Computer Vision HW4, Binary Morphology Report

tags: NTU CS Computer Vision Writeup Report

NTU CSIE, R08922024, Alfons Hwu
Prequisites and env as the following

```
Ubuntu WSL for windows with jupyter notebook
Python3.6.7
OpenCV for image IO
Matplotlib for displaying image
```

a, dilation

```
def dilation(a, b):
    ra, ca = a.shape
    res = np.zeros(a.shape, dtype = 'int32')

    for ai in range(ra):
        for aj in range(ca):
            if a[ai, aj] == 0xff:

                # assign original image position
                res[ai, aj] = 0xff
                for b_each in b:
                    bi, bj = b_each
                    if  ai + bi >= 0 and ai + bi < ra \
                    and aj + bj >= 0 and aj + bj < ca:
                        # extend the value
                        res[ai + bi, aj + bj] = 0xff

    return res
```

b, erosion

```
def erosion(a, b):
    ra, ca = a.shape # original image
    res = np.zeros(a.shape, dtype = 'int32')

    for ai in range(ra):
        for aj in range(ca):
            if a[ai, aj] > 0:

                ok = 1
                for b_each in b:
                    bi, bj = b_each
                    if ai + bi >= ra or aj + bj >= ca \
                    or ai + bi <  0  or aj + bj <  0  \
                    or a[ai + bi, aj + bj] == 0:
                        ok = 0
                        break

                if ok == 1:
                    res[ai, aj] = 255

    return res
```

c, closing

```
def closing(a, b):
    return erosion(dilation(a, b), b)
```

d, opening

```
def opening(a, b):
    return dilation(erosion(a, b), b)
```

e, hit and miss transformation

```
def hit_and_miss(a, j, k):
    a_c = (-a) + 255
    a_j = erosion(a, j)
    a_k = erosion(a_c, k)
    res = (a_j + a_k) / 2

    # intersect with add up and div 2 to see whether still 255
    return (res == 255) * 255
```