

## Sessió 3

Primer farem les lectures de les imatges i les passarem a escala de grisos.

```
caracol1original = imread('_61A5845.jpg');  
caracol2original = imread('_61A5855.jpg');  
caracol3original = imread('_61A5861.jpg');  
  
montage({caracol1original, caracol2original, caracol3original})
```



```
caracol1 = rgb2gray((imread('_61A5845.jpg')));  
caracol2 = rgb2gray((imread('_61A5855.jpg')));  
caracol3 = rgb2gray((imread('_61A5861.jpg')));  
  
montage({caracol1, caracol2, caracol3})
```



Creem dos filtres de Sobel, un horitzontal i un vertical.

```
Sv = [-1, 0, 1; -2, 0, 2; -1, 0, 1];
Sh = [-1, -2, -1; 0, 0, 0; 1, 2, 1];
```

Apliquem aquests filtres a cada imatge.

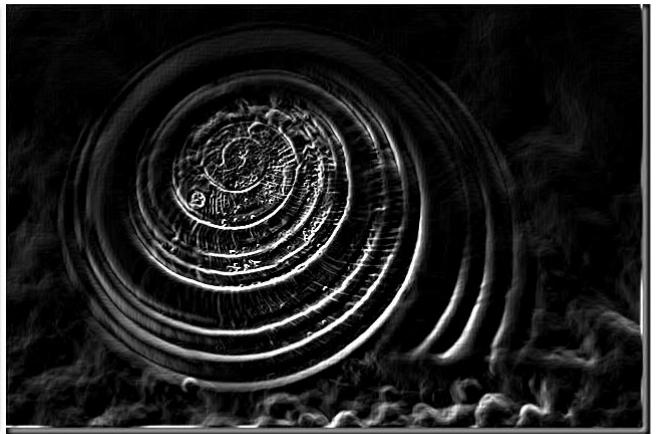
```
caracol1sv = imfilter(caracol1, Sv);
caracol1sh = imfilter(caracol1, Sh);
caracol1s = abs(caracol1sh) + abs(caracol1sv);

caracol2sv = imfilter(caracol2, Sv);
caracol2sh = imfilter(caracol2, Sh);
caracol2s = abs(caracol2sh) + abs(caracol2sv);

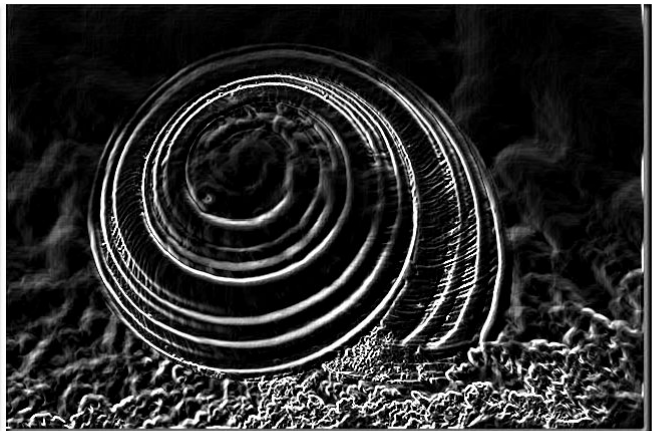
caracol3sv = imfilter(caracol3, Sv);
caracol3sh = imfilter(caracol3, Sh);
caracol3s = abs(caracol3sh) + abs(caracol3sv);
```

Visualitzem els contorns.

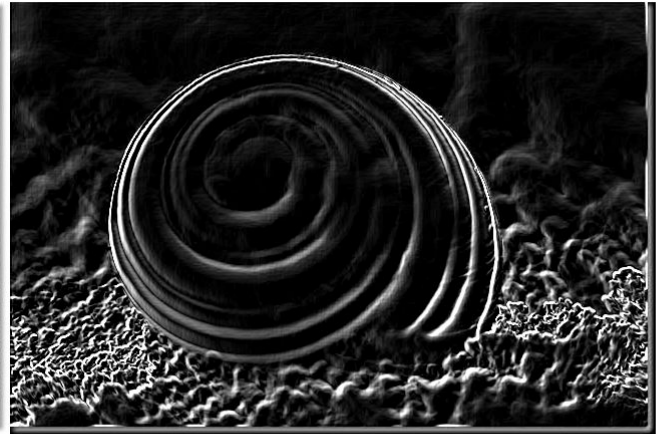
```
montage({caracol1, caracol1s})
```



```
montage({caracol2, caracol2s})
```



```
montage({caracol3, caracol3s})
```



Fem un recorregut de la imatge i fem que els valors del mig de la imatge tinguin un pes més gran que els dels extrems.

```
caracol1size = size(caracol1s);
multiplicador = 0;
m1 = uint64(0);
for i = 1:(caracol1size(1)/2)
    for j = 1:(caracol1size(2)/2)
        m1 = m1 + uint64(caracol1s(i, j)) * multiplicador;
        m1 = m1 + uint64(caracol1s(i, caracol1size(2) - j)) * multiplicador;
        m1 = m1 + uint64(caracol1s(caracol1size(1) - i, j)) * multiplicador;
        m1 = m1 + uint64(caracol1s(caracol1size(1) - i, caracol1size(2) - j)) * multiplicador;
    end
    multiplicador = multiplicador + 1;
end

caracol3size = size(caracol3s);
multiplicador = 0;
m3 = uint64(0);
for i = 1:(caracol3size(1)/2)
    for j = 1:(caracol3size(2)/2)
        m3 = m3 + uint64(caracol3s(i, j)) * multiplicador;
        m3 = m3 + uint64(caracol3s(i, caracol3size(2) - j)) * multiplicador;
        m3 = m3 + uint64(caracol3s(caracol3size(1) - i, j)) * multiplicador;
        m3 = m3 + uint64(caracol3s(caracol3size(1) - i, caracol3size(2) - j)) * multiplicador;
    end
    multiplicador = multiplicador + 1;
end

caracol2size = size(caracol2s);
multiplicador = 0;
m2 = uint64(0);
for i = 1:(caracol2size(1)/2)
    for j = 1:(caracol2size(2)/2)
        m2 = m2 + uint64(caracol2s(i, j)) * multiplicador;
```

```

        m2 = m2 + uint64(caracol2s(i, caracol2size(2) - j)) * multiplicador;
        m2 = m2 + uint64(caracol2s(caracol2size(1) - i, j)) * multiplicador;
        m2 = m2 + uint64(caracol2s(caracol2size(1) - i, caracol2size(2) - j)) * multiplicador;
    end
    multiplicador = multiplicador + 1;
end

```

Imprimim els valors.

```
disp(m1)
```

886498615

```
disp(m2)
```

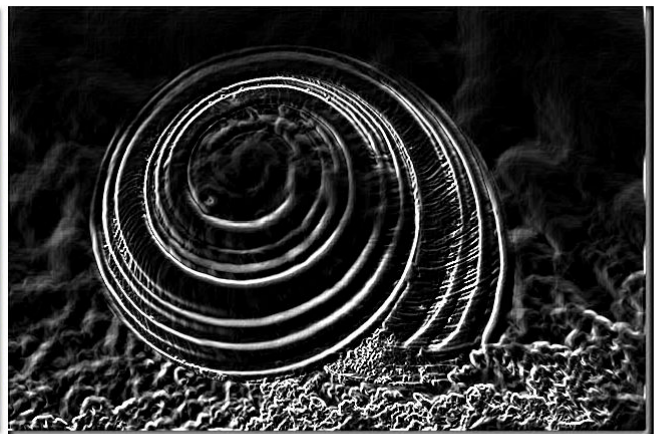
1182018262

```
disp(m3)
```

1055501862

Com podem veure la segona imatge té un valor més alt ja que els contorns que té es veuen millor.

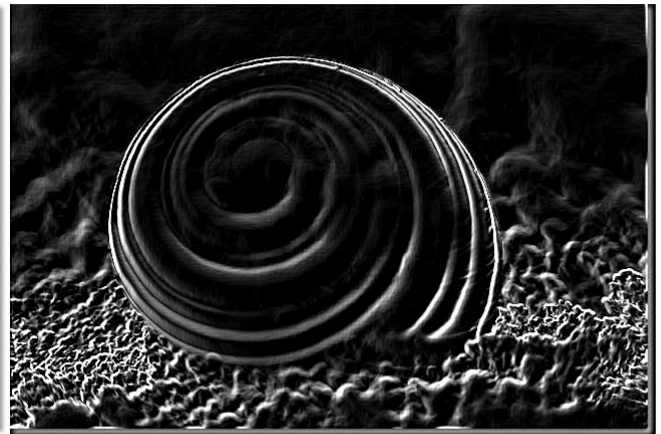
```
montage({caracol2original, caracol2s})
```



La segona seria la tercera imatge.

```
montage({caracol3original, caracol3s})
```





Com a última imatge la primera.

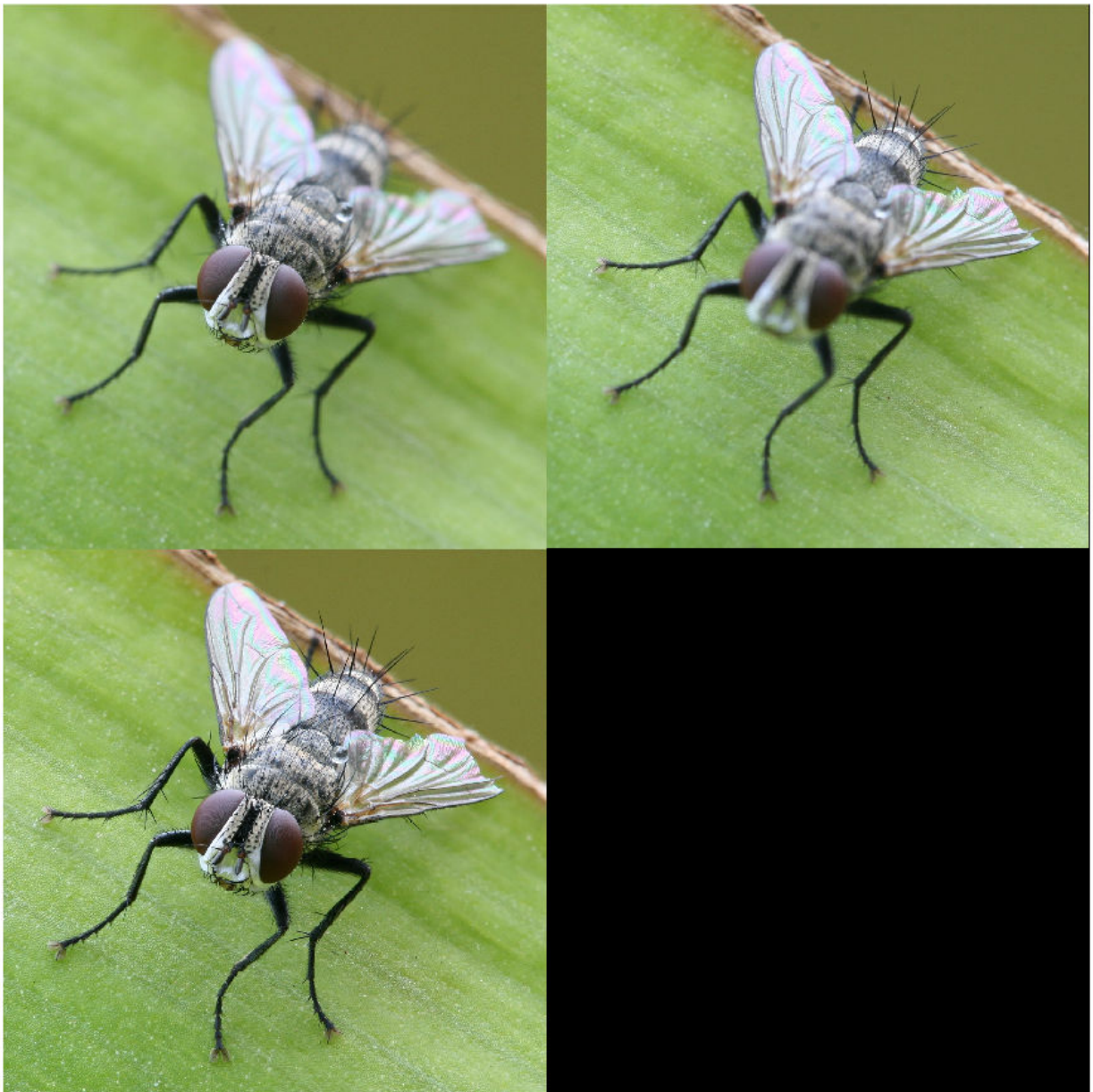
```
montage({caracol1original, caracol1s})
```



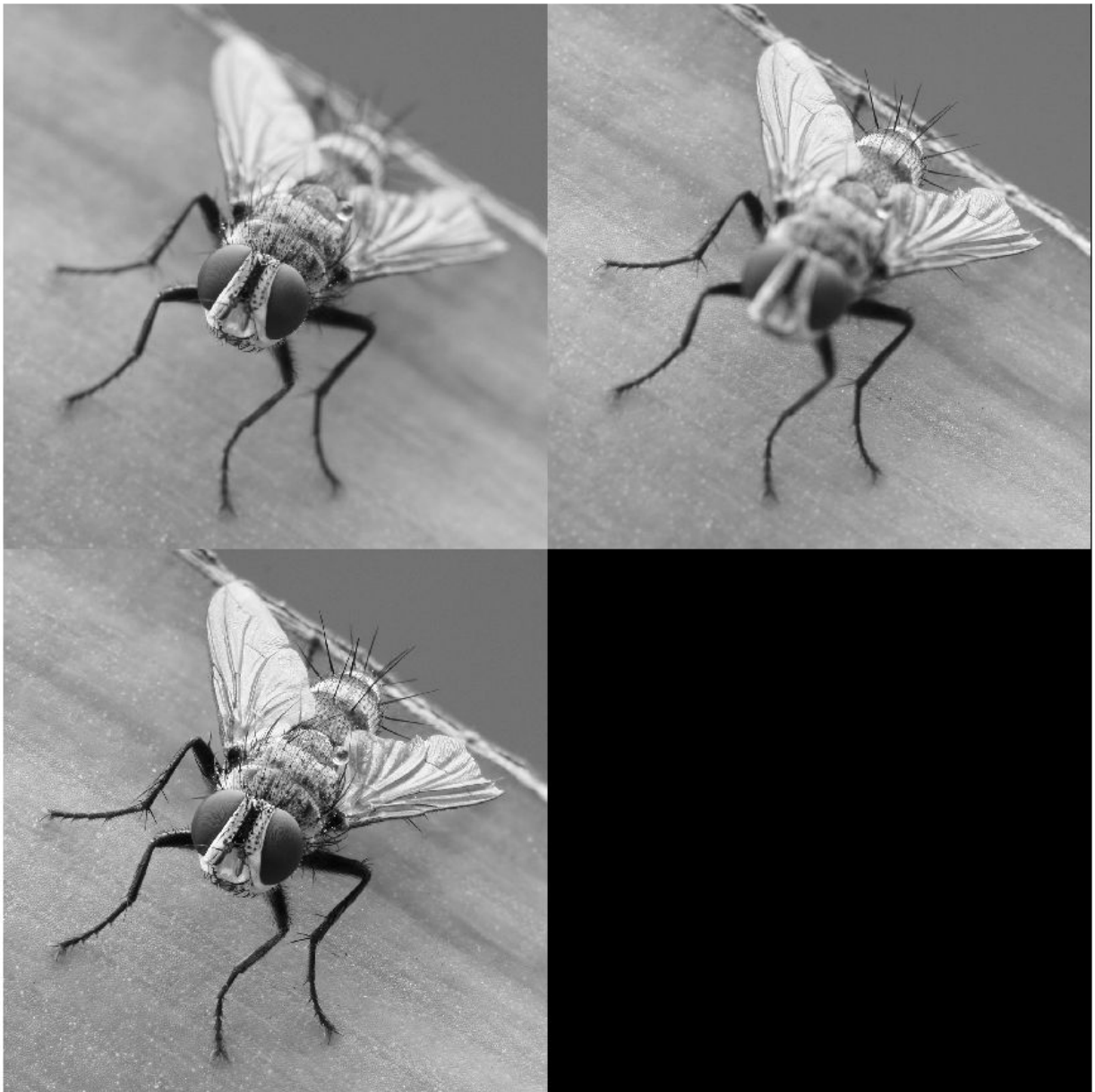
El resultat és bastant coherent amb la imatge guanyadora, el conjunt de parts de la imatge que està millor enfocada és la que ha obtingut més puntuació amb la heurística escollida.

Proba amb una mosca

```
caracol1original = imread('1.jpg');  
caracol2original = imread('2.jpg');  
caracol3original = imread('3.jpg');  
  
montage({caracol1original, caracol2original, caracol3original})
```



```
caracol1 = rgb2gray((imread('1.jpg')));  
caracol2 = rgb2gray((imread('2.jpg')));  
caracol3 = rgb2gray((imread('3.jpg')));  
  
montage({caracol1, caracol2, caracol3})
```



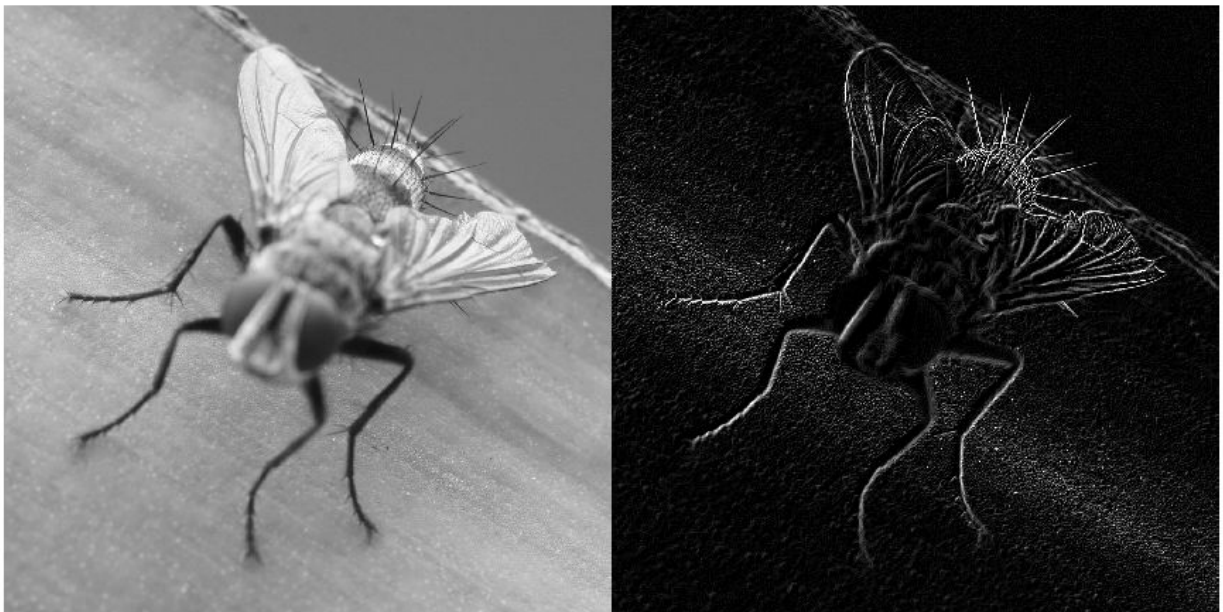
```
Sv = [-1, 0, 1; -2, 0, 2; -1, 0, 1];  
Sh = [-1, -2, -1; 0, 0, 0; 1, 2, 1];  
  
caracol1sv = imfilter(caracol1, Sv);  
caracol1sh = imfilter(caracol1, Sh);  
caracol1s = abs(caracol1sh) + abs(caracol1sv);  
  
caracol2sv = imfilter(caracol2, Sv);  
caracol2sh = imfilter(caracol2, Sh);  
caracol2s = abs(caracol2sh) + abs(caracol2sv);
```



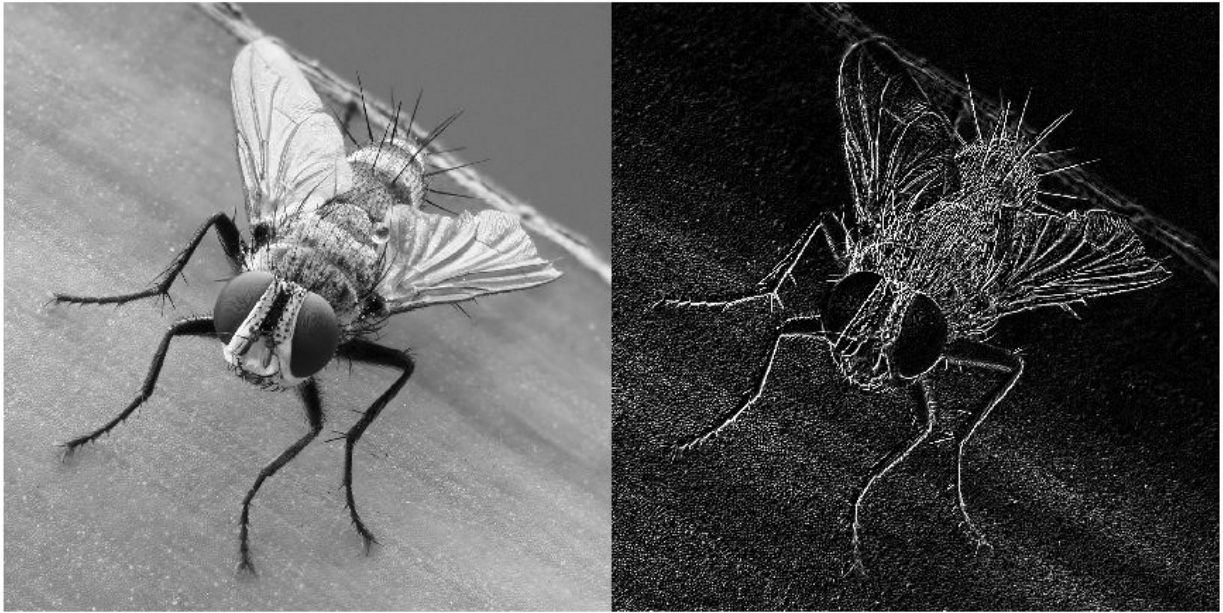
```
caracol3sv = imfilter(caracol3, Sv);  
caracol3sh = imfilter(caracol3, Sh);  
caracol3s = abs(caracol3sh) + abs(caracol3sv);  
  
montage({caracol1, caracol1s})
```



```
montage({caracol2, caracol2s})
```



```
montage({caracol3, caracol3s})
```



```
caracol1size = size(caracol1s);
multiplicador = 0;
m1 = uint64(0);
for i = 1:(caracol1size(1)/2)
    for j = 1:(caracol1size(2)/2)
        m1 = m1 + uint64(caracol1s(i, j)) * multiplicador;
        m1 = m1 + uint64(caracol1s(i, caracol1size(2) - j)) * multiplicador;
        m1 = m1 + uint64(caracol1s(caracol1size(1) - i, j)) * multiplicador;
        m1 = m1 + uint64(caracol1s(caracol1size(1) - i, caracol1size(2) - j)) * multiplicador;
    end
    multiplicador = multiplicador + 1;
end

caracol3size = size(caracol3s);
multiplicador = 0;
m3 = uint64(0);
for i = 1:(caracol3size(1)/2)
    for j = 1:(caracol3size(2)/2)
        m3 = m3 + uint64(caracol3s(i, j)) * multiplicador;
        m3 = m3 + uint64(caracol3s(i, caracol3size(2) - j)) * multiplicador;
        m3 = m3 + uint64(caracol3s(caracol3size(1) - i, j)) * multiplicador;
        m3 = m3 + uint64(caracol3s(caracol3size(1) - i, caracol3size(2) - j)) * multiplicador;
    end
    multiplicador = multiplicador + 1;
end

caracol2size = size(caracol2s);
```

```

multiplicador = 0;
m2 = uint64(0);
for i = 1:(caracol2size(1)/2)
    for j = 1:(caracol2size(2)/2)
        m2 = m2 + uint64(caracol2s(i, j)) * multiplicador;
        m2 = m2 + uint64(caracol2s(i, caracol2size(2) - j)) * multiplicador;
        m2 = m2 + uint64(caracol2s(caracol2size(1) - i, j)) * multiplicador;
        m2 = m2 + uint64(caracol2s(caracol2size(1) - i, caracol2size(2) - j)) * multiplicador;
    end
    multiplicador = multiplicador + 1;
end

disp(m1)

```

5614889850

```
disp(m2)
```

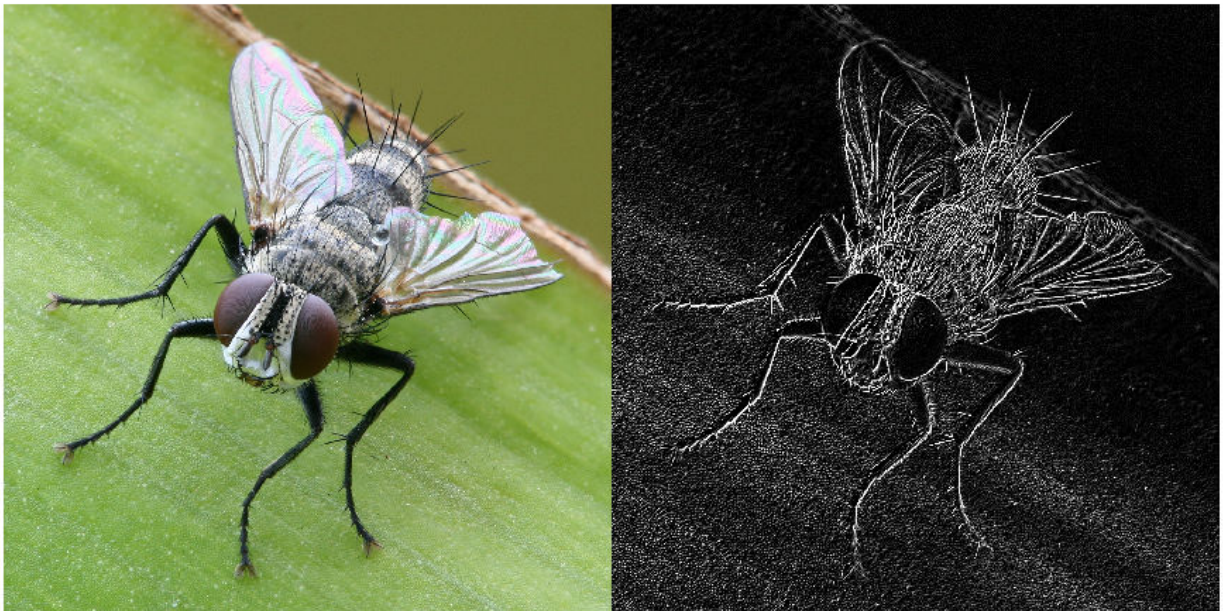
8561093084

```
disp(m3)
```

15213174524

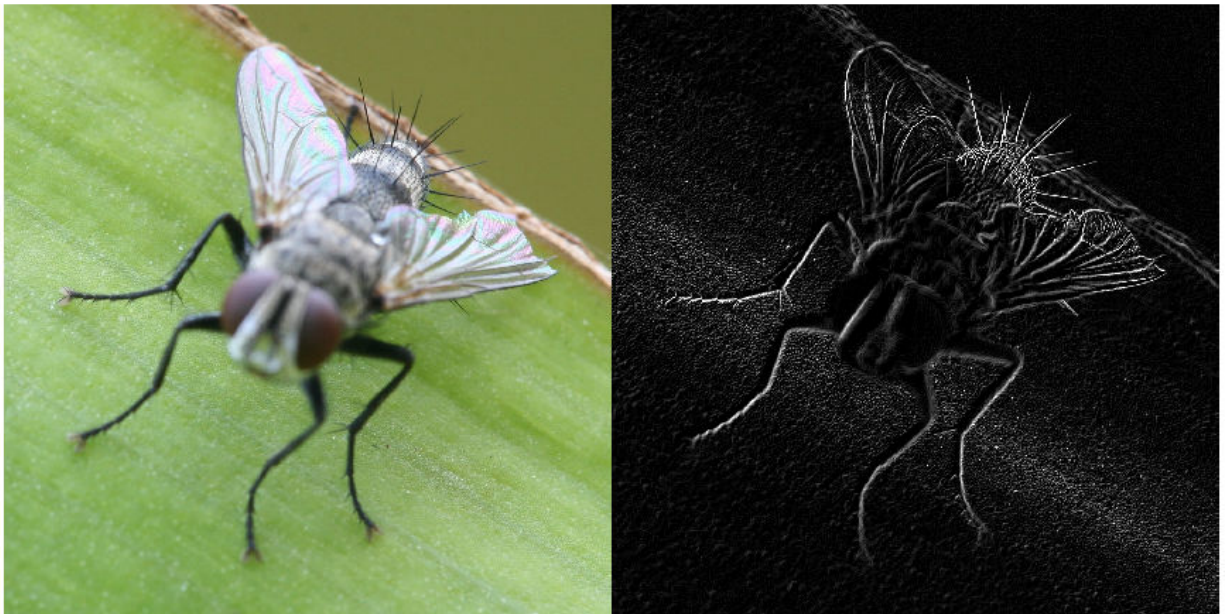
Com podem la imatge3 que és la que s'ha creat a partir de fer stacking amb les altres dues és la que té una millor puntuació. Més això podem dir que la nostra heurística funciona correctament.

```
montage({caracol3original, caracol3s})
```



```
montage({caracol2original, caracol2s})
```





```
montage({caracol1original, caracol1s})
```

