# iGAN을 활용한 '캐릭터 자판기' Project 중간평가

000조 고민주 오영진 오현지

### INDEX

- 01 Project 목적
- 02 iGAN 논문 & 모델
- 03 Project 진행상황
- 04 Project 변경사항

# 01 Project 목적

캐릭터의 단순 생성이 아닌, 게임의 분위기와 스토리에 맞는 캐릭터 창작 필요 캐릭터 디자이너의 요구사항에 따른 다양한 캐릭터 생성이 필요





# 01 Project 목적



# 01 Project 목적

게임 캐릭터 디자이너가 게임 속 특정 배경이나 컨셉이 필요한 캐릭터를 디자인할 때 원하는 이미지의 캐릭터 초안을 제공



"Generative Visual Manipulation on the Natural Image Manifold"

2016년 기존 original-GAN에 GVM(Generative Visual Manipulation)기술을 추가하여 발표한 논문으로 사용자의 요구에 따른 다양한 이미지의 형태 변환을 시행하였다.

DCGAN을 이용하여 단순한 이미지 생성

+

AlexNet, HogNet을 이용하여 interactive하게 이미지 형태 변환

Generator와 Discriminator 학습 (DCGAN)

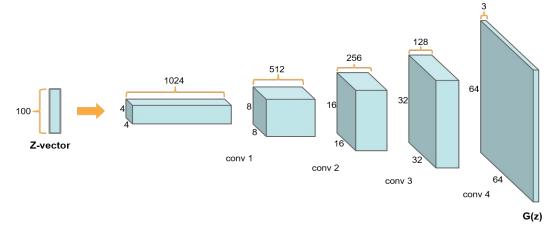
Z-vector 학습 (AlexNet)

Interactive Model (HogNet)

#### Model Parameter

Model constructing input Parameter				
n_layers	3	# of convolution layers		
n_f	128	# of (last) feature channels		
npx	64	height = width		
nc	3	# of image channels(RGB)		
nz	100	# of dim for Z		
niter	25	# of iter at starting learning rate		
niter_decay	25	# iter to linearly decay learning rate to zero		
init_sz	4	# of first feature map size(4*4)		
fs	5	Fractionally-strided convolution filter size		
bs	32	batch_size		
Ir	0.0002	learning_rate		
niter	25	# of iter before decay Ir		
niter_decay	25	# of inter after decay Ir		
b1	0.5	Momentum		

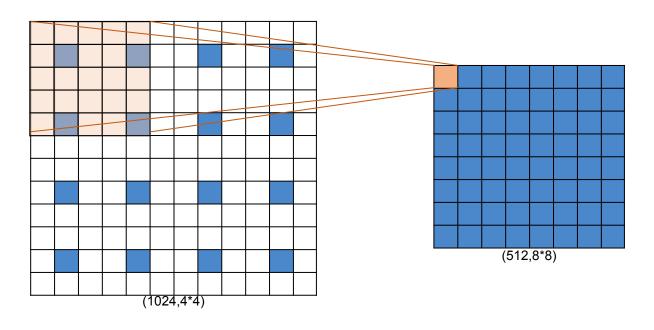
Generator (DCGAN)



Parameter(G)	Project & reshape	Layer 1	Layer 2	Layer 3	Layer 4
Weight	(100,1024*4*4)	(1024,512,5,5)	(512,256,5,5)	(256,128,5,5)	(128,3,5,5)
Gain	4*4*1024	512	256	128	3
Bias	4*4*1024	512	256	128	3
Activation F	Relu	Relu	Relu	Relu	Tanh
Batch Norm	0	0	0	0	X

### Generator (DCGAN)

- Fractionally-strided convolution



Discriminator (DCGAN)

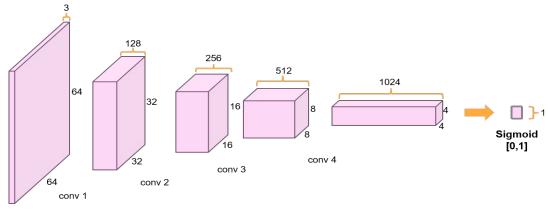
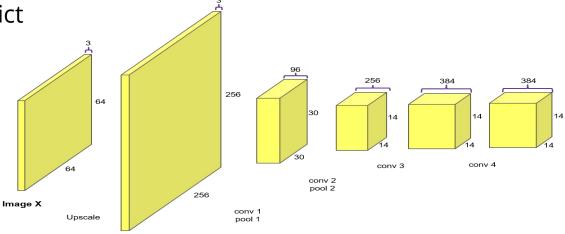


Image X

Parameter(D)	Project & reshape	Layer 1	Layer 2	Layer 3	Layer 4
Weight	(128,3,5,5)	(256,128,5,5)	(512,256,5,5)	(1024,512,5,5)	(1024*4*4*4)
Gain	X	256	512	1024	1
Bias	X	256	512	1024	1
Activation F	X	LeakyRelu	LeakyRelu	LeakyRelu	Sigmoid
Batch Norm	X	0	0	0	X

Z-vector predict (AlexNet)



Parameter(P)	Upscale(x4)	Layer 1	Layer 2	Layer 3	Layer 4
Weight	(3,256,256)	(96,3,11,11)	(256,96,5,5)	(384,256,3,3)	(384,384,3,3)
Filter	X	11*11	5*5(padding)	3*3(padding)	3*3(padding)
stride	X	4	X	Χ	X
pool	Χ	3*3 ,stride=2	3*3 ,stride=2	Χ	Χ

### Z-vector predict (AlexNet)

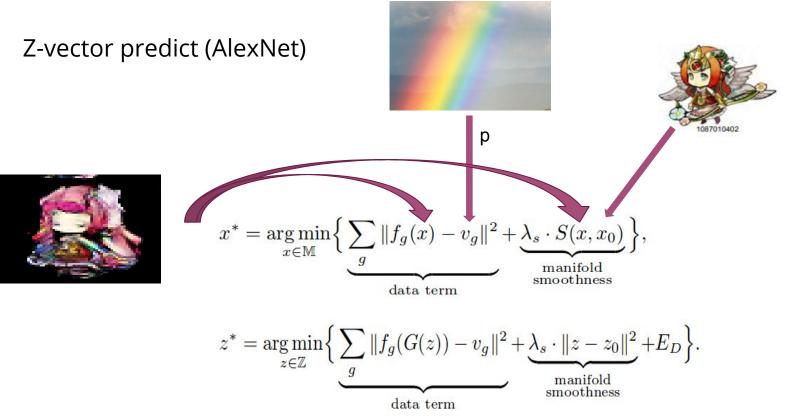
$$x^* = \underset{x \in \tilde{\mathbb{M}}}{\arg \min} \ \mathcal{L}(x, \ x^R).$$

 $z^* = \underset{z \in \tilde{\mathbb{Z}}}{\operatorname{arg \, min}} \ \mathcal{L}(G(z), \ x^R).$ 

#### cost

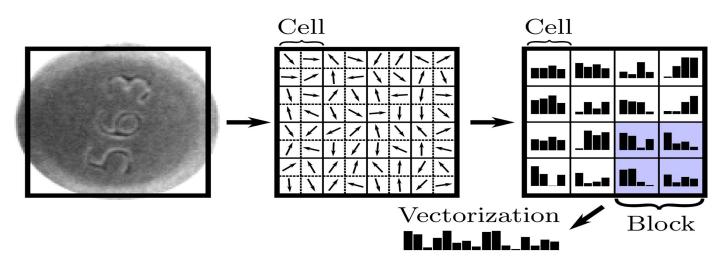
pixel wise error를 사용한것 보다 AlexNet Conv4의 결과인 feature map간의 error를 사용했을 때 더 좋은 성능이 나온다.

$$heta_P^* = rg \min_{ heta_P} \sum_n \mathcal{L}(G(P(x_n^R; heta_P)), \, x_n^R),$$

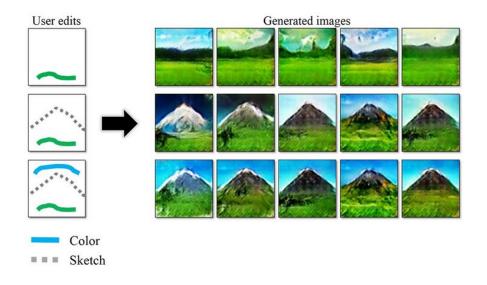


### Interactive Model (HogNet)

- 대상 영역을 일정 크기의 셀로 분할
- 각 셀마다 edge 픽셀(gradient magnitude가 일정 값 이상인 픽셀)들의 방향에 대한 히스토그램을 구함
- 히스토그램 bin 값들을 일렬로 연결하여 벡터 생성



학습 dataset은 64x64 shoes, handbag, outdoor image dataset으로 구분 - 64x64 outdoor dataset으로 학습한 후 iGAN 실행한 결과



### 성능 지표

- 내부 모델끼리(HOG\_NET/Predict\_deep\_learning) 비교하기 위해 사용한 성능지표



<= pixel\_wise error =>



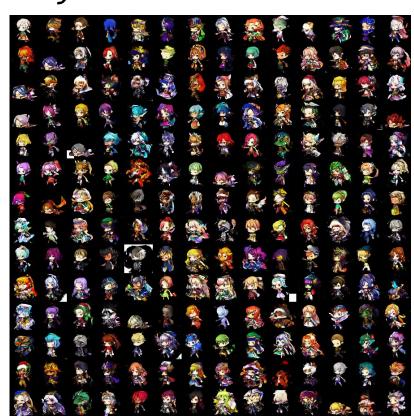
### 게임 캐릭터 dataset 학습

- 약 2만개의 캐릭터 이미지 64x64 사이즈로 분할
- 캐릭터 dataset HDF5 파일로 생성 (charac\_64.hdf5)
- 학습 parameter(slide 8)로 학습

### 코드 수정

- 학습 dataset에 맞게 코드 수정하여 모델 학습
- 코드 실행 결과





UI 코드 수정 및 연동



#### Drawing Tool 가이드라인 (1)

사용자 입력에 따라 그려진 캐릭터가 변해가는 과정을 보고, 원하는 이미지로 고정 및 저장 할 수

있다.



#### Drawing Tool 가이드라인 (2)

사용자 입력에 따라 오른쪽 화면에 16개의 캐릭터가 생성되어서 나온다.

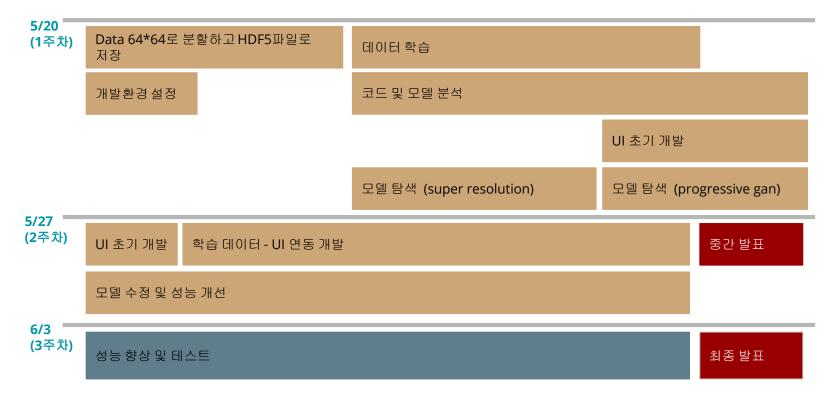


#### Drawing Tool 가이드라인 (3)

이미지 파일을 넣으면 이미지와 비슷한 캐릭터 생성되어 오른쪽 화면에 나온다.



차후 16개의 이미지 넣을 예정

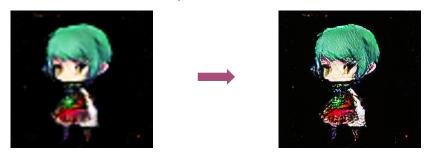


# 04 Project 변경사항

#### 변경 사항 및 이슈

### Super Resolution 포기

- 성능이 좋지 않음. Generated 이미지는 super\_resolution을 진행하면 깨짐 현상이 더 부각됨.



### **Progressive GAN**

- GPU 성능 문제(16G memory로 학습 -> 코드 수정시간 부족)

## 04 Project 변경사항 (TO DO)

#### 모델 성능 향상 방안

### 해상도 높이기

- 고해상도 캐릭터 dataset(1,034개)를 128x128로 resize
- 64x64 DCGAN(3layers)에서 128x128 DCGAN(4 layers)로 변경하여 학습
- Image Augmentation을 통해 40,000개의 데이터 추출
- batch size = 64 에서 batch size = 32로 변경
- momentum 0.5/0.8
- z-vector 100 -> 200
- 4개 조합 학습중 ....

## 04 Project 변경사항 (TO DO)

#### 개선 사항

### 다양한 캐릭터 생성

- 배경에 맞는 캐릭터 생성시
- AlexNet을 활용한 predicted Z-vector에 noise를 추가하여 다양한 캐릭터를 생성하게 조정
- [-1,1] 범위에서 uniform distribution을 따르는 noise vector 16개 생성
- predicted Z-vector에 noise vector weight = 1.5로 하여 16개의 변형된 캐릭터 생성































## 역할분담

고민주

- 논문 분석
- UI 코드 수정

오영진

- 논문 분석
- 코드 수정
- 모델 학습

오현지

- 논문 분석
- 코드 수정
- 모델 탐색

# Reference

#### **iGAN**

-(논문)Jun-Yan et."Generative Visual Manipulation on the Natural Image Manifold", in European Conference on Computer Vision (ECCV). 2016. (https://arxiv.org/pdf/1609.03552.pdf)

-(코드)https://github.com/junyanz/iGAN

-(Youtube 영상)https://www.youtube.com/watch?v=9c4z6YsBGQ0

#### **SR-GAN**

-(논문)https://arxiv.org/abs/1501.00092v3

-(코드)https://github.com/tegg89/SRCNN-Tensorflow

#### DC-GAN

-(논문)https://arxiv.org/abs/1511.0643

-(코드)https://github.com/SeitaroShinagawa/DCGAN-chainer

#### **Progressive-GAN**

-(논문)https://arxiv.org/abs/1710.10196

-(코드)https://github.com/tkarras/progressive\_growing\_of\_gans

#### Data

-http://yurudora.com/tkool

-http://machinelearninguru.com/deep\_learning/data\_preparation/hdf5/hdf5.html [HDF5 file 만들고 사용하기]

#### UI

-https://wiki.python.org/moin/PyQt/A%20 full%20 widget%20 waiting%20 indicator

-https://www.tutorialspoint.com/pyqt/pyqt\_layout\_management.htm

-http://pyqt.sourceforge.net/Docs/PyQt4/qstackedwidget.html

-http://nullege.com/codes/search?cq=PyQt4.QtGui.QMessageBox.button

# Reference

### **HOG(Histograms of Oriented Gradients)**

-https://blog.naver.com/jh\_h1022/220861693757

-(E是)https://lear.inrialpes.fr/pubs/2005/DT05/hog\_cvpr2005.pdf

