NTIRE 2025 Image Super-Resolution ($\times$4) Challenge Factsheet
Learning to upsample for One-step diffusion model

Donghun Ryou[2] Inju Ha[1] Bohyung Han[1,2]
Computer Vision Laboratory, [1]ECE & [2]IPAI, Seoul National University
{dhryou, hij1112, bhhan}@snu.ac.kr

1. Introduction

This factsheet template is meant to structure the description of the contributions made by each participating team in the NTIRE 2025 challenge on image super-resolution ($\times$4).

Ideally, all the aspects enumerated below should be addressed. The provided information, the codes/executables, and the achieved performance on the testing data are used to decide the awardees of the NTIRE 2025 challenge.

Reproducibility is a must and needs to be checked for the final test results in order to qualify for the NTIRE awards.

The main winners will be decided based on overall performance and a number of awards will go to novel, interesting solutions and to solutions that stand up as the best in a particular subcategory the judging committee will decide. Please check the competition webpage and forums for more details.

The winners, the awardees and the top-ranking teams will be invited to co-author the NTIRE 2025 challenge report and to submit papers with their solutions to the NTIRE 2025 workshop. Detailed descriptions are appreciated.

The factsheet, source codes/executables, trained models should be sent to **all of the NTIRE 2025 challenge organizers (Zheng Chen, Jue Gong, Jingkai Wang, Kai Liu, Lei Sun, Zongwei Wu, Yulun Zhang, and Radu Timofte)** by email.

2. Email final submission guide

To: zhengchen.cse@gmail.com
leosun0331@gmail.com
g1017325431@gmail.com
normal.kliu@gmail.com
jingkaiwang100@gmail.com
zongwei.wu@uni-wuerzburg.de
yulun100@gmail.com
Radu.Timofte@uni-wuerzburg.de
cc: your_team_members
Title: NTIRE 2025 Image Super-Resolution ($\times$4) Challenge - TEAM_NAME - TEAM_ID

To get your TEAM_ID, please register at Google Sheet. Please fill in your Team Name, Contact Person, and Contact Email in the first empty row from the top of the sheet. Body contents should include:

a team name
b team leader's name and email address
c rest of the team members
d user names on NTIRE 2025 CodaLab competitions
e Code, pre-trained model, and factsheet download command, e.g. `git clone ...`, `wget ...`
f Result download command, e.g. `wget ...`
- Please provide different URLs in e) and f)

Factsheet must be a compiled pdf file together with a zip with .tex factsheet source files. Please provide a detailed explanation.

3. Code Submission

The code and trained models should be organized according to the GitHub repository. This code repository provides the basis for comparing the various methods in the challenge. **Code scripts based on other repositories will not be accepted.** Specifically, you should follow the steps below.

1. Git clone the repository
2. Put your model script under the `models/[Team_ID]_[Model_Name]` folder
3. Put your pretrained model under the `model_zoo/[Team_ID]_[Model_Name]` folder
4. Modify `model_path` in `test.py`. Modify the imported models
5. `python test.py` (restore images, details in GitHub)
6. `python eval.py` (eval results, details in GitHub)

Please send us the command to download your code, e.g. `git clone [Your repository link]` When submitting the code, please remove the input and output images in the (any) `data` folder to save the bandwidth.

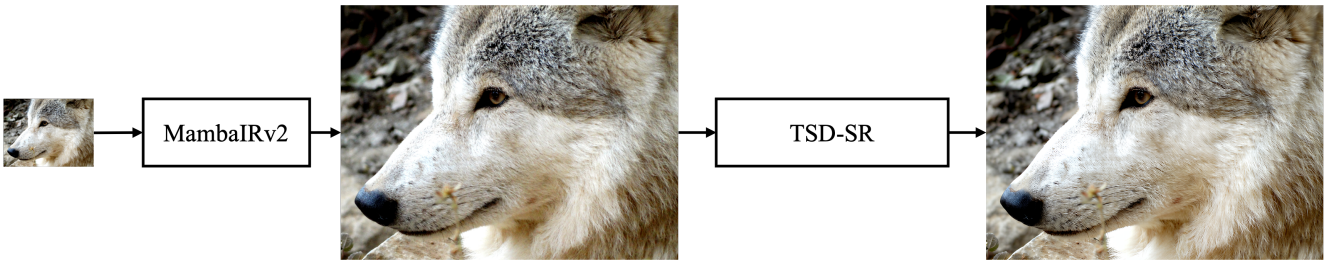

Figure 1. The architecture of our pipeline.

4. Factsheet Information

4.1. Team details

- **Team name**
  SNUCV
- **Team leader name**
  Donghun Ryou
- **Team leader address, phone number, and email**
  Address: 70-15, Gwanak-ro 12-gil, Gwanak-gu, Seoul, Republic of Korea
  Phone number: +82 010-4879-3794
  Email: dhryou@snu.ac.kr
- **Rest of the team members**
  Inju Ha, Bohyung Han
- **Team website URL (if any)**
  https://cv.snu.ac.kr/
- **Affiliation**
  Seoul National University
- **Affiliation of the team and/or team members with NTIRE 2025 sponsors (check the workshop website)**
- **User names and entries on the NTIRE 2025 Codalab competitions (development/validation and testing phases)**
  dhryou
- **Best scoring entries of the team during the development/validation phase**
- **Link to the codes/executables of the solution(s)** https://github.com/dhryougit/NTIRE2025_SR_SNUCV

4.2. Method details

- **General method description (How is the network designed?)**
  As illustrated in Figure 1, we first employ a classic super-resolution (SR) model to perform 4× upsampling of the input image. Subsequently, a one-step diffusion model is applied to further enhance the super-resolution process. For the 4× upsampling model, we adopt MambaIRv2 [3], while the diffusion model utilizes the TSD-SR [2] architecture. To fully leverage the generative prior of the diffusion model, we refrain from additional training and directly utilize its pretrained weights. Instead, we focus on fine-tuning the upsampling model.

- **Training strategy**
  Let us denote the diffusion model as M, the upsampler model as U, the input image as I, and the ground-truth image as I_{gt}. To ensure that the generated output $M(U(I))$ and the intermediate upsampler output $U(I)$ accurately resemble the ground-truth I_{gt}, we define a loss function comprising the LPIPS loss [8], denoted $\mathcal{L}_{\text{LPIPS}}$, and the L1 loss, denoted $\mathcal{L}_1$. These losses are formulated as:

$$\mathcal{L}_{\text{accuracy}} = \mathcal{L}_1(U(I), I_{gt}) + \alpha \cdot \mathcal{L}_{\text{LPIPS}}(M(U(I)), I_{gt}), \tag{1}$$

where $\mathcal{L}_{\text{LPIPS}}$ captures perceptual similarity between the final output and the ground-truth, and $\mathcal{L}_1$ enforces pixel-wise accuracy between the upsampler output and the ground-truth.
In addition, to enhance the perceptual quality of the generated image $M(U(I))$, we utilize a no-reference CLIP-based Image Quality Assessment (IQA) loss [6, 7], denoted $\mathcal{L}_{\text{CLIP}}$. This loss leverages the CLIP model's text encoder, $\text{CLIP}_{\text{text}}$, and image encoder, $\text{CLIP}_{\text{image}}$. We define two text prompts: a positive quality reference, "Good photo", and a negative quality reference, "Bad photo". Their embeddings are computed and normalized as: $T_{\text{pos}} = \text{norm}(\text{CLIP}_{\text{text}}(\text{"Good photo"}))$ and $T_{\text{neg}} = \text{norm}(\text{CLIP}_{\text{text}}(\text{"Bad photo"}))$, where $\text{norm}(\cdot)$ denotes L2 normalization.
The generated image's feature representation is obtained as:

$$F_{\text{pred}} = \text{norm}(\text{CLIP}_{\text{image}}(M(U(I)))). \tag{2}$$

Similarities between the image features and the text em-

beddings are then computed:

$$S_{\text{pos}} = F_{\text{pred}} \cdot T_{\text{pos}}^{\top}, \quad S_{\text{neg}} = F_{\text{pred}} \cdot T_{\text{neg}}^{\top}, \tag{3}$$

where S_{pos} and S_{neg} represent the similarity to the positive and negative references, respectively. The CLIP loss is defined to maximize similarity to "Good photo" and minimize similarity to "Bad photo":

$$\mathcal{L}_{\text{CLIP}} = 1 - S_{\text{pos}} + S_{\text{neg}}. \tag{4}$$

Therefore, the total training loss is:

$$\mathcal{L} = \mathcal{L}_{\text{accuracy}} + \beta \cdot \mathcal{L}_{\text{CLIP}}. \tag{5}$$

This combined loss ensures both perceptual fidelity and structural consistency between the generated output and the ground-truth image. The hyperparameters α and β are set to 0.1 and 0.5, respectively.

- **Training details**
  When training the upsampler MambaIRv2, we utilized the AdamW [5] optimizer with hyperparameters $\beta_1 = 0.9$ and $\beta_2 = 0.999$, and a learning rate of 1×10^{-5}. The ground-truth (GT) images were cropped into patches of size 256×256. Training was conducted over a total of 100,000 iterations with a batch size of 4. The training dataset was composed of a mixture of DIV2K [1] and LSDIR [4] datasets.

References

[1] Eirikur Agustsson and Radu Timofte. Ntire 2017 challenge on single image super-resolution: Dataset and study. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2017. 3

[2] Linwei Dong, Qingnan Fan, Yihong Guo, Zhonghao Wang, Qi Zhang, Jinwei Chen, Yawei Luo, and Changqing Zou. Tsd-sr: One-step diffusion with target score distillation for real-world image super-resolution. *arXiv preprint arXiv:2411.18263*, 2024. 2

[3] Hang Guo, Yong Guo, Yaohua Zha, Yulun Zhang, Wenbo Li, Tao Dai, Shu-Tao Xia, and Yawei Li. Mambairv2: Attentive state space restoration. *arXiv preprint arXiv:2411.15269*, 2024. 2

[4] Yawei Li, Kai Zhang, Jingyun Liang, Jiezhang Cao, Ce Liu, Rui Gong, Yulun Zhang, Hao Tang, Yun Liu, Denis Demandolx, et al. Lsdir: A large scale dataset for image restoration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1775–1787, 2023. 3

[5] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 3

[6] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR, 2021. 2

[7] Jianyi Wang, Kelvin CK Chan, and Chen Change Loy. Exploring clip for assessing the look and feel of images. In *AAAI*, 2023. 2

[8] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018. 2