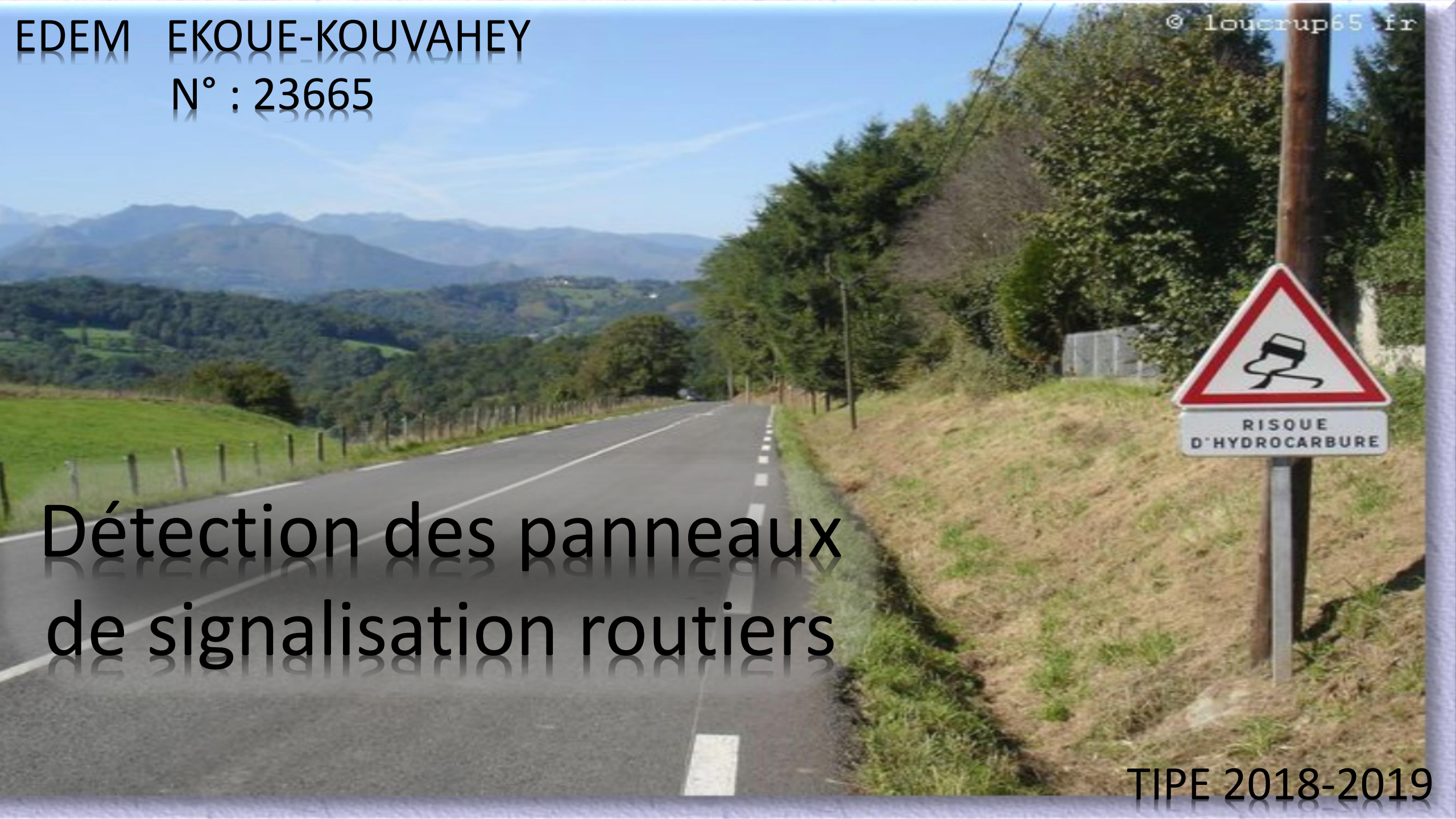


EDEM EKOUE-KOUVAHEY  
N° : 23665

© loucrup65.fr



Détection des panneaux  
de signalisation routiers

TIPE 2018-2019

# Problématique

---

Il convient alors de se demander comment procéder pour détecter les panneaux de signalisations.

# PLAN

---

I

- Acquisition et analyse des panneaux routiers

II

- Détections des couleurs dans les panneaux routiers

III

- Détections de formes des panneaux

IV

- Conclusion

# I- Acquisition et analyse des panneaux de signalisations

## A-) Analyse des panneaux de signalisations

HYPOTHESE : On se limitera dans notre étude aux panneaux rouges et bleus



Pixel

$$PIXEL = [r, v, b]$$

- ON CALCULE LA MOYENNE SUR L'IMAGE DE CHAQUE COMPOSANTE

R = moyenne de la composante rouge sur l'image

B = moyenne de la composante bleue sur l'image

V = moyenne de la composante verte sur l'image



# Exemple1 : Panneau rouge



*Taille = (n, m)*

- $P = n \times m$
- $R = 127,32$
- $V = 98,50$
- $B = 87,40$

## Exemple2: Panneau bleu



*Taille = (n, m)*

- $P = n \times m$

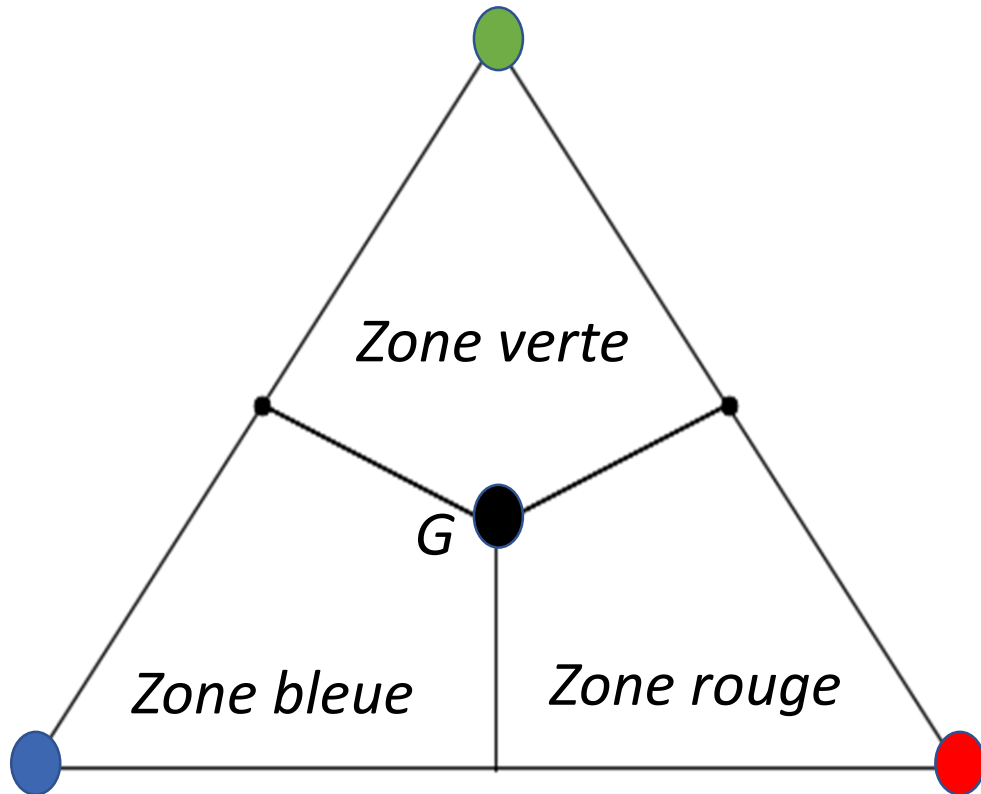
- $R = 107,82$

- $V = 149,93$

- $B = 198,13$

# II- Détections de couleurs des panneaux

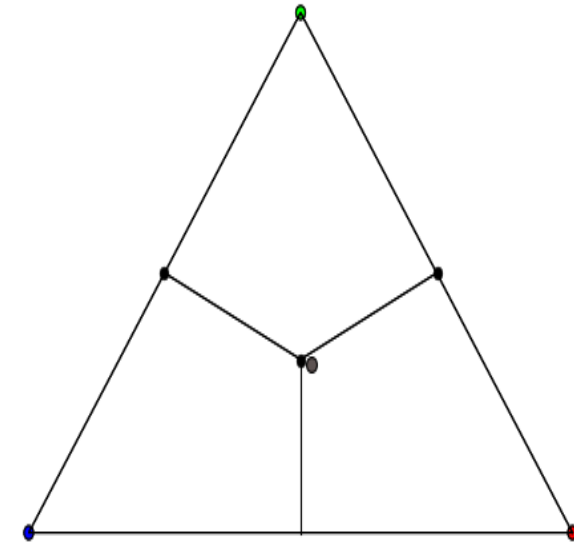
## *A – Triangle de couleurs de Maxwell*



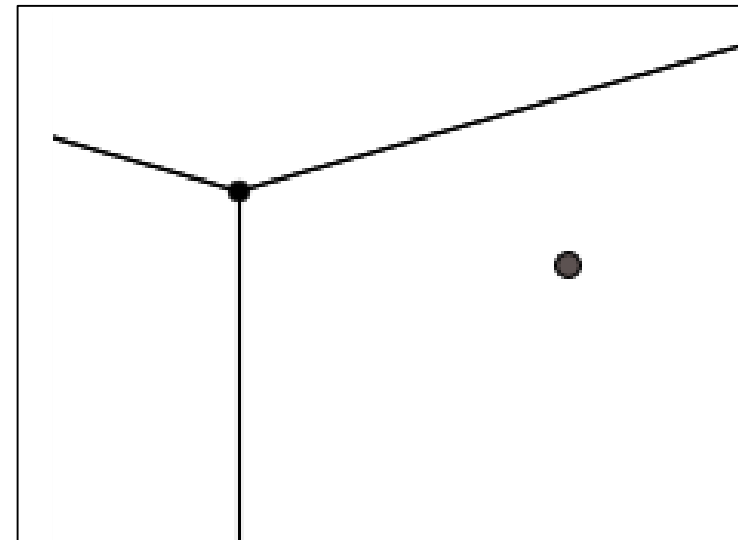
- $\alpha = \frac{R}{R+V+B}$  ;  $\beta = \frac{V}{R+V+B}$  ;  $\gamma = \frac{B}{R+V+B}$
- $\alpha + \beta + \gamma = 1$
- $(\alpha - \gamma; \beta)$  = coordonnées du panneau dans le triangle de Maxwell

# Exemple1: Panneau rouge

- $\alpha = 0,35$
- $\beta = 0,32$
- $\gamma = 0,31$
- $\alpha - \gamma = 0,04$



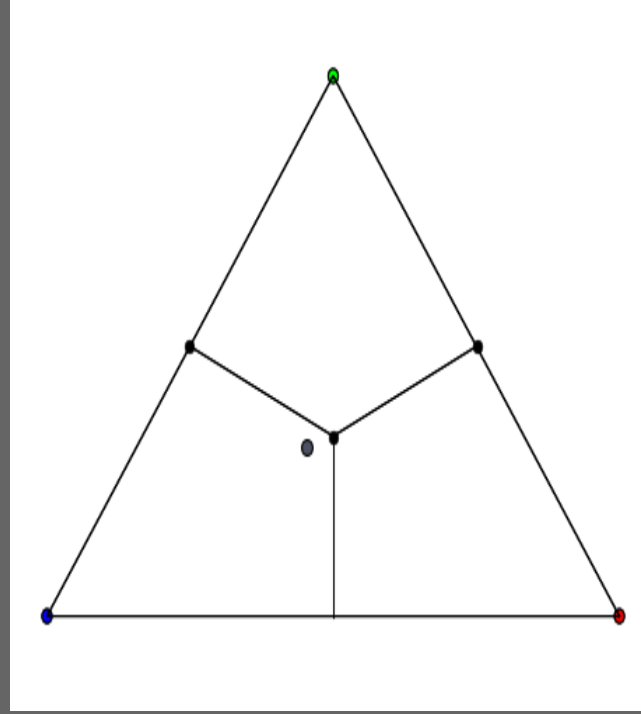
*zoom*



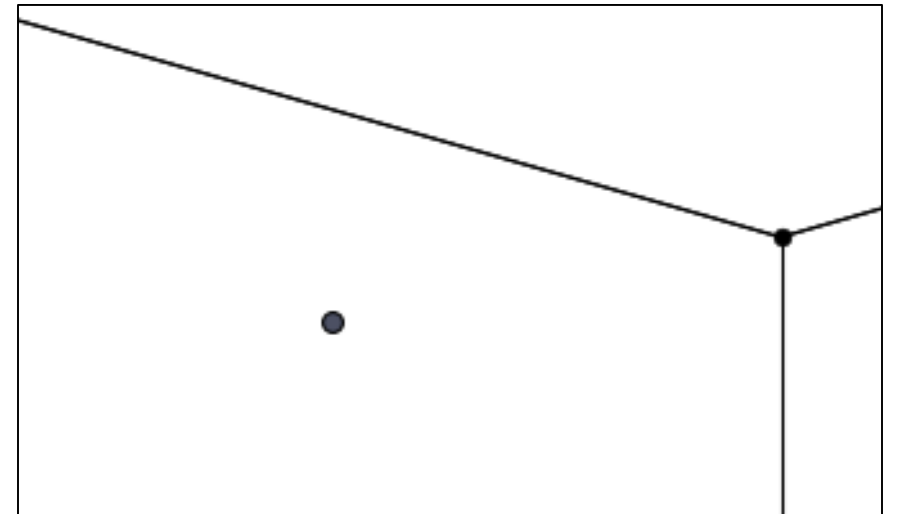


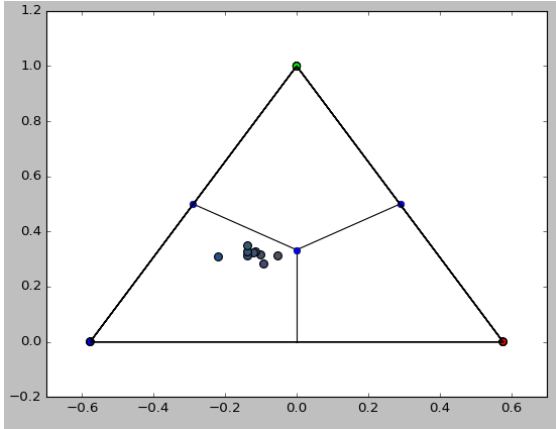
## Exemple2: Panneau bleu

- $\alpha = 0,29$
- $\beta = 0,31$
- $\gamma = 0,38$
- $\alpha - \gamma = -0,09$

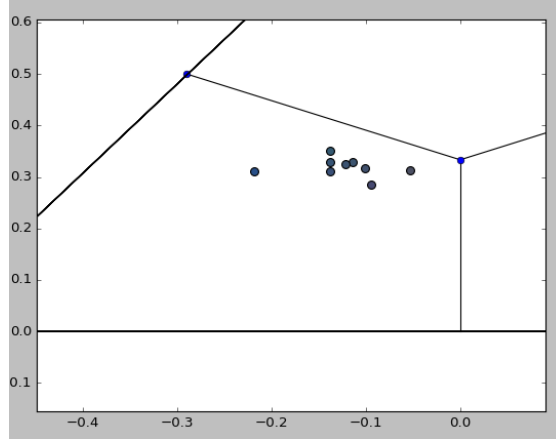


*zoom*

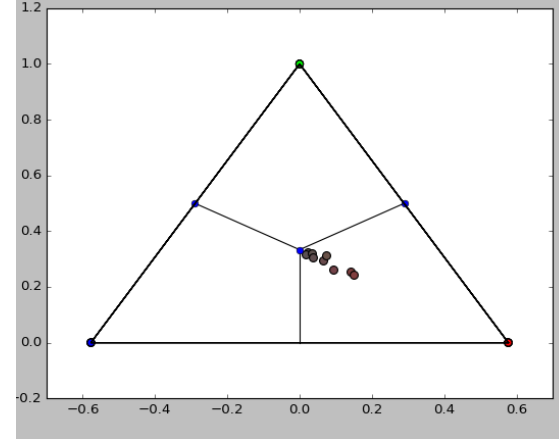




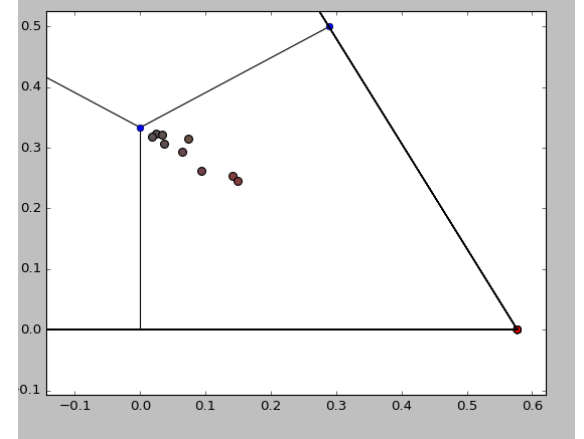
*Panneaux bleus*



*zoom*

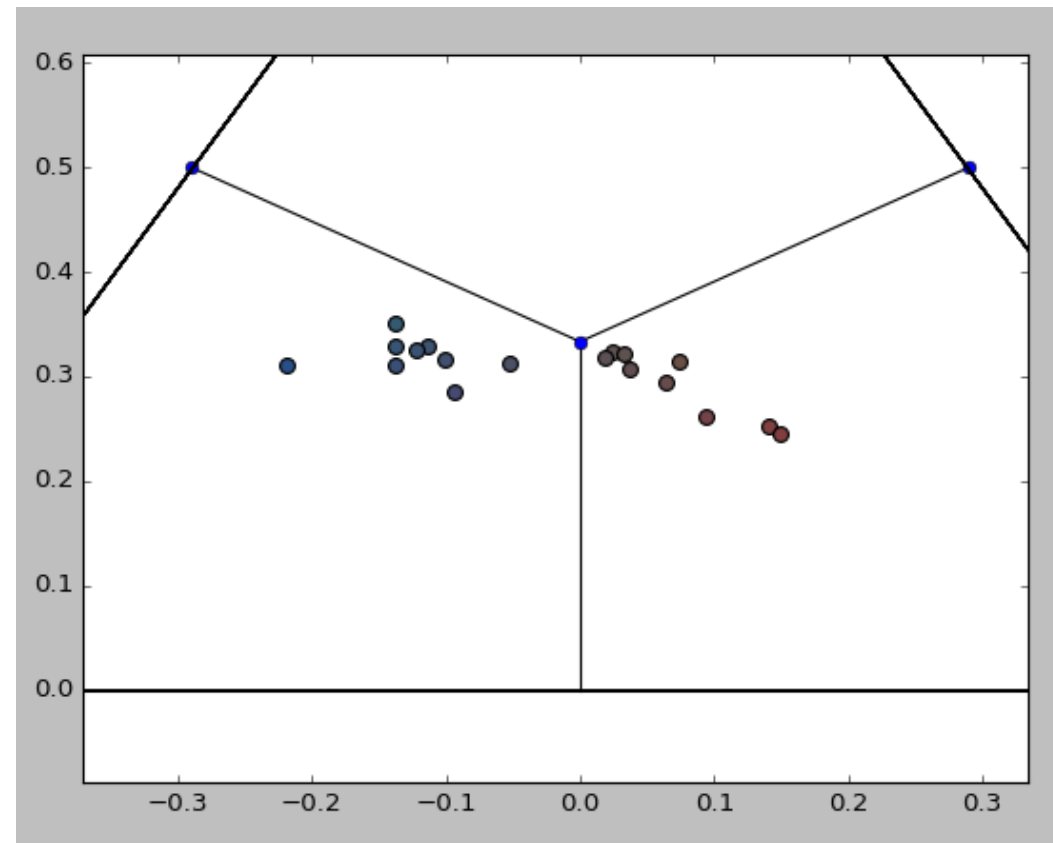
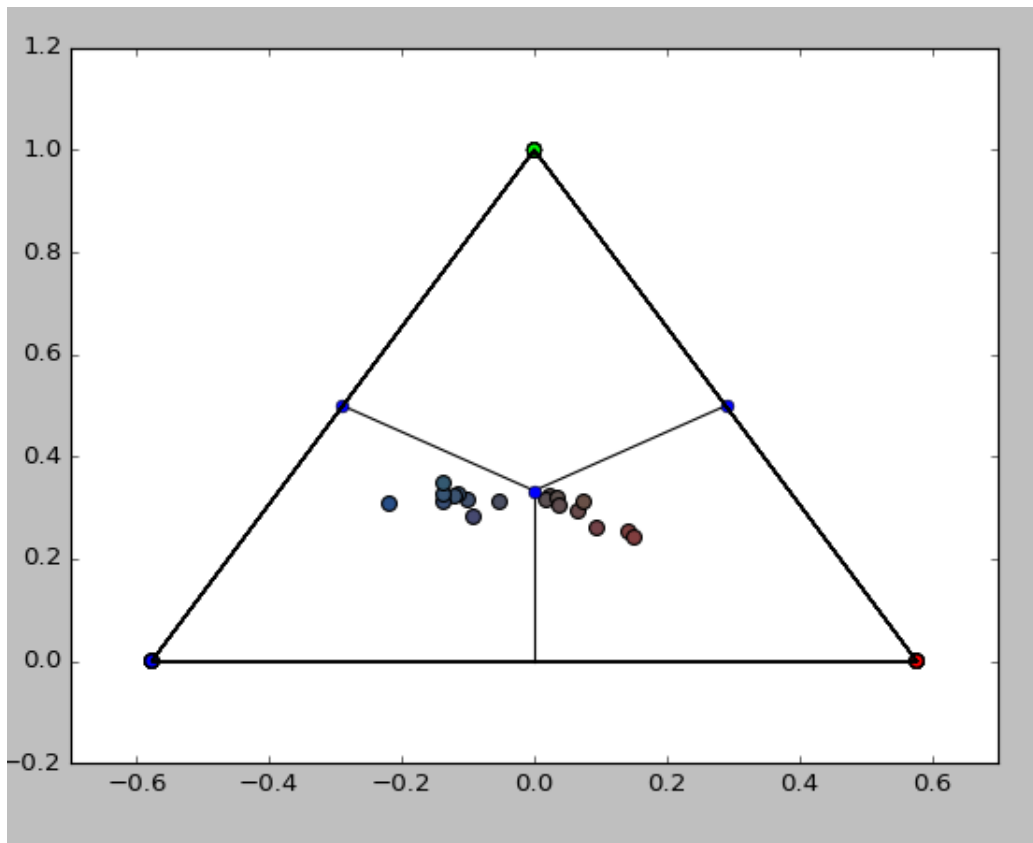


*Panneaux rouges*



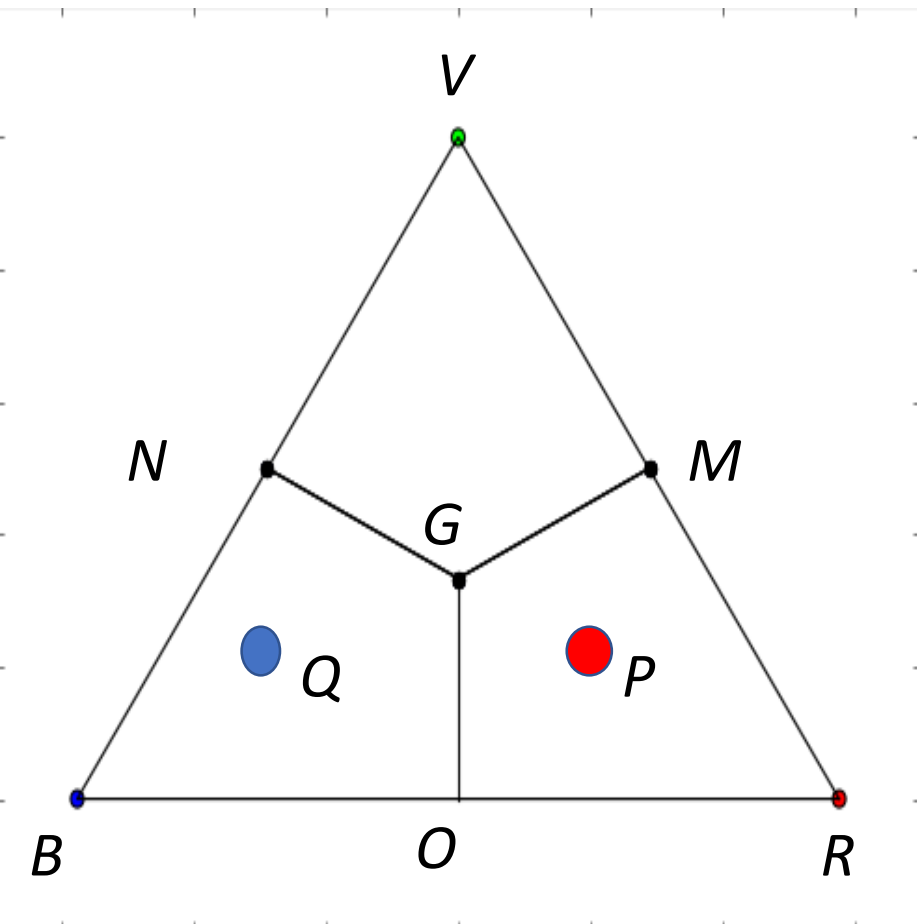
*zoom*

# Remplissage du triangle de Maxwell



Panneaux rouges et bleus

# B- Critères d'identification des panneaux rouges et bleus



- ROGM
  - $(OG) h : x \longrightarrow 0$
  - $(GM) f : x \longrightarrow \frac{1}{\sqrt{3}}x + \left(\frac{1}{3}\right)$
  - $(MR) v : x \longrightarrow -1.74x + 1$
- BOGN
  - $(OG) h : x \longrightarrow 0$
  - $(GN) g : x \longrightarrow -\frac{1}{\sqrt{3}}x + \left(\frac{1}{3}\right)$
  - $(NB) u : x \longrightarrow 1.74x + 1$

# Exemple1: Identification d'un panneau quelconque



- $S = (0,093; 0,26)$
- Abscisse  $>0$  ;  $v < \text{ordonnée} < f$

```
>>> identification(PR7,f,g,h,v)  
'le panneau est rouge'
```

## Exemple2: Identification d'un panneau bleu quelconque

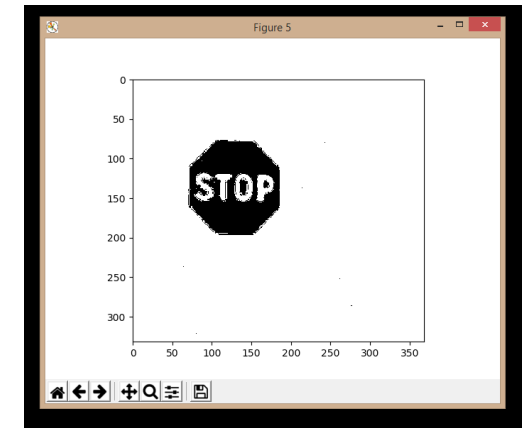
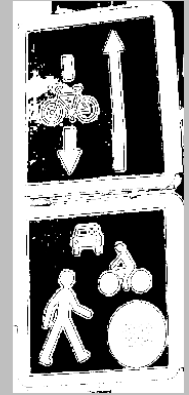


- $S = (-0,13; 0,32)$
- $\text{Abcisse} < 0$  et  $g < \text{ordonnée} < u$

```
>>> identification(PB7,f,g,h,v)  
'le panneau est bleu'
```



# Seuillage des images après indentation

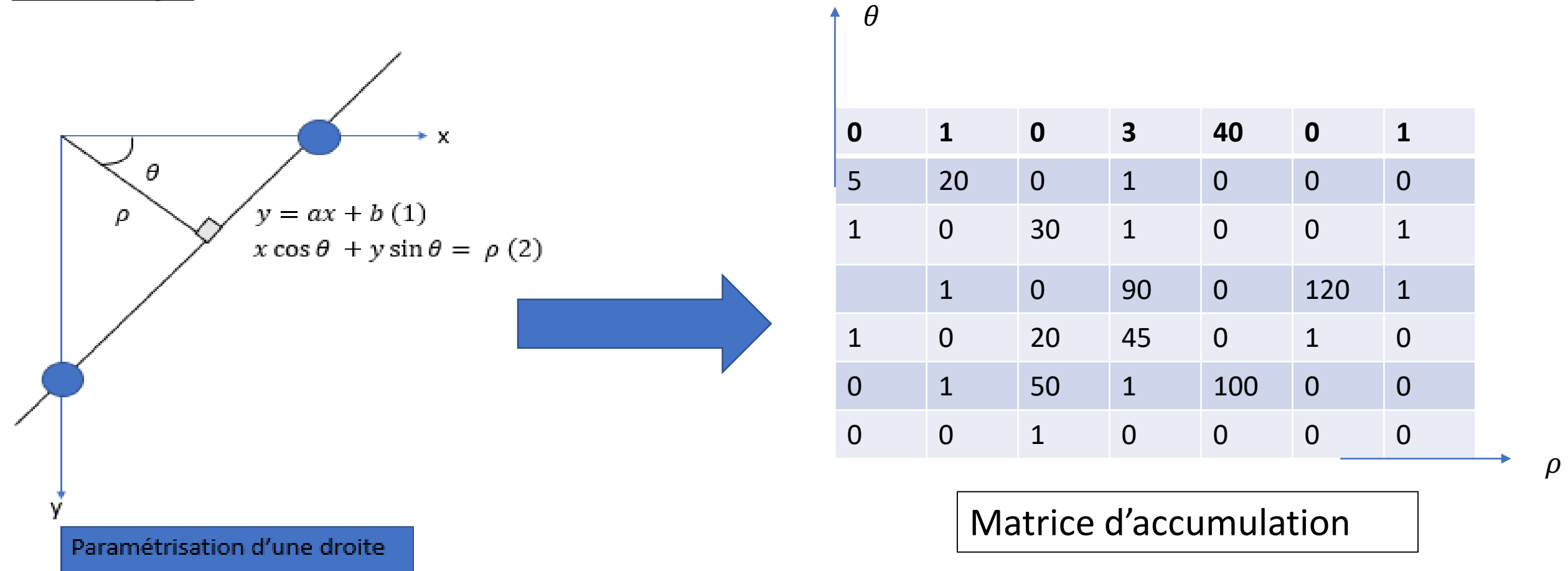


### III -) Détections des formes de panneaux

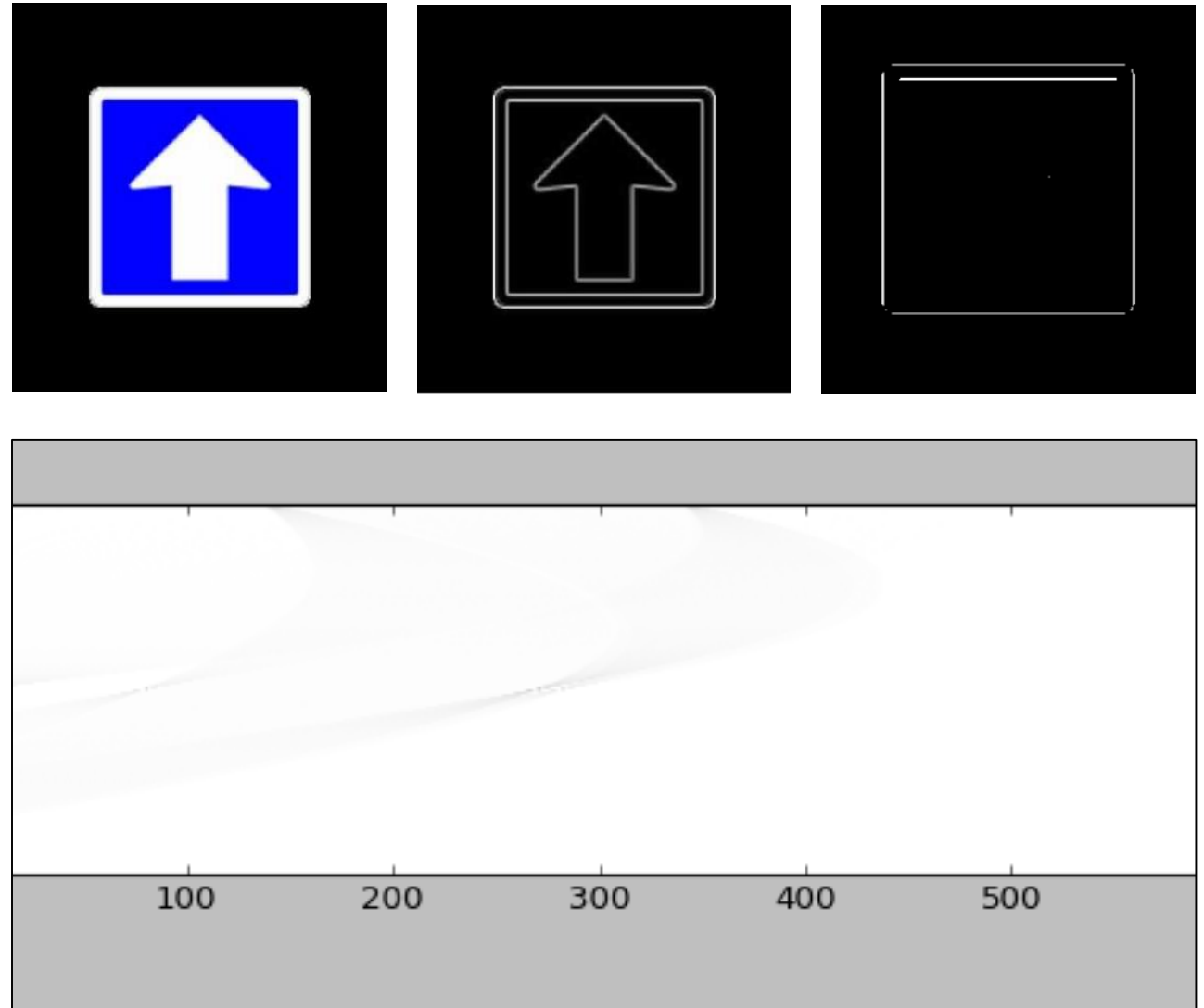
HYPOTHESE : on va se limiter à l'étude des formes carrées , triangulaires

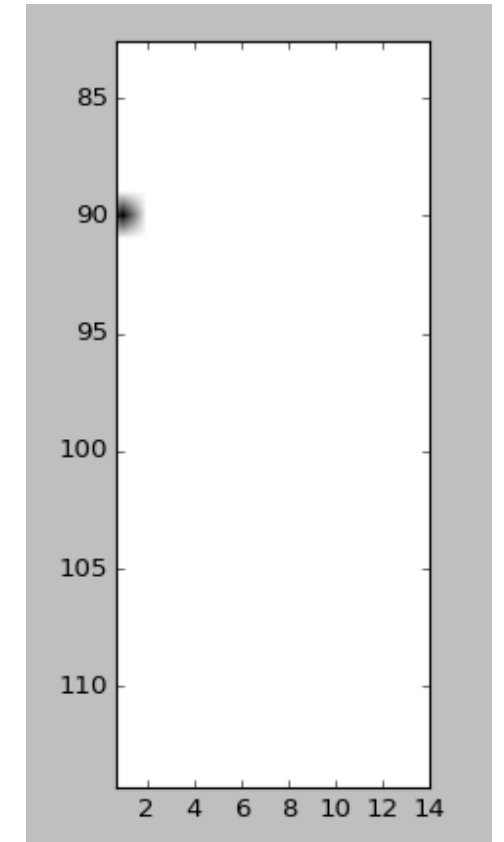
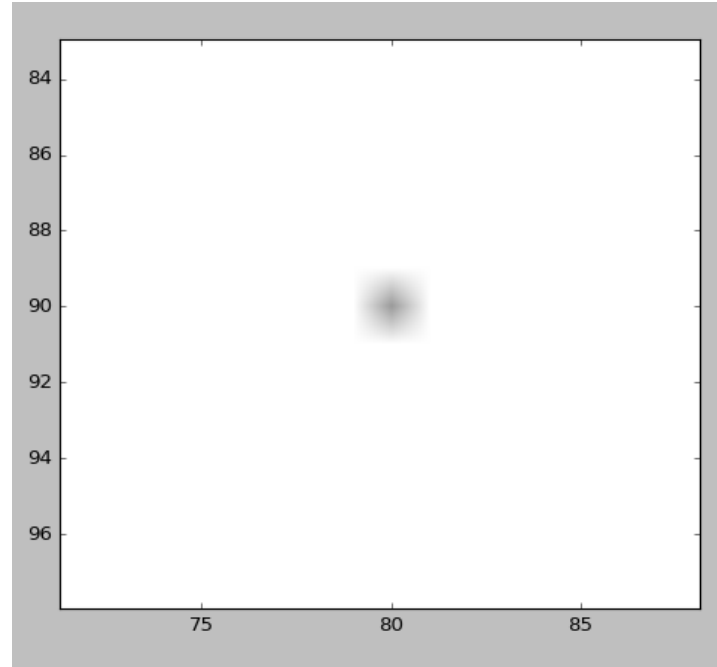
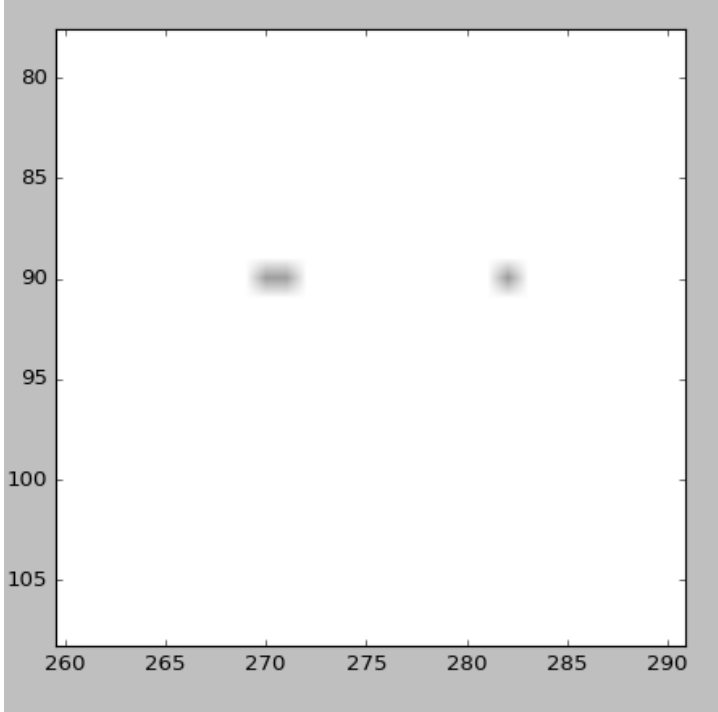
- détection de contours faites au préalable

#### 1- Détections d'un panneau de formes carrées ou triangulaire par la Transformée de Hough



Exemple1:  
détection d'un  
panneau carré



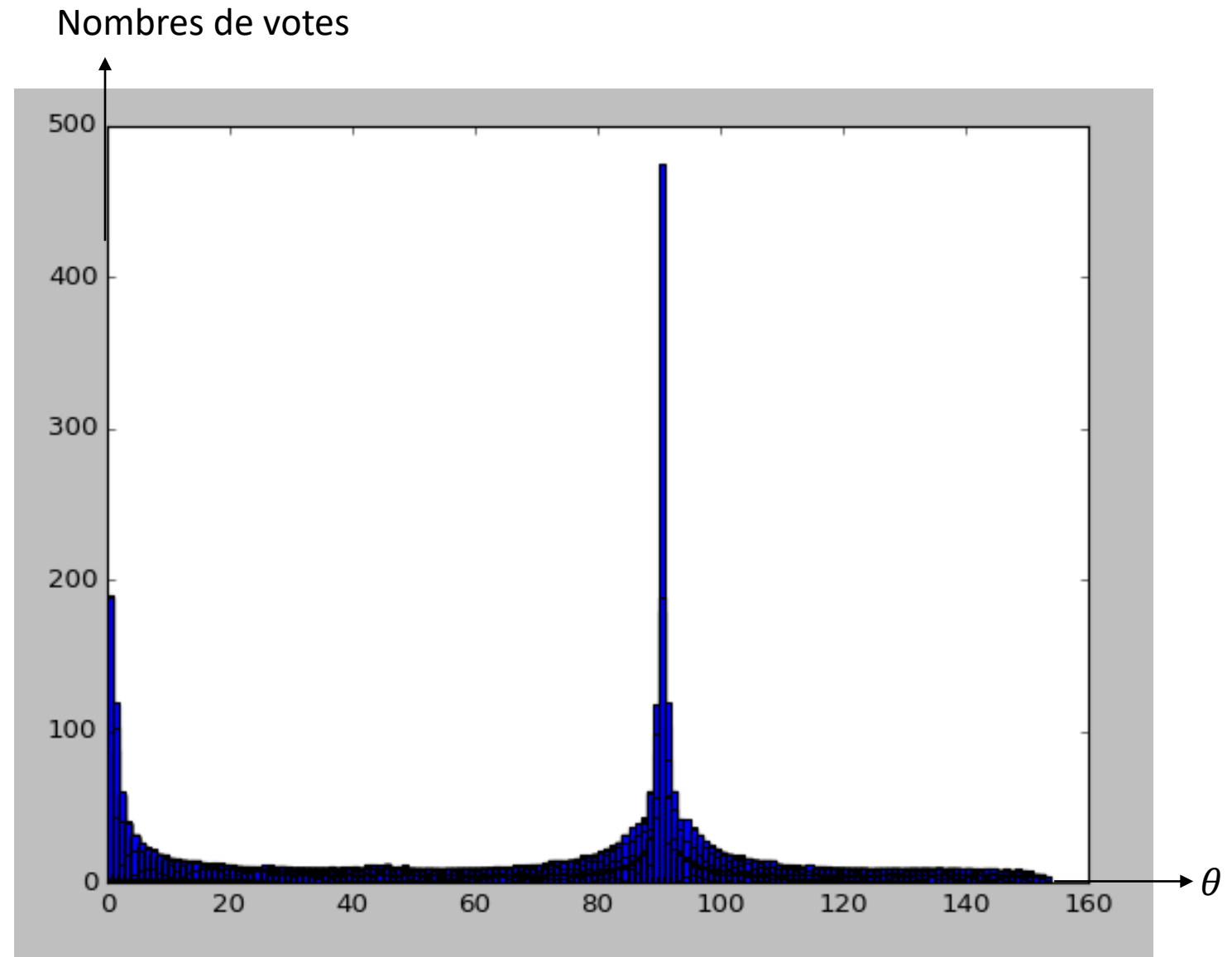


# Zoom sur la Matrice (carré)

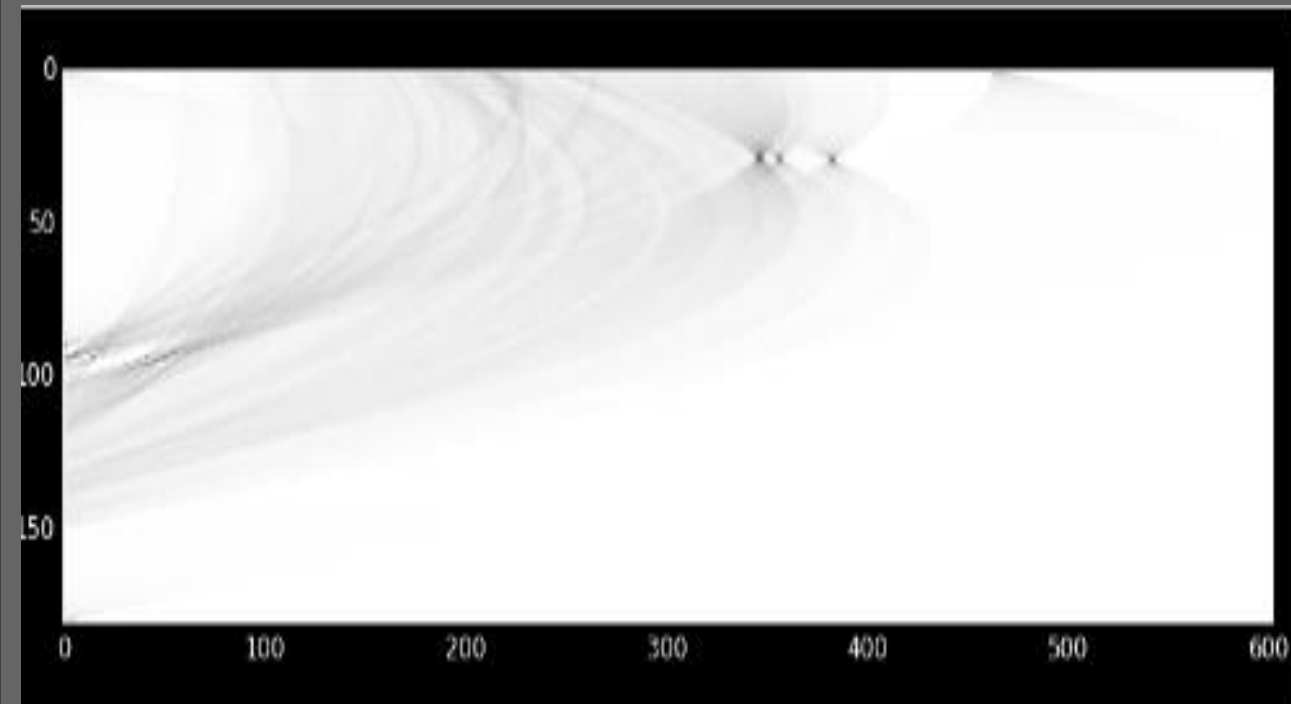
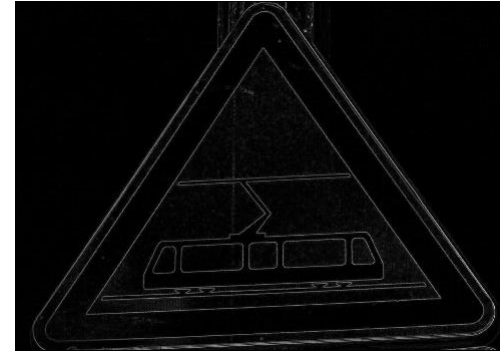
---

# Histogramme d'accumulation pour le carré

---

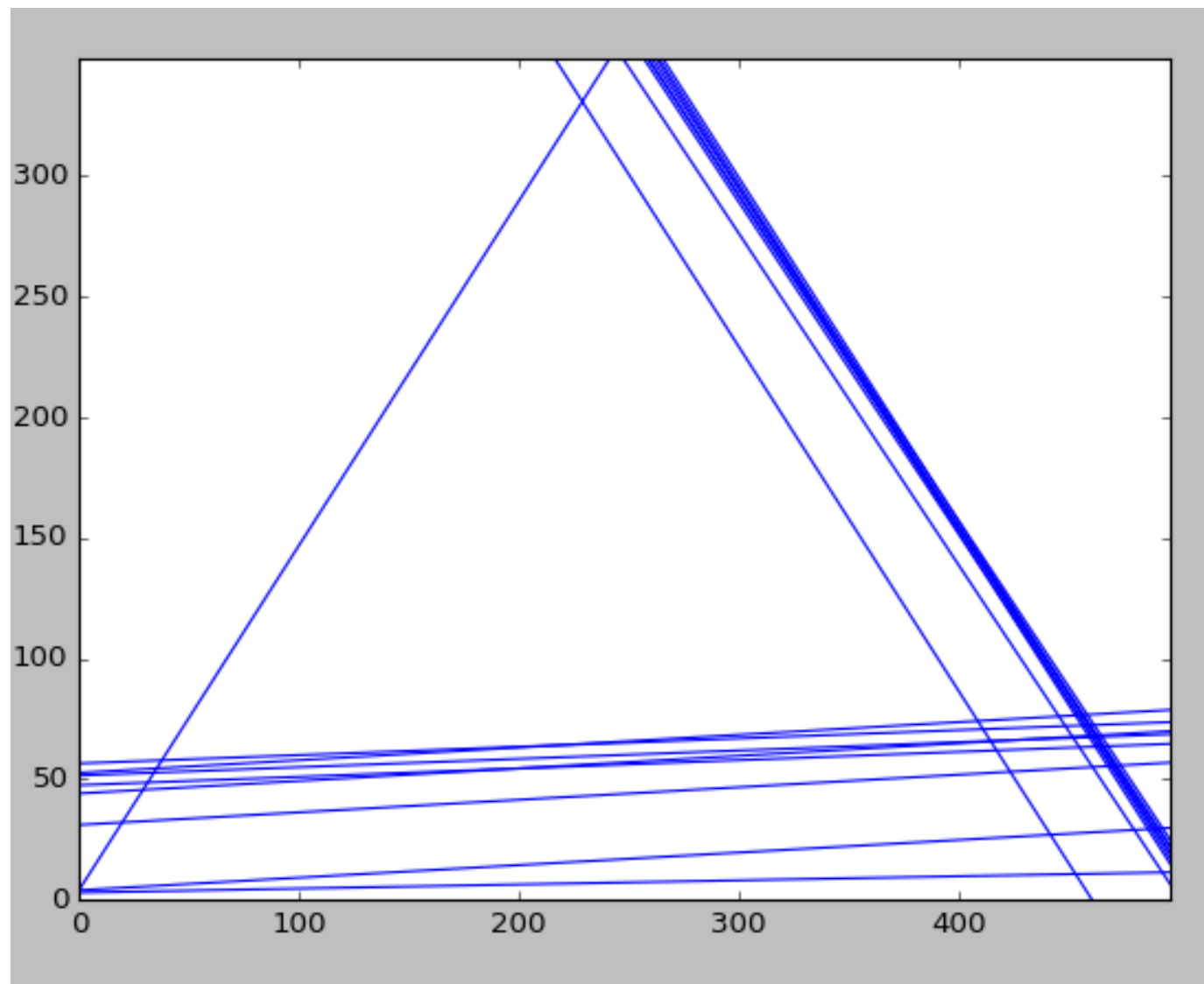


## Exemple2: détection d'un panneau triangulaire

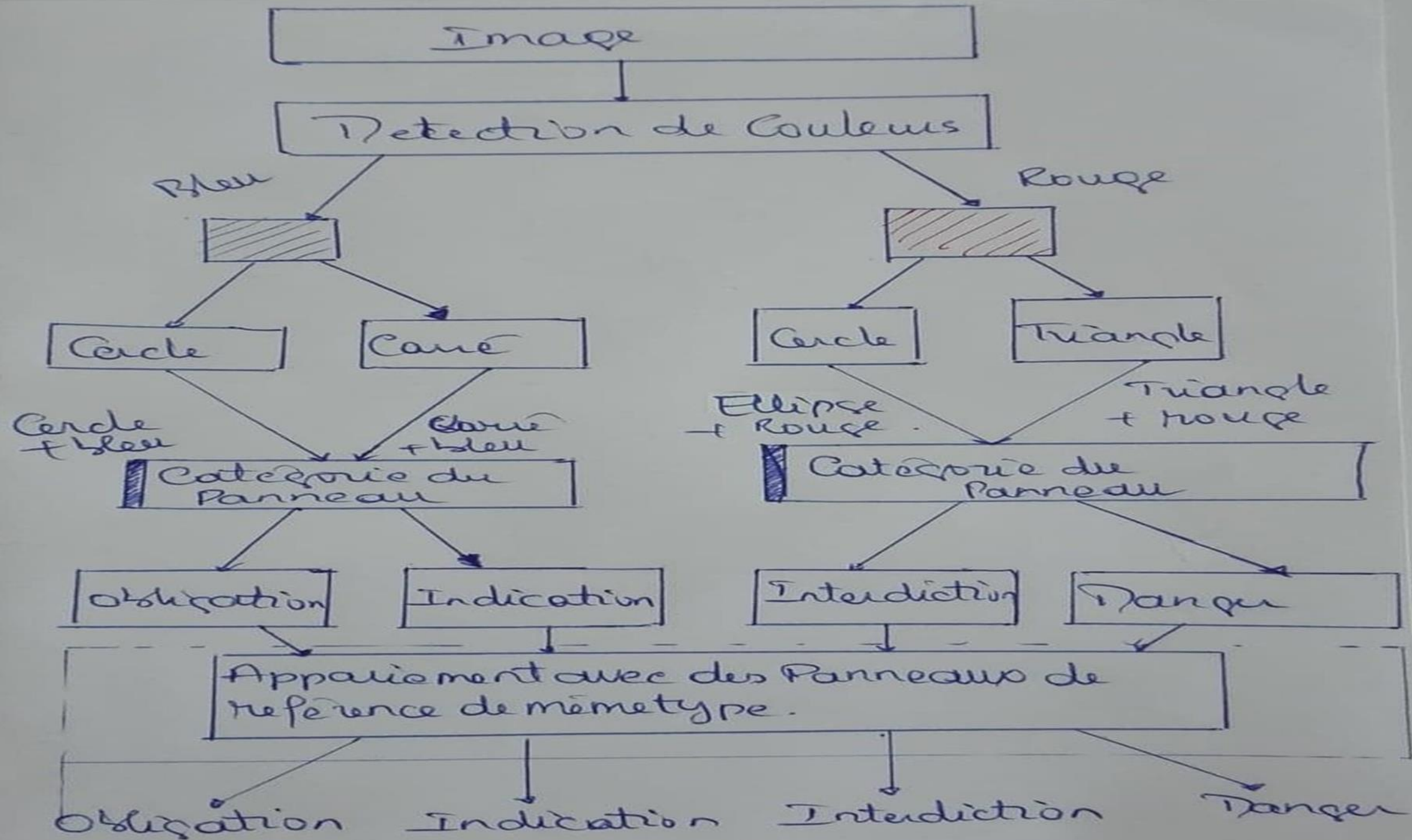




Tracé des lignes de la  
matrice  
d'accumulation pour  
le triangle



## IV-)Conclusion



- ANNEXE 1:

```
## Matrice d'accumulation
```

```
(Ny,Nx) = A.shape
```

```
rho= 1.0
```

```
theta=1.0
```

```
Ntheta = int(180.0/theta)
```

```
Nrho = int(math.floor(math.sqrt(Nx*Nx+Ny*Ny))/rho)
```

```
dtheta = math.pi/Ntheta
```

```
drho = math.floor(math.sqrt(Nx*Nx+Ny*Ny))/Nrho
```

```
accum = np.zeros((Ntheta,Nrho))
```

```
for j in range(Ny):
```

```
    for i in range(Nx):
```

```
        if A[j][i]!=0:
```

```
            for itheta in range(Ntheta):
```

```
                theta = itheta*dtheta
```

```
                rho = i*math.cos(theta)+(Ny-j)*math.sin(theta)
```

```
                irho = int(rho/drho)
```

```
                if (irho>0) and (irho<Nrho):
```

```
                    accum[itheta][irho] += 1
```

```
## représentation de la matrice d'accumulation
```

```
plt.figure(7)
plt.figure(figsize=(12,6))
plt.imshow(accum,cmap=cm.gray),plt.show()
```

```
##seuillage de la matrice d'accumulation
```

```
s= 250
```

```
accums = np.copy(accum)
```

```
for itheta in range(Ntheta):
    for irho in range(Nrho):
        if accum[itheta][irho]<s:
            accums[itheta][irho] = 0
```

```
plt.figure(8)
plt.figure(figsize=(12,6))
```

```
plt.imshow(accums,cmap=cm.gray)
```

```
plt.show()
```

```
## on va creer une liste de valeurs definissant les points
```

```
lignes = []
```

```
for itheta in range(Ntheta):
```

```
    for irho in range(Nrho):
```

```
        if accums[itheta][irho]!=0:
```

```
            lignes.append((irho*drho,itheta*dtheta))
```

## Annexe2:

```
X = np.linspace(-1.0/math.sqrt(3),1.0/math.sqrt(3))
plt.plot(X,f(X),'r')
plt.plot(X,g(X),'b')
plt.plot([0,0],[0,1/3])
plt.plot(X,v(X),'k')
plt.plot(X,h(X),'k')
plt.show()

def identification(H,f,g,h,v):
    T = point(moyenne(H))
    k = T[0]
    l = T[1]

    X = np.linspace(-1.0/math.sqrt(3),1.0/math.sqrt(3))
    for x in X:
        if k > 0 and l < f(x) and l < v(x):
            return 'le panneau est rouge'

        if k < 0 and l < g(x) and l < h(x):
            return 'le panneau est bleu'
```



## ## tracé du triangle de maxwell

```
a = 1.0/math.sqrt(3)
plt.scatter([a,0,-a],[0,1,0],s=40,c=[(1,0,0),(0,1,0),(0,0,1)])
plt.plot([a,0,-a,a],[0,1,0,0],'k-')
plt.axis([-0.7,0.7,-0.2,1.2],)
```

```
#plt.plot([0,0],[1,0],"k")
#plt.plot([1/math.sqrt(3),-0.3],[0,0.5],"k")
#plt.plot([-1/math.sqrt(3),0.3],[0,0.5],"k")
```

```
plt.plot([0.29,0],[0.5,1/3],"k") ## segment à droite rouge
```

```
plt.plot([-0.29,0],[0.5,1/3],"k") ## segment à gauche bleu
```

```
plt.plot([0,0],[0,1/3],"k") ## segment du milieu
```

##

## ## placement de points dans le triangle de maxwell

```
def point(rvb):
    somme = rvb[0]+rvb[1]+rvb[2]
    r = rvb[0]*1.0/somme
    v = rvb[1]*1.0/somme
    b = rvb[2]*1.0/somme
    a = 1.0/math.sqrt(3)
    L = []
    plt.scatter([a,0,-a],[0,1,0],s=40,c=[(1,0,0),(0,1,0),(0,0,1)])

    x =(r-b)/math.sqrt(3)
    y = v

    L.append(x)
    L.append(y)

    plt.scatter([(r-b)/math.sqrt(3)],[v],s=50,c=(r,v,b))
    plt.plot([a,0,-a,a],[0,1,0,0],'k-')
    return L
```

## ## fonctions critères pour être un panneau rouge ou bleue

```
def f(x): ## droite passant par le point rouge et passant par le barycentre
    return (((1/3)-0.5)/(0-0.29))*x + (1/3)
```

```
def g(x): ## droite passant par le point bleu et passant par le barycentre
    return ((1/3-0.5)/(0.29))*x + 1/3
```

```
def h(x): ## droite passant passant par le point bleu et passant par le point vert
    return ((0.5)/(-0.29+a))*x+1
```

```
def v(x): ## droite passant par le point rouge et passant par le point vert
    return (-0.5/(a-0.29))*x+1
```