



딥러닝으로 덕업일치하기

NAVER AI LAB 이가영

* 네이버 웹툰에 있을 때 진행한 연구ㅎㅎ;

딥러닝

모델

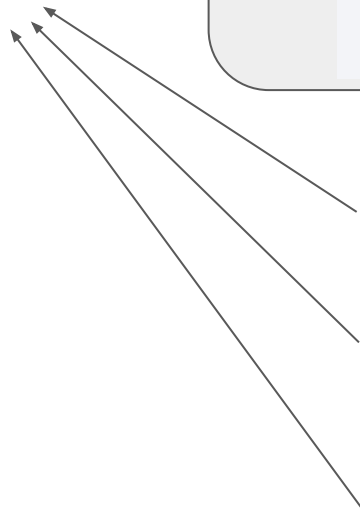
인풋
(img)



```
class Net(nn.Module):  
    def __init__(self):  
        super(Net, self).__init__()  
        self.conv1 = nn.Conv2d(3, 6, 5)  
        self.pool = nn.MaxPool2d(2, 2)  
        self.conv2 = nn.Conv2d(6, 16, 5)  
        self.fc1 = nn.Linear(16 * 5 * 5, 120)  
        self.fc2 = nn.Linear(120, 84)  
        self.fc3 = nn.Linear(84, 10)  
  
    def forward(self, x):  
        x = self.pool(F.relu(self.conv1(x)))  
        x = self.pool(F.relu(self.conv2(x)))  
        x = x.view(-1, 16 * 5 * 5)  
        x = F.relu(self.fc1(x))  
        x = F.relu(self.fc2(x))  
        x = self.fc3(x)  
        return x
```

아웃풋
(label)

피드백



오토바이

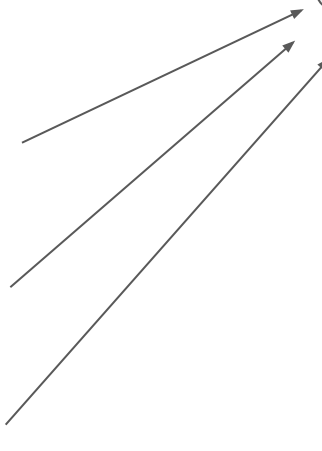


오토바이



자동차

데이터



img2img 분야



오토바이

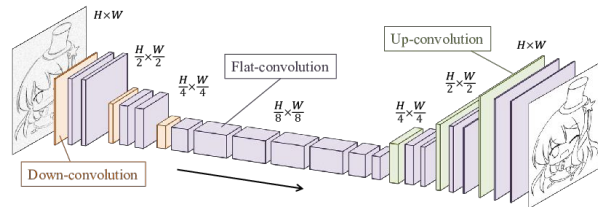


Image-to-Image Translation with Conditional Adversarial Networks, Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, Alexei A. Efros CVPR17

Labels to Street Scene

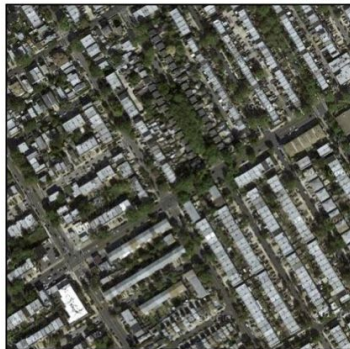


input



output

Aerial to Map

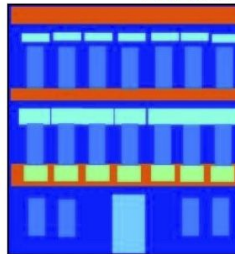


input



output

Labels to Facade



input



output

BW to Color



input



output

Day to Night



input



output

Edges to Photo



input



output

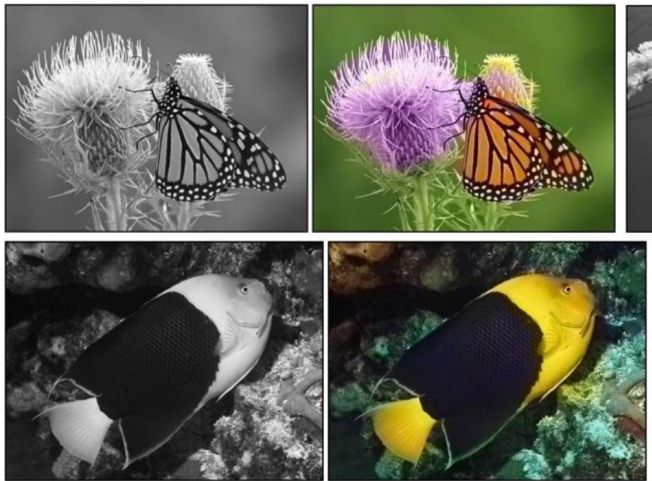
덕심을 자극하는 img2img 분야

2D 전문가로서 그냥 지나칠 수 없는 수많은 덕후들...



자동 채색

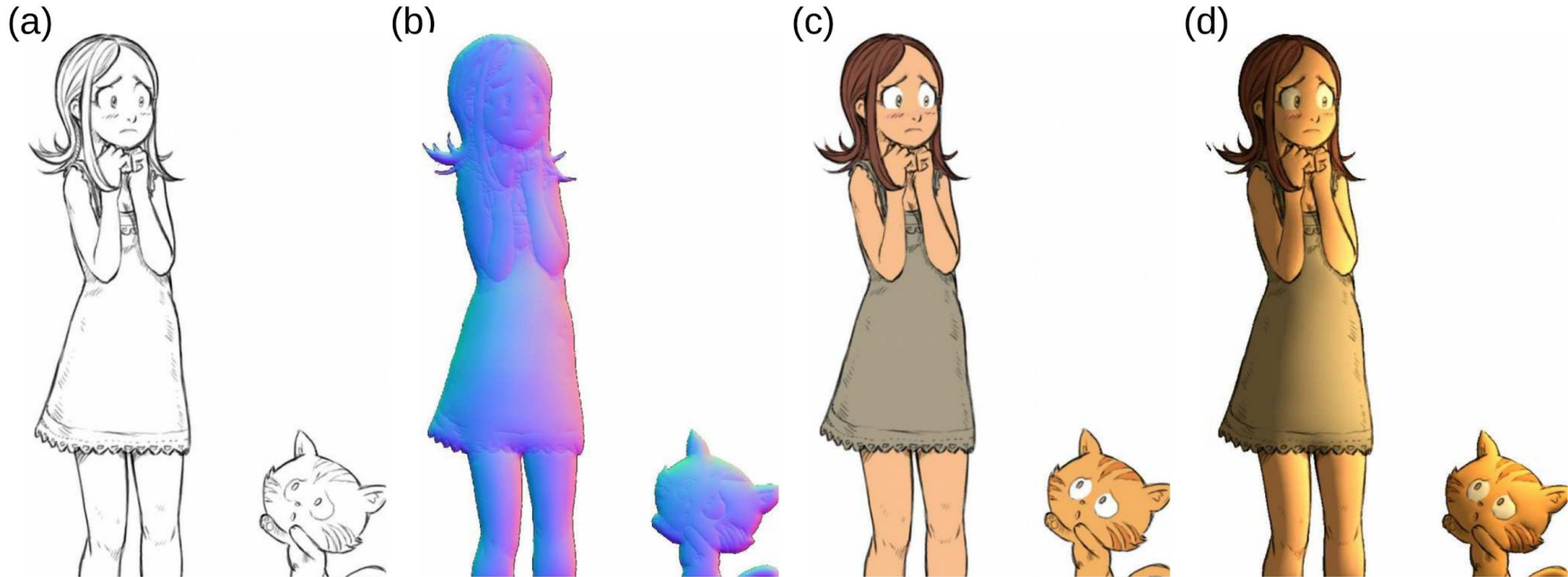
Colorful Image Colorization, Richard Zhang et al. (ECCV 16)



User-Guided Deep Anime Line Art
Colorization with Conditional Adversarial
Networks, Yuanzheng Ci et al. (ACMM 18)



자동 shading



Deep Normal Estimation for Automatic Shading of Hand-Drawn Characters, Matis
Hudon et al. ECCVW_2018

Selfie2Anime



site: <https://selfie2anime.com/>

paper : U-GAT-IT: Unsupervised Generative Attentional Networks with Adaptive Layer-Instance Normalization for Image-to-Image Translation, Junho Kim, Minjae Kim, Hyeonwoo Kang, Kwanghee Lee, ICLR20

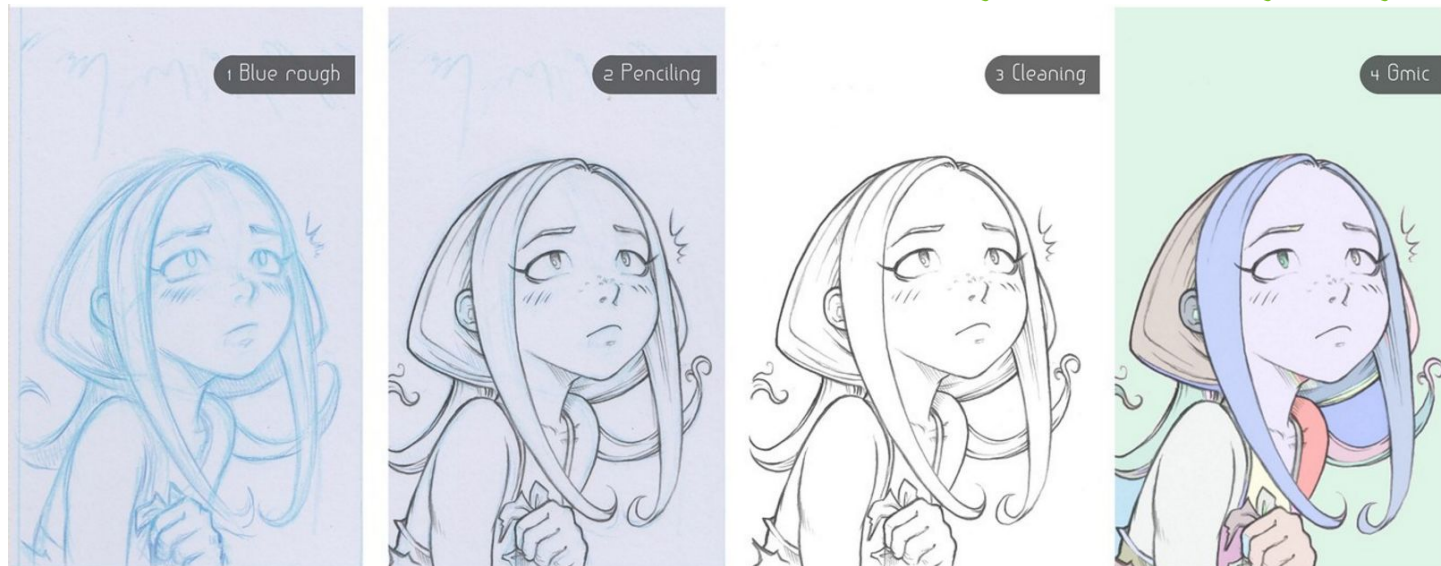
Pose to char

Full-body High-resolution Anime Generation with
Progressive Structure-conditional Generative
Adversarial Networks, Koichi Hamada et al.
ECCVW 18



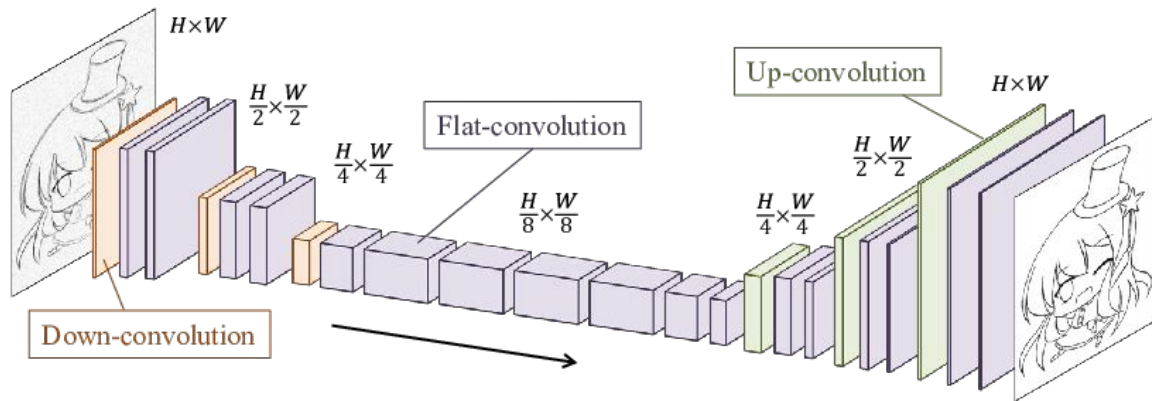
진행한 연구 : 스케치 선따기

© <https://www.davidrevoy.com>, David Revoy, CC-BY-4.0

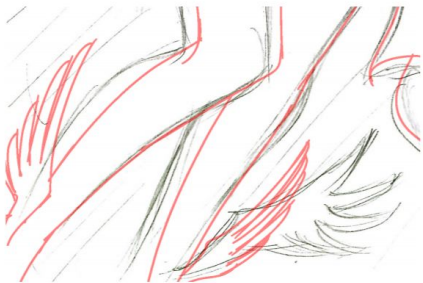


선따기

기존 연구 : 스케치 선따기

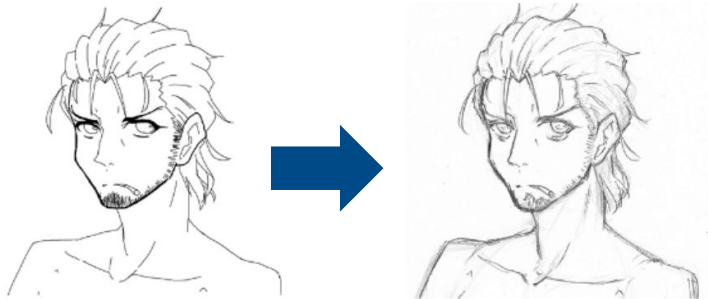


가장 큰 문제는 데이터



실제로 짝(pair)이 있는 데이터 :

- 수집하기에는 수가 너무 적음
- 스케치에서 선 따면서 그림이 종종 바뀜



이전 논문에서 제안한 방식 :

선화 위에 다시 스케치를 그려 내서 사용하는 방식

- 어쨌든 사람이 그려야 해서 모으기 귀찮다

스케치를 자동으로 생성하면 어떨까

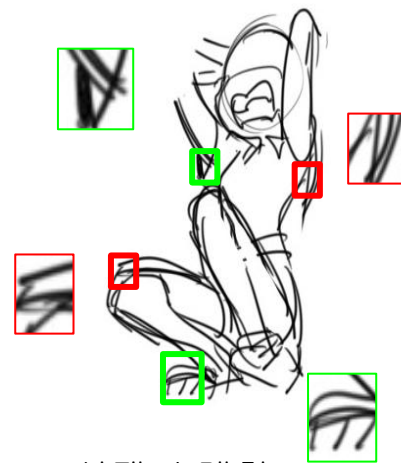


사람이 스케치를 만들지 말고, 적당히 시뮬레이션해서 스케치를 만들어서 데이터로 써 보자!

MUNIT 등

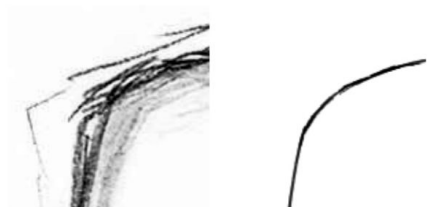
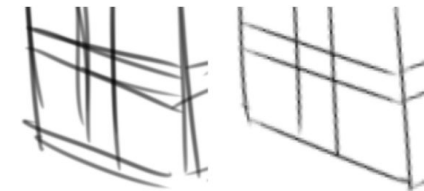
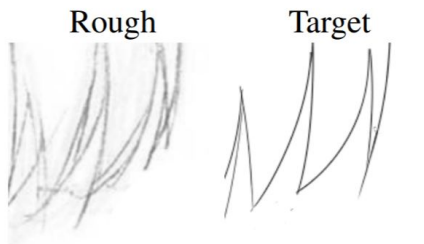
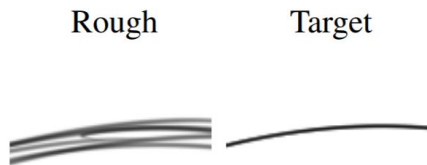
Unsupervised img to img 모델

- 짝 이미지가 없이 스케치, 펜션 이미지들만 있어도 변환하는 함수 학습 가능



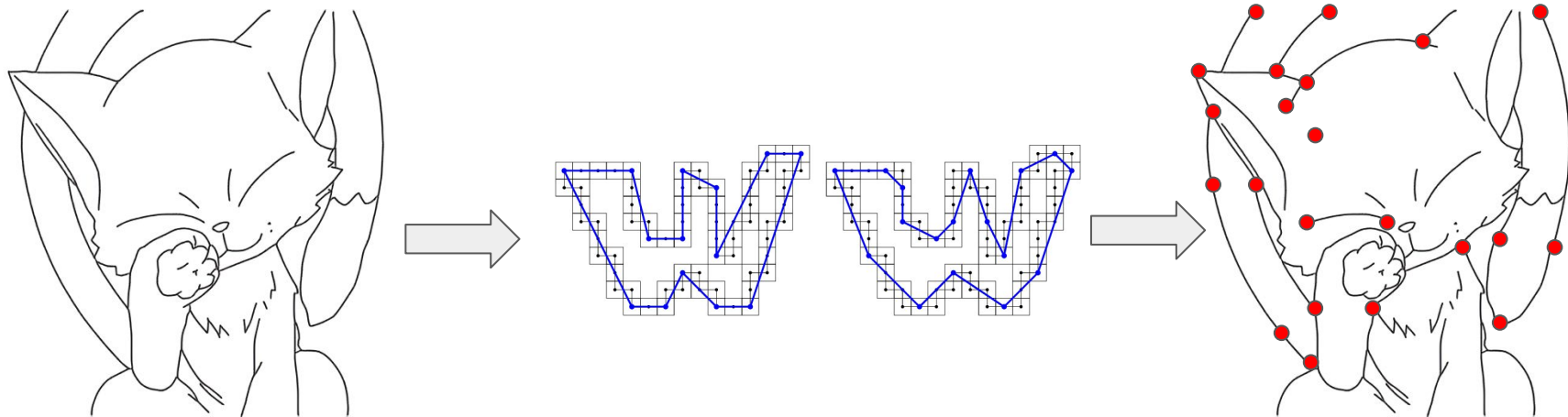
실제 스케치

스케치 선 관찰

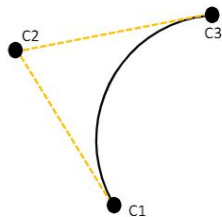


- 원래 선과 기울기나 위치가 다른 선
- 한번에 못 긋고 여러 번 그음
- 짧은 선분으로 나눠서 그리기도
- 선분 사이에 틈도 있음

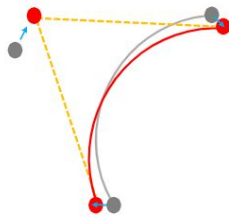
Potrace를 이용해서 vectorize하기



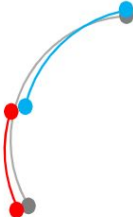
선을 shake it shake it



(a)



(b)



(c)



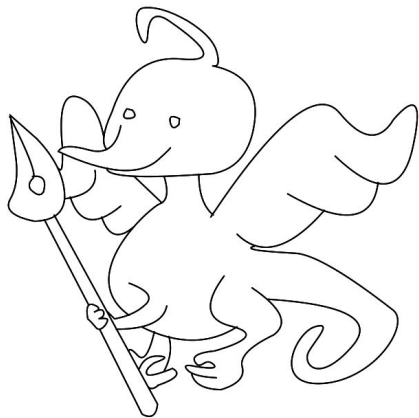
(d)

$$B(c_1, c_2, c_3) = (1 - t)^2 c_1 + 2(1 - t)tc_2 + t^2 c_3,$$

컨트롤 점의 위치를 약간씩 움직이기 -> 위치와 기울기가 약간 다른 선
커브를 중간에 끊어서 따로 따로 노이즈 주기 -> 여러 짧은 선을 겹쳐 그리는 방식
한 선분에 여러 번 겹쳐서 생성 -> 여러 번에 걸쳐서 겹쳐 그리는 스케치 선 묘사

노이즈의 강도 조절

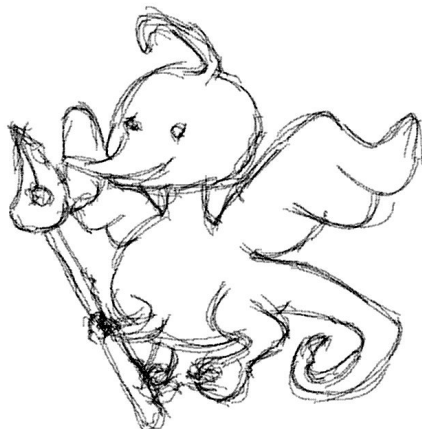
Input Line



$\alpha = 30, M = 1$



$\alpha = 30, M = 3$



$\alpha = 50, M = 4$



M : 겹치는 횟수, α : 노이즈의 세기

단순한 이미지 프로세싱과 비교

Input Line



Slur



Add Noises



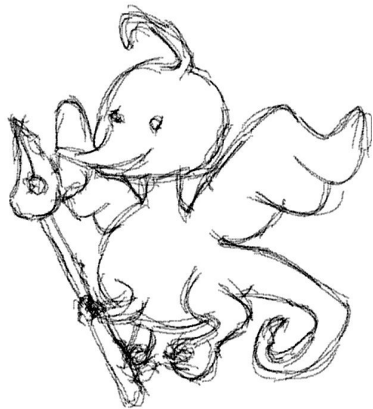
Our Augmentation



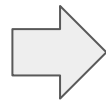
음영 및 배경 추가



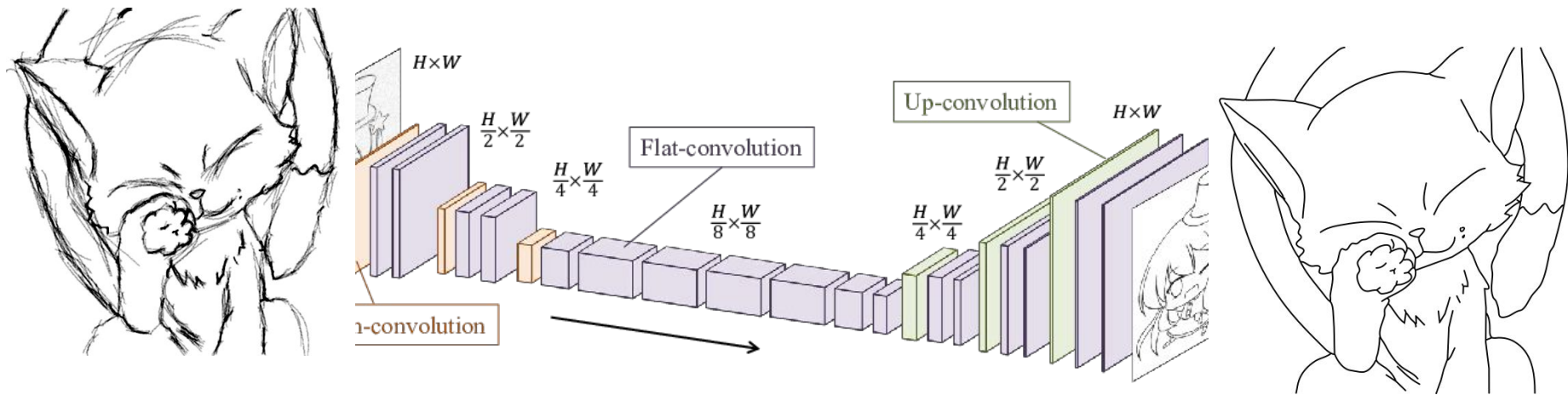
텍스처 추가



+



모델 학습



적당한 모델에 생성한 짝 데이터를 넣어서 학습하기

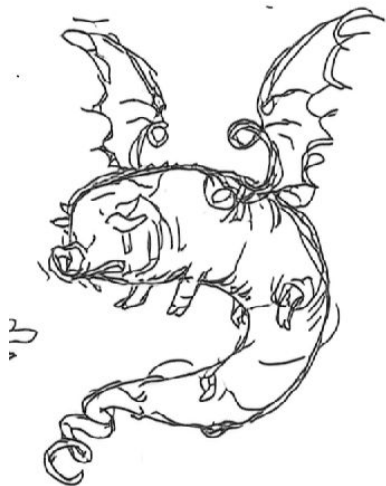
결과

© <https://www.davidrevoy.com>, David Revoy, CC-BY-4.0



Input Sketch

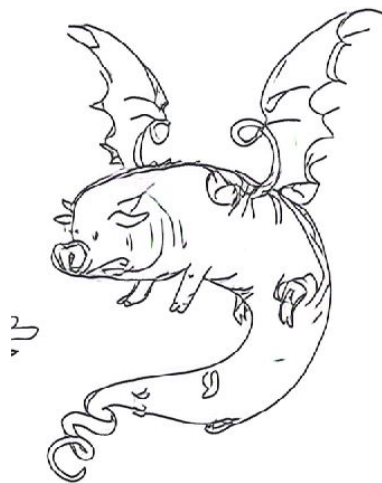
* Resized to 512 X 512



CycleGAN



Mastering
Sketching



Our method

결과

© <https://www.davidrevoy.com>, David Revoy, CC-BY-4.0



Input Sketch

* Resized to 512 X 512



CycleGAN



Mastering
Sketching



Our method

결과



결과



감사합니다!