

Methodology

1. We will use the OpenCV library from Python 3 from anaconda navigator(anaconda3)
2. OpenCV provides a module called ml that has many machine learning algorithms bundled into it. Some of the algorithms include **Bayes Classifier**, **K-Nearest Neighbors**, **Support Vector Machines**, **Decision Trees**, **Neural Networks**, and so on.
3. How to increase the number of images in a dataset:
 - 3.1. Using paint to change size and position
 - 3.2. Using powerpoint to rotate the images one degree at a time
 - 3.3. Using OpenCV to add white noise to each image
4. Photo samples that I am going to take and use to learn my machine is something like this which is in a microscopic scale (500 Nanometer)

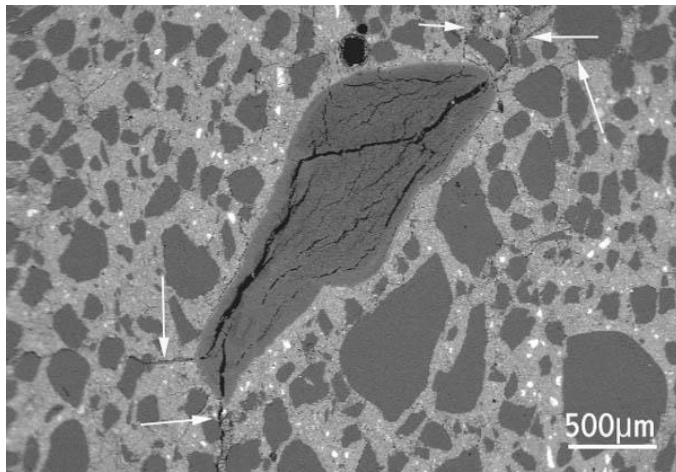


Figure 1

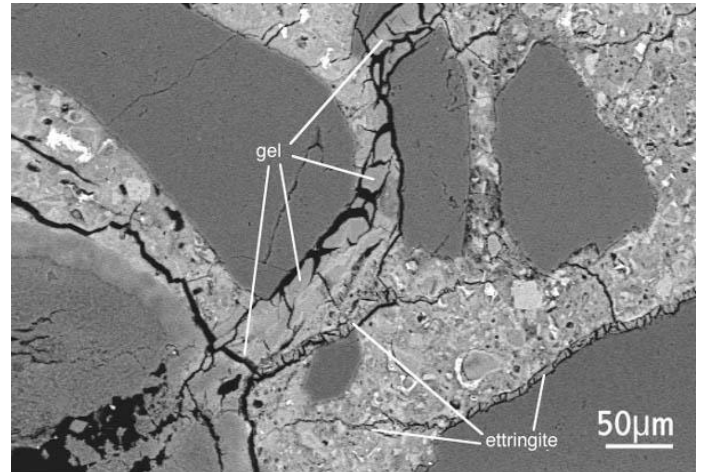


Figure 2

5. The code we use for recognizing the characteristic of cracks is:

5.1.A Combined Corner and Edge Detector code in OpenCV

```
import cv2
import numpy as np

filename = 'chessboard2.jpg'
img = cv2.imread(filename)
gray = cv2.cvtColor(img,cv2.COLOR_BGR2GRAY)

# find Harris corners
gray = np.float32(gray)
dst = cv2.cornerHarris(gray,2,3,0.04)
dst = cv2.dilate(dst,None)
ret, dst = cv2.threshold(dst,0.01*dst.max(),255,0)
dst = np.uint8(dst)

# find centroids
ret, labels, stats, centroids = cv2.connectedComponentsWithStats(dst)

# define the criteria to stop and refine the corners
criteria = (cv2.TERM_CRITERIA_EPS + cv2.TERM_CRITERIA_MAX_ITER, 100, 0.001)
corners = cv2.cornerSubPix(gray,np.float32(centroids),(5,5),(-1,-1),criteria)

# Now draw them
res = np.hstack((centroids,corners))
res = np.int0(res)
img[res[:,1],res[:,0]]=[0,0,255]
img[res[:,3],res[:,2]] = [0,255,0]

cv2.imwrite('subpixel5.png',img)
```

5.2.Object Detection code in OpenCV

```
import numpy as np
import cv2

face_cascade = cv2.CascadeClassifier('haarcascade_frontalface_default.xml')
eye_cascade = cv2.CascadeClassifier('haarcascade_eye.xml')

img = cv2.imread('sachin.jpg')
gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)

then

faces = face_cascade.detectMultiScale(gray, 1.3, 5)
for (x,y,w,h) in faces:
    img = cv2.rectangle(img,(x,y),(x+w,y+h),(255,0,0),2)
    roi_gray = gray[y:y+h, x:x+w]
    roi_color = img[y:y+h, x:x+w]
    eyes = eye_cascade.detectMultiScale(roi_gray)
    for (ex,ey,ew,eh) in eyes:
```

```
cv2.rectangle(roi_color, (ex, ey), (ex+ew, ey+eh), (0, 255, 0), 2)

cv2.imshow('img', img)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

Purpose

Our mission is to characterize the cracks of normal concrete and then recycled aggregate concrete to teach the machine so the classifier algorithms can classify the normal concrete from the recycled one.