PixelPyramids: Exact Inference Models from Lossless Image Pyramids
– Supplemental Material –

Shweta Mahajan[1] Stefan Roth[1,2]
[1]Department of Computer Science, TU Darmstadt [2] hessian.AI

We provide the proof of Lemma 3.1 in the main paper, as well as additional details on the datasets and the implementation. We further include additional qualitative examples.

A. Proof of Lemma 3.1

Lemma 3.1. *Let the sampled image* $\mathbf{I}$ *be of resolution* $[N_0, N_0, C]$*, then the worst-case number of steps* T *(length of the critical path) required is* $\mathcal{O}(\log N_0)$.

Proof. At the first sampling step, *i.e.* at the coarsest level, the spatial dimensionality of the image is $\left[N_0/2^{\lfloor (L+1)/2 \rfloor}, N_0/2^{\lfloor L/2 \rfloor}, C\right]$. At each level starting from the coarsest, the spatial resolution increases by a factor of 2, alternatingly along the rows and columns. Thus, to generate an image of size $[N_0, N_0, C]$ starting from an image of size $[1, 1, C]$, the number of levels of the coarse-to-fine pyramid to traverse equals

$$L = 2 \cdot \log N_0. \tag{8}$$

At the coarsest level of spatial resolution 1×1, with its fully autoregressive PixelCNN structure, the number of sampling steps is equal to the spatial dimension and, therefore, is given as

$$T_L = 1. \tag{9}$$

Let the number of squeeze operations applied at level i be n_{S_i} such that we obtain $4^{n_{S_i}}$ subsampled images from $\mathbf{F}_i$. This implies that the number of sequential steps at level i is

$$T_i = 4^{n_{S_i}}, \tag{10}$$

assuming that each autoregressive sampling step can be carried out in parallel (hence in constant time). Therefore, the total number of sequential steps (length of the critical path) required for sampling is

$$\begin{aligned} T &= \sum_{i=1}^{L} T_i = T_L + \sum_{i=1}^{L-1} T_i \\ &= 1 + \sum_{i=1}^{L-1} 4^{n_{S_i}} \end{aligned} \tag{11}$$

Under the assumption that the number of squeeze operations at any level is constant with $\mathcal{O}(1) \ni 4^{n_{S_i}} \ll N_0$, we obtain the the number of sampling steps required as

$$T \in 2 \cdot \log N_0 \cdot \mathcal{O}(1) = \mathcal{O}(\log N_0) \tag{12}$$

□

B. Additonal Implementation Details

Datasets. CelebA-HQ [23] consists of 30K images of which 26K are used for training, 1000 images for validation, and 3000 images are provided in the test set.

The LSUN [49] bedroom, church outdoor, and tower datasets consist of 3M, 126K, and 700K images, respectively, in the training set. The validation set consists of 300 images for each of the datasets. For training at 128 × 128 resolution, similar to [25, 31], the images are first center cropped to the spatial resolution of 256 × 256 and then resized to a spatial resolution of 128 × 128.

Further, ImageNet [38] consists of 1.3M images in the training set and 50K images in the test set. We follow [32, 35] for resizing the images to size 128 × 128, where the images are first cropped along the longer spatial dimension and then resized to the desired spatial resolution.

Optimization. Similar to [31, 39], we use Adam [24] with an initial learning rate of 10^{-3}. Additionally, the parameters are taken as $\beta_1 = 0.95$, $\beta_2 = 0.9995$ and Polyak averaging is set to 0.9995. The learning rate decays exponentially at a rate of 0.999995.

Dataset	Parameter Count
CelebA-HQ (256 × 256)	166M
CelebA-HQ (1024 × 1024)	349M
LSUN (128 × 128)	224M
ImageNet (128 × 128)	346M

Table 6. Parameter count of PixelPyramids across different datasets.

In Tab. 6 we provide the number of parameters for different datasets. The number of parameters varies depending on the size and the complexity of the dataset. Within the U-Net

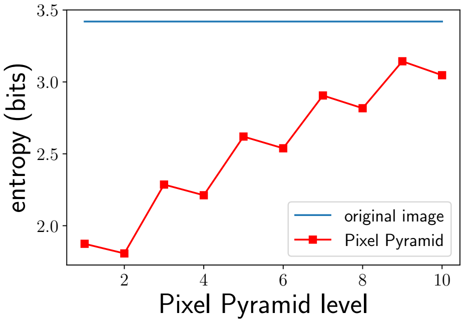

Figure 10. *Low-entropy decomposition.* Entropy values at different pyramid levels on ImageNet (128 × 128). In contrast to the other datasets considered in the main paper, the entropy values for the decomposition of ImageNet are not significantly reduced compared to the original image.

module at each level, the number of channels increases by a factor of two for every downsamling/upsampling layer. The first layer has 64 channels for the CelebA-HQ (256 × 256, 1024 × 1024) and LSUN (128 × 128) datasets. Owing to the multimodality of the dataset, ImageNet (128×128) even at 128 × 128 resolution requires same number of parameters as CelebA-HQ 1024 × 1024, as we need to increase the width of the first layer of the U-Net module to 128 channels. This design choice is further supported by the Paired Pyramid decomposition of ImageNet 128 × 128 (Fig. 10), which shows that the entropy is not reduced as significantly for the fine components as is the case for the CelebA-HQ dataset (*cf*. Fig. 4).[1]

C. Additional Results

In Fig. 11 we show the entropy values for the fine components of the Paired Pyramid representation of the low-resolution CIFAR10 [29] dataset. In comparison to the CelebA-HQ dataset (*cf*. Fig. 4), where the entropy values are significantly reduced, *e.g.* for $\mathbf{F}_1$ from 5.51 bits for the original pixel values to 3.14 bits for the fine component, the entropy of the fine components for the CIFAR10 dataset does not reduce significantly, even less so than for ImageNet (128 × 128). This limits the applicability of our PixelPyramids framework to such multimodal low-resolution datasets, since the fine components of the Paired Pyramid representation have similarly high entropy as the original image. This is further observed in Tab. 7, where the density estimates on CIFAR10 with fully autoregressive approaches are better compared to that with PixelPyramids with its partial autoregressive structure. We thus focus on high-resolution images here, since this is where the key limitations of existing exact inference models currently lie.

[1] The entropy values across different datasets are not comparable due to different preprocessing procedures.

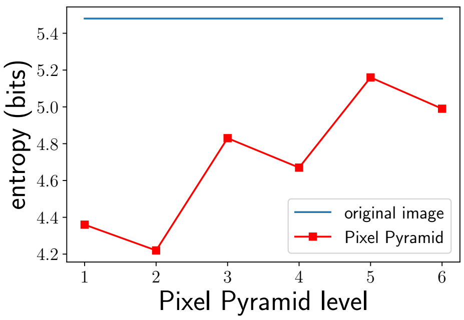

Figure 11. *Low-entropy decomposition.* Entropy values at different pyramid levels on CIFAR10. The entropy values for the decomposition are not significantly reduced compared to the original image for this multimodal low-resolution dataset.

Method	bits/dim. (↓)
PixelCNN [45]	3.03
PixelCNN++ [39]	2.92
Glow [25]	3.35
Flow++ [18]	3.08
Residual Flow [51]	3.28
MaCoW [31]	3.16
PixelPyramids *(ours)*	3.19

Table 7. Evaluation on the 8-bit CIFAR10 (32 × 32) dataset.

D. Additional Examples

To show that the pixel outliers in the images synthesized with PixelPyramids can indeed be resolved, we include Fig. 12, where we apply a median filter to remove the artifacts resulting from the cyclic shift of pixel values in the vicinity of 0 and 255 (Algorithm 1). The pixel outliers are detected using the Isolation Forest algorithm [52] in the HSV space and the outlier pixels are replaced using a median filter over an 7 × 7 neighborhood.

Algorithm 1: Improvement of images synthesized with PixelPyramids using pixel outlier detection. m is the filter size for the median filter and n is the pixel neighborhood for outliers.

```
Sample $\hat{\mathbf{I}} \sim p_{\theta,\phi_L}(\mathbf{I}_0)$ ;            // Sample the image
$\hat{\mathbf{I}}_{\text{med}} \leftarrow \texttt{medianfilter}(\hat{\mathbf{I}}, m)$;    // Median filter
$\hat{\mathbf{I}}_{\text{hsv}} \leftarrow \texttt{rgb2hsv}(\hat{\mathbf{I}})$;              // Convert to HSV
$\mathbf{O}_{\text{mask}} \leftarrow \texttt{isolationforest}(\hat{\mathbf{I}}_{\text{hsv}}, n)$;  // Outliers
$\hat{\mathbf{I}}[\mathbf{O}_{\text{mask}}] \leftarrow \hat{\mathbf{I}}_{\text{med}}[\mathbf{O}_{\text{mask}}]$;            // Assign median
```

We include additional qualitative examples obtained from our PixelPyramids on the high resolution datasets CelebA-HQ (256×256; Fig. 13), CelebA-HQ (1024×1024; Fig. 14), LSUN (bedroom, church outdoor, and tower, (128 × 128); Figs. 15 to 17), and ImageNet (128 × 128; Fig. 18). PixelPyramids can synthesize high quality and

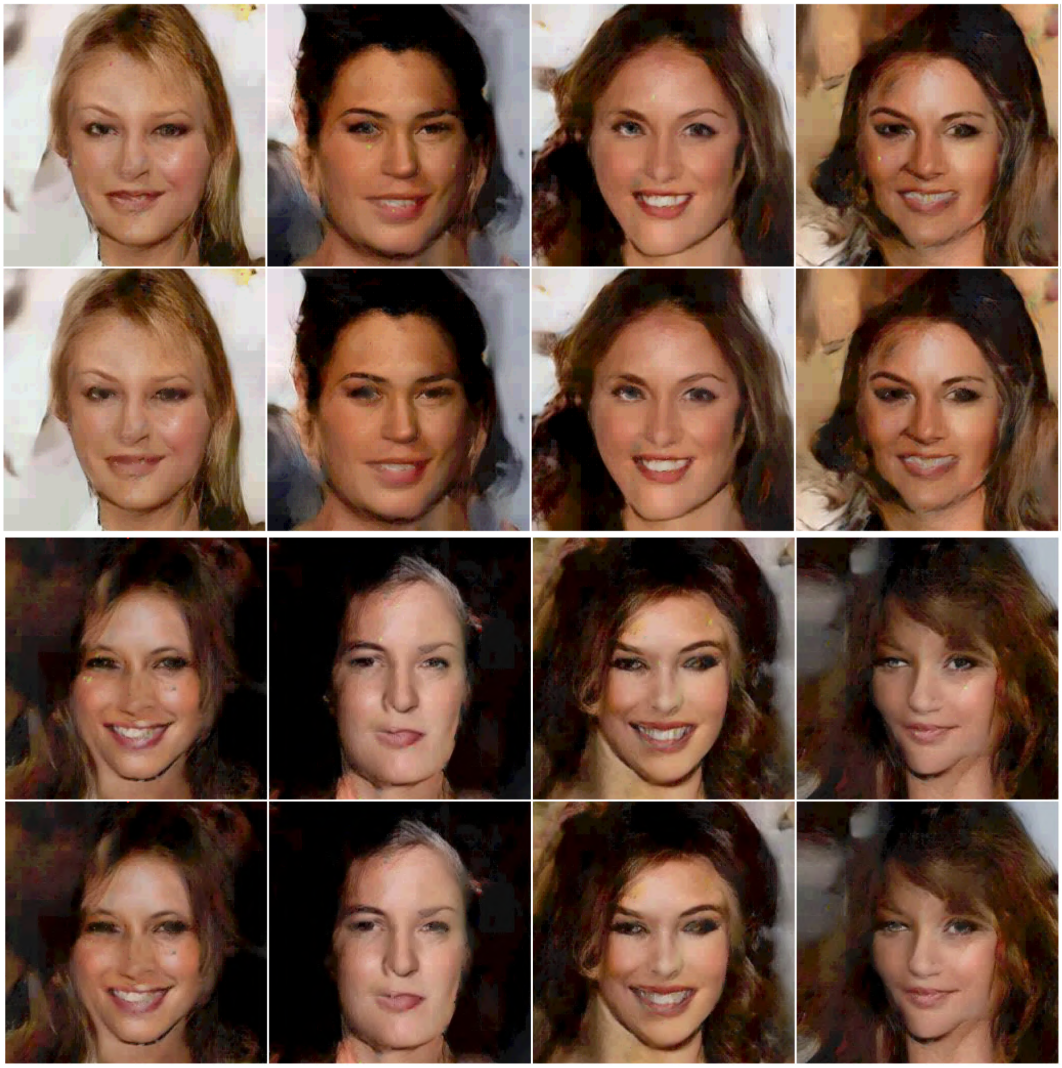

Figure 12. *Removal of pixel outliers from images synthesized with PixelPyramids on 5-bit CelebA-HQ (*256×256*):* Images generated with our PixelPyramids framework (*row 1 & 3*); generated images after the application of pixel outlier removal using a median filter (*rows 2 & 4*, see text).

diverse samples, capturing important visual properties in varied high-resolution datasets. Furthermore, in Figs. 19 and 20 we show the applicability of our PixelPyramids framework to the task of super-resolution, where fine details are iteratively added to the coarse input image (128×128) at every level of PixelPyramids to generate a high-resolution output with a spatial resolution of 1024×1024. We obtain an average PSNR($\uparrow$) of 27.25dB compared to 23.18dB for the baseline bicubic kernel [30] and LPIPS($\downarrow$) of 0.28 compared to 0.51 with a bicubic kernel.

References

[51] Jens Behrmann, Will Grathwohl, Ricky T. Q. Chen, David Duvenaud, and Jörn-Henrik Jacobsen. Invertible residual networks. In *ICML*, pages 573–582, 2019. 2

[52] Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. Isolation forest. In *ICDM*, pages 413–422. IEEE Computer Society, 2008. 2

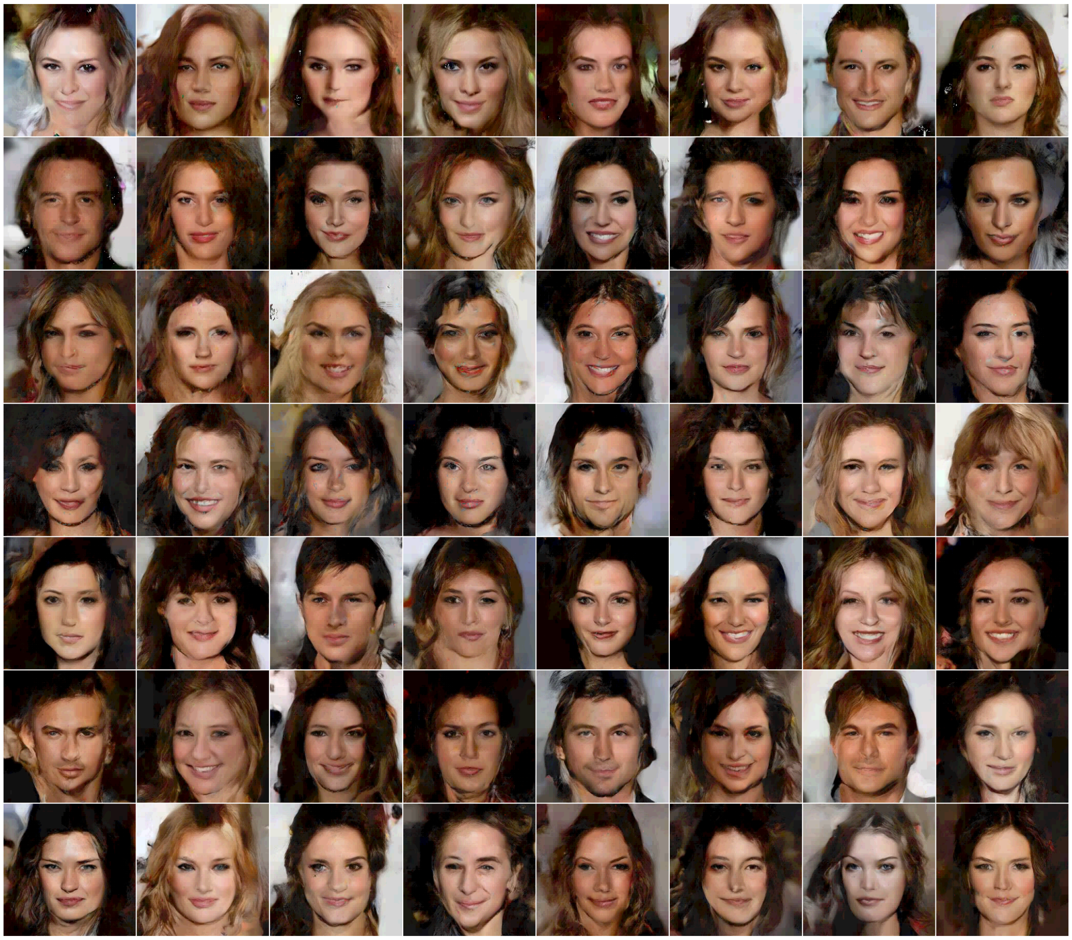

Figure 13. Random samples from 5-bit CelebA-HQ (256 × 256).

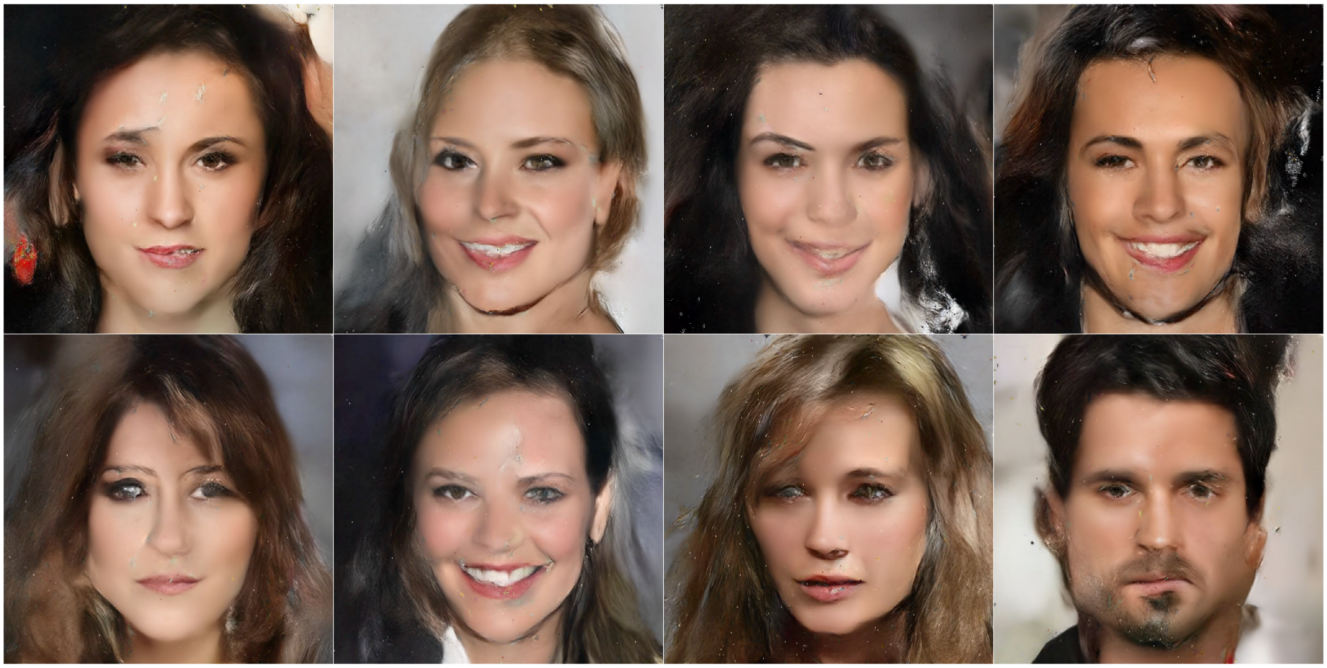

Figure 14. Random samples from 8-bit CelebA-HQ (1024×1024).

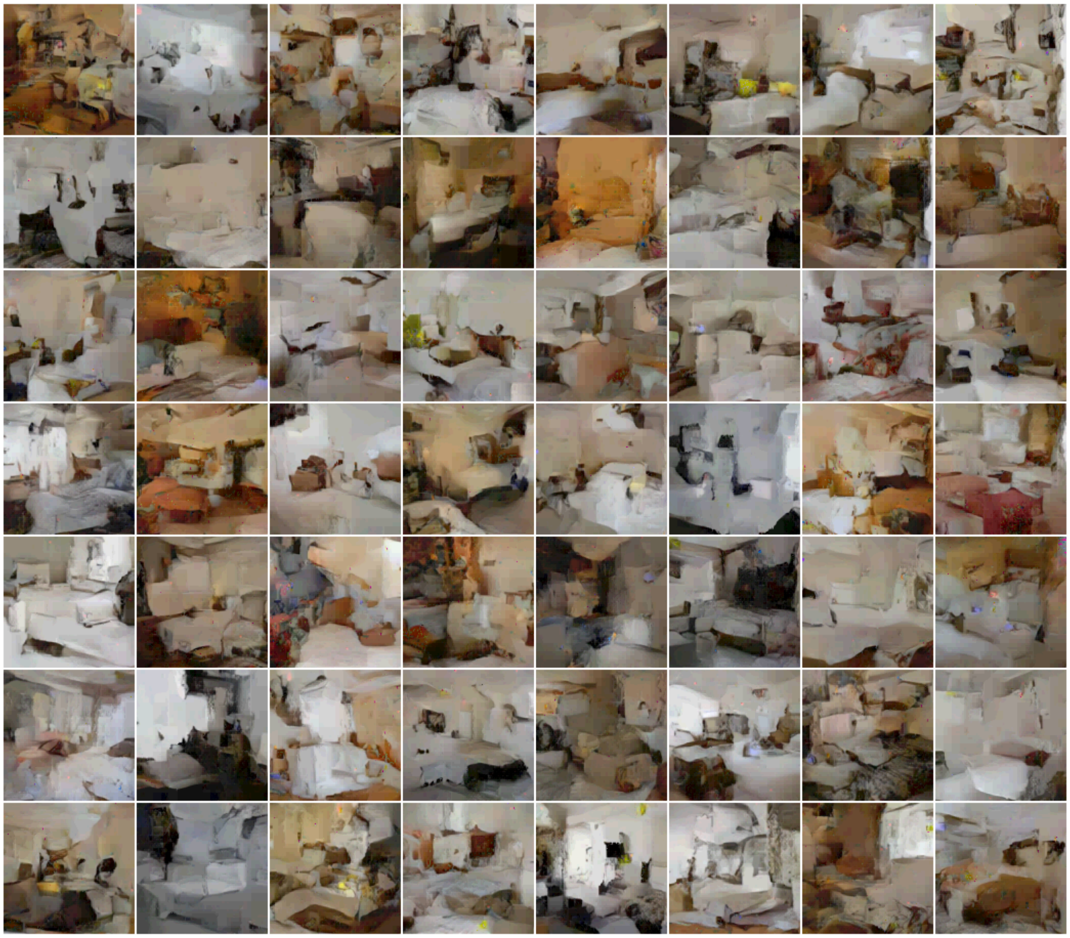

Figure 15. Random samples from 5-bit LSUN bedroom (128 × 128).

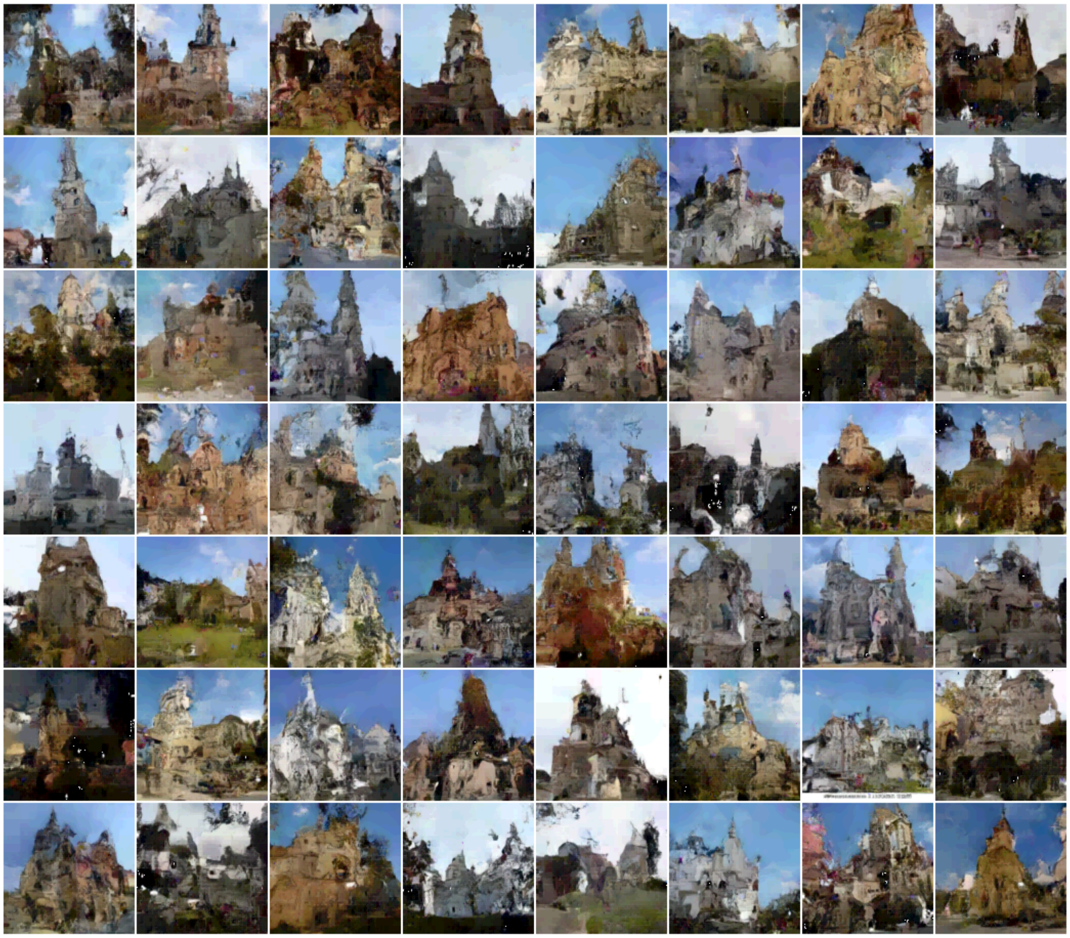

Figure 16. Random samples from 5-bit LSUN church (128 × 128).

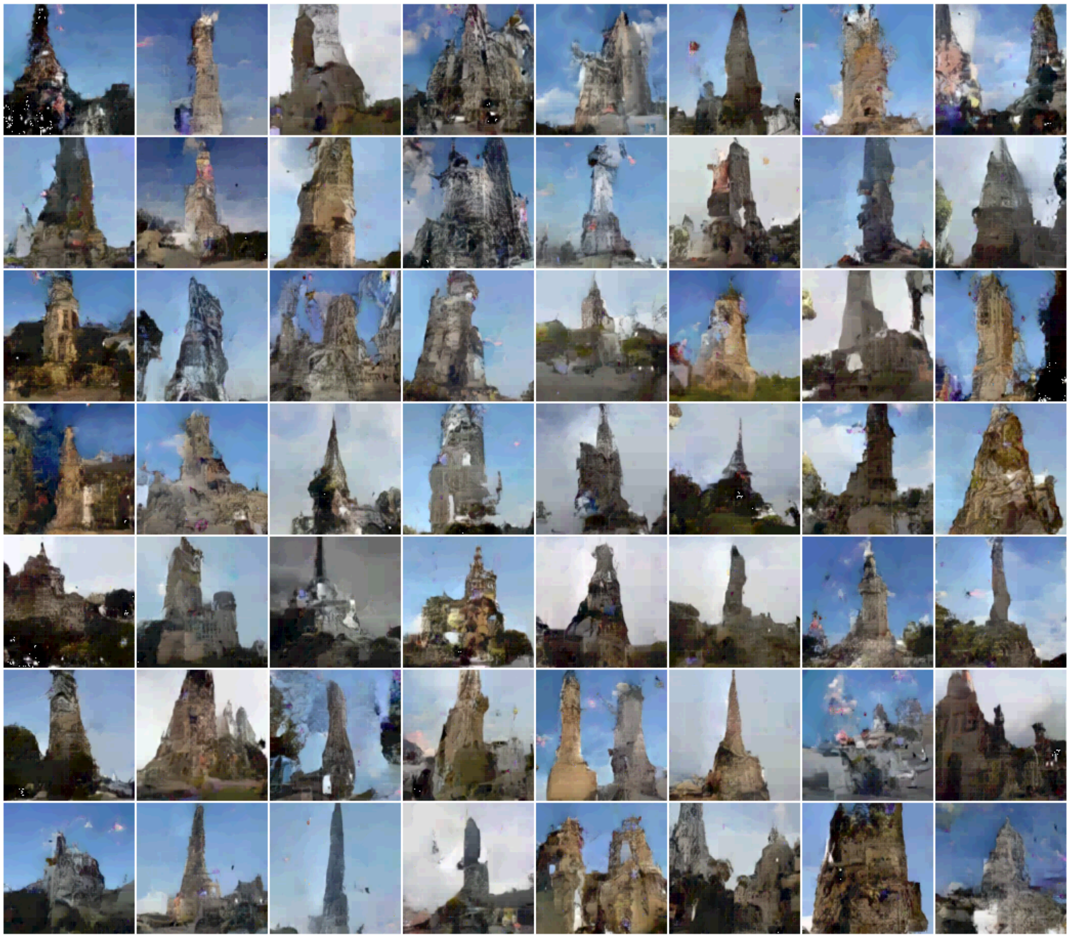

Figure 17. Random samples from 5-bit LSUN tower (128 × 128).

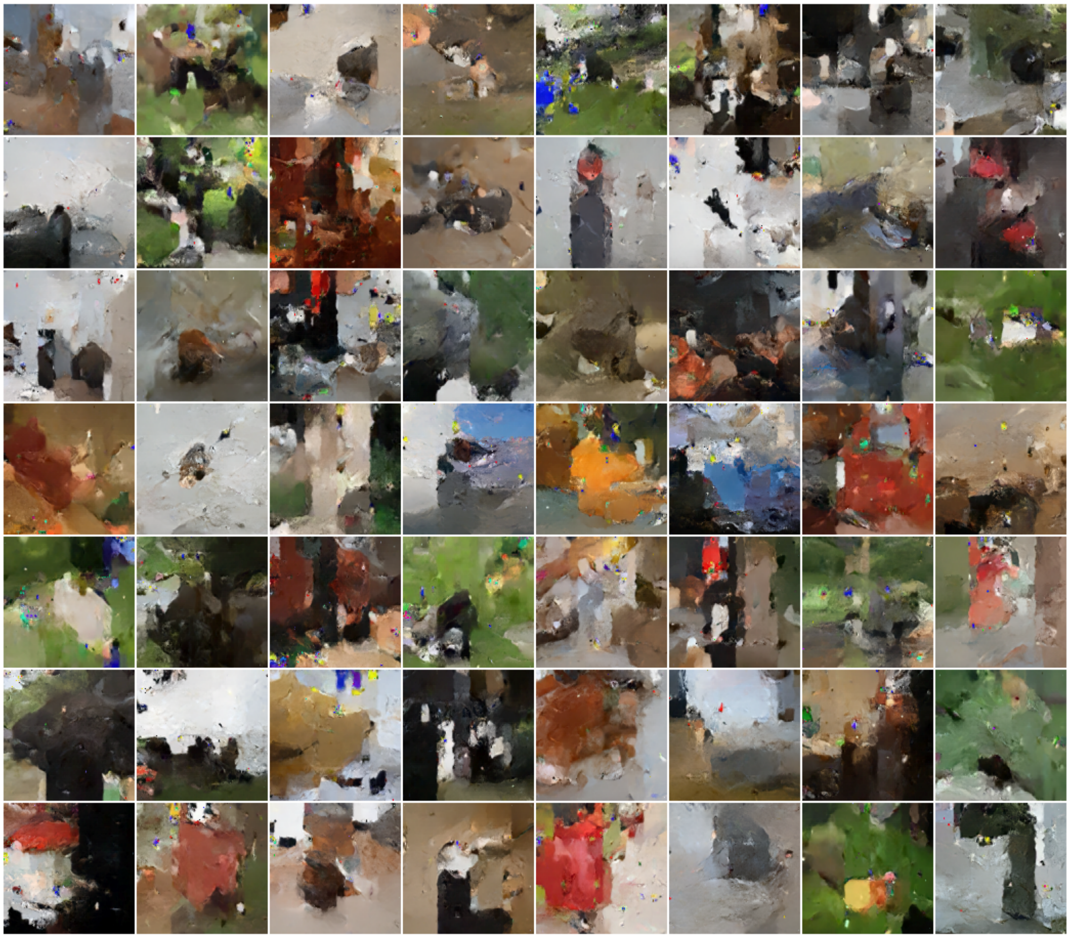

Figure 18. Random samples from 8-bit ImageNet (128 × 128).

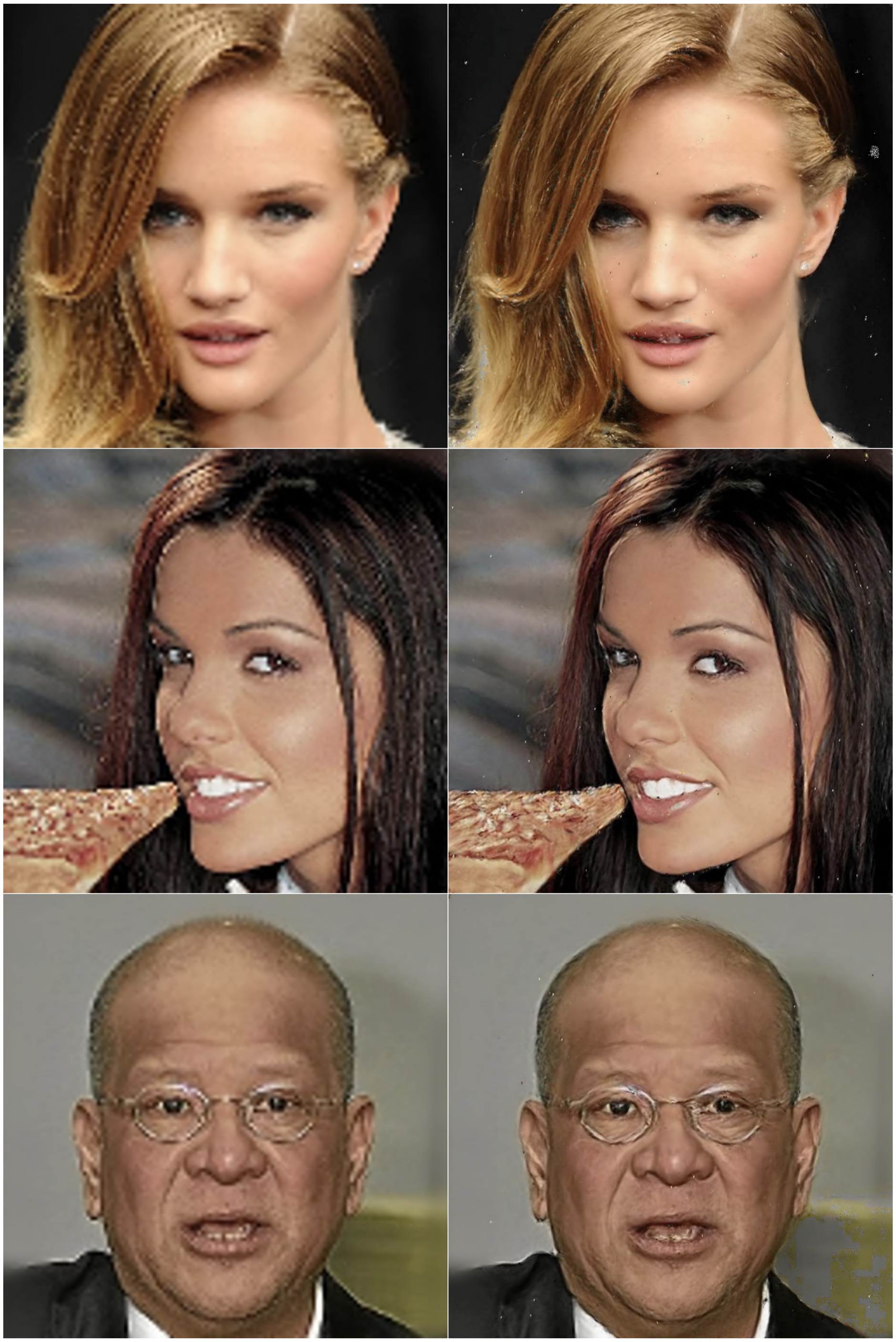

Figure 19. Super-resolution with PixelPyramids resizing a 128×128 image to 1024×1024 on the 8-bit CelebA-HQ (1024×1024).

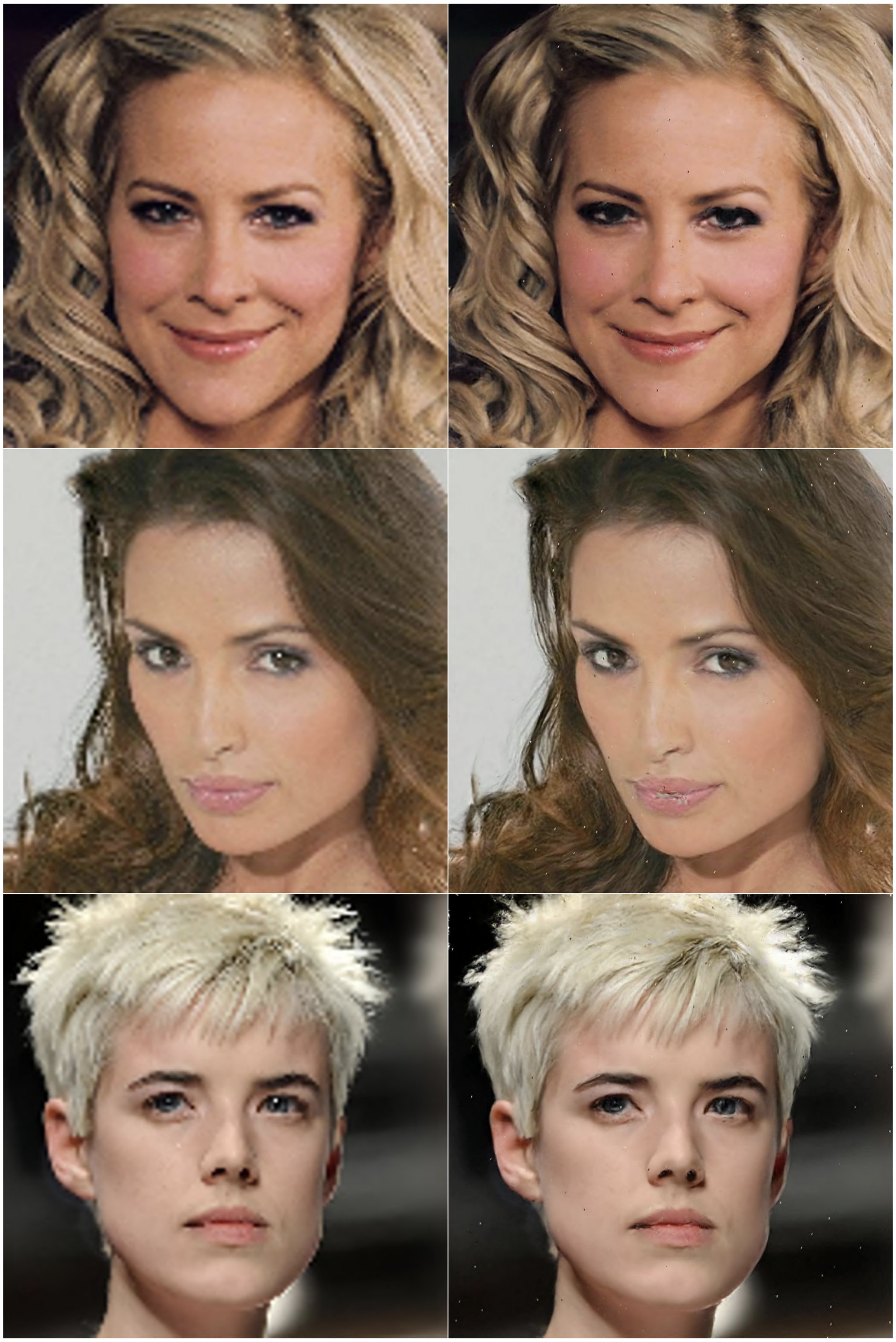

Figure 20. Super-resolution with PixelPyramids resizing a 128 × 128 image to 1024 × 1024 on the 8-bit CelebA-HQ (1024 × 1024).