1. Perform basic Image Handling and processing operations on the image. • Read an image in python

and Convert an Image to Grayscale

AIM:

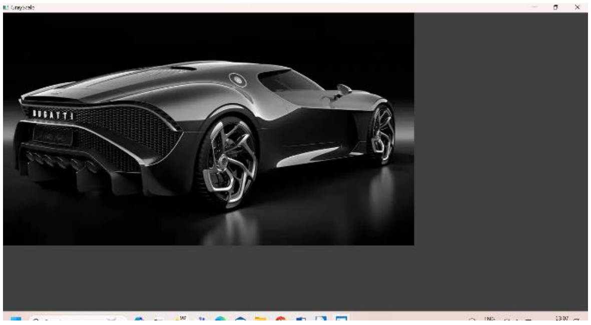
To Perform Basic Operations to Read Image and Convert to Grayscale using Python

PROGRAM

```
import cv2

import numpy as np

kernel = np.ones((5,5),np.uint8)

print(kernel)

path = r"C:\Users\ACER\Downloads\Buagti.jpg"

img =cv2.imread(path)

imgGray = cv2.cvtColor(img,cv2.COLOR_BGR2GRAY)

cv2.imshow("GrayScale",imgGray)

cv2.waitKey(0)
```

INPUT

OUTPUT



2. Perform basic Image Handling and processing operations on the image.• Read an image in python and Convert an Image to Blur using GaussianBlur.

AIM:

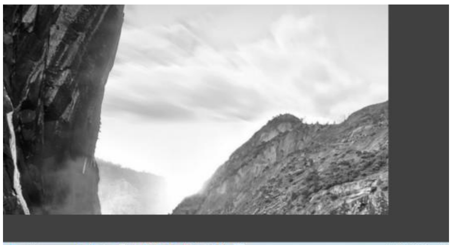
To Perform Basic Operations to Read Image and Convert to Blur using GaussianBlur.

PROGRAM:

```
import cv2

import numpy as np

kernel = np.ones((5,5),np.uint8)

print(kernel)

path = r"C:\Users\ACER\Downloads\Waterfall .jpg"

img =cv2.imread(path)

imgGray = cv2.cvtColor(img,cv2.COLOR_BGR2GRAY)

imgBlur = cv2.GaussianBlur(imgGray,(7,7),0)

cv2.imshow("Img Blur",imgBlur)

cv2.waitKey(0)
```

INPUT:

OUTPUT:



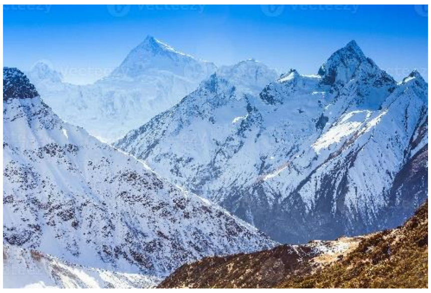
3. Perform basic Image Handling and processing operations on the image• Read an image in python and Convert an Image to show outline using Canny function

AIM: To Perform Basic Operations to Convert image to show outline Canny function in Python.

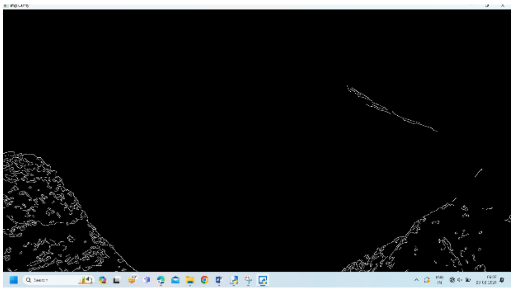
PROGRAM:

```
import cv2

import numpy as np

kernel = np.ones((5,5),np.uint8)

print(kernel)

path = r"C:\Users\ACER\Downloads\MOUNTAIN.jpg"

img =cv2.imread(path)

imgGray = cv2.cvtColor(img,cv2.COLOR_BGR2GRAY)

imgBlur = cv2.GaussianBlur(imgGray,(7,7),0)

imgCanny = cv2.Canny(imgBlur,100,200)

cv2.imshow("Img Canny",imgCanny)

cv2.waitKey(0)
```

INPUT:



OUTPUT:



4. Perform basic Image Handling and processing operations on the image• Read an image in python and Dilate an Image using Dilate function.

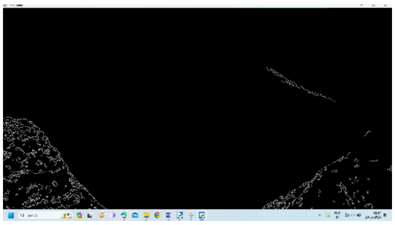
AIM: To Perform Basic Operations to Read Image and Dilate an Image using Python.

PROGRAM:

```
import cv2

import numpy as np

kernel = np.ones((5,5),np.uint8)

print(kernel)

path = r"C:\Users\ACER\Downloads\Waterfall .jpg"

img =cv2.imread(path)

imgGray = cv2.cvtColor(img,cv2.COLOR_BGR2GRAY)

imgBlur = cv2.GaussianBlur(imgGray,(7,7),0)

imgCanny = cv2.Canny(imgBlur,100,200)

imgDilation = cv2.dilate(imgCanny,kernel , iterations = 10)

imgEroded = cv2.erode(imgDilation,kernel,iterations=2)

cv2.imshow("Img Erosion",imgEroded)

cv2.waitKey(0)
```

INPUT:

OUTPUT:



5. Perform basic Image Handling and processing operations on the image• Read an image in python and Erode an Image using erode function

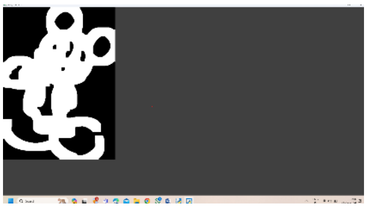
AIM: The Aim of the experiment is to Read an image in python and Erode an Image using erode function.

PROGRAM:

```
import cv2

import numpy as np

kernel = np.ones((5,5),np.uint8)

print(kernel)

path = r"C:\Users\ACER\Downloads\MICKYMOUSE.JPG"

img =cv2.imread(path)

imgGray = cv2.cvtColor(img,cv2.COLOR_BGR2GRAY)

imgBlur = cv2.GaussianBlur(imgGray,(7,7),0)

imgCanny = cv2.Canny(imgBlur,100,200)

imgDilation = cv2.dilate(imgCanny,kernel , iterations = 10)

imgEroded = cv2.erode(imgDilation,kernel,iterations=2)

cv2.imshow("Img Erosion",imgEroded)

cv2.waitKey(0)
```

INPUT:

OUTPUT:



6. Perform basic video processing operations on the captured video• Read captured video in python and display the video, in slow motion and in fast motion.

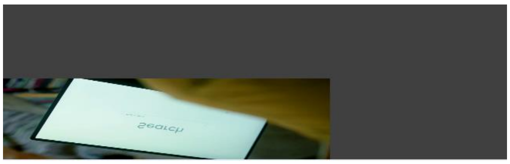
AIM: The Aim of the Experiment is to Read captured video in python and display the video, in slow motion and in fast motion.

PROGRAM:

```
import cv2

cap = cv2.VideoCapture(r"C:\Users\ACER\Pictures\www.5MovieRulz.gg - Tiger Nageswara Rao (2023) Telugu DVDScr x264 MP3 750MB.mkv")

if not cap.isOpened():

   print("Error opening video file")

while cap.isOpened():

   ret, frame = cap.read()

   if ret:

      cv2.imshow('Frame', frame)

      if cv2.waitKey(250) & 0xFF == ord('q'):

        break

   else:

      break

cap.release()

cv2.destroyAllWindows()
```

INPUT:

OUTPUT:



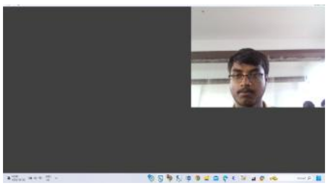
7. Capture video from web Camera and Display the video, in slow motion and in fast motion operations on the captured video

AIM: The Aim is to Capture video from web Camera and Display the video, in slow motion and in fast motion operations on the captured video.

PROGRAM:

```
import cv2
cap = cv2.VideoCapture(0)
height = int(cap.get(cv2.CAP_PROP_FRAME_HEIGHT))
width = int(cap.get(cv2.CAP_PROP_FRAME_WIDTH))
fps = cap.get(cv2.CAP_PROP_FPS)
path = "output.mp4"
fourcc = cv2.VideoWriter_fourcc(*'mp4v')
output = cv2.VideoWriter(path, fourcc, 2, (width, height))
while True:
   ret, frame = cap.read()
   cv2.imshow("frame", frame)
   output.write(frame)
   k = cv2.waitKey(24)
   if k == ord("q"):
      break
cap.release()
output.release()
cv2.destroyAllWindows()
```

OUTPUT:



8. Scaling an image to its Bigger and Smaller sizes.

AIM:

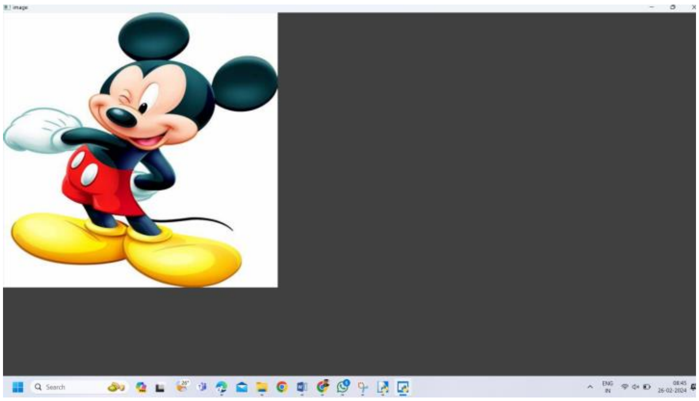
The Aim is resize the image from bigger to smaller size.

PROGRAM:

```
import cv2

import numpy as np

kernel = np.ones((5,5),np.uint8)

img = cv2.imread(r"C:\Users\ACER\Downloads\MICKYMOUSE.JPG",cv2.IMREAD_COLOR)

img = cv2.resize(img,(600,600))

cv2.imshow("image",img)

cv2.waitKey(0)
```

INPUT:

OUTPUT:



9. Perform Rotation of an image to clockwise and counter clockwise direction.

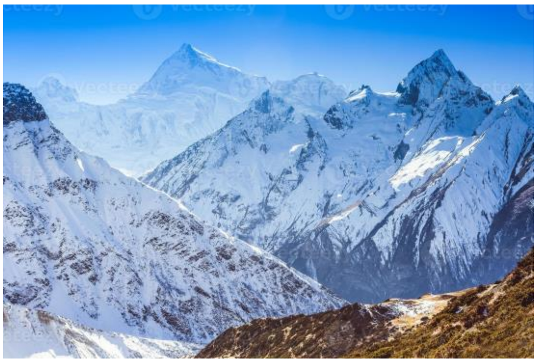
AIM:

The Aim of the Experiment is to perform Rotation of an image along 90 degree.

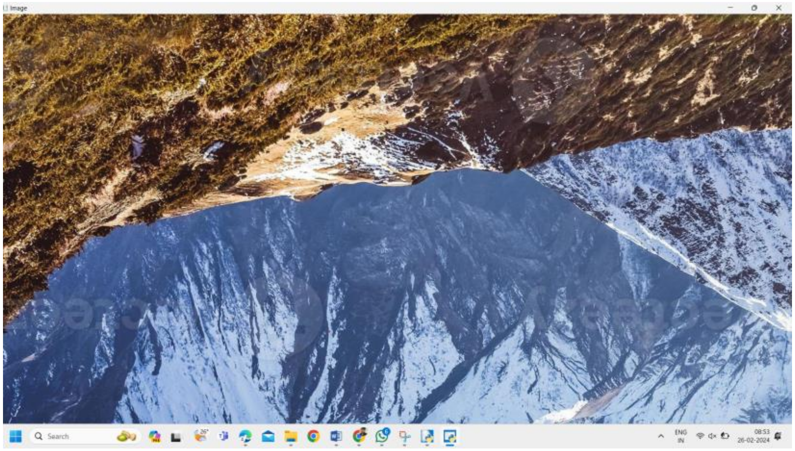
PROGRAM:

```
import cv2

path = r"C:\Users\ACER\Downloads\MOUNTAIN.jpg"

src = cv2.imread(path)

window_name = 'Image'

image = cv2.rotate(src, cv2.ROTATE_180)

cv2.imshow(window_name, image)

cv2.waitKey(0)
```

INPUT:



OUTPUT:



ROTATION ALONG 180 DEGREE

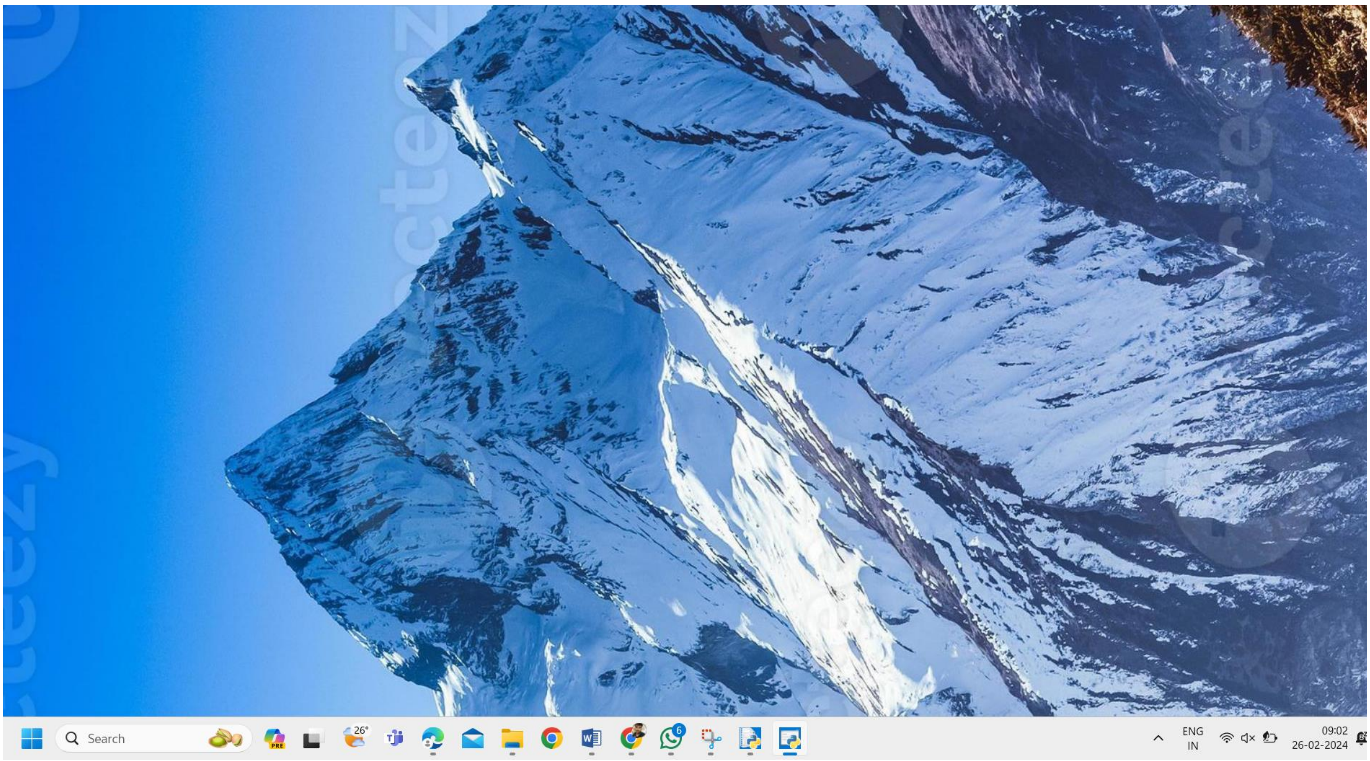
AIM:

The Aim of the Experiment is to perform Rotation of an image along 180 degree.

PROGRAM:

```
import cv2

path = r"C:\Users\ACER\Downloads\MOUNTAIN.jpg"

src = cv2.imread(path)

window_name = 'Image'

image = cv2.rotate(src, cv2.ROTATE_90_COUNTERCLOCKWISE)

cv2.imshow(window_name, image)

cv2.waitKey(0)
```

OUTPUT:



10.ROTATION ALONG 270 DEGREE

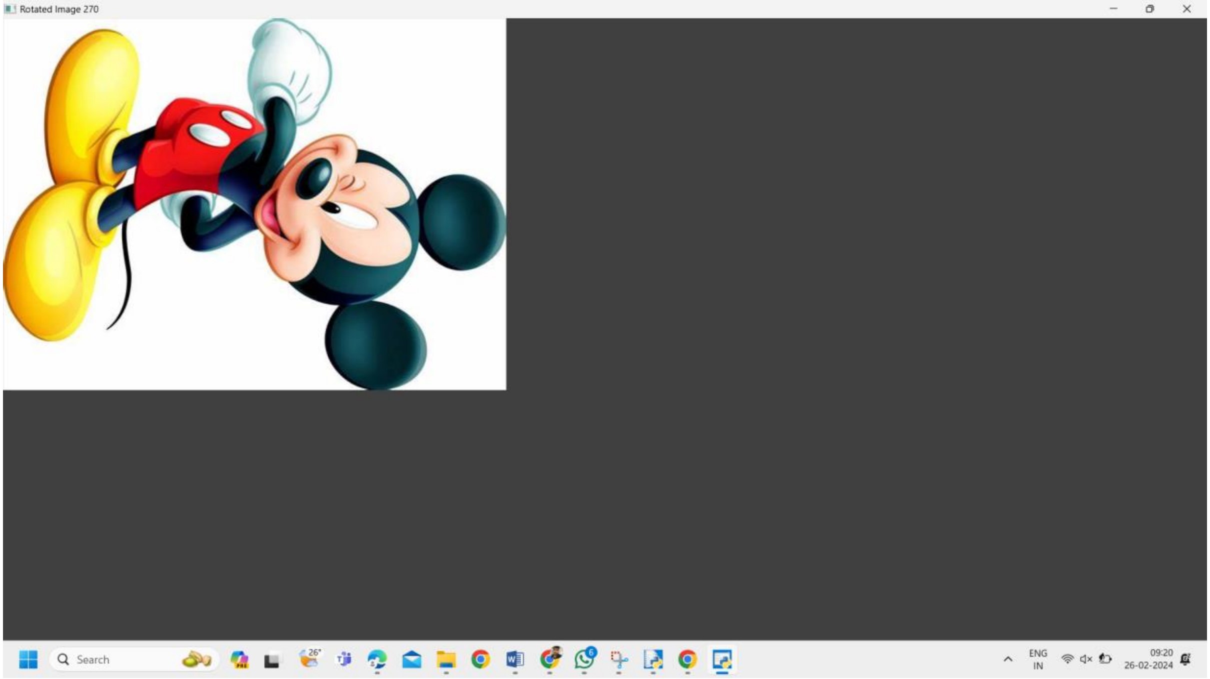
AIM:

The Aim of the Experiment is to perform Rotation of an image along 270 degree.

PROGRAM:

```
import cv2

path = r"C:\Users\ACER\Downloads\MICKYMOUSE.JPG"

src = cv2.imread(path)

image_rotated_270 = cv2.rotate(src, cv2.ROTATE_90_CLOCKWISE)

cv2.imshow('Rotated Image 270', image_rotated_270)

cv2.waitKey(0)

cv2.destroyAllWindows()
```

OUTPUT:


Rotated Image 270
Search
ENG
IN
09:20
26-02-2024