MLDS HW3-2

TAs
b05902013@csie.ntu.edu.tw
b05902127@csie.ntu.edu.tw

Baseline Model (5/15 Update)

- Generator
  - noise_input = (100,);
  - text_input = (119,);
  - # num of (hair, eyes) pairs
  - text_emb = Dense(256,‘relu’)(text_input);
  - concatenate([noise_input, text_emb]);
  - Dense(4*4*512); Reshape((4, 4, 512));
  - Batchnorm(mom=0.9); Relu;
  - Conv2DTranspose(256, kernel=5);
  - Batchnorm(mom=0.9); Relu;
  - Conv2DTranspose(128, kernel=5);
  - Batchnorm(mom=0.9); Relu;
  - Conv2DTranspose(64, kernel=5);
  - Batchnorm(mom=0.9); Relu;
  - Conv2DTranspose(3, kernel=5);
  - Tanh;
- Training
  - Adam(lr = 0.0002, beta = 0.5)
- Discriminator
  - image_input = (64,64,3);
  - text_input = (119,);
  - text_emb = Dense(256,’relu’)(text_input);
  - text_emb = Reshape((1,1,256))(text_emb);
  - tiled_emb = tile(text_emb, [1,4,4,1]);
  - Conv2D(64 ,kernel=5)(image_input); LeakyRelu;
  - Conv2D(128, kernel=5);
  - Batchnorm(mom=0.9); LeakyRelu;
  - Conv2D(256, kernel=5);
  - Batchnorm(mom=0.9); LeakyReLu;
  - Conv2D(512, kernel=5);
  - Batchnorm(mom=0.9);
  - image_feat = LeakyRelu;
  - concatenate([image_feat, tiled_emb]);
  - Conv2D(512, kernel=1, strides=(1,1));
  - Flatten;
  - Dense(1, ‘sigmoid’);

Outline

- Timeline
- Task Descriptions
- Model & Training tips
- Submission & Rules
- Q&A

Timeline

Three Parts in HW3

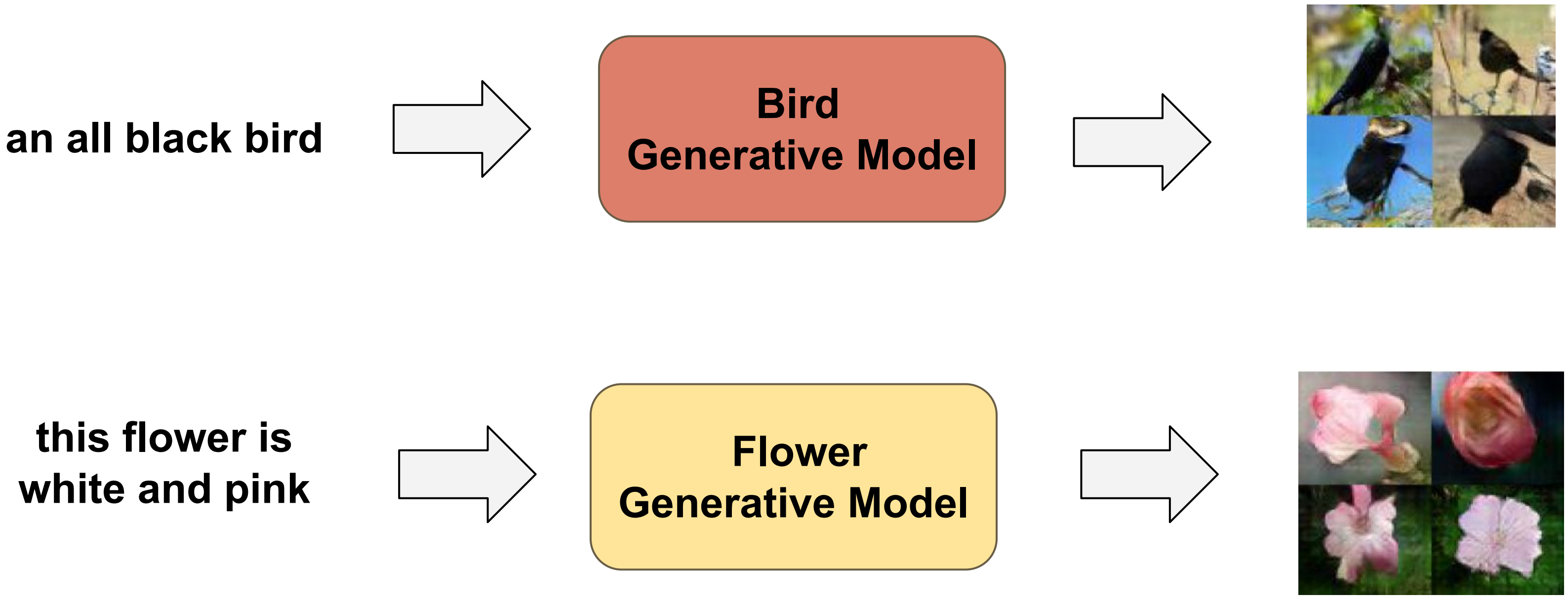
- (3-1) Image Generation
- (3-2) Text-to-Image Generation
- (3-3) Style Transfer

Schedule

- **4/30 or 5/4 :**
    - Release HW3-1
- **5/7 or 5/11 :**
    - Present HW 3-1
    - Release HW 3-2, HW 3-3
- **5/14 or 5/18 : Break**
- **5/21 or 5/25:**
    - Present HW 3-2, HW 3-3

Task Descriptions

HW3-2: Text-to-Image Generation 1/2



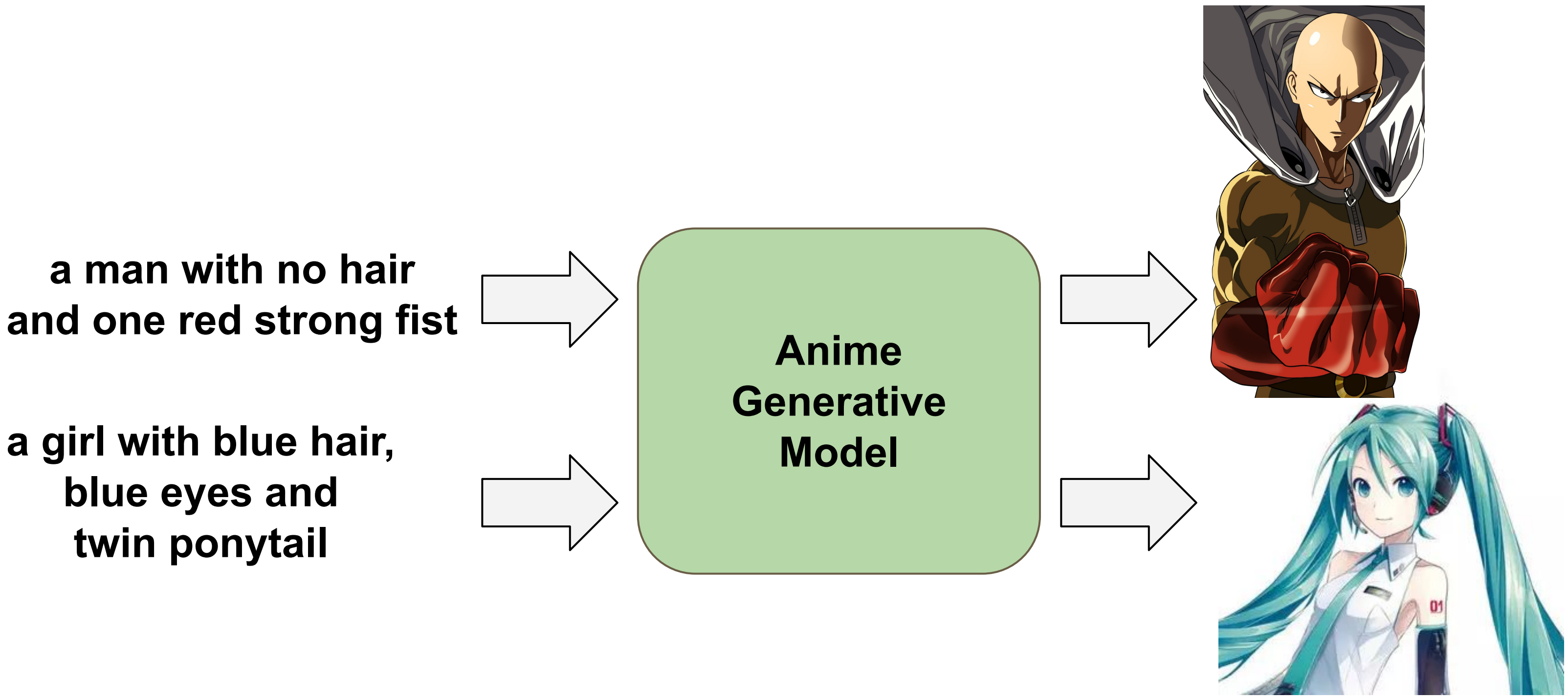
an all black bird

Bird Generative Model

this flower is white and pink

Flower Generative Model

HW3-2: Text-to-Image Generation 2/2



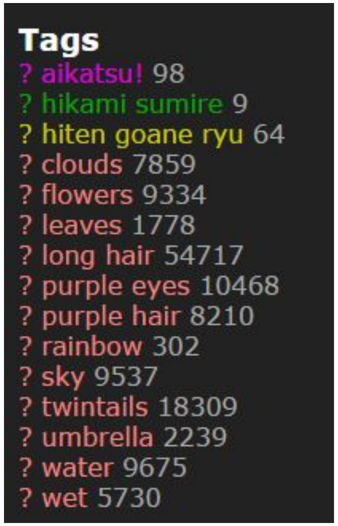
a man with no hair
and one red strong fist

a girl with blue hair,
blue eyes and
twin ponytail

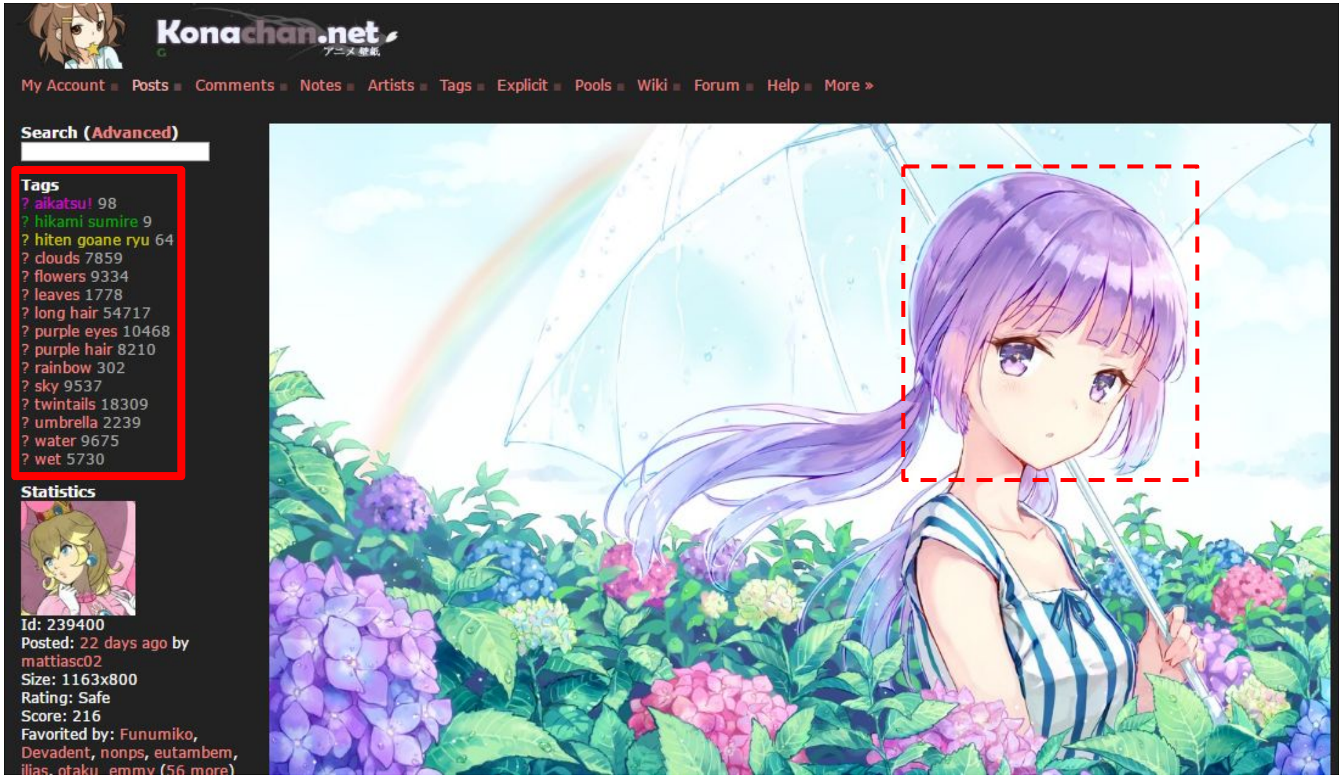
Anime
Generative
Model

Data Collections 1/2

- Anime dataset



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



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

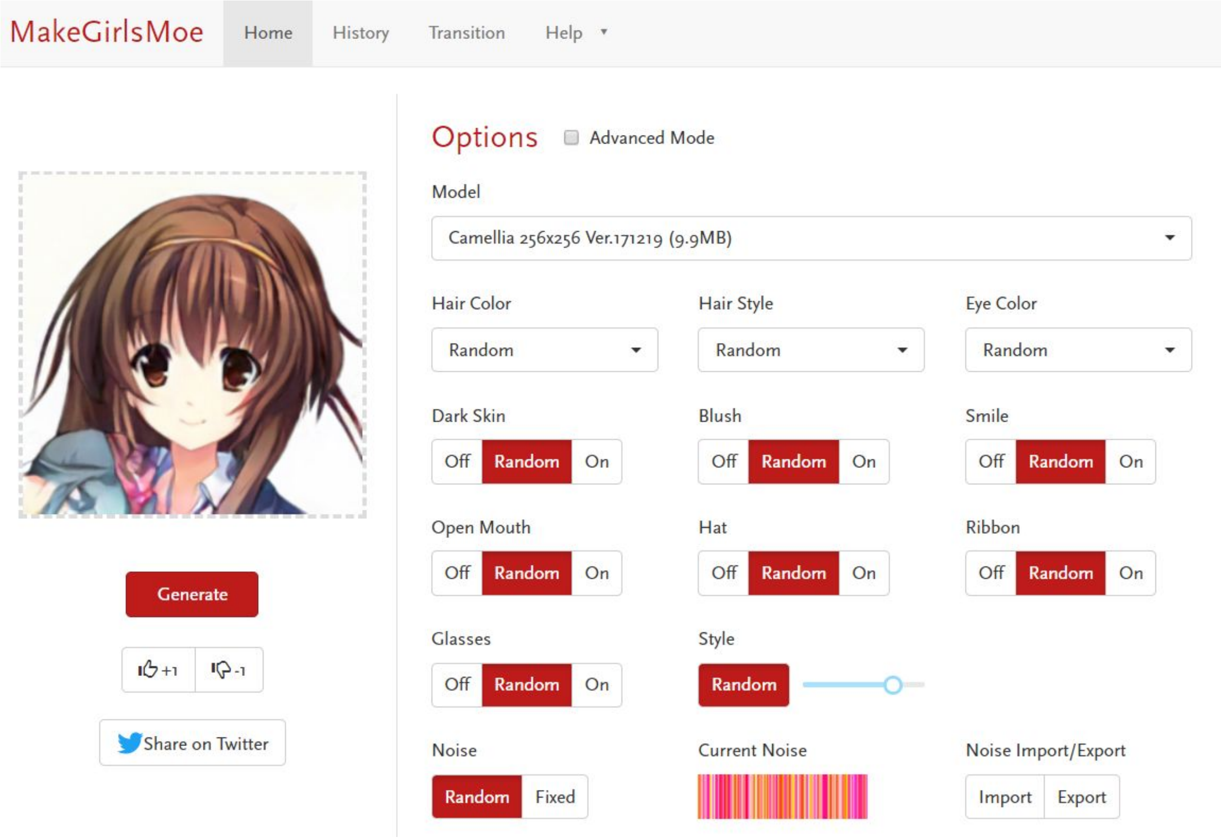
Statistics
Id: 239400
Posted: 22 days ago by mattiasc02
Size: 1163x800
Rating: Safe
Score: 216
Favorited by: Funumiko, Devadent, nonps, eutambem, ilias, otaku_emmy (56 more)

http://konachan.net/post/show/239400/aikatsu-clouds-flowers-hikami_sumire-hiten_goane_r

感謝樊恩宇助教蒐集data

Data Collections 2/2

- Extra data



Options □ Advanced Mode

Model: Camellia 256x256 Ver.171219 (9.9MB)

Hair Color: Random

Hair Style: Random

Eye Color: Random

Dark Skin: Off [✓ Random] On

Blush: Off [✓ Random] On

Smile: Off [✓ Random] On

Open Mouth: Off [✓ Random] On

Hat: Off [✓ Random] On

Ribbon: Off [✓ Random] On

Glasses: Off [✓ Random] On

Style: [✓ Random]

Noise: [✓ Random] Fixed

Current Noise

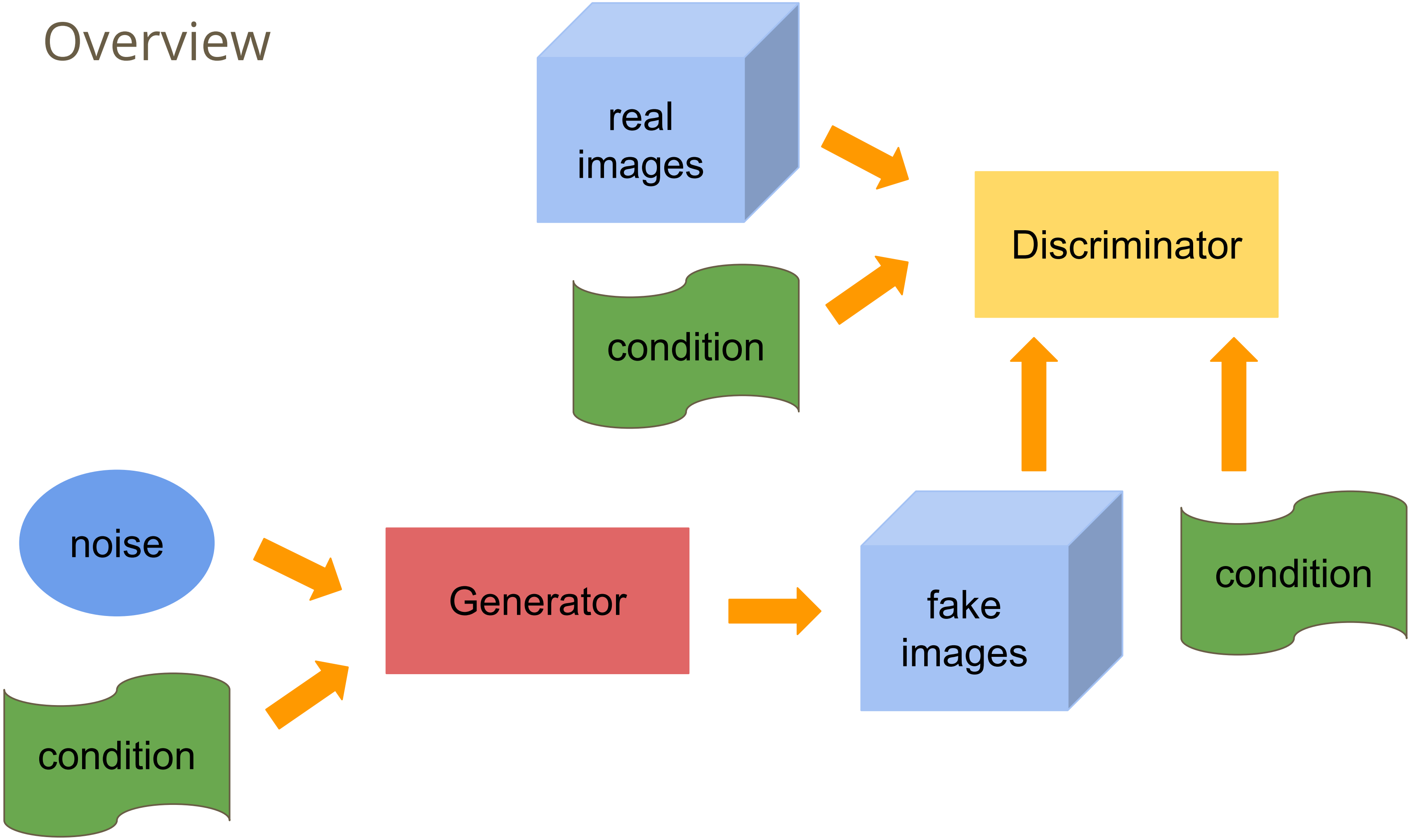
Noise Import/Export: Import Export

https://make.girls.moe/#/　　感謝吳宗翰同學收集並處理資料

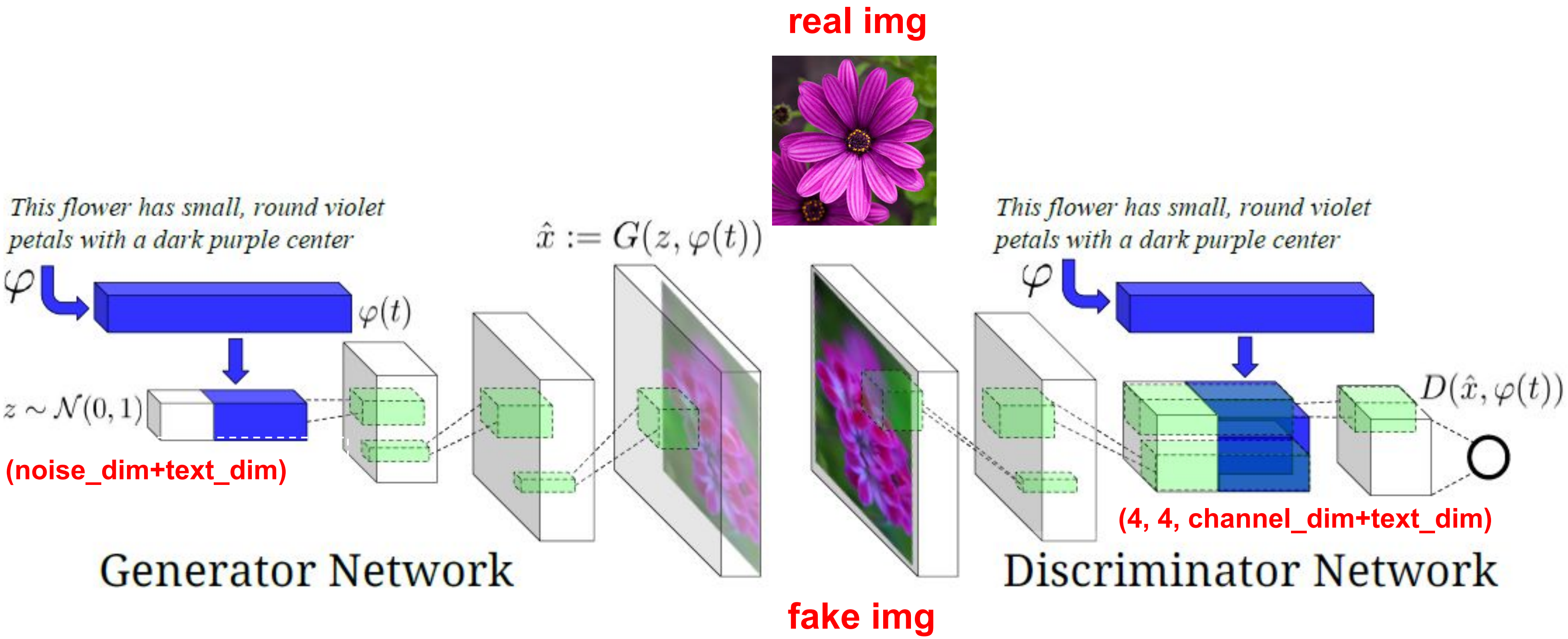
Model & Training Tips

Conditional GAN

- Overview

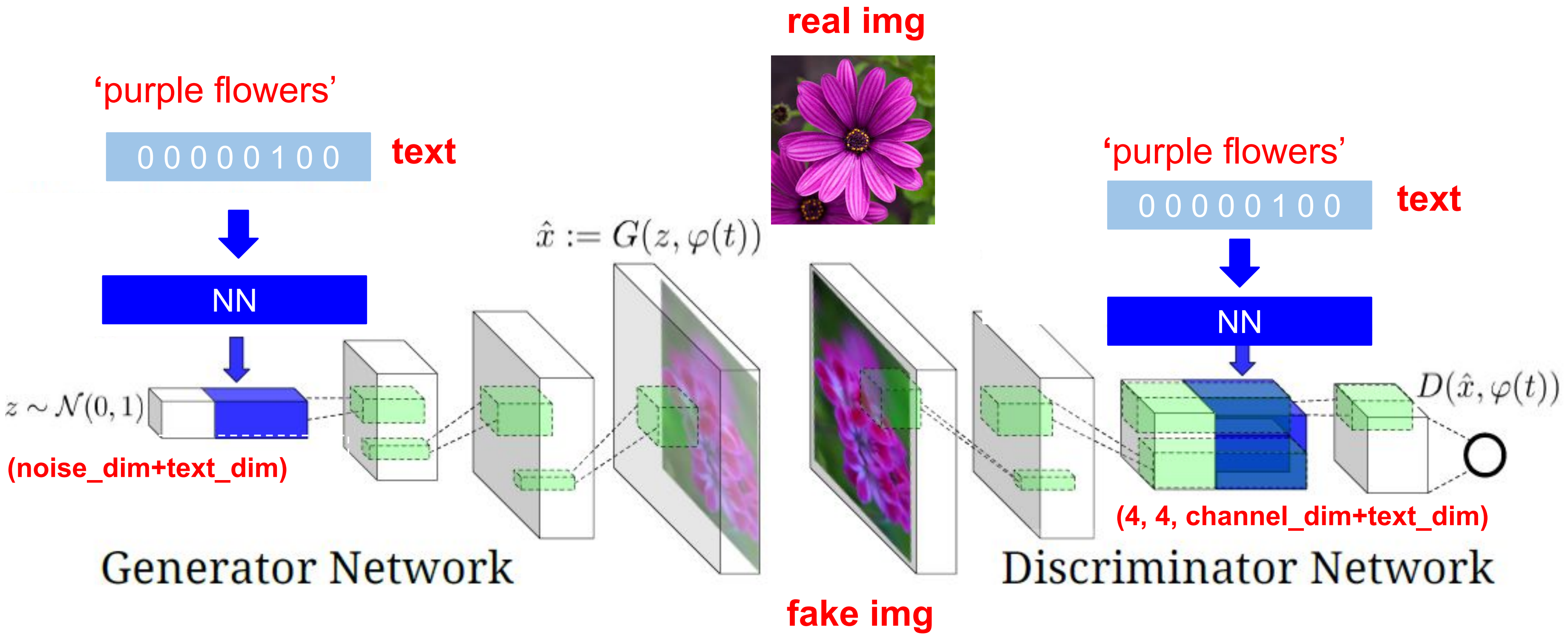


Conditional GAN for Text-to-Image Generation 1/2


real img
This flower has small, round violet petals with a dark purple center
φ
$\varphi(t)$
$z \sim \mathcal{N}(0,1)$
(noise_dim+text_dim)
$\hat{x} := G(z, \varphi(t))$
Generator Network
fake img
This flower has small, round violet petals with a dark purple center
φ
$D(\hat{x}, \varphi(t))$
(4, 4, channel_dim+text_dim)
Discriminator Network

ref: https://arxiv.org/pdf/1605.05396.pdf

Conditional GAN for Text-to-Image Generation 2/2



ref: https://arxiv.org/pdf/1605.05396.pdf

Tip for Training

- Discriminator Output:
  - (real image, right text): 1
  - (fake image, right text): 0
  - (wrong image, right text): 0

- Different objective function
  - Wasserstein GAN (WGAN)
  - Improved WGAN (WGAN-GP)
  - Auxiliary Classifier GAN (ACGAN)
  - StackGAN

ACGAN 1/2

Discriminator should also be able to do a classification task.

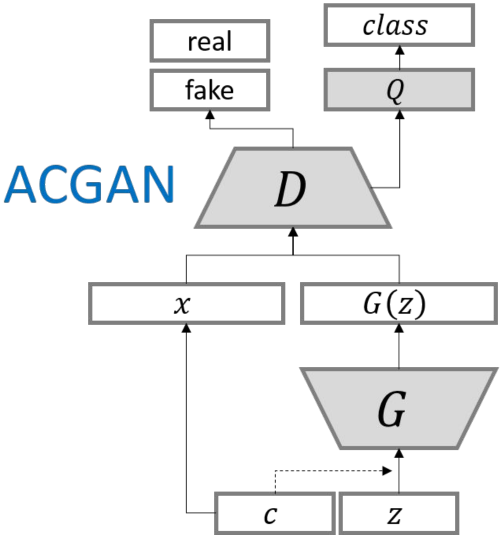
$$L_{D,Q}^{ACGAN} = L_D^{GAN} + E[P(class = c|x)] + E[P(class = c|G(z))]$$

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

ACGAN 2/2

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

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



D is trained to maximize $\mathbf{L_s + L_c}$ while **G** is trained to maximize $\mathbf{L_c - L_s}$

ref:https://arxiv.org/pdf/1610.09585.pdf

StackGAN

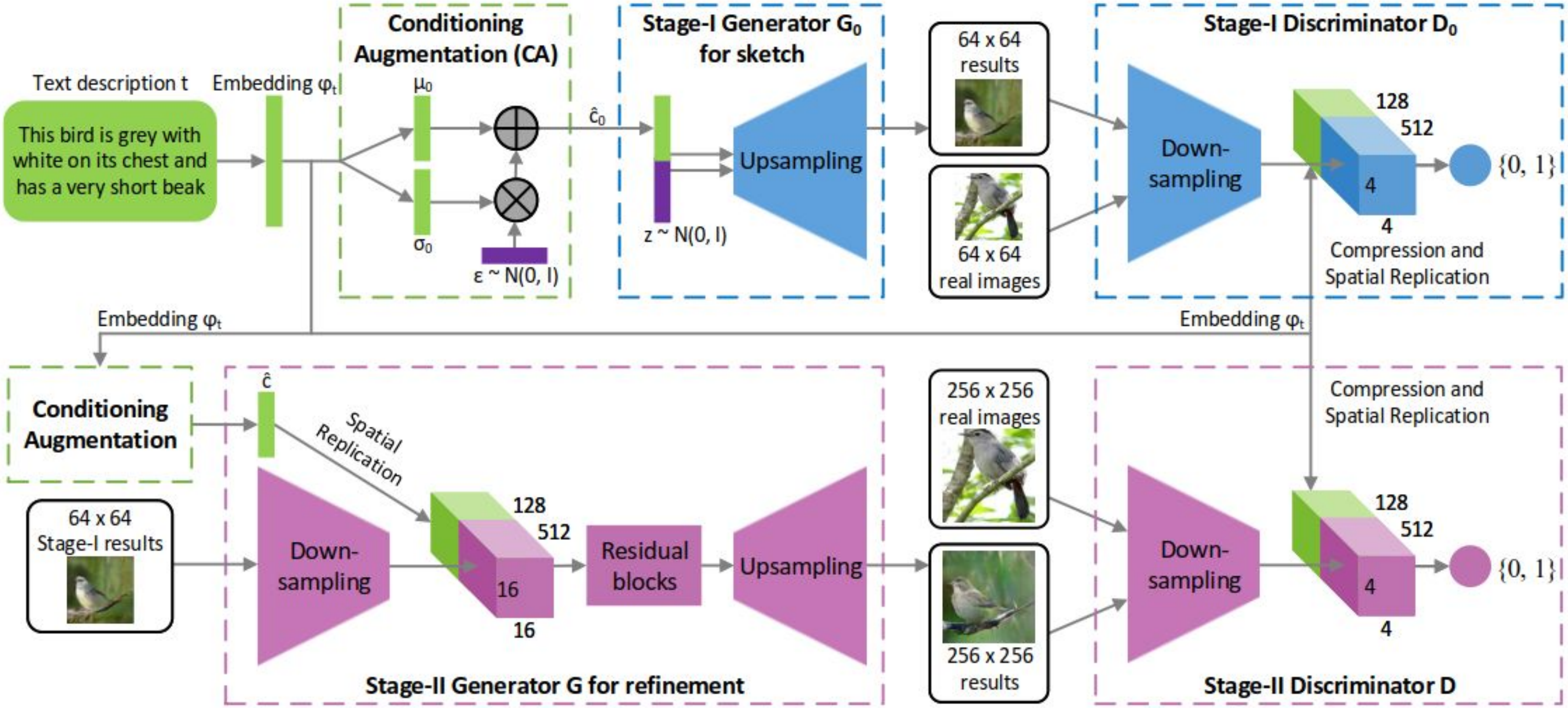
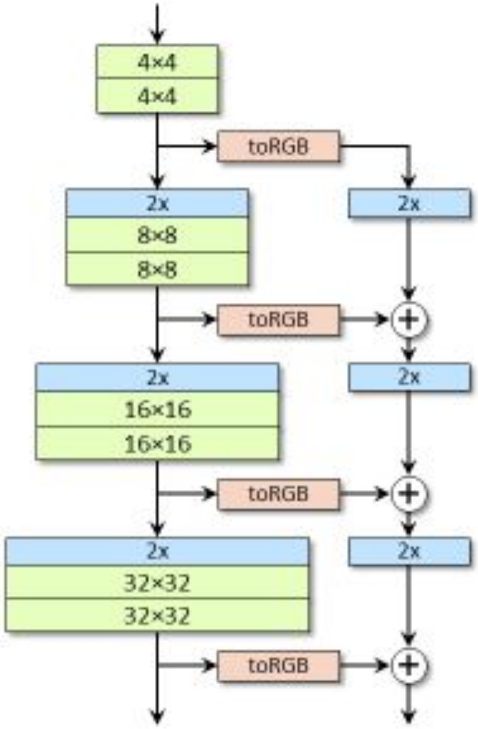


Figure 2. The architecture of the proposed StackGAN. The Stage-I generator draws a low-resolution image by sketching rough shape and basic colors of the object from the given text and painting the background from a random noise vector. Conditioned on Stage-I results, the Stage-II generator corrects defects and adds compelling details into Stage-I results, yielding a more realistic high-resolution image.

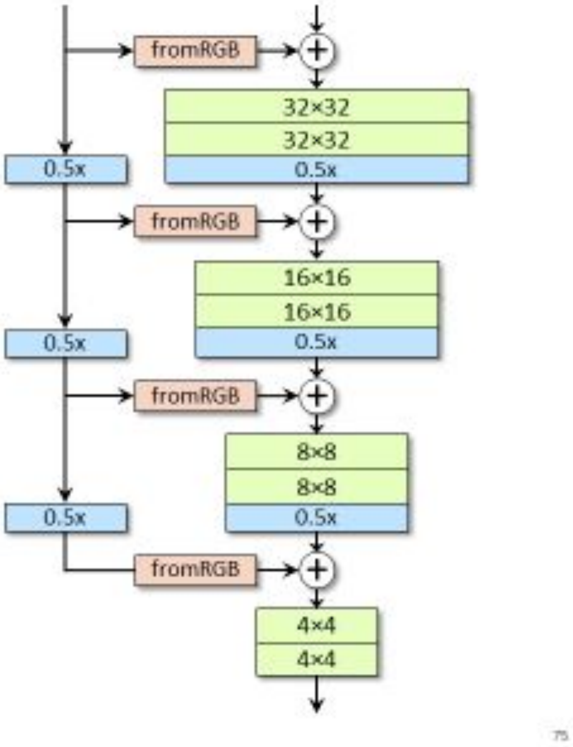
ref:https://arxiv.org/pdf/1612.03242.pdf

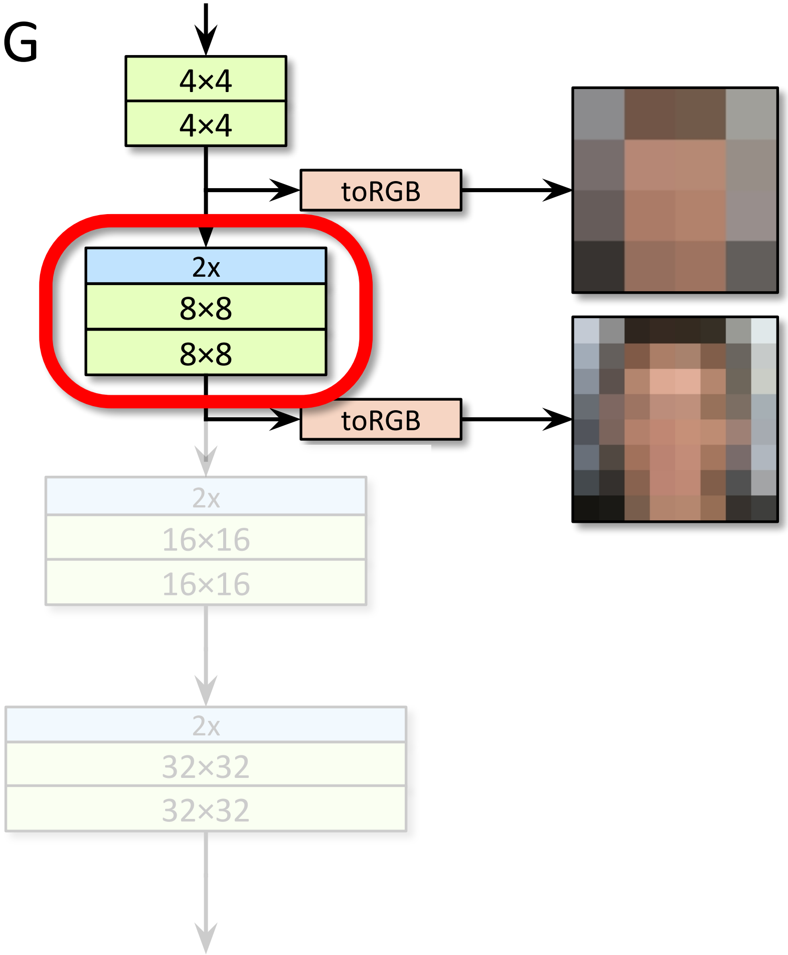
Progressive GAN

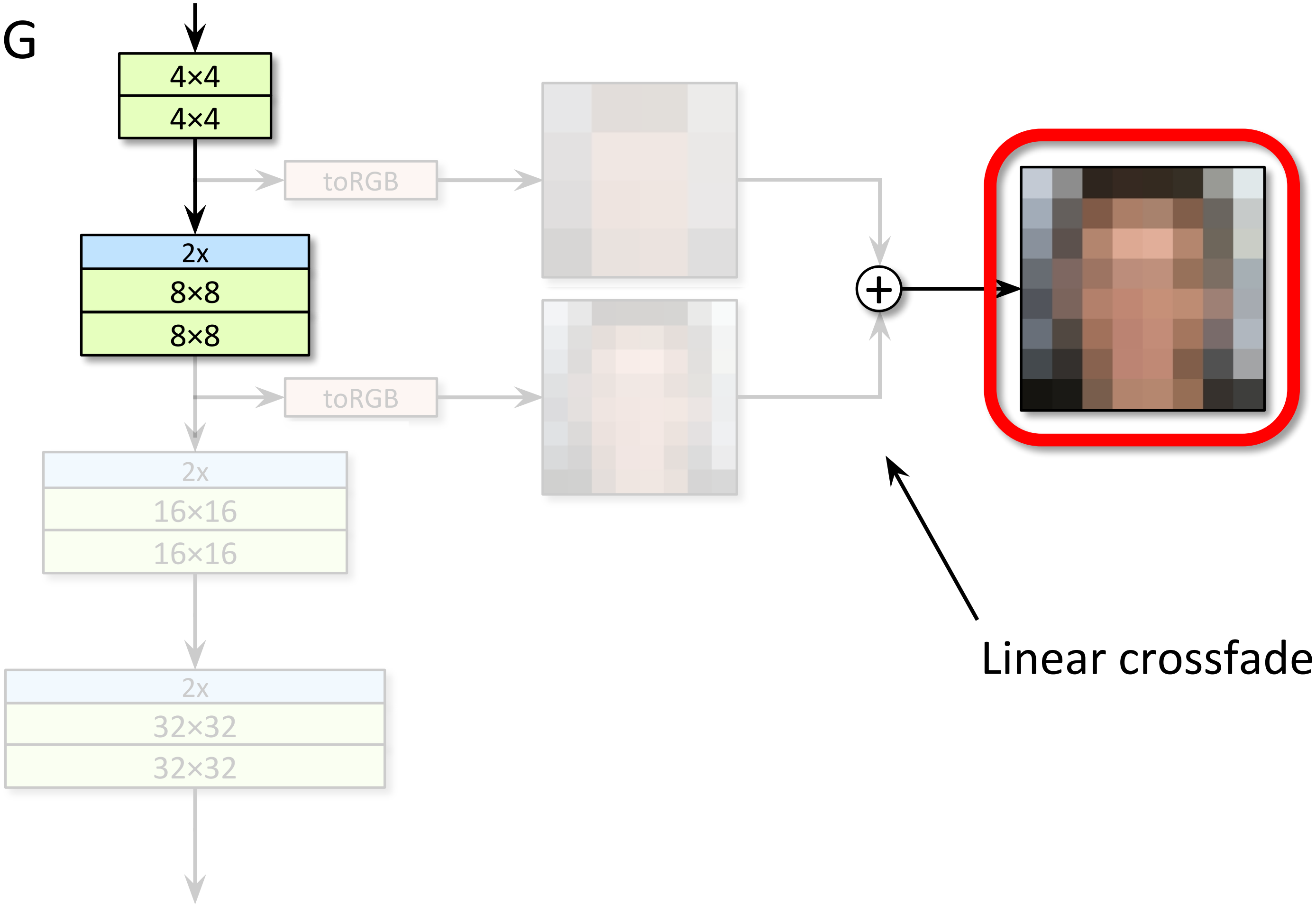
Generator



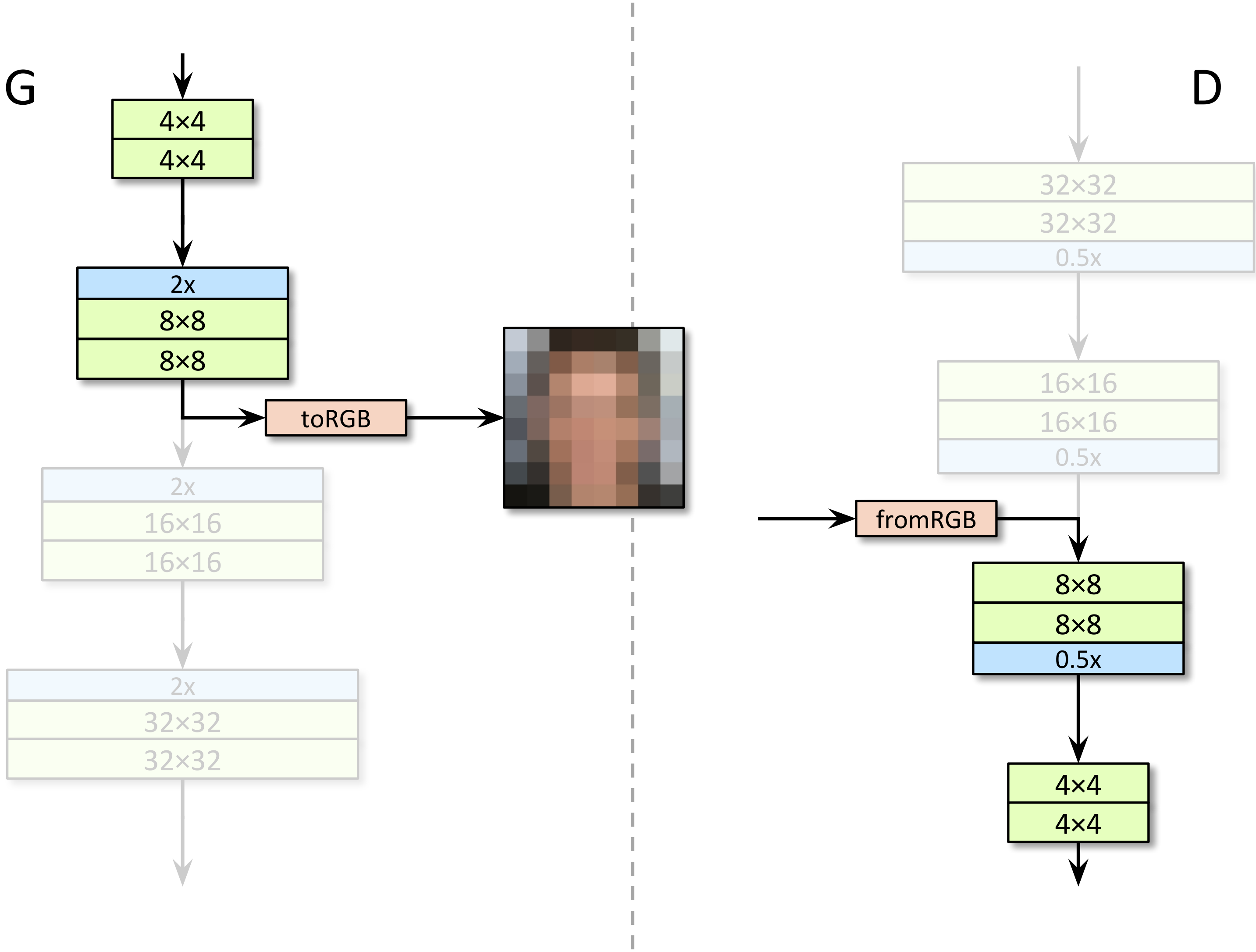
Discriminator




G
4×4
4×4
toRGB
2x
8×8
8×8
toRGB
2x
16×16
16×16
2x
32×32
32×32



Linear crossfade



G

4×4
4×4

2x
8×8
8×8

toRGB

2x
16×16
16×16

2x
32×32
32×32

D

32×32
32×32
0.5x

16×16
16×16
0.5x

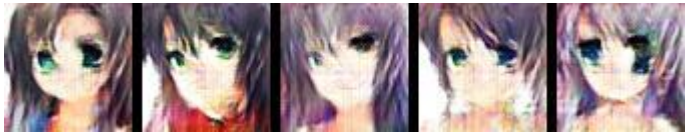
fromRGB

8×8
8×8
0.5x

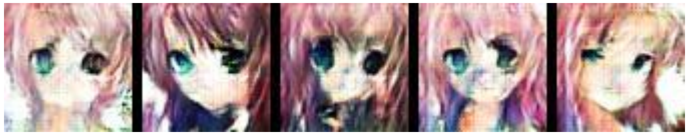
4×4
4×4

Little Results

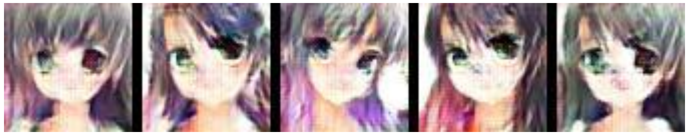
input text: black hair blue eyes



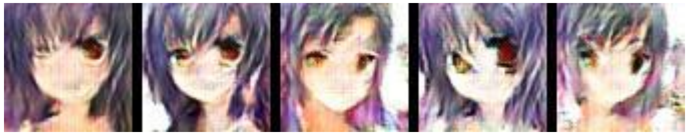
input text: pink hair green eyes



input text: green hair green eyes

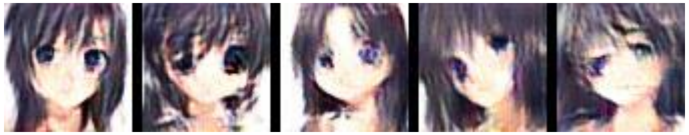


input text: blue hair red eyes

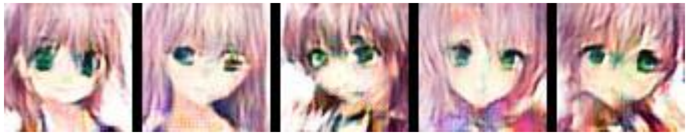


GAN result

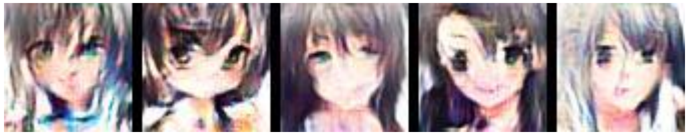
input text: black hair blue eyes



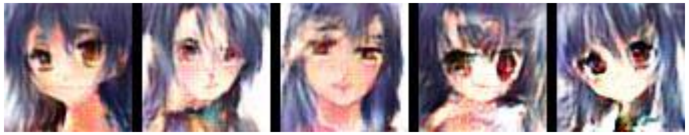
input text: pink hair green eyes



input text: green hair green eyes

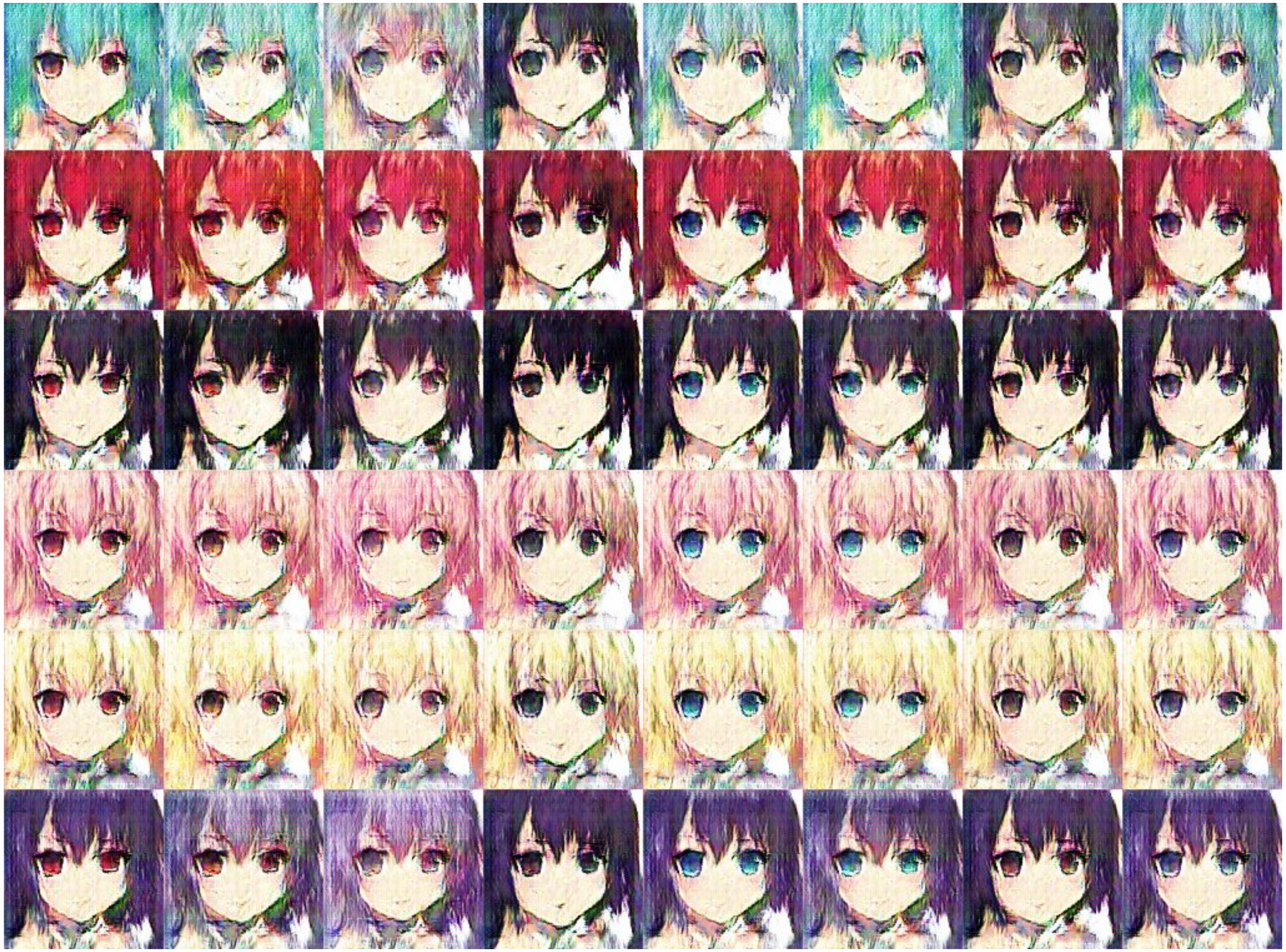


input text: blue hair red eyes



WGAN-GP result

Little Results

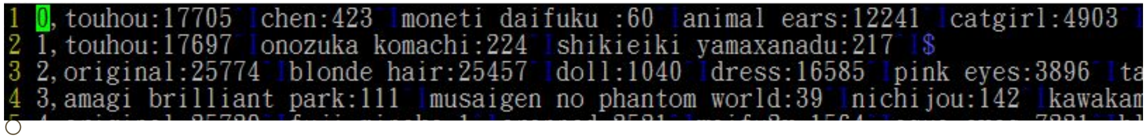


Submission & Grading

Data & format

- Anime Dataset
  - training data: 33.4k (image, tags) pair
  - faces/, tags.csv, **testing_tags.txt**
- training tags file format
  - img_id <comma> tag1 <colon> #_post <tab> tag2 <colon> #_post

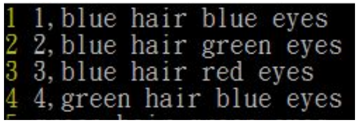
blue eyes
red hair
short hair



```
0,touhou:17705	chen:423	moneti daifuku :60	animal ears:12241	catgirl:4903
1,touhou:17697	onozuka komachi:224	shikieiki yamaxanadu:217	$
2,original:25774	blonde hair:25457	doll:1040	dress:16585	pink eyes:3896	ta
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
  - testing text only includes **‘color hair’** and **‘color eyes’**, only alphabetic char involved.



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

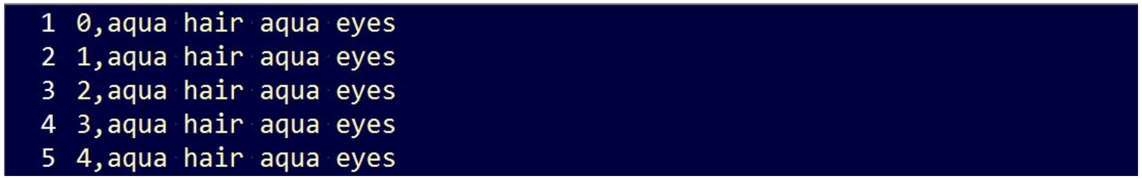
Testing Text Content

- **‘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'.

Data & format

- Extra data
  - training data: 36.7k (image, tags) pair
  - images/, tags.csv
- training tags file format
  - img_id <comma> hair tag <space> eyes tag
  - tags in extra data only includes ‘color hair’ and ‘color eyes’

black eyes
red hair



```
0,aqua hair aqua eyes
1,aqua hair aqua eyes
2,aqua hair aqua eyes
3,aqua hair aqua eyes
4,aqua hair aqua eyes
```

tags.csv

Extra Data & format (2019 Spring Update)

Raw data

- Raw data
  - all_data.zip, all_tags.csv (~25.6GB)
  - tags是用 | 分隔的，意義如下
  - id | hair_style | smile | hat | eyes color | hair color | dark skin |blush | glasses
  - image是 512 x 512 的人臉照
- Processed + Selected data
  - tag.npy, large_image.npy (~1.9GB)
  - tag是120維的one hot，詳細資訊請於const.py查看。
  - large_image是個128 x 128的人臉照，並且有crop過。
  - 你們可以自己用把120維的one-hot壓成22維tag，並討論結果會怎麼樣。
    - (共12種髮色，10種瞳色)，與之前的dataset一樣。
  - 如果要壓成 64 x 64，可以自己resize。
- **Raw data tag很髒！如要做cGAN可以自己處理一下dataset。**
- Processed data的 tag是有做小小的處理該篩選的，雖然好很多但也有點小髒。

Crop data

testing_tags

- 以下這五個tag個產生5張當做你的結果。
  - pink hair black eyes
  - black hair purple eyes
  - red hair red eyes
  - aqua hair green eyes
  - blonde hair orange eyes

Data Link

- Anime Dataset
- Extra Data
- Crypko.ai Data

HW3 Grading Policy:

- HW3-1 Passing baseline , Experiment of tips
- HW3-2 Generate testing_tags image

HW3-2 Report Questions

- Model Description
  - Describe the models you use to, including the model architecture, objective function for G and D.
- Experiment settings and observation
  - Show generated images
- 嘗試把WGAN, WGAN-GP, LSGAN等loss funcion設計的精神融入cGAN/ACGAN/stack GAN等，並討論其是否成功，或嘗試解釋你的設計為什麼合邏輯。
  - 例如WGAN-GP的設計直接套用cGAN，那麼tag呢？
  - 例如LSGAN的設計套到ACGAN的synthesis score，合理嘛？那麼classify的score呢？
  - 例如WGAN有辦法和ACGAN做結合嘛？

Output Format Requirement

- We recommend that each generated image resized to **at least 64 x 64**
- Generate 25 image into one png
    - sample code is in baseline.py
    - 為防止同學產生的圖片不一致，請同學使用baseline.py裡的**save_imgs()**
- 放在report或討論的時候，請先用64 x 64做好baseline。之後你可以嘗試generate更大的image size，並討論你怎麼改或選擇model的，還有中間所遇到的困難(例如比較難train，或是有diminished gradient發生之類的)。(Optional)
- 只需要放圖片在report or slide 即可，不用reproduce。

Q&A

b05902013@csie.ntu.edu.tw
b05902127@csie.ntu.edu.tw