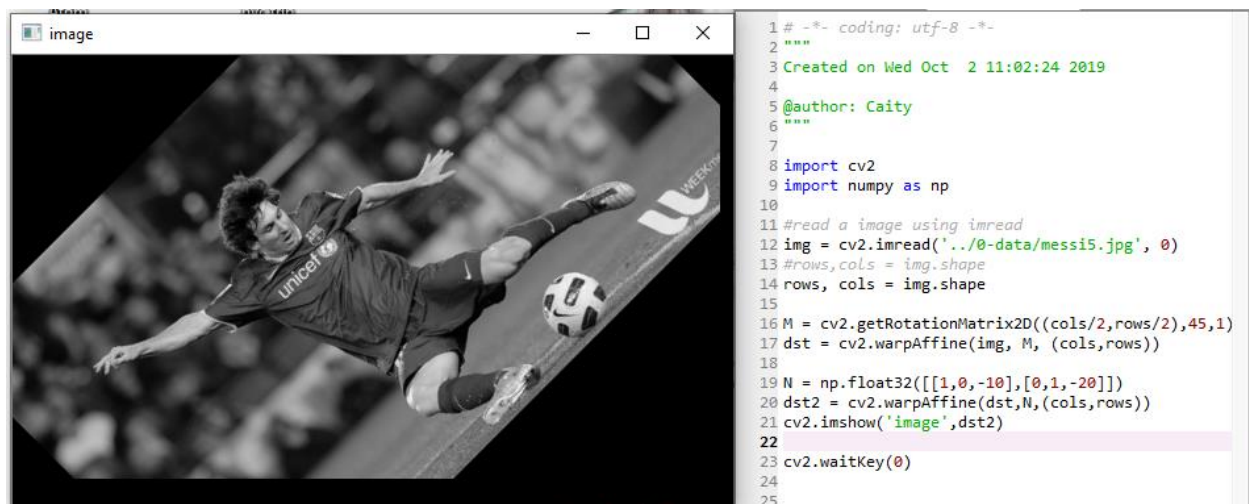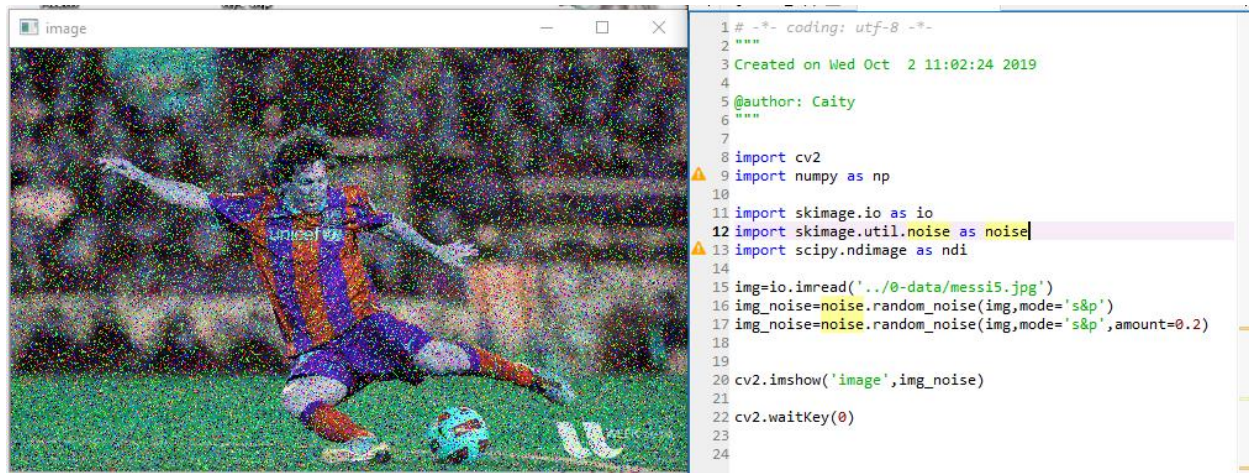**Caitlin Harris**

CSC 317: Homework 2

4 November 2019

1. Geometric Transformation – Find an image and rotate the image 45 degrees counter-clockwise and then translate the image with the displacement values of (-10, -20) to the (x, y) coordinates of all pixels. Show the image after each transformation.
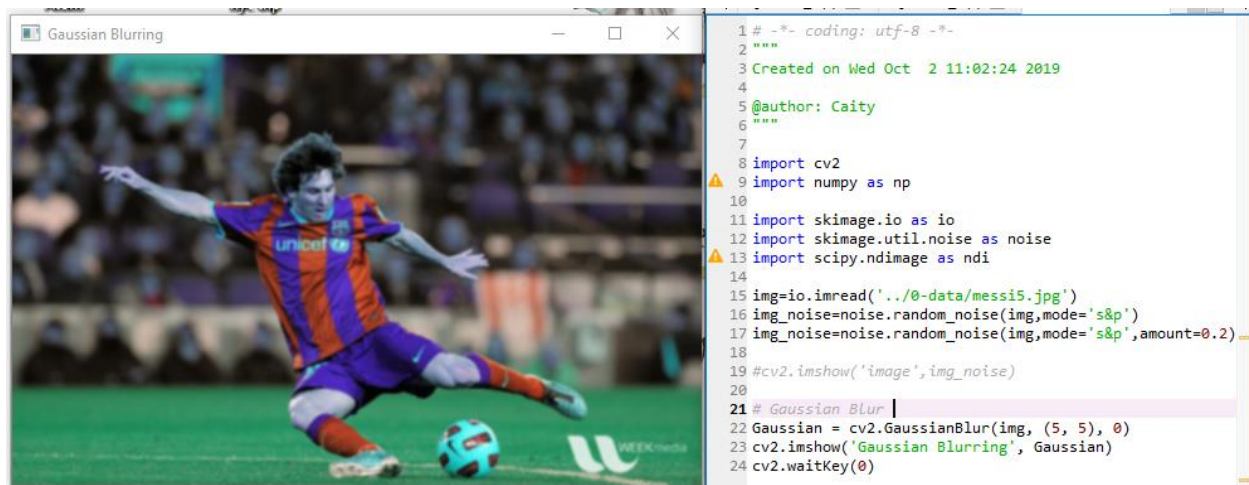
2. Smoothing Images –Take an image, add 20% salt and pepper noise, compare the effect of blurring to remove the noise with box filter (average filter), Gaussian filter, and median filter. Which filter gives the best result? (Hint: Use Chapter 8 of our textbook for adding noises to an image.)
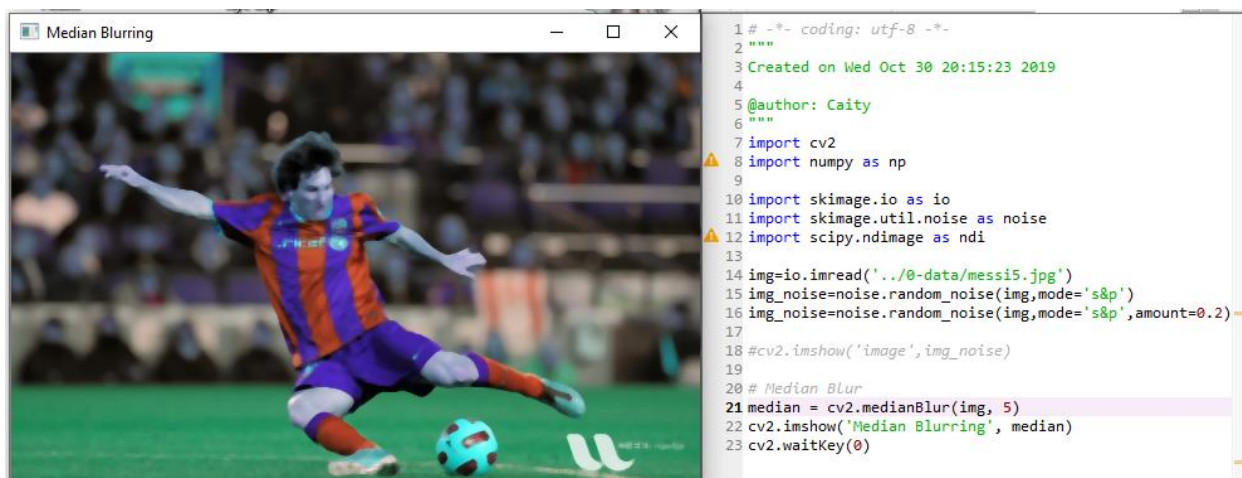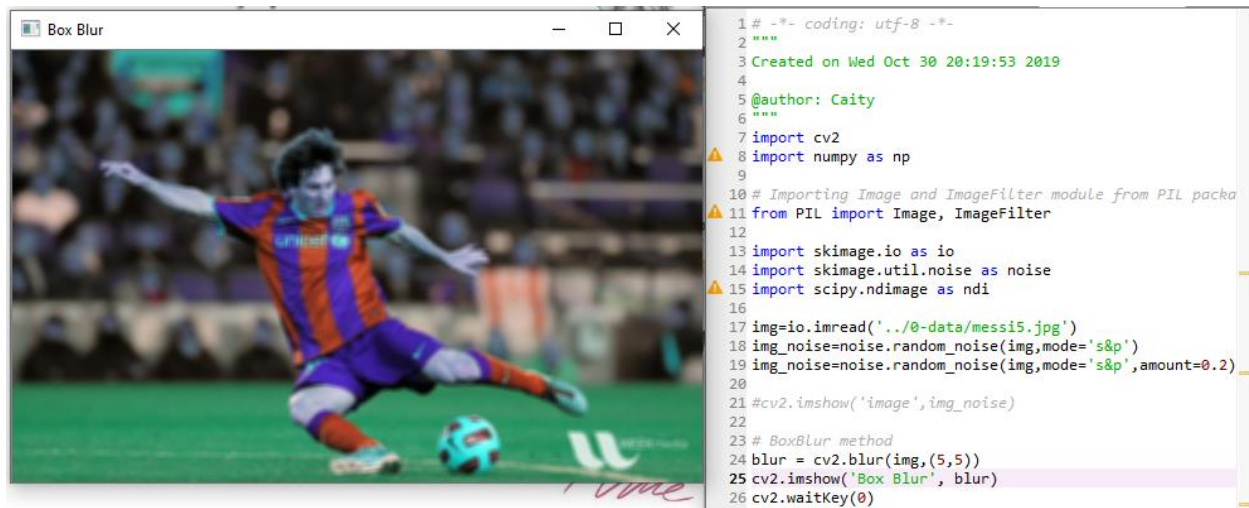
20% salt and pepper noise:


```python
# -*- coding: utf-8 -*-
"""
Created on Wed Oct  2 11:02:24 2019

@author: Caity
"""

import cv2
import numpy as np

import skimage.io as io
import skimage.util.noise as noise
import scipy.ndimage as ndi

img=io.imread('../0-data/messi5.jpg')
img_noise=noise.random_noise(img,mode='s&p')
img_noise=noise.random_noise(img,mode='s&p',amount=0.2)


cv2.imshow('image',img_noise)

cv2.waitKey(0)
```

Gaussian Filter:


```python
# -*- coding: utf-8 -*-
"""
Created on Wed Oct  2 11:02:24 2019

@author: Caity
"""

import cv2
import numpy as np

import skimage.io as io
import skimage.util.noise as noise
import scipy.ndimage as ndi

img=io.imread('../0-data/messi5.jpg')
img_noise=noise.random_noise(img,mode='s&p')
img_noise=noise.random_noise(img,mode='s&p',amount=0.2)

#cv2.imshow('image',img_noise)

# Gaussian Blur
Gaussian = cv2.GaussianBlur(img, (5, 5), 0)
cv2.imshow('Gaussian Blurring', Gaussian)
cv2.waitKey(0)
```

Median Filter:


```python
# -*- coding: utf-8 -*-
"""
Created on Wed Oct 30 20:15:23 2019

@author: Caity
"""
import cv2
import numpy as np

import skimage.io as io
import skimage.util.noise as noise
import scipy.ndimage as ndi

img=io.imread('../0-data/messi5.jpg')
img_noise=noise.random_noise(img,mode='s&p')
img_noise=noise.random_noise(img,mode='s&p',amount=0.2)

#cv2.imshow('image',img_noise)

# Median Blur
median = cv2.medianBlur(img, 5)
cv2.imshow('Median Blurring', median)
cv2.waitKey(0)
```

Box Filter:


```
1 # -*- coding: utf-8 -*-
2 """
3 Created on Wed Oct 30 20:19:53 2019
4
5 @author: Caity
6 """
7 import cv2
8 import numpy as np
9
10 # Importing Image and ImageFilter module from PIL packa
11 from PIL import Image, ImageFilter
12
13 import skimage.io as io
14 import skimage.util.noise as noise
15 import scipy.ndimage as ndi
16
17 img=io.imread('../0-data/messi5.jpg')
18 img_noise=noise.random_noise(img,mode='s&p')
19 img_noise=noise.random_noise(img,mode='s&p',amount=0.2)
20
21 #cv2.imshow('image',img_noise)
22
23 # BoxBlur method
24 blur = cv2.blur(img,(5,5))
25 cv2.imshow('Box Blur', blur)
26 cv2.waitKey(0)
```

Conclusion:

The notes state that the median filter is most effective in removing salt and pepper noise, I slightly disagree in this instance. Judging by the pictures above, the Gaussian Filter seems to remove the most noise from the picture, making it the least blurry option.

3. Find the convolution of x[t] * h[t] if x[t]=[1, 2, 3, 1], t=0,1,2,and 3; h[t] = [1, 1, 1], t=-1, 0, 1. Show your computation.