Attention-Guided Masked Autoencoders For Learning Image Representations Supplementary Material

A. Implementation Details

A.1. Autoencoder Architecture

Our attention-guided masked autoencoder architecture is inspired by the vanilla MAE [8], whereby we use a vanilla ViT [6] for both the encoder and the decoder. Our models operate with a patch size of 16×16, and our encoder is a ViT-L with 16 heads and a depth of 24 blocks which produces $1024 - d$ embeddings. We keep our decoder depth constant at 8 blocks with a width of $512 - d$ which is narrower than our encoder.

For the DINO model, we choose the ViT-B with patch size 16×16, pre-trained fully unsupervised on ImageNet-1K [5]. This transformer is also used for generating the attention maps for TokenCut. To conduct our Grad-CAM ablation, we implement an ImageNet-1K supervised pre-trained ResNet-50.

A.2. Data Pre-Processing

All images used for pre-training are resized to 224×224 pixels. We only apply random horizontal flipping 50% of the time and normalization afterwards. Whenever our input image is flipped, we flip our attention map as well. To generate our attention map, we pass each image through the different object discovery networks and their respective post-processing pipeline. To simplify repeating our conducted experiments, we serialize all attention maps. All attention maps are output with size 14×14 given patches of dimension 16×16.

A.3. Training Details

To enable a leveled evaluation, both the vanilla MAE and our model are trained with the same basic settings, following He *et al*. [8]. All models are trained on a single A100 80GB with a batch size of $B = 512$. We keep the AdamW optimizer [12] and adopt a half-cycle cosine learning rate schedule after 40 warmup epochs [11]. Our learning rate μ is calculated using a linear scaling rule with a base learning rate of 0.00015 [7]: $\mu = 0.00015 \cdot B/256$. The learning rate is decayed after the warmup phase in the described fashion.

A.4. Pre-training Protocol

To pre-train our models, both the vanilla MAE and ours, we keep the training configuration, shown in Table 1, the same to ensure a fair comparison of the techniques. All configurations again follow He *et al*. [8].

Component	Value
Data Augmentation	RandomResizedCrop
Batch Size	512
Base Learning Rate	$1.5e^{-4}$
Learning Rate Schedule	Half-Cycle Cosine Decay [11]
Warmup Epochs [11]	40
Optimizer	AdamW [12]
Optimizer Momentum	0.9, 0.95

Table 1. **Pre-training protocol details.**

Component	Value
Data Augmentation	RandomResizedCrop
Batch Size	4096
Base Learning Rate	0.1
Learning Rate Schedule	Half-Cycle Cosine Decay [11]
Warmup Epochs [11]	10
Training Epochs	90
Optimizer	LARS [17]
Optimizer Momentum	0.9

Table 2. **Linear probing protocol details.**

A.5. Linear Probing Protocol

For our linear probing experiments, we freeze the entire backbone and finetune a single linear layer on the dataset. We evaluate the model on the validation split every epoch. We provide a detailed overview of our configuration in Table 2 below. All configured parameters are set equally to the implementation from He *et al*. [8].

A.6. Low-Shot Finetuning

For MAE and our *AttG* ViT-L/16 models, we adapt the low-shot finetuning protocol from Assran *et al*. [1]. For

the 1% evaluation (roughly 13 images per class), we finetune the models for 50 epochs with a peak learning rate of $1e^{-3}$ and use only crop and flip as augmentations. For the 10% finetuning, we add the extensive augmentations from the original MAE finetuning protocol introduced by He *et al*. [8]. For the SemMAE ViT-B [9], we find it improves results to reduce the learning rate to $5e^{-4}$, increase the epochs to 90 while keeping crop & flip for 1% and the extensive augmentations for 10%. For SimMIM [16] and BEiT [3], we found that increasing the learning rate in comparison to the MAE models significantly improved their results. For the BEiT ViT-L, we finetune for 50 epochs with a learning rate of $5e^{-3}$, with crop and flip for 1%, as well as the extensive MAE augmentation scheme for 10%. For the SimMIM ViT-B, we also use a learning rate of $5e^{-3}$ and increase the warmup epochs to 20 instead of 5 for the other models. We finetune for 100 epochs with the same augmentations. ViT-L models use learning rate layer decay of 0.75, ViT-Bs use 0.65. We always set weight decay to 0.05.

A.7. k-NN Classification Protocol

We follow the k-NN classification protocol by Caron *et al*. [4]. We extract embeddings for all images in the training and validation split. We then perform a weighted k-NN classification for $k \in \{5, 10, 20, 100, 200\}$ with a temperature of 0.07. We calculate our k-NN results with the official script from the DINO code release [1] by Caron *et al*. [4] and append the model loading part with the Python functions for checkpoint loading from the official repositories of the models we report numbers for.

A.8. Few-Shot Protocol

For our few-shot classification experiments with linear probing, we mostly follow the linear probing protocol described in Section A.5. We observe that due to the small dataset size with reduced training samples, mostly fewer than 4096, our learning rate calculation seems to output a learning rate too small for effective experimentation. Therefore, we calculate the learning rate as if the batch size would be the same by accumulating the gradients from multiple batches to match the initial batch size.

A.9. Semantic Segmentation

We conduct semantic segmentation experiments on ADE20k [19] and NYUv2 [13] following the protocols from Bachmann *et al*. [2]. We train a ConvNeXt-like [10] segmentation head on top of our model for both datasets and report our detailed hyperparameters in Table 6.

[1] https://github.com/facebookresearch/dino/blob/main/eval_knn.py

A.10. Taskonomy

As detailed in the main text, we perform experiments on a variety of tasks from the Taskonomy [18] benchmark. Specifically, inspired by Bachmann *et al*. [2], we train a DPT-like [14] head on top of the model to predict Depth, Edges, 2D Keypoints, 3D Keypoints and Occlusion, then report the average L1 loss and rank on the test dataset. For this, we use the tiny version of the dataset and limit the training split to 800 samples and the validation split to 200 samples. We test on the full dataset and report our detailed result in Table 3. Our attention-guided MAE outperforms the vanilla MAE across 4 of 5 tasks of the benchmark. We detail our hyperparameters in Table 7.

A.11. Code And Dataset Release

We will release our code upon acceptance. Our repository is based on the original MAE implementation provided by the authors, therefore ensuring a leveled comparison of our reported results. We are also planning to release the guidance maps used to train our model, so that our results can be easily reproduced.

B. Comparison To Contrastive Methods

In Table 4, we provide a comparison to contrastive methods, and show that our method *AttG* is able to lower the gap for linear evaluations compared to the vanilla MAE.

C. Finetuning Results

We report our finetuning results on ImageNet in Table 5. Both models have been pre-trained for 800 epochs. Our training settings follow He *et al*. [8].

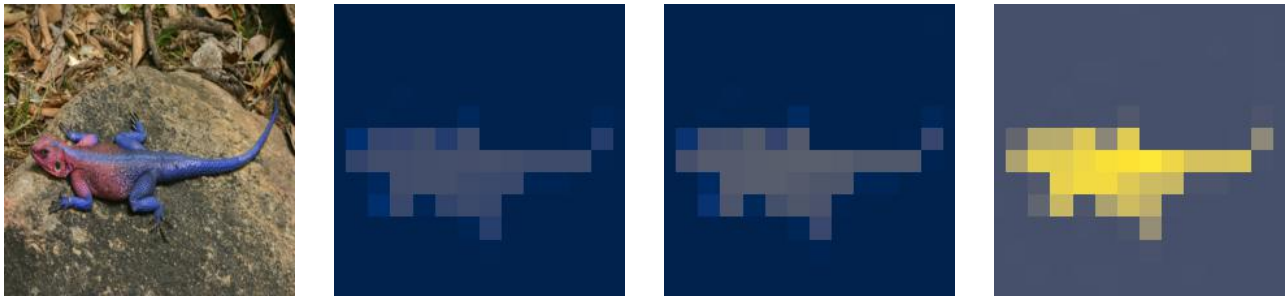
In our experiments, we observe both methods to have a comparable performance when finetuned on IN1K with a minor disadvantage for our method.

D. Visualization of Mask Scaling

We provide a visual comparison of different temperature parameters. Figure 1 shows the attention maps in the different stages of the scaling process, beginning with the normalized output of the object discovery network, followed by the temperature scaled version and the exponentially scaled map. By applying the temperature scaling before the exponential function, we avoid assigning additional weight in our guidance loss to background patches with zero values. After scaling with the exponential function, the weight of the background patches is changed to 1. This is also reflected in the visualization, comparing the background in Figure 1c to 1d. When applied to the reconstruction loss, this results in the loss of background patches being left unchanged, since they are simply multiplied by 1. If the order of scaling operations were to be reversed, the result of the

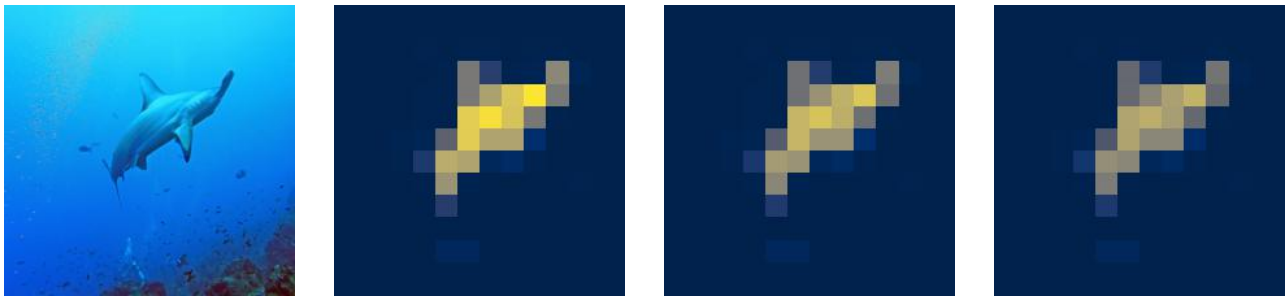
	Depth ($\cdot 10^{-2}$)	Edges ($\cdot 10^{-3}$)	Occlusion ($\cdot 10^{-4}$)	2D Keypoints ($\cdot 10^{-4}$)	3D Keypoints ($\cdot 10^{-2}$)	Average loss ($\cdot 10^{-2}$)	Average rank
MAE	3.810	6.927	5.803	2.745	**4.558**	1.829	1.8
AttG	**3.770**	**6.922**	**5.781**	**2.659**	4.595	**1.828**	**1.2**

Table 3. **Detailed Taskonomy Results.**



(a) Original (b) Normalized (c) Temperature-Scaled (d) Final output, exp. scaled

Figure 1. **Visualization of our scaling operations.** All attention maps are displayed with a fixed color scale from dark blue for 0 and bright yellow for 3.8, since the latter value presents the maximum of the guidance map when setting $\tau = 0.75$. At the normalization step, most background patches are valued at 0.



(a) Original (b) $\tau = 0.7$ (c) $\tau = 0.75$ (d) $\tau = 0.8$

Figure 2. **Visualization of guidance maps with different temperatures τ.** We fix the color scale from dark blue for 1.0 and bright yellow for 4.2, the minimum and maximum values of the guidance map when setting $\tau = 0.7$.

	Method	Size	Epochs	k-NN	Linear
CL	DINO	ViT-B	300	**76.1**	**78.2**
	iBOT	ViT-B	800	71.5	74.4
	AttnMask	ViT-B	100	72.8	76.1
MIM	SimMIM	ViT-B	800	9.4	56.7
	SemMAE	ViT-B	800	45.1	65.0
	BEiT	ViT-L	800	11.4	73.5
	MAE	ViT-L	800	52.8	73.5
	+ *AttG (Ours)*	ViT-L	800	**56.2**	**74.4**
	MAE	ViT-L	1600	50.9	75.1
	+ *AttG (Ours)*	ViT-L	1600	**59.0**	**75.9**

Table 4. ***k*-NN classification and linear probing on ImageNet for contrastive learning (CL) and masked image modeling (MIM) based pre-training techniques.**

Method	Epochs	Top-1 Accuracy
MAE	800	**85.9**
Ours	800	85.6

Table 5. **Finetuning results on ImageNet.**

exponential function would be divided by the temperature parameter, giving the background patches additional weight in the reconstruction loss, even though they are not part of the object.

E. Visualization of Different Temperatures

In Figure 3, we visually present the effects of scaling the attention map with different temperature parameters. By choosing a lower temperature, the additional emphasis on the object in the guidance loss is increased, and vice versa.

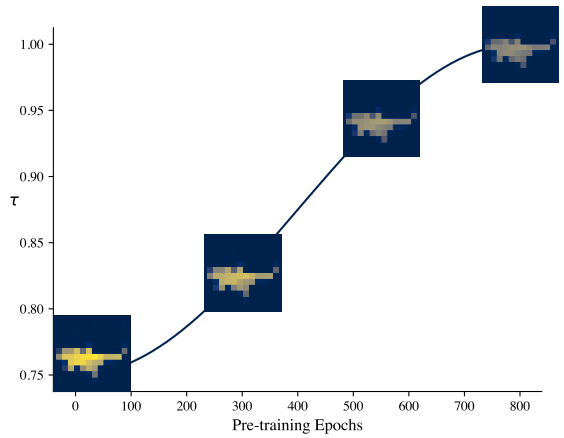


Figure 3. **Visualization of our guidance maps along the temperature schedule.**

This is reflected in the visualizations. The stronger the yellow of the patch is, the higher the values used for weighting the reconstruction loss at this position. Figure 3 visualizes the influence of τ on our attention maps.

Component	Value
Batch Size	16
Learning Rate	$3e^{-4}$
Warmup Learning Rate	$1e^{-6}$
Network Architecture	ConvNeXt [10]
Warmup Epochs [11]	3
Training Epochs	64
Optimizer	AdamW [12]

Table 6. **Semantic Segmentation protocol details.**

Component	Value
Batch Size	8
Learning Rate	$3e^{-4}$
Warmup Learning Rate	$1e^{-6}$
Network Architecture	DPT [14]
Warmup Epochs [11]	3
Training Epochs	64
Optimizer	AdamW [12]

Table 7. **Taskonomy protocol details.**

F. Random Guidance Map Ablations

We also evaluate using a random guidance map with the value range equal to that of our semantic guidance maps in order to investigate the effect of the semantic information in the guidance, but also to rule out a simple regularization by our method. Identical to our main approach, we apply the random guidance map to the reconstruction loss. Table 8 shows that our method goes beyond just simply regularizing the loss and points towards the benefits of our guidance through semantic information.

Guidance Map	k-NN	Linear
Random Map	48.0	70.7
TokenCut [15]	**57.0**	**77.1**

Table 8. **Ablation of different uses of the attention map.** Implementing a random guidance map yields wore results than using our semantic guidance map, therefore pointing towards the benefits of inducing the semantic information into the training process.

G. Input Masking Ratio

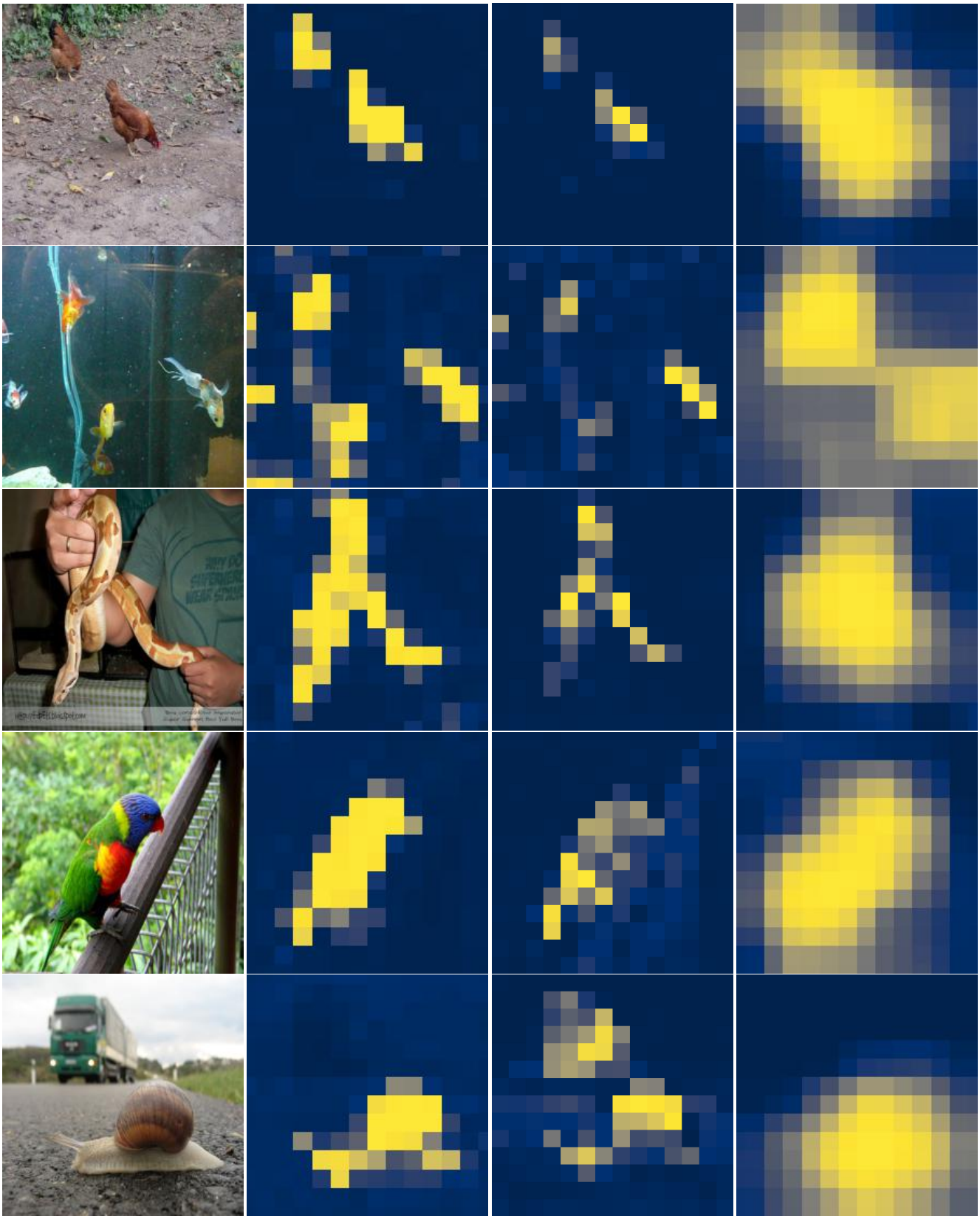
He *et al*. [8] find that for the vanilla MAE, randomly masking 75% of the input image is most effective for pre-training. As shown in Table 9, this is consistent with our findings. When increasing or decreasing the masking ratio by 5%, we observe that top-1 accuracy for linear probing drops. All models have been pre-trained for 400 epochs with $\tau = 0.75$ without scheduling and TokenCut attention maps for loss guidance.

Masking Ratio	Linear
0.7	76.6
0.75	**77.1**
0.8	76.5

Table 9. **Masking Ratios**

H. Additional Object Discovery Samples

Figure 4 illustrates additional examples of maps obtained from the different object discovery networks. The qualitative difference becomes visually apparent across all five examples.



(a) Original (b) TokenCut (c) DINO (d) Grad-CAM

Figure 4. **More attention map visualizations from our object discovery models.**

References

[1] Mahmoud Assran, Quentin Duval, Ishan Misra, Piotr Bojanowski, Pascal Vincent, Michael Rabbat, Yann LeCun, and Nicolas Ballas. Self-supervised learning from images with a joint-embedding predictive architecture, 2023. 1

[2] Roman Bachmann, David Mizrahi, Andrei Atanov, and Amir Zamir. Multimae: Multi-modal multi-task masked autoencoders. In *European Conference on Computer Vision*, pages 348–367. Springer, 2022. 2

[3] Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. Beit: Bert pre-training of image transformers. In *International Conference on Learning Representations*, 2021. 2

[4] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9650–9660, 2021. 2

[5] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 1

[6] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2020. 1

[7] Priya Goyal, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. Accurate, large minibatch sgd: Training imagenet in 1 hour. *arXiv preprint arXiv:1706.02677*, 2017. 1

[8] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16000–16009, 2022. 1, 2, 4

[9] Gang Li, Heliang Zheng, Daqing Liu, Chaoyue Wang, Bing Su, and Changwen Zheng. Semmae: Semantic-guided masking for learning masked autoencoders. *arXiv preprint arXiv:2206.10207*, 2022. 2

[10] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11976–11986, 2022. 2, 4

[11] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016. 1, 4

[12] Ilya Loshchilov and Frank Hutter. Fixing weight decay regularization in adam. *CoRR*, abs/1711.05101, 2017. 1, 4

[13] Pushmeet Kohli Nathan Silberman, Derek Hoiem and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *ECCV*, 2012. 2

[14] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 12179–12188, 2021. 2, 4

[15] Yangtao Wang, Xi Shen, Yuan Yuan, Yuming Du, Maomao Li, Shell Xu Hu, James L Crowley, and Dominique Vaufreydaz. Tokencut: Segmenting objects in images and videos with self-supervised transformer and normalized cut. *arXiv preprint arXiv:2209.00383*, 2022. 4

[16] Zhenda Xie, Zheng Zhang, Yue Cao, Yutong Lin, Jianmin Bao, Zhuliang Yao, Qi Dai, and Han Hu. Simmim: A simple framework for masked image modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9653–9663, 2022. 2

[17] Yang You, Igor Gitman, and Boris Ginsburg. Large batch training of convolutional networks. *arXiv preprint arXiv:1708.03888*, 2017. 1

[18] Amir R. Zamir, Alexander Sax, William Shen, Leonidas J. Guibas, Jitendra Malik, and Silvio Savarese. Taskonomy: Disentangling task transfer learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 2

[19] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 633–641, 2017. 2