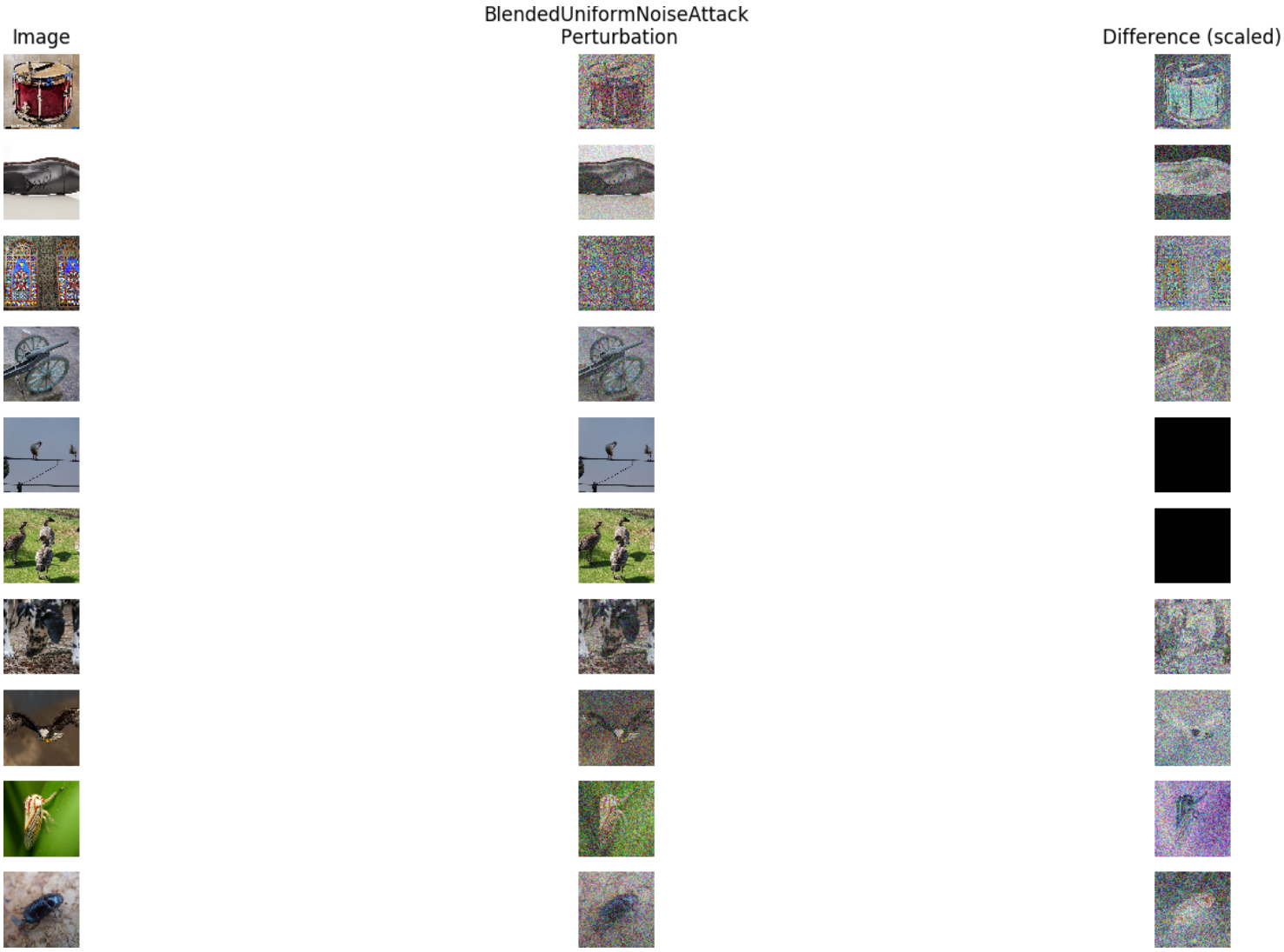
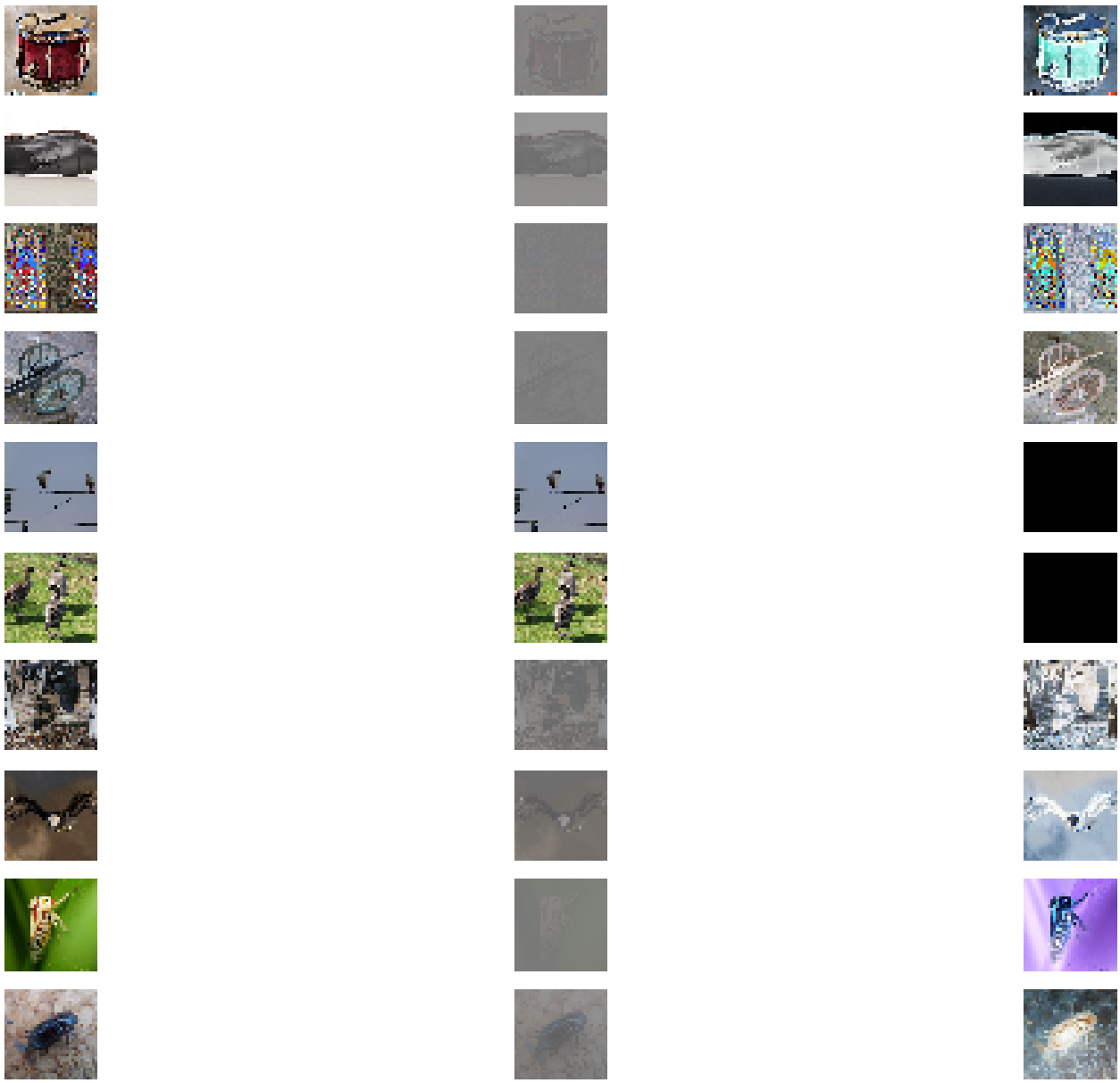
```
import foolbox
import keras
import numpy as np
from keras.applications.resnet50 import ResNet50
import matplotlib.pyplot as plt
import matplotlib.image as mpimg
import sklearn
import sys
# instantiate model
keras.backend.set_learning_phase(0)
kmodel = ResNet50(weights='imagenet')
preprocessing = dict(flip_axis=-1, mean=np.array([104, 116, 123]))  # RGB
to BGR and mean subtraction
fmodel = foolbox.models.KerasModel(kmodel, bounds=(0, 255),
preprocessing=preprocessing)
for attacks in
['BlendedUniformNoiseAttack','ContrastReductionAttack','FGSM','SinglePixe
lAttack','SaliencyMapAttack']:
    for i in range(10):
        # get source image and label, unique idx: 0-19
        image, label =
foolbox.utils.samples(dataset='imagenet',index=10+i,batchsize=1)
        image = image[0]
        label = label[0]
        # apply attack on source image
        if attacks == 'FGSM':
            attack = foolbox.v1.attacks.FGSM(fmodel)
        elif attacks == 'BlendedUniformNoiseAttack':
            attack = foolbox.v1.attacks.BlendedUniformNoiseAttack(fmodel)
        elif attacks == 'ContrastReductionAttack':
            attack = foolbox.v1.attacks.ContrastReductionAttack(fmodel)
        elif attacks == 'SinglePixelAttack':
            attack = foolbox.v1.attacks.SinglePixelAttack(fmodel)
        else:
            attack = foolbox.v1.attacks.SaliencyMapAttack(fmodel)
        adversarial = attack(image, label)
        # if the attack fails, adversarial will be None and a warning will
be printed
        plt.subplot(10, 3, (i*3)+1)
        plt.axis('off')
        if i==0:
            plt.title('Image')
        plt.imshow(image.astype(np.uint8))
        plt.subplot(10, 3, (i*3)+2)
        plt.axis('off')
        if i == 0:
            plt.title(attacks+'\n Perturbation')
        plt.imshow(adversarial.astype(np.uint8))
        plt.subplot(10, 3, (i*3)+3)
        plt.axis('off')
```

```
        if i == 0:
            plt.title('Difference (scaled)')
        diff = adversarial - image
        plot_diff = 255 * (diff - np.min(diff)) / (np.max(diff) -
np.min(diff)+sys.float_info.epsilon)
        plt.imshow(plot_diff.astype(np.uint8))
    plt.show()
```



ContrastReductionAttack

Image | Perturbation | Difference (scaled)



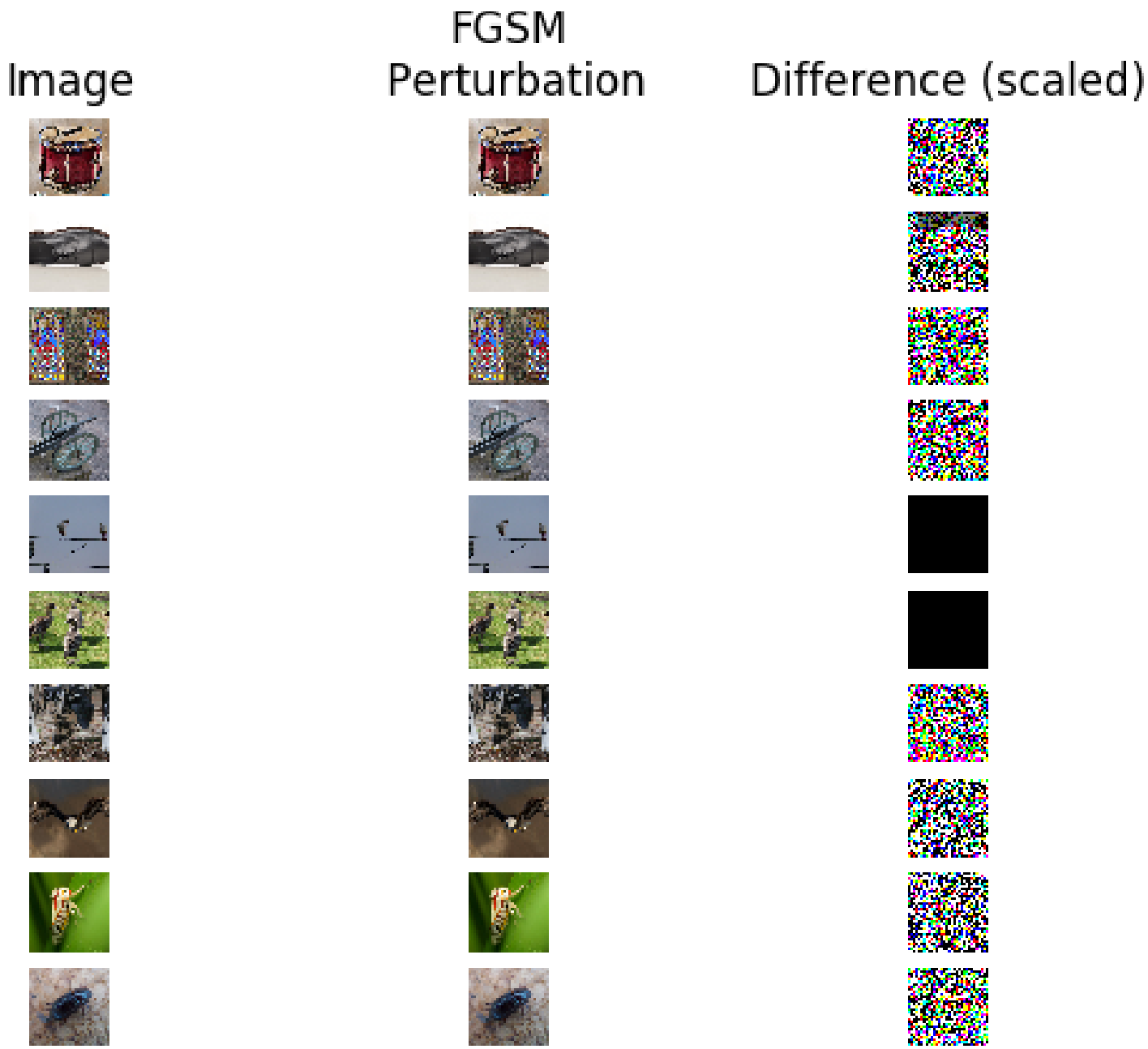


Image
FGSM
Perturbation
Difference (scaled)

All Single Pixel Attacks failed
None returned for all attacks on VGG16

SaliencyMapAttack

Image | Perturbation | Difference (scaled)