EGYPTIAN SPACE AGENCY

EGYPTIAN UNIVERSITIES TRAINING SATELLITE PROJECT
EUTS

<< Cloud detection and segmentation >>

Prepared by

<<Benha University>>

<< Computer and Artifcial Intelligence — AI>> -

<<Team 35>>

EUTS
Egyptian Universities Training Satellite

Introduction

EUTS
Egyptian Universities Training Satellite

The importance of images that taken from satellite

EgSA
وكالة الفضاء المصرية
Egyptian Space Agency

EUTS
Egyptian Universities Training Satellite

- Nano and microsatellite image and its applications:
  1. agriculture and forestry
  2. weather forecast and Meteorological
  3. Fishing and ocean science
  4. Geology and cartography
  5. regional planning
  6. the war…etc.

EgSA
وكالة الفضاء المصرية
Egyptian Space Agency

There is problem in these images …

What is the solution…?

Software

- cloud detection
- cloud segmentation

hardware

- Raspberry pi and Camera pi

Simulation

- Arduino
- Stepper motor
- Slider
- Power bank

- The increasing demand for high resolution and Suitable shots for use and processing images from nano and microsatellites conflicts with the strict bandwidth constraints for downlink transmission. A possible approach to mitigate this problem consists in reducing the amount of data to transmit to ground through on-board processing of images. In this project, we use a custom Convolutional Neural Network (CNN) to select images eligible for transmission to ground.

- The project begins with live cloud detection then it takes a shot to make segmentation for this cloud. The images transmitted to ground are those that present less than 70% of cloudiness in a frame. We train and test the Neural network –the test set we achieve 88% of accuracy, The innovation aspect of our work concerns not only cloud detection but in general cloud segmentation to make decision if the image is Suitable for use and processing. Our work should enable a new era of edge applications and enhance remote sensing applications directly on-board satellite.

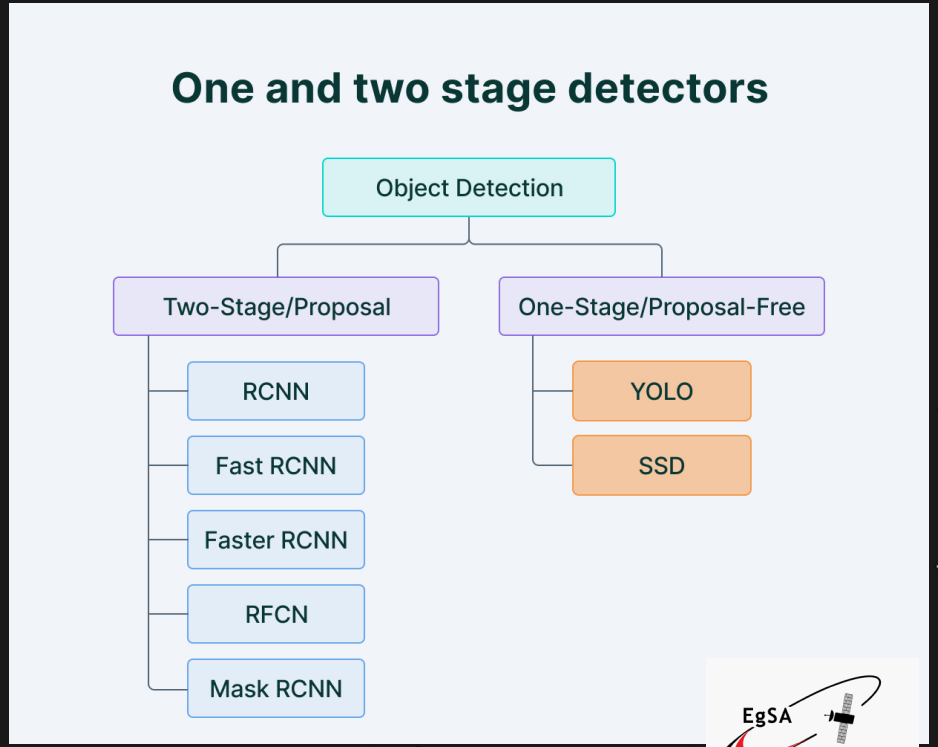
EUTS
Egyptian Universities Training Satellite

Object Detection

Yolov5

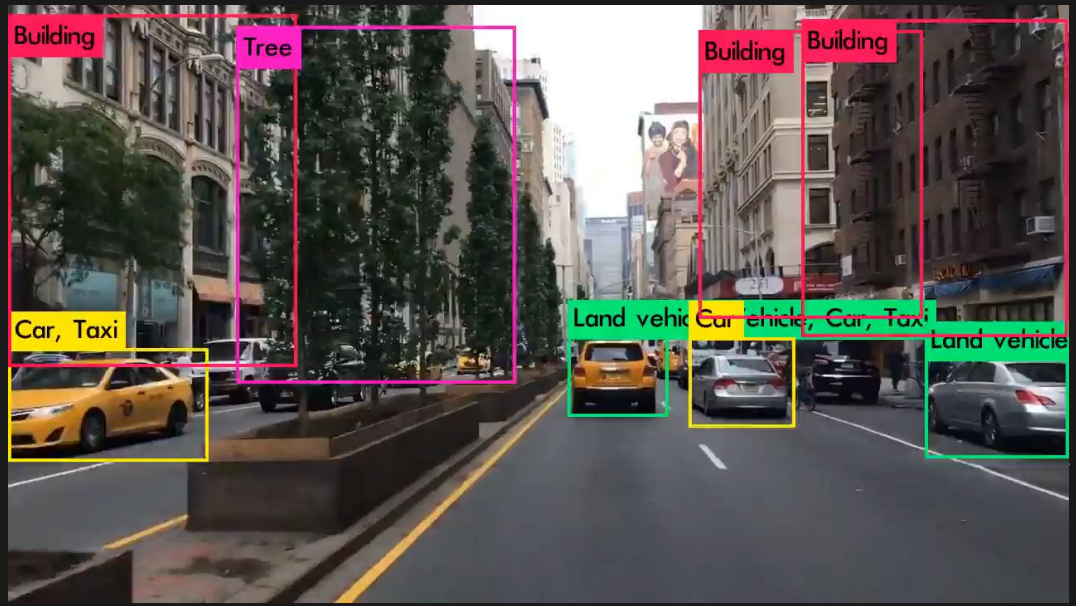
Object Detection

- Two-stage object detection refers to the use of algorithms that break down the object detection problem statement into the following two-stages:
    - Detecting possible object regions.
    - Classifying the image in those regions into object classes.
- Popular two-step algorithms like Fast-RCNN and Faster-RCNN typically use a Region Proposal Network that proposes regions of interest that might contain objects.



Yolo (You Only Look Once)

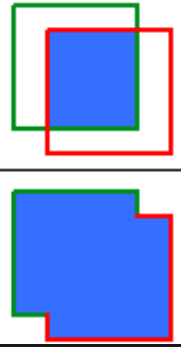
YOLO is a much faster algorithm than its counterparts, running at as high as 45 FPS.

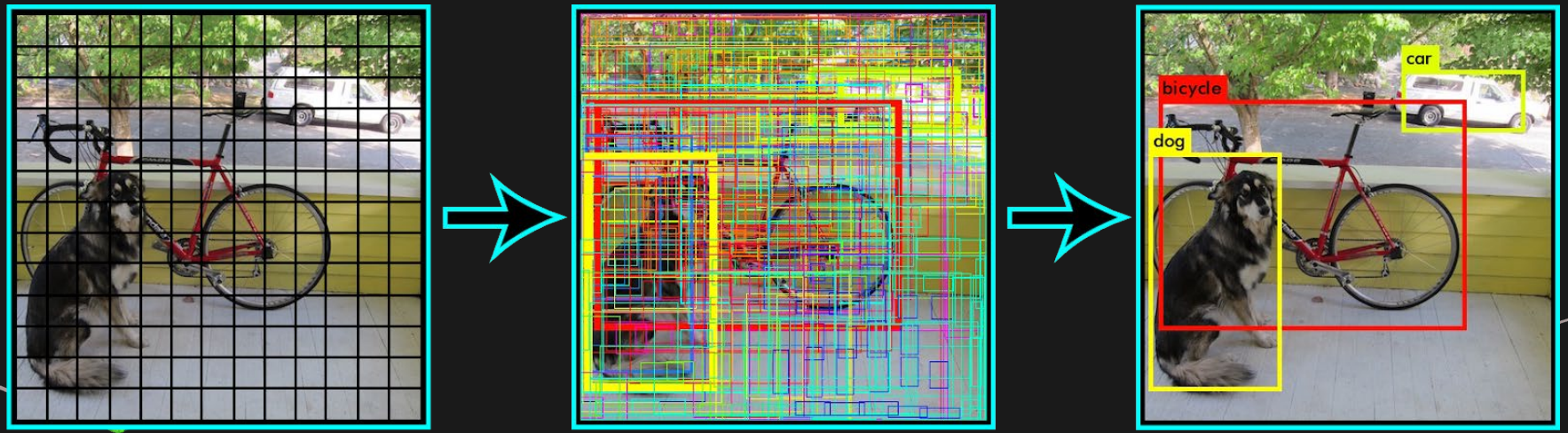


Intersection over Union (IoU)

An IoU value > 0.5. is taken as a positive prediction, while an IoU value < 0.5 is a negative prediction.

$$IOU = \frac{\text{area of overlap}}{\text{area of union}} =$$





Create Labels

zidane.txt

```
0  0.481719 0.634028 0.690625 0.713278
0  0.741094 0.524306 0.314750 0.933389
27 0.364844 0.795833 0.078125 0.400000
```

YOLO Architecture

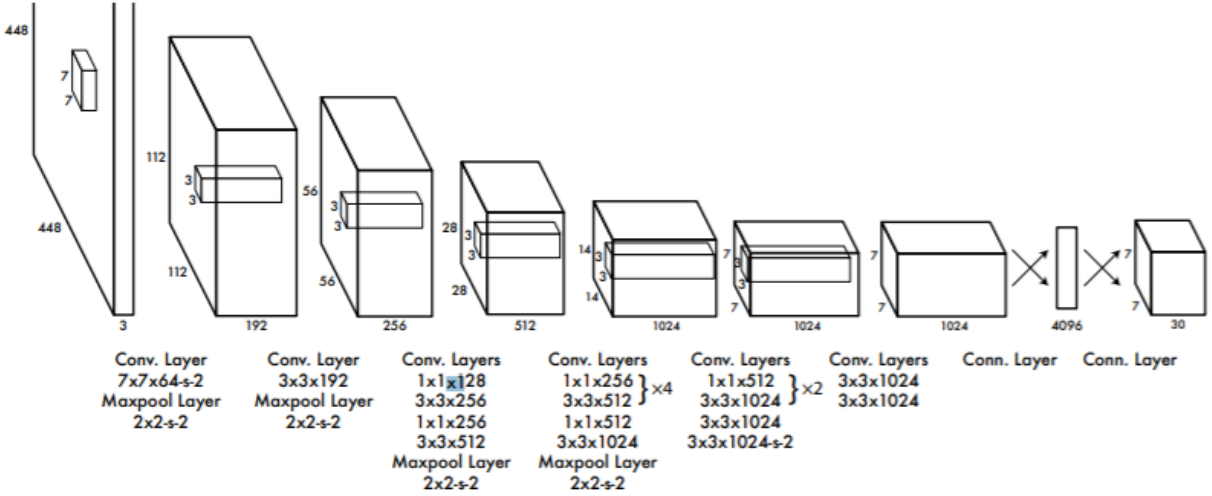


Figure 3: The Architecture. Our detection network has 24 convolutional layers followed by 2 fully connected layers. Alternating 1×1 convolutional layers reduce the features space from preceding layers. We pretrain the convolutional layers on the ImageNet classification task at half the resolution (224×224 input image) and then double the resolution for detection.

YOLO code

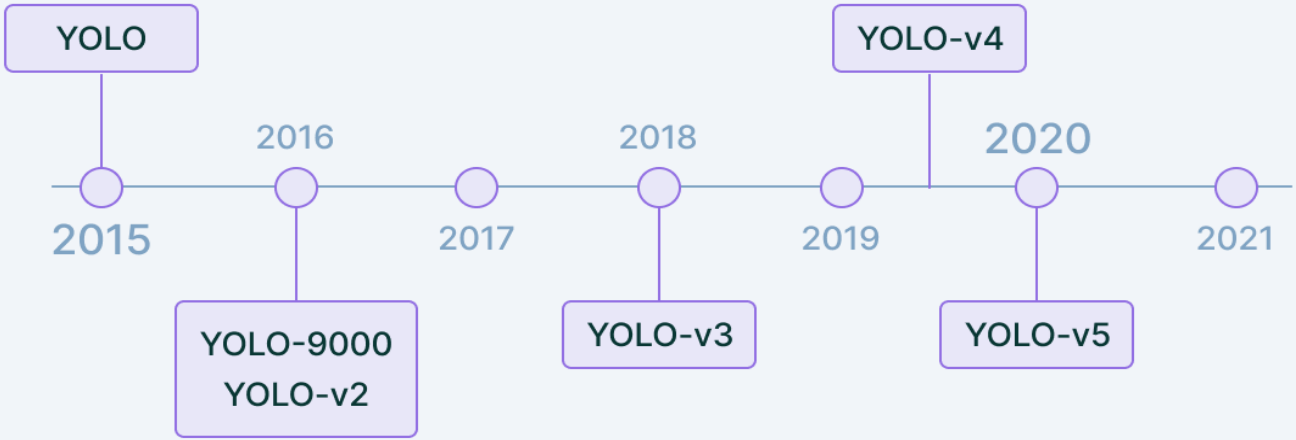
```
model = torch.hub.load('ultralytics/yolov5','custom',path = 'D:/UI/exp13/weights/best.pt')

img = 'D:/Dataset3/image/1 (235).jpg'

results = model(img)

results.show()
```

YOLO timeline



EUTS
Egyptian Universities Training Satellite

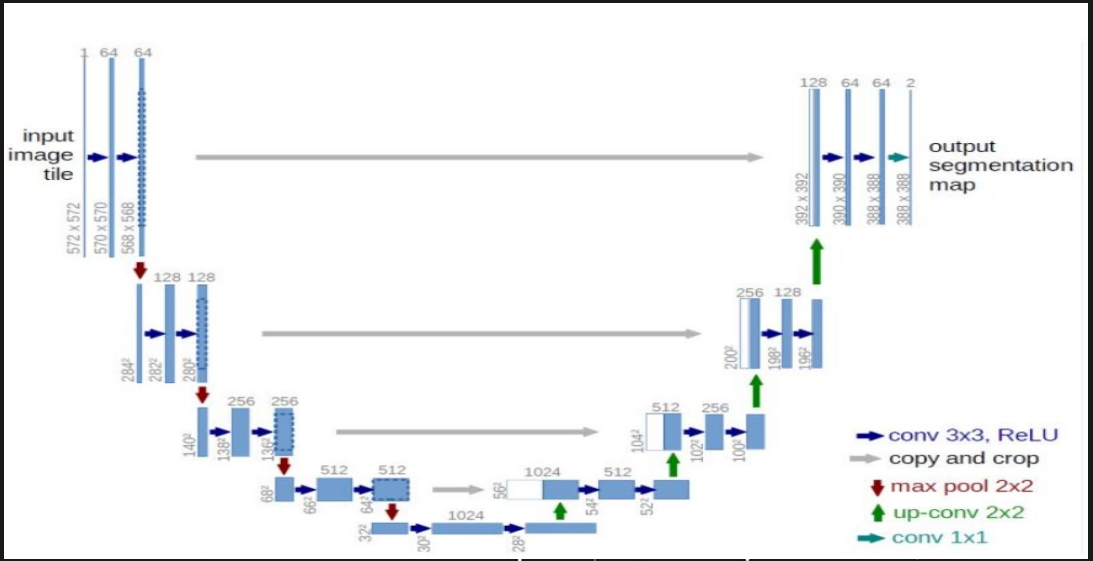
Semantic segmentation

U-net

Our aim from the project:

- To know the ratio of the clouds from the images that taken by satellites to send them to the station on the earth or not is this image can't be useful.

- So, we use semantic segmentation to clouds by learning our model to know the clouds from images using Unet style neural network.

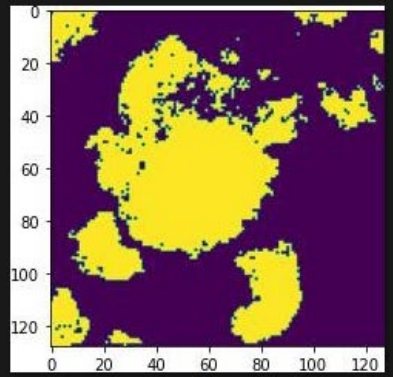


How to get the model ?

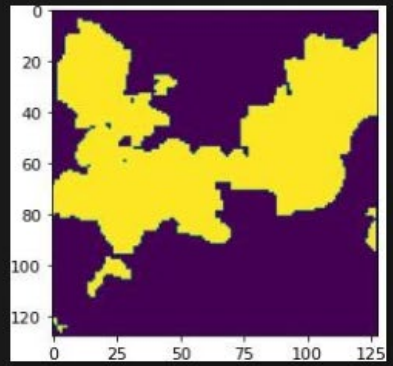
- We import to liner application a dataset of cloud images from satellites separating them into inputs which are original clouds and outputs which are masked cloud images.
- Then we get the model by cloud segmentation accuracy equals 92%.

mix 1 (14).jpg

mix 1 (19).jpg



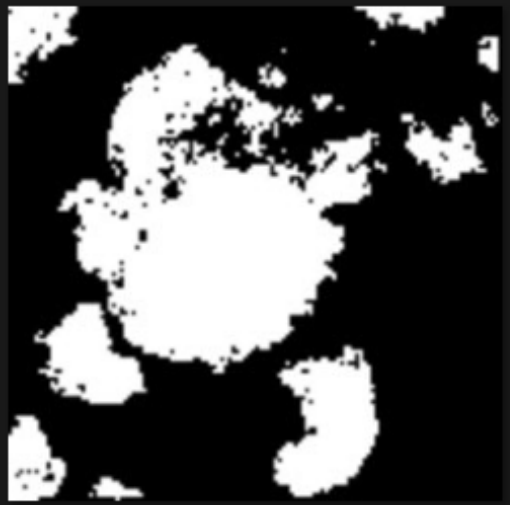
plt 1 (14).jpg



plt 1 (19).jpg

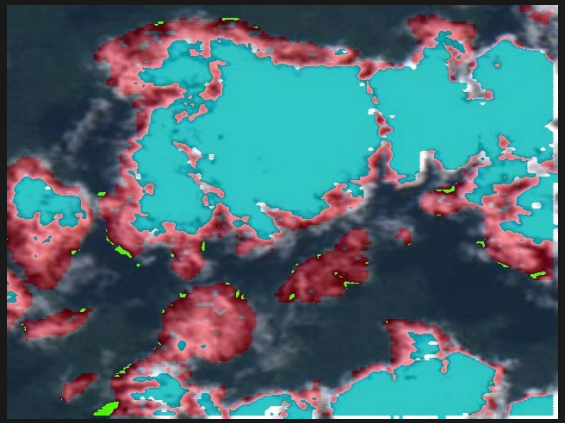
Converting the segmented output plot into binary image

- After getting the segmented clouds convert it into binary image and counting the white pixels which have value 1 the pixels of the cloud from the whole pixels of the out put image.



The acceptance :

- If the ratio of clouds is more than **40%** from image so the image is not accepted to be sent to the earth station , but if the ratio less than **40%** it will e accepted to be sent.



Accepted image

Not accepted image

EUTS
Egyptian Universities Training Satellite

Raspberry pi 3 Model B

- Raspberry pi is based on ARM architecture
- Specifically ARM cortex A
- The released date 29/2/2016
- This model uses microcontroller BCM2837 and it’s produced by Broadcom company
- instruction set : ARMV8-A
- Microarchitecture : Cortex-A53
- GPU : 400MHZ
- Cores : 4
- Threads : 4
- Frequency: 1.2 GHz
- Cache memory : 512 KB
- Max memory capacity 1 GB
- Memory types :LPDDR-900 SDRAM
- Max display resolution: 1080p@30fps
- Video decoding: H.264 1080p@30fps
- Max video capture: 1080p@30fps

- Video output : HDMI/composite
- Audio output : HDMI/Headphone
- 10/100M Ethernet
- Wi-Fi: 2.4GHz/5GHz 802.11 b/g/n
- Bluetooth: Bluetooth 4.1
- 4 USB 2 ports
- Full-size HDMI CSI (Camera Serial Interface) camera port for connecting a camera
- DSI (Display Serial Interface) display port for connecting a touchscreen display
- Micro SD port
- Micro USB power port (up to 2.5A)

- PWM (pulse-width modulation) pins:
  - Software PWM is available on all pins
  - Hardware PWM is available on these pins only:
    - GPIO12, GPIO13, GPIO18, GPIO19 SPI pins
- SPI0: GPIO9 (MISO), GPIO10 (MOSI), GPIO11 (SCLK), GPIO8 (CE0), GPIO7 (CE1)
- SPI1: GPIO19 (MISO), GPIO20 (MOSI), GPIO21 (SCLK), GPIO18 (CE0), GPIO17 (CE1), GPIO16 (CE2)
- I2C pins: Data: (GPIO2), Clock (GPIO3)
- UART Pins: TX (GPIO14) ,RX (GPIO15)

- Then we needed to enable a wireless communication between the raspberry pi And the laptop as simulation between the satellite and the ground space station

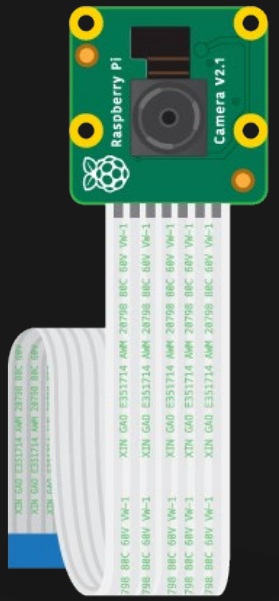
EUTS
Egyptian Universities Training Satellite

Raspberry Pi Camera

There are two versions of the Camera Module:

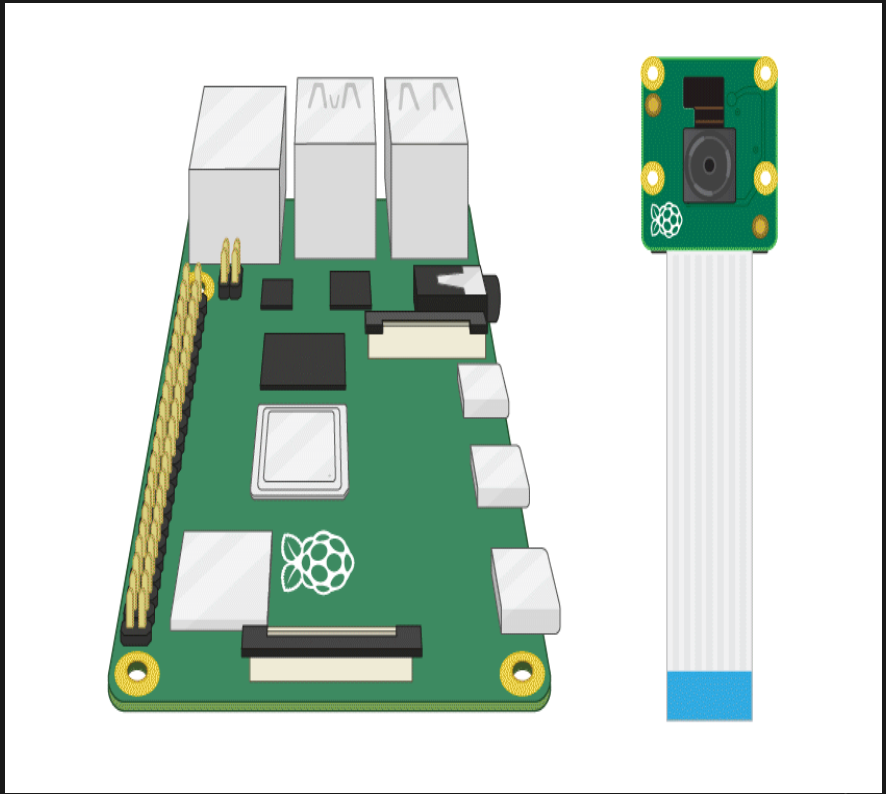
1- The standard version: which is designed to take pictures in normal light.

2- The NoIR version: which doesn't have an infrared filter, so you can use it together with an infrared light source to take pictures in the dark.

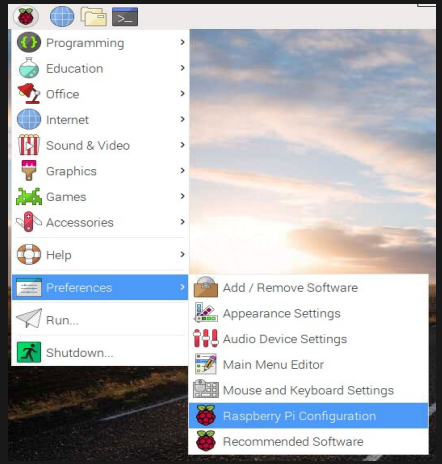


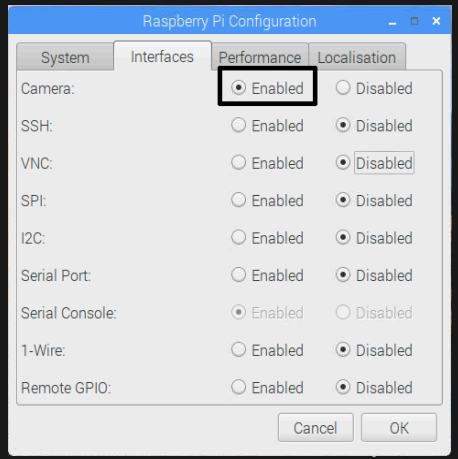
EgSA
وكالة الفضاء المصرية
Egyptian Space Agency

Connection:



- Start up your Raspberry Pi.
- Go to the main menu and open the Raspberry Pi Configuration tool.
- Select the Interfaces tab and ensure that the camera is enabled :
- Reboot your Raspberry Pi.

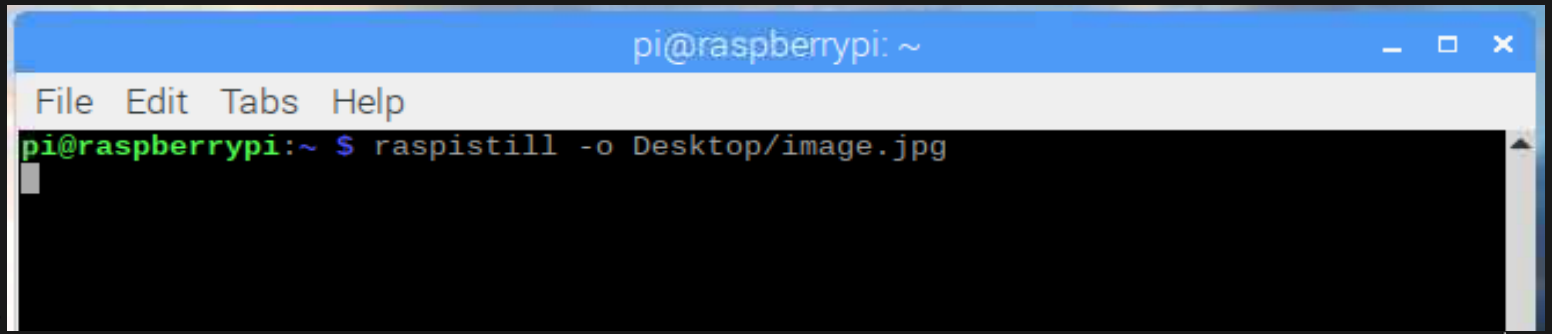




How to control the Camera Module via the command line ?

- Open a terminal window by clicking the black monitor icon in the taskbar :

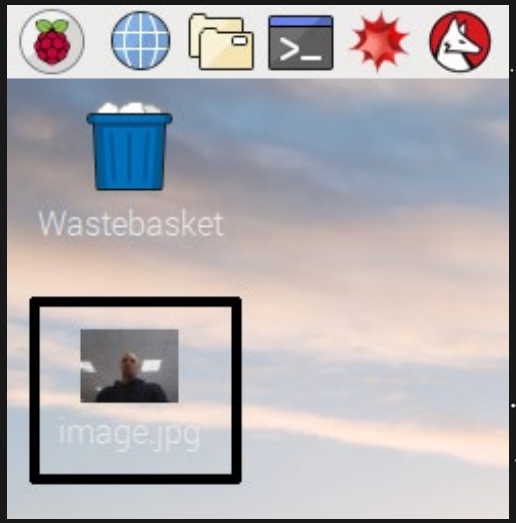
- Type in the following command to take a still picture and save it to the Desktop:



- Press Enter to run the command.

(When the command runs, you can see the camera preview open for five seconds before a still picture is taken).

- Look for the picture file icon on the Desktop and double-click the file icon to open the picture.



Code

```
def camera(self):
        cap = cv2.VideoCapture(0)
        while cap.isOpened():
            time.sleep(1 / int(self.TSpinbox1.get()))
            ret, frame = cap.read()
            cv2.imshow('Camera',frame)
            if cv2.waitKey(10) & 0xFF == ord('q'):
                break
            elif cv2.waitKey(10) & 0xFF == ord('c'):
                cv2.imwrite('images/'+str(uuid.uuid1())+'.jpg',frame)
                print("done")
                self.TLabel1.configure(background="#ffffff")
                self.TLabel1.configure(foreground="#000000")
                self.val.set("Smile")
                break
        cap.release()
        cv2.destroyAllWindows()
```

Cv2.resize

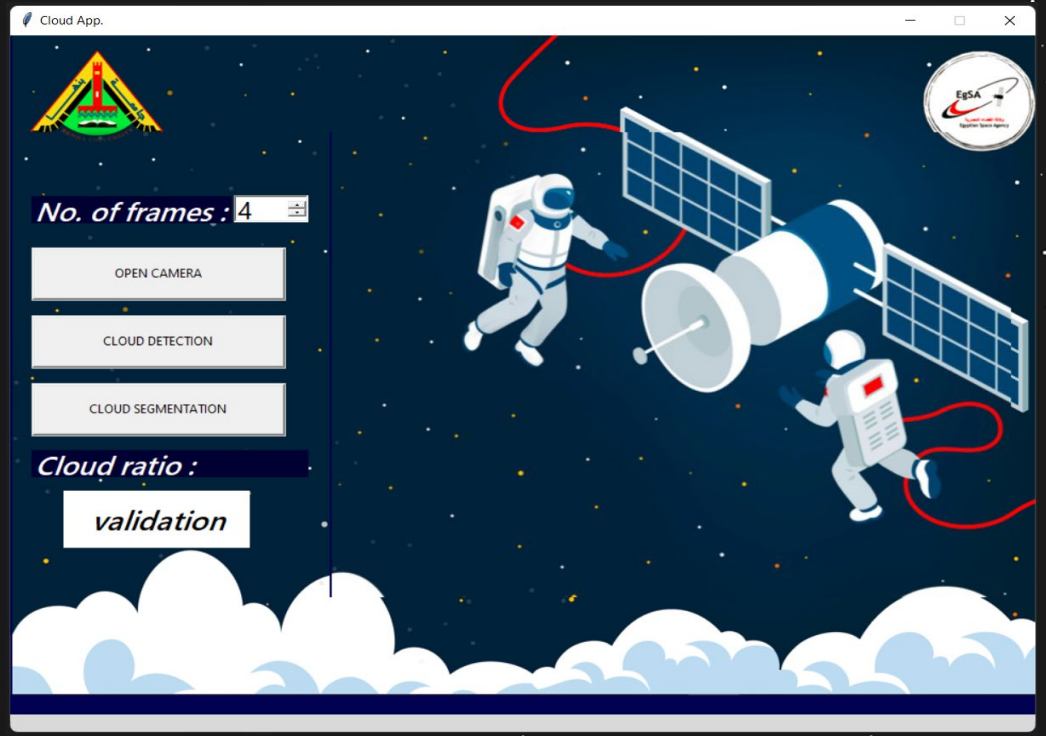
```
img_cv = cv2.resize(img_cv1, [orimage.shape[1],orimage.shape[0]])
orimages = cv2.resize(orimage,[640,480])
mix = cv2.resize(cv2.convertScaleAbs(segmentation_output_cv, alpha=(1)), [640,480])
```

EgSA
وكالة الفضاء المصرية
Egyptian Space Agency

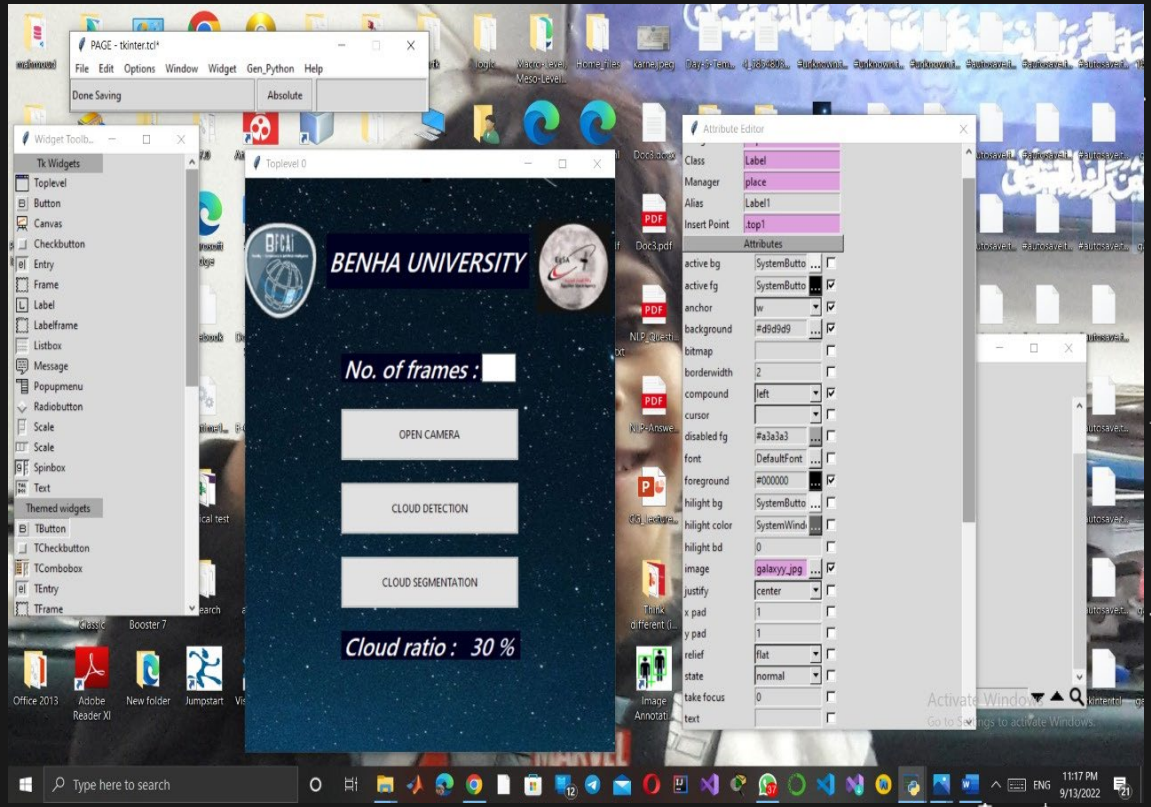

GUI
Tkinter

We used a simple GUI to represent our work on it by using Tkinter

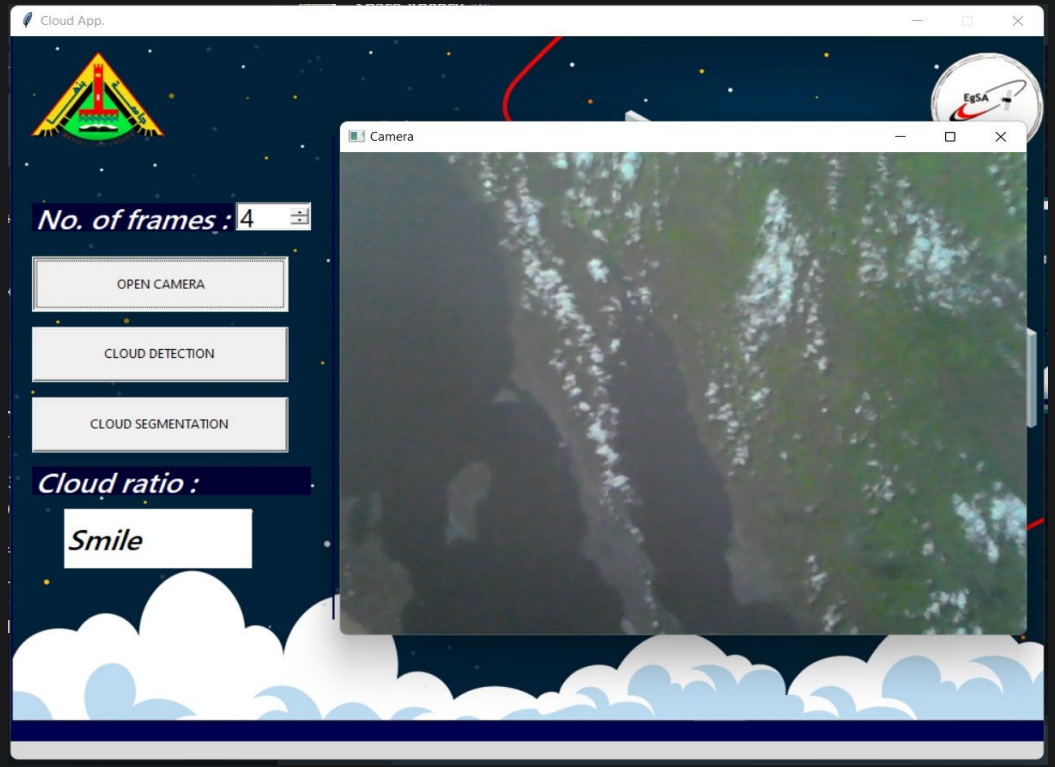
First we can change the number of frames so that the speed of camera will change as we want.



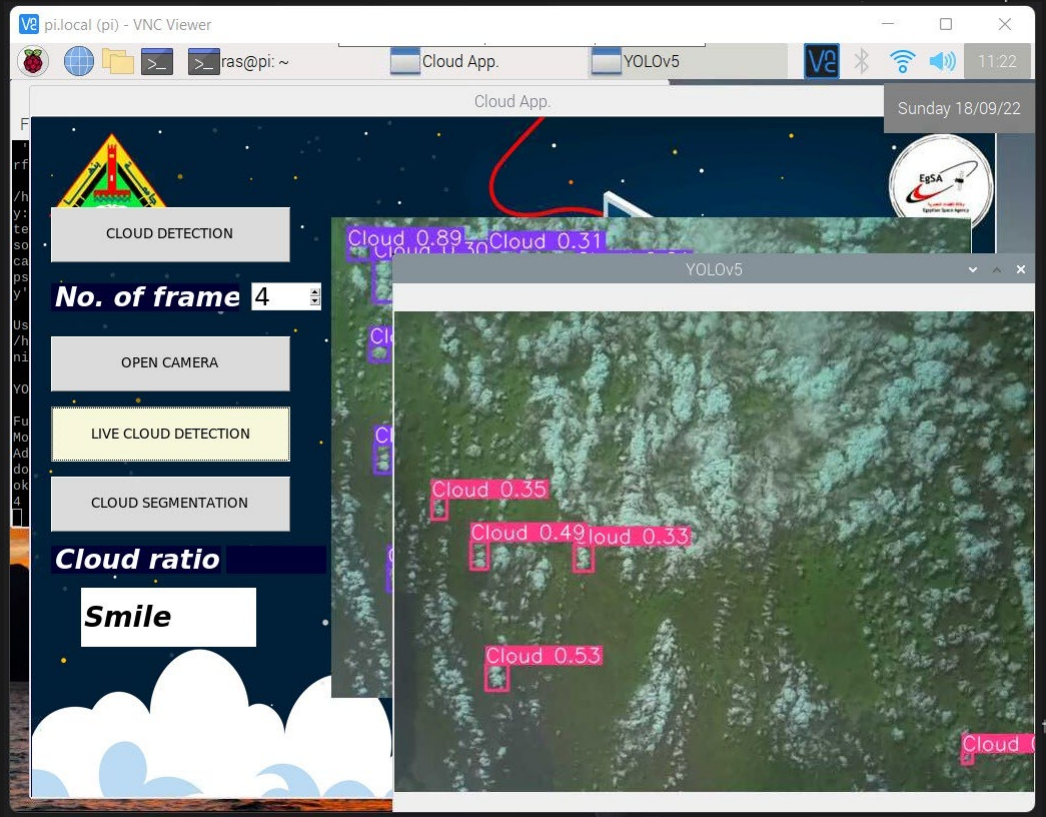
- We used a simple GUI to represent our work on it by using Tkinter
- We use drag and drop techniques by using tkinter



Then we can open camera by “Open camera” button and take a shoot of cloud image then the photo saved in images folder

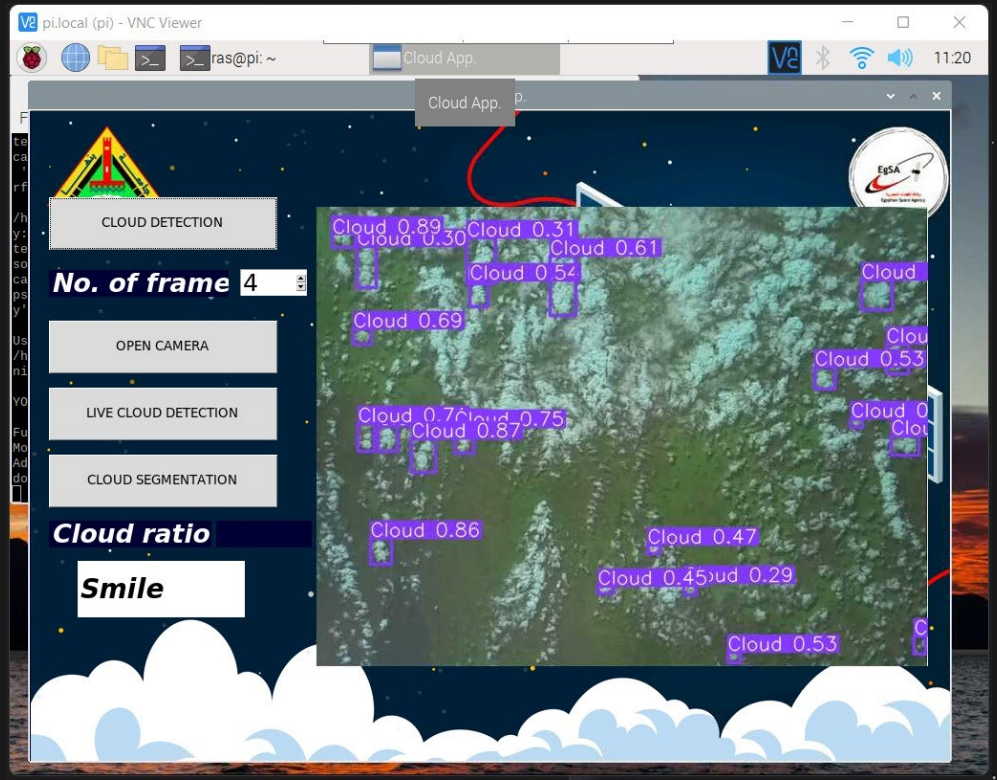


We can operate live cloud detection by using“Live cloud detection” button

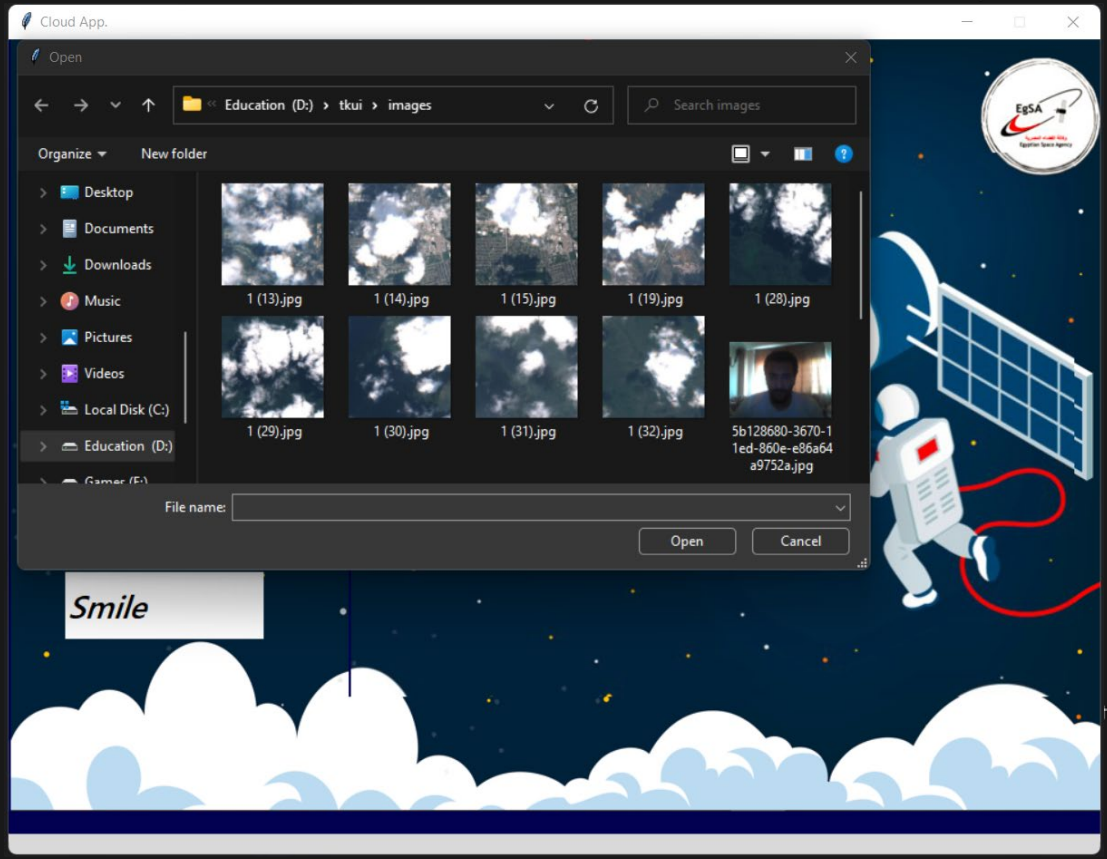


The cloud can be detected when we choose

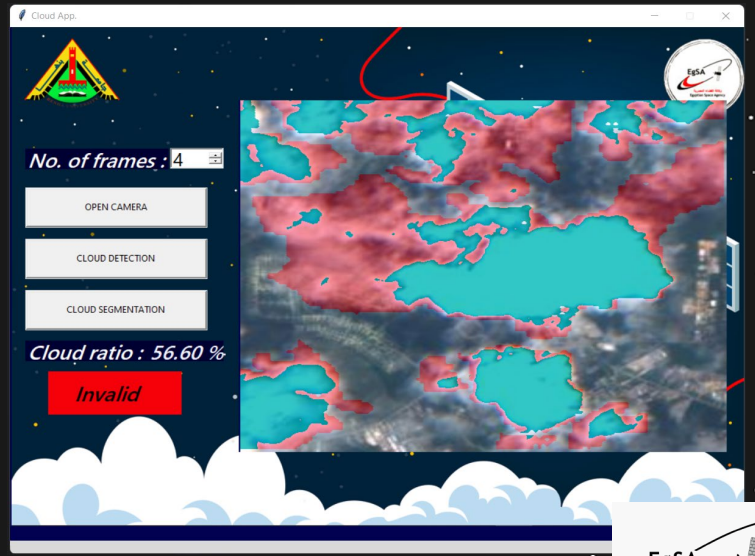
a photo by using“Cloud detection” button



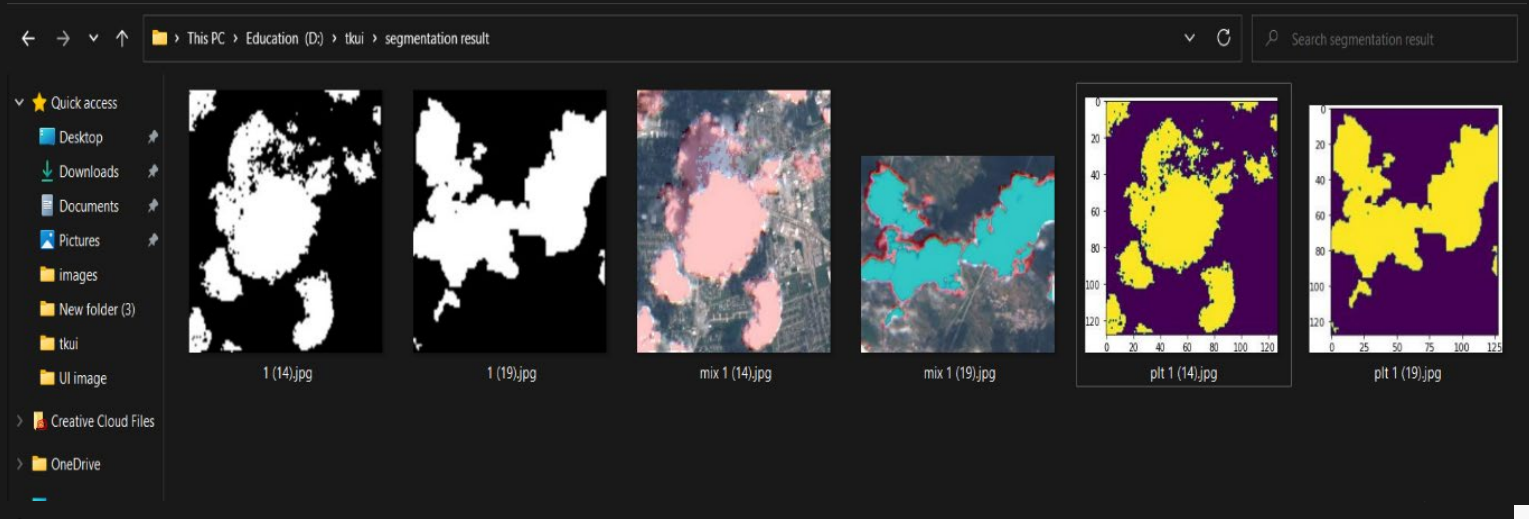
After that when clicking on “Cloud segmentation” button new window will appear and we can choose the image that we want to make segmentation on it.



Then the segmented image will apear on the screen and also the cloud ratio will appear declaring the validity of the image by making a label that inform “Valid” if the cloud ratio is less than 40% or “Invalid” if the cloud ratio more than 40%.



Finally the segmented image will be saved in segmentation result folders by different forms.

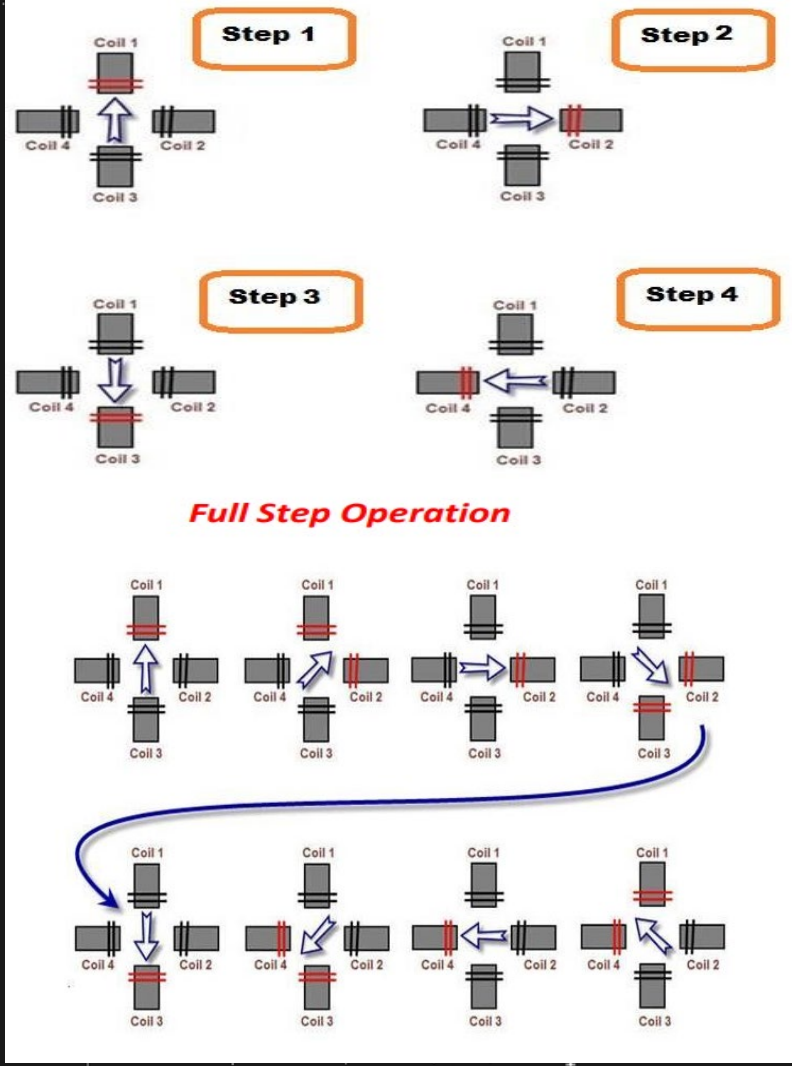


EUTS
Egyptian Universities Training Satellite

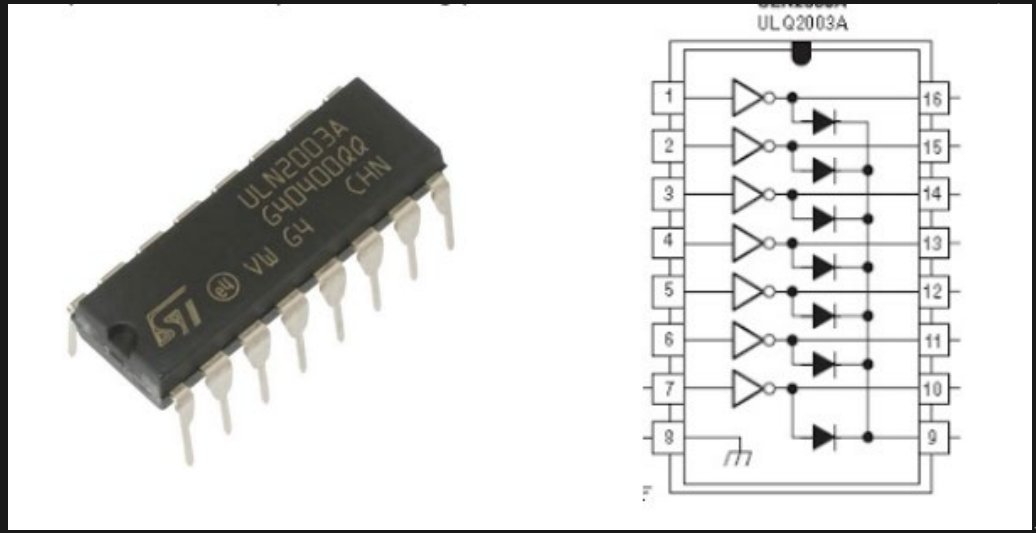
Stepper motor

When current flows through coil “1" the magnet is attracted and moves one step forward. Then, coil “1" is turned off and coil “2" is turned on. Now, the magnet takes another step, and so on. For a stepper motor to move, these coils should be activated in a correct sequence. In this simple orientation of coils, we assume that the step is 90 degree. For that the motor makes a full rotation in 4 steps only. But in reality, the motor may have more coils to achieve smaller step

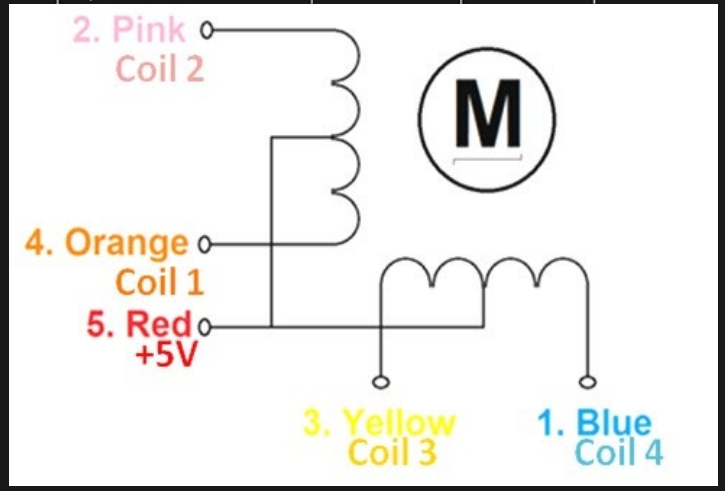
- In second image is used to achieve higher precision by having the rotor moving half step at time.



- This IC combines 7 Darlington pairs, each one has input for the base and output for its collector. If the input is 0, the output is GND if the input is 1, the output is floating pin



EUTS
Egyptian Universities Training Satellite



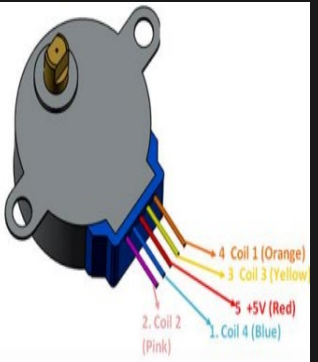
Half Step Mode Clockwise

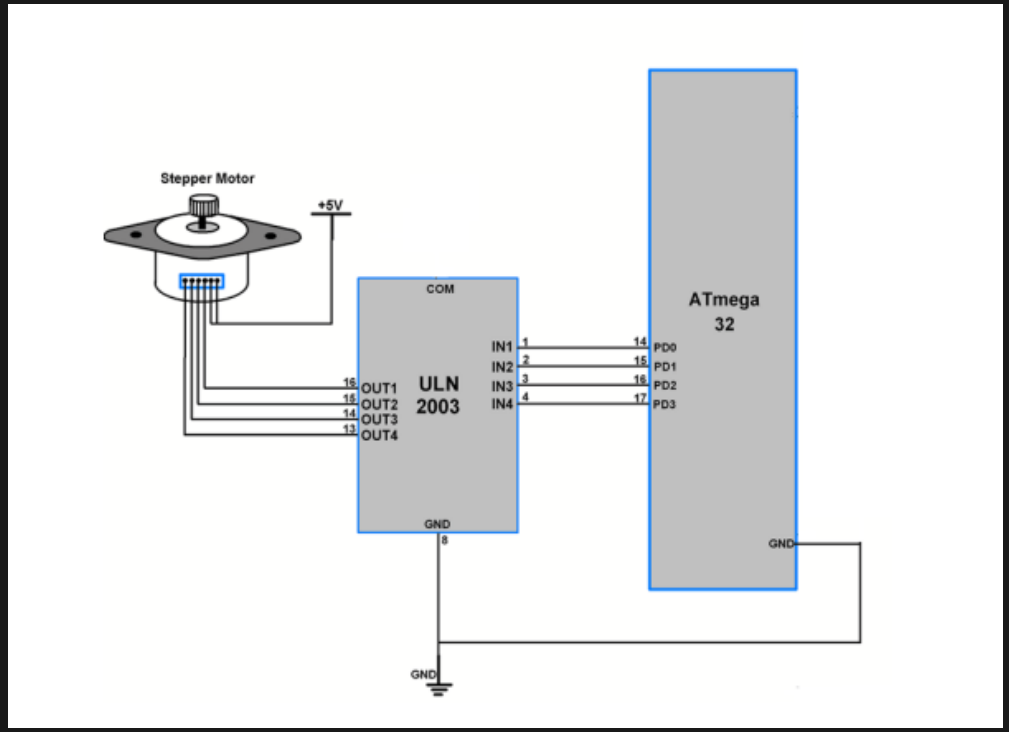
4 Orange	3 Yellow	2 Pink	1 Blue
1	0	0	1
0	0	0	1
0	0	1	1
0	0	1	0
0	1	1	0
0	1	0	0
1	1	0	0
1	0	0	0

Full Step Mode Clockwise

4 Orange	3 Yellow	2 Pink	1 Blue
0	0	0	1
0	0	1	0
0	1	0	0
1	0	0	0

Rated voltage :	5VDC
Number of Phase	4
Stride Angle	5.625°/64
Frequency	100Hz
In-traction Torque	>34.3mN.m(120Hz)
Self-positioning Torque	>34.3mN.m
Friction torque	600-1200 gf.cm
Pull in torque	300 gf.cm




Stepper Motor
+5V
COM
ULN
2003
OUT1
OUT2
OUT3
OUT4
IN1
IN2
IN3
IN4
GND
ATmega
32
PD0
PD1
PD2
PD3
GND
GND

Team members:

1- Dina Mohamed Elsayed

2- Farah Mohamed Kamel

3- Razan Bassel Said

4- Mohamed Hassan

5- Mohamed Khalid

6- Mohamed Saeed

EUTS
Egyptian Universities Training Satellite

Thank you