Grail-AD: Request-Driven Algorithm Evolution for 2D Graphite Material Packing

Anonymous authors
Paper under double-blind review

Abstract

Growing large single-crystal graphite enables structural-superlubric devices, but converting grown crystals into products requires packing independently rotatable rectangular items into irregular usable regions while maintaining safe distances for laser cutting kerf. The physical crystals are expensive to acquire and their morphology is difficult to control. We release a dataset of large-scale single-crystal graphite flakes and propose Grail-AD, a closed-loop framework that connects a scale-controlled crystal image generator to a population of evolving packing algorithms. The generator combines pre-trained feature alignment, scale control, and representation-based retrieval to supply diverse crystal images. The packing evolution employs an LLM-driven heuristic search with a composite objective combining geometrically valid packing count and a spacing reward — with an image-request mechanism through which algorithms query specific morphologies from the generator; cases that promote progress are retained while unsolvable ones are rejected, forming a task-directed data engine. Retrospective evaluation of 45 runs spanning three search frameworks, five image-supply schemes, and three repeats on a frozen independent 100-image test set shows that conditioned requests yield a higher retrospective maximum than matched Fixed runs in 22 of 27 comparisons. OpenEvolve improves in all nine matched comparisons; Real Request averages 34,638 versus 27,094 for Fixed, a 27.8% gain. MLES Request improves in all three matched comparisons, with a 10.7% mean gain. Because the reported score for each run is the retrospective maximum over its candidate pool, these results characterize candidate-discovery capability under a fixed search budget rather than unbiased deployment performance. This approach may extend to the product design and manufacturing of other two-dimensional crystal materials, including transferred heterostructures.

1 Introduction

The growth of centimeter-scale single-crystal graphite provides the material foundation for manufacturing structural-superlubricity devices (Dienwiebel et al., 2004; Zhang et al., 2022). Cutting rectangular graphite flakes from single crystals for superlubric devices still requires solving a morphology-constrained algorithmic problem: avoiding grain boundaries, cracks, and wrinkles, packing as many specified-size rectangles as possible within irregular usable regions, and maintaining spacing required for laser cutting. Material utilization therefore depends on whether the packing algorithm can adapt to single-crystal graphite images of varying scales and connectivity structures.

Studying this problem requires abundant crystal images, but growth and acquisition costs are high, and scale and morphology are difficult to control on demand, making it very difficult to obtain enough cases experimentally. Collecting this dataset required approximately $50,000 in experimental investment and yielded 133 single-crystal graphite morphology images. We therefore build a conditional generative model on these images, which can then controllably produce large quantities of single-crystal graphite images at different scales and morphologies, thereby supplying the diverse cases needed for algorithm development.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

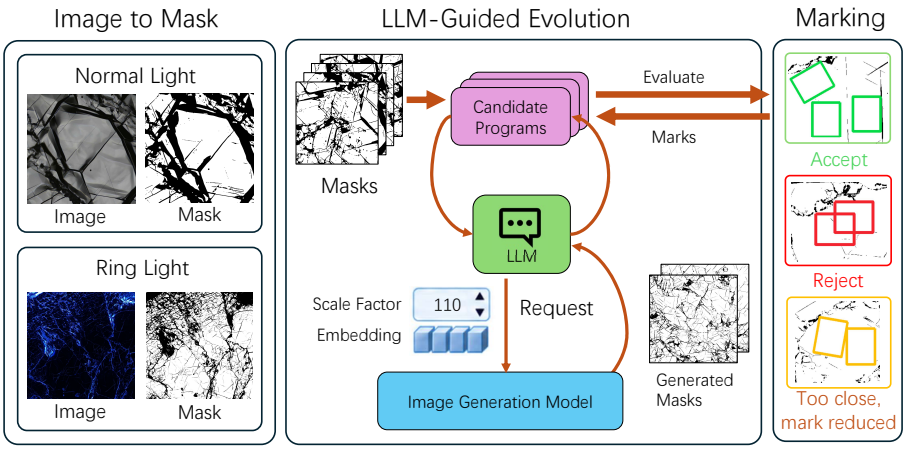


Figure 1: Closed-loop overview of Grail-AD. Images from both acquisition modalities are recognized to obtain usable-region masks; the LLM evolves candidate packing programs based on layout evaluation feedback and requests new cases with specified scale and morphology. Generated images are converted to masks by the recognizer and returned to the evolution loop. The right side shows three types of feedback: layout acceptance, overlap rejection, and spacing reward reduction.

A natural follow-up question is: which images should be generated in the next batch to drive algorithmic progress?

We propose Grail-AD, which embeds image requests into LLM-driven heuristic evolution (Figure 1). The generator, recognizer, and packing program form a closed loop in which image generation serves algorithm evolution, and algorithm progress guides subsequent image requests. On a frozen independent 100-image test set, conditioned requests yield higher retrospective maxima than matched Fixed runs in 22 of 27 comparisons across 45 runs. OpenEvolve improves in all nine matched comparisons, with Real Request averaging 34,638 versus 27,094 for Fixed, a 27.8% gain. MLES Request improves in all three matched comparisons, with a 10.7% mean gain. These retrospectively selected maxima measure candidate-discovery capability under a fixed search budget rather than unbiased deployment performance.

Our contributions are as follows:

- Crystal scan data and packing task. To the best of our knowledge, we release the first scan-image dataset of centimeter-scale single-crystal graphite flakes, constructed with approximately $50,000 in investment. The data includes two acquisition modalities, Normal Light and Ring Light, and provides a processing pipeline for obtaining candidate usable regions from images.
- Scale-controllable crystal image generation model and framework. We build a conditional generative model based on StyleGAN2-ADA with a topological style adapter that encodes scale as FiLM modulation parameters, enabling controlled generation of graphite images at arbitrary scales with morphology controlled through latent variable sampling. Generated images are converted into binary usable-region masks through the recognizer, providing packing instances for downstream algorithm evaluation.
- Program evolution with additional image supply. We compare fixed cases, frozen-pool selection, and deterministic code-conditioned case generation during LLM-guided program search, and measure the candidates discovered under a common 50-round budget.

2 Dataset, Image Processing, and Packing Task

2.1 Materials and Dual-Modality Acquisition

The dataset is derived from microscopy scans of CVD-grown graphite flakes (Zhang et al., 2022; 2025). Structural superlubricity studies have shown that crystal contact orientation and edge conditions significantly affect friction (Dienwiebel et al., 2004; Hu et al., 2024); friction mechanisms at grain-boundary undulations have also been investigated at graphene interfaces (Gao et al., 2021). This work focuses on the irregular usable regions formed by these material structures in the images and the resulting packing requirements.

We construct a graphite scanning dataset covering Normal Light and Ring Light acquisition modalities across different crystal scales and morphologies. The scanning system comprises a three-axis translation stage, a microscopy imaging module, and illumination devices: Normal Light preserves the bright-field appearance, while a custom Ring Light suppresses specular reflection and emphasizes surface boundaries. Each acquisition modality produces raw images that, after modality-specific processing, enter the same generation and packing pipeline. Lamp fixtures and installation examples are given in Appendix A.

2.2 From Local Scans to Task Instances

A single microscope frame covers only a local region of the sample, so we first scan a regular grid with overlapping fields of view, then register and fuse adjacent frames into large-field mosaics. The mosaics preserve structural continuity across fields of view and are used to construct packing instances; local patches extracted from source images are used to train the image generation model. The masks produced by the recognizer from both sources constitute the usable regions actually used by the algorithm.

Source image files and patches are counted separately; adjacent tiles are not counted as independent crystal samples. Acquisition details and data statistics are in Appendix A.

2.3 From Images to Binary Usable Regions

Let $I \in [0,255]^{H\times W\times 3}$ denote the original RGB image and $\widehat{I}$ the generated image. A modality-specific recognizer M_m maps both to a binary mask, where $m \in \{\text{Normal Light}, \text{Ring Light}\}$. Define

$$U_I = \{(x,y) : M_m(I)_{y,x} = 1\}. \tag{1}$$

U_I is the set of candidate usable pixels, not the original image; the packing program reads only this binary region, and everything outside the image is unavailable. The recognizer is selected according to the acquisition modality and applied to both real and generated images after modality-appropriate preprocessing, producing a unified binary mask where 1 indicates usable. Examples of correspondence among original images, training images, and masks are given in Appendix B.

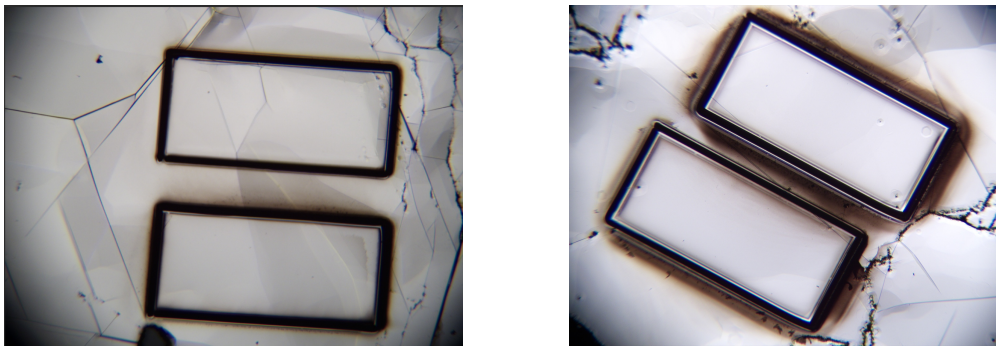
A product pose $q = (x, y, \theta)$ defines an independently rotatable $w \times h$ rectangle $R(q)$, with $\theta \in [0, \pi)$. We adopt $(w, h) = (500, 400)$ pixels as the benchmark specification shared by all algorithms when the task is in $(W, H) = (4096, 4096)$, comparing packing strategies under a fixed product requirement. Rectangle coordinates are consistent with the pixel coordinates of the input mask.

2.4 Geometric Validity and Spacing Reward

Given the program output, verification yields a set of rectangles $S = \{q_1, \ldots, q_N\}$: each rectangle lies entirely within U_I, and no two have positive-area overlap. Define the nearest-neighbor boundary distance

$$d_i = \min_{j\neq i} \operatorname{dist}(R(q_i), R(q_j)), \tag{2}$$

where distance is Euclidean and zero upon contact; when only one rectangle is present, $d_i = +\infty$. In engineering terms, $d_i \geq \delta$ indicates that the recommended cutting spacing is met, and $0 \leq d_i < \delta$ indicates that this spacing is not met.



(a) $d > \delta$ (b) $d < \delta$

Figure 2: Bright-field microscope images of two rectangular product proposals on a graphite flake. The black borders are laser-cutting kerf, and the safe distance retained between the two rectangles corresponds to the spacing constraint δ in Equation 3.

This work adopts a scoring scheme with hard geometric-validity check and soft spacing reward, with $\delta = 20$ task pixels. Consistent with the actual evaluator, define

$$b_i = \begin{cases} 0, & 0 < d_i < \delta, \\ 1, & d_i \geq \delta, \end{cases} \qquad S_{\text{safety}} = \sum_{i=1}^{N} b_i, \qquad F(I, S) = 100N + S_{\text{safety}}. \tag{3}$$

$S_{\text{safety}} \in [0, N]$ is the unnormalized sum of rewards. Each geometrically valid rectangle receives 100 points, plus 1 additional point if the reward condition is met; rectangles with insufficient spacing are still counted in N. Quantity and spacing are compared through a weighted sum rather than strict lexicographic order; for the same quantity, the one with higher spacing reward scores higher. Candidates that go out of bounds, fall into unavailable regions, or overlap are not counted in N.

2.5 Scale Description of Crystal Structure

Scale refers to the characteristic size of candidate regions, not the image magnification factor. Let $C_1, \ldots, C_n$ be the eight-connected regions output by the frozen geometric segmenter, sorted in decreasing order of area. In the generator calibration, they are extracted separately from the non-wrinkle classes of this segmenter and then merged and sorted; the resulting value serves as a structural-scale proxy on the image rather than a count of independent physical grains. Define

$$y(I) = \frac{1}{K} \sum_{k=1}^{K} |C_k|, \qquad \ell(I) = \sqrt{y(I)}, \qquad K = \min(10, n). \tag{4}$$

The ten largest regions are taken to reduce the influence of numerous small connected components on the scale description while avoiding having the scale determined solely by the largest component. The square root of the mean area yields a pixel length, facilitating the setting of scale requests; when fewer than ten components are present, all connected components are used, and zero is taken when no connected components exist. This geometric segmentation and the packing usable-region recognition serve distinct purposes; the specific measurement conventions are given in Appendix C.

3 Grail-AD: Request-Driven Algorithm Evolution

3.1 Controllable Image Generation and Morphology Retrieval

The scale-controllable crystal image generator uses conditional StyleGAN2-ADA as its backbone, injecting scale conditions into early synthesis representations via a topological style

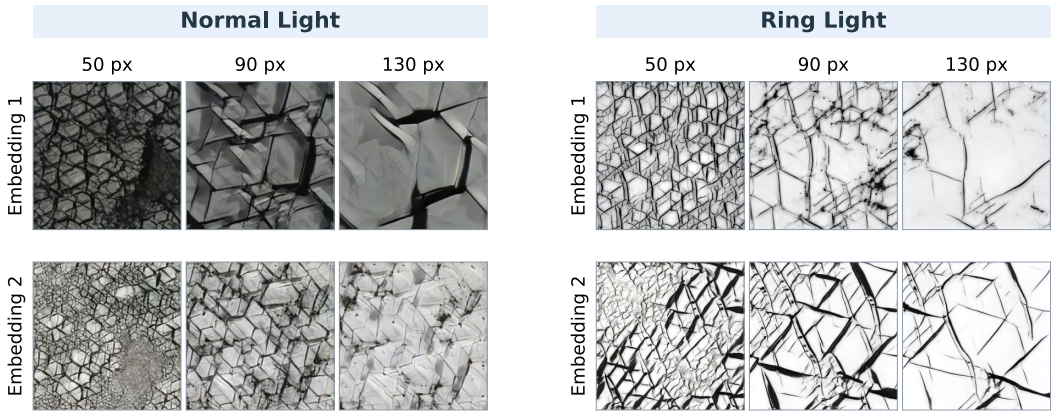


Figure 3: Controllable generation examples under Normal Light and Ring Light. Columns indicate target scale; rows show different sampled representations; generated images are subsequently converted to packing masks by the corresponding recognizer. The full matrix is given in Appendix B.

adapter that encodes the requested scale as FiLM modulation parameters, with latent variables providing morphological variation. The generator outputs images $\hat{I} = G_m(z, s)$, not packing masks directly; these are subsequently converted to usable regions by M_m. Model architecture and training details are given in Appendix B.

3.2 Operational Image Supply During Program Search

The search maintains a population of packing programs $\mathcal{A}_t$. Candidate programs are proposed using the feedback available to each search framework and are evaluated on ten fixed real cases D_0. Depending on the image-supply scheme, two additional cases are also evaluated and recorded.

The request rules use detection outcomes from completed training rounds. Let K be the number of preceding rounds considered. Images newly detected during those rounds define the target scale and, where applicable, the target image embedding; when no such image exists, the ten fixed cases D_0 provide the fallback. Request samples two scale-matched cases from the frozen inventory pool using a seed derived from the repetition and evaluation indices. Embedding Request instead retrieves the two scale-matched inventory cases with the highest embedding similarity to the recently detected images. Real Request uses the same scale and embedding-based reference cases to request two new images from the frozen GraphiteGenerator, then converts its outputs to packing masks. Random Extra selects two inventory cases independently of the detection history and candidate code. Per-image scores and detection outcomes are recorded, while the primary training fitness follows the framework-specific rules in Appendix D. The independent test images are never used to choose or generate training cases.

This closed loop changes the data encountered by the search. The GraphiteGenerator remains fixed and serves as a reusable source of synthetic cases. Detection feedback from preceding rounds determines the scales and image features requested in subsequent rounds.

4 Experiments: From Controllable Generation to Algorithmic Gains

4.1 Scale Response

Across ten request levels in the 50–140 pixel range, 10 images are generated per level. In Figure 4, the center line of each box plot is the per-level median; the MAE is the mean absolute scale error over all 100 images. The measured median increases from 49.01 to

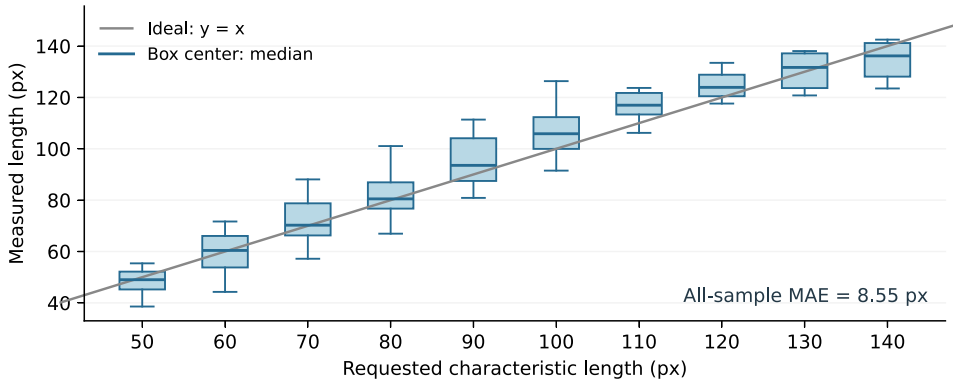


Figure 4: Requested scale versus measured scale. Ten images per level; boxes show the interquartile range with the median as the center line, and whiskers extend to the most extreme observation within 1.5× IQR. The gray line is $y = x$. The MAE is the mean absolute error over all 100 images.

136.18 pixels with an MAE of 8.55 pixels, confirming that scale requests systematically change the size of generated structures.

4.2 Representation Similarity Evaluation

Both models share 20 real reference images and each uses 20 generated images. R–R, R–G, and G–G denote real–real, real–generated, and generated–generated ResNet cosine similarity, respectively. The scale-controllable model shows highly overlapping distributions across all three pairings; the early generator exhibits elevated G–G but depressed R–G. This result supports the balance between visual consistency and generation diversity achieved by the scale-controllable model; complete matrices, violin plots, and summary tables are presented in Appendix C.

4.3 Qwen Main Matrix and Unified Independent Evaluation

We construct $3 \times 5 \times 3 = 45$ runs from three program search frameworks (OpenEvolve, EOH, MLES), five image supply schemes (Fixed, Request, Embedding Request, Real Request, Random Extra), and three repeats. Each run uses the same initial program and a 50-round budget, producing 6,602 traceable candidates in total. All 45 runs completed candidate collection and unified evaluation on the frozen independent test set; pseudocode and training protocols for the five image supply schemes are given in Appendix D.

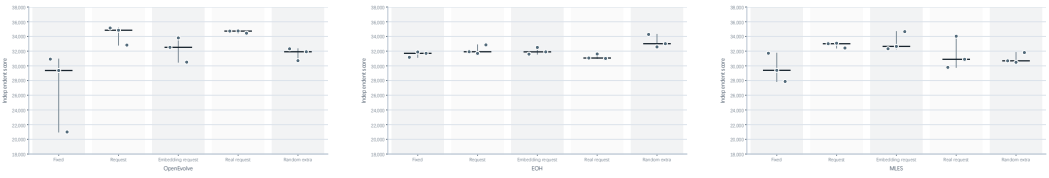
We evaluate every collected candidate on the same frozen 100-image real evaluation set. For each candidate pool $\mathcal{C}$, we plot the cumulative retrospective maximum $B_{100} = \max_{c \in \mathcal{C}} F_{100}(c)$ over search rounds. This staircase describes how the search discovers candidates that also score well on a broader set of real packing cases excluded from program-training feedback. Because the maximizing candidate is identified using this evaluation set, B_{100} is a descriptive candidate-discovery statistic, not a once-only test estimate for a program selected without access to this set.

4.4 High-Scoring Candidates Discovered by Request-Driven Image Supply

The three conditional request schemes yield 27 pairings against the same-framework, same-repeat Fixed baseline, of which 22 achieve a higher 100-image retrospective maximum. Request wins 8 and loses 1; Embedding Request wins all 9; Real Request wins 5 and loses 4. Aggregating across all frameworks and repeats, Request has a mean pairing difference of +3642 and a mean per-pair relative change of +14.1%; Embedding Request has a mean pairing difference of +3048; Real Request has a mean pairing difference of +3033. These overall means are positive, but Table 1 also preserves cross-framework variation: all three

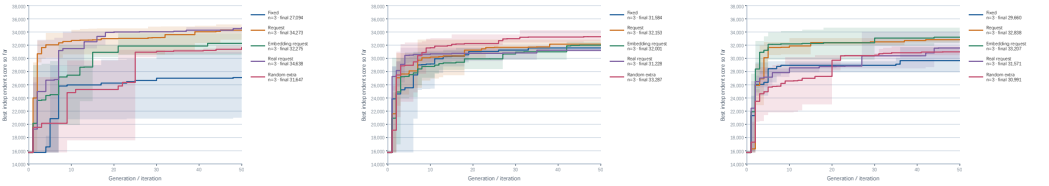
Table 1: Three-repeat means on the 100-image frozen independent test set. Parentheses show the mean difference from Fixed within the same framework and repeat; the rightmost column gives the total wins across three frameworks. Complete per-repeat results are in Appendix Table 5.

Image supply	OpenEvolve	EOH	MLES	Wins/9
Fixed	27,094	31,584	29,660	–
Request	34,273 (+7179)	32,153 (+570)	32,838 (+3177)	8
Embedding Request	32,275 (+5180)	32,001 (+417)	33,207 (+3547)	9
Real Request	34,638 (+7544)	31,228 (−356)	31,571 (+1911)	5
Random Extra	31,647 (+4552)	33,287 (+1703)	30,991 (+1331)	8



(a) OpenEvolve (b) EOH (c) MLES

Figure 5: Retrospective candidate maxima on the 100-image test set across three frameworks. Dots are per-repeat scores; the horizontal line is the median and vertical lines span the inter-repeat range. Complete values are in Appendix Table 5.



(a) OpenEvolve (b) EOH (c) MLES

Figure 6: Fifty-round candidate discovery process on the 100-image test set. Solid lines are the mean cumulative maximum over three repeats; shaded bands show the inter-repeat range.

conditional requests win under OpenEvolve, whereas Real Request under EOH has a lower mean than Fixed.

The improvement is most pronounced on OpenEvolve: all nine pairings exceed Fixed. Real Request averages 34,638, 27.8% above the Fixed mean of 27,094; MLES Request wins all three repeats with a 10.7% gain. Gains on EOH are weaker, with Real Request 1.1% below Fixed, indicating that request benefits depend on searcher dynamics. Figure 5 summarizes the final distribution, and Figure 6 shows the cumulative discovery process.

4.5 Random Extra Image Control

To assess whether the gains are associated with providing additional images, we add a Random Extra arm: each iteration also provides two extra images, selected by random seed. Random Extra exceeds Fixed in 8 of 9 matched pairings, with a mean difference of +2529. Under OpenEvolve, both Request and Real Request exceed Random Extra in all three repeats (by 2,626 and 2,991 points on average, respectively); across the three conditional requests, 8 of 9 pairings exceed Random Extra. This pattern suggests an additional benefit from candidate-conditioned case selection under OpenEvolve, while the advantage varies by search framework. Complete results are in Appendix Table 6.

4.6 Result Traceability

All 45 runs share a unified initial program, 50-round budget, and 100 frozen test images; each candidate retains its code hash, round number, and per-image evaluation. All result files pass candidate key checksums and scoring verification. Appendix D provides the five image supply schemes, the complete matrix, and pairing summaries.

5 Related Work and Conclusion

Materials and manufacturing. Early experiments on graphite superlubricity revealed the roles of contact orientation and incommensurability (Dienwiebel et al., 2004); subsequent work further quantified the contribution of graphite edge contacts to friction (Hu et al., 2024), and analyzed the dissipation mechanism of graphene grain boundary undulations (Gao et al., 2021). Centimeter-scale single-crystal graphite growth (Zhang et al., 2022), stable microscale superlubricity (Huang et al., 2023), and automatic transfer (Chen et al., 2023) have advanced manufacturing applications. This work studies the downstream packing problem: containers are determined by usable regions in images rather than pre-given single polygons (Leão et al., 2020; Martinez-Sykora et al., 2017).

Generation and algorithm evolution. StyleGAN2-ADA and Projected GAN study image generation under limited data and pretrained features (Karras et al., 2020; Sauer et al., 2021); instance generation and unsupervised environment design connect data construction to task performance (Geng et al., 2023; Dennis et al., 2020). FunSearch, EoH, ReEvo, and LLM4AD support LLM-driven program or heuristic search (Romera-Paredes et al., 2024; Liu et al., 2024a; Ye et al., 2024; Liu et al., 2024b). Grail-AD integrates controllable crystal image generation, mask conversion, and image requests into this process, enabling algorithmic progress to guide subsequent case acquisition.

Data selection and active acquisition. Active learning selects the most informative unlabeled samples for annotation, reducing labeling cost while maintaining model quality (Settles, 2012). Curriculum learning and data selection strategies further show that the order and composition of training data affect optimization outcomes (Bengio et al., 2009). In our setting, the request mechanism plays a role analogous to data acquisition: the controller decides which scales and morphologies to generate next based on algorithm feedback. Unlike active learning, which selects from a fixed unlabeled pool, our setting generates new instances on demand; unlike curriculum learning, which orders a fixed dataset, the request controller shapes the data distribution itself. This distinction makes the mechanism closer to experiment design, where the choice of what to measure is itself part of the optimization process.

6 Discussion

The benefit of request-driven image supply depends on the program search framework. All nine conditioned-request comparisons exceed Fixed under OpenEvolve, and eight exceed Random Extra. Under MLES, Request and Embedding Request each exceed both controls in all three repeats. EOH shows weaker gains against Fixed, and Random Extra exceeds every conditioned scheme in the matched repeats. These results suggest that the value of candidate-conditioned cases depends on how the search framework uses the supplied images.

Future requests can target connectivity, boundary complexity, or specific failure modes rather than only scale and representation clusters; the controller can also move from heuristic rules to a learned acquisition policy. Effective requests should balance case discriminability, remaining optimization headroom, and relevance to the real task distribution. In our setting, where real data is expensive but new instances can be generated on demand, the request mechanism provides a lightweight way to close the loop between data acquisition and algorithm progress without requiring generator retraining.

7 Conclusion

We present Grail-AD, which connects real material characterization data, controllable crystal image generation, and LLM-driven algorithm evolution for image-constrained packing of 2D graphite materials. The system has three parts: a real microscopy data organization and processing pipeline for single-crystal graphite morphology and packing; a scale-controllable crystal image generator based on representation retrieval that produces task instances at different scales and morphologies, convertible into packing-usable regions; and a dynamic image-supply mechanism in which additional cases are selected or generated as program search proceeds, rather than using only a fixed case set.

On a frozen independent 100-image test set, conditioned requests produce higher retrospective candidate maxima than matched Fixed runs in 22 of 27 comparisons. The benefit is strongest and most consistent under OpenEvolve: Real Request reaches a three-repeat mean of 34,638, a 27.8% gain over Fixed, while eight of the nine conditioned-request comparisons also exceed Random Extra. Under MLES, Request and Embedding Request each exceed both controls in all three repeats. EOH favors Random Extra over the conditioned schemes, showing that the best image supply strategy depends on the search framework. These results establish gains in candidate discovery within the tested 50-round budget; whether they translate into deployment performance or other high-data-cost domains remains to be tested.

AI Use Statement

We used generative AI in both the research method and the preparation of this paper. In the experiments, a Qwen large language model proposed candidate packing programs for the OpenEvolve, EOH, and MLES search runs. A conditional StyleGAN2-ADA model generated synthetic graphite images used to construct training and requested cases. Independent evaluation used frozen benchmarks of real graphite images.

During the research and writing process, we used OpenAI ChatGPT and Codex to discuss experimental protocols and controls; implement and debug experiment orchestration and evaluation code; organize image and score data; check numerical summaries and interpret results; and create or revise tables, plots, diagrams, LaTeX, and manuscript text. These tools also assisted with Chinese–English translation and manuscript review for clarity and consistency. Quantitative results were derived from executed runs and frozen evaluation records, not from model-generated estimates. No formal proofs or qualitative thematic analyses were part of this study. The authors take responsibility for the final methods, results, figures, text, and claims.

References

Yoshua Bengio, Jérémie Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In Proceedings of the 26th International Conference on Machine Learning, pp. 41–48, 2009.

Li Chen, Cong Lin, Diwei Shi, Xuanyu Huang, Quanshui Zheng, Jinhui Nie, and Ming Ma. Fully automatic transfer and measurement system for structural superlubric materials. Nature Communications, 14:6323, 2023. doi: 10.1038/s41467-023-41859-6.

Michael Dennis, Natasha Jaques, Eugene Vinitsky, Alexandre Bayen, Stuart Russell, Andrew Critch, and Sergey Levine. Emergent complexity and zero-shot transfer via unsupervised environment design. In Advances in Neural Information Processing Systems, volume 33, 2020.

Martin Dienwiebel, Gertjan S. Verhoeven, Namboodiri Pradeep, Joost W. M. Frenken, Jennifer A. Heimberg, and Henny W. Zandbergen. Superlubricity of graphite. Physical Review Letters, 92:126101, 2004. doi: 10.1103/PhysRevLett.92.126101.

Xiang Gao, Wengen Ouyang, Michael Urbakh, and Oded Hod. Superlubric polycrystalline graphene interfaces. Nature Communications, 12:5694, 2021. doi: 10.1038/s41467-021-25750-w.

Zijie Geng, Xijun Li, Jie Wang, Xiao Zhang, Yongdong Wu, Yuan Li, Feng Li, and Jianye Wu. A deep instance generative framework for MILP solvers under limited data availability. In Advances in Neural Information Processing Systems, volume 36, 2023.

Hengqian Hu, Jin Wang, Kaiwen Tian, Quanshui Zheng, and Ming Ma. The effects of disordered edge and vanishing friction in microscale structural superlubric graphite contact. Nature Communications, 15:10830, 2024. doi: 10.1038/s41467-024-55069-1.

Xuanyu Huang, Tengfei Li, Jin Wang, Kai Xia, Zipei Tan, Deli Peng, Xiaojian Xiang, Bin Liu, Ming Ma, and Quanshui Zheng. Robust microscale structural superlubricity between graphite and nanostructured surface. Nature Communications, 14:2931, 2023. doi: 10.1038/s41467-023-38680-6.

Tero Karras, Miika Aittala, Janne Hellsten, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Training generative adversarial networks with limited data. In Advances in Neural Information Processing Systems, volume 33, 2020.

Aline A. S. Leão, Franklina M. B. Toledo, José Fernando Oliveira, Maria Antónia Carravilla, and Ramón Alvarez-Valdés. Irregular packing problems: A review of mathematical models. European Journal of Operational Research, 282(3):803–822, 2020. doi: 10.1016/j.ejor.2019.04.045.

Fei Liu, Xialiang Tong, Mingxuan Yuan, Xi Lin, Fu Luo, Zhenkun Wang, Zhichao Lu, and Qingfu Zhang. Evolution of heuristics: Towards efficient automatic algorithm design using large language model. In Proceedings of the 41st International Conference on Machine Learning, 2024a.

Fei Liu, Rui Zhang, Zhuoliang Xie, Rui Sun, Kai Li, Xi Lin, Zhenkun Wang, Zhichao Lu, and Qingfu Zhang. LLM4AD: A platform for algorithm design with large language model. arXiv preprint arXiv:2412.17287, 2024b.

Antonio Martinez-Sykora, Ramon Alvarez-Valdes, Julia Bennell, Ruben Ruiz, and Jose M. Tamarit. Matheuristics for the irregular bin packing problem with free rotations. European Journal of Operational Research, 258(2):440–455, 2017. doi: 10.1016/j.ejor.2016.09.043.

Bernardino Romera-Paredes, Mohammadamin Barekatain, Alexander Novikov, Matej Balog, M. Pawan Kumar, Emilien Dupont, Francisco J. R. Ruiz, Jordan S. Ellenberg, Pengming Wang, Omar Fawzi, et al. Mathematical discoveries from program search with large language models. Nature, 625:468–475, 2024. doi: 10.1038/s41586-023-06924-6.

Axel Sauer, Kashyap Chitta, Jens Müller, and Andreas Geiger. Projected GANs converge faster. In Advances in Neural Information Processing Systems, volume 34, 2021.

Burr Settles. Active learning. Synthesis Lectures on Artificial Intelligence and Machine Learning, 6(1):1–114, 2012.

Haoran Ye, Jiarui Wang, Zhiguang Cao, Federico Berto, Chuanbo Hua, Haeyeon Kim, Jinkyoo Park, and Guojie Song. ReEvo: Large language models as hyper-heuristics with reflective evolution. In Advances in Neural Information Processing Systems, volume 37, 2024.

Liyuan Zhang, Meihui Wang, Dongho Jeon, Yongqiang Meng, Sun Hwa Lee, Myeonggi Choe, Yunqing Li, Mengran Wang, Sherilyn J. Lu, Zonghoon Lee, Won Kyung Seong, and Rodney S. Ruoff. Synthesis and properties of mirror-like large-grain graphite films. Nature Communications, 16:7180, 2025. doi: 10.1038/s41467-025-62227-6.

Zhibin Zhang, Mingchao Ding, Ting Cheng, Ruixi Qiao, Mengze Zhao, Mingyan Luo, Enze Wang, Yufei Sun, Shuai Zhang, Xingguang Li, et al. Continuous epitaxy of single-crystal graphite films by isothermal carbon diffusion through nickel. Nature Nanotechnology, 17 (12):1258–1264, 2022. doi: 10.1038/s41565-022-01230-0.

A Data Acquisition and Sample Organization

The dataset spans two acquisition modalities: 46 source images under Normal Light and 87 under Ring Light. The complete dataset is described in Section 2; this appendix supplements the organization of the subset used for generative model training.

A.1 Acquisition Modalities and Scanning

Under Normal Light, flat regions appear as bright background while grain boundaries, cracks, and wrinkles appear dark; Ring Light suppresses specular reflection and enhances boundary and topography contrast (Figure 8). Each appearance is processed by its corresponding recognizer.

The X/Y motion of the three-axis stage performs planar scanning, with the Z axis used for focusing. Stitching uses SIFT registration, phase-correlation refinement, and robust row/column localization, followed by dynamic-programming seam and feathering blend. Raw frames, setup photographs, the stitching pipeline, and statistics are expanded in this appendix.

A.2 Acquisition Parameters and Stitching Implementation

The imaging module uses a $5\times$ objective and a global-shutter industrial camera (60 fps). Raw frames are 3072×2048 pixels, calibrated at $1.28\,\mu$m per pixel. The three-axis translation stage has 100 mm X/Y travel and 50 mm Z travel, with repeat positioning accuracy better than $5\,\mu$m; PID-controlled stepper motors provide a planar scan step of approximately 1.1 mm.

Stitching uses grid adjacency, robust translation estimation, and seam blending; Algorithm 1 gives the full procedure.

A.3 Raw Frames and Acquisition Setup

Raw microscope frames support local registration; stitched mosaics recover large-field structure; training patches are local samples derived from source images; masks are the binary regions produced by recognition for packing use. Figure 7 shows two pairs of adjacent raw frames to illustrate the acquired content; Figure 8 shows the Ring Light setup that suppresses specular reflection.

A.4 Stitching Visualization

Figure 9 illustrates the pipeline from grid acquisition through adjacent-patch matching to seam blending. Its purpose is to combine locally clear fields of view into a structurally continuous large image, not to increase the number of independent crystals.

A.5 Released Data and Derived Patch Statistics

A.5.1 Source Images and Preprocessing

Microscopy texture images were acquired by two independent imaging systems corresponding to two illumination modalities: 46 normal-incidence (Normal Light) source images and 87 oblique ring-light (Ring Light) source images, totaling 133. Counts are based on source files and do not directly correspond to the number of independent physical samples. After acquisition, all source images were cropped to the central 50 % of the original frame and downsampled to 50 % of the original resolution.

A.5.2 Patch Extraction

Each full-field stitched image is divided into fixed-size patches for model training and evaluation. Using a 50 % overlap sliding window, 43,421 patches are extracted from the 133 source images; these are overlapping derived patches, not 43,421 independent crystals.

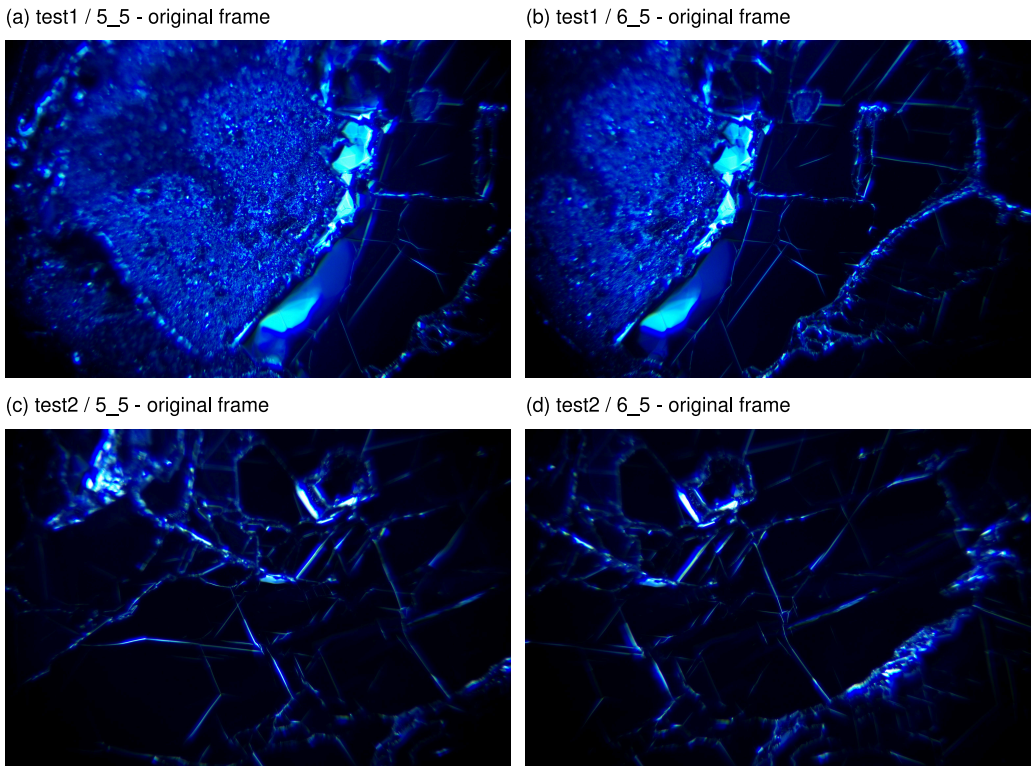
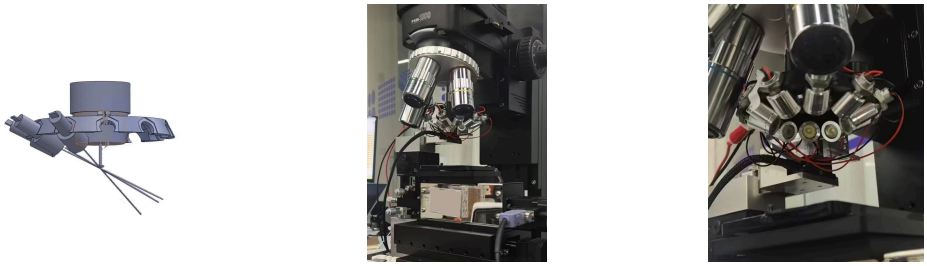


Figure 7: Ring Light raw scanning frames of the dataset. (a,b) Adjacent positions $(5, 5)$ and $(6, 5)$ in the test1 sequence; (c,d) the same index positions in the test2 sequence. Each raw frame is 3072×2048 pixels, shown at proportional scale without cropping, preserving original color and full field of view; identical indices across sequences do not correspond to the same physical location.



(a) (b) (c)

Figure 8: Ring Light acquisition setup for suppressing specular reflection and enhancing boundary contrast. (a) Model of the ring light luminaire: three small lamps are mounted on the left, the tilted thin lines indicate the beam centerlines of the lamps, the luminaire sleeves onto the microscope tube, and the bottom of the inverted-T extension below the tube marks the ideal focal point; (b) front view mounted on the microscope and translation stage; (c) bottom view mounted on the microscope and translation stage; labels have been obscured to comply with double-blind review requirements.

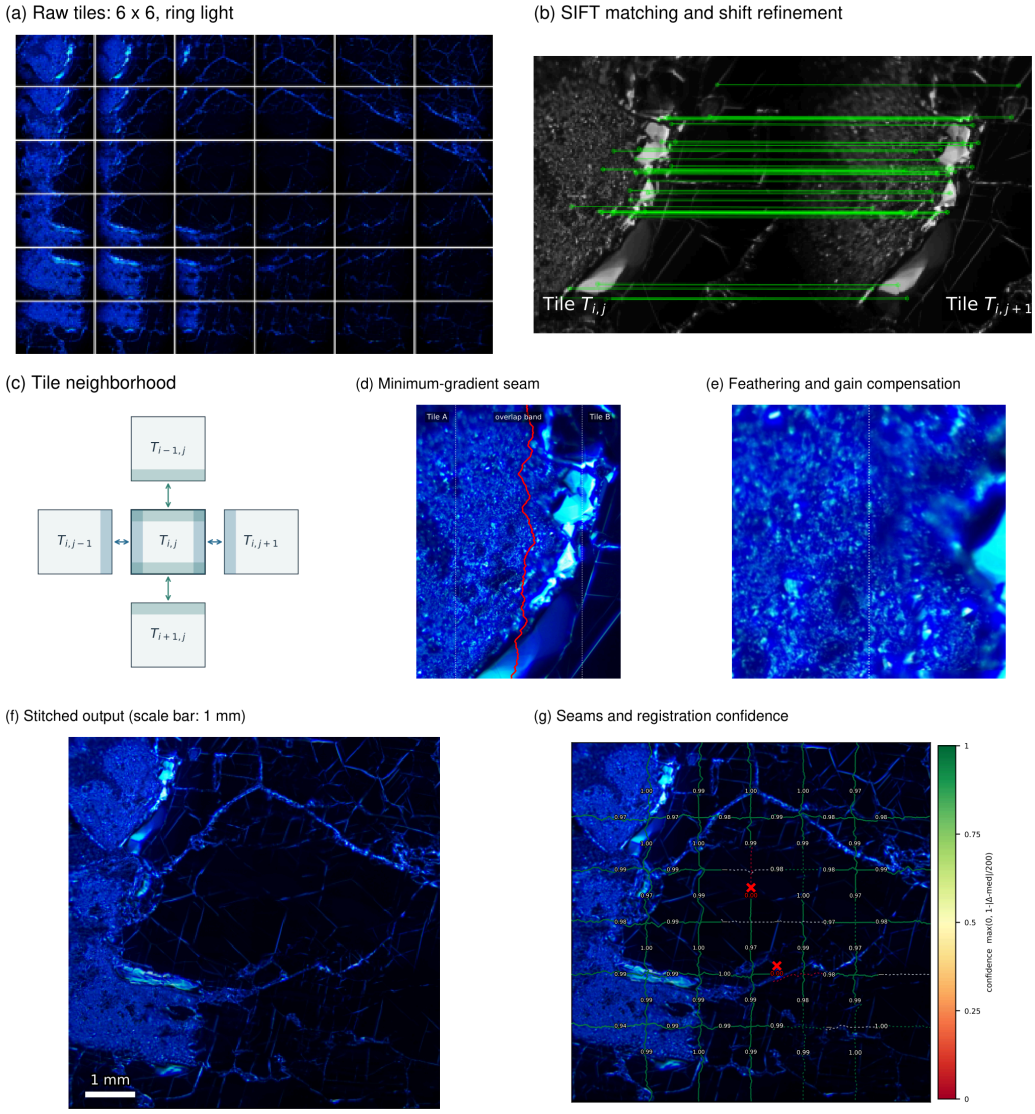


Figure 9: Processing example from raw scanning to stitched output. (a) Ring Light 6×6 raw frames; (b) SIFT feature matching with displacement refinement; (c) patch adjacency schematic (inter-patch spacing is schematic only); (d) dynamic-programming seam: red line is the seam, dashed lines mark the overlap boundary; (e) local seam after feathering and gain compensation; (f) stitched output, scale bar is 1 mm; (g) stitching canvas and DP seam lines recomputed from raw frames; colors and numbers indicate displacement consistency score $c = \max(0, 1 - |\Delta - \text{median}(\Delta)|/200)$, with medians computed separately along horizontal and vertical directions. This score is a confidence proxy rather than a calibrated probability; dashed lines indicate gating failure with median-step fallback, red crosses indicate $c \leq 0.5$, and gray dashes indicate no match.

Algorithm 1 Grid-Constrained Robust Image Stitching

Require: Frames T_{ij}; crop window W; crack masks K_{ij}
Ensure: Mosaic I and aligned crack map K

$(E_h, E_v) \leftarrow \text{GridAdjacency}(T)$
$C_{ij} \leftarrow \text{Crop}(T_{ij}, W)$; $S_{ij} \leftarrow \text{GrayDownsample}(T_{ij}, 0.25)$
for all $e = (a, b) \in E_h \cup E_v$ **do**
  $P_e \leftarrow \text{SIFTMatch}(S_a, S_b, 0.75)$
  $J_e \leftarrow \text{RANSACInliers}(P_e, 1.0 \text{ px})$
  if $|J_e| \geq 4$ **then**
    $r_e \leftarrow 4 \,\text{median}_{(p,q)\in J_e}(q - p)$
    $\delta_e \leftarrow -\text{PhaseRefine}(C_a, C_b, r_e)$
  else
    $\delta_e \leftarrow \perp$
  end if
end for
$m_h \leftarrow \text{median}\{\delta_e^x : e \in E_h, \delta_e \neq \perp\}$
$m_v \leftarrow \text{median}\{\delta_e^y : e \in E_v, \delta_e \neq \perp\}$
Use nominal directional steps if either valid set is empty.
for all $e \in E_h \cup E_v$ **do**
  $(d, m) \leftarrow (x, m_h)$ if $e \in E_h$, else (y, m_v)
  if $\delta_e \neq \perp$ and $|J_e| \geq 8$ and $|\delta_e^d - m| \leq 200$ px **then**
    $\widehat{d}_e \leftarrow \delta_e^d$
  else
    $\widehat{d}_e \leftarrow m$ ▷ Median fallback
  end if
end for
$X \leftarrow \text{RowPrefixSum}(\widehat{d}_{E_h})$; $Y \leftarrow \text{ColumnPrefixSum}(\widehat{d}_{E_v})$
$(I, K) \leftarrow \text{AllocateCanvas}(X, Y, W)$
for all (i, j) in row-major order **do**
  $\pi \leftarrow \text{MinGradientSeam}(I, C_{ij}, X_{ij}, Y_{ij})$
  $I \leftarrow \text{GainFeatherBlend}(I, C_{ij}, \pi, 60 \text{ px})$
  $K \leftarrow \text{HardSeamAssign}(K, K_{ij}, \pi)$
end for
return (I, K)

Table 2: Dataset statistics summary.

Statistic	Value
Acquisition modality	Normal Light / Ring Light
Single-frame field of view (raw frames here)	approx. 3.9 mm × 2.6 mm
Pixel calibration	1.28 μm per pixel
Source image count (Normal / Ring)	46 / 87 (133 total)
Extracted patch count	43,421 (50 % overlap)
Estimated experimental cost	approx. 50,000 USD

A.5.3 Statistics and Experimental Cost

Table 2 summarizes the main dataset statistics.

Dataset construction covers sample preparation, assembly and calibration of the custom optical system, and large-scale scanning with quality inspection. Direct experimental cost is estimated at approximately 50,000 USD, covering sample preparation consumables, equipment depreciation, and labor. This dataset provides a relatively scarce training data source for machine-learning research on graphite surface morphology.

Table 3: Data sources and evaluation subsets for the generative model.

Data level	Count and purpose
Acquired source images	46 Normal Light, 87 Ring Light, recorded by project source file
Scale-controllable generator deduplicated training list	2,020 images, deduplicated by exact content hash, for scale-distribution and training-reference diagnostics
Ring Light fine-tuning list	4,344 images, for color-transform domain adaptation and structural nearest-neighbor retrieval
Public visual reference	Fixed 20-image extraction from the scale-controllable crystal image generator training list, shared by both models

A.6 Training Subset and Provenance Tracking

The 100-image evaluation set was randomly sampled to ensure independence from the data used during algorithm evolution. Specifically, each evaluation image either comes from a different original source image than those underlying the real masks used for initial packing training and any requested data, or, if drawn from the same original source image, comes from a spatially non-overlapping region.

B Generator Training and Cross-Modality Examples

B.1 Training Interface

Training images are extracted as 1000×1000 patches and downsampled to 500×500 content; the network uses 512×512 input/output, and released inference center-crops to 500×500, which will be upsampled to required size before running the algorithm. The scale-controllable crystal image generator uses a conditional StyleGAN2-ADA backbone, adding rank-64 topological adapters on the first seven style vectors. The latent dimension is 512 and the conditional dimension is 29; Figure 11 shows the network connections.

Early joint constrained training combines the adversarial objective, projected-feature objective, and a frozen geometric regressor. Denoting the predicted condition as $\hat{s}$, the point-calibration and scale-pairing term is

$$\begin{aligned}\mathcal{L}_{\text{scale}} = \rho(\hat{s}(G(z,s)) - s) + \rho(\hat{s}(G(z,s_\ell)) - s_\ell) \\ + \rho(\hat{s}(G(z,s_h)) - s_h) + [m(s_h - s_\ell) - (\hat{s}_h - \hat{s}_\ell)]_+^2,\end{aligned} \tag{5}$$

where ρ is smooth-L_1 and pairs cover adjacent, mid-span, and endpoint scales. Auxiliary condition dropout reduces metadata shortcuts; topological differences between same-scale latent pairs provide a diversity signal. The enabled loss terms follow the per-stage configuration.

B.2 Original and Generated Images for Both Modalities

Figure 10 illustrates the conversion from local source images to training content and candidate masks. Both groups use $1\times$ local crops centered at 1000×1000 pixels and downsampled to 500×500. Normal Light preserves RGB; candidate masks are obtained by Otsu thresholding, adaptive thresholding, and morphological processing. For Ring Light source images, candidate regions are extracted where $R + G + B \leq 50$.

Ring Light training content uses the grayscale transform $R' = G' = B' = \max(0, 255 - 2(R+G+B))$. Figure 12 shows generation examples from the Normal Light model and the Ring Light fine-tuned model at 50, 70, 90, 110, and 130 pixels; Embeddings 1–4 label four selected sampling representations. The two models exhibit distinct brightness and texture styles while preserving scale-varying structural response.

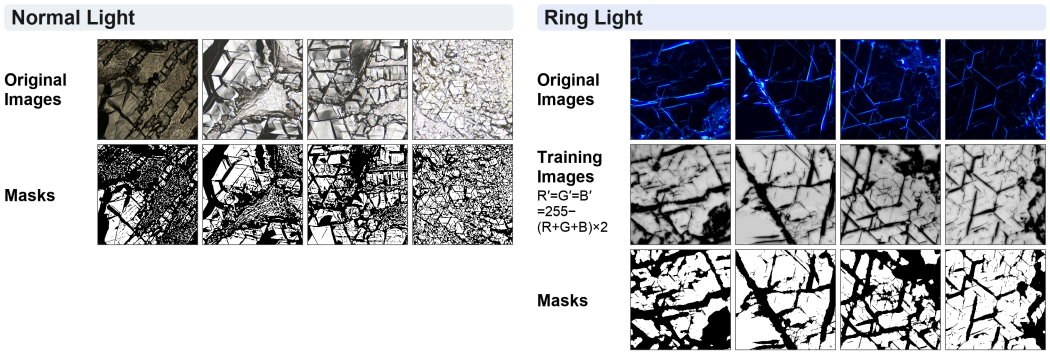


Figure 10: Preprocessing from local source images to training content. Normal Light: 1× Normal Light case with brightfield candidate mask. Ring Light: 1× Ring Light case, transformed training content, and candidate mask. White in the mask indicates candidate-available regions; black indicates unavailable regions.

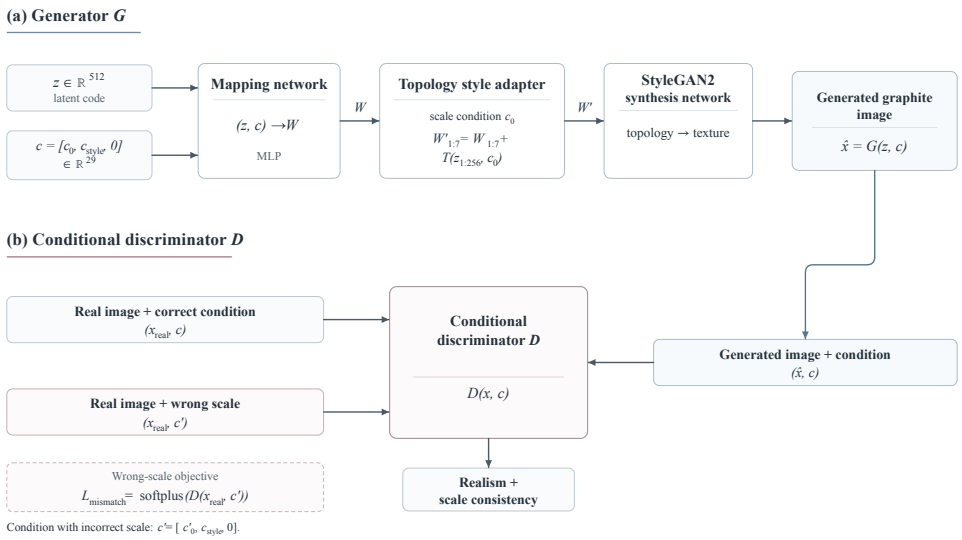


Figure 11: Conditional generative training architecture. Latent variables and conditions pass through the mapping network, topological style adapter, and StyleGAN2 synthesis network to produce images; the discriminator compares real matches, generated matches, and real mismatched conditions. Which loss terms are enabled and their weights are determined by the corresponding stage configuration.

C Quantitative Evaluation of the Scale-Controllable Crystal Image Generator

C.1 Scale Calibration and OOD Extrapolation

Scale measurement uses the frozen geometric segmenter from the training configuration, extracting eight-connected components within each non-wrinkle class $\{1, 2, 3, 4, 6\}$. The average area of the largest $\min(10, n)$ components is converted to the 500-pixel field of view, and the square root gives the characteristic length.

Figure 13 compares the scale response with the training label distribution. The purple curve is the empirical median from a 50–140 pixel scan, 10 images per step for 100 images total; the feature-length MAE is 8.55 pixels. The box plots show a 0–200 pixel scan with

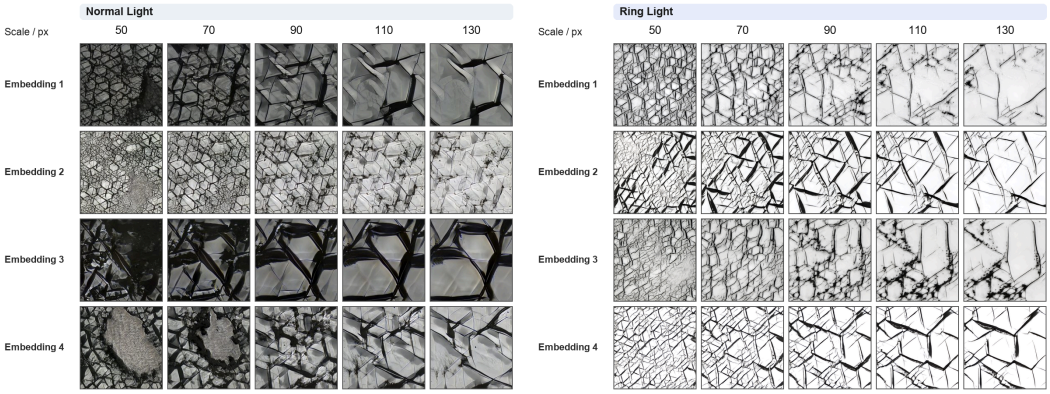


Figure 12: Representation–scale examples from the Normal Light model (left) and Ring Light model (right). Images from the Normal Light model show richer brightness gradation with faint color, but these details interfere with usable-region determination; the Ring Light model does not have this issue and can filter candidate regions directly by grayscale thresholding.

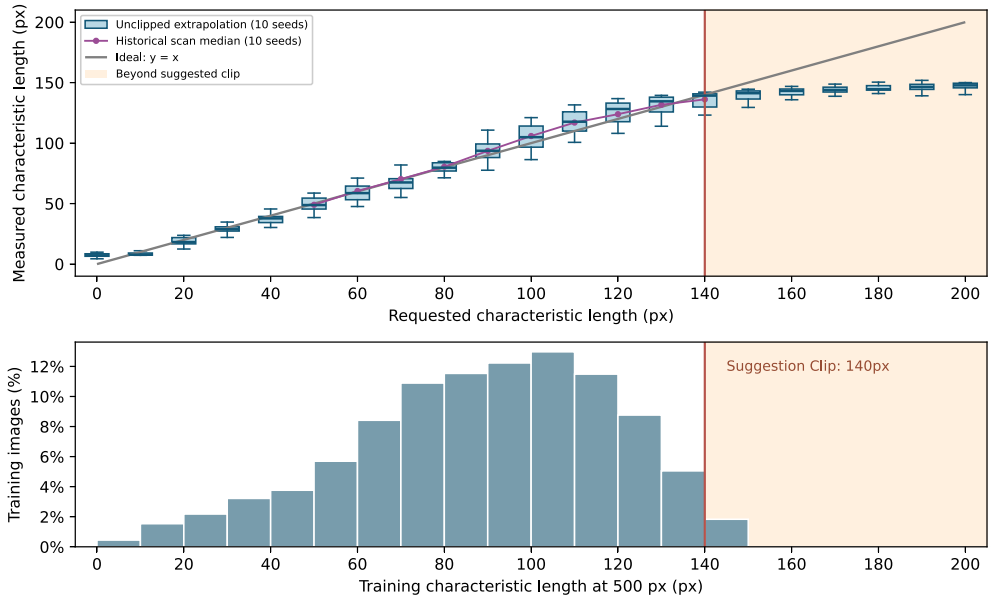


Figure 13: Scale control response of the scale-controllable crystal image generator compared with the training label distribution. Top: extrapolation box plots with 10 random seeds per step; the box center line is the median, and the purple curve is also the empirical median from the historical scan; the two groups are reported independently. The gray line is $y = x$, representing the ideal region. Bottom: scale histogram of the training images. The red line is the recommended truncation value of 140 pixels.

upper clipping removed, step 10, 10 images per step, 210 images total. The two groups are reported separately; the training set contains 2,020 images with a label-scale median of 92.15 pixels.

Response within the training scale range follows the ideal trend; beyond the covered range it gradually saturates, corresponding mainly to out-of-distribution (OOD) scale requests.

Table 4: Generated representation comparison under a common real reference. Both models use 20 real and 20 generated images; differences are referenced to the real–real cosine mean of 0.7342.

Model	Real–Generated mean	Difference from real baseline (95% interval)
Early image generator	0.6173	-0.1169 $[-0.1666, -0.0657]$
Scale-controllable crystal image generator	0.7387	$+0.0046$ $[-0.0353, 0.0351]$

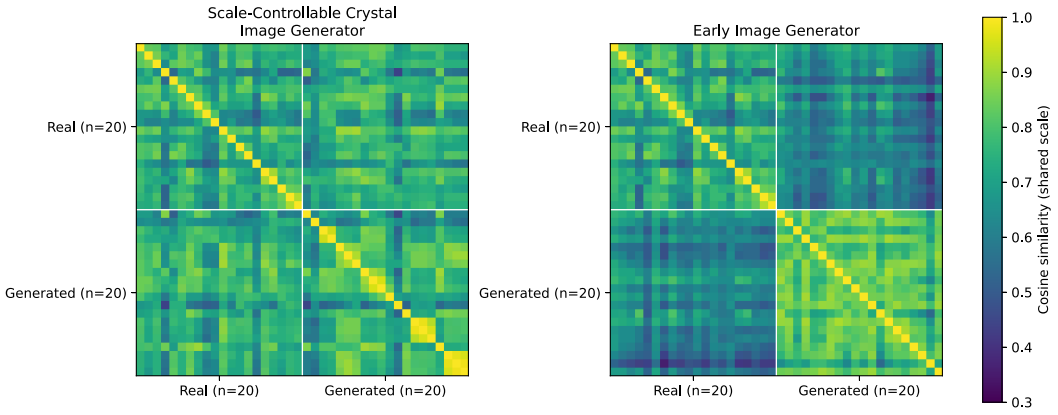


Figure 14: Real–generated ResNet cosine matrices. Left: scale-controllable crystal image generator; right: early image generator. Each matrix is composed of the same 20 real images and 20 generated images from the corresponding model. Both panels share the feature extraction protocol and color scale; self-similarity diagonals are displayed as 1.0.

C.2 Balanced Real–Generated Representation Comparison

Both models share 20 real images sampled from the training set, each using 20 generated images; sample selection does not depend on similarity.

Real and scale-controllable crystal image generator images are center-cropped to 500×500 and Lanczos-downsampled to the early image generator's native 128×128; all three groups are then uniformly resized to 224×224 with ImageNet normalization. A frozen ResNet-18 extracts layer-1–layer-4 global-pooled features concatenated to 960 dimensions, with each feature view L_2-normalized separately. Real–real (R–R) and generated–generated (G–G) statistics each use 190 distinct non-self pairs; real–generated (R–G) uses all 400 cross-group pairs; pairs sharing an image are not independent samples. Self-similarity diagonals in the matrix are displayed as 1.0 but excluded from distribution statistics.

Figures 14 and 15 show the representation matrices and similarity distributions. For the scale-controllable crystal image generator, R–R, R–G, and G–G means are 0.7342, 0.7387, and 0.7505, respectively, with highly overlapping distributions. In contrast, the early image generator has a G–G mean as high as 0.8173 but an R–G mean of only 0.6173, indicating that generated samples are overly similar to each other and insufficiently matched to real textures. The scale-controllable model therefore achieves a better balance between visual representation consistency and sample diversity.

The R–G minus R–R mean differences for the two models are $+0.0046$ and -0.1169, respectively. A 1,000-round stratified bootstrap is performed by real filename source group (15 groups) and generated random unit (9 seeds for the scale-controllable crystal image generator, 20 latent samples for the early image generator); the 95% intervals of the difference are $[-0.0353, 0.0351]$ and $[-0.1666, -0.0657]$, respectively.

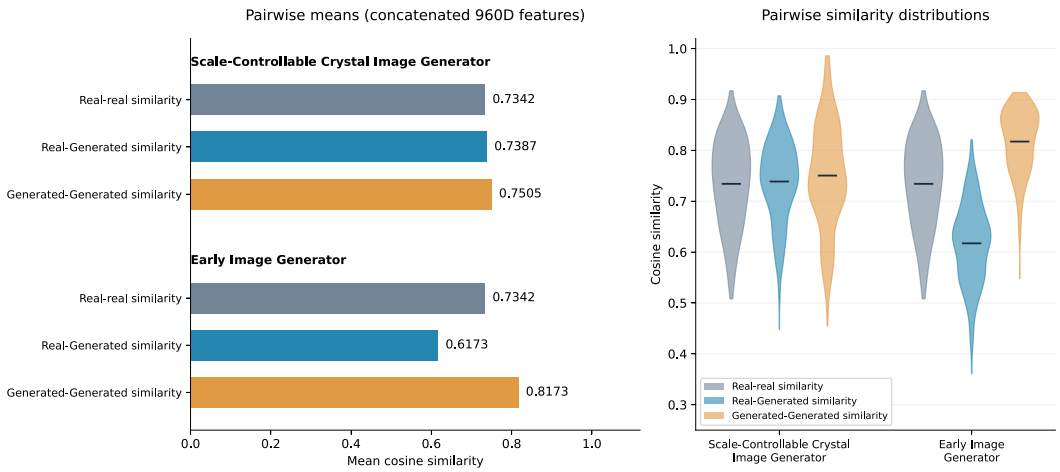


Figure 15: ResNet concatenated representation similarity comparison for the two models. Left: average cosine similarity for R–R, R–G, and G–G, with means annotated at bar tops. Right: violin plots of the three pairing similarities, with short dashes indicating means. Within-group statistics exclude self-pairs. Samples are identical to those in Figure 14.

D Image Supply Protocols and Complete Results for Qwen Program Search

D.1 Image Supply Protocols and Candidate Evaluation Pseudocode

The following pseudocode describes the image supply rules for each candidate evaluation. All five schemes—Fixed, Request, Embedding Request, Real Request, and Random Extra—are run three times each in OpenEvolve, EOH, and MLES, forming 45 matrices with 50 rounds per group. D_0 denotes the shared ten original training images, P the frozen inventory image pool, e the candidate evaluation index, $\rho(e)$ its search-round index, c the candidate code, r the repetition number, and K the number of preceding completed rounds used to identify newly detected images. All images are first converted to usable-region masks with the same semantics, then processed by the same packing validator.

Implementation and scoring. For each repetition, let A_t be the union of training images detected by candidates evaluated in completed round t, and let $N_t = A_t \setminus \bigcup_{j<t} A_j$ be the images newly detected in that round. Before candidate evaluation e, define $H_{r,e} = \bigcup_{t=\max(1,\rho(e)-K)}^{\rho(e)-1} N_t$ and $B_{r,e} = H_{r,e}$ when nonempty, otherwise $B_{r,e} = D_0$. Thus improvement refers to additional images detected during the preceding K completed rounds; the fallback applies when no such image exists. Scale is computed from the usable-region masks of $B_{r,e}$ using a rule frozen before the runs. Request randomly selects two scale-matched images from P using the repetition seed and evaluation index. Embedding Request selects the two scale-matched images in P with the highest similarity to the frozen embedding of $B_{r,e}$. Real Request uses the same scale and embedding-based exemplars to condition a request for two new images from the GraphiteGenerator. Ties and insufficient scale-matched inventory are resolved by a frozen rule. Random Extra selects images using the repetition seed and evaluation index, independent of detection history and candidate code. Only completed training-round records are used to construct $H_{r,e}$; the 100 independent test images are never used for image selection. Each framework retains its native training fitness; cross-scheme conclusions use only the aggregate score F_{100} on the 100 frozen independent test images. All candidates are evaluated with the same 100-image test set and packing validator, with candidate hash and completion status verified.

Among the 27 pairings of the three conditioned requests against Fixed, 22 are positive. By scheme, Request is 8–1, Embedding Request is 9–0, and Real Request is 5–4. By framework, OpenEvolve is 9–0, MLES is 8–1, and EOH is 5–4. The three request schemes

Algorithm 2 Fixed: Ten-Image Fixed Baseline

Require: Candidate code c, fixed training images D_0
Ensure: Training feedback f and per-image record
$D \leftarrow D_0$
for each $I \in D$ do
    Execute $c(I)$ and validate rectangle validity, non-overlap, and spacing
end for
$f \leftarrow \sum_{I \in D_0} F(I, c)$
return f and per-image record

Algorithm 3 Request: Randomly Select Scale-Matched Inventory Images

Require: c, D_0, P, r, e, K, completed detection history
Ensure: Training feedback f and per-image record
$B \leftarrow \text{RecentNewDetections}(r, e, K)$; if $B = \varnothing$, set $B \leftarrow D_0$
$s \leftarrow \text{Scale}(B)$
$P_s \leftarrow \text{ScaleMatch}(P, s)$
$D_+ \leftarrow \text{UniformSampleTwo}(P_s, \text{Hash}(r, e))$
$D \leftarrow D_0 \cup D_+$
for each $I \in D$ do
    Execute and validate $c(I)$; record $F(I, c)$ and whether I was detected
end for
$f \leftarrow \text{TrainingFitness}(\{F(I, c) : I \in D\})$
return f and per-image record

Algorithm 4 Embedding Request: Retrieve Scale-Matched Similar Images

Require: c, D_0, P, r, e, K, completed detection history, frozen image encoder
Ensure: Training feedback f and per-image record
$B \leftarrow \text{RecentNewDetections}(r, e, K)$; if $B = \varnothing$, set $B \leftarrow D_0$
$s \leftarrow \text{Scale}(B)$; $z \leftarrow \text{ImageEmbedding}(B)$
$P_s \leftarrow \text{ScaleMatch}(P, s)$
$D_+ \leftarrow \text{TopTwoBySimilarity}(P_s, z)$
$D \leftarrow D_0 \cup D_+$
for each $I \in D$ do
    Execute and validate $c(I)$; record $F(I, c)$ and whether I was detected
end for
$f \leftarrow \text{TrainingFitness}(\{F(I, c) : I \in D\})$
return f and per-image record

Algorithm 5 Real Request: Generate Images Conditioned on Embedding Request

Require: c, D_0, P, r, e, K, completed detection history, frozen image encoder and Graphite-Generator
Ensure: Training feedback f and per-image record
$B \leftarrow \text{RecentNewDetections}(r, e, K)$; if $B = \varnothing$, set $B \leftarrow D_0$
$s \leftarrow \text{Scale}(B)$; $z \leftarrow \text{ImageEmbedding}(B)$
$P_s \leftarrow \text{ScaleMatch}(P, s)$
$E \leftarrow \text{TopTwoBySimilarity}(P_s, z)$
$D_+ \leftarrow \text{GenerateTwoGraphiteImages}(s, z, E, r, e)$
$D \leftarrow D_0 \cup D_+$
for each $I \in D$ do
    Convert usable region, execute and validate $c(I)$; record $F(I, c)$ and whether I was detected
end for
$f \leftarrow \text{TrainingFitness}(\{F(I, c) : I \in D\})$
return f and per-image record

Algorithm 6 Random Extra: Additional Images Independent of Candidate Code

Require: c, D_0, P, r, e
Ensure: Training feedback f and per-image record
$q \leftarrow \text{Hash}(r, e)$ ▷ Image selection is independent of candidate code c
$D_+ \leftarrow \text{SelectTwo}(P, q)$
$D \leftarrow D_0 \cup D_+$
for each $I \in D$ do
    Execute and validate $c(I)$, record $F(I, c)$
end for
$f \leftarrow \text{TrainingFitness}(\{F(I, c) : I \in D\})$ ▷ Randomly selected extra images contribute to training feedback
return f and per-image record

Table 5: Retrospective best scores of the full 45-run Qwen matrix on the 100-image frozen independent test set. F/R/E/G/X denote Fixed, Request, Embedding Request, Real Request, and Random Extra, respectively. Each group is strictly limited to 50 rounds; the best scores in the table characterize the candidate-discovery upper bound.

Framework	Repeat	F	R	E	G	X
OpenEvolve	A	30,915	35,148	32,520	34,729	32,320
	B	29,367	34,846	33,802	34,744	30,704
	C	21,001	32,825	30,502	34,441	31,916
	Mean	27,094	34,273	32,275	34,638	31,647
EOH	A	31,162	31,916	31,594	31,064	34,275
	B	31,886	31,694	32,506	31,613	32,572
	C	31,703	32,850	31,902	31,007	33,014
	Mean	31,584	32,153	32,001	31,228	33,287
MLES	A	31,714	33,009	32,320	29,795	30,683
	B	29,391	33,077	32,646	34,029	30,482
	C	27,876	32,427	34,656	30,889	31,809
	Mean	29,660	32,838	33,207	31,571	30,991

Table 6: Average pairwise differences on the 100-image frozen independent test set. $\Delta_X = X - \text{Fixed}$; the remaining three columns are the differences between conditioned requests and Random Extra; each cell is the average over three repeats.

Framework	Δ_X	Request$-X$	Embedding$-X$	Real$-X$
OpenEvolve	+4552	+2626	+628	+2991
EOH	+1703	−1134	−1286	−2059
MLES	+1331	+1846	+2216	+580

under OpenEvolve average 6,634 points above Fixed per pairing, the most consistent gain in the full matrix.

D.2 Complete Numerical Results for Random Extra Image Control

All nine Random Extra groups across the three frameworks completed 50 rounds with the 100-image frozen independent test. The inventory pool is pre-frozen, and image selection depends only on the repetition seed and evaluation index. Table 6 computes average pairwise differences by same-framework, same-repeat retrospective best scores, separating the effect of "additional cases" from "conditioned image selection.".

In the full matrix, Random Extra vs. Fixed is 8 wins and 1 loss, with an average pairwise difference of +2529. OpenEvolve further shows an advantage for conditioned selection: Request and Real Request are each 3–0 vs. Random Extra, while Embedding Request is 2–1; across the three schemes, 8 of 9 pairings are higher, with an average additional gain of

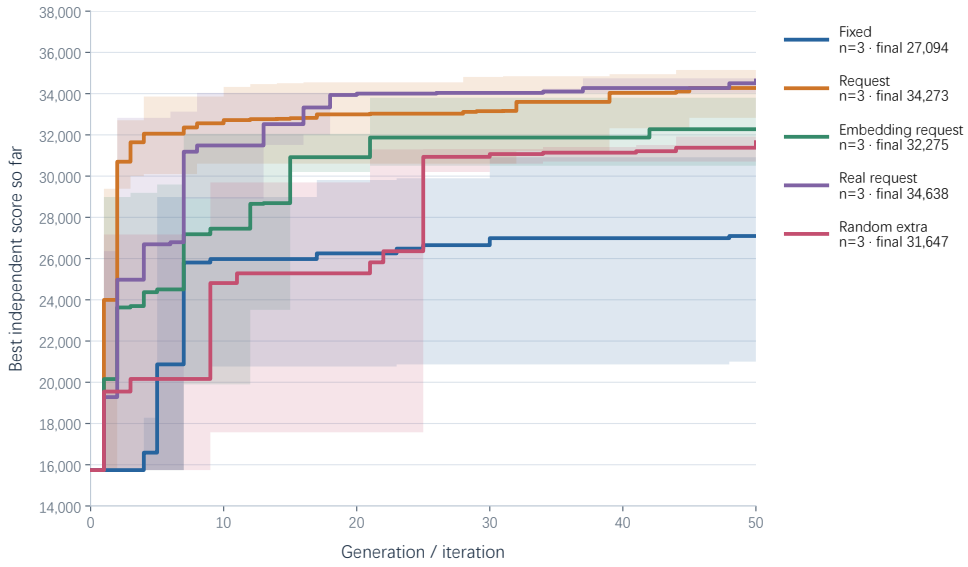


Figure 16: OpenEvolve 50-round candidate discovery process on the 100-image frozen independent test set. Solid lines are the average cumulative best score over three repeats; shaded bands show the range across repeats. Legend numbers are the round-50 averages; this figure characterizes candidate discovery trajectories.

2,082 points. In MLES, Request and Embedding Request each exceed Random Extra in all three repeats, while Real Request does so in one. In EOH, Random Extra exceeds all three conditioned schemes in every repeat. The preferred image supply scheme therefore varies with the search framework.

D.3 Conclusions and Interpretation Boundaries on the 100-Image Frozen Independent Test

The full 45-run evaluation on the frozen 100-image test set shows higher retrospective candidate maxima for conditioned requests than for Fixed in 22 of 27 matched pairings. All nine OpenEvolve pairings exceed Fixed, and eight of nine exceed Random Extra. OpenEvolve Real Request has the highest three-repeat mean in the matrix, 34,638, with a range of only 303 points across repeats and a 27.8% gain over Fixed. MLES Request and Embedding Request each win all three comparisons against Fixed and Random Extra. EOH shows a different pattern: Random Extra outperforms each conditioned scheme in all three repeats. These results support a framework-dependent benefit for request-driven image supply within the tested search budget.