de 53 graus i 22 pixels amb una obertura:



Observem que la imatge segueix no essent neta. Intentem eliminar la resta d'elements amb un seguit d'operacions de tancament i obertura amb diversos elements estructurants, amb l'objectiu d'eliminar per una part aquells elements que siguin més petits que la mida dels que representen els nostres 4 triangles, i d'altre banda per tancar forats entre elements que estiguin molt junts entre ells, per tal que aquests formin una figura més grossa de la que tenen aquelles que identifiquen els nostres 4 triangles, i amb el residu de l'operació de l'obertura eliminar aquestes figures més grans. Finalment arribem al següent resultat:

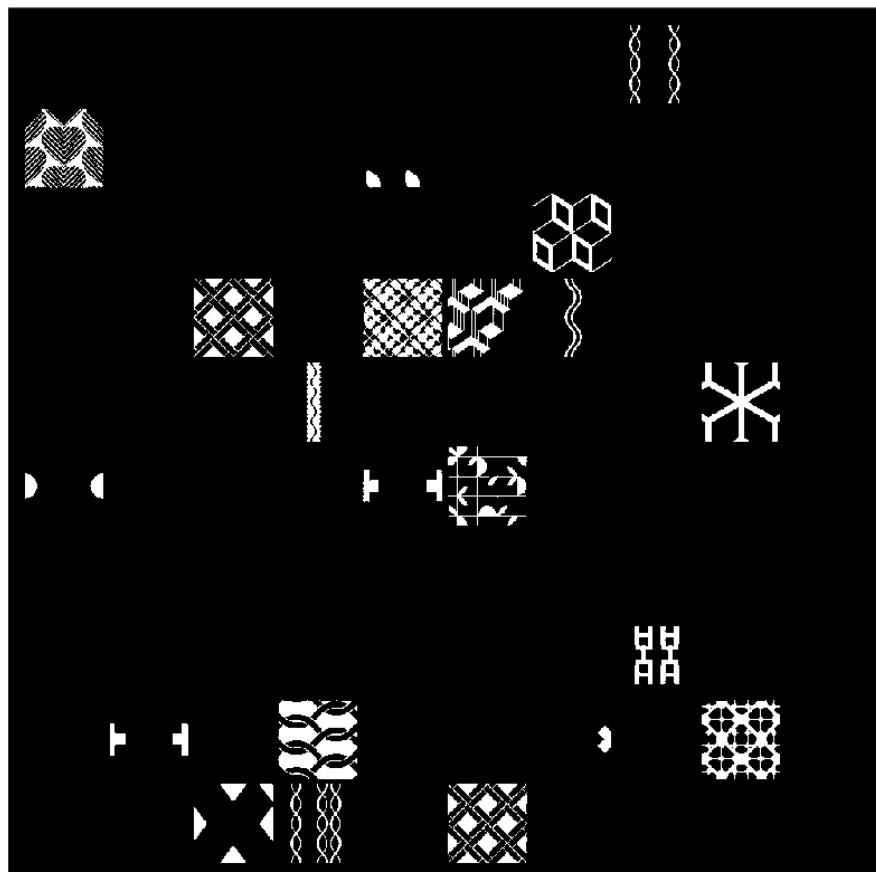


La imatge és més neta, però encara ens falta progrés. Fem una reconstrucció d'aquesta imatge amb l'original:

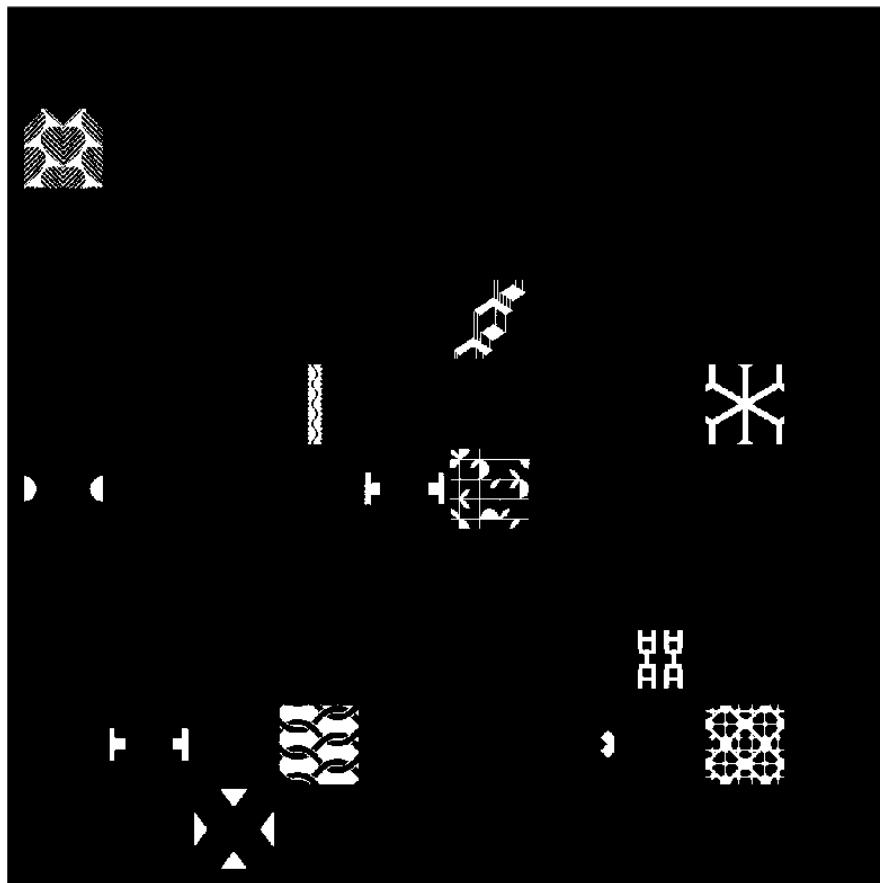


Tal com hem dit abans, els triangles també es veuen identificats per unes línies de 125 graus. Per tant, tot el que hem fet fins ara també ho podem fer sobre un filtre principal d'obertura amb un element estructurant de línia de 125 graus, en comptes de 53.

Si ho fem, i també fem la reconstrucció de les formes que ens queden, output:



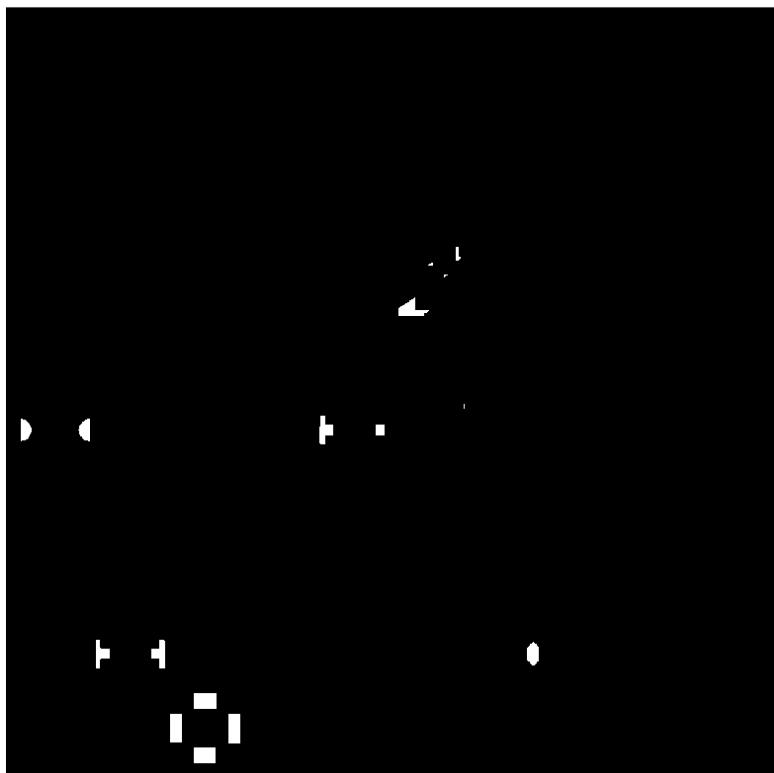
Ara, el que podem fer és la intersecció entre aquestes dues reconstruccions que hem fet, per tal que ens quedi una imatge encara més neta. Ens queda això:



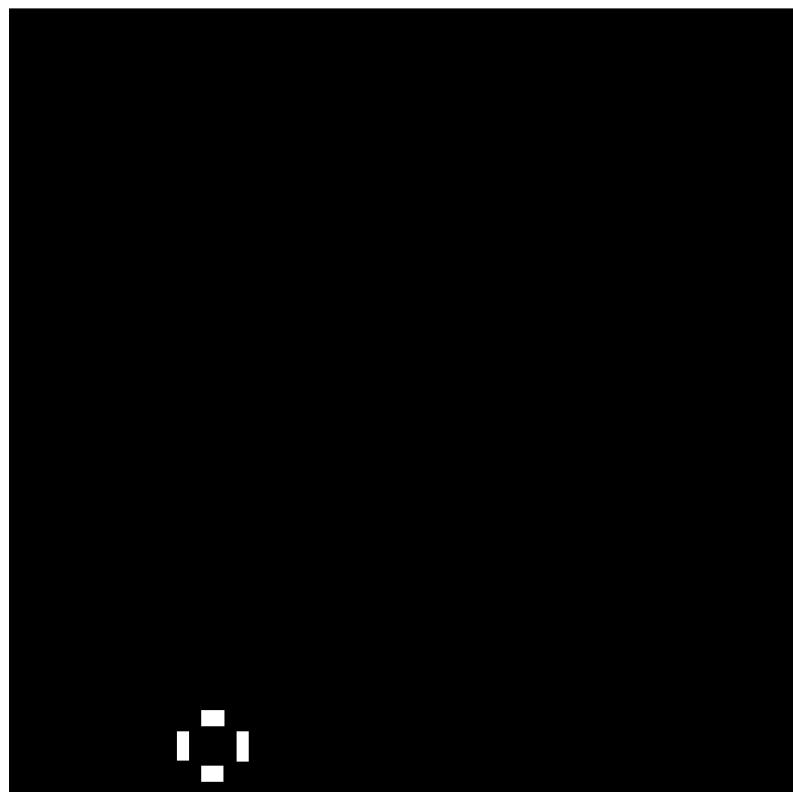
Si observem les formes que no volem, podem veure que si fem una operació de tancament sobre un element estructurant adient les podrem unificar de forma que és quedin en figures uniformes i per tant més grans que les que volem:



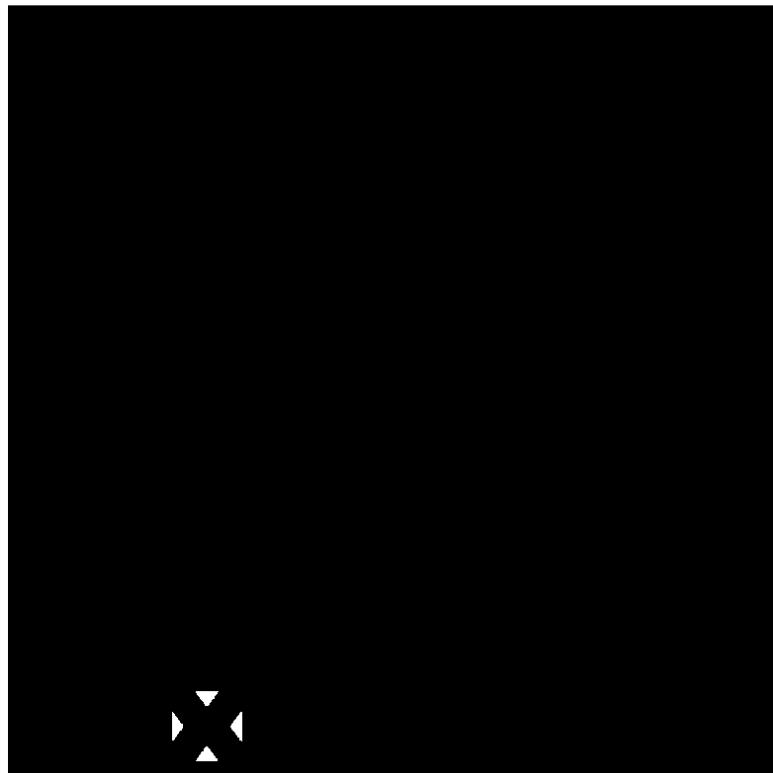
Si ara ens quedem amb el residu respecte el que ens queda si apliquem una operació d'obertura, eliminem els objectes grans:



Ara les figures que volem són totes més grans que les altres. Per tant, si apliquem operacions d'obertura:

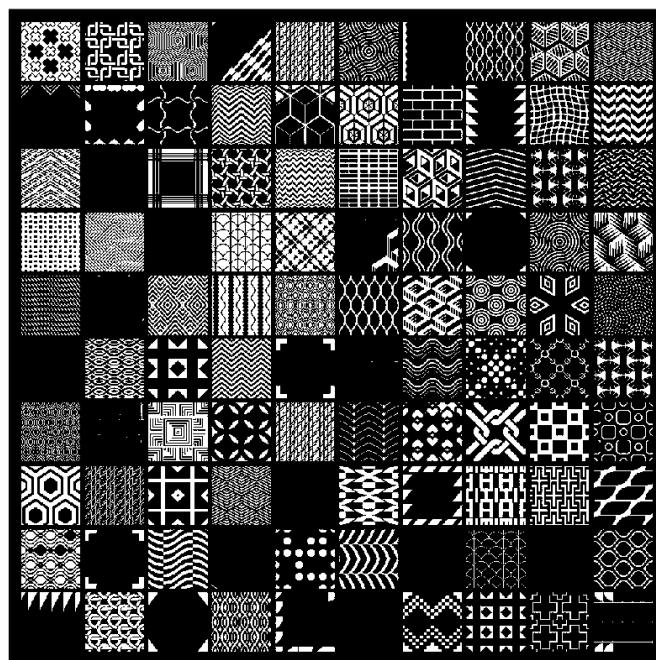


Reconstruim aquesta imatge amb la orginal i ja tenim reconeguts els triangles grans:

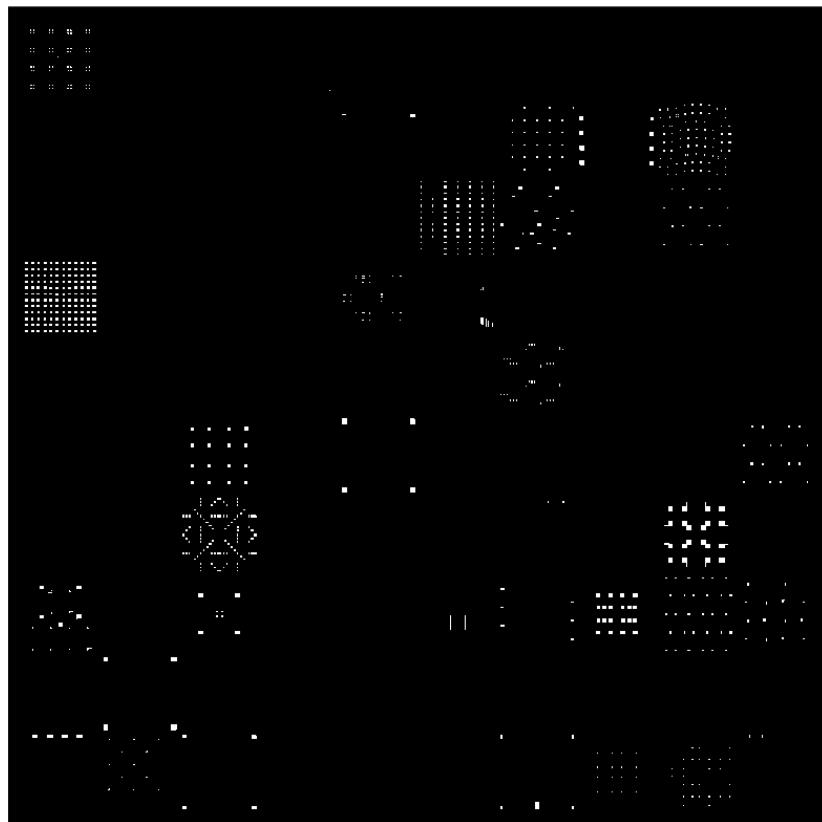


Anem ara a per els 4 triangles més petits:

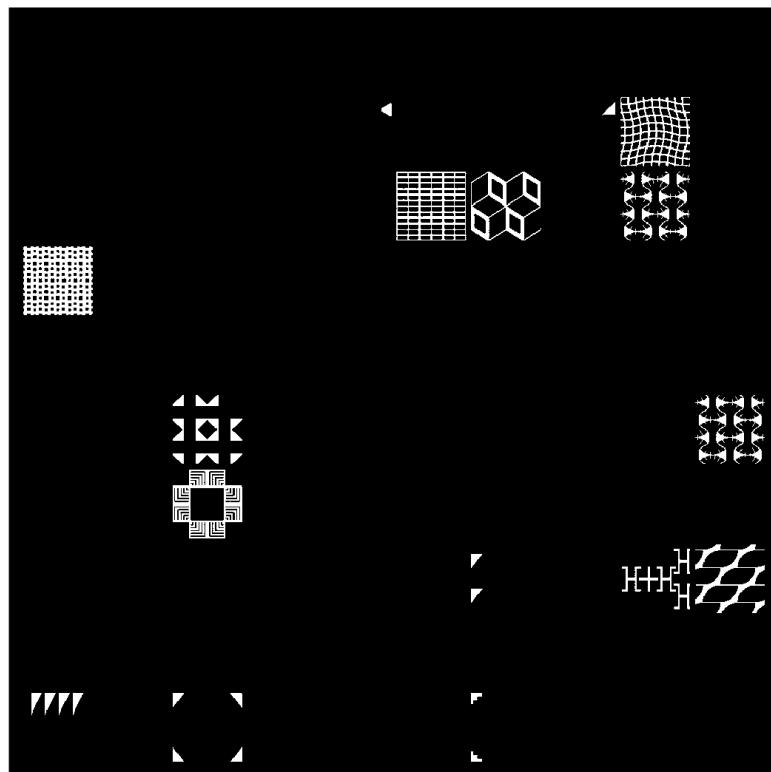
Al tractar-se de figures petites, podem començar realitzant un seguit d'operacions d'obertura amb diferents elements estructurants grans, reconstruir la imatge resultant i quedar-nos amb la imatge diferència respecte la imatge original, per eliminar totes aquelles figures que siguin de mida més gran:



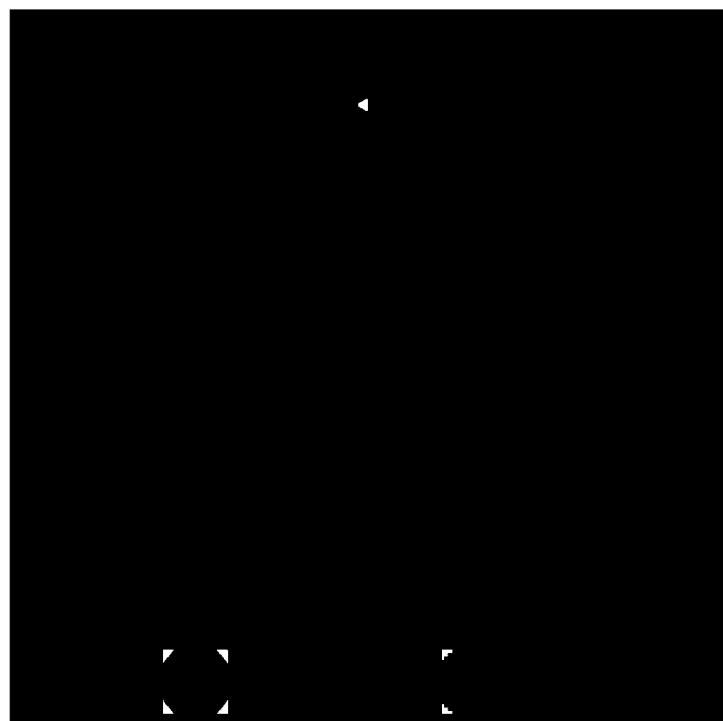
Això ens ha descalificat poques figures. El que podem fer és, aprofitant que a l'extrem dels triangles triangles intersequen línies de 90 i 0 graus, aplicar dos operaicons d'obertura speradament, una amb un element estructurant que sigui una línia de 0 graus, i un altre amb un element estructurant que sigui una línia de 90 graus, i calcular la intersecció de les dues imatges que queden:



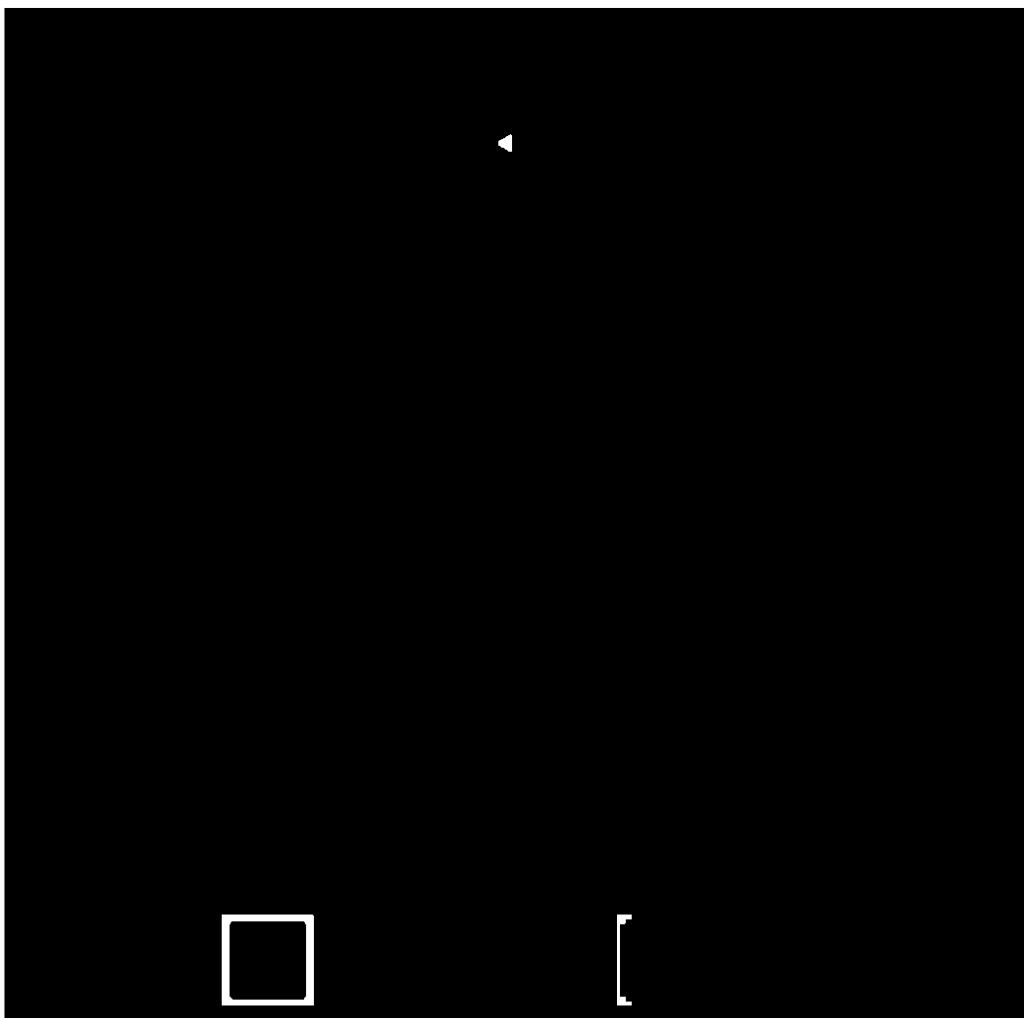
A continuació, podem aplicar l'operador open per a eliminar moltes de les formes que tenen una mida molt petita, i reconstruim també la imatge per veure com podem eliminar la resta:



Ara tenim un cas similar a un anterior, en que moltes de les formes que no volem estan juntes entre elles. Si apliquem l'operador de tancament per unificar aquestes formes, i ens quedem amb la diferència de l'aplicació de l'operador d'obertura respecte la imatge que tenim, eliminarem totes aquestes formes i tindrem una imatge encara millor:



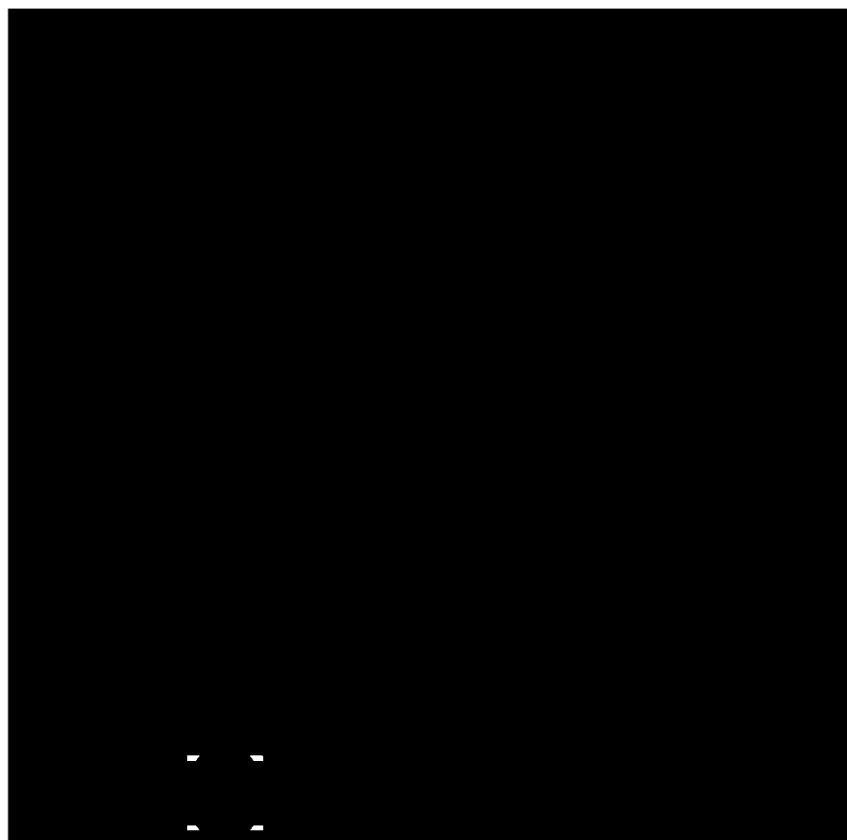
Per eliminar la resta de figures que no volem, primer podem aplicar una operació de tancament que formi un quadrat sobre les 4 figures que reconeixen els nostres triangles:



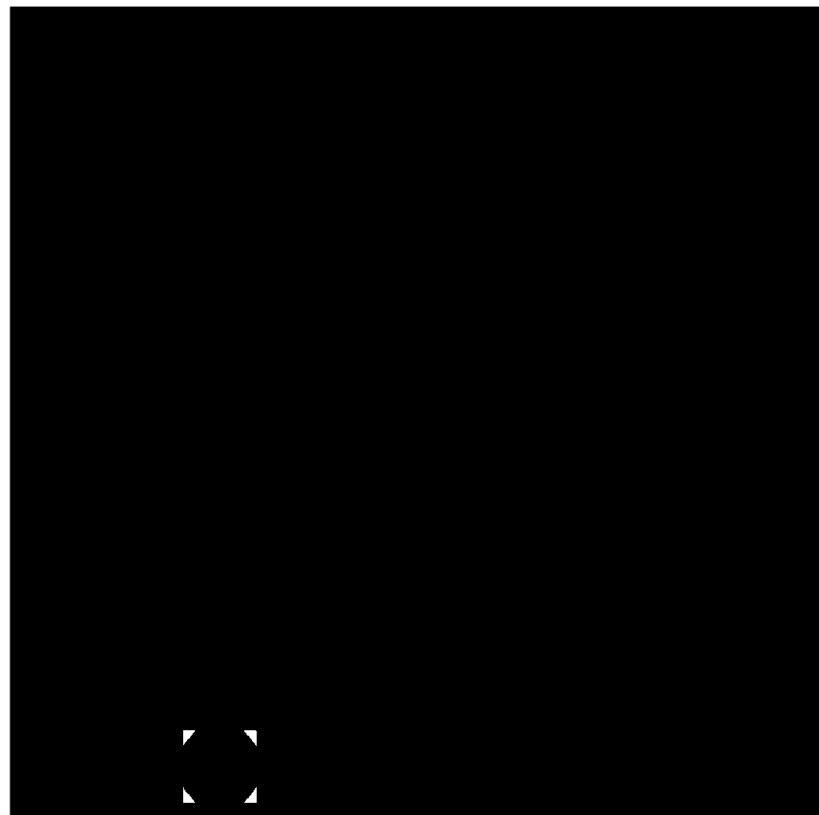
D'aquesta manera ara si apliquem una operació d'obertura amb un element línia de grau 0 i de 20 pixels longitud, només acabaràn sobrevisquer dues lines en el nostre patró:



Ara si fem una intersecció entre aquesta imatge i aquella en que s'identificaven les quatre formes que reconeixien el nostre triangle, obtenim:



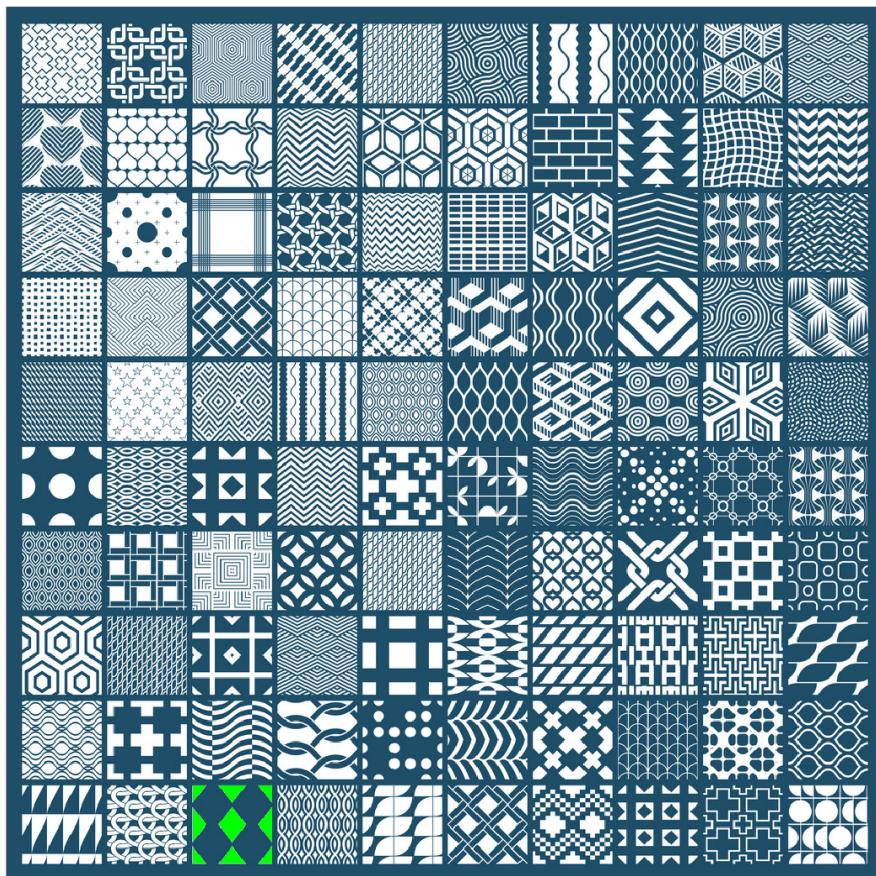
Reconstruim la imatge i ja tenim els triangles:



Si fem la unió de totes les formes del patró:



Si fem un overlay ja tenim el patró identificat:



El codi:

```
clc
clear
close all

%% Llegir imatge
I = imread('24486978.bmp');
I = rgb2gray(I);
%imshow(I);

%% Decisió de llindar
h = imhist(I);
%figure
%plot(h);
```

```

BW = I > 200;
%figure
imshow(BW);

%% Patró 2

% Diagonal central
EE = strel('diamond', 15);
A = imopen(BW, EE);
%figure
imshow(A)

% Eliminar petits espuris
EE2 = strel('diamond', 16);
A = and(A, not(imopen(A, EE2)));
%figure
imshow(A)
A = imopen(A, EE);
%figure
imshow(A)

EE3 = strel('square', 16);
%A = imopen(A, EE3);
A = and(A, not(imopen(A, EE3)));
%figure
imshow(A)

EE4 = strel('diamond', 10);
A = imopen(A, EE4);
%figure
imshow(A)

A = imreconstruct(A, BW);
%figure
imshow(A)

%% Triangles grans

% 55 graus

EE5 = strel('line', 22, 53);
B = imopen(BW, EE5);
%figure
imshow(B)

EE6 = strel('square', 10);
B = imclose(B, EE6);
%figure
imshow(B)

EE8 = strel('disk', 5, 8);
B = imclose(B, EE8);
%figure
imshow(B)

EE7 = strel('square', 5);
B = and(B, not(imopen(B, EE7)));
%figure
imshow(B);

```

```

EE9 = strel('line', 22, 53);
B = imopen(B, EE9);
%figure
%imshow(B);

EE10 = strel('square', 38);
B = imclose(B, EE10);
%figure
%imshow(B)

EE12 = strel('disk', 5, 8);
B = and(B, not(imopen(B, EE12)));
%figure
%imshow(B);

B = imreconstruct(B, BW);
%figure
%imshow(B)

% 125 graus

EE15 = strel('line', 22, 125);
D = imopen(BW, EE15);
%figure
%imshow(D)

EE16 = strel('square', 10);
D = imclose(D, EE16);
%figure
%imshow(D)

EE18 = strel('disk', 5, 8);
D = imclose(D, EE18);
%figure
%imshow(D)

EE17 = strel('square', 5);
D = and(D, not(imopen(D, EE17)));
%figure
%imshow(D);

EE19 = strel('line', 23, 125);
D = imopen(D, EE19);
%figure
%imshow(D);

EE20 = strel('square', 30);
D = imclose(D, EE20);
%figure
%imshow(D)

EE22 = strel('disk', 1, 8);
D = and(D, not(imopen(D, EE22)));
%figure
%imshow(D);

```

```

%% interseccio 55 i 120 graus

D= imreconstruct(D, BW);
%figure
%imshow(D)
F = and(B, D);

%figure
%imshow(F)

%% eliminar espuris pels triangles grans

EE25 = strel('square', 40);
F = imclose(F, EE25);
%figure
%imshow(F)

EE26 = strel('square', 21);
F = and(F, not(imopen(F, EE26)));
%figure
%imshow(F)

EE27 = strel('rectangle', [40, 5]);
F = and(F, not(imopen(F, EE27)));
%figure
%imshow(F)

EE28 = strel('rectangle', [38, 5]);
D = imopen(F, EE28);
%figure
%imshow(D)

EE29 = strel('rectangle', [10, 28]);
F = imopen(F, EE29);
%figure
%imshow(F)

F = or(F, D);
%figure
%imshow(F)

F = imreconstruct(F, BW);
%figure
%imshow(F)

%% els triangles petits

EE31 = strel('disk', 6, 8);
W = imopen(BW, EE31);
%figure
%imshow(W)

EE31 = strel('square', 10);
E = imopen(BW, EE31);
%figure

```

```

%imshow(E)

EE31 = strel('diamond', 6);
R = imopen(BW, EE31);
%figure
%imshow(R)

T = or(W, E);
T = or(T, R);
%figure
%imshow(T)

T = and(BW, not(imreconstruct(T, BW)));
%figure
%imshow(T)

EE30 = strel('line', 13, 90);
W = imopen(T, EE30);
%figure
%imshow(W)

EE31 = strel('line', 13, 0);
D = imopen(T, EE31);
%figure
%imshow(D)

T = and(D, W);
%figure
%imshow(T)

EE32 = strel('square', 3);
T = imopen(T, EE32);
%figure
%imshow(T)

EE33 = strel('line', 6, 45);
D = imopen(T, EE33);
%figure
%imshow(D)

T = imreconstruct(T, BW);
D = imreconstruct(D, BW);
T = and(T, not(D));
%figure
%imshow(T)

EE34 = strel('square', 50);
T = imclose(T, EE34);
%figure
%imshow(T)

EE35 = strel('square', 10);
D = imopen(T, EE35);
%figure
%imshow(D)

```

```

T = imreconstruct(T, BW);
D = imreconstruct(D, BW);
T = and(T, not(D));
%figure
%imshow(T)

EE36 = strel('square', 70);
D = imclose(T, EE36);
%figure
%imshow(D)

EE37 = strel('line', 20, 0);
D = imopen(D, EE37);
%figure
%imshow(D)

T = and(T, D);
%figure
%imshow(T)

T = imreconstruct(T, BW);
%figure
%imshow(T)

FINAL = or(A, F);
FINAL = or(FINAL, T);
%figure
%imshow(FINAL)

IMG = imread('24486978.bmp');
IMG(:,:,1) = IMG(:,:,1) .* uint8(not(FINAL));
IMG(:,:,3) = IMG(:,:,3).* uint8(not(FINAL));
IMG(:,:,2) = IMG(:,:,2) + uint8(FINAL)*255;

figure
imshow(IMG);

```