Oral Cancer Prediction using Convultion Neural Network(CNN)

Data Processing

1) Importing the neccessary python Libraries

In [1]:
```
!pip install opencv-python
```

```
Collecting opencv-python
  Using cached opencv_python-4.5.2.54-cp38-cp38-win_amd64.whl (34.7 MB)
Requirement already satisfied: numpy>=1.17.3 in c:\programdata\anaconda3\lib\site-pac
kages (from opencv-python) (1.19.2)
Installing collected packages: opencv-python
Successfully installed opencv-python-4.5.2.54
```

In [2]:
```
import numpy as np
import cv2
import os
import random
import matplotlib.pyplot as plt
import pickle
```

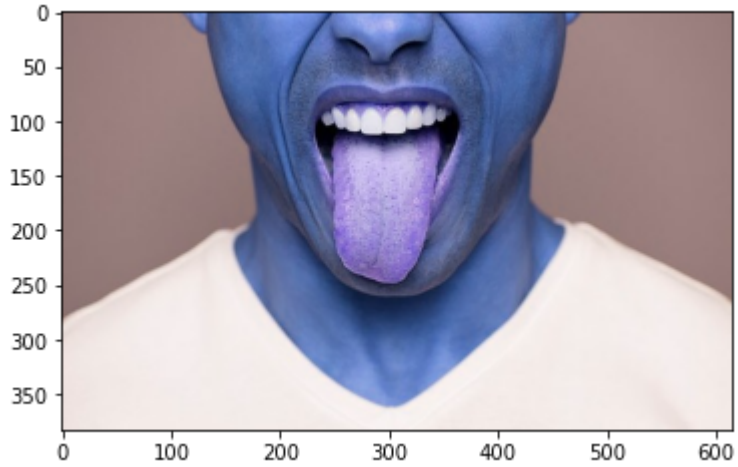
2) Importing the dataset folders containg images of mouth and tongue with oral cancer and without oral cancer

In [3]:
```
DIRECTORY = r"C:\Users\dell\Desktop\machine-learning-
ex\tpcs\OralCancer"
CATEGORIES = ['cancer','non-cancer']
```

In [4]:
```
for category in CATEGORIES:
    folder = os.path.join(DIRECTORY,category)
    for img in os.listdir(folder):
        img_path = os.path.join(folder,img)
        print(img_path)
        break
```

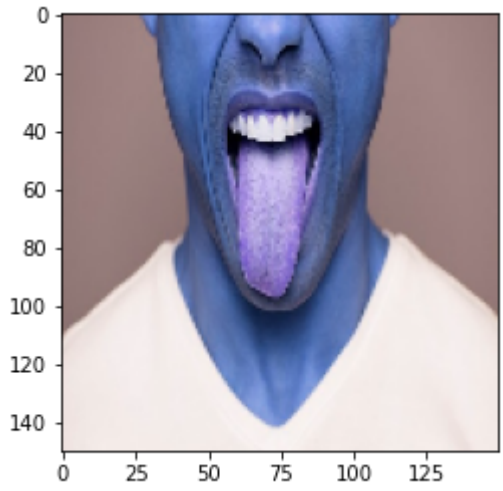
```
C:\Users\dell\Desktop\machine-learning-ex\tpcs\OralCancer\cancer\cancer(1).jpeg
C:\Users\dell\Desktop\machine-learning-ex\tpcs\OralCancer\non-cancer\12654650-6954555
-image-a-22_1556101508834-Edited.jpg
```

In [5]:
```
for category in CATEGORIES:
    folder = os.path.join(DIRECTORY,category)
    for img in os.listdir(folder):
        img_path = os.path.join(folder,img)
        img_array = cv2.imread(img_path)
        plt.imshow(img_array)
        break
```



In [6]:

```
Img_size = 150
for category in CATEGORIES:
    folder = os.path.join(DIRECTORY,category)
    for img in os.listdir(folder):
        img_path = os.path.join(folder,img)
        img_array = cv2.imread(img_path)
        img_array = cv2.resize(img_array,(Img_size,Img_size))
        plt.imshow(img_array)
        break
```



In [7]:

```
Img_size = 150
data = []
for category in CATEGORIES:
    folder = os.path.join(DIRECTORY,category)
    label = CATEGORIES.index(category)
    for img in os.listdir(folder):
        img_path = os.path.join(folder,img)
        img_array = cv2.imread(img_path)
        img_array = cv2.resize(img_array,(Img_size,Img_size))
        data.append([img_array, label])
```

In [8]:

```
len(data)
```

Out[8]:

```
131
```

In [9]:
```
random.shuffle(data)
```

In [10]:
```
data[0]
```

Out[10]:
```
[array([[[149, 179, 218],
         [138, 182, 235],
         [141, 190, 252],
         ...,
         [172, 198, 245],
         [179, 206, 253],
         [197, 209, 241]],

        [[152, 181, 220],
         [143, 187, 240],
         [130, 179, 241],
         ...,
         [176, 200, 241],
         [184, 206, 248],
         [201, 214, 246]],

        [[158, 188, 227],
         [137, 181, 234],
         [127, 176, 239],
         ...,
         [190, 208, 242],
         [196, 212, 246],
         [203, 215, 247]],

        ...,

        [[149, 179, 227],
         [131, 171, 234],
         [129, 169, 234],
         ...,
         [130, 164, 234],
         [130, 164, 234],
         [116, 170, 236]],

        [[154, 184, 231],
         [136, 176, 239],
         [134, 174, 240],
         ...,
         [125, 159, 229],
         [125, 159, 229],
         [124, 176, 236]],

        [[148, 179, 226],
         [142, 182, 245],
         [137, 177, 243],
         ...,
         [135, 169, 239],
         [138, 172, 242],
         [126, 173, 226]]], dtype=uint8),
 0]
```

In [11]:
```
x = []
y = []
for features,labels in data:
    x.append(features)
    y.append(labels)
```

In [12]:
```
x = np.array(x)
y = np.array(y)
```

In [13]:
```

```

```
len(x),len(y)
```

Out[13]:
```
(131, 131)
```

In [14]:
```
pickle.dump(x,open('x.pkl','wb'))
pickle.dump(y,open('y.pkl','wb'))
```