An Image is Worth 16 $\times$ 16 Tokens: Visual Priors for Efficient Image Synthesis with Transformers

Robin Rombach* **Patrick Esser*** **Björn Ommer**
IWR, HCI, Heidelberg University

Abstract

Designed to learn long-range interactions on sequential data, transformers continue to show state-of-the-art results on a wide variety of tasks. In contrast to CNNs, they contain no inductive bias that prioritizes local interactions. This makes them expressive, but also computationally infeasible for long sequences, such as high-resolution images. We demonstrate the effectiveness of combining the inductive bias of CNNs with the expressivity of transformers to enable applications in high-resolution image synthesis tasks. This allows us to investigate effects of sequence permutations on the autoregressive prediction task. Finally, we present results on the potential of our approach to serve as a universal image synthesis model.

1 Introduction

Transformers are on the rise—they are now the de-facto standard architecture for language tasks [1, 2, 3, 4] and are increasingly adapted in other areas such as audio [5] and vision [6, 7]. In contrast to the predominant vision architecture, convolutional neural networks (CNNs), the transformer architecture contains no built-in inductive prior on the locality of interactions and is therefore free to learn arbitrary relationships among its inputs. However, this generality also implies that it *has to* learn all relationships, whereas CNNs have been designed to exploit the two-dimensional structure of images. Thus, the increased expressivity of transformers comes with large computational costs. Energy and time requirements of state-of-the-art transformer based models limit the ability to perform extensive experiments with these models. Observations that transformers tend to learn convolutional structures [7] thus beg the question: Do we have to re-learn everything we know about images from scratch each time we train a vision model, or can we efficiently encode prior knowledge while still retaining the flexibility of transformers? We hypothesize that low-level processing of images is well described by a locally connected structure, i.e. a convolutional architecture, whereas this structural assumption ceases to be effective at higher levels. This suggests an approach that fuses CNNs and transformers to combine the benefits of both: General purpose, efficient CNN feature extraction, combined with the ability to learn transformer models at greatly reduced costs with *the same* architecture across different tasks. Specifically, we employ a two-stage approach and use discrete representation learning to learn a CNN-based VQVAE [8] model which produces a compressed representation of its input. This compact discrete representation of the input is then processed by a transformer model via autoregressive next-token prediction.

2 Combining Vision Specific Architectures with Generic Transformers

Preliminaries The defining characteristic of the transformer architecture [1] is that it models interactions between its inputs solely through attention [9, 10, 11] which enables them to faithfully handle interactions between inputs regardless of their relative position to one another. Originally applied to language tasks, inputs (z_i) to the transformer were given by tokens, but arbitrary discrete signals, such as those obtained from audio or images, can be used. Each layer of the transformer then consists of an attention mechanism, which allows for interaction between inputs at different positions,

1st NeurIPS workshop on Interpretable Inductive Biases and Physically Structured Learning (2020), virtual.

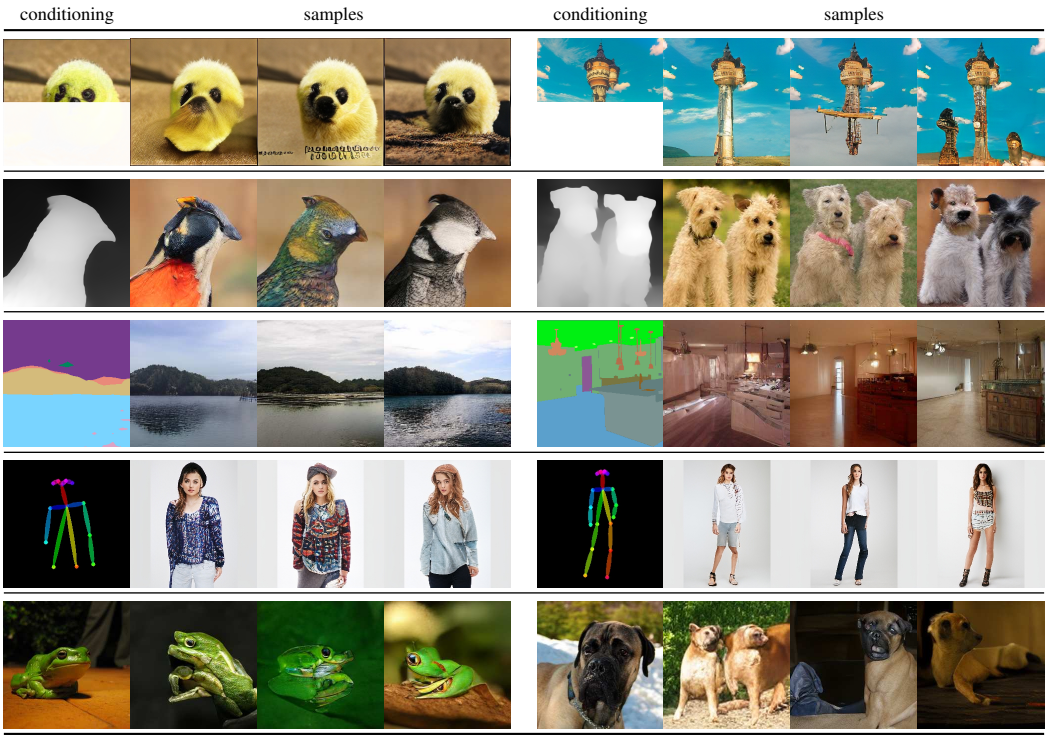

Figure 1: Transformers with efficient priors unify a wide range of image synthesis tasks. We show 256×256 synthesis results across different conditioning inputs and datasets, all obtained with the same approach to exploit inductive biases of CNN architectures in combination with the expressivity of transformer architectures.

followed by a position-wise fully connected network, which is applied to all positions independently. Since the attention mechanism relies on the computation of inner products between all pairs of elements in the sequence, its computational complexity increases quadratically with the sequence length. While the ability to consider interactions between *all* elements is the reason transformers efficiently learn long-range interactions, it is also the reason transformers quickly become infeasible, especially on images, where the sequence length itself scales quadratically with the resolution. However, the two-dimensional structure of images suggests that local interactions are particularly important. CNNs exploit this structure by restricting interactions between input variables to a local neighborhood defined by the kernel size of the convolutional kernel. Applying a kernel thus results in costs that scale linearly with the overall sequence length (the number of pixels in the case of images) and quadratically in the kernel size, which, in modern CNN architectures, is often fixed to a small constant such as 3×3. This inductive bias towards local interactions thus leads to efficient computations, but the wide range of specialized layers which are introduced into CNNs to handle different synthesis tasks [12, 13, 14, 15, 16] suggest that this bias is often too restrictive.

Approach To benefit from the efficiency of CNNs and the flexibility of transformers, we first learn a CNN-based autoencoder and use its latent code as inputs to a transformer. Through the size of the latent code we can control the trade-off between computational costs for training the transformer, and the quality of reconstructions which is an upper bound on synthesis performance. In particular, we make use of vector quantization [8] and train a VQVAE variant which results in discrete latent codes from which an input can be faithfully reconstructed. Our VQVAE is trained with a combination of perceptual and adversarial losses, which enables a large reduction in sequence length while retaining visually pleasing synthesis results. In contrast to previous works which applied pixel-based [17, 18] and transformer-based autoregressive models [6] on top of a VQVAE, we aim to push the limits of efficiency of transformer-based models by deliberately introducing a CNN-based VQVAE which benefits from suitable inductive biases for images, and mainly rely on the transformer architecture after the point where this inductive prior breaks down. As we show in the subsequent experiments, this approach yields an efficient and universal image synthesis model.

Efficiency Gains To assess the efficiency gains of our approach, we compare results between training a transformer directly on pixels, and training it on top of a VQVAE's latent code, given a *fixed* computational budget. We follow [6] and learn a dictionary of 512 RGB values on CIFAR10,

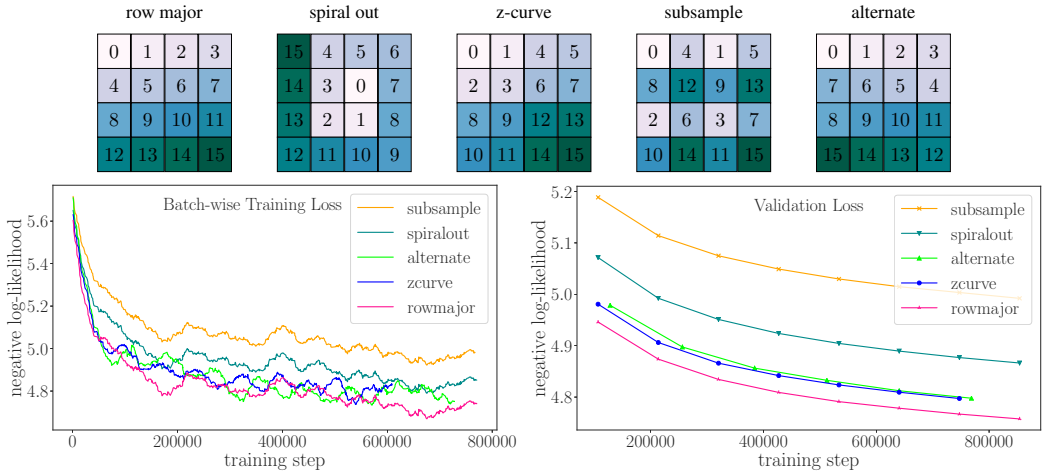

Figure 2: Top: All sequence permutations we investigate, illustrated on a 4×4 grid. Bottom: The transformer architecture is permutation invariant but next-token prediction is not: The average loss on the validation split of ImageNet, corresponding to the negative log-likelihood, differs significantly between different prediction orderings. Among our choices, the commonly used row-major order performs best.

such that each of its images is represented by a sequence of $32 \times 32 = 1024$ inputs, each taking integer values between 0 and 511. Additionally, we train the same transformer architecture on top of a VQVAE with a latent code of size $16 \times 16 = 256$ and a dictionary of 1024 values. We observe improvements of 18.63% for FID scores and 14.08 times faster sampling of images.

Quadratic complexity of transformers with respect to the sequence length means that efficiency gains become even more drastic at higher resolutions. Moreover, due to redundancy in pixels, we observe that we do not need to scale the discrete latent representation proportionally to the image resolution. In fact, 16×16 codes still produce perceptually good reconstruction for images of size 256×256, such that within this regime we have (up to the first stage training costs, which are one-time) constant computational costs. This enables the use of transformers for efficent high resolution image synthesis and allows us to perform the experiments of the following sections which would be otherwise infeasible. All subsequent experiments use a VQVAE with a latent code of size 16×16 and a dictionary of 1024 values to represent images of resolution 256×256, and a GPT2-medium architecture (307 M parameters) [3] is used for the transformer.

3 Introducing Vision Specific Biases within Generic Transformers

On the Inherent Ordering of Image Data For the "classical" domain of transformer models, natural language, the order of tokens is defined by the language at hand. For images and their discrete representations, in contrast, it is not clear which linear ordering to use. Intuitively, the difficulty to predict the next token depends on the available context for that prediction. To investigate our hypothesis that, for image data, this task is dependent on the choice of prediction ordering, we consider the following five different *permutations* of the input sequence of codebook indices: (i) **row major**, where the image representation is unrolled from top left to bottom right. (ii) **spiral out**, which incorporates the prior assumption that most images show a *centered* object. (iii) **z-curve**, also known as *z-order* or *morton curve*, which introduces the prior of *preserved locality* when mapping a 2D image representation onto a 1D sequence. (iv) **subsample**, where prefixes correspond to subsampled representations. (v) **alternate**, which is related to *row major*, but alternates the direction of unrolling every row. For a graphical visualization of these permutation variants, see Fig. 2.

To analyze the effect of each permutation, we first train a VQVAE in the settings of Sec. 2 on the ImageNet2012 dataset [19] on inputs of size 256×256. Then, given this VQVAE, we train a transformer (same settings as in Sec. 2) for each permutation variant in a controlled setting, i.e. we fix initialization, batch-size and computational budget for each variant. Fig. 2 shows the evolution of negative log-likelihood for each variant as a function of training iterations. Interestingly, *row major* performs best in terms of this metric, whereas the more hierarchical *subsample* prior does not induce any helpful bias for this task. Fig. 13 shows samples of each model variant. We observe that the two worst performing models in terms of negative log-likelihood (*subsample* and *spiral out*) tend to produce more textural samples, while the other variants synthesize samples with much more recognizable structures. Overall we can conclude that the autoregressive task is *not* permutation-invariant, but the commonly used *row major* ordering [17, 6] outperforms other orderings.

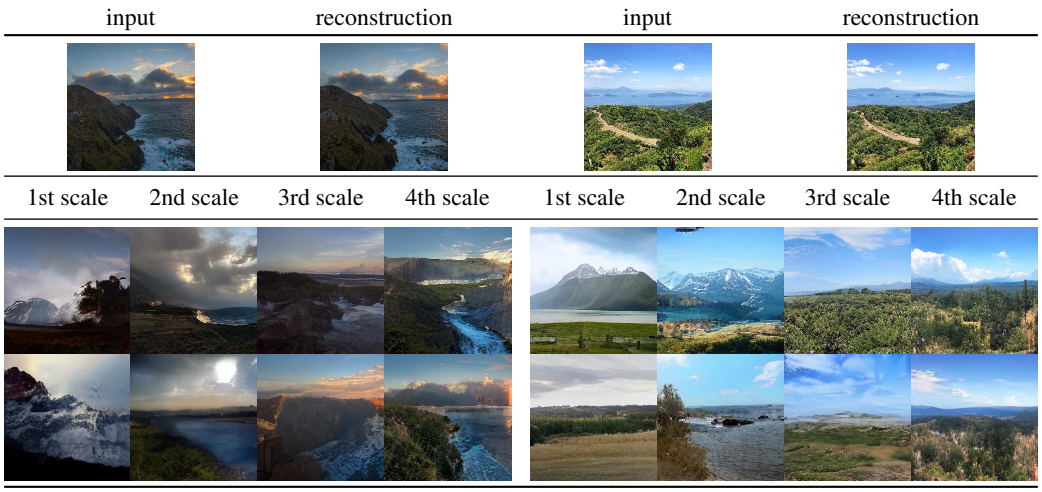

Figure 3: Scale-dependent samples for given inputs demonstrate control through multiscale-priors. See Sec. 3.

Integrating Multiscale Priors Besides investigating the effect of different prediction orderings on image generating transformers, we also include another popular inductive bias: Modeling an image through multiple different scales. In this setting, an image is iteratively generated in a *coarse-to-fine* manner [20]. Given a discrete 16×16 representation of an image x, we consider all subscales with spatial size equal to a power of 2. To compute these scales, we first average pool with a kernel size and stride of $16, 8, 4, 2$, respectively, and then re-quantize with the codebook. We then model the image in an autoregressive fashion over each scale, where each scale representation is unrolled seperately. Fig. 3& 7 show results of this approach on a dataset of landscape images. Generating an image autoregressively over the different scales now gives control on the output variance of the image. The very first 1×1 scale determines the overall appearance (e.g. color) of the image, whereas using increasingly more scales produces samples which are perceptually closer to the input image.

4 A Unified Model for Different Conditional Image Synthesis Tasks

The versatility and generality of the transformer architecture makes it a promising candidate for conditional image synthesis. Here, additional information c such as class labels or segmentation maps are introduced and our goal is to learn the distribution of possible outcomes. Thus, the model has to predict the probability of the next token given all previous tokens *and* the additional conditioning information c, such that the conditional likelihood reads $p(z|c) = \prod_i p(z_i|z_{<i}, c)$. We propose to model this task by simply prepending a discrete representation of the conditioning information of interest to the discrete sequence (z_i) which describes the image. We exploit the fact that c can itself be a sequence (c_i), and, for each conditioning, learn a seperate discrete representation with a VQVAE. Using the same settings as before (i.e. image size 256×256, latent size 16×16), we perform various conditional image synthesis experiments (for additional details see Sec. A.4):

(i): **Semantic image synthesis**, where we condition on unrolled discrete representations of semantic segmentation masks of the ADE20K [21] and a web-scraped landscapes dataset; see Fig. 8 and Fig. 9.

(ii): **Depth-to-image**, where we include a 3D dimensional prior (depth information) in the model. Again, we condition on a discrete representation of this additional information and obtain high-quality results for both the restricted ImageNet [22] and the ImageNet2012 dataset, see Fig. 1, 4 and Fig. 5.

(iii): **Pose-guided person synthesis:** Instead of using the semantically rich information of either segmentation or depth maps, Fig. 6 shows that the same approach as for the previous experiments can be used to build a shape-conditional generative model on the DeepFashion [23] dataset.

(iv): **Class-conditional image synthesis** Here, the conditioning information c is a single index describing the class label of interest. It can be directly concatenated with the corresponding sequence (z_i), increasing the total length of the sequence by one. Results on conditional sampling for both the RIN and IN datasets can are demonstrated in Fig. 10 and Fig. 11.

All of these examples make use of the same methodology. Instead of requiring task specific architectures or modules, the flexibility of the transformer allows us to learn appropriate interactions for each task, while the image-specific CNN architecture of the VQVAE—which can be *reused* across different tasks—leads to short sequence lengths. In combination, the presented approach can be understood as an efficient, general purpose mechanism for conditional image synthesis.

References

[1] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," 2017.

[2] A. Radford, "Improving language understanding by generative pre-training," 2018.

[3] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Language models are unsupervised multitask learners," 2019.

[4] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei, "Language models are few-shot learners," 2020.

[5] R. Child, S. Gray, A. Radford, and I. Sutskever, "Generating long sequences with sparse transformers," 2019.

[6] M. Chen, A. Radford, R. Child, J. Wu, H. Jun, P. Dhariwal, D. Luan, and I. Sutskever, "Generative pretraining from pixels," 2020.

[7] Anonymous, "An image is worth 16x16 words: Transformers for image recognition at scale," in *Submitted to International Conference on Learning Representations*, 2021, under review. [Online]. Available: https://openreview.net/forum?id=YicbFdNTTy

[8] A. van den Oord, O. Vinyals, and K. Kavukcuoglu, "Neural discrete representation learning," 2018.

[9] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," 2016.

[10] Y. Kim, C. Denton, L. Hoang, and A. M. Rush, "Structured attention networks," 2017.

[11] A. P. Parikh, O. Täckström, D. Das, and J. Uszkoreit, "A decomposable attention model for natural language inference," 2016.

[12] T. Park, M.-Y. Liu, T.-C. Wang, and J.-Y. Zhu, "Semantic image synthesis with spatially-adaptive normalization," 2019.

[13] P. Zhang, B. Zhang, D. Chen, L. Yuan, and F. Wen, "Cross-domain correspondence learning for exemplar-based image translation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 5143–5153.

[14] A. Siarohin, S. Lathuilière, S. Tulyakov, E. Ricci, and N. Sebe, "First order motion model for image animation," in *Conference on Neural Information Processing Systems (NeurIPS)*, December 2019.

[15] P. Zhu, R. Abdal, Y. Qin, and P. Wonka, "Sean: Image synthesis with semantic region-adaptive normalization," 2019.

[16] T. Zhou, S. Tulsiani, W. Sun, J. Malik, and A. A. Efros, "View synthesis by appearance flow," 2017.

[17] A. van den Oord, N. Kalchbrenner, O. Vinyals, L. Espeholt, A. Graves, and K. Kavukcuoglu, "Conditional image generation with pixelcnn decoders," 2016.

[18] A. Razavi, A. van den Oord, and O. Vinyals, "Generating diverse high-fidelity images with vq-vae-2," 2019.

[19] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 2009, pp. 248–255.

[20] S. Reed, A. van den Oord, N. Kalchbrenner, S. G. Colmenarejo, Z. Wang, D. Belov, and N. de Freitas, "Parallel multiscale autoregressive density estimation," 2017.

[21] B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso, and A. Torralba, "Semantic understanding of scenes through the ade20k dataset," *arXiv preprint arXiv:1608.05442*, 2016.

[22] S. Santurkar, D. Tsipras, B. Tran, A. Ilyas, L. Engstrom, and A. Madry, "Computer vision with a single (robust) classifier," in *ArXiv preprint arXiv:1906.09453*, 2019.

[23] Z. Liu, P. Luo, S. Qiu, X. Wang, and X. Tang, "Deepfashion: Powering robust clothes recognition and retrieval with rich annotations," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[24] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *CVPR*, 2018.

[25] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," *CVPR*, 2017.

[26] A. Krizhevsky, G. Hinton *et al.*, "Learning multiple layers of features from tiny images," 2009.

[27] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," 2017.

[28] R. Ranftl, K. Lasinger, D. Hafner, K. Schindler, and V. Koltun, "Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2020.

[29] Y. Liao, K. Schwarz, L. Mescheder, and A. Geiger, "Towards unsupervised learning of generative models for 3d controllable image synthesis," in *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[30] P. Esser, E. Sutter, and B. Ommer, "A variational u-net for conditional appearance and shape generation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8857–8866.

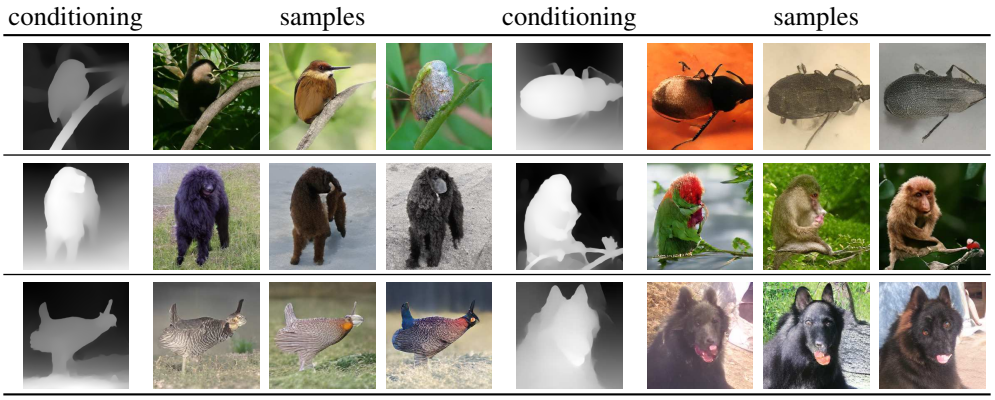

Figure 4: Conditional samples for the depth-to-image model on restricted ImageNet.

A Supplementary

A.1 Models

VQVAE The architecture of our VQVAE model is the same for all experiments with 256×256 images, which includes VQVAEs learned on semantic segmentation masks or depth maps as in Sec. 4. More specifically, we downsample from $256 \times 256 \times 3$ to a discrete representation of size $16 \times 16 \times 256$ using strided convolutions and residual blocks and quantize this representation with 1024 codes. For upscaling, the encoder architecture is reversed. We train the model with a combination of a variant of perceptual loss (LPIPS, [24]) and a patch discriminator as in [25]. For quantization, we employ a standard vector quantization loss as described in [8], where we use a "commitment" factor $\beta = 0.25$. For the experiments on CIFAR-10 (see Sec. 2) we only downsample once but otherwise adhere to this training protocol.

Transformer Our transformer model is identical to the GPT2-medium architecture [3] (307 M parameters), which we train with a batch size $b = 12$ and a learning rate of $b \cdot 4.5 \cdot 10^{-6}$ for all experiments on the ImageNet dataset and the multiscale prior in Sec. 3. The experiments on other visual priors (Sec. 4) use the same architecture in the GPT2-small setting. We generally produce samples with a temperature $t = 1.0$ and a top-k cutoff at $k = 100$.

A.2 Datasets

We use the following datasets for our experiments:

- CIFAR-10 [26], to compare the effectivness of our approach vs. a pure transformer in image space. See Sec. 2.
- DeepFashion [23], for pose-guided person and fashion synthesis. See Sec. 4 for results.
- ADE20K [21], for semantic image synthesis of indoor and outdoor scenes, see Sec. 4.
- ImageNet2012 [19] and Restricted ImageNet [22], for experiments on the effects of different inductive biases such as sequence ordering on image synthesis with transformers; furthermore class conditional, unconditional and depth-to-image synthesis; Sec. 3 and Sec. 4.
- A web-scraped dataset of landscapes images, for semantic image synthesis (where segmentation maps are obtained with the model by [27]) and incorporation of multiscale priors for scale-dependent image synthesis.

Furthermore, we utilize the pretrained model by [28] to extract depth maps from the ImageNet dataset.

A.3 Additional Synthesis Results

The following pages provide additional plots of synthesized images for depth-to-image (Fig. 4, 5), pose-guided (Fig. 6), multiscale (Fig. 7), semantic (Fig. 8, 9), class-conditional (Fig. 10, 11) and unconditional (Fig. 12, 13) image synthesis, respectively.

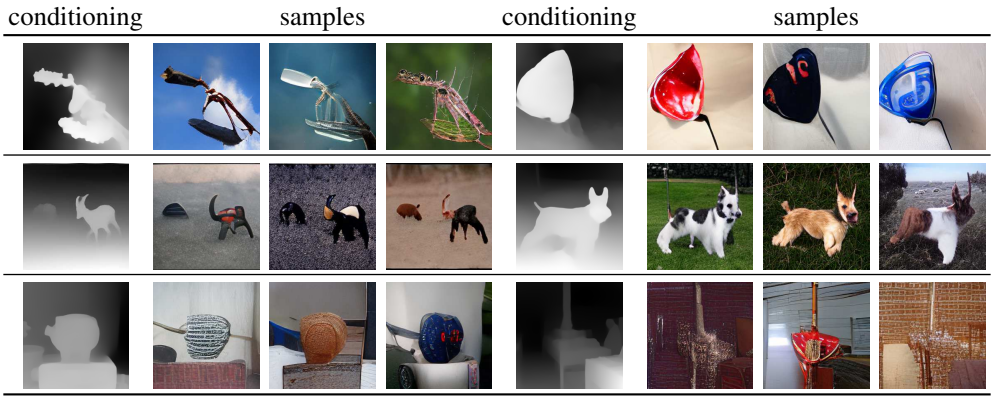

Figure 5: Conditional samples for the depth-to-image model on ImageNet.

A.4 Additional Details on Conditional Image Synthesis Tasks

The following provides an overview of our experiments with different conditionings (see Sec. 4), which were all executed with the same hyperparameters (in particular, an image size of 256×256 and alatent size of 16×16). Note that we use *the same* architecture for the VQVAE and transformer model (GPT2-medium) across *all* tasks.

Semantic Image Synthesis The task of semantically and spatially controllable image synthesis through segmentation masks received significant attention in the last few years [12, 13, 15], where approaches were mostly based on CNNs and directly modified inputs in image space. In contrast, we propose to use a transformer model, which, conditioned on a discrete representation $c = (c_i)$ of a segmentation map, generates a discrete representation (z_i) of the training image data. As described above, these discrete representations are obtained independently of the training of the transformer using discrete representation learning (VQVAE), which both for z and c produce representations of spatial size 16×16. Results for the ADE20K and a web-scraped landscapes dataset can be found in Fig. 8 and Fig. 9, indicating high-fidelity and variance in the synthesized outputs.

Depth-to-Image Recent work on generative image modeling [29] suggests that pure 2D models fail to produce coherent outputs since they lack an understanding of the actual 3D world. We address this problem by using *depth maps* to introduce 3D-knowledge into the model. We propose to include this powerful visual prior by first learning a discrete representation of depth maps, and, in analogy to the approch for segmentation maps, learn the conditional density $p(z|c)$ by concatenating this representation with the image representation (z_i). Results of 256×256 images are depicted in Fig. 1, 4 and Fig. 5, indicating high quality in synthesis both for the Restricted ImageNet and the more challenging ImageNet2012 dataset.

Pose-Guided Person Synthesis Instead of the semantically rich prior/information of either segmentation masks or depth maps, many computer vision problems involve sparse information. A particular application is conditional appearance and shape generation from keypoints [30]. Fig. 6 shows that the same approach as for the previous experiments can be used to build a shape-conditional generative model on the DeepFashion dataset. Again, discrete representation learning is used to produce discrete latent representations of both the RGB images and the shape images (keypoints) of the dataset, and the conditioning information is concatentated with the image sequence.

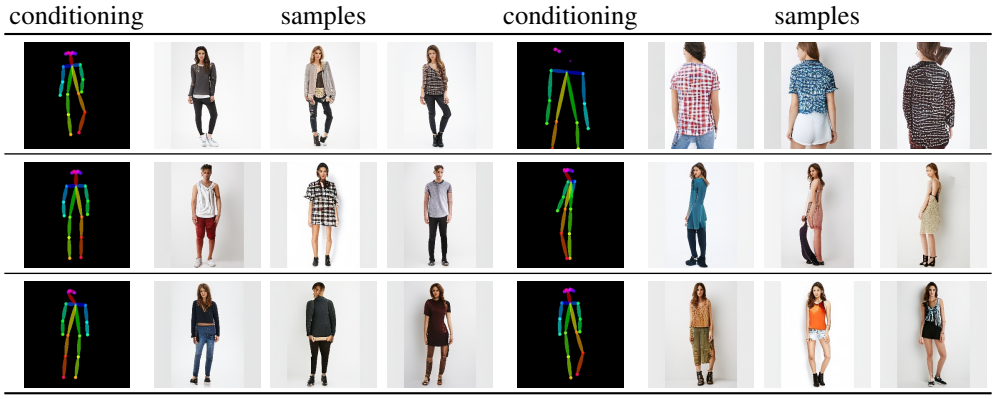

Figure 6: Conditional samples for the pose-guided synthesis model via keypoints.

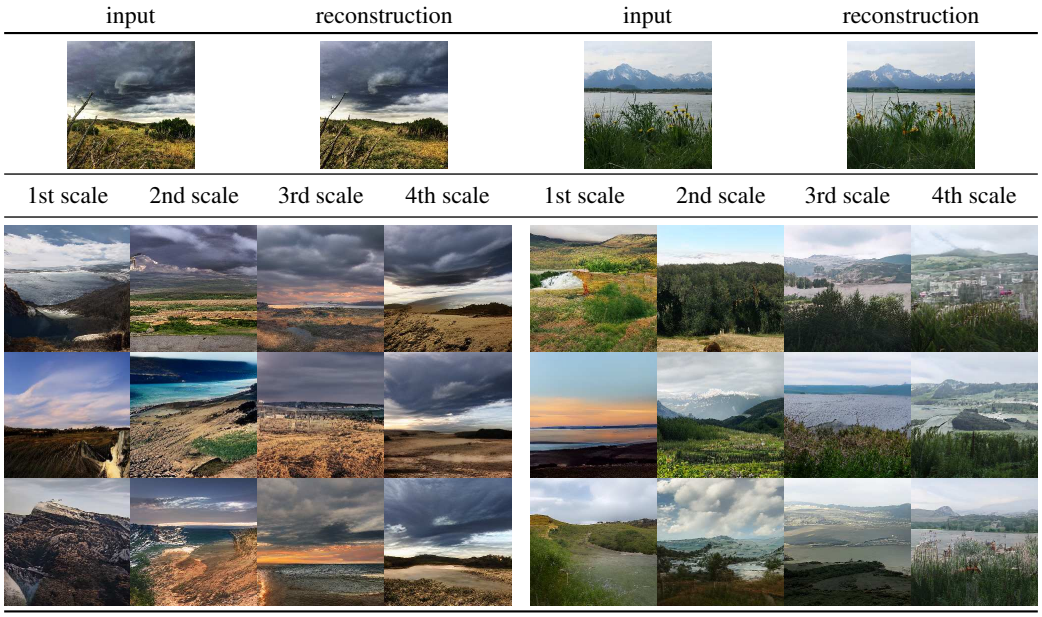

Figure 7: Scale-dependent samples for given inputs demonstrate control through multiscale-priors. See Sec. 3.

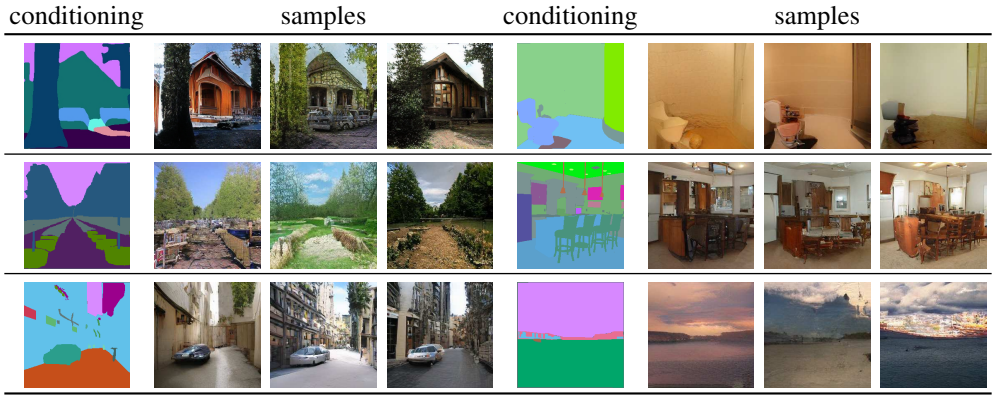

Figure 8: Samples from the semantically guided model trained on ADE20K.

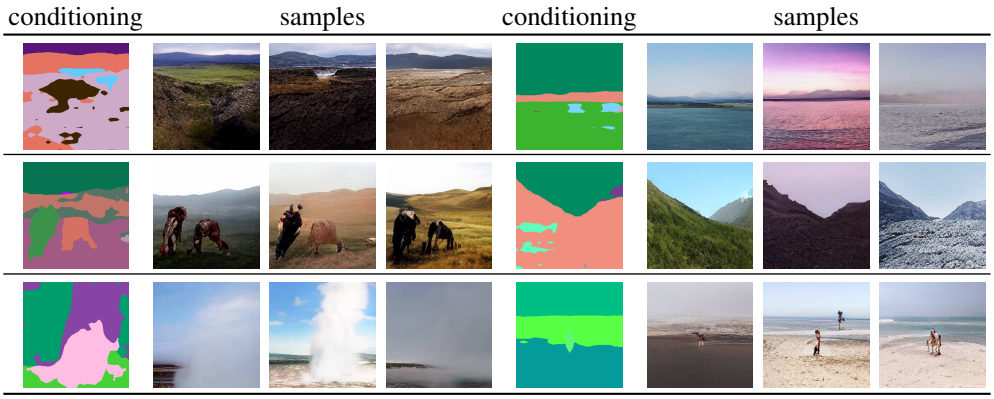

Figure 9: Samples from the semantically guided model trained on our landscapes dataset.

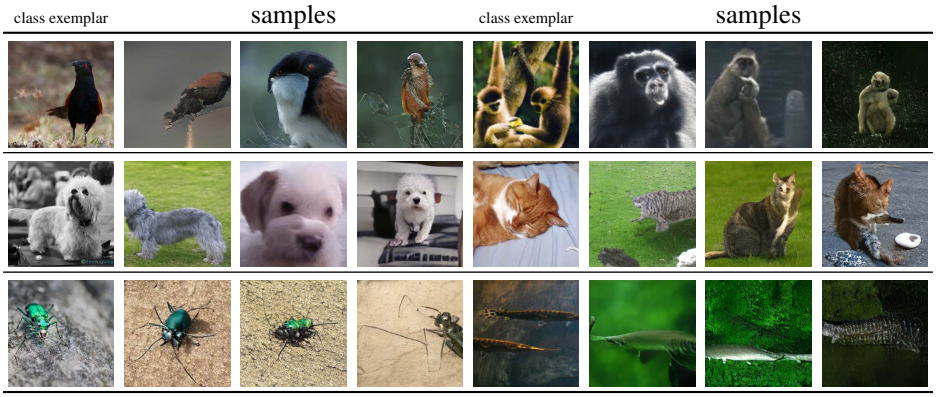

Figure 10: Samples produced by the class-conditional model trained on restricted ImageNet.

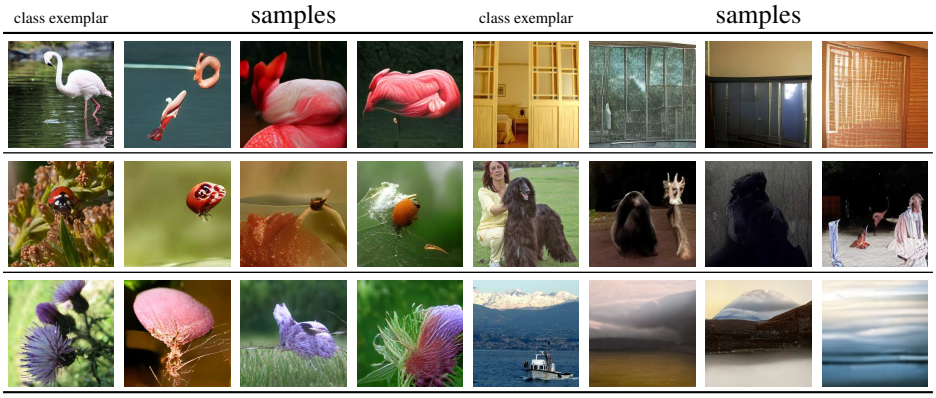

Figure 11: Samples synthesized by the class-conditional model trained on ImageNet.

Figure 12: Samples produced by an unconditional model trained on restricted ImageNet.

Figure 13: Random samples from transformer models trained with different inductive priors for next pixel prediction as described in Sec. 3.