In [ ]:

```
import torch
import torch.nn as nn
import torchvision
import torchvision.transforms as transforms
from torchvision import transforms
from torchvision.datasets import ImageFolder
import matplotlib.pyplot as plt
from PIL import Image
import sys
import os
import glob
import cv2
```

Data Augmentation:

flip horizontally and vertically

Image augmentation is one useful technique in building convolutional neural networks that can increase the size of the training set without acquiring new images. The idea is simple; duplicate images with some kind of variation so the model can learn from more examples. I use OpenCV library to increase the number of my datatraining set images with flip horizantall and vertically and also rotate 90 degree methods.

My priviuose training set has almpst 34000 images and after data Augmentation it increased to more than 270000 images, almost 8 times.

Flip functions:

In [ ]:

```
def main():
    path_of_the_directory = "/Users/azadehbaghdadi/Desktop/New research-CNN/Proj
    for filename in os.listdir(path_of_the_directory):
        img_dir = os.path.join(path_of_the_directory,filename)
        jpg_ext = ".jpg"
        path = "/Users/azadehbaghdadi/Desktop/New research-CNN/Project/Data/test
        angles = [0]
        for file_name in glob.iglob(os.path.join(img_dir, "*" + jpg_ext)):
            original_img = Image.open(file_name)
            new_image = original_img.resize((128, 128))
            for angel in angles:
                rot_suffix = "_r{:03d}{:s}".format(angel, jpg_ext)
                file_name_rot = file_name.replace(jpg_ext, rot_suffix)
                vertical_img = new_image.transpose(method=Image.FLIP_TOP_BOTTOM)
                vertical_img.save(file_name_rot)
                print("Rotated: {:s} by {:3d} degrees to {:s}".format(file_name,

        angles = [1]
        for file_name in glob.iglob(os.path.join(img_dir, "*" + jpg_ext)):
            original_img = Image.open(file_name)
            new_image = original_img.resize((128, 128))
            for angel in angles:
                rot_suffix = "_r{:03d}{:s}".format(angel, jpg_ext)
                file_name_rot = file_name.replace(jpg_ext, rot_suffix)
                horz_img = new_image.transpose(method=Image.FLIP_LEFT_RIGHT)
                horz_img.save(file_name_rot)
                print("Rotated: {:s} by {:3d} degrees to {:s}".format(file_name,




if __name__ == "__main__":
    print("Python {:s} on {:s}\n".format(sys.version, sys.platform))
    main()
    print("\nDone.")
```

Rotate functions:

In [ ]:

```
def main():
    path_of_the_directory = "/Users/azadehbaghdadi/Desktop/New research-CNN/Proj
    for filename in os.listdir(path_of_the_directory):
        img_dir = os.path.join(path_of_the_directory,filename)
        jpg_ext = ".jpg"
        angles = [90]
        for file_name in glob.iglob(os.path.join(img_dir, "*" + jpg_ext)):
            #image = Image.open(file_name)
            original_img = Image.open(file_name)
            new_image = original_img.resize((128, 128))
            for angle in angles:
                rot_suffix = "_r{:03d}{:s}".format(angle, jpg_ext)
                file_name_rot = file_name.replace(jpg_ext, rot_suffix)
                image_rot = new_image.rotate(angle)
                image_rot.save(file_name_rot)
                print("Rotated: {:s} by {:3d} degrees to {:s}".format(file_name,


if __name__ == "__main__":
    print("Python {:s} on {:s}\n".format(sys.version, sys.platform))
    main()
    print("\nDone.")
```