

- *Supplementary Material* -
Real-Time Neural Materials using Block-Compressed Features

C. Weinreich*, L. De Oliveira*, A. Houdard* and G. Nader*
Ubisoft La Forge

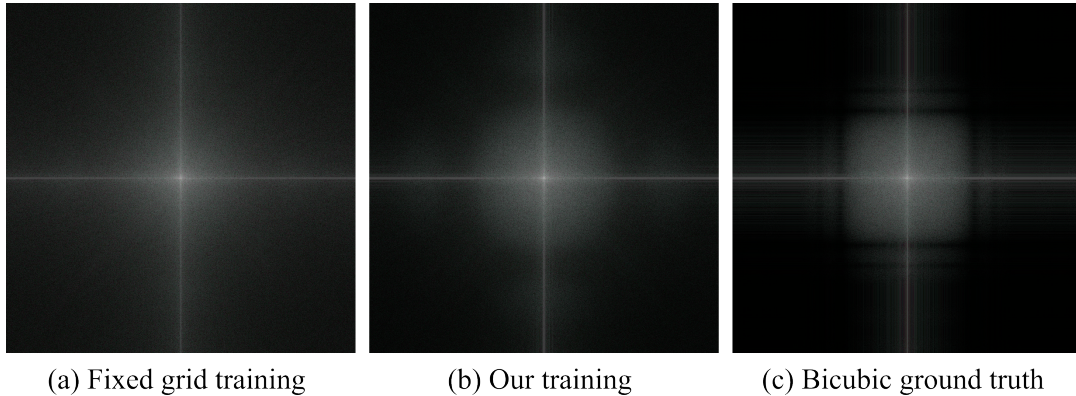


Figure 1: Magnitude of the Discrete Fourier Transform of a $\times 4$ magnification of the diffuse part of Ukulele model for (a) fixed grid training (b) our continuous training and (c) the ground-truth magnification with bicubic filter.

1 High-frequency behavior

Training large neural networks on a discrete grid may yield high-frequency artifacts when sampling in-between the points of the grid. Indeed, when no additional regularization is added, overfitting a highly non-linear model on a discrete set of point does not constraint the behavior outside these locations. This may cause undesired behavior in a rendering context, such as flickering effects or high-frequency artifact when close. In order to mitigate this issue, [VSW*23] rely on post-process filtering. Our approach however does not suffer from this issue for three main reasons

1. Our training that aims at minimizing L2 functional norm between the filtered ground truth and the model acts as a regularizer as illustrated in Figure 1.
2. MLP are K-Lipschitz, and point 1. together with the fact that our decoder is very small keep K small. As a result this limits its high-frequency response. This is illustrated in Table 1, that shows the computed spectral norm upper bound of the MLP in various scenarios.
3. We are not using positional encoding, and the bilinear interpolations of fixed resolution features we use as input of the decoder are frequency limited by the highest resolution feature.

Figure 2 together with Lipschitz bound from Table 1 illustrate these points with a $\times 16$ magnification of an 8×8 area. With our continuous training approach, both 64×2 and 16×1 hidden layers MLP show limited high-frequency responses whereas with a fixed-grid training, the 64×2 MLP has more high-frequency (visible in the top-left to bottom-right diagonal of the Fourier spectrum). Another point to note is that adding positional embedding together with the fixed grid training as in [VSW*23] drastically increase the high-frequency response. This allows the method from [VSW*23] to reconstruct better some high frequency details at a cost of artifacts in extreme magnification cases.

*Equal Contribution, order determined by coin toss

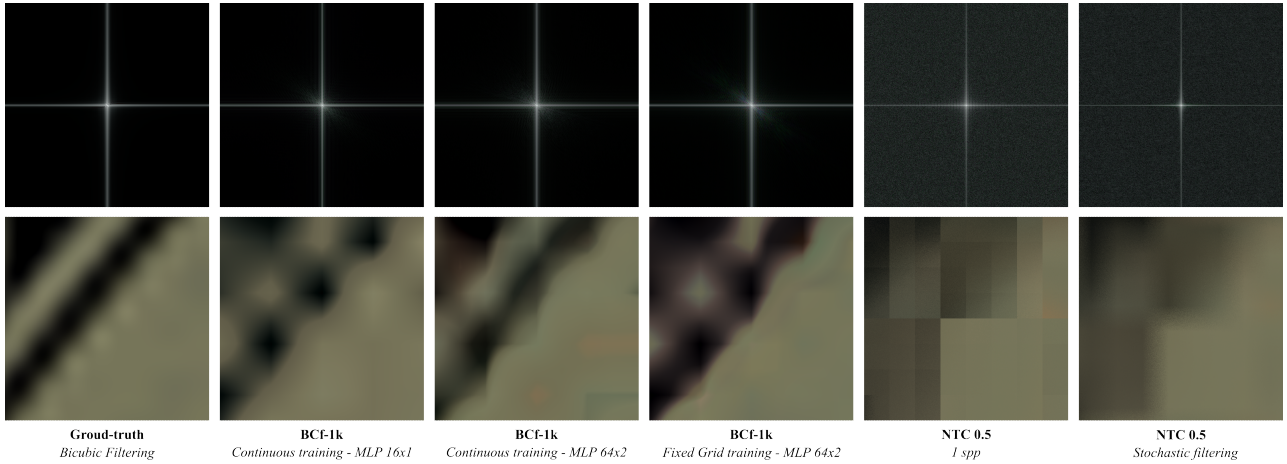


Figure 2: $16\times$ magnification zoom on an 8×8 pixel area of Ukulele model. (a) Magnitude of the Discrete Fourier Transform (b) Diffuse. From left to right, the ground truth bicubic interpolation ($\times 16$), our **BCf-1k** method as described in the paper, the same with a larger 64×2 MLP, a fixed-grid training with same parameters, the result of NTC 0.5 if directly sampled at 1 spp and NTC 0.5 with stochastic filtering.

Regularization effect of training on continuous uv The training process using uniformly sampled uv coordinates rather than regularly-spaced uv grids acts as a regularizer. The regularizing effect can be measured by estimating an upper-bound of the Lipschitz constants of the decoder MLP. In our experiments, training on uniformly sampled uvs produced MLPs with a lower upper-bound of the Lipschitz constant. In practice, an MLP with a lower Lipschitz constant would produce a smoother output as the rate of change is bounded by a lower value. Table 1 shows measured upper bounds for different training processes. The upper-bound is computed by multiplying the largest singular-value of each layer of the MLP.

Table 1: Evaluation of a spectral-norm derived upper-bound of the MLP Lipschitz coefficient K for different scenarios.

	Spectral Bound
BCf-1k 16×1	5.75
BCf-1k 64×2	241.54
BCf-1k 64×2 + fixed grid	1029.02
NTC 0.5	5213.04

2 Optimal BC Partition selection

In order to supplement the experiment from the paper about partition selection method, we constructed a toy image designed to have an optimal BC partition set. We then learned a custom model of one BC neural feature of the same resolution than the target toy image and ran our method with our partition selection heuristic then with random partition selection. The results are presented in Figure 4. This illustrates that our heuristic is able to retrieve the right partition set when it exists, and that choosing a bad partition set can downgrade the quality of the reconstruction (more than 16dB difference in PSNR in this case).

3 Larger Network

We also experimented with pairing our neural BC features with a larger network with 2 hidden layers of size 64 and positional encoding similar to the ones in [VSW*23], making the size of the input grow to 20. While this larger architecture yielded better results (see. table 4), the increase in performance is marginal, not to mention the potential impact on temporal stability (as discussed in section 1). We found that having a neural texture set according to the

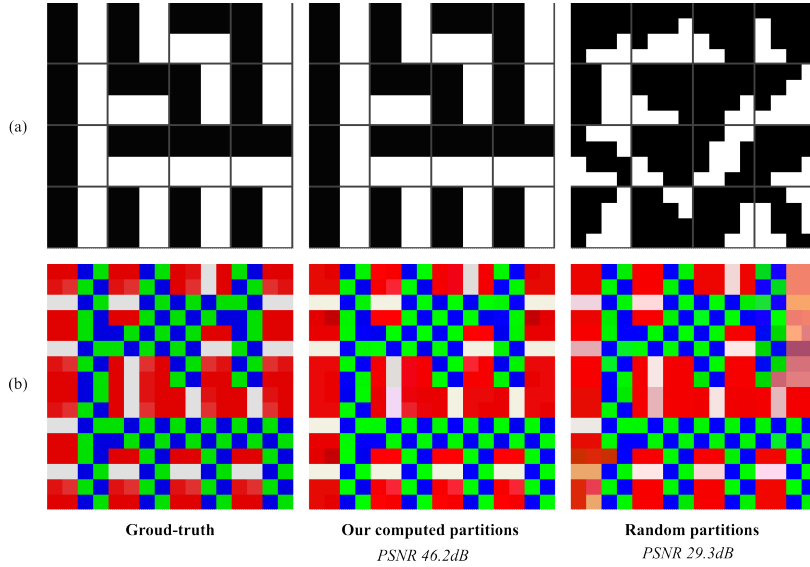


Figure 3: Left, Designed toy image (a) with its optimal BC partition brute force computed (b). Compressed image (a) with partitions (b) found with our propose heuristic. Compressed image (a) with random partitions (b).

Table 2: Evaluation of the reconstruction metrics (PSNR, SSIM and FLIP) on the whole dataset when pairing the BCf-feature with a 64×2 MLP and grid positional encodings [VSW*23]

	BCf-0.5K	BCf-1K	BCf-2K	BCf-2K++
PSNR (\uparrow)	28.79	32.30	36.29	38.13
SSIM (\uparrow)	0.80	0.88	0.95	0.97
FLIP (\downarrow)	0.148	0.096	0.066	0.057

BCf-2k++ configuration, emulating having 6 channels in the first layer of neural features, have more impact on the reconstruction quality than increasing the size of the network. This suggests that, only having a larger network is not sufficient to increase the quality and that having higher dimensional features is key here.

4 Non-Power-of-Two Neural Features

In the paper, we propose to use only power of two shaped neural features for convenience when using mipmaps and to be aligned with the texture standards of most game engines. However, our training pipeline allows neural features of any shape. For sake of completeness, we experimented with sets of features with non power of two shape. For instance we have tested the following profiles with 4 set of features and resolution : **BCf-1K** [1024,512,256,128], **P1** [644,636,544,460], **P2** [1084,344,164,72], **P3** [976,596,112,100], **P4** [856,636,352,260]. All these configurations have the same compression rate and are equivalent to our BCf-1K Table 3 shows that when using various profiles with 4 textures having the same total number of pixel than the BCf-1K profile does not drastically change. For that reason, we only focused on power of two shaped features in this work.

5 Detailed dataset performance

Table 4 contains the PSNR, SSIM and FLIP metrics for the entire dataset for both the 16×1 and 64×2 MLPs.

Table 3: Final PSNR of five feature shape profiles on Ukulele2K data from dataset.

	BCf-1K	P1	P2	P3	P4
PSNR (\uparrow)	34.12	33.78	33.7	33.73	34.03

Table 4: Evaluation of the reconstruction metrics (PSNR, SSIM and FLIP) on the whole dataset

		16×1 MLP			64×2 MLP		
		PSNR (\uparrow)	SSIM (\uparrow)	FLIP (\downarrow)	PSNR (\uparrow)	SSIM (\uparrow)	FLIP (\downarrow)
antique katana	BCf-2K++	39.22	0.97	0.043	40.14	0.98	0.04
	BCf-2K	38.33	0.97	0.047	38.84	0.98	0.043
	BCf-1K	33.88	0.94	0.066	34.39	0.95	0.063
	BCf-0.5K	30.58	0.91	0.091	30.6	0.9	0.099
bench vice	BCf-2K++	34.87	0.95	0.068	35.62	0.93	0.067
	BCf-2K	32.51	0.91	0.077	32.93	0.93	0.074
	BCf-1K	29.22	0.84	0.108	29.39	0.83	0.105
	BCf-0.5K	27.28	0.77	0.136	27.11	0.73	0.155
cannon	BCf-2K++	38.83	0.96	0.065	39.46	0.97	0.063
	BCf-2K	36.64	0.94	0.079	37.35	0.95	0.074
	BCf-1K	32.95	0.88	0.112	33.49	0.89	0.113
	BCf-0.5K	30.62	0.84	0.16	30.37	0.8	0.18
cardboard box	BCf-2K++	41.26	0.98	0.044	41.33	0.98	0.039
	BCf-2K	39.42	0.97	0.051	39.65	0.97	0.044
	BCf-1K	35.96	0.94	0.068	36.47	0.94	0.06
	BCf-0.5K	33.05	0.91	0.084	32.86	0.88	0.09
ceramic roof	BCf-2K++	39.8	0.97	0.049	40.56	0.98	0.044
	BCf-2K	37.75	0.96	0.056	38.3	0.97	0.046
	BCf-1K	34.15	0.91	0.085	34.75	0.92	0.075
	BCf-0.5K	30.01	0.82	0.142	30.65	0.84	0.13
Chandelier	BCf-2K++	41.47	0.98	0.033	42.11	0.98	0.032
	BCf-2K	39.88	0.97	0.04	40.51	0.97	0.038
	BCf-1K	36.46	0.95	0.054	36.82	0.95	0.054
	BCf-0.5K	32.67	0.93	0.082	32.17	0.9	0.087
denim fabric	BCf-2K++	37.71	0.97	0.057	33.63	0.95	0.073
	BCf-2K	37.09	0.95	0.061	32.29	0.94	0.075
	BCf-1K	25.23	0.76	0.136	26	0.77	0.118
	BCf-0.5K	22.9	0.63	0.173	23.67	0.68	0.168
garden gloves	BCf-2K++	32.17	0.94	0.102	32.83	0.95	0.096
	BCf-2K	29.83	0.9	0.133	30.37	0.92	0.117
	BCf-1K	26.58	0.82	0.165	26.7	0.8	0.176
	BCf-0.5K	23.69	0.7	0.251	23.75	0.67	0.268
garden gnome	BCf-2K++	41.59	0.98	0.035	42.09	0.98	0.036
	BCf-2K	40.1	0.97	0.046	41.15	0.98	0.039
	BCf-1K	37.49	0.96	0.052	38.16	0.96	0.053
	BCf-0.5K	34.57	0.94	0.074	33.8	0.92	0.083
garden sprinkler	BCf-2K++	36.47	0.96	0.084	37.04	0.97	0.082
	BCf-2K	34.25	0.94	0.102	34.79	0.95	0.102
	BCf-1K	30.37	0.89	0.135	30.6	0.88	0.145
	BCf-0.5K	27.73	0.82	0.199	27.57	0.79	0.236
lubricant spray	BCf-2K++	39.9	0.97	0.033	41.05	0.98	0.031
	BCf-2K	38.84	0.97	0.044	39.79	0.98	0.036
	BCf-1K	34.13	0.93	0.049	35.3	0.95	0.045
	BCf-0.5K	30.67	0.89	0.066	30.8	0.9	0.069
metal tool chest	BCf-2K++	39.14	0.97	0.054	40.46	0.97	0.018
	BCf-2K	36.94	0.96	0.076	39.05	0.96	0.021
	BCf-1K	33.61	0.93	0.086	34.82	0.92	0.031
	BCf-0.5K	30.17	0.89	0.123	31.13	0.85	0.047
MetalPlates	BCf-2K++	40.22	0.97	0.023	39.44	0.98	0.051
	BCf-2K	38.82	0.96	0.021	37.67	0.97	0.06
	BCf-1K	34.37	0.91	0.032	33.98	0.93	0.083
	BCf-0.5K	30.62	0.84	0.052	30.23	0.88	0.132
painted concrete	BCf-2K++	36.6	0.96	0.051	36.88	0.96	0.048
	BCf-2K	34.41	0.93	0.06	34.64	0.94	0.065
	BCf-1K	29.86	0.85	0.086	30.13	0.85	0.085
	BCf-0.5K	27.59	0.76	0.123	27.73	0.77	0.12
PavingStones	BCf-2K++	31.92	0.95	0.121	32.3	0.95	0.096
	BCf-2K	29.52	0.92	0.106	29.78	0.92	0.106
	BCf-1K	25.81	0.76	0.166	26	0.77	0.159
	BCf-0.5K	23.39	0.59	0.239	23.53	0.61	0.231
Terrazzo	BCf-2K++	40.5	0.98	0.04	40.42	0.99	0.047
	BCf-2K	39.26	0.98	0.048	40.04	0.98	0.042
	BCf-1K	35.63	0.95	0.069	36.1	0.95	0.063
	BCf-0.5K	29.16	0.86	0.142	30.16	0.88	0.119
treasure chest	BCf-2K++	36.06	0.95	0.072	36.41	0.96	0.069
	BCf-2K	33.9	0.92	0.087	34.29	0.94	0.086
	BCf-1K	30.24	0.85	0.124	30.56	0.85	0.12
	BCf-0.5K	27.28	0.78	0.164	26.82	0.75	0.192
Ukulele	BCf-2K++	40.35	0.97	0.051	40.88	0.98	0.047
	BCf-2K	38.41	0.95	0.066	39.32	0.97	0.054
	BCf-1K	34.17	0.91	0.09	34.98	0.91	0.083
	BCf-0.5K	31.41	0.87	0.116	31.7	0.85	0.126
Wood	BCf-2K++	31.29	0.94	0.112	31.77	0.95	0.099
	BCf-2K	28.53	0.9	0.123	28.79	0.91	0.123
	BCf-1K	24.63	0.68	0.208	25.1	0.71	0.2
	BCf-0.5K	22	0.5	0.291	22.34	0.53	0.281

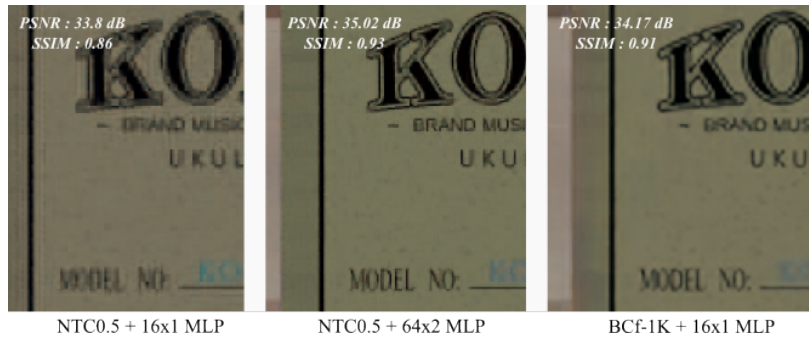


Figure 4: Pairing NTC with a small network diminishes its performance. The high dimensional of the input (81 in this case) makes the small MLP with a hidden layer of size 16 less efficient.

References

- [VSW*23] VAIDYANATHAN K., SALVI M., WRONSKI B., AKENINE-MÖLLER T., EBELIN P., LEFOHN A.: Random-Access Neural Compression of Material Textures. In *Proceedings of SIGGRAPH (2023)*.