

```
%load_ext autoreload
%autoreload 2

The autoreload extension is already loaded. To reload it, use:
%reload_ext autoreload

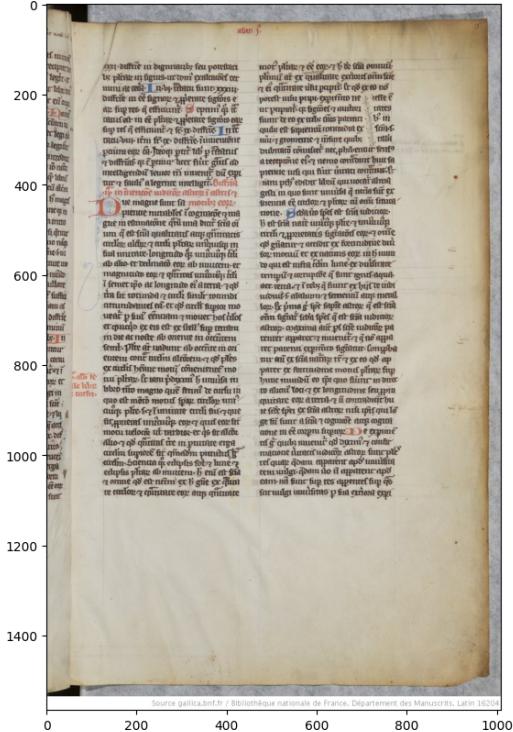
import cv2
import matplotlib.pyplot as plt
import numpy as np
from sklearn.metrics import mean_squared_error, mean_absolute_error

from functions.utils import *
from functions.cart import *
from functions.crop import *

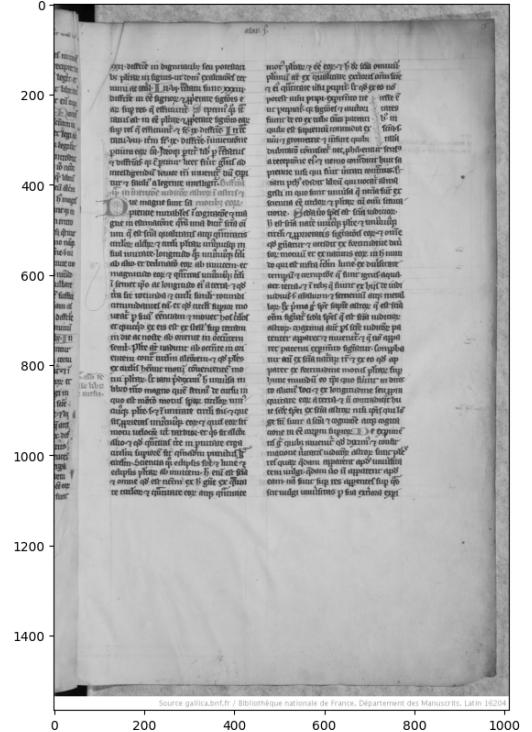
image = cv2.imread("../ImagesCodicologie/Corpus de jeu/Lebec/Tractactus_diversi_super_scie

# Change color to RGB (from BGR)
image = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
# convert to grayscale
gray = cv2.cvtColor(image, cv2.COLOR_RGB2GRAY)
# normalize the image
norm_image = gray/255.0
#f_image = ft_image(norm_image)
# Display the images
f, (ax1,ax2) = plt.subplots(1, 2, figsize=(20,10))
ax1.imshow(image)
ax2.imshow(norm_image, cmap="gray")
#ax2.imshow(f_image, cmap="gray")
```

<matplotlib.image.AxesImage at 0x1f0b32c8d50>



Source gallica.bnf.fr / Bibliothèque nationale de France, Département des Manuscrits, Latm 162v-4

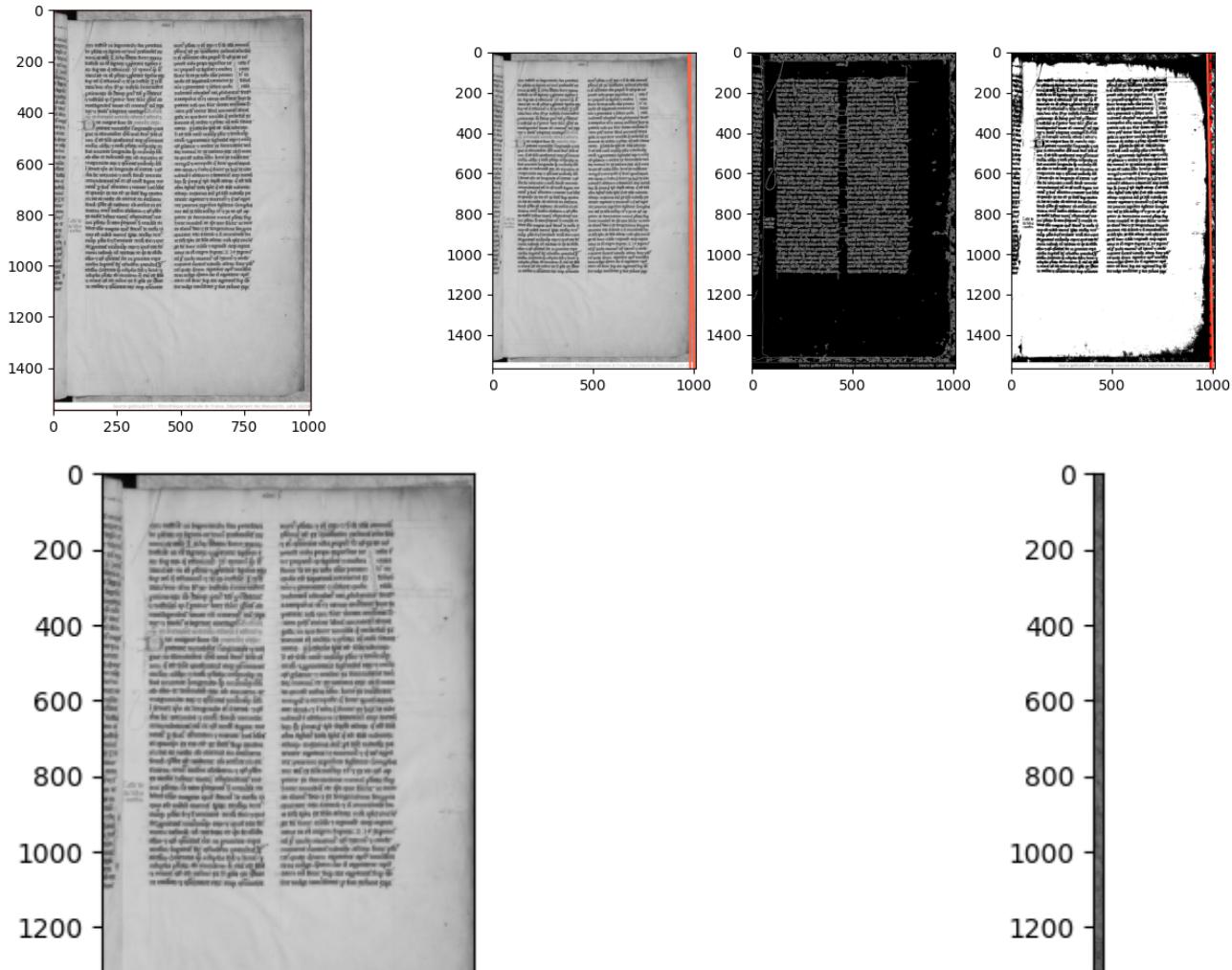


Source gallica.bnf.fr / Bibliothèque nationale de France, Département des Manuscrits, Latm 162v-4

split_pages(gray, c=1)



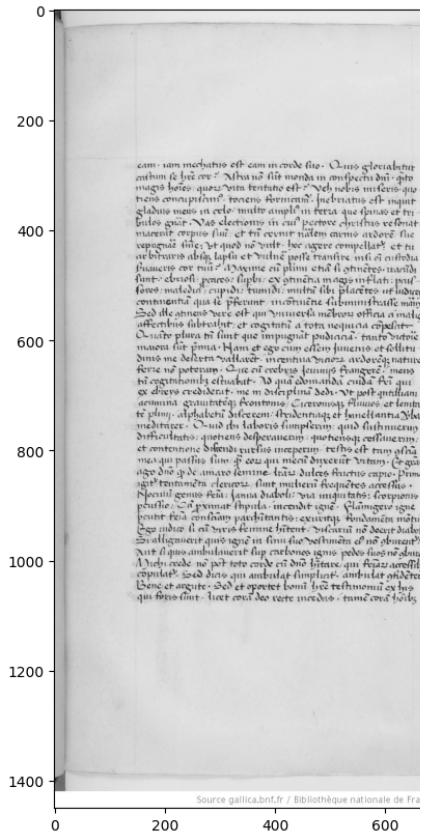
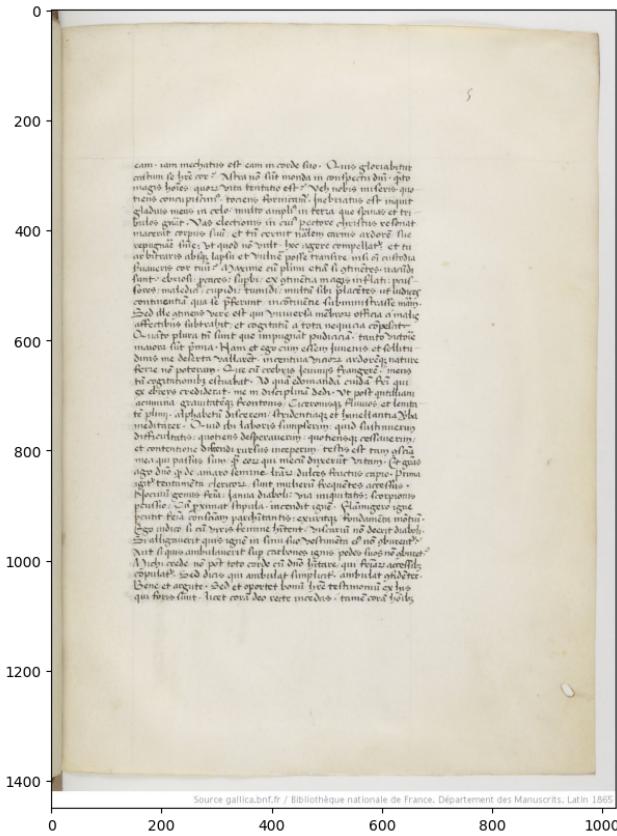
173.8570040337367
 11591 candidat lines amongst 23598 lines found
 1566 pixels calculated on final candidate line
 $-0.9998477 \ 0.017452458 \ 974 \ -17$
 $r = 0.8518518518518519$ with split at 987
 $r \max = 0.8518518518518519$



```
image = cv2.imread("../ImagesCodicologie/Corpus de jeu/Emilie Cottreau-Gabillet/BnF Lat 1
```

```
# Change color to RGB (from BGR)
image = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
# convert to grayscale
gray = cv2.cvtColor(image, cv2.COLOR_RGB2GRAY)
# normalize the image
norm_image = gray/255.0
#f_image = ft_image(norm_image)
# Display the images
f, (ax1,ax2) = plt.subplots(1, 2, figsize=(20,10))
ax1.imshow(image)
ax2.imshow(norm_image, cmap="gray")
#ax2.imshow(f_image, cmap="gray")
```

```
<matplotlib.image.AxesImage at 0x1f0a80dff10>
```



split_pages(gray,c=1)

213.9217389547414

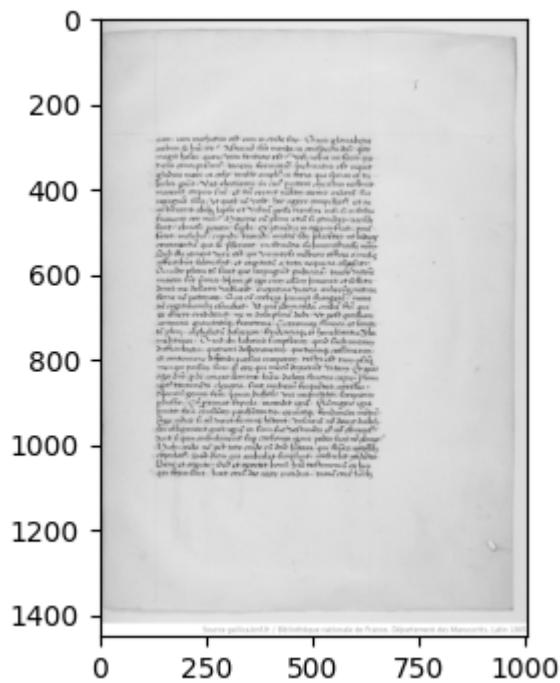
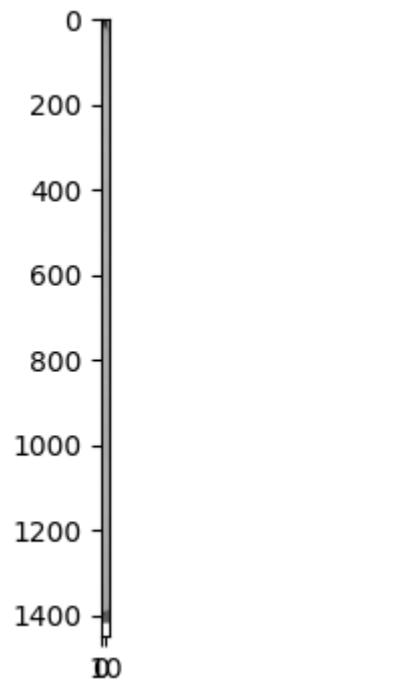
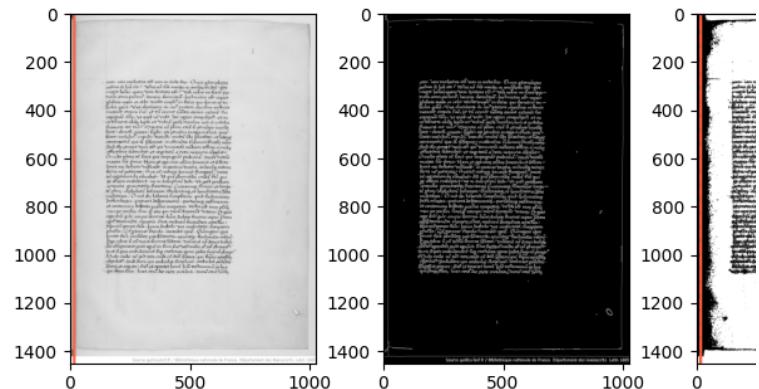
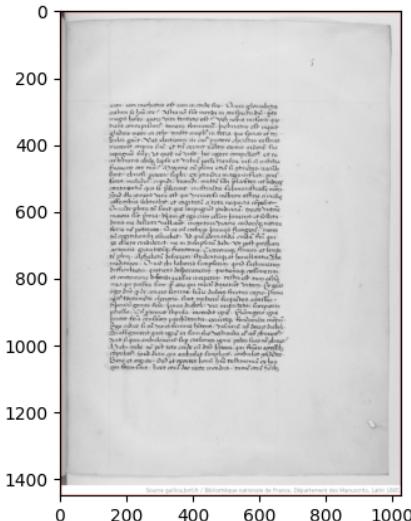
6856 candidat lines

[1. 1. 1. ... -0.9271838 -0.9612617 -0.9659258]

0

1450 pixels calculated on final candidate line

1.0 0.0 18 0

 $r = 0.9793103448275862$ with split at 18 $r_{\max} = 0.9793103448275862$ 

(18,

```
[array([[120, 91, 80, ..., 191, 195, 197],
       [131, 117, 87, ..., 185, 190, 192],
       [115, 126, 97, ..., 191, 188, 189],
       ...,
       [255, 255, 255, ..., 255, 255, 255],
       [255, 255, 255, ..., 255, 255, 255],
       [255, 255, 255, ..., 255, 255, 255]], dtype=uint8),
```

array([[200, 203, 205, ..., 229, 229, 229],

[200, 203, 205, ..., 229, 229, 229],

#plt.hist(gray.flatten(), bins=50)

hist, vals = np.histogram(gray, bins=50)

print('hist =', hist, np.sum(hist), len(hist))

print('vals =', vals, len(vals))

i = np.argmax(hist)

```
print(vals[i],vals[i+1])
xy = [(x,y) for y in range(len(gray)) for x in range(len(gray[0])) if (gray[y,x]>=vals[i])
xy = np.array(xy)
plt.figure(figsize=(20,10))
plt.imshow(gray,cmap='gray')
plt.scatter(xy[:,0],xy[:,1],color='teal',marker='.',s=1)
```

```

hist = [      3       7      13      62     103      329      497     1081     1384     2596
    2831    4194    5045    4426    5523    4355    5070    3716    4426    3350
    4059    3409    4180    3051    3589    3625    2878    3661    3207    4266
    4023    5385    4705    7596    9428   17820   16546   16076   27414   44495
  114024  175474  392877  346922  165007  19536   3352    923    2166  26095] 1484800 50
vals = [ 28.    32.54  37.08  41.62  46.16  50.7   55.24  59.78  64.32  68.86
    73.4   77.94  82.48  87.02  91.56  96.1   100.64 105.18 109.72 114.26
  118.8   123.34 127.88 132.42 136.96 141.5   146.04 150.58 155.12 159.66
  164.2   168.74 173.28 177.82 182.36 186.9   191.44 195.98 200.52 205.06

```

```
image = cv2.imread("../ImagesCodicologie/Corpus de jeu/Julien/Corpus B/Fr_578_110v.jpg")
```

```

# Change color to RGB (from BGR)
image = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
# convert to grayscale
gray = cv2.cvtColor(image, cv2.COLOR_RGB2GRAY)
# normalize the image
norm_image = gray/255.0
#f_image = ft_image(norm_image)
# Display the images
f, (ax1,ax2) = plt.subplots(1, 2, figsize=(20,10))
ax1.imshow(image)
ax2.imshow(norm_image, cmap="gray")
#ax2.imshow(f_image, cmap="gray")

```

```
<matplotlib.image.AxesImage at 0x1f0ac72be90>
```



```
#plt.hist(gray.flatten(),bins=50)
```

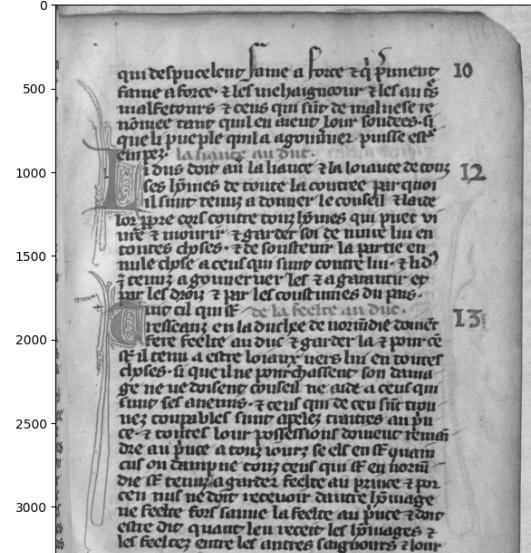
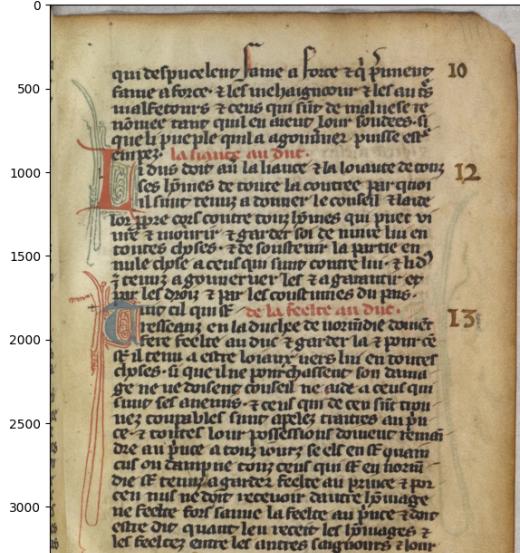
```
hist,vals = np.histogram(gray,bins=50)
print('hist =',hist,np.sum(hist),len(hist))
print('vals =',vals,len(vals))
i = np.argmax(hist)
print(vals[i],vals[i+1])
xy = [(x,y) for y in range(len(gray)) for x in range(len(gray[0])) if (gray[y,x]>=vals[i])
xy = np.array(xy)
plt.figure(figsize=(20,10))
plt.imshow(gray,cmap='gray')
plt.scatter(xy[:,0],xy[:,1],color='tomato',marker='.',s=1)
```

```
hist = [ 5513  3338  3628  3744  4534  7187  10471  14183  19851  26954
        40096  39125  44391  46010  47163  48210  47862  46769  45708  44280
        51341  41824  39892  38920  38722  38401  38475  39338  39356  39470
        49024  41907  42508  44255  47389  49123  54005  63648  76396  102068
        196712 247619 348657 554794 821736 979793 698180 278458 198874 5894] 5855796 50
vals = [ 0.    5.1   10.2  15.3  20.4  25.5  30.6  35.7  40.8  45.9  51.   56.1
        61.2   66.3  71.4  76.5  81.6  86.7  91.8  96.9  102.  107.1  112.2  117.3
        122.4  127.5  132.6  137.7  142.8  147.9  153.  158.1  163.2  168.3  173.4  178.5
        183.6  188.7  193.8  198.9  204.  209.1  214.2  219.3  224.4  229.5  234.6  239.7
        244.8  249.9  255. ] 51
229.4999999999997 234.6

image = cv2.imread("../ImagesCodicologie/Corpus de jeu/Julien/Fr_5963.jpg")

# Change color to RGB (from BGR)
image = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
# convert to grayscale
gray = cv2.cvtColor(image, cv2.COLOR_RGB2GRAY)
# normalize the image
norm_image = gray/255.0
#f_image = ft_image(norm_image)
# Display the images
f, (ax1,ax2) = plt.subplots(1, 2, figsize=(20,10))
ax1.imshow(image)
ax2.imshow(norm_image, cmap="gray")
#ax2.imshow(f_image, cmap="gray")
```

<matplotlib.image.AxesImage at 0x1f14c687fd0>



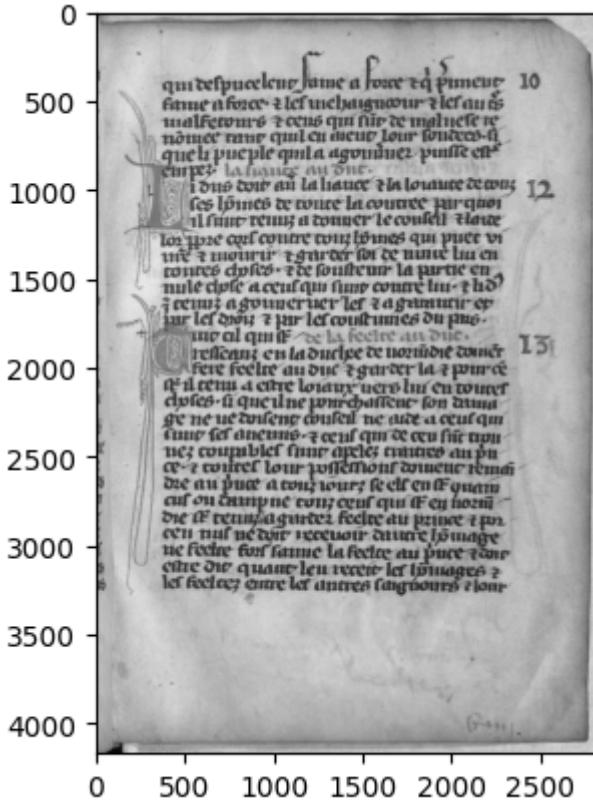
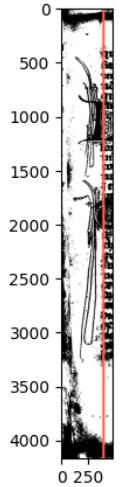
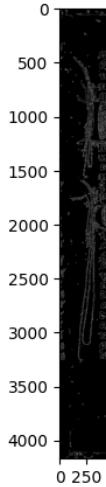
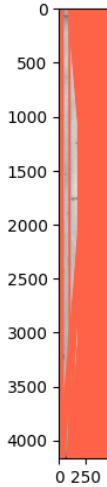
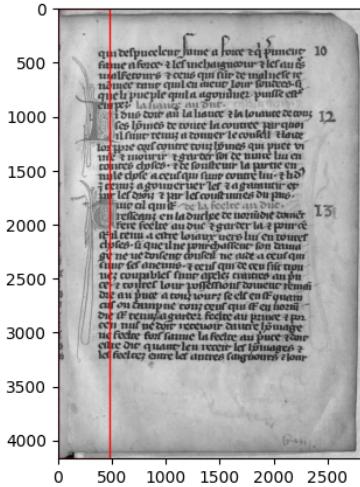
Pour l'exemple au dessous, l'arrière plan se voit à droite mais pas à gauche, et l'algo les coupe bien.



```
split_pages(gray,dir='l',c=6)
split_pages(gray,dir='r',c=6)
#split_pages(gray,dir='u')
#split_pages(gray,dir='d')
#split_pages(gray,dir='m')
```

168.22407046990395

1199 candidat lines amongst 1201 lines found

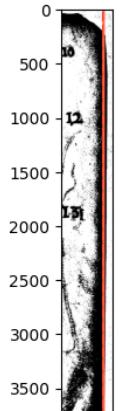
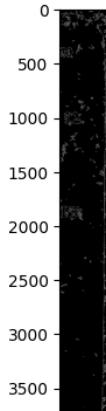
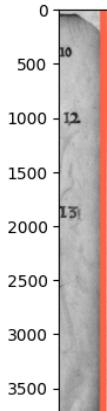
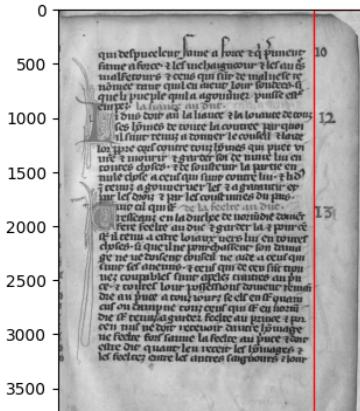
 $r \max = 0.5735188294555049$ 

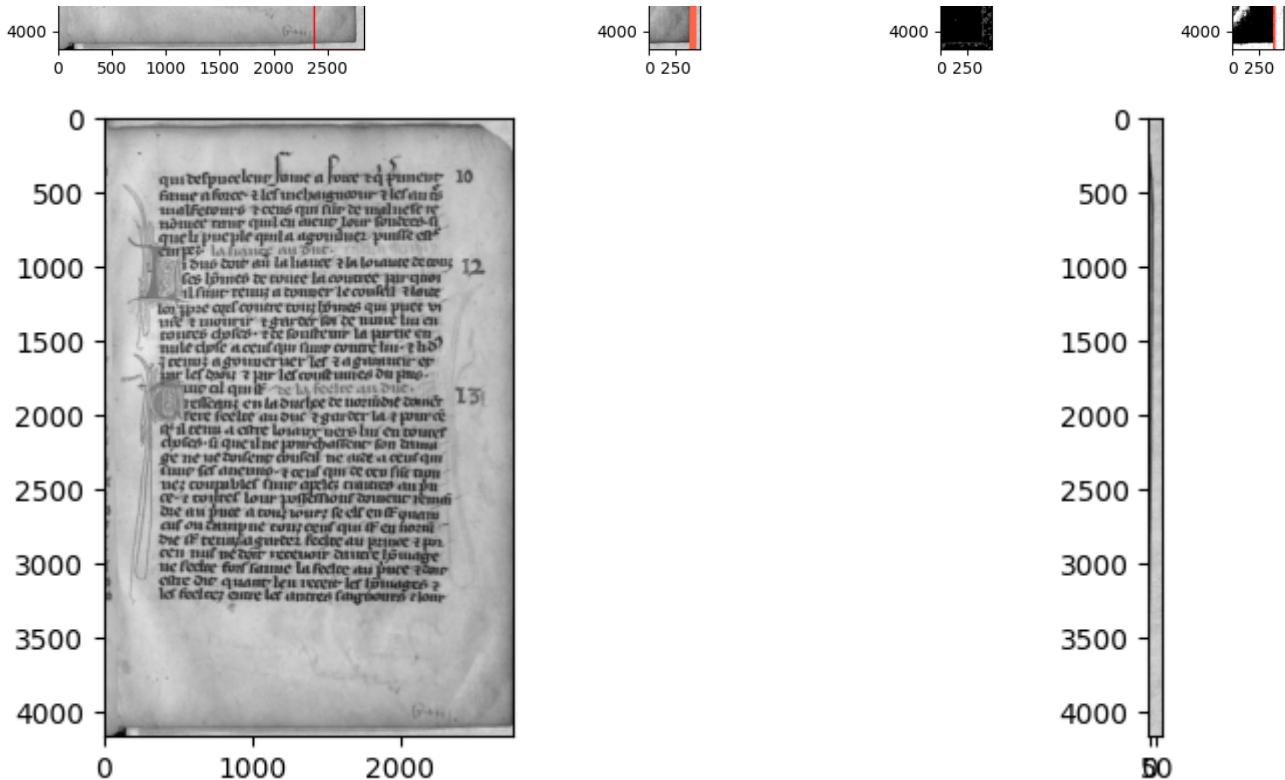
165.03485845394934

170 candidat lines amongst 173 lines found

4169 pixels calculated on final candidate line

1.0 0.0 391 0

 $r = 0.8944591029023746$ with split at 2761 $r \max = 0.8944591029023746$ 



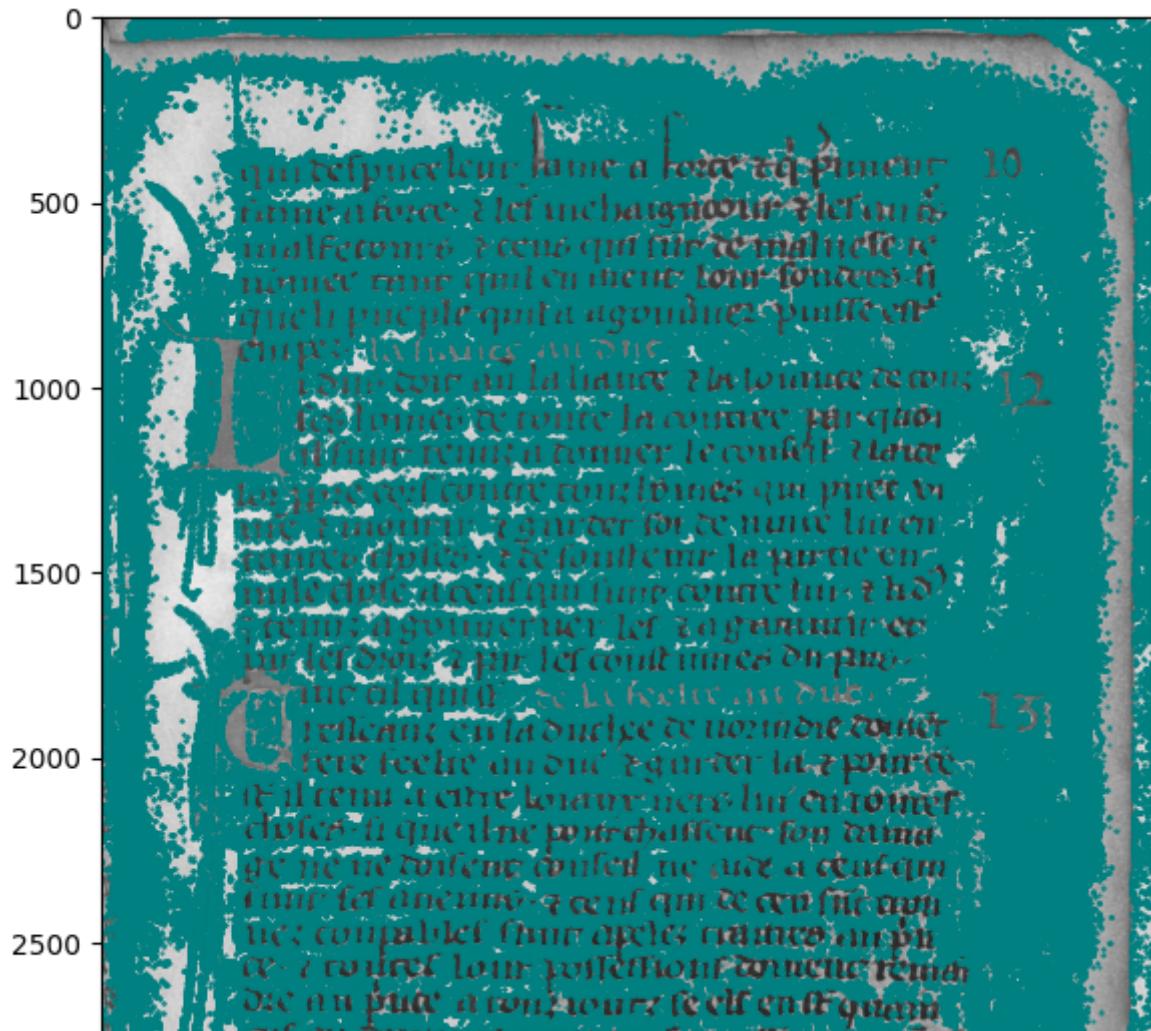
```
(2761,
[array([[ 53,  56,  53, ..., 200, 204, 206],
       [ 49,  51,  50, ..., 201, 205, 204],
       [ 34,  35,  41, ..., 202, 207, 203],
       ...,
       [122, 123, 124, ..., 191, 188, 186],  

#plt.hist(gray.flatten(),bins=50)
hist,vals = np.histogram(gray,bins=50)
print('hist =',hist,np.sum(hist),len(hist))
print('vals =',vals,len(vals))
i = np.argmax(hist)
print(vals[i],vals[i+1])
xy = [(x,y) for y in range(len(gray)) for x in range(len(gray[0])) if (gray[y,x]>=vals[i])
xy = np.array(xy)
plt.figure(figsize=(20,10))
plt.imshow(gray,cmap='gray')
plt.scatter(xy[:,0],xy[:,1],color='teal',marker='.',s=1)
```

```

hist = [    29      955     682     235     548     3235    13030    56439   146732
 203978  302843  300082  207542  193568  130497  76726   79954    71224
  53966   68669    72402    66165    97557   111741   97129   124926   136710
 163861  153502  234345  294391  278345  447962  642238  665676  1072620
1295345 1065434 1166853  858024  445194  275355  110227  38972    21237
  8059    1250     170      9      3] 11856636 50
vals = [  3.     7.68  12.36  17.04  21.72  26.4    31.08  35.76  40.44  45.12
 49.8   54.48  59.16  63.84  68.52  73.2    77.88  82.56  87.24  91.92
 96.6  101.28 105.96 110.64 115.32 120.    124.68 129.36 134.04 138.72
143.4  148.08 152.76 157.44 162.12 166.8  171.48 176.16 180.84 185.52
190.2  194.88 199.56 204.24 208.92 213.6  218.28 222.96 227.64 232.32
237.  ] 51
171.48 176.16
<matplotlib.collections.PathCollection at 0x1f0a175bf10>

```



```
plt.imshow(black_white(gray), cmap='gray')
```

<matplotlib.image.AxesImage at 0x1f0b577bd10>

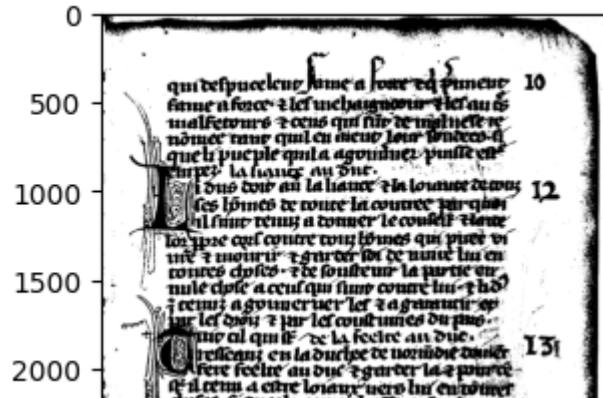


image = cv2.imread("../ImagesCodicologie/Corpus de jeu/Emilie Cottreau-Gabillet/BnF Lat 7

```
# Change color to RGB (from BGR)
image = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
# convert to grayscale
gray = cv2.cvtColor(image, cv2.COLOR_RGB2GRAY)
# normalize the image
norm_image = gray/255.0
#f_image = ft_image(norm_image)
# Display the images
f, (ax1,ax2) = plt.subplots(1, 2, figsize=(20,10))
ax1.imshow(image)
ax2.imshow(norm_image, cmap="gray")
#ax2.imshow(f_image, cmap="gray")
```

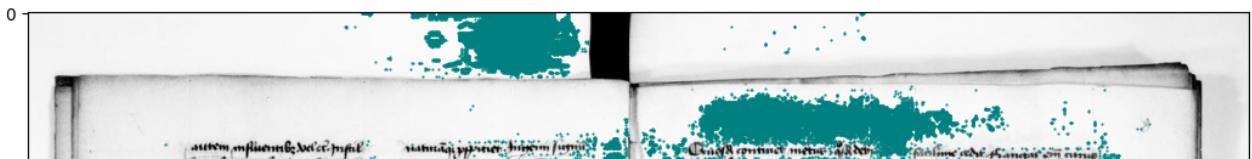
```
<matplotlib.image.AxesImage at 0x1f0b417cbd0>
```



```
#plt.hist(gray.flatten(),bins=50)
hist,vals = np.histogram(gray,bins=50)
print('hist =',hist,np.sum(hist),len(hist))
print('vals =',vals,len(vals))
i = np.argmax(hist)
print(vals[i],vals[i+1])
xy = [(x,y) for y in range(len(gray)) for x in range(len(gray[0])) if (gray[y,x]>=vals[i])
xy = np.array(xy)
plt.figure(figsize=(20,10))
plt.imshow(gray,cmap='gray')
plt.scatter(xy[:,0],xy[:,1],color='teal',marker='.',s=1)
```

```
hist = [ 12915  4510   3795   3658   3728   3925   4041   4166   4230   4389
        5384  4653  4876  4792  4948  4945  4951  4985  4982  5063
        6263  5255  5327  5547  5929  6126  6090  6040  6386  6448
        8010  6864  7170  7402  7916  8298  8724  9472  10536 11613
       16430  16766 21613 28005 37016 55909 91559 165477 285531 487646] 1450304 50
vals = [ 0.    5.1  10.2  15.3  20.4  25.5  30.6  35.7  40.8  45.9  51.   56.1
        61.2  66.3  71.4  76.5  81.6  86.7  91.8  96.9 102.  107.1 112.2 117.3
       122.4 127.5 132.6 137.7 142.8 147.9 153.  158.1 163.2 168.3 173.4 178.5
       183.6 188.7 193.8 198.9 204.  209.1 214.2 219.3 224.4 229.5 234.6 239.7
      244.8 249.9 255. ] 51
249.8999999999998 255.0
```

<matplotlib.collections.PathCollection at 0x1f0d1a25210>

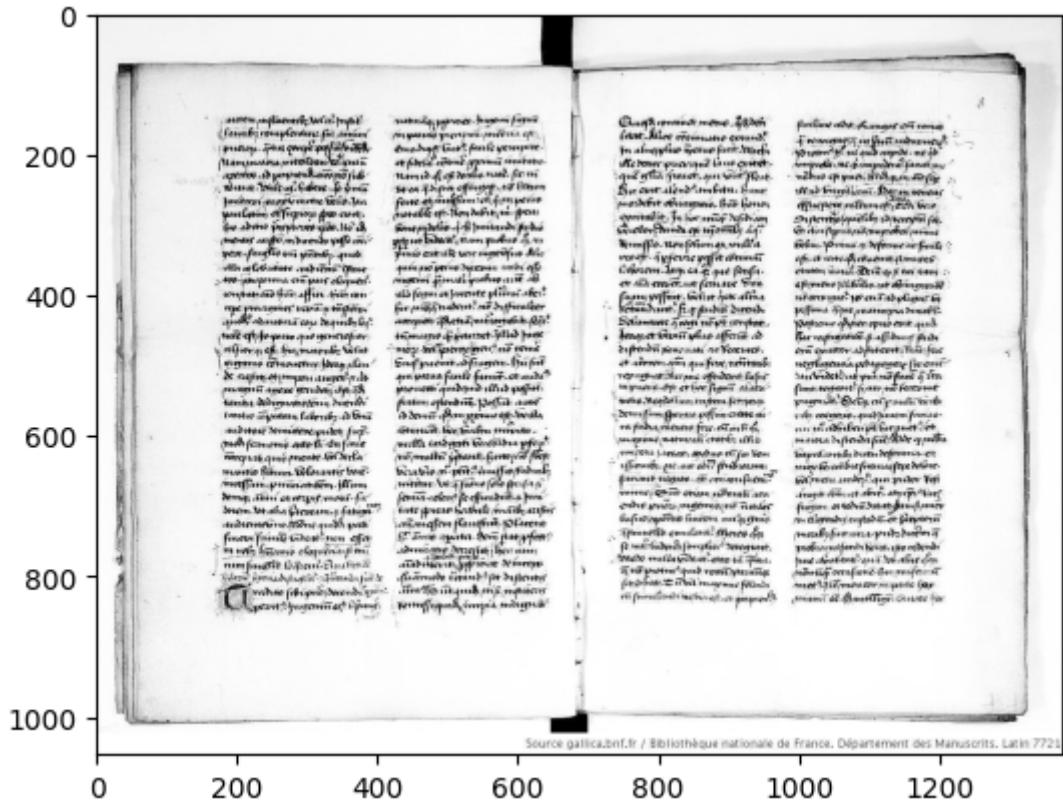


```
w = np.array([[0,-1,0],[-1,5,-1],[0,-1,0]])
```

```
cvd = convolve2d(gray,w)
```

```
plt.imshow(cvd,cmap='gray')
```

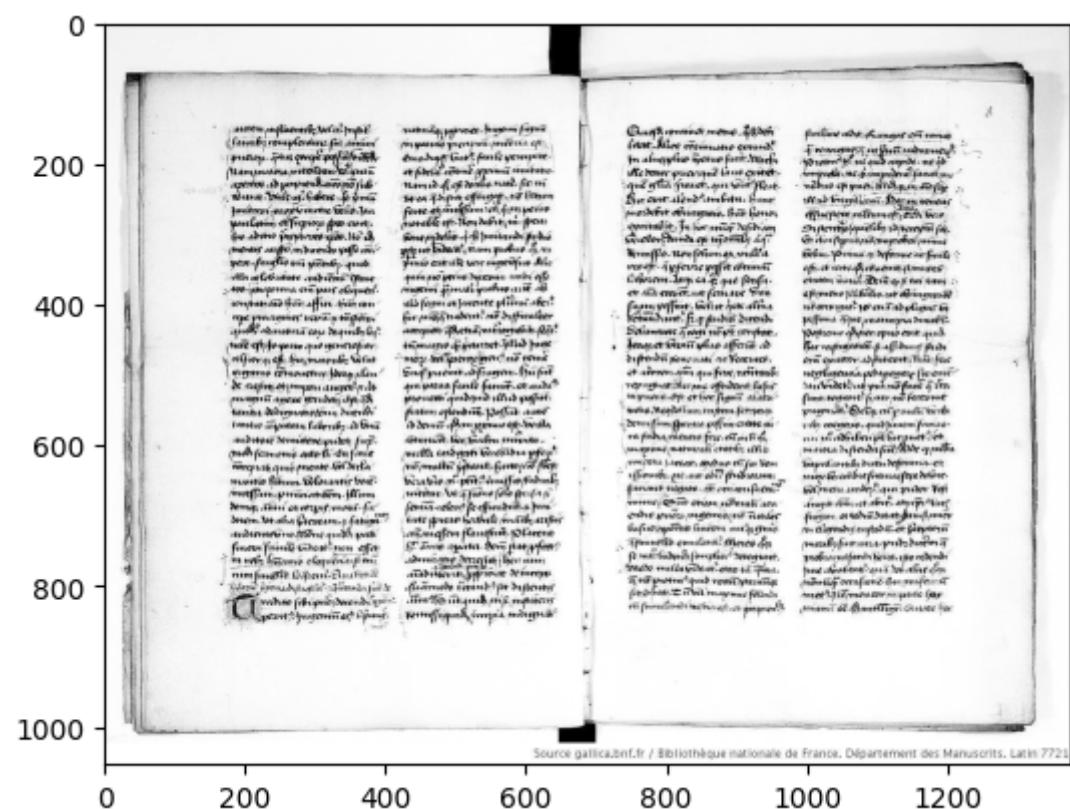
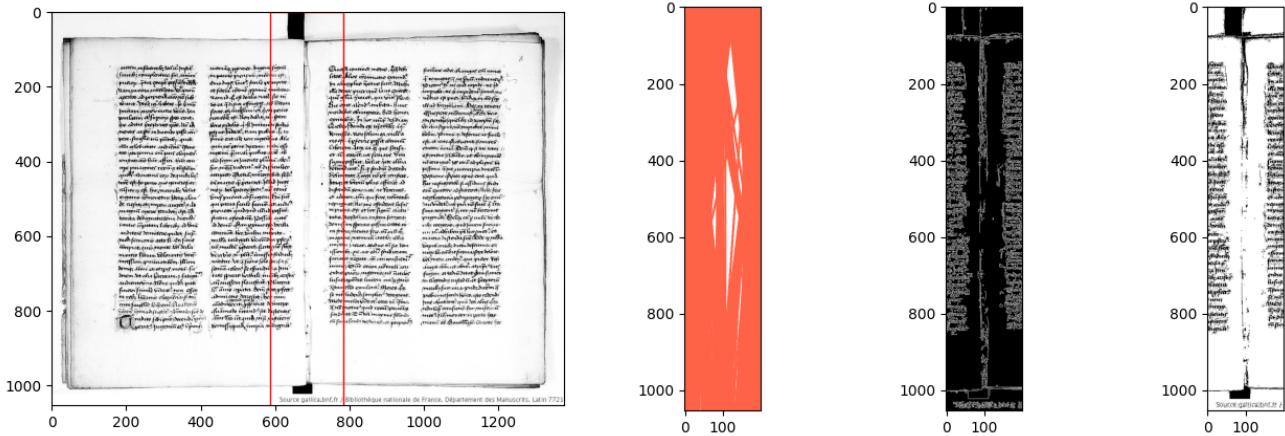
<matplotlib.image.AxesImage at 0x1f0a180b850>



```
split_pages(cvd)
```

223.57856736242886

282 candidat lines



On n'arrive pas à couper car le milieu est trop clair...

```
image = cv2.imread("../ImagesCodicologie/Corpus de jeu/Emilie Cottreau-Gabillet/BnF Lat 7

# Change color to RGB (from BGR)
image = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
# convert to grayscale
gray = cv2.cvtColor(image, cv2.COLOR_RGB2GRAY)
# normalize the image
norm_image = gray/255.0
#f_image = ft_image(norm_image)
# Display the images
f, (ax1,ax2) = plt.subplots(1, 2, figsize=(20,10))
```

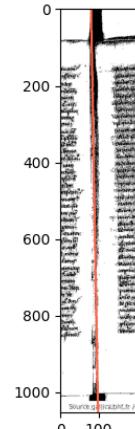
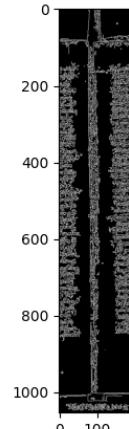
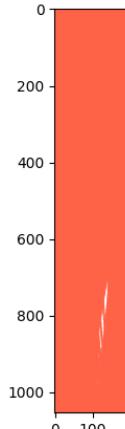
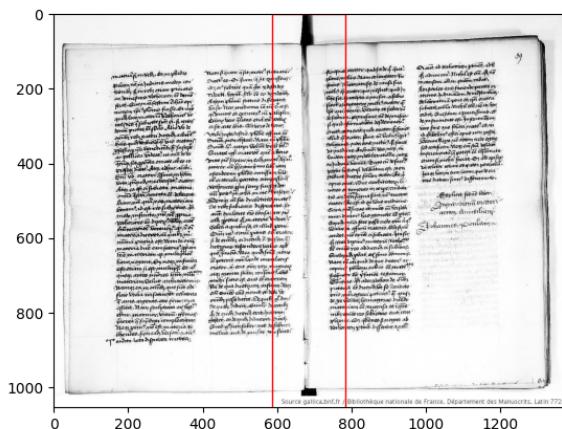
```
ax1.imshow(image)
ax2.imshow(norm_image, cmap="gray")
#ax2.imshow(f_image, cmap="gray")
```

```
<matplotlib.image.AxesImage at 0x1f0a80e6950>
```



```
split_pages(gray, cvd=True)
```

218.66951138519923
 562 candidat lines
 $r_{\max} = 0.6574952561669829$



On n'arrive pas à couper car le milieu est trop clair...

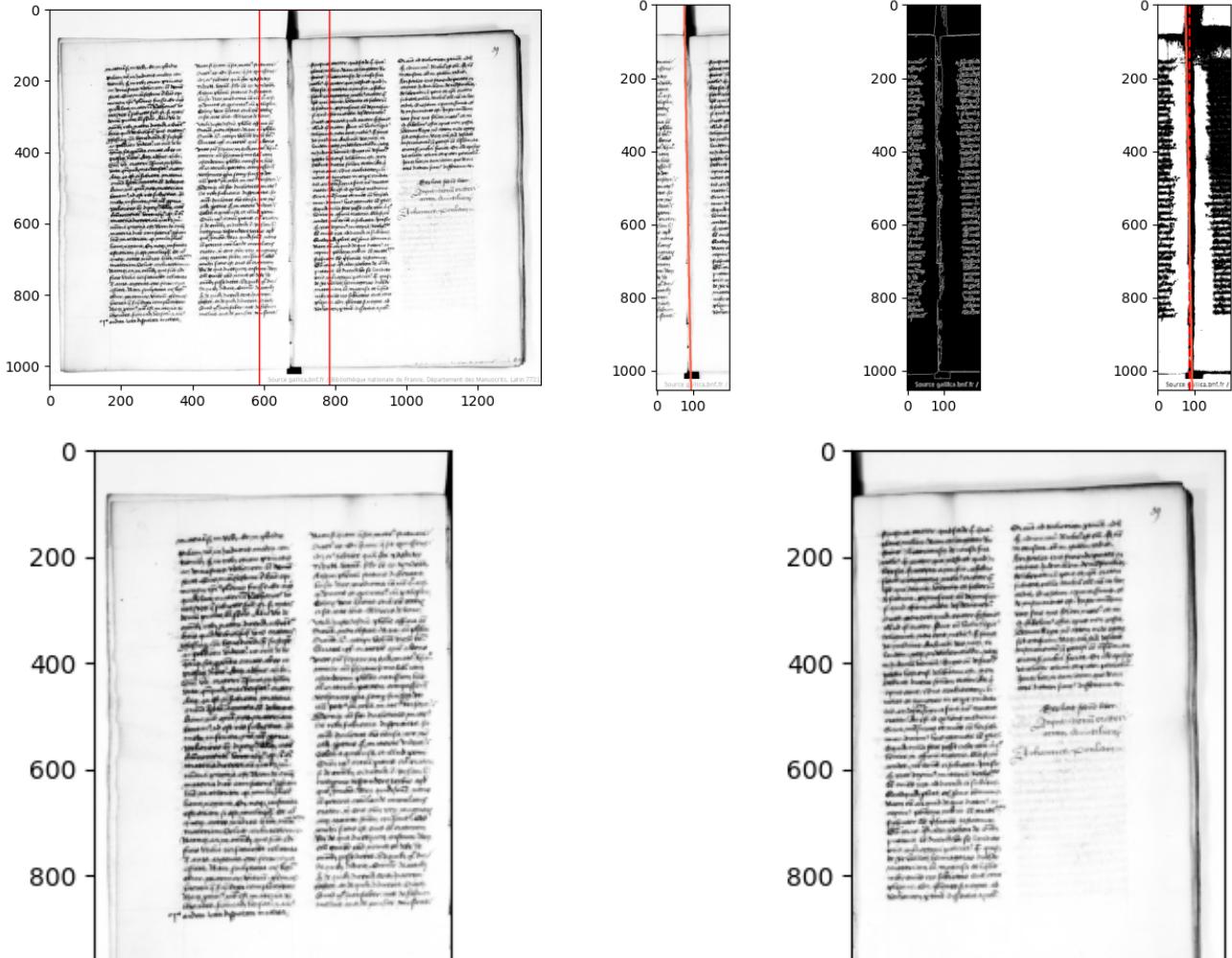


Les exemples où cela marche bien au dessous

0 200 400 600 800 1000 1200

`split_pages(norm_image)`

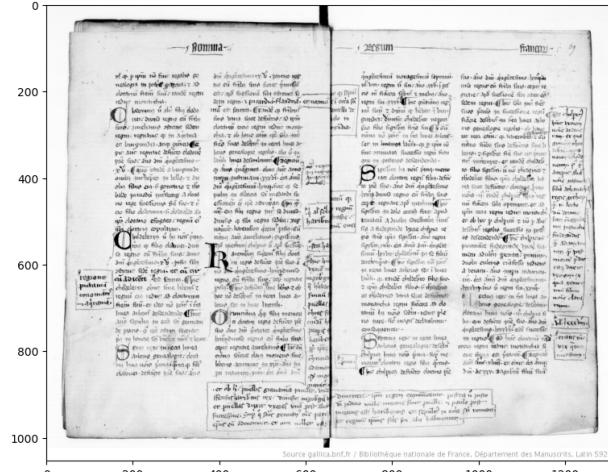
221.50518026565464
 199 candidat lines
 1054 pixels calculated on final candidate line
 $-77.0 \ 3.1241393 \ -0.9998477 \ 0.017452458 \ 76 \ -1$
 $r = 0.9184060721062619$ with split at 673
 $r_{\max} = 0.9184060721062619$



```
image = cv2.imread("../ImagesCodicologie/Corpus de jeu/Emilie Cottreau-Gabillet/BnF Lat 5
```

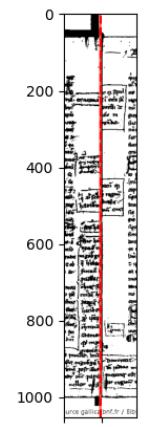
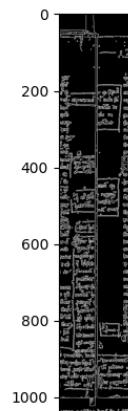
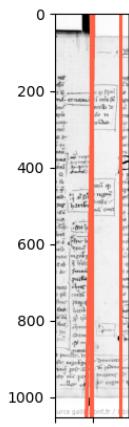
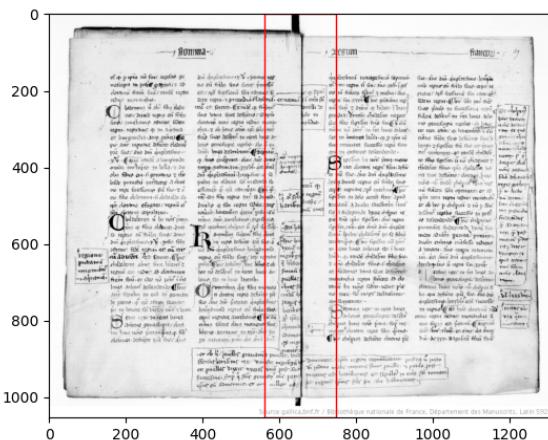
```
# Change color to RGB (from BGR)
image = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
# convert to grayscale
gray = cv2.cvtColor(image, cv2.COLOR_RGB2GRAY)
# normalize the image
norm_image = gray/255.0
#f_image = ft_image(norm_image)
# Display the images
f, (ax1,ax2) = plt.subplots(1, 2, figsize=(20,10))
ax1.imshow(image)
ax2.imshow(norm_image, cmap="gray")
#ax2.imshow(f_image, cmap="gray")
```

<matplotlib.image.AxesImage at 0x1f0a7d0a3d0>



split_pages(norm_image)

407 candidat lines
 1054 pixels calculated
 95.0 0.0 1.0 0.0 95 0
 $r = 0.9079696394686907$ with split at 656



$r_{\max} = 0.9079696394686907$

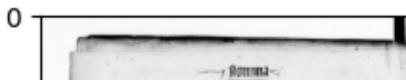
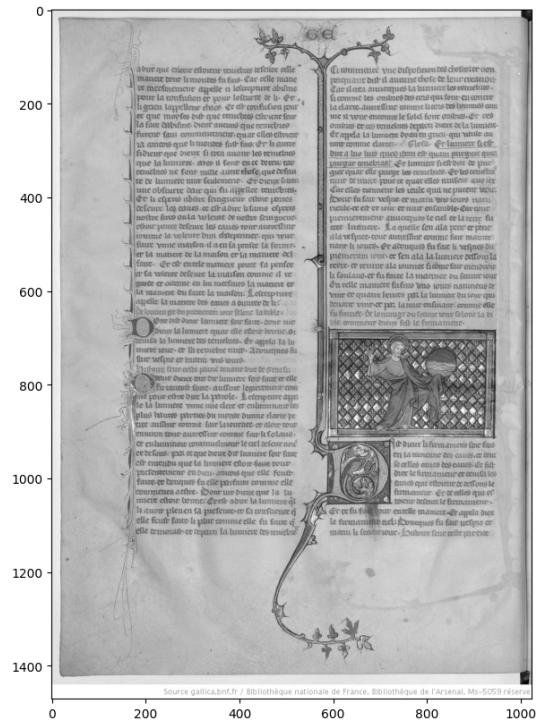


image = cv2.imread("../ImagesCodicologie/Corpus de jeu/Emilie Cottereau-Gabillet/Arsenal 5

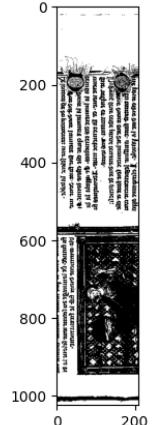
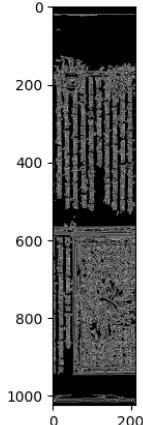
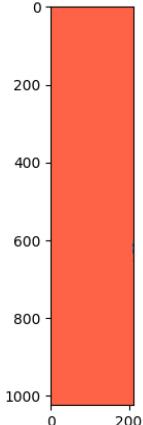
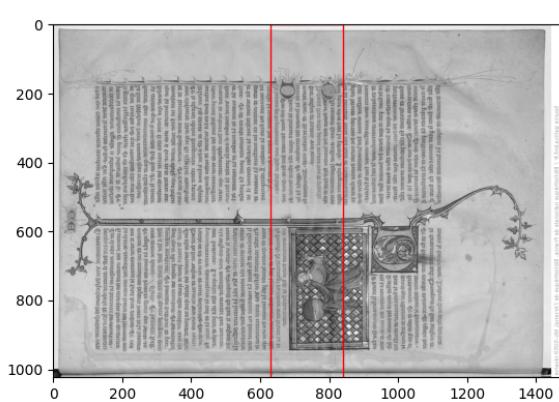
```
# Change color to RGB (from BGR)
image = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
# convert to grayscale
gray = cv2.cvtColor(image, cv2.COLOR_RGB2GRAY)
# normalize the image
norm_image = gray/255.0
#f_image = ft_image(norm_image)
# Display the images
f, (ax1,ax2) = plt.subplots(1, 2, figsize=(20,10))
ax1.imshow(image)
ax2.imshow(norm_image, cmap="gray")
#ax2.imshow(f_image, cmap="gray")
```

<matplotlib.image.AxesImage at 0x1f0a7cd4950>



split_pages(gray)

1182 candidat lines



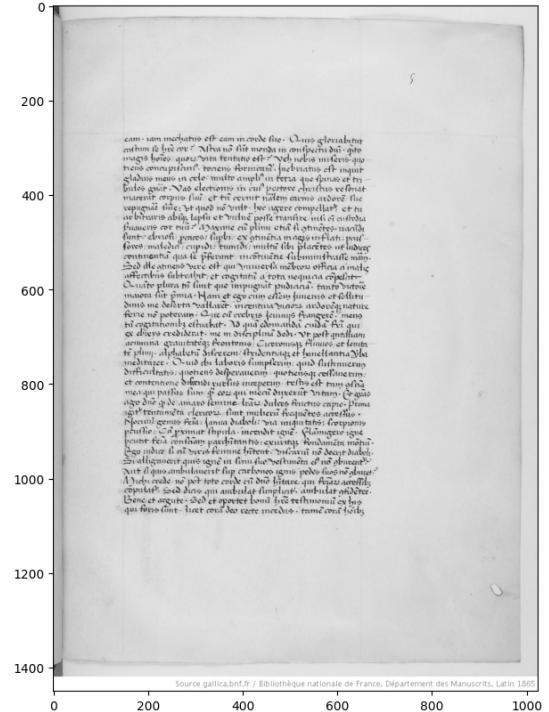
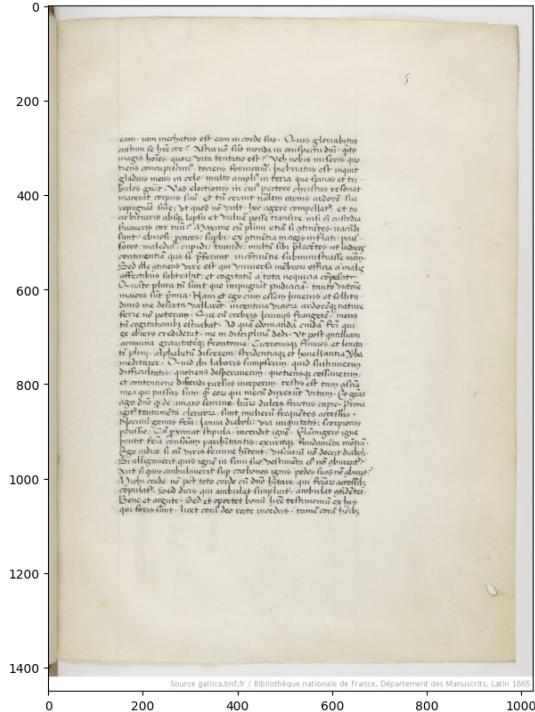
$r_{\max} = 0.7818574514038876$



```
image = cv2.imread("../ImagesCodicologie/Corpus de jeu/Emilie Cottreau-Gabillet/BnF Lat 1
```

```
# Change color to RGB (from BGR)
image = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
# convert to grayscale
gray = cv2.cvtColor(image, cv2.COLOR_RGB2GRAY)
# normalize the image
norm_image = gray/255.0
#f_image = ft_image(norm_image)
# Display the images
f, (ax1,ax2) = plt.subplots(1, 2, figsize=(20,10))
ax1.imshow(image)
ax2.imshow(norm_image, cmap="gray")
#ax2.imshow(f_image, cmap="gray")
```

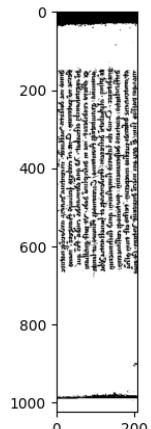
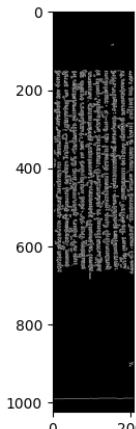
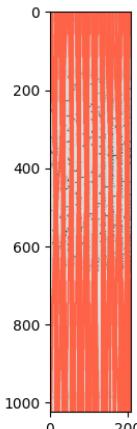
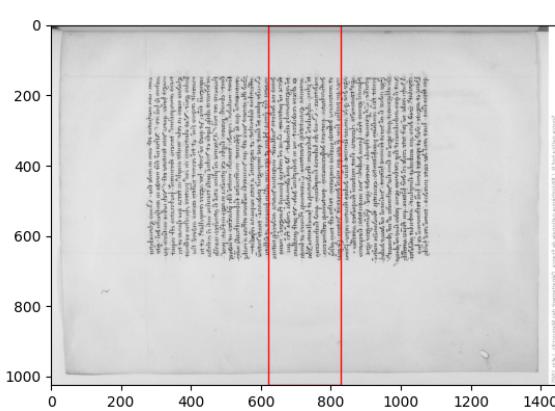
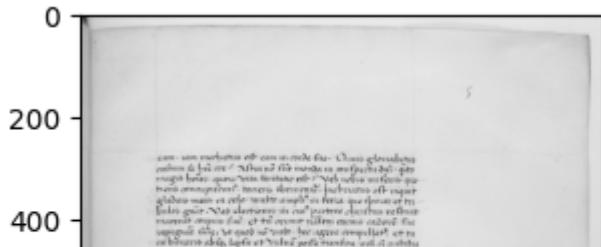
<matplotlib.image.AxesImage at 0x1f0a80e80d0>



split_pages(norm_image)

206.77237642728366

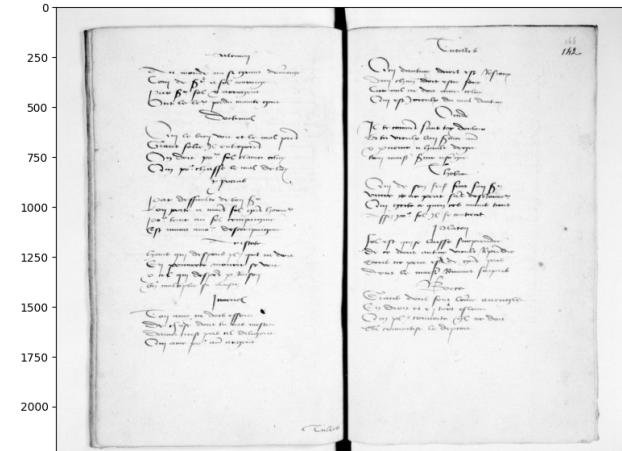
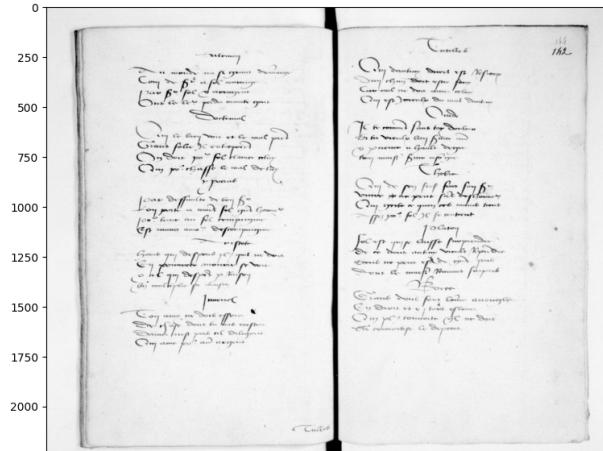
69 candidat lines

 $r_{\max} = 0.4025670945157527$ 

```
image = cv2.imread("../ImagesCodicologie/Corpus de jeu/Julien/Fr_1370_141v_142r.jpg")
```

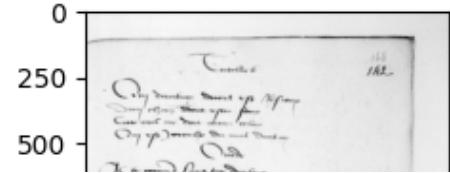
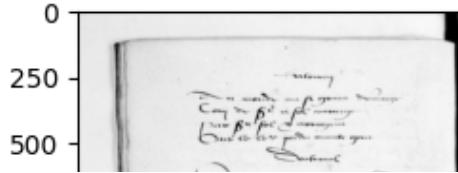
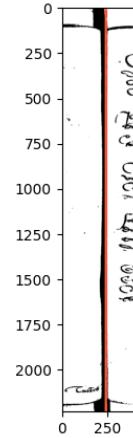
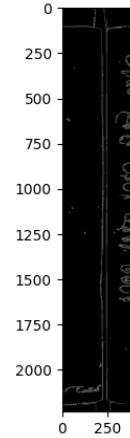
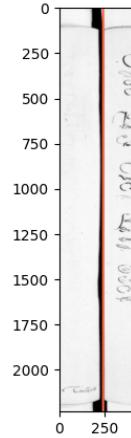
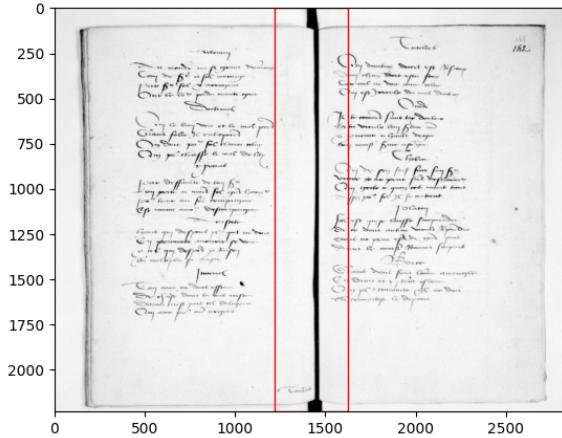
```
# Change color to RGB (from BGR)
image = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
# convert to grayscale
gray = cv2.cvtColor(image, cv2.COLOR_RGB2GRAY)
# normalize the image
norm_image = gray/255.0
#f_image = ft_image(norm_image)
# Display the images
f, (ax1,ax2) = plt.subplots(1, 2, figsize=(20,10))
ax1.imshow(image)
ax2.imshow(norm_image, cmap="gray")
#ax2.imshow(f_image, cmap="gray")
```

<matplotlib.image.AxesImage at 0x1f0b331c390>



split_pages(gray)

223.7907172273298
 37 candidat lines amongst 39 lines found
 2233 pixels calculated on final candidate line
 $1.0 \ 0.0 \ 243 \ 0$
 $r = 0.9650694133452754$ with split at 1461
 $r \max = 0.9650694133452754$



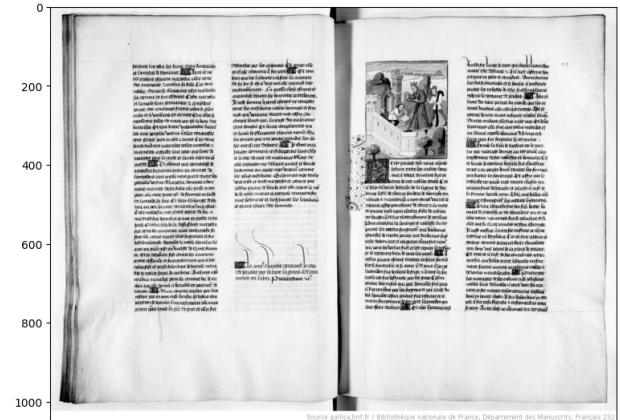
```
image = cv2.imread("../ImagesCodicologie/Corpus de jeu/Emilie Cottreau-Gabillet/BnF Fr 23
```

```
# Change color to RGB (from BGR)
image = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
# convert to grayscale
gray = cv2.cvtColor(image, cv2.COLOR_RGB2GRAY)
# normalize the image
norm_image = gray/255.0
#f_image = ft_image(norm_image)
# Display the images
f, (ax1,ax2) = plt.subplots(1, 2, figsize=(20,10))
ax1.imshow(image)
ax2.imshow(norm_image, cmap="gray")
#ax2.imshow(f_image, cmap="gray")
```

<matplotlib.image.AxesImage at 0x1f0b6bff390>



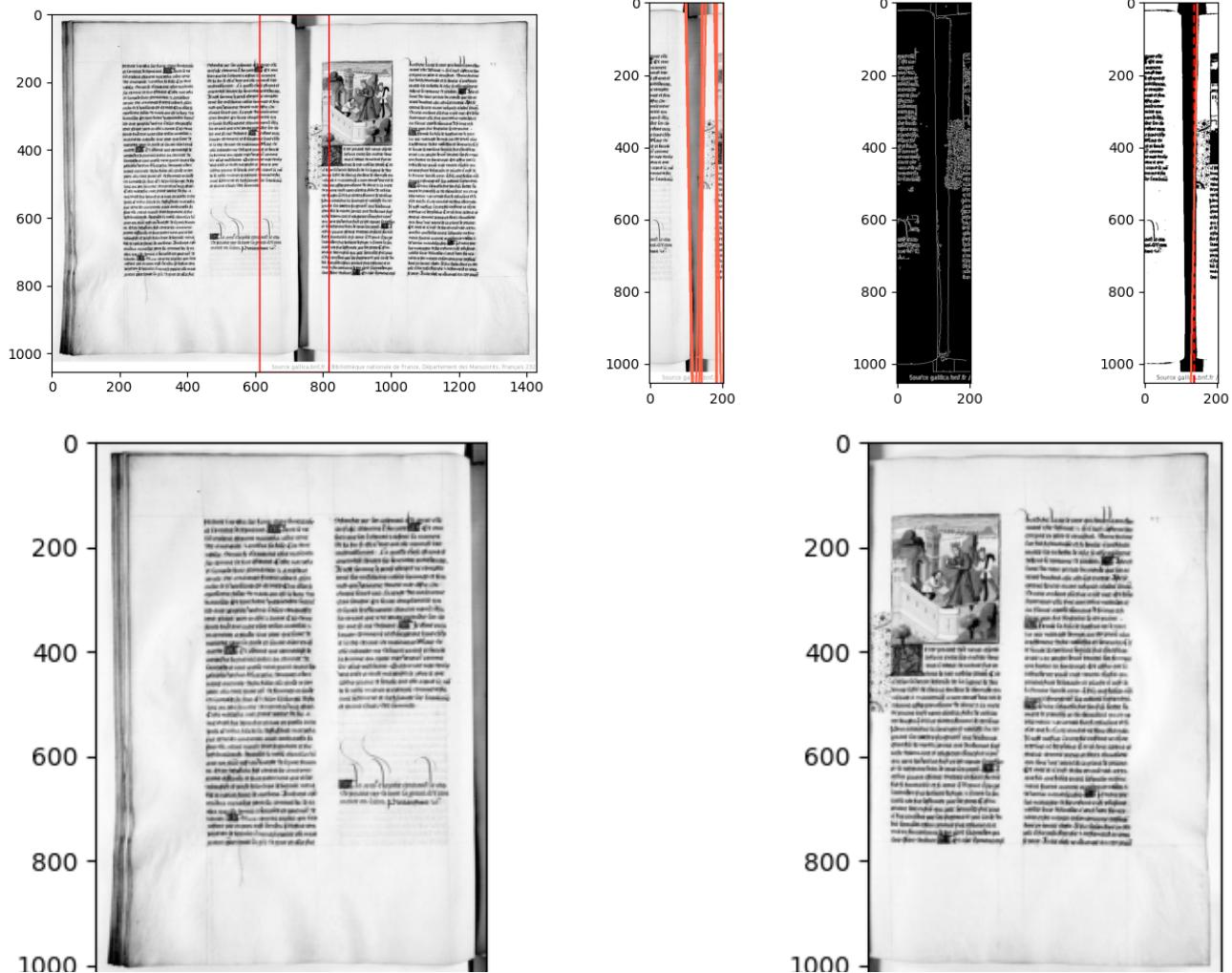
Source gallica.bnf.fr / Bibliothèque nationale de France, Département des Manuscrits, Français 23r



Source gallica.bnf.fr / Bibliothèque nationale de France, Département des Manuscrits, Français 23v

split_pages(norm_image)

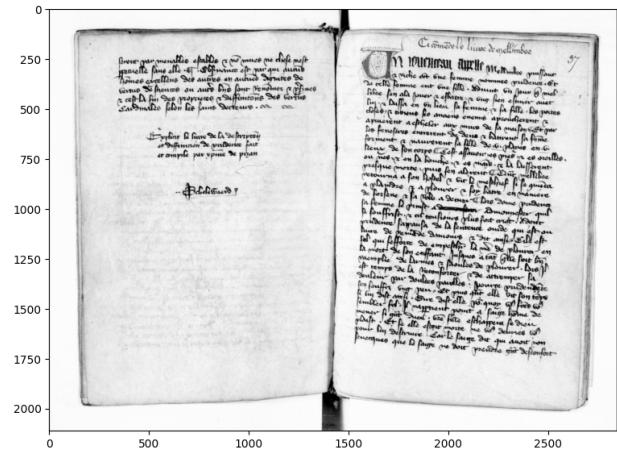
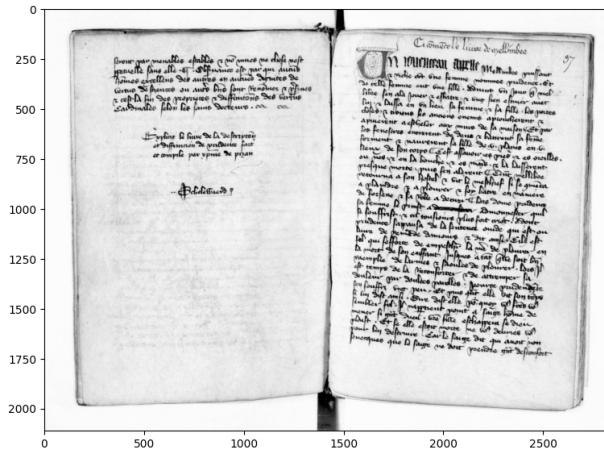
204.22040079603832
 71 candidat lines amongst 71 lines found
 1054 pixels calculated on final candidate line
 $0.9998477 \ 0.017452406 \ 148 \ 2$
 $r = 0.976280834914611$ with split at 750
 $r_{\text{max}} = 0.976280834914611$



```
image = cv2.imread("../ImagesCodicologie/Corpus de jeu/Julien/Fr_2240_36v_37r.jpg")
```

```
# Change color to RGB (from BGR)
image = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
# convert to grayscale
gray = cv2.cvtColor(image, cv2.COLOR_RGB2GRAY)
# normalize the image
norm_image = gray/255.0
#f_image = ft_image(norm_image)
# Display the images
f, (ax1,ax2) = plt.subplots(1, 2, figsize=(20,10))
ax1.imshow(image)
ax2.imshow(norm_image, cmap="gray")
#ax2.imshow(f_image, cmap="gray")
```

<matplotlib.image.AxesImage at 0x1f0b6acd4d0>



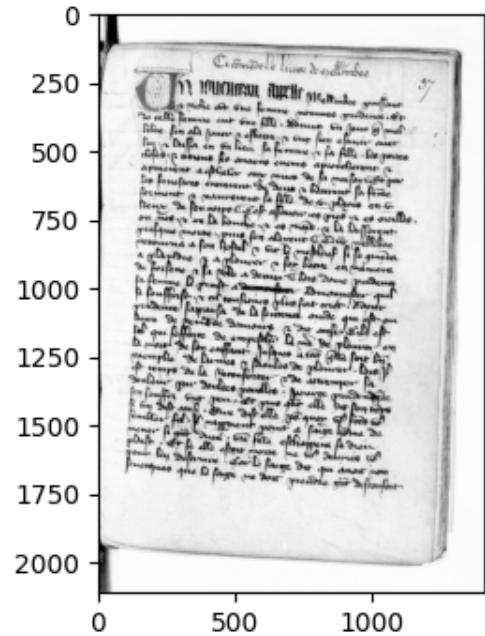
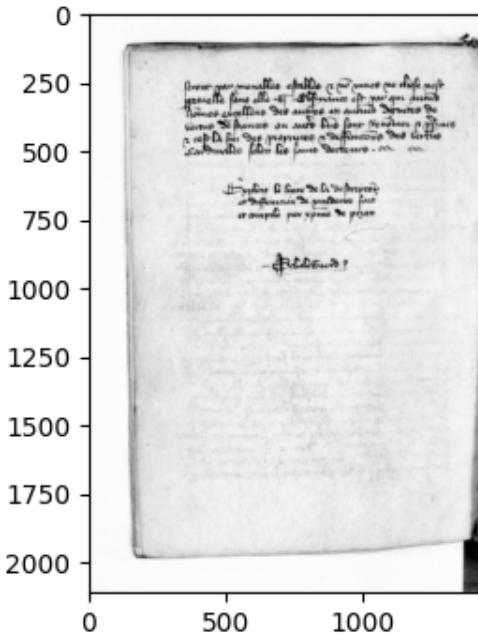
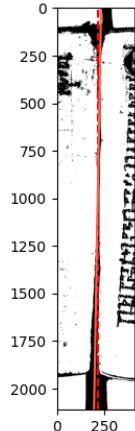
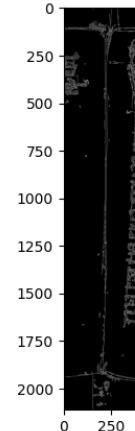
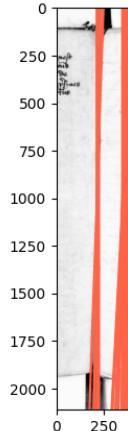
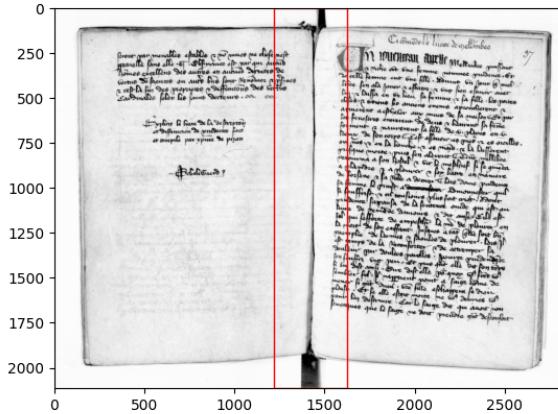
split_pages(norm_image)

220.28284685086155

159 candidat lines amongst 160 lines found

2112 pixels calculated on final candidate line

0.9998477 0.017452406 232 4

 $r = 0.87926136363636$ with split at 1431 $r_{\max} = 0.87926136363636$ 

(1431,

```
[array([[0.97254902, 0.97254902, 0.97254902, ..., 0.95686275, 0.95686275,
       0.95686275],
      [0.97254902, 0.97254902, 0.97254902, ..., 0.95686275, 0.95686275,
       0.95686275],
      [0.97254902, 0.97254902, 0.97254902, ..., 0.96078431, 0.96078431,
       0.96078431],
      ...,
      [0.97647059, 0.97647059, 0.97647059, ..., 0.3254902 , 0.32941176,
```

image = cv2.imread("../ImagesCodicologie/Corpus de jeu/Emilie Cottreau-Gabillet/BnF Lat 5

Change color to RGB (from BGR)

image = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)

convert to grayscale

gray = cv2.cvtColor(image, cv2.COLOR_RGB2GRAY)

normalize the image

norm_image = gray/255.0

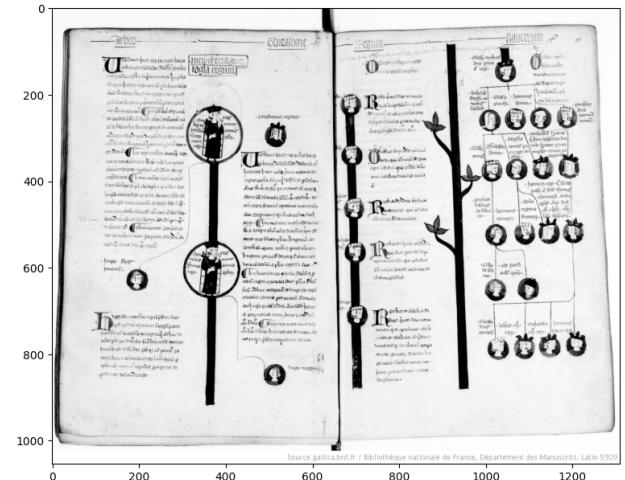
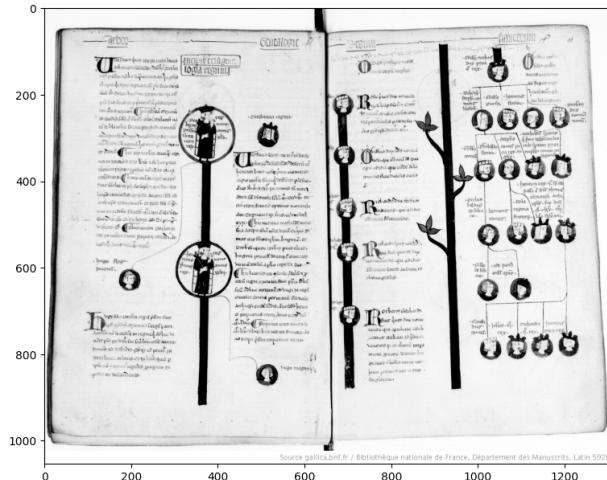
#f_image = ft_image(norm_image)

Display the images

f, (ax1,ax2) = plt.subplots(1, 2, figsize=(20,10))

```
ax1.imshow(image)
ax2.imshow(norm_image, cmap="gray")
#ax2.imshow(f_image, cmap="gray")
```

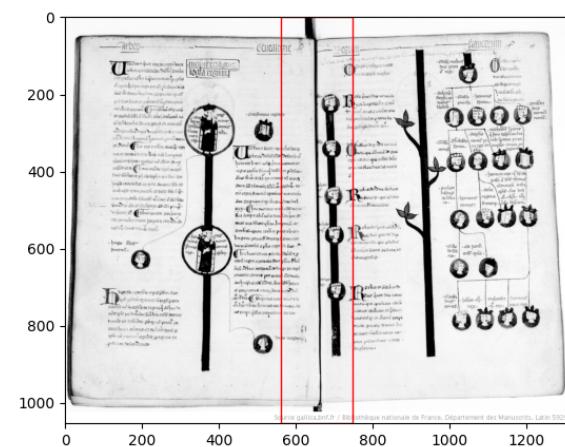
```
<matplotlib.image.AxesImage at 0x1f095cd6850>
```



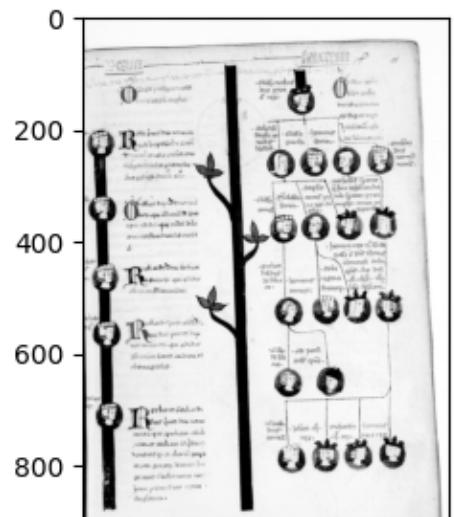
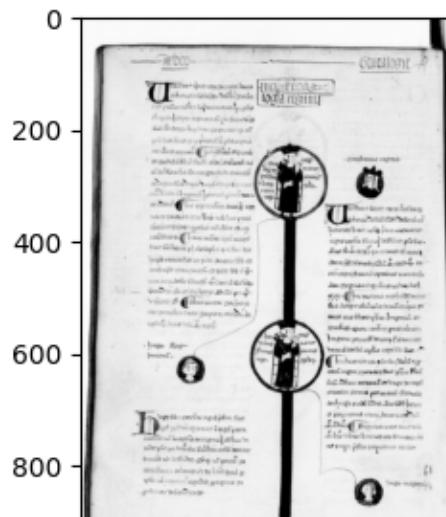
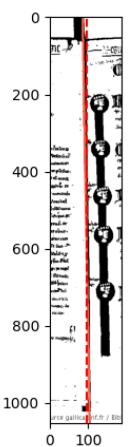
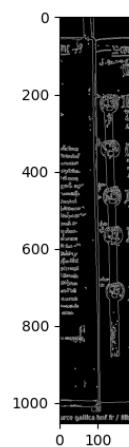
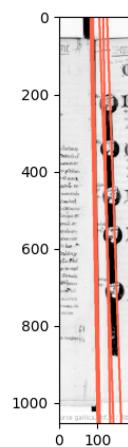
```
split_pages(gray)
```

35 candidat lines

```
-83.0 3.1241393 -0.9998477 0.017452458 82 -1
-89.0 3.1241393 -0.9998477 0.017452458 88 -1
-116.0 3.1241393 -0.9998477 0.017452458 115 -2
-106.0 3.106686 -0.99939084 0.03489945 105 -3
-126.0 3.106686 -0.99939084 0.03489945 125 -4
-87.0 3.1241393 -0.9998477 0.017452458 86 -1
r = 0.967741935483871 with split at 656
```



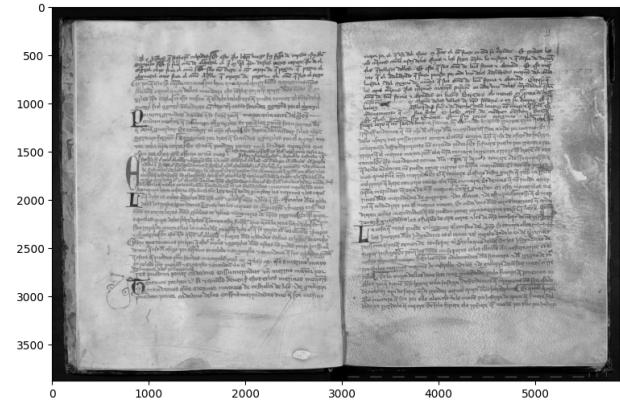
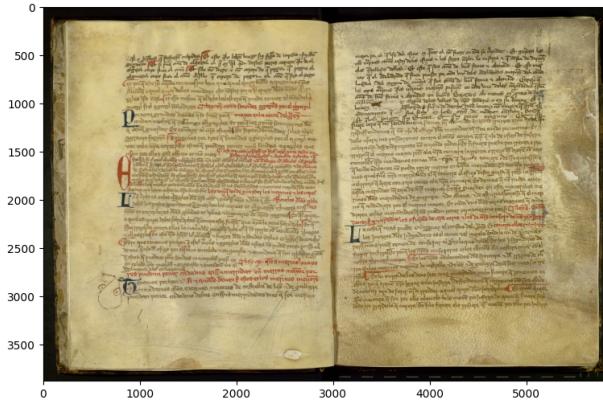
r max = 0.967741935483871



```
image = cv2.imread("../ImagesCodicologie/Corpus de jeu/Foronda/FF-BNE19438, fol. 11.tif")
```

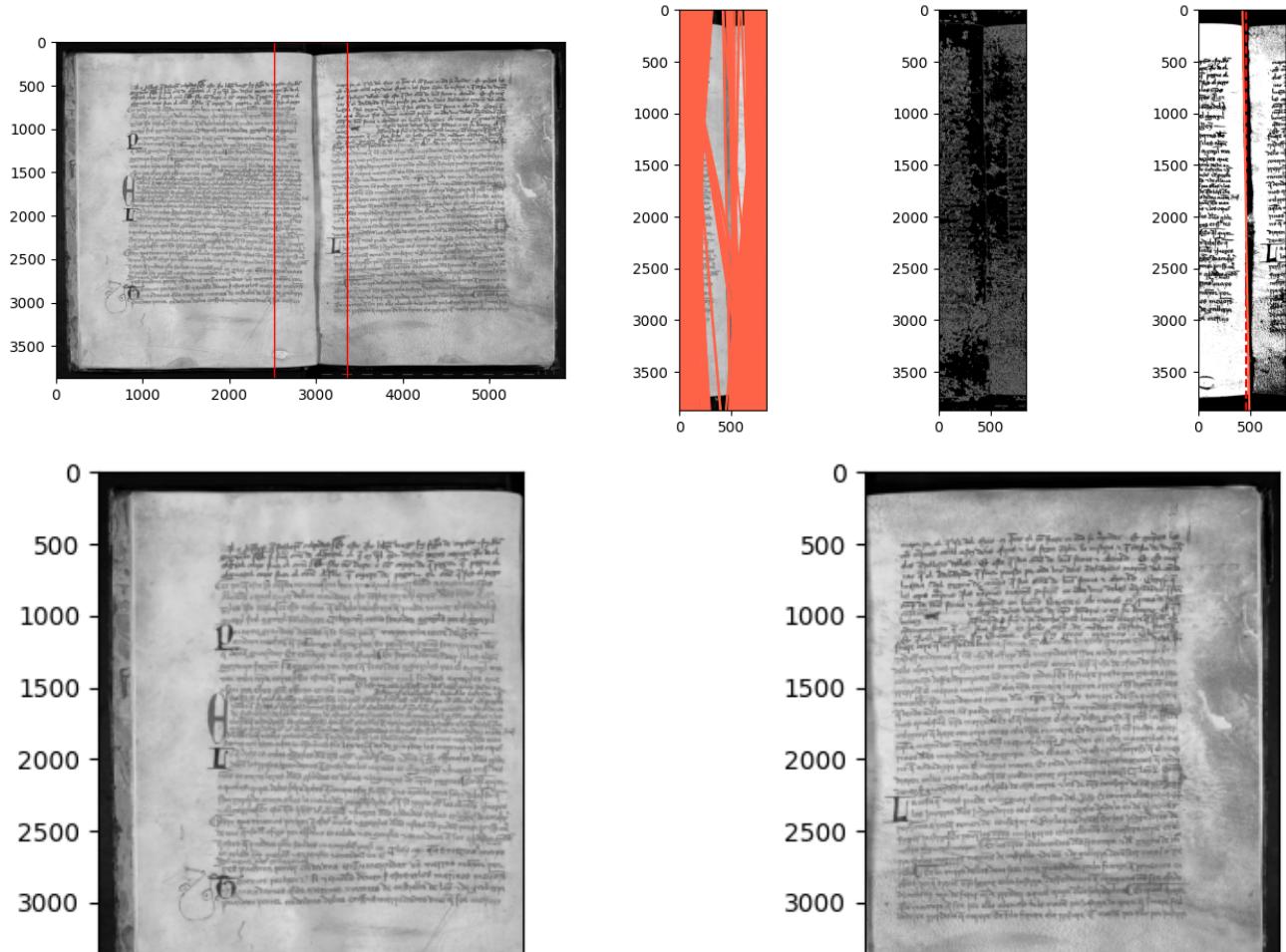
```
# Change color to RGB (from BGR)
image = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
# convert to grayscale
gray = cv2.cvtColor(image, cv2.COLOR_RGB2GRAY)
# normalize the image
norm_image = gray/255.0
#f_image = ft_image(norm_image)
# Display the images
f, (ax1,ax2) = plt.subplots(1, 2, figsize=(20,10))
ax1.imshow(image)
ax2.imshow(norm_image, cmap="gray")
#ax2.imshow(f_image, cmap="gray")
```

<matplotlib.image.AxesImage at 0x1f10e8e1fd0>



split_pages(gray)

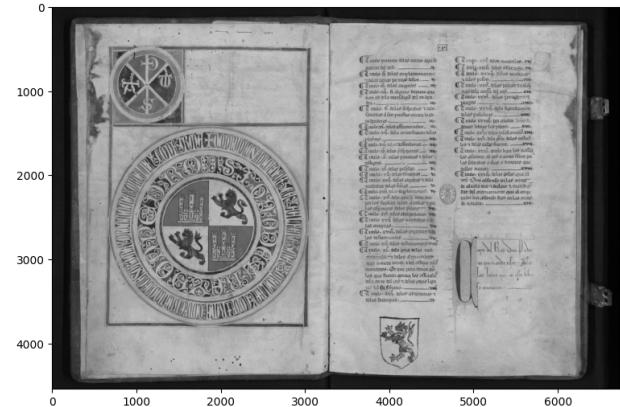
154.7670759822657
 19187 candidat lines amongst 41332 lines found
 3875 pixels calculated on final candidate line
 $-0.9998477 \ 0.017452458 \ 428 \ -7$
 $r = 0.8957419354838709$ with split at 2975
 $r_{\max} = 0.8957419354838709$



```
image = cv2.imread("../ImagesCodicologie/Corpus de jeu/Foronda/FF-BNE15-7, fol. 1v-2.tif")
```

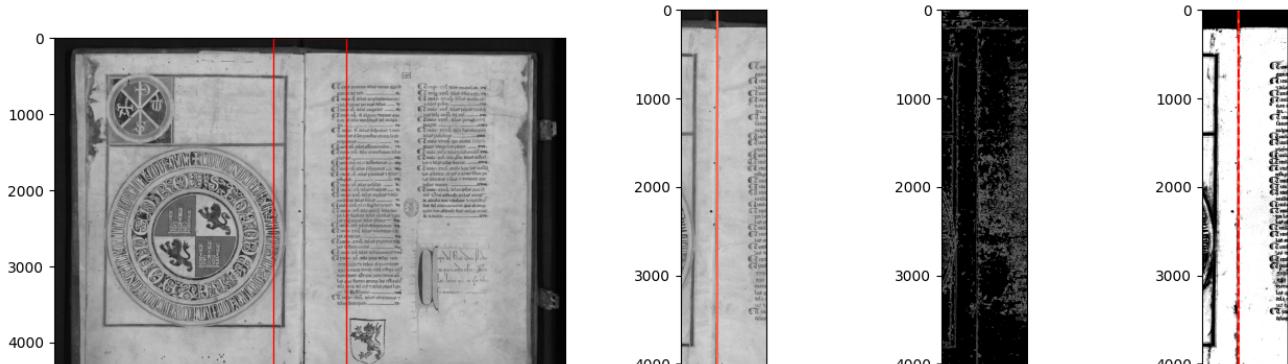
```
# Change color to RGB (from BGR)
image = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
# convert to grayscale
gray = cv2.cvtColor(image, cv2.COLOR_RGB2GRAY)
# normalize the image
norm_image = gray/255.0
#f_image = ft_image(norm_image)
# Display the images
f, (ax1,ax2) = plt.subplots(1, 2, figsize=(20,10))
ax1.imshow(image)
ax2.imshow(norm_image, cmap="gray")
#ax2.imshow(f_image, cmap="gray")
```

<matplotlib.image.AxesImage at 0x1f0b6c9e2d0>



split_pages(norm_image)

```
161.58898293644057
22576 candidat lines
4543 pixels calculated on final candidate line
410.0 0.0 1.0 0.0 410 0
r = 0.8591239269205371 with split at 3296
```



```
image = cv2.imread("../ImagesCodicologie/Corpus de jeu/Julien/Corpus B/Lat_12883_34r.jpg")
```

```
# Change color to RGB (from BGR)
image = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
# convert to grayscale
gray = cv2.cvtColor(image, cv2.COLOR_RGB2GRAY)
# normalize the image
norm_image = gray/255.0
#f_image = ft_image(norm_image)
# Display the images
f, (ax1,ax2) = plt.subplots(1, 2, figsize=(20,10))
ax1.imshow(image)
ax2.imshow(norm_image, cmap="gray")
#ax2.imshow(f_image, cmap="gray")
```

```
<matplotlib.image.AxesImage at 0x1f0a1a57f10>
```



```
split_pages(gray)
```

113.47166571942786

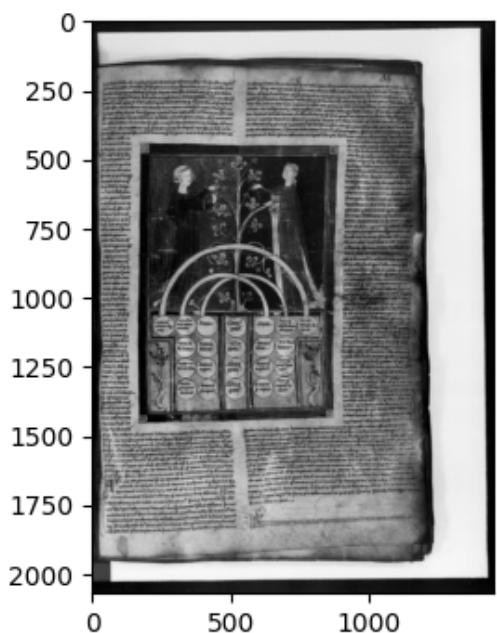
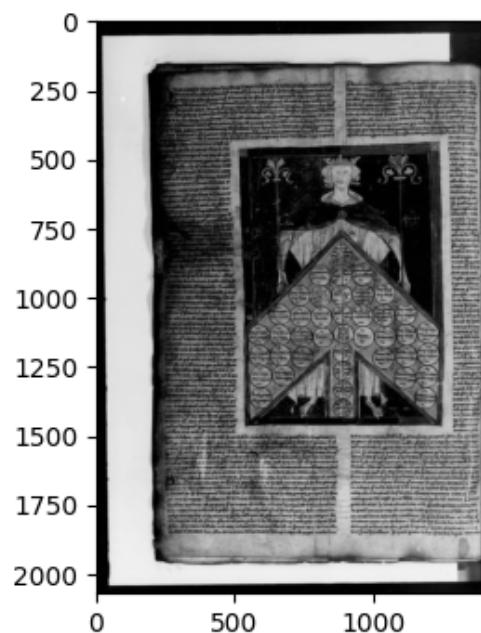
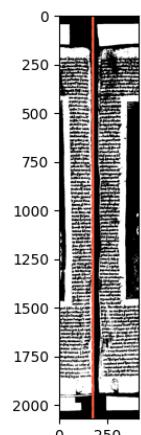
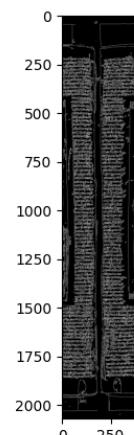
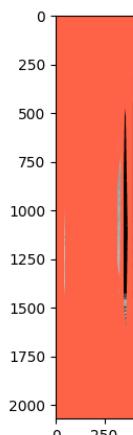
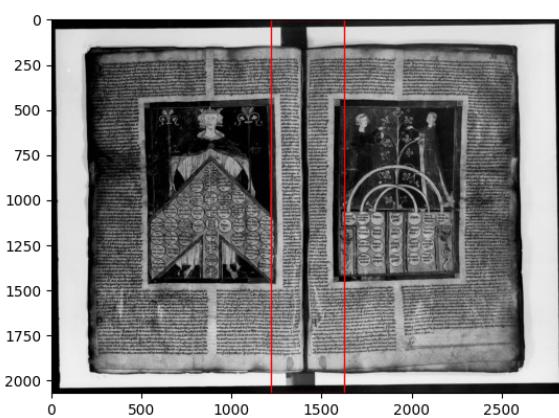
5370 candidat lines

2070 pixels calculated on final candidate line

174.0 0.0 1.0 0.0 174 0

r = 0.8570048309178744 with split at 1392

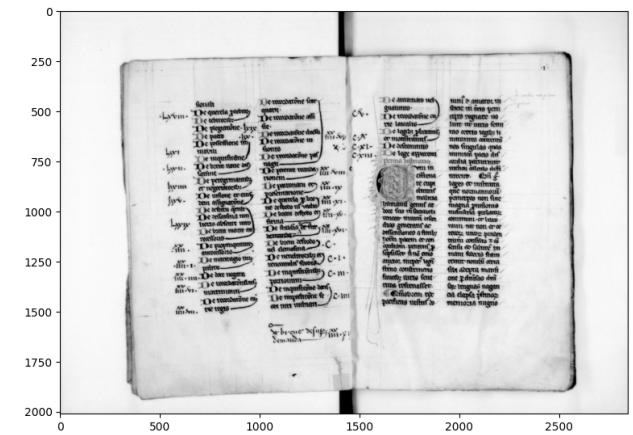
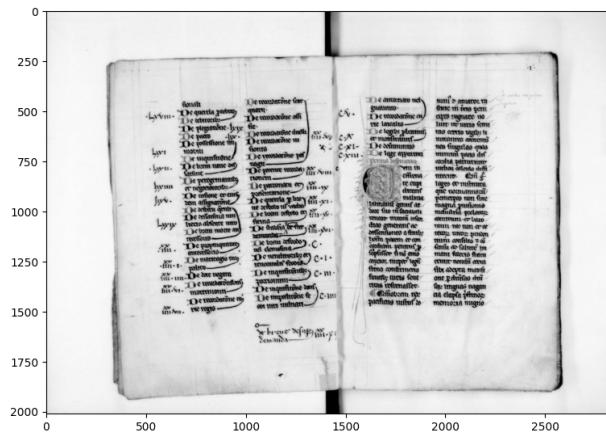
r max = 0.8570048309178744



```
image = cv2.imread("../ImagesCodicologie/Corpus de jeu/Julien/Corpus B/Lat_14689_3r.jpg")

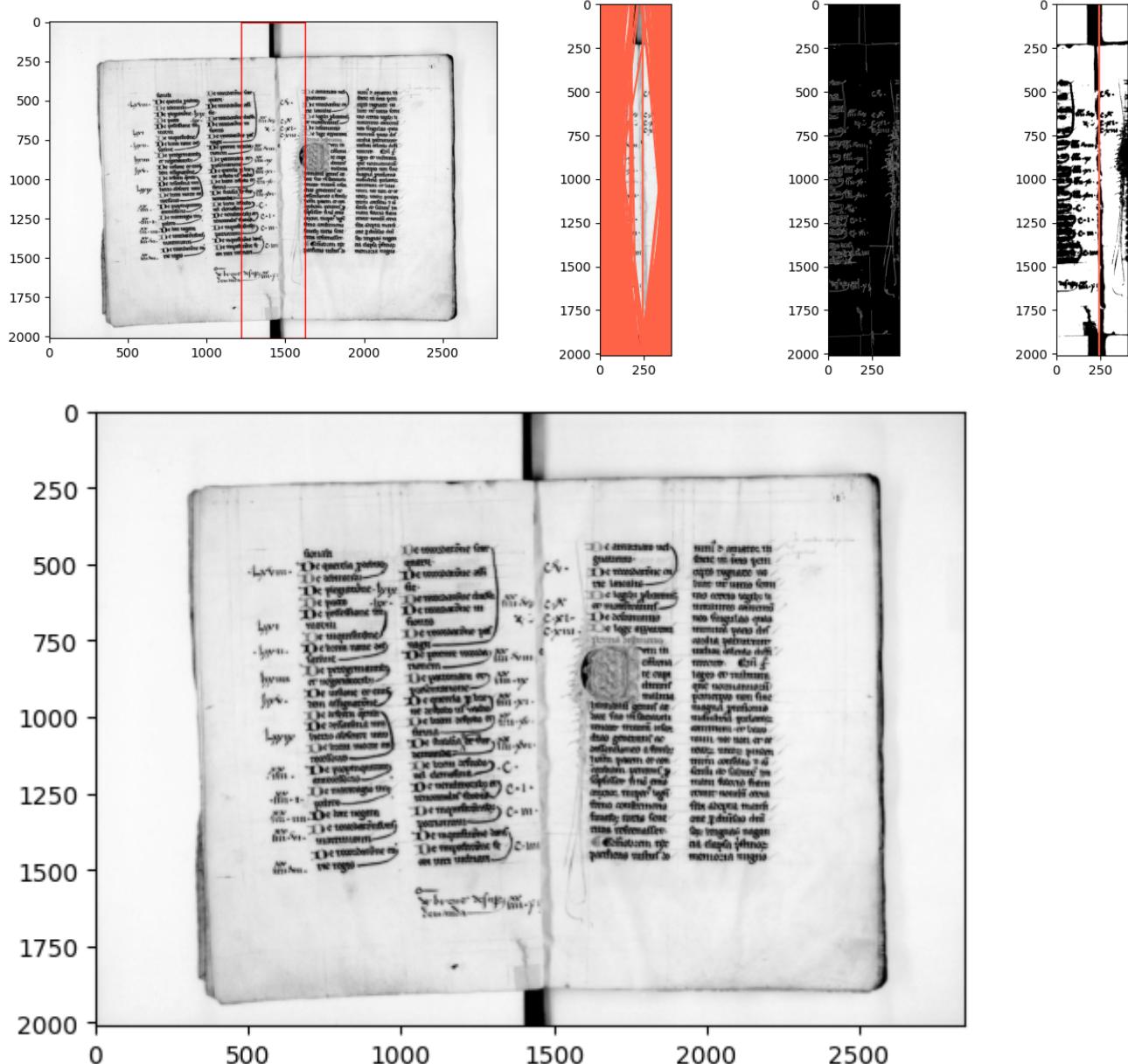
# Change color to RGB (from BGR)
image = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
# convert to grayscale
gray = cv2.cvtColor(image, cv2.COLOR_RGB2GRAY)
# normalize the image
norm_image = gray/255.0
#f_image = ft_image(norm_image)
# Display the images
f, (ax1,ax2) = plt.subplots(1, 2, figsize=(20,10))
ax1.imshow(image)
ax2.imshow(norm_image, cmap="gray")
#ax2.imshow(f_image, cmap="gray")
```

<matplotlib.image.AxesImage at 0x1f0b3df1850>



split_pages(norm_image)

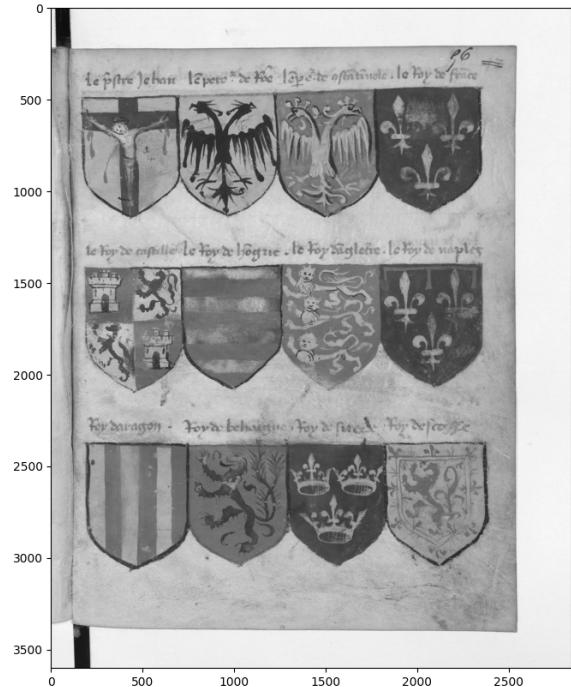
203.08997328419184
 355 candidat lines
 $r_{\max} = 0.6877175534559921$



```
image = cv2.imread("../ImagesCodicologie/Corpus de jeu/Julien/Corpus B/Fr_5930_56r.jpg")
```

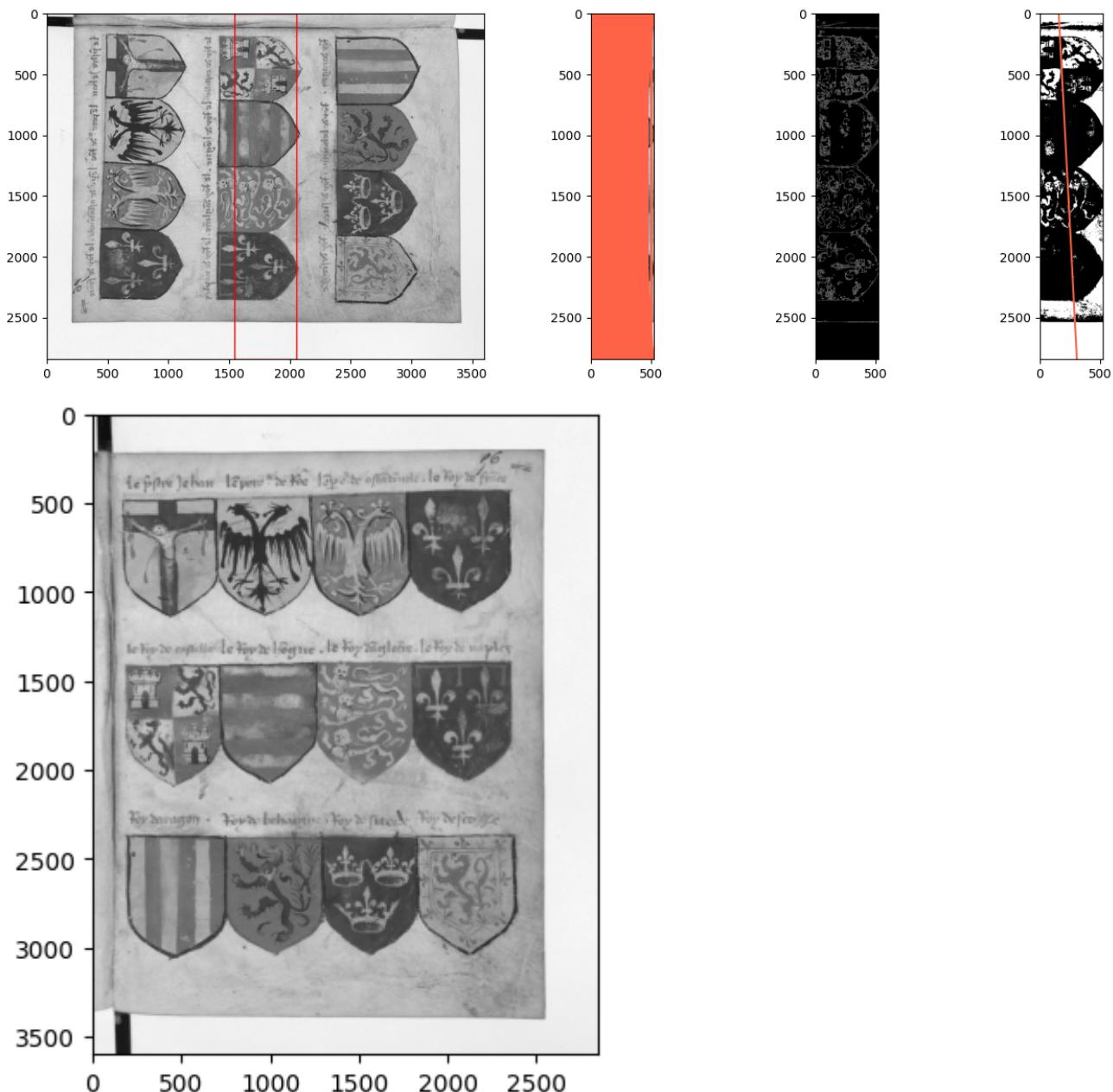
```
# Change color to RGB (from BGR)
image = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
# convert to grayscale
gray = cv2.cvtColor(image, cv2.COLOR_RGB2GRAY)
# normalize the image
norm_image = gray/255.0
#f_image = ft_image(norm_image)
# Display the images
f, (ax1,ax2) = plt.subplots(1, 2, figsize=(20,10))
ax1.imshow(image)
ax2.imshow(norm_image, cmap="gray")
#ax2.imshow(f_image, cmap="gray")
```

<matplotlib.image.AxesImage at 0x1f0b3d23f90>



split_pages(gray)

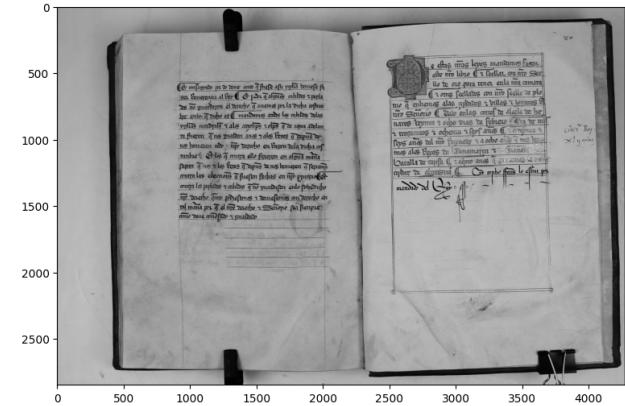
141.57093576397375
 1956 candidat lines
 $r_{\max} = 0.7879746835443038$



```
image = cv2.imread("../ImagesCodicologie/Corpus de jeu/Foronda/FF-RBMEYII7_fol-79v-80.jpg"

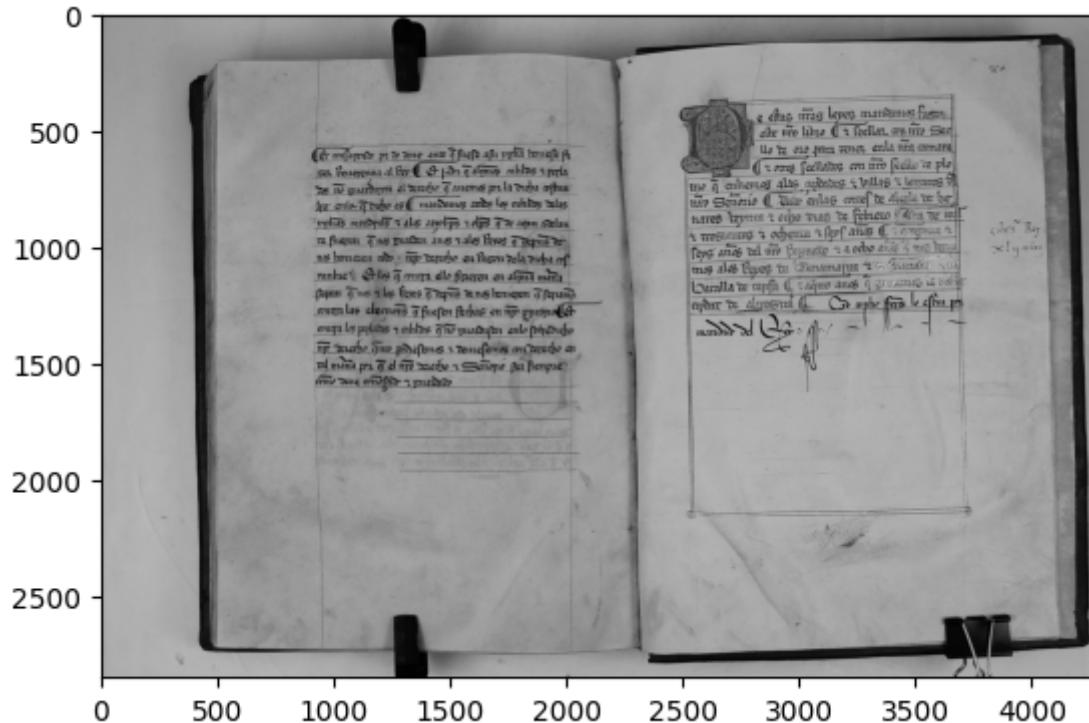
# Change color to RGB (from BGR)
image = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
# convert to grayscale
gray = cv2.cvtColor(image, cv2.COLOR_RGB2GRAY)
# normalize the image
norm_image = gray/255.0
#f_image = ft_image(norm_image)
# Display the images
f, (ax1,ax2) = plt.subplots(1, 2, figsize=(20,10))
ax1.imshow(image)
ax2.imshow(norm_image, cmap="gray")
#ax2.imshow(f_image, cmap="gray")
```

<matplotlib.image.AxesImage at 0x1f0b6bebed0>



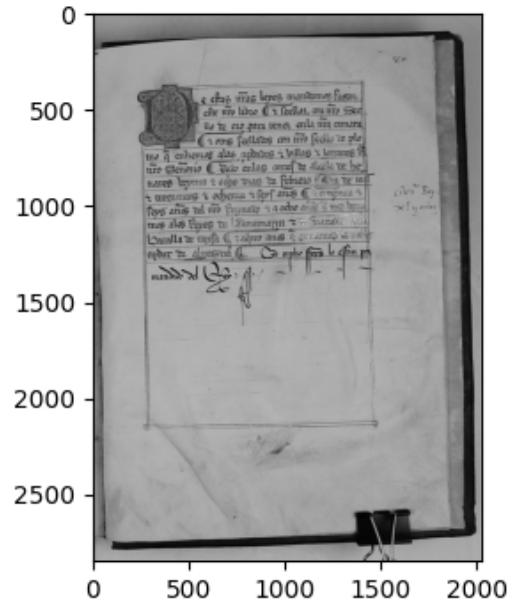
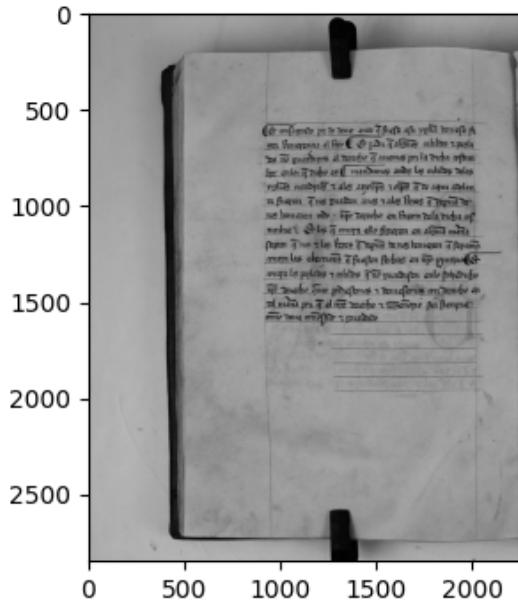
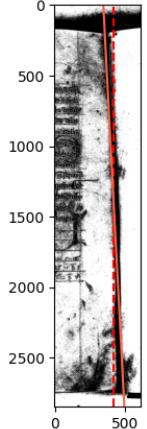
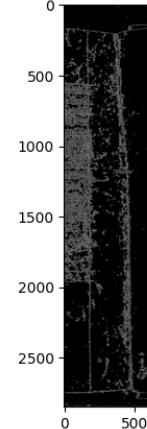
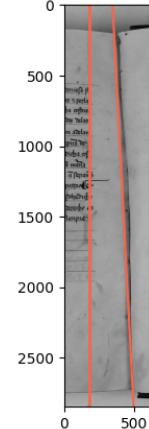
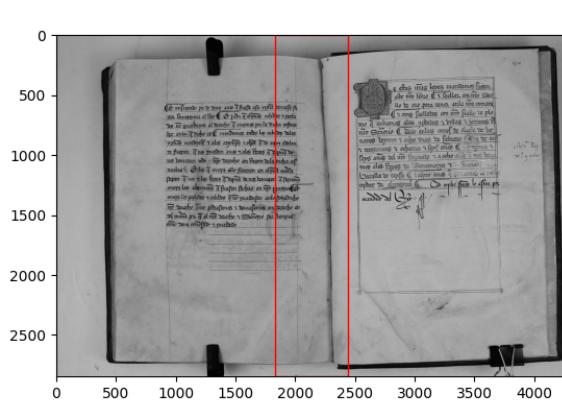
```
w = np.array([[0,-1,0],[-1,5,-1],[0,-1,0]])
cvd = convolve2d(gray,w)
plt.imshow(cvd,cmap='gray')
```

<matplotlib.image.AxesImage at 0x1f0a7e1bd50>



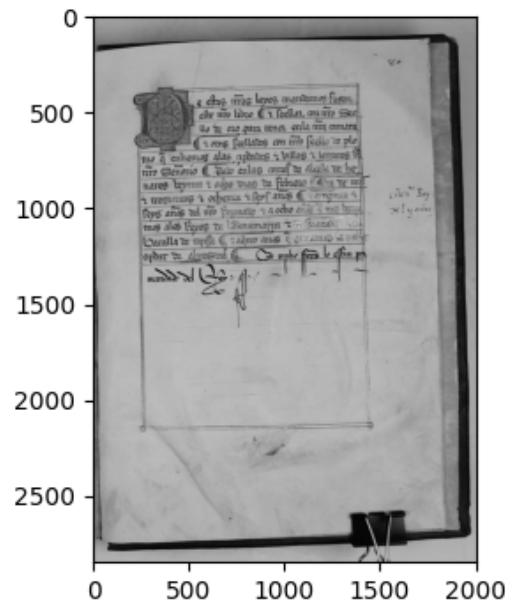
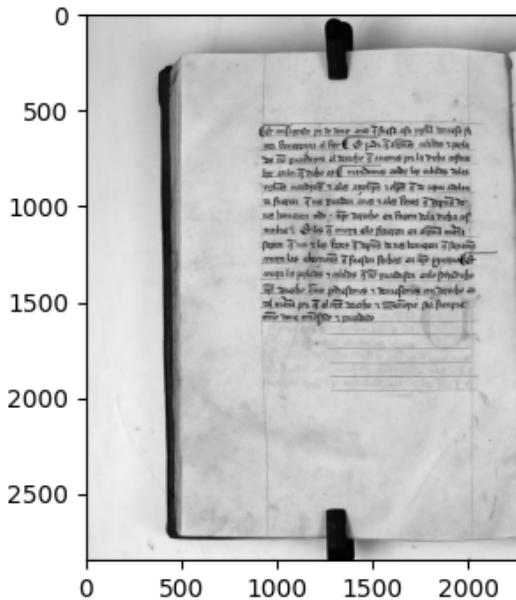
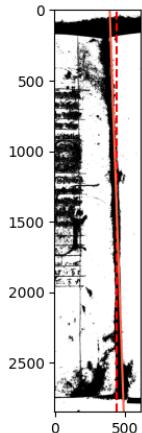
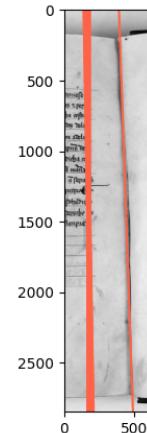
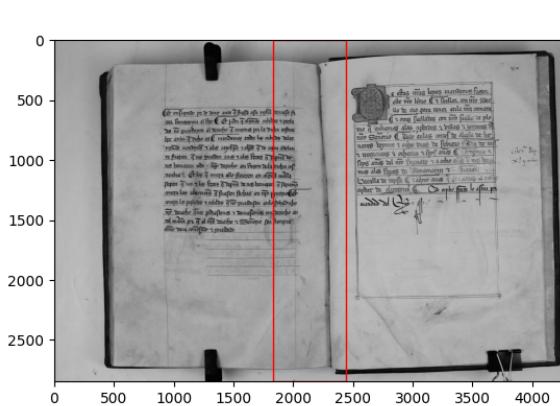
```
split_pages(cvd)
```

146.0984247631637
 5502 candidat lines
 2848 pixels calculated on final candidate line
 $-346.0 \ 3.0892327 \ -0.9986295 \ 0.052336052 \ 345 \ -18$
 $r = 0.9027387640449438$ with split at 2250
 $r \max = 0.9027387640449438$



```
split_pages(gray)
```

145.8841842916942
 400 candidat lines
 2848 pixels calculated on final candidate line
 $-392.0 \ 3.106686 \ -0.99939084 \ 0.03489945 \ 391 \ -13$
 $r = 0.9094101123595506$ with split at 2271
 $r_{\max} = 0.9094101123595506$



Produits payants Colab - Résilier les contrats ici

