특집논문 (Special Paper)
방송공학회논문지 제29권 제6호, 2024년 11월 (JBE Vol.29, No.6, November 2024)
https://doi.org/10.5909/JBE.2024.29.6.981
ISSN 2287-9137 (Online) ISSN 1226-7953 (Print)

임의 스케일 이미지 초해상화를 위한 효율적 확산 모델 기반 이미지 프라이어 활용 암시적 이미지 함수 학습

황 인 제[a)], 이 우 진[a)], 김 문 철[a)‡]

LIIFusion: Learning Implicit Image Function Using Image Prior Generated by an Efficient Diffusion Model for Arbitrary-Scale Image Super-Resolution

Inje Hwang[a)], Woojin Lee[a)], and Munchurl Kim[a)‡]

요 약

이미지 초해상화(Super-Resolution, SR)는 저해상도(Low-Resolution, LR) 이미지에서 고해상도(High-Resolution, HR) 이미지를 복원하는 작업을 목표로 한다. 기존의 이미지 SR 방법은 고정된 스케일에 제한되는 경우가 많아, 원하는 스케일이 사전에 알려지지 않았을 때 최적의 결과를 얻기 어려울 수 있다. 본 논문에서는 암시적 신경 표현(Implicit Neural Representation, INR)과 확산 모델(Diffusion Model, DM)을 결합하여 임의의 스케일로 유연한 SR을 수행하는 효과적인 새로운 SR 방법인 LIIFusion을 제안한다. LIIFusion은 다음과 같은 방법을 사용해 연산 복잡도를 줄이면서도 임의 스케일의 SR 이미지에서 그럴듯한 세부 특징을 생성하였다. (1) 이미지를 해상도가 없는 잠재 벡터로 표현해 잠재 공간에서 확산 과정을 수행한다. 확산 과정을 통해 샘플링된 잠재 벡터는 HR 이미지 정보를 담고 있는 프라이어(Prior)로, LR 이미지 특징맵에 HR 정보를 주입하는데 사용된다. 벡터는 해상도가 없기 때문에 해당 방법은 일반적인 잠재 공간 확산 과정보다도 더 낮은 연산 복잡도를 가진다. (2) 프라이어가 주입된 특징맵을 간접적으로 참고할 수 있는 시프트 윈도우 크로스 어텐션(Shifted Window Cross-Attention)을 활용하여 공간 정보를 프라이어가 주입되지 않은 특징맵에서 가져오도록 네트워크를 설계하였다. 프라이어는 해상도가 없기 때문에 프라이어가 주입된 특징맵은 HR 이미지에 대한 정보는 있으나, 공간 정보가 훼손되었을 가능성이 높다. 따라서 프라이어가 주입된 특징맵을 직접 사용하기보다 이를 참고해 고품질의 정보를 본래 특징맵에서 뽑아내는 것이 더 효과적이다. 시프트 윈도우 크로스 어텐션을 통해 나온 고품질의 특징맵은 이후 INR인 조건부 LIIF에 입력되어 연속적인 이미지 함수를 학습하는데 도움을 준다. 연속 이미지 함수는 원하는 픽셀의 위치와 크기를 주면 해당 픽셀의 RGB 값을 출력하기 때문에 어떤 스케일이든 SR이 가능하다. 앞서 언급한 방법들 덕분에 LIIFusion은 전통적인 이미지 공간에서의 확산 모델보다 더 효율적이고 확장 가능하다. 또한 다양한 데이터셋에 대한 실험 결과, LIIFusion은 최신 SR 방법들 대부분을 성능 비교에서 능가하였다.

Abstract

Image Super-Resolution (SR) is a task which aims to reconstruct a High-Resolution (HR) image from a Low-Resolution (LR) image. Existing SR methods are often limited to a fixed scale, which can lead to suboptimal results when the desired scale is not known in advance. This paper proposes a novel and effective SR method, called LIIFusion, that combines Implicit Neural Representation (INR) and a Diffusion Model (DM) to achieve flexible SR with arbitrary scales. LIIFusion employs the following methods to generate plausible details in super-resolution (SR) images at arbitrary scales while reducing computational complexity. (1) The image is represented as a latent vector without spatial resolution, and the diffusion process is carried out in the latent space. The latent vector sampled through the diffusion process serves as a Prior that contains HR image information, used to inject

HR information into the LR image feature map. Since the vector does not have a resolution, this approach has lower computational complexity compared to typical latent space diffusion processes. (2) The network is designed to retrieve spatial information from the feature map that has not had the Prior injected, using Shifted Window Cross-Attention, which allows for indirect reference to the Prior-injected feature map. Due to the lack of resolution, the feature map with the Prior injected may contain HR information but could have damaged spatial information. Thus, instead of directly using the Prior-injected feature map, it is more effective to refer to it and extract high-quality information from the original feature map. The high-quality feature map derived through Shifted Window Cross-Attention is then input into the Conditional LIIF, an INR, which aids in learning a continuous image function. Since the continuous image function can output the RGB value for a given pixel's position and size, SR is achievable at any scale. Thanks to the aforementioned methods, LIIFusion is more efficient and scalable compared to conventional diffusion models in the image space. Additionally, experimental results on various datasets demonstrate that LIIFusion outperforms most state-of-the-art SR methods in terms of performance.

Keyword : Arbitrary-Scale SR, INR, Latent Space Diffusion, Prior

Ⅰ. 서 론

이미지 초해상화(SR)는 저해상도(LR) 이미지를 고해상도(HR) 이미지로 복원하는 과정이다. SR은 의학, 보안, 컴퓨터 비전 등 다양한 응용 분야에서 사용되며 중요한 역할을 한다. 그러나 대부분의 기존 연구는 고정된 스케일에 대해서만 효과적인 SR을 수행한다. 그 결과, 훈련 시 사용된 스케일과 일치하지 않는 경우 최적의 품질을 얻기 어렵다[17].

이 문제를 해결하기 위해 암시적 신경 표현(Implicit Neural Representation, INR) 기반 방법인 LIIF[5]와 GAN (Generative Adversarial Network) 기반 방법인 GCFSR[6]이 제안되었다. INR 기반 방법들은 이미지를 암시적인 연속 표현 함수로 학습하고, 각 픽셀 좌표마다 RGB 값을 구하는 방식으로 SR을 시도한다. 그러나 회귀 손실(Regression Loss) 함수로 학습된 LIIF는 오버스무딩(Over-Smoothing)된 가장자리와 부자연스러운 결과를 보여준다. 또한, GCFSR도 유사하게 특정 스케일에 맞추기 위해 특징 조절을 시도하지만, 이미지의 스케일이 커질수록 이미지에 어울리지 않는 어색한 결과를 생성한다. 더욱이, GAN은 모드 붕괴 문제 같은 학습의 어려움이 있는 것으로 잘 알려져 있다.

한편, 확산 모델(DM)은 최근 SR 분야에서 많이 주목 받기 시작하였다. DM 기반 방법들은 DM의 우수한 데이터 분포 추정 성능을 바탕으로 SR 분야에서 뛰어난 결과를 보여주고 있다[15-17,19]. 그러나 대부분의 DM 방법들은 고정 스케일의 SR에만 우수한 성능을 보인다. IDM[17]은 INR과 DM을 결합하여 이 문제를 해결했다. IDM은 디코더(Decoder)에 INR을 적용하여 고정된 스케일이 아닌 임의 스케일의 이미지에서 확산 과정을 수행한다.

비록 IDM[17]이 사실적인 세부 특징을 잘 표현하는 우수한 SR 결과를 보여주었지만, 두 가지 한계점이 있다. 첫째, IDM은 이미지 공간에서 확산 과정을 수행하기 때문에 연산 복잡도가 높은 편이다. 이는 이미지의 분포를 추정하기 위해서는 SR된 이미지와 동일한 해상도에서 확산 과정이 이루어져야 하기 때문이다. SR 스케일 값이 높을수록 연산

a) 한국과학기술원 전기및전자공학부(KAIST Electrical Engineering)
‡ Corresponding Author : 김문철(Munchurl Kim)
E-mail: mkimee@kaist.ac.kr
Tel: +82-42-350-7419
ORCID: https://orcid.org/0000-0003-0146-5419
※ 이 논문의 연구 결과 중 일부는 한국방송·미디어공학회 2024년 하계학술대회에서 발표한 바 있음.
※ 이 논문은 2023년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임 (No. RS-2022-00144444,딥러닝 기반 정적 및 동적 장면의 공간 영상 표현 학습 및 렌더링 연구).
※ This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No. RS-2022-00144444, Deep Learning Based Visual Representational Learning and Rendering of Static and Dynamic Scenes).
· Manuscript October 23, 2024; Revised October 24, 2024; Accepted November 5, 2024.

복잡도도 그에 따라 증가한다. 또한 이미지 공간에서 확산 과정을 수행하면 디노이징(Denoising)을 위한 많은 단계가 필요하여 샘플링 시간이 증가한다. 둘째, IDM은 비효율적인 연산을 수행한다. 일반적인 SR 수행은 LR 이미지에서 시작해 HR 이미지를 복원한다. 그러나 IDM은 HR 이미지 생성을 노이즈(Noise)에서 시작한다. 이는 이미 LR 이미지 내에 존재하는 콘텐츠(영상 정보)를 복원하기 위해 불필요한 연산을 수행하는 것이며, LR 이미지의 컨텍스트(Context)에 맞지 않는 디테일(Fake Details)을 생성할 가능성이 있다[19].

이 논문에서는 임의 스케일 SR을 가능하게 하는 LIIFusion 프레임워크를 제안한다. 제안한 LIIFusion은 IDM과 유사하게 INR과 DM을 결합하지만, 이미지 공간 대신 잠재 공간에서 확산 과정을 수행하는 잠재 공간 확산(Latent Space Diffusion) 방법을 사용한다. 해당 방법은 높은 해상도의 이미지를 낮은 해상도의 잠재 표현으로 바꾸어 확산 과정을 수행함으로써, 기존 이미지 공간 확산 방법에 비해 낮은 연산 복잡도와 적은 디노이징 횟수를 가진다. LIIFusion은 기존 잠재 공간 확산 방법에서 더 나아가 해상도가 있는 잠재 표현 대신 해상도가 없는 잠재 벡터에 대해 잠재 공간 확산 과정을 수행한다. 그 결과 더 낮은 연산 복잡도를 갖는 동시에 DM과 다른 모듈을 공동으로 학습(Joint Learning)할 수 있을 정도로 디노이징 단계를 크게 줄였다. LIIFusion에서 잠재 벡터는 HR 이미지의 정보를 갖고 있는 프라이어로 활용된다. DM을 사용하여 프라이어의 분포를 학습하고, HR 정보가 담긴 프라이어를 학습한 분포로부터 샘플링해 LR 이미지 특징맵에 주입함으로써 LR 이미지를 SR한다. 고정된 스케일로만 SR을 수행하는 DiffIR[19]과 달리, LIIFusion은 임의 스케일에 대해 SR을 수행하기 때문에 더 정교한 프라이어 활용 방법이 필요하다. 공간 정보가 없는 프라이어를 특징맵에 주입하면 특징맵에 존재하는 공간 정보가 훼손될 수 있다. 따라서 프라이어가 주입된 특징맵을 SR에 바로 사용할 경우 아티팩트(Artifact)가 발생할 가능성이 높다. 이를 막기 위해 LIIFusion에서는 시프트 윈도우 크로스 어텐션을 사용한다. 프라이어가 주입된 특징맵을 쿼리(Query)로 두고 프라이어가 주입되지 않은 특징맵을 키(Key)와 밸류(Value)로 둬서 프라이어 정보를 참고하되, 공간 정보는 본래 특징맵에서 가져오도록 네트워크를 설계하였다. 본 논문의 주요 기여분은 다음과 같다.

1. 잠재 벡터 확산

IDM은 임의 스케일 SR을 위해 이미지 공간에서 확산 과정을 수행했으나, 해당 방법은 연산 복잡도가 높고 많은 디노이징 단계를 필요로 한다. 본 논문에서는 확산 과정을 잠재 공간, 특히 해상도가 없는 벡터를 활용해 수행함으로써 연산 복잡도와 디노이징 단계를 크게 줄였다. 또한 노이즈 대신 LR 이미지로부터 HR 이미지를 만들기 때문에 효율적인 연산을 수행하며, LR 이미지의 컨텍스트에 적합한 디테일을 생성한다.

2. 프라이어 간접 활용 구조

임의 스케일 SR은 모든 스케일에 대해 동일한 프라이어를 사용하기 때문에 고정 스케일 SR에 비해 더 정교한 프라이어 활용 방법이 필요하다. 공간 정보가 없는 프라이어를 특징맵에 주입하면 특징맵의 공간 정보가 훼손될 수 있으며, 이를 SR에 바로 사용하면 아티팩트가 생길 가능성이 높다. 따라서 본 논문에서는 시프트 윈도우 크로스 어텐션을 활용해 프라이어를 간접적으로 참고하고, 공간 정보는 프라이어를 주입하지 않은 본래 특징맵에서 가져오도록 네트워크를 설계했다. 또한 관련 실험을 통해 해당 방법이 유효함을 확인하였다.

II. 관련 연구

1. 회귀 기반 이미지 초해상화

전통적으로 이미지 SR은 쌍입방(Bicubic) 보간법과 같은 수학적 방법을 기반으로 수행되었다. 그러나 SRCNN[3]이 컨볼루션 신경망(CNN)의 가능성을 보여준 이후, 많은 CNN 기반 이미지 SR 모델들이 제안되었다. 예를 들어, EDSR[12]은 모델의 값 범위에 유연성을 제공하기 위해 배치 정규화(Batch Normalization)를 제거하는 방법을 제안하였다. 또한, RCAN[13]은 중요한 특징에 집중하기 위해 채널

어텐션(Channel-Attention)을 도입하였다. CNN 기반 모델들은 SR에서 주목할 만한 결과를 보여주었지만, 위치적으로 떨어진 픽셀 간의 연관성을 만들 수 없었다. 그 결과, 트랜스포머 기반 구조가 이미지 SR 작업에서 효과적인 대안이 되었다. 예를 들어, SwinIR[11]은 셀프 어텐션(Self-Attention)의 연산 복잡도를 줄이기 위해 시프트 윈도우 셀프 어텐션(Shifted Window Self-Attention)을 도입하여 이미지 SR에서 인상적인 결과를 달성하였다. HAT[4]은 더 많은 픽셀을 활용하는 것을 이미지 SR의 핵심으로 제안하며, 채널 어텐션과 시프트 윈도우 셀프 어텐션을 결합하였다.

2. 확산 기반 이미지 초해상화

최근 확산 모델을 이미지 복원 문제에 적용하는 연구가 다수 진행되고 있는데, 그 중 이미지 SR에 적용하는 연구 역시 활발히 진행되고 있다. 예를 들어, SR3[16]는 이미지 공간에서 조건부 확산 과정을 적용하여 자연스러운 SR 결과를 보여주었다. 그러나 이 과정은 이미지 공간에서 수행되기 때문에 연산 복잡도가 높다. 이 문제를 해결하기 위해 LDM[15]은 잠재 공간에서의 확산 모델을 제안하였다. 이 접근 방식에서는 이미지가 잠재 표현으로 인코딩되고, 확산 과정이 잠재 공간에서 수행된다. 결과적으로 LDM은 SR3에 비해 연산 복잡도를 낮추면서도 비슷한 수준의 SR 결과를 보이는데 성공하였다. DiffIR[19]은 노이즈로부터 SR 이미지를 생성하는 기존 방법의 비효율성을 지적하며, 잠재 표현을 이미지 복원의 프라이어로 활용하였다. 해상도를 가진 LDM의 잠재 표현과 달리 DiffIR은 해상도가 없는 벡터를 잠재 표현으로 활용하여 연산 복잡도를 더욱 낮추었으며, 더 적은 모델 파라미터만으로 LDM의 이미지 SR 성능을 능가하였다.

3. 임의 스케일 이미지 초해상화

앞서 언급된 모든 모델들은 고정된 스케일로만 이미지 SR을 최적으로 수행할 수 있다. 그러나 임의의 스케일로 이미지 SR을 수행하려는 몇 가지 시도가 있었다. 예를 들어, LIIF[5]는 이미지의 연속적인 표현 함수를 학습하여 훈련 중에 보지 못한 스케일로 이미지를 SR할 수 있다. 그러나 LIIF는 회귀 손실 함수로 학습되었기 때문에 자연스러운 SR 결과를 생성하는데 한계가 있었다. 또 다른 시도로는 GAN을 사용하는 방법으로서, GCFSR[6]은 인코더(Encoder)와 생성자(Generator)의 특징을 조절하여 해당 스케일에 더 적합한 특징을 만들어 SR을 수행하였다. 적절한 스케일에서는 자연스러운 결과를 보여주었지만, 이미지에 비해 스케일 값이 너무 큰 경우 여러 아티팩트가 발생하는 부자연스러운 결과를 생성하였다. 이는 GCFSR이 이미지의 연속적인 표현 함수를 학습할 수 있지만, LIIF의 표현 함수만큼 정확하지 않음을 의미한다. IDM[17]은 확산 모델과 LIIF를 결합하여 임의의 스케일에서 사실적인 이미지를 생성하였다. GCFSR과 유사하게 조건부 특징과 생성된 특징을 조절한 후, 여러 암시적 신경 표현을 사용하여 추정된 노이즈를 SR하였다. IDM은 SR3[16]를 큰 차이로 능가하고 LIIF에 비해 자연스러운 SR 이미지를 보여주었지만, DiffIR[19]에서 지적한 연산 복잡도 및 컨텍스트에 맞지 않는 디테일 복원 문제를 해결하지 못하였다.

III. 제안 방법

제안 방법인 LIIFusion 프레임워크를 설명하기 앞서, 사용될 기호를 설명한다. $\mathbf{z}\in\mathbb{R}^{4C}$는 이미지 복원을 위한 프라이어를 나타내며, $\mathbf{M}\in\mathbb{R}$은 EDSR-Baseline[12] 인코더에서 출력된 인코딩된 이미지(특징맵)를 나타낸다. 특징맵 $\mathbf{M}$은 프라이어 $\mathbf{z}$와 조건부 암시적 이미지 표현 함수를 사용해 임의의 스케일로 SR된다. 스케일 팩터값 $s\in\mathbb{R}_{\geq 1}$은 매 반복(Iteration)마다 특정 범위 내에서 무작위로 선택한다.

제안하는 모델의 전체 구조는 <그림 1>에 나와 있다. LIIFusion은 다음과 같이 두 단계로 훈련한다. (1) 조건부 암시적 이미지 표현 함수를 학습하기 위해 첫 번째 단계에서는 EDSR-Baseline 인코더, 프라이어 인코더(Prior Encoder), 프라이어 인젝터(Prior Injector), 그리고 조건부 LIIF(Conditional LIIF)를 훈련시킨다. 이 단계에서는 프라이어 인코더에서 $\mathbf{z}$를 얻는다. (2) 두 번째 단계에서는 사전 훈련된 조건부 LIIF와 EDSR-Baseline 인코더, 그리고 새롭게 추가된 확산 모델과 조건 인코더(Condition Encoder)를

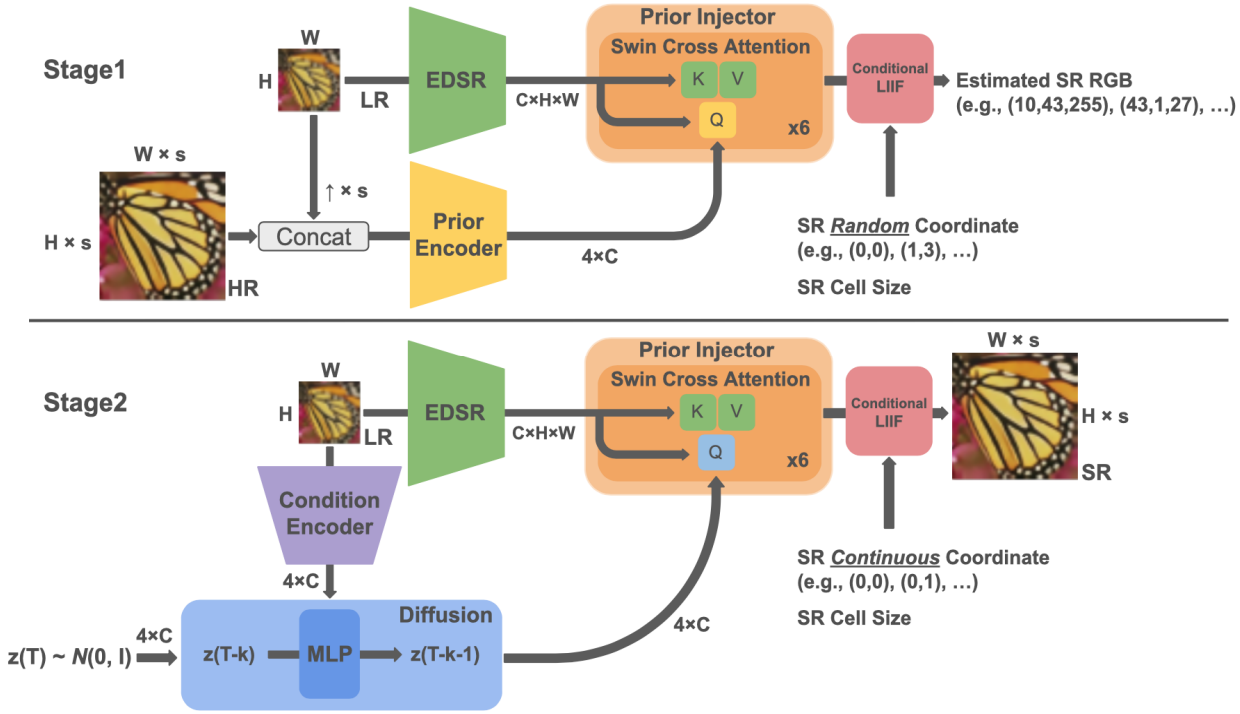


그림 1. LIIFusion의 전체 구조. 1단계의 프라이어 인코더는 2단계에서 고품질 프라이어를 샘플링하기 위해 확산 모델로 대체된다. 2단계의 EDSR-Baseline 인코더, 프라이어 인젝터, 그리고 조건부 LIIF는 1단계에서 사전 학습된다.
Fig. 1. Overall Structure of LIIFusion. The Prior Encoder in stage 1 is replaced to the Diffusion Model in stage 2 to sample high quality Prior. In stage 2, EDSR-Baseline Encoder, Prior Injector, and Conditional LIIF are pretrained in stage 1.

동시에 훈련한다. 첫 번째 단계와 달리, 이번에는 확산 모델에서 $\mathbf{z}$를 얻는다. 사전 학습된 프라이어 인코더에서 HR 이미지 정보가 담긴 $\mathbf{z}_0$를 얻어 확산 모델을 훈련하는데 사용한다. 조건부 잠재 벡터 $\mathbf{z}_c$는 조건 인코더에서 출력되어 확산 과정에 주입된다. 그 결과, 확산 모델은 프라이어의 분포를 학습하고, 이미지의 세부 사항을 재구성하는 데 도움을 주는 고품질의 프라이어를 샘플링한다.

1. 조건부 암시적 이미지 표현 함수 학습

<그림 2>는 첫 번째 단계의 학습 과정을 보여준다. 우선 EDSR-Baseline 인코더는 LR 이미지 정보를 가진 특징맵 $\mathbf{M}$을 LR 이미지로부터 추출한다. 프라이어 인코더는 DiffIR[19]의 CPEN과 유사한 구조를 가지고 있으며, LR-HR 이미지 쌍을 받아 HR 이미지 정보가 담긴 프라이어 $\mathbf{z}\in\mathbb{R}^{4C}$를 추출한다. 이때 LR 이미지를 HR 이미지와 동일한 해상도로 맞추기 위해 쌍입방 업샘플링을 수행하여 LR$\uparrow$을 만든다. 프라이어 인젝터는 특징맵 $\mathbf{M}$과 프라이어 $\mathbf{z}$를 받아 HR 이미지 정보가 들어있는 고품질의 특징맵 $\mathbf{M}''$을 출력한다. 프라이어 $\mathbf{z}$는 프라이어 인젝터 내부의 MLP를 통과하며 $\mathbf{z}_a\in\mathbb{R}^C$와 $\mathbf{z}_b\in\mathbb{R}^C$로 분리된다. 벡터 $\mathbf{z}_a$는 $\mathbf{M}$에 요소별(Element-Wise)로 곱해지고 $\mathbf{z}_b$는 요소별로 더해져 조건부 특징맵 $\mathbf{M}'\in\mathbb{R}^{C\times H\times W}$를 생성한다:

$$\mathbf{z}=\text{PriorEncoder}(\text{Concat}(\text{LR}\uparrow,\text{HR})), \tag{1}$$

$$\mathbf{z}_a,\mathbf{z}_b=\text{MLP}(\mathbf{z}), \tag{2}$$

$$\mathbf{M}'=\mathbf{z}_a\odot\mathbf{M}\oplus\mathbf{z}_b. \tag{3}$$

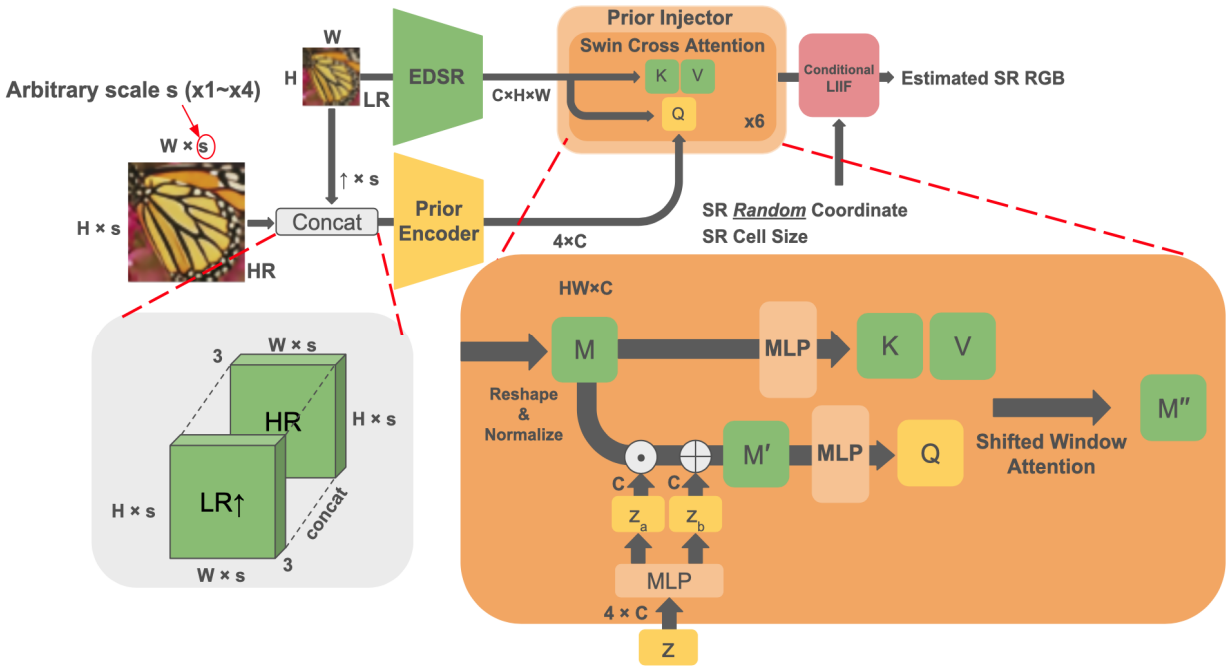


그림 2. 첫 번째 단계의 구조. EDSR-baseline 인코더, 프라이어 인코더, 프라이어 인젝터, 그리고 조건부 LIIF가 학습된다. 프라이어 인젝터는 아티팩트를 방지하기 위해 시프트 윈도우 크로스 어텐션을 수행한다.
Fig. 2. Structure of Stage 1. We train EDSR-Baseline Encoder, Prior Encoder, Prior Injector, and Conditional LIIF. Prior Injector performs Shifted Window Cross-Attention to avoid artifacts.

여기서 요소별 곱셈은 $\odot$로, 요소별 덧셈은 $\oplus$로 표시했다. Concat(A,B)는 이미지 A와 이미지 B를 포개어 구성하는 결합 오퍼레이터를 나타낸다. 만약 $\mathbf{z}_a$를 분산으로 보고 $\mathbf{z}_b$를 평균으로 본다면, 식 (3)은 StyleGAN[18]의 AdaIN과 유사하다. 그러나 StyleGAN2[9]에서 지적했듯이, 특징맵 $\mathbf{M}$의 직접적인 수정은 부자연스러운 아티팩트를 생성할 수 있으며, 프라이어는 해상도가 없는 벡터이기 때문에 특징맵에 주입될 경우 공간정보를 훼손할 가능성이 있다. 이러한 아티팩트를 피하기 위해, 제안 방법에서는 <그림 2>에서 보듯이 조건부 특징맵 $\mathbf{M}'$과 원본 특징맵 $\mathbf{M}$ 사이에 시프트 윈도우 크로스 어텐션을 수행한다. 이 방법은 쿼리 Q만 $\mathbf{M}'$에서 오고, 키 K와 밸류 V는 $\mathbf{M}$에서 오는 간접적인 프라이어 주입 방법이다. 이를 통해 HR 이미지의 정보를 참고하되, 공간 정보는 본래 특징맵 $\mathbf{M}$에서 가져올 수 있다. 또한 연산 복잡도를 줄이기 위해 SwinIR[11]의 시프트 윈도우 셀프 어텐션처럼 윈도우 기반 방식의 크로스 어텐션을 수행한다. 프라이어 인젝터에서는 총 6번의 시프트 윈도우 크로스 어텐션이 수행된다. 조건부 LIIF는 프라이어가 주입된 특징맵 $\mathbf{M}''$과 SR 이미지의 픽셀 크기, 그리고 SR 이미지 픽셀의 위치를 받아 해당하는 픽셀의 RGB 값을 출력한다. <그림 3>은 조건부 LIIF가 SR 이미지 픽셀의 RGB 값을 출력하는 과정을 보여준다. 우선 특징맵 $\mathbf{M}''$을 이루고 있는 각각의 잠재 코드 $\mathbf{m}'' \in \mathbb{R}^{C\times1\times1}$은 LIIF[5]의 특징 펼치기(Feature Unfolding)를 통해 인접한 8개의 잠재 코드와 포개어져 새 잠재 코드 $\mathbf{b}\in\mathbb{R}^{9C\times1\times1}$를 형성한다. 잠재 코드 $\mathbf{b}$의 위치가 $\mathbf{x}_b \in \mathbb{R}^{2\times1\times1}$이고 SR 이미지 픽셀 위치가 $\mathbf{x}_q \in \mathbb{R}^{2\times1\times1}$일 때, SR 이미지 픽셀은 잠재 코드 $\mathbf{b}$로부터 $\mathbf{d} = \mathbf{x}_q - \mathbf{x}_b$만큼 떨어져 있다. 조건부 LIIF는 잠재 코드 $\mathbf{b}$와 SR 이미지 픽셀 크기 $\mathbf{c}\in\mathbb{R}^{2\times1\times1}$, 그리고 SR 이미지 픽셀과 잠재 코드 $\mathbf{b}$간 상대적 거리 $\mathbf{d}$를 모두 포개어 MLP에 입력해 해당 픽셀의 RGB 값 $\mathbf{p}\in\mathbb{R}^{3\times1\times1}$를 구한다:

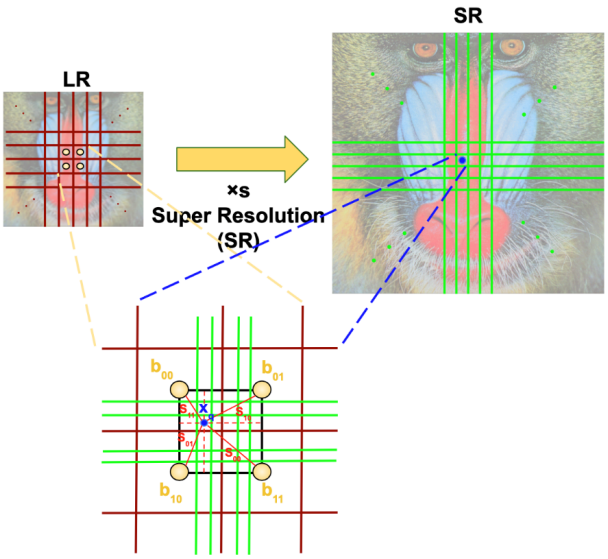


그림 3. 조건부 LIIF의 SR 이미지 픽셀 RGB 값을 구하는 과정. SR 이미지 픽셀에 인접한 잠재 코드 $\mathbf{b}_{00}$, $\mathbf{b}_{01}$,$\mathbf{b}_{10}$, 그리고 $\mathbf{b}_{11}$의 로컬 앙상블을 통해 해당 픽셀의 RGB 값을 구한다.
Fig. 3. The process of obtaining a SR image pixel RGB value of Conditional LIIF. The RGB value of a pixel is obtained through the Local Ensemble of the latent codes $\mathbf{b}_{00}$, $\mathbf{b}_{01}$,$\mathbf{b}_{10}$, and $\mathbf{b}_{11}$ adjacent to the SR image pixel.

$$\mathbf{p}= \mathrm{MLP}\left(\mathrm{Concat}\left(\mathbf{b}, \mathbf{c}, \mathbf{x}_q-\mathbf{x}_b\right)\right). \tag{4}$$

하나의 잠재 코드만 사용하면 주변과 어울리지 않는 RGB 값이 출력될 수도 있다. 주변과 조화로운 픽셀 RGB 값을 출력하기 위해 LIIF의 로컬 앙상블(Local Ensemble)을 사용한다. $\mathbf{x}_q$에 위치한 SR 이미지 픽셀의 RGB 값을 연산할 경우, 우선 해당 픽셀과 인접한 잠재 코드 $\mathbf{b}_{00}$, $\mathbf{b}_{01}$,$\mathbf{b}_{10}$, 그리고 $\mathbf{b}_{11}$를 가지고 각각 RGB 값을 구한다. 이때 $\mathbf{x}_q$에서 수평으로 선을 긋고 수직으로 선을 그으면 <그림 3>처럼 $\mathbf{b}_{00}$, $\mathbf{b}_{01}$,$\mathbf{b}_{10}$, 그리고 $\mathbf{b}_{11}$을 모두 꼭짓점으로 하는 큰 직사각형이 작은 직사각형 4개로 나뉜다. $\mathbf{b}_{00}$, $\mathbf{b}_{01}$,$\mathbf{b}_{10}$, 그리고 $\mathbf{b}_{11}$을 모두 꼭짓점으로 하는 큰 직사각형의 넓이를 s, $\mathbf{b}_{00}$와 $\mathbf{x}_q$를 꼭짓점으로 갖는 작은 직사각형의 넓이를 s_{11}, $\mathbf{b}_{01}$와 $\mathbf{x}_q$를 꼭짓점으로 갖는 작은 직사각형의 넓이를 s_{10}, $\mathbf{b}_{10}$와 $\mathbf{x}_q$를 꼭짓점으로 갖는 작은 직사각형의 넓이를 s_{01}, 마지막으로 $\mathbf{b}_{11}$와 $\mathbf{x}_q$를 꼭짓점으로 갖는 작은 직사각형의 넓이를 s_{00}이라 하자. 이때 조건부 LIIF가 최종적으로 출력하는 픽셀의 RGB 값은 다음과 같다:

$$\mathbf{p}= \sum_{t\in\{00,01,10,11\}} \frac{s_t}{s}\times\mathrm{MLP}\left(\mathrm{Concat}\left(\mathbf{b}_t, \mathbf{c}, \mathbf{x}_q-\mathbf{x}_{b_t}\right)\right). \tag{5}$$

$\mathbf{x}_{b_{00}}$, $\mathbf{x}_{b_{01}}$, $\mathbf{x}_{b_{10}}$, 그리고 $\mathbf{x}_{b_{11}}$는 각각 잠재 코드 $\mathbf{b}_{00}$, $\mathbf{b}_{01}$,$\mathbf{b}_{10}$, 그리고 $\mathbf{b}_{11}$의 위치를 나타낸다. 이렇게 구한 각 픽셀의 RGB 값과 SR 이미지의 각 픽셀 RGB 값의 MSE 손실을 계산하고 이를 역전파(Back Propagation)한다. 이때 연산 복잡도를 줄이기 위해 모든 픽셀 대신 무작위로 샘플링된 소수의 픽셀에 대해서만 손실을 계산한다. 이러한 방법이 가능한 이유는 네트워크가 조건부 암시적 이미지 표현 함수를 학습하기 때문이다. 네트워크를 통해 출력된 각 픽셀의 RGB 값은 조건부 암시적 이미지 함수의 함숫값이며, 소수의 샘플링된 함숫값에 대한 손실을 계산하는 것만으로도 함수를 학습하는 것이 가능하기 때문이다. 이를 통해 EDSR-Baseline 인코더, 프라이어 인코더, 프라이어 인젝터, 그리고 조건부 LIIF의 파라미터가 업데이트된다.

2. 확산 모델로의 전환

이 단계의 주요 목표는 프라이어 인코더를 확산 모델로 대체하는 것이다. DiffIR[19]과 유사하게, 디노이징 횟수를 4회로 설정하고, 확산 과정 중 조건을 제공하기 위해 LR 이미지를 조건 인코더로 인코딩하여 확산 모델에 제공한다. 조건 인코더는 프라이어 인코더와 거의 동일한 구조를 가지고 있지만, LR 이미지만 입력으로 받는다. 디노이징 과정은 DDIM[8] 스타일로 수행된다. 순방향 과정은 프라이어에 노이즈를 추가하며, 다음과 같이 설명할 수 있다:

$$q(\mathbf{z}_t|\mathbf{z}_0)= \mathrm{N}\left(\mathbf{z}_t; \sqrt{\bar{a}_t}\mathbf{z}_0, (1-\bar{a}_t)\mathbf{I}\right). \tag{6}$$

여기서 $\sqrt{\bar{a}_t}=\prod_{i=0}^{t}a_i$이고 a_i는 사전 정의된 분산에서 유래된 값이다. $\mathbf{z}_0$는 사전 학습된 프라이어 인코더에서 출력

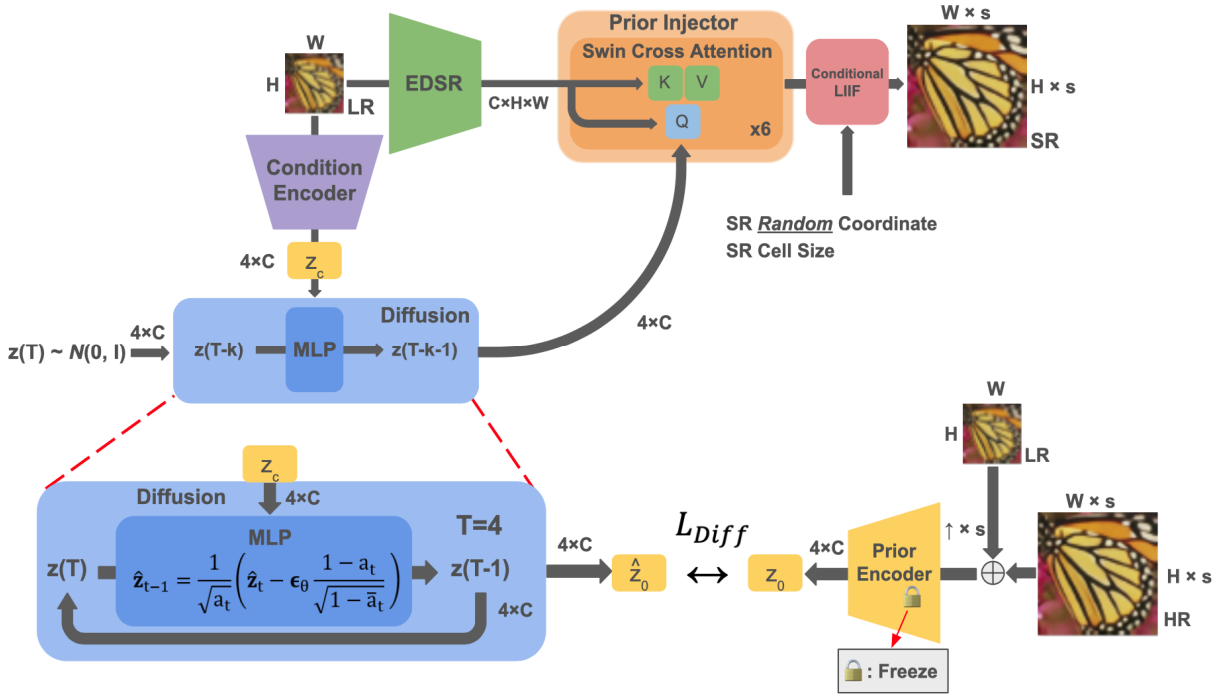


그림 4. 두 번째 단계의 확산 모델 손실 계산 과정. 프라이어 인코더는 확산 모델로 대체된다. 프라이어 인코더는 고정되어 있으며, 확산 모델을 위해 가상 HR 프라이어를 생성한다. 기존의 확산 모델 학습과 달리, 총 4회의 디노이징 횟수로도 충분하기 때문에 확산 과정의 마지막 출력물에 대해 확산 모델 손실을 계산한다.

Fig. 4. Diffusion Loss of Stage 2. Prior Encoder is replaced to the Diffusion Model. Prior Encoder is fixed and generates Pseudo HR Prior for the Diffusion Model. Unlike the conventional diffusion training, we calculate Diffusion Loss only for the last output of the diffusion process as it only needs 4 steps to sample a Prior.

된 프라이어이고 $\mathbf{z}_t$는 순방향 과정 중 t번째 단계에서의 노이즈가 추가된 프라이어를 나타낸다. 역방향 과정은 디노이징 과정으로, 노이즈가 있는 프라이어에서 노이즈를 제거한다. 해당 과정은 다음과 같이 수식으로 나타낼 수 있다:

$$\hat{\mathbf{z}}_{t-1} = \frac{1}{\sqrt{a_t}}\left(\hat{\mathbf{z}}_t - \epsilon_\theta(\hat{\mathbf{z}}_t, \mathbf{z}_c) \times \frac{1-a_t}{\sqrt{1-\bar{a}_t}}\right). \quad (7)$$

여기서 $\epsilon_\theta(\hat{\mathbf{z}}_t, \mathbf{z}_c)$는 확산 모델에서 추정된 노이즈이며, $\hat{\mathbf{z}}_t$와 $\hat{\mathbf{z}}_{t-1}$는 각각 역방향 과정 중 t번째와 t-1번째 단계에서의 프라이어를 나타낸다. 여러 횟수가 필요한 기존 확산 과정과 달리, 제안 방법은 오직 4회의 디노이징 과정만 수행하면 되기 때문에 디노이징 과정 4회를 처음부터 끝까지 모두 수행한다. 따라서 확산 모델의 손실도 디노이징 과정을 마친 후 계산된다. 해당 과정은 <그림 4>에 나와 있다. 확산 모델 손실(Diffusion Loss) 함수는 식 (8)과 같다:

$$L_{Diff} = \frac{1}{4C}\sum_{i=1}^{4C} |\hat{\mathbf{z}}_0^{\,i} - \mathbf{z}_0^{\,i}|. \quad (8)$$

$\hat{\mathbf{z}}_0$는 확산 모델에서 출력된 디노이징 과정을 끝낸 프라이어를 나타낸다. 사전 학습된 프라이어 인코더는 확산 모델을 위한 가상 HR 프라이어(Pseudo HR Prior) $\mathbf{z}_0$를 제공하며, 두 번째 단계에선 업데이트되지 않는다. $\hat{\mathbf{z}}_0^{\,i}$와 $\mathbf{z}_0^{\,i}$는 각각 $\hat{\mathbf{z}}_0$와 $\mathbf{z}_0$의 i번째 채널의 값을 의미한다. 두 번째 단계에서는 확산 과정 횟수가 적은 덕분에 확산 모델을 포함한 제안 모델 내 모든 모듈을 공동으로 동시에 학습시키는 것

$$L_{total} = MSE(I_{SR}, I_{HR}) + \lambda_p Perceptual(I_{SR}, I_{HR}) + \lambda_{GAN} GAN(I_{SR}, I_{HR}) + \lambda_{Diff} L_{Diff}. \quad (9)$$

이 가능하다. 두 번째 단계에서는 첫 번째 단계에서 사전 훈련된 세 가지 모듈인 EDSR-Baseline 인코더, 프라이어 인젝터, 그리고 조건부 LIIF를 사용한다. 프라이어 주입 방법은 첫 번째 단계와 동일하지만, 두 번째 단계에서는 샘플링된 프라이어가 HR 이미지 프라이어 대신 주입된다. MSE 손실 함수만으로는 오버스무딩 문제를 해결하기에 충분하지 않기 때문에 DiffIR처럼 MSE 손실 함수에 사전 학습된 VGG19[23] 네트워크를 이용한 인지 손실(Perceptual Loss) 함수와 GAN 손실 함수를 추가한다. 이 두 새로운 손실 함수를 적용하기 위해선 기존의 샘플링된 RGB 값이 아닌 이미지 자체가 필요하다. 따라서 첫 번째 단계와는 달리 두 번째 단계에서는 모든 픽셀의 RGB 값을 추정한다. 이후 이를 재구성해 이미지 형태로 만듦으로써 두 손실 함수 모두에 입력할 수 있다. SR 이미지를 I_{SR}, HR 이미지를 I_{HR} 이라 할 때 두 번째 단계의 최종 손실 함수는 식 (9)와 같다. λ_p는 0.1, λ_{GAN}와 λ_{Diff}는 5e-2로 설정하였다.

Ⅳ. 실 험

1. 데이터셋

실험을 위해 자연 이미지 DIV2K 데이터셋[1]과 얼굴 이미지 FFHQ 데이터셋[24]을 사용하여 훈련을 진행하였다. 테스트에는 DIV2K 검증 데이터셋과 네 가지 벤치마크(Set5[2], Set14[20], B100[14], Urban100[7]), 그리고 CelebA-HQ[25] 데이터셋을 사용하였다.

2. 구현 세부 사항

제안 방법(LIIFusion)과 비교 방법들의 성능을 공정하게 비교하기 위해 자연 이미지는 LIIF[5]와 동일한 방법으로 학습시켰으며, 얼굴 이미지는 IDM[17]과 동일한 방법으로 학습시켰다. 자연 이미지의 경우 학습을 위해 각 이미지마다 ×1에서 ×4 사이의 무작위 스케일이 균등한 확률로 선택되었다. 스케일은 고정하지 않고 각 반복이 끝날 때마다 다시 샘플링하였다. 테스트의 경우, DIV2K 검증 데이터셋은 나머지 데이터셋과 SR 스케일을 다르게 적용하였다. DIV2K 검증 데이터셋의 경우, 총 8개의 스케일(×2에서 ×30배 사이)을 선택하여 결과를 측정하였다. 이 중 3개의 스케일은 학습 중에 사용하였던 스케일(In-Distribution)이고, 나머지 5개는 훈련 중에 보지 못한 스케일(Out-of-Distribution)이다. 4개 벤치마크의 경우 5개의 스케일(×2에서 ×8배 사이)에서 SR 결과를 측정하였다. 이 중 3개의 스케일은 In-Distribution이고, 나머지 2개는 Out-of-Distribution 스케일이다. 모델에 입력될 LR 이미지는 In-Distribution 스케일을 평가할 경우 데이터셋에서 제공하는 다운샘플링된 ×2, ×3, ×4 스케일 이미지를 사용하였고, Out-of-Distribution을 평가할 경우 원본 이미지를 쌍입방 다운샘플링하여 생성하였다. 얼굴 이미지의 경우 각 이미지마다 ×1에서 ×8 사이의 무작위 스케일을 균등한 확률로 선택했으며, 각 반복이 끝날 때마다 다시 샘플링하였다. 테스트의 경우 CelebA-HQ 데이터셋 중 첫 번째부터 100 번째까지의 이미지들(CelebA-HQ100)에 대해 ×5.3과 ×7의 In-Distribution 스케일과 ×10, ×10.7, 그리고 ×12의 Out-of-Distribution 스케일 SR 결과를 측정하였다. 모든 스케일에 대해 LR 이미지는 원본 이미지를 쌍입방 다운샘플링하여 생성하였다.

첫 번째 학습 단계에서는 Adam 옵티마이저[10]와 MSE 손실 함수를 사용하였다. 초기 학습률은 10^{-4}로 설정했으며, 매 200 에포크 마다 절반으로 줄어들도록 설정하였다. 학습에는 총 1,000 에포크가 소요되었다. 두 번째 학습 단계의 경우, 역시 Adam 옵티마이저를 사용하였으며, MSE 손실 함수와 함께 GAN 손실 함수, 인지 손실 함수, 확산 손실 함수를 사용하였다. 학습률은 3×10^{-5}로 설정했으며, 동일한 학습률로 자연 이미지는 700 에포크 동안, 얼굴 이미지는 1000 에포크 동안 학습을 진행하였다. 두 단계 모두에서 기본 EDSR-baseline[12] 인코더가 특징맵을 추출하는 데 사용되었다. 첫 번째 단계의 훈련은 사전 학습된 모듈 없이 처음부터 진행되었으며, 결과는 PSNR로만 측정하였다. 두 번째 단계의 훈련에서는 첫 번째 단계에서 학습된 파라미터를 사용하였다. 모델은 앞서 언급한 손실 함수로 훈련했으며, PSNR 및 LPIPS로 평가하였다. 마지막으로, 본 논문의 모든 실험에서 12GB NVIDIA TITAN XP 두 개를 사용하였다.

3. 실험 결과 분석

3.1 조건부 암시적 이미지 표현 함수 학습

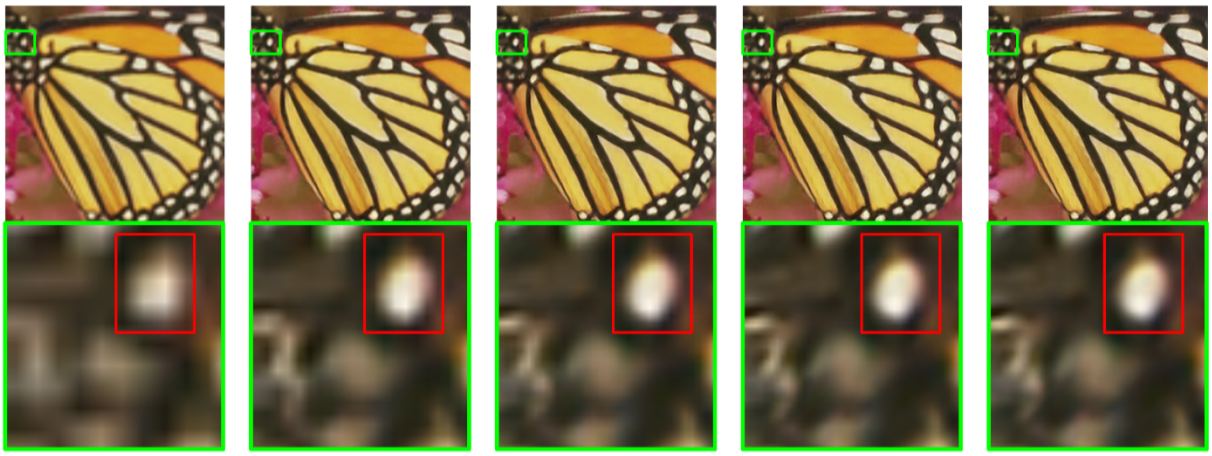
프라이어의 효과를 확인하기 위해 본 논문에서 제안한 LIIFusion과 LIIF-EDSR[5]을 비교하였다. <표 1>과 <표 2>에 나와 있듯이, 제안 방법의 첫 번째 단계 모델은 모든 데이터셋에서 거의 모든 스케일에 대해 LIIF-EDSR 보다 높은 PSNR 성능을 보였다(×2 스케일에서 유사하거나 일부 더 높

표 1. 첫 번째 단계 - DIV2K 검증 데이터셋에 대한 정량적 비교 (PSNR). 가장 우수한 성능은 굵은 글씨로 표시하였다.
Table 1. Stage 1 - Quantitative comparison on DIV2K Validation dataset (PSNR). Best performance is marked in bold.

Methods	In-Distribution			Out-of-Distribution				
	×2	×3	×4	×6	×12	×18	×24	×30
Bicubic	31.07	28.27	26.70	24.87	22.34	21.09	20.29	19.72
LIIF-EDSR[5]	**34.68**	30.97	29.01	26.76	23.72	22.17	21.19	20.49
Ours	34.60	**30.98**	**29.06**	**26.84**	**23.78**	**22.22**	**21.22**	**20.52**

표 2. 첫 번째 단계 - 벤치마크 데이터셋에 대한 정량적 비교 (PSNR). 가장 우수한 성능은 굵은 글씨로 표시하였다.
Table 2. Stage 1 - Quantitative comparison on benchmark datasets (PSNR). Best performance is marked in bold.

Datasets	Methods	In-Distribution			Out-of-Distribution	
		×2	×3	×4	×6	×8
Set5[2]	Bicubic	31.81	28.62	26.70	24.20	22.75
	LIIF-EDSR[5]	**37.99**	34.40	32.24	28.96	26.98
	Ours	37.97	**34.51**	**32.36**	**29.03**	**27.03**
Set14[20]	Bicubic	28.33	25.73	24.25	22.50	21.41
	LIIF-EDSR[5]	33.66	30.34	28.62	26.45	24.94
	Ours	**33.77**	**30.43**	**28.75**	**26.60**	**25.09**
B100[14]	Bicubic	28.28	25.88	24.64	23.22	22.37
	LIIF-EDSR[5]	32.17	29.10	27.60	25.84	24.79
	Ours	**32.18**	**29.15**	**27.67**	**25.94**	**24.88**
Urban100[7]	Bicubic	25.44	23.00	21.69	20.19	19.30
	LIIF-EDSR[5]	**32.15**	28.22	26.15	23.79	22.45
	Ours	32.13	**28.39**	**26.43**	**24.05**	**22.69**



×2 ×4 ×6 ×8 ×10

그림 5. 첫 번째 단계 – Set5에 대한 정성적 결과. 스케일이 증가함에 따라 빨간색 상자 안의 그리드 패턴이 사라지는 것을 확인할 수 있다.
Fig. 5. Stage 1 - Qualitative result on Set5. As increasing scale, the grid pattern in red box vanishes.

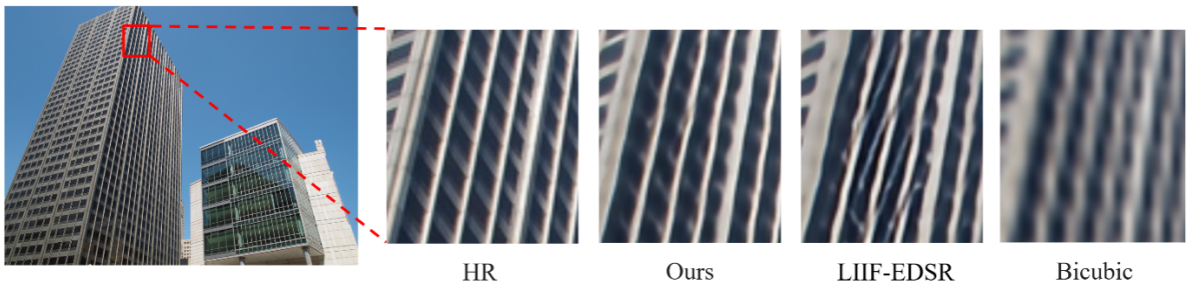


그림 6. 첫 번째 단계 - Urban100 ×4 SR에 대한 정성적 결과. LIIFusion이 LIIF-EDSR에 비해 건물의 선을 더 잘 복원하였다.
Fig. 6. Stage 1 - Qualitative result on Urban100 ×4 SR. LIIFusion restored the lines of the building better than LIIF-EDSR.

은 PSNR 성능을 보임). 제안 모델은 In-Distribution 스케일 뿐만 아니라 Out-of-Distribution 스케일에서도 좋은 성능을 보여주었으며, 이는 제안 모델의 임의 스케일 SR의 일반화 학습 능력을 보여준다. 결과 이미지는 <그림 5>와 <그림 6>에 나와 있다. 이러한 개선은 시프트 윈도우 크로스 어텐션을 통해 모델에 HR 프라이어를 주입함으로써 이루어진 것이며, 프라이어가 이미지 업샘플링 정보를 갖고 있음을 입증한다. 특히 <표 2>를 보면, 제안 모델은 Urban100에서 가장 큰 성능 향상을 보였다.

3.2 확산 모델로의 전환

<표 3>은 DIV2K 검증 데이터셋에서 제안 모델과 확산 모델 기반 고정 스케일 SR 모델인 LDM[15]과 DiffIR[19], 회귀 기반 임의 스케일 SR 모델인 SRNO[22], 그리고 정규화 흐름(Normalizing Flow)을 이용한 임의 스케일 생성형 SR 모델인 LINF[21]를 정량적으로 비교한 수치를 보여준다. 제안 모델은 ×4 스케일에서 DiffIR 보다 낮은 LPIPS 성능을 보였다. 그러나 또다른 확산 모델 기반 고정 스케일 SR 모델인 LDM 보다는 높은 성능을 보였으며, 모든 스케일에서 최신 임의 스케일 SR 모델들보다 더 높은 LPIPS 성능을 보였다. 이러한 경향은 벤치 마크 데이터셋인 Set5, Set14, B100, 그리고 Urban100에서도 이어졌다. <표 4>에서 제안 모델은 모든 임의 스케일 SR 모델보다 높은 LPIPS 성능을 보였다. 확산 모델 기반 고정 스케일 SR 모델들과 비교했을 때, 여전히 DiffIR 보다 낮은 LPIPS 성능을 보였으나, LDM 보다는 높은 LPIPS 성능을 보였다. 제안 모델의 결과 이미지는 <그림 7>, <그림 8>, 그리고 <그림 9>에 나와있다.

<표 5>는 CelebA-HQ 데이터셋 중 첫 번째부터 100 번째까지의 이미지들만 사용한 CelebA-HQ100 데이터셋에 대한 제안 모델 및 임의 스케일 SR모델들[5,17]의 결과이다. DiffIR, LDM, LINF, 그리고 SRNO는 얼굴 데이터셋에 대해 훈련하지 않았기 때문에 비교대상에서 제외했으며, LIIF-EDSR은 얼굴 데이터셋인 CelebA-HQ 데이터셋에 대해 훈련했으나, ×1에서 ×4 사이 스케일로 훈련했기 때문에

표 3. 두 번째 단계 - DIV2K 검증 데이터셋에 대한 정량적 비교 (PSNR/LPIPS). 가장 우수한 성능은 굵은 글씨로 표시하였으며, 두 번째로 우수한 성능은 밑줄로 표시하였다. LDM과 DiffIR은 고정 스케일 SR 모델이기 때문에 ×4 결과만 표기하였다.
Table 3. Stage 2 - Quantitative comparison on DIV2K Validation dataset (PSNR/LPIPS). Best performance is marked in bold, and second-best performance is underlined. Since LDM and DiffIR are fixed-scale SR models, only the ×4 results are listed.

Methods	In-Distribution						Out-of-Distribution									
	×2		×3		×4		×6		×12		×18		×24		×30	
	PSNR	LPIPS	PSNR	LPIPS	PSNR	LPIPS	PSNR	LPIPS	PSNR	LPIPS	PSNR	LPIPS	PSNR	LPIPS	PSNR	LPIPS
LDM[15]	-	-	-	-	23.48	0.2170	-	-	-	-	-	-	-	-	-	-
DiffIR[19]	-	-	-	-	29.13	**0.0871**	-	-	-	-	-	-	-	-	-	-
LINF[21]	33.73	0.0635	29.93	0.1159	27.94	0.1736	26.04	0.2985	23.67	0.4989	22.13	0.6060	21.13	0.6640	20.44	0.6965
SRNO[22]	**34.85**	0.0850	**31.11**	0.1916	**29.16**	0.2636	**26.90**	0.3634	**23.84**	0.5262	**22.28**	0.6114	**21.26**	0.6591	**20.55**	0.6918
Ours	33.39	**0.0334**	29.58	**0.0758**	27.86	0.1238	25.81	**0.2054**	22.99	**0.4014**	21.54	**0.5091**	20.61	**0.5765**	19.94	**0.6294**

표 4. 두 번째 단계 - 벤치마크 데이터셋에 대한 정량적 비교 (PSNR/LPIPS). 최고 성능은 굵은 글씨로, 두 번째로 우수한 성능은 밑줄로 표시하였다. LDM과 DiffIR은 고정 스케일 SR 모델이기 때문에 ×4 결과만 표기하였다.

Table 4. Stage 2 - Quantitative comparison on benchmark datasets (PSNR/LPIPS). Best and second-best results are marked in bold and underlined, respectively. Since LDM and DiffIR are fixed-scale SR models, only the ×4 results are listed.

Datasets	Methods	In-Distribution						Out-of-Distribution			
		×2		×3		×4		×6		×8	
		PSNR	LPIPS	PSNR	LPIPS	PSNR	LPIPS	PSNR	LPIPS	PSNR	LPIPS
Set5[2]	LDM[15]	-	-	-	-	28.18	0.1754	-	-	-	-
	DiffIR[19]	-	-	-	-	31.46	**0.0628**	-	-	-	-
	LINF[21]	37.07	0.0387	33.39	0.0669	31.02	0.0866	27.88	0.1732	26.69	0.2541
	SRNO[22]	**38.15**	0.0564	**34.53**	0.1223	**32.39**	0.1747	**29.05**	0.2550	**27.06**	0.3236
	Ours	36.77	**0.0226**	33.23	**0.0482**	31.18	0.0777	27.87	**0.1440**	26.06	**0.2186**
Set14[20]	LDM[15]	-	-	-	-	25.74	0.2179	-	-	-	-
	DiffIR[19]	-	-	-	-	27.46	**0.1178**	-	-	-	-
	LINF[21]	32.73	0.0709	29.26	0.1296	27.55	0.1889	25.71	0.3136	24.74	0.3955
	SRNO[22]	**33.83**	0.0910	**30.50**	0.2026	**28.78**	0.2787	**26.55**	0.3926	**25.05**	0.4543
	Ours	32.68	**0.0472**	29.23	**0.0957**	27.70	0.1458	25.78	**0.2399**	24.36	**0.3175**
B100[14]	LDM[15]	-	-	-	-	25.19	0.2596	-	-	-	-
	DiffIR[19]	-	-	-	-	26.45	**0.1498**	-	-	-	-
	LINF[21]	31.39	0.1138	28.21	0.1735	26.62	0.2383	25.21	0.3816	24.64	0.4860
	SRNO[22]	**32.27**	0.1443	**29.19**	0.2790	**27.67**	0.3636	**25.91**	0.4848	**24.87**	0.5592
	Ours	31.22	**0.0613**	27.99	**0.1259**	26.66	0.1867	25.17	**0.2869**	24.19	**0.3794**
Urban100[7]	LDM[15]	-	-	-	-	23.39	0.1864	-	-	-	-
	DiffIR[19]	-	-	-	-	26.04	**0.1006**	-	-	-	-
	LINF[21]	31.22	0.0520	27.26	0.1147	25.15	0.1836	23.11	0.3313	22.27	0.4151
	SRNO[22]	**32.62**	0.0597	**28.57**	0.1470	26.50	0.2144	**24.07**	0.3311	**22.69**	0.4180
	Ours	31.31	**0.0354**	27.37	**0.0853**	25.44	0.1384	23.26	**0.2346**	22.01	**0.3238**

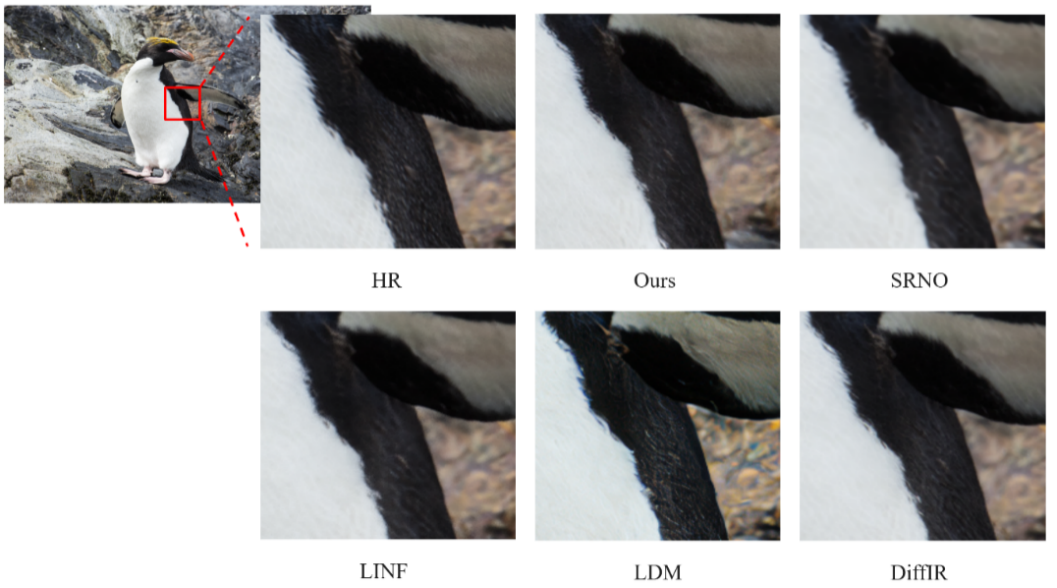


그림 7. 두 번째 단계에서 DIV2K 검증 데이터셋 ×4 SR에 대한 정성적 결과. SRNO는 오버스무딩된 SR 이미지를 보여주었으며, LINF는 노이즈가 있는 SR 이미지를 보여주었다. LDM은 선명하지만 아티팩트가 있는 SR 이미지를 생성했다. 반면 LIIFusion과 DiffIR은 선명하면서 자연스러운 SR 이미지를 생성하였다.

Fig. 7. Qualitative result of stage 2 on DIV2K Validation dataset ×4 SR. SRNO produced an oversmoothed SR image, and LINF showed an SR image with noise. LDM generated a sharp SR image but with some artifacts. In contrast, LIIFusion and DiffIR generated SR images that were both sharp and natural-looking.

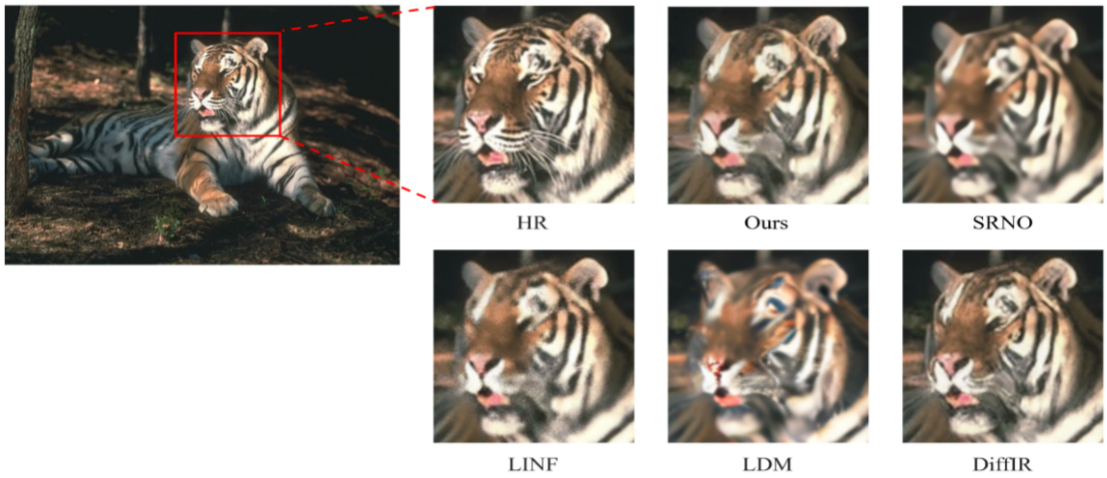


그림 8. 두 번째 단계에서 B100 ×4 SR에 대한 정성적 결과. SRNO는 오버스무딩된 SR 이미지를 보여주었으며, LINF와 LDM은 각각 노이즈가 있는 SR 이미지와 아티팩트가 있는 SR 이미지를 생성했다. 반면 LIIFusion과 DiffIR은 그럴듯한 세부 특징을 가진 자연스러운 SR 이미지를 생성했으며, DiffIR의 결과가 LIIFusion 보다 더 선명했다.

Fig. 8. Qualitative result of stage 2 on B100 ×4 SR. SRNO showed an oversmoothed SR image, while LINF and LDM generated SR images with noise and artifacts, respectively. In contrast, LIIFusion and DiffIR produced natural SR images with plausible fine details, with DiffIR's result appearing sharper than LIIFusion.

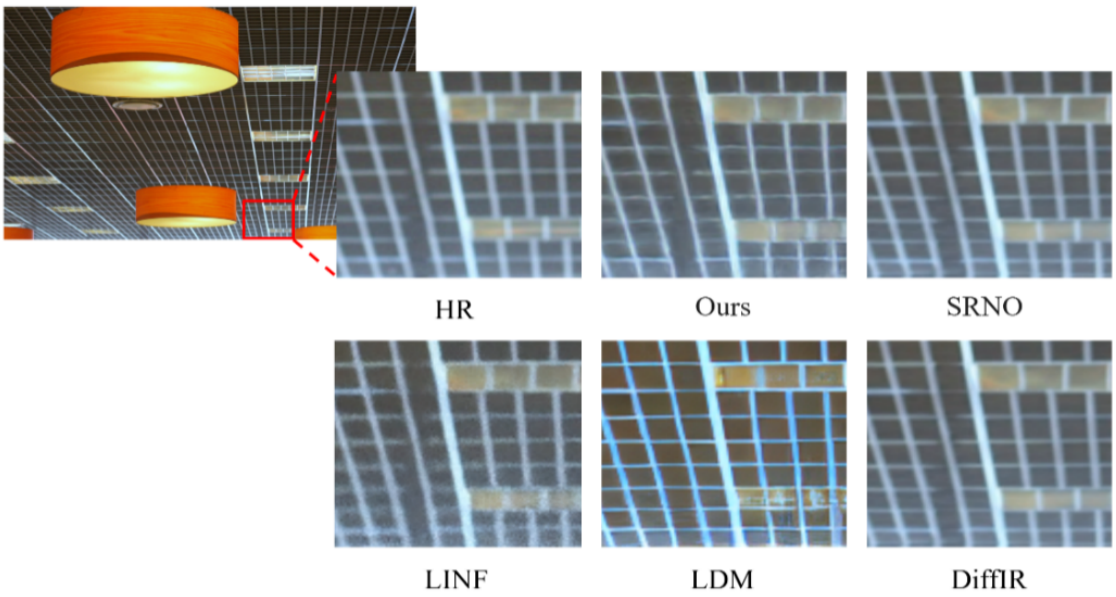


그림 9. 두 번째 단계에서 Urban100 ×4 SR에 대한 정성적 결과. LINF는 노이즈가 있는 이미지를 생성했으며, LDM이 생성한 이미지는 선명하긴 하지만 아티팩트가 있고 색이 달랐다. 반면 LIIFusion 과 SRNO, 그리고 DiffIR은 HR 이미지와 상당히 유사한 SR 결과를 보여주었다. 특히 LIIFusion이 생성한 SR 이미지는 SRNO와 DiffIR의 SR 결과 이미지에 비해 더 선명했다.

Fig. 9. Qualitative result of stage 2 on Urban100 ×4 SR. LINF generated an image with noise, while the image generated by LDM was sharp but had artifacts and color differences. In contrast, LIIFusion, SRNO, and DiffIR produced SR results that were quite similar to the HR image. Notably, the SR image generated by LIIFusion appeared sharper than the SR results from SRNO and DiffIR.

표 5. 두 번째 단계 - CelebA-HQ100 데이터셋에 대한 정량적 비교 (PSNR/LPIPS). 최고 성능은 굵은 글씨로, 두 번째로 우수한 성능은 밑줄로 표시하였다. IDM은 긴 추론 시간으로 인해 논문에서 수치를 발췌하였다.
Table 5. Stage 2 - Quantitative comparison on CelebA-HQ100 dataset (PSNR/LPIPS). Best and second-best results are marked in bold and underlined, respectively. Due to the long inference time of IDM, the figures are extracted from the paper.

Methods	In-Distribution				Out-of-Distribution					
	×5.3		×7		×10		×10.7		×12	
	PSNR	LPIPS	PSNR	LPIPS	PSNR	LPIPS	PSNR	LPIPS	PSNR	LPIPS
LIIF-EDSR[5]	**26.19**	0.1017	**26.19**	0.1571	**25.20**	0.2435	**25.14**	0.2573	**25.04**	0.2823
IDM[17]	23.34	<u>0.0526</u>	23.55	<u>0.0736</u>	23.46	**0.1171**	23.30	**0.1238**	23.06	<u>0.1800</u>
Ours	<u>24.75</u>	**0.0451**	<u>24.17</u>	**0.0727**	<u>23.85</u>	<u>0.1183</u>	<u>23.81</u>	<u>0.1254</u>	<u>23.75</u>	**0.1470**

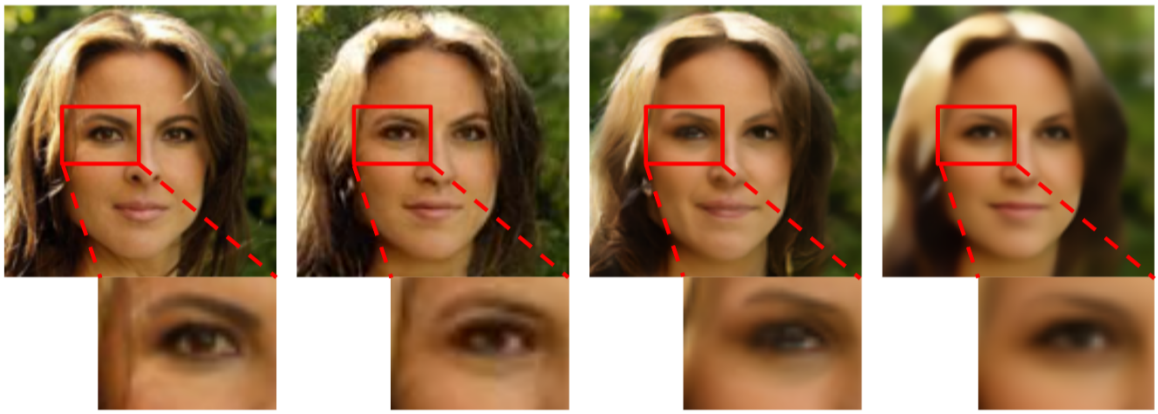


그림 10. 두 번째 단계에서 CelebA-HQ100 ×8 SR에 대한 정성적 결과. LIIF-EDSR은 오버스무딩된 SR 이미지를 보여주었으며, LIIFusion과 IDM은 선명한 SR 이미지를 생성했다. 특히 LIIFusion은 IDM보다 더 LR 이미지의 컨텍스트에 적합한 SR 결과를 보여주었는데, IDM의 SR 이미지는 두 눈동자의 색이 다르지만 LIIFusion의 SR 이미지는 동일한 색이다.
Figure 10. Qualitative result of stage 2 on CelebA-HQ100 ×8 SR. LIIF-EDSR showed an oversmoothed SR image, while LIIFusion and IDM generated sharp SR images. Notably, LIIFusion provided SR results more suited to the context of the LR image than IDM. In the SR image generated by IDM, the colors of the two pupils are different, whereas in the SR image generated by LIIFusion, the colors of both pupils remain identical.

바로 비교할 수 없었다. 그래서 FFHQ 데이터셋에 대해 ×1에서 ×8 사이 스케일로 LIIF-EDSR을 학습시켜 비교하였다. IDM은 특유의 긴 추론 시간으로 인해 직접 성능을 평가하는 대신 논문에 나와있는 수치를 발췌하였다. 제안 모델은 ×12 스케일을 제외한 모든 스케일에서 IDM과 비슷한 LPIPS 성능을 보였으며, ×12 스케일에선 IDM을 크게 앞섰다. 이는 제안 모델이 IDM 보다 더 우수한 일반화 성능을 갖고 있음을 의미한다. 또한 <그림 10>을 보면 제안 모델은 동일한 눈동자 색을 가진 SR 이미지를 생성했지만, IDM은 서로 다른 눈동자 색을 가진 SR 이미지를 생성했다. 이는 제안 모델이 LR 이미지의 컨텍스트에 적합한 디테일을 생성할 수 있다는 것을 의미한다. IDM은 노이즈로부터 SR 이미지를 생성하기 때문에 LR 이미지의 컨텍스트와 맞지 않는 디테일을 생성한다.

마지막으로 <표 6>에서 확산 모델 기반 SR 모델인 LDM, DiffIR, 그리고 IDM과 제안 모델의 평균 추론 시간을 비교하였다. 이미지 공간에서 확산 과정을 수행하는 IDM이 가장 추론 시간이 길었으며, 잠재 공간에서 확산 과정을 수행하지만 해상도를 가진 잠재 표현을 다루는 LDM이 그다음으로 길었다. DiffIR과 제안 모델은 잠재 벡터를 활용한 확산 과정 덕분에 짧은 추론 시간을 보여주었다. 특히 제안 모델은 DIV2K 검증 데이터셋에서 IDM에 비해 약 1,000배 이상 빠른 추론 속도를 보였다.

표 6. 두 번째 단계 - Set5 및 DIV2K 검증 데이터셋에 대한 평균 테스트 시간 비교 (초). 가장 우수한 결과는 굵은 글씨로 표시하였으며, 두 번째로 우수한 결과는 밑줄로 표시하였다.
Table 6. Stage 2 - Average test time comparison on Set5 & DIV2K Validation datasets (Seconds). Best is marked in bold, and second-best is underlined.

Datasets	Methods	Average test time (sec)
Set5[2]	LDM[15]	4.3
	DiffIR[19]	<u>1.5</u>
	IDM[17]	1260
	Ours	**0.9**
DIV2K Validation[1]	LDM[15]	190.6
	DiffIR[19]	<u>3.6</u>
	IDM[17]	6540
	Ours	**3**

4. 주요 컴포넌트 소거 실험 (Ablation Study)

4.1 시프트 윈도우 크로스 어텐션의 효과

제안 방법의 요소별 효과성을 실험하기 위해 DiffIR[19]처럼 프라이어가 주입된 특징맵을 SR에 바로 사용하는 모델을 학습시켰다. 프라이어를 쿼리 Q, 키 K, 그리고 밸류 V에 모두 주입해 시프트 윈도우 셀프 어텐션을 수행하였으며, 이를 QKV 프라이어(QKV Prior) 모델로 표기하였다. 그러나 <표 7>에서 보듯이 밸류 V를 프라이어로 직접 변경하면 성능이 낮아졌다. 이는 해상도가 없는 프라이어로 인해 특징맵의 공간 정보가 훼손되었음을 의미한다. 따라서 제안 방법에서는 프라이어를 쿼리 Q에만 주입하고, 프라이어가

표 7. 첫 번째 단계 - 벤치마크 데이터셋에서 프라이어 활용 방법에 따른 비교 (PSNR). 더 나은 성능은 굵은 글씨로 표시하였다. 평가는 200 에포크를 기준으로 수행하였다.
Table 7. Stage 1 - Comparison according to the Prior utilization on benchmark datasets (PSNR). Better performance is marked in bold. Evaluation was conducted based on 200 epochs

Datasets	Methods	In-Distribution ×2	In-Distribution ×3	In-Distribution ×4	Out-of-Distribution ×6	Out-of-Distribution ×8
Set5[2]	LIIFusion-QKV prior	37.73	34.14	32.00	28.67	26.73
	LIIFusion-Q prior (Ours)	**37.75**	**34.30**	**32.18**	**28.92**	**26.95**
Set14[20]	LIIFusion-QKV prior	33.46	30.21	28.47	26.29	24.81
	LIIFusion-Q prior (Ours)	**33.49**	**30.29**	**28.57**	**26.45**	**24.93**
B100[14]	LIIFusion-QKV prior	32.04	29.01	27.51	25.77	24.72
	LIIFusion-Q prior (Ours)	**32.08**	**29.04**	**27.57**	**25.84**	**24.79**
Urban100[7]	LIIFusion-QKV prior	**31.74**	27.95	25.93	23.62	22.31
	LIIFusion-Q prior (Ours)	31.64	**28.02**	**26.07**	**23.73**	**22.40**

표 8. 첫 번째 단계 - 벤치마크 데이터셋에서 HR Prior 내 Scale 정보의 유무에 따른 비교 (PSNR). 더 나은 성능은 굵은 글씨로 표시하였다. 두 모델은 HR Prior에 Scale 정보 존재의 유무를 제외하고는 모든 조건이 동일하다. 평가는 200 에포크를 기준으로 수행하였다.
Table 8. Stage 1 - Comparison according to the presence or absence of scale factor information within the HR Prior on benchmark datasets (PSNR). Better performance is marked in bold. The two models are equal in all conditions except whether there is information about scale factor in the HR prior or not. Evaluation was conducted based on 200 epochs

Datasets	Methods	In-Distribution ×2	In-Distribution ×3	In-Distribution ×4	Out-of-Distribution ×6	Out-of-Distribution ×8
Set5[2]	LIIFusion-w/ scale	**37.92**	**34.39**	**32.20**	27.00	26.67
	LIIFusion-w/o scale (Ours)	37.75	34.30	32.18	**28.92**	**26.95**
Set14[20]	LIIFusion-w/ scale	**33.56**	**30.34**	**28.61**	25.08	24.75
	LIIFusion-w/o scale (Ours)	33.49	30.29	28.57	**26.45**	**24.93**
B100[14]	LIIFusion-w/ scale	**32.14**	**29.09**	**27.58**	24.99	24.66
	LIIFusion-w/o scale (Ours)	32.08	29.04	27.57	**25.84**	**24.79**
Urban100[7]	LIIFusion-w/ scale	**31.95**	**28.13**	**26.09**	22.25	22.20
	LIIFusion-w/o scale (Ours)	31.64	28.02	26.07	**23.73**	**22.40**

주입된 특징맵과 주입되지 않은 특징맵 사이에서 시프트 윈도우 크로스 어텐션을 수행하였다. 이를 Q 프라이어(Q Prior) 모델로 표기하고, 앞선 실험결과에서 사용된 LIIFusion과 동일한 모델임을 밝힌다. <표 7>에서 확인할 수 있듯이, Q 프라이어 모델의 성능이 QKV 프라이어 모델보다 대부분 PSNR 성능이 더 우수하였다

4.2 스케일 팩터를 반영한 프라이어

<표 1>과 <표 2>에서 ×2 스케일의 경우 Set 14를 제외한 나머지 데이터셋에서는 유의미한 성능 향상이 없거나 미미하지만 저하된 성능을 보였다. 이 문제를 개선하기 위해 프라이어에 스케일 팩터에 대한 정보를 넣어주었다. <표 8>에서 볼 수 있듯이, 스케일 팩터 정보를 사용한 후 In-Distribution 스케일인 ×2~×4에 대해서는 상당한 성능 향상이 있었다. 그러나 Out-of- Distribution 스케일인 ×6 또는 그 이상의 스케일에서는 큰 성능 저하가 발생하였다. 이는 LIIFusion이 스케일 정보에 지나치게 의존하게 되어 일종의 스케일 과적합(Scale Overfitting) 학습이 발생했음을 의미한다. Out-of-Distribution 스케일을 고려하지 않는다면 스케일 정보를 포함하는 프라이어를 사용하는 것이 더 나을 수도 있으나, 임의 스케일 SR에는 적합하지 않기 때문에 최종 모델(LIIFusion) 구현에서는 적용하지 않았다.

Ⅵ. 토론 및 결론

1. 결론

이 논문에서는 효율적이고 자연스러운 임의 스케일 이미지 SR 방법을 제안하였다. 제안 방법은 잠재 공간에서의 확산을 통해 HR 이미지를 대체할 수 있는 프라이어를 생성하고, 구성 요소들이 유기적으로 작동하도록 하는 올인원(All-in-One) 모델을 구현하였다. 많은 기존 모델들이 연산 효율성과 자연스러운 출력 이미지를 보여주었지만, 대부분은 고정된 스케일에서만 잘 수행되었다. 반면, 제안 모델은 연산 효율성과 함께 자연스러운 출력 이미지를 임의의 스케일에서 모두 복원할 수 있음을 보여주었다. 그 결과, IDM[17]과 같은 이미지 공간 확산 모델에 비해 훨씬 빠른 이미지 생성 속도를 제공하며, 이미지의 디테일을 더 충실히 복원하는 장점을 가진다.

2. 한계점 및 향후 과제

우리는 첫 번째 단계에서 프라이어 주입이 대부분의 스케일에서 효과적인 방법임을 입증하였다. 그러나 ×2 스케일에서는 미미한 수준이지만 성능 저하가 있었다. 이는 제안 모델이 비교적 SR을 수행하기 어려운 높은 스케일에 더 집중하여, 낮은 스케일에서는 프라이어가 크게 도움이 되지 않았기 때문으로 추측된다. 주요 컴포넌트 소거 실험에서 모델에 추가적인 스케일 정보를 제공하는 실험을 수행하였으며, 제안 모델은 비로소 낮은 스케일에서도 향상된 결과를 보였다. 그러나 앞서 언급한 방법은 높은 스케일에서 열등한 결과를 보였기 때문에 일반화할 수 없었다. 이는 스케일에 따라 프라이어를 다르게 처리해야 함을 의미할 수 있다. 향후 과제로는 이러한 문제를 해결할 계획이다.

참 고 문 헌 (References)

[1] Eirikur Agustsson and Radu Timofte. Ntire 2017 challenge on single image super-resolution: Dataset and study. In 2017 IEEE Conference on Computer Vision and Pattern Recogni- tion Workshops (CVPRW), 2017.
doi: https://doi.org/10.1109/CVPRW.2017.150

[2] Marco Bevilacqua, A. Roumy, Christine Guillemot, and Marie-Line Alberi-Morel. Low-complexity single image super-resolution based on nonnegative neighbor embedding. 09 2012.

[3] Xiaoou Tang Chao Dong, Chen Change Loy. Image super-resolution using deep convolutional networks. TPAMI, 2016.
doi: https://doi.org/10.1109/TPAMI.2015.2439281

[4] Xiangyu Chen, Xintao Wang, Wenlong Zhang, Xiangtao Kong, Yu Qiao, Jiantao Zhou, and Chao Dong. Hat: Hybrid attention transformer for image restoration. arXiv preprint arXiv:2309.05239, 2023.
doi: https://doi.org/10.48550/arXiv.2309.05239

[5] Yinbo Chen, Sifei Liu, and Xiaolong Wang. Learning continuous image representation with local implicit image function. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 8628 – 8638, 2021.
doi: https://doi.org/10.1109/CVPR46437.2021.00852

[6] Jingwen He, Wu Shi, Kai Chen, Lean Fu, and Chao Dong. Gcfsr: a generative and controllable face super resolution method without facial and gan priors. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 1889 – 1898, 2022.
doi: https://doi.org/10.1109/CVPR52688.2022.00193

[7] Jia-Bin Huang, Abhishek Singh, and Narendra Ahuja. Single image super-resolution from transformed self-exemplars. In 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 5197 – 5206, 2015.
doi: https://doi.org/10.1109/CVPR.2015.7299156

[8] Song, J., Meng, C., & Ermon, S. Denoising diffusion implicit models. arXiv preprint arXiv:2010.02502, 2020.
doi: https://doi.org/10.48550/arXiv.2010.02502

[9] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of StyleGAN. In Proc. CVPR, 2020.
doi: http://dx.doi.org/10.1109/CVPR42600.2020.00813

[10] Kingma, D. P. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980, 2014.
doi: https://doi.org/10.48550/arXiv.1412.6980

[11] Jingyun Liang, Jiezhang Cao, Guolei Sun, Kai Zhang, Luc Van Gool, and Radu Timofte. Swinir: Image restoration using swin transformer. arXiv preprint arXiv:2108.10257, 2021.
doi: https://doi.org/10.48550/arXiv.2108.10257

[12] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, July 2017.
doi: https://doi.org/10.1109/CVPRW.2017.151

[13] Zhang, Yulun Li, Kunpeng, Li, Kai Wang, Lichen, Zhong, Bineng, and Fu, Yun. “Image Super-Resolution Using Very Deep Residual Channel Attention Networks.” ECCV, 2018.
doi: https://doi.org/10.1007/978-3-030-01234-2_18

[14] Martin, D., Fowlkes, C., Tal, D., and Malik, J. “A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics.” In Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001, volume 2, pages 416 – 423 vol.2, 2001.
doi: https://doi.org/10.1109/ICCV.2001.937655

[15] Rombach, Robin, et al. “High-resolution image synthesis with latent diffusion models.” Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR). 2022.
doi: http://dx.doi.org/10.1109/CVPR52688.2022.01042

[16] Saharia, Chitwan, Ho, Jonathan, Chan, William, Salimans, Tim, Fleet, David J., and Norouzi, Mohammad. “Image super-resolution via iterative refinement.” arXiv:2104.07636, 2021.
doi: https://doi.org/10.48550/arXiv.2104.07636

[17] Zeng, Bohan, Xu, Sheng, Li, Yanjing, Luo, Xiaoyan, Liu, Jianzhuang, Zhen, Xiantong, Gao, Sicheng, Liu, Xuhui, and Zhang, Baochang. “Implicit diffusion models for continuous super-resolution.” In CVPR, 2023.
doi: https://doi.org/10.1109/CVPR52729.2023.00966

[18] Karras, Tero, Laine, Samuli, and Aila, Timo. “A style-based generator architecture for generative adversarial networks.” In CVPR, 2019.
doi: https://doi.org/10.1109/CVPR.2019.00453

[19] Xia, Bin, Zhang, Yulun, Wang, Shiyin, Wang, Yitong, Wu, Xinglong, Tian, Yapeng, Yang, Wenming, and Van Gool, Luc. “DiffIR: Efficient diffusion model for image restoration.” ICCV, 2023.
doi: https://doi.org/10.1109/iccv51070.2023.01204

[20] Zeyde, Roman, Michael Elad, and Matan Protter. “On single image scale-up using sparse-representations.” Curves and Surfaces: 7th International Conference, Avignon, France, June 24-30, 2010.
doi: https://doi.org/10.1007/978-3-642-27413-8_47

[21] Yao, Jie-En, et al. “Local implicit normalizing flow for arbitrary-scale image super-resolution.” Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2023.
doi: http://dx.doi.org/10.1109/CVPR52729.2023.00177

[22] Wei, Min, and Xuesong Zhang. “Super-resolution neural operator.” Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2023.
doi: http://dx.doi.org/10.1109/CVPR52729.2023.01750

[23] S. Karen and Z. Andrew. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556, 2014.
doi: https://doi.org/10.48550/arXiv.1409.1556

[24] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In CVPR, 2019.
doi: https://doi.org/10.1109/CVPR.2019.00453

[25] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. arXiv:1710.10196, 2017.
doi: https://doi.org/10.48550/arXiv.1710.10196

저 자 소 개

황 인 제

- 2023년 2월 : 한국과학기술원 전기및전자공학부 학사
- 2023년 9월 ~ 현재 : 한국과학기술원 전기및전자공학부 석사과정
- ORCID : https://orcid.org/0009-0000-6352-4299
- 주관심분야 : 컴퓨터 비전, 딥러닝, 초해상화, 생성 모델

저 자 소 개

이 우 진

- 2023년 2월 : 한국과학기술원 전기및전자공학부 학사
- 2023년 3월 ~ 현재 : 한국과학기술원 전기및전자공학부 석사과정
- ORCID : https://orcid.org/0009-0000-0937-5907
- 주관심분야 : 컴퓨터 비전, 딥러닝, 객체 탐지, 초해상화, 생성 모델

김 문 철

- 1996년 8월 : University of Florida Electrical and Computer Engineering 박사
- 2001년 2월 ~ 2009년 3월 : 한국정보통신대학교 공학부 조교수/부교수
- 2009년 3월 ~ 현재 : 한국과학기술원 전기및전자공학부 부교수/정교수
- ORCID : https://orcid.org/0000-0003-0146-5419
- 주관심분야 : 컴퓨터 비전, 딥러닝, 영상 처리, 비디오 코딩