Journal of Korea Multimedia Society Vol. 22, No. 1, January 2019(pp. 44-54)
https://doi.org/10.9717/kmms.2019.22.1.044

CycleGAN을 이용한 야간 상황 물체 검출 알고리즘

조상흠[†+], 이용[††+], 나재민[†††], 김영빈[††††], 박민우[†††††], 이상환[††††††], 황원준[†††††††]

CycleGAN-based Object Detection under Night Environments

Sangheum Cho[†+], Ryong Lee[††+], Jaemin Na[†††], Youngbin Kim[††††],
Minwoo Park[†††††], Sanghwan Lee[††††††], Wonjun Hwang[†††††††]

ABSTRACT

Recently, image-based object detection has made great progress with the introduction of Convolutional Neural Network (CNN). Many trials such as Region-based CNN, Fast R-CNN, and Faster R-CNN, have been proposed for achieving better performance in object detection. YOLO has showed the best performance under consideration of both accuracy and computational complexity. However, these data-driven detection methods including YOLO have the fundamental problem is that they can not guarantee the good performance without a large number of training database. In this paper, we propose a data sampling method using CycleGAN to solve this problem, which can convert styles while retaining the characteristics of a given input image. We will generate the insufficient data samples for training more robust object detection without efforts of collecting more database. We make extensive experimental results using the day-time and night-time road images and we validate the proposed method can improve the object detection accuracy of the night-time without training night-time object databases, because we converts the day-time training images into the synthesized night-time images and we train the detection model with the real day-time images and the synthesized night-time images.

Key words: CycleGAN, Data Sampling, Image-to-Image Translation

1. 서 론

최근 컴퓨터 비전 분야는 Convolutional Neural Network (CNN)의 도입으로 인해 매우 큰 진보를 이루었다. 이로 인해 CNN을 사용한 방식이 점차 주류가 되어가고 동시에 기존의 연구에서 사용되어 왔던 hand-crafted feature는 CNN을 사용하여 추출한 특징들로 대체되었다. 이처럼 CNN을 사용한 방식이 주류가 될 수 있었던 이유는 크게 2가지로 들 수 있는데, 1) Deep Convolutional Neural Networks (Deep CNNs) 와 2) 충분한 양의 라벨링된 데이터이다. 위의 2가지 조건이 전제가 된 경우 CNN은 매우 효과적

※ Corresponding Author : Wonjun Hwang, Address: (16499) Ajou Univ., Worldcupro 206, Yeongtong-gu, Suwon, Korea, TEL : +82-31-219-2632, FAX : +82-31-219-1621, E-mail : wjhwang@ajou.ac.kr
(+: Equal contribution)
Receipt date : Nov. 2, 2018, Revision date : Dec. 19, 2018
Approval date : Dec. 22, 2018
† Dept. of Software and Computer Engineering, Ajou University (E-mail : chothird@ajou.ac.kr)
†† Korea Institute of Science and Technology Information (E-mail : ryonglee@kisti.re.kr)
††† Dept. of Software and Computer Engineering, Ajou University (E-mail : osial46@ajou.ac.kr)
†††† Dept. of Software and Computer Engineering, Ajou University (E-mail : dudqls1994@ajou.ac.kr)
††††† Korea Institute of Science and Technology Information (E-mail : pminwoo@kisti.re.kr)
†††††† Korea Institute of Science and Technology Information (E-mail : sanglee@kisti.re.kr)
††††††† Dept. of Software and Computer Engineering, Ajou University
※ This research was supported by a project 'Establishing a System for Sharing and Disseminating Research Data(K-18-L11-C03)' of Korea Institute of Science and Technology(KISTI), Korea.

인 특징 추출 수단이 될 수 있다. 하지만 충분한 양의 라벨링된 데이터의 확보는 어플리케이션 측면에서 매우 번거롭고 비용이 드는 과정이다. 따라서 해당 도메인에서 학습 데이터가 부족한 경우 이러한 문제를 해결하기 위한 여러 방법이 제시되었다. 그중 하나가 Image-to-Image translation [1, 2, 4, 13, 14, 15]를 이용한 학습 데이터 생성이다. 본 논문에서는 Image-to-Image translation [1, 2, 4, 13, 14, 15] 네트워크를 이용하여 데이터 샘플을 확보하고 객체 검출 네트워크를 통해 이 방식의 효용성을 입증하고자 한다.

Image-to-Image translation [1, 2, 4, 13, 14, 15]는 주어진 이미지를 다른 이미지로 변환하는 분야를 가리킨다. 이 분야는 시작 도메인의 이미지에서 해당 이미지의 주요 특징은 유지한 상태로 목표 도메인의 이미지로 변환하는 것을 목표로 한다. 예를 들어 반고흐의 작품을 모네의 화풍으로 변환시키거나 [1], 주어진 인물 사진의 화난 표정을 웃는 표정으로 바꾸어 다시 생성하는 작업 [2] 등이 있다. 이러한 분야는 Generative Adversarial Network (GAN) [3]의 도입과 더불어 활발한 연구가 진행되었으며 그 결과 segmentation map 혹은 edge map에서 사진을 다시 생성하거나 [4], 어느 특정 계절대의 이미지를 또 다른 계절대의 이미지로 변환 하거나 [1], 해상도 개선을 위해서 사용 [21] 되는 등 여러 응용 범위에서의 연구가 진행되었다.

그리고 객체 검출은 주어진 이미지 내에서 원하는 물체가 어느 위치에 있는지 특정하는 분야를 가리킨다. 최근 몇 년간 CNN을 사용한 방식이 컴퓨터 비전 분야에서 큰 성공을 거두면서, 이에 따라 객체 검출 기들의 성능도 크게 향상되었다. 이후 성능이 어느 정도 안정되면서 detection 속도를 증가시키는 것에 초점이 맞춰지게 되는데, 대표적인 네트워크로는 Single Shot multi-box Detector (SSD) [5]와 YOLO (You Only Look Once) [6]가 있다. SSD는 입력 이미지에서 추출해 내는 특징 맵 (feature map)을 여러 개의 크기로 만들어서 큰 특징 맵에서는 작은 물체의 검출을, 작은 특징 맵에서는 큰 물체의 검출을 하게 만든 방식의 네트워크이고, YOLO는 전체 특징 맵을 격자로 나누어 각각의 격자에 대한 결과를 얻어내는 방식의 네트워크이다.

정확도와 detection 속도에 있어 큰 발전을 이루었지만 여전히 해결해야할 문제들은 남아있는데, 그 중 하나가 야간 영상에서의 검출 문제이다. YOLO [6]의 경우 야간 영상에서 detection이 제대로 이루어지지 않는 문제가 있다. YOLO를 학습시키기 위해 주로 사용하는 dataset으로는 PASCAL dataset [7]과 COCO dataset [8]이 있는데, 대부분의 이미지가 낮 시간대의 이미지이기 때문에 이를 이용하여 학습한 네트워크는 낮 시간대의 물체에만 반응하게 된다. 이러한 원인 때문에 기존의 학습된 네트워크로는 밤 시간대의 물체에 한하여 검출 성능이 떨어질 수밖에 없다. 밤 시간대의 물체도 검출하기 위해서는 네트워크를 학습시킬 때 학습 데이터에 충분한 양의 밤 시간대의 이미지를 추가해야 하며, 추가된 이미지 각각에 라벨 정보가 추가되어야 한다.

네트워크 학습에 있어서 데이터의 부족 문제를 해결하기 위해, 본 논문에서는 2개의 생성자 (Generator)와 구별자 (Discriminator)를 기반으로 하는 Cycle-Consistent Adversarial Networks (CycleGAN) [1]을 사용하여 데이터를 샘플링하는 기법을 제안한다. Berkeley Deep Drive (BDD) dataset [9]의 낮 시간대 이미지와 밤 시간대 이미지를 사용하여 CycleGAN [1]을 학습시킨 후 이렇게 학습된 CycleGAN [1]을 이용하여 라벨이 있는 낮 시간대의 이미지를 밤 시간대의 이미지로 변환한다. 이렇게 생성된 밤 시간대의 이미지를 학습 dataset에 추가해 YOLO [6]를 다시 학습시켜 야간 영상에서 검출 성능이 떨어지는 문제를 해결하고자 한다. 실험을 통해 Image-to-Image Translation [1, 2, 4, 13, 14, 15]를 이용하여 생성된 이미지를 네트워크의 학습 데이터로써 사용할 수 있음을 입증한다.

본 논문의 구성은 다음과 같다. 2장에서 본 논문의 기본 이론이 되는 GAN [3]과 이를 이용한 task (Image-to-Image Translation [1, 2, 4, 13, 14, 15], Neural Style Transfer [16, 17])에 대하여 정의한 다음, 3장에서 제안 방법과 그 베이스라인이 되는 CycleGAN [1]의 특성에 대해 설명한다. 그리고 4장에서는 제안한 방법으로 생성한 데이터를 이용한 실험 및 그 결과에 관해 설명, 마지막으로 5장에서 결론을 맺는다.

2. 관련 연구

2.1 Generative Adversarial Network (GAN) [3]

Adversarial Network (GAN) [3]은 generative

model을 학습하는 새로운 방법으로 제시되었다. GAN [3]은 2개의 적대적인 네트워크로 구성되어 있다. Generative 네트워크 G는 기존의 데이터 분포를 학습하여 이와 최대한 유사한 데이터를 생성해내는 것이 목적인 반면 구별자 네트워크 D는 입력으로 주어진 데이터가 G가 생성한 것인지 실제 데이터 분포에서 추출한 것인지 구분해내는 것이 목적이다.

실제 데이터 분포를 학습하기 위해, 생성자 G는 random noise 분포에서 실제 데이터 분포로 이어지는 매핑 함수 (mapping function)를 다층 퍼셉트론 (Multi-Layer Perceptron)의 형태로 구축한다. 반면, 구별자 D는 주어진 입력을 받아 해당 입력이 실제 데이터에서 왔을 확률을 결과값으로 내놓는다. G와 D는 동시에 학습되며, 입력으로 사용되는 random noise를 z라고 했을 경우 G는 z로부터 생성한 결과물이 구별자에서 실제 데이터로 판별될 확률을 최대한으로 늘리는 방향으로, D는 실제 데이터와 생성된 데이터의 판별 성능 자체를 높이는 방향으로 학습하게 된다. 즉, 생성자 G로부터 생성된 결과가 실제 데이터로 판별될 확률을 최소한으로 줄이는 방향으로 학습되는 것이다. 이것이 동시에 이루어지면 생성자와 구별자 두 플레이어의 mini-max game의 형태가 되는데 이를 식으로 나타내면 식 (1)과 같다.

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

그러나 GAN [3]은 생성된 영상의 떨어지는 완성도와 모델 자체의 학습 시 안정성 등 여러 문제를 안고 있었다. 완성도의 경우 MNIST와 같이 비교적 단순하면서도 작은 해상도를 지닌 이미지는 언뜻 보기에 자연스러운 이미지를 생성할 수 있었지만, CIFAR-10과 같이 보다 복잡하고 상대적으로 큰 해상도의 이미지에 대해서는 자연스러운 이미지를 생성하는데 어려움을 겪었다. LAPGAN [10]의 경우 GAN과 Laplacian Pyramid의 구조를 혼합하여 기존 GAN [3]보다는 상대적으로 높은 해상도에서도 그럴듯한 이미지를 생성할 수 있었지만, Laplacian Pyramid 구조로부터 기인하는 에러로 인해 이미지 내 물체들이 흔들린 것처럼 생성하는 문제가 여전히 남아있었다. 이러한 문제를 해결하기 위해 Deep Convolutional Generative Adversarial Network (DCGAN) [11]에서는 대부분의 상황에서 안정적으로 학습이 되는 새로운 GAN 네트워크를 제안하고 수많은 실험을 통해 이를 입증하였다.

Conditional GAN (cGAN) [12]에서는 vanilla GAN에 조건부 확률의 개념을 도입하여 원하는 데이터를 생성할 수 있도록 네트워크를 개량한다. 입력으로 사용하는 random noise z에 부가 조건 y를 추가하여 학습시킴으로써 y를 통해 생성되는 결과물을 조정할 수 있도록 하였다. 입력이 변화되면서 생성자와 구별자 사이의 목적 함수 V에도 변화가 생기게 되는데 변화된 식은 식 (2)와 같이 나온다.

$$\min_G \max_D V(D,G) = E_{x\sim P_{data}(x)}[logD(x\,|\,y)] + E_{z\sim P_{data}(z)}[log\{1 - D(G(z\,|\,y))\}] \quad (2)$$

2.2 Image-to-Image Translation [1, 2, 4, 13, 14, 15]

GAN의 도입과 이와 관련된 후속 연구들이 나오면서 Image-to-Image Translation [1, 2, 4, 13, 14, 15] 분야는 큰 발전을 이루었다. 비지도 학습에서의 학습 성능이 지도 학습보다는 떨어질 수밖에 없다는 사실은 GAN의 학습 측면에서도 마찬가지였기 때문에 Pix2Pix [4]는 cGAN [12]의 개념을 이용하여 Image-to-Image Translation 문제를 Supervised Learning의 측면에서 해결하고자 하였다. Pix2Pix [4]에서는 loss function으로 기존의 GAN loss인 adversarial loss 이외에도 이미지 사이의 L1 loss를 조합해서 사용하였다. 이를 통해 생성된 이미지가 보다 실제 이미지와 가깝게 생성될 수 있도록 유도하였다. 그러나 이 경우 시작 도메인에 해당하는 각 이미지가 라벨의 역할을 했기 때문에 목표 도메인과 시작 도메인은 서로 짝에 해당하는 이미지를 가지고 있어야 한다는 한계가 있었다.

이러한 문제점을 해결하기 위해 데이터가 서로 짝을 이루지 않는 도메인 사이의 이미지 변환인 Unpaired image-to-image translation [1, 2, 13, 14, 15]를 위한 네트워크들이 제안되었다. 그 중 Coupled GAN (CoGAN) [13]은 weight를 공유하는 2개의 생성자를 사용함으로써 짝을 이루지 않는 시작 도메인과 목표 도메인의 공통된 분포를 학습하고자 시도하였으며, Unsupervised Image-to-Image Translation Networks (UNIT) [14]에서는 Variational Auto-Encoder (VAEs)를 CoGAN [13]과 조합하여 시작 도메인과 목표 도메인에서의 각 데이터가 같은 noise 분포의 데이터로 변환되도록 하는 제약조건을 CoGAN [13]에 추가하였다. CycleGAN [1]과 Learning

to Discover Cross-Domain Relations with Generative Adversarial Networks (DiscoGAN) [15]의 경우 cycle consistency의 개념을 적용하여 입력 이미지와 변환된 이미지 사이의 공통되는 주요 특징을 보존하고자 했다.

2.3 Neural Style Transfer [16, 17]

Neural Style Transfer [16, 17]는 image-to-image translation을 수행할 수 있는 또 다른 방법이다. 서로 다른 도메인 사이를 연결해주는 매핑 함수, 즉 생성자를 학습시킨다는 측면에서는 image-to-image translation에서의 작업과 공통점을 가진다. 그러나 특정 noise 분포에서 임의의 noise를 추출하여 이를 기반으로 이미지를 생성하는 방식이 아닌, 이미지를 기반으로 해당 이미지의 주요 특징과 다른 도메인에서의 스타일을 합성하는 방식을 사용한다.

3. 제안 방법

3.1 CycleGAN [1] 개요

이미지 변환을 통한 데이터 생성을 위한 베이스라인 모델로서, 이 논문에서는 CycleGAN [1]을 채택하였다. CycleGAN [1]의 학습 목적은 2개의 도메인 사이를 오갈 수 있는 2개의 생성자와 2개의 구별자를 학습시키는 것이다. 2개의 도메인을 각각 X, Y라 하고 X => Y 방향으로의 생성자를 F, Y => X 방향으로의 생성자를 G라 했을 때 X 도메인의 구별자는 X 도메인의 데이터 x와 Y 도메인에서 G를 통해 변환된 $G(y)$를 보다 잘 판별하는 것을 목표로 한다. 반대로 Y 도메인의 구별자는 Y 도메인의 데이터 y와 X 도메인에서 F를 통해 변환된 $F(x)$를 보다 잘 판별하는 것을 목표로 한다. 이를 식으로 나타내면 식 (3), (4)와 같다.

$$\min_G \max_D V(D_X, G) = E_{x \sim P_{data}(x)}[log\, D_X(x)] + E_{y \sim P_{data}(y)}[log\,\{1 - D_X(G(y))\}] \quad (3)$$

$$\min_G \max_D V(D_Y, F) = E_{y \sim P_{data}(y)}[log\, D_Y(y)] + E_{x \sim P_{data}(x)}[log\,\{1 - D_Y(F(x))\}] \quad (4)$$

이 Adversarial loss 이외에도 Cycle Consistency Loss라는 개념을 도입하여 2개의 도메인의 데이터가 불균형한 경우에도 학습할 수 있도록 만들었다. 단순히 한 방향으로의 mapping만이 아니라 변환되었다가 다시 복원되는 mapping까지 고려하여, 원래의 도메인으로 돌아왔을 때 제대로 복원되었는지 확인하는 것이다. Cycle Consistency Loss를 식으로 표현하면 식 (5)와 같다.

$$L_{Cycle} = E_{x \sim P_{data}(x)}[||F(G(x) - x||_1] + E_{y \sim P_{data}(y)}[||G(F(y) - y||_1] \quad (5)$$

3.2 제안 방법의 개요

두 도메인 사이의 데이터가 짝을 이루지 않는, 즉 불균형한 도메인 사이에서도 제대로 학습이 가능한 CycleGAN [1]의 특성을 이용한다. 이 제안 방법의 목적은 CycleGAN [1]을 이용하여 특정 도메인의 데이터를 생성, 이를 이후 classification 혹은 detection 네트워크에 사용하는 것을 목적으로 한다. 식 (3), (4), (5)에서 도메인 X를 데이터가 상대적으로 많은 dataset 혹은 변환하기를 원하는 시작 도메인에 대응시킨다. 그리고 반대로 도메인 Y를 데이터가 상대적으로 적은 dataset 혹은 변환되기를 원하는 목표 도메인에 대응시킨다. 이 경우 3.1에서 설명한 2개의 생성자 F, G를 각각 GD->N, GN->D로 표기한다. 본 논문에서 제안하는 전체적인 CycleGAN [1]의 학습 흐름은 Fig. 1과 같다. 이 논문에서 사용한 CycleGAN [1]은 Resnet [18] 9-block 기반의 생성자와 구별자을 한 쌍으로 하여 총 2쌍의 네트워크로 구성되었고, Activation Function은 ReLU, Optimizer는 Adam [19]을 사용하였다. 이후 학습시킨 CycleGAN [1]에서 GD->N을 떼어내어 Y 도메인의 데이터를 생성한다. 이 데이터는 X 도메인의 데이터로부터 생성한 것이므로 X 도메인의 annotation (bounding

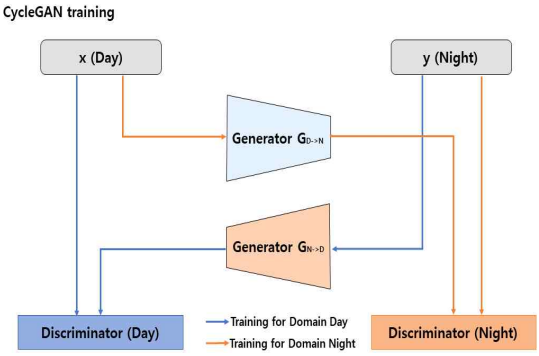


Fig. 1. The flow of CycleGAN [1] training procedure for proposed method.

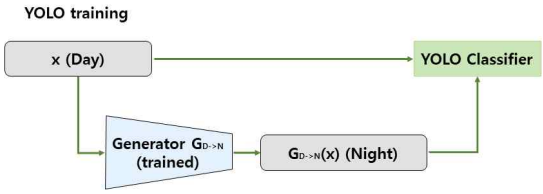


Fig. 2. The flow of YOLO [6] training using Generator of CycleGAN [1].

box, 클래스 등)를 공유한다. 이를 바탕으로 YOLO [6]를 학습시킬 수 있으며 그 흐름은 Fig. 2와 같다.

4. 실험 결과 및 고찰

4.1 BDD dataset [9]

BDD dataset [9]은 test set 20,000장을 포함, 총 100,000장의 이미지와 그에 대한 annotation을 포함한 dataset이다. 각 이미지에 대한 annotation에는 해당 이미지의 전체적인 도메인 (날씨, 장소, 시간대)와 이미지 내 각 물체들의 클래스와 bounding box 등의 정보가 포함되어 있다. BDD dataset [9] 중 training set과 validation set의 시간 도메인별 구성은 Table 1에 있다. BDD dataset [9]의 시간대별 클래스는 Day, Night, Dawn/Dusk, Undefined의 총 4가지 클래스로 이루어져 있다. Day, Night Dawn/Dusk의 경우 각 클래스에 해당하는 이미지는 해당 클래스의 시간대에 수집한 이미지이며, 시간대를 확인할 수 없는 이미지는(예 : 터널이나 건물 내부에서 수집한 경우) Undefined로 분류하였다. 본 논문에서는 BDD dataset [9]의 시간 도메인 중 Day와 Night을 선택하여 사용하였다. BDD dataset [9]의 예시는 Fig. 3에 있다. CycleGAN [1]의 경우 Training Set의 Daytime과 Night 도메인의 이미지 전부를 사용하여 학습을 시키고, YOLO [6]에서는 Day domain 5,000장, Night domain 5,000장씩을 무작위로 추출하여 실험을 위한 학습에 사용하였다. YOLO [6]에서 학습시킨 클래스는 Car, Person, Truck, Bus의 총 4종이며 추출된 도메인에 포함된 클래스별 구성은 Table 2, Table 3과 같다.

Table 1. The distribution of BDD dataset [9] according to weather conditions

	Training Set	Validation Set
Day	36,800	5,258
Night	28,028	3,929
Dawn/Dusk	5,033	778
Undefined	139	35
Total	70,000	10,000

Fig. 3. The example images of BDD dataset [9]. The top row represents the images of Day time, the middle represents the Day time images applied bounding box information from annotation of BDD dataset [9], and the bottom row represents the images of Night time. In the middle row, blue bounding box indicates the location of Car objects, and green bounding box indicates the location of Person.

Table 2. The distribution of objects included in selected training set for experiments

	Day	Night
Car	54,998	45,694
Person	8,986	3,476
Truck	3,079	938
Bus	1,133	378

Table 3. The distribution of objects included in selected test set for experiments

	Day	Night
Car	32,785	27,421
Person	5,606	2,156
Truck	1,789	576
Bus	668	239

4.2 실험 프로토콜

BDD dataset [9]의 Day domain과 Night domain의 이미지를 활용하여 CycleGAN [1]을 학습시켰다. 이후 CycleGAN [1]을 사용하여 생성한 데이터의 학습 데이터로서의 유효성을 입증하기 위해 총 3가지 프로토콜로 YOLO [6]의 학습을 진행하였다. 학습을 위한 프로토콜은 다음과 같다. ① Day 클래스의 이미지만 추출하여 학습, ② Day 클래스와 Night 클래스의 이미지를 균등하게 추출하여 학습, ③ Day 클래스를 추출하고 이와 별개로 다른 Day 클래스의 이미지를 Night 클래스로 변환하여 학습에 사용하는 방식으로 실험하였다. YOLO [6]의 학습을 위해 무작위로 추출한 10,000장의 이미지 (Day 클래스 5,000장, Night 클래스 5,000장)에서 각 이미지가 포함하고 있는 물체들을 추출하여 학습에 사용하였으며 그 분포는 Table 2와 같다. 이후 학습된 YOLO [6]를 이용하여 Day domain과 Night domain에서의 테스트를 진행하였다. test set은 BDD dataset [9]에서 무작위로 3,000장을 추출했으며, 추출된 이미지 내에 존재하는 물체들을 이용하여 YOLO [6]의 테스트를 진행하였다. 테스트에 사용된 물체들의 분포는 Table 3과 같다. 그리고 테스트 결과로 각 클래스 별 mean Average Precision (mAP)를 계산하여 프로토콜별로 그 성능을 비교하였다.

4.3 CycleGAN [1]을 이용한 이미지 변환 결과

4.2에서 제안한 방법대로, YOLO [6]를 학습하기 전 CycleGAN [1]을 사용하여 Day 클래스의 데이터를 Night 클래스의 데이터로 변환하는 작업을 수행하였다. 그 결과에 대한 예시는 Fig. 4에 있다. 변환 전과 변환 후의 이미지를 비교해보면, 이미지 내 물체나 전반적인 특징들의 훼손보다는 이미지의 명암 (혹은 픽셀값) 문제가 큰 것을 알 수 있다. Fig. 4의 오른쪽 두 열에서 볼 수 있듯이, 일부 물체의 명암이 제대로 변환되지 않았거나, 혹은 건물이나 구조물 사이에 낀 배경에서의 명암 변환이 제대로 이루어지지 않는 경우가 있다. 그 문제는 유독 붉은 색을 표현하는 R channel에서 두드러지게 나타났다. 명암 또는 픽셀값에 대한 histogram 정보는 Fig. 5에 있다. Fig. 5에 따르면, 변환이 비교적 잘 된 이미지와 문제가

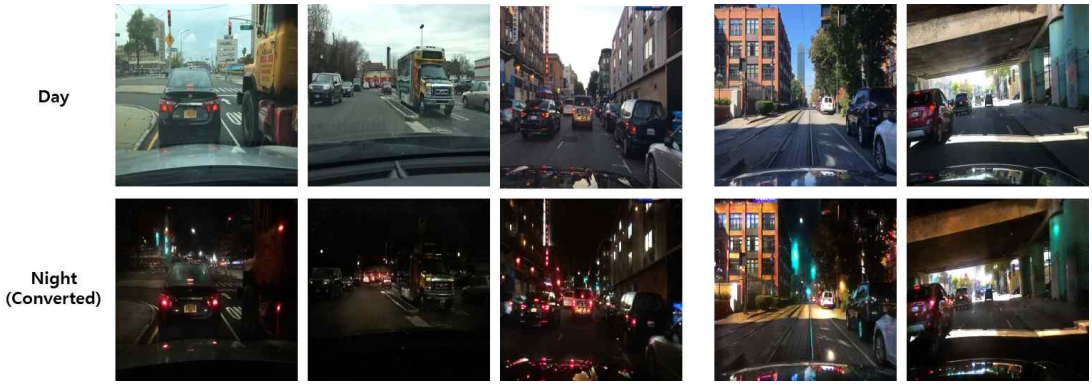


Fig. 4. The result of image translation using CycleGAN [1]. The upper row shows the original Day time image, and the lower row shows the image converted from Day to Night.

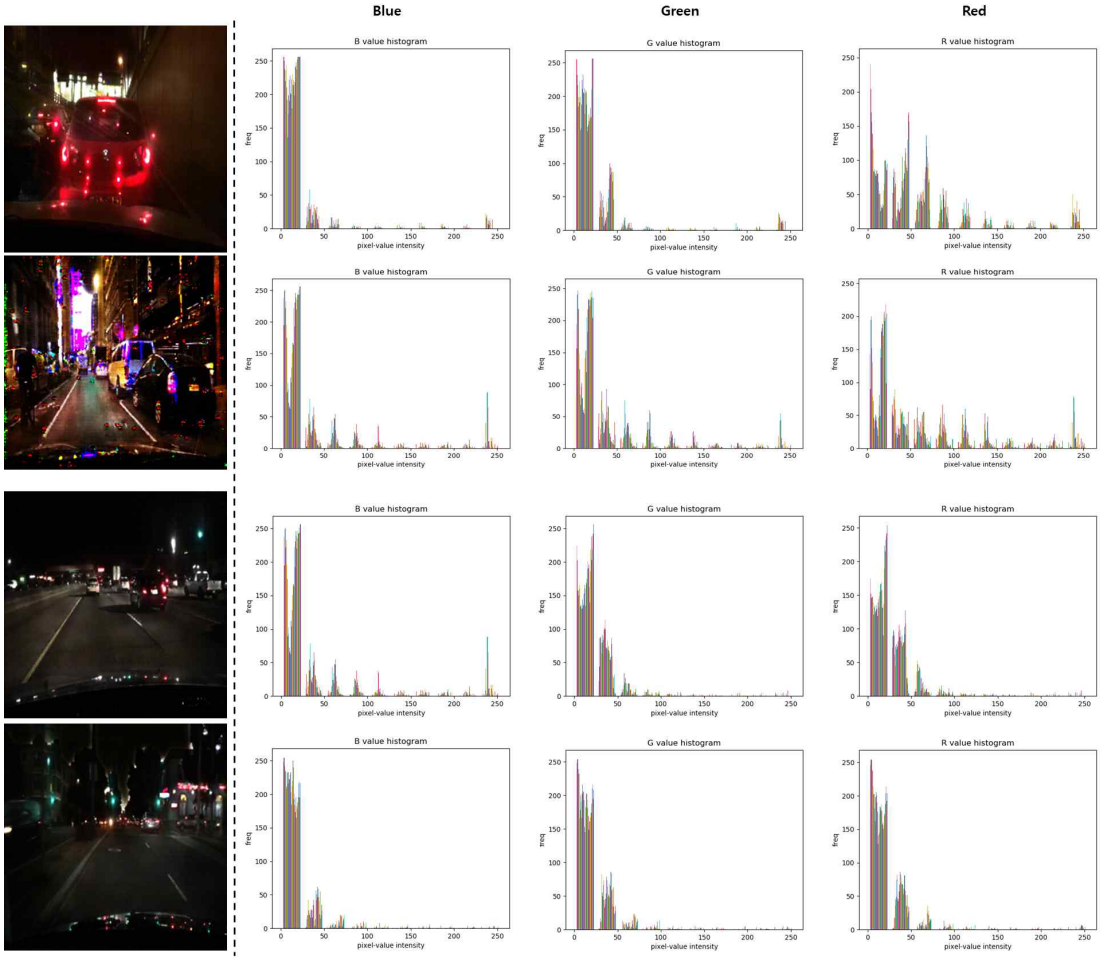


Fig. 5. The histograms of pixel values after conversion using CycleGAN [1]. The leftmost column is the original converted image, and the right shows the corresponding B, G, R pixel histograms of the converted image. The above two images have a problem during the conversion, and the below two images are cases that are converted well.

있는 이미지를 비교했을 때 B channelG channel에서의 차이보다는 R channel의 픽셀값의 차이가 크게 나타남을 볼 수 있다.

4.4 YOLO [6] 결과

CycleGAN [1]을 이용한 이미지 변환 후, 이를 YOLO [6]의 training set에 포함시켜 학습을 진행하였다. 그리고 Day 클래스와 Night 클래스의 데이터를 넣어 각각의 테스트를 진행하였다. 테스트의 성능 지표는 IoU (Intesection of Union)에 해당하는 영역을 기준으로 한 mAP (mean Average Precision)를 이용하였으며, 실험 결과에 대한 요약은 Table 4에 있다. Table 4에 따르면, Day 도메인의 데이터로 테스트를 실행했을 경우 BDD dataset [9]에서 Day 도메인의 이미지만 추출하여 학습했을 때 28.77%,

Table 4. The overall performances of the proposed method

Training	Test (mAP)	
	Day	Night
Day	28.77%	19.14%
Day + Night	29.77%	26.81%
Day + Night (Converted) (Proposed Method)	28.36%	23.40%

BDD dataset [9]에서 Day 도메인의 이미지와 Night 도메인의 이미지를 균등하게 추출하여 학습했을 경우 29.77 %, BDD dataset [9]에서 Day 도메인의 이미지를 추출하고 여기에 이와는 별개로 Day 도메인의 다른 이미지를 CycleGAN [1]을 통하여 Night 도메인으로 변환하여 이를 합친 dataset으로 학습시킨 결과가 28.36 % 로 나왔다. 이 중 가장 기본이 되는 첫 번째 프로토콜과 제안 방식인 마지막 프로토콜을 비교해보면 Day의 경우 YOLO [6] 학습시 허용오차 범위 내에 포함되지만 소폭 감소함을 확인할 수 있다. 반면 Night 도메인의 데이터로 테스트를 실행했을 경우 BDD dataset [9]에서 Day 도메인의 이미지만 추출하여 학습했을 때 19.14 %, BDD dataset [9]에서 Day 도메인의 이미지와 Night 도메인의 이미지를 균등하게 추출하여 학습했을 경우 26.81 %, 마지막으로 Day 도메인과 변환시킨 Night 도메인의 이미지를 합친 dataset으로 학습시킨 결과는 23.40 % 로 나왔다. CycleGAN [1]으로 변환시킨 Night 이미지를 섞었을 경우 변환시키지 않은 기존의 Night 이미지를 섞었을 때보다는 성능이 떨어지지만, 단순히 Day만 사용했을 때의 성능보다는 어느 정도 개선이 있음을 확인하였다. 성능 차이에 대한 예시 이미지는 Fig. 6에 있다.

Table 4에 대한 상세한 결과는 Table 5에 있다. Table 5에 따르면 Night 도메인에서의 테스트에서 Car나 Person의 검출 성능 자체는 개선이 별로 보이지 않는다. 그러나 Car나 Person에 비해 비교적 데이터가 적은 편인 Truck이나 Bus에서 꽤 높은 성능 개선율을 보였다. 이는 Car, Person 클래스의 개별적인 물체 크기와 Bus, Truck 클래스의 물체의 개별적인 물체 크기의 차이에서 비롯된 것으로 보인다. 실제 Car, Person 클래스와 Bus, Truck 클래스의 평균적인 크기에 차이가 존재하기 때문에 같은 거리에 있을지라도 이미지 내에서 차지하는 픽셀 수(혹은 크기)에 차이가 있다. 그리고 Fig. 4, Fig. 6에서 볼 수 있듯이, 이미지 내에서 차지하는 픽셀 수가 작은 샘플들의 대부분이 Car 클래스와 Person 클래스에 분포되어 있음을 알 수 있다. 크기가 작은 샘플들이 많기 때문에 라벨링된 영역과 검출된 영역 사이의 겹치는 영역인 Intersection 부분이 상대적으로 큰 샘플들이 많은 Bus, Truck 클래스에 비해 작을 수밖에 없고, 이로 인해 검출 오류가 발생하여 동일한 크기의 오류가 발생하더라도 Car, Person 클래스의 검출 성능은 Bus, Truck 클래스의 성능에 비해 영향을 크게 받을 수밖에 없다. 이러한 원인으로 인해 Car 클래스와 Person 클래스의 검출 성능 개선율이 Bus, Truck 클래스의 검출 성능 개선율에 비해 떨어지는 현상이 야기된 것으로 보인다. 물론 Table 2, Table 3에서 볼 수 있듯이 Car, Person 클래스와 Bus, Truck 클래스 사이의 샘플 개수의 차이도 크기 때문에 차후 이어지는 실험에서는 Truck과 Bus 클래스를 포함한 샘플들을 추가시키거나, 이 논문에서 실험에 사용한 샘플들의 data augmentation 등의 전처리 작업을 통해 클래스 사이의 샘플 균등성을 확보해야 할 것이다.

5. 결 론

본 논문에서는 네트워크의 학습에 있어서 데이터 샘플이 적은 클래스 혹은 dataset의 데이터 부족으로 인한 문제를 해결하기 위해 CycleGAN을 기반으로 하는 데이터 샘플링 기법을 제안하였다. 이 샘플링으로 생성한 데이터의 유효성을 입증하기 위해 기존의 dataset에서 학습시킨 YOLO와 생성한 data를 포함한 dataset으로 학습시킨 YOLO를 비교하는 실험을 진행하였다. 그 결과 크지는 않지만 어느 정도 유효

Table 5. The detailed performances according to different classes at the proposed method

Training	Test (mAP)							
	Day				Night			
	Car	Bus	Truck	Person	Car	Bus	Truck	Person
Day	45.74%	25.40%	24.78%	19.16%	33.22%	15.96%	14.32%	13.05%
Day + Night	44.71%	28.65%	27.31%	18.44%	41.40%	26.84%	22.73%	16.26%
Day + Night(Converted) (Proposed Method)	43.89%	23.72%	27.40%	18.42%	33.34%	26.24%	20.64%	13.38%

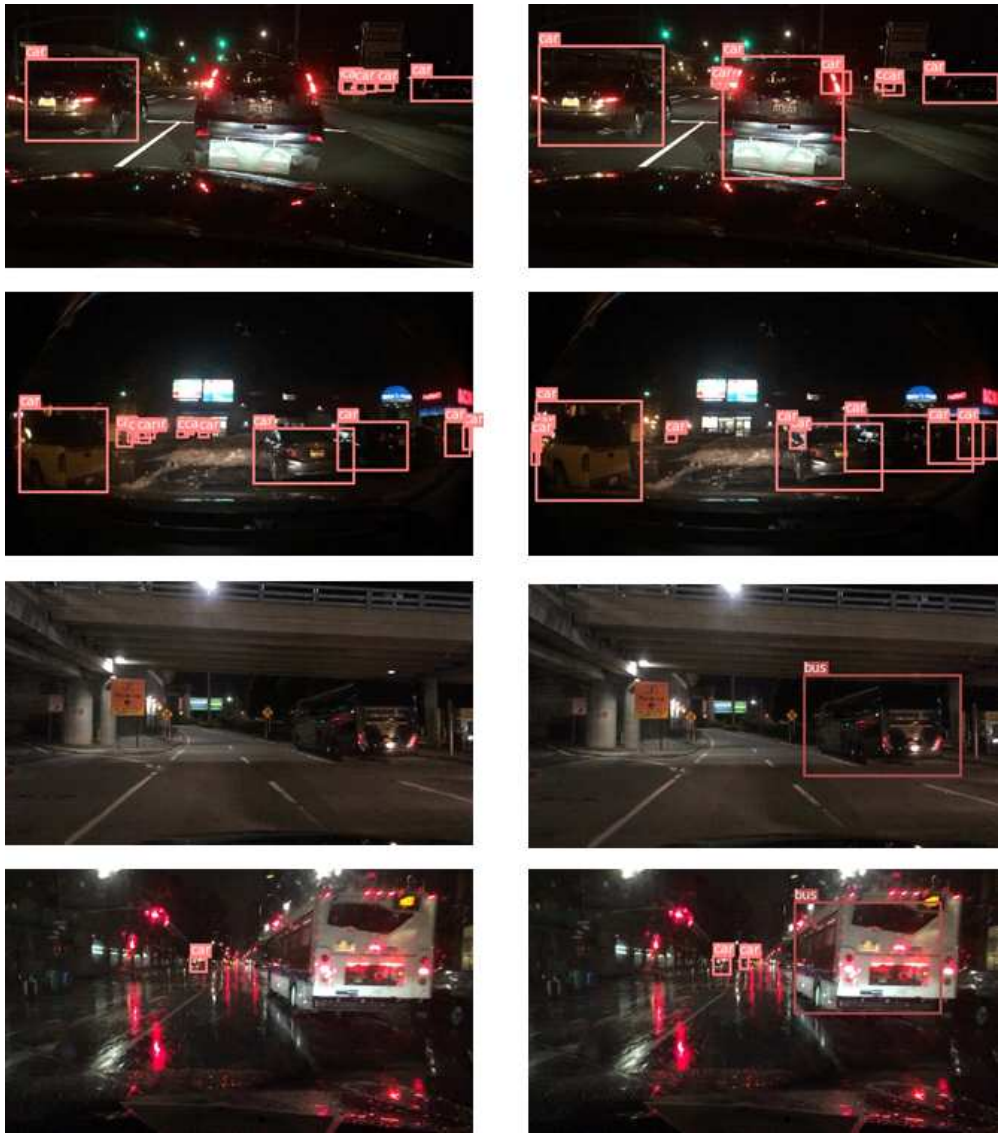


Fig. 6. The example results of the proposed method. Left images are results of YOLO algorithm trained using only Day class data, and right images are results of YOLO algorithm trained using Day class and synthesized Night class data. By the proposed method, YOLO can detect undetected objects such as a bus or cars under the night time.

한 성능 향상을 확인할 수 있었으며, 이로써 Cycle GAN으로 생성한 데이터들도 학습에 활용할 수 있음이 입증되었다. 이를 활용하여 향후 데이터 샘플이 부족한 dataset의 문제를 해결할 수 있을 것이다.

REFERENCE

[1] J.Y. Zhu, T. Park, P. Isola, and A.A. Efros, "Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks," *Proceeding of International Conference on Computer Vision*, pp. 2242-2251, 2017.

[2] Y. Choi, M. Choi, M. Kim, JW. Ha, S. Kim, J. Choo, et al., "StarGAN: Unified Generative Adversarial Networks for Multi-Domain Image-

to-Image Translation," *Proceeding of International Conference on Computer Vision and Pattern Recognition*, pp. 8789-8797, 2018.

[3] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, et al., "Generative Adversarial Nets," *Proceeding of Conference on Neural Information Processing Systems*, pp. 1-9, 2014.

[4] P. Isola, J.Y. Zhu, T. Zhou, and A.A. Efros, "Image-to-Image Translation with Conditional Adversarial Networks," *Proceeding of International Conference on Computer Vision and Pattern Recognition*, pp. 5967-5976, 2017.

[5] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, CY. Fu, et al., "SSD: Single Shot Multibox Detector," *Proceeding of European Conference on Computer Vision*, pp. 21-37, 2016.

[6] J. Redmon and A. Farhadi, "YOLO9000: Better, Faster, Stronger," *Proceeding of International Conference on Computer Vision and Pattern Recognition*, pp. 6517-6525, 2017.

[7] M. Everingham, L. Van Gool, C.K.I. Williams, J. Winn, and A. Zisserman, "The PASCAL Visual Object Classes (VOC) Challenge," *International Journal of Computer Vision*, Vol. 88, No. 2, pp. 303-338, 2010.

[8] T.Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, et al., "Microsoft COCO: Common Objects in Context," *Proceeding of European Conference on Computer Vision*, pp. 740-755, 2014.

[9] F. Yu, W. Xian, Y. Chen, F. Liu, M. Liao, V. Madhavan, et al., "BDD100K: A Diverse Driving Video Database with Scalable Annotation Tooling," *arXiv Preprint arXiv:1805.04687*, 2018.

[10] E. Denton, S. Chintala, A. Szlam, and R. Fergus, "Deep Generative Image Models Using a Laplacian Pyramid of Adversarial Networks," *Proceeding of Conference on Neural Information Processing Systems*, pp. 1486-1494, 2015.

[11] A. Radford, L. Metz, and S. Chintala, "Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks," *Proceeding of International Conference on Learning Representations*, pp. 1-16, 2016.

[12] M. Mirza and S. Osindero, "Conditional Generative Adversarial Nets," *arXiv Preprint arXiv:1411.1784*, 2014.

[13] M.Y. Liu and O. Tuzel, "Coupled Generative Adversarial Networks," *Proceeding of Conference on Neural Information Processing Systems*, pp. 469-477, 2016.

[14] M.Y. Liu, T. Breuel, and J. Kautz, "Unsupervised Image-to-Image Translation Networks," *Proceeding of Conference on Neural Information Processing Systems*, pp. 700-708, 2017.

[15] T. Kim, M. Cha, H. Kim, J.K. Lee, and J. Kim, "Learning to Discover Cross-Domain Relations with Generative Adversarial Networks," *Proceeding of International Conference on Machine Learning*, pp. 1857-1865, 2017.

[16] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual Losses for Real-Time Style Transfer and Super-Resolution," *Proceeding of European Conference on Computer Vision*, pp. 694-711, 2016.

[17] L.A. Gatys, A.S. Ecker, and M. Bethge, "Image Style Transfer Using Convolutional Neural Networks," *Proceeding of International Conference on Computer Vision and Pattern Recognition*, pp. 2414-2423, 2016.

[18] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," *Proceeding of International Conference on Computer Vision and Pattern Recognition*, pp. 770-778, 2016.

[19] D.P. Kingma and J.L. Ba, "ADAM: A Method for Stochastic Optimization," *Proceeding of International Conference on Learning Representations*, pp. 1-15, 2015.

[20] J. Redmon, S. Divvala, R. Girshick, and A.

Farhadi, “You only Look Once: Unified, Real-Time Object Detection,” *Proceeding of International Conference on Computer Vision and Pattern Recognition*, pp. 779-788, 2016.

[21] H.S. Ha and B.Y. Hwang, “Enhancement Method of CCTV Video Quality Based on SRGAN,” *Journal of Korea Multimedia Society*, Vol. 21, No. 9, pp. 1027-1034, 2018.

조 상 흠

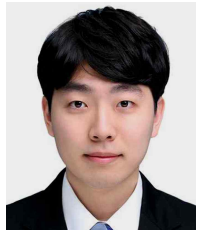
2017년 아주대학교 소프트웨어학과 학사
2017년 아주대학교 컴퓨터공학과 석사 재학 중

이 용

1988년 한국항공대학교 공학사
2001년 일본 교토대학 공학석사
2003년 일본 교토대학 공학 박사
2003년~2008년 삼성종합기술원 전문연구원
2008년~20011년 일본 효고현립대학 정보미디어연구실 특임강사/특임준교수
2011년~2013년 일본 NICT연구소 연구원
2017년~현재 일본 관서학원대학 사회정보학연구센터 객원연구원
2013년~현재 한국과학기술정보연구원(KISTI) 연구데이터허브센터 선임연구원

나 재 민

2018년 아주대학교 소프트웨어학과 학사
2018년 아주대학교 컴퓨터공학과 석사 재학 중



김 영 빈

2018년 아주대학교 디지털 미디어학과 학사
2018년 아주대학교 컴퓨터공학과 석사 재학 중

박 민 우

1992년 충남대학교 전산학과 이학사
2004년 충남대학교 컴퓨터과학 석사
1996년 현재 한국과학기술정보연구원 선임연구원

이 상 환

1992년 울산대학교 전자계산학과 공학사
2004년 고려대학교 SW공학과 공학석사
2018년 서울시립대학교 컴퓨터공학과 공학박사
1995년~2010년 한국과학기술정보연구원(KISTI) 선임연구원
2015년~현재 한국과학기술정보연구원(KISTI) 연구데이터허브센터 센터장

황 원 준

1999년 고려대학교 전자 공학과 학사
2001년 고려대학교 전자 공학과 석사
2001년~2016년 삼성종합기술원 전문연구원
2016년 KAIST 전기 및 전자공학부 박사
2016년 아주대학교 소프트웨어학과 조교수