# Attributing Statistics to Synthesis Quality in Correlation-Based Texture Models

**Anonymous Author(s)**
Affiliation
Address
email

## Abstract

Learning strong and interpretable representations for textures is fundamental in many computer vision tasks, particularly texture synthesis, where the aim is to match the intricate statistical patterns of one texture to generate new syntheses. With modern deep learning architectures, it is difficult to obtain interpretable and attributable features with their highly over-parameterized representation spaces despite their strong task performance. More traditional approaches to representing texture on the other hand, rely on highly interpretable, hand-picked statistic sets, but often at the cost of performance. In order to bridge the gap between these two approaches and obtain performant yet interpretable texture features, we introduce a new texture representation model. Our method combines the interpretable neuroscience-based multi-scale pyramid filter structure of traditional well-tested texture models with the power of pairwise-correlation approaches. This analysis-by-synthesis model generates texture images with similar quality to style-transfer based approaches. With our interpretable approach, we create a organizational structure for our statistics, breaking them into families, and evaluating the contribution of these families to synthesis quality. We then use contrastive learning to identify which statistics are most and least important for differentiating textures, and show that this ordering transfers to synthesis quality. By attributing synthesis quality to a subset of interpretable statistics, we are able to reduce the number of parameters to below that of previous methods while retaining similar or better synthesis quality.

## 1 Introduction

While deep-network-based texture models have been successful at generating synthetic textures, biologically inspired approaches grounded in neuroscience, such as multi-scale pyramid models, have gained renewed interest for their more interpretable and attributable features. These models capture key statistical properties of textures that align with both behavioral and physiological findings, allowing them to also serve as models of early visual processing in humans. For instance, Portilla and Simoncelli's parametric texture synthesis model [Portilla and Simoncelli, 2000] uses responses from V1-like filters to summarize textures through a set of summary statistics, mimicking the texture representation in early visual areas. Such models have proven to be effective not only in producing perceptually plausible textures but also in explaining neural responses to natural scenes [Ziemba et al., 2016]. They offer a biologically interpretable framework for understanding how textures are encoded, with the potential to explain phenomena including texture segregation, surface perception, and perceptual grouping. This interplay between neuroscience and texture synthesis holds promise for advancing our understanding of how the brain processes complex visual information in natural environments.

Submitted to 38th Conference on Neural Information Processing Systems (NeurIPS 2024). Do not distribute.

Here, we further this line of work by incorporating the power and flexibility of style-transfer based texture synthesis approaches into a biologically feasible and interpretable model. We demonstrate that our single-layer method can generate syntheses on par with the quality of deep-networks, while retaining interpretable features, for which we create an organizational framework of statistical families. We use this framework to attribute texture synthesis quality to a subset of the statistical families. Finally, we use contrastive learning to learn a reduced set of interpretable statistics, and show that this reduced set produces quality syntheses with a small parameter set.

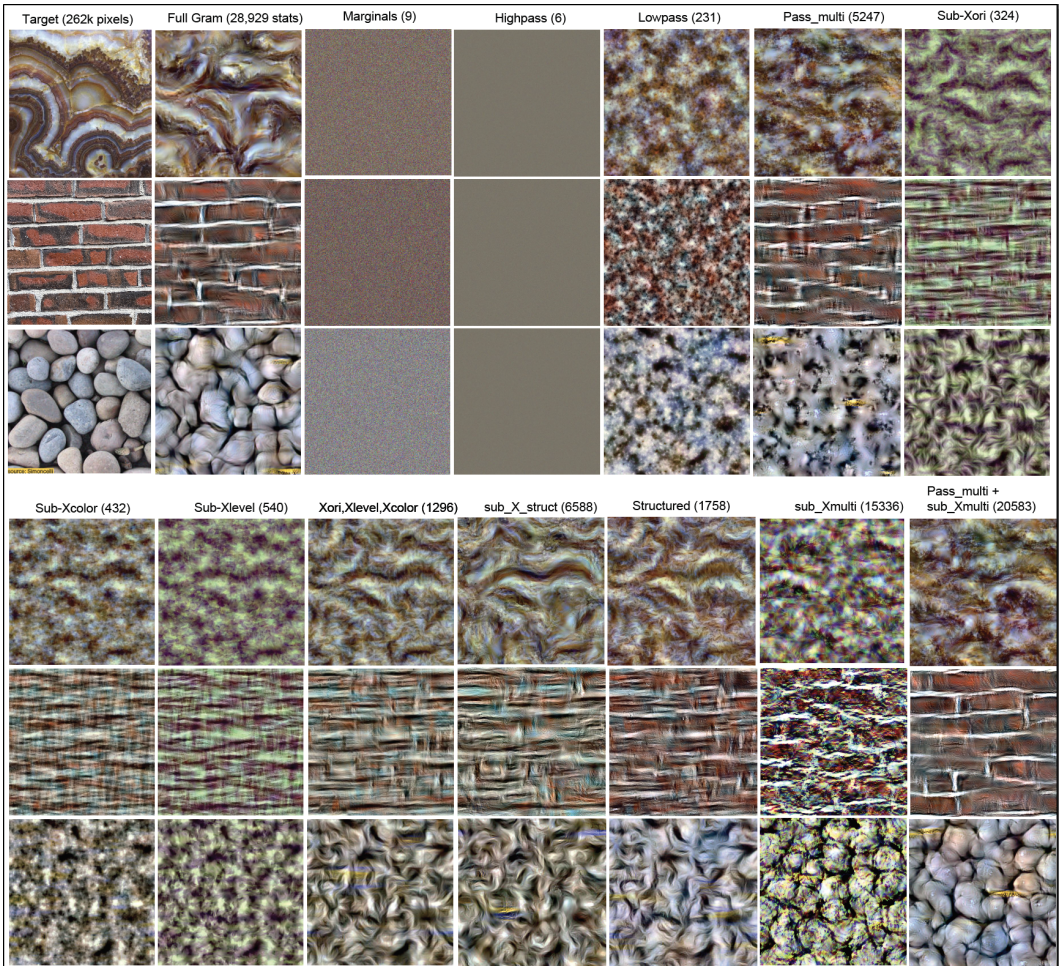


Figure 1: Depleted syntheses optimized using statistics subsets defined by categorical families. Families with 'structured' statistics pairs where only one parameter differs generate poor synthesis compared to larger families of unstructured pairs.

## 2 Previous Work

A wide body of research from the neuroscience, vision science, psychology, and computer science literature has worked towards creating valid, testable models of human texture perception. For texture synthesis specifically, approaches have historically ranged from modeling texture in pixel [Julesz, 1962], fourier [Matsuyama et al., 1983], and multi-scale pyramid [Burt and Adelson, 1987, Heeger and Bergen, 1995] space. Such texture models are often measured for quality as models of peripheral vision [Rosenholtz et al., 2012, Freeman and Simoncelli, 2011] using statistics sets similar to the Portilla & Simoncelli model [Portilla and Simoncelli, 2000]. The interpretability of these statistics sets has enabled the study of which textures succeed and fail [Brown et al., 2021], which statistics are necessary and sufficient [Koevesdi et al., 2023] for synthesis, and their interaction [Balas, 2006].

Progress in deep learning has enabled methods to synthesize spatial textures without hand-picked statistics sets. Matching the gram matrix from deep network layers [Gatys et al., 2015] as in style transfer [Gatys et al., 2016] has produced successful both texture and peripheral syntheses [Deza et al., 2017, Wallis et al., 2017], and been shown to work on single layer networks [Ustyuzhaninov et al., 2022]. While these methods are able to create successful syntheses, they have three major disadvantages: 1) They are highly over-parameterized compared to pyramid based models, with 1-2 orders of magnitude more statistics. 2) Unlike multi-scale pyramid representations, deep network hidden layers bear little resemblance to human visual neuroscience. 3) The statistics are based on correlations of a deep network, removing the interpretability that comes along with hand-picked statistics.

# 3 Model

We design a simple model (Fig. 4) that combines the interpretable and biologically-inspired pyramid filters of Portilla & Simoncelli [Portilla and Simoncelli, 2000] with the power and flexibility of the Gram matrix representation [Gatys et al., 2015]. Our model uses the multi-scale pyramid representation from [Brown et al., 2021], convolving each color opponent channel of an input image with pyramid filters at individual combinations of orientation, scale, and color. We then treat each pyramid image as a channel, calculating the pair-wise correlations between each pyramid image pair, collapsing over space, resulting in a Gram matrix. We use the upper triangle and diagonal of this matrix as the full statistics set, combined with a set of 3 marginal statistics per color channel for downstream analysis.

We use this as an analysis-by-synthesis model to generate novel texture images by matching the statistics of arbitrary input images. We show significant improvement over the standard multi-scale-pyramid synthesis method [Portilla and Simoncelli, 2000], and similar quality results to much larger, neural-network based synthesis methods [Gatys et al., 2015] (Figure 5).

# 4 Statistical Families

An advantage of our method is that our statisitcs are correlations between known pyramid filters that can be interpreted, and quality of syntheses attributed to different statistical families. We organize these families into two groups: one of statistics with one or more non-subband (pyramid level) filter image, and one group with correlations exclusively between subband images (Table A.4). We further organize the subband group into sub-groups, based on which properties are shared and differ between the two pyramid images correlated.

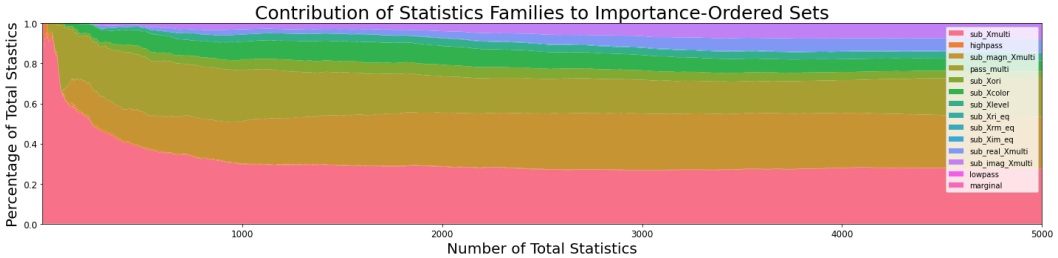


Figure 2: Using the contrastively learned importance ordering, we visualize the percent contribution of individual families to the total set, for varying statistics set size. Families sub_Xmulti, highpass, and pass_multi contribute most when selecting for the most important statisics (left).

The hand-curated statistics sets from previous models [Portilla and Simoncelli, 2000] rely heavily on correlation statistics from groups like the first 6 sub-band families, which we call 'structured' statistics. These share all properties except for one (i.e. sub_Xlevel share orientation, phase, and color at different scales). To attribute the quality of synthesis to different family groups, we performed ablation experiments, synthesizing textures with subsets of the statistical families (Figure 1). Interestingly, we find that these 'structured' correlation statistics generate poor synthesis even

when combined together. By contrast, unstructured statistics generate superior syntheses, though we note these represent a majority of the statistics.

## 5 Parameter Reduction

To further investigate the contribution of individual texture families to synthesis quality, we use contrastive learning to reduce the correlation statistics to a 100 element feature vector (Figure 4). We crop texture images from [Cimpoi et al., 2014], and train a single fully connected layer using InfoNCE [Oord et al., 2018] to group texture crops from the same parent texture image in latent space. We find that the single trained layer is sufficient to group same textures in embedding space (Fig. 6).

Next, to evaluate the importance of different statistics to the contrastive learning task, we ordered statistics from most to least important based on both the absolute value of their weightings (Figure 7), as well as by their Shapley values [Roth, 1988] (not shown, similar results). Again, we find correlations between sub-band filter outputs with few or no shared attributes (sub_Xmulti) are most important to the contrastive learning task. By contrast, correlations between lowpass filter outputs are least important for grouping textures. We visualize the relative contributions of individual families to the total statistics set when ordered by importance (Figure 2), and find that sub_Xmulti, lowpass, and pass_multi families make an out-sized contribution to the most important statistics.

Finally, we evaluate the transfer of importance in the contrastive learning task to synthesis quality. To do this, we again perform an ablation experiment, but selectively optimize selecting statistics based on their importance to the contrastive learning task (Figure 3). Using the most important statistics, we are able to achieve good syntheses that outperform the Portilla & Simoncelli model with only 5,000 of the total 29,000 correlation statistics, for a reduction of 83%. Statistic groups of the same size from the least important statistics set are extremely poor, and randomly selected are only slightly better (Figure 8).

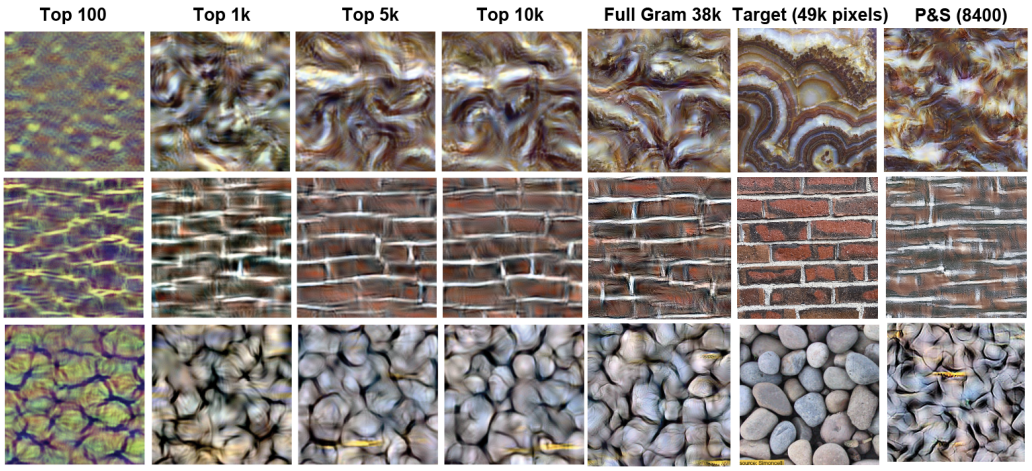


Figure 3: Depleted syntheses optimized using statistics subsets defined by order from contrastive learning. Top ordered groups demonstrate high quality syntheses, despite small set sizes.

## 6 Discussion

We present a human vision inspired texture model that synthesizes textures using pyramid-based correlation statistics, bridging insights from neuroscience and machine learning to incorporate both biological plausibility and interpretability. We demonstrate quality texture syntheses from our full model, then combine our model with contrastive learning to reduce the number of parameters significantly, and demonstrate quality syntheses with a reduced model. Finally, we provide a method of categorizing correlation statistics into families for interpretability, enabling future work to explore the relative contributions of these statistics to the final syntheses.

# References


Benjamin J Balas. Texture synthesis and perception: Using computational models to study texture representations in the human visual system. *Vision research*, 46(3):299–309, 2006.

Rachel Brown, Vasha DuTell, Bruce Walter, Ruth Rosenholtz, Peter Shirley, Morgan McGuire, and David Luebke. Efficient dataflow modeling of peripheral encoding in the human visual system. *arXiv preprint arXiv:2107.11505*, 2021.

Peter J Burt and Edward H Adelson. The laplacian pyramid as a compact image code. In *Readings in computer vision*, pages 671–679. Elsevier, 1987.

Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3606–3613, 2014.

Arturo Deza, Aditya Jonnalagadda, and Miguel Eckstein. Towards metamerism via foveated style transfer. *arXiv preprint arXiv:1705.10041*, 2017.

Jeremy Freeman and Eero P Simoncelli. Metamers of the ventral stream. *Nature neuroscience*, 14(9): 1195–1201, 2011.

Leon Gatys, Alexander S Ecker, and Matthias Bethge. Texture synthesis using convolutional neural networks. *Advances in neural information processing systems*, 28, 2015.

Leon A Gatys, Alexander S Ecker, and Matthias Bethge. Image style transfer using convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2414–2423, 2016.

David J Heeger and James R Bergen. Pyramid-based texture analysis/synthesis. In *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, pages 229–238, 1995.

Bela Julesz. Visual pattern discrimination. *IRE transactions on Information Theory*, 8(2):84–92, 1962.

Christian Koevesdi, Vasha DuTell, Anne Harrington, Mark Hamilton, William T Freeman, and Ruth Rosenholtz. Stattexnet: Evaluating the importance of statistical parameters for pyramid-based texture and peripheral vision models. In *NeuRIPS 2023 Workshop on Gaze Meets ML*, 2023.

Dong C Liu and Jorge Nocedal. On the limited memory bfgs method for large scale optimization. *Mathematical programming*, 45(1):503–528, 1989.

Takashi Matsuyama, Shu-Ichi Miura, and Makoto Nagao. Structural analysis of natural textures by fourier transformation. *Computer vision, graphics, and image processing*, 24(3):347–362, 1983.

L. McInnes, J. Healy, and J. Melville. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. *ArXiv e-prints*, February 2018.

Leland McInnes, John Healy, Nathaniel Saul, and Lukas Grossberger. Umap: Uniform manifold approximation and projection. *The Journal of Open Source Software*, 3(29):861, 2018.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

Javier Portilla and Eero P Simoncelli. A parametric texture model based on joint statistics of complex wavelet coefficients. *International journal of computer vision*, 40(1):49–70, 2000.

Ruth Rosenholtz, Jie Huang, Alvin Raj, Benjamin J Balas, and Livia Ilie. A summary statistic representation in peripheral vision explains visual search. *Journal of vision*, 12(4):14–14, 2012.

Alvin E Roth. *The Shapley value: essays in honor of Lloyd S. Shapley*. Cambridge University Press, 1988.

Ivan Ustyuzhaninov, Wieland Brendel, Leon Gatys, and Matthias Bethge. What does it take to generate natural textures? In *International conference on learning representations*, 2022.

Thomas Wallis, Christina Funke, Alexander Ecker, Leon Gatys, Felix Wichmann, and Matthias Bethge. Towards matching peripheral appearance for arbitrary natural images using deep features. *Journal of Vision*, 17(10):786–786, 2017.

Corey M Ziemba, Jeremy Freeman, J Anthony Movshon, and Eero P Simoncelli. Selectivity and tolerance for visual texture in macaque v2. *Proceedings of the National Academy of Sciences*, 113 (22):E3140–E3149, 2016.

# A Appendix

## A.1 Model

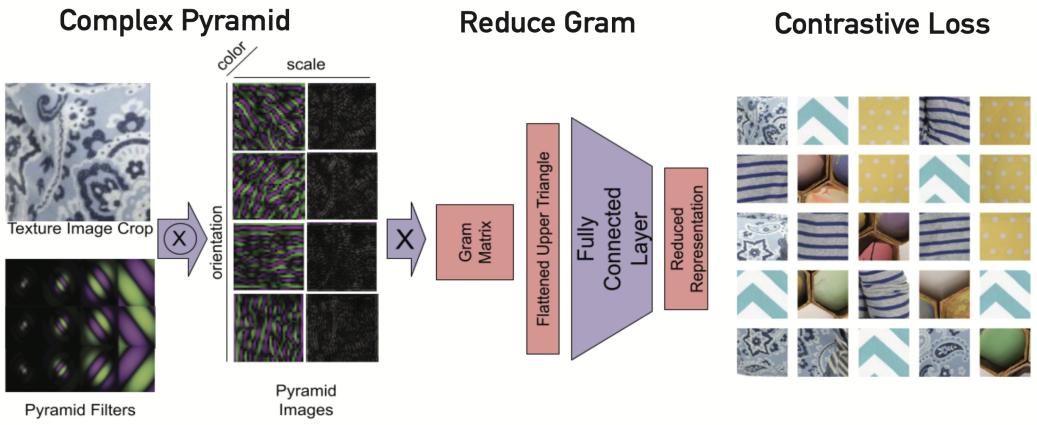


Figure 4: Our model decomposes images into pyramid stacks with varying scales, orientations, and color using a multi-scale pyramid filter bank. Pair-wise correlations between all pyramid images are represented in the gram matrix. The upper triangle of this gram matrix represents the full statistics set. To learn a reduced statistics set, a single fully connected layer compresses to a reduced representation. This fully connected layer is trained using a contrastive loss, self-supervised by training on crops from the same texture image.

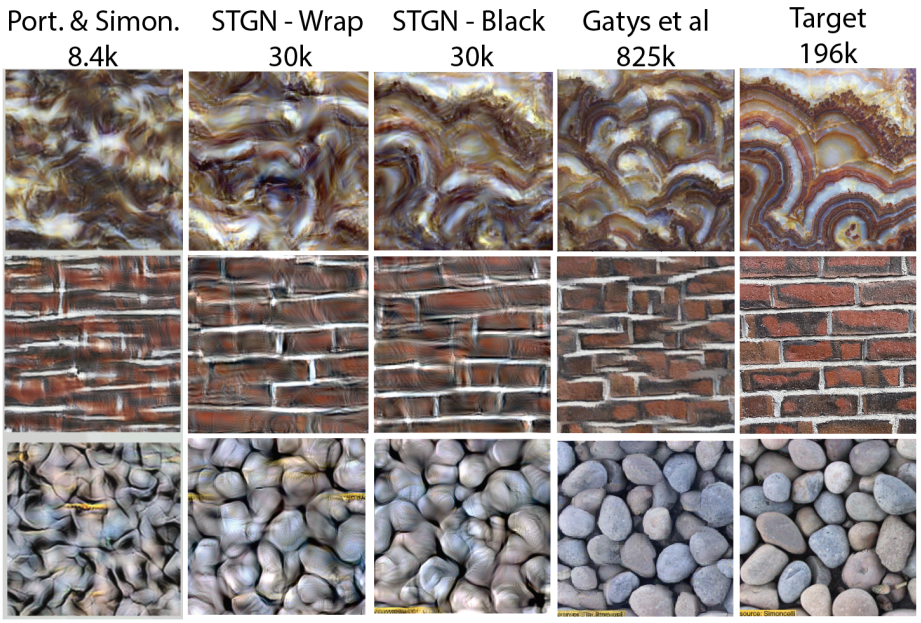


Figure 5: We generate texture image syntheses with better quality than Portilla & Simoncelli, and with similar performance to Gatys et al, but with a smaller model. STGN-Wrap uses a wrapped Fourier transform, conserving purely spatially-invariant information, while STGN-Black uses zero-padding, and retains more spatial image content. Gatys et al synths are from unofficial Pytorch repo for [Gatys et al., 2015] (https://github.com/trsvchn/deep-textures)

## A.2 Contrastive Learning

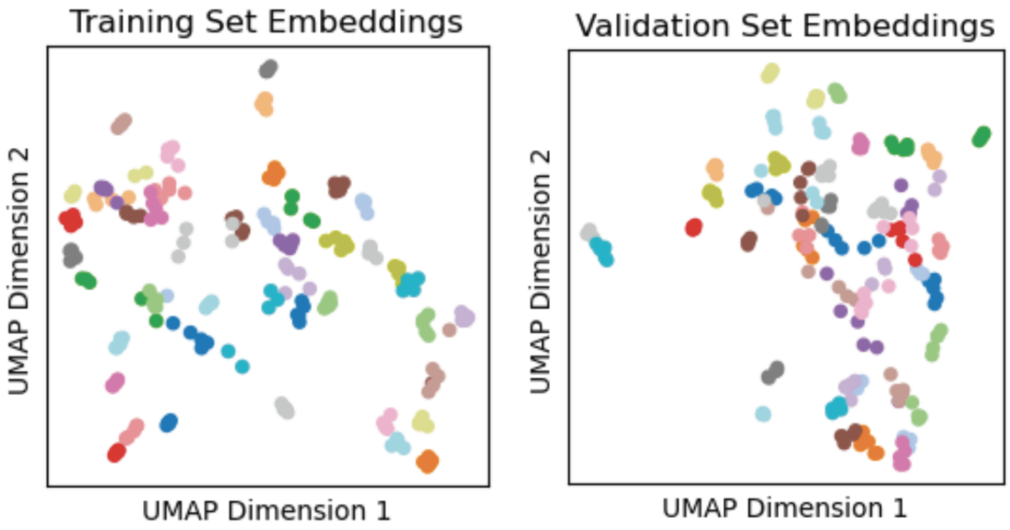


Figure 6: We verify the success of training with UMAP [McInnes et al., 2018, McInnes et al., 2018] to visualize the reduced latent space trained through contrastive learning. For both the training set (left), and validation set (right), crops from the same texture (plotted as same color) lie very close in latent space.

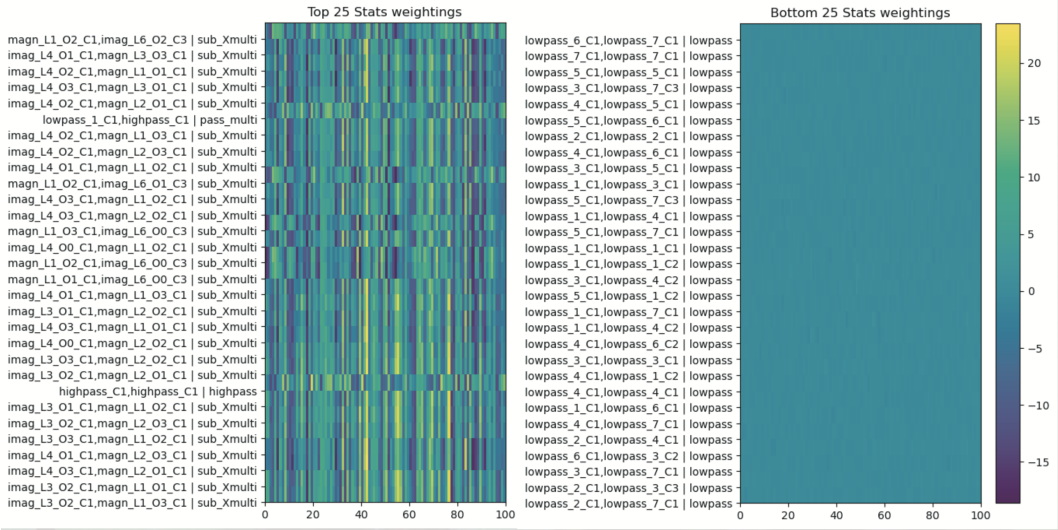


Figure 7: Correlation statistics by the absolute value of the magnitude of their learned weightings. Highest weighted correlation statistics (left) are from the sub_Xmulti group, corresponding to correlations between subband filter outputs with few or no shared attributes. Lowest weighted correlation statistics (right) are from the lowpass group, corresponding to correlation statistics between lowpass filter outputs.

## A.3 Synthesis Procedure

To synthesize images, we first seed a noise image, initializing a tensor with a random normal distribution in the range [0,1] and of the same size as our target image. We then use the codebase

from [Brown et al., 2021] to decompose the image into 3 color-opponent channels, and convolve each channel with with a pyramid filter bank with 4 spatial scales, real/imaginary pyramids, plus 3 marginal statistics per color channel. We then follow the method of [Gatys et al., 2015], calculating the gram matrix of the filter responses, collapsing over the orientation, scale, and color channel dimensions, measuring the pairwise correlation between all filter outputs. We reduce this to the upper-triangle plus diagonal. We calculate the MSE loss on the difference between these statistics for the target and synthesis video, and back-propagate this loss using the L-BFGS optimizer [Liu and Nocedal, 1989] using default hyper-parameters. We run 1000 iterations for each synthesis, which takes approximately 2 minutes on a single GPU.

### A.4 Statistics Families

| **Non-Subband Pyramid Stats** | | | |
|---|---|---|---|
| **Group Name** | **Stat A** | **Stat B** | **Number Stats** |
| marginal | mean ∣ var ∣ std | mean ∣ var ∣ std | 9 |
| highpass | highpass | highpass | 6 |
| lowpass | lowpass | lowpass | 231 |
| pass_multi | highpass ∣ lowpass | X | 5247 |
| **Subband Pyramid Stats** | | | |
| **Group Name** | **Stats Same** | **Stats Differ** | **Stats** |
| sub_Xori | Level, Color, Pyr | Ori | 324 |
| sub_Xcolor | Level, Ori, Pyr | Color | 432 |
| sub_Xlevel | Color, Ori, Pyr | Level | 540 |
| sub_Xri_eq | Level, Color, Ori | RealXImag | 72 |
| sub_Xrm_eq | Level, Color, Ori | RealXMag | 72 |
| sub_Xim_eq | Level, Color, Ori | ImagXMag | 72 |
| sub_real_Xmulti | Real | Level, Color, Ori | 2196 |
| sub_imag_Xmulti | Imag | Level, Color, Ori | 2196 |
| sub_magn_Xmulti | Magn | Level, Color, Ori | 2196 |
| sub_Xmulti | - | Pyr, Level, Color, Ori | 15336 |

### A.5 Ablation Control

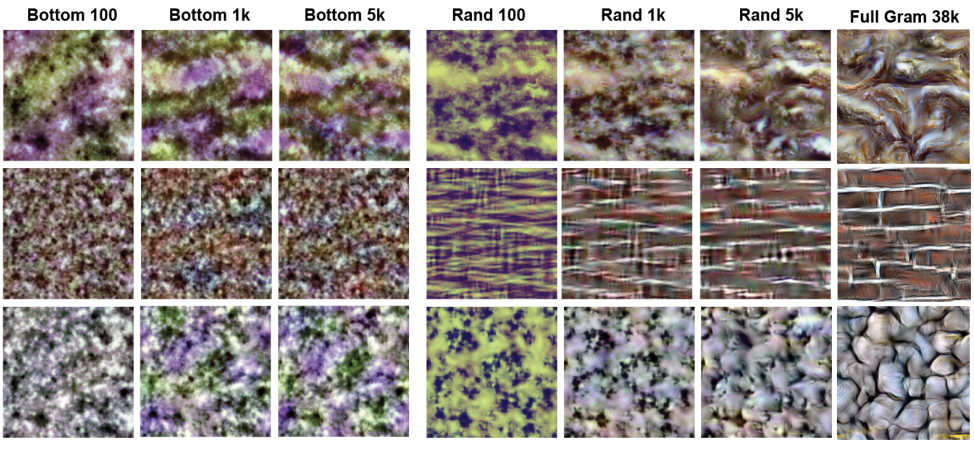


Figure 8: Depleted syntheses optimized using statistics subsets defined by reverse order from contrastive learning.