

# Fake/Real Face Classification

Assignment 4.

Yonsei University

Computer Vision 2019 Spring

# 1. Introduction

- **Fake/Real Face Classification**

We are going to classify whether given input face image is real or fake.



**Fake**



**Real**



**Fake**

## 2. Dataset

---

1. We provide the dataset for training/testing your networks
2. Our dataset consists of three classes
  - **Real:** real face images
  - **Fake:** synthesized fake images. These are generated manually by human experts, using tools like Adobe Photoshop.
  - **gan:** generated fake images. These are fake face images generated by our deep network.

## 2. Dataset (cont'd)

---

1. There are 3 directories inside dataset.zip, each represents each class.
2. Overall directory structure of dataset.zip is as follows.
  - dataset/
    - real/
      - real images...
    - gan/
      - generated images by deep network...
    - fake/
      - generated images by humans...

## 2. Dataset

---

- If you want to use additional dataset for training, you are free to use.
- Please explain about your dataset in your report, if you used additional data.

# 3. Network

---

1. No limitation for any implementation details of your network.
2. You are free to decide any details of your implementation, including **framework**, **network architecture**, **loss function**, **optimizer** or etc.
  - For this assignment, you can use tensorflow, pytorch or any other deep learning framework.
  - Even if you want to use different languages, that's OK for this assignment.
  - Just make sure that details of your implementation is explained in your report

## 4. Team Project

---

1. Please note that this assignment is a **team project**.
2. Team members can be one ~ three.

## 5. Submission: Output text

---

1. Output text file should be a text file which contains result of your model, on testing images.
2. Each line of your output text file should be a result of each input image, and format of the line should like below. Make sure that new line is **unix style** (`\n`).

**image\_name, probability\_of\_real**

3. For probability\_of\_real, it should be rounded up to .7f
4. Example of output text file:

```
000001.jpg, 0.0001111  
000002.jpg, 0.9982222  
000003.jpg, 0.6321111  
000004.jpg, 0.7112222
```



## 5. Submission: Output text (cont'd)

---

1. For this assignment, there will be a competition for some portion of grading.
2. In grading server, we will provide you leaderboard, where you can check rank of your network.
3. At the final phase, you will be graded based on your rank. (for some portion of your score)
4. Details of competition, testing images, and grading server will be announced soon.

# 5. AUROC

---

1. For measuring performance of your model, we will use **AUROC**.
2. AUROC is **Area Under ROC** curve.
3. For calculating AUROC, we will draw ROC curve using your output text file, and then calculate area under ROC curve.
4. We will provide code for calculating AUROC using output text file, and this will be announced soon.
5. For more information about AUROC, please see the link below.

<http://gim.unmc.edu/dxtests/roc3.htm>

## 5. Submission: Report

---

1. In report, you should explain details of your implementation.
2. Below contents **should** be included in your report.
  - Details of your network architecture
  - Basic information of your code  
(language you use, framework, OS or system you run your codes...)
  - Training details  
(training duration, optimizer, loss, learning rate, and any details of your hyper parameters)
  - Analysis of your network  
(sample classification result of your network, for which case your model fails to classify, and expected reasons of failure cases, or success cases.)

## 5. Submission (cont'd)

---

- These are the things you should submit as a submission.

### **1. Output text file (on grading server)**

→ details of grading server, and testing images will be announced soon.

### **2. Report, your project codes (on YSCEC)**

**Ex)**

**2019123456\_assignment4.zip**

└ **report.pdf**

└ **directory that has your project**

# 6. Grading Policy

---

## 1. Performance measurement (on grading server, using your output text file.) – 50 pt

- your AUROC score exceeds our baseline (30 pt)
- you got high rank in a competition (20 pt)

## 2. Report, codes (on YSCEC) – 50 pt

- Basic information of your code (5 pt)
- Details of your network architecture (15 pt)
- Training details (15 pt)
- Analysis of your network (15 pt)

## 7. Due Date

---

1. ~ 6/7 23:55
2. For late submission, -10% per each day.