Optimizing HPC Performance for CDC-Based Storage and Reconstruction of High-Resolution Imagery

Jooho Kim*, Yifan Yang†, Anish Shakya‡, and Jacob Kelly§
*Institute for a Disaster Resilient Texas (IDRT), Texas A&M University, College Station, TX, USA
†Department of Geography, Texas A&M University, College Station, TX, USA
‡Department of Marine and Coastal Environmental Science, Texas A&M University, Galveston, TX, USA
§Department of Computer Science and Engineering, Texas A&M University, College Station, TX, USA

***Abstract*—High-resolution imagery is increasingly used for community asset management, environmental monitoring, and disaster response, but a single field mission can generate hundreds of gigabytes of data, which increases long-term storage and transfer costs. To reduce this storage burden, this study evaluates an existing conditional diffusion compression (CDC) [1] workflow for high-resolution image storage and reconstruction on GPU supercomputers. We do not propose a new compression model. Instead, we study how CDC behaves as an HPC workload and identify practical operating points that balance compression ratio, GPU memory use, reconstruction speed, image quality, and downstream analysis utility. Using 100 RGB drone images from Galveston, Texas, we test different CDC checkpoints and tiled reconstruction settings on NCSA DeltaAI GH200 [2]. The balanced checkpoint with 256 × 256 tiled reconstruction reduces peak GPU memory from 51.96 GB to 1.57 GB and lowers reconstruction time from 143.17 to 79.34 seconds per image compared with full-image reconstruction. This setting keeps a 79.98× compression ratio while preserving useful image quality and downstream detection and segmentation performance. CDC can therefore support long-term storage of high-resolution imagery when reconstruction is treated as a system optimization problem.**

I. Summary

Drone imagery is valuable after disasters because it preserves roof damage, vehicle access, road conditions, and other fine visual details. Recent work also examines bi-temporal street-view damage assessment and satellite-to-street synthesis for disaster intelligence [3], [4]. Those same details make the data hard to use at scale. A single collection can create hundreds of gigabytes of imagery, and repeated reconstruction can turn a storage problem into a GPU scheduling problem. For SC, the core question is therefore a systems question: how should a data-centric AI workflow choose a compression and reconstruction operating point that fits memory, queue time, storage constraints, and downstream analysis needs?

We treat CDC as an existing reconstruction path, not as an original algorithmic contribution. We chose CDC over classical codecs such as JPEG 2000 [5] and WebP because those codecs are CPU-cheap but plateau at modest ratios, whereas CDC reaches 80–140× above 30 dB PSNR; its GPU cost is precisely the open systems question. The contribution is a measured HPC study of how checkpoint choice and tiled reconstruction change resource behavior on NCSA DeltaAI GH200. The workflow processes 100 RGB drone images from Galveston, Texas. For the main N50 tradeoff matrix, each configuration records wall time, images per hour, peak GPU memory, bits per pixel, compression ratio, PSNR, SSIM [6], LPIPS [7], seam-related artifacts, and visual error maps. YOLOv8s detection [8] and SAM masks [9] are used only as utility checks that an efficient setting preserves the signals emergency managers inspect.

TABLE I
Selected full-resolution DeltaAI operating points on 50 drone images.

Setting	Sec./img	Img/hr	GB	Ratio	F1	Roof IoU
High quality 512	88.51	40.7	2.96	29.73×	0.8805	0.9001
Balanced 256	79.34	45.4	1.57	79.98×	0.8705	0.8989
Max comp. 256	78.91	45.6	1.57	139.89×	0.8679	0.8956

The central result is that full-image diffusion reconstruction is feasible but poorly matched to routine HPC use. With the balanced checkpoint b00064, full-image reconstruction takes 143.17 seconds per image and peaks at 51.96 GB of GPU memory. The same checkpoint with 256×256 tiled reconstruction takes 79.34 seconds per image, reaches 45.4 images per hour on one GH200 GPU, and peaks at 1.57 GB. This is a 44.6% runtime reduction and more than a $30\times$ reduction in peak GPU memory relative to the full-image balanced run. It also keeps a $79.98\times$ compression ratio, 33.14 dB PSNR, 0.8768 SSIM, and 0.001826 LPIPS. The asymmetry is explained by how the diffusion decoder scales: peak memory follows tile area, whereas runtime follows total pixels plus per-tile overhead, so tiling removes almost all memory pressure but only part of the compute. The 512×512 tile remains a quality-safe backup.

The downstream measurements support this operating-point choice. On human vehicle and roof labels, original images reach 0.8829 F1. Balanced 256×256 reconstruction reaches 0.8705 F1, a relative drop of about 1.4%, while maximum compression reaches 0.8679 F1. The stricter mAP@0.5:0.95 metric is more sensitive, which means localization quality

should be reported separately from classification stability. SAM is used similarly as a mask-stability check. All 1,013 vehicle prompts and 1,201 roof prompts return usable masks. For balanced 256×256, vehicle masks reach 0.7346 mean IoU and 0.8380 Dice, while roof masks reach 0.8989 mean IoU and 0.9447 Dice.

The claim is thus not that CDC is a new model, but that CDC becomes practical for full-resolution drone imagery on a GPU supercomputer when treated as a systems operating-point problem that reports storage savings, GPU memory, throughput, scalar quality, visible artifacts, and downstream utility together. Balanced checkpoint b00064 with 256×256 tiling is recommended because it converts a memory-heavy diffusion job into a low-memory, higher-throughput workflow while preserving disaster-relevant utility. The study is limited to one site, 100 RGB images, and a single GPU. Next steps are multi-GPU and multi-node scaling on DeltaAI, comparison against A100/H100 nodes, a cost model that predicts the optimal tile size from a GPU memory budget, extension to multispectral and thermal imagery and additional sites, and integration into the IDRT disaster-response pipeline.

REFERENCES

[1] R. Yang and S. Mandt, "Lossy image compression with conditional diffusion models," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 36, 2023.

[2] National Center for Supercomputing Applications, "DeltaAI user documentation," https://docs.ncsa.illinois.edu/systems/deltaai/, 2024, accessed: 2026-10-01.

[3] Y. Yang, L. Zou, B. Zhou, D. Li, B. Lin, J. Abedin, and M. Yang, "Hyperlocal disaster damage assessment using bi-temporal street-view imagery and pre-trained vision models," *Computers, Environment and Urban Systems*, vol. 121, p. 102335, 2025.

[4] Y. Yang, L. Zou, and W. Jepson, "Satellite-to-street: Synthesizing post-disaster views from satellite imagery via generative vision models," 2026. [Online]. Available: https://arxiv.org/abs/2603.20697

[5] A. Skodras, C. Christopoulos, and T. Ebrahimi, "The JPEG 2000 still image compression standard," *IEEE Signal Processing Magazine*, vol. 18, no. 5, pp. 36–58, 2001.

[6] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.

[7] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 586–595.

[8] G. Jocher, A. Chaurasia, and J. Qiu, "Ultralytics YOLOv8," https://github.com/ultralytics/ultralytics, 2023.

[9] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, P. Dollár, and R. Girshick, "Segment anything," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023, pp. 4015–4026.