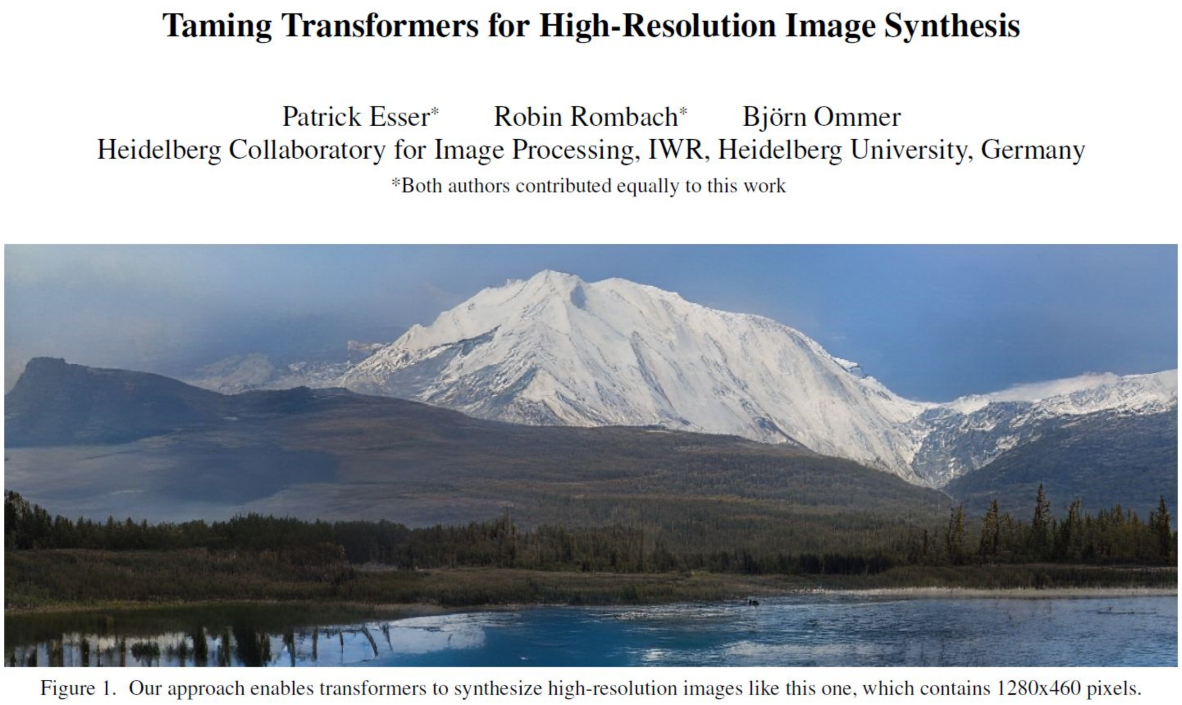


Taming Transformers for High-Resolution Image Synthesis

Patrick Esser* Robin Rombach* Björn Ommer
Heidelberg Collaboratory for Image Processing, IWR, Heidelberg University, Germany
*Both authors contributed equally to this work

Figure 1. Our approach enables transformers to synthesize high-resolution images like this one, which contains 1280x460 pixels.

CVPR’21 Oral

DAVIAN Vision Study
2021.06.07

양소영

Reference

https://arxiv.org/abs/2012.09841

https://github.com/CompVis/taming-transformers#more-resources

Table

1. VQVAE
2. Problem setting
3. Method: VQGAN
4. Experiments

1. VQVAE, NeurIPS 2017

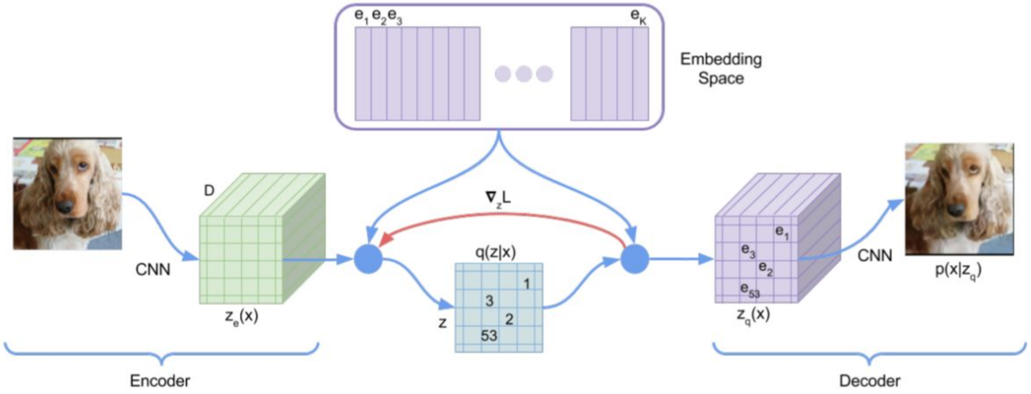
Neural Discrete Representation Learning

Aaron van den Oord
DeepMind
avdnoord@google.com

Oriol Vinyals
DeepMind
vinyals@google.com

Koray Kavukcuoglu
DeepMind
korayk@google.com

$$L = \log p(x|z_q(x)) + \|\text{sg}[z_e(x)] - e\|_2^2 + \beta\|z_e(x) - \text{sg}[e]\|_2^2,$$



https://arxiv.org/abs/1711.00937

2. Problem setting

In contrast to CNNs, **Transformers** contain **no inductive bias that prioritizes local interactions.**

- Transformer architecture contains no built-in inductive prior on the locality of interactions and is therefore free to learn complex relationships among its inputs.
- ***However***, this generality also implies that it has to learn all relationships, whereas CNNs have been designed to exploit prior knowledge about strong local correlations within images.
- **Thus**, **the increased expressivity of transformers comes with quadratically increasing computational costs,** because all pairwise interactions are taken into account.

3. Method - hypothesis

Hypothesis : that low-level image structure is well described by a local connectivity, i.e. a convolutional architecture.

Novelty :

- use a convolutional approach to efficiently learn a codebook of context-rich visual parts and, subsequently, learn a model of their global compositions.
- utilize an adversarial approach to ensure that the dictionary of local parts captures perceptually important local structure to alleviate the need for modeling low-level statistics with the transformer architecture.

3. Model architecture

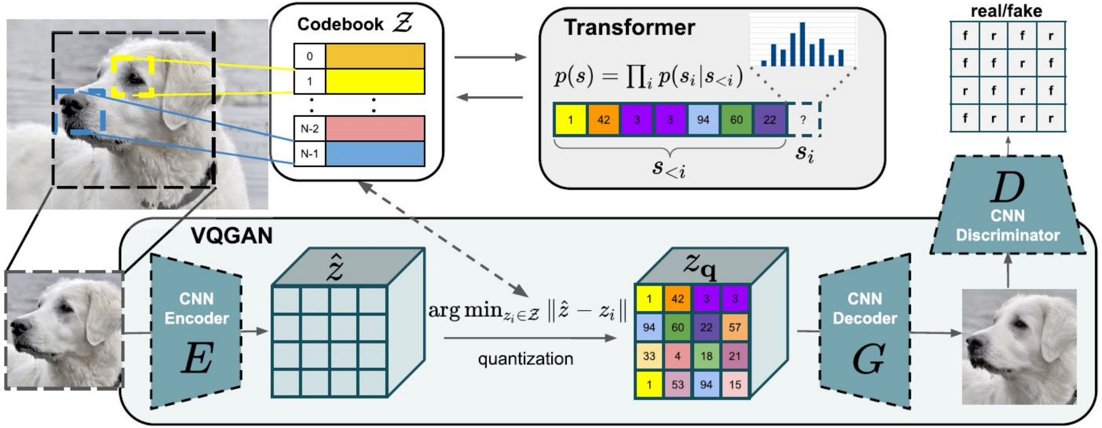


Figure 2. Our approach uses a convolutional *VQGAN* to learn a codebook of context-rich visual parts, whose composition is subsequently modeled with an autoregressive transformer architecture. A discrete codebook provides the interface between these architectures and a patch-based discriminator enables strong compression while retaining high perceptual quality. This method introduces the efficiency of convolutional approaches to transformer based high resolution image synthesis.

3.1. Learning effective codebook

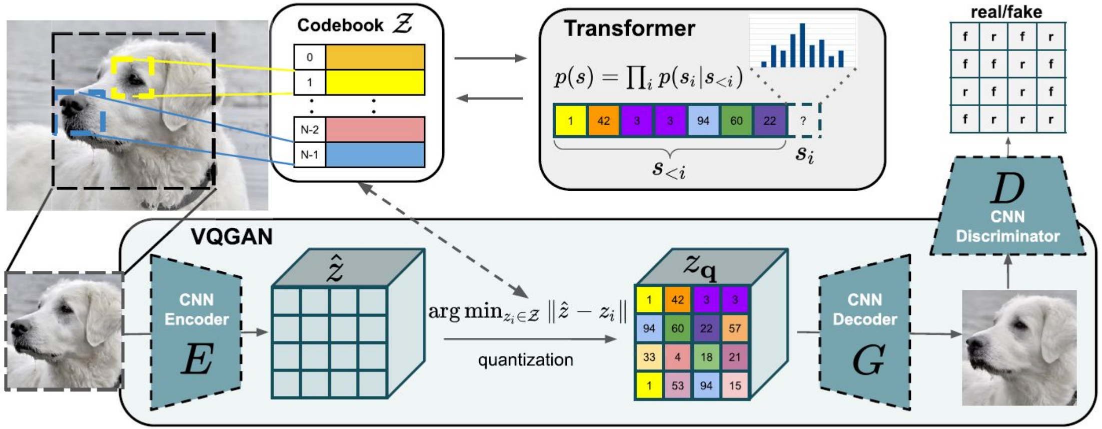


Figure 2. Our approach uses a convolutional *VQGAN* to learn a codebook of context-rich visual parts, whose composition is subsequently modeled with an autoregressive transformer architecture. A discrete codebook provides the interface between these architectures and a patch-based discriminator enables strong compression while retaining high perceptual quality. This method introduces the efficiency of convolutional approaches to transformer based high resolution image synthesis.

$$\mathcal{L}_{\mathrm{GAN}}(\{E, G, \mathcal{Z}\}, D) = [\log D(x) + \log(1 - D(\hat{x}))] \quad (5)$$

$$\hat{x} = G(z_{\mathbf{q}}) = G\left(\mathbf{q}(E(x))\right). \quad (3)$$

$$\mathcal{L}_{\mathrm{VQ}}(E, G, \mathcal{Z}) = \|x - \hat{x}\|^2 + \|\mathrm{sg}[E(x)] - z_{\mathbf{q}}\|_2^2 + \beta\|\mathrm{sg}[z_{\mathbf{q}}] - E(x)\|_2^2. \quad (4)$$

3.1. Learning effective codebook

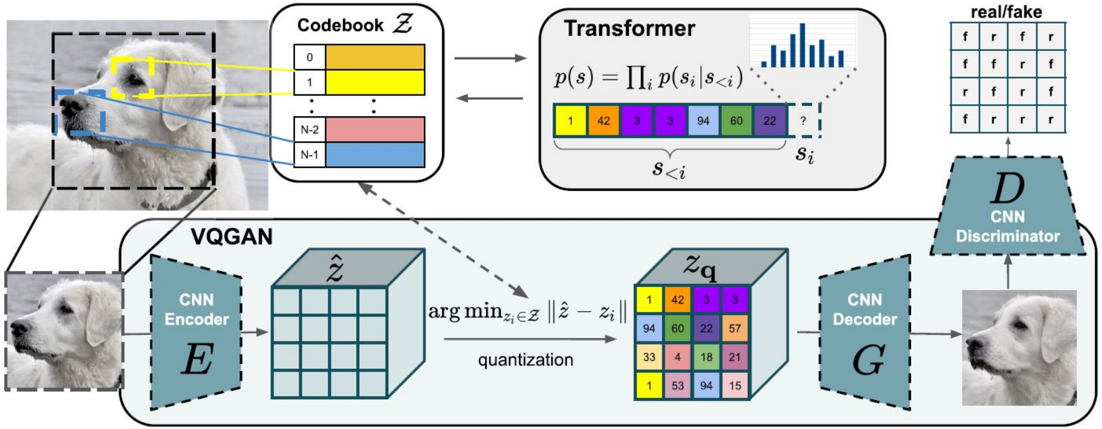


Figure 2. Our approach uses a convolutional *VQGAN* to learn a codebook of context-rich visual parts, whose composition is subsequently modeled with an autoregressive transformer architecture. A discrete codebook provides the interface between these architectures and a patch-based discriminator enables strong compression while retaining high perceptual quality. This method introduces the efficiency of convolutional approaches to transformer based high resolution image synthesis.

The complete objective for finding the optimal compression model $\mathcal{Q}^* = \{E^*, G^*, \mathcal{Z}^*\}$ then reads

$$\mathcal{Q}^* = \underset{E,G,\mathcal{Z}}{\arg\min}\, \max_{D} \mathbb{E}_{x\sim p(x)} \Big[\mathcal{L}_{\text{VQ}}(E, G, \mathcal{Z}) + \lambda \mathcal{L}_{\text{GAN}}(\{E, G, \mathcal{Z}\}, D) \Big], \quad (6)$$

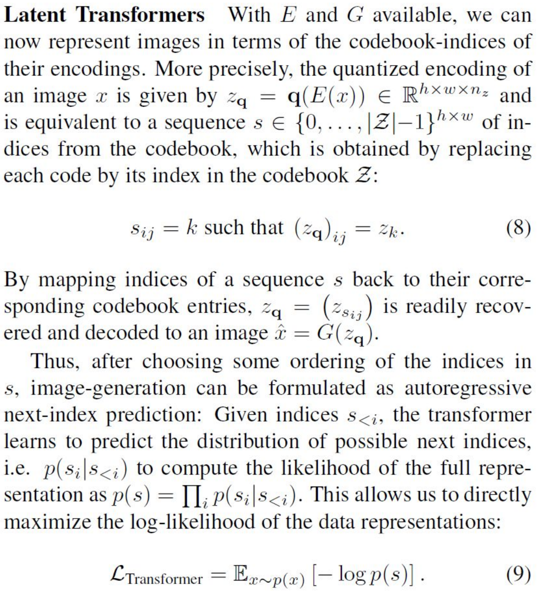
where we compute the adaptive weight λ according to

$$\lambda = \frac{\nabla_{G_L}[\mathcal{L}_{\text{rec}}]}{\nabla_{G_L}[\mathcal{L}_{\text{GAN}}] + \delta} \quad (7)$$

$$\mathcal{L}_{\text{GAN}}(\{E, G, \mathcal{Z}\}, D) = [\log D(x) + \log(1 - D(\hat{x}))] \quad (5)$$

$$\mathcal{L}_{\text{VQ}}(E, G, \mathcal{Z}) = \|x - \hat{x}\|^2 + \quad \|\text{sg}[E(x)] - z_{\mathbf{q}}\|_2^2 + \beta \|\text{sg}[z_{\mathbf{q}}] - E(x)\|_2^2. \quad (4)$$

3.2. Learning composition of images with transformers



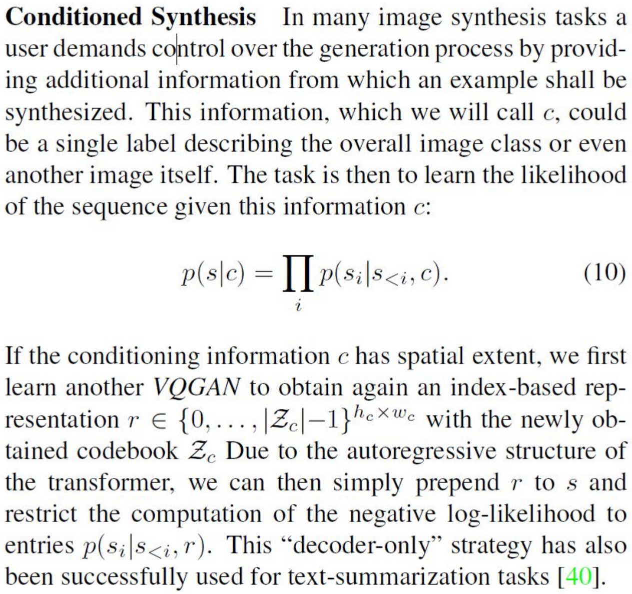
Latent Transformers With E and G available, we can now represent images in terms of the codebook-indices of their encodings. More precisely, the quantized encoding of an image x is given by $z_{\mathbf{q}} = \mathbf{q}(E(x)) \in \mathbb{R}^{h \times w \times n_z}$ and is equivalent to a sequence $s \in \{0, \dots, |\mathcal{Z}|-1\}^{h \times w}$ of indices from the codebook, which is obtained by replacing each code by its index in the codebook $\mathcal{Z}$:

$$s_{ij} = k \text{ such that } (z_{\mathbf{q}})_{ij} = z_k. \tag{8}$$

By mapping indices of a sequence s back to their corresponding codebook entries, $z_{\mathbf{q}} = (z_{s_{ij}})$ is readily recovered and decoded to an image $\hat{x} = G(z_{\mathbf{q}})$.

Thus, after choosing some ordering of the indices in s, image-generation can be formulated as autoregressive next-index prediction: Given indices $s_{<i}$, the transformer learns to predict the distribution of possible next indices, i.e. $p(s_i|s_{<i})$ to compute the likelihood of the full representation as $p(s) = \prod_i p(s_i|s_{<i})$. This allows us to directly maximize the log-likelihood of the data representations:

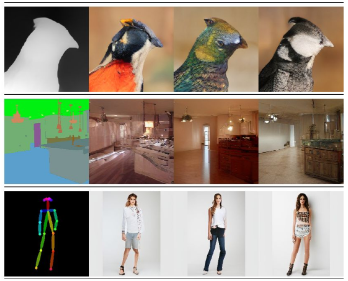
$$\mathcal{L}_{\text{Transformer}} = \mathbb{E}_{x \sim p(x)} \left[-\log p(s) \right]. \tag{9}$$



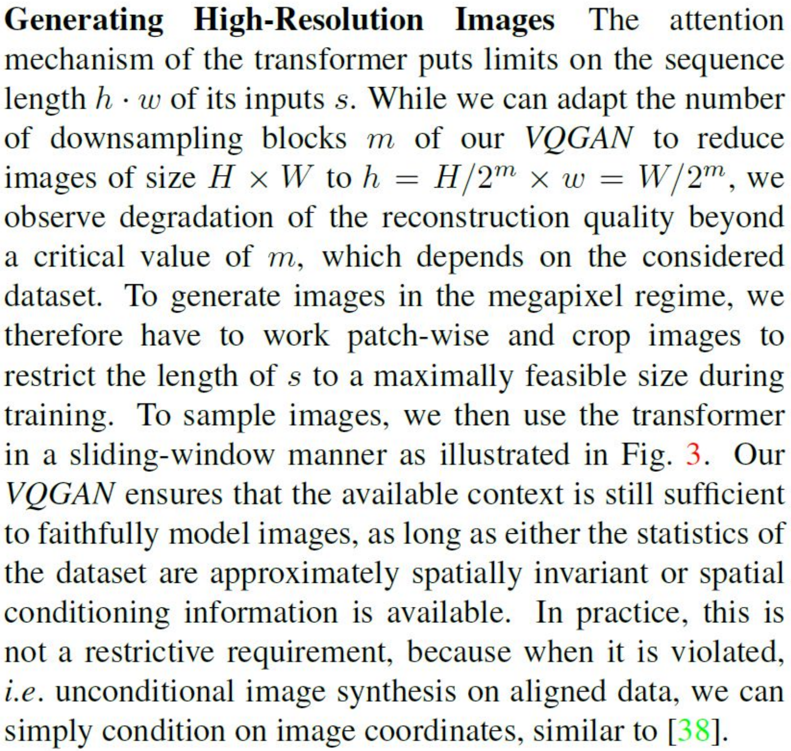
Conditioned Synthesis In many image synthesis tasks a user demands control over the generation process by providing additional information from which an example shall be synthesized. This information, which we will call c, could be a single label describing the overall image class or even another image itself. The task is then to learn the likelihood of the sequence given this information c:

$$p(s|c) = \prod_i p(s_i|s_{<i}, c). \tag{10}$$

If the conditioning information c has spatial extent, we first learn another *VQGAN* to obtain again an index-based representation $r \in \{0, \dots, |\mathcal{Z}_c|-1\}^{h_c \times w_c}$ with the newly obtained codebook $\mathcal{Z}_c$ Due to the autoregressive structure of the transformer, we can then simply prepend r to s and restrict the computation of the negative log-likelihood to entries $p(s_i|s_{<i}, r)$. This "decoder-only" strategy has also been successfully used for text-summarization tasks [40].



3.2. Learning composition of images with transformers



Generating High-Resolution Images The attention mechanism of the transformer puts limits on the sequence length $h \cdot w$ of its inputs s. While we can adapt the number of downsampling blocks m of our *VQGAN* to reduce images of size $H \times W$ to $h = H/2^m \times w = W/2^m$, we observe degradation of the reconstruction quality beyond a critical value of m, which depends on the considered dataset. To generate images in the megapixel regime, we therefore have to work patch-wise and crop images to restrict the length of s to a maximally feasible size during training. To sample images, we then use the transformer in a sliding-window manner as illustrated in Fig. 3. Our *VQGAN* ensures that the available context is still sufficient to faithfully model images, as long as either the statistics of the dataset are approximately spatially invariant or spatial conditioning information is available. In practice, this is not a restrictive requirement, because when it is violated, *i.e*. unconditional image synthesis on aligned data, we can simply condition on image coordinates, similar to [38].

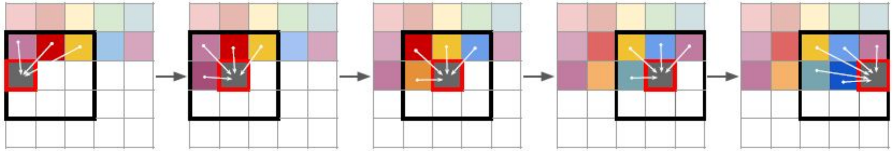


Figure 3. Sliding attention window.

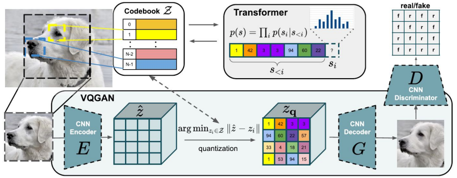
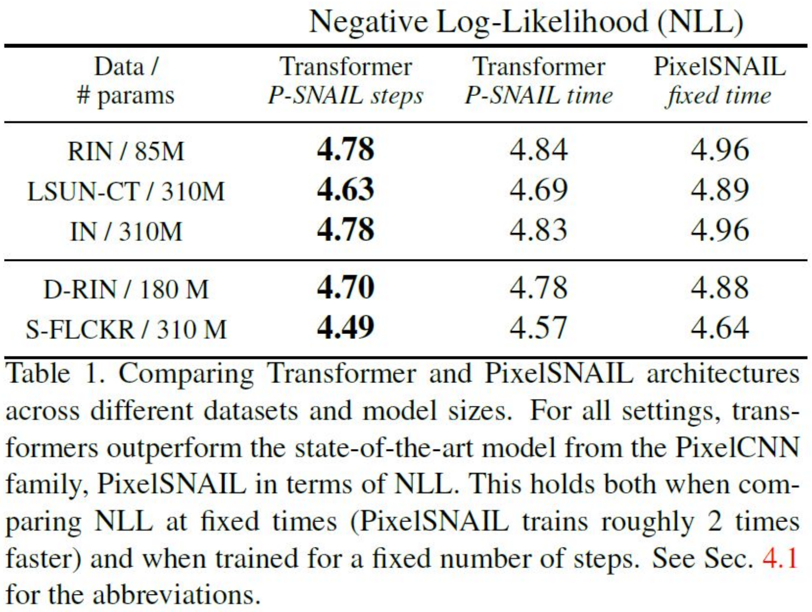


Figure 2. Our approach uses a convolutional *VQGAN* to learn a codebook of context-rich visual parts, whose composition is subsequently modeled with an autoregressive transformer architecture. A discrete codebook provides the interface between these architectures and a patch-based discriminator enables strong compression while retaining high perceptual quality. This method introduces the efficiency of convolutional approaches to transformer based high resolution image synthesis.

4. Experiments: (1) transformer latent space

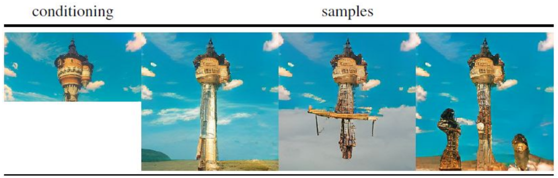


	Negative Log-Likelihood (NLL)		
Data / # params	Transformer *P-SNAIL steps*	Transformer *P-SNAIL time*	PixelSNAIL *fixed time*
RIN / 85M	**4.78**	4.84	4.96
LSUN-CT / 310M	**4.63**	4.69	4.89
IN / 310M	**4.78**	4.83	4.96
D-RIN / 180 M	**4.70**	4.78	4.88
S-FLCKR / 310 M	**4.49**	4.57	4.64

Table 1. Comparing Transformer and PixelSNAIL architectures across different datasets and model sizes. For all settings, transformers outperform the state-of-the-art model from the PixelCNN family, PixelSNAIL in terms of NLL. This holds both when comparing NLL at fixed times (PixelSNAIL trains roughly 2 times faster) and when trained for a fixed number of steps. See Sec. 4.1 for the abbreviations.

- VQGAN with m = 4
- img size = 256 x 256, latent size = 16 x 16
- The results shows that the transformer consistently outperforms PixelSNAIL across all tasks when trained for the same amount of time and the gap increases even further when trained for the same number of steps.
- These results demonstrate that gains of transformers carry over to our proposed two-stage setting.

4. Experiments: (2) Image synthesis



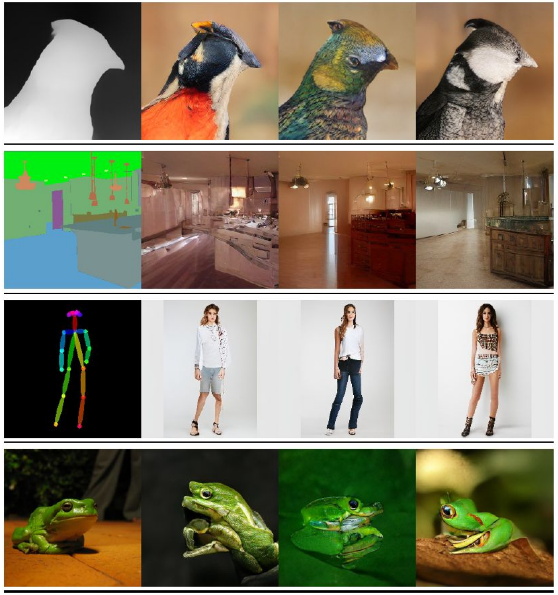


Figure 4. Transformers within our setting unify a wide range of image synthesis tasks. We show 256 × 256 synthesis results across different conditioning inputs and datasets, all obtained with the same approach to exploit inductive biases of effective CNN based *VQGAN* architectures in combination with the expressivity of transformer architectures. Top row: Completions from unconditional training on ImageNet. 2nd row: Depth-to-Image on RIN. 3rd row: Semantically guided synthesis on ADE20K. 4th row: Pose-guided person generation on DeepFashion. Bottom row: Class-conditional samples on RIN.

(i): **Semantic image synthesis**, where we condition on semantic segmentation masks of ADE20K [71], a web-scraped landscapes dataset (S-FLCKR) and COCO-Stuff [6]. Results are depicted in Figure 4, 5 and Fig. 6.

(ii): **Structure-to-image**, where we use either depth or edge information to synthesize images from both RIN and IN (see Sec. 4.1). The resulting depth-to-image and edge-to-image translations are visualized in Fig. 4 and Fig. 6.

(iii): **Pose-guided synthesis:** Instead of using the semantically rich information of either segmentation or depth maps, Fig. 4 shows that the same approach as for the previous experiments can be used to build a shape-conditional generative model on the DeepFashion [41] dataset.

(iv): **Stochastic superresolution**, where low-resolution images serve as the conditioning information and are thereby upsampled. We train our model for an upsampling factor of 8 on ImageNet and show results in Fig. 6.

(v): **Class-conditional image synthesis:** Here, the conditioning information c is a single index describing the class label of interest. Results on conditional sampling for the RIN dataset are demonstrated in Fig. 4.

4. Experiments: (2) Image synthesis

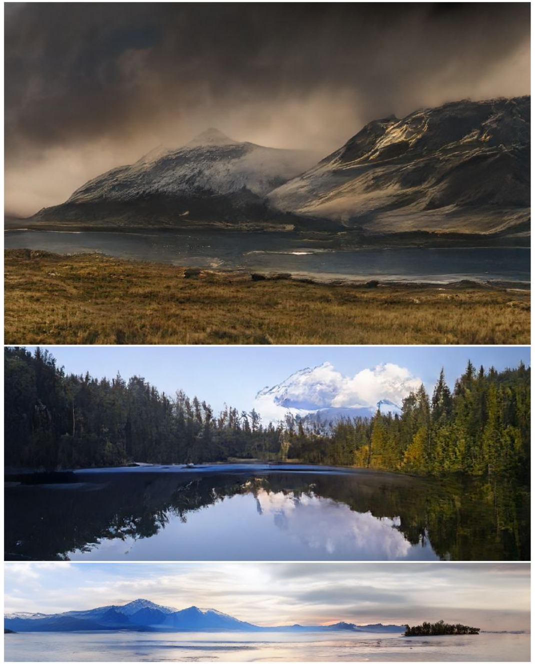


Figure 5. Samples generated from semantic layouts on S-FLCKR. Sizes from top-to-bottom: 1280 × 832, 1024 × 416 and 1280 × 240 pixels. Best viewed zoomed in. A larger visualization can be found in the appendix, see Fig 17.

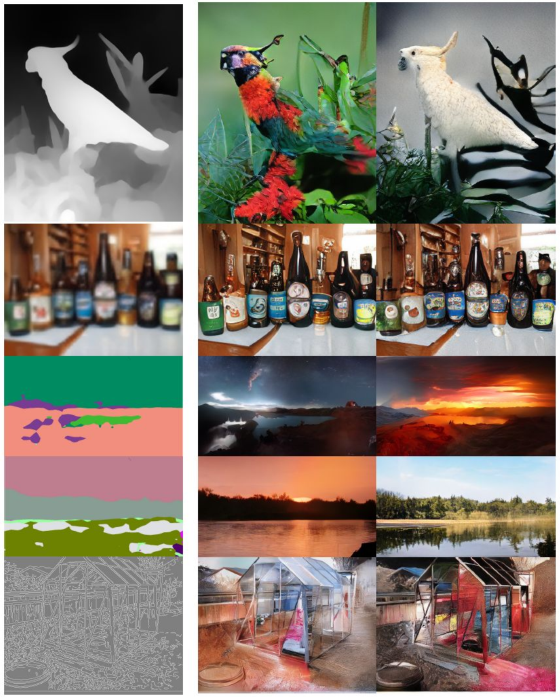


Figure 6. Applying the sliding attention window approach (Fig. 3) to various conditional image synthesis tasks. Top: Depth-to-image on RIN, 2nd row: Stochastic superresolution on IN, 3rd and 4th row: Semantic synthesis on S-FLCKR, bottom: Edge-guided synthesis on IN. The resulting images vary between 368 × 496 and 1024 × 576, hence they are best viewed zoomed in.

(i): **Semantic image synthesis**, where we condition on semantic segmentation masks of ADE20K [71], a web-scraped landscapes dataset (S-FLCKR) and COCO-Stuff [6]. Results are depicted in Figure 4, 5 and Fig. 6.

(ii): **Structure-to-image**, where we use either depth or edge information to synthesize images from both RIN and IN (see Sec. 4.1). The resulting depth-to-image and edge-to-image translations are visualized in Fig. 4 and Fig. 6.

(iii): **Pose-guided synthesis:** Instead of using the semantically rich information of either segmentation or depth maps, Fig. 4 shows that the same approach as for the previous experiments can be used to build a shape-conditional generative model on the DeepFashion [41] dataset.

(iv): **Stochastic superresolution**, where low-resolution images serve as the conditioning information and are thereby upsampled. We train our model for an upsampling factor of 8 on ImageNet and show results in Fig. 6.

(v): **Class-conditional image synthesis:** Here, the conditioning information c is a single index describing the class label of interest. Results on conditional sampling for the RIN dataset are demonstrated in Fig. 4.

4. Experiments: (3) Context-rich vocabularies

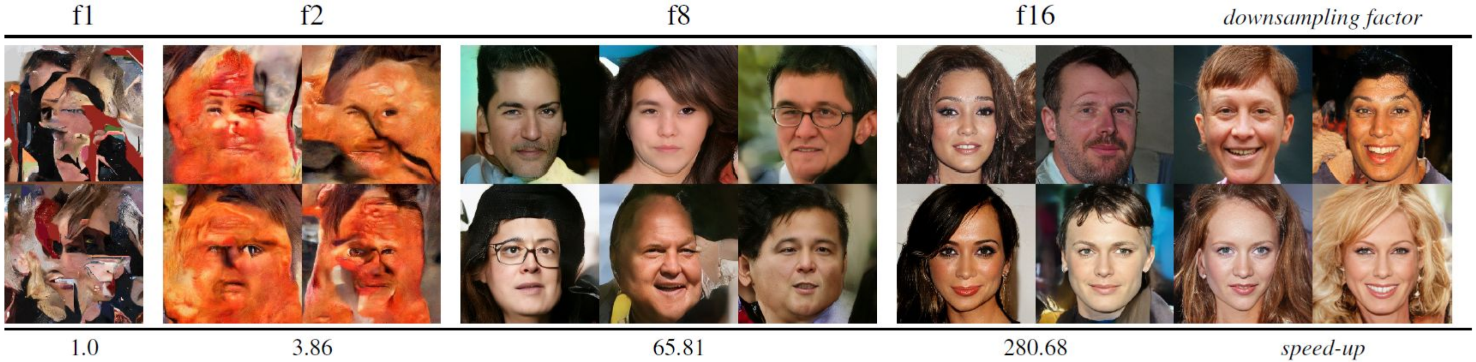
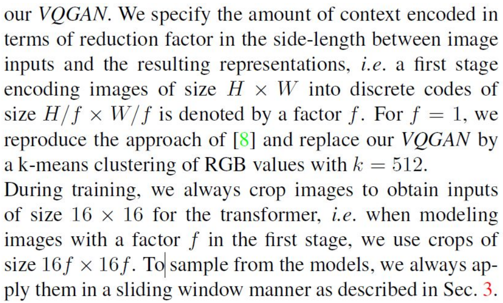
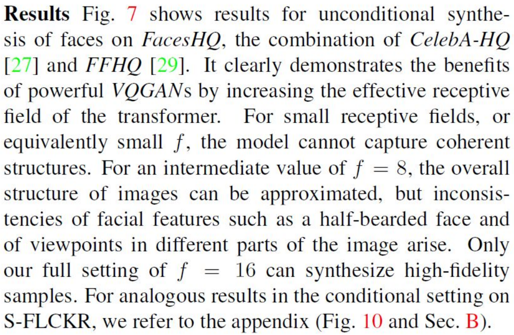


Figure 7. Evaluating the importance of effective codebook for HQ-Faces (CelebA-HQ and FFHQ) for a fixed sequence length $|s|= 16{\cdot}16 = 256$. Globally consistent structures can only be modeled with a context-rich vocabulary (right). All samples are generated with temperature $t = 1.0$ and top-k sampling with $k = 100$. Last row reports the speedup over the f1 baseline which operates directly on pixels and takes 7258 seconds to produce a sample on a NVIDIA GeForce GTX Titan X.

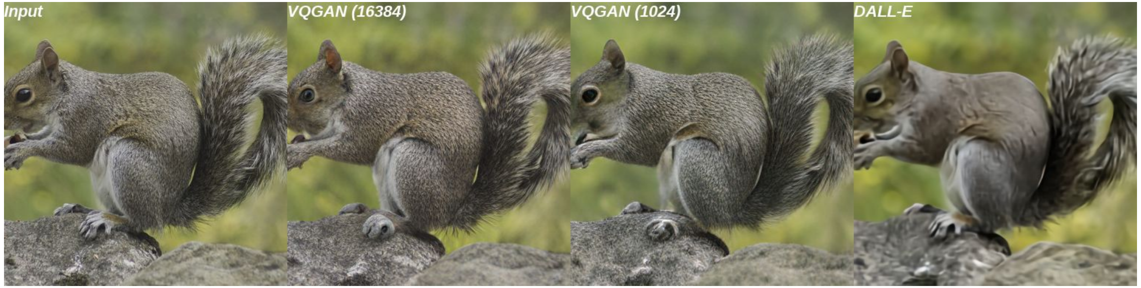


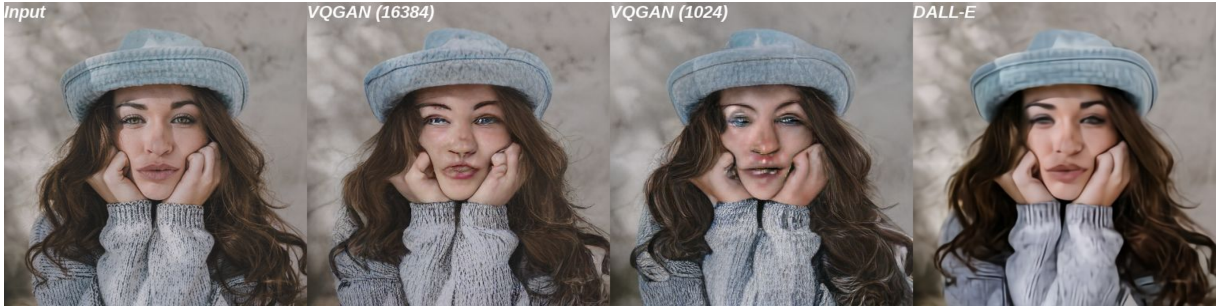
our *VQGAN*. We specify the amount of context encoded in terms of reduction factor in the side-length between image inputs and the resulting representations, *i.e.* a first stage encoding images of size $H \times W$ into discrete codes of size $H/f \times W/f$ is denoted by a factor f. For $f = 1$, we reproduce the approach of [8] and replace our *VQGAN* by a k-means clustering of RGB values with $k = 512$.
During training, we always crop images to obtain inputs of size 16×16 for the transformer, *i.e.* when modeling images with a factor f in the first stage, we use crops of size $16f \times 16f$. To sample from the models, we always apply them in a sliding window manner as described in Sec. 3.



Results Fig. 7 shows results for unconditional synthesis of faces on *FacesHQ*, the combination of *CelebA-HQ* [27] and *FFHQ* [29]. It clearly demonstrates the benefits of powerful *VQGAN*s by increasing the effective receptive field of the transformer. For small receptive fields, or equivalently small f, the model cannot capture coherent structures. For an intermediate value of $f = 8$, the overall structure of images can be approximated, but inconsistencies of facial features such as a half-bearded face and of viewpoints in different parts of the image arise. Only our full setting of $f = 16$ can synthesize high-fidelity samples. For analogous results in the conditional setting on S-FLCKR, we refer to the appendix (Fig. 10 and Sec. B).

4. Compared with VQVAE in DALLE and different vocab size

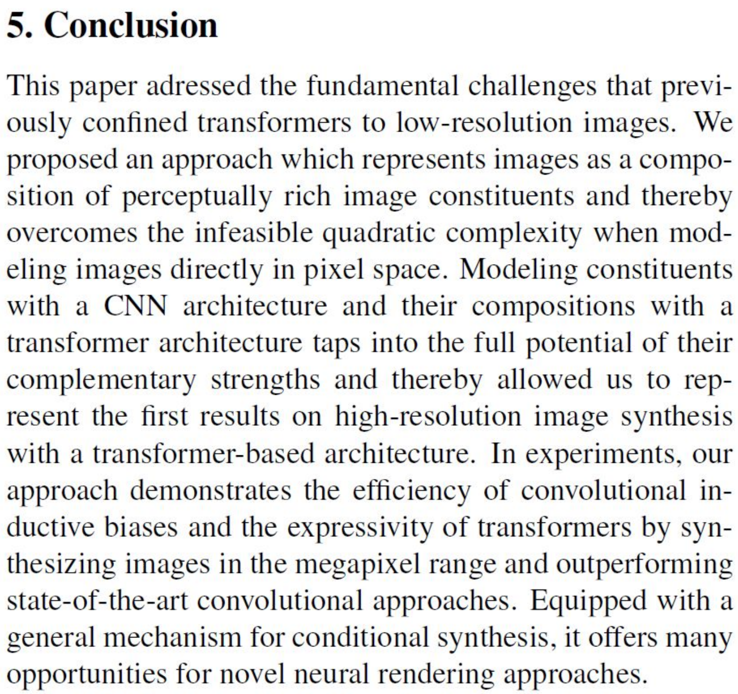




Faces are particularly difficult for the VQGAN to get right and the reconstructions of DALL-E's first stage appear more presentable. However, it should also be noted that the latter has been trained on a dataset which is roughly 400 times larger than the dataset (ImageNet) that the VQGAN was trained on. Thus, training the VQGAN on a larger dataset, or fine-tuning it on a dataset containing more faces, could improve the perceptual quality of reconstructed faces (and VQGANs trained on face datasets only do not show this problem).

https://colab.research.google.com/github/CompVis/taming-transformers/blob/master/scripts/reconstruction_usage.ipynb#scrollTo=DZCMKe-Ptapi

Conclusion



5. Conclusion

This paper adressed the fundamental challenges that previously confined transformers to low-resolution images. We proposed an approach which represents images as a composition of perceptually rich image constituents and thereby overcomes the infeasible quadratic complexity when modeling images directly in pixel space. Modeling constituents with a CNN architecture and their compositions with a transformer architecture taps into the full potential of their complementary strengths and thereby allowed us to represent the first results on high-resolution image synthesis with a transformer-based architecture. In experiments, our approach demonstrates the efficiency of convolutional inductive biases and the expressivity of transformers by synthesizing images in the megapixel range and outperforming state-of-the-art convolutional approaches. Equipped with a general mechanism for conditional synthesis, it offers many opportunities for novel neural rendering approaches.