Utility routines

The area around (x,y) in M is "colored" in color nr and the new
Matrix is returned (non-recursive version) (very slow)

```
fill(M, x, y, nr) :=
    Stack ← (x  y)^T
    index ← 0
    color ← M_{x,y}
    while index ≥ 0
        (x  y) ← Stack^⟨index⟩^T
        index ← index − 1
        if (0 ≤ x ≤ rows(M) − 1) ∧ (0 ≤ y ≤ cols(M) − 1) ∧ (M_{x,y} = color)
            M_{x,y} ← nr
            neighbours ← [(x, y−1)  (x, y+1)  (x−1, y)  (x+1, y)]
            for i ∈ 0..3
                Stack^⟨index+1+i⟩ ← neighbours_{0,i}
            index ← index + 4
    return M
```

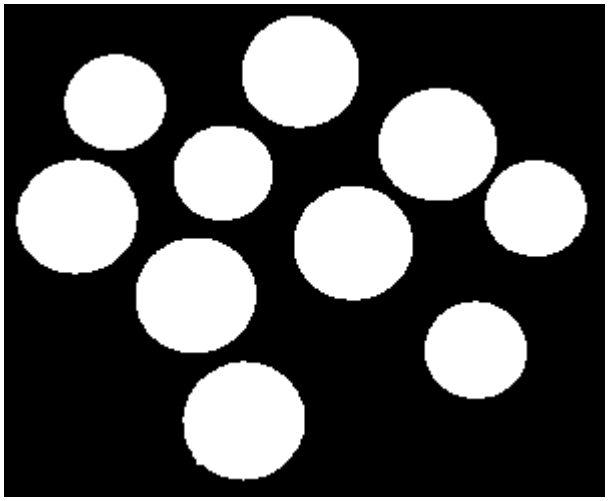
Input M1 is assumed to be a matrix with values 0 or 1. A blob is assumed to be "color" 1(white)

```
count_blobs(M) :=
    R ← rows(M)
    C ← cols(M)
    count ← 0
    for r ∈ 0..R − 1
        for c ∈ 0..C − 1
            if M_{r,c} = 1
                count ← count + 1
                M ← fill(M, r, c, count + 1)
    (count, M)
```

Utility routines

$Img := READ\_IMAGE("coins.jpg")$

$Img_{clean} := erode8(dilate8(binarize\_auto(Img), 1), 1)$



Img

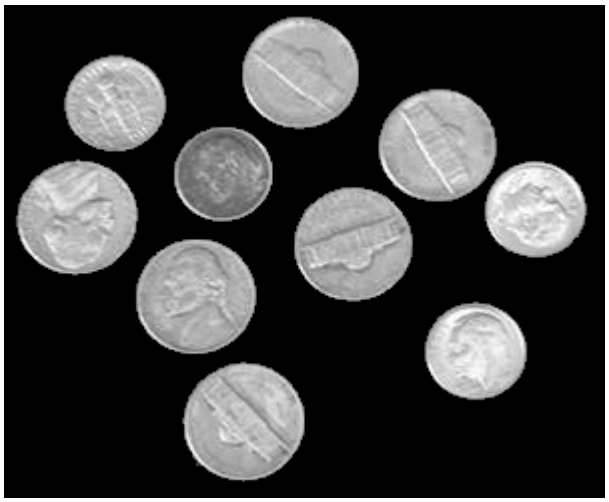
$Img_{clean} \cdot 255$

$\begin{pmatrix} nr_{blobs} \\ Img2 \end{pmatrix} := count\_blobs(Img_{clean})$

$nr_{blobs} = 10$

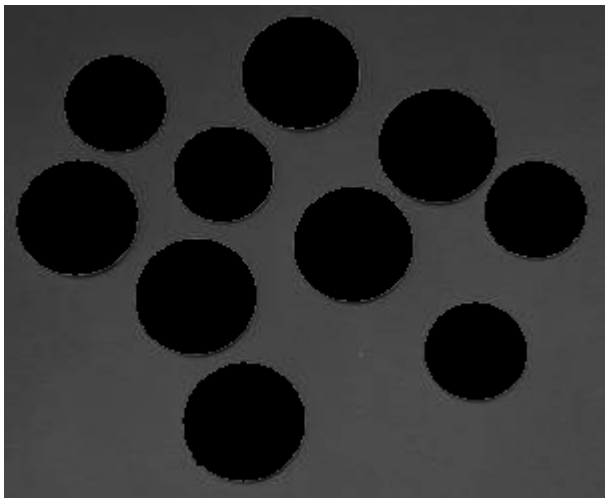
Img2 is a matrix where the regions are denoted by values from 2 up. There should be no value 1 in that matrix.

We can use Img.clean to mask all objects



$\overrightarrow{(Img \cdot Img_{clean})}$

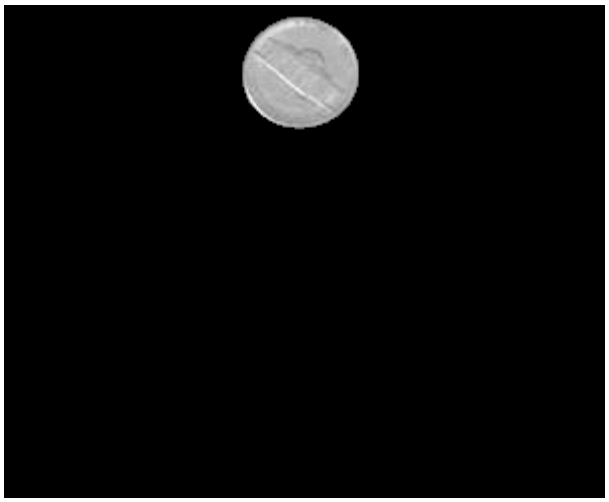
or inverse mask



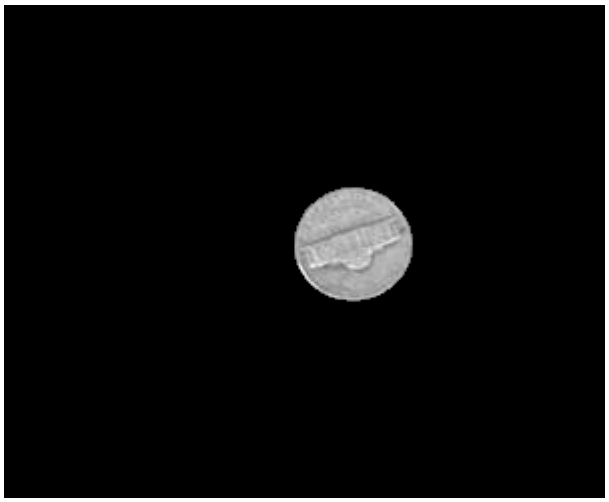
$\overrightarrow{[Img \cdot (1 - Img_{clean})]}$

We may also use Img2 to mask all but a single object (remember, object 1 is denoted by 2's, etc.

$coin := 1$ $\overrightarrow{Mask := [(coin + 1) = Img2]}$



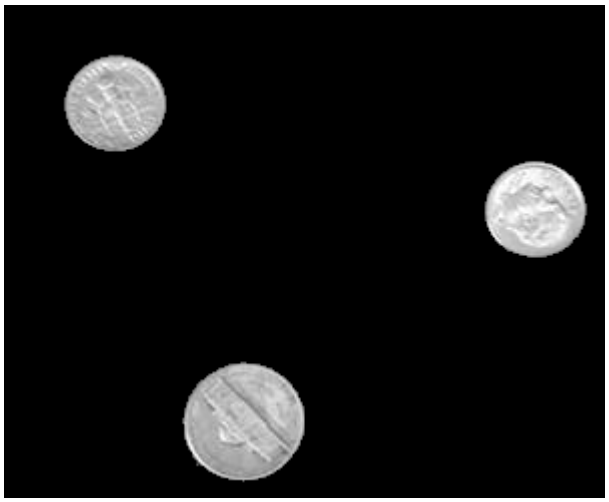
$\overrightarrow{[(Img)\cdot(Mask)]}$



$\overrightarrow{[(Img)\cdot(Mask)]}$

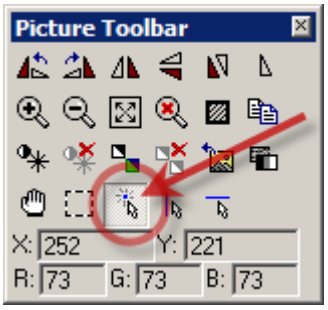
Or lets show coins nr. 2, 6 and 10

$\overrightarrow{Mask2 := (Img2 = 3 \vee Img2 = 7 \vee Img2 = 11)}$



$\overrightarrow{[(Img)\cdot(Mask2)]}$

Chose "Mouse Selects Pixel" in Picture Toolbar and click on any object



$nr := Img2_{(XY_1),XY_0}$ $\overrightarrow{Mask := (Img2 = nr)}$

Object number $nr - 1 = 7$