

이미지 생성을 위해 노이즈를 이용한 GAN 시스템

배상중¹ · 김민규¹ · 정회경^{2*}

GAN System Using Noise for Image Generation

Sangjung Bae¹ · Mingyu Kim¹ · Hoekyung Jung^{2*}

¹Graduate Student, Department of Computer Engineering, Paichai University, Daejeon, 35345 Korea

^{2*}Professor, Department of Computer Engineering, Paichai University, Daejeon, 35345 Korea

요 약

생성적 적대 신경망(GAN, Generative Adversarial Network)은 두 개의 신경망을 대립하여 이미지를 생성하는 방법이다. 이미지를 생성할 때 랜덤으로 생성한 노이즈를 재배열하여 이미지를 생성하는데 이러한 방법으로 생성된 이미지는 노이즈에 따라 생성이 잘 이루어지지 않고, 이미지의 픽셀이 적은 경우 제대로 된 이미지를 생성하기 어렵다는 문제점이 발생할 수 있다. 또한 데이터 분류에서 데이터가 쌓이는 속도와 크기가 증가되는데 이들을 라벨링하는 데는 많은 어려움이 있다. 본 논문에서는 이를 해결하기 위해 랜덤으로 생성하던 노이즈에 실제 데이터를 사용하여 노이즈를 생성하고 이를 기반으로 이미지를 생성하는 기법을 제안한다. 제안하는 시스템은 기존에 있는 이미지를 기반으로 하는 이미지를 생성하는 것이므로 좀 더 자연스러운 이미지의 생성이 가능하다는 것을 확인하였고 이를 학습에 이용할 경우 기존의 생성적 적대 신경망을 사용한 방법보다 더 높은 적중률을 보임을 확인하였다.

ABSTRACT

Generative adversarial networks are methods of generating images by opposing two neural networks. When generating the image, randomly generated noise is rearranged to generate the image. The image generated by this method is not generated well depending on the noise, and it is difficult to generate a proper image when the number of pixels of the image is small. In addition, the speed and size of data accumulation in data classification increases, and there are many difficulties in labeling them. In this paper, to solve this problem, we propose a technique to generate noise based on random noise using real data. Since the proposed system generates an image based on the existing image, it is confirmed that it is possible to generate a more natural image, and if it is used for learning, it shows a higher hit rate than the existing method using the hostile neural network respectively.

키워드 : 머신 러닝, 생성적 적대 신경망, 준 지도 학습, DCGAN, GAN

Keywords : Machine Learning, Generative Adversarial Networks, Semi-Supervised Learning, DCGAN, GAN

Received 16 March 2020, Revised 29 March 2020, Accepted 14 April 2020

* Corresponding Author Hoekyung Jung(E-mail:hkjung@pcu.ac.kr, Tel:+82-42-520-5640)

Professor, Department of Computer Engineering, Paichai University, Daejeon 35345, Korea

Open Access <http://doi.org/10.6109/jkiice.2020.24.6.700>

print ISSN: 2234-4772 online ISSN: 2288-4165

©This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License(<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.
Copyright © The Korea Institute of Information and Communication Engineering.

I. 서론

네트워크의 급속한 성장에 따라 생성되는 데이터 또한 기하급수적으로 증가하고 있지만 이를 학습에 사용하는데 많은 어려움이 존재하고 있다[1-3]. 그 중 하나가 라벨링의 어려움 때문이다. 라벨링은 사람이 진행하게 되는데 사람들은 서둘러 이를 처리하는 경우가 많으며 모든 사람이 같은 방식으로 이해하기에는 많은 어려움이 따른다. 이러한 문제 때문에 나온 점점 준 지도 학습의 중요성이 커질 것이라고 예상하고 있다[4].

생성적 적대 신경망은 이러한 준 지도 학습의 중 하나이며 주로 이미지를 생성하는 데 사용된다. 하지만 생성적 적대 신경망에서 가장 큰 주제가 되었던 것은 안정화에 있었다. MiniMax 또는 Saddle Problem을 풀어야 하는 생성적 적대 신경망은 이론적으로는 Fixed solution으로 수렴하지만 실제로는 가정이 깨지면서 불안정한 구조적 단점이 있다[5]. 이후 나오는 연구인 DCGAN은 대부분의 상황에서 안정되게 학습되며 특정 필터들이 이미지의 특성을 학습하고 이를 기반으로 이미지를 생성할 수 있었다는 점과 이후 나오는 논문들에 기반이 되었다는 점에서 많은 기여를 하고 있다[6].

그러나 생성적 적대 신경망은 기본적으로 노이즈를 기반으로 이미지를 생성하기 때문에 랜덤으로 생성된 노이즈에 따라 생성된 이미지의 품질이 달라지며 특히 적은 픽셀을 가진 이미지를 학습하는 경우 이미지를 잘 생성하지 못하며, 기존에 이미지를 학습시켜야 하기 때문에 일정량의 이미지가 존재하지 않는다면 학습이 힘들며, 합성을 하는 경우에도 정도를 지정할 수 없다는 문제점이 있다[7].

이를 해결하기 위해 본 논문에서는 랜덤으로 생성하였던 노이즈를 제어하여 이미지를 생성하는 기법을 제안한다. 실제 이미지를 기반으로 생성한 노이즈를 사용하여 생성되는 이미지의 품질을 보장하였고 노이즈를 통하여 생성된 이미지를 통해 학습을 진행하였을 경우 기존의 생성적 적대 신경망에 비해 높은 정확도를 보이는 것을 확인하였다.

II. 관련연구

2.1. 생성적 적대 신경망(GAN)

GAN(Generative Adversarial Networks)[8]은 2014년 이안 굿펠로우(Ian Goodfellow)가 저술한 논문이며 이를 기반으로 한 다양한 후속 연구가 진행되고 있다. GAN은 생성적 적대 신경망이라는 이름처럼 두 개의 신경망의 경쟁을 통해 학습하고 결과물을 만들어낸다. 두 신경망 모델은 ‘생성자(Generator)’와 ‘판별자(Discriminator)’로 불리며 생성자는 실제에 가까운 거짓 데이터를 생성하는 것이 목표로 판별자는 생성자가 생성한 거짓 데이터를 밝혀내는 것을 목적으로 한다. 식 (1)은 GAN에 대한 식을 나타낸다.

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

먼저 판별자가 실제 이미지에 대해 학습을 진행한다. 다음은 생성자가 노이즈를 이용하여 가짜 이미지를 생성하고 이 생성된 이미지에 대해 판별자를 통해 이미지가 거짓 이미지라고 판별하게 한다. 이 때, 생성자는 판별자가 거짓이라고 판별된 이유를 이용하여 모델을 변경한다. 이러한 방법을 반복 한다면 생성자는 판별자가 판별하지 못할 정도의 이미지를 생성해 낸다는 것이 GAN의 기본 이론이다.

하지만 두 개의 신경망이 MiniMax 문제를 제대로 해결하지 못하면서 생성자와 판별자의 균형이 깨지면서 불안정해지는 문제점이 있다.

2.2. DCGAN(Deep Convolution GAN)

DCGAN은 이러한 GAN의 문제점을 해결하기 위하여 GAN의 구조에 Convolutional 구조를 합친 구조를 가지고 있다[9].

먼저 미분이 되지 않는 부분을 convolution으로 대체하여 생성자와 판별자가 각각의 특징을 학습할 수 있게 해주었다. 이는 “안경을 쓰고 있는 남자들”에서 평균값을 구하고, “안경이 없는 남자”와 “안경이 없는 여자”를 만들어 내는 입력값에 대하여 평균값을 구한 후 각각을 빼고 더하면 “안경이 있는 여자”의 이미지를 만들 수 있는 역할을 하며 생성자가 이미지를 외워서 보여주는 것이 아니라는 것을 의미한다. 또 다른 특징으로 배치 정규화(Batch Normalization)를 사용하여 기존 GAN의 문

제점인 Mode Collapsing을 완화시켰다[10].

2.3. 문제점 및 해결 방안

생성자는 랜덤으로 생성한 노이즈를 재배치하여 이미지를 생성해낸다. 하지만 이 노이즈에 따라서 많은 문제가 발생하기도 한다. 그 중 하나가 위에서 설명한 Mode Collapsing이 발생하고 적은 픽셀을 가진 이미지를 학습하는 경우 이미지를 잘 생성하지 못하며, 합성을 하는 경우에도 정도를 지정할 수 없다는 문제점이 있다. 또한 기존에 이미지를 학습시켜야 하기 때문에 일정량의 이미지가 존재하지 않는다면 학습이 힘들다는 문제점이 있다.

이러한 문제점을 바탕으로 본 논문에서는 노이즈를 제어하는 생성적 적대 신경망을 제안한다. 노이즈는 실제 사진을 바탕으로 설정하되 어느 정도 이미지의 정확도에 따라 어느 정도 차이를 주며 이를 재배치하여 생성자가 이미지를 생성할 수 있도록 한다.

생성자와 판별자의 균형이 깨짐으로 생기는 Mode Collapsing 문제는 생성자와 판별자의 입력 값을 비슷하게 맞춰줌으로써 완화시키고 적은 픽셀의 이미지라도 기준이 되는 이미지가 존재하기 때문에 실제의 이미지와 비슷한 정도의 품질을 가질 수 있다.

노이즈 제어를 사용한 방법을 구현하고 그 결과를 확인한다. 노이즈 컨트롤을 사용한 경우와 사용하지 않은 경우를 비교하고 제안한 시스템의 성능을 검증한다.

III. 시스템 설계

3.1. 전체 시스템 설계

그림 1은 제안하는 시스템의 구조도를 나타낸 것이다. 먼저 입력 데이터는 Cifar-10을 사용하였다. Cifar-10 데이터 셋은 비행기, 자동차, 새, 고양이, 사슴, 개, 개구리, 말, 배, 트럭이라는 총 10개의 레이블로 이루어진 컬러 이미지이며 각각의 레이블마다 32*32 크기의 훈련 이미지 5000장 1000장의 테스트 이미지로 구성되어 있다. 이미지 전처리에는 이미지의 크기를 재설정하거나 이미지를 코드 값으로 변경하는 일을 진행한다. 노이즈 생성기는 입력 이미지를 기반으로 하는 노이즈를 생성해낸다. 네트워크에는 생성적 적대 신경망의 생성

자 네트워크와 판별자 네트워크가 있으며 그 이외에도 이미지를 구분하기 위한 신경망을 따로 구축하였다. 생성자 네트워크는 노이즈를 입력받아 이미지를 생성하는 일을 수행하고 판별자 네트워크는 생성자와 대립하면서 생성자가 생성한 이미지를 검증한다. 이미지 분류 네트워크는 일반적인 생성된 이미 판별 값을 주 출력 단계에서는 코드로 이루어진 이미지를 파일로 만들어 저장하는 역할을 진행한다.

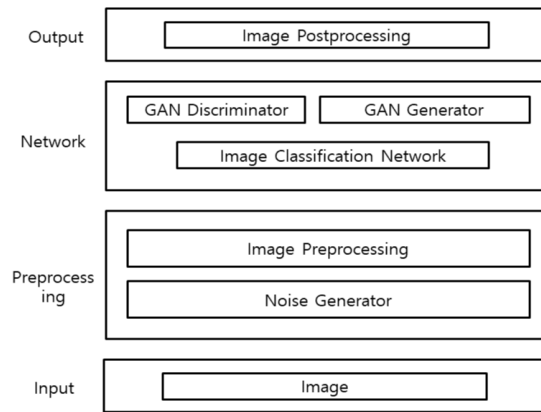


Fig. 1 Structure Diagram of the Proposed System

그림 2는 전체 시스템 흐름도이다. 먼저 이미지 전처리에서는 이미지를 실험 데이터와 검증 데이터로 나누고 이를 컴퓨터가 이해할 수 있도록 코드로 변경시킨다. 다음은 판별자를 학습시키고 이를 복사하여 이미지 구분기를 생성한다. 초기에는 판별기와 구분기 두 개는

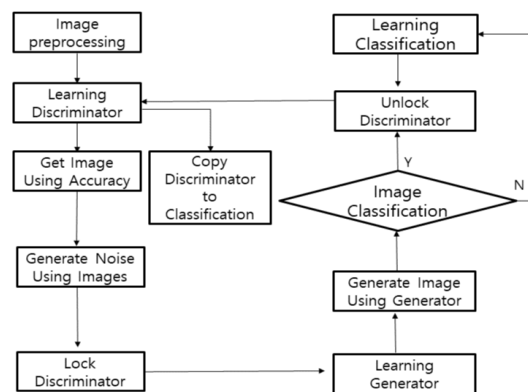


Fig. 2 System Flowchat

동일 데이터로 동일 구조로 진행하므로 복사를 해도 무관하다. 그러면서 학습하면서 각 이미지에 대한 분류 정확도를 측정하여 메모리에 저장해 둔다. 그 후 정확도와 노이즈를 생성하고 판별자는 생성자가 학습하는 동안에 영향을 미치지 않도록 잠금을 걸어둔다. 생성자 학습이 끝나면 생성된 노이즈와 학습이 끝난 생성자를 이용하여 새로운 이미지를 생성한다. 생성된 이미지가 배치 사이즈보다 적다면 기존의 생성적 적대 신경망처럼 판별자에 걸려둔 잠금을 해제하고 다시금 판별자를 학습한다.

3.2. 노이즈 제어를 사용한 생성적 적대 신경망 설계

그림 3은 노이즈 제어를 사용하여 생성적 적대 신경망의 흐름도이다. 주어진 레이블이 있는 이미지를 사용하여 GAN을 학습시킨다. 학습이 끝난 GAN의 판별자를 이용하여 레이블이 없는 이미지를 기반으로 하여 이미지를 생성한다. 생성된 이미지는 다시 판별자를 통해 이미지를 판별하고 이 이미지가 처음 판별된 결과와 같다면 이 이미지에 대해 GAN을 학습하고 다르다면 이미지를 생성하지 못했기 때문에 버리고 다음 에포크를 진행하여 새로운 이미지를 생성한다.

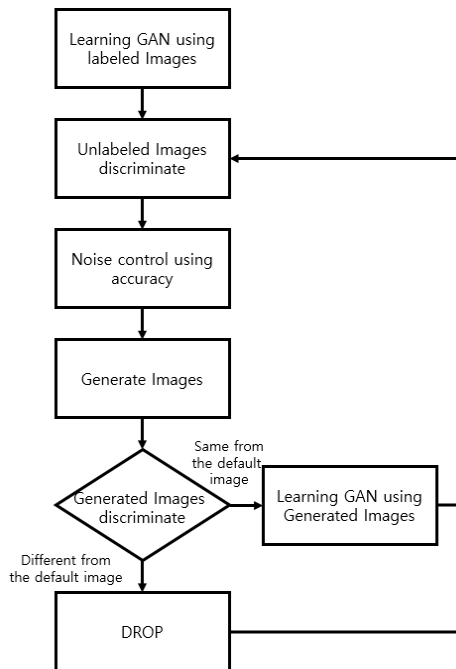


Fig. 3 GAN with Noise Control Flowchart

IV. 시스템 구현

해당 시스템을 구현하기 위해 프로세서는 i5-4670, 메모리는 16G, 그래픽 카드는 GeForce GTX 760가 장착된 컴퓨터에 Windows 10 Pro를 설치하여 사용하였다. 또한 빠른 학습을 위해 GPU를 사용하기 위하여 CUDA 10.0을 설치하였으며 Jupyter Notebook에서 Python3.5를 사용하여 시스템을 구현하였다.

4.1. 노이즈 제어와 생산적 적대 신경망 구현

그림 4는 노이즈 컨트롤을 사용했을 때 기본이 된 이미지와 그 이미지를 기반으로 노이즈를 구현했을 때 나오는 이미지이다. 기본 생성적 적대 신경망을 사용하여 생성된 이미지는 알아보기 힘든 이미지가 대부분이다. 그러나 이에 비해 노이즈 컨트롤을 사용하여 생성된 이미지는 기존의 방식을 사용한 것보다 더 실제 이미지와 비슷하게 생성된 것을 볼 수 있다. 그러나, 그림에서 기본이 된 이미지와 생성된 이미지를 묶었을 경우 1행 3열, 2행 2열, 4행 3열의 이미지는 판별자가 개구리, 말, 비행기로 인식하였고 이 상태에서 생성자가 학습을 진행하여 이미지를 생성하였기에 알기 힘들지만 그 이외의 이미지는 알아볼 수 있을 정도의 이미지가 나온 것을 볼 수 있으며, 직관적으로 보았을 경우 노이즈 제어를 사용하여 생성된 이미지가 더 현실과 가깝게 생성된 것으로 볼 수 있다.



Fig. 4 Generated Image Using Noise Control

4.2. 노이즈 제어를 사용한 시스템 구현

그림 5는 학습과정을 나타낸다. 학습은 클래스별 이미지 1000장씩 넣어서 총 10000장을 사용하여 진행하

였다. 한 번의 반복이 진행될 때 마다 판별자 학습, 노이즈 생성, 생성자 학습을 진행하고 생성자는 다시 각 클래스별 1000장의 이미지를 생성해 내어 구분기를 통하여 생성이 제대로 이루어진 이미지만을 저장한다. 구분기는 클래스별 1000장의 이미지가 생성되면 구분기를 학습시키고 다음 반복으로 넘어간다.

```

Dis learning end
1000/1000 [=====] - 21s 2ms/step
noise create
gen learning end
Epoch 1/10
10000/10000 [=====] - 11s 1ms/step - loss: 2.9003 - acc: 0.2646
Epoch 2/10
10000/10000 [=====] - 13s 1ms/step - loss: 1.6444 - acc: 0.4158
Epoch 3/10
10000/10000 [=====] - 12s 1ms/step - loss: 1.4741 - acc: 0.4777
Epoch 4/10
10000/10000 [=====] - 10s 997us/step - loss: 1.3714 - acc: 0.5166
Epoch 5/10
10000/10000 [=====] - 10s 1ms/step - loss: 1.2829 - acc: 0.5564
Epoch 6/10
10000/10000 [=====] - 10s 999us/step - loss: 1.1845 - acc: 0.5798
Epoch 7/10
10000/10000 [=====] - 28s 3ms/step - loss: 1.0940 - acc: 0.6175
Epoch 8/10
10000/10000 [=====] - 27s 3ms/step - loss: 1.0150 - acc: 0.6446
Epoch 9/10
10000/10000 [=====] - 10s 1ms/step - loss: 0.9649 - acc: 0.6703
Epoch 10/10
10000/10000 [=====] - 10s 997us/step - loss: 0.8606 - acc: 0.7057
10000/10000 [=====] - 3s 283us/step
    
```

Fig. 5 Learning Process

그림 6은 노이즈 제어없이 생성적 적대 신경망으로 생성한 이미지와 노이즈 제어를 사용하여 생성된 이미지를 사용하여 학습하였을 때 10번의 에포크 당 평균 정확도를 나타낸 것이다. 기존의 생성적 적대 신경망으로 생성한 이미지를 학습시켰을 경우 약 35%, 노이즈 컨트롤을 사용하여 생성된 이미지를 학습시켰을 경우 약 49%의 정확도를 보인 것을 알 수 있다. 두 가지의 방법을 비교해본다면 14%이상의 차이를 보이는 것을 알 수 있으며 이는 제안하는 방법을 학습 데이터가 적은 경우에 사용한다면 데이터의 양을 늘리기 위해 사용하였던 기존의 생성적 적대 신경망보다 더 좋은 성능을 보였다

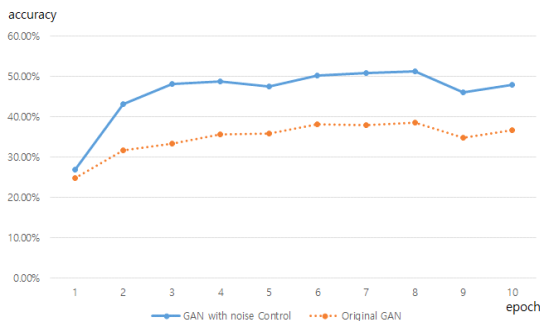


Fig. 6 Classification Accuracy Using Generated Image

는 점을 알 수 있다. 이는 제안한 시스템이 생성된 이미지의 기초가 실제 이미지에서 기반이 되었고 이를 사용하여 생성된 이미지의 품질이 기존의 방식보다 뛰어난 것으로 판단된다.

V. 결론

생성적 적대 신경망은 준 지도 학습의 대표적인 기술 중 하나이다. 그러나 구조로부터 기인하는 노이즈로 인해 물체들이 마치 흔들리게 보이거나 생성자와 판별자의 불균형으로 인하여 생성되는 이미지의 품질이 현저히 낮은 문제가 존재한다.

이를 해결하기 위해 본 논문에서는 기존의 생성적 적대 신경망에 노이즈를 제어하는 기법을 제안하였다. 노이즈를 제어함으로써의 이점은 생성된 이미지의 품질을 유지할 수 있다는 점과, 생성된 이미지를 다시 학습에 사용하여도 정확도가 상승한다는 이점을 얻을 수 있었다. 구현에서 보았듯이 노이즈 제어를 사용한 경우가 사용하지 않은 경우보다 더 품질이 좋은 이미지를 생성하였고 이를 다시 학습에 이용하더라도 기존의 시스템보다 약 14%의 성능 향상시킬 수 있다는 결론을 내릴 수 있었다. 이는 영상 합성이나 영상 복원, 영상 교체에서 주로 사용하였던 기존의 생성적 적대 신경망을 노이즈 컨트롤을 추가함으로써 데이터 양이 적은 경우, 특히 데이터의 생성되는 데이터의 양은 많으나 이를 전부 라벨링하기 힘든 최근의 경우 제안한 방법을 사용하는 것으로 라벨링은 물론이고 이를 학습에 사용할 수 있다는 점에서 많은 이점을 볼 수 있을 것으로 사료된다.

향후 연구는 라벨링의 정확도를 높이는 방안에 대해 연구 및 다른 종류의 GAN을 활용한 연구를 지속해야 할 것이다.

ACKNOWLEDGEMENT

This work was supported by the research grant of Pai Chai University in 2020.

REFERENCES

- [1] K. T. Kim, W. Lee, E. U. Cha, M. Y. Sin, and J. U. Kim, "The way to make training data for deep learning model to recognize keywords in product catalog image at E-commerce," *Journal of Korea Intelligent Information Systems Society*, vol. 24, 1-23, 2018.
- [2] I. H. Lee, "Outage Analysis and Power Allocation for Distributed Space-Time Coding-Based Cooperative Systems over Rayleigh Fading Channels," *Journal of Information and Communication Convergence Engineering*, vol. 15, no. 1, pp. 21-27, Mar. 2017.
- [3] A. Suryadibrata, and K.B. Kim, "Ganglion Cyst Region Extraction from Ultrasound Images Using Possibilistic C-Means Clustering Method," *Journal of Information and Communication Convergence Engineering*, vol. 15, no. 1, pp. 49-52, Mar. 2017.
- [4] D. Kim, J. G. Joung, K.A.Sohn, and H. Shin, "Knowledge boosting:A graph-based integration approach with multi-omics data and genomic knowledge for cancer clinical outcome prediction," *Journal of the American Medical Informatics Association*, vol. 22, no. 1, pp.109-120, 2015.
- [5] Y. J. Jeon and Y. W. Cho, "An Implementation of Othello Game Player Using ANN based Records Learning and Minimax Search Algorithm," *Journal of the Korean Institute of Electrical Engineers*, vol. 67, no. 12, pp. 1657-1664, 2018.
- [6] A. Radford, L. Metz, and S. Chintala, "Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks," *Cornell University*, 2016.
- [7] Y. Han and H. J. Kim, "Face Morphing Using Generative Adversarial Networks," *Journal of Digital Contents Society*, vol. 19, no. 3, pp. 435-443, 2018.
- [8] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, and D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, "Generative Adversarial Networks," *Cornell University*, 2016.
- [9] M. S. Ko, H. K. Roh, and K. H. Lee, "GANMOOK: Generative adversarial network to stylize images like ink wash painting," in *Proceedings of the Korea Computer Congress*, pp. 793-795, 2017.
- [10] L. C. Yang, S. Y. Chou, Y. H. Yang, "MidiNet:A convolutional generative adversarial network for symbolic domain music generation," in *Proceeding sof the 18th International Society of Music Information Retrieval Conference*, pp. 324-331, 2018.



배상중(Sangjung Bae)

2018년 배재대학교 컴퓨터공학과(공학사)
2020년 배재대학교 컴퓨터공학과(공학석사)
※관심분야 : IoT, Machine Learning



김민규(Mingyu Kim)

2013년 ~ 2017년 배재대학교 컴퓨터공학과(공학사)
2018년 ~ 2020년 배재대학교 컴퓨터공학과(공학석사)
2020년 ~ 배재대학교 컴퓨터공학과 재학 중(공학박사)
※관심분야 : BigData, Embedded System



정회경(Hoekyung Jung)

1985년 광운대학교 컴퓨터공학과(공학사)
1987년 광운대학교 컴퓨터공학과(공학석사)
1993년 광운대학교 컴퓨터공학과(공학박사)
1994년 ~ 현재 배재대학교 컴퓨터공학과 교수
※관심분야 : 멀티미디어 문서정보처리, USN, IoT, Machine Learning, BigData, Embedded System