
Supplementary Material for MoRIC

Anonymous Author(s)

Affiliation

Address

email

Contents

- 1. Open resources
- 2. Dataset for region-based coding
- 3. Additional experimental results
- 4. Instructions for baseline and MoRIC implementation
- 5. Instructions for adaptive chain coding
- 6. Paper Checklist

1 Open resources

(a) Datasets. We release region-annotated datasets for region-based coding, including DAVIS (18 images), Kodak (24 images), and CLIC2020 (41 images). Additionally, we introduce a new fixed-background dataset featuring a cartoon toy scenario (see Section 2) as an application example.

(b) Adaptive chain coding. We open-source both lossy and lossless variants of C* coding to facilitate future research on the region-based codec, with detailed instructions in Section. 5

(c) Code and checkpoints. Code and checkpoints are provided. (See instructions in Section 4.4.)

2 Dataset for region-based coding

For the DAVIS dataset (18 images), we directly use the contours from the original object masks, all included in the submitted materials.

For the Kodak and CLIC2020 (professional validation) datasets, we use SAM2[8] to segment and extract object contours, yielding 24 samples for Kodak and 41 samples for CLIC2020, each with corresponding contour masks (see submitted materials for details). Some examples of our region-based dataset are visualized in Fig. 1.

Notably, the current region selection is based on areas of contrast, aiming to explore the potential of region-based compression. These region masks are not optimized and can be further refined for enhanced performance.

For practical applications, we introduce a fixed-background dataset featuring a cartoon toy scenario¹. The background is fixed through environmental control and post-processing when needed. Currently,

¹This custom dataset is constructed using real-world objects to evaluate our method. It includes several custom-designed, 3D-printed toy figurines, as well as a commonly available object (a turtle-shaped figure from Kinder Surprise products). The 3D-printed models are independently created and loosely inspired by everyday toy designs. All figures are used solely for non-commercial, academic research. Any associated trademarks or design rights remain the property of their respective owners.

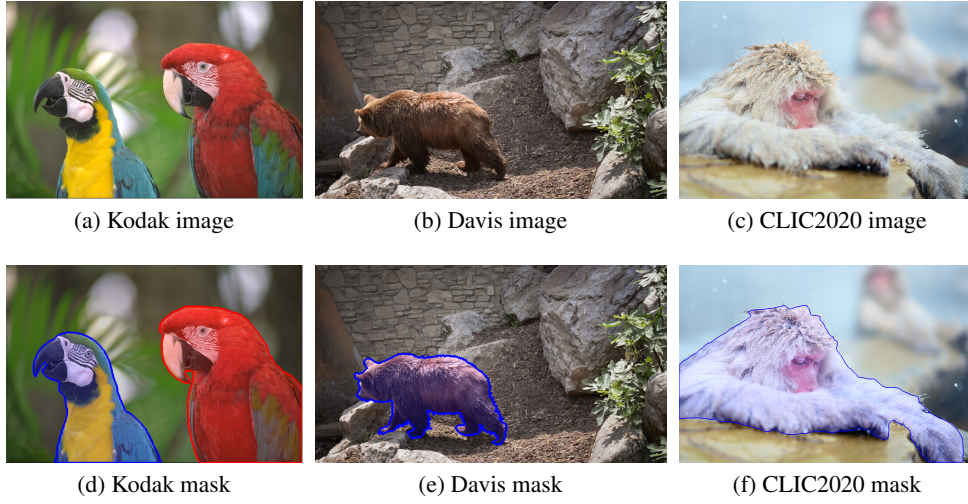


Figure 1: Examples of the region-based datasets (Kodak, Davis, and CLIC2020).

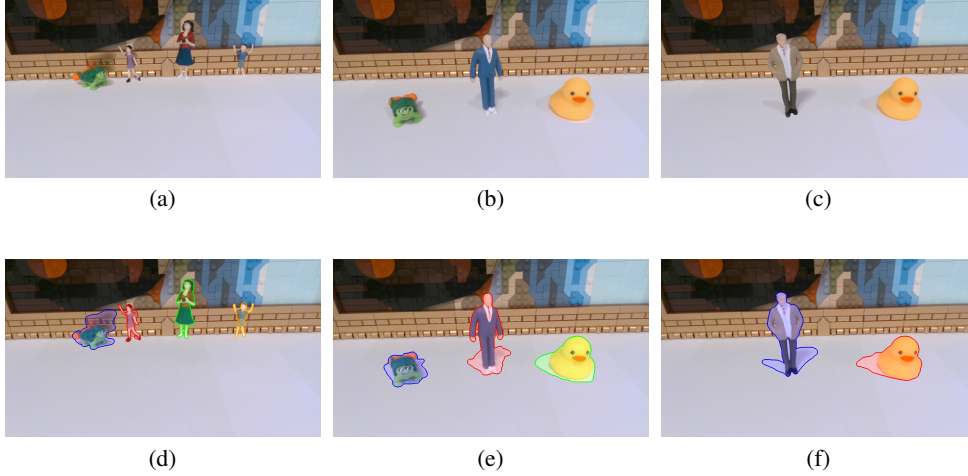


Figure 2: Examples of the fixed-background toy dataset with their corresponding region masks.

it includes 8 images at a resolution of 1920×1080 , with two examples shown in Fig. 2. In the future, we plan to expand this dataset with additional fixed-background scenarios, such as industrial assembly lines, surveillance, and game-generated environments, to support broader real-world use cases. Related experiments are presented in Section 3.2.

Representativeness: this small dataset is constructed for controlled evaluation of our proposed method, focusing on background-fixed scenario. It includes several 3D-printed or toy-like figures to emulate distinct visual instances. While not intended to represent the full diversity of real-world object distributions, the dataset supports consistent and repeatable comparison across controlled settings (fixed background). We do not claim universal generalization, and acknowledge the limitations of toy-based and synthetic objects in capturing real-world complexity. Our goal is to demonstrate the potential application of precise region-based coding schemes.

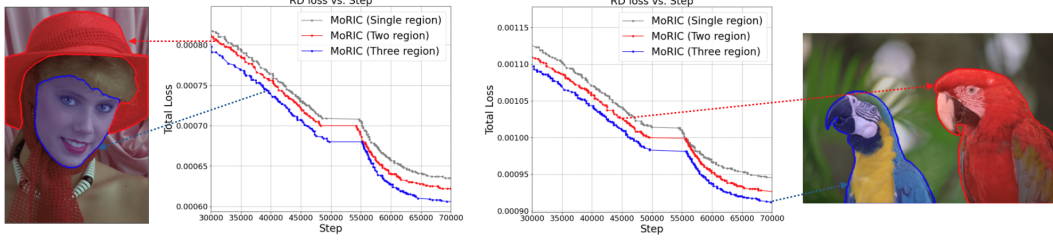


Figure 3: Training convergence across different region configurations, where MoRIC with three regions converges with least training steps.

Table 1: Performance of MoRIC for standard full image compression on CLIC2020 dataset.

BD-rate [vs. VTM 19.1] -2.59%		BD-rate [vs. VTM 19.1] -7.43%	
Rate [bits per pixel]	PSNR [dB]	Rate [bits per pixel]	PSNR [dB]
0.0926	30.541	0.0888	30.632
0.1610	32.502	0.1612	32.648
0.2408	34.079	0.2455	34.260
0.3517	35.716	0.3593	35.931
0.4994	37.336	0.5140	37.544
0.7841	39.445	0.8006	39.683
(a) C3		(b) MoRIC	

3 Additional experimental results

3.1 Analysis of training convergence

Similar to prior work on input space partitioning [9, 7, 6, 1], MoRIC provides potential of accelerating the overfitted encoding by specializing distinct INRs to local regions. As shown in Fig. 3, MoRIC with a three-region setting converges faster than the two- and single-region configurations, reaching similar RD performance up to 3000 steps earlier. While this demonstrates theoretical efficiency, actual encoding latency still depends on deployment factors such as memory and hardware optimizations.

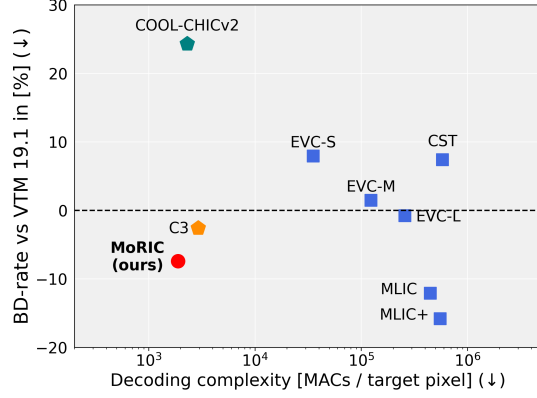
3.2 Experiments on CLIC2020 dataset

We further evaluate MoRIC on the another standard compression dataset (CLIC2020) using a fixed two-region configuration. As shown in Table 1 and Fig. 4, MoRIC achieves state-of-the-art performance among overfitted codecs, with a BD-rate reduction of -7.43% over VTM-19.1 and a significant gain of -4.54% compared to C3. These results validate the effectiveness of MoRIC’s region-based compression strategy, which achieves leading performance under low-complexity constraints (2000 MACs/pixel). Due to computational limitations, we currently did not explore alternative region configurations or architectures for CLIC2020 dataset, where further improvements are expected.

3.3 Experiments on fixed-background dataset

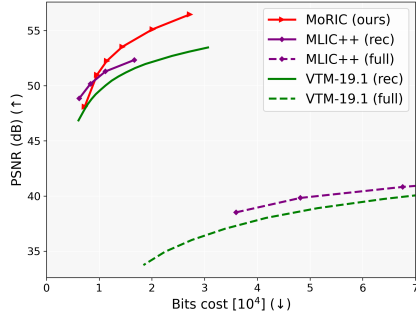
To demonstrate practical applicability, we evaluate MoRIC on our fixed-background dataset (Fig. 2) using VTM 19.1 and MLIC++ as baselines. In our experiments, MoRIC encodes only the object regions, reconstructing the rest using a fixed background. For baselines, we adopt two strategies: (1) compress each sample independently to capture all detailed changes, as in VTM-19.1 (full) and MLIC++ (full) schemes. (2) compress only the smallest bounding box covering the object regions while reusing the fixed background elsewhere, as in VTM-19.1 (rec) and MLIC++ (rec) schemes.

As shown in Fig.5a, MoRIC significantly outperforms traditional full-image coding methods such as MLIC++ (full) and VTM (full), highlighting the benefits of leveraging fixed backgrounds information through precise region-wise compression. Moreover, MoRIC also surpasses MLIC++ (rec) and VTM

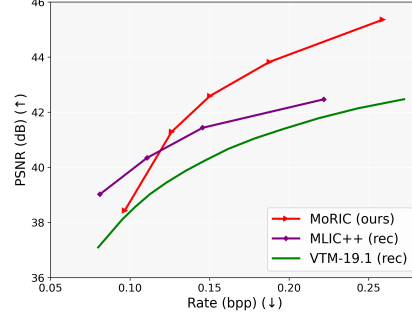


(a)

Figure 4: RD performance vs. decoding complexity on CLIC2020 dataset.



(a)



(b)

Figure 5: Experimental results on the fixed-background toy dataset: (a) RD performance for full-image reconstruction. (b) RD performance for single-object compression using a fixed background, where baselines encode only the target bounding box.

64 (rec), demonstrating that its accurate contour-based coding outperforms simple rectangular-region
 65 strategies (see Figs.5b).

66 4 Instructions for baseline

67 This section provides detailed instructions for implementing main baselines and our MoRIC.

68 4.1 Neural codec implementations

69 For autoencoder-based neural codecs, EVC [10] and MLIC++ [3], we use their official implemen-
 70 tations and report the results as published in their respective papers. For overfitted codecs, C3 [4]
 71 results are taken directly from the original paper, while COOL-CHICv4 [5] results are obtained from
 72 its official code or reported results.

73 4.2 VTM implementations

74 The implementations of VTM are based on CompressAI [2], with detailed commands given as:

VTM-19.1 Code Example

```

## Convert input image to yuv 444
bitdepth = 10
arr = np.asarray(self._load_img(in_filepath))
fd, yuv_path = mkstemp(suffix=".yuv")
out_filepath = os.path.splitext(yuv_path)[0] + ".bin"
rgb = torch.from_numpy(arr.copy()).float() / (2**8 - 1)
arr = np.clip(rgb2ycbcr(rgb).numpy(), 0, 1)
arr = (arr * (2**bitdepth - 1)).astype(np.uint16)

## Encode
cmd = [self.encoder_path, "-i", yuv_path,
      "-c", cfg_path, "-q", quality,
      "-o", output_path, "-b", out_filepath,
      "-wdt", width, "-hgt", height,
      "-fr", "1", "-f", "1",
      "--InputChromaFormat=444",
      "--InputBitDepth=10",
      "--ConformanceWindowMode=1"]

## Decode
cmd = [self.decoder_path, "-b", out_filepath,
      "-o", yuv_path, "-d", 10]

## Convert YCbCr content to rgb
arr = ycbcr2rgb(torch.from_numpy(arr.copy())).numpy()
rec_arr = ycbcr2rgb(torch.from_numpy(rec_arr.copy())).numpy()

```

75

Table 2: Raw datapoints of MoRIC for single object compression task.

Rate [bits per pixel]	PSNR [dB]	Rate [bits per pixel]	PSNR [dB]
0.183	26.741	0.215	24.493
0.267	28.724	0.375	28.585
0.364	30.356	0.616	32.301
0.503	32.124	0.803	34.330
0.702	34.057	1.060	36.432
0.960	36.090	1.282	38.006
1.191	37.611	1.487	39.234
1.385	38.770		

(a) Kodak dataset

(b) Davis dataset

Table 3: Raw datapoints of MoRIC for standard full image compression task.

Rate [bits per pixel]	PSNR [dB]	Rate [bits per pixel]	PSNR [dB]
0.151	29.103	0.169	28.834
0.232	30.723	0.257	30.503
0.354	32.608	0.390	32.428
0.524	34.618	0.577	34.503
0.734	36.679	0.809	36.629
0.922	38.171	1.010	38.228
1.088	39.351	1.196	39.511
1.421	41.293	1.559	41.793

(a) Kodak dataset

(b) Davis dataset

76 4.3 Detailed datapoints of MoRIC

77 The detailed datapoints of MoRIC is provided in Table. 2 and 3.

78 4.4 MoRIC implementations

79 With the provided MoRIC code, we provide the commands for the implementations:

MoRIC Code Example

```
## Install the requirements:  
See the requirements.txt  
## If use lossy C* algorithm for region, run:  
python train.py --context_arm 8 --dim_arm_mod 8 \\  
--lambda_rate_list 1e-3 --sythesis_features 5 --if_lossy True  
  
## If use lossless C* algorithm for region, run:  
python train.py --context_arm 8 --dim_arm_mod 8 \\  
--lambda_rate_list 1e-3 --sythesis_features 5
```

80

81 5 Instructions for adaptive chain coding

82 C* coding is an adaptive chain coding approach for lossy contour compression tailored for region-
83 based overfitted codecs. In addition to this, we also open-source some alternative lossless chain-coding
84 methods to facilitate future research over this region-based coding method.

85 5.1 C* coding Example

86 Given the function in our submitted codes, C* coding can be run as:

C* coding

```
## If use lossless C* algorithm for region, run:  
eval_border_rate_num = get_border_bits_c_star(mask_path,it,  
T=5,thread=10,rate=0.3)  
## If use lossy C* algorithm for region, run:  
total_bits,_,_,_ = get_lossy_border_bits(mask_path,it,  
T,phi_t=np.pi/8,  
M=3,iteration)
```

87

88 5.2 Vertex chain coding

89 With the functions provided in the submitted files, we provide the instructions for the VCC method:

VCC Example

```
mask = cv2.imread(mask_path, cv2.IMREAD_GRAYSCALE)  
contours, _ = cv2.findContours(mask, cv2.RETR_EXTERNAL,  
cv2.CHAIN_APPROX_NONE)  
contour_reordered = contours[0]  
outer_vertices = get_outer_vertices(mask, contour_reordered)  
vcc_result = compute_vcc(outer_vertices, mask)
```

90

91 5.3 3OT

92 With the functions provided in the submitted files, we provide the instructions for the 3OT method:

3OT Example

```
mask = cv2.imread(mask_path, cv2.IMREAD_GRAYSCALE)
contours, _ = cv2.findContours(mask, cv2.RETR_EXTERNAL,
                               cv2.CHAIN_APPROX_NONE)

contour_reordered = contours[0]
outer_vertices = get_outer_vertices(mask, contour_reordered)
three_ot_result = compute_3ot(outer_vertices)
```

93

94 5.4 NAD

95 With the functions provided in the submitted files, we provide the instructions for the NAD method:

NAD Example

```
mask = cv2.imread(mask_path, cv2.IMREAD_GRAYSCALE)
contours, _ = cv2.findContours(mask, cv2.RETR_EXTERNAL,
                               cv2.CHAIN_APPROX_NONE)

contour_reordered = contours[0]
outer_vertices = get_outer_vertices(mask, contour_reordered)
nad_result = compute_nad(contour_reordered)
```

96

97 References

- 98 [1] Maor Ashkenazi and Eran Treister. Towards croppable implicit neural representations. In *The*
99 *Thirty-eighth Annual Conference on Neural Information Processing Systems*.
- 100 [2] Jean Bégaint, Fabien Racapé, Simon Feltman, and Akshay Pushparaja. Compressai: A Py-
101 Torch library and evaluation platform for end-to-end compression research. *arXiv preprint*
102 *arXiv:2011.03029*, 2020.
- 103 [3] Wei Jiang and Ronggang Wang. MLIC++: Linear complexity multi-reference entropy mod-
104 eling for learned image compression. In *ICML 2023 Workshop Neural Compression: From*
105 *Information Theory to Applications*.
- 106 [4] Hyunjik Kim, Matthias Bauer, Lucas Theis, Jonathan Richard Schwarz, and Emilien Dupont.
107 C3: High-performance and low-complexity neural compression from a single image or video.
108 In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*,
109 pages 9347–9358, 2024.
- 110 [5] Thomas Leguay, Théo Ladune, Pierrick Philippe, and Olivier Deforges. Improved encoding for
111 overfitted video codecs. *arXiv preprint arXiv:2501.16976*, 2025.
- 112 [6] Ruofan Liang, Hongyi Sun, and Nandita Vijaykumar. Coordx: Accelerating implicit neu-
113 ral representation with a split mlp architecture. In *International Conference on Learning*
114 *Representations*.
- 115 [7] Ke Liu, Feng Liu, Haishuai Wang, Ning Ma, Jiajun Bu, and Bo Han. Partition speeds up learning
116 implicit neural representations based on exponential-increase hypothesis. In *Proceedings of the*
117 *IEEE/CVF International Conference on Computer Vision*, pages 5474–5483, 2023.
- 118 [8] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma,
119 Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. SAM 2: Segment
120 anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024.
- 121 [9] Vishwanath Saragadam, Jasper Tan, Guha Balakrishnan, Richard G Baraniuk, and Ashok
122 Veeraraghavan. Miner: Multiscale implicit neural representation. In *European Conference on*
123 *Computer Vision*, pages 318–333. Springer, 2022.

- ¹²⁴ [10] Guo-Hua Wang, Jiahao Li, Bin Li, and Yan Lu. Evc: Towards real-time neural image compres-
¹²⁵ sion with mask decay. *International Conference on Learning Representations*, 2023.

NeurIPS Paper Checklist

1. Claims

Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

Answer: [\[Yes\]](#)

Justification: The main claims in the abstract and introduction are consistent with the core contributions and scope of our paper: proposing a region-based overfitted codec for image compression.

Guidelines:

- The answer NA means that the abstract and introduction do not include the claims made in the paper.
- The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
- The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. Limitations

Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: [\[Yes\]](#)

Justification: We provide a section for limitations with some potential solutions for them.

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. Theory assumptions and proofs

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [NA]

Justification: We study a learning-based neural codec without relying on formal assumptions or theoretical proofs. The conclusions of the paper are based on extensive numerical experiments, which are consistently supported throughout the paper.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. Experimental result reproducibility

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes] ,

Justification: We include an open-resource section to introduce the open-sourced components of our work. The code and model checkpoints are submitted as part of the supplementary materials. A more complete version of the codebase, including detailed documentation and usage instructions, will be released publicly after the review process.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general, releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
 - (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
 - (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
 - (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
 - (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in

some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. Open access to data and code

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We include an open-resource section to introduce the open-sourced components of our work. The code and model checkpoints are submitted as part of the supplementary materials. A more complete version of the codebase, including detailed documentation and usage instructions, will be released publicly after the review process.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so “No” is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. Experimental setting/details

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: We provide all details in the appendix. Additionally, code with instructions are provided in the supplementary materials.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. Experiment statistical significance

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [No]

Justification: Not applicable to our setting, where deterministic coding is used.

Guidelines:

- The answer NA means that the paper does not include experiments.

- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. Experiments compute resources

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: Appendix provides all the platform details, training time, and decoding time.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. Code of ethics

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics <https://neurips.cc/public/EthicsGuidelines>?

Answer: [Yes]

Justification: The research conducted in the paper conform with the NeurIPS Code of Ethics.

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. Broader impacts

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: We provide this in Section.B of our appendix.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. Safeguards

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [No]

Justification: Our work does not involve models or datasets considered high risk for misuse. The models and datasets used are standard compression dataset and do not pose notable security, privacy, or misuse concerns. The small test dataset we introduce is made by ourselves independently, all objects are 3D printed or properly credited.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. Licenses for existing assets

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: All assets used in this work, including datasets, pretrained models, and code libraries, are properly cited in the paper. Their licenses and terms of use are respected.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.

- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, paperswithcode.com/datasets has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. New assets

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [\[Yes\]](#)

Justification: We introduce new assets in the form of model code, pretrained checkpoints, and a small validation set. These are accompanied by clear documentation outlining usage instructions, model structure, and dependencies. The materials are included in the supplementary files, and a more comprehensive open-source release with full documentation will follow after the review process.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. Crowdsourcing and research with human subjects

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [\[NA\]](#)

Justification: Our work does not involve crowdsourcing or experiments with human subjects. Therefore, no participant instructions, screenshots, or compensation details are applicable.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. Institutional review board (IRB) approvals or equivalent for research with human subjects

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [\[NA\]](#)

Justification: This work does not involve human subjects or human-related data collection, and therefore does not require IRB approval or equivalent ethical review.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. Declaration of LLM usage

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: No large language models (LLMs) are used as part of the core methodology or experimental design in this research. Any LLM usage, if any, was limited to minor writing support and did not influence the scientific content or originality of the work.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (<https://neurips.cc/Conferences/2025/LLM>) for what should or should not be described.