

# 딥 러닝 기반 이미지 압축 기법의 성능 비교 분석

이용환\*·김흥준\*\*†

\*원광대학교 디지털콘텐츠공학과, \*\*경상국립대학교 컴퓨터과학부

## Comparison Analysis of Deep Learning-based Image Compression Approaches

Yong-Hwan Lee\* and Heung-Jun Kim\*\*†

\*Dept. of Digital Contents, Wonkwang University,

\*\*†School of Computer Sciences, Gyeongsang National University

### ABSTRACT

Image compression is a fundamental technique in the field of digital image processing, which will help to decrease the storage space and to transmit the files efficiently. Recently many deep learning techniques have been proposed to promise results on image compression field. Since many image compression techniques have artifact problems, this paper has compared two deep learning approaches to verify their performance experimentally to solve the problems. One of the approaches is a deep autoencoder technique, and another is a deep convolutional neural network (CNN). For those results in the performance of peak signal-to-noise and root mean square error, this paper shows that deep autoencoder method has more advantages than deep CNN approach.

**Key Words** : Image Compression, Encoding and Decoding, Deep Learning, Deep Autoencoder, Deep Convolutional Neural Network

### 1. 서 론

최근 이미지 압축은 딥 러닝(Deep Learning) 영역에서 주요한 연구 주제로 부각되면서 딥 러닝 기술을 활용한 다양한 압축 기법들이 연구되고 있다[1, 8]. 딥 러닝과 콘볼루션 신경망(Convolutional Neural Networks)은 다양한 컴퓨터 비전 분야에서 높은 성능적 우위를 보였으며, 이미지 압축 분야까지 활용성이 넓혀지고 있다[2, 3, 26]. 최근 두드러진 성능을 보이는 학습 기반 이미지 압축 알고리즘으로, 대표적인 Deep Autoencoder [4]과 Deep CNN 접근 방법[5]이 사용되고 있다. Autoencoder은 데이터를 고차원공간에서 저차원공간으로 변환하는 딥러닝 모델이며, 데이터를 압축(Encoding)하는 인코더 부분과 원본 이미지로 재구성

(Decoding)하는 디코더 부분으로 구성된다[6]. 콘볼루션 신경망(CNN)은 2차원 데이터와 함께 동작하도록 설계된 특별한 유형의 신경망 모델이며, 네트워크의 주요 블록인 여러 콘볼루션 계층으로 구성된다[7]. 입력 필터는 콘볼루션과 특징 맵에서의 반복적인 입력 콘볼루션을 적용하여 연산하며, 특징 맵은 입력 이미지에서 검출된 특징의 강도와 위치를 포함하는 활성화된 맵이 된다. CNN은 이미지 분류와 같은 특정 애플리케이션 모델링의 제약 조건에서 특정 데이터 셋에 대한 많은 수의 필터를 병렬적으로 자동 학습할 수 있다는 장점이 있다[9].

본 연구에서는 다양한 이미지 압축 기법들 중에서도 딥 러닝을 통한 이미지 압축에 집중하고, 대표적인 알고리즘을 살펴본다. 본 논문에서는 2개의 알고리즘으로, David이 제안한 Autoencoder [4]와 Lukas가 제안한 CNN [5]을 선별하여 분석한다.

†E-mail: thinkthe@gnu.ac.kr

Autoencoder 구조는 이미지를 압축 및 압축 해제하기 위한 대칭적 컨볼루션 계층과 잔차 블록(Residual Block) 계층의 집합으로 구성된다[10]. 인코더에서는 이미지의 차원을 여러 특징 맵(Feature Map)으로 줄이고, 특징 맵은 이미지의 밀도 높게 채워진 칼라 값을 포함한다. 디코더에서는 원본 이미지의 원래 값에 가깝게 재구성한다. 각 특징 맵은 특징 맵의 기능 샘플들 간의 비트 할당을 제어하는 중요도 맵(Importance Map)으로 변조되며, 출력 이미지의 최종 압축률을 결정하는 중요한 요소가 된다. 중요도 맵은 이미지의 중요한 특징 샘플을 식별하는 일련의 학습된 컨볼루션 계층에 의해 생성된다. 특정 이미지에서 특징 맵의 비트 깊이를 공간적으로 다양한 방식의 중요도 마스크 세트를 생성한다. 이에 따라, 최종 특징 맵은 특정 비율에서 이미지를 표현하는데 가장 의미 있는 정보만을 보유하게 된다. 양자화(Quantization) 연산에서는 훈련 문제를 해결하기 위해 소프트 양자화 기법을 적용하며, 속도 제어를 위해 중요도 마스크 값을 기반으로 속도 추정을 구현하였다. 절단된 특징 맵을 압축하는 엔트로피 코딩은 JPEG2000 표준 압축을 기반으로 개발되었다. 엔트로피 코딩 결과의 대부분은 최하위 비트 영역에서 모두 0으로 수렴하는 특징 맵 특성과 일치하도록 컨텍스트 모델로 조정된다.

Deep CNN은 이미지 분할, 이미지 분류 및 객체 검출 분야에서 필수적인 컴퓨터 비전 기술이 되었다[11, 12, 27]. Lukas의 CNN 기본 개념은 다음의 3가지 특성으로 요약된다. 첫 번째는 새로운 딥 러닝 기반 컨볼루션 신경망 접근 방식으로 JPEG 압축 이미지 데이터에서 압축 아티팩트를 개선하는 기술이다. 두 번째는 적은 비전 작업으로 조정 가능한 딥 네트워크 훈련 방식을 제시한다. 세 번째는 압축 아티팩트 억제(Compression Artifact Suppression)에 대한 컨볼루션 넷의 현재 상태 성능을 보여주는 네트워크 속성을 강조하기 위해 이미지 데이터셋의 성능을 평가한다[13].

본 논문의 구성은 다음과 같다. 2장에서는 딥러닝 기술을 활용한 대표적인 이미지 압축 기법을 상세히 살펴보고, 3장에서는 공개소스를 통해 알고리즘을 측정하고 4장에서 결론으로 마무리한다.

## 2. 압축 알고리즘

본 절에서는 딥 러닝 기반 이미지 압축 연구에서 대표적인 2가지 알고리즘을 살펴본다. Pavel은 CNN을 활용하여 아티팩트를 제거하는 기능을 제안하였다[14]. CNN 구조에서는 엣지 강조 손실 함수, 대칭 가중치 초기화 및 스킵 연결을 사용하여 회귀 작업을 위한 딥 러닝 네트워크의 훈련을 방해하는 장애 요소를 제거하였다.

### 2.1 Deep CNN 구조

전체 네트워크 구조는 Fig 1과 같다[5]. A에서 D블록은 각 블록당 2개의 컨볼루션 계층으로 구성되며, 채널 수가 1에서 128로 증가한 다음 256으로 증가한다. 컨볼루션 네트워크의 주요 경로는 Fig 1에서 파란색으로 표시되어 있다. 따라서, 표준 학습 방법을 사용하여 학습할 수 없는 12개 계층의 컨볼루션 네트워크를 얻게 된다[15]. D의 가중치를 변경하기 위한 그라디언트는 출력에서 A, B, C, D, D를 통해 전달된다. Fig 1에서 녹색으로 표시된 경로는 단일 컨볼루션 계층을 사용하여 네트워크 내에서 저해상도 이미지의 재구성 경로를 개선하기 위해 사용된다. 그런 다음, 가장 가까운 이웃 보간법을 사용하여 출력을 2배로 업-샘플링하고 특징 맵을 연결하여 Fig 1에서 노란색으로 표시된 경로와 평행한 전체 컨볼루션 계층을 생성한다. 가능한 정확도를 줄이기 위해 배치 정규화가 수행되고, 평균과 분산은 최대 정확도로 네트워크의 완전 수렴을 피하는 배치-배치 지터(Batch-to-Batch Jitter)를 반영한다[16, 17].

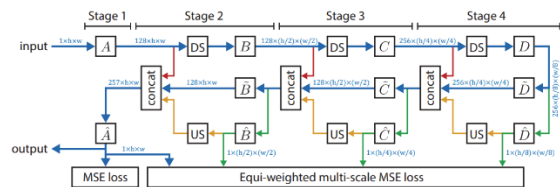


Fig. 1. Structure of Deep Convolutional Neural Network [5]. The paths are color coded: Main path (Blue), Concatenation of lower-level features (Red), Multi-scale output paths (Green), Re-use of multi-scale outputs (Yellow).

### 2.2 손실 함수

MSE 기준은 참조 이미지의 변화를 수정하여 네트워크 학습 중에 최소화된다[14]. 학습 절차를 개선하기 위해 네트워크 내에서 전체 해상도 출력과 저해상도 출력을 포함한다. MSE를 계산하기 위한 기준은 각각 4, 16 및 64 픽셀을 평균하여 입력 이미지를 다운-샘플링하여 계산한다. 각 출력의 평균 제곱 오차(Mean-squared Error)를 계산하여 다중 스케일(Multi-scale) 손실 함수를 계산한다. 출력 손실에 기여하는 전체 해상도 출력 이미지의 MSE를 최소화하기 위해 여러 단계에서 학습이 계속적으로 진행되며, 최적화 목표를 달성하기 위해 네트워크의 미세 조정(Fine Tuning)이 수행된다 [18].

### 2.3 Deep Autoencoder 구조

Autoencoder는 숨겨진 계층( $h$ )의 힌트를 통해 입력 데이터를 출력에 복사하도록 훈련된 피드-전방(Feed-forward)

신경망을 갖는다. 네트워크 구조는 크게 2개 영역으로 구성된다[19]. 첫번째 영역은 입력 데이터를 잠재 공간 표현 또는 수식(1)에 표현된 코드로 압축하는 인코더와 같은 영역이다. 두번째 영역은 수식(2)로 표현된 잠재 공간 표현에서 입력을 재구성하는 디코더 부분이다.

$$h = f(x) \tag{1}$$

$$r = g(h) \tag{2}$$

$h$ 는 숨겨진 계층을,  $x$ 는 입력 계층을,  $r$ 는 재구성된 계층을,  $g$ 는 인코더와 디코더를 의미한다. 이러한 함수는 수식(3)으로 표시할 수 있다.

$$g(f(x)) = r \tag{3}$$

학습 과정은  $L(x, g(f(x)))$  으로 표현되며,  $L$ 은 MSE로 표현되는  $x$ 로부터 발산하는 손실함수  $g(f(x))$ 를 나타낸다.

Autoencoder은 Stacked Autoencoder 또는 Deep Autoencoder로 설명되는 여러 숨겨진 계층을 갖는다. Autoencoder의 추가 계층은 적은 계산 시간으로 다수의 복잡하고 비선형 관계를 나타내는 코딩의 장점을 제공한다[20]. Fig.2는 다중 계층을 갖는 Deep Autoencoder 신경망 구조를 보여준다. 이러한 구조는 훈련 과정에서 그라디언트가 사라지는 문제가 있다. 이 문제를 해결하기 위해, 사전 훈련 단계에서 초기 가중치를 임의로 할당한 후, 매개변수 튜닝을 위해 BPN을 통합한다.

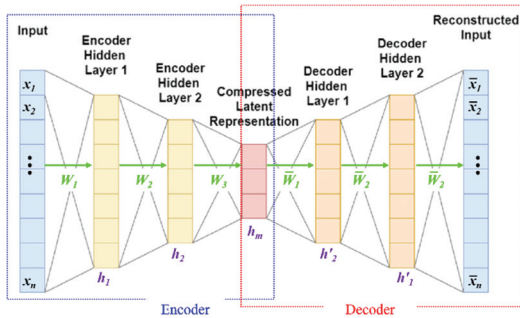


Fig. 2. Structure of Deep Autoencoder Neural Network.

### 2.4 Deep Boltzmann Machine 학습

다중 계층을 갖는 비선형 Autoencoder에서의 가중치 최적화의 복잡도 문제는 Salakhutdinov가 제안한 사전 훈련 방식을 적용하여 해결한다[21]. Deep Boltzmann Machine (DBM)에서는 계층 간의 무방향(Undirected) 연결을 포함하며, 복잡해진 내부 표현을 학습하는 기능을 갖는다. 레이

블이 지정되지 않은 데이터를 학습하고 원하는 데이터로 미세 조정하는 계층별 훈련 절차를 수행한다. 다른 네트워크 구조(Deep belief network와 Deep CNN)와 비교하여, 하향식 피드백에 의해 달성된 간접 단계와 상향식 프로세스가 추가되어 DPM이 불명확한 입력에 대해 불확실성을 보다 명확하게 통합할 수 있다. 이를 통해 보다 좋은 촉진 학습 생성 모델을 제공할 수 있다. DBP 구조를 보다 명확하게 생성하기 위해서는 에너지 함수(Energy Function)를 지정해야 한다[22]. 2개 계층 모델을 위한 에너지 함수 식은 다음과 같다.

$$E_{DBM}(v, h^{(1)}, h^{(2)}, \theta) = -v^T W h^{(1)} - h^{(1)T} V h^{(2)} - d^{(1)T} h^{(1)} - d^{(2)T} h^{(2)} - b^T v \tag{4}$$

수식(4)에서  $\theta = \{W, V, d^{(1)}, d^{(2)}, b\}$  을 의미한다. DBM은 두 정점 사이의 이분 그래프로 간주된다. 평균장 분포(Mean-field Distribution)에 따라 하한을 늘리고 수렴에 도달할 때까지 반복적인 연산을 수행한다.

Fig. 3(1)은 DBM을 생성하기 위해 정리되지 않은 Restricted Boltzmann Machine의 학습 스택에 3개의 숨겨진 계층이 있는 사전 학습 과정을 보여준다. 3개의 모듈로 구성하여 단일 모델을 형성하면 첫번째와 두번째 숨겨진 계층으로 들어오는 전체 입력이 반으로 줄어든다, 계층 복제는 제거된다[21]. Fig 3(2)의 Deep Boltzmann Machine은 다수

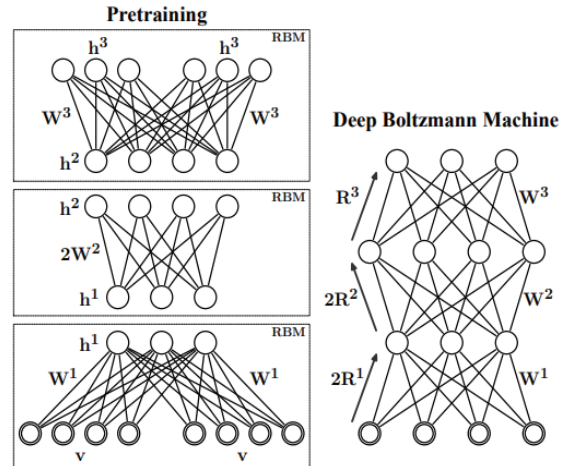


Fig. 3. Pre-training (1) and Deep Boltzmann Machine (2) : Left (1) is pre-training of DBM with 3 hidden layers consists of learning a stack of RBM's that are then composed to create a DBM. Right (2) is resulting DBM, where the parameters  $\{R^1, R^2, R^3\}$  define the recognition model [21].

개의 숨겨진 계층으로 구성된 심층 생성 무방향 모델 (Deep Generative Undirected Model)을 갖는다. 저급-레벨 특징의 학습에 영향을 미치기 위해 하향식 연결 패턴에서 동작한다.  $R^1$ ,  $R^2$  및  $R^3$ 는 인식 모델의 가중치를 의미하며, 하향식 피드백의 단점을 보완하기 위해 가중치는 각 계층에서 2배값을 적용한다.

### 3. 실험결과

#### 3.1 데이터셋

ImageNet은 대략 22,000개 범주를 갖는 1,500만개 이상의 다양한 고해상도 이미지 데이터 셋으로 구성된다[23]. 이 중에서 다양한 유형의 약 200개 이미지를 샘플링하여 선별하고 사전처리 없이 서로 다른 파일 포맷의 이미지를 2개의 알고리즘에 적용하여 실험한다[24].

#### 3.2 실험 및 논의

JPEG 포맷 이미지에 대한 PSNR 및 MSE 값을 계산하여 평가하며, 평가 결과는 Table 1과 같다. PSNR과 MSE 값을 계산하는 식은 수식(5)과 수식(6)과 같다[25]. Deep CNN과 Deep Autoencoder 방식에 대한 PSNR과 RMSE를 평가하고, 다양한 이미지 데이터셋에 대한 성능을 비교하였다.

$$PSNR = 10 \times \log_{10} \left( 255^2 / \sqrt{MSE} \right) \quad (5)$$

$$MSE = \frac{1}{N} \times \sum_i \sum_j (x, y) - F(x, y))^2 \quad (6)$$

**Table 1.** Comparison of PSNR and RMSE for different images using Deep Autoencoder and Deep CNN

Algorithms	Images	PSNR	RMSE
Deep Autoencoder [4]	Lighthouse	52.64	0.005
	Sherlock	51.17	0.003
	Office	54.97	0.012
	Yellow Lily	57.61	0.008
	Indian Corn	53.75	0.038
	Lama	52.18	0.014
Deep CNN [5]	Lighthouse	28.89	4.918
	Sherlock	34.81	3.252
	Office	29.47	5.141
	Yellow Lily	33.85	3.531
	Indian Corn	24.83	7.193
	Lama	31.68	4.165

Table 1에서 보듯이, 서로 다른 이미지에 대해 Deep Autoencoder 방식의 경우, PSNR 값이 51dB ~ 58dB 사이의

값을 보였으며, Deep CNN 방식은 24dB ~ 35dB 사이의 값을 가졌다. 즉, Deep CNN 기법에 비해, Deep Autoencoder 기법에서 PSNR 값이 크게 증가하였다. RMSE(Root MSE) 값을 비교하면, Deep CNN 방식은 3.2 ~ 7.1 사이의 값을 보였으며, Deep Autoencoder 방식은 0.002 ~ 0.014 사이의 값으로 크게 감소하였다. 즉, PSNR 및 RMSE를 성능 매개변수 값을 평가할 경우, Autoencoder 기법이 Deep CNN 기법보다 이미지 압축 애플리케이션에서는 보다 효율적인 압축 방식이라는 것으로 결과에서 알 수 있다.

### 4. 결론

본 연구에서는 JPEG 이미지 압축을 위한 Deep CNN 기법과 Deep Autoencoder 기법에 대해 다중 스케일 손실 함수로 학습하고, 성능적 비교 분석을 수행하였다. 다양한 이미지에 대해 PSNR과 RMSE를 평가 요소로 비교 실험한 결과, Deep Autoencoder 기법이 Deep CNN 기법에 비해 PSNR 값이 증가하고 RMSE 값을 감소하는 것으로 나타났다. 수치상으로, Deep Autoencoder 방식이 Deep CNN 기법보다 보다 효율적인 이미지 압축 알고리즘임을 실험적으로 검증하였다. 이러한 알고리즘은 JPEG 특정 이미지 압축에 국한되지 않고, 보다 광범위한 애플리케이션 영역에서 학습 기반 이미지 압축 방식으로 활용될 수 있다.

### 감사의 글

본 연구는 2023년도 정부(미래창조과학부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임(과제번호: 2021R1A2C1012947).

### 참고문헌

1. Manish I. Patel, Sirali Suthar, Jil Thakar, "Survey on Image Compression using Machine Learning and Deep Learning", Proceedings of the International Conference on ICICCS, pp.1103-1105, 2019.
2. Baba Fakruddin Ali, Prakash R., "Overview on Machine Learning in Image Compression Techniques", Innovations in Power and Advanced Computer Technologies, 2021.
3. Hajar Maseeh Yasin, Adnan Mohsin Abdulazeez, "Image Compression based on Deep Learning: A Review", Asian Journal of Research in Computer Science, vol.8, issue.1, pp.62-76, 2021.
4. David Alexandre, Chih-Peng Chang, Wen-Hsiao Peng, Hsueh-Ming Hang, "An Autoencoder-based Learned Image Compressor: Description of Challenge Proposal

- by NCTU”, Workshop on Computer Vision and Pattern Recognition (CVPR), 2019. [Available at] <https://arxiv.org/abs/1902.07385v1>
5. Lukas Cavigelli, Pascal Hager, Luca Benini, “CAS-CNN : A Deep Convolutional Neural Network for Image Compression Artifact Suppression”, International Joint Conference on Neural Networks (IJCNN), 2017.
  6. Website; <https://wiki.pathmind.com/deep-autoencoder>
  7. Website; <https://wiki.pathmind.com/convolutional-network>
  8. Y.H. Lee, “Study on Image Compression Algorithm with Deep Learning”, Journal of the Semiconductor & Display Technology, vol.21, no.4, 2022.
  9. Rikiya Yamashita, Mizuho Nishio, Richard Kinh Gian Do, Kaori Togashi, “Convolutional neural networks: an overview and application in radiology”, Insights into Imaging, vol.9, pp.611-629, 2018.
  10. Guoan Cheng, Ai Matsune, Qiuyu Li, Leilei Zhu, Huaijuan Zang, Shu Zhan, “Encoder-Decoder Residual Network for Real Super-resolution”, Conference on Computer Vision and Pattern Recognition, 2019.
  11. Shervin Mimaee, Yuri Boykov, Fatih Porikli, Antonio Plaza, Nasser Kehtarnavaz, “Image Segmentation using Deep Architecture: A Survey”, IEEE Transaction on Pattern Analysis and Machine Intelligence, vol.44, issue.7, 2022.
  12. Ian H. Witten, Eibe Frank, Mark A. Hall, Christopher J. Pal, Data Mining (Fourth Edition), Practical Machine Learning Tools and Techniques, 2017.
  13. Chao Dong, Chen Change Loy, Kaiming He, Xiaoou Tang, “Learning a Deep Convolutional Network for Image Super-Resolution”, Lecture Notes in Computer Science, vol.8692, pp.184-199, 2014.
  14. Pavel Svoboda, Michal Hradis, David Barina, Pavel Zemcik, “Compression Artifacts Removal Using Convolutional Neural Networks”, International Conference in Central Europe on Computer Graphics and Visualization (WSCG), 2016.
  15. Jonathan Long, Evan Shelhamer, Trevor Darrell, “Fully Convolutional Networks for Semantic Segmentation”, Computer Vision and Pattern Recognition (CVPR), 2015.
  16. Sharan Chetlur, Cliff Woolley, Philippe Vandermersch, Jonathan Cohen, John Tran, “cuDNN: Efficient Primitives for Deep Learning”, <https://arxiv.org/abs/1410.0759>, 2014.
  17. Zhangyang Wang, Ding Liu, Shiyu Chang, Qing Ling, Yingzhen Yang, Thomas S. Huang, “D3: Deep Dual-Domain Based Fast Restoration of JPEGCompressed Images”, IEEE Conference on Computer Vision and Pattern Recognition, 2016.
  18. Huibin Chang, Michael K. Ng, Tiejong Zeng, “Reducing Artifacts in JPEG Decompression Via a Learned Dictionary”, IEEE Transactions on Signal Processing, Vol.62, Issue.3, 2014.
  19. Lijun Zhao, Huihui Bai, Anhong Wang, Yao Zhao, “Multiple Description Convolutional Neural Networks for Image Compression”, IEEE Transactions on Circuits and Systems for Video Technology, Vol.29, Issue.8, 2019.
  20. Zhengxue Cheng, Heming Sun, Masaru Takeuchi, Jiro Katto, “Energy Compaction-Based Image Compression Using Convolutional AutoEncoder”, IEEE Transactions on Multimedia, Vol.22, Issue.4, 2020.
  21. Ruslan Salakhutdinov, Hugo Larochelle, “Efficient Learning of Deep Boltzmann Machines”, Proceeding of International Conference on Artificial Intelligence and Statistics (AISTATS), 2010.
  22. Ruslan Salakhutdinov, Geoffrey Hinton, “A Better Way to Pretrain Deep Boltzmann Machines”, Advances in Neural Information Processing Systems (NIPS), 2012.
  23. Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, Li Fei-Fei, “ImageNet: A large-scale hierarchical image database”, IEEE Conference on Computer Vision and Pattern Recognition, 2009.
  24. Website; <https://www.image-net.org/download.php>
  25. Giuseppe Valenzise, Andrei Purica, Vedad Hulusic, Marco Cagnazzo, “Quality Assessment of Deep-Learning-Based Image Compression”, IEEE International Workshop on Multimedia Signal Processing (MMSP), 2018.
  26. Y.K. Lee, Y.H. Lee, “Implementation of Moving Object Recognition based on Deep Learning”, Journal of the Semiconductor & Display Technology, vol.17, no.2, 2018.
  27. H.C. Ahn, Y.H. Lee, “A Research of CNN-based Object Detection for Multiple Object Tracking in Image”, Journal of the Semiconductor & Display Technology, vol.18, no.3, pp.110-114, 2019.

---

접수일: 2023년 3월 16일, 심사일: 2023년 3월 22일,  
 게재확정일: 2023년 3월 23일