image_mary_poppins

February 4, 2023

1 Quelques images de Mary Poppins

Le notebook télécharge une petite vidéo de Mary Poppins, extrait une dizaine de secondes, convertit la vidéo sous forme d'images, enlève le fond de chaque image et recompose une vidéo.

[1]:
```
from jyquickhelper import add_notebook_menu
add_notebook_menu()
```

[1]:
```
<IPython.core.display.HTML object>
```

Le module utilise moviepy et pytube3.

[2]:
```
%matplotlib inline
```

1.1 Télécharger une vidéo et extraire une partie

On choisit une vidéo de Mary Poppins : Supercalifragilisticexpialidocious.

[3]:
```
from pytube import YouTube
from pytube.exceptions import RegexMatchError
try:
    y = YouTube("https://www.youtube.com/watch?v=DYacXAd3eAo")
except RegexMatchError as e:
    print("Youtube... changes sometimes, don't ask me why", e)
```

```
Youtube… changes sometimes, don't ask me why regex pattern (yt\.akamaized\.net
/\)\s*\|\|\s*.*?\s*c\s*&&\s*d\.set\([^,]+\s*,\s*(?P<sig>[a-zA-Z0-9$]+)\() had
zero matches
```

[4]:
```
import os
name = 'Supercalifragilisticexpialidocious (from Mary Poppins) - Julie Andrews Dick␣
↪Van Dyke.mp4'
if not os.path.exists(name):
    from code_beatrix.art.video import download_youtube_video
    name = download_youtube_video(tag='DYacXAd3eAo', res=None)
name
```

[4]:
```
'Marie Poppins - Supercalifragilisticexpilialidocious.mp4'
```

On extraire une dizaine de secondes.

[5]:
```
from code_beatrix.art.video import video_extract_video, video_save
vid = video_extract_video(name, '00:00:32', '00:00:45')
video_save(vid, "mpoppins.mp4")
```

1.2 Convertir la vidéo en images

[6]:
```
folder = "images"
import os
if not os.path.exists(folder):
    os.mkdir(folder)
```

[7]:
```
from code_beatrix.art.video import video_enumerate_frames
imgs = list(video_enumerate_frames("mpoppins.mp4", folder=folder))
imgs[:5]
```

[7]:
```
['images\\images_0000.jpg',
 'images\\images_0001.jpg',
 'images\\images_0002.jpg',
 'images\\images_0003.jpg',
 'images\\images_0004.jpg']
```

[8]:
```
from IPython.display import Image
Image(imgs[0], width=360)
```

[8]:

[9]:
```
Image(imgs[10], width=360)
```

[9]:

[10]:
```
Image(imgs[-1], width=360)
```

[10]:

1.3 Extraire les personnages de l'image

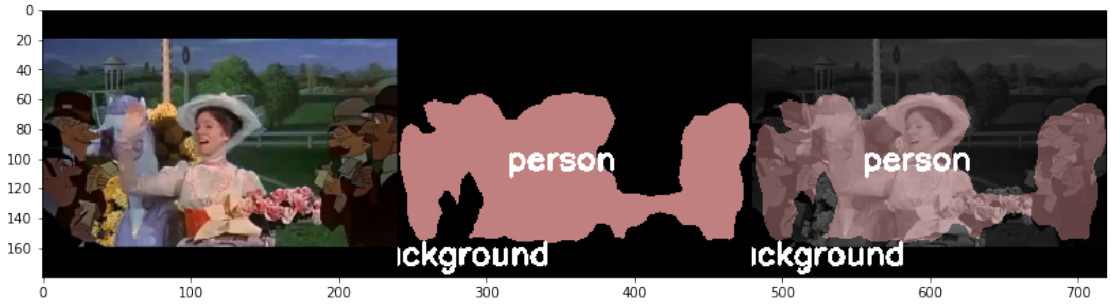
On essaye différentes résolutions. On s'aperçoit que les personnages de dessin animé sont aussi considérés comme des personnes.

[11]:
```
from code_beatrix.ai import DLImageSegmentation
model = DLImageSegmentation()
```

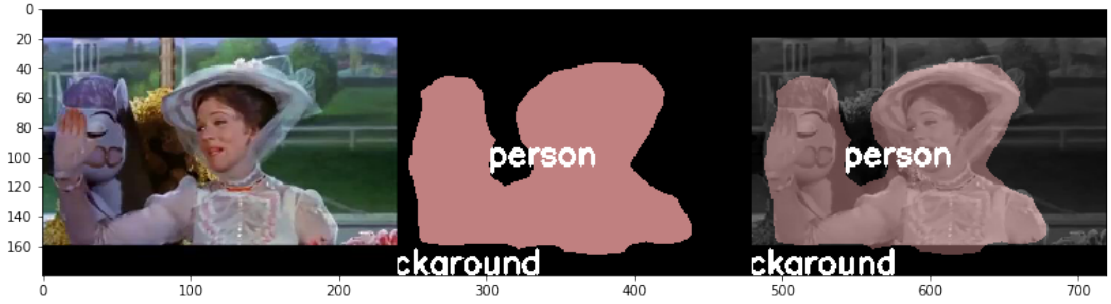
[12]:
```
feat, pred = model.predict(imgs[0], resize=('max2', 400))
```

[13]:
```
viz = model.plot(feat, pred)
```

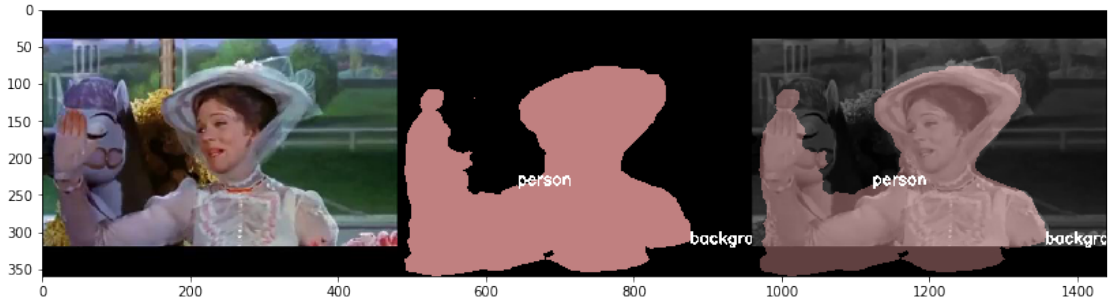
[14]:
```
import matplotlib.pyplot as plt
fig, ax = plt.subplots(1, 1, figsize=(14, 12))
ax.imshow(viz);
```



[15]:
```
feat, pred = model.predict(imgs[10], resize=('max2', 400))
viz = model.plot(feat, pred)
fig, ax = plt.subplots(1, 1, figsize=(14, 12))
ax.imshow(viz);
```



[16]:
```
feat, pred = model.predict(imgs[10])
viz = model.plot(feat, pred)
fig, ax = plt.subplots(1, 1, figsize=(14, 12))
ax.imshow(viz);
```



1.4 Combiner les personnages avec une autre image

[17]:
```
pred[pred!=0]
```

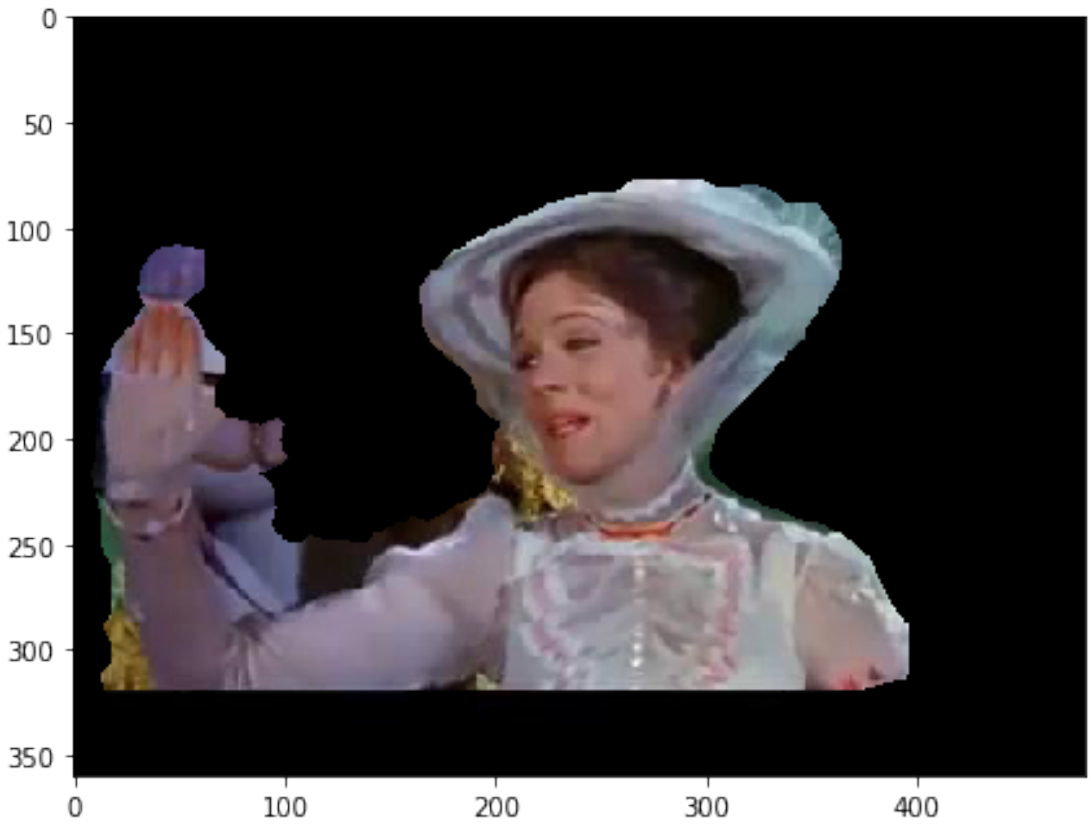
[17]:
```
array([15, 15, 15, …, 15, 15, 15])
```

[18]:
```
feat.shape
```

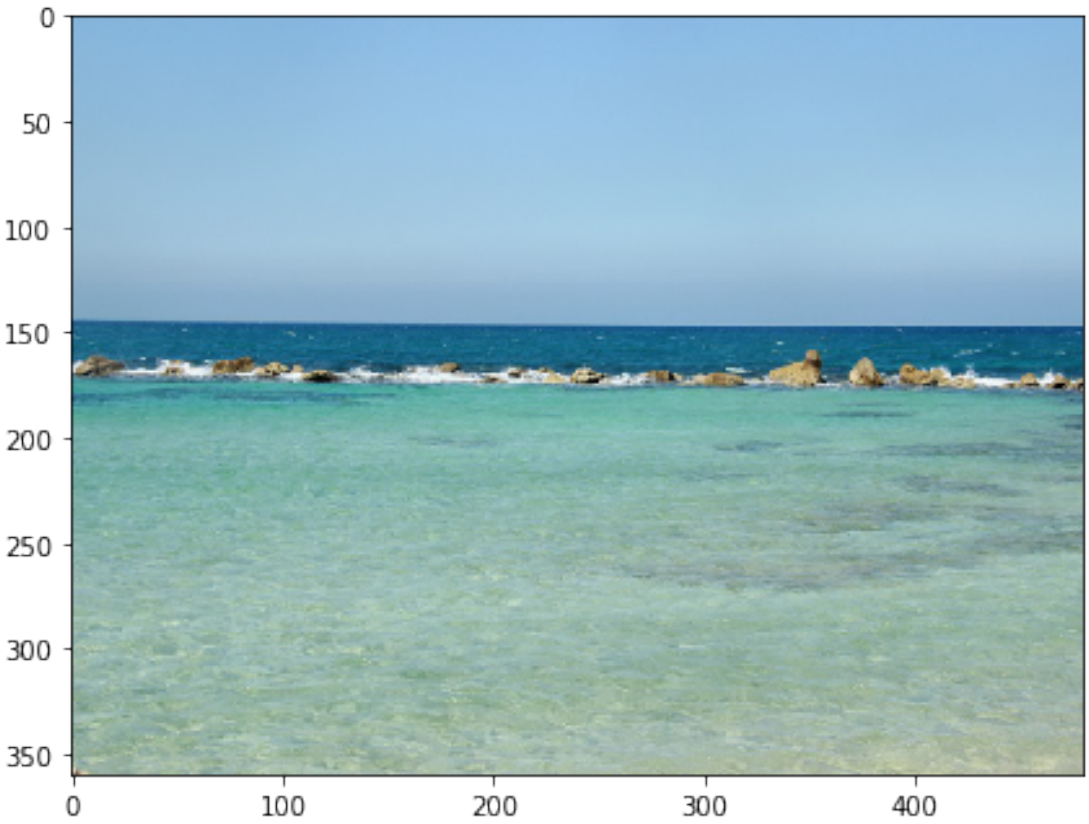
[18]:
```
(360, 480, 3)
```

[19]:
```
feat[pred!=15] = 0
```

[20]:
```
fig, ax = plt.subplots(1, 1, figsize=(7, 6))
ax.imshow(feat);
```



[21]:
```
from skimage.io import imread
from skimage.transform import resize
from skimage import img_as_ubyte
img = imread("data/plage.jpg")
img = img_as_ubyte(resize(img, (360, 480)))
fig, ax = plt.subplots(1, 1, figsize=(7, 6))
ax.imshow(img);
```

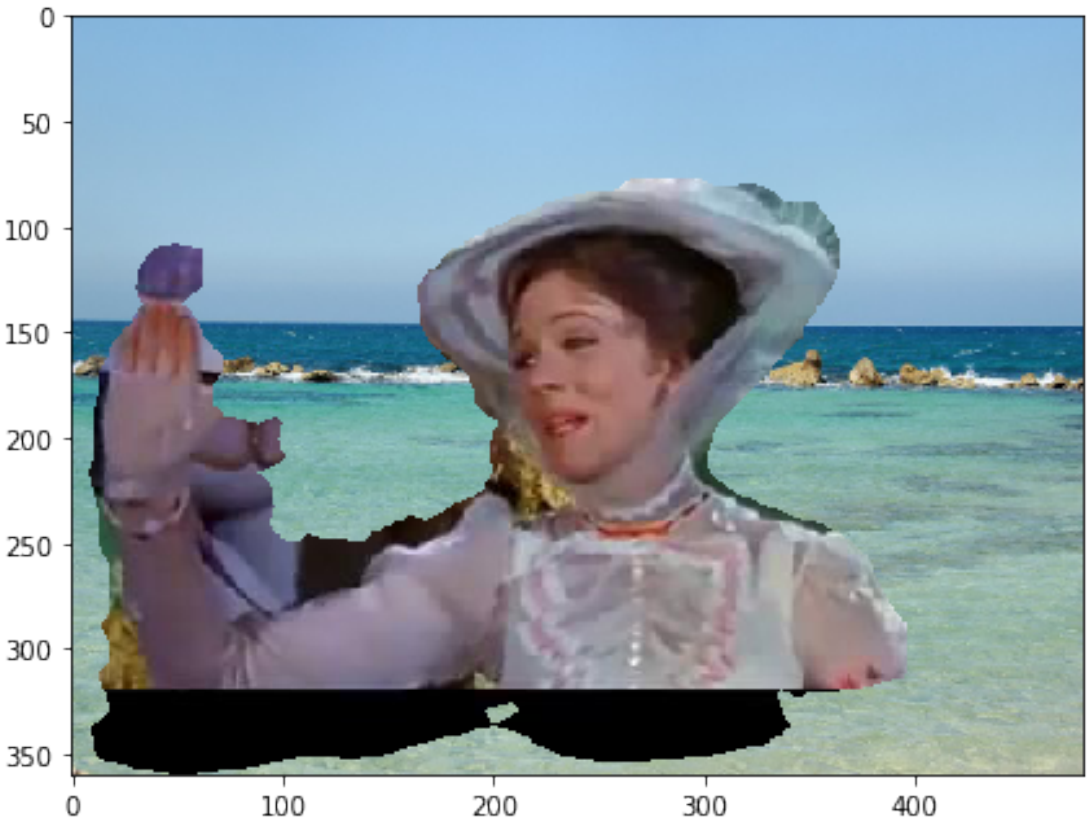


[22]:
```
img.shape, feat.shape
```

[22]:
```
((360, 480, 3), (360, 480, 3))
```

[23]:
```
img[pred==15] = feat[pred==15]
```

[24]:
```
fig, ax = plt.subplots(1, 1, figsize=(7, 6))
ax.imshow(img);
```



[25]: