

# 머신러닝을 활용한 자동 채색 시스템 알고리즘 비교 분석

이송은<sup>1</sup>, 이지연<sup>2</sup>, 김나현<sup>3</sup>, 김진환<sup>4</sup>

<sup>1,2,3,4</sup> 동국대학교 정보통신공학과

juliegogo200@naver.com, cjstkdmdm@naver.com, nellykim577@naver.com, wlsghks3109@naver.com

## Comparison Analysis of Automatic Coloring System Algorithm Using Machine Learning

Lee song eun, Lee Ji Yeon, Kim Na Heon, Kim Jin Hwan

<sup>1,2,3,4</sup> Information and Communication Engineering, Dongguk University

### 요 약

현재 머신러닝(Machine Learning) 기술은 기존의 머신러닝과 조합 및 변형 되어 조금 더 발전 된 형태로 연구되어지고 있다. 따라서 수많은 알고리즘이 개발되고 있는 시점이다. 본 연구는 최근 좋은 결과로 관심을 받고있는 GAN(Generative Adversarial Net)을 중심으로 IT 기술의 머신러닝과 그림을 조합하여 자동채색을 목적으로 GAN 알고리즘을 비교하고 분석하고자 한다. GAN 알고리즘들 가운데서 'Conditional GAN'과 'Wasserstein GAN'을 사용하여 자동채색을 적용시켰고, 가장 부합한 알고리즘을 찾고 성능을 비교하여 어떠한 알고리즘이 '자동채색' 목적에 더 부합한지 비교하고 판단 한다.

### 1. 서 론

현재 IT 기술의 발달로 급격하게 새로운 기술들이 생겨나고 발전하고 있다. 그 중 알파고의 영향으로 많은 IT 기술의 중심이 머신러닝으로 대두되고 있다. 이러한 관심과 함께 새로운 알고리즘이 개발되고 있다. 다양한 알고리즘은 각각의 장단점을 가지고 있고, 어떤 주제에 활용할지에 따라 그 성능에도 차이가 있다. 본 논문에서는 자동 채색 시스템이라는 주제를 바탕으로 한다. 자동 채색 시스템을 Generative Adversarial Nets 의 두 종류인 Conditional GAN 과 Wasserstein GAN 에 적용시켜, 결과를 비교해보고, 다양한 각도에서 성능을 비교 및 분석 해본다.

본 논문의 구성은 다음과 같다. 2 장에서는 GAN 과

이에서 파생된 알고리즘인 Conditional GAN 과 Wasserstein GAN 의 원리를 설명하고, 각 알고리즘의 구성 설명을 한다. 이어서 3 장에서는 Convolutional Neural Network 의 원리를 설명한다. 4 장에서는 이 두 알고리즘을 사용하여 얻게 된 결과를 비교하고 5 장에서는 결과에 대한 이유를 분석하여 자동 채색이라는 목적에 가장 알맞은 알고리즘을 선택한다. 마지막으로 6 장에서는 결론을 맺는다.

### 2. GAN 알고리즘

#### 2.1 GAN(Generative Adversarial Net)이란

GAN 은 지도학습인 CNN(Convolution Neural Net)과는 달리 비지도 학습 방식을 따른다. 라벨링을 통해 데이터를 분류하여 학습하는 방식과 다르게

비지도 학습방식은 데이터 각각이 어떤 의미가 있는지 모른다. 하지만 데이터들 간의 상관 관계를 통해 군집화(Clustering) 또는 분포 추정(Underlying Probability Density Estimation)과정을 거쳐 학습하게 된다.

GAN(Generative Adversarial Nets)은 그림 1을 보면 크게 두 부분으로 나누어져 있다. 데이터를 만들어 내는 Generator와 이렇게 해서 만들어진 데이터를 분석하는 Discriminator가 서로 대립하면서 서로의 성능을 점차 개선해 나가는 개념이다. Generator는 input data의 distribution을 알아내려고 한다. 만약 Generator가 정확히 data distribution을 구현할 수 있다면 Generator가 내놓은 sample은 완벽히 data와 구별할 수 없다. 한편 Discriminator는 현재 자기가 분석하는 sample이 training data인 것인지 Generator로 부터 만들어 낸 sample인지 구별하여 각각의 경우에 대한 확률을 추정한다.

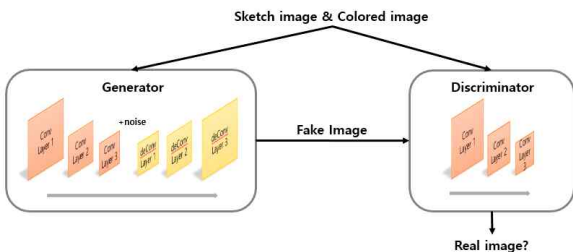


그림 1. GAN 구성도

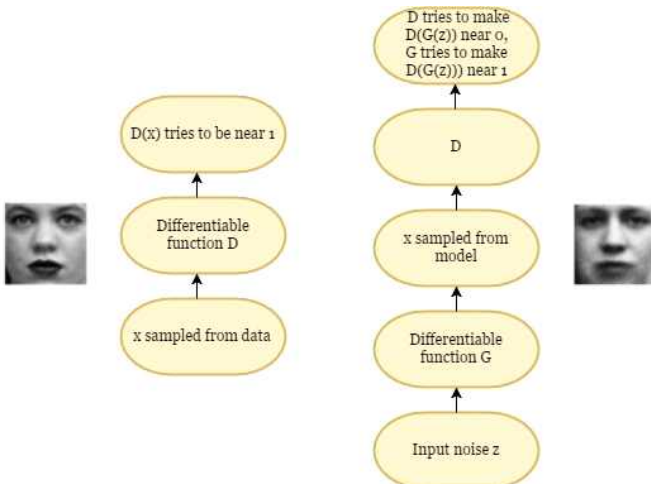


그림 2. GAN에서 G와 D 흐름도

real data에서 선택된 sample  $x$ 는  $D(x)=1$ 이 성립하고,

noise distribution으로 부터 생성된 noise  $z$ 를 sample  $x$ 에 적용하여 sample  $x$ 를 만들게 된다. Discriminator는  $D(Z(x))=0$ 이 되도록 노력한다.  $D(Z(x))=1$ 이 된다는 것은 fake 이미지가 원래의 데이터로 생각된다는 뜻이다. 종합적으로 보게 되면 D는 실수할 확률을 낮추기 위해 반대로 G는 D가 실수할 확률을 높이기 위해 노력한다. 이러한 과정을 반복하면서 학습을 진행하게 된다.

## 2.2 cGAN(Conditional GAN)이란?

cGAN(Conditional Generative adversarial net)은 데이터 군집화에 따른 분류 알고리즘이다. cGAN은 생성자 Generator를 통해서 데이터에 노이즈를 생성하고 판별자 Discriminator(X)를 통해 데이터의 분포 및 분류 알고리즘을 구현한다.

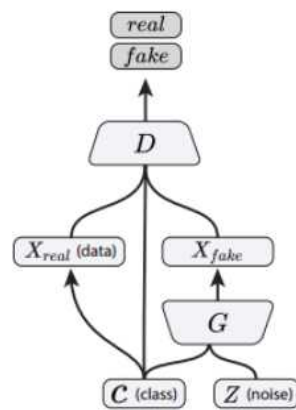


그림 3. cGAN 순서도

그리고 cGAN의 손실식은 아래와 같이 나타낼 수 있다.

$$\min_G \max_D V(D, G) = \mathbb{E}_{x \sim p_{data}(x)} [\log D(x, y)] + \mathbb{E}_{z \sim p_z(z)} [\log(1 - D(G(z, y), y))] \quad (1)$$

cGAN 은 Generator 와 Discriminator 로 구성되어 있다. Generator 는 convolution neural network 를 이용하여 이미지를 학습하고, deconvolution 하여 노이즈가 더해져 새로운 이미지를 생성하게 된다. Discriminator 또한 convolution neural network 를 이용하여 진짜 데이터를 학습하고 Generator 가 생성한 이미지를 실제 이미지와의 유사도를 비교하게 된다. 이때 손실도가 0 에 가까울 수록 유사도가 높다. 이때 Discriminator 의 값이 0 에 가까울 수록 손실도가 0 에 가깝다. Discriminator 의 값이 무한할수록 즉, 학습량이 많을 수록 손실의 정도가 크다. 따라서 적절한 데이터에 따른 학습량을 맞추는것이 cGAN 학습의 관건이 된다.

### 2.3 wGAN(Wasserstein GAN)이란?

wGAN(Wasserstein GAN)은 데이터 확률분포의 수렴값에 따른 데이터 분류 알고리즘이다. wGAN 에서는 Discriminator 가 아닌 Critic 으로 데이터 확률 분포 및 분류 알고리즘을 구현한다.

Critic 모델에서는 수렴값에 대한 정확한 측정 기준을 제시함으로써 학습 진행 상황을 유의미한 값으로 측정할 수 있으며 output 의 성능을 높일 수 있다. 아래의 식은 Wasserstein distance 를 나타낸다.

$$\begin{aligned}
 W(\mathbb{P}, \mathbb{Q}) &= \inf_{\gamma \in \Pi(\mathbb{P}, \mathbb{Q})} \int d(x, y) \gamma(dx dy) \\
 &= \inf_{\gamma \in \Pi(\mathbb{P}, \mathbb{Q})} \mathbb{E}^\gamma [d(X, Y)]
 \end{aligned}
 \tag{2}$$

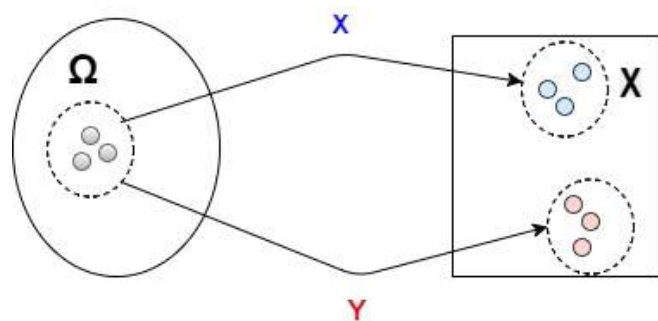


그림 4. 확률분포와 최적화 경로

wGAN 은 데이터의 확률분포 경계값을 계산한다. 이때 경계값에 극한값을 가지는 특정분포에 대해서 상한값과 하한값을 지정해 준다. 이를 지정해 줌으로써 모든 특정 분포 집합을 상한값과 하한값으로 구할 수 있기 때문에 분포집합을 유의미한 값으로 측정 가능하다.

서로 다른 두 특정분포에 대한 가중치는 최적화 경로로 계산된다. (가중치의 감소) 각분포에 대한 특정 위치는 지정된 데이터 확률분포에 따른 상한값과 하한값의 평균값인 sample 로 설정했다. 이때 최적화 알고리즘으로 RMSProp 을 사용하여 학습률을 분산시켜서 가중치의 수량에 대한 평균값을 구한다.

경로에 대한 가중치를 축소하게되면 sample 의 충돌문제를 개선하여서 cGAN 보다 더 안정적인 학습이 가능해진다. 다만, 다양한 데이터값들이 존재하기 때문에 최적화 경로로 Output 을 개선하려면 상당한 시간이 걸린다. 일반적으로 Discriminator 보다 Critic 은 더 많은 학습을 수반한다.

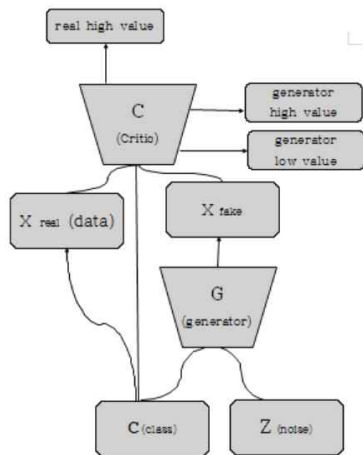


그림 5. WGAN 순서도

### 3. CNN(Convolutional Neural Network)

영상에서 특정위치에 있는 픽셀들은 그 주변에 있는 일부 픽셀들과 correlation 이 높고 거리가 멀어질수록 그 영향은 감소하게 된다. 이러한 특징을 이용하여 CNN에서는 이미지를 해석하여 인식알고리즘을 사용할 경우 이미지 전체픽셀을 같은 중요도로 처리하는 대신 특정 범위를 한정하여 처리하는 과정에서 효성성을 증가 시킨다. 또한 Local 정보를 활용하여 공간적으로 인접한 신호들에 대한 correlation 관계를 비선형 필터를 적용하여 추출하는 과정 거친다. 이때 필터를 여러 개 적용하게 되면 다양한 local 특징을 추출해 낼 수 있다.

Convolution layer에서는 convolution feature를 추출하는 layer로 이미지에서 의미있는 특징들을 추출한다. 이미지라는 데이터 타입의 특성상 많은 데이터를 요구하고 처리하게 되는데 이러한 feature를 줄이기 위해 subsampling 하는 과정을 거친다. 다음 convolution layer와 pooling layer에서 나온 feature들을 이용하여 분류하는 layer를 통해 마지막으로 학습하는 과정을 겪는다. CNN 처리

과정은단순하게 분류기로 구성 된 것이 아니라 특징을 추출하는 단계가 포함되어 있기 때문에, raw image에 대해 직접 operation이 가능하며, 기존 알고리즘과 달리 별도의 전처리 단계를 필요로하지 않는다. 특징 추출과 topology invariance를 얻기 위해 filter와 sub-sampling을 거치며 이 과정을 반복적으로 수행하여 local feature로부터 global를 얻어 낸다.

대부분의 영상 인식 알고리즘에서는 특징을 추출하기 위해 filter를 사용다. 보통 5 by 5 나 3 by 3 필터를 사용하는데 필터에 사용하는 계수들의 값에 따라 각각 다른 특징을 얻을 수 가 있다. 일반적으로 이 계수들은 특정 목적에 따라 고정되지만 CNN에서 사용하는 계수들은 학습을 통해 결정된다는 점이 다르다.

CNN에서는 max-pooling 방식의 sub-sampling 과정을 거친다. 간단히 설명하자면 각 window에서 가장 큰 자극만을 선택하는 것이다. 이 과정(convolution + sub-sampling) 과정을 여러 번 거치게 되면 이미지 전체를 대표할 수 있는 global한 특징을 얻을 수 있게 된다. 이렇게 얻은 특징을 학습시키면 topology 변화에 강인한 인식 능력을 갖게 된다.

### 4. 본 논문의 알고리즘

#### 4.1 구성

실험을 위한 알고리즘은 다음과 같이 구성된다. 첫째, 비교를 위한 두개의 알고리즘은 공통적으로 그림 1의 구성을 따르고 둘째, Generator의 세부적인 구성도는 표 1을 따른다. Discriminator는 Generator의 Layer

5 까지의 구성을 한다. 이 때, activation function 은 Generator 와 Discriminator 모두 sigmoid 를 사용한다.

Layer	Type	In_channels	out_channels	normalization	activation
1	Conv	256	128	BN	ReLU
2	Conv	128	64	BN	ReLU
3	Conv	64	32	BN	ReLU
4	Conv	32	16	BN	ReLU
5	Conv	16	8	BN	ReLU
6	Deconv	8	16	BN	ReLU
7	Deconv	16	32	BN	ReLU
8	Deconv	32	64	BN	ReLU
9	Deconv	64	128	BN	ReLU
10	Deconv	128	256	BN	ReLU

Conv.: 2D convolution, Deconv.: 2D convolution, BN: batch normalization  
Layer 6과 Layer 4, Layer 7과 Layer 3, Layer 8과 Layer 2, Layer 9와 Layer 1의 각각의 out channel은 Receptive Field를 적용한다.

표 1. convolution 구성도

이를 바탕으로 구성된 Wasserstein GAN 과 Conditional GAN 알고리즘을 구성하고, Wasserstein GAN 의 Critic 은 optimizer 를 RMSProp 를 사용하고, Conditional GAN 의 Discriminator 는 Adam 을 사용하여 학습 한다.

## 4.2 알고리즘 비교

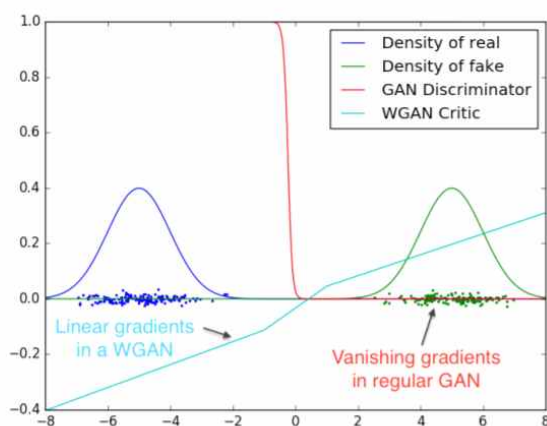


그림 7. Discriminator 와 Critic 의 비교

y 축은 각 알고리즘의 학습률을 나타내고 x 축은 fake 데이터와 real 데이터간의 손실도(+손실도가 큼)를 나타낸다. 위의 그래프에서 GAN 의 Discriminator 는

sigmoid 형을 띄고, wGAN 의 Critic 은 linear 형을 띈다.

Discriminator 는 초기부터 높은 학습률을 나타내고 Fake 데이터의 밀도가 높아질때 손실도가 증가하고 학습률이 급격하게 0 에 수렴한다. 즉, 손실이 있을 때 학습은 중단된다. Critic 또한 초기부터 낮은 학습률을 나타내고 Fake 데이터의 밀도가 높아질때 손실도가 증가하지만 Discriminator 와 달리 낮은 학습률은 선형적으로 향상된다.

즉, wGAN 의 학습율은 데이터 양에 따라 지속적으로 증가한다. 그러나 학습율이 Discriminator 만큼 높아지지 못한다. cGAN 은 적절한 Real 과 Fake 데이터를 상관시켜 학습하는것이 중요하며 wGAN 보다 빠르게 학습 시킬 수 있다.

## 5. 결과

### 5.1 채색된 그림에 따른 결과



## 그림 8. 결과 영상

(a) input, (b) cGAN output, (c) wGAN output

### 5.2 결과 분석

본 논문은 두가지 알고리즘에 따라 각각 같은 데이터셋을 학습 시킴으로 인해서 각기 다른 결과물을 얻었다. 학습을 통해 얻은 결과물은 그림 8 (a) 원본 사진을 학습시켜 (b)와 (c)의 결과를 얻었다.

cGAN의 결과인 (b)의 경우 원본에서 이미지 경계선을 정확히 인식하고 경계선 안을 채색 했다. 이미지의 번짐현상이 wGAN을 이용한 (c)의 결과보다 적다.

wGAN의 결과인 (c) 경우에는 색의 정확도는 높지만 경계선을 인식하는 정도에서 차이가 났다. 그림 8(c)에서 보이듯이 번짐현상이 일어 났다.

wGAN은 Critic 모델을 통해 안정적인 결과를 얻게 되지만 이 과정에서 많은 시간이 소요되고 학습 데이터가 추가되는 부분에서 Discriminator와의 연동 부분에 있어서 변수들을 잘못 다룰 경우 학습이 되지 않거나 결과가 나오지 않는 경우가 빈번하게 발생한다. 상대적으로 cGAN은 wGAN보다 시간이 절약되어 학습이 더 빠르게 구현된다. 또한, Critic 모델을 사용하지 않기 때문에 쉽게 알고리즘을 구현할 수 있다.

### 참고문헌

[1] Martin.A, Soumith.C, Leon.B, "Wasserstein GAN", Cornell University Library, arXiv:1701.07875v2, 2017.

[2] Yagmur.G, Umut.G, Rob.van, Marcel.A.J, "Convolutional Sketch Inversion", Cornell University Library, arXiv:1606.03073v19, 2016.

[3] Mehdi.M, Simon.O, "Conditional Generative Adversarial Nets", Cornell University Library, arXiv:1411.1784, 2014.

[4] Kvfrans, "Outline Colorization through Tandem Adversarial Networks", Cornell University Library, arXiv:1704.08834v1, 2017.

[3]

<https://www.slideshare.net/ssuser7e10e4/wasserstein-gan-i>  
[http://www.cs.toronto.edu/~tijmen/csc321/slides/lecture\\_slides\\_lec6.pdf](http://www.cs.toronto.edu/~tijmen/csc321/slides/lecture_slides_lec6.pdf)

(<http://wiseodd.github.io/techblog/2016/12/24/conditional-gan-tensorflow/>)에서 첨부

(<https://arxiv.org/pdf/1706.06918.pdf>)

#### [학술대회 논문작성시 유의사항]

##### 1. 논문 페이지 수

- 2 - 4 쪽 이내

##### 2. 용지 및 여백처리

- 용지: A4, 용지방향 - 세로  
- 여백: 위 쪽 30mm, 아래 쪽 20mm,  
왼 쪽 10mm, 오른 쪽 10mm

##### 3. 논문구성

- 아래 순서대로 작성하며,

① ~ ⑥항목은 1 단(column)

⑦ ~ ⑪항목은 2 단으로 구성

① 제목(국문)

② 저자명(국문) \* 발표자는 공동저자와 구분 처리

(예) 홍길동<sup>0</sup>

③ 소속(국문)

④ 저자 E-mail Address

⑤ 제목(영문)

⑥ 저자명(영문) \* 발표자는 공동저자와 구분처리

(예) Kil-Dong Hong<sup>0</sup>

⑦ 소속(영문)

⑧ 요약

⑨ 본문

- 장 및 절에 해당되는 번호는 아라비아 숫자 로 각각 1., 1.1 등과 같이 표기

- 그림의 명칭은 하단에, 표는 상단에 각각 그림 1 및 표 1 로 표기

⑩ 참고문헌

- 본문중에 참고문헌 번호를 쓰고, 그 문헌을 참고문헌란에 인용한 순서대로 기술.

- 기술 순서는 저자, 제목, 학술지명, 권, 호, 쪽수, 발행년도 순으로 작성.

⑪ 부록(해당사항이 있는 경우만 작성)

##### 4. 기타

- 논문작성폰트 임의사용 가능, 글자크기는 9pt 이상 사용, 논문파일은 PDF 로 제출 권장합니다.

- 샘플논문(KCC2012 발표):

<http://www.kiise.or.kr/conference02/data/sample2.pdf>

- 논문심사는 저자와 심사위원 상호 비공개로 진행됩니다. 따라서, 심사용(저자정보 삭제)과 출판용(저자정보 포함)으로 나눠 제출합니다. 심사용은 투고 시, 출판용은 심사 후 지정된 수정기간 중에 각 업로드 하시면 됩니다.