

# Text and picture drawing

Function call: cv2.putText(img, str, origin, font, size, color, thickness)

The parameters are: picture, added text, upper left corner coordinate (integer), font, font size, color, and font thickness.

Font types are as follows:

Font Type	Description
FONT_HERSHEY_SIMPLEX	Normal size sans-serif font
FONT_HERSHEY_PLAIN	Small size sans-serif font
FONT_HERSHEY_DUPLEX	Normal size sans-serif font (more complex than FONT_HERSHEY_SIMPLEX)
FONT_HERSHEY_COMPLEX	Normal size serif font
FONT_HERSHEY_TRIPLEX	Normal size serif font (more complex than FONT_HERSHEY_COMPLEX)
FONT_HERSHEY_COMPLEX_SMALL	Smaller version of FONT_HERSHEY_COMPLEX
FONT_HERSHEY_SCRIPT_SIMPLEX	Handwritten style font
FONT_HERSHEY_SCRIPT_COMPLEX	More complex variant of FONT_HERSHEY_SCRIPT_SIMPLEX
FONT_ITALIC	Marked as italic font

- Start Docker

After entering the Raspberry Pi 5 desktop, open a terminal and run the following command to start the container corresponding to Dofbot:

```
./Docker_Ros.sh
```

Access Jupyter Lab within Docker:

```
IP:9999 // Example: 192.168.1.11:9999
```

Code path:/root/Dofbot/4.opencv/3.draw\_picture/06\_draw\_world.ipynb

```
import cv2
import numpy as np

img = cv2.imread('yahboom.jpg',1)

font = cv2.FONT_HERSHEY_SIMPLEX
```

```

cv2.rectangle(img,(200,100),(500,400),(0,255,0),3)

# 1 dst 2 text content 3 coordinates 4 5 font size 6 color 7 thickness 8 line
type

cv2.putText(img,'Yahboom',(250,50),font,1,(200,200,0),2,cv2.LINE_AA)

#cv2.imshow('src',img)

#cv2.waitKey(0)

```

```

import matplotlib.pyplot as plt

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

plt.imshow(img)

plt.show()

```

