

000 QUANTDEMOIRE: QUANTIZATION WITH OUTLIER 001 AWARE FOR IMAGE DEMOIRÉING 002

003 *Supplementary Material*

004
 005
 006
 007 **Anonymous authors**
 008 Paper under double-blind review
 009
 010
 011

012 1 MORE ABLATION

013
 014 We ablate the effect of the sampling rate on both the smooth (γ_1) and bound (γ_2) stages. As shown in
 015 Tab. 2, $\gamma_1/\gamma_2 = 10^{-3}$ consistently achieves the best performance. Larger rates introduce excessive
 016 outliers, while smaller rates lead to biased estimation. These results confirm that a moderate sampling
 017 rate provides the most stable and accurate range estimation, aligning with our design motivation.
 018

019 Table 1: Ablation on sampling rate in activation quantizer. Evaluation is conducted on UDHM, 4bit.

(a) Smooth sampling rate (γ_1).				(b) Bound sampling rate (γ_2).			
Rate	10^{-2}	10^{-3}	10^{-4}	Rate	10^{-2}	10^{-3}	10^{-4}
PSNR \uparrow	19.92	20.92	20.80	PSNR \uparrow	20.88	20.92	20.89
SSIM \uparrow	0.6941	0.7570	0.7505	SSIM \uparrow	0.7558	0.7570	0.7474
LPIPS \downarrow	0.4141	0.3171	0.3221	LPIPS \downarrow	0.3183	0.3171	0.3388

028 2 IMPLEMENTATION DETAILS OF COMPARISON METHODS

029
 030 **MinMax.** We apply per-channel quantization to the weights and per-tensor quantization to the
 031 activations. During the calibration stage, we directly use the minimum and maximum activation
 032 values of the activation tensor as the quantization boundaries.

033 **Percentile.** We also use per-channel quantization to the weights and per-tensor quantization to the
 034 activations. For both weights and activations, the 99.9%-th and 0.1%-th percentiles of all values are
 035 used as the quantization boundaries.

036 **2DQuant.** We first employ the DOBI (Liu et al., 2024) method to search for quantizer parameters.
 037 Then we train the model using the quantization parameters obtained from DQC (Liu et al., 2024),
 038 with a training configuration of 100 epochs, a batch size of 4, and a learning rate of 10^{-2} . All other
 039 training settings remain consistent with those used in our proposed method.

040 **SVDQuant.** We first apply SmoothQuant (Xiao et al., 2023) to mitigate outliers in the activations.
 041 Next, we perform an SVD decomposition on the weights and compute both the low-rank branch and
 042 the residual branch. To ensure fairness in our experiments and to prevent the low-rank branch from
 043 introducing an excessive number of additional parameters, we set the rank to 2.

045 2.1 VARIANTS OF SVDQUANT

046
 047 We observed that SVDQuant suffers from a substantial performance drop when quantized to extremely
 048 low bit-widths (3 and 4 bits). To investigate this issue, we conducted further experiments with higher
 049 ranks. The results show that when the rank is increased to 8, SVDQuant achieves a significant
 050 performance improvement. This indicates that at lower ranks, SVDQuant fails to effectively capture
 051 outlier weights. However, when the rank is set to 8, although the performance improves, the relatively
 052 small parameter size of the convolutional layers in ESDNet leads SVDQuant to introduce an excessive
 053 number of additional parameters. Consequently, SVDQuant is not well-suited for convolutional
 layers with a small number of weight parameters.

Table 2: Results of SVDQuant (rank = 8). Evaluation is conducted on UDHM, 3 and 4bit.

	Bit	W4A4	W3A3
PSNR ↑		18.6998	17.2048
SSIM ↑		0.7428	0.6707
LPIPS ↓		0.3615	0.5033

Table 3: Compression ratios of Params and Ops at 8/6/4/3-bit. Ops are measured with an input size of $3 \times 224 \times 224$. Our QuantDmoire maintains efficiency and performance.

Method	Bit (w/a)	Params (M) (↓Ratio)	Ops (G) (↓Ratio)	PSNR	UHDM SSIM	LPIPS
ESDNet (Yu et al., 2022)	32/32	5.93 (↓0%)	13.52 (↓0%)	22.12	0.7956	0.2551
MinMax (Jacob et al., 2018)	8/8	1.49 (↓74.90%)	3.38 (↓75.00%)	21.50	0.7727	0.2596
Percentile (Li et al., 2019)	8/8	1.49 (↓74.90%)	3.38 (↓75.00%)	19.38	0.7744	0.2784
2DQuant (Liu et al., 2024)	8/8	1.49 (↓74.90%)	3.38 (↓75.00%)	21.20	0.7827	0.2749
SVDQuant (Li et al., 2024)	8/8	1.82 (↓69.29%)	5.44 (↓59.75%)	21.80	0.7907	0.2580
QuantDmoire (ours)	8/8	1.53 (↓74.25%)	3.47 (↓74.31%)	22.00	0.7932	0.2555
MinMax (Jacob et al., 2018)	6/6	1.12 (↓81.14%)	2.54 (↓81.25%)	20.48	0.7648	0.2828
Percentile (Li et al., 2019)	6/6	1.12 (↓81.14%)	2.54 (↓81.25%)	18.44	0.7562	0.2957
2DQuant (Liu et al., 2024)	6/6	1.12 (↓81.14%)	2.54 (↓81.25%)	20.02	0.7595	0.2893
SVDQuant (Li et al., 2024)	6/6	1.45 (↓75.53%)	4.60 (↓65.97%)	20.86	0.7602	0.3015
QuantDmoire (ours)	6/6	1.16 (↓80.43%)	2.64 (↓80.49%)	21.61	0.7874	0.2572
MinMax (Jacob et al., 2018)	4/4	0.75 (↓87.38%)	1.69 (↓87.50%)	16.51	0.5255	0.6786
Percentile (Li et al., 2019)	4/4	0.75 (↓87.38%)	1.69 (↓87.50%)	16.85	0.6701	0.4639
2DQuant (Liu et al., 2024)	4/4	0.75 (↓87.38%)	1.69 (↓87.50%)	17.07	0.6288	0.5117
SVDQuant (Li et al., 2024)	4/4	1.08 (↓81.78%)	3.76 (↓72.19%)	14.68	0.5762	0.6559
QuantDmoire (ours)	4/4	0.79 (↓86.61%)	1.80 (↓86.68%)	21.08	0.7626	0.3068
MinMax (Jacob et al., 2018)	3/3	0.56 (↓90.51%)	1.27 (↓90.63%)	12.53	0.3802	0.8630
Percentile (Li et al., 2019)	3/3	0.56 (↓90.51%)	1.27 (↓90.63%)	14.79	0.5206	0.6869
2DQuant (Liu et al., 2024)	3/3	0.56 (↓90.51%)	1.27 (↓90.63%)	11.20	0.2465	0.8460
SVDQuant (Li et al., 2024)	3/3	0.90 (↓84.89%)	3.34 (↓75.30%)	14.83	0.4547	0.7289
QuantDmoire (ours)	3/3	0.61 (↓89.69%)	1.38 (↓89.78%)	19.12	0.6839	0.4567

3 MORE COMPRESSION RATIO

A more comprehensive comparison of compression ratios is presented in Tab. 3. Our method achieves performance close to that of the full-precision model, while maintaining a high compression ratio.

4 MORE MATHEMATICAL PROOF

A more comprehensive comparison of compression ratios is presented in Tab. 3. Our method achieves performance close to that of the full-precision model, while maintaining a high compression ratio.

5 MORE DISTRIBUTION VISUALIZATIONS

Additional distributions of weights and activations are presented in Fig. 3 and Fig. 5. It can be observed that the activations in most convolutional layers approximately follow either an exponential or a Gaussian distribution, while the weights in the majority of convolutional layers exhibit an approximately Gaussian distribution.

6 MORE QUALITATIVE RESULTS

Additional visual comparisons are presented in Fig. 1 and 2. Our method demonstrates clear advantages under the 3-, 4-, and 6-bit settings, as well as across all datasets.

REFERENCES

Benoit Jacob, Skirmantas Kligys, Bo Chen, Menglong Zhu, Matthew Tang, Andrew Howard, Hartwig Adam, and Dmitry Kalenichenko. Quantization and training of neural networks for efficient integer-arithmetic-only inference. In *CVPR*, 2018.

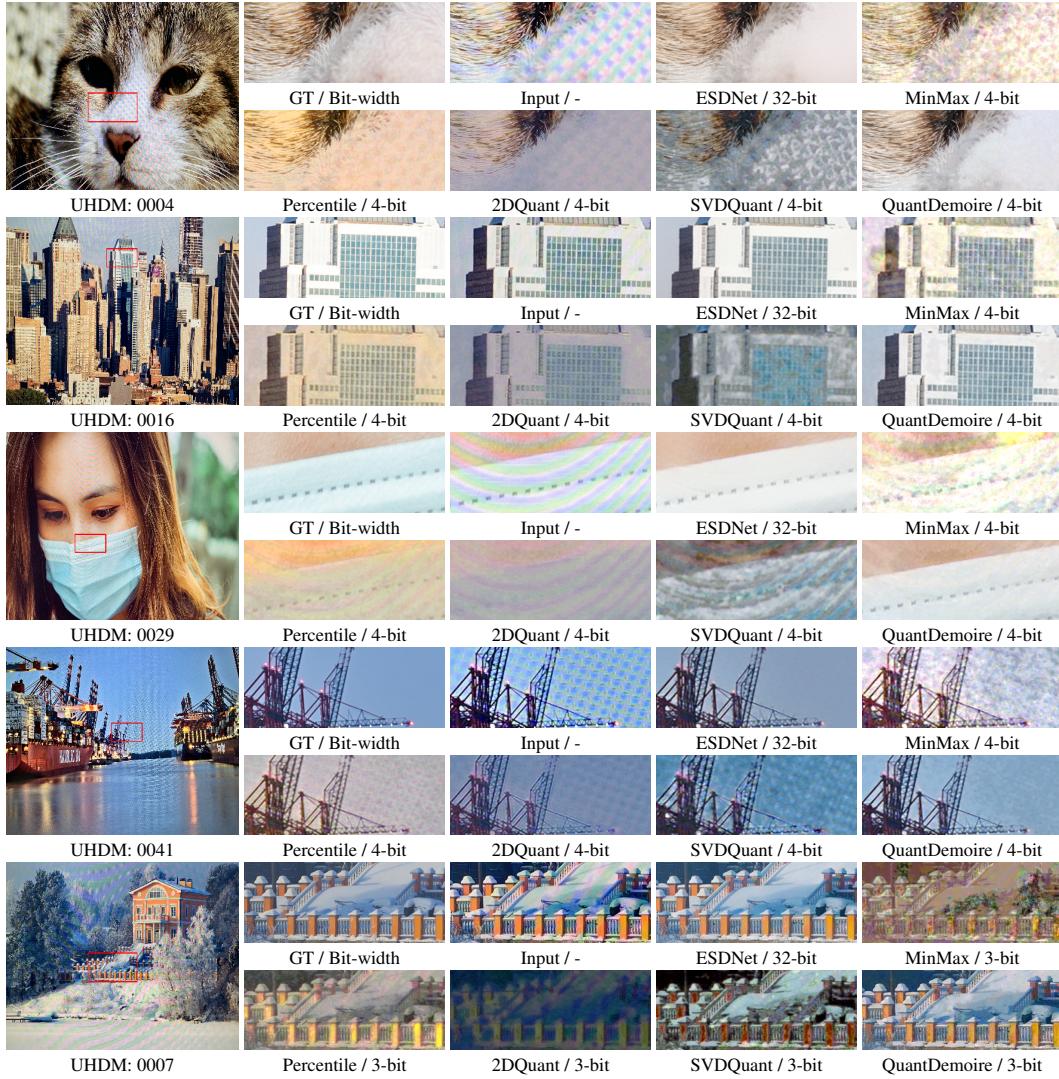


Figure 1: Visual comparison on the UHDM (Yu et al., 2022) dataset.

Muyang Li, Yujun Lin, Zhekai Zhang, Tianle Cai, Xiuyu Li, Junxian Guo, Enze Xie, Chenlin Meng, Jun-Yan Zhu, and Song Han. Svdquant: Absorbing outliers by low-rank components for 4-bit diffusion models. *arXiv preprint arXiv:2411.05007*, 2024.

Rundong Li, Yan Wang, Feng Liang, Hongwei Qin, Junjie Yan, and Rui Fan. Fully quantized network for object detection. In *CVPR*, 2019.

Kai Liu, Haotong Qin, Yong Guo, Xin Yuan, Linghe Kong, Guihai Chen, and Yulun Zhang. 2dquant: Low-bit post-training quantization for image super-resolution. In *NeurIPS*, 2024.

Guangxuan Xiao, Ji Lin, Mickael Seznec, Hao Wu, Julien Demouth, and Song Han. Smoothquant: Accurate and efficient post-training quantization for large language models. In *ICML*, 2023.

Xin Yu, Peng Dai, Wenbo Li, Lan Ma, Jiajun Shen, Jia Li, and Xiaojuan Qi. Towards efficient and scale-robust ultra-high-definition image demoiréing. In *ECCV*, 2022.

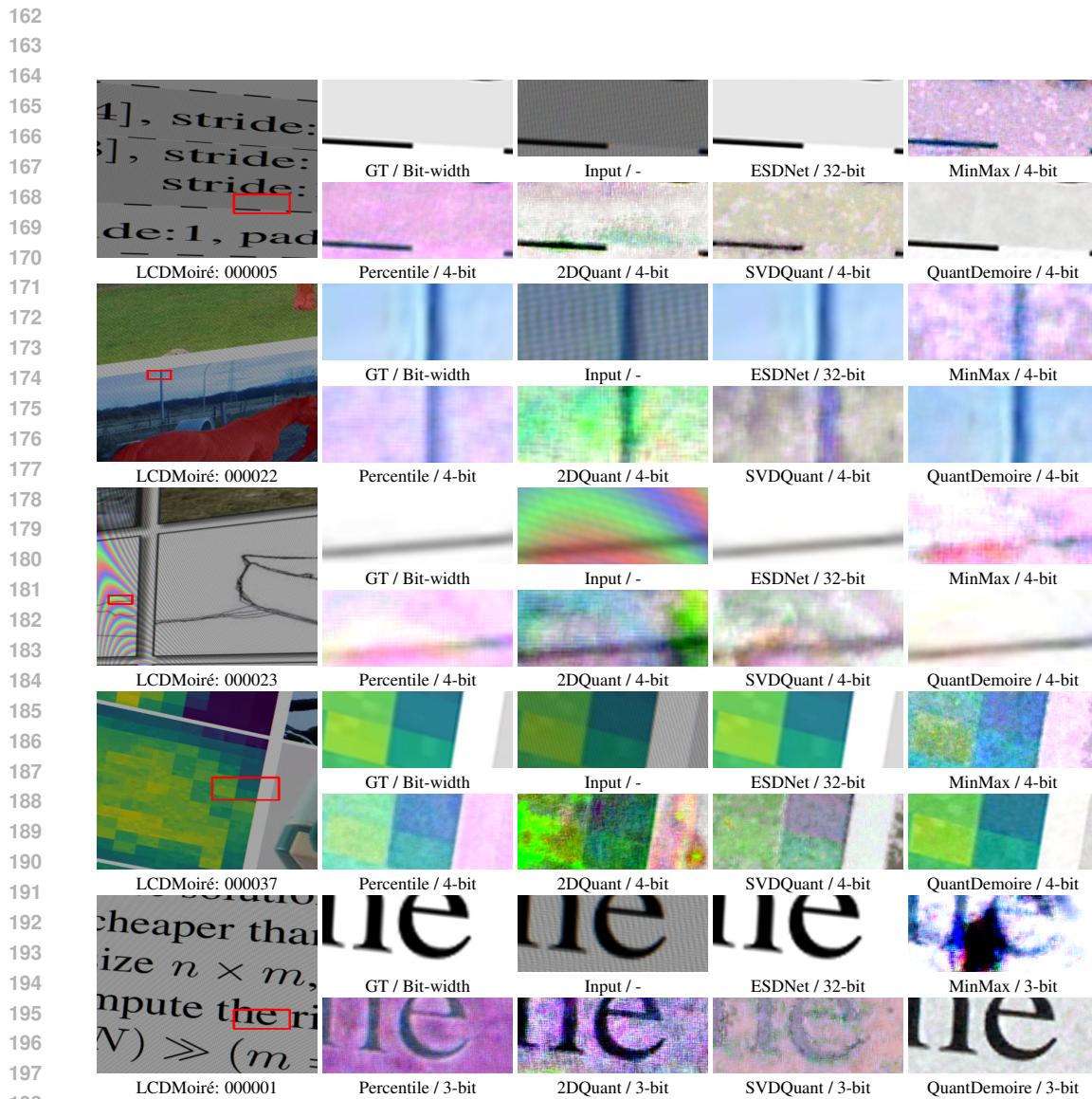


Figure 2: Visual comparison on the LCDMoiré (Yuan et al., 2019) dataset.

Shanxin Yuan, Radu Timofte, Gregory Slabaugh, Aleš Leonardis, Bolun Zheng, Xin Ye, Xiang Tian, Yaowu Chen, Xi Cheng, Zhenyong Fu, et al. Aim 2019 challenge on image demoiréing: Methods and results. In *ICCVW*, 2019.

216
217
218
219
220
221
222
223

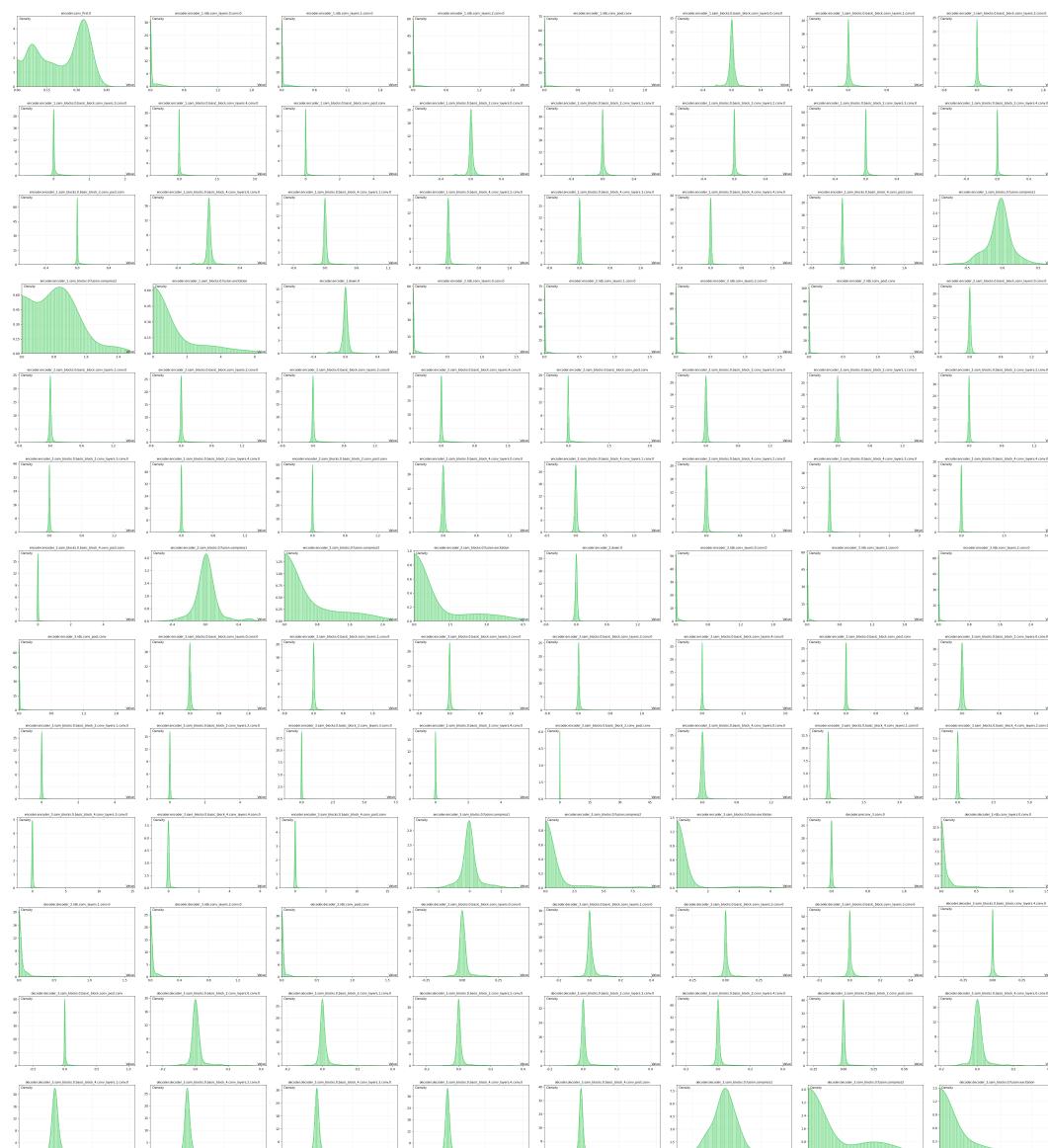
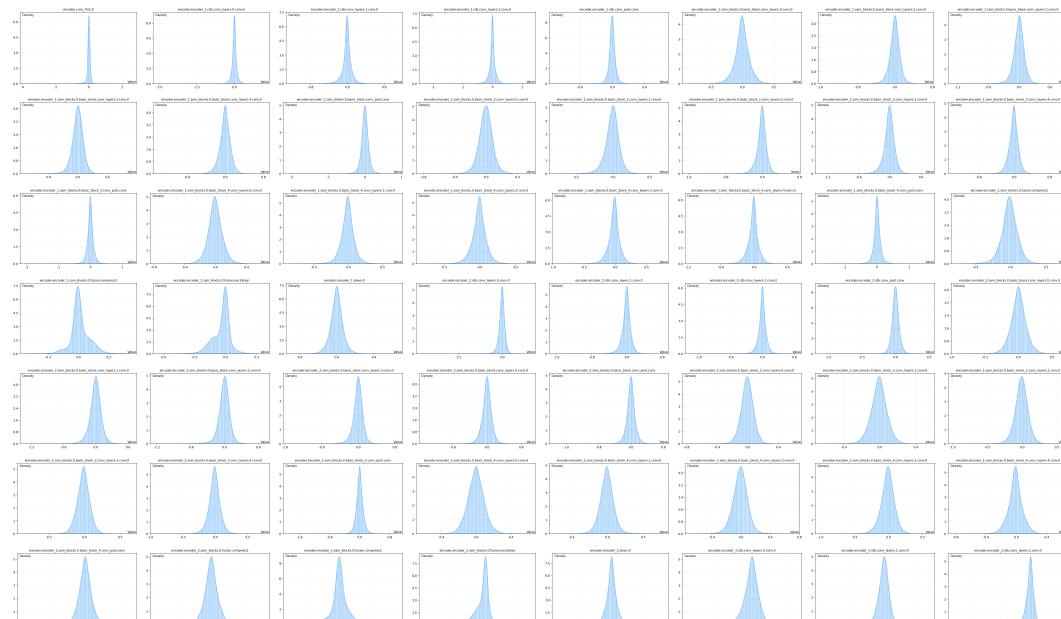
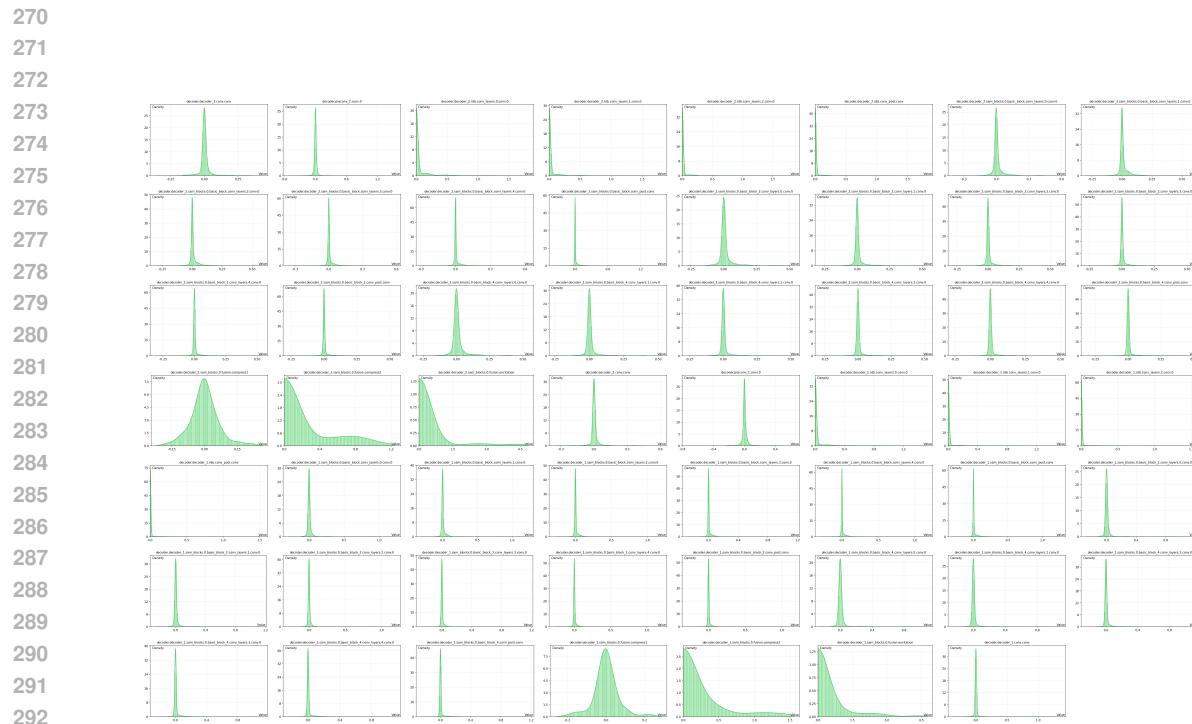


Figure 3: More distribution of activation (Part1).

224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260

261
262
263
264
265
266
267
268
269



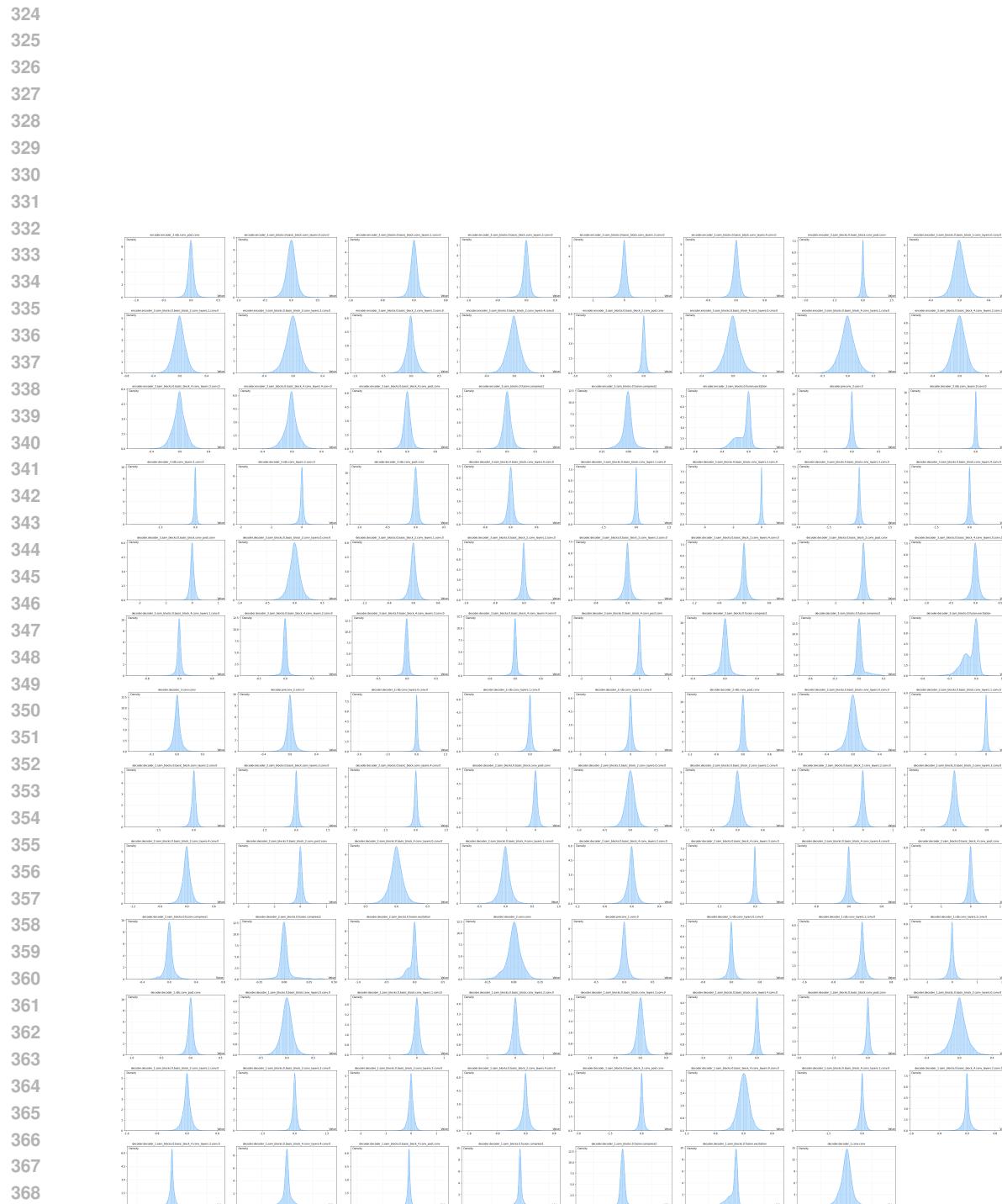


Figure 6: More distribution of weight (Part2).

370
371
372
373
374
375
376
377