

---

---

# **ADL x MLDS 2017 Fall**

## **HW4 - Generative Adversarial Networks**

 [adlxmls@gmail.com](mailto:adlxmls@gmail.com) 

---

---

# Updates

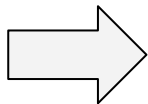
1. (12/11 17:43): p24, Toolkit version and script usage.
2. (12/14 15:39): p16, About Batch Normalization or not.
3. (12/14 15:49): p20, Ls and Lc mean “Log-likelihood”.
4. ~~(12/16 14:46): p26, Deadline should be at **12/31/2017**(Sun.)~~
5. (12/17 18:30): p23, Testing Text Content
6. (12/24 02:33): p17, Formula bug fixed (expectation)
7. (12/24 02:33): p17, Formula bug fixed (minus sign)
8. (1/3 13:55): p26-28, Submission Format

# Outline

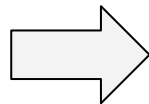
- Task Introduction
  - Text2image generation
  - Dataset collection
- Model
  - Conditional GAN
  - Tips for training
    - Discriminator loss function
    - Objective function
- Submission and grading

# Task Introduction - text2image generation

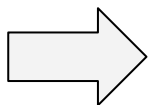
**an all black bird**



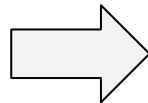
**Bird  
Generative Model**



**this flower is  
white and pink**



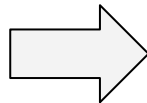
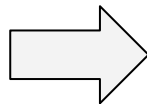
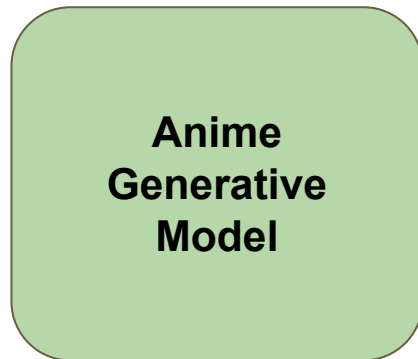
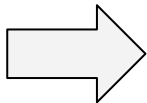
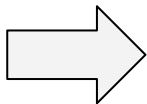
**Flower  
Generative Model**



# Task Introduction - text2image generation

**a man with no hair  
and one red strong fist**

**a girl with blue hair,  
blue eyes and  
twin ponytail**

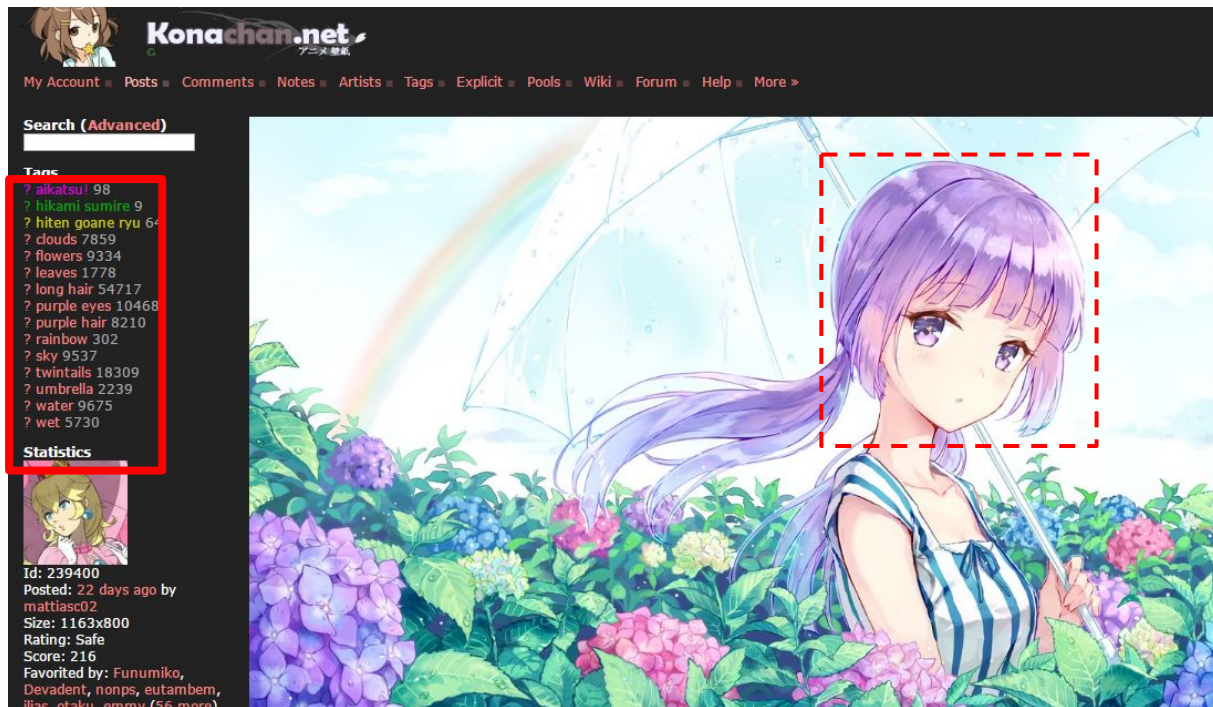


# Data Collection

## Tags

? aikatsu! 98  
? hikami sumire 9  
? hiten goane ryu 64  
? clouds 7859  
? flowers 9334  
? leaves 1778  
? long hair 54717  
? purple eyes 10468  
? purple hair 8210  
? rainbow 302  
? sky 9537  
? twintails 18309  
? umbrella 2239  
? water 9675  
? wet 5730

Not all tags are useful



Konachan.net  
アニメ看板

My Account » Posts » Comments » Notes » Artists » Tags » Explicit » Pools » Wiki » Forum » Help » More »

Search (Advanced)

Tags

- ? aikatsu! 98
- ? hikami sumire 9
- ? hiten goane ryu 64
- ? clouds 7859
- ? flowers 9334
- ? leaves 1778
- ? long hair 54717
- ? purple eyes 10468
- ? purple hair 8210
- ? rainbow 302
- ? sky 9537
- ? twintails 18309
- ? umbrella 2239
- ? water 9675
- ? wet 5730

Statistics

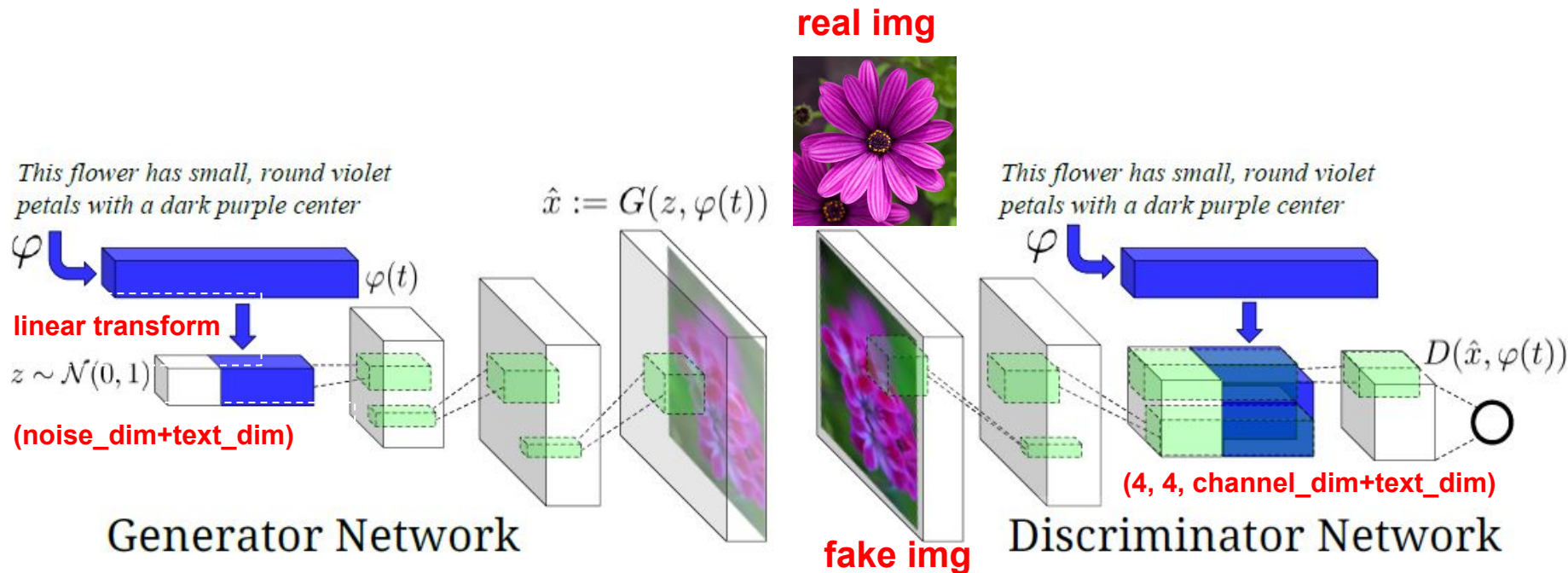
Id: 239400  
Posted: 22 days ago by mattias02  
Size: 1163x800  
Rating: Safe  
Score: 216  
Favorited by: Funumiko, Devadent, nonps, eutambem, illas, otaku, emmy (56 more)

[http://konachan.net/post/show/239400/aikatsu-clouds-flowers-hikami\\_sumire-hiten\\_goane\\_r](http://konachan.net/post/show/239400/aikatsu-clouds-flowers-hikami_sumire-hiten_goane_r)

感謝樊恩宇助教蒐集data

# Model and training tips

# Conditional GAN for text2image generation

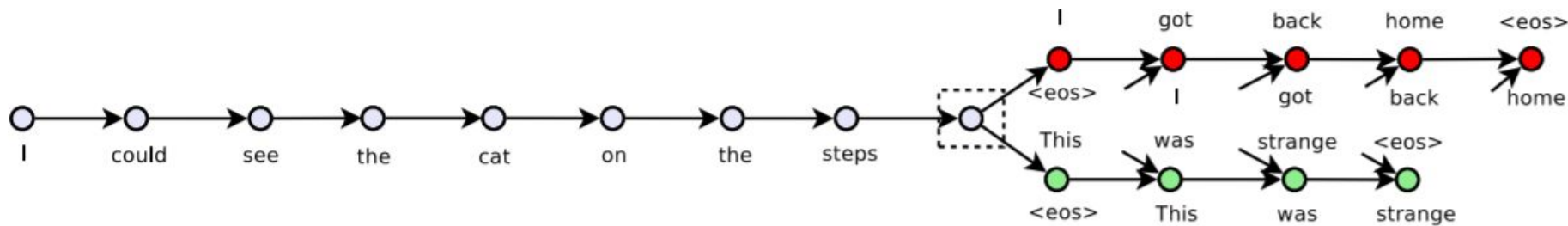




# Details for training

- Updates between Generator and Discriminator
  - 1 : 1 or 2 : 1
- ADAM with  $lr = 0.0002$ , momentum = 0.5
- gaussian or uniform noise dim = 100
- batch size = 64
- epoch = 300

# Text feature process tool - Skip-thought vector



skip-thought source code:

[https://github.com/tensorflow/models/tree/master/research/skip\\_thoughts#download-pretrained-models-optional](https://github.com/tensorflow/models/tree/master/research/skip_thoughts#download-pretrained-models-optional)

No matter which tool you use to process text input, please make sure you include that pre-trained model in your repository to let us run your code successfully.

# Image process tool - skimage and scipy.misc

```
In [35]: # convert img to tensor
In [36]: import skimage
In [37]: import skimage.io
In [38]: img = skimage.io.imread('sample.jpg')
In [39]: # resize img
In [40]: import skimage.transform
In [41]: img_resized = skimage.transform.resize(img, (64, 64))
In [42]: img.shape
Out[42]: (96, 96, 3)
In [43]: img_resized.shape
Out[43]: (64, 64, 3)
```

Install:

- `sudo apt-get install python-skimage`
- `sudo pip install --user numpy scipy`

```
In [48]: # convert tensor to img
In [49]: import scipy.misc
In [50]: scipy.misc.imsave('sample_resize.jpg', img_resized)
In [51]: ls sam
sample.jpg          sample_resize.jpg
In [51]: ls sample
```

# Little Demo

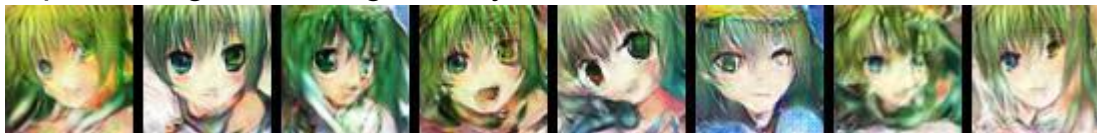
input text: black hair blue eyes



input text: pink hair green eyes



input text: green hair green eyes



input text: blue hair red eyes



# Tips for training

- Discriminator output:
  - (real img, right text): 1
  - (fake img, right text): 0
  - (real img, wrong text): 0
  - (wrong img, right text): 0
- Different objective function
  - Wasserstein GAN (WGAN)
  - Improved W-GAN
  - Auxiliary Classifier GAN (ACGAN)

# Wasserstein GAN

The output of D is thus not probability anymore.  
The D loss turn to be a measure of distance.

$$L_D^{WGAN} = E[D(x)] - E[D(G(z))]$$

$$L_G^{WGAN} = E[D(G(z))]$$

$$W_D \leftarrow \text{clip\_by\_value}(W_D, -0.01, 0.01)$$

# Wasserstein GAN

- In each training iteration: **No sigmoid for the output of D**

Learning  
D

Repeat  
k times

- Sample  $m$  examples  $\{x^1, x^2, \dots, x^m\}$  from data distribution  $P_{data}(x)$
- Sample  $m$  noise samples  $\{z^1, z^2, \dots, z^m\}$  from the prior  $P_{prior}(z)$
- Obtaining generated data  $\{\tilde{x}^1, \tilde{x}^2, \dots, \tilde{x}^m\}$ ,  $\tilde{x}^i = G(z^i)$
- Update discriminator parameters  $\theta_d$  to maximize
  - $\tilde{V} = \frac{1}{m} \sum_{i=1}^m D(x^i) - \frac{1}{m} \sum_{i=1}^m D(\tilde{x}^i)$
  - $\theta_d \leftarrow \theta_d + \eta \nabla \tilde{V}(\theta_d)$  **Weight clipping**

Learning  
G

Only  
Once

- Sample another  $m$  noise samples  $\{z^1, z^2, \dots, z^m\}$  from the prior  $P_{prior}(z)$
- Update generator parameters  $\theta_g$  to minimize
  - $\tilde{V} = \frac{1}{m} \sum_{i=1}^m \log D(x^i) - \frac{1}{m} \sum_{i=1}^m D(G(z^i))$
  - $\theta_g \leftarrow \theta_g - \eta \nabla \tilde{V}(\theta_g)$

# Wasserstein GAN

## Implementation Notes :

- Do not apply sigmoid at the output of D
- Clip the weight of D
- Use RMSProp instead of Adam
- Train more iteration of D (the paper use 5)
- ~~• Do not use batch normalization~~
- Do not apply batch normalization on **Discriminator** only when using **Improved W-GAN loss (WGAN-GP loss)**  
(ref: <https://arxiv.org/pdf/1704.00028.pdf>, page.4)

ref:<https://arxiv.org/pdf/1701.07875.pdf>



# Improved W-GAN

Do not clip the weight of D but to add a new objective called “Gradient Penalty”.

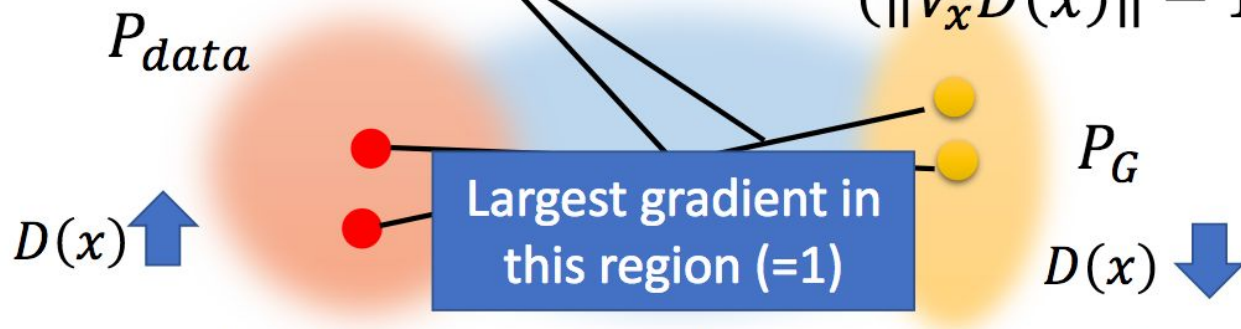
$$L_D^{WGAN\_GP} = L_D^{WGAN} - \lambda E[(\|\nabla D(\alpha x + (1 - \alpha)G(z))\|_2 - 1)^2]$$

$$L_G^{WGAN\_GP} = L_G^{WGAN}$$

# Improved W-GAN

$$W(P_{data}, P_G) \approx \max_D \{ E_{x \sim P_{data}} [D(x)] - E_{x \sim P_G} [D(x)] - \lambda E_{x \sim P_{penalty}} [\max(0, \|\nabla_x D(x)\| - 1)] \}$$

$$(\|\nabla_x D(x)\| - 1)^2$$



# ACGAN

Discriminator should also be able to do a classification task.

$$L_{D,Q}^{ACGAN} = L_D^{GAN} + E[P(class = c|x)] + E[P(class = c|G(z))]$$

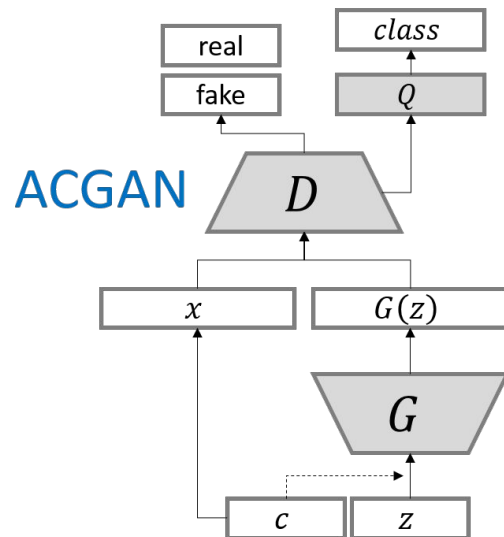
$$L_G^{ACGAN} = L_G^{GAN} + E[P(class = c|G(z))]$$

# ACGAN

$$L_S = E[\log P(S = \text{real} \mid X_{\text{real}})] + \\ E[\log P(S = \text{fake} \mid X_{\text{fake}})]$$

$$L_C = E[\log P(C = c \mid X_{\text{real}})] + \\ E[\log P(C = c \mid X_{\text{fake}})]$$

**D** is trained to maximize  $\mathbf{L}_s + \mathbf{L}_c$   
while **G** is trained to maximize  $\mathbf{L}_c -$   
 $\mathbf{L}_s$

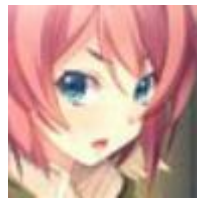


# Submission and Grading

# Homework 4 package

- Anime Dataset:

- training data: 33.4k (image, tags) pair
- faces/, tags.csv, sample\_testing.txt



blue eyes  
red hair  
short hair

- training tags file format

- img\_id <comma> tag1 <colon> #\_post <tab> tag2 <colon> ...

```
1 0,touhou:17705 |chen:423 |moneti daifuku :60 |animal ears:12241 |catgirl:4903 |
2 1,touhou:17697 |onozuka komachi:224 |shikieiki yamaxonadu:217 |$
3 2,original:25774 |blonde hair:25457 |doll:1040 |dress:16585 |pink eyes:3896 |ta
4 3,amagi brilliant park:111 |musaigen no phantom world:39 |nichijou:142 |kawakam
```

tags.csv

- testing text file format

- testing\_text\_id <comma> testing\_text

```
1 1,blue hair blue eyes
2 2,blue hair green eyes
3 3,blue hair red eyes
4 4,green hair blue eyes
```

sample  
testing\_text.txt

- testing text only includes **'color hair'** and **'color eyes'**, only alphabetic char involved.
- Data download link:
  - <https://drive.google.com/open?id=1bXXeEzARYWsvUwbW3SA0meulCR3nIhDb>
  - If you want to do something cool beyond generating faces, mail us. We will give you original images

# Testing Text Content

## **[color hair]:**

'orange hair', 'white hair', 'aqua hair', 'gray hair',  
'green hair', 'red hair', 'purple hair', 'pink hair',  
'blue hair', 'black hair', 'brown hair', 'blonde hair'.

## **[color eyes]:**

'gray eyes', 'black eyes', 'orange eyes',  
'pink eyes', 'yellow eyes', 'aqua eyes', 'purple eyes',  
'green eyes', 'brown eyes', 'red eyes', 'blue eyes'.

# Allowed packages

- Allowed package includes:
  - PyTorch v0.2.0
  - tensorflow r1.3
  - Keras 2.0.7
  - MXNet 0.11.0
  - CNTK 2.2
  - python 2.7/3.5/3.6
    - In .sh file, please call “python2” or “python3.5” or “python3.6”
    - Ex. python2 GAN\_train.py



# Submission on Github

- Only one branch **master** is needed
- **master** stores the model by **using GAN structure**
- Remember to put your **pre-trained models or download scripts** so that we can run your code successfully

# Submission

- Deadline: ~~12/31/2017(Sun.) 23:59:59 (UTC+8)~~
- New Deadline: 1/7/2018(Sun.) 23:59:59 (UTC+8)
- **ADLxMLDS2017/hw4** should contain the following files:
  - **run.sh, extra\_run.sh(optional)**
  - **train.py, generate.py**
  - **(pre-)trained\_model, samples/**
  - **report.pdf**
  - If some files are too big, upload to your cloud and download them when running your run.sh
  - You can merge train.py and generate.py to a single file
- TAs will run your scripts in the following order to generate images
  - bash run.sh [testing\_text.txt]
  - bash extra\_run.sh [testing\_text.txt] (optional)
  - both scripts must output in **10 minutes**.

# Output Format Requirement

- The generated images should be in Directory **samples/**
  - ~~make sure it's **empty** before we run your code~~
  - 為加速批改, 請大家繳交時 **就將產生的結果傳到 samples/**, 助教利用 script reproduce 時也請同學將結果輸出到這個資料夾。為保證 reproduce 結果相同, 請同學將 random 的部份固定
  - Please put your **generated images into samples/** to speed up the peer review process. Also, please fix your random seed so that TAs can reproduce your images.
- For each condition, you must generate **5 images**
- [testing\_text.txt] contains the following three conditions:
  - 1, blue hair red eyes
  - 2, black hair blue eyes
  - 3, red hair green eyes
- Each generated image must be resized to **64 x 64**

# Output Format Requirement

- Generated img should be named as
  - **"sample\_(testing\_text\_id)\_(sample\_id).jpg"**
  - **"extra\_sample\_(testing\_text\_id)\_(sample\_id).jpg"** (optional)
- Example:

```
extra_sample_1_1.jpg extra_sample_2_2.jpg extra_sample_3_3.jpg sample_1_4.jpg sample_2_5.jpg
extra_sample_1_2.jpg extra_sample_2_3.jpg extra_sample_3_4.jpg sample_1_5.jpg sample_3_1.jpg
extra_sample_1_3.jpg extra_sample_2_4.jpg extra_sample_3_5.jpg sample_2_1.jpg sample_3_2.jpg
extra_sample_1_4.jpg extra_sample_2_5.jpg sample_1_1.jpg sample_2_2.jpg sample_3_3.jpg
extra_sample_1_5.jpg extra_sample_3_1.jpg sample_1_2.jpg sample_2_3.jpg sample_3_4.jpg
extra_sample_2_1.jpg extra_sample_3_2.jpg sample_1_3.jpg sample_2_4.jpg sample_3_5.jpg
```

# 組別互評

- We will put your generated images in the grading platform
- Link will be sent to your mail after HW deadline
- Answer **2** scores for each image
  - How the image fits the text
  - How the image looks real
- Scores should be integer from 1 to 5
  - 1 to 5 corresponding to (super bad, bad, average, good, super good)
- You may score your results, so be fair when your are scoring :)

# 組別互評

- Separate scores with a comma (score for matching text, score for reality)
- Example:

3 → Gray hair green eyes



4, 5

Ok ✓

press ENTER

# What report should cover?

- Model description(2%)
  - Must include model structure, objective function for G and D
- How do you improve your performance (2%)
- Experiment settings and observation (2%)
- No more than 5 pages
- Please written in Chinese (unless you don't know how to type Chinese)

# Grading Policy (18%)

- Wrong output format will not be graded
- Report (6%)
- 限時任務(2%) [To be announced]
- Score others' generated images(2%)
- Peer feedback(4%)
- Code (4%)
  - You will be scored only if you use GAN and output results in 10 minutes
  - Fix random seed so we can reproduce your results
- Bonus
  - style-transfer (2%)



# Special Mission

- Open a folder ADLxMLDS2017/hw4/early/
- The condition is: red hair, green eyes
- Please generate 5 images corresponding to this condition
- Period: 2017/12/18~2017/12/24(23:59)

# Other Policy

- Late policy: 25% off per day late afterwards.  
[Delay form will be announced afterwards]
- No plagiarism is allowed.

# TA hours

- If you have other questions,
  - please contact TAs via [adlxmlds@gmail.com](mailto:adlxmlds@gmail.com)
  - post your questions on [facebook group](#)
  - go to TA office hours
    - 季大中 Mon 17:30-19:00 (徳田524)
    - 葉政杰 Fri 15:30-17:00 (電二531)
    - 楊靖平 Thu 15:30-17:00 (電二531)

# Q&A

1. samples/ 裡可以有 .gitignore 或是 .gitkeep 之類的檔案嗎？有什麼方式可以讓 git track empty dir?  
Ref:<https://stackoverflow.com/questions/115983/how-can-i-add-an-empty-directory-to-a-git-repository>
2. scipy version: 1.0.0
3. 請問 "#\_post" 代表的意思是？會需要用到嗎？---> "img\_id <comma> tag1 <colon> #\_post <tab> tag2 <colon> ... 答：不會用到
4. 關於testing text的格式請參照 p22, 兩個敘述間是 **沒有逗點**的, 內容只會從p23有的出, **不保證髮色眼色會同時出現, 也沒有保證兩個敘述的順序**, 但至少會有一個敘述, 且有多個敘述時不會互相衝突。
5. 可以使用torchvision, 版本為0.2.0
6. train.py generate.py可以合併成一個檔案, 只要 script確定能處理好就好
7. 限時任務跟真的testing時都是一組敘述產生**5**張圖片
8. 請問我們可以自己將 DATASET中沒標到hair和eyes的資料標好嗎？或是將有多種顏色的 hair和 eyes重新自己標嗎？答：**可以, 另外對data做一些augmentation也是可以的**