

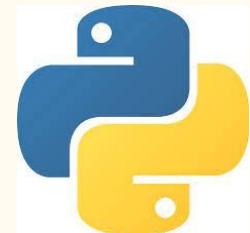
EDITH Presentation



Team D
Chingis
Eunmin Kim
Gong He
Soohun Park

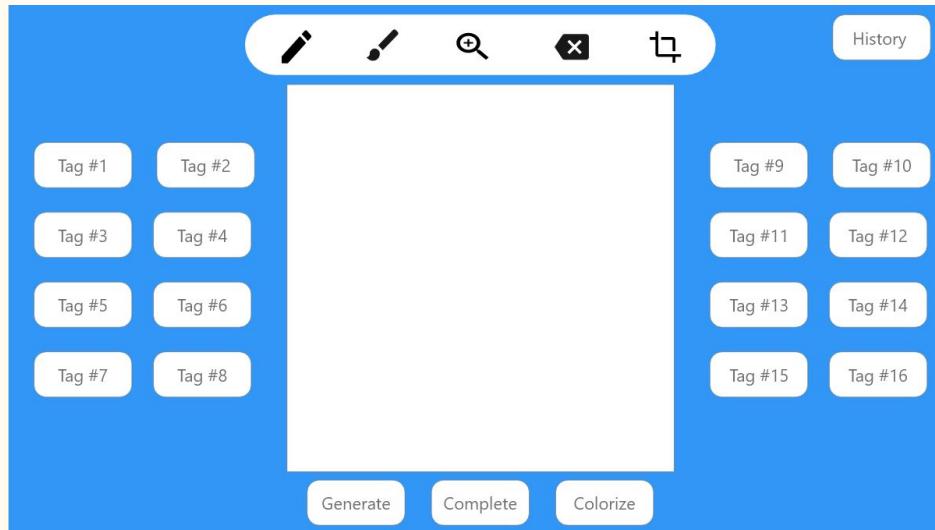
Implementation

- Frontend: JQuery
- Backend: Django / FastAPI
- ML/AI: Pytorch, AWS
- APIs: Fast API (with Python), AWS or Heroku
- Open Source Borrowed:
 - Female dataset : Kaggle
(<https://www.kaggle.com/scribbless/another-anime-face-dataset>)
 - Face crop AI Model:
(https://github.com/nagadomi/lbpcascade_animeface)
- Programming Languages: Python, Javascript



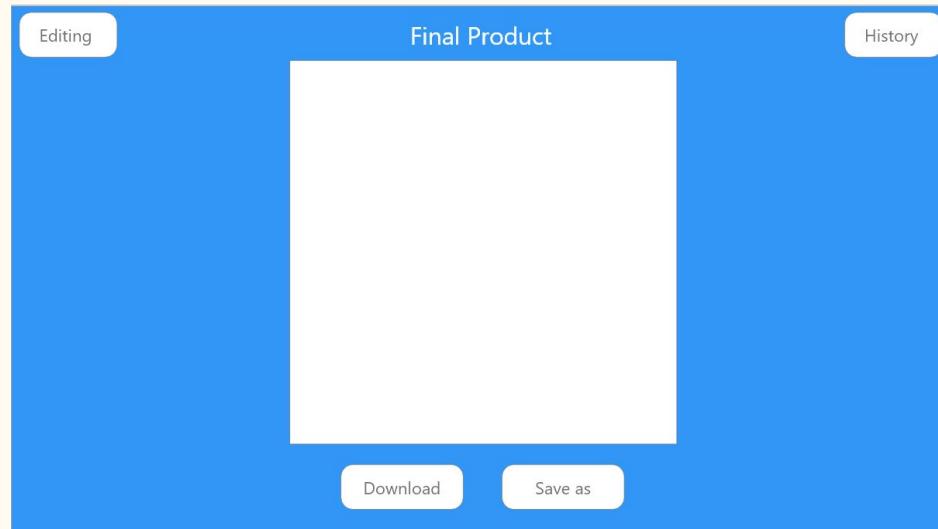
Design

Screen 1



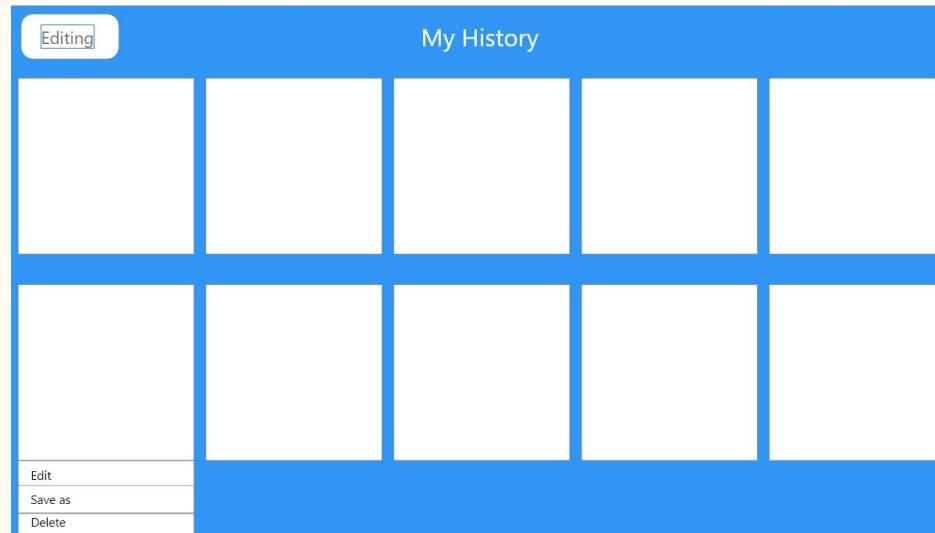
Design

Screen 2



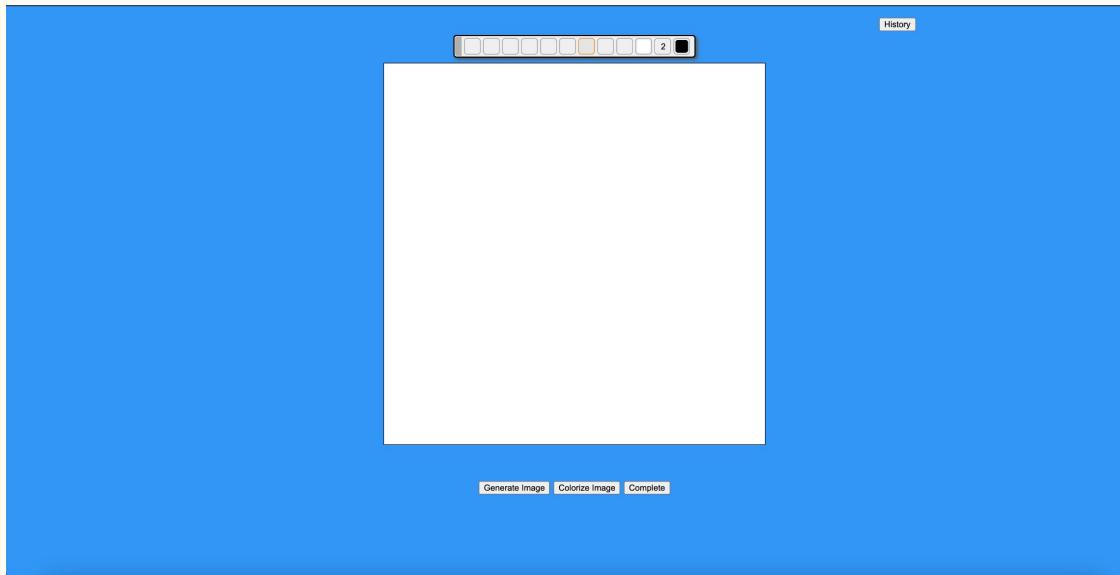
Design

Screen 3



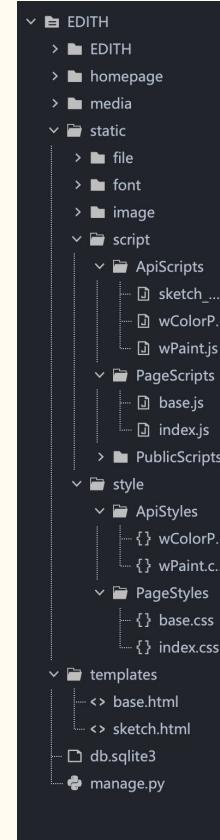
Frontend Start

Frontend: Screen 1 (JQuery)



Backend

- Start Django project in Django 3.0.6 version.
- Start “homepage” app at Django project to process all page requests.
- Start “api” app at Django project to process all api requests.
- Add session energy to Django project setting.



Completed backend path

Editing Tools

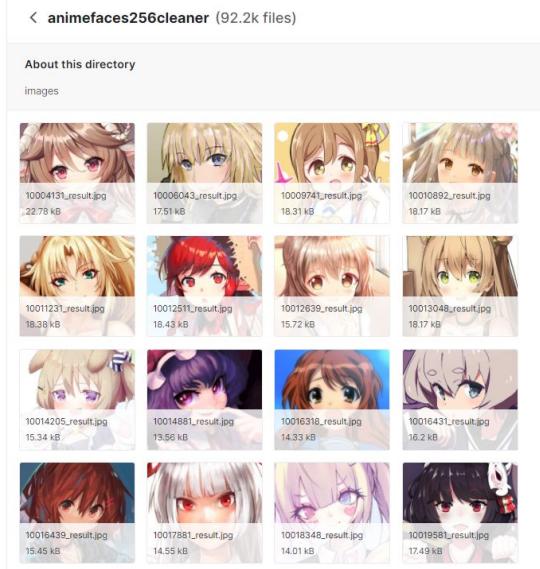
- Free movement
- Undo
- Redo
- Clear
- Rectangle
- Ellipse
- Line
- Pencil
- Text
- Eraser
- Weight
- Color



Editing Tools

Dataset Preparation

Plenty of female dataset(kaggle)



No male dataset



Dataset Crawling

safebooru.or

Safebooru

My Account Posts Comments Aliases Artists Tags Pools Forum Stats Twitter Favourites

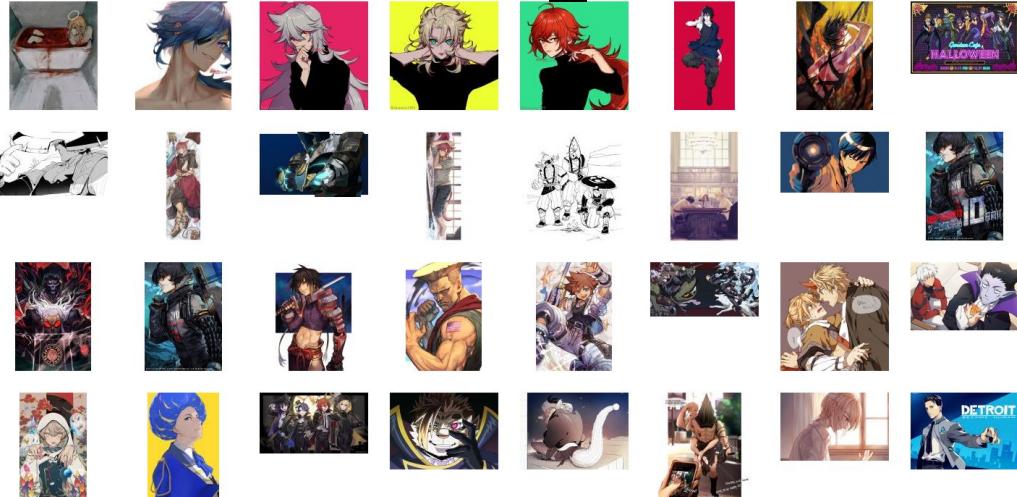
List 3D List Upload Random Contact Us About Help ToS Privacy Policy

Search Search (Supports wildcard *)

Tags

+ - chainsaw man 2759
+ - genghis Impact 25123
Character
+ - kaiya (genghis impact) 327
+ - iboy 300383
+ - male devil (chainsaw man) 107
+ - angel wings 10609
+ - bathtub 1268
+ - blood 44160
+ - bloody wife 40
+ - blue eyes 694916
+ - blue hair 318019
+ - brown hair 620319
+ - closed mouth 266386
+ - collarbone 184653
+ - dark-skinned male 1898
+ - date 67700
+ - eyepatch 35362
+ - face 26324
+ - halo 15303
+ - highres 1499465
+ - light brown hair 25562
+ - long hair 1561396
+ - looking at viewer 988977
+ - male focus 120911
+ - normaoz0 54
+ - male 8328
+ - ojo 55
+ - one eye covered 61211
+ - parted lips 127133
+ - portrait 20287
+ - red nose 516240

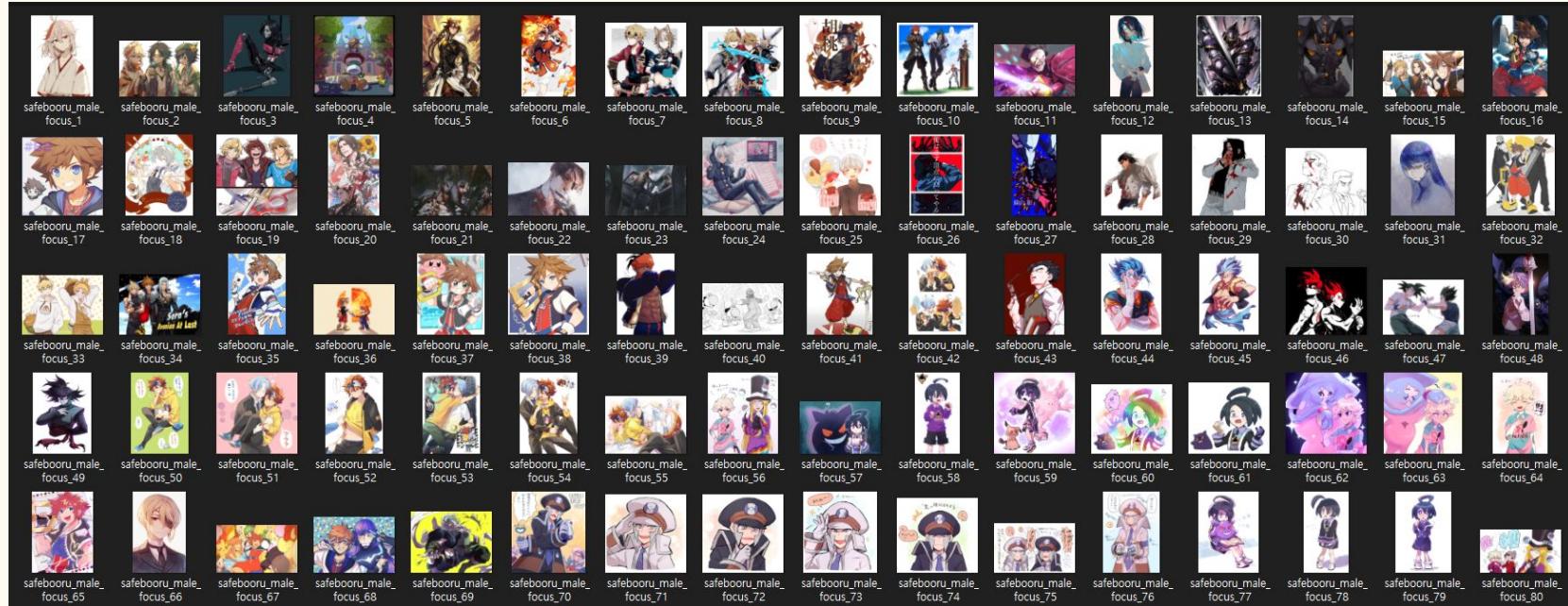
Copyright : male_focus



Dataset Crawling - First Try

26,636 images

BUT we need face dataset



Dataset Crawling - First Try

Anime-Face-Detector

A Faster-RCNN based anime face detector.

This detector is trained on 6000 training samples and 641 testing samples, randomly selected from the dataset which is crawled from top 100 pixiv daily ranking.

Thanks to [OpenCV based Anime face detector](#) written by nagadomi, which helps labelling the data.

The original implementation of Faster-RCNN using Tensorflow can be found [here](#)



30,112 images

Challenge #1: Dataset

Simply crawling images result in many noisy samples. Noisy samples include random crops or non-human images.



Violence



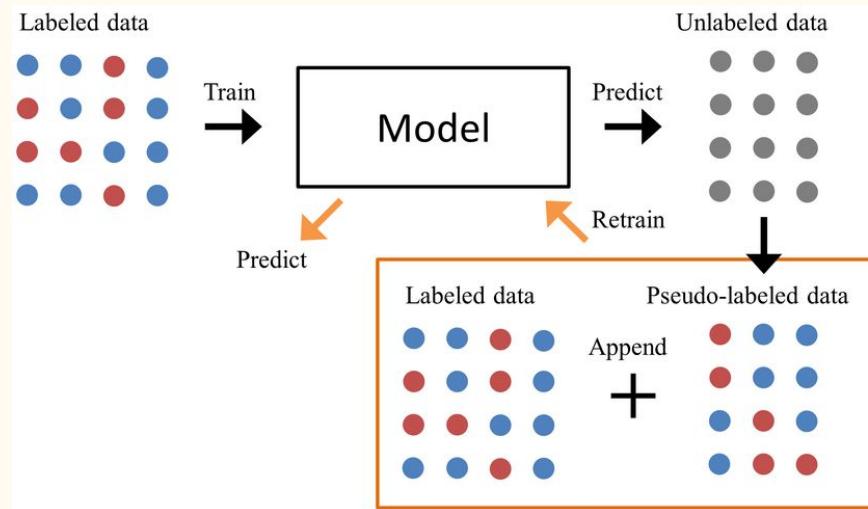
Non-human



Bad crops

Trial and error: Pseudo Labeling

It can distinguish male and females but it cannot distinguish good and bad samples.



Solution: Metric Learning

We can mine kNN of good samples by using embeddings of the images.

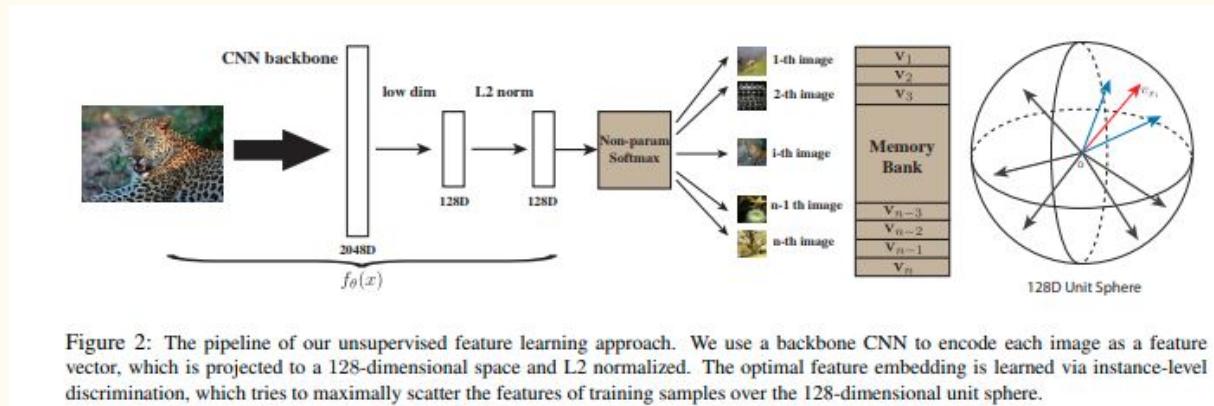


Figure 2: The pipeline of our unsupervised feature learning approach. We use a backbone CNN to encode each image as a feature vector, which is projected to a 128-dimensional space and L2 normalized. The optimal feature embedding is learned via instance-level discrimination, which tries to maximally scatter the features of training samples over the 128-dimensional unit sphere.

$$P(i|\mathbf{v}) = \frac{\exp(\mathbf{v}_i^T \mathbf{v} / \tau)}{\sum_{j=1}^n \exp(\mathbf{v}_j^T \mathbf{v} / \tau)},$$

Dataset Crawling - Second Try

Crawled 9,386 images → Cropped face part → 10,669 images

Crawled also the meta tags in the html file

```
safebooru_male_focus_1.jpg 1boy,bangs,black_vest,brown_hair,collarbone,collared_shirt,commentary_request,dress_shirt,eyebrows_vis  
safebooru_male_focus_2.jpg 1boy,black_bblindfold,black_gloves,black_legwear,blindfold,boots,child,gloves,jumping,long_sleeves,male_  
safebooru_male_focus_3.jpg 2boys,ahat_(ragnarok_online),bangs,bat,black_coat,black_hair,blonde_hair,blue_eyes,choker,coat,commenta  
safebooru_male_focus_4.jpg 1boy,blurry,blurry_background,boku_no_hero_academia,ceiling,commentary,english_commentary,forehead,from  
safebooru_male_focus_5.jpg 1boy,ahat_(ragnarok_online),ahoge,bangs,blonde_hair,blue_eyes,blush,bow,commentary_request,corset,cross  
safebooru_male_focus_6.jpg 1boy,ahat_(ragnarok_online),ahoge,alternate_color,bangs,blonde_hair,blue_eyes,blush,bow,commentary_requ  
safebooru_male_focus_7.jpg 1boy,bangs,blue_eyes,brown_cape,brown_gloves,brown_headwear,brown_pants,brown_shirt,cape,card,card_in_m  
safebooru_male_focus_8.jpg 1boy,bangs,black_hair,en'&#039;en_no_shouboutai,hair_between_eyes,highres,looking_at_viewer,male_focus,s  
safebooru_male_focus_9.jpg 2boys,ahat_(ragnarok_online),ahoge,animal_collar,animal_ear_fluff,animal_ears,bangs,blonde_hair,blue_ey  
safebooru_male_focus_10.jpg 1boy,absurdres,bangs,black_hair,en'&#039;en_no_shouboutai,fire,glowing,glowing_eyes,hair_between_eyes,h  
safebooru_male_focus_11.jpg 1boy,animal_ears,bangs,belt,blue_shirt,blush_stickers,brown_bag,brown_belt,brown_eyes,brown_hair,brown
```

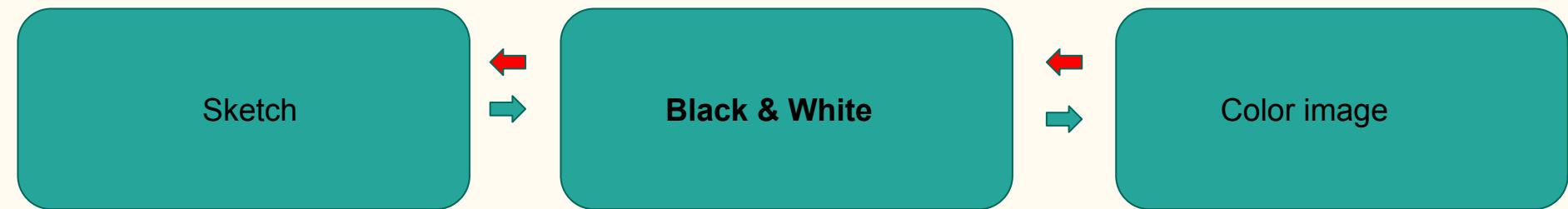
Total 12K images

Male : 6k images

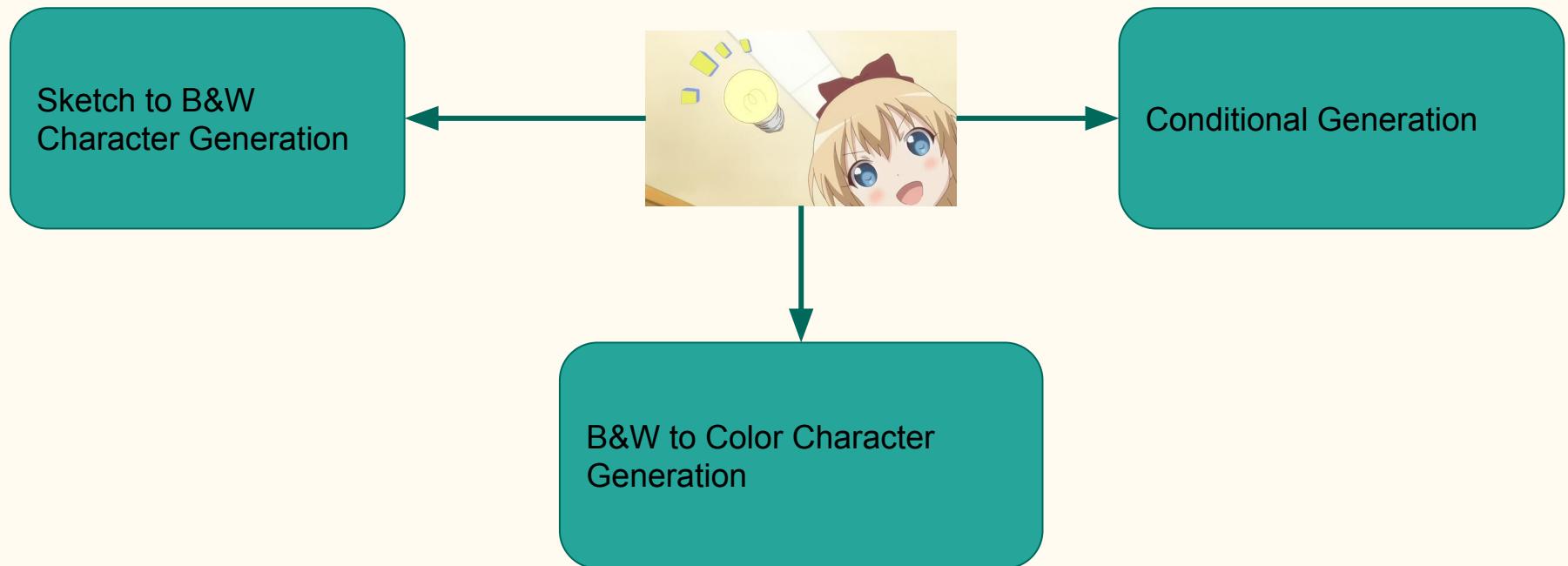
Female : 6k images

Ready to train

Drawing steps

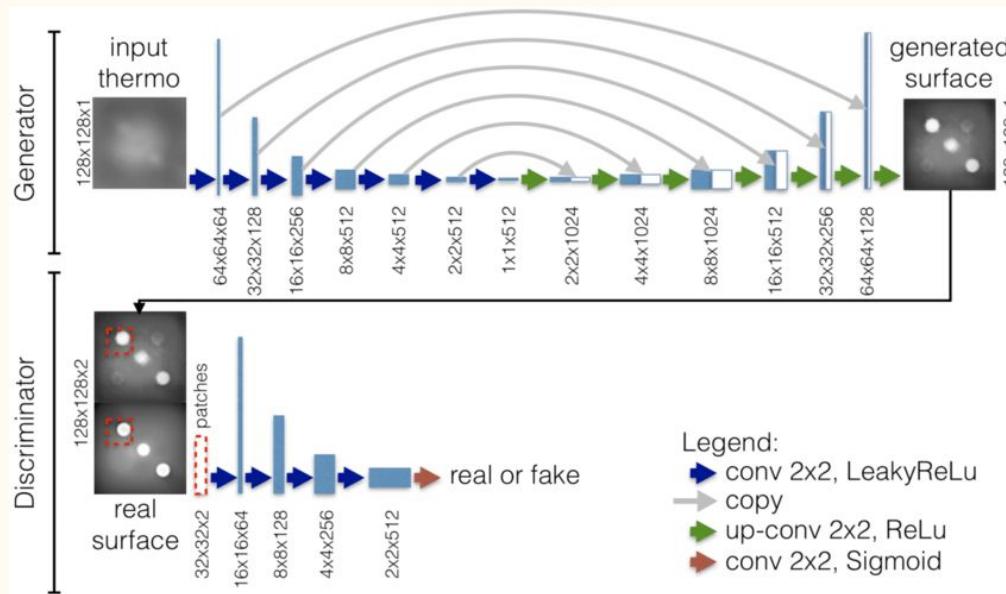


Machine Learning

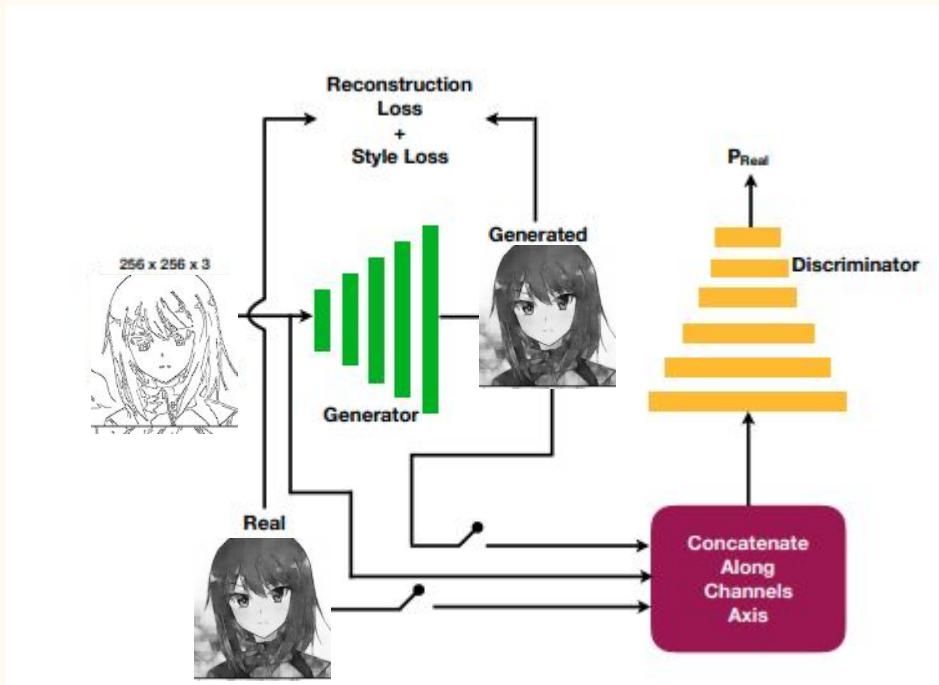


Pix2Pix GAN

“Image-to-Image Translation with Conditional Adversarial Networks” proposed by Isola et al. at CVPR 2017.



Framework



Our loss function

Discriminator loss

$$\max_D \mathbb{E}_{\mathbf{x} \sim p_{\text{Real}}} [\log D(\mathbf{x})] + \mathbb{E}_{\mathbf{z} \sim p_{\text{Noise}}} [\log(1 - D(G(\mathbf{z})))]$$

Generator loss

$$\min_G -\mathbb{E}_{\mathbf{z} \sim p_{\text{Noise}}} [\log D(G(\mathbf{z}))]$$

Reconstruction loss

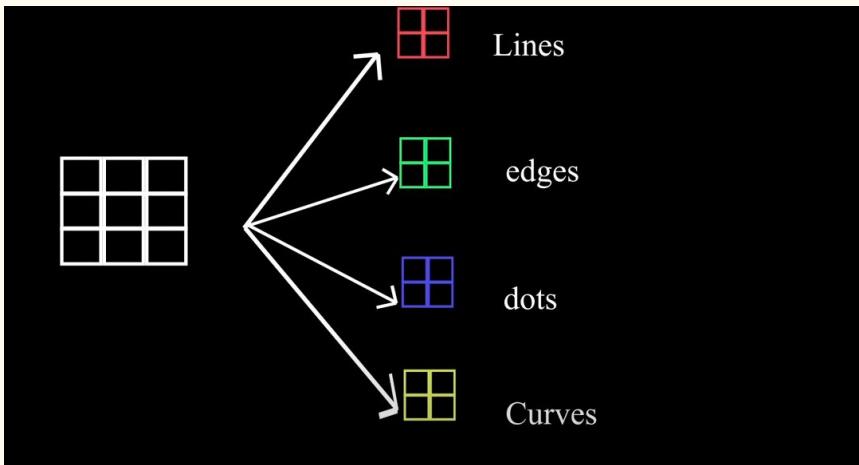
$$\|M_{\text{fake}} - M_{\text{real}}\|_2$$

Style loss

$$G_{ij} = \sum_k F_{ik} F_{jk}$$

$$L_{\text{style}} = \frac{1}{4N^2M^2} (G_{ij} - A_{ij})^2$$

Gram matrix intuition



Degree of correlation = Style of an image

$$\begin{array}{ccc} \begin{matrix} \textcolor{red}{\square} & & \\ & \textcolor{green}{\square} & \\ & & \textcolor{blue}{\square} \end{matrix} & * & \begin{matrix} \textcolor{red}{\square} & \textcolor{green}{\square} & \textcolor{blue}{\square} & \textcolor{yellow}{\square} \\ \textcolor{green}{\square} & \textcolor{red}{\square} & \textcolor{blue}{\square} & \textcolor{yellow}{\square} \\ \textcolor{blue}{\square} & \textcolor{blue}{\square} & \textcolor{red}{\square} & \textcolor{yellow}{\square} \\ \textcolor{yellow}{\square} & \textcolor{yellow}{\square} & \textcolor{yellow}{\square} & \textcolor{red}{\square} \end{matrix} & = & \begin{matrix} & & \\ & & \\ & & \\ & & \\ & & \\ & & \end{matrix} \\ \text{Gram matrix} & & & & \\ \mathbf{G} = \mathbf{V} \cdot \mathbf{V}^T & & & & \end{array}$$

Sketch to B&W Character Generation

Preprocessed images (Bilateral filter -> Canny edge detector)



Generated Images (Model is still training at the moment)



Pix2Pix

B&W to Color Character Generation

Preprocessed images (leave some color marks)

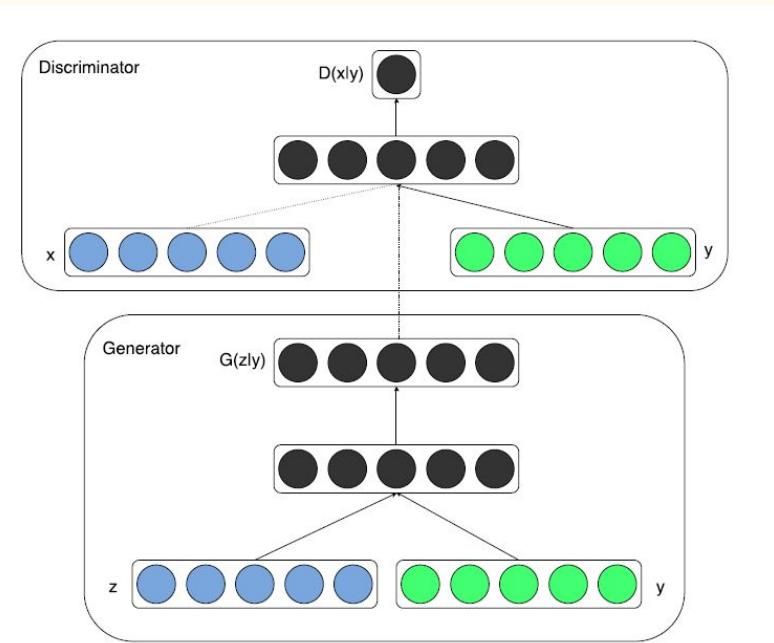


Generated Images



Conditional GAN

“Generative Adversarial Text to Image Synthesis” by Reed et al.



Algorithm 1 GAN-CLS training algorithm with step size α , using minibatch SGD for simplicity.

```
1: Input: minibatch images  $x$ , matching text  $t$ , mis-matching  $\hat{t}$ , number of training batch steps  $S$ 
2: for  $n = 1$  to  $S$  do
3:    $h \leftarrow \varphi(t)$  {Encode matching text description}
4:    $\hat{h} \leftarrow \varphi(\hat{t})$  {Encode mis-matching text description}
5:    $z \sim \mathcal{N}(0, 1)^Z$  {Draw sample of random noise}
6:    $\hat{x} \leftarrow G(z, h)$  {Forward through generator}
7:    $s_r \leftarrow D(x, h)$  {real image, right text}
8:    $s_w \leftarrow D(x, \hat{h})$  {real image, wrong text}
9:    $s_f \leftarrow D(\hat{x}, h)$  {fake image, right text}
10:   $\mathcal{L}_D \leftarrow \log(s_r) + (\log(1 - s_w) + \log(1 - s_f))/2$ 
11:   $D \leftarrow D - \alpha \partial \mathcal{L}_D / \partial D$  {Update discriminator}
12:   $\mathcal{L}_G \leftarrow \log(s_f)$ 
13:   $G \leftarrow G - \alpha \partial \mathcal{L}_G / \partial G$  {Update generator}
14: end for
```

Conditional Generation

Generated Images (still in the process...)



Challenge #2: Generation Model

- Discriminator becomes over confident, Generator struggles to fool the Discriminator.
- Mode Collapse.
- The tra



How we are addressing it.

Over confidence:

- Label smoothing (0.9).
- Adding noise to inputs of Discriminator.

Generator struggles to fool the Discriminator:

- Double or triple steps for Generator per Discriminator update.
- Different learning rates (0.0004 and 0.0002 for Discriminator and Generator respectively).

Mode collapse:

- Gradient penalty as in “Wasserstein GAN” by Arjovsky et al.

To try:

- Different architecture since it might have a better capacity.

Evaluation

1. The **Fréchet inception distance (FID)** is a metric used to assess the quality of images created by a generative model, like a generative adversarial network (GAN). Unlike the earlier inception score (IS), which evaluates only the distribution of generated images, the FID compares the distribution of generated images with the distribution of real images that were used to train the generator.
2. Evaluate visually.



Limitations

- Constraints/Assumptions when user sketches:
 - Is a human (Not an animal or inanimate object)
 - Only face + neck, not full body
 - Includes desired key features (Nose, eyes, lips, etc)

Question Time