Closing

```
In [ ]: import numpy as np
        import cv2
        def pad(img,shp):
                p=np.zeros((shp[0]+1,shp[1]+1))
                p[1:,1:]=np.copy(img)
                p[0,1:]=img[0]
                p[1:,0]=img[:,0]
                p[0,0]=img[0,0]
                p[-1,0]=img[-1,0]
                return p

        def comp_erosion(sample,metric):
                for i in range(2):
                        for j in range(2):
                                if sample[i,j]!=metric[i,j]:
                                        return False
                return True

        def comp_dilation(sample,metric):
                for i in range(2):
                        for j in range(2):
                                if sample[i,j]==metric[i,j]:
                                        return True
                return False

        def slice(img):
                temp=np.array(img)
                print(temp.shape)
                int_slice=np.zeros((temp.shape[0],temp.shape[1],8))
                for x in range(8):
                        int_slice[:,:,x]=temp%(2)
                        temp=(temp/2).astype(int)
                return int_slice

        def stitch(int_slice,shp):
                out=np.zeros(shp)
                for x in range(8):
```

```python
                out=out+int_slice[:,:,x]*(2**x)
        return out


img=cv2.imread('cat.jpeg',0)
shp=img.shape
temp=pad(img,shp)
int_slice=slice(temp)
struct_el=np.array([[1,1],[1,1]])
int_slice_new=np.zeros((shp[0],shp[1],8))
for x in range(8):
        for i in range(shp[0]):
                for j in range(shp[1]):
                        if comp_dilation(int_slice[i:i+2,j:j+2,x],struct_el):
                                int_slice_new[i,j,x]=1


int_slice=np.array(int_slice_new)
int_slice_new=np.zeros((shp[0],shp[1],8))
for x in range(8):
        test=pad(int_slice[:,:,x],shp)
        for i in range(shp[0]):
                for j in range(shp[1]):
                        if comp_erosion(test[i:i+2,j:j+2],struct_el):
                                int_slice_new[i,j,x]=1


out=stitch(int_slice_new,shp)
out=np.array(out, dtype = np.uint8)
print(img)
print(out)
cv2.imshow('image', img)
cv2.imshow('closed', out)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

```
(351, 529)
[[14 21 29 ...  7  7  7]
 [13 20 28 ...  6  6  6]
 [12 19 28 ...  6  6  6]
 ...
 [ 8  7  5 ...  0  0  0]
 [ 9  8  6 ...  0  0  0]
 [10  9  7 ...  0  0  0]]
[[14 14 29 ...  7  7  7]
 [14 14 29 ...  7  7  7]
 [13 13 29 ...  6  6  6]
 ...
 [ 7  7  6 ...  0  0  0]
 [ 9  9  7 ...  0  0  0]
```

```
 [ 9  9 11 ...  0  0  0]]
```

In [ ]: