Appendix of
Unleashing Vanilla Vision Transformer with Masked Image Modeling for Object Detection

Yuxin Fang1*  Shusheng Yang1*  Shijie Wang1*  Yixiao Ge2,3  Ying Shan2,3  Xinggang Wang1†

1School of EIC, Huazhong University of Science & Technology  
2Tencent AI Lab  
3ARC Lab, Tencent PCG

Table 1: COCO object detection and instance segmentation results using Mask R-CNN on COCO val & test-dev set respectively. Their results are consistent.

| Method       | COCO val                          | COCO test-dev                      |
|--------------|-----------------------------------|------------------------------------|
| MIMDET-B     | 51.7 APbbox / 46.1 APmask         | 51.8 APbbox / 46.3 APmask          |
| MIMDET-L     | 54.3 APbbox / 48.2 APmask         | 54.5 APbbox / 48.7 APmask          |

Table 2: Params, FLOPs & ft epochs comparisons with Li et al. [5] using Mask R-CNN.

| Backbone     | params (M) | FLOPs (T) | ft epochs | APbbox | APmask |
|--------------|------------|-----------|-----------|--------|--------|
| Li et al.-B  | 111        | 0.8       | 100       | 50.3   | 44.9   |
| MIMDET-B     | 128        | 0.9       | 36        | 51.7   | 46.1   |
| Li et al.-L  | 331        | 1.9       | 100       | 53.3   | 47.2   |
| MIMDET-L     | 349        | 2.1       | 36        | 54.3   | 48.2   |

A. Appendix

Architecture of ConvStem. We adopt a minimalist Con-vStem design, i.e., by simply stacking 3×3 regular convolutions with a stride of 2 and doubled feature dimensions. Each convolutional layer is followed by a layer normalization [1] and a GELU activation [4]. The detailed configurations are given in Architecture 1.

Hyper-parameters and Model Configurations. Hyper-parameters and model configurations for fine-tuning on the COCO dataset are shown in Table 3. Since the vanilla ViT encoder is already pre-trained while the task layer is trained from scratch, the learning rate of the ViT encoder part is divided by a “lr multiplier” and the learning rate for the task layer is multiplied by a “lr multiplier”.

The implicit reconstruction process of ViT encoder is driven by the supervision from the Mask R-CNN detector.

Results on COCO test-dev set and comparisons with COCO val set results are shown in Table 1, which imply that our models & settings are not biased towards val set.

Feature Visualizations Figure 1 and 2 visualizes some backbone & FPN feature maps with a stride of 4 for both [5] and our MIMDET. The stride-4 backbone feature of [5] is obtained from a stride-16 ViT encoder feature via upsampling

Architecture 1 -ConvStem for ViT-Base (PyTorch Style), which can help preserve low-level details, produce higher resolution hierarchical features for FPN, and introduce 2D inductive biases for the ViT encoder & detector.

# Number of Parameters: 4.1M.
ConvStem(
  ModuleList(
    (0): Sequential(
      (0): Conv2d(3, 96, kernel_size=(3, 3), stride=(2, 2), padding=(1, 1), bias=False)
      (1): LayerNorm2d(96, eps=1e-06, affine=True) & GELU()
    )
    (1): Sequential(
      (0): Conv2d(96, 192, kernel_size=(3, 3), stride=(2, 2), padding=(1, 1), bias=False)
      (1): LayerNorm2d(192, eps=1e-06, affine=True) & GELU() # Input for FPN P2.
    )
    (2): Sequential(
      (0): Conv2d(192, 384, kernel_size=(3, 3), stride=(2, 2), padding=(1, 1), bias=False)
      (1): LayerNorm2d(384, eps=1e-06, affine=True) & GELU() # Input for FPN P3.
    )
    (3): Sequential(
      (0): Conv2d(384, 768, kernel_size=(3, 3), stride=(2, 2), padding=(1, 1), bias=False)
      (1): LayerNorm2d(768, eps=1e-06, affine=True) & GELU()
      (2): Conv2d(768, 768, kernel_size=(1, 1), stride=(1, 1)) # Input for ViT-Base Enc.
    )))

*Equal contribution. † Xinggang Wang (xgwang@hust.edu.cn) is the corresponding author. This work was done when Shusheng Yang was interning at ARC Lab, Tencent PCG.
Table 3: **Hyper-parameters and model configurations for COCO fine-tuning with Mask R-CNN.** We report the average number of FLOPs and inference time for the first 100 images in the COCO val set following [2] on a V100 GPU. Hyper-parameters for Cascade Mask R-CNN and RetinaNet are same as Mask R-CNN.

| Backbone       | lr     | lr multiplier | weight decay | drop path | ft epochs | params (M) | FLOPs (G) | inf. time (s) |
|----------------|--------|---------------|--------------|-----------|-----------|------------|-----------|---------------|
| MIMDet-Base    | $8e^{-5}$ | 2             | 0.1          | 0.1       | 36        | 128        | 933       | 0.29          |
| MIMDet-Large   | $8e^{-5}$ | 3.5          | 0.1          | 0.1       | 36        | 349        | 2082      | 0.58          |

Figure 1: **Feature visualizations and comparisons of some stride-4 backbone and FPN feature maps.** The feature maps of [5] is obtained from our re-implementation which successfully reproduces its reported results.

Using two stride-2 transposed convolutions with $2 \times 2$ kernel. The resulting features suffer from very strong “checkerboard artifacts [6]”. If we look closer, the evidence of ViT attention’s window partition emerges. Thanks to FPN, the noise can be mitigated to some extent. However, many low-level details are still fuzzy. On the other hand, our ConvStem in MIMDet can always produce clear and tidy features, which is beneficial to both the ViT encoder as well as the Mask R-CNN detector.
Figure 2: Feature visualizations and comparisons of some stride-4 backbone and FPN feature maps. The feature maps of [5] is obtained from our re-implementation which successfully reproduces its reported results.
References

[1] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. arXiv preprint arXiv:1607.06450, 2016.

[2] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In ECCV, 2020.

[3] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross B. Girshick. Mask r-cnn. In ICCV, 2017.

[4] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). arXiv preprint arXiv:1606.08415, 2016.

[5] Yanghao Li, Saining Xie, Xinlei Chen, Piotr Dollar, Kaiming He, and Ross Girshick. Benchmarking detection transfer learning with vision transformers. arXiv preprint arXiv:2111.11429, 2021.

[6] Augustus Odena, Vincent Dumoulin, and Chris Olah. Deconvolution and checkerboard artifacts. Distill, 2016.