试题要求

任务目标：

针对深度学习图像识别模型的自动化测试框架，设计并实现一个 Python 实现的基于 TensorFlow 的深度学习图像识别模型的自动化测试方法，采用特定的方式，根据提供的训练数据集和待测数据集，由待测数据集尽量生成使得模型出错但是和原始数据“相似度”高的测试数据。

测试对象：

针对某个图像识别模型的待测数据集进行测试。对模型未知，对训练数据集和待测数据集已知。其中训练数据集将使用 Fashion-MNIST 数据集。（黑盒测试）

算法评估：

1. 每位同学提交的算法脚本将会在后台测试 1000 张图片，依据生成的对抗样本图片计算得分。
2. 得分前提为输出的样本成功让模型识别出错（黑盒攻击成功），出现错误分类，否则不得分。
3. 对于成功生成的对抗样本，与原始样本进行对比，计算 ASS（SSIM），即平均结构相似性，参考论文 Image quality assessment: from error visibility to structural similarity。
4. 设置时间阈值，超出阈值作为惩罚项计分。
5. 最后计算 1000 张图片的平均得分作为最终得分。

提交要求：

1. 提交项目源代码
2. Python 库要求：只限于 python3.6 及以上，TensorFlow1.8 及以上，numpy，scipy，pandas，pillow，keras，cv2（只允许使用这些库，不允许使用其他开源的库）
3. 项目源代码打包成 AITest.zip，包含一个 python 程序
4. 程序主要包含一个 main.py 文件（但不限于只有一个文件，可互相调用），里面包含 aiTest 方法（**一定要在 main.py 中**）。
   后台测试时**只调用 aiTest 方法**，该方法的输入参数（及其顺序）和输出参数（及其顺序）如下：

```
def aiTest(images,shape)
  return generate_images
```

4.1 输入：

images: 一批图片，类型为：numpy.ndarray

shape: 该批图片的 shape，类型为：tuple

（例如：（1000，28，28，1），1000 表示输入图片的数量，28*28*1 表示单张图片的 shape）

4.2 输出：

Generate_images:同输入图片相同的 shape 的修改后的批量图片数据（同 images 一一对应，例如 images 中的第一张图片修改后的数据，即位于 generate_images 中的第一张）

5. 提交项目要求：

5.1 zip 包命名规范：学号_姓名.zip (如：191250111_张三.zip)

5.2 zip 包内包含一个 attack 文件夹(说明文档可选，非必须)

5.3 attack 文件夹下面需要**直接包含 main.py 文件**（其他辅助代码文件，辅助数据文件的代码结构不做要求，确保 mian.py 能正确调用到即可）

5.4 以上要求必须遵守，否则后果自负。