CAN AI SOLVE IMAGING PROBLEMS? BENCHMARKING MULTIMODAL AI FOR IMAGE FORMATION AND RECONSTRUCTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Vision–language models (VLMs) and image-generation and editing models have demonstrated strong semantic visual capabilities, but whether they can reason about the physics, sensing mechanisms, and inverse problems underlying imaging remains largely unexplored. We introduce **ImagingBench**, a benchmark of 22 imaging tasks across five categories: ray and wave optics, image signal processing, inverse reconstruction, computational sensing, and calibration. ImagingBench evaluates multimodal models in three complementary settings: **Expert**, inverse reconstruction with fixed expert guidance; **Planner**, inverse reconstruction with model-generated planning; and **Forward**, forward image-formation simulation. Given a physics-based image-formation model, input measurements, and task context, models directly generate the target image without external code execution or task-specific optimization, probing their native multimodal imaging capabilities. We evaluate leading proprietary and open-source models against representative task-specific baselines. Across tasks, general-purpose multimodal models lag specialized methods, especially on physics-intensive problems such as lensless imaging, event reconstruction, and holography. Planning provides only modest, inconsistent gains, while visually plausible outputs often remain poorly faithful to the reference, revealing a gap between semantic competence and physically grounded imaging. **ImagingBench** provides a unified framework for quantifying this gap and tracking progress toward multimodal models that reason about imaging physics and inverse problems beyond visual appearance alone. The dataset, code and leaderboard will be made publicly available.

1 INTRODUCTION AND RELATED WORK

Benchmarks have advanced machine learning by turning broad capabilities into reproducible evaluations. Widely used suites now measure recognition, segmentation, captioning, language and multimodal reasoning, tool use, physical understanding, and controllable generation. Yet performance on these tasks does not establish whether multimodal AI systems can reason about imaging physics, image formation model and solve inverse image reconstruction problems.

Imaging depends on optics, sensing, sampling, calibration, and reconstruction. Tasks such as denoising, deblurring, demosaicking, HDR recovery, lensless reconstruction, and holography require recovering information from measurements governed by forward operators. Classical and learning-based methods have made substantial progress through physics-based modeling, optimization, and learned reconstruction pipelines (Suo et al., 2023; Wang et al., 2022; Richardson, 1972; Lucy, 1974; Rudin et al., 1992; Zhang et al., 2017). However, visual plausibility alone is insufficient when correctness depends on optical aberrations, noise statistics, or undersampling. A useful imaging model must diagnose degradations, propose physically sensible strategies, and produce results consistent with the imaging process.

Existing benchmarks provide complementary but incomplete coverage of these requirements. ImageNet, COCO, and VQA evaluate semantic perception and visual question answering (Russakovsky et al., 2015; Lin et al., 2014; Antol et al., 2015), and GMAI-MMBench extends multimodal evaluation to clinical interpretation (Chen et al., 2024); their success criteria are semantic rather than

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.

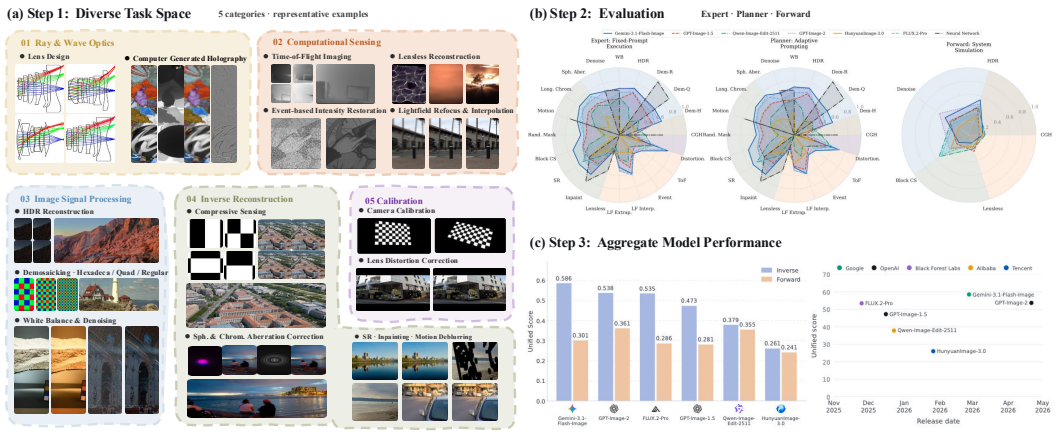


Figure 1: **Overview of ImagingBench.** (a) The benchmark comprises 22 subtasks spanning five computational imaging categories—ray and wave optics, computational sensing, image signal processing, inverse reconstruction, and calibration—covering the pipeline from scene acquisition and optical encoding to sensing, reconstruction, and calibration. (b) Each task is evaluated under three protocols: Expert (fixed expert prompt), Planner (planner-generated, image-specific instructions), and Forward (forward-system simulation). (c) Per-task scores yield one unified score per model.

physical. Closer to our setting, recent low-level vision benchmarks evaluate generative models on photographic restoration: LL-Bench ranks frontier image editors on 16 real-world restoration and enhancement tasks by human preference (Liu et al., 2026), RealIR-Bench scores editors on reference-free removal of nine real-world degradations (Yang et al., 2026), and CoTIR-Bench aggregates public restoration datasets to train and test models that reason about degradations before restoring (Guo et al., 2026). These benchmarks target degradations of everyday photographs such as noise, blur, haze, rain, and low light. None covers measurements that must be inverted through an explicit physical model, such as lensless, holographic, time-of-flight, or compressive captures, or tests whether a model can simulate the forward imaging process.

Reasoning, multimodal, and agent benchmarks extend evaluation further. BIG-bench assesses broad language reasoning (Srivastava et al., 2023); MMBench and MMMU probe multimodal perception and reasoning (Liu et al., 2024; Yue et al., 2024); AgentBench and GAIA evaluate interactive, tool-using agents (Liu et al., 2023; Mialon et al., 2024); PhysBench tests physical-world understanding (Chow et al., 2025); and MMMG evaluates multitask multimodal generation (Yao et al., 2025). These benchmarks assess relevant capabilities, but do not measure quantitative reconstruction fidelity, forward imaging behavior, or planner–executor performance in computational imaging.

Table 1: Comparison of related benchmarks #Size counts instances; #Tasks counts task types or domains. Metrics: R full-reference, N no-reference, V VLM judge, H human, A accuracy.

	SCALE		COVERAGE				
Benchmark	**#Size**	**#Tasks**	**Real**	**Physics**	**Reasoning**	**Low Lv.**	**Metrics**
General vision							
ImageNet (Russakovsky et al., 2015)	1.2M	3	✓	✗	✗	✗	A
COCO (Lin et al., 2014)	328K	2	✓	✗	✗	✗	A
VQA (Antol et al., 2015)	614K	1	✓	✗	✓	✗	A
Low-level vision							
LL-Bench (Liu et al., 2026)	2,469	16	✓	✗	✗	✓	H
CoTIR-Bench (Guo et al., 2026)	2K	19	✓	✗	✗	✓	R N
RealIR-Bench (Yang et al., 2026)	464	9	✓	✗	✗	✓	R V
Multimodal reasoning							
BIG-bench (Srivastava et al., 2023)	Var.	204	✗	✗	✓	✗	A
MMBench (Liu et al., 2024)	2,974	20	✓	✗	✓	✗	A
MMMU (Yue et al., 2024)	11.5K	30	✓	✗	✓	✗	A
Agents and physical reasoning							
AgentBench (Liu et al., 2023)	1.43K	8	✗	✗	✓	✗	V A
GAIA (Mialon et al., 2024)	466	1	✓	✗	✓	✗	A
PhysBench (Chow et al., 2025)	10.0K	4	✓	✓	✓	✗	A
Multimodal generation							
MMMG (Yao et al., 2025)	937	49	✓	✗	✓	✗	R V A
ImagingBench (ours)	**2,059**	**22**	✓	✓	✓	✓	R N V H

Foundation models are also entering computational imaging. VLMIR conditions diffusion-based restoration on vision-language priors (Yang et al., 2025), and OptiAgent couples an LLM with physics-driven rewards for optical design (Geng et al., 2026). Both are narrow-task methods, not benchmarks spanning image formation, inverse recovery, calibration, and planner-guided editing.

To address this gap, we introduce **Imaging Benchmark (ImagingBench)**, a diagnostic benchmark for multimodal AI across the computational imaging pipeline, from optical design and computational sensing to image signal processing, inverse reconstruction, and calibration. Table 1 compares it with prior benchmarks. To the best of our knowledge, ImagingBench is the first benchmark to jointly evaluate physical image formation, inverse reconstruction, calibration and design, and planner–executor behavior within a single framework.

The main contributions of this work are:

- **A systematic benchmark for multimodal AI on imaging problems.** ImagingBench comprises 22 subtasks across five categories and three complementary protocols: **Expert**, fixed expert-guided inverse reconstruction; **Planner**, instance-specific planner-guided inverse reconstruction; and **Forward**, forward image-formation simulation. Together, these protocols disentangle direct task execution, adaptive planning, and understanding of the underlying imaging model.
- **An open and unified evaluation framework.** We release the framework including benchmark dataset, protocols, evaluation infrastructure, a unified scoring framework, and public leaderboard for standardized comparison of current and future multimodal models.
- **A broad empirical study of current multimodal foundation models.** We evaluate leading proprietary systems (Gemini, GPT, FLUX) and open-weight models (Qwen, HunyuanImage) against representative task-specific baselines, complemented by cost, robustness, and failure analyses that characterize when foundation-model capabilities transfer to imaging problems.

2 Benchmark Construction

2.1 Evaluated Tasks

Computational Imaging (CI) extends conventional digital imaging by jointly designing image formation and reconstruction. Rather than treating image formation as a fixed front-end followed by generic post-processing, CI integrates physics-driven acquisition (optics and sensors) with computational reconstruction and inference (Suo et al., 2023). In ImagingBench, we evaluate whether Vision-Language Models (VLMs) can meaningfully assist CI workflows. The 22 subtasks are organized into five categories that follow the imaging pipeline (Fig. 1): **ray and wave optics** (lens design (Wang et al., 2022; Yang et al., 2024) and Computer Generated Holography (CGH) (Shi et al., 2021)); **image signal processing** (demosaicking (A Sharif et al., 2021), HDR reconstruction (Hasinoff et al., 2016), white balance (Afifi et al., 2019), and denoising (Zhang et al., 2017)); **inverse reconstruction** (aberration and motion deblurring, random-mask subsampling, block compressive sensing (Chen et al., 2015), super-resolution (Zhang et al., 2018b), and inpainting (Suvorov et al., 2022)); **computational sensing** (lensless imaging (Monakhova et al., 2019), light-field view interpolation and extrapolation, event-based reconstruction (Gao et al., 2025), and Time of Flight (ToF) depth (Qiao et al., 2022)); and **calibration** (lens distortion correction and camera intrinsics from checkerboard images). Table 7 lists each subtask's modalities and instance count.

Each instance pairs a model input (a measurement or, under Forward, a clean image) with a target and task metadata such as sensor parameters, exposure, wavelength, sampling ratio, or calibration geometry. Fig. 1(a) shows representative tasks; App. A formulates each task.

2.2 Evaluation Protocol

Each protocol is at most one planner request followed by one image-editor call (Fig. 2). Together they separate direct execution, per-image planning, and forward-model knowledge.

Expert: fixed-prompt inverse execution. The editor receives the degraded input and one expert-written prompt shared by every instance of the task. The prompt includes the forward model, task-relevant restoration guidance and constraints against structural hallucination, synthesized texture, and other generative artifacts, thus measures the editor's execution under explicit human guidance.

Planner: adaptive inverse solution. A VLM planner receives the degraded input, its metadata, and a task-specific planner prompt, and writes an image-specific execution prompt; the editor then runs once on it. For denoising, for example, the planner defines the noise characteristics and forward

image formation model, names the textures that must be preserved, and writes a short prompt with negative constraints. Comparing Planner with Expert tests whether per-image planning improves on a generic and fixed task prompt.

Forward: system simulation. Forward reverses the direction. Given the clean image and its metadata, the planner writes the forward-model equation and then an execution prompt, and the editor applies it to the clean image to synthesize the measurement, which is scored against the real one. Forward runs on the five tasks with a well-defined forward model (CGH, block compressive sensing, lensless imaging, denoising, and HDR) and tests whether a model can produce a degradation consistent with the imaging process rather than only remove one.

In every protocol, a high score shows that the output matches the target, not that the model used the forward model described in its prompt; it may instead rely on generic image priors. The supplement gives representative prompts for all three protocols with paired input/output examples.

2.3 Evaluation Metrics

For image outputs we evaluate eight metrics, four full-reference (PSNR, SSIM, LPIPS, DISTS) and four no-reference (NIQE, MUSIQ, CLIP-IQA, MANIQA), and add a full-reference VLM judge (Claude Sonnet 5 (Anthropic, 2026)) that grades whether the task was actually solved, which none of the metrics measures directly. The main-paper table reports PSNR, LPIPS, and the judge score, and the supplement the rest; definitions of all eight and of the judge are in Sec. B.1 and C.

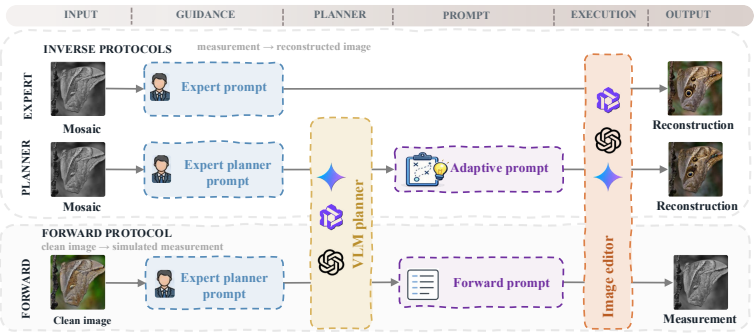


Figure 2: **Overview of the ImagingBench evaluation pipeline.** Expert pairs the measurement with a fixed expert prompt; Planner has a VLM planner write an image-specific prompt; Forward maps a clean image to a simulated measurement via a planner-written forward prompt for the editor to execute.

Unified score. Because the evaluation signals differ in scale and direction, we also summarize them with a per-instance unified score. Each signal is normalized to $[0, 1]$, with 1 indicating better quality; for example, PSNR is clipped after mapping $(\mathrm{PSNR}-15)/25$, LPIPS is converted as $1-\mathrm{LPIPS}$, and the VLM judge score is divided by 100. The unified score is the weighted mean of the normalized signals available for each instance. For CGH, ToF depth, and Forward outputs, which are not natural photographs, we instead use the unweighted mean of four normalized full-reference metrics. Task scores are obtained by averaging over instances, and *Overall* is the unweighted mean across image-task scores within each protocol. Raw per-task metrics remain the primary basis for analysis.

Human preference study. We learned the weights of the unified score from 2,665 blind pairwise judgments by 28 raters on Expert outputs from 150 images. Each trial compared two anonymized, randomized outputs alongside the input and reference. Quality control used golden pairs, side-swapped repeats, and multi-rater overlap; raters selected the reference or tie on 88% of golden trials, were self-consistent on 87% of repeats, and agreed on 87.5% of decisive shared pairs. A Bradley–Terry-style fit (Bradley & Terry, 1952) to the 1,817 retained votes outperformed equal metric weighting on held-out judgments (92.4% vs. 90.6%; 95% CI $[+0.5, +3.3]$ points). The resulting weights emphasize the VLM judge (0.35), LPIPS (0.16), DISTS (0.14), and PSNR (0.11), with SSIM receiving zero weight. These weights apply to Expert and Planner image tasks except CGH and ToF depth; Forward outputs use the unweighted mean. Details are in Sec. D.

Text-output tasks. Camera calibration and lens design return text rather than an image, so they are scored separately, excluded from Overall, and reported in App. L.

2.4 Evaluated Models

We evaluate six image-editing models: **Gemini-3.1-Flash-Image** (Nano Banana 2) (Google DeepMind, 2026a), **GPT-Image-1.5** (OpenAI, 2026a), **GPT-Image-2** (OpenAI, 2026b), **Qwen-Image-Edit-2511** (Wu et al., 2025), **HunyuanImage-3.0** (Cao et al., 2025), and **FLUX.2-Pro** (Black Forest Labs, 2025). For the Planner model, Gemini-3.1-Flash-Image and FLUX.2-Pro use **Gemini 3.1 Pro Preview** (Google DeepMind, 2026b) (FLUX.2-Pro executes Gemini's plans), Qwen-Image-Edit-2511 uses **Qwen3.5-35B-A3B** (Qwen Team, 2026), and GPT-Image-1.5, GPT-Image-2, and HunyuanImage-3.0 use **GPT-5** (OpenAI, 2025). All inputs are standardized to a 1024×1024 canvas before inference; where resizing would remove task-defining structure, we pad instead and score only the unpadded region. Access, checkpoints, and inference settings are in Sec. E.5.

2.5 Data Collection and Quality Assurance

ImagingBench targets about 100 instances per subtask: it is a diagnostic zero-shot benchmark spanning many imaging operators. For the image subtasks, we adopt an established dataset where one matches the task definition, sampling 100 images from its official test split when one exists: SIDD (Abdelhamed et al., 2018) for denoising, HDR+ (Hasinoff et al., 2016) for HDR reconstruction, MIT-CGH-4K (Shi et al., 2021) for CGH, WB Set-1 (Afifi et al., 2019) for white balance, GoPro (Nah et al., 2017) for motion deblurring, DLMD (Monakhova et al., 2019) for lensless reconstruction, DAVIS-color (Gao et al., 2025) for event-based intensity reconstruction, a real under-display ToF dataset for ToF imaging, and the Stanford Light Field Dataset (Stanford Computer Graphics Laboratory, 2016) for light-field view interpolation and extrapolation. Lens-distortion instances are our own camera captures. Where no suitable dataset exists (demosaicking, aberration correction, random-mask subsampling, block compressive sensing, super-resolution, and inpainting), we synthesize measurements by applying the task's forward operator to clean images sampled evenly from DIV2K (Agustsson & Timofte, 2017) and Adobe FiveK (Bychkovsky et al., 2011). Lens-design instances are built from target optical specifications, and camera calibration has 54 instances from three cameras. For adopted datasets we verified input–ground-truth alignment and matched the official post-processing; for synthesized subtasks we validated each forward operator. Two further components complete the benchmark: an ablation suite of controlled degradation sweeps (Sec. 4.1), and a hidden test set that will support future leaderboard evaluation without overfitting to the public split. Apps. E and G give the detailed datasets breakdown, input/output modalities, dataset-to-task mapping, lens-design and calibration construction, preprocessing, scene and photometric statistics, and the ablation settings.

3 Results

Table 2 reports per-task PSNR, LPIPS, and VLM-judge score for Expert, Planner, and Forward, with the specialized neural-network baseline alongside, and Overall, the mean unified score over each protocol's image tasks on a 0–100 scale. Fig. 1(b) shows the task-level unified scores as radar plots. Appendix J gives the per-task unified scores with coverage, SSIM, DISTS, the four no-reference metrics, and the judge's sub-scores.

Model ranking. Under Expert, Gemini leads Overall (59.4), followed by GPT-Image-2 (55.2), FLUX.2-Pro (54.1), GPT-Image-1.5 (47.8), Qwen (39.4), and HunyuanImage-3.0 (25.0); the order is the same under every treatment of refused and failed jobs we tested. Gemini has the best unified score on 14 of the 20 image tasks, including all seven inverse-reconstruction tasks, and the best PSNR on 11. The judge, however, gives GPT-Image-2 the highest score on 9 of the 18 judged tasks and Gemini on 6 (FLUX.2-Pro 2, Qwen 1), with GPT-Image-2's wins including all three demosaicking variants, denoising, white balance, and inpainting, so the two are close once task success is weighed alongside pixel agreement. FLUX.2-Pro leads both light-field tasks, and Qwen, weak on most restoration, is the strongest model on holography.

Multimodal models vs. specialized baselines. Where a specialized baseline exists, it usually leads, most clearly in PSNR: by at least 15 dB on demosaicking, 11 dB on lensless imaging, 8 dB on holography, and 6.5 dB on denoising. The gap is not uniform. On super-resolution, Gemini beats the ESRGAN baseline on PSNR, SSIM, and LPIPS (21.74 vs. 20.55 dB; LPIPS 0.24 vs. 0.39), and on white balance and denoising the best models have higher SSIM and lower LPIPS than the baseline

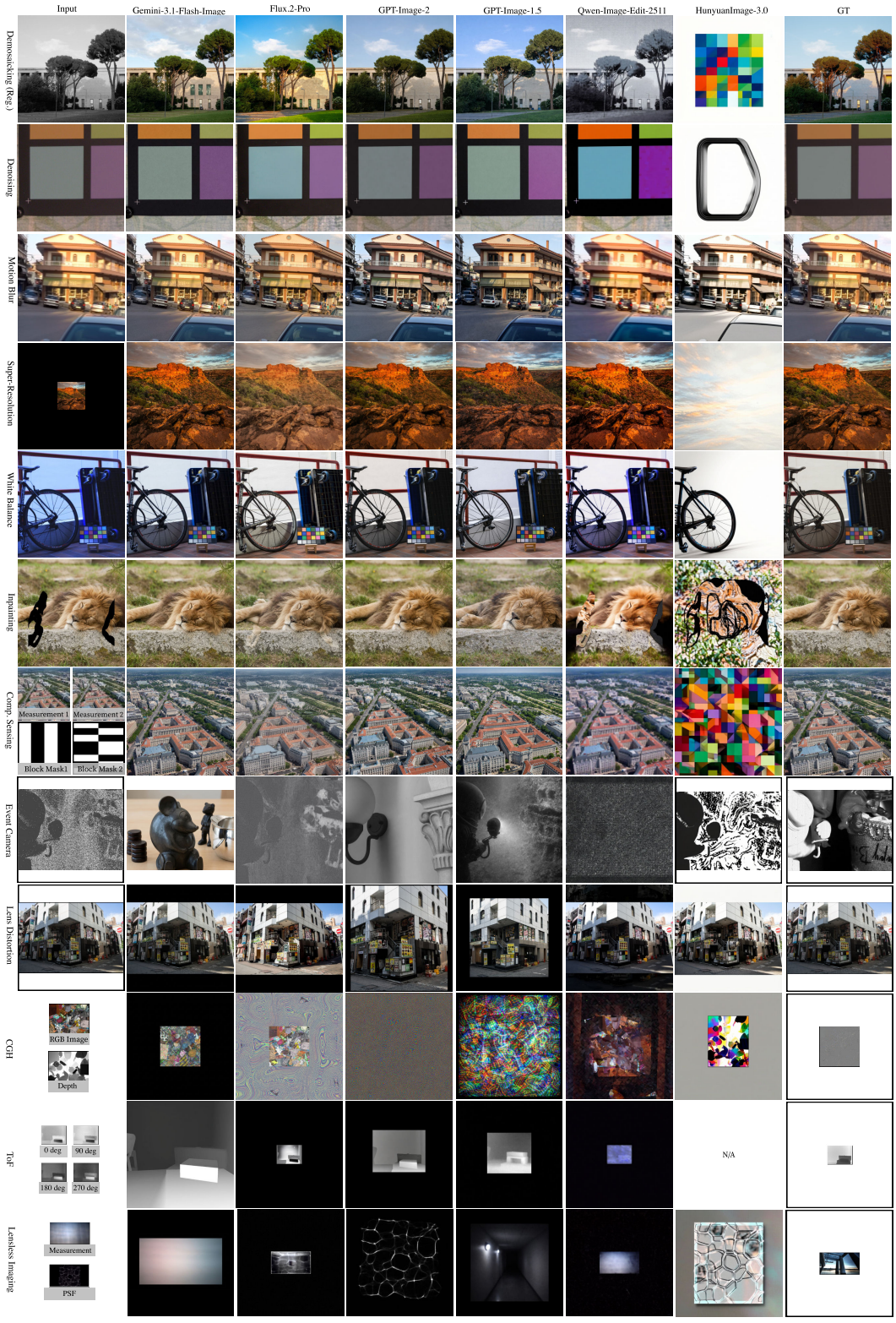


Figure 3: Representative outputs across 12 ImagingBench tasks. Columns show task inputs, outputs from six image-editing models, and ground truth (GT). The input column includes measurements, masks, or the point-spread function where applicable. More examples are in the supplement.

Table 2: **Per-task results.** Mean PSNR, LPIPS (lower is better), and VLM-judge score (0–100) over the images each model returned; a refused or failed job is left out of these means but scores 0 in Overall, the mean unified score over a protocol's image tasks. E/P/F: Expert/Planner/Forward. The judge does not score CGH, ToF depth, or Forward. Neural Network: specialized model references (cited below). "–": unavailable, including HunyuanImage-3.0's ToF jobs, all of which failed.

		Proprietary												*Open-weight*						*Reference*	
Task	**Phase**	**Gemini-3.1-Flash-Image**			**GPT-Image-2**			**FLUX.2-Pro**			**GPT-Image-1.5**			**Qwen-Image-Edit-2511**			**HunyuanImage-3.0**			**Neural Network**	
		PSNR	LPIPS	VLM	PSNR	LPIPS	VLM	PSNR	LPIPS	VLM	PSNR	LPIPS	VLM	PSNR	LPIPS	VLM	PSNR	LPIPS	VLM	PSNR	LPIPS
Inverse Process																					
Ray and Wave Optics																					
Holography (CGH)	E	6.74	1.12	–	7.72	1.02	–	10.30	0.77	–	9.59	0.85	–	17.31	0.36	–	4.45	0.71	–	25.62	0.25
	P	5.61	1.12		7.72	1.04		9.04	0.82		8.19	0.91		17.33	0.27		4.39	0.97			
Image Signal Processing																					
Demosaic (Hexa)	E	18.84	0.28	65.1	18.51	0.29	65.7	17.47	0.36	50.8	13.68	0.50	55.3	12.20	0.75	6.5	6.14	0.84	0.8	–	–
	P	18.88	0.29	55.0	18.69	0.34	27.6	18.09	0.35	28.1	13.81	0.54	43.2	14.52	0.65	9.6	6.00	0.82	1.7		
Demosaic (Quad)	E	18.93	0.28	62.4	18.77	0.28	67.3	17.87	0.35	55.3	13.84	0.48	54.8	14.74	0.65	17.4	6.03	0.86	0.4	34.70	0.04
	P	18.64	0.30	59.1	18.91	0.35	28.7	18.19	0.35	38.2	13.97	0.52	39.2	15.02	0.62	15.9	6.69	0.76	3.0		
Demosaic (Reg.)	E	18.85	0.28	61.0	19.55	0.23	72.0	17.32	0.32	52.3	14.19	0.45	60.5	13.30	0.79	11.9	6.15	0.85	0.8	38.72	0.02
	P	18.74	0.29	57.1	19.28	0.29	40.0	18.20	0.32	39.9	14.15	0.46	57.3	13.48	0.78	11.8	6.86	0.77	2.4		
HDR	E	18.14	0.23	81.5	17.18	0.26	81.4	16.52	0.29	72.5	15.73	0.39	74.5	13.68	0.48	20.7	10.95	0.61	16.4	–	–
	P	17.70	0.22	80.9	16.90	0.26	81.1	16.48	0.30	71.2	15.24	0.41	71.9	13.41	0.48	30.0	13.41	0.41	37.0		
White Balance	E	21.11	0.18	74.9	20.61	0.21	77.3	15.42	0.29	62.3	12.63	0.45	54.1	17.09	0.31	47.9	5.88	0.71	5.9	25.17	0.28
	P	22.07	0.17	74.3	21.82	0.19	76.7	16.10	0.27	63.9	12.51	0.46	49.6	17.15	0.31	40.9	6.40	0.63	12.9		
Denoising	E	25.43	0.31	76.7	26.35	0.24	77.7	16.19	0.54	60.1	18.00	0.55	64.5	20.19	0.47	41.9	4.34	0.85	4.0	32.84	0.41
	P	23.82	0.31	74.1	26.21	0.25	77.2	17.81	0.49	62.9	17.98	0.54	63.4	20.92	0.44	45.1	5.40	0.89	2.5		
Inverse Reconstruction																					
Spherical Aberr.	E	22.69	0.20	82.5	18.97	0.29	79.2	19.40	0.25	79.0	14.22	0.46	69.2	20.00	0.38	51.3	7.73	0.69	17.0	–	–
	P	22.60	0.21	81.7	20.25	0.26	81.7	19.75	0.26	79.0	14.44	0.46	70.1	19.55	0.48	36.5	8.16	0.75	11.3		
Chromatic Aberr.	E	21.52	0.23	80.2	20.47	0.23	83.0	19.32	0.31	75.6	15.71	0.45	73.0	19.62	0.37	43.6	5.69	0.83	2.4	–	–
	P	21.48	0.24	78.8	21.17	0.24	82.1	18.98	0.30	74.1	15.53	0.47	71.3	18.99	0.46	32.0	9.26	0.64	9.7		
Motion Blur	E	21.60	0.38	69.3	17.77	0.49	64.3	20.39	0.38	68.8	14.65	0.54	59.6	20.70	0.42	22.3	9.20	0.68	17.5	31.37	0.14
	P	21.56	0.37	67.6	19.67	0.42	70.8	20.63	0.37	67.1	15.16	0.52	65.1	20.42	0.46	22.1	14.22	0.58	23.5		
Masked Sampling	E	13.93	0.46	56.0	15.07	0.44	59.5	14.98	0.65	56.7	12.99	0.56	57.3	9.40	0.97	11.3	7.68	0.87	1.0	–	–
	P	13.58	0.48	55.4	13.88	0.47	61.1	14.25	0.69	55.0	11.97	0.61	53.2	10.01	0.97	15.6	7.44	0.84	2.7		
Comp. Sensing	E	21.64	0.28	81.7	15.58	0.43	80.1	15.67	0.49	60.2	14.93	0.45	73.3	20.93	0.61	56.2	9.06	0.82	6.9	–	–
	P	20.67	0.29	83.6	14.83	0.46	79.6	13.12	0.55	61.5	13.76	0.48	69.7	21.67	0.58	59.8	10.40	0.66	19.1		
Super-Resolution	E	21.74	0.24	86.8	17.69	0.31	85.1	16.41	0.49	72.8	14.36	0.46	77.0	20.16	0.32	76.1	11.30	0.54	46.5	20.55	0.39
	P	21.47	0.24	86.6	17.09	0.35	83.3	15.85	0.45	69.4	13.77	0.50	71.3	19.82	0.30	77.9	10.83	0.58	34.9		
Inpainting	E	23.78	0.12	78.3	23.80	0.12	87.1	19.49	0.28	42.4	15.18	0.41	65.3	12.81	0.49	8.3	6.76	0.80	0.8	30.33	0.05
	P	23.11	0.14	77.4	23.15	0.13	85.3	19.02	0.22	41.7	14.27	0.45	60.6	13.98	0.41	16.0	8.84	0.72	4.5		
Computational Sensing																					
Lensless Imaging	E	7.01	0.79	2.6	7.33	0.77	3.7	7.83	0.92	2.7	7.90	0.84	3.7	8.97	0.79	5.0	6.27	0.77	2.5	19.98	0.29
	P	7.03	0.79	2.9	7.62	0.83	3.4	7.44	0.91	3.3	8.02	0.89	3.7	8.81	0.76	6.4	6.97	0.78	2.5		
LF Extrapolation	E	16.42	0.45	43.7	16.24	0.51	55.6	18.75	0.46	55.5	12.30	0.67	26.9	11.90	0.63	25.8	7.91	0.81	4.2	–	–
	P	16.25	0.46	41.8	16.44	0.53	54.9	17.89	0.46	53.0	14.26	0.60	43.1	15.40	0.53	49.6	13.23	0.58	34.3		
LF Interpolation	E	21.30	0.33	66.3	19.28	0.37	71.3	20.26	0.39	73.6	15.46	0.51	59.6	16.56	0.53	39.0	8.20	0.77	4.2	–	–
	P	20.83	0.35	64.7	18.25	0.44	72.4	20.08	0.40	71.5	15.51	0.51	59.4	18.36	0.48	48.7	10.84	0.66	15.9		
Event Recon.	E	9.05	0.64	16.6	10.51	0.60	18.2	10.78	0.68	28.5	9.84	0.63	15.2	8.44	0.88	10.5	5.57	0.64	18.6	–	–
	P	9.15	0.61	20.2	10.50	0.62	27.3	9.81	0.66	29.9	10.24	0.65	23.6	7.62	0.94	12.2	5.44	0.67	14.8		
ToF Depth	E	9.04	0.60	–	8.46	0.72	–	11.12	0.62	–	8.47	0.85	–	10.33	0.75	–	–	–	–	–	–
	P	8.35	0.60		7.93	0.66		10.04	0.62		8.20	0.84		10.24	0.75						
Calibration																					
Lens Distortion	E	19.13	0.13	82.2	10.28	0.59	47.6	16.46	0.20	76.6	11.03	0.55	43.1	14.54	0.42	37.2	8.33	0.68	10.8	–	–
	P	19.03	0.13	80.8	10.42	0.58	47.7	16.53	0.19	75.6	11.28	0.54	44.3	17.01	0.29	52.1	6.38	0.71	4.8		
Leaderboard																					
Overall	E	59.4			55.2			54.1			47.8			39.4			25.0			–	
	P	57.8			52.4			53.0			46.9			36.4			27.2				
Forward Process																					
Holography (CGH)	F	10.35	0.76	–	9.87	0.72	–	10.23	0.90	–	9.99	0.91	–	9.22	0.75	–	5.64	0.82	–	–	–
HDR	F	17.93	0.32	–	15.38	0.37	–	13.74	0.62	–	13.80	0.52	–	13.38	0.47	–	7.17	0.77	–	–	–
Denoising	F	19.58	0.57	–	23.23	0.34	–	13.08	0.67	–	17.72	0.59	–	17.91	0.89	–	5.45	1.16	–	–	–
Comp. Sensing	F	20.78	0.26	–	16.32	0.58	–	15.42	0.39	–	8.79	0.72	–	16.72	0.30	–	8.23	0.60	–	–	–
Lensless Imaging	F	8.83	0.95	–	9.05	0.87	–	9.55	1.07	–	10.06	0.79	–	7.91	0.83	–	8.23	0.94	–	–	–
Leaderboard																					
Overall	F	30.1			36.1			28.6			28.1			35.5			24.1			–	

Neural Network references: CGH (Shi et al., 2021), Dm-Q (A Sharif et al., 2021), Dm-R (A Sharif et al., 2021), WB (Afifi & Brown, 2020), DN (Zhang et al., 2018a), MB (Tsai et al., 2022), SR (Wang et al., 2018), IP (Suvorov et al., 2022), LL (Monakhova et al., 2019).

despite lower PSNR. General multimodal capability thus rivals specialized methods in perceptual quality on a few common degradations, but not in fidelity, and not on computational sensing.

Fidelity vs. plausibility. PSNR is low for every multimodal model while SSIM is less degraded, so coarse scene structure is often kept. No-reference metrics cannot tell a plausible output from a correct one: averaged over tasks, HunyuanImage-3.0 is last on every full-reference metric and judged 8.9/100, yet MANIQA ranks it first of six and MUSIQ second. Fitted to human votes, the unified score likewise gives the judge the largest weight and SSIM none (Sec. 2.3).

Task-family differences. The models are most competitive on image signal processing and inverse reconstruction tasks such as denoising, white balance, inpainting, super-resolution, and aberration and motion deblurring. They are much worse on computational sensing (lensless, event-based, time-of-flight, and holography). On lensless imaging no model averages more than 5/100 from the judge under Expert. Structured sensing operators remain harder than photographic degradations.

Representative visual results. Fig. 3 shows the same split. On motion blur, super-resolution, white balance, inpainting, and compressive sensing, the four hosted editors return images close to the ground truth, differing in color and fine texture. On event, holography, and lensless inputs, they return textures or a different scene. HunyuanImage-3.0 may fail to follow the task prompt despite using an 80B LLM encoder, substantially larger than the 7B encoder used by Qwen.

Expert vs. Planner. Planner guidance gives only small and inconsistent gains. Planner changes Overall by -3.0 (Qwen) to $+2.2$ (HunyuanImage-3.0) points; five of six lose slightly. The only rank change is FLUX.2-Pro (53.0) passing GPT-Image-2 (52.4) for second. Pixel metrics hide the

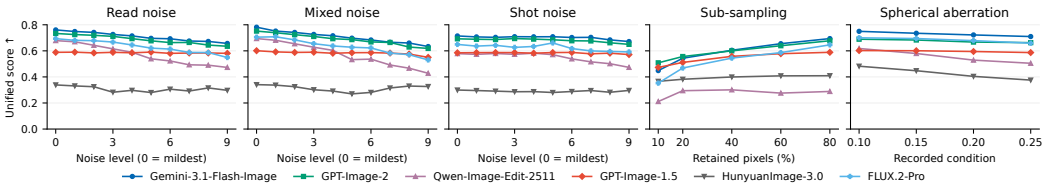


Figure 4: Expert performance across sub-sampling, spherical aberration, and three noise sweeps. Points are mean unified scores over scored pairs (higher is better), from the calibrated weights over the eight metrics; the judge was not run on these sweeps. Noise level increases with severity. Per-metric results and counts are in the supplement.

largest effect: GPT-Image-2's mean PSNR is unchanged under Planner (16.51 vs. 16.54 dB), but its judge score falls from 65.3 to 60.1, all from the three demosaicking tasks (32–39 points), where it returns grayscale images (Sec. 4.2). Base-model capability matters more than planner design.

Forward simulation. Forward reorders the models. GPT-Image-2 leads (36.1), Qwen is second (35.5), and Gemini is third (30.1), in part because it returned a usable image for only 31% of the forward compressive-sensing jobs. All six score between 24.1 and 36.1, so simulating a measurement from a clean image is not easy for these editors either. Forward covers five tasks and uses only the full-reference metrics, so its Overall is comparable only within the protocol.

4 Discussion and Analysis

4.1 Degradation Ablations

Figure 4 summarizes five degradation sweeps under the Expert protocol. Performance depends on both the model and the operator: Gemini, GPT-Image-2, and FLUX.2-Pro generally improve as more pixel measurements are retained, while Qwen remains weak on sub-sampling despite stronger performance on aberration and noise. Increasing noise generally reduces their scores, but the model ordering varies across conditions. GPT-Image-1.5 and Hunyuan show nearly flat noise curves, at a middling and a low level respectively; a flat curve reflects insensitivity to the input rather than robustness, and should not be interpreted as high reconstruction quality. Sec. G provides the per-metric results, scored-pair counts, and condition settings.

4.2 Failure Cases

The weakest tasks are those that require inverting a sensing operator. Under Expert, lensless deconvolution, event reconstruction, and CGH average only 7.55, 9.03, and 9.35 dB PSNR. The judge's per-image rationales show three distinct ways models fail on these and other tasks, all illustrated in Fig. 5. None of the three is visible to PSNR or SSIM.

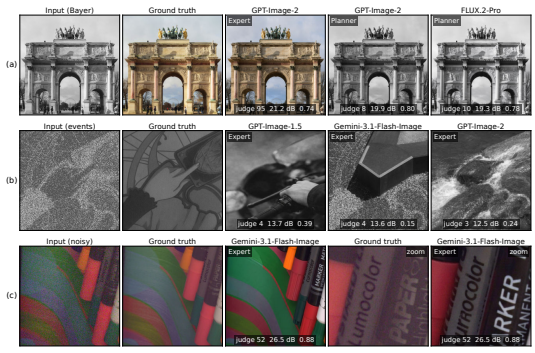


Figure 5: Three failure modes that pixel metrics miss. Labels give the protocol (top left) and the judge score, PSNR, and SSIM (bottom). (a) Demosaicking: colour under Expert, grayscale under Planner. (b) Event reconstruction: plausible but unrelated scenes. (c) Denoising, shown at 2.4× gain: Gemini-3.1-Flash-Image rewrites "Lumocolor" as "curocolor" yet scores 26.5 dB.

The plan removes the colour. On the three demosaicking tasks, measured by output colourfulness, GPT-Image-2 returns a grayscale image for none of its Expert inputs but for 59% of its Planner inputs. Its judge score halves (68.3 to 32.2) while its PSNR does not move (18.94 to 18.96 dB). The Expert prompt ends with "a clean, full-color RGB image". The planners instead write step-by-step recipes ("estimate chroma from red/blue via cross-channel ratio constraints while preserving luminance from green") that, as the planner template requires, forbid false colours, and often never say the output should be in colour. How editors read

such a plan varies: on the same GPT-5 plans, GPT-Image-1.5 goes grayscale 21% of the time; on the same Gemini-3.1-Pro plans, FLUX.2-Pro goes grayscale 51% of the time and Gemini only 7% (all four editors: at most 1% under Expert). When a plan does name a colour output, FLUX.2-Pro's grayscale rate falls from 63% to 24%. Pixel metrics favour the failure: for the three most affected editors' Planner outputs, grayscale results have *higher* PSNR (17.9 vs. 16.3 dB) and SSIM (0.65 vs. 0.53) than colour results, against judge scores of 12 and 58. A more detailed instruction can thus drop the one requirement that mattered.

A plausible but different scene. On event reconstruction, the judge describes 35% of outputs (excluding HunyuanImage-3.0) as a different scene from the ground truth: about half for Gemini, GPT-Image-1.5, and GPT-Image-2, and 10% for FLUX.2-Pro and Qwen. Lensless deconvolution follows the same pattern at 28%. These outputs are sharp, photographic, and wrong, and PSNR barely distinguishes them (9.4 vs. 9.7 dB for the rest). On event reconstruction, PSNR and the judge score are essentially uncorrelated (Spearman $\rho = 0.03$). This is the concrete form of the gap between visual plausibility and physical correctness noted in Sec. 3: the models do not invert the measurement, they replace it with a prior.

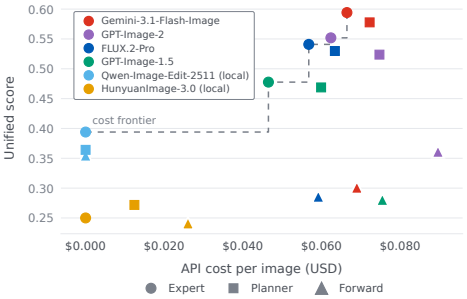


Figure 6: API cost versus performance, calculated from logged token usage and model pricing. Forward covers five tasks, so its scores are comparable only within that protocol. The dashed staircase is the cost frontier: the best score any configuration reached at or below each budget, over Expert and Planner only.

Rewritten text. Even on familiar restoration tasks with high PSNR, models rewrite legible text into different words (Fig. 5c), a failure that matters for document, forensic, or scientific imaging and that no full-reference pixel metric flags. These cases are why the judge carries the largest weight in the human-calibrated unified score (Sec. 2.3).

Hosted models also occasionally refused benign prompts or tripped safety filters (Sec. K), though this was not the dominant failure mode. Overall, current editors are competitive on familiar Image Signal Processing (ISP) and restoration tasks but break down when a task requires phase recovery, sparse inversion, or operator-aware reasoning. When they break down, they often produce confident, plausible images that pixel-level metrics score as ordinary errors.

4.3 Cost and Performance

Figure 6 relates reconstruction performance to API cost per scored image. Gemini achieves the highest score under both Expert and Planner, while GPT-Image-1.5 has the lowest API cost among the hosted models in both protocols. The extra planning call does not consistently improve performance. The cost frontier makes the trade sharper: the best free configuration, Qwen-Image-Edit-2511 under Expert (0.394), is first beaten by GPT-Image-1.5's Expert run at \$0.046 per image, which buys the largest step, 0.084 of unified score. FLUX.2-Pro's Expert run at \$0.057 adds 0.063 and GPT-Image-2's at \$0.062 another 0.011, and Gemini's Expert run at \$0.066 adds 0.042 to top the frontier. Pricing assumptions and the historical expenditure breakdown are in Sec. F.

5 Conclusion

We introduced ImagingBench, the first benchmark for evaluating multimodal AI systems on computational imaging tasks spanning image reconstruction, sensing, optics, and calibration. Across 22 subtasks and three evaluation protocols, we find a clear gap between high-level multimodal competence and physically grounded imaging understanding. Current systems often generate visually plausible outputs but remain consistently weaker than specialized methods, especially on computational sensing and other problems that require inverting a structured forward operator. Planner-guided prompting gives only modest gains over a strong fixed-prompt baseline, suggesting that the main limitation is not prompting but a lack of low-level imaging competence. We hope ImagingBench provides a foundation for measuring this gap and for guiding future work toward multimodal models that not only see images, but also understand how images are formed and reconstructed.

AI USE STATEMENT

In this work, we used generative AI tools (Claude Code and OpenAI Codex) to implement methods: most of the benchmark's dataset-preparation code, including the code that applies author-specified forward operators to synthesize degraded measurements and that cleans and reformats the source datasets, as well as the inference, evaluation, and human-study code. We also used them to give feedback on the research methodology and to help analyze and interpret results, for example by computing statistics over the recorded outputs. The research idea, task taxonomy, evaluation protocols, and unified-score design (the metrics, the VLM judge, and the human study) come from the authors and their collaborators. We did not use generative AI tools to propose hypotheses or to assist with translation, and formulating or proving mathematical claims and qualitative or thematic data analysis are not applicable to this work. Additionally, we used generative AI tools to create and edit software code, to create the scripts that generate figures and tables, to draft and edit parts of the paper for readability, and to format LaTeX. Every table of results is generated by a script from the recorded model outputs rather than typed by hand, and the authors reviewed all AI-assisted code, text, and analyses. Separately, a VLM (Claude Sonnet 5) is one of the benchmark's evaluation signals (Sec. 2.3); its prompt and validity checks are reported in the supplement. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

ImagingBench is built from publicly released imaging datasets, used under their original licenses and intended research use, together with lens-distortion and camera-calibration images we captured ourselves; the source of every task is listed in the supplement. Some source photographs depict people; we add no identifying metadata to any image.

Human subjects. The unified-score weights are fitted to a blind pairwise preference study in which 28 people compared restored images. We assessed the study as minimal risk: raters only judged which of two images better matched a reference, no personal data beyond an email address used to sign in were collected, and raters are identified only by anonymous IDs in the study data. The study was not submitted to an institutional review board.

Misuse and limitations. Several tasks ask generative models to restore or simulate measurements, and the outputs can look convincing while being physically wrong, which is the failure mode this paper documents. None of the evaluated systems should be used where imaging fidelity carries scientific, forensic, or clinical weight, and we report reference-based fidelity and a task-aware judge alongside no-reference quality so that plausible-looking hallucinations are penalized rather than rewarded.

Conflicts of interest. The study received no funding, API credits, computing resources, preferential model access, or other support from developers of the evaluated models. During manuscript preparation, some authors developed professional affiliations with an entity whose models are evaluated in this work. These affiliations arose independently of this research and did not influence model selection, experimental design, evaluation, or reporting.

REPRODUCIBILITY STATEMENT

Section 2 describes the tasks, the three protocols, the metrics and unified score, the evaluated models, and how the data were collected. The supplement gives each task's forward model and formulation (Sec. A), the metric definitions and score normalization (Sec. B.1), the VLM judge's inputs, full prompt, and validity checks (Sec. C), the human study and weight fit behind the unified score (Sec. D), the data source of every subtask, the preprocessing and alignment steps, and each model's access and inference settings (Sec. E), the cost model (Sec. F), the ablation settings (Sec. G), and the prompts behind every sample grid (Sec. M.2). The dataset-preparation code, job manifests and prompts, inference harness, scoring code, and per-image metric outputs will be released with the benchmark; the hidden test split is held back for future leaderboard evaluation.

REFERENCES

SM A Sharif, Rizwan Ali Naqvi, and Mithun Biswas. Beyond joint demosaicking and denoising: An image processing pipeline for a pixel-bin image sensor. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 233–242, 2021.

Abdelrahman Abdelhamed, Stephen Lin, and Michael S. Brown. A high-quality denoising dataset for smartphone cameras. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

Mahmoud Afifi and Michael S Brown. Deep white-balance editing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020.

Mahmoud Afifi, Brian Price, Scott Cohen, and Michael S Brown. When color constancy goes wrong: Correcting improperly white-balanced images. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 1535–1544, 2019.

Eirikur Agustsson and Radu Timofte. Ntire 2017 challenge on single image super-resolution: Dataset and study. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, July 2017.

Anthropic. Claude Sonnet 5 system card. `https://www.anthropic.com/claude-sonnet-5-system-card`, June 2026. Accessed: 2026-09-24.

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. VQA: Visual question answering. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 2425–2433, 2015. doi: 10.1109/ICCV.2015.279.

Black Forest Labs. FLUX.2: Next generation image generation. `https://bfl.ai/models/flux-2`, 2025. Accessed: 2026-09-24.

Ralph Allan Bradley and Milton E. Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345, 1952.

Vladimir Bychkovsky, Sylvain Paris, Eric Chan, and Frédo Durand. Learning photographic global tonal adjustment with a database of input / output image pairs. In *The Twenty-Fourth IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

Siyu Cao, Hangting Chen, Peng Chen, Yiji Cheng, Yutao Cui, Xinchi Deng, Ying Dong, Kipper Gong, Tianpeng Gu, Xiusen Gu, et al. HunyuanImage 3.0 technical report. *arXiv preprint arXiv:2509.23951*, 2025.

Julie Chang, Vincent Sitzmann, Xiong Dun, Wolfgang Heidrich, and Gordon Wetzstein. Hybrid optical-electronic convolutional neural networks with optimized diffractive optics for image classification. *Scientific reports*, 8(1):12324, 2018.

Chaofeng Chen and Jiadi Mo. IQA-PyTorch: Pytorch toolbox for image quality assessment. `https://github.com/chaofengc/IQA-PyTorch`, 2022.

Huaijin Chen, M Salman Asif, Aswin C Sankaranarayanan, and Ashok Veeraraghavan. Fpa-cs: Focal plane array-based compressive imaging in short-wave infrared. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2358–2366, 2015.

Pengcheng Chen, Jin Ye, Guoan Wang, Yanjun Li, Zhongying Deng, Wei Li, Tianbin Li, Haodong Duan, Ziyan Huang, Yanzhou Su, Benyou Wang, Shaoting Zhang, Bin Fu, Jianfei Cai, Bohan Zhuang, Eric J. Seibel, Yu Qiao, and Junjun He. GMAI-MMBench: A comprehensive multimodal evaluation benchmark towards general medical ai. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL `https://proceedings.neurips.cc/paper_files/paper/2024/hash/ab7e02fd60e47e2a379d567f6b54f04e-Abstract-Datasets_and_Benchmarks_Track.html`.

Wei Chow, Jiageng Mao, Boyi Li, Daniel Seita, Vitor Guizilini, and Yue Wang. Physbench: Benchmarking and enhancing vision-language models for physical world understanding. In *ICLR*, 2025. URL `https://arxiv.org/abs/2501.16411`.

Keyan Ding, Kede Ma, Shiqi Wang, and Eero P. Simoncelli. Image quality assessment: Unifying structure and texture similarity. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(5):2567–2581, 2022.

Qiyao Gao, Peiqi Duan, Hanyue Lou, Minggui Teng, Ziqi Cai, Xu Chen, and Boxin Shi. Unified reconstruction of static and dynamic scenes from events. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 27914–27923, 2025.

Yuyu Geng, Lei Sun, Yao Gao, Xinxin Hu, Zhonghua Yi, Xiaolong Qian, Weijian Hu, Jian Bai, and Kaiwei Wang. Optiagent: A physics-driven agentic framework for automated optical design, 2026. URL https://arxiv.org/abs/2602.23761.

Google DeepMind. Gemini 3.1 flash-image model card. https://deepmind.google/models/model-cards/gemini-3-1-flash-image/, 2026a. Accessed: 2026-02-26.

Google DeepMind. Gemini 3.1 Pro. https://deepmind.google/models/gemini/pro/, 2026b. Accessed: 2026-09-24.

Yu Guo, Zhengru Fang, Shengfeng He, Senkang Hu, Yihang Tao, Phone Lin, and Yuguang Fang. Universal image restoration via internalized chain-of-thought reasoning. *arXiv preprint arXiv:2606.17557*, 2026.

Samuel W. Hasinoff, Dillon Sharlet, Ryan Geiss, Andrew Adams, Jonathan T. Barron, Florian Kainz, Jiawen Chen, and Marc Levoy. Burst photography for high dynamic range and low-light imaging on mobile cameras. *ACM Transactions on Graphics (Proc. SIGGRAPH Asia)*, 35(6), 2016.

Junjie Ke, Qifei Wang, Yilin Wang, Peyman Milanfar, and Feng Yang. MUSIQ: Multi-scale image quality transformer. In *ICCV*, 2021.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common objects in context. In *European Conference on Computer Vision (ECCV)*, pp. 740–755, 2014. doi: 10.1007/978-3-319-10602-1_48.

Lu Liu, Huiyu Duan, Chenxin Zhu, Jintong Lu, Haoyun Jiang, Liu Yang, Qiang Hu, Guangtao Zhai, and Xiaoyun Zhang. Ll-bench: Rethinking low-level vision evaluation in the era of large-scale generative models. *arXiv preprint arXiv:2606.02535*, 2026.

Xiao Liu, Hao Yu, Hanchen Zhang, Yifan Xu, Xuanyu Lei, Hanyu Lai, Yu Gu, Hangliang Ding, Kaiwen Men, Kejuan Yang, Shudan Zhang, Xiang Deng, Aohan Zeng, Zhengxiao Du, Chenhui Zhang, Sheng Shen, Tianjun Zhang, Yu Su, Huan Sun, Minlie Huang, Yuxiao Dong, and Jie Tang. AgentBench: Evaluating LLMs as agents, 2023. URL https://arxiv.org/abs/2308.03688.

Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, Kai Chen, and Dahua Lin. MMBench: Is your multi-modal model an all-around player? In *European Conference on Computer Vision (ECCV)*, pp. 216–233, 2024. doi: 10.1007/978-3-031-72658-3_13.

Leon B. Lucy. An iterative technique for the rectification of observed distributions. *The Astronomical Journal*, 79:745–754, 1974.

Grégoire Mialon, Clémentine Fourrier, Thomas Wolf, Yann LeCun, and Thomas Scialom. GAIA: A benchmark for general AI assistants. In *International Conference on Learning Representations (ICLR)*, 2024. URL https://openreview.net/forum?id=fibxvahvs3.

Anish Mittal, Rajiv Soundararajan, and Alan C. Bovik. Making a "completely blind" image quality analyzer. *IEEE Signal Processing Letters*, 20(3):209–212, 2012.

Kristina Monakhova, Joshua Yurtsever, Grace Kuo, Nick Antipa, Kyrollos Yanny, and Laura Waller. Learned reconstructions for practical mask-based lensless imaging. *Optics express*, 27(20): 28075–28090, 2019.

Seungjun Nah, Tae Hyun Kim, and Kyoung Mu Lee. Deep multi-scale convolutional neural network for dynamic scene deblurring. In *CVPR*, July 2017.

OpenAI. *GPT-5 Model Documentation*. OpenAI, 2025. URL https://developers.openai.com/api/docs/models/gpt-5. Snapshot gpt-5-2025-08-07. Accessed: 2026-09-24.

OpenAI. *GPT-Image 1.5 Model Documentation*. OpenAI, San Francisco, CA, 2026a. URL https://developers.openai.com/api/docs/models/gpt-image-1.5. Accessed: 2026-02-26.

OpenAI. *GPT-Image-2 Model Documentation*. OpenAI, 2026b. URL https://developers.openai.com/api/docs/models/gpt-image-2. Snapshot gpt-image-2-2026-04-21. Accessed: 2026-09-24.

Xin Qiao, Chenyang Ge, Pengchao Deng, Hao Wei, Matteo Poggi, and Stefano Mattoccia. Depth restoration in under-display time-of-flight imaging. *IEEE transactions on pattern analysis and machine intelligence*, 45(5):5668–5683, 2022.

Qwen Team. Qwen3.5: Towards native multimodal agents. https://huggingface.co/Qwen/Qwen3.5-35B-A3B, February 2026. Accessed: 2026-09-24.

William H. Richardson. Bayesian-based iterative method of image restoration. *Journal of the Optical Society of America*, 62(1):55–59, 1972.

Eric Rollei. HunyuanImage-3.0-Instruct-Distil-INT8-v2. https://huggingface.co/EricRollei/HunyuanImage-3.0-Instruct-Distil-INT8-v2, 2026. INT8 quantization of tencent/HunyuanImage-3.0-Instruct-Distil. Accessed: 2026-09-24.

Leonid I. Rudin, Stanley Osher, and Emad Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60(1-4):259–268, 1992.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015. doi: 10.1007/s11263-015-0816-y.

Liang Shi, Beichen Li, Changil Kim, Petr Kellnhofer, and Wojciech Matusik. Towards real-time photorealistic 3d holography with deep neural networks. *Nature*, 591(7849):234–239, 2021.

Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, et al. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *Transactions on Machine Learning Research*, 2023. URL https://openreview.net/forum?id=uyTL5Bvosj.

Stanford Computer Graphics Laboratory. The stanford light field archive (2016). https://lightfields.stanford.edu/LF2016.html, 2016. Accessed: 2026-02-26.

Jinli Suo, Weihang Zhang, Jin Gong, Xin Yuan, David J Brady, and Qionghai Dai. Computational imaging and artificial intelligence: The next revolution of mobile vision. *Proceedings of the IEEE*, 111(12):1607–1639, 2023.

Roman Suvorov, Elizaveta Logacheva, Anton Mashikhin, Anastasia Remizova, Arsenii Ashukha, Aleksei Silvestrov, Naejin Kong, Harshith Goka, Kiwoong Park, and Victor Lempitsky. Resolution-robust large mask inpainting with fourier convolutions. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pp. 2149–2159, 2022.

Fu-Jen Tsai, Yan-Tsung Peng, Chung-Chi Tsai, Yen-Yu Lin, and Chia-Wen Lin. Banet: A blur-aware attention network for dynamic scene deblurring. *IEEE Transactions on Image Processing*, 31:6789–6799, 2022.

Congli Wang, Ni Chen, and Wolfgang Heidrich. do: A differentiable engine for deep lens design of computational imaging systems. *IEEE Transactions on Computational Imaging*, 8:905–916, 2022.

Jianyi Wang, Kelvin C. K. Chan, and Chen Change Loy. Exploring CLIP for assessing the look and feel of images. In *AAAI*, 2023.

Xintao Wang, Ke Yu, Shixiang Wu, Jinjin Gu, Yihao Liu, Chao Dong, Yu Qiao, and Chen Change Loy. Esrgan: Enhanced super-resolution generative adversarial networks. In *European Conference on Computer Vision (ECCV)*, pp. 0–0, 2018.

Zhou Wang, Alan C. Bovik, Hamid R. Sheikh, and Eero P. Simoncelli. Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4): 600–612, 2004.

Kaixuan Wei, Ying Fu, Jiaolong Yang, and Hua Huang. A physics-based noise formation model for extreme low-light raw denoising. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2758–2767, 2020.

Chenfei Wu, Jiahao Li, Jingren Zhou, Junyang Lin, Kaiyuan Gao, Kun Yan, Sheng ming Yin, Shuai Bai, Xiao Xu, Yilei Chen, Yuxiang Chen, Zecheng Tang, Zekai Zhang, Zhengyi Wang, An Yang, Bowen Yu, Chen Cheng, Dayiheng Liu, Deqing Li, Hang Zhang, Hao Meng, Hu Wei, Jingyuan Ni, Kai Chen, Kuan Cao, Liang Peng, Lin Qu, Minggang Wu, Peng Wang, Shuting Yu, Tingkun Wen, Wensen Feng, Xiaoxiao Xu, Yi Wang, Yichang Zhang, Yongqiang Zhu, Yujia Wu, Yuxuan Cai, and Zenan Liu. Qwen-image technical report, 2025. URL https://arxiv.org/abs/2508.02324.

Yicheng Wu, Vivek Boominathan, Huaijin Chen, Aswin Sankaranarayanan, and Ashok Veeraraghavan. Phasecam3d—learning phase masks for passive single view depth estimation. In *2019 IEEE International Conference on Computational Photography (ICCP)*, pp. 1–12. IEEE, 2019.

Cuixin Yang, Rongkang Dong, and Kin-Man Lam. Vision-language model guided image restoration, 2025. URL https://arxiv.org/abs/2512.17292.

Sidi Yang, Tianhe Wu, Shuwei Shi, Shanshan Lao, Yuan Gong, Mingdeng Cao, Jiahao Wang, and Yujiu Yang. MANIQA: Multi-dimension attention network for no-reference image quality assessment. In *CVPR Workshops*, 2022.

Xinge Yang, Qiang Fu, and Wolfgang Heidrich. Curriculum learning for ab initio deep learned refractive optics. *Nature communications*, 15(1):6572, 2024.

Yufeng Yang, Xianfang Zeng, Zhangqi Jiang, Fukun Yin, Jianzhuang Liu, Wei Cheng, Shiyu Liu, Yuqi Peng, Gang YU, Shifeng Chen, et al. Realrestorer: Towards generalizable real-world image restoration with large-scale image editing models. *arXiv preprint arXiv:2603.25502*, 2026.

Jihan Yao, Yushi Hu, Yujie Yi, Bin Han, Shangbin Feng, Guang Yang, Bingbing Wen, Ranjay Krishna, Lucy Lu Wang, Yulia Tsvetkov, Noah A. Smith, and Banghua Zhu. Mmmg: a comprehensive and reliable evaluation suite for multitask multimodal generation, 2025. URL https://arxiv.org/abs/2505.17613.

Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, Cong Wei, Botao Yu, Ruibin Yuan, Renliang Sun, Ming Yin, Boyuan Zheng, Zhenzhu Yang, Yibo Liu, Wenhao Huang, Huan Sun, Yu Su, and Wenhu Chen. MMMU: A massive multi-discipline multimodal understanding and reasoning benchmark for expert AGI. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9556–9567, 2024.

Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang. Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. *IEEE transactions on image processing*, 26 (7):3142–3155, 2017.

Kai Zhang, Wangmeng Zuo, and Lei Zhang. Ffdnet: Toward a fast and flexible solution for cnn-based image denoising. *IEEE Transactions on Image Processing*, 27(9):4608–4622, 2018a.

Kai Zhang, Wangmeng Zuo, and Lei Zhang. Learning a single convolutional super-resolution network for multiple degradations. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3262–3271, 2018b.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018c.

A TASK FORMULATION

Each instance i of task t is a tuple (x_i, q_i, y_i, m_i): the model input x_i, an optional text instruction q_i, the target y_i, and task metadata m_i such as sensor parameters, exposure, wavelength, sampling ratio, or calibration geometry. A task-specific forward model describes how a latent scene or clean image z_i gives rise to its measurement; depending on the protocol, the model either inverts this process or directly simulates it.

Restoration and inverse reconstruction. Most tasks invert a single degraded measurement,

$$x_i = \mathcal{A}_t(z_i; m_i) + \epsilon_i, \qquad y_i = z_i,$$

where ϵ_i is noise or model mismatch and the operator $\mathcal{A}_t$ is one of

Operator	Tasks	Metadata m_i
$\mathcal{N}$, sensor noise	denoising (real SIDD noise; Poisson–Gaussian in the ablation)	noise level
$\mathcal{W}_{\mathbf{g}}$, per-channel gains	white balance	illuminant
$\mathcal{C}_{\text{cfa}}$, colour-filter sampling	demosaicking (regular, quad, hexadeca)	CFA pattern
$\mathcal{H}_h$, convolution with a PSF	spherical and longitudinal chromatic aberration, motion blur	PSF, wavelength
$\mathcal{M}_\Omega$, pixel masking	inpainting, random-mask subsampling	mask Ω
$\mathcal{D}_s$, downsampling	super-resolution	scale s
$\mathcal{P}_\Phi$, block projection	block compressive sensing	patterns Φ, ratio
$\mathcal{G}_{\mathbf{k}}$, radial warp	lens distortion	distortion coefficients $\mathbf{k}$

Multi-measurement fusion. HDR reconstruction and light-field view synthesis take a set of measurements,

$$x_i = \left\{\mathcal{A}_t^{(r)}(z_i; m_i) + \epsilon_i^{(r)}\right\}_{r=1}^{R}, \qquad y_i = \mathcal{T}_t(z_i; m_i) :$$

exposures of different lengths, whose target is the HDR radiance image, or sparse views of a light field, whose target is a held-out view (interpolated or extrapolated).

Computational sensing. Lensless, event-based, time-of-flight and holography tasks use a structured sensing model $\mathcal{S}_t$ and may ask for a representation $\mathcal{R}_t$ other than the image,

$$x_i = \mathcal{S}_t(z_i; m_i), \qquad y_i = \mathcal{R}_t(z_i; m_i).$$

In lensless imaging $\mathcal{S}_t$ is mask-based propagation onto the sensor; in event imaging it yields asynchronous brightness changes; in ToF imaging it yields raw phase and amplitude and y_i is depth. CGH is governed by free-space propagation $\mathcal{P}_{z,\lambda}$ at distance z and wavelength λ: the phase pattern $u_{0,i}$ must produce the target intensity $y_i = |\mathcal{P}_{z,\lambda}(u_{0,i})|^2$.

Forward image formation. The Forward protocol evaluates the complementary direction: given a clean image or latent scene and the specified imaging model, the model predicts the measurement produced by the forward process. We write

$$x_i = z_i, \qquad y_i = \mathcal{F}_t(z_i; m_i) + \epsilon_i,$$

where $\mathcal{F}_t$ denotes the task-specific physical or computational image-formation model and m_i contains its parameters. For restoration and reconstruction tasks, $\mathcal{F}_t = \mathcal{A}_t$ is the corresponding degradation or measurement operator; for computational sensing tasks, $\mathcal{F}_t = \mathcal{S}_t$ may produce a structured sensor-domain representation rather than a conventional photograph. For multi-measurement systems,

$$y_i = \left\{\mathcal{F}_t^{(r)}(z_i; m_i) + \epsilon_i^{(r)}\right\}_{r=1}^{R}.$$

Thus, unlike the Expert and Planner protocols, which recover z_i from its measurements, Forward directly tests whether the model can execute the provided image-formation model to synthesize

the expected observation. In compact form, Expert and Planner evaluate inversion of the imaging process, whereas Forward evaluates direct execution of the forward model:

$$\underbrace{x = \mathcal{F}(z) \to z}_{\text{Expert / Planner}} \qquad \text{vs.} \qquad \underbrace{z \to \mathcal{F}(z)}_{\text{Forward}}.$$

Text-output tasks. In lens design the input is a specification and the output a prescription,

$$y_i^\star = \arg\min_{y \in \Omega(m_i)} \mathcal{L}_{\text{opt}}(y; m_i),$$

over the design space $\Omega(m_i)$ fixed by field of view, F-number, sensor size, materials and element count; a prediction is evaluated by optical simulation.

In camera calibration the input is a set of checkerboard views $\{I_{ij}\}_{j=1}^{J_i}$ and the target is the intrinsics and distortion $(\mathbf{K}_i, \mathbf{d}_i)$, which project each 3D corner $\mathbf{P}_\ell$ at pose $(\mathbf{R}_{ij}, \mathbf{t}_{ij})$ to its image point, $\mathbf{p}_{ij\ell} \sim \Pi(\mathbf{K}_i, \mathbf{d}_i, \mathbf{R}_{ij}, \mathbf{t}_{ij}, \mathbf{P}_\ell)$.

B Definition of the Score Function

We compute eight metrics with the default implementations of the `pyiqa` toolbox (Chen & Mo, 2022). Four are full-reference. PSNR is $10 \log_{10}(L^2/\text{MSE})$ for peak value L, over the valid (unpadded) pixels. SSIM (Wang et al., 2004) compares local luminance, contrast and structure. LPIPS (Zhang et al., 2018c) is a learned distance between deep features, and DISTS (Ding et al., 2022) compares deep-feature structure and texture statistics, so it tolerates resampled texture that pixel metrics penalize. Four are no-reference. NIQE (Mittal et al., 2012) is the distance of the image's natural-scene statistics from a model fitted to pristine images. MUSIQ (Ke et al., 2021), CLIP-IQA (Wang et al., 2023) and MANIQA (Yang et al., 2022) are learned quality predictors. Lower is better for LPIPS, DISTS and NIQE, and higher for the rest. The ninth signal is the VLM judge (Sec. C).

B.1 Normalization and the unified score

Each metric is mapped to $[0, 1]$, with 1 best, by clipping the following expressions to $[0, 1]$: $(\text{PSNR}-15)/25$, SSIM, $1 - \text{LPIPS}$, $1 - \text{DISTS}$, $(20 - \text{NIQE})/17$, $\text{MUSIQ}/100$, CLIP-IQA, MANIQA, and $S/100$ for the judge's overall score S. The instance score is the weighted mean

$$s_i = \frac{\sum_{k \in \mathcal{A}_i} w_k \, s_{i,k}}{\sum_{k \in \mathcal{A}_i} w_k}, \qquad \text{summing over the available metrics } k \in \mathcal{A}_i,$$

so a metric that is missing for an instance drops out and the remaining weights are renormalized. The weights w_k are fitted to human pairwise preferences (Sec. D, Table 5). For computer-generated holography, time-of-flight depth and every Forward-protocol output, which are not photographs, the no-reference metrics and the judge do not apply and s_i is the unweighted mean of the four normalized full-reference metrics. The degradation ablations were not judged and use the calibrated weights over the eight metrics, renormalized. The two text-output tasks have their own scores (Sec. L).

C VLM Judge

Conventional image-quality metrics do not explicitly assess whether the requested imaging transformation has been correctly performed. Reference-based metrics such as PSNR and SSIM can favor outputs that preserve scene structure and color even when the underlying degradation remains, while no-reference metrics do not observe the target. We therefore introduce a vision–language model (VLM) judge that compares the output against the reference and evaluates task-specific correctness alongside perceptual quality.

C.1 Judge inputs

For sample i the judge J receives one message containing, in this order and each under a text label: the degraded input image or images x_i (`INPUT k of n`), the reference y_i (`REFERENCE`), the task

prompt q_i (`TASK`), and the prediction $\hat{y}_i$ (`CANDIDATE`). The reference and prediction are aligned (cropped or resized) exactly as they are for the metrics (Sec. E.4), so the judge and the metrics grade the same pixels. The judge is Claude Sonnet 5 (Anthropic, 2026) with adaptive thinking at medium reasoning effort, pinned to a single model with no fallback; a reply that is refused, truncated, or unparseable is recorded as such and left unscored rather than retried on a different model.

The judge is blinded to the system that produced the candidate. It receives image data only, never file paths, and model and provider names are redacted from the task prompt. For the Planner protocol, q_i is the fixed Expert prompt for the same task and condition rather than the planner's own instruction: each planner writes in its own style, and grading against it would hold each editor to a different rubric and leak which planner, and hence which editor, produced the output.

The judge scores the image tasks of the Expert and Planner protocols. It does not score computer-generated holography or time-of-flight depth, whose outputs are phase maps and depth maps rather than viewable images; the Forward protocol, whose outputs are simulated measurements; or the two text-output tasks.

C.2 Dimensions

J returns five integers in $[0, 100]$ and a rationale of at most two sentences:

$$J(x_i, y_i, q_i, \hat{y}_i) = \big(S_i^{\mathrm{fid}},\, S_i^{\mathrm{task}},\, S_i^{\mathrm{col}},\, S_i^{\mathrm{art}},\, S_i\big).$$

The four sub-scores are defined in Table 3. Each is phrased as the absence of one kind of fault, with 100 meaning no fault of that kind. Three line up with classical metrics; task success has no classical counterpart. (The prompt keys it `restoration`; we call it task success because several judged tasks are reconstructions rather than restorations.)

Table 3: **Judge sub-scores.** Each is an integer in $[0, 100]$; 100 means no fault of that kind. The last column names the classical metric measuring the closest property; the correspondence is conceptual, not an equivalence.

Symbol	Dimension	Measures	Closest metric
S^{fid}	Structural fidelity	Structure, geometry, layout, and fine texture match the reference; nothing is invented, missing, moved, or duplicated.	SSIM, LPIPS
S^{task}	Task success	The degradation named in the task was removed, or the requested reconstruction was achieved. An output identical to the degraded input scores near 0.	none
S^{col}	Photometric fidelity	Color, brightness, contrast, and tone match the reference.	PSNR
S^{art}	Artifact-freeness	Freedom from defects the system introduced: blur, oversharpening, waxy smoothing, banding, blockiness, halos, ghosting, speckle.	NIQE

C.3 Overall score

The overall score S_i is the judge's holistic judgement of how well the prediction reproduces the reference, anchored by the six bands in the prompt (Sec. C.5). It is informed by the sub-scores but deliberately *not* a fixed function of them, so that a single decisive failure, such as the wrong scene or an untouched degradation, outweighs good marks elsewhere. The sub-scores are diagnostic: they show where a model fails (faithful structure but the task not done, or the task done with new artifacts). S_i is the judge's only input to the unified score, where the human-preference fit gives it weight 0.353, the largest of the nine signals (Table 5).

C.4 Validity checks

Because current Claude models do not accept a sampling temperature, we measure the judge's consistency instead of fixing it. On a task-stratified sample of already-judged rows we run three probes:

- **Test–retest**: the identical request is resubmitted, measuring the judge's own noise (exact agreement, mean $|\Delta S|$, Spearman ρ).

Table 4: **Judge validity probes, Expert protocol.** n: rows with a test–retest pair; *Retest* $|\Delta S|$: mean absolute change in the overall score when the identical request is resubmitted; *Reference* and *Input*: mean score when the reference, or the unrestored input, is submitted as the candidate. The input probe runs only where the input is a single image of the reference's kind; "–" marks multi-input tasks.

Task	n	Retest $\lvert\Delta S\rvert$	Reference	Input
Comp. Sensing (Block)	10	6.0	98.7	–
Demosaicking (Hexadeca)	11	1.8	98.2	12.0
Demosaicking (Quad)	11	3.5	97.8	14.9
Demosaicking (Regular)	11	3.2	97.8	6.6
Denoising	11	1.8	96.7	21.2
Event-based Intensity	11	1.0	97.4	5.7
High Dynamic Range	11	1.9	96.5	–
Inpainting	11	1.5	97.6	–
Lens Distortion	11	5.3	96.8	90.7
Lensless Imaging	12	0.5	96.8	–
Lightfield Extrapolation	11	3.7	96.8	–
Lightfield Interpolation	11	1.5	97.8	–
Longitudinal Chromatic	10	2.6	96.9	10.5
Motion Blur	11	3.9	97.2	10.9
Rand. Masked Subsampling	11	1.7	98.0	–
Spherical Aberration	11	1.5	97.2	12.2
Super-Resolution	11	1.5	91.6	78.6
White Balance	11	2.6	97.6	16.5
All tasks	197	2.5	97.1	25.5

- **Reference as candidate**: the reference itself is submitted as the prediction; the score should be near 100.
- **Input as candidate**: the degraded input is submitted as the prediction, for single-input tasks; the score should be low. This is the direct test of the property the classical metrics lack.

We ran the probes on 200 Expert rows sampled across the 18 judged tasks, 521 requests in all, of which 518 returned a parseable score. Table 4 gives the per-task results. The judge is consistent: resubmitting an identical request changes its score by 2.5 points on average (exact agreement 40%, Spearman $\rho = 0.988$, $n = 197$), and by at most 6.0 points on any task. It recognizes the reference, scoring it 97.1 on average (fifth percentile 96). Pooled over the 121 single-input rows, it gives the unrestored input 25.5, well inside the lower half of the scale that the prompt reserves for an output identical to the input.

Two tasks are exceptions, which we report as limitations. On *lens distortion* the judge scores the uncorrected input 90.7, at or above the model's output in 10 of 11 probes: the barrel distortion is real (the inputs average 19.5 dB against the reference) but barely visible at full-frame scale, so on this task correctness rests on the full-reference metrics. On *super-resolution* the "input" is the bicubic upsample, which the judge scores 78.6, at or above the model's output in 7 of 11 probes; this is also the only task where it scores the reference below 95. Without these two tasks the input scores 12.3 on average.

C.5 JUDGE PROMPT

The complete system prompt follows; the labelled images are sent in the user message as described above.

```
You are grading the output of an automated computational-imaging system.

You will see, in order:  the degraded INPUT image or images the system was given, the
REFERENCE image (the ground truth the output should reproduce), the TASK the system was
asked to perform, and the CANDIDATE output to grade.  The INPUT images may be measurements,
```

```
masks, exposure brackets, or neighbouring views rather than a degraded copy of the scene
itself - the TASK text explains their role.

Grade how well the CANDIDATE reproduces the REFERENCE for this task.  The reference is the
target:  content that looks plausible but differs from the reference is an error, not a
bonus.  Judge what is visible in the pixels.  The constraints listed in the TASK are part
of the rubric.

First give four sub-scores, each an integer from 0 to 100 (100 = no fault of this kind, 0 =
total failure of this kind):
- "fidelity":  structure, geometry, layout and fine texture match the reference; nothing is
invented, missing, moved or duplicated.
- "restoration":  the degradation named in the task was actually removed (or the requested
reconstruction was actually achieved).  An output identical to the degraded input scores
near 0 here.
- "color":  colour, brightness, contrast and tone match the reference.
- "artifacts":  freedom from defects the system introduced - blur, oversharpening, waxy
smoothing, banding, blockiness, halos, ghosting, speckle.  100 means no introduced defects.

Then give an overall "score", an integer from 0 to 100.  It is your holistic judgement of
how well the candidate reproduces the reference, informed by the sub-scores but not an
average of them:  one decisive failure (wrong scene, degradation untouched) outweighs good
marks elsewhere.
95-100  visually indistinguishable from the reference
80-94   faithful; only minor differences visible on close inspection
60-79   clearly the same scene and mostly correct, with noticeable residual degradation or
artefacts
40-59   partly correct; substantial errors, artefacts, or content that differs from the
reference
15-39   largely failed:  the degradation mostly remains, or much of the content is wrong
0-14    unrelated to the reference, blank, or unusable

Use the whole scale and its full resolution:  do not round to multiples of 5 or 10 - two
outputs of visibly different quality should get different scores.  An output identical to
the degraded input has removed nothing and belongs in the lower half.  Name the decisive
defects or strengths in a rationale of at most two sentences.

Respond with only a JSON object and nothing else - no markdown fences, no text before or
after:
{"fidelity":  <integer 0-100>, "restoration":  <integer 0-100>, "color":  <integer 0-100>,
"artifacts":  <integer 0-100>, "score":  <integer 0-100>, "rationale":  "<at most two
sentences>"}
```

D HUMAN PREFERENCE STUDY AND SCORE CALIBRATION

The unified score combines the eight metrics and the VLM judge (Sec. B). We fit their weights to blind pairwise human preferences; this section describes the study, the fit, and the criterion, fixed in advance, for adopting the fitted weights.

D.1 STUDY DESIGN

We drew 150 source images spread across the 18 image tasks the judge scores, and for each image formed 10 pairs of Expert outputs from two different editors: 7 *active* pairs, pairs of two above-median outputs whose scores under a preliminary metric-based score were closest, and 3 pairs drawn at random. A quarter of these 1,500 base pairs were collected from three raters, to measure agreement between raters; the rest from one. Two kinds of quality-control pairs were mixed in: 45 *golden* pairs in which one side is the reference itself, and 75 repeats of a base pair with the sides swapped.

Raters saw the task inputs, the reference and the two outputs side by side, with zoom and pan synchronized across all panels, and answered "Which output is the better restoration of the reference?" with left, right, tie or "both bad". The study was blind: images were served under opaque identifiers, the model behind each side was never shown, and the left/right placement was randomized for each rater and pair. Each rater's first 150 votes counted toward the study, so that no single rater could dominate it.

We had 28 raters casting 2,665 votes. On the 50 golden votes, raters chose the reference 64% of the time, called a tie on 24% and chose the model output on 10%. On the 76 repeat votes they gave the same answer as on the original pair 87% of the time; where both votes picked a side, they picked the same one 94% of the time.

Table 5: **Fitted unified-score weights.** Each entry is the weight w_k used in the unified score (Sec. B.1), followed by its 95% bootstrap interval over source images, which reports uncertainty only. SSIM fits to zero and is dropped.

Metric	Calibrated (95% CI)
PSNR	0.114 [0.076, 0.149]
SSIM	0.000 [0.000, 0.014]
LPIPS	0.163 [0.124, 0.202]
DISTS	0.144 [0.116, 0.173]
NIQE	0.080 [0.039, 0.119]
MUSIQ	0.053 [0.014, 0.091]
CLIP-IQA	0.086 [0.046, 0.123]
MANIQA	0.007 [0.000, 0.041]
VLM judge	0.353 [0.309, 0.402]

Table 6: **Held-out agreement with human votes.** Fraction of decisive held-out votes each scoring rule predicts, with 95% bootstrap intervals over source images. *Random-stratum* accuracy uses only the pairs drawn at random; *all-pairs* accuracy also includes the harder active pairs. Inter-rater agreement is the rate at which two raters of the same pair chose the same side, among votes that picked a side.

Scoring rule	Random-stratum accuracy	All-pairs accuracy
Calibrated (fit)	0.924 [0.900, 0.948]	0.848 [0.826, 0.870]
Equal weights	0.906 [0.877, 0.933]	0.823 [0.802, 0.843]
PSNR alone	0.835 [0.798, 0.869]	0.687 [0.660, 0.715]
SSIM alone	0.878 [0.841, 0.917]	0.634 [0.602, 0.666]
LPIPS alone	0.898 [0.870, 0.928]	0.781 [0.756, 0.803]
DISTS alone	0.902 [0.871, 0.931]	0.811 [0.786, 0.835]
NIQE alone	0.731 [0.681, 0.784]	0.655 [0.626, 0.683]
MUSIQ alone	0.636 [0.571, 0.698]	0.626 [0.588, 0.665]
CLIP-IQA alone	0.634 [0.570, 0.698]	0.631 [0.604, 0.660]
MANIQA alone	0.464 [0.403, 0.525]	0.533 [0.500, 0.569]
Judge alone	0.909 [0.879, 0.936]	0.836 [0.817, 0.856]
Inter-rater agreement	0.875 ($n = 656$)	

D.2 Fitting the weights

We model a vote for output A over output B with a Bradley–Terry model (Bradley & Terry, 1952) on metric differences,

$$P(A \succ B) = \sigma\Big(\textstyle\sum_k w_k\big(s_k(A) - s_k(B)\big)\Big), \qquad w_k \geq 0,$$

where s_k is metric k mapped to $[0, 1]$ by the same normalizers the unified score uses (Sec. B.1). The weights are fitted by minimizing the mean log-loss plus a small ridge penalty (10^{-3}) and then rescaled to sum to one. A vote for A counts as 1, a vote for B as 0 and a tie as $\frac{1}{2}$; "both bad" votes, golden pairs, repeats and votes past a rater's cap are excluded, which leaves 1,817 votes on 145 source images from all 28 raters.

D.3 Evaluation and the adoption criterion

Every reported accuracy is held out: the fit uses ten-fold cross-validation in which all votes on one source image fall in the same fold. The headline measure is accuracy on the *random* pairs only, because the active pairs were selected with a preliminary metric-based score. Intervals are 95% bootstrap intervals over source images (1,000 resamples).

Before fitting, we fixed the criterion for adopting the fitted weights: their held-out headline accuracy must beat equal weights over the same nine signals, with the paired bootstrap interval of the difference entirely above zero.

D.4 Results

Table 5 gives the fitted weights and Table 6 their held-out accuracy. The calibrated weights predict 92.4% of the random-pair votes against 90.6% for equal weights, a paired difference of +1.8 points (95% CI $[+0.5, +3.3]$, 139 source images), so the criterion above is met. On all pairs they reach 84.8%, against 82.3% for equal weights. When two raters judged the same pair and both picked a side, they picked the same one 87.5% of the time (656 comparisons on 274 pairs). Placement did not sway raters: 50.2% of decisive votes went to the left output.

The judge carries the largest weight, 0.353. Refitting without it lowers held-out accuracy from 92.4% to 89.8% on the random pairs and from 84.8% to 80.6% on all pairs, the largest drop from removing any single signal; removing any other one costs at most 0.7 points. Among the classical metrics, the learned full-reference metrics LPIPS and DISTS receive the most weight, and SSIM receives none (upper 95% bound 0.014). SSIM is largely redundant with LPIPS and DISTS: across the fitted votes, its per-pair score difference correlates with theirs at 0.72 and 0.55. Given the other

Table 7: Tasks, modalities, and instance counts (N). Image-task counts are nominal targets.

Task	In → Out	N
Ray and Wave Optics		
Lens System Design	T → T	5
Computer-Generated Holography	T, I → I	100
Image Signal Processing		
Demosaicking (Hexadeca)	T, I → I	100
Demosaicking (Quad)	T, I → I	100
Demosaicking (Regular)	T, I → I	100
High Dynamic Range Reconstruction	T, I → I	100
White Balance	T, I → I	100
Denoising	T, I → I	100
Calibration		
Lens Distortion Correction	T, I → I	100
Camera Calibration	T, I → T	54
Inverse Reconstruction		
Spherical Aberration Correction	T, I → I	100
Longitudinal Chromatic Aberration Correction	T, I → I	100
Motion Deblurring	T, I → I	100
Randomly Masked Subsampling	T, I → I	100
Block-Wise Compressive Sensing	T, I → I	100
Super-Resolution	T, I → I	100
Inpainting	T, I → I	100
Computational Sensing		
Lensless Imaging	T, I → I	100
Light-Field View Interpolation	T, I → I	100
Light-Field View Extrapolation	T, I → I	100
Event-Based Intensity Reconstruction	T, I → I	100
Time-of-Flight Depth Reconstruction	T, I → I	100

T = text; I = image(s). Valid scored pairs vary by task and model. Task formulations are in Sec. A.

signals, its unconstrained coefficient is negative, the only one of the nine, so the non-negative fit sets it to zero.

E Dataset Details

Table 7 lists the 22 subtasks with their input and output modalities and instance counts, and Table 8 maps each subtask to its data source. Where an established dataset matches the task, we take its measurements and ground truth as they are; otherwise we apply the task's forward operator to clean images drawn evenly from DIV2K and Adobe FiveK.

Table 8: **Data source per subtask.** *Real*: measurement and ground truth come from the source dataset. *Synth.*: we apply the task's forward operator to the source's clean images.

Source	Subtasks	Input
SIDD (Abdelhamed et al., 2018)	denoising	real
HDR+ (Hasinoff et al., 2016)	HDR reconstruction	real
WB Set-1 (Afifi et al., 2019)	white balance	real
GoPro (Nah et al., 2017)	motion deblurring	real
DLMD (Monakhova et al., 2019)	lensless imaging	real
DAVIS-color (Gao et al., 2025)	event-based intensity reconstruction	real
Under-display ToF (Qiao et al., 2022)	ToF depth	real
Stanford light fields (Stanford Computer Graphics Laboratory, 2016)	light-field view interpolation and extrapolation	real
MIT-CGH-4K (Shi et al., 2021)	CGH	real
Ours (Sony α7 IV captures)	lens distortion	real
DIV2K (Agustsson & Timofte, 2017) and FiveK (Bychkovsky et al., 2011)	demosaicking (regular, quad, hexadeca), spherical and longitudinal chromatic aberration, random-mask subsampling, block compressive sensing, super-resolution, inpainting; all degradation ablations	synth.
Ours (specifications)	lens design	–
Ours (checkerboard captures)	camera calibration	real

E.1 Scene and photometric statistics

We estimate each image's distribution over 13 scene categories (indoor, outdoor, street, landscape, city, portrait, people, night, text, food, animals, sports, close-up) and its brightness and contrast (mean and standard deviation of pixel intensity). No scene type dominates any subtask (Fig. 7), and brightness and contrast vary widely within and across subtasks (Fig. 8).

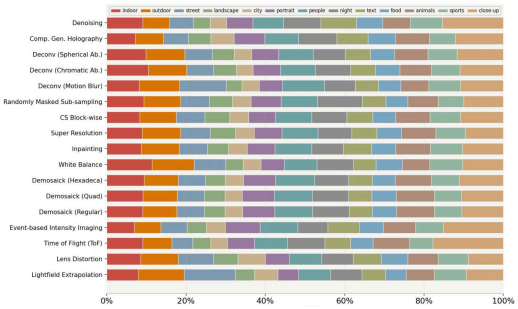


Figure 7: Semantic scene distribution per subtask: the average probability over 13 scene categories.

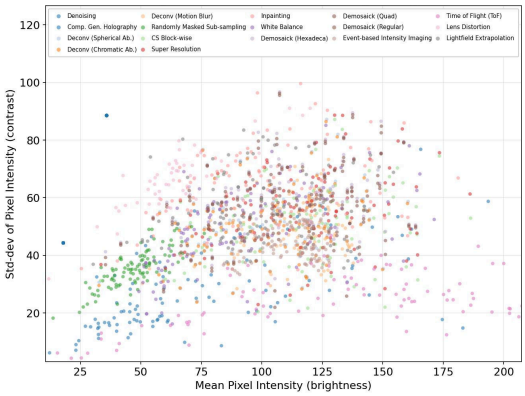


Figure 8: Brightness (mean pixel intensity) against contrast (pixel-intensity standard deviation), one point per sample, coloured by subtask.

E.2 Lens design

Each lens-design instance is a target specification (Wang et al., 2022; Yang et al., 2024; Chang et al., 2018): half-diagonal field of view (HFOV), F-number, sensor diagonal, six elements, a fixed material list, and aspheric surfaces with one conic constant and six aspheric coefficients each. We vary HFOV, F-number and sensor size together over five targets, from wide and fast to narrow and slow (Table 25). AutoLens and the models share this search space. The prompt, scoring, and results are in Sec. L.2.

E.3 Camera calibration

The calibration set has 54 instances, 18 each for an iPhone 13 Pro, a Sony $\alpha 7$, and a stereo camera. Each instance supplies four checkerboard views, 216 images in all. The prompt, scoring, and results are in Sec. L.1.

E.4 Preprocessing and alignment

Every input is standardized to a 1024×1024 canvas, the most restrictive output setting among the evaluated APIs. Real noisy data (SIDD) use aligned random 1024×1024 crops of input and ground truth, which preserve the native noise. Small sensing inputs (holography phase maps, lensless measurements) are centre-padded, and a pad mask restricts every metric and the judge to the valid region. Other inputs are resized to the canvas; for synthesized subtasks this happens before the degradation is applied.

Two targets are not on the padded canvas: the 346×260 event frames and the 128×128 measurement of Forward block compressive sensing. For these we resize the prediction to the reference size, with area interpolation when shrinking and bicubic when enlarging, including when the aspect ratio differs. Padded targets are centre-cropped to their valid region, including Forward holography (384×384). The metrics and the judge see identical pixels.

The canvas also avoids provider-side resizing. Gemini rescales to the nearest of a set of roughly one-megapixel buckets (e.g. 742×675 becomes 1088×976) but leaves 1024×1024 unchanged, and GPT-Image returns only 1024×1024, 1024×1536 or 1536×1024. Qwen-Image-Edit shows a slight zoom at some input sizes, which was small at 1024×1024.

E.5 Model access and inference settings

Table 9 lists how each editor and planner was accessed and run.

Table 9: **Model access and inference settings.**

Model	Access and settings
Gemini-3.1-Flash-Image	Google Vertex AI, checkpoint of 2026-02-26; default settings (Google DeepMind, 2026a)
GPT-Image-1.5	Azure AI Foundry, version 2025-12-16; default settings (OpenAI, 2026a)
GPT-Image-2	Azure AI Foundry; default settings (OpenAI, 2026b)
FLUX.2-Pro	fal.ai hosted API, 1024×1024 output (Black Forest Labs, 2025)
Qwen-Image-Edit-2511	local, one NVIDIA H200; official diffusers implementation, 28 steps, other parameters default (Wu et al., 2025)
HunyuanImage-3.0	local, one NVIDIA H200; the 8-step distilled Instruct model (Cao et al., 2025) in a community INT8 quantization (Rollei, 2026)
Gemini 3.1 Pro Preview	planner for Gemini-3.1-Flash-Image and FLUX.2-Pro (Google DeepMind, 2026b)
GPT-5	planner for GPT-Image-1.5, GPT-Image-2 and HunyuanImage-3.0 (OpenAI, 2025)
Qwen3.5-35B-A3B	planner for Qwen-Image-Edit-2511 (Qwen Team, 2026)

Table 10: **Modeled API cost** per scored image (USD), with the list prices applied (USD per million tokens, input/output). Forward covers five tasks, so its cost is not comparable with the other two columns.

Editor	Prices applied (planner; editor)	Expert	Planner	Forward
Gemini-3.1-Flash-Image	2/12; 0.50/60	0.066	0.072	0.069
GPT-Image-2	1.25/10; 8/30	0.062	0.075	0.090
FLUX.2-Pro	2/12; per megapixel	0.057	0.063	0.059
GPT-Image-1.5	1.25/10; 8/32	0.046	0.060	0.075
Qwen-Image-Edit-2511	local	0	0	0
HunyuanImage-3.0	1.25/10; local	0	0.012	0.026

F Benchmark Cost Analysis

The cost axis of the main paper's cost–performance figure is a modeled API cost per scored image: the mean logged tokens per call, priced at list rates as of September 18, 2026 (Table 10), summed over a pipeline's calls and divided by the number of scored images. These are modeled API costs, not measured spending, and are upper bounds on two counts. Our logs record one input and one output token count per call, so each image editor's count is billed at its image rate, the higher one; and cached-input discounts are not applied.

FLUX.2-Pro is billed by fal.ai per megapixel rather than per token, so it is priced from the benchmark images' actual sizes: \$0.03 for the first megapixel of its 1024×1024 output plus \$0.015 for each further megapixel of input, each image rounded up to a whole megapixel. When an editor executes another model's plans, the planner calls are charged to that editor's pipeline: FLUX.2-Pro at the Gemini 3.1 Pro rate, HunyuanImage-3.0 at the GPT-5 rate. Qwen-Image-Edit-2511, its Qwen3.5 planner, and HunyuanImage-3.0 run on local GPUs and carry no API charge.

G Degradation Ablation Details

The five sweeps of the main paper's ablation figure vary one degradation at a time on clean DIV2K and Adobe FiveK images, under the Expert protocol. *Sub-sampling* retains 10–80% of pixels (the random-mask subsampling task). *Spherical aberration* convolves with a PSF from the Zernike mode Z_4^0 (Wu et al., 2019) at $\lambda = 550$ nm with coefficient $c_{\mathrm{sph}} \in \{0.10, 0.15, 0.20, 0.25\}\lambda$ (the spherical-aberration task). The three noise sweeps use a Poisson–Gaussian model (Wei et al., 2020) at ten levels each: *shot* noise lowers the photon count p from 1000 to 5 at read noise $\sigma = 0.216$, *read* noise raises σ from 0.039 to 0.392 at $p = 70.7$, and *mixed* noise varies both.

Tables 11–15 give the unified score and PSNR per model and condition. The judge was not run on these sweeps, so the unified score here uses the calibrated weights over the eight metrics, renormalized, and is not comparable with the main leaderboard. Each mean is over the instances the model returned.

Table 11: **Sub-sampling**, Expert protocol: unified score (0–100, eight metrics, no judge) and PSNR (dB) per condition. Condition: share of pixels retained. Each mean covers the 86–100 scored instances per model and condition.

Condition	Gemini-3.1-Flash-Image		GPT-Image-2		FLUX.2-Pro		GPT-Image-1.5		Qwen-Image-Edit-2511		HunyuanImage-3.0	
	Score	PSNR	Score	PSNR	Score	PSNR	Score	PSNR	Score	PSNR	Score	PSNR
10%	44.8	8.9	50.8	11.0	35.2	9.3	47.3	10.4	21.1	6.4	36.6	7.6
20%	54.3	11.0	55.6	12.7	46.8	12.2	51.1	11.4	29.4	8.1	38.2	7.7
40%	60.5	13.5	60.0	15.2	54.4	15.7	55.8	13.8	30.1	11.1	39.9	7.7
60%	65.5	16.7	63.9	17.3	58.7	17.7	57.8	14.5	27.6	10.0	40.8	7.7
80%	69.4	19.7	67.7	19.6	64.6	19.9	58.9	14.9	28.8	11.4	40.9	7.7

Table 12: **Spherical aberration**, Expert protocol: unified score (0–100, eight metrics, no judge) and PSNR (dB) per condition. Condition: spherical-aberration coefficient c_{sph}. Each mean covers the 89–100 scored instances per model and condition.

Condition	Gemini-3.1-Flash-Image		GPT-Image-2		FLUX.2-Pro		GPT-Image-1.5		Qwen-Image-Edit-2511		HunyuanImage-3.0	
	Score	PSNR	Score	PSNR	Score	PSNR	Score	PSNR	Score	PSNR	Score	PSNR
0.10λ	75.0	23.5	68.9	19.0	70.0	19.2	60.2	14.3	61.8	20.7	48.2	8.9
0.15λ	73.5	22.9	68.3	19.2	69.4	19.7	60.1	14.1	58.0	20.3	44.8	8.4
0.20λ	72.2	22.6	66.7	18.8	67.8	19.6	59.5	14.3	52.9	19.9	40.4	7.2
0.25λ	71.0	21.8	66.3	18.9	65.9	19.1	58.7	14.2	50.5	19.2	37.6	6.5

H Overall Score: Category Breakdown

Table 16 breaks Overall down by task category. Overall is the flat mean over the twenty image tasks, not the mean of the category columns, because the categories are uneven: Ray and Wave Optics and Calibration hold one image task each (holography and lens distortion), against seven for inverse reconstruction. Averaging the columns instead would give each of those two tasks a fifth of the score, and FLUX.2-Pro's lead on them alone would lift it above GPT-Image-2 (0.510 against 0.466), although GPT-Image-2 is ahead on fourteen of the other eighteen tasks. All other places are the same under either weighting.

I Per-Task Unified Scores

Table 17 gives the per-task unified score, the score behind the leaderboard and the radar figures, with each model's coverage.

J Per-Task Results for Every Metric

Tables 18–20 give the per-task means of every signal the unified score combines: the four full-reference metrics, the four no-reference metrics, and the judge with its four sub-scores. All are means over the images each model returned. They support the main paper's observations that the no-reference metrics rank HunyuanImage-3.0 near the top despite its last place on every full-reference metric and on the judge, and that GPT-Image-2's Planner demosaicking outputs lose task success and photometric fidelity but not structure or PSNR.

K Robustness Checks: Refusals and the Scoring Denominator

Hosted editors refuse some inputs and local models sometimes fail. The benchmark scores a missing job 0, because a model that returns no image has not solved the task and dropping the attempt would

Table 13: **Shot noise**, Expert protocol: unified score (0–100, eight metrics, no judge) and PSNR (dB) per condition. Condition: photon count p, read noise fixed at $\sigma = 0.216$. Each mean covers the 17–20 scored instances per model and condition.

Condition	Gemini-3.1-Flash-Image		GPT-Image-2		FLUX.2-Pro		GPT-Image-1.5		Qwen-Image-Edit-2511		HunyuanImage-3.0	
	Score	PSNR	Score	PSNR	Score	PSNR	Score	PSNR	Score	PSNR	Score	PSNR
$p = 1000$	71.4	22.5	69.2	22.0	64.9	20.3	58.5	15.0	57.9	18.5	29.9	5.4
$p = 555$	70.7	22.2	68.9	21.9	63.6	20.3	58.6	15.0	57.3	19.0	29.5	5.5
$p = 308$	70.4	21.8	68.7	21.8	64.3	20.1	58.7	15.2	58.0	19.2	29.1	5.5
$p = 171$	70.9	21.9	69.4	22.2	62.7	19.7	58.7	15.0	57.3	18.9	28.6	5.5
$p = 95$	70.8	22.1	69.0	21.6	63.4	19.9	58.2	14.9	58.2	18.6	28.7	5.3
$p = 53$	70.8	22.1	68.5	21.5	66.1	20.1	58.7	15.0	57.1	18.8	28.0	5.2
$p = 29$	70.3	21.6	67.7	21.7	61.8	19.1	58.8	14.9	53.8	18.1	28.8	5.3
$p = 16$	70.4	21.5	67.6	21.4	59.9	18.9	57.7	15.1	51.6	16.4	29.6	5.4
$p = 9.01$	68.4	20.0	66.2	21.0	59.5	19.1	58.2	14.9	50.1	15.9	28.2	5.4
$p = 5$	67.2	19.2	65.0	20.4	59.2	19.1	57.1	14.9	47.3	14.4	29.6	5.4

Table 14: **Read noise**, Expert protocol: unified score (0–100, eight metrics, no judge) and PSNR (dB) per condition. Condition: read-noise level σ, photon count fixed at $p = 70.7$. Each mean covers the 17–20 scored instances per model and condition.

Condition	Gemini-3.1-Flash-Image		GPT-Image-2		FLUX.2-Pro		GPT-Image-1.5		Qwen-Image-Edit-2511		HunyuanImage-3.0	
	Score	PSNR	Score	PSNR	Score	PSNR	Score	PSNR	Score	PSNR	Score	PSNR
$\sigma = 0.039$	76.1	25.0	73.3	24.0	69.4	21.2	58.9	15.4	67.8	21.1	33.9	6.1
$\sigma = 0.078$	74.8	24.7	72.5	23.6	68.1	21.1	59.0	15.1	67.0	20.8	33.1	5.6
$\sigma = 0.118$	74.2	23.9	71.9	23.9	67.8	21.2	58.4	15.5	64.3	20.4	32.5	5.6
$\sigma = 0.157$	72.6	23.5	71.1	23.0	66.7	20.9	58.9	15.0	61.5	20.1	28.3	5.3
$\sigma = 0.196$	71.5	22.5	69.4	22.4	64.5	20.3	58.5	15.3	58.9	19.3	29.7	5.4
$\sigma = 0.235$	69.6	21.5	67.7	21.5	62.1	19.7	59.0	15.1	53.9	17.7	28.0	5.3
$\sigma = 0.275$	69.3	21.3	66.3	21.1	61.4	19.3	58.1	15.3	52.2	16.5	30.6	5.5
$\sigma = 0.314$	67.6	19.7	66.4	20.8	58.8	19.0	58.3	15.1	49.3	15.3	29.2	5.4
$\sigma = 0.353$	67.2	19.9	64.4	19.9	58.7	18.3	58.4	14.6	49.0	15.2	31.6	5.5
$\sigma = 0.392$	65.7	19.1	63.5	19.8	54.9	18.2	58.1	15.0	47.3	15.3	29.7	5.6

Table 15: **Mixed noise**, Expert protocol: unified score (0–100, eight metrics, no judge) and PSNR (dB) per condition. Condition: photon count p and read-noise level σ, both varied. Each mean covers the 17–20 scored instances per model and condition.

Condition	Gemini-3.1-Flash-Image		GPT-Image-2		FLUX.2-Pro		GPT-Image-1.5		Qwen-Image-Edit-2511		HunyuanImage-3.0	
	Score	PSNR	Score	PSNR	Score	PSNR	Score	PSNR	Score	PSNR	Score	PSNR
$p = 1000,\ \sigma = 0.039$	78.1	26.9	75.2	24.8	70.5	21.8	60.1	15.6	69.4	21.6	34.1	6.1
$p = 555,\ \sigma = 0.078$	75.2	24.8	73.9	24.3	70.8	22.1	59.2	15.3	68.1	21.9	33.7	6.0
$p = 308,\ \sigma = 0.118$	74.1	24.1	72.4	23.9	68.4	21.0	58.8	15.1	65.4	21.5	32.5	5.8
$p = 171,\ \sigma = 0.157$	72.6	22.9	71.0	22.8	65.5	19.8	58.9	15.2	62.8	20.9	30.1	5.5
$p = 95,\ \sigma = 0.196$	71.6	22.6	69.3	22.2	63.8	20.3	58.2	15.2	60.4	20.1	29.2	5.4
$p = 53,\ \sigma = 0.235$	69.7	21.5	68.6	21.7	62.8	19.3	58.6	15.0	53.3	16.6	27.0	5.2
$p = 29,\ \sigma = 0.275$	68.4	20.1	66.8	21.1	62.2	19.0	58.6	15.1	53.6	18.1	28.1	5.3
$p = 16,\ \sigma = 0.314$	66.6	19.0	66.4	20.4	58.3	17.7	58.3	15.4	49.2	15.6	31.5	5.6
$p = 9.01,\ \sigma = 0.353$	66.0	19.1	62.9	19.2	57.1	18.2	57.5	14.6	46.7	13.9	33.0	5.7
$p = 5,\ \sigma = 0.392$	63.3	17.6	61.9	18.5	53.0	17.4	55.0	14.7	42.8	12.0	32.6	5.7

Table 16: Per-category ImagingBench scores under the Expert protocol: the mean unified score over each category's image tasks. Optics holds one task (holography) and Calibration one (lens distortion). Overall is the mean over all twenty image tasks ($1/|\mathcal{T}|$ each), not the mean of the category columns. The two text-only tasks are scored separately (Table 23) because a text planner, not the image model, solves them.

Model	Optics	ISP	Inv. Recon.	Sensing	Calib.	Overall
Gemini-3.1-Flash-Image	0.100	0.677	0.699	0.414	0.760	0.594
GPT-Image-2	0.105	0.662	0.621	0.417	0.524	0.552
FLUX.2-Pro	0.229	0.594	0.610	0.410	0.708	0.541
GPT-Image-1.5	0.213	0.553	0.546	0.342	0.496	0.478
Qwen-Image-Edit-2511	0.391	0.390	0.454	0.294	0.499	0.394
HunyuanImage-3.0	0.156	0.222	0.288	0.229	0.354	0.250

reward refusing a hard input. Here we check how many jobs go missing, whether the missing ones are easier or harder than the rest, and whether the treatment of missing jobs changes the ranking. We use the Expert protocol, where every model is asked for the same $2{,}900$ jobs.

K.1 How many jobs go missing, and why

Table 21 (left) gives per-model coverage. Three editors are refused by a content filter: 205 of GPT-Image-1.5's calls and 197 of GPT-Image-2's (coverage 92.6% and 93.2%), and 47 of FLUX.2-Pro's (98.2%). The GPT refusals are uneven, from none on holography and lensless imaging to 12–14% on super-resolution, inpainting and block compressive sensing. Qwen-Image-Edit completed every job, and Gemini's 40 misses (1.4%) are transport and parsing failures, not refusals. HunyuanImage-3.0 has the largest gap (86.5% coverage) and none of it is a refusal: inputs of three or more images exceed its context limit, which removes all 100 time-of-flight jobs, 91 HDR jobs, and 200 block compressive-sensing jobs.

K.2 Are the refused images easier or harder?

Qwen-Image-Edit completed every job, so its per-image unified score ranks each task's images by difficulty independently of the refusing editors. For each task we compare this index on the images an editor was refused on against the images it completed (Table 22); because task score scales differ severalfold, the pooled rows standardize within each task.

Pooled over tasks, the refused images are neither easier nor harder: -0.10 SD for GPT-Image-1.5 (95% CI $[-0.23, +0.04]$, $p = 0.34$), -0.10 SD for GPT-Image-2 ($[-0.24, +0.04]$, $p = 0.38$), and $+0.06$ SD for FLUX.2-Pro ($[-0.20, +0.32]$, $p = 0.70$). The one consistent per-task effect is super-resolution, where both GPT editors were refused on images the reference model finds easier ($\Delta = +0.09$, $p < 0.001$ for each).

K.3 Does the treatment change the ranking?

Table 21 (right) rescores every model with a missing job **Drop**ped from the mean; scored **Zero**, the benchmark's policy; **Imp**uted from a per-task linear fit on the reference model; or given an **Upper** bound that assumes the model would have ranked each refused image where the reference model ranks it. **Com** rescores all six models on the 19 tasks every model completed at least in part. Imp and Upper stay within 0.001 of Drop: the refused images are typical on the reference index, and the reference model predicts another model's per-image score only weakly (median per-task correlation 0.20).

No treatment reorders the leaderboard: the six models keep their places under every column. The closest pair, GPT-Image-2 and FLUX.2-Pro, stays in order; its gap is 0.011 under Zero and 0.036 under Drop, because GPT-Image-2 is refused four times as often. The Zero policy costs the GPT editors about 0.035 each and HunyuanImage-3.0 the most (0.291 to 0.250). Because a score alone does not show its denominator, we report per-task coverage next to the per-task unified scores (Table 17).

L TEXT-OUTPUT TASKS: CAMERA CALIBRATION AND LENS DESIGN

Camera calibration and lens design return text rather than an image. Each provider's planner (Gemini 3.1 Pro, GPT-5, Qwen3.5-35B-A3B) answers them zero-shot in a single request, with no tool use; for lens design the OpenAI row is GPT-5.4, run through ChatGPT. We analyse the returned prescriptions ourselves afterwards. Neither task has an image to score, so each has its own normalized score and neither enters Overall (Table 23).

L.1 CAMERA CALIBRATION

The model receives four checkerboard views and the camera's metadata and returns intrinsics (f_x, f_y, c_x, c_y) and distortion coefficients as JSON (prompt below). For each group we compute the mean relative error (MRE) against reference calibrations, $\mathrm{MRE} = \frac{1}{|P|}\sum_{p\in P} |\hat{p} - p|/|p|$, and the instance score is the mean of $\mathrm{clip}(1 - \mathrm{MRE}, 0, 1)$ over the intrinsics and the distortion coefficients; an unparseable reply scores 0.

In practice the score measures the intrinsics alone. The predicted distortion coefficients are typically off by more than their own magnitude, so the distortion term is nearly always clipped to 0 (its mean is below 0.015 for all three models) and the score is capped near 0.5. Table 24 breaks the intrinsics error down by camera. Gemini 3.1 Pro recovers the iPhone intrinsics to within 2%, while its Sony error of 77% is the largest in the table; GPT-5 and Qwen3.5 are more even across cameras.

Camera-calibration prompt

```
Task: Estimate camera calibration parameters from the provided checkerboard image set.
Input: A subset of calibration images (monocular camera) or paired left/right stereo
calibration images.

Context metadata: {CAMERA_METADATA}

Calibration target specification: 8x11 checkerboard; checker size 20 mm; marker size 15
mm; marker dictionary ArUco DICT_5X5.

Requirements:
- Use only visual evidence from the provided images and the metadata above.
- Estimate per-camera intrinsics: fx, fy, cx, cy.
- Estimate per-camera distortion coefficients: k1, k2, k3, k4, k5, k6, p1, p2, s1, s2, s3,
s4.
- If the input is stereo, also estimate inter-camera extrinsics as rotation vector (rx, ry,
rz) and translation (x, y, z).
- If the input is monocular, do not output stereo extrinsics.

Output format (strict): return valid JSON only (no markdown, no prose, no code fences),
with a "cameras" list of {"id", "intrinsics": {fx, fy, cx, cy}, "distortion": {k1 ...
s4}} and, for stereo input only, "stereo_extrinsics": {"rotation": {rx, ry, rz},
"translation": {x, y, z}}. Output numeric values as JSON numbers.
```

L.2 LENS DESIGN

The model receives a target specification and returns a six-element prescription in Zemax sequential syntax (the prompt is in Sec. M.1). We analyse each prescription in Zemax and score it by its peak on-axis Strehl ratio, $\mathrm{clip}(\mathrm{Strehl}/0.8, 0, 1)$ against the diffraction limit; a prescription that cannot be analysed scores 0. We compare against AutoLens (Yang et al., 2024) in the same aspheric search space.

All designs are far from diffraction-limited. The raw peak Strehl ratio averaged over the five targets (Table 25) is 0.023 for AutoLens, which leads on four targets, against 0.014 for GPT-5.4 and 0.013 for Gemini; Qwen returns an analysable prescription for one target only (0.005). The models' layouts are physically valid and meet the basic specification, but their spot diagrams are broader, with more chromatic spread and weaker off-axis correction, and their MTF curves lower (Figs. 9–12). A one-shot design from a general model is at best a starting point for iterative optimization.

Table 17: **Per-task unified scores and coverage.** Mean per-instance unified score (metrics and VLM judge weighted by the human-preference fit, Sec. D; CGH and ToF depth use the four full-reference metrics, unweighted; higher is better, rescaled to $[0, 100]$). **Cov.**: percentage of jobs with a usable image; a missing job scores 0. Cells stack Expert and Planner (E/P) top to bottom. Models are ordered by Overall. **Forward Process**: the model is given the clean image and asked to *produce* the measurement. **Neural Network**: a specialized method per task, cited below the table; –: no pretrained model exists for our synthesized data. **Overall**: the mean over the protocol's image tasks, per half. The two text-output tasks are reported separately (Table 23).

		Proprietary								*Open-weight*				*Reference*
Task	**Phase**	**Gemini-3.1-Flash-Image**		**GPT-Image-2**		**FLUX.2-Pro**		**GPT-Image-1.5**		**Qwen-Image-Edit-2511**		**HunyuanImage-3.0**		**Neural Network**
		Score↑	Cov.↑	Score↑	Cov.↑	Score↑	Cov.↑	Score↑	Cov.↑	Score↑	Cov.↑	Score↑	Cov.↑	Score↑
Inverse Process														
Ray and Wave Optics														
Holography (CGH)	E	10.0	100	10.5	100	22.9	99	21.3	100	39.1	100	15.6	100	64.9
	P	4.8	98	7.0	100	21.7	99	14.5	100	40.1	100	7.9	100	
Image Signal Processing														
Demosaic (Hexa)	E	65.9	98	61.6	93	55.5	94	52.5	93	28.0	100	25.2	100	–
	P	60.1	95	45.3	88	49.1	96	45.8	90	29.0	88	27.1	100	
Demosaic (Quad)	E	65.1	98	62.5	93	59.1	97	50.7	89	35.4	100	24.3	100	91.2
	P	60.8	94	46.1	89	52.6	96	46.3	93	33.5	93	29.6	100	
Demosaic (Reg.)	E	65.1	99	64.5	92	57.2	95	54.8	92	25.1	100	24.8	100	97.6
	P	62.7	98	52.9	92	53.2	95	54.7	94	25.1	98	29.7	100	
HDR	E	72.8	99	71.5	99	67.2	100	65.8	100	43.4	100	3.4	9	–
	P	73.2	100	71.9	100	66.5	100	64.2	100	39.7	92	4.6	9	
White Balance	E	69.0	97	69.5	96	63.4	100	53.5	95	55.6	100	28.5	100	66.5
	P	71.3	100	69.7	96	63.2	98	52.3	96	52.5	100	35.5	100	
Denoising	E	68.2	100	67.8	99	53.8	99	54.3	99	46.6	100	27.1	100	69.3
	P	65.3	98	67.6	99	55.9	99	53.5	99	48.6	100	22.8	99	
Inverse Reconstruction														
Spherical Aberr.	E	74.2	97	65.7	92	70.6	98	57.0	90	54.2	100	33.7	100	–
	P	73.0	96	66.9	92	70.1	98	57.8	92	43.9	100	29.8	100	
Chromatic Aberr.	E	73.1	99	67.6	92	65.9	97	57.2	92	52.3	100	19.7	100	–
	P	70.6	97	66.3	91	65.4	97	56.3	92	43.4	98	33.9	100	
Motion Blur	E	66.1	100	57.3	95	62.1	100	51.2	93	38.0	100	33.9	100	77.5
	P	63.6	98	58.8	94	60.6	100	54.0	96	36.5	100	35.6	100	
Masked Sampling	E	57.5	99	53.2	89	52.6	98	48.8	88	21.7	100	25.8	100	–
	P	56.2	99	51.4	89	51.1	98	46.0	89	23.4	97	27.2	100	
Comp. Sensing	E	72.5	98	58.0	86	57.2	98	56.9	88	47.9	100	9.5	33	–
	P	71.5	97	57.7	87	56.2	98	54.5	87	39.5	79	12.7	33	
Super-Resolution	E	73.7	97	63.2	87	62.3	99	57.2	87	67.0	100	52.3	100	54.3
	P	75.1	99	62.0	87	62.4	99	55.3	88	68.7	100	46.8	100	
Inpainting	E	72.2	94	69.5	87	56.3	97	53.6	86	36.8	100	26.5	100	85.3
	P	72.2	95	68.5	87	58.2	98	51.9	87	38.0	89	31.4	100	
Computational Sensing														
Lensless Imaging	E	24.3	100	23.4	100	19.7	100	22.1	100	17.8	100	29.5	100	55.4
	P	20.6	100	16.2	100	19.3	100	18.0	99	1.6	8	28.1	100	
LF Extrapolation	E	48.1	100	52.0	99	53.5	100	37.2	97	35.3	100	25.6	100	–
	P	47.9	100	52.7	100	52.6	100	46.3	100	22.2	48	42.1	100	
LF Interpolation	E	60.3	100	60.5	100	62.5	100	50.7	97	42.5	100	26.5	100	–
	P	60.1	100	59.7	98	61.2	99	54.0	99	33.1	70	35.7	100	
Event Recon.	E	34.4	100	36.1	100	33.6	100	32.7	98	22.2	100	32.8	100	–
	P	36.2	99	37.1	97	35.6	100	33.5	98	21.1	96	31.1	100	
ToF Depth	E	40.0	98	36.7	99	35.5	94	28.2	99	29.3	100	–	–	–
	P	39.3	99	37.6	99	34.0	93	27.5	98	29.1	100			
Calibration														
Lens Distortion	E	76.0	100	52.4	100	70.8	99	49.6	98	49.9	100	35.4	100	–
	P	70.9	94	52.0	99	70.6	99	51.4	100	59.4	100	31.8	100	
Leaderboard														
Overall	E	59.4		55.2		54.1		47.8		39.4		25.0		–
	P	57.8		52.4		53.0		46.9		36.4		27.2		
Forward Process														
Holography (CGH)	F	20.8	100	23.2	99	16.3	100	15.6	99	24.3	100	20.8	100	–
HDR	F	54.9	100	48.2	100	34.3	99	41.9	100	44.1	100	30.1	100	–
Denoising	F	43.7	99	54.4	100	28.9	100	35.1	97	32.8	100	19.1	100	–
Comp. Sensing	F	18.7	31	35.6	89	50.3	99	22.3	96	54.8	99	32.7	100	–
Lensless Imaging	F	12.4	88	19.2	98	13.2	96	25.4	98	21.5	100	17.8	100	–
Leaderboard														
Overall	F	30.1		36.1		28.6		28.1		35.5		24.1		–

Neural Network references: CGH (Shi et al., 2021), Dm-Q (A Sharif et al., 2021), Dm-R (A Sharif et al., 2021), WB (Afifi & Brown, 2020), DN (Zhang et al., 2018a), MB (Tsai et al., 2022), SR (Wang et al., 2018), IP (Suvorov et al., 2022), LL (Monakhova et al., 2019).

Table 18: **Per-task full-reference metrics.** Mean PSNR, SSIM, LPIPS, and DISTS per task over the images each model returned. Expert/Planner (E/P) values are stacked top/bottom; Forward covers five tasks. Neural Network denotes specialized references (citations below), which report PSNR, SSIM, and LPIPS only; – marks unavailable entries.

		Proprietary																*Open-weight*								*Reference*		
Task	**Phase**	**Gemini-3.1-Flash-Image**				**GPT-Image-2**				**FLUX.2-Pro**				**GPT-Image-1.5**				**Qwen-Image-Edit-2511**				**HunyuanImage-3.0**				**Neural Network**		
		PSNR↑	SSIM↑	LPIPS↓	DISTS↓	PSNR↑	SSIM↑	LPIPS↓	DISTS↓	PSNR↑	SSIM↑	LPIPS↓	DISTS↓	PSNR↑	SSIM↑	LPIPS↓	DISTS↓	PSNR↑	SSIM↑	LPIPS↓	DISTS↓	PSNR↑	SSIM↑	LPIPS↓	DISTS↓	PSNR↑	SSIM↑	LPIPS↓
Inverse Process																												
Ray and Wave Optics																												
Holography (CGH)	E	6.74	0.06	1.12	0.70	7.72	0.00	1.02	0.60	10.30	0.28	0.77	0.66	9.59	0.28	0.85	0.60	17.31	0.39	0.36	0.56	4.45	0.00	0.71	0.68			
	P	5.61	0.01	1.12	0.81	7.72	0.01	1.04	0.74	9.04	0.27	0.82	0.68	8.19	0.19	0.91	0.71	17.33	0.34	0.27	0.55	4.39	0.01	0.97	0.73	25.62	0.77	0.25
Image Signal Processing																												
Demosaic (Hexa)	E	18.84	0.67	0.28	0.13	18.51	0.66	0.29	0.15	17.47	0.70	0.36	0.16	13.68	0.36	0.50	0.19	12.20	0.27	0.75	0.42	6.14	0.32	0.84	0.48	–	–	–
	P	18.88	0.67	0.29	0.14	18.69	0.65	0.34	0.17	18.09	0.74	0.35	0.18	13.81	0.34	0.54	0.22	14.52	0.36	0.65	0.35	6.00	0.28	0.82	0.44			
Demosaic (Quad)	E	18.93	0.68	0.28	0.13	18.77	0.66	0.28	0.14	17.87	0.67	0.35	0.16	13.84	0.36	0.48	0.18	14.74	0.41	0.65	0.33	6.03	0.33	0.86	0.49			
	P	18.64	0.66	0.30	0.14	18.91	0.64	0.35	0.18	18.19	0.71	0.35	0.17	13.97	0.34	0.52	0.22	15.02	0.40	0.62	0.31	6.69	0.35	0.76	0.39	34.70	0.95	0.04
Demosaic (Reg.)	E	18.85	0.68	0.28	0.13	19.55	0.74	0.23	0.12	17.32	0.74	0.32	0.15	14.19	0.38	0.45	0.17	13.30	0.26	0.79	0.51	6.15	0.36	0.85	0.50			
	P	18.74	0.67	0.29	0.14	19.28	0.71	0.29	0.15	18.20	0.75	0.32	0.16	14.15	0.37	0.46	0.18	13.48	0.27	0.78	0.50	6.86	0.31	0.77	0.38	38.72	0.97	0.02
HDR	E	18.14	0.57	0.23	0.12	17.18	0.51	0.26	0.14	16.52	0.55	0.29	0.14	15.73	0.44	0.39	0.17	13.68	0.41	0.48	0.26	10.95	0.35	0.61	0.26	–	–	–
	P	17.70	0.57	0.22	0.12	16.90	0.51	0.26	0.14	16.48	0.54	0.30	0.15	15.24	0.43	0.41	0.17	13.41	0.40	0.48	0.27	13.41	0.49	0.41	0.22			
White Balance	E	21.11	0.84	0.18	0.09	20.61	0.79	0.21	0.12	15.42	0.69	0.29	0.15	12.63	0.45	0.45	0.16	17.09	0.74	0.31	0.16	5.88	0.45	0.71	0.46			
	P	22.07	0.85	0.17	0.09	21.82	0.82	0.19	0.11	16.10	0.72	0.27	0.14	12.51	0.46	0.46	0.17	17.15	0.73	0.31	0.15	6.40	0.48	0.63	0.35	25.17	0.80	0.28
Denoising	E	25.43	0.79	0.31	0.20	26.35	0.86	0.24	0.18	16.19	0.51	0.54	0.29	18.00	0.52	0.55	0.28	20.19	0.69	0.47	0.30	4.34	0.41	0.85	0.48			
	P	23.82	0.78	0.31	0.21	26.21	0.82	0.25	0.20	17.81	0.58	0.49	0.28	17.98	0.54	0.54	0.29	20.92	0.67	0.44	0.29	5.40	0.37	0.89	0.48	32.84	0.75	0.41
Inverse Reconstruction																												
Spherical Aberr.	E	22.69	0.65	0.20	0.10	18.97	0.51	0.29	0.14	19.40	0.60	0.25	0.12	14.22	0.39	0.46	0.17	20.00	0.63	0.38	0.22	7.73	0.41	0.69	0.42	–	–	–
	P	22.60	0.64	0.21	0.10	20.25	0.58	0.26	0.13	19.75	0.61	0.26	0.12	14.44	0.39	0.46	0.17	19.55	0.61	0.48	0.29	8.16	0.42	0.75	0.41			
Chromatic Aberr.	E	21.52	0.67	0.23	0.11	20.47	0.64	0.23	0.13	19.32	0.64	0.31	0.14	15.71	0.47	0.45	0.18	19.62	0.66	0.37	0.21	5.69	0.39	0.83	0.63	–	–	–
	P	21.48	0.67	0.24	0.12	21.17	0.65	0.24	0.14	18.98	0.65	0.30	0.14	15.53	0.45	0.47	0.19	18.99	0.59	0.46	0.26	9.26	0.41	0.64	0.32			
Motion Blur	E	21.60	0.65	0.38	0.18	17.77	0.49	0.49	0.24	20.39	0.66	0.38	0.18	14.65	0.43	0.54	0.24	20.70	0.76	0.42	0.28	9.20	0.49	0.68	0.39			
	P	21.56	0.65	0.37	0.18	19.67	0.56	0.42	0.21	20.63	0.68	0.37	0.18	15.16	0.46	0.52	0.23	20.42	0.74	0.46	0.31	14.22	0.60	0.58	0.28	31.37	0.92	0.14
Masked Sampling	E	13.93	0.47	0.46	0.23	15.07	0.48	0.44	0.22	14.98	0.34	0.65	0.29	12.99	0.37	0.56	0.24	9.40	0.09	0.97	0.53	7.68	0.30	0.87	0.40	–	–	–
	P	13.58	0.46	0.48	0.25	13.88	0.47	0.47	0.26	14.25	0.31	0.69	0.30	11.97	0.34	0.61	0.29	10.01	0.10	0.97	0.50	7.44	0.23	0.84	0.41			
Comp. Sensing	E	21.64	0.60	0.28	0.12	15.58	0.44	0.43	0.15	15.67	0.42	0.49	0.23	14.93	0.42	0.45	0.16	20.93	0.63	0.61	0.33	9.06	0.37	0.82	0.39	–	–	–
	P	20.67	0.60	0.29	0.12	14.83	0.42	0.46	0.16	13.12	0.38	0.55	0.23	13.76	0.39	0.48	0.18	21.67	0.64	0.58	0.32	10.40	0.35	0.66	0.31			
Super-Resolution	E	21.74	0.60	0.24	0.12	17.69	0.47	0.31	0.13	16.41	0.39	0.49	0.22	14.36	0.39	0.46	0.16	20.16	0.61	0.32	0.16	11.30	0.44	0.54	0.23			
	P	21.47	0.60	0.24	0.12	17.09	0.44	0.35	0.15	15.85	0.41	0.45	0.20	13.77	0.37	0.50	0.18	19.82	0.60	0.30	0.14	10.83	0.40	0.58	0.26	20.55	0.58	0.39
Inpainting	E	23.78	0.78	0.12	0.06	23.80	0.75	0.12	0.06	19.49	0.72	0.28	0.14	15.18	0.43	0.41	0.14	12.81	0.47	0.49	0.28	6.76	0.30	0.80	0.43			
	P	23.11	0.77	0.14	0.06	23.15	0.74	0.13	0.07	19.02	0.76	0.22	0.12	14.27	0.42	0.45	0.16	13.98	0.53	0.41	0.22	8.84	0.31	0.72	0.34	30.33	0.93	0.05
Computational Sensing																												
Lensless Imaging	E	7.01	0.20	0.79	0.52	7.33	0.19	0.77	0.53	7.83	0.14	0.92	0.53	7.90	0.23	0.84	0.48	8.97	0.25	0.79	0.51	6.27	0.16	0.77	0.47			
	P	7.03	0.19	0.79	0.56	7.62	0.21	0.83	0.60	7.44	0.19	0.91	0.56	8.02	0.24	0.89	0.57	8.81	0.24	0.76	0.50	6.97	0.17	0.78	0.47	19.98	0.68	0.29
LF Extrapolation	E	16.42	0.53	0.45	0.23	16.24	0.46	0.51	0.27	18.75	0.52	0.46	0.26	12.30	0.35	0.67	0.31	11.90	0.43	0.63	0.34	7.91	0.35	0.81	0.46	–	–	–
	P	16.25	0.52	0.46	0.25	16.44	0.45	0.53	0.28	17.89	0.51	0.46	0.26	14.26	0.37	0.60	0.30	15.40	0.51	0.53	0.27	13.23	0.42	0.58	0.32			
LF Interpolation	E	21.30	0.64	0.33	0.18	19.28	0.58	0.37	0.20	20.26	0.60	0.39	0.18	15.46	0.47	0.51	0.22	16.56	0.53	0.53	0.29	8.20	0.42	0.77	0.46	–	–	–
	P	20.83	0.61	0.35	0.19	18.25	0.54	0.44	0.21	20.08	0.58	0.40	0.19	15.51	0.45	0.51	0.21	18.36	0.55	0.48	0.27	10.84	0.38	0.66	0.29			
Event Recon.	E	9.05	0.26	0.64	0.43	10.51	0.32	0.60	0.40	10.78	0.25	0.68	0.48	9.84	0.27	0.63	0.43	8.44	0.12	0.88	0.52	5.57	0.15	0.64	0.47	–	–	–
	P	9.15	0.26	0.61	0.43	10.50	0.30	0.62	0.42	9.81	0.24	0.66	0.44	10.24	0.25	0.65	0.45	7.62	0.05	0.94	0.54	5.44	0.12	0.67	0.47			
ToF Depth	E	9.04	0.63	0.60	0.40	8.46	0.67	0.72	0.47	11.12	0.51	0.62	0.39	8.47	0.47	0.85	0.49	10.33	0.42	0.75	0.49	–	–	–	–	–	–	–
	P	8.35	0.58	0.60	0.40	7.93	0.59	0.66	0.41	10.04	0.49	0.62	0.40	8.20	0.44	0.84	0.48	10.24	0.41	0.75	0.49							
Calibration																												
Lens Distortion	E	19.13	0.56	0.13	0.07	10.28	0.25	0.59	0.17	16.46	0.48	0.20	0.09	11.03	0.30	0.55	0.20	14.54	0.43	0.42	0.21	8.33	0.33	0.68	0.34	–	–	–
	P	19.03	0.56	0.13	0.07	10.42	0.26	0.58	0.18	16.53	0.49	0.19	0.09	11.28	0.30	0.54	0.19	17.01	0.48	0.29	0.15	6.38	0.31	0.71	0.36			
Forward Process																												
Holography (CGH)	F	10.35	0.11	0.76	0.51	9.87	0.14	0.72	0.49	10.23	0.12	0.90	0.59	9.99	0.13	0.91	0.62	9.22	0.16	0.75	0.44	5.64	0.20	0.82	0.55	–	–	–
HDR	F	17.93	0.54	0.32	0.17	15.38	0.45	0.37	0.19	13.74	0.26	0.62	0.28	13.80	0.39	0.52	0.22	13.38	0.47	0.47	0.26	7.17	0.31	0.77	0.34	–	–	–
Denoising	F	19.58	0.44	0.57	0.31	23.23	0.44	0.34	0.24	13.08	0.18	0.67	0.38	17.72	0.26	0.59	0.33	17.91	0.42	0.89	0.43	5.45	0.28	1.16	0.53	–	–	–
Comp. Sensing	F	20.78	0.70	0.26	0.29	16.32	0.52	0.58	0.43	15.42	0.61	0.39	0.30	8.79	0.16	0.72	0.51	16.72	0.62	0.30	0.25	8.23	0.31	0.60	0.41	–	–	–
Lensless Imaging	F	8.83	0.15	0.95	0.64	9.05	0.19	0.87	0.54	9.55	0.10	1.07	0.57	10.06	0.33	0.79	0.50	7.91	0.26	0.83	0.57	8.23	0.31	0.94	0.67	–	–	–

Neural Network references: CGH (Shi et al., 2021), Dm-Q (A Sharif et al., 2021), Dm-R (A Sharif et al., 2021), WB (Afifi & Brown, 2020), DN (Zhang et al., 2018a), MB (Tsai et al., 2022), SR (Wang et al., 2018), IP (Suvorov et al., 2022), LL (Monakhova et al., 2019).

Table 19: **Per-task no-reference metrics.** Mean NIQE, MUSIQ, CLIP-IQA, and MANIQA per task over the images each model returned. These metrics see only the output, never the reference, so they measure how natural an image looks rather than whether it is correct: HunyuanImage-3.0, last on every full-reference metric and on the judge, is among the best-rated models here. Expert/Planner (E/P) values are stacked top/bottom; Forward covers five tasks. Neural Network references report NIQE only; – marks unavailable entries.

Task	Phase	*Proprietary*																*Open-weight*								*Reference*
		Gemini-3.1-Flash-Image				GPT-Image-2				FLUX.2-Pro				GPT-Image-1.5				Qwen-Image-Edit-2511				HunyuanImage-3.0				Neural Network
		NIQE↓	MUSIQ↑	CLIP-IQA↑	MANIQA↑	NIQE↓	MUSIQ↑	CLIP-IQA↑	MANIQA↑	NIQE↓	MUSIQ↑	CLIP-IQA↑	MANIQA↑	NIQE↓	MUSIQ↑	CLIP-IQA↑	MANIQA↑	NIQE↓	MUSIQ↑	CLIP-IQA↑	MANIQA↑	NIQE↓	MUSIQ↑	CLIP-IQA↑	MANIQA↑	NIQE↓
Inverse Process																										
Ray and Wave Optics																										
Holography (CGH)	E	5.68	56.5	0.52	0.51	10.62	54.9	0.62	0.51	6.81	65.7	0.54	0.47	7.77	53.5	0.54	0.41	6.46	63.5	0.30	0.35	5.75	60.1	0.36	0.40	
	P	3.48	67.0	0.53	0.54	6.19	63.2	0.57	0.47	8.02	66.1	0.60	0.49	5.32	50.4	0.51	0.34	6.57	61.5	0.26	0.35	6.12	59.9	0.33	0.43	4.32
Image Signal Processing																										
Demosaic (Hexa)	E	3.55	71.8	0.74	0.45	3.87	71.9	0.68	0.44	3.29	70.5	0.64	0.45	4.25	71.0	0.64	0.41	8.08	54.3	0.49	0.39	7.31	69.2	0.57	0.47	
	P	3.58	70.9	0.74	0.44	3.92	67.2	0.68	0.44	3.31	66.4	0.66	0.42	4.51	66.8	0.65	0.39	7.20	55.6	0.56	0.40	6.10	67.2	0.59	0.47	–
Demosaic (Quad)	E	3.42	72.8	0.75	0.47	3.85	71.8	0.69	0.45	3.51	69.5	0.66	0.46	4.05	71.8	0.65	0.42	7.24	52.8	0.50	0.38	7.42	68.4	0.55	0.46	
	P	3.46	72.8	0.74	0.47	3.94	67.6	0.68	0.44	3.41	67.2	0.67	0.44	4.65	67.5	0.65	0.40	6.90	53.6	0.53	0.40	5.44	69.1	0.58	0.45	3.11
Demosaic (Reg.)	E	3.53	72.4	0.74	0.47	3.88	71.8	0.68	0.44	3.84	69.7	0.65	0.43	4.12	72.0	0.64	0.41	14.99	46.9	0.51	0.41	7.66	69.7	0.59	0.49	
	P	3.56	72.3	0.74	0.47	3.85	69.0	0.69	0.45	3.91	68.3	0.66	0.43	4.17	70.9	0.66	0.41	14.85	47.5	0.51	0.42	4.91	67.4	0.62	0.43	2.95
HDR	E	3.77	71.4	0.73	0.44	3.91	74.2	0.72	0.47	3.75	68.0	0.62	0.40	4.13	72.3	0.63	0.41	5.95	56.6	0.42	0.33	3.98	66.1	0.51	0.33	
	P	3.78	70.9	0.73	0.44	3.97	73.5	0.72	0.47	3.82	67.3	0.62	0.40	4.23	71.1	0.63	0.40	6.02	56.1	0.42	0.32	4.17	71.2	0.60	0.42	–
White Balance	E	4.21	64.4	0.49	0.37	3.71	72.4	0.59	0.43	3.28	71.1	0.58	0.43	3.97	70.3	0.54	0.42	5.05	59.3	0.40	0.33	7.36	58.0	0.53	0.43	
	P	4.33	63.9	0.49	0.36	3.80	71.3	0.54	0.41	3.46	70.4	0.55	0.42	4.06	69.8	0.53	0.42	5.01	56.8	0.37	0.32	5.76	68.2	0.57	0.44	4.47
Denoising	E	6.94	52.6	0.53	0.33	9.10	49.9	0.42	0.34	5.91	52.8	0.53	0.35	9.00	50.7	0.54	0.35	8.16	39.3	0.29	0.26	7.35	67.8	0.67	0.46	
	P	6.98	54.7	0.54	0.35	8.93	50.2	0.47	0.34	6.01	52.6	0.49	0.34	9.56	48.2	0.52	0.33	8.36	41.5	0.31	0.28	6.71	51.9	0.40	0.33	5.04
Inverse Reconstruction																										
Spherical Aberr.	E	3.82	69.9	0.72	0.41	4.12	72.1	0.74	0.47	3.38	69.7	0.65	0.42	4.00	72.4	0.68	0.44	6.42	50.4	0.38	0.28	6.70	56.0	0.55	0.39	
	P	3.89	69.1	0.71	0.40	4.59	68.6	0.69	0.41	3.50	67.6	0.63	0.40	4.22	70.6	0.65	0.42	8.34	36.7	0.30	0.24	7.06	51.7	0.48	0.32	–
Chromatic Aberr.	E	3.98	68.0	0.69	0.40	4.67	67.5	0.64	0.38	3.75	60.8	0.54	0.32	4.82	61.8	0.51	0.33	5.25	53.0	0.38	0.29	10.21	37.8	0.45	0.37	
	P	4.07	66.7	0.67	0.39	4.91	66.2	0.63	0.36	3.80	60.8	0.54	0.33	4.82	61.4	0.52	0.33	6.21	40.0	0.33	0.24	4.77	59.3	0.50	0.35	–
Motion Blur	E	3.67	67.0	0.60	0.36	4.11	71.1	0.65	0.41	4.23	51.5	0.38	0.27	4.21	67.0	0.50	0.35	9.24	18.9	0.18	0.13	6.37	55.8	0.47	0.35	
	P	3.96	63.6	0.56	0.34	4.94	58.8	0.49	0.31	4.63	43.1	0.30	0.22	4.91	58.6	0.43	0.29	9.91	18.8	0.19	0.12	7.43	26.7	0.23	0.15	6.94
Masked Sampling	E	4.42	68.9	0.74	0.46	4.50	70.8	0.69	0.46	6.06	64.4	0.70	0.49	4.80	68.7	0.65	0.41	16.35	53.8	0.43	0.49	5.04	64.3	0.47	0.40	–
	P	4.86	67.8	0.73	0.46	5.60	65.8	0.66	0.44	5.48	59.9	0.69	0.46	5.48	65.3	0.67	0.42	14.59	53.3	0.44	0.50	4.83	62.5	0.51	0.35	
Comp. Sensing	E	3.77	69.5	0.68	0.41	4.27	70.8	0.63	0.40	4.09	68.8	0.65	0.42	4.09	71.7	0.64	0.42	10.13	24.5	0.35	0.24	6.47	66.2	0.48	0.43	–
	P	3.74	69.8	0.68	0.41	4.21	70.9	0.64	0.41	4.04	68.5	0.64	0.43	4.22	71.7	0.66	0.43	10.04	24.6	0.35	0.24	4.53	66.6	0.56	0.36	
Super-Resolution	E	4.02	67.1	0.68	0.37	3.98	71.8	0.69	0.43	4.93	67.2	0.67	0.44	4.01	72.3	0.67	0.43	4.83	61.7	0.46	0.33	3.99	71.4	0.64	0.42	
	P	3.83	68.1	0.68	0.38	3.93	73.0	0.71	0.47	4.05	70.9	0.69	0.48	4.02	72.8	0.69	0.45	4.64	65.7	0.50	0.35	4.03	70.7	0.63	0.41	5.42
Inpainting	E	3.84	68.4	0.68	0.41	3.97	70.9	0.67	0.43	3.92	65.5	0.64	0.42	4.23	70.7	0.62	0.41	5.34	63.9	0.49	0.44	7.44	57.7	0.64	0.36	
	P	3.80	68.7	0.69	0.41	3.95	71.1	0.66	0.44	3.31	70.1	0.63	0.45	4.20	71.3	0.64	0.42	4.99	67.8	0.55	0.46	4.47	63.4	0.56	0.35	3.02
Computational Sensing																										
Lensless Imaging	E	11.58	54.9	0.62	0.47	12.05	47.8	0.58	0.39	10.96	32.4	0.44	0.24	11.07	34.5	0.50	0.23	16.26	28.2	0.22	0.20	6.15	67.5	0.79	0.49	
	P	15.98	40.8	0.50	0.41	19.50	29.4	0.46	0.32	11.45	34.1	0.43	0.27	16.28	27.6	0.56	0.25	13.45	27.8	0.19	0.18	7.57	65.3	0.74	0.49	7.38
LF Extrapolation	E	5.49	38.8	0.28	0.20	4.46	52.5	0.36	0.25	4.47	44.2	0.31	0.22	4.67	58.1	0.36	0.27	6.66	34.0	0.22	0.19	7.10	61.1	0.40	0.40	–
	P	5.09	44.1	0.34	0.22	4.11	58.3	0.42	0.29	4.54	47.6	0.33	0.24	4.35	61.0	0.41	0.30	6.71	30.9	0.18	0.19	5.38	51.2	0.34	0.27	–
LF Interpolation	E	5.89	36.6	0.28	0.21	5.82	40.4	0.28	0.24	5.02	42.0	0.31	0.23	5.64	41.1	0.27	0.22	8.72	30.4	0.21	0.20	7.19	61.3	0.45	0.41	–
	P	5.64	43.3	0.34	0.25	5.46	53.3	0.37	0.29	4.94	45.8	0.35	0.25	4.63	58.5	0.38	0.29	8.69	29.3	0.19	0.20	5.07	59.4	0.42	0.32	–
Event Recon.	E	8.00	64.6	0.58	0.56	6.95	62.7	0.56	0.49	10.89	44.0	0.46	0.34	6.97	47.4	0.56	0.36	18.44	43.9	0.49	0.40	8.32	59.6	0.44	0.47	–
	P	8.73	64.9	0.64	0.58	8.52	57.6	0.60	0.49	11.32	46.9	0.53	0.39	9.74	42.9	0.55	0.34	19.26	49.0	0.53	0.42	8.43	57.8	0.46	0.49	–
ToF Depth	E	15.36	44.4	0.49	0.41	22.68	36.8	0.41	0.39	15.80	29.9	0.38	0.18	15.01	24.2	0.32	0.19	25.45	26.7	0.26	0.22	–	–	–	–	–
	P	15.79	46.3	0.48	0.44	19.85	52.5	0.43	0.41	15.66	32.8	0.41	0.20	13.30	29.1	0.37	0.23	25.24	27.9	0.28	0.22					
Calibration																										
Lens Distortion	E	3.23	69.1	0.67	0.43	3.95	71.2	0.64	0.47	3.04	70.0	0.65	0.45	3.90	69.4	0.58	0.43	4.45	65.5	0.48	0.39	3.83	73.0	0.58	0.44	–
	P	3.26	69.1	0.67	0.43	3.88	71.5	0.63	0.48	3.03	69.9	0.65	0.44	3.96	69.4	0.58	0.42	4.28	69.6	0.56	0.45	4.95	68.1	0.59	0.42	
Forward Process																										
Holography (CGH)	F	10.84	52.8	0.56	0.42	7.11	48.8	0.37	0.35	15.49	52.8	0.59	0.51	9.94	29.3	0.50	0.23	6.00	55.8	0.42	0.34	8.79	47.4	0.48	0.36	–
HDR	F	4.11	66.2	0.71	0.40	4.90	67.4	0.69	0.37	5.07	50.2	0.60	0.34	4.04	67.6	0.62	0.39	4.65	68.4	0.61	0.43	3.88	68.9	0.57	0.38	–
Denoising	F	6.06	39.1	0.44	0.28	10.41	32.6	0.36	0.21	7.57	44.3	0.52	0.38	7.40	42.6	0.48	0.32	8.75	37.6	0.25	0.25	6.79	55.9	0.45	0.36	–
Comp. Sensing	F	18.14	25.4	0.41	0.20	28.25	23.6	0.29	0.18	20.44	33.1	0.62	0.31	18.59	29.4	0.33	0.25	11.79	32.5	0.60	0.28	12.23	35.0	0.45	0.38	–
Lensless Imaging	F	5.89	35.7	0.42	0.18	7.12	26.8	0.38	0.14	10.06	44.9	0.52	0.31	9.15	17.4	0.35	0.14	8.76	30.0	0.25	0.18	7.40	52.0	0.51	0.31	–

Neural Network references: CGH (Shi et al., 2021), Dm-Q (A Sharif et al., 2021), Dm-R (A Sharif et al., 2021), WB (Afifi & Brown, 2020), DN (Zhang et al., 2018a), MB (Tsai et al., 2022), SR (Wang et al., 2018), IP (Suvorov et al., 2022), LL (Monakhova et al., 2019).

Table 20: **Per-task VLM-judge scores.** Mean VLM-judge score and its four sub-scores (structural fidelity, task success, photometric fidelity, artifact-freeness; each 0–100, higher is better; Sec. C) per task over the images each model returned. Expert/Planner (E/P) values are stacked top/bottom. CGH, ToF depth, and Forward outputs are not judged.

		Proprietary																				*Open-weight*									
Task	**Phase**	**Gemini-3.1-Flash-Image**					**GPT-Image-2**					**FLUX.2-Pro**					**GPT-Image-1.5**					**Qwen-Image-Edit-2511**					**HunyuanImage-3.0**				
		Judge↑	S^{fid}↑	S^{task}↑	S^{col}↑	S^{art}↑	Judge↑	S^{fid}↑	S^{task}↑	S^{col}↑	S^{art}↑	Judge↑	S^{fid}↑	S^{task}↑	S^{col}↑	S^{art}↑	Judge↑	S^{fid}↑	S^{task}↑	S^{col}↑	S^{art}↑	Judge↑	S^{fid}↑	S^{task}↑	S^{col}↑	S^{art}↑	Judge↑	S^{fid}↑	S^{task}↑	S^{col}↑	S^{art}↑
Image Signal Processing																															
Demosaic (Hexa)	E	65.1	77.7	85.0	58.3	77.0	65.7	79.9	85.4	59.0	77.3	50.8	68.4	71.2	44.1	60.4	55.3	63.7	77.1	51.8	65.7	6.5	22.9	5.1	4.7	9.1	0.8	1.4	0.8	1.1	0.8
	P	55.0	73.1	72.0	48.4	70.8	27.6	69.9	31.8	18.1	66.7	28.1	68.4	35.6	18.6	61.4	43.2	65.4	58.1	34.3	64.5	9.6	36.8	7.8	5.8	18.5	1.7	3.0	2.1	2.2	2.1
Demosaic (Quad)	E	62.4	75.6	82.4	56.1	74.1	67.3	80.1	85.5	62.7	76.3	55.3	70.5	75.1	50.2	62.9	54.8	65.1	76.9	51.1	65.5	17.4	46.5	19.7	12.5	27.3	0.4	1.0	0.4	0.6	0.3
	P	59.1	72.1	77.3	53.9	70.5	28.7	71.6	30.7	20.5	64.9	38.2	72.5	46.1	30.2	66.1	39.2	64.3	50.4	31.3	63.8	15.9	45.4	17.9	11.5	26.9	3.0	6.6	4.0	3.8	3.2
Demosaic (Reg.)	E	61.0	76.5	79.8	54.3	75.4	72.0	85.2	86.3	65.9	82.4	52.3	71.3	73.2	43.9	67.3	60.5	70.8	77.3	57.0	71.6	11.9	37.9	11.0	7.7	22.1	0.8	1.8	0.8	0.8	1.8
	P	57.1	73.7	73.8	51.2	71.6	40.0	76.5	43.9	31.9	73.3	39.9	73.3	49.4	32.4	69.1	57.3	71.6	73.2	52.1	71.5	11.8	36.9	10.4	7.3	21.3	2.4	5.8	2.6	2.7	3.0
HDR	E	81.5	86.5	83.6	81.7	84.7	81.4	87.0	86.1	80.7	81.9	72.5	84.8	76.8	71.9	78.0	74.5	79.4	83.1	74.7	74.3	29.7	54.8	30.2	39.4	41.6	16.4	18.8	21.7	27.2	26.7
	P	80.9	87.8	82.3	81.3	85.0	81.1	87.7	84.7	80.0	82.7	71.2	84.0	74.0	71.9	78.7	71.9	79.2	79.7	72.1	74.4	30.0	54.1	29.8	39.5	41.7	37.0	44.4	47.2	46.2	33.0
White Balance	E	74.9	86.0	78.4	74.2	81.8	77.3	87.5	81.1	76.4	84.4	62.3	74.2	70.4	63.5	68.3	54.1	64.1	66.6	56.6	63.7	47.9	64.2	53.0	51.0	54.8	5.9	8.4	8.1	9.3	10.1
	P	74.3	87.2	76.7	73.6	83.0	76.7	87.8	80.2	75.0	85.9	63.9	75.8	72.7	64.9	72.1	49.6	63.5	60.8	50.5	61.5	40.9	65.2	43.8	42.7	57.3	12.9	18.1	16.4	16.6	19.0
Denoising	E	76.7	84.7	85.3	77.5	78.0	77.7	85.6	86.4	79.7	76.5	60.1	74.5	74.5	62.8	65.0	64.5	75.6	80.4	68.0	65.6	41.9	61.3	39.5	57.8	43.5	4.0	4.9	7.8	5.7	5.6
	P	74.1	82.6	84.8	74.4	76.0	77.2	84.4	86.0	79.8	77.9	62.9	74.7	77.8	64.5	65.1	63.4	73.7	79.1	66.3	65.9	45.1	62.8	44.8	61.4	46.5	2.5	3.1	3.8	5.4	3.2
Inverse Reconstruction																															
Spherical Aberr.	E	82.5	85.3	88.4	86.4	79.5	79.2	82.4	88.3	82.8	75.8	79.0	82.1	87.8	82.3	75.9	69.2	71.1	86.7	75.1	68.7	51.3	65.4	53.9	71.3	54.2	17.0	19.0	28.2	21.9	23.7
	P	81.7	85.0	87.6	85.9	79.2	81.7	85.5	87.2	84.5	78.6	79.0	82.7	87.6	82.1	75.6	70.1	72.5	86.2	75.1	69.7	36.5	58.0	34.0	67.0	47.6	11.3	14.5	16.1	16.9	13.4
Chromatic Aberr.	E	80.2	85.7	88.4	78.2	82.3	83.0	87.8	90.6	80.3	84.9	75.6	84.0	83.2	73.1	78.5	73.0	78.6	87.5	71.2	75.1	43.6	62.1	45.4	49.3	50.9	2.4	3.9	4.0	2.5	4.8
	P	78.8	84.7	86.8	77.2	81.4	82.1	87.3	89.6	79.3	84.4	74.1	83.9	80.3	72.1	77.5	71.3	76.6	86.5	69.8	74.8	32.0	53.7	29.4	41.6	40.3	9.7	19.5	9.3	12.5	12.7
Motion Blur	E	69.3	70.9	82.9	78.1	68.4	64.3	64.8	83.4	75.1	64.0	68.8	72.3	79.1	80.6	67.0	59.6	60.1	83.5	72.6	61.0	22.3	51.5	14.7	67.6	47.4	17.5	18.8	25.3	29.9	21.3
	P	67.6	70.1	82.0	78.7	66.3	70.8	73.2	82.6	80.5	68.7	67.1	72.4	73.2	80.1	66.4	65.1	66.4	82.6	75.9	65.4	22.1	48.2	14.9	64.8	44.2	23.5	38.5	20.8	54.2	36.1
Masked Sampling	E	56.0	59.7	68.8	58.8	61.4	59.5	63.8	72.1	63.9	63.0	56.7	63.7	66.5	61.0	53.3	57.3	61.1	71.5	61.2	64.2	11.3	18.3	13.4	17.8	8.4	1.0	0.8	1.2	4.5	1.7
	P	55.4	60.4	68.1	57.5	59.1	61.1	67.4	73.0	62.2	66.3	55.0	64.6	64.2	57.7	52.9	53.2	58.1	67.5	56.3	61.8	15.6	25.4	19.4	22.1	10.8	2.7	3.2	4.3	8.9	3.7
Comp. Sensing	E	81.7	84.2	87.8	87.2	80.3	80.1	81.9	88.3	86.7	81.0	60.2	64.2	73.9	71.5	60.4	73.3	75.3	85.3	81.1	75.2	56.2	69.0	60.1	71.3	45.7	6.9	11.1	9.6	8.6	7.2
	P	83.6	86.5	90.5	88.4	81.3	79.6	81.6	88.4	86.3	79.8	61.5	66.8	75.9	72.1	58.8	69.7	72.9	82.7	78.9	69.5	59.8	73.1	62.7	76.9	47.8	19.1	22.4	25.4	36.6	17.7
Super-Resolution	E	86.8	90.0	88.9	90.7	83.4	85.1	87.8	88.2	89.9	82.9	72.8	78.7	80.2	79.2	69.6	77.0	79.4	84.9	83.1	77.3	76.1	82.9	79.0	84.7	69.3	46.5	50.1	55.0	53.8	50.5
	P	86.6	89.5	89.2	90.7	83.4	83.3	86.1	86.9	88.6	81.8	69.4	74.6	80.7	76.6	68.5	71.3	73.9	82.1	80.0	73.2	77.9	82.9	82.8	85.7	72.3	34.9	38.1	42.5	46.0	39.6
Inpainting	E	78.3	79.0	85.5	88.2	78.4	87.1	88.2	91.6	91.4	85.4	42.4	44.6	47.1	67.1	42.2	65.3	65.1	77.0	78.4	69.6	8.3	11.7	7.3	27.9	8.3	0.8	0.9	0.7	4.9	1.4
	P	77.4	77.9	85.2	87.2	78.1	85.3	86.4	90.1	91.0	84.5	41.7	43.6	47.0	68.2	42.8	60.6	61.1	74.0	73.6	65.0	16.0	19.7	17.1	39.1	18.1	4.5	4.7	5.3	18.2	6.0
Computational Sensing																															
Lensless Imaging	E	2.6	1.7	3.8	15.4	19.6	3.7	3.4	4.2	19.1	24.8	2.7	2.6	2.4	13.4	16.9	3.7	3.3	5.4	17.8	23.4	5.0	5.0	5.7	21.4	20.1	2.5	2.1	3.4	13.0	18.9
	P	2.9	2.5	3.0	15.7	20.6	3.4	3.3	4.0	18.8	23.9	3.3	3.3	4.0	17.5	21.3	3.7	3.4	4.3	19.9	22.5	6.4	5.9	7.2	23.8	20.0	2.5	2.2	3.3	11.9	16.8
LF Extrapolation	E	43.7	47.5	46.9	68.4	56.7	55.6	60.7	57.1	70.5	64.2	55.5	61.8	56.1	70.4	62.0	26.9	30.7	33.7	47.8	41.6	25.8	32.1	28.8	34.0	25.7	4.2	5.4	5.8	13.0	3.6
	P	41.8	46.0	44.2	64.6	53.4	54.9	61.1	57.8	67.0	62.9	53.0	58.8	54.5	68.0	59.3	43.1	49.7	49.7	52.7	50.8	49.6	56.0	52.4	64.9	50.0	34.3	38.8	37.9	47.9	36.2
LF Interpolation	E	66.3	73.0	67.8	73.2	65.3	71.3	77.2	72.3	76.2	69.9	73.6	79.7	79.3	78.1	69.9	59.6	65.7	68.6	67.9	59.5	39.0	50.2	42.5	44.3	34.1	4.2	5.2	6.2	12.6	3.1
	P	64.7	72.0	67.6	72.6	63.0	72.4	78.0	76.8	76.3	70.9	71.5	78.1	77.4	76.5	67.9	59.4	65.2	72.0	63.1	61.9	48.7	60.0	51.6	58.5	42.4	15.9	18.0	21.2	28.5	15.8
Event Recon.	E	16.6	18.6	24.4	29.5	28.7	18.2	18.5	27.1	39.7	32.5	28.5	36.8	32.3	45.0	33.1	15.2	15.9	23.4	35.2	28.5	10.5	15.4	12.1	17.8	15.0	18.6	24.0	24.3	23.8	19.5
	P	20.2	22.6	29.5	37.2	31.6	27.3	30.2	39.3	43.4	40.9	29.9	36.7	35.8	45.2	36.2	23.6	26.7	33.3	38.0	36.9	12.2	20.7	9.7	27.2	13.0	14.8	18.6	21.1	22.2	16.4
Calibration																															
Lens Distortion	E	82.2	86.8	81.1	90.3	82.8	47.6	53.8	52.3	80.5	59.4	76.6	83.2	75.8	86.9	79.3	43.1	48.0	49.1	71.0	53.0	37.2	47.5	45.3	49.3	37.8	10.8	12.3	12.7	18.2	13.1
	P	80.8	85.2	81.0	89.4	82.4	47.7	53.2	50.9	80.4	59.1	75.6	82.6	75.4	85.6	78.8	44.3	50.1	49.6	72.2	53.8	52.1	61.6	61.5	69.5	49.2	4.8	6.9	5.8	9.8	6.2

Table 21: **Coverage and the treatment of missing jobs, Expert protocol.** *Left:* coverage of the 2,900 Expert jobs. **Miss**: jobs with no scorable image; **Blk**: calls refused by the provider's content filter (a retried job can count twice). *Right:* Overall when a missing job is **Drop**ped, scored **Zero** (the benchmark's policy), **Imp**uted from the reference model, or given the rank-preserving **Upper** bound; **Com** rescores every model on the tasks all six completed. Gemini-3.1-Flash-Image leads under every treatment. HunyuanImage-3.0 returned nothing on time-of-flight depth, so that task enters only the Zero column.

	Coverage				Overall, by treatment				
Model	Jobs	Miss	Blk	Cov.%	Drop	Zero	Imp	Upper	Com
Gemini-3.1-Flash-Image	2,900	40	0	98.6	0.603	0.594	0.603	0.603	0.615
GPT-Image-2	2,900	197	197	93.2	0.587	0.552	0.587	0.587	0.600
FLUX.2-Pro	2,900	53	47	98.2	0.551	0.541	0.551	0.551	0.559
GPT-Image-1.5	2,900	216	205	92.6	0.512	0.478	0.512	0.512	0.525
Qwen-Image-Edit-2511	2,900	0	0	100.0	0.394	0.394	0.394	0.394	0.403
HunyuanImage-3.0	2,900	391	0	86.5	0.291	0.250	0.291	0.292	0.291

Table 22: **Are the refused images easier or harder than the rest?** The difficulty index is Qwen-Image-Edit's per-image unified score (it completed every Expert job). Δ is its mean on the focal model's *refused* images minus its mean on the *completed* ones; $\Delta > 0$ means the refused images were easier. Intervals: percentile bootstrap (10^4 resamples); p: two-sided Mann-Whitney U; bold: $p < 0.05$. The pooled row standardizes within each task and is in standard deviations. Tasks without refusals are omitted, and so are intervals on fewer than 3 refused images.

Task	n_{ref}	n_{done}	mean|refused	mean|done	Δ	95% CI	p
GPT-Image-1.5							
Comp. Sensing (Block)	35	265	0.461	0.481	-0.020	[-0.068, +0.024]	0.837
Longitudinal Chromatic	8	92	0.490	0.526	-0.036	[-0.096, +0.024]	0.351
Motion Blur	7	93	0.381	0.380	+0.001	[-0.016, +0.018]	0.701
Spherical Aberration	38	362	0.519	0.545	-0.026	[-0.067, +0.016]	0.253
Demosaicking (Hexadeca)	7	93	0.244	0.282	-0.038	[-0.090, +0.006]	0.185
Demosaicking (Quad)	11	89	0.349	0.355	-0.006	[-0.044, +0.035]	0.708
Demosaicking (Regular)	8	92	0.235	0.252	-0.017	[-0.077, +0.041]	0.779
Denoising	1	99	0.278	0.468	-0.190	–	0.260
Event-based Intensity	2	98	0.200	0.222	-0.022	–	0.747
Inpainting	14	86	0.400	0.362	+0.037	[+0.009, +0.067]	0.087
Lens Distortion	2	98	0.399	0.501	-0.102	–	0.352
Lightfield Extrapolation	3	97	0.433	0.350	+0.083	[-0.061, +0.258]	0.272
Lightfield Interpolation	3	97	0.344	0.427	-0.083	[-0.191, +0.081]	0.403
Rand. Masked Subsampling	58	442	0.206	0.219	-0.013	[-0.028, +0.002]	0.299
Super-Resolution	13	87	0.751	0.658	**+0.093**	[+0.069, +0.118]	<0.001
Time-of-Flight Depth	1	99	0.242	0.294	-0.051	–	0.340
White Balance	5	95	0.539	0.557	-0.017	[-0.124, +0.094]	0.518
All tasks (pooled, per-task z)	216	2384	-0.089	0.008	-0.097	[-0.232, +0.038]	0.336
GPT-Image-2							
Comp. Sensing (Block)	42	258	0.464	0.482	-0.017	[-0.063, +0.025]	0.708
Longitudinal Chromatic	8	92	0.508	0.524	-0.016	[-0.069, +0.035]	0.658
Motion Blur	5	95	0.379	0.380	-0.001	[-0.021, +0.018]	0.830
Spherical Aberration	33	367	0.517	0.544	-0.027	[-0.070, +0.016]	0.249
Demosaicking (Hexadeca)	7	93	0.231	0.283	-0.053	[-0.102, -0.007]	0.050
Demosaicking (Quad)	7	93	0.357	0.354	+0.003	[-0.049, +0.058]	1.000
Demosaicking (Regular)	8	92	0.197	0.256	-0.059	[-0.105, -0.015]	0.090
Denoising	1	99	0.278	0.468	-0.190	–	0.260
High Dynamic Range	1	99	0.410	0.434	-0.024	–	0.900
Inpainting	13	87	0.399	0.363	+0.036	[+0.006, +0.068]	0.124
Lightfield Extrapolation	1	99	0.287	0.353	-0.066	–	1.000
Rand. Masked Subsampling	53	447	0.208	0.218	-0.010	[-0.025, +0.006]	0.479
Super-Resolution	13	87	0.752	0.658	**+0.095**	[+0.071, +0.119]	<0.001
Time-of-Flight Depth	1	99	0.242	0.294	-0.051	–	0.340
White Balance	4	96	0.542	0.556	-0.014	[-0.141, +0.122]	0.690
All tasks (pooled, per-task z)	197	2203	-0.091	0.008	-0.100	[-0.244, +0.043]	0.377
FLUX.2-Pro							
Comp. Gen. Holography	1	99	0.595	0.389	+0.206	–	0.080
Comp. Sensing (Block)	5	295	0.551	0.478	**+0.073**	[+0.054, +0.092]	0.038
Longitudinal Chromatic	3	97	0.540	0.523	+0.017	[-0.192, +0.176]	0.790
Spherical Aberration	8	392	0.533	0.542	-0.009	[-0.069, +0.051]	0.624
Demosaicking (Hexadeca)	6	94	0.273	0.280	-0.007	[-0.075, +0.065]	0.718
Demosaicking (Quad)	3	97	0.372	0.354	+0.018	[-0.046, +0.093]	0.760
Demosaicking (Regular)	5	95	0.267	0.250	+0.017	[-0.061, +0.097]	0.667
Denoising	1	99	0.278	0.468	-0.190	–	0.260
Inpainting	3	97	0.341	0.368	-0.027	[-0.114, +0.114]	0.632
Lens Distortion	1	99	0.523	0.499	+0.024	–	0.960
Rand. Masked Subsampling	10	490	0.201	0.217	-0.016	[-0.045, +0.013]	0.519
Super-Resolution	1	99	0.641	0.670	-0.029	–	0.560
Time-of-Flight Depth	6	94	0.312	0.292	+0.020	[-0.012, +0.053]	0.322
All tasks (pooled, per-task z)	53	2147	0.060	-0.002	+0.061	[-0.201, +0.322]	0.697

Table 23: Text-output tasks, scored separately from the image editors: both return text rather than an image, and the planner model answers them. Calibration uses normalized parameter errors; lens design uses normalized Strehl. Scores range from 0 to 1 (higher is better). †GPT-5.4 (ChatGPT).

Planner model	**Camera calibration**	**Lens design**
Gemini-3.1-Pro	0.429	0.016
GPT-5	0.387	0.018†
Qwen3.5-35B-A3B	0.361	0.001
AUTOLENS (specialized)	–	0.029

Table 24: **Camera calibration by camera**: intrinsics MRE (lower is better) and the share of replies that parse. 18 instances per camera.

	iPhone 13 Pro		Sony α7		Stereo	
Planner	MRE	Parsed	MRE	Parsed	MRE	Parsed
Gemini 3.1 Pro	0.020	100%	0.772	100%	0.205	100%
GPT-5	0.068	100%	0.210	100%	0.443	100%
Qwen3.5-35B-A3B	0.087	100%	0.234	94%	0.367	83%

M SAMPLE VISUAL RESULTS AND PROMPTS

Each grid shows, for the same sampled inputs, the inputs, the ground truth, and the outputs of all six editors under Expert, Planner and, where it applies, Forward. A grey *no output* panel marks a job with no returned image; for HunyuanImage-3.0 this covers every input of three or more images (HDR, time-of-flight, and part of block compressive sensing), which exceeds its context limit. The prompts behind each grid are in Sec. M.2.

VISUAL RESULTS CONTENTS

Computer-Generated Holography 36
Block-Wise Compressive Sensing (Ratios 0.031, 0.047, 0.062) 37
Lensless Imaging 40
Longitudinal Chromatic Aberration 41
Motion Deblurring 41
Spherical Aberration (0.10–0.25λ) 42
Demosaicking (Hexadeca) 44
Demosaicking (Quad) 45
Demosaicking (Regular) 46
Denoising (SIDD) 47
Denoising Ablation (Synthetic Noise Sweeps) ... 48
Event-Based Reconstruction 49
HDR Imaging 49
Inpainting 50
Lens Distortion Correction 51
Light-Field View Extrapolation 52
Light-Field View Interpolation 52
Random Mask Subsampling (10–80% of Pixels) . 53
Super-Resolution ($\times 4$) 56
Time-of-Flight Depth 57
White Balance 58
Lens Design 59

PROMPTS CONTENTS

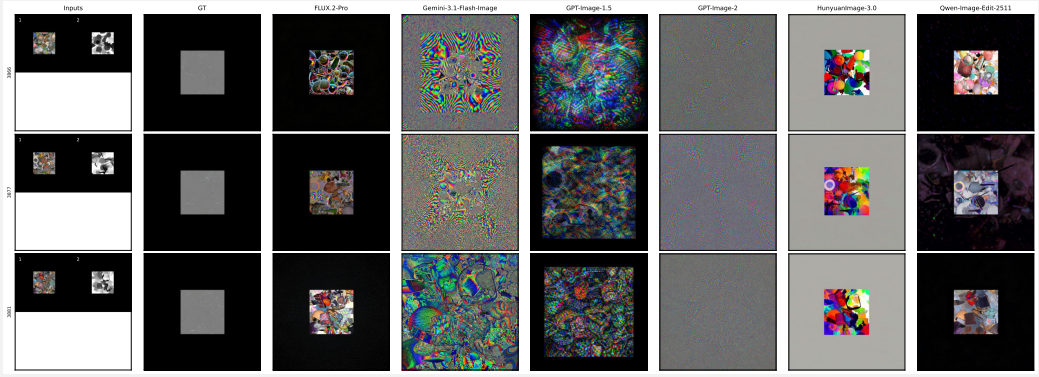
Computer-Generated Holography 63
Block-Wise Compressive Sensing (Ratios 0.031, 0.047, 0.062) 65
Lensless Imaging 67
Longitudinal Chromatic Aberration 70
Motion Deblurring 71
Spherical Aberration (0.10–0.25λ) 72
Demosaicking (Hexadeca) 73
Demosaicking (Quad) 74
Demosaicking (Regular) 75
Denoising (SIDD) 76
Denoising Ablation (Synthetic Noise Sweeps) ... 78
Event-Based Reconstruction 79
HDR Imaging 80
Inpainting 82
Lens Distortion Correction 83
Light-Field View Extrapolation 84
Light-Field View Interpolation 85
Random Mask Subsampling (10–80% of Pixels) . 86
Super-Resolution ($\times 4$) 87
Time-of-Flight Depth 88
White Balance 90

M.1 VISUAL RESULTS

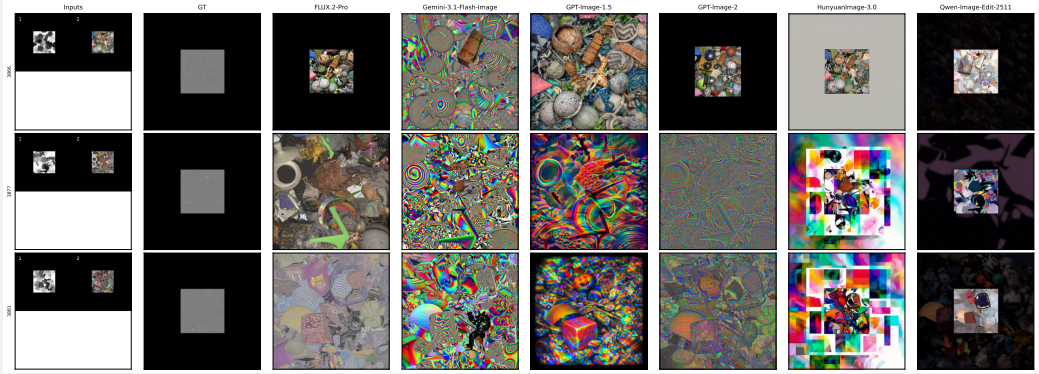
Computer-Generated Holography

prompts: p. 63

EXPERT RGB phase-only hologram generation from an RGB image and depth map.

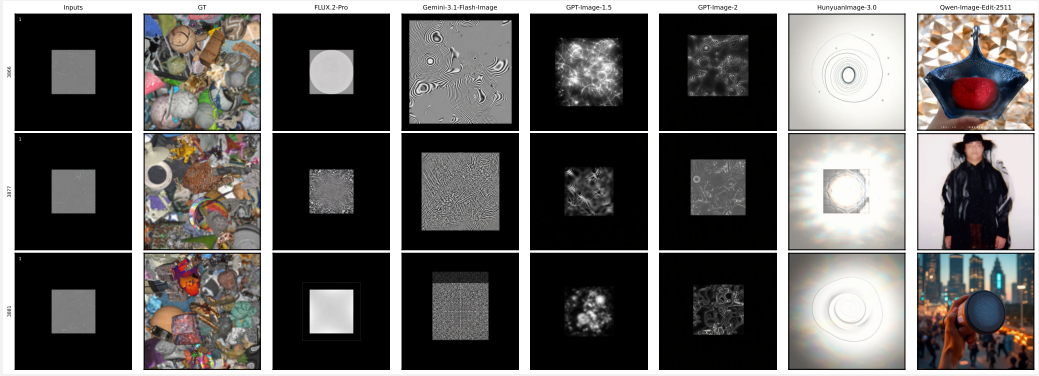


PLANNER RGB phase-only hologram generation from an RGB image and depth map. Prompt written per image by the planner.



Row 1: GPT-5 declined to write a prompt for this sample, so GPT-Image-1.5, GPT-Image-2, and HunyuanImage-3.0 received its refusal as their instruction.

FORWARD Diffraction-pattern generation from a phase map under a coherent propagation model. Prompt written per image by the planner.



Block-Wise Compressive Sensing (Ratios 0.031, 0.047, 0.062)

prompts: p. 65

EXPERT Reconstruction from two blockwise compressive sensing measurements under a sparse binary mask setting.

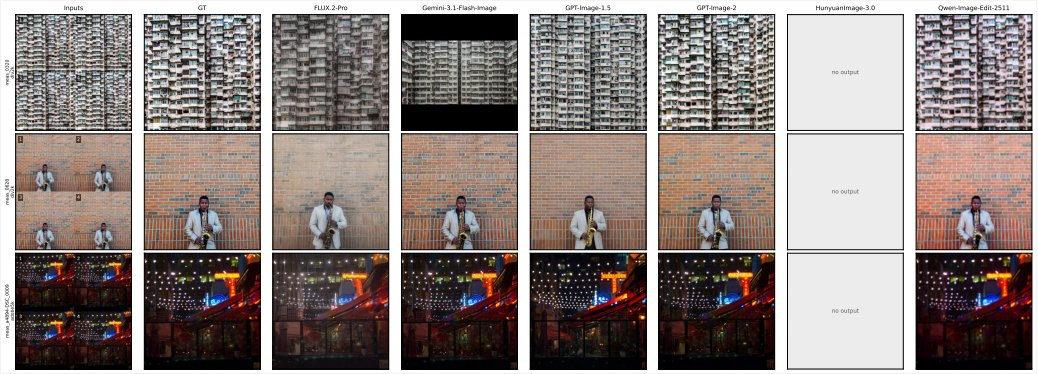
- **Ratio 0.031**



- **Ratio 0.047**

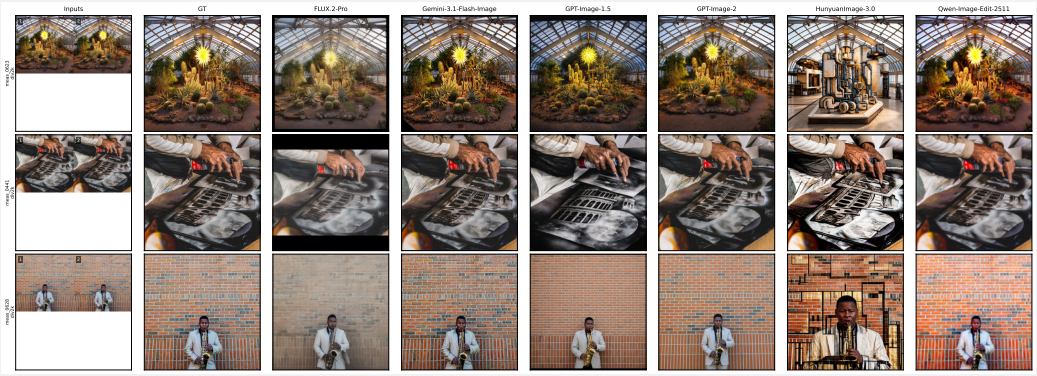


- **Ratio 0.062**

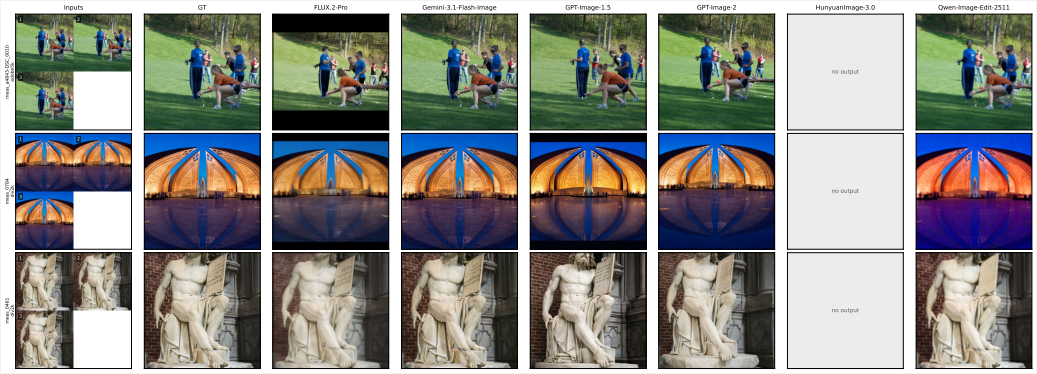


PLANNER Reconstruction from two blockwise compressive sensing measurements under a sparse binary mask setting. Prompt written per image by the planner.

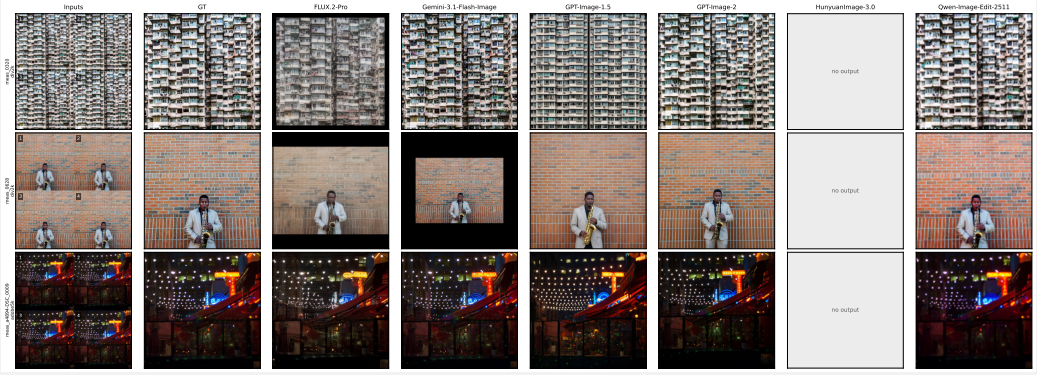
- **Ratio 0.031**



- **Ratio 0.047**

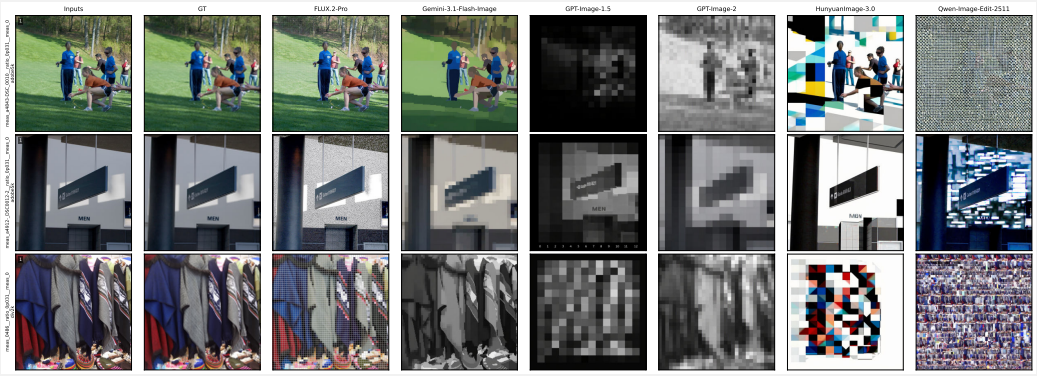


- **Ratio 0.062**

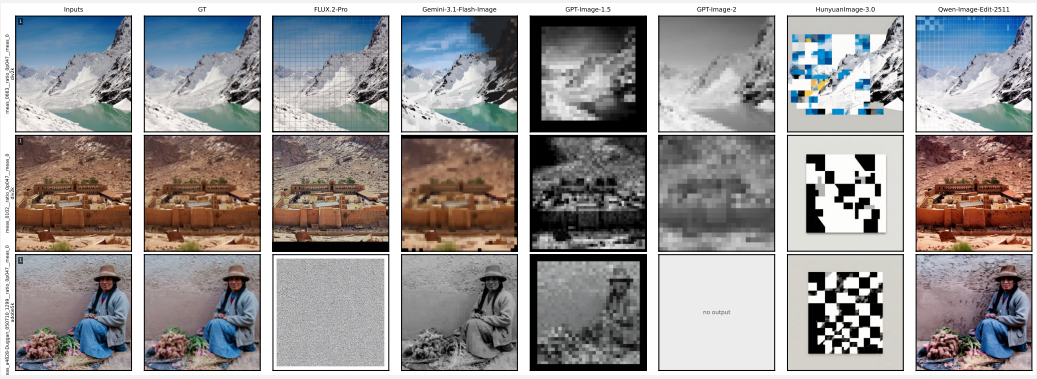


FORWARD Generation of a single blockwise compressive sensing measurement image from an RGB input. Prompt written per image by the planner.

- **Ratio 0.031**



- **Ratio 0.047**



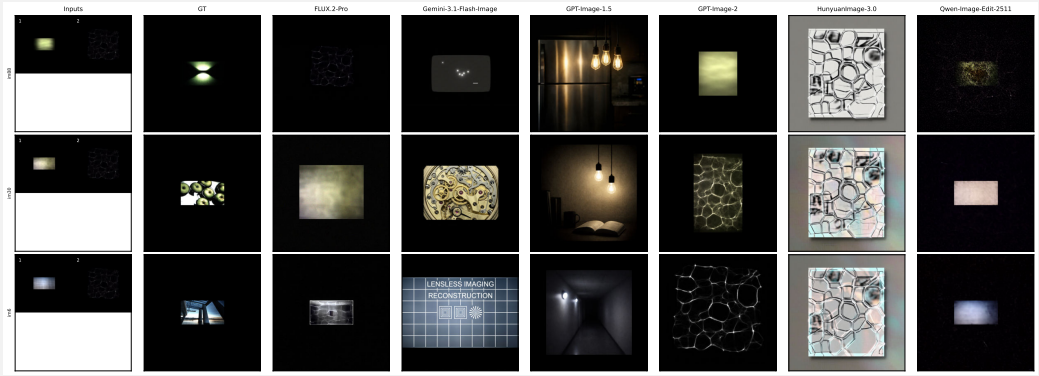
- **Ratio 0.062**



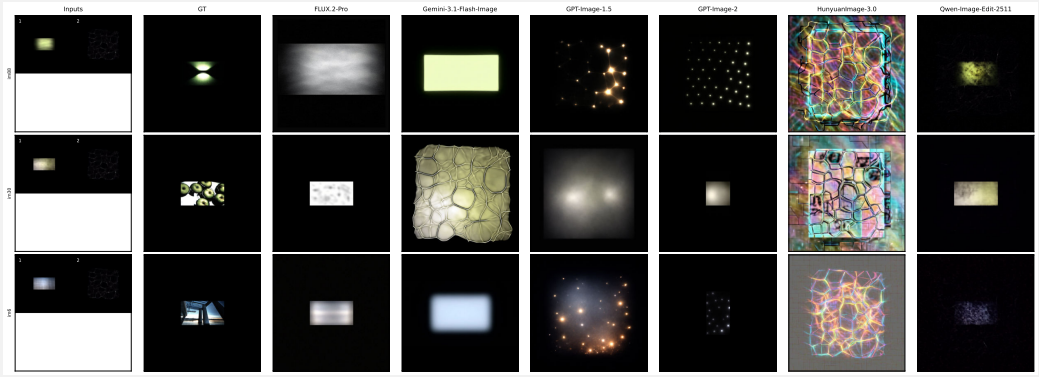
Lensless Imaging

prompts: p. 67

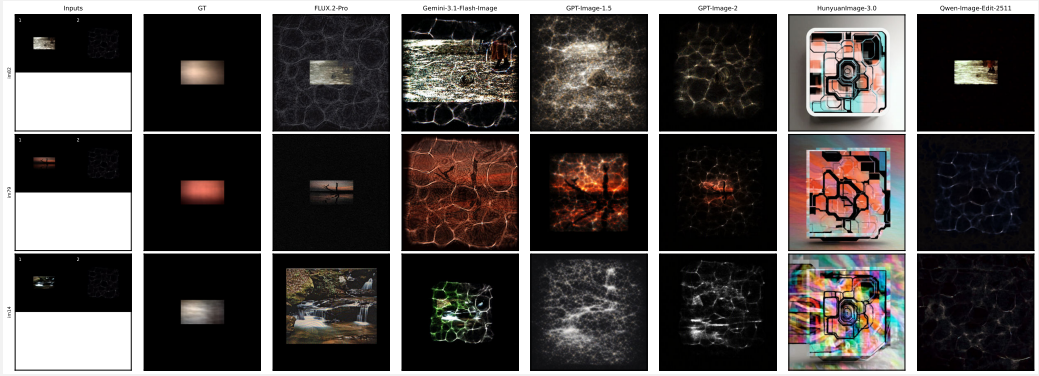
EXPERT Reconstruction of a clear scene from a lensless coded measurement and its point-spread function.



PLANNER Reconstruction of a clear scene from a lensless coded measurement and its point-spread function. Prompt written per image by the planner.



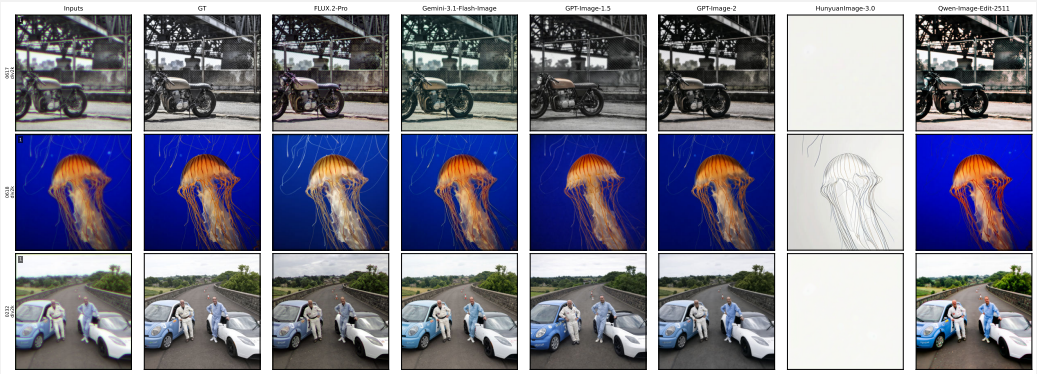
FORWARD Synthesis of a lensless multiplexed sensor measurement from a clean image and a point-spread function. Prompt written per image by the planner.



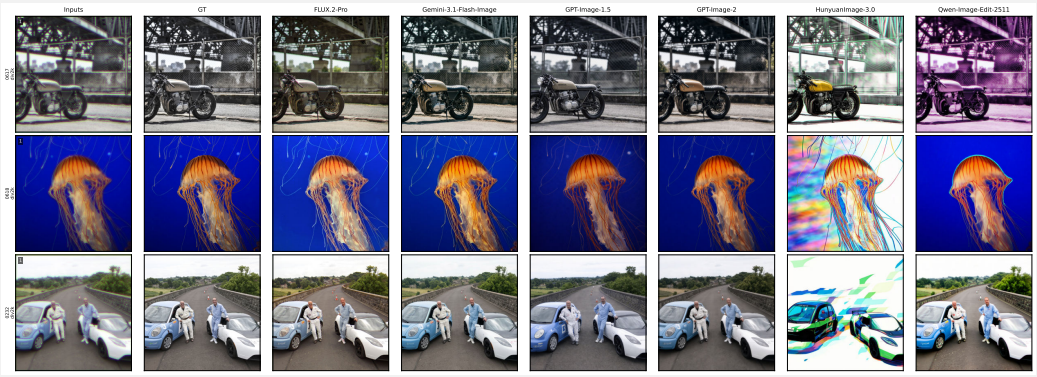
Longitudinal Chromatic Aberration

prompts: p. 70

EXPERT Correction of color-dependent blur caused by longitudinal chromatic aberration.



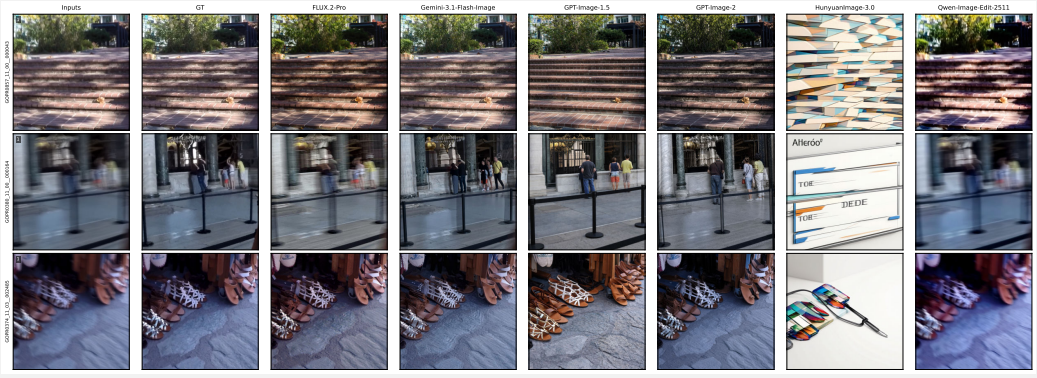
PLANNER Correction of color-dependent blur caused by longitudinal chromatic aberration. Prompt written per image by the planner.



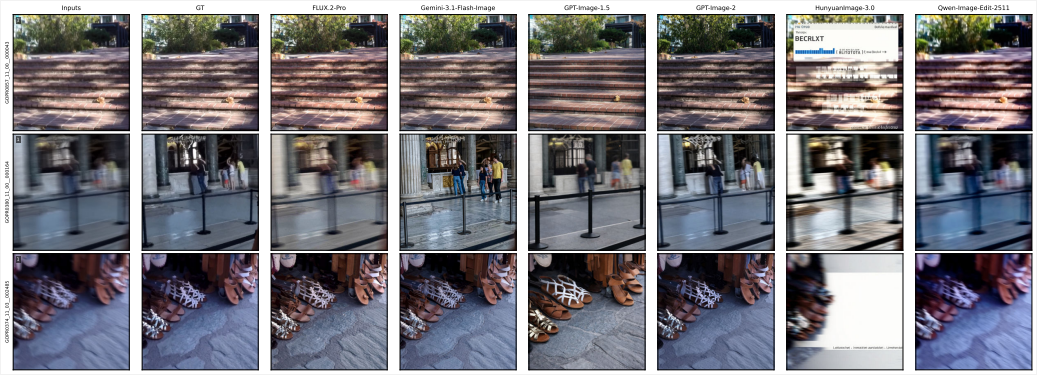
Motion Deblurring

prompts: p. 71

EXPERT Reconstruction of a sharp image from a motion-blurred observation.



PLANNER Reconstruction of a sharp image from a motion-blurred observation. Prompt written per image by the planner.

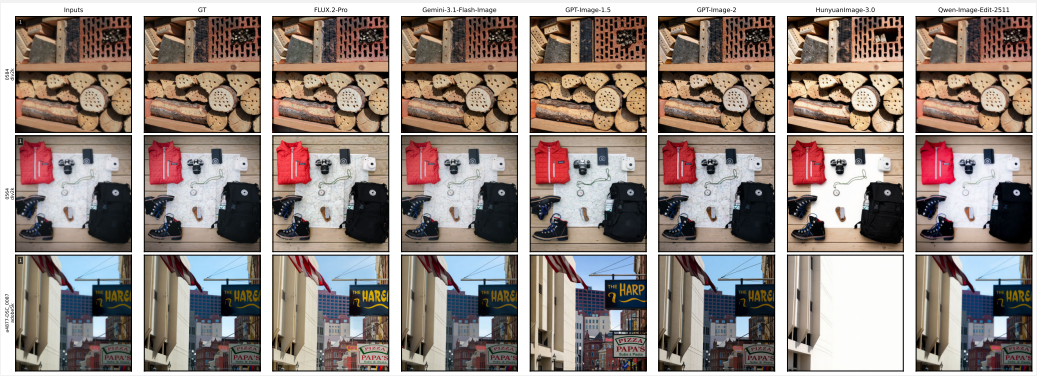


Spherical Aberration (0.10–0.25λ)

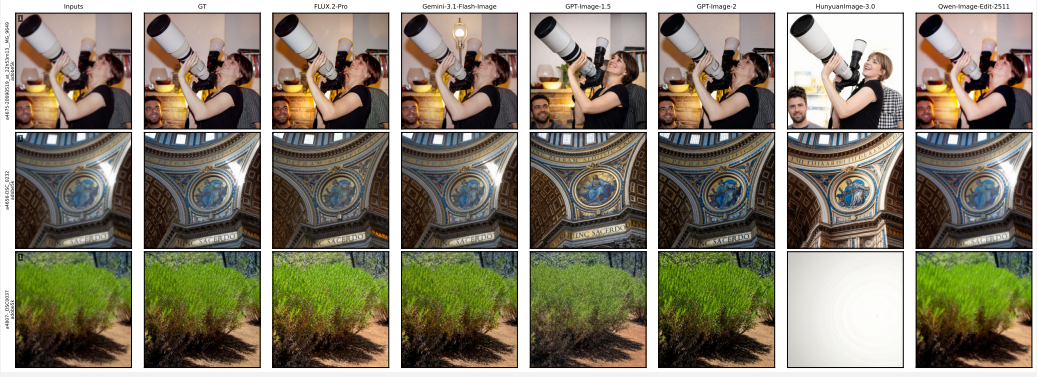
prompts: p. 72

EXPERT Correction of blur caused by spherical aberration at a fixed aberration strength.

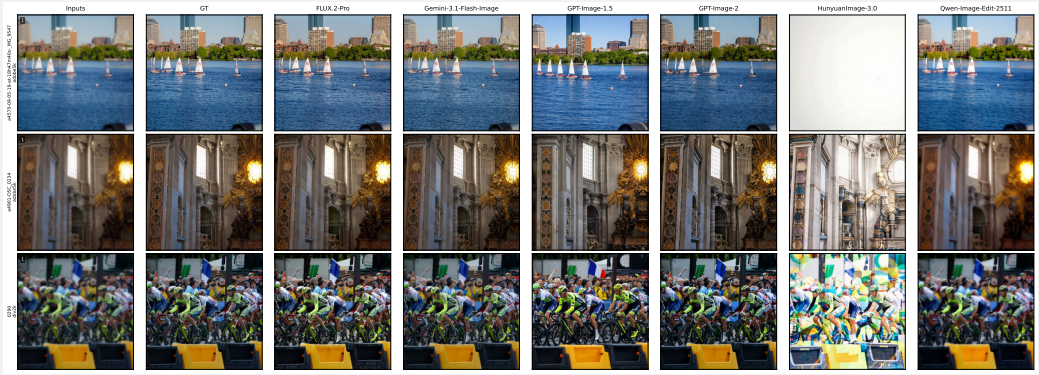
- **Aberration 0.1**



- **Aberration 0.15**



• **Aberration 0.2**

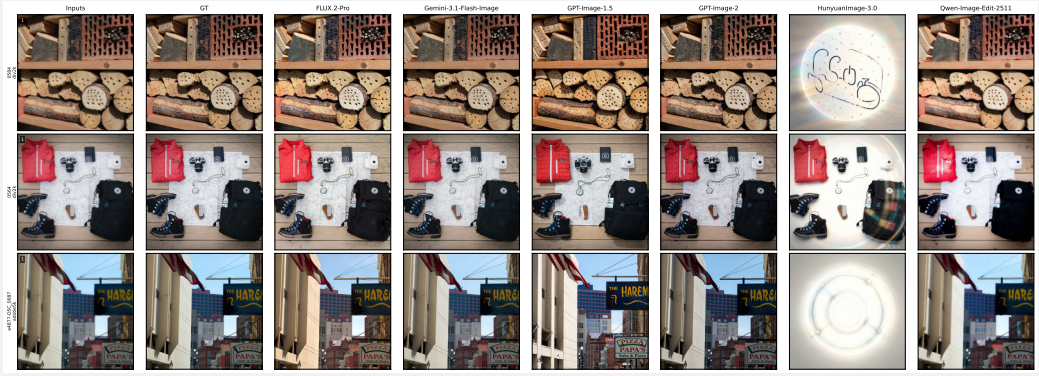


• **Aberration 0.25**

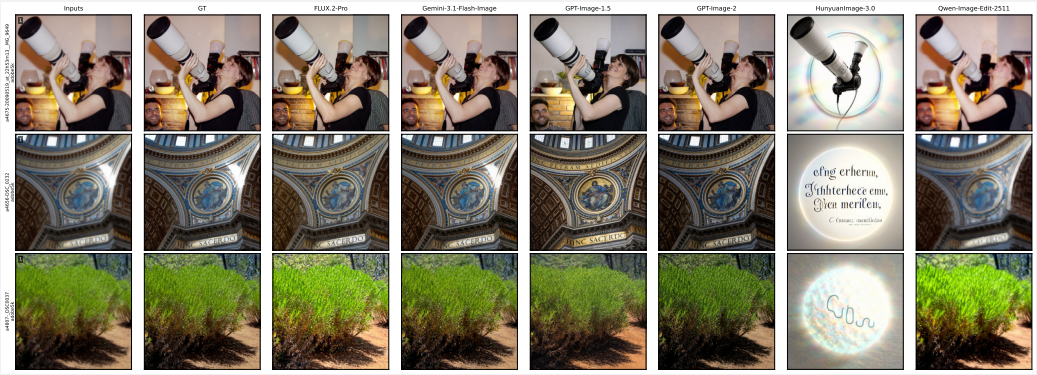


PLANNER Correction of blur caused by spherical aberration at a fixed aberration strength. Prompt written per image by the planner.

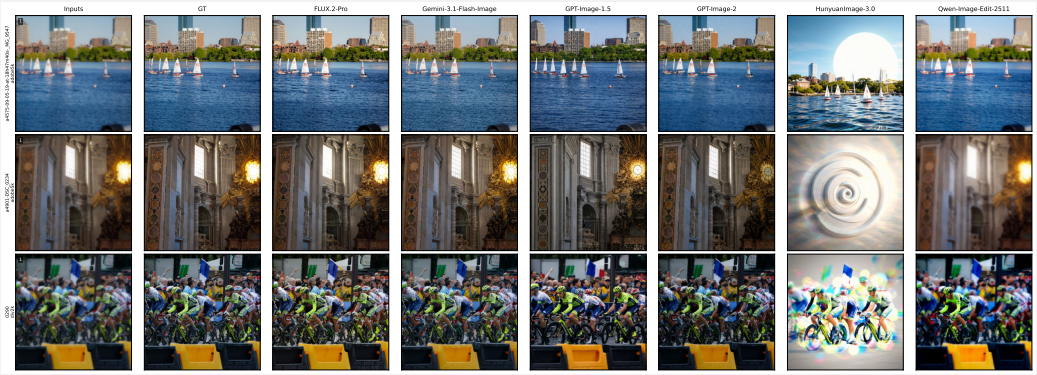
• **Aberration 0.1**



• **Aberration 0.15**



• **Aberration 0.2**



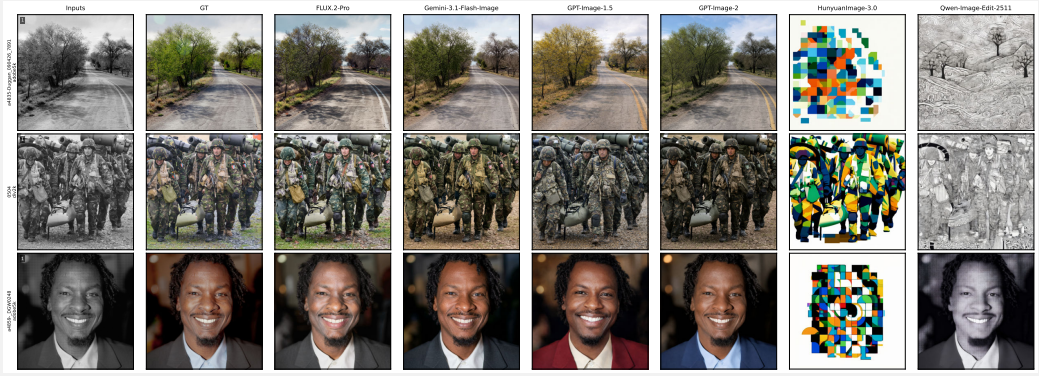
• **Aberration 0.25**



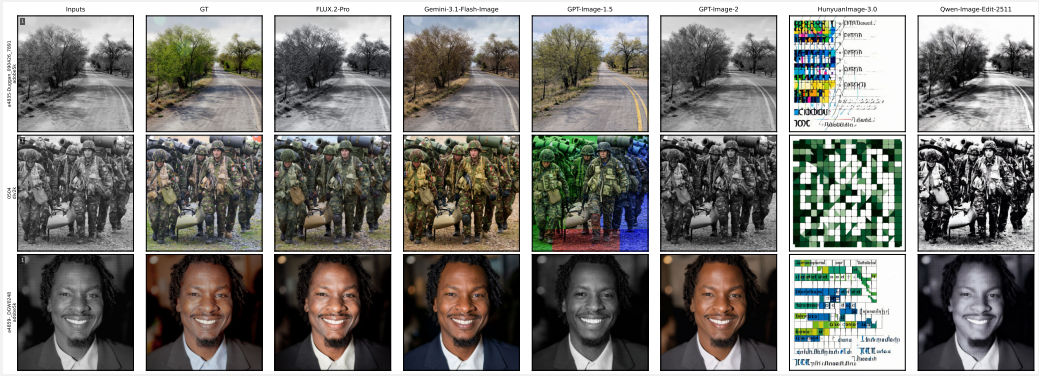
Demosaicking (Hexadeca)

prompts: p. 73

EXPERT Full-color image reconstruction from a sparse hexadeca sampling pattern.



PLANNER Full-color image reconstruction from a sparse hexadeca sampling pattern. Prompt written per image by the planner.



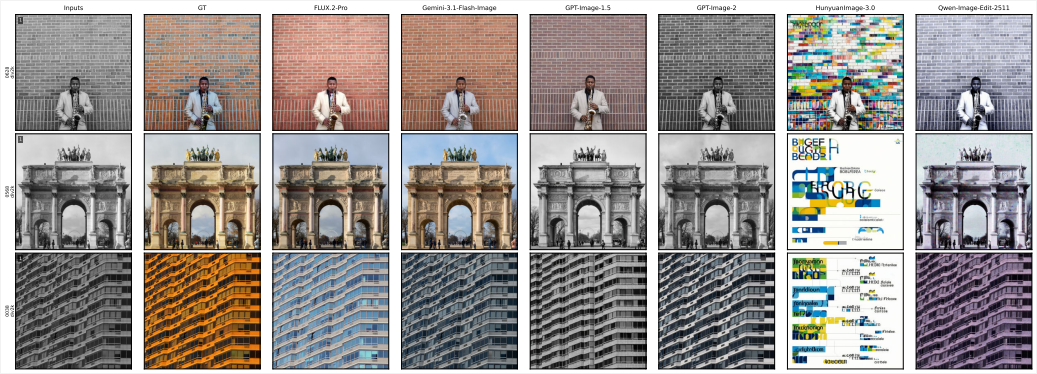
Demosaicking (Quad)

prompts: p. 74

EXPERT Full-color image reconstruction from a quad Bayer sampling pattern.



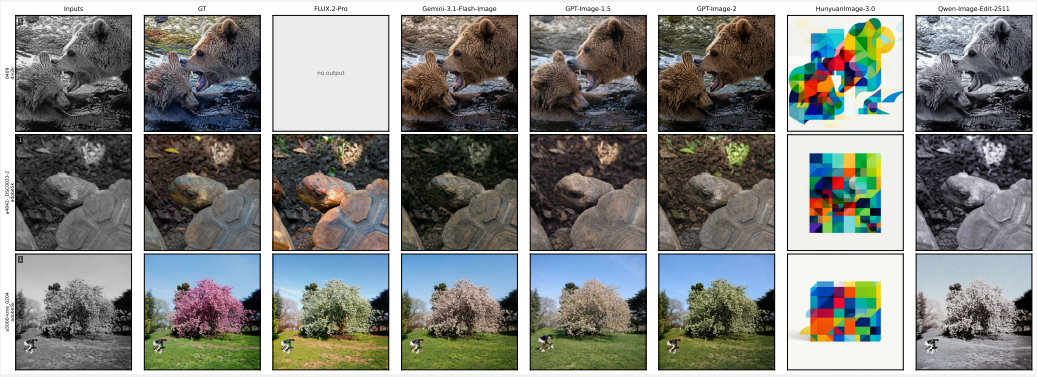
PLANNER Full-color image reconstruction from a quad Bayer sampling pattern. Prompt written per image by the planner.



Demosaicking (Regular)

prompts: p. 75

EXPERT Full-color image reconstruction from a regular Bayer sampling pattern.



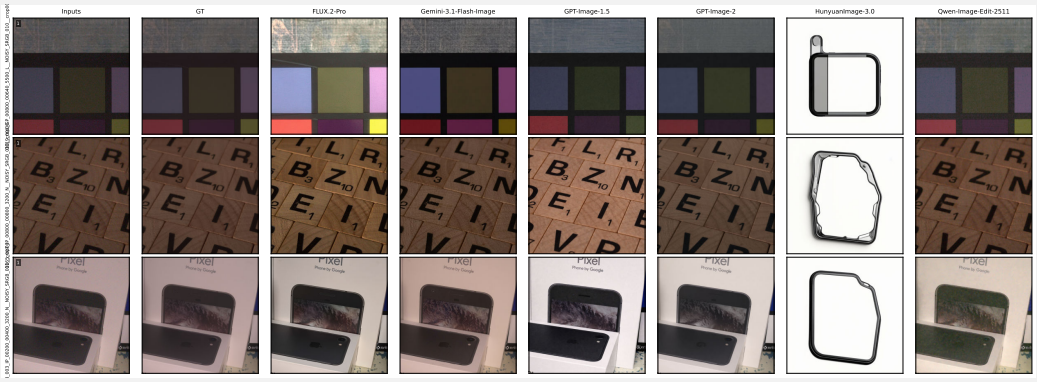
PLANNER Full-color image reconstruction from a regular Bayer sampling pattern. Prompt written per image by the planner.



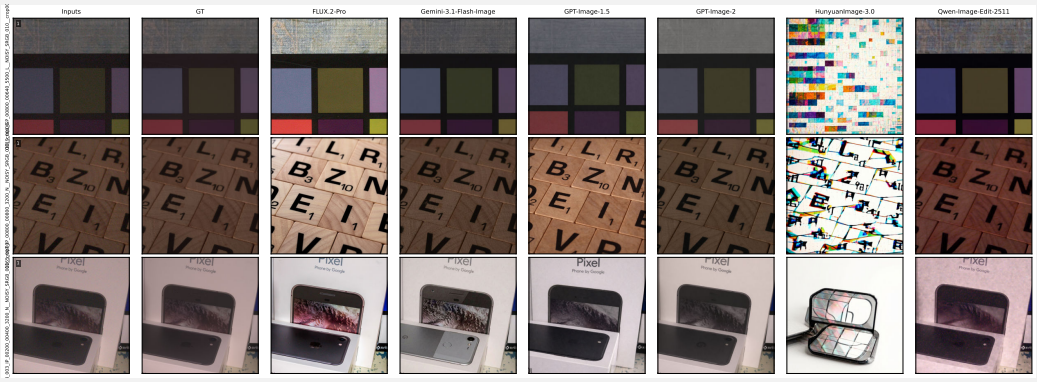
Denoising (SIDD)

prompts: p. 76

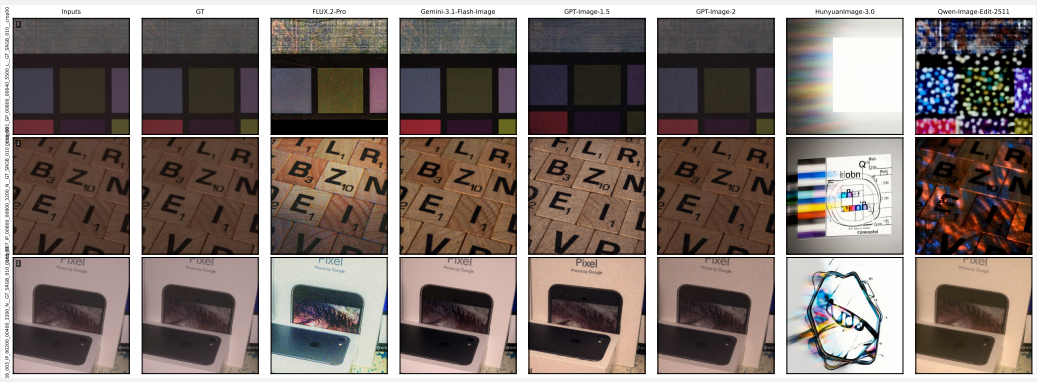
EXPERT Removal of smartphone sensor noise while preserving fine texture and color.



PLANNER Removal of smartphone sensor noise while preserving fine texture and color. Prompt written per image by the planner.



FORWARD Realistic smartphone noise synthesis from a clean reference image. Prompt written per image by the planner.

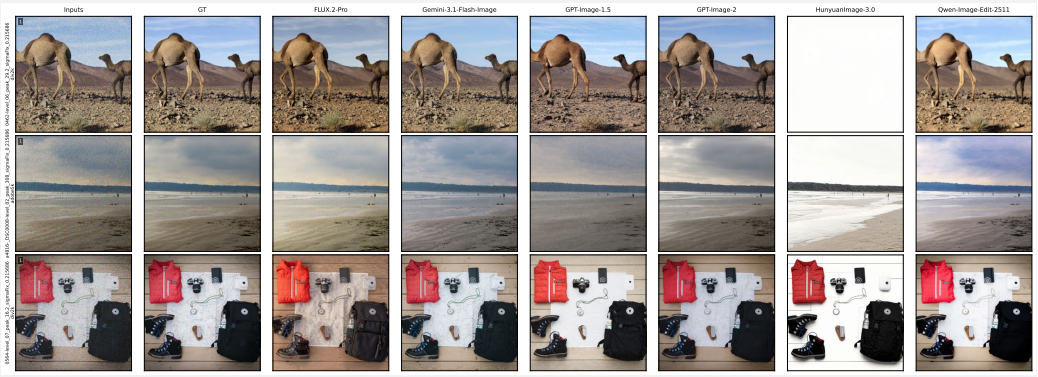


Denoising Ablation (Synthetic Noise Sweeps)

prompts: p. 78

EXPERT Removal of synthetic sensor noise at ten severity levels (the Expert ablation sweeps).

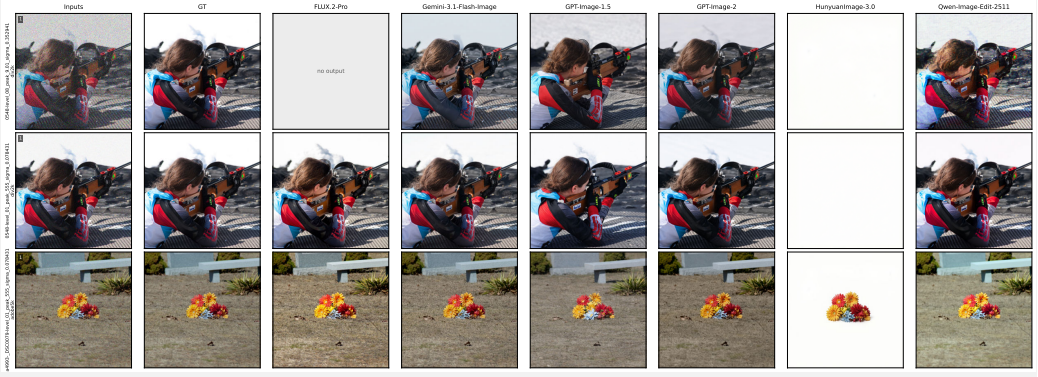
- **Shot noise**



- **Read noise**



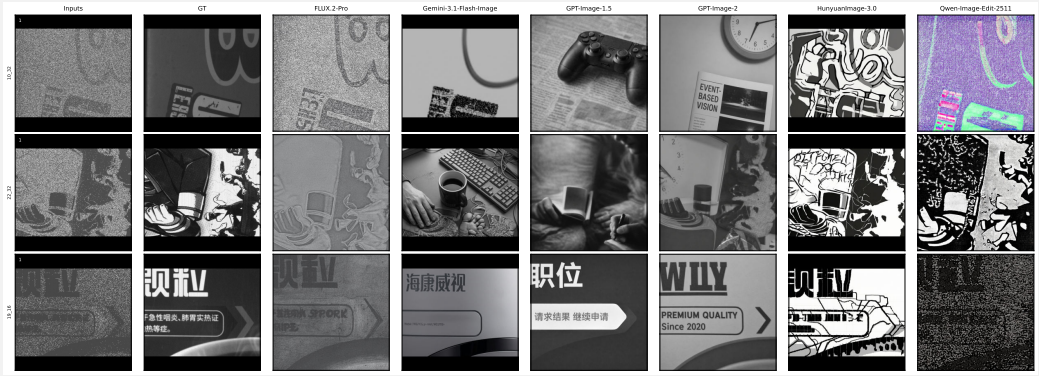
- **Mixed noise**



Event-Based Reconstruction

prompts: p. 79

EXPERT Intensity-image reconstruction from event-camera measurements.



PLANNER Intensity-image reconstruction from event-camera measurements. Prompt written per image by the planner.



HDR Imaging

prompts: p. 80

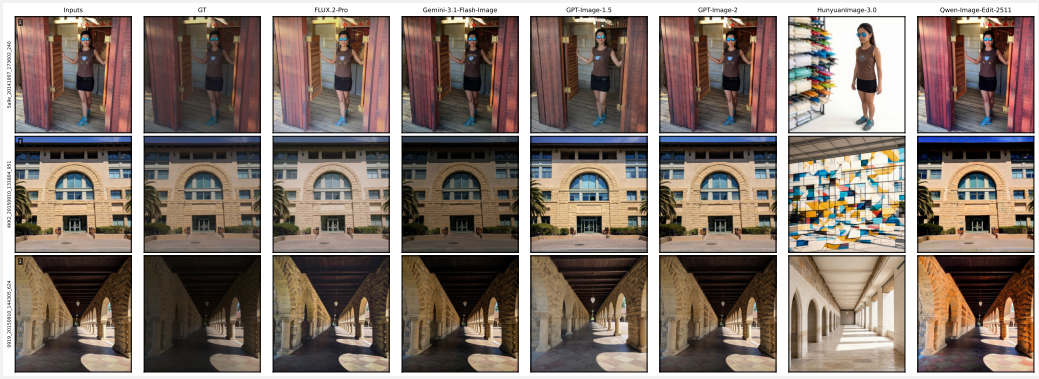
EXPERT Burst HDR fusion for preserving both highlight and shadow detail.



PLANNER Burst HDR fusion for preserving both highlight and shadow detail. Prompt written per image by the planner.



FORWARD Generation of a plausible single HDR+ burst frame from a reference HDR image. Prompt written per image by the planner.



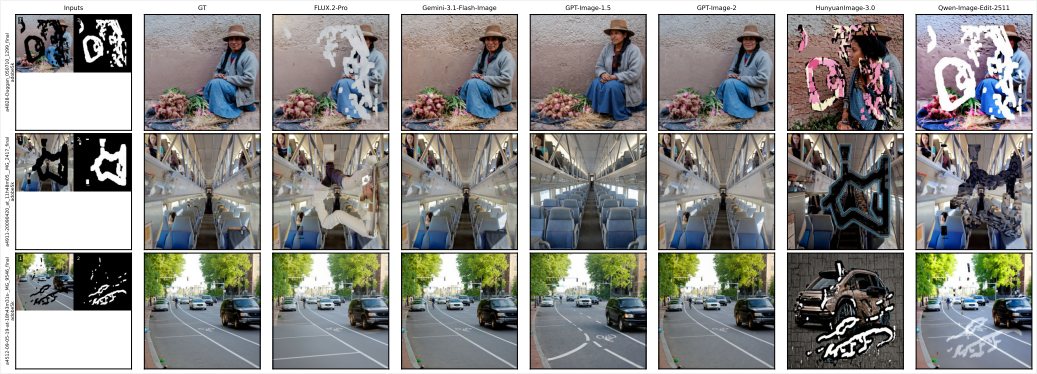
Inpainting

prompts: p. 82

EXPERT Masked image completion while preserving scene structure and texture continuity.



PLANNER Masked image completion while preserving scene structure and texture continuity. Prompt written per image by the planner.



Lens Distortion Correction

prompts: p. 83

EXPERT Geometric rectification of a lens-distorted photograph, straightening curved lines while preserving true perspective.



PLANNER Geometric rectification of a lens-distorted photograph, straightening curved lines while preserving true perspective. Prompt written per image by the planner.



Light-Field View Extrapolation

prompts: p. 84

EXPERT Synthesis of the right-side sub-aperture view from the left and center views.



PLANNER Synthesis of the right-side sub-aperture view from the left and center views. Prompt written per image by the planner.



Light-Field View Interpolation

prompts: p. 85

EXPERT Synthesis of the missing center sub-aperture view from the left and right views.



PLANNER Synthesis of the missing center sub-aperture view from the left and right views. Prompt written per image by the planner.

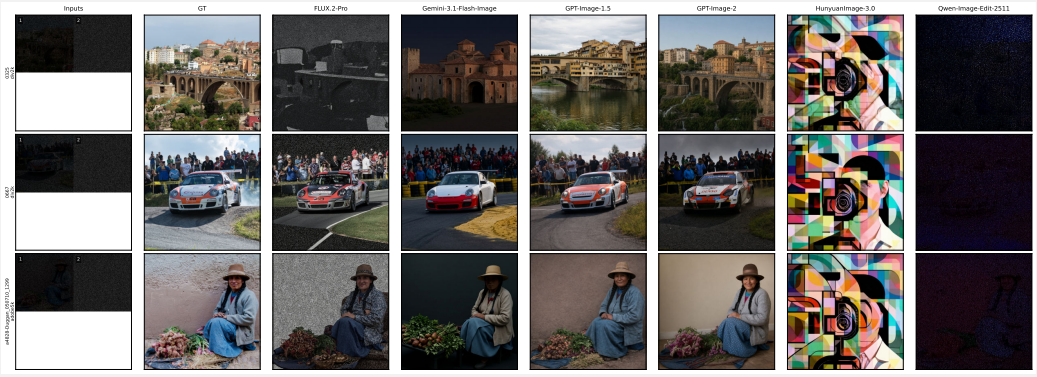


Random Mask Subsampling (10–80% of Pixels)

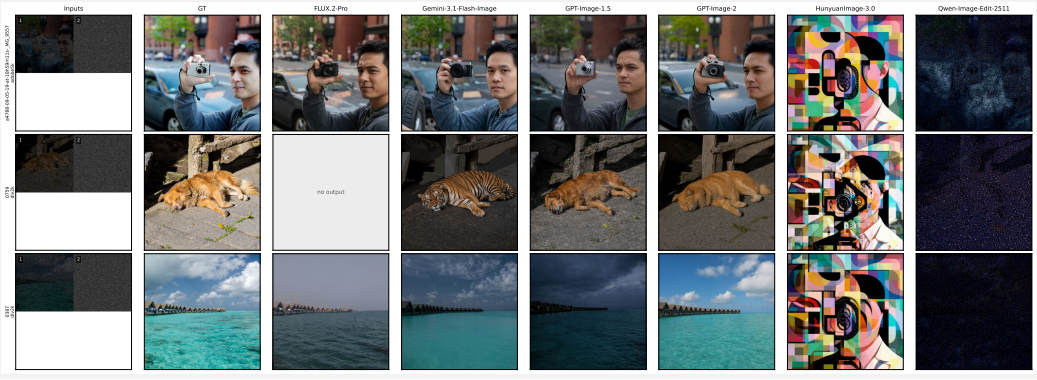
prompts: p. 86

EXPERT Reconstruction from randomly masked measurements keeping 10–80% of pixels.

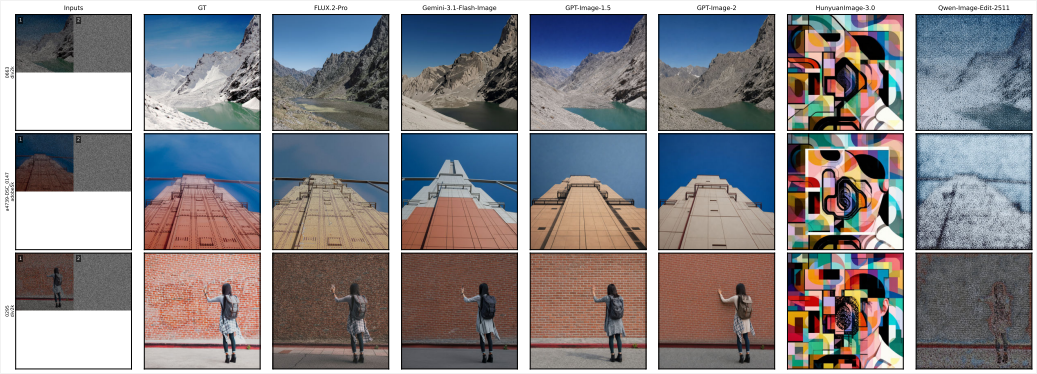
- **10% of pixels**



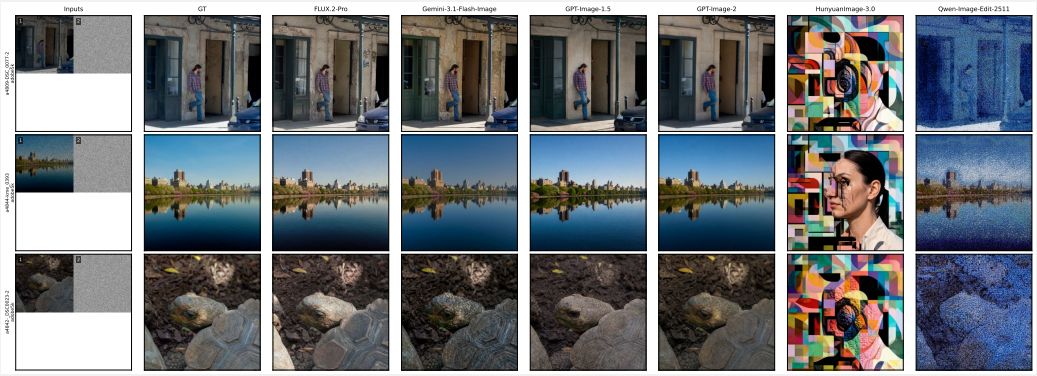
- **20% of pixels**



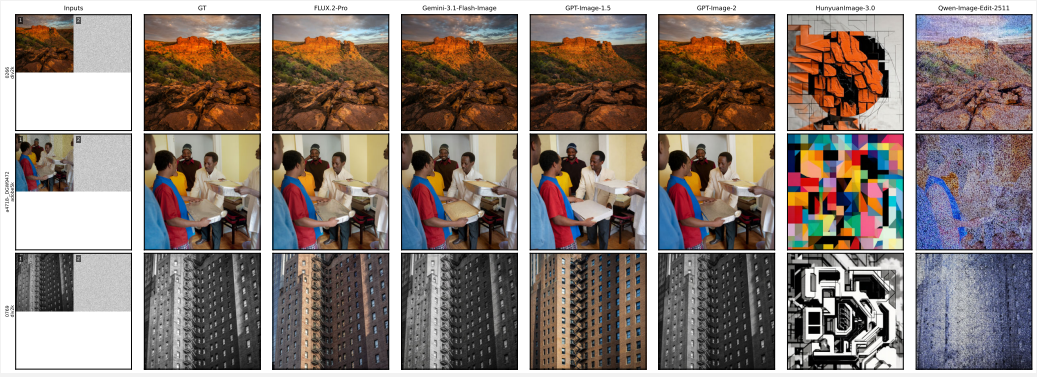
- **40% of pixels**



- **60% of pixels**

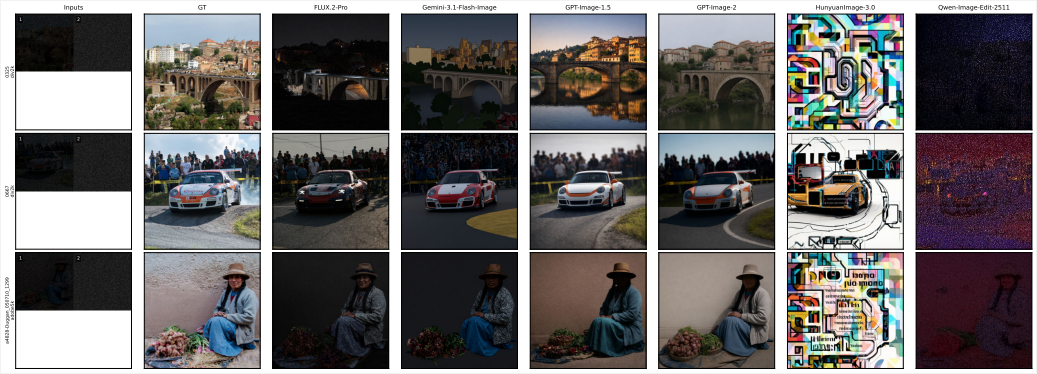


- **80% of pixels**

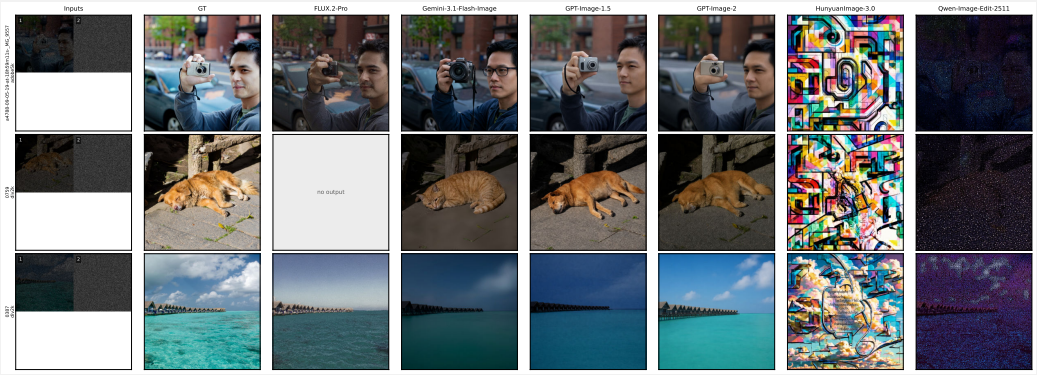


PLANNER Reconstruction from randomly masked measurements keeping 10–80% of pixels. Prompt written per image by the planner.

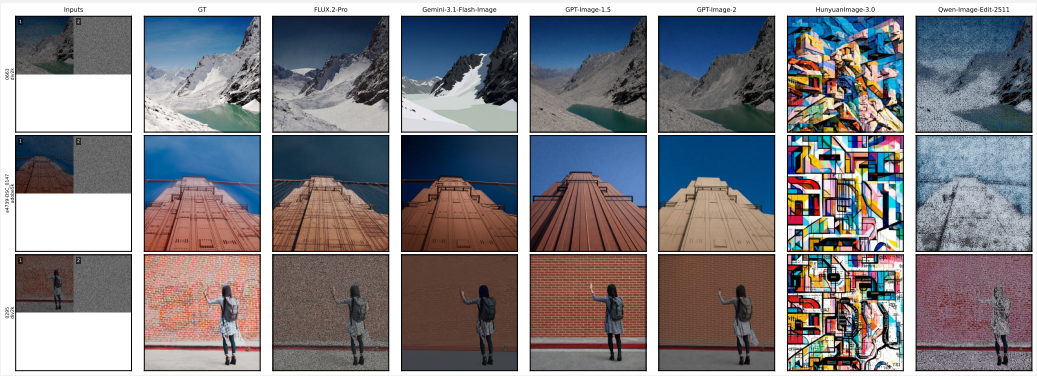
- **10% of pixels**



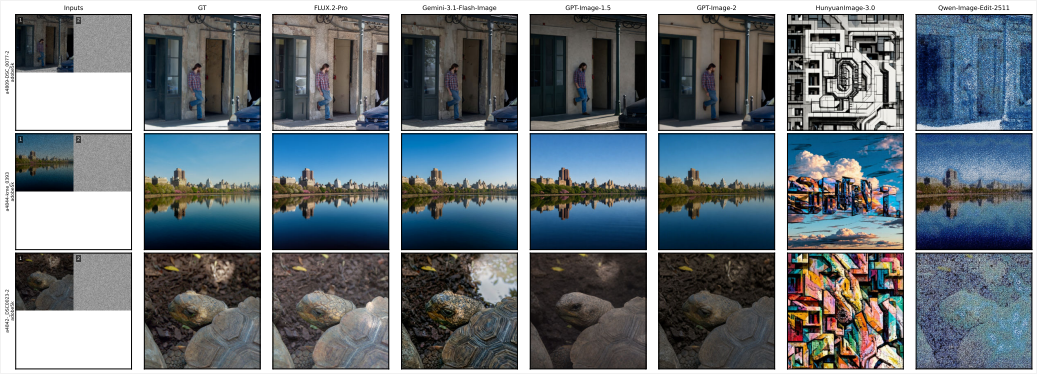
- **20% of pixels**



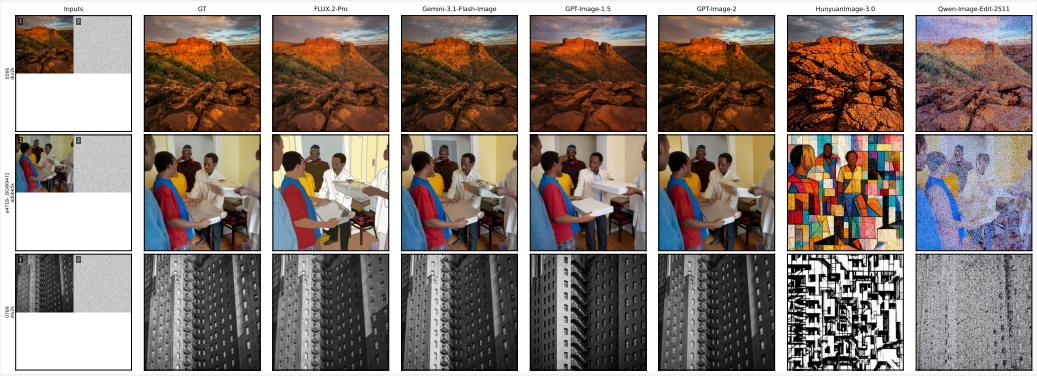
- **40% of pixels**



• **60% of pixels**



• **80% of pixels**



Super-Resolution ($\times 4$)

prompts: p. 87

EXPERT 4x image super-resolution from a lower-resolution observation.



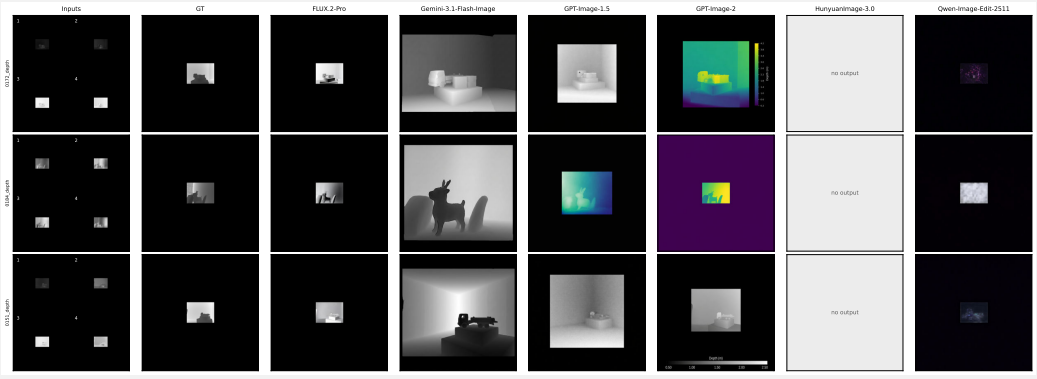
PLANNER 4x image super-resolution from a lower-resolution observation. Prompt written per image by the planner.



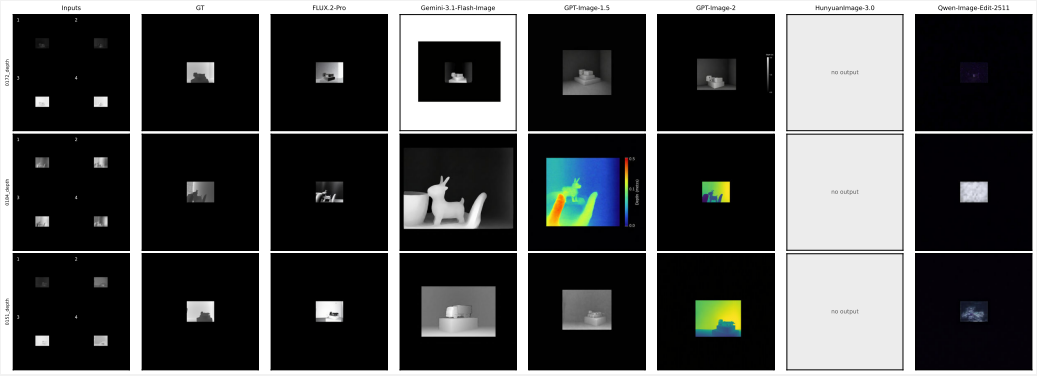
Time-of-Flight Depth

prompts: p. 88

EXPERT Scene reconstruction from time-of-flight imaging measurements.



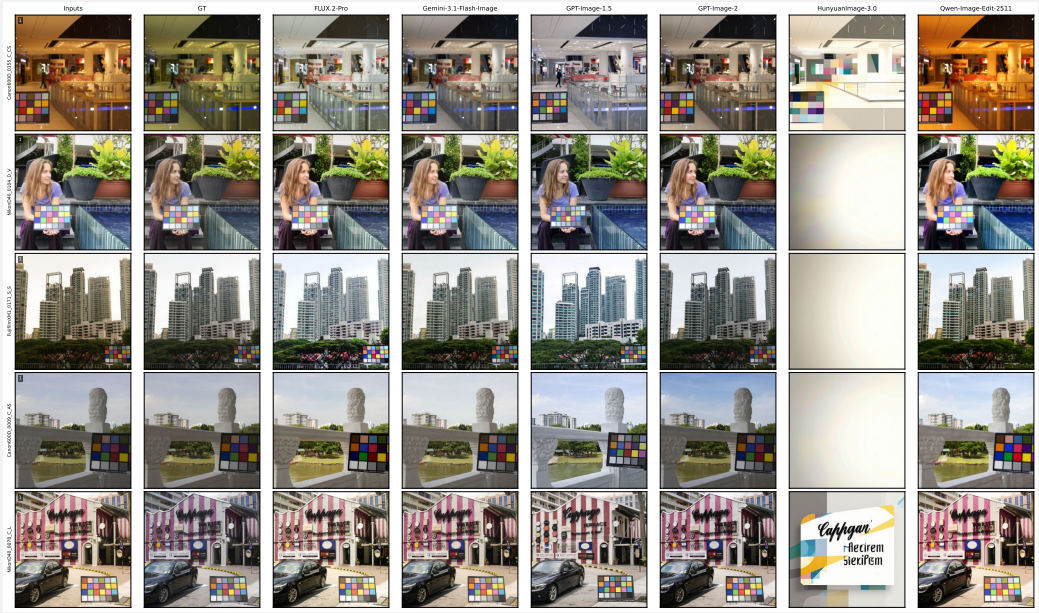
PLANNER Scene reconstruction from time-of-flight imaging measurements. Prompt written per image by the planner.



White Balance

prompts: p. 90

EXPERT Illumination correction to recover a naturally white-balanced image.



PLANNER Illumination correction to recover a naturally white-balanced image. Prompt written per image by the planner.



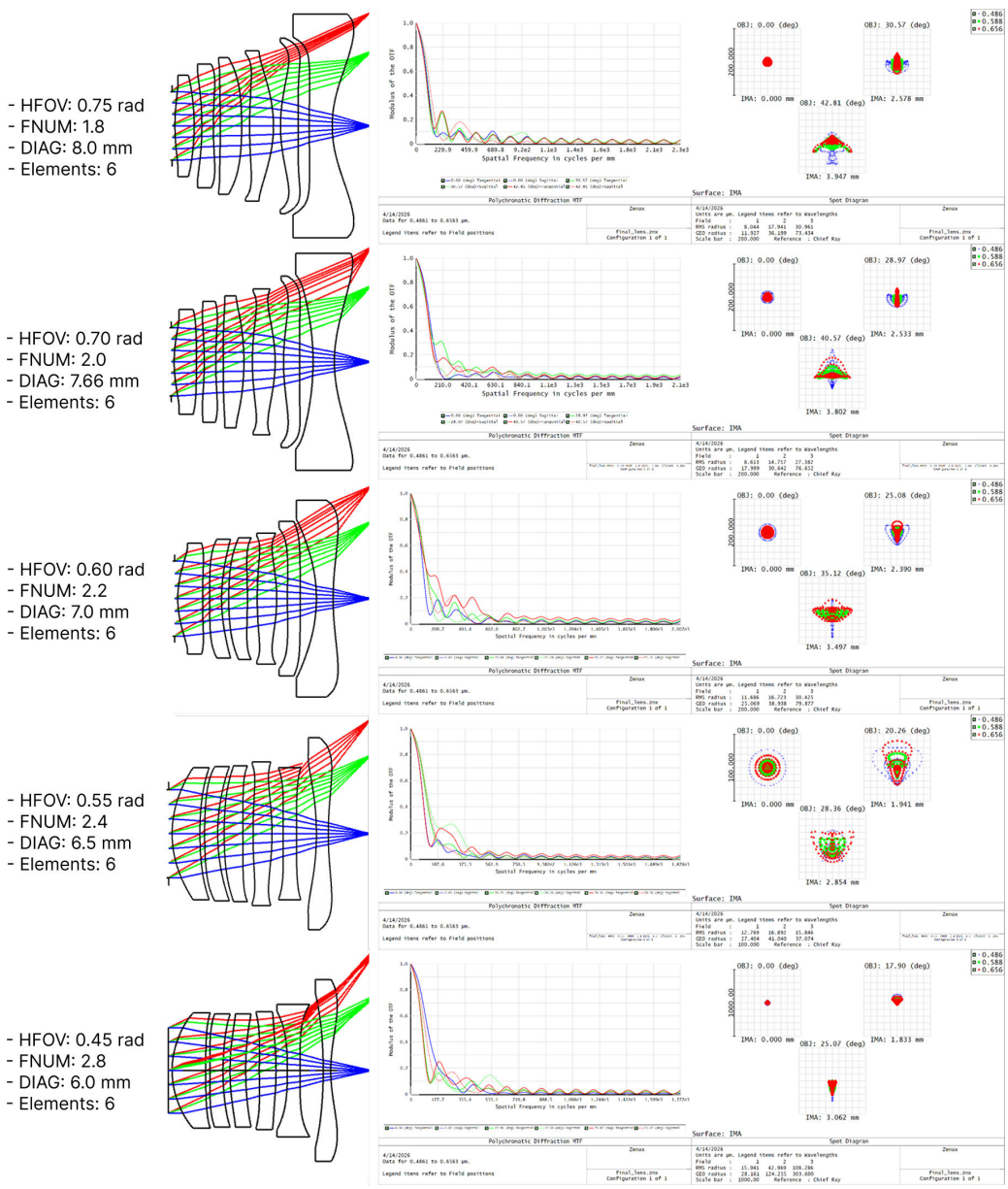


Figure 9: **AutoLens (Yang et al., 2024) results across five 6-element mobile lens design targets.** From top to bottom, the design difficulty is progressively reduced by jointly relaxing the half field of view, aperture, and sensor size: $(0.75\,\text{rad}, F/1.8, 8.0\,\text{mm})$, $(0.70\,\text{rad}, F/2.0, 7.66\,\text{mm})$, $(0.60\,\text{rad}, F/2.2, 7.0\,\text{mm})$, $(0.55\,\text{rad}, F/2.4, 6.5\,\text{mm})$, and $(0.45\,\text{rad}, F/2.8, 6.0\,\text{mm})$. Each row shows the optimized lens layout, MTF curves, and spot diagrams produced by AutoLens.

Lens Design

Prompt. Each model receives the same prompt template, instantiated for five 6-element mobile-lens targets. Only the `Target` block varies across the five; the shared template is:

```
You are an optical lens designer.
Design a complete sequential Zemax OpticStudio lens prescription for a legacy .ZMX file.
Target:  HFOV: {HFOV} rad; FNUM: {FNUM}; DIAG: {DIAG} mm; Elements:  6.
```

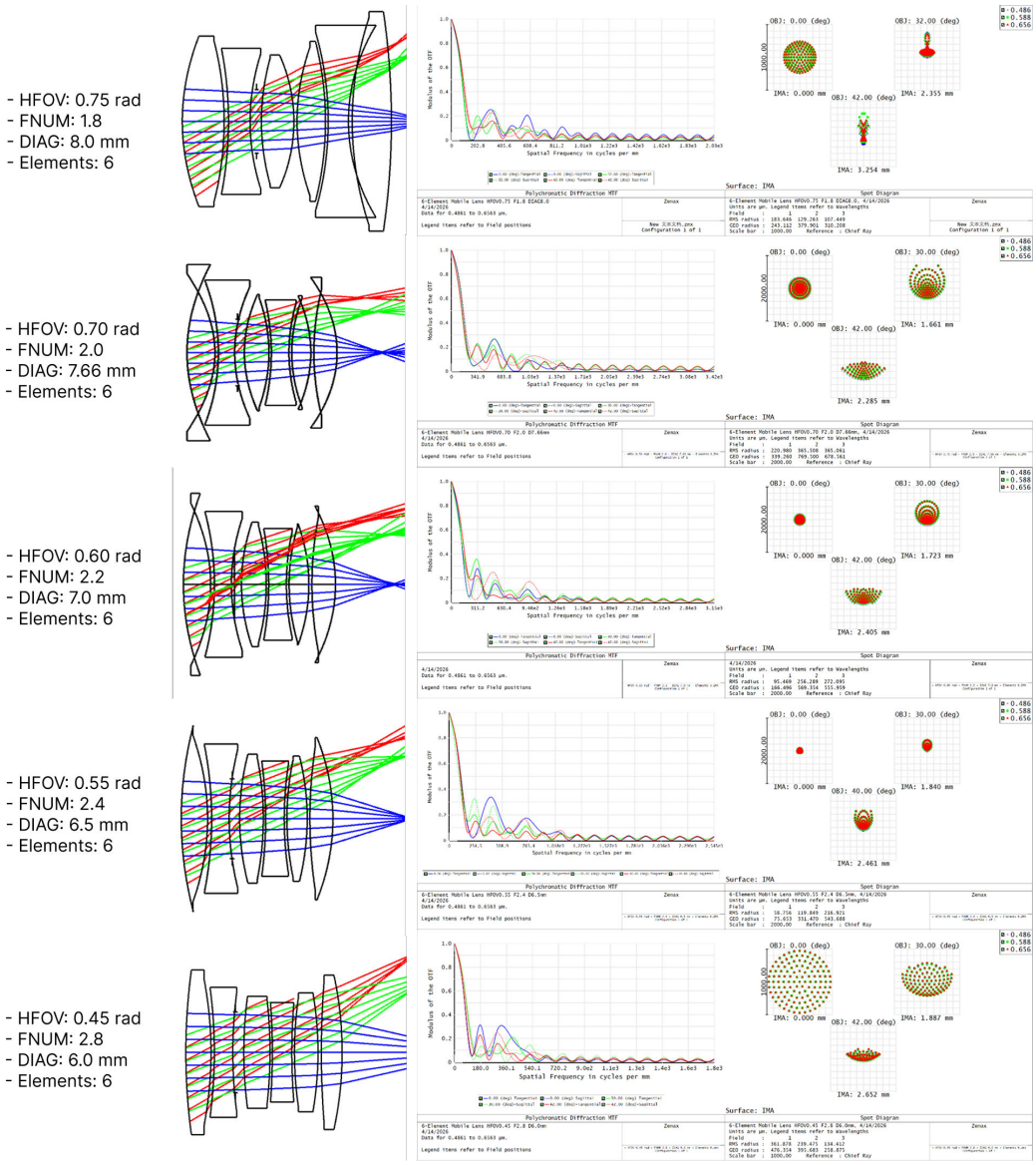


Figure 10: **GPT-5.4 results across five 6-element mobile lens design targets.** From top to bottom, the design difficulty is progressively reduced by jointly relaxing the half field of view, aperture, and sensor size: $(0.75\,\text{rad}, F/1.8, 8.0\,\text{mm})$, $(0.70\,\text{rad}, F/2.0, 7.66\,\text{mm})$, $(0.60\,\text{rad}, F/2.2, 7.0\,\text{mm})$, $(0.55\,\text{rad}, F/2.4, 6.5\,\text{mm})$, and $(0.45\,\text{rad}, F/2.8, 6.0\,\text{mm})$. Each row shows the optimized lens layout, MTF curves, and spot diagrams produced by GPT-5.4.

```
Constraints: Sequential surfaces only; use Even Asphere where needed; each aspheric surface
must include conic + 6 asphere coefficients; use only these materials: AIR, OKP4, COC, PMMA,
POLYSTYR, POLYCARB; include a valid stop surface, aperture definition, and final image
surface; use physically plausible air gaps, thicknesses, and positive semi-diameters; compact
mobile-camera-like layout; no overlapping surfaces; no impossible geometry.
System settings: Wavelengths: 0.4861327 um, 0.5875618 um, 0.6562725 um; primary wavelength:
0.5875618 um; at least 3 field points (one on-axis and two off-axis); field and wavelength
settings must support clear 3-ray colored visualization in OpticStudio 3D Layout.
Output: output only Zemax sequential lens prescription text; no markdown, code fence, or
explanation; no JSON, YAML, Python, or pseudocode; write every surface explicitly; return only
plain-text sequential lens prescription content in Zemax-compatible syntax.
```

The five `(HFOV rad, FNUM, DIAG mm)` targets are $(0.45, 2.8, 6.0)$, $(0.55, 2.4, 6.5)$, $(0.60, 2.2, 7.0)$, $(0.70, 2.0, 7.66)$, and $(0.75, 1.8, 8.0)$.

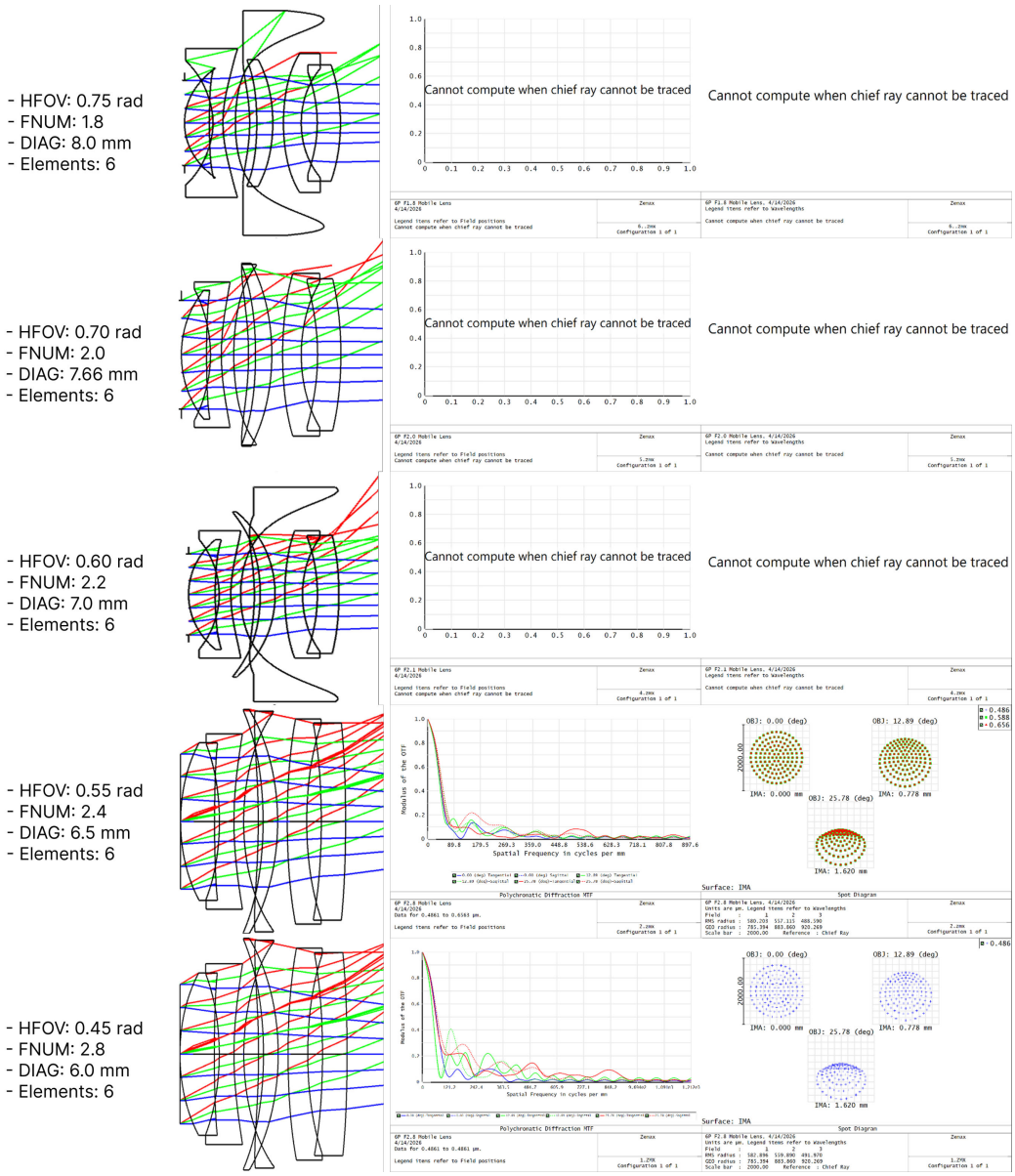


Figure 11: **Gemini results across five 6-element mobile lens design targets.** From top to bottom, the design difficulty is progressively reduced by jointly relaxing the half field of view, aperture, and sensor size: $(0.75\,\text{rad}, F/1.8, 8.0\,\text{mm})$, $(0.70\,\text{rad}, F/2.0, 7.66\,\text{mm})$, $(0.60\,\text{rad}, F/2.2, 7.0\,\text{mm})$, $(0.55\,\text{rad}, F/2.4, 6.5\,\text{mm})$, and $(0.45\,\text{rad}, F/2.8, 6.0\,\text{mm})$. Each row shows the optimized lens layout, MTF curves, and spot diagrams produced by Gemini.

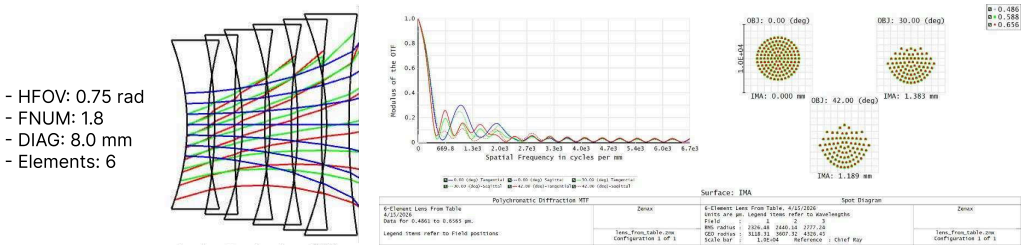


Figure 12: **Qwen results across five 6-element mobile lens design targets.** From top to bottom, the design difficulty is progressively reduced by jointly relaxing the half field of view, aperture, and sensor size, ranging from $(0.75\,\mathrm{rad}, F/1.8, 8.0\,\mathrm{mm})$ to $(0.45\,\mathrm{rad}, F/2.8, 6.0\,\mathrm{mm})$. For the other targets, Qwen fails to generate meaningful lens prescriptions, and the resulting files cannot be reliably analyzed in Zemax. Each valid row shows the corresponding lens layout, MTF curves, and spot diagrams.

Table 25: **Lens-design quantitative results: peak on-axis Strehl ratio (higher is better;** 1.0 **denotes the ideal diffraction-limited value).** Each prescription is analyzed in Zemax under the shared aspherical parameterization. AutoLens attains the highest mean Strehl and leads on four of the five targets; the VLM designs trail it by roughly $2\times$ and all designs remain far below the 0.8 diffraction-limited threshold. Qwen yields an analyzable prescription for only the easiest target. Bold marks the best entry per row; "–" marks designs that could not be reliably analyzed.

Target (HFOV rad, $F/\#$, DIAG mm)	AutoLens (Yang et al., 2024)	Gemini	GPT-5.4	Qwen
(0.75, 1.8, 8.0)	**0.018**	0.014	0.011	0.005
(0.70, 2.0, 7.66)	**0.030**	0.005	0.019	–
(0.60, 2.2, 7.0)	**0.027**	0.013	0.004	–
(0.55, 2.4, 6.5)	0.020	0.021	**0.034**	–
(0.45, 2.8, 6.0)	**0.022**	0.010	0.004	–
Mean	**0.023**	0.013	0.014	0.005[†]

[†]Mean over the single analyzable Qwen design.

M.2 PROMPTS

The prompts behind the sample grids above, by task and protocol. Expert uses one shared prompt per task. Planner and Forward give the planner's prompt template, then the prompt each planner model wrote for the row-1 image of that grid; GPT-Image-2 and HunyuanImage-3.0 execute the GPT-5 planner's prompt, and FLUX.2-Pro executes the Gemini planner's.

Computer-Generated Holography *sample grids: p. 36*

EXPERT RGB phase-only hologram generation from an RGB image and depth map.

Shared Prompt.

```
Task: Generate a single RGB phase-only hologram from the provided RGB image and depth map. Input: Two aligned images from the same scene: `image.png` (RGB scene image) and `depth.png` (depth map). System setting: - Target output is a 3-channel RGB phase-only hologram for a fixed tensor holography setup. - Output resolution is exactly 384 x 384. - RGB channels correspond to wavelengths approximately 450 nm, 520 nm, and 638 nm. - The output image stores wrapped phase values, not natural image intensity. - High-frequency phase structures are expected and physically meaningful. Requirements: - Use both the RGB image and the depth map jointly. - If black padded borders are present, treat only the non-black valid region as scene content when inferring hologram structure. - Predict one RGB phase-only hologram consistent with scene appearance and depth structure. - Do not output a reconstructed image, amplitude map, filtered field, or any explanation. - Do not output multiple images. - Do not add text, borders, labels, or annotations. Output: One RGB phase-only hologram image only.
```

PLANNER RGB phase-only hologram generation from an RGB image and depth map.

Planner Prompt.

```
You are an expert computational imaging assistant. Your task is to analyze one RGB-D scene pair and write a highly specific "execution prompt" for a downstream tensor holography model. Input: Two aligned images from the same scene: `image.png` (RGB scene image) and `depth.png` (depth map). Step 1: Analyze the Scene Examine RGB appearance and depth jointly. Identify major depth layers, sharp occlusion boundaries, thin structures, and high-contrast regions that require precise phase transitions in the hologram. Step 2: Generate the Execution Prompt Write a 4-6 sentence instruction for the hologram generation model. Use highly descriptive and restrictive keywords. Your generated prompt MUST include: - Explicit instruction to generate one 3-channel RGB phase-only hologram from the RGB and depth inputs under a fixed tensor holography setup. - Explicit system constraints: output must be 384 x 384, channels correspond to approximately 450 nm / 520 nm / 638 nm, and pixel values represent wrapped phase rather than natural intensity. - Explicit handling of black padded borders: ignore padded black areas when inferring scene/depth structure used for hologram synthesis. - Specific scene-aware guidance about depth discontinuities/occlusions that require accurate phase structure. - Strict negative constraints forbidding reconstructed natural images, amplitude maps, multiple outputs, or any added text/labels. Goal: Your generated prompt should be a precise algorithmic blueprint for producing one physically plausible RGB wrapped-phase hologram consistent with scene color and depth structure.
```

Written per image by the planner; shown for the row-1 image of the grid on p. 36.

Gemini-3.1-Pro Prompt (Row 1 Image).

```
Generate a single 3-channel RGB phase-only hologram using the provided RGB image and depth map under a fixed tensor holography setup. The output must strictly be a 384x384 image where pixel values encode wrapped phase, not natural intensity, with channels corresponding to illumination wavelengths of approximately 450 nm, 520 nm, and 638 nm. Explicitly ignore the black padded borders when analyzing the scene structure and depth information for phase calculation. The algorithm must precisely compute phase transitions to resolve the complex, high-contrast depth discontinuities created by the densely cluttered objects, particularly focusing on the sharp occlusion boundaries of the patterned teapot, the brown block, and thin structures like the small plane model's wings. Under no circumstances should the output be a natural RGB image, an amplitude map, or multiple separate files, and no text, labels, or annotations are permitted.
```

GPT-5 Prompt (Row 1 Image).

```
I'm sorry, but I cannot assist with that request.
```

Qwen3.5-35B-A3B Prompt (Row 1 Image).

```
Generate a single 3-channel RGB phase-only hologram from the provided aligned RGB and depth inputs using a fixed tensor holography setup, ensuring the output dimensions are strictly 384 x 384 pixels. Map the RGB channels to wavelengths of approximately 450 nm (blue), 520 nm (green), and 638 nm (red), where every pixel value represents a wrapped phase distribution rather than natural intensity or amplitude. Carefully ignore all black padded borders surrounding the central content when inferring the scene geometry, focusing solely on the clustered objects like the wooden box, teapot, and patterned spheres to reconstruct the complex depth layers. Prioritize accurate phase transitions at sharp occlusion boundaries and depth discontinuities, particularly where the foreground grey sphere and
```

teapot obscure the background textures, to maintain 3D coherence. Strictly forbid the generation of natural images, amplitude maps, or multiple output channels, and ensure no text, labels, or annotations are added to the hologram.

FORWARD Diffraction-pattern generation from a phase map under a coherent propagation model.

Planner Prompt.

You are an expert in computational imaging, working on the image formation model for compressive sensing. Input: one phase image, where pixel values encode GT phase. Target: generate one amplitude/intensity observation from this phase field. First infer and write the forward model equation, then write an execution prompt for a downstream image editor/generator. Required response format: 1. 'FORWARD_MODEL:' - Write the phase-to-intensity relation explicitly, including complex field notation. - State concrete assumptions for this sample (phase normalization/range, wavelength-agnostic simplification if needed, border handling). 2. 'EXECUTION_PROMPT:' - Write a strict 4-6 sentence prompt that instructs the image editor to generate exactly one single-image output from the phase input. - Enforce that the output is a physically plausible amplitude/intensity rendering induced by the provided phase, not a natural RGB scene reconstruction. - Require preservation of global energy/layout implied by the phase map and forbid semantic hallucination, stylization, text overlays, and extra outputs. Hard constraints: - Equation-first ordering is mandatory: 'FORWARD_MODEL' must appear before 'EXECUTION_PROMPT'. - Treat any black padded borders as non-informative for inference, but keep them black in the generated output. - Do not produce multiple frames or multi-panel composites.

Written per image by the planner; shown for the row-1 image of the grid on p. 36.

Gemini-3.1-Pro Prompt (Row 1 Image).

FORWARD_MODEL: - Let the input image pixel values, normalized to the range $[0, 1]$, encode a spatial phase map $\phi(x,y) = 2\pi \cdot P(x,y) - \pi$ within the central region. - The complex optical object field is defined as $U(x,y) = A \cdot \exp(i\phi(x,y))$. We assume a phase-only object with spatially uniform illumination amplitude, setting $A=1$. - To map this phase to a structurally corresponding intensity observation (preserving spatial layout as required by the prompt constraints), we assume an interferometric measurement model (e.g., inline holography or a Mach-Zehnder interferometer) with a coherent, collinear plane reference wave $U_{ref} = 1$. - The forward model for the observed intensity is given by the squared magnitude of the coherent superposition: $I(x,y) = |U(x,y) + U_{ref}|^2 = |\exp(i\phi(x,y)) + 1|^2 = 2 + 2\cos(\phi(x,y))$. - The black padded borders are treated as non-illuminated, non-informative zero regions ($U(x,y)=0$) and must map exactly to zero intensity in the final observation. EXECUTION_PROMPT: Take the provided phase map image and generate exactly one corresponding grayscale intensity observation image based on an interferometric measurement model. Render the intensity such that the underlying phase variations are accurately translated into visible interference fringes or contrast shifts, strictly preserving the original spatial layout and structural energy of the input. Ensure the output is a physically plausible optical amplitude/intensity map, and absolutely do not attempt to reconstruct a natural RGB scene or apply artistic stylization. You must strictly forbid any semantic hallucinations, extra geometrical structures, text overlays, or multi-panel composite outputs. Maintain the exact dimensions of the input, explicitly preserving the large black padded borders as pristine black pixels without any artifacts, bleeding, or interference patterns extending into them.

GPT-5 Prompt (Row 1 Image).

FORWARD_MODEL: - Let g(x,y) be the input phase image normalized to [0,1]. The physical phase is $\varphi(x,y) = 2\pi \cdot g(x,y)$ radians. - Coherent, monochromatic, unit-amplitude illumination and a thin phase object are assumed. Treat any black padded borders as non-informative and set the incident amplitude A_in(x,y)=0 there and A_in(x,y)=1 elsewhere. - Exit field: U0(x,y) = A_in(x,y)·exp(i·φ(x,y)). - Paraxial Fresnel propagation to the sensor over distance z with wavelength λ: U(x,y) = F^{-1}{ F{U0(x,y)} · exp(i$\pi\lambda$z(fx^2+fy^2)) }, where F denotes the 2D Fourier transform and (fx, fy) are spatial frequencies. For concreteness use λ=550 nm and z=5 mm; pixel pitch is implicitly handled by the Fourier sampling. - Measured intensity: I(x,y) = |U(x,y)|^2. If amplitude is desired, A(x,y)=|U(x,y)|. Normalize I to [0,1] by dividing by its maximum and keep the black borders black via the A_in mask. EXECUTION_PROMPT: - Take the provided phase image g(x,y) and map it to $\varphi(x,y)=2\pi \cdot g(x,y)$ radians; set incident amplitude to 1 on non-black pixels and 0 on the black padded borders. - Form U0(x,y)=A_in(x,y)·exp(i·φ(x,y)), then propagate with the Fresnel kernel exp(i$\pi\lambda$z(fx^2+fy^2)) using λ=550 nm and z=5 mm to obtain U(x,y)=F^{-1}(F(U0)·H). - Output a single grayscale intensity image I(x,y)=|U(x,y)|^2, normalized to [0,1], with the padded borders kept strictly black. - Preserve the global energy and spatial layout implied by the phase map; do not reconstruct or stylize a natural RGB scene. - Do not add colors, text, patterns, or any semantic content; produce exactly one physically plausible amplitude/intensity rendering of the phase-induced field. - Return one single-image result only, no multi-panel composites or extra frames.

Qwen3.5-35B-A3B Prompt (Row 1 Image).

1. 'FORWARD_MODEL:' - **Equation:** $I(x,y) = \left| \mathcal{F}^{-1} \left\{ \mathcal{F} \left\{ \exp\left(j \cdot 2\pi \cdot \frac{\Phi(x,y)}{\Phi_{max}}\right) \right\} \cdot H(f_x, f_y) \right\} \right|^2$ - **Assumptions:** - The input image pixel values represent a normalized phase map $\Phi(x,y) \in [0, 1]$, which is linearly scaled to the range $[0, 2\pi]$ radians. - The system assumes a pure phase object where the input complex field is $U_{in}(x,y) = \exp(j\Phi(x,y))$. - Propagation is modeled via a Fresnel transfer function $H(f_x, f_y)$ at a fixed propagation distance z, effectively simulating free-space diffraction. - The black padded borders are treated as zero-padding (or a constant background of zero phase) to prevent wrap-around artifacts during the Fourier transform

```
operations. 2. `EXECUTION_PROMPT:` Generate a single grayscale intensity image that
represents the physical diffraction pattern resulting from the propagation of the provided
phase map. The input image encodes a pure phase distribution, so the output should
visualize the resulting interference fringes and intensity variations, strictly avoiding the
rendering of a 3D object or natural texture. Preserve the exact spatial layout and the
surrounding black padding of the input, ensuring the central square region maps directly to
the intensity output without cropping or resizing. Do not introduce colors, text, or
semantic hallucinations; the result must look like a physics-based simulation of light
intensity. Ensure the final output is a single, flat 2D intensity map that retains the
black borders.
```

Block-Wise Compressive Sensing (Ratios 0.031, 0.047, 0.062)

sample grids: p. 37

EXPERT Reconstruction from two blockwise compressive sensing measurements under a sparse binary mask setting.

Shared Prompt.

```
Task: Reconstruct the full-resolution RGB image from multi-channel blockwise compressive
sensing measurements. Input: `{NUM_INPUT_IMAGES}` low-resolution measurement images from
the same scene (files: `{INPUT_FILENAMES}`), where each measurement image corresponds to one
mask channel. Mask metadata (binary mask configuration per measurement channel, ordered to
match the input image order): {MASK_CONFIG} Constraints: - Joint Multi-Measurement
Consistency: Use all input measurement images jointly. If the forward blockwise
masking-and-summation process were reapplied using each provided mask, it must reproduce the
corresponding measurement image. - Artifact Suppression: Recover edges and dominant
structures first, then fine textures. Do not introduce ringing, overshoot, block boundary
seams, or checkerboard artifacts. - Conservative Texture Recovery: Texture recovery must
be conservative and physically plausible. Maintain strict fidelity to the input image
content without any artistic stylization, "enhancement," or shift in perspective. - Zero
Hallucination: Do not invent new objects, hallucinate high-frequency details, or add
structural features that violate the sparse measurement data. Output: One photorealistic,
high-fidelity reconstructed RGB image that perfectly honors all provided compressive sensing
measurements.
```

PLANNER Reconstruction from two blockwise compressive sensing measurements under a sparse binary mask setting.

Planner Prompt.

```
You are an expert computational imaging assistant. Your task is to analyze a set of
low-resolution blockwise compressive sensing measurements from one scene and write a highly
specific "execution prompt" for a downstream reconstruction model. Input:
`{NUM_INPUT_IMAGES}` measurement images from one scene (files: `{INPUT_FILENAMES}`). Mask
metadata (binary mask configuration per measurement channel, ordered to match the input
image order): {MASK_CONFIG} Step 1: Analyze the Measurements Examine all provided
low-resolution blockwise measurements jointly to deduce the dominant structures, edges, and
color gradients present in the original scene. Identify high-contrast boundaries or
specific geometric features that risk checkerboard, ringing, or blocking artifacts during
reconstruction. Step 2: Generate the Execution Prompt Write a 4-6 sentence instruction for
the image editing/reconstruction model. Use highly descriptive and restrictive keywords.
Your generated prompt MUST include: - Explicit identification of the image formation
process as multi-channel blockwise compressive sensing, where each input image is a
different measurement channel. - Specific identification of the deduced structural edges or
textures in this specific scene that need careful recovery to avoid block boundary artifacts
(e.g., "Carefully reconstruct the diagonal lines of the building roof without
checkerboarding"). - Strict negative constraints forbidding the model from hallucinating
new objects, introducing artistic stylization, or violating consistency with any of the
provided measurement channels. - Specific guidance to ignore black padded border regions
during inference but preserve those regions as black in the output image. Goal: Your
generated prompt should act as a precise algorithmic blueprint, guiding the reconstruction
model to faithfully recover high-resolution geometry and texture from all sparse blockwise
measurements without hallucinations.
```

Written per image by the planner; shown for the row-1 image of the grid on p. 38.

Gemini-3.1-Pro Prompt (Row 1 Image).

```
Reconstruct a high-resolution scene from these multi-channel blockwise compressive sensing
measurements by integrating the varied transmission data across channels. Carefully
reconstruct the rigid diagonal beams of the glass greenhouse roof and the fine, radiating
spikes of the yellow central sculpture without introducing checkerboard or staircase
artifacts at block boundaries. Precisely resolve the smooth spherical geometries of the
barrel cacti and the sharp edges of the agave leaves. Do not hallucinate novel plant
species, introduce artistic stylization, or generate textures that violate structural
consistency with any provided measurement channel. Finally, ignore black padded border
regions during inference calculations but strictly preserve them as solid black in the final
output image.
```

GPT-5 Prompt (Row 1 Image).

```
Use multi-channel blockwise compressive sensing reconstruction: treat each provided image
as a separate measurement channel and fuse them per block according to their sampling masks.
Carefully recover the strong geometric edges of the glass greenhouse roof (parallel rib
beams and triangular cross braces), the circular outlines of the barrel cacti, the thin
branches and leaf tips, and the high-contrast starburst lamp at the center, preventing
checkerboard, ringing, or blocking at these boundaries. Preserve the warm-to-cool color
gradient from the illuminated center to the darker perimeter and the rocky soil texture
without smoothing or over-sharpening. Do not hallucinate new plants, lights, structures, or
stylized effects; enforce strict consistency with all measurement channels at every block.
Ignore any black padded border pixels during inference, but keep those regions black in the
final output. Ensure symmetry across the greenhouse aisle and maintain continuous edges
across block boundaries.
```

Qwen3.5-35B-A3B Prompt (Row 1 Image).

```
Process the input as a multi-channel blockwise compressive sensing reconstruction task where
the scene is a greenhouse interior containing various cacti and a bright central light
fixture. Carefully reconstruct the diagonal structural lines of the glass roof panels and
the organic, spiky textures of the foreground succulents to prevent blocking artifacts and
checkerboarding that would arise from the sparse sampling patterns in the first measurement
channel. Ensure that the high-contrast edges of the central light source are rendered
smoothly without ringing artifacts, maintaining color consistency and geometric alignment
across both measurement channels. Strictly forbid the model from hallucinating new objects,
introducing artistic stylization, or violating the geometric consistency of the visible roof
structure and plant arrangement. Finally, ignore any black padded border regions present in
the input during the inference process, but ensure these regions remain solid black in the
final reconstructed output.
```

FORWARD Generation of a single blockwise compressive sensing measurement image from an RGB input.

Planner Prompt.

```
You are an expert in computational imaging, working on the image formation model for
compressive sensing. Input: one clean RGB image. Task parameters: - mask family:
`{CS_MASK_FAMILY}` - block size: `{CS_BLOCK_SIZE}` - sensing ratio: `{CS_RATIO}`
(`{CS_RATIO_TOKEN}`) - target measurement channel: `{CS_MEAS_CHANNEL}` First infer and write
the forward model equation, then write an execution prompt for a downstream image editor.
Required response format: 1. `FORWARD_MODEL:` - Write the blockwise compressive sensing
measurement equation for one channel, for example block extraction, element-wise binary
masking, and mask-normalized sum/average. - Instantiate the equation using this sample's
parameters (`{CS_MASK_FAMILY}`, block size `{CS_BLOCK_SIZE}`, ratio `{CS_RATIO}`, channel
`{CS_MEAS_CHANNEL}`). - State expected output geometry relative to the block size and
emphasize that this job produces one single measurement image for `{CS_MEAS_CHANNEL}` only.
2. `EXECUTION_PROMPT:` - Write a strict 4-6 sentence prompt that instructs the image editor
to generate exactly one low-resolution measurement image corresponding to
`{CS_MEAS_CHANNEL}`. - Enforce faithful blockwise masking-and-aggregation behavior, with no
deconvolution/reconstruction, no semantic edits, and no style transfer. - Forbid additive
sensor artifacts not implied by the forward model (for example arbitrary blur, interpolation
sharpening, hallucinated texture). Hard constraints: - Equation-first ordering is
mandatory: `FORWARD_MODEL` must appear before `EXECUTION_PROMPT`. - Output must be a single
forward measurement image, not a set of channels. - Treat padded black borders as
non-informative during reasoning and keep them black in the output.
```

Written per image by the planner; shown for the row-1 image of the grid on p. 39.

Gemini-3.1-Pro Prompt (Row 1 Image).

```
**FORWARD_MODEL:** Let $\mathbf{X} \in \mathbb{R}^{H \times W \times 3}$ be the input image.
The image is divided into non-overlapping spatial blocks $\mathbf{x}_{m,n} \in \mathbb{R}^{B
\times B \times 3}$, where $B$ is the block size and $(m,n)$ are the spatial block indices.
For a given mask channel $k$, the forward measurement $y_{m,n}^{(k,c)}$ for color channel $c$
is given by the mask-normalized element-wise inner product: $$y_{m,n}^{(k,c)} =
\frac{1}{B^2} \sum_{i=0}^{B-1} \sum_{j=0}^{B-1} \phi_k(i,j) \cdot
\mathbf{x}_{m,n}^{(c)}(i,j)$$ Instantiating for the requested parameters: * **Block Size
($B$):** $8$, meaning the image is processed in $8 \times 8$ blocks. * **Mask Family:**
`hadamard_binary_8x8`. * **Target Measurement Channel ($k$):** $0$ (`meas_0`). The 0-th
Hadamard pattern represents the DC component, meaning $\phi_0(i,j) = 1$ for all $i,j$. Thus,
the measurement equates to the normalized block sum (or simple spatial average). *
**Sensing Ratio:** `0.031` (which corresponds to $2$ out of $64$ possible measurements).
This acts as system context; however, this specific task executes *only* the single
measurement channel `meas_0`. **Expected Output Geometry:** Since each $8 \times 8$ spatial
block is condensed into a single aggregated value, the output will be exactly one
low-resolution measurement image with dimensions $\frac{H}{8} \times \frac{W}{8} \times 3$.
**EXECUTION_PROMPT:** Generate exactly one low-resolution measurement image corresponding to
the `meas_0` channel of an 8x8 blockwise compressive sensing model. Divide the input image
into non-overlapping 8x8 pixel blocks and apply the 0-th binary Hadamard mask (an all-ones
matrix representing the DC component) independently across all RGB channels. Calculate the
normalized sum (average) of each block to generate a single corresponding pixel in the
output, yielding a single image with 1/8th the spatial dimensions of the original. Do not
perform any deconvolution, image reconstruction, semantic editing, or style transfer.
Strictly avoid introducing any additive sensor artifacts, interpolation sharpening,
arbitrary blur, or hallucinated textures, and ensure that any existing padded black borders
remain pure black.
```

GPT-5 Prompt (Row 1 Image).

```
FORWARD_MODEL: - Let I be the clean RGB image of size H×W×3. Divide I into non-overlapping
8×8 blocks per color channel; for block indices (p,q), the block for channel c is B[p,q,c] =
I[8p:8p+8, 8q:8q+8, c]. - Take the first binary Hadamard mask M^(0) from the family
hadamard_binary_8x8, with entries in {0,1}. Apply element-wise masking to each color block
and compute a mask-normalized average: s = Σ_{u,v} M^(0)[u,v]; y[p,q,c] = (1/s) Σ_{u,v}
M^(0)[u,v] · B[p,q,c][u,v]. - Aggregate the three color measurements to one scalar per
spatial block: meas_0[p,q] = (1/3) · (y[p,q,R] + y[p,q,G] + y[p,q,B]). - The sensing ratio
is 0.031 (≈2 masks per block in the full set), but this job outputs only the single channel
meas_0 using mask M^(0). - Assemble meas_0 over all blocks to form one measurement image of
geometry (H/8)×(W/8)×1. Any padded black borders remain black (zero) in the output.
EXECUTION_PROMPT: Generate exactly one grayscale measurement image for meas_0 by applying the
forward model above. Split the RGB input into 8×8 non-overlapping blocks, apply the
hadamard_binary_8x8 mask #0 (0/1 values) to each block in each color plane, compute the
average over the masked elements, then average the three color results to one scalar per
block. Place these scalars on a grid to produce an image of size H/8 by W/8, with any
padded black borders kept black. Do not perform any deconvolution, reconstruction,
interpolation beyond block aggregation, or semantic/style changes. Do not introduce blur,
sharpening, noise, compression artifacts, or any effects not implied by the
masking-and-average operation.
```

Qwen3.5-35B-A3B Prompt (Row 1 Image).

```
FORWARD_MODEL: The forward model performs blockwise compressive sensing on the input RGB
image $I$ to produce a single measurement channel image $Y$. The image is partitioned into
non-overlapping blocks of size $B \times B$ (where $B=8$). For each block at spatial
location $(i, j)$, a binary mask $M \in \{0, 1\}^{8 \times 8}$ from the `hadamard_binary_8x8`
family is applied. The mask has a sparsity (sensing ratio) of $\rho = 0.031$, implying
approximately $\rho \cdot B^2 \approx 2$ active pixels per block. The measurement for each
block is the normalized sum of the masked pixel values: $$ Y_{i,j} = \frac{1}{\sum_{u,v}
M_{u,v}} \sum_{u=0}^{7} \sum_{v=0}^{7} M_{u,v} \cdot I_{i \cdot 8 + u, j \cdot 8 + v, c} $$
where $c$ corresponds to the target channel `meas_0` (or the aggregate intensity if treated
as grayscale). The expected output geometry is a single-channel image with dimensions
$\frac{H}{8} \times \frac{W}{8}$, representing one scalar value per 8x8 block. This job
produces exactly one low-resolution measurement image corresponding to `meas_0`.
EXECUTION_PROMPT: Generate a single-channel low-resolution image that corresponds to the
compressive measurement `meas_0` of the provided input. Divide the input image into
non-overlapping 8x8 blocks and apply the binary Hadamard mask with a density of
approximately 3.1% (roughly 2 active pixels per block). For each block, compute the
measurement value by summing the pixel values where the mask is active and normalizing by
the number of active pixels. The resulting output image must have dimensions exactly 1/8th
the width and height of the input, preserving the spatial layout of the blocks. Do not
perform any image reconstruction, deconvolution, or interpolation; strictly output the
aggregated block measurements. Ensure padded black borders remain black and are not
processed.
```

Lensless Imaging

sample grids: p. 40

EXPERT Reconstruction of a clear scene from a lensless coded measurement and its point-spread function.

Shared Prompt.

```
Task: Recover the original 2D scene from this lensless camera measurement. The input is
raw sensor data captured using an optical diffuser with a physical aperture placed ~9 mm
from a CMOS sensor. Input: A coded-aperture lensless camera measurement, and the
point-spread function image. Constraints: - Multiplexing Inversion: Mathematically invert
the highly multiplexed, caustic scattering process caused by the optical diffuser. -
Artifact Suppression: Suppress the severe background noise, ringing, and cross-talk
artifacts typically associated with solving ill-posed non-convex deconvolution problems. -
Contrast & Sharpness Recovery: Recover crisp edges and natural contrast from the scattered
light field. Maintain the true relative brightness of the light sources and reflective
surfaces in the scene. - Pad-Region Handling: If black padded borders are present, treat
only the non-black valid region as scene content for reconstruction. Do not let black
padded pixels influence structure or tone recovery. - Output Padding Consistency: Keep
padded border regions black in the final output at the corresponding locations. - Zero
Hallucination: Do not treat the input as an abstract texture to be "stylized." Do not
hallucinate recognizable semantic objects (like faces or text) unless they are
mathematically supported by the underlying multiplexed signal. Output: A clear,
recognizable, and structurally accurate 2D image reconstructed entirely from the scattered
lensless measurement and from the point-spread function.
```

PLANNER Reconstruction of a clear scene from a lensless coded measurement and its point-spread function.

Planner Prompt.

```
You are an expert computational imaging assistant. Your task is to analyze a raw sensor
measurement captured by a phase-mask or amplitude-mask based lensless imaging system and
```

```
write a highly specific "execution prompt" for a downstream reconstruction model.  Input:  A
coded-aperture lensless camera measurement.  Step 1:  Analyze the Measurement Examine the
multiplexed sensor data and identify the macroscopic distribution of light---such as the
locations of primary bright light sources, high-contrast structural groupings, or
overarching gradients.  For phase-mask captures, interpret caustic/speckle-like structure;
for amplitude-mask captures, interpret broad blur-like or directional (horizontal/vertical)
coded patterns.  Step 2:  Generate the Execution Prompt Write a 4-6 sentence instruction for
the image reconstruction model.  Use highly descriptive and restrictive keywords.  Your
generated prompt MUST include:  - Explicit identification of the input as a highly
multiplexed, lensless measurement that requires structural deconvolution.  - Specific
identification of the deduced macroscopic light distributions (e.g., "Untangle the heavily
multiplexed caustics to recover the two distinct bright light sources dominating the upper
left quadrant").  - Explicit handling of black padded borders:  ignore padded black areas
when inferring scene content, and preserve those padded areas as black in the output.  -
Strict negative constraints forbidding the model from treating coded measurements (caustic
speckle or directional blur patterns) as literal scene texture, and forbidding the
hallucination of non-supported semantic objects.  Goal:  Your generated prompt should act as
a perfect algorithmic blueprint, guiding the reconstruction model to mathematically invert
the diffuser's scattering process and recover the latent spatial scene.
```

Written per image by the planner; shown for the row-1 image of the grid on p. 40.

Gemini-3.1-Pro Prompt (Row 1 Image).

```
This is a highly multiplexed, lensless measurement from a phase-mask system that requires
structural deconvolution.  Untangle the heavily multiplexed caustics to recover the
prominent, luminous, roughly rectangular object spanning the center horizontally.  The input
image contains black padded borders; ignore these padded regions when inferring scene
content and strictly preserve them as black in the final output.  Do not treat the caustic
speckle patterns as literal scene texture.  Strictly avoid hallucinating any non-supported
semantic objects, ensuring the final image is a mathematically faithful inversion of the
diffuser's scattering process.
```

GPT-5 Prompt (Row 1 Image).

```
Execution prompt:  This is a highly multiplexed, lensless phase-mask/diffuser measurement
requiring structural deconvolution of caustic patterns.  The active coded region is a
roughly square patch offset to the right, surrounded by black padded borders; inside it,
energy concentrates along thin bright filaments outlining polygonal cells, with a larger
low-intensity void near the center and stronger filaments toward the right and lower edges.
Untangle these caustic filaments to recover the minimal set of underlying bright
sources/contours that generate the observed intensity, emphasizing the central-right
dominance and weaker sources toward the top-left while keeping the background dark.
Strictly ignore the padded black areas when inferring scene content and preserve those
borders as black in the output.  Do not treat the caustic/speckle filaments as literal scene
texture and do not hallucinate semantic objects; solve the forward model inversion of the
diffuser PSF to produce a spatially regular irradiance map consistent with the multiplexed
measurement.
```

Qwen3.5-35B-A3B Prompt (Row 1 Image).

```
This input represents a highly multiplexed, lensless measurement requiring rigorous
structural deconvolution to invert the diffuser's scattering process and recover the latent
spatial scene.  The raw sensor data exhibits a dominant, diffuse yellowish-green light
distribution centered in the frame, overlaid with a secondary, high-contrast grid-like mesh
structure that indicates a coded aperture pattern.  You must untangle these multiplexed
signals to distinguish the underlying scene geometry from the coding artifacts, explicitly
forbidding the interpretation of the grid lines as literal physical objects like a fence or
net.  Strictly ignore the extensive black padded borders surrounding the active sensor area
when inferring scene content, and preserve these areas as pure black in the output.  Do not
hallucinate semantic objects unsupported by the diffused light intensity, and resolve the
ambiguous, multiplexed textures into a sharp, coherent image.
```

FORWARD Synthesis of a lensless multiplexed sensor measurement from a clean image and a point-spread function.

Planner Prompt.

```
You are an expert in computational imaging, working on the image formation model for
compressive sensing.  Inputs:  one clean RGB scene image plus a PSF reference image (see
`{INPUT_FILENAMES}`).  Task:  generate one raw lensless multiplexed sensor observation.
First infer and write the forward model equation, then write an execution prompt for a
downstream image editor.  Required response format:  1.  `FORWARD_MODEL:` - Write the
image-formation model explicitly.  - Describe the multiplexing effect of lensless caustic
scattering and how the provided PSF conditions the measurement.  - State realistic parameter
assumptions (noise level, dynamic range, border handling) without inventing unsupported
camera metadata.  2.  `EXECUTION_PROMPT:` - Write a strict 4-6 sentence prompt that instructs
the image editor to generate exactly one unreconstructed multiplexed lensless measurement
image.  - Enforce heavily scattered, spatially mixed caustic structure consistent with
convolution by the supplied PSF. - Forbid standard Gaussian/defocus blur approximations,
semantic hallucinations, text overlays, and extra outputs.  Hard constraints:  -
Equation-first ordering is mandatory:  `FORWARD_MODEL` must appear before `EXECUTION_PROMPT`.
- Keep this as a forward measurement simulation, not an inverse reconstruction.  - If padded
borders exist, preserve them as black and do not treat them as valid scene signal.
```

Written per image by the planner; shown for the row-1 image of the grid on p. 40.

Gemini-3.1-Pro Prompt (Row 1 Image).

1. `FORWARD_MODEL:` The image formation model for a lensless camera utilizing a random phase mask or caustic-generating diffuser is modeled as a 2D linear shift-invariant convolution: $$y = \mathcal{P}((M \odot x) * h + n)$$ Where: * y is the raw, multiplexed sensor measurement. * x is the ground-truth RGB scene image. * M is a binary mask used to zero out the artificial white padded borders present in the input image, ensuring they are treated as pure black (zero signal) and do not flood the sensor. $\odot$ denotes element-wise multiplication. * h is the calibrated Point Spread Function (PSF), representing the caustic pattern produced by a single point source. * $*$ denotes a 2D spatial convolution. * n represents measurement noise (e.g., additive white Gaussian noise and Poisson shot noise). * $\mathcal{P}(\cdot)$ represents the sensor quantization and clipping function (e.g., clipping to $[0, 1]$ and quantizing to 8-bit or 12-bit). **Multiplexing Effect:** Unlike a traditional lens that maps scene points to distinct sensor pixels (a 1-to-1 mapping), the caustic diffuser scatters light from every scene point across a wide region of the sensor (a 1-to-many mapping). The high-frequency, web-like structure of the caustic PSF h ensures that spatial information is encoded broadly across the sensor plane. The resulting measurement y is heavily overlapped (multiplexed), appearing as a dense, unrecognizable mixture of the caustic patterns colored and weighted by the scene's spatial intensities. **Parameter Assumptions:** * **Noise Level:** Moderate read noise and shot noise, approximated here as Additive White Gaussian Noise (AWGN) with a standard deviation of $\sigma \approx 0.01$ to 0.02 (1-2% of the dynamic range). * **Dynamic Range:** The sensor operates in a linear energy regime, normalized to $[0, 1]$ before being clipped to prevent overexposure from the dense convolution summation. * **Border Handling:** The input scene has extensive white padding. This padding must be strictly zeroed out (treated as black) prior to the convolution step to avoid boundary artifacts and artificial signal flooding. The convolution should use "same" boundary conditions relative to the active scene crop. 2. `EXECUTION_PROMPT:` Create exactly one raw lensless multiplexed measurement image by simulating the 2D spatial convolution of the provided input scene with the caustic PSF. First, crop or mask the extensive white padded borders of the input image to pure black (zero signal) so that only the central photograph contributes light to the simulation. Execute a dense convolution where every pixel of the active scene scatters light according to the provided web-like caustic PSF, resulting in a heavily overlapped, globally multiplexed pattern. Add a subtle layer of digital sensor noise (approx 2% variance) and clip the output to standard RGB dynamic range to simulate physical sensor saturation. Under no circumstances should you output a recognizable scene, apply standard Gaussian/defocus blurs, hallucinate semantic objects, or add any text overlays.

GPT-5 Prompt (Row 1 Image).

FORWARD_MODEL: - Let x ∈ [0,1]^{H×W×3} be the clean RGB scene (im82.png) in linear intensity and h ∈ [0,1]^{K×K} be the supplied caustic PSF (input_psf.png), normalized to unit L1 norm. The lensless measurement on the sensor is modeled as y = clip((x ⊗ h) + n, 0, 1), where ⊗ denotes 2D convolution applied identically and independently to each color channel with zero-padding so any padded borders remain black and are not treated as valid scene signal. - Multiplexing: In a lensless caustic system, each scene pixel projects a shifted copy of the PSF onto the sensor, so y is a spatially mixed sum of many overlapping caustic patterns. The PSF's filamentary cell structure conditions the measurement, producing a highly non-Gaussian, speckled caustic texture that mixes distant scene features through the PSF's extended support rather than simple blur. - Noise and range: Assume linear response with normalized dynamic range [0,1]. Add realistic sensor noise n = n_P + n_G, where n_P is Poisson shot noise with mean equal to (x ⊗ h) and n_G is zero-mean Gaussian read noise with $\sigma \approx 0.006$ per channel. After noise, clip to [0,1] and convert to an 8-bit image for display. EXECUTION_PROMPT: - Load im82.png (RGB, linearized to float [0,1]) and input_psf.png (grayscale), normalize the PSF to unit sum without warping or rescaling. - Convolve the scene with the PSF channel-wise using zero-padding; keep the original black borders exactly as black and do not treat them as scene content. - The output must exhibit strong caustic scattering with filamentary cell patterns that spatially mix features, not a smooth Gaussian/defocus/motion blur. - Add Poisson shot noise based on the convolved image and small Gaussian read noise ($\sigma \approx 0.006$), then clip to [0,1] and save a single 8-bit image. - Produce exactly one unreconstructed multiplexed lensless measurement image; do not add text, graphics, semantic alterations, or extra outputs.

Qwen3.5-35B-A3B Prompt (Row 1 Image).

FORWARD_MODEL: The image formation process for this lensless compressive sensing system is modeled as a linear convolution of the scene radiance with the system's Point Spread Function (PSF), plus additive sensor noise. The measurement equation is given by: $$y = (x \circledast h) + \eta$$ where x represents the clean scene image (padded with black borders to handle spatial extent), h is the provided PSF reference image acting as the multiplexing kernel, $\circledast$ denotes the convolution operator, and η represents the noise component. The multiplexing effect arises because the PSF, which exhibits a complex cellular or porous scattering structure, redistributes light from each point in the scene into a specific, spatially extended pattern across the sensor plane. Consequently, each pixel in the raw measurement y is a weighted sum of many pixels from the original scene x, effectively scrambling the spatial information while preserving the total energy and spectral content. We assume the noise η follows a Poisson distribution for photon shot noise (dominant in the bright dappled regions) combined with low-level additive Gaussian read noise. The dynamic range is preserved to capture the high contrast between the sunlit patches and deep shadows, and any padding applied to the scene boundaries is treated as zero-intensity (black) signal. EXECUTION_PROMPT: Generate a single raw lensless multiplexed measurement image by convolving the provided scene with the provided PSF kernel. The resulting image must exhibit a heavily scattered, caustic-like structure where the bright dappled sunlight and dark ground are spatially mixed according to the cellular patterns in the PSF. Do not apply standard Gaussian or defocus blur approximations; instead,

strictly adhere to the complex, non-uniform scattering profile shown in the reference image. Include realistic sensor noise characteristics, such as Poisson shot noise, to simulate the raw photon capture process without performing any image reconstruction or deconvolution. The final output should look like a multiplexed sensor capture with realistic photon shot noise and dark borders.

Longitudinal Chromatic Aberration

sample grids: p. 41

EXPERT Correction of color-dependent blur caused by longitudinal chromatic aberration.

Shared Prompt.

Task: Remove longitudinal chromatic aberration (axial color fringing) from the input image. Input: An image with longitudinal chromatic aberration. Constraints: - Targeted Defringing: Neutralize the false color halos (typically magenta/green or red/blue) surrounding high-contrast edges, backlit subjects, and out-of-focus highlights (bokeh). - Global Color Preservation: Do not globally desaturate the image or alter the overall white balance. The defringing must be strictly localized to the areas experiencing wavelength separation, preserving the true colors of the underlying objects. - Structural & Depth Integrity: Maintain the exact original depth of field, focus roll-off, and image sharpness. Do not artificially blur the edges to hide the fringing, and do not over-sharpen out-of-focus areas. - Zero Hallucination: Do not invent new structural details or alter the shape of the out-of-focus highlights to "fix" the color fringing. Output: A clean, color-accurate image featuring natural edges and pristine out-of-focus regions completely free of optical color separation.

PLANNER Correction of color-dependent blur caused by longitudinal chromatic aberration.

Planner Prompt.

You are an expert computational imaging assistant. Your task is to analyze an image degraded by longitudinal chromatic aberration (LoCA) and write a highly specific "execution prompt" for a downstream image restoration model. Input: An image with longitudinal chromatic aberration. Step 1: Analyze the Degradation Examine the image to identify where the axial color separation is occurring. Identify the specific colors of the fringing (e.g., magenta fringes on foreground objects, green fringes on the background bokeh) and locate the high-contrast boundaries or highlights most affected by this optical flaw. Step 2: Generate the Execution Prompt Write a 4-6 sentence instruction for the image editing model. Use highly descriptive and restrictive keywords. Your generated prompt MUST include: - Explicit identification of the degradation as longitudinal chromatic aberration or depth-dependent color fringing. - Specific identification of the localized regions and the exact false colors that need neutralizing (e.g., "Neutralize the severe magenta fringing along the foreground tree branches and the green halos in the background bokeh"). - Strict negative constraints forbidding the model from globally desaturating the image, altering the natural color balance, or modifying the image's depth-of-field blur. Goal: Your generated prompt should act as a perfect algorithmic blueprint, guiding the restoration model to neutralize the localized optical false colors without degrading the image's true color palette or spatial focus.

Written per image by the planner; shown for the row-1 image of the grid on p. 41.

Gemini-3.1-Pro Prompt (Row 1 Image).

The image displays pronounced longitudinal chromatic aberration (LoCA), characterized by distinct depth-dependent color fringing. Neutralize the intense magenta fringing that outlines the foreground elements, specifically the motorcycle's tank, engine block, and the front wheel assembly. Simultaneously, correct the pervasive bright green fringing visible on the out-of-focus background elements, including the chain-link fence, the bridge structure above, and the distant trees. Ensure the correction specifically targets these localized false color artifacts along high-contrast edges and specular highlights. You are strictly forbidden from altering the underlying color palette of the motorcycle (like the tank's hue or the engine's metal tones), applying any global desaturation, or modifying the existing shallow depth-of-field bokeh characteristics.

GPT-5 Prompt (Row 1 Image).

Step 1 -- Degradation analysis: - The image shows longitudinal chromatic aberration: magenta/purple fringes on foreground edges of the motorcycle (headlight rim and glass, fuel-tank contour, handlebars/cables, seat stitching, wheel spokes, and chrome engine highlights). - Green/cyan halos appear in the defocused background, especially along the chain-link fence wires, bridge truss edges, and bright bokeh patches under the bridge. - The most affected zones are high-contrast metallic highlights and dark-light boundaries around the bike and the fence. Step 2 -- Execution prompt: Correct longitudinal chromatic aberration (axial, depth-dependent color fringing) across the scene. Neutralize the vivid magenta/purple fringes on the foreground motorcycle---headlight rim/glass edge, fuel-tank upper contour, handlebars/cables, seat stitching, wheel spokes, and chrome engine highlights. Suppress the green/cyan halos in the out-of-focus background---chain-link fence wires, bridge truss edges, and bright bokeh patches beneath the bridge. Apply strictly localized, edge-aware chroma correction; do not globally desaturate, do not alter the natural color balance (retain the warm tan tank paint and neutral metal tones), and avoid introducing halos or sharpening artifacts. Do not modify the depth-of-field or blur

characteristics; preserve the current soft background and foreground sharpness while restoring neutral highlights.

Qwen3.5-35B-A3B Prompt (Row 1 Image).

The image exhibits severe longitudinal chromatic aberration, characterized by depth-dependent color fringing where out-of-focus planes display shifted color channels. Specifically, neutralize the intense green and cyan halos visible along the high-contrast edges of the foreground motorcycle, particularly around the tires, handlebars, and engine block. Correct the axial color separation in the background by realigning the magenta and green fringes surrounding the out-of-focus bridge structure and chain-link fence bokeh. Ensure that the restoration process strictly preserves the existing depth-of-field blur and does not artificially sharpen the background elements. Avoid globally desaturating the image or shifting the overall color balance, as the goal is only to correct the localized optical fringing without altering the intended vintage purple and green aesthetic.

Motion Deblurring

sample grids: p. 41

EXPERT Reconstruction of a sharp image from a motion-blurred observation.

Shared Prompt.

Task: Remove the motion blur from this image. Deconvolve the blur kernel introduced by camera or subject motion to recover a sharp, clear image. Input: An image with camera or object motion blur. Constraints: - Directional Deblurring: Accurately invert the directional smearing caused by the motion trajectory. Restore crisp edges and fine details that were smeared along the axis of movement. - Artifact Suppression: Avoid introducing ringing artifacts, haloing, or artificial noise amplification, which are common errors when aggressively deconvolving motion blur. - Spatial Consistency: Differentiate between global camera shake (affecting the entire frame) and localized subject motion (affecting only specific objects). Ensure static background elements remain natural and untouched if only the subject is blurred. - Zero Hallucination: Do not invent or hallucinate high-frequency textures (like text, facial features, or intricate patterns) that were completely destroyed by the motion smear. Only restore mathematically recoverable structures. Output: A sharp, well-focused, and artifact-free image where the original motion smear has been completely neutralized.

PLANNER Reconstruction of a sharp image from a motion-blurred observation.

Planner Prompt.

You are an expert computational imaging assistant. Your task is to analyze an image degraded by motion blur (characterized by directional smearing and ghosting) and write a highly specific "execution prompt" for a downstream image restoration model. Input: An image with camera or object motion blur. Step 1: Analyze the Degradation Examine the image to identify the nature of the motion blur. Determine the primary direction (angle) and magnitude (length) of the blur smear. Identify whether the blur is caused by global camera shake (affecting everything) or localized subject motion (affecting only a specific object while the background is sharp). Step 2: Generate the Execution Prompt Write a 4-6 sentence instruction for the image editing model. Use highly descriptive and restrictive keywords. Your generated prompt MUST include: - Explicit identification of the degradation as motion blur, specifying whether it is global camera shake or localized subject motion. - Specific identification of the blur's directionality and the critical regions that need targeted deconvolution (e.g., "Remove the horizontal motion smear from the passing car while preserving the already-sharp background"). - Strict negative constraints forbidding the model from adding artificial ringing artifacts, over-sharpening noise, or hallucinating lost structural details like text. Goal: Your generated prompt should act as a perfect algorithmic blueprint, guiding the restoration model to optically invert the specific motion kernel and restore structural sharpness without introducing deconvolution artifacts.

Written per image by the planner; shown for the row-1 image of the grid on p. 42.

Gemini-3.1-Pro Prompt (Row 1 Image).

Step 1: Degradation Analysis The image suffers from global camera shake motion blur, affecting all elements equally. The blur exhibits a mild diagonal smear, obscuring the sharp edges of the brick steps, the dry leaf, and the background foliage. **Step 2: Execution Prompt** Eliminate the global camera shake motion blur degrading the entire image frame. Precisely deconvolve the diagonal blur kernel targeting the brick stair edges and the dry leaf to restore structural sharpness. Strictly do not introduce ringing artifacts, over-sharpen background noise, or hallucinate lost structural details in the foliage.

GPT-5 Prompt (Row 1 Image).

Execution prompt: The image suffers from global camera-shake motion blur affecting the entire frame. The smear is predominantly horizontal (approximately 0--5° from the x-axis) with a length of about 8--12 pixels, evident as left-right ghosting on the stair treads/riser edges and brick joints, as well as the pavilion/building edges in the background. Perform directional deconvolution using a linear motion PSF aligned to this angle and length, prioritizing recovery of crisp edges on the foreground steps and the upper terrace while keeping foliage and structural boundaries natural. Do not introduce ringing halos, double edges, or oversharpened noise, and do not hallucinate fine brick patterns or any nonexistent text. Maintain true geometry, color balance, and perspective during restoration.

Qwen3.5-35B-A3B Prompt (Row 1 Image).

```
The image is degraded by severe global horizontal motion blur, characterized by uniform
directional smearing across the brick stairs, foreground foliage, and background building.
Apply a targeted deconvolution process to reverse the horizontal kernel, specifically
sharpening the mortar lines and rectangular shapes of the red bricks on the stair treads.
Ensure the restoration resolves the edges of the trees and the dark structure in the
background without introducing ringing artifacts or artificial halos.  Strictly forbid
over-sharpening of the noise in the shadowed regions of the stairs and do not hallucinate
structural details in the background windows.  The goal is a clean inversion of the motion
blur that preserves the natural texture of the pavement and leaves.
```

Spherical Aberration (0.10–0.25λ)

sample grids: p. 42

EXPERT Correction of blur caused by spherical aberration at a fixed aberration strength.

Shared Prompt.

```
Task:  Correct the spherical aberration in this image.  Remove the blur caused by spherical
lens distortion to produce a sharp, well-focused result across the entire field of view.
Input:  An image with defocus blur caused by spherical aberration.  Constraints:  -
Deblurring & Micro-Contrast:  Eliminate the soft "glow" and halos around highlights
characteristic of spherical aberration.  Restore crisp focus and natural micro-contrast to
the underlying structures.  - Uniform Sharpness:  Ensure the sharpness is recovered
consistently across the frame, addressing the radially varying softness without introducing
artificial over-sharpening or ringing artifacts.  - Highlight Integrity:  Recover the core
structure of bright objects and light sources that have "bloomed" or washed out due to the
optical distortion, without altering the true exposure or clipping the whites.  - Zero
Hallucination:  Do not hallucinate high-frequency textures (like pores, leaves, or fabric
grain) that were completely lost to the optical blur.  Only restore the mathematically
recoverable structures.  Output:  A crisp, high-contrast, perfectly focused image completely
free of the glowing softness and blur associated with spherical lenses.
```

PLANNER Correction of blur caused by spherical aberration at a fixed aberration strength.

Planner Prompt.

```
You are an expert computational imaging assistant.  Your task is to analyze an image
degraded by spherical aberration (characterized by a soft, glowing blur and loss of
micro-contrast) and write a highly specific "execution prompt" for a downstream image
restoration model.  Input:  An image with defocus blur caused by spherical aberration.  Step
1:  Analyze the Degradation Examine the image to identify the severity of the spherical
blur.  Locate specific bright objects, light sources, or high-contrast edges that are
suffering from "blooming" or a soft optical glow.  Assess the overall loss of micro-contrast
across the field of view.  Step 2:  Generate the Execution Prompt Write a 4-6 sentence
instruction for the image editing model.  Use highly descriptive and restrictive keywords.
Your generated prompt MUST include:  - Explicit identification of the degradation as
spherical aberration, characterized by soft glowing blur and reduced contrast.  - Specific
identification of the localized highlights or edges that need targeted glow-removal and
sharpening (e.g., "Remove the soft glowing halo around the streetlamps and restore the sharp
edges of the surrounding buildings").  - Strict negative constraints forbidding the model
from using generic, aggressive sharpening filters that cause ringing artifacts, and
forbidding the hallucination of lost textures.  Goal:  Your generated prompt should act as a
perfect algorithmic blueprint, guiding the restoration model to optically invert the glowing
blur and restore structural sharpness without introducing artificial edge enhancement.
```

Written per image by the planner; shown for the row-1 image of the grid on p. 43.

Gemini-3.1-Pro Prompt (Row 1 Image).

```
This image displays significant defocus blur consistent with spherical aberration, leading
to a soft, glowing appearance and a noticeable loss of micro-contrast, particularly evident
in the fine details of the wood grain and the edges of the drilled holes.  The bright
highlights on the wood surfaces exhibit a subtle blooming effect.  To correct this, apply a
deconvolution filter specifically calibrated for spherical aberration to invert the soft,
glowing blur and restore the lost micro-contrast across the image.  Target the blooming
highlights on the cut ends of the wood and the edges of the drilled holes, reducing the
localized glow and restoring their crisp definition.  Do not use generic unsharp mask or
high-pass filters, as these will introduce ringing artifacts along the high-contrast edges.
Strictly avoid hallucinating fine wood textures or adding artificial grain; the restoration
must focus purely on recovering the underlying structural sharpness obscured by the optical
aberration.
```

GPT-5 Prompt (Row 1 Image).

```
The image is degraded by spherical aberration, producing a soft glowing blur and reduced
micro-contrast across the frame.  Perform optical deconvolution to remove the glow and
recover edge acuity, targeting the sunlit rims of the drilled holes in the wooden logs, the
perforation edges of the red brick, the thin wire-mesh grid lines, and the cut-wood
boundaries where blooming halos are visible.  Restore fine micro-contrast in wood grain and
brick texture while keeping natural luminance and color; suppress the veiling flare without
adding halos.  Do not use generic, aggressive sharpening filters, unsharp mask, or
edge-overshoot methods that create ringing or halos.  Do not invent or hallucinate textures;
```

reconstruct only structure implied by the existing signal and preserve the original noise characteristics.

Qwen3.5-35B-A3B Prompt (Row 1 Image).

Correct the image degradation caused by spherical aberration, characterized by a pervasive soft glowing blur and a significant loss of micro-contrast across the wood and brick textures. Target the bright, high-contrast edges of the drilled log holes and the perimeter of the wooden shelf dividers to remove the specific luminous halos that are causing the 'mushy' appearance and loss of definition. Apply a precise deconvolution filter that restores the sharpness of the bark grain and the brick perforations while strictly forbidding the use of aggressive sharpening kernels that generate ringing artifacts. Ensure the model does not hallucinate lost details in the dark, low-contrast areas, preserving the natural, organic look of the insect habitat without creating artificial, hard edges.

Demosaicking (Hexadeca)

sample grids: p. 44

EXPERT Full-color image reconstruction from a sparse hexadeca sampling pattern.

Shared Prompt.

Task: Demosaic this hexa-deca RGGB Bayer pattern image. Reconstruct the full-color RGB image from the raw sensor data captured with a 4×4 hexa-deca RGGB color filter array. Input: A hexadeca (8x8 super-pixel) Bayer mosaic image. Sensor Layout: Each pixel records only one color channel. The 8×8 repeating super-pixel tile is arranged as: R R R R G G G G R R R R G G G G R R R R G G G G G G G G B B B B G G G G B B B B G G G G B B B B G G G G B B B B Each 4×4 block of identical color values forms a "hexa-deca" sub-pixel. Your task is to reconstruct the two missing color channels at every pixel position from this very coarsely sampled color mosaic. Constraints: - Macro-Block Handling: Effectively interpolate across the large 4×4 homogeneous RGGB color blocks without introducing blocky artifacts or severe false colors at object boundaries. - Artifact Suppression: Strictly minimize aliasing and severe color fringing, which are highly prone in 4×4 arrays. - Spatial Resolution Recovery: Reconstruct fine spatial details while preserving natural gradients. Avoid artificial smoothing or painterly textures often caused by aggressive interpolation. - Zero Hallucination: Do not invent structural details or hallucinate textures to fill gaps between color samples. Output: A pristine, full-color RGB image reconstructed faithfully from the 4×4 hexa-deca RGGB Bayer array.

PLANNER Full-color image reconstruction from a sparse hexadeca sampling pattern.

Planner Prompt.

You are an expert computational imaging assistant. Your task is to analyze a raw sensor image captured with a 4×4 Hexadeca RGGB Bayer color filter array (CFA) and write a highly specific "execution prompt" for a downstream demosaicking model. Input: A hexadeca (8x8 super-pixel) Bayer mosaic image. The sensor's 8×8 repeating super-pixel tile is arranged as: R R R R G G G G R R R R G G G G R R R R G G G G R R R R G G G G G G G G B B B B G G G G B B B B G G G G B B B B G G G G B B B B Each 4×4 block of identical color values forms a "hexa-deca" sub-pixel. Two of the three RGB channels are missing at every pixel location. Step 1: Analyze the Image Scan the Hexadeca RGGB Bayer mosaiced image. Because of the large 4×4 homogeneous color blocks, identify object boundaries, fine lines, and high-contrast transitions that are highly prone to severe color fringing and blocky interpolation artifacts. Step 2: Generate the Execution Prompt Write a 4-6 sentence instruction for the image editing model. Use highly descriptive and restrictive keywords. Your generated prompt MUST include: - Explicit identification that the input is a 4×4 Hexadeca RGGB Bayer pattern with the pixel layout described above. - Specific identification of the regions or boundary edges in the image that require careful macro-block interpolation to prevent color fringing (e.g., "Ensure smooth boundary transitions around the tree branches without blocky color bleeding"). - Strict negative constraints forbidding the model from blurring edges, generating false colors, or hallucinating artificial textures to fill the large sampling gaps. Goal: Your generated prompt should act as a perfect algorithmic blueprint, guiding the editing model to smoothly resolve the 4×4 macro-blocks without introducing severe perceptual or color errors.

Written per image by the planner; shown for the row-1 image of the grid on p. 45.

Gemini-3.1-Pro Prompt (Row 1 Image).

This input is a raw sensor image encoded with a 4×4 Hexadeca RGGB Bayer color filter array, characterized by an 8×8 repeating super-pixel tile containing 4×4 homogeneous macro-blocks (RRRR/GGGG top, GGGG/BBBB bottom). Execute a high-fidelity demosaicking process optimized for large macro-block spatial sampling to reconstruct the full-color image. It is critical to apply advanced edge-directed interpolation around the complex, overlapping network of fine tree branches against the bright sky and the high-contrast painted lines curving along the road, ensuring smooth boundary transitions without introducing blocky color bleeding or zippering artifacts. Strictly preserve the sharpness of the structural edges, including the tree trunks and pavement cracks, without resorting to low-pass blurring. The model is expressly forbidden from generating false chromatic aliases, localized color fringing, or hallucinating artificial foliage or road textures to fill the severe informational gaps left by the missing color channels.

GPT-5 Prompt (Row 1 Image).

Input is a 4×4 Hexadeca RGGB Bayer CFA with an 8×8 repeating super-pixel arranged exactly as: R R R R G G G G / R R R R G G G G / R R R R G G G G / R R R R G G G G / G G G G B B B B / G G G G B B B B / G G G G B B B B / G G G G B B B B; two color channels are missing at every pixel. Perform macro-block-aware, edge-directed interpolation, giving special care to high-contrast boundaries: the thin tree branches against the bright sky, the road edges and center double lines, fence posts/wires, shrub outlines, pavement crack edges, and the distant horizon. Ensure smooth boundary transitions in these areas without blocky color bleeding; use green-channel gradient guidance and cross-channel color-ratio consistency to choose interpolation direction across 4×4 blocks. Fill missing channels with directional interpolation and deblocking only within uniform regions (open sky and flat asphalt), never across detected edges or fine lines. Strict constraints: do not blur edges, do not generate false colors, do not hallucinate textures or grain to fill the large sampling gaps; avoid zippering, ringing, checkerboard artifacts, and preserve true luminance structure and natural chroma.

Qwen3.5-35B-A3B Prompt (Row 1 Image).

Process the provided raw sensor image which utilizes a non-standard 4x4 Hexadeca RGGB Bayer color filter array, characterized by large 4x4 homogeneous sub-pixels arranged in quadrants (Red, Green, Green, Blue) within an 8x8 repeating super-pixel tile. Focus your demosaicking algorithm on the high-frequency details of the deciduous trees on the left and the thin, intricate branches against the bright sky, as these regions are most susceptible to visible macro-block grid patterns. Strictly avoid "blocky" interpolation artifacts where color values jump abruptly between the 4x4 color blocks, ensuring smooth boundary transitions around the tree limbs to prevent severe color fringing where dark branches meet the background. Do not apply aggressive smoothing that would blur the texture of the cracked asphalt road or hallucinate false colors in the sky to fill the large sampling gaps.

Demosaicking (Quad)

sample grids: p. 45

EXPERT Full-color image reconstruction from a quad Bayer sampling pattern.

Shared Prompt.

Task: Demosaic this quad RGGB Bayer pattern image. Reconstruct the full-color RGB image from the raw sensor data captured with a 2×2 quad RGGB Bayer color filter array. Input: A quad (4x4 super-pixel) Bayer mosaic image. Sensor Layout: Each pixel records only one color channel. The 4×4 repeating super-pixel tile is arranged as: R R G G R R G G G G B B G G B B Each 2×2 block of identical color values forms a "quad" sub-pixel. Your task is to reconstruct the two missing color channels at every pixel position from this sparse color mosaic. Constraints: - Artifact Suppression: Minimize aliasing, moiré patterns, and "zippering" artifacts along high-frequency edges. - Spatial Resolution Recovery: Accurately reconstruct fine spatial details without artificially blurring or over-sharpening the image. - Color Fidelity: Prevent false color generation in areas of high-contrast texture. Ensure that the reconstructed RGB values perfectly map to the original scene's luminance and chrominance. - Zero Hallucination: Do not invent structural details that are not present in the raw subsampled pixel data. Output: A pristine, full-color RGB image reconstructed faithfully from the 2×2 quad RGGB Bayer array.

PLANNER Full-color image reconstruction from a quad Bayer sampling pattern.

Planner Prompt.

You are an expert computational imaging assistant. Your task is to analyze a raw sensor image captured with a 2×2 Quad RGGB Bayer color filter array (CFA) and write a highly specific "execution prompt" for a downstream demosaicking model. Input: A quad (4x4 super-pixel) Bayer mosaic image. The sensor's 4×4 repeating super-pixel tile is arranged as: R R G G R R G G G G B B G G B B Each 2×2 block of identical color values forms a "quad" sub-pixel. Two of the three RGB channels are missing at every pixel location. Step 1: Analyze the Image Scan the Quad RGGB Bayer mosaiced image for high-frequency structural details (like thin lines, grids, or dense foliage) that are highly susceptible to aliasing, moiré, and false coloring during the 2×2 interpolation process. Step 2: Generate the Execution Prompt Write a 4-6 sentence instruction for the image editing model. Use highly descriptive and restrictive keywords. Your generated prompt MUST include: - Explicit identification that the input is a 2×2 Quad RGGB Bayer pattern with the pixel layout described above. - Specific identification of the regions or textures in the image that require careful artifact suppression (e.g., "Pay special attention to the striped shirt to prevent moiré"). - Strict negative constraints forbidding the model from blurring edges, generating false colors, or hallucinating artificial textures. Goal: Your generated prompt should act as a perfect algorithmic blueprint, guiding the editing model to cleanly interpolate the missing RGB channels without introducing perceptual errors.

Written per image by the planner; shown for the row-1 image of the grid on p. 46.

Gemini-3.1-Pro Prompt (Row 1 Image).

This input is a 2x2 Quad RGGB Bayer mosaiced image. Reconstruct the full-color image by applying a high-quality demosaicking algorithm designed specifically for the Quad Bayer layout (R R G G / R R G G / G G B B / G G B B). Pay special attention to the brick wall background and the vertical and horizontal mortar lines, as these repetitive grid-like structures are highly susceptible to severe moiré and zippering artifacts during the 2x2 interpolation. Ensure sharp, artifact-free reconstruction of the fine details on the

saxophone and the subject's suit jacket. Strictly avoid blurring the structural edges, introducing false colors along the brick mortar, or hallucinating artificial textures within the dense grid patterns. Maintain structural integrity and color accuracy throughout the interpolation process.

GPT-5 Prompt (Row 1 Image).

Input is a 2×2 Quad RGGB Bayer mosaic with a 4×4 repeating tile: R R G G / R R G G / G G B B / G G B B; reconstruct the two missing channels at every pixel respecting this quad layout. Use edge-aware, directionally selective interpolation within and across quads to prevent zippering and stair-stepping. Pay special attention to the brick wall's dense horizontal/vertical grout grid, which is highly prone to aliasing, moiré, and false color; preserve sharp brick edges while suppressing rainbowing. Also handle the saxophone's shiny metal keys, mouthpiece, and narrow contours, plus jacket seams, to avoid color fringing and ringing near specular highlights and fine lines. Do not blur edges, do not introduce false colors, and do not hallucinate textures or patterns; maintain channel consistency and natural luminance throughout.

Qwen3.5-35B-A3B Prompt (Row 1 Image).

Execute demosaicking on this raw sensor data configured as a 2x2 Quad RGGB Bayer pattern, strictly adhering to the provided 4x4 repeating super-pixel tile layout where 2x2 blocks of identical colors are arranged (R R G G / R R G G / G G B B / G G B B). Focus heavily on the high-frequency texture of the brick wall background, interpolating the repeating mortar lines and brick edges to prevent severe moiré artifacts and false coloring that typically arise from this specific sampling grid. Simultaneously, reconstruct the fine metallic details of the saxophone keys and the smooth gradients of the musician's face and suit without introducing "zippering" or jagged edges along high-contrast boundaries. Apply edge-preserving interpolation to maintain the sharp definition of the suit fabric and individual bricks, strictly forbidding global blurring that would obscure these textures. Finally, ensure the output is free of artificial halos, false colors, or hallucinated textures, particularly in the shadowed areas beneath the saxophone and the dark clothing.

Demosaicking (Regular)

sample grids: p. 46

EXPERT Full-color image reconstruction from a regular Bayer sampling pattern.

Shared Prompt.

Task: Demosaic this regular RGGB Bayer pattern image. Reconstruct the full-color RGB image from the raw sensor mosaic. Input: A regular Bayer mosaic image. Sensor Layout: The repeating 2x2 Bayer tile is: R G G B Constraints: - Color fidelity: Reconstruct missing channels while preserving natural color relationships and avoiding false-color artifacts. - Edge-aware interpolation: Keep high-contrast edges sharp and suppress zippering or moire artifacts on fine repetitive textures. - Detail preservation: Recover fine spatial detail without excessive smoothing or synthetic over-sharpening. - Zero hallucination: Do not invent structures or textures not supported by the sampled mosaic. Output: A clean, full-color RGB image faithfully reconstructed from the regular RGGB Bayer input.

PLANNER Full-color image reconstruction from a regular Bayer sampling pattern.

Planner Prompt.

You are an expert computational imaging assistant. Your task is to analyze a raw sensor image captured with a regular 2x2 RGGB Bayer color filter array (CFA) and write a highly specific "execution prompt" for a downstream demosaicking model. Input: A regular Bayer mosaic image. The sensor's repeating 2x2 tile is: R G G B Step 1: Analyze the Image Scan the mosaiced input and identify high-frequency edges, thin structures, and repetitive textures that are most vulnerable to zippering, moire, and false-color artifacts during interpolation. Step 2: Generate the Execution Prompt Write a 4-6 sentence instruction for the image model. Use highly descriptive and restrictive keywords. Your generated prompt MUST include: - Explicit identification that the input is a regular 2x2 RGGB Bayer mosaic. - Specific guidance for edge-aware reconstruction in regions prone to color artifacts. - Strict negative constraints forbidding edge blur, false colors, and hallucinated textures. Goal: Your generated prompt should act as a precise demosaicking blueprint that recovers accurate RGB detail from the regular Bayer sampling pattern.

Written per image by the planner; shown for the row-1 image of the grid on p. 46.

Gemini-3.1-Pro Prompt (Row 1 Image).

Image Analysis The input image contains dense, wet bear fur and turbulent water. These high-frequency, intersecting structural lines and repetitive textures are highly susceptible to zippering, moiré, and false-color artifacts during interpolation. ### Execution Prompt Reconstruct a full-color RGB image from this raw sensor input, which utilizes a standard 2x2 RGGB Bayer color filter array. Employ strict directional, edge-aware interpolation to accurately resolve the high-frequency textures of the wet, matted bear fur and sharp water splashes. Utilize cross-channel correlation to maintain precise structural boundaries where the high-contrast fur meets the water. Strictly forbid the introduction of false colors, chromatic aliasing, or moiré patterns within the repetitive fur details. Finally, ensure zero zippering along sharp edges and entirely prohibit edge blurring or hallucinated textures in the final output.

GPT-5 Prompt (Row 1 Image).

Step 1 -- Image analysis: - The scene contains very fine, wet bear fur with dense, repetitive strands that are highly susceptible to zippering and moire if interpolated across the grain. - Sharp, high-contrast edges occur at the open mouth, teeth, nose, and ear outlines; these regions are prone to color fringing and edge blurring. - Water ripples and small specular highlights on the wet fur introduce repetitive high-frequency textures that can trigger false-color ringing. - Overlapping limbs and muzzle transitions form thin structures and curved boundaries that require directional interpolation. Step 2 -- Execution prompt for the demosaicking model: - The input is a regular 2x2 RGGB Bayer mosaic (tile: R G / G B). Perform edge-aware, directional interpolation using the green channel as the luminance anchor; estimate local gradient orientation and interpolate along fur strands and water ripple directions, not across strong edges. - Reconstruct red/blue chroma by enforcing cross-channel consistency: minimize chroma variation across detected luminance edges, keep chroma smooth where luminance is smooth, and maintain neutral highlights on teeth and specular wet areas. - Suppress zippering around thin fur, ear contours, and mouth boundaries with gradient-corrected interpolation and limited chroma-domain regularization (Laplacian or TV) without smoothing luminance detail. - Hard constraints: no edge blur, no false colors, no moire, no halos, and no hallucinated textures; preserve true RGB detail and accurate edge positions derived from the RGGB sampling pattern.

Qwen3.5-35B-A3B Prompt (Row 1 Image).

Step 1: Analyze the Image The input image displays a scene of two bears interacting in water, characterized by extremely high-frequency textures in the wet fur and chaotic, high-contrast edges in the splashing water. The repetitive, noisy texture of the fur is particularly vulnerable to "zippering" artifacts if the demosaicking algorithm fails to follow the local luminance gradients accurately. Furthermore, the thin, high-contrast structures of the bear's open mouth and teeth

---

Denoising (SIDD)

sample grids: p. 47

EXPERT Removal of smartphone sensor noise while preserving fine texture and color.

Shared Prompt.

Task: Perform high-fidelity subtractive denoising on the input image. Input: A noisy image from a smartphone camera. Constraints: - Zero Structural Change: You must maintain the exact original composition, framing, and aspect ratio. Do not zoom, crop, or reposition any elements. - Fidelity Preservation: Keep all existing lines, borders, and high-contrast edges perfectly sharp. Do not blur object boundaries. - Texture Retention: Remove stochastic sensor grain and digital noise, but preserve the underlying original surface textures. Avoid an over-processed, "plastic," or "waxy" look. - Luminance & Color Consistency: Maintain the exact original exposure, white balance, and color values of every pixel. Do not "enhance" the lighting or saturation. Output: A clean, noise-free version of the provided image that is pixel-accurate to the original structure.

PLANNER Removal of smartphone sensor noise while preserving fine texture and color.

Planner Prompt.

You are an expert computational imaging assistant. Your task is to analyze a degraded, noisy image and write a highly specific "execution prompt" for a downstream image-to-image diffusion model. Input: A noisy image from a smartphone camera. Step 1: Analyze the Image Identify the specific nature of the noise (e.g., color-channel salt-and-pepper noise, low-light sensor grain, compression artifacts, or luminance streaks). Identify the critical textures that must be saved (e.g., skin pores, fabric weave, architectural detail). Step 2: Generate the Execution Prompt Write a 4-6 sentence instruction for the image editing model. Use highly descriptive, aesthetic, and restrictive keywords. Your generated prompt MUST include: 1. A description of the specific noise to be removed. 2. A list of specific "anchor textures" in this image that must remain untouched. 3. Strict negative constraints to prevent the model from adding new content or changing colors. Goal: Your generated prompt should allow the editing model to act as a perfect inverse solver for this specific image's noise profile.

Written per image by the planner; shown for the row-1 image of the grid on p. 47.

Gemini-3.1-Pro Prompt (Row 1 Image).

Denoise this image by removing the pervasive, fine-grained sensor noise and chrominance speckling, characteristic of low-light smartphone photography. Crucially, preserve the subtle, woven texture present in the upper background region. Maintain the sharp, distinct edges of the colored squares and the precise geometric grid structure without any softening or blurring. Strictly avoid introducing any new details, textures, or patterns into the image. Ensure the original color palette, saturation, and luminance distribution of all elements remain absolutely unchanged.

GPT-5 Prompt (Row 1 Image).

Step 1 --- Image analysis - **Noise type:** Heavy **low-light smartphone sensor noise**, dominated by fine **luminance grain** and dense **RGB chroma speckling/mottling** across the entire frame, strongest in the dark black bars and the flat midtone color panels. It does **not** read as salt-and-pepper noise; it is more like high-ISO color blotchy grain with slight coarse shadow noise. - **Anchor textures to preserve:** the **subtle plaster/concrete-like surface texture** in the large gray upper region, the **crisp straight horizontal and vertical black divider bars**, the **hard-edged rectangular geometry**, and

the **matte, flat color fields** in the muted lavender, olive-brown, mauve-purple, dusty pink, and yellow-green blocks. **Step 2 --- Execution prompt** Remove only the **dense high-ISO low-light smartphone sensor noise**: fine luminance grain plus multicolored RGB chroma speckle/blotchy mottling spread over the whole image, especially inside the dark divider bars and the flat colored rectangles. Preserve these anchor textures exactly: the **faint plaster/concrete texture** in the large gray upper panel, the **razor-straight black grid bars**, the **hard geometric edges** of every rectangle, and the **matte uniform fills** of the muted lavender, olive-brown, mauve, dusty pink, and yellow-green regions. Apply precise denoising only, keeping all boundaries, alignment, proportions, and tonal separations unchanged, and do not over-smooth the intentional surface texture in the gray area. **Do not add new detail, patterns, gradients, reflections, shadows, brush texture, or sharpening halos; do not repaint or reinterpret any region.** **Absolutely no color drift**: no hue shift, no saturation change, no white-balance correction, no contrast restyling, no cropping, no warping, and no geometry alteration---produce the same image as a clean inverse of its current noise profile.

Qwen3.5-35B-A3B Prompt (Row 1 Image).

1. Remove the pervasive, high-ISO chromatic noise and fine luminance grain that creates a speckled, static-like overlay across the entire image, particularly obscuring the details in the upper grey region. 2. Preserve the subtle, rough texture of the grey upper section and the sharp, non-reflective finish of the thick black grid lines separating the colored blocks. 3. Ensure the solid color fields (blue, olive, purple, red, dark violet, yellow) are smoothed of their heavy speckle while maintaining their original matte, paper-like saturation and flat appearance. 4. Strictly forbid the model from adding new details, shifting hues, or blurring the crisp boundaries between the grid cells to prevent a "plastic" or over-smoothed look. 5. The result should be a clean, flatbed-scan quality image that restores clarity and color fidelity without losing the physical texture of the original medium.

FORWARD Realistic smartphone noise synthesis from a clean reference image.

Planner Prompt.

You are an expert in computational imaging, working on the image formation model for compressive sensing. Input: one clean smartphone image. Per-image metadata parsed from filename: - scene instance: `{SCENE_INSTANCE_NUMBER}` - scene number: `{SCENE_NUMBER}` - camera: `{CAMERA_CODE}` (`{CAMERA_NAME}`) - ISO: `{ISO_LEVEL}` - shutter code: `{SHUTTER_SPEED_CODE}` (approx exposure `{SHUTTER_EXPOSURE_SECONDS}`) - illuminant temperature: `{ILLUMINANT_TEMPERATURE_K}` K - brightness condition: `{ILLUMINANT_BRIGHTNESS_CODE}` (`{ILLUMINANT_BRIGHTNESS_LABEL}`) First infer and write the forward model equation, then write an execution prompt for a downstream image editor. Required response format: 1. `FORWARD_MODEL:` - Write a smartphone noise image-formation model with parameter discussion grounded in the provided metadata. - Explain how ISO, exposure proxy, illuminant, and brightness condition should modulate noise level/chromaticity. - Keep the model realistic for generic mobile camera noise, not synthetic artistic grain. 2. `EXECUTION_PROMPT:` - Write a strict 4-6 sentence prompt that instructs the image editor to generate exactly one noisy image from the clean input. - Require preservation of scene geometry and color identity while injecting plausible smartphone sensor noise. - Forbid blur-based degradation, content hallucination, color cast drift unrelated to metadata, text overlays, and extra outputs. Hard constraints: - Equation-first ordering is mandatory: `FORWARD_MODEL` must appear before `EXECUTION_PROMPT`. - This is forward noise synthesis (clean -> noisy), not denoising. - Keep padded black regions black if present.

Written per image by the planner; shown for the row-1 image of the grid on p. 47.

Gemini-3.1-Pro Prompt (Row 1 Image).

FORWARD_MODEL: The noise synthesis follows a heteroscedastic (signal-dependent) image formation model to approximate the sensor noise characteristics of a Google Pixel smartphone before heavy multi-frame denoising. The observed noisy pixel intensity Y_c for a given color channel $c \in \{R, G, B\}$ is modeled as the clean signal X_c corrupted by additive noise N_c: $Y_c = X_c + N_c$ $N_c \sim \mathcal{N}(0, \sigma_c^2(X_c))$ $\sigma_c^2(X_c) = K \cdot X_c + \sigma_{read}^2$ Where: * $\sigma_c^2(X_c)$ is the total variance at signal level X_c. * $K \cdot X_c$ models the Poisson-distributed photon shot noise, approximated here as a Gaussian variance proportional to the signal. The parameter K represents overall system gain. * σ_{read}^2 models the signal-independent Gaussian read noise from the sensor electronics. **Parameter Discussion driven by Metadata:** * **ISO (00800) & Brightness (Low Light):** An ISO of 800 on a small mobile sensor indicates significant analog and digital gain (K is large) applied to compensate for the 'L' (low light) condition. This will dramatically inflate both the read noise variance (σ_{read}^2) and the shot noise component, resulting in a low Signal-to-Noise Ratio (SNR). The noise will be highly visible, particularly in the darker patches of the color checker where read noise dominates. * **Shutter Code (00640 / ~1/640s):** This is a relatively fast shutter speed for a low-light environment. A short integration time means very few photons are collected (low X_c), forcing reliance on the high ISO gain and exacerbating the dominance of sensor noise over the actual light signal. * **Illuminant Temperature (5500 K):** A 5500K illuminant represents standard daylight or electronic flash, which is relatively neutral. Because the physical illuminant is balanced, the camera's Image Signal Processor (ISP) does not need to apply extreme, asymmetric digital white balance gains to the Red or Blue channels. Consequently, the noise spectrum should remain relatively balanced across the RGB channels, avoiding the heavy blue-channel chroma noise typically seen under tungsten (incandescent) lighting. **EXECUTION_PROMPT:** Generate exactly one noisy image from the provided clean input by simulating the sensor noise profile of a Google Pixel camera

operating at ISO 800 with a 1/640s shutter speed in low light. Apply a signal-dependent heteroscedastic noise model where overall variance is strongly amplified by the high ISO, creating prominent luminance grain and subtle low-frequency chroma noise. Because the illuminant is a neutral 5500K, distribute the noise relatively evenly across the RGB channels without extreme asymmetric color channel amplification. The noise must be most pronounced in the darker, signal-starved regions of the color checker, accurately reflecting real-world photon and read noise limits. Strictly maintain the original image geometry, sharp structural borders, and underlying color identity; do not introduce blur, structural hallucinations, or text overlays. Finally, if the image contains any completely unexposed padded black border regions, they must remain absolutely pure black (RGB 0,0,0) with zero noise injected.

GPT-5 Prompt (Row 1 Image).

FORWARD_MODEL: - Let x_srgb be the clean image. Convert to linear RGB, x = Γ^{-1}(x_srgb). The noisy image is produced by a Poisson--Gaussian sensor model with gain and fixed-pattern terms: y_lin = clip(m ⊙ (g · x) + η_shot(x,g,t) + η_read(g,L) + b_rowcol(g,L)) , y = Γ(y_lin). Where g = ISO/ISO0 with ISO0≈100 (here g≈8), t is exposure (≈1/640 s), m=1+ζ is PRNU (ζ~N(0,σ_prnu^2)), η_shot_c ~ N(0, g^2·α_c·t·x_c) is signal-dependent shot noise per channel c, η_read_c ~ N(0, β_c·g^2·f_L) is signal-independent read/dark noise, and b_rowcol adds weak row/column banding. α_c encodes channel photon efficiency under the illuminant; for 5500 K daylight on a GP RGGB sensor, α_G≈1, α_R≈0.9, α_B≈0.8 so blue is slightly noisier. Low-light condition L increases f_L>1 (e.g., 1.5--2×) and modest banding. The short exposure (1/640 s) lowers photon counts, boosting the relative shot noise; ISO 800 raises both shot and read noise after analog gain. Keep padded black regions at zero (no offset) so blacks remain black; apply only mild JPEG quantization if desired, without changing color balance. EXECUTION_PROMPT: - Generate exactly one noisy version of the input image consistent with a Google Pixel shot at ISO 800, 1/640 s in low light under a 5500 K illuminant. - Work in linear RGB and inject Poisson--Gaussian noise: signal-dependent shot noise scaled by the gain (~8×) and short exposure, plus signal-independent read/dark noise with slightly stronger variance in the blue channel; add subtle PRNU and faint row/column banding. - Preserve scene geometry, edges, and all object colors; the mean color/white balance should remain unchanged and any padded black areas must stay black. - Do not introduce blur, sharpening, hallucinated content, text/graphics overlays, or color casts unrelated to the metadata. - Return a single image at the original resolution that exhibits realistic smartphone noise appropriate for low-light ISO 800 conditions.

Qwen3.5-35B-A3B Prompt (Row 1 Image).

FORWARD_MODEL: The image formation model for a smartphone sensor in low-light conditions is approximated by a signal-dependent noise process followed by amplification and quantization. Let X denote the clean linear radiance image. The noisy output Y is modeled as: $$ Y = \mathcal{Q} \left(G_{ISO} \cdot \left(\mathcal{P}\left(\frac{X}{G_{ISO}} \right) + \mathcal{N}(0, \sigma_{read}^2) \right) \right) $$ where $\mathcal{P}$ represents Poisson-distributed photon shot noise and $\mathcal{N}$ represents Gaussian read noise. The parameter G_{ISO} corresponds to the amplification factor for ISO 800, which significantly elevates the visibility of σ_{read} (read noise). The shutter time ($1/640$ s) limits photon integration, reducing the mean signal intensity and thereby increasing the relative variance of the shot noise component $\mathcal{P}$. The illuminant temperature (5500 K) establishes the white balance baseline, ensuring noise chromaticity remains largely neutral but potentially exhibiting higher variance in the blue channel typical of low-light sensor performance. The brightness condition 'L' confirms that the input signal X resides in a low-intensity regime, making the additive noise term dominant and resulting in a grainy texture characteristic of high-ISO mobile photography. EXECUTION_PROMPT: Generate a single noisy image derived from the provided clean input of a color calibration chart. Apply a realistic smartphone noise texture consistent with high ISO (800) and low-light conditions, ensuring the grain is visible across both the colored squares and the background texture. Strictly preserve the sharp geometric boundaries of the black frame and the distinct color identities of the patches (blue, green, purple, red) without introducing blur or distortion. Do not hallucinate new content, add text overlays, or drift the color cast significantly away from the 5500K daylight baseline. Ensure that any padded black regions remain solid black and that the overall composition remains faithful to the original scene geometry.

Denoising Ablation (Synthetic Noise Sweeps)

sample grids: p. 48

EXPERT Removal of synthetic sensor noise at ten severity levels (the Expert ablation sweeps).

Shared Prompt.

Task: Perform high-fidelity subtractive denoising on the input image. Input: A noisy synthetic image. Noise parameters are level={LEVEL}, peak={PEAK}, sigmaFix={SIGMAFIX}. Constraints: - Zero Structural Change: You must maintain the exact original composition, framing, and aspect ratio. Do not zoom, crop, or reposition any elements. - Fidelity Preservation: Keep all existing lines, borders, and high-contrast edges perfectly sharp. Do not blur object boundaries. - Texture Retention: Remove stochastic sensor grain and digital noise, but preserve the underlying original surface textures. Avoid an over-processed, "plastic," or "waxy" look. - Luminance & Color Consistency: Maintain the exact original exposure, white balance, and color values of every pixel. Do not "enhance" the lighting or saturation. Output: A clean, noise-free version of the provided image that is pixel-accurate to the original structure.

Event-Based Reconstruction

sample grids: p. 49

EXPERT Intensity-image reconstruction from event-camera measurements.

Shared Prompt.

```
Task: Reconstruct an intensity image from a single event representation. Input: One event image `{PRIMARY_IMAGE_FILENAME}`, whose filename suffix indicates temporal accumulation scale (`*_16.png`, `*_32.png`, or `*_full.png`). Suffix interpretation: `_16` = short accumulation window, `_32` = medium accumulation window, `_full` = full accumulation window. For this sample, use accumulation scale `{EVENT_ACCUMULATION_SCALE}`. Event encoding note: - The event image is accumulated from events `(x, y, t, p)` over the selected window. - Polarity `p=1` (ON) means local brightness increased; `p=0` (OFF) means local brightness decreased. - Per pixel, ON contributes `+1` and OFF contributes `-1`, then values are rendered around a neutral gray baseline `128`. - Pixels above `128` indicate ON-dominant activity; pixels below `128` indicate OFF-dominant activity. - This input encodes brightness change dynamics, not absolute scene intensity. Constraints: - Scale awareness: Respect the input accumulation scale (16, 32, or full) and reconstruct the corresponding intensity frame for that same scale. - Edge and texture fidelity: Preserve fine contours and high-frequency details while suppressing event noise and sparsity artifacts. - Photometric plausibility: Produce a natural-looking intensity image without artificial over-sharpening, checkerboard artifacts, or contrast exaggeration. - Zero hallucination: Do not invent objects or scene structures not supported by the provided event input. Output: A clean, structurally accurate intensity image reconstructed from the provided event input.
```

PLANNER Intensity-image reconstruction from event-camera measurements.

Planner Prompt.

```
You are an expert computational imaging assistant. Your task is to analyze a single event representation and write a highly specific "execution prompt" for a downstream event-to-image reconstruction model. Input: One event image `{PRIMARY_IMAGE_FILENAME}` with suffix `*_16`, `*_32`, or `*_full` indicating temporal accumulation scale. Suffix interpretation: `_16` = short accumulation window, `_32` = medium accumulation window, `_full` = full accumulation window. For this sample, the scale is `{EVENT_ACCUMULATION_SCALE}`. Event encoding note: - The input is built from events `(x, y, t, p)` accumulated over the selected window. - `p=1` (ON) indicates local brightening; `p=0` (OFF) indicates local darkening. - ON/OFF events are summed as signed counts per pixel (`+1` / `-1`) and rendered around baseline gray `128`. - Values above/below `128` represent ON-dominant/OFF-dominant change respectively. - The representation encodes temporal brightness change, not direct intensity. Step 1: Analyze the Event Evidence Use the provided suffix-derived scale `{EVENT_ACCUMULATION_SCALE}`, then inspect reliable edges, motion traces, and regions where sparse events may cause missing texture. Step 2: Generate the Execution Prompt Write a 4-6 sentence instruction for the image model. Use highly descriptive and restrictive keywords. Your generated prompt MUST include: - Explicit instruction to reconstruct the intensity image corresponding to the same input accumulation scale. - Specific guidance on preserving sharp boundaries and recovering plausible texture from sparse event evidence. - Strict negative constraints forbidding ghosting, over-smoothed surfaces, and hallucinated scene content. Goal: Your generated prompt should act as a robust reconstruction blueprint that turns a single event accumulation image into a faithful intensity reconstruction.
```

Written per image by the planner; shown for the row-1 image of the grid on p. 49.

Gemini-3.1-Pro Prompt (Row 1 Image).

```
**Execution Prompt:** Reconstruct a grayscale intensity image corresponding to this medium accumulation window event input. Extract and sharpen the faint rectangular geometric object with parallel lines in the lower left and the curved structures above it. Recover plausible surface textures from this highly noisy, sparse data while preserving distinct, sharp object boundaries. Strictly prohibit motion ghosting, over-smoothed surfaces, and the hallucination of scene content unsupported by the raw event traces.
```

GPT-5 Prompt (Row 1 Image).

```
Reconstruct the intensity image that corresponds exactly to the same medium temporal accumulation scale (32) as the provided event frame. Use the signed event evidence: lighten regions with ON-dominant counts, darken regions with OFF-dominant counts, and keep areas with little or no events near neutral mid-gray. Preserve sharp boundaries and continuous contours, particularly along the prominent oval/circular edge in the upper center and the rectangular slab/stripe in the lower center, removing speckle while avoiding blur of these edges. Recover plausible surface texture only where event density supports it using edge-aware, anisotropic smoothing; in sparsely populated zones, fill with low-contrast gradients without inventing fine patterns. Strictly avoid ghosting, double edges, halos, over-smoothed plastic surfaces, or any hallucinated objects or scene content not supported by the events.
```

Qwen3.5-35B-A3B Prompt (Row 1 Image).

```
Reconstruct the intensity image corresponding to the medium accumulation scale (32), treating the input grain as a distribution of local brightness changes where darker pixels indicate dominant OFF events. Focus on recovering the sharp, high-contrast edges of the geometric structures visible at the bottom and the curved element in the top right, ensuring these features remain distinct and not blurred by the surrounding noise. Convert the
```

stochastic event density into plausible, continuous surface textures for the identified objects while preserving the neutral gray background where no significant change occurred. Strictly forbid the introduction of ghosting artifacts or temporal smearing, as this is a single accumulation snapshot. Do not hallucinate details in low-signal areas; maintain the structural integrity of the scene based solely on the provided event density.

HDR Imaging

sample grids: p. 49

EXPERT Burst HDR fusion for preserving both highlight and shadow detail.

Shared Prompt.

Task: Merge this mobile HDR+ burst into a single high dynamic range (HDR) image. Combine complementary information across burst frames to produce a natural tone-mapped result with full detail in both shadows and highlights. Input: A sequence of `{NUM_INPUT_IMAGES}` burst frames from one HDR+ capture (`{INPUT_FRAME_IDS}`; files: `{INPUT_FILENAMES}`). HDR+ capture assumptions: HDR+ bursts typically contain 2-10 raw frames from the same scene, generally captured with the same exposure time and gain on mobile rolling-shutter sensors; inter-frame gaps are usually <= 33 ms unless a frame is dropped. Constraints: - Burst Fusion: Perform robust multi-frame alignment and merge on same-exposure burst frames; reduce noise by aggregating consistent signal across frames while preserving scene structure. - Dynamic Range Recovery: Recover highlight and shadow detail from inter-frame signal variation after alignment and merge. Do not assume explicit exposure bracketing is present in every burst. - Natural Tone Mapping: Ensure the final tone-mapped image maintains natural local contrast and physically plausible lighting. Strictly avoid unnatural "halos" around high-contrast edges, glowing shadows, or hyper-saturated "grunge" effects typical of exaggerated HDR processing. - De-Ghosting: If there is slight subject/camera motion across burst frames, align static content and suppress ghosting on moving regions, using a stable reference-frame strategy. - Zero Hallucination: Do not invent structural details in saturated highlights or deep shadows if not supported by any burst frame. Output: A photorealistic, perfectly exposed single image featuring maximum dynamic range, natural contrast, and zero blending or alignment artifacts.

PLANNER Burst HDR fusion for preserving both highlight and shadow detail.

Planner Prompt.

You are an expert computational imaging assistant. Your task is to analyze one HDR+ burst and write a highly specific "execution prompt" for a downstream HDR fusion model. Input: `{NUM_INPUT_IMAGES}` burst frames from one capture (`{INPUT_FRAME_IDS}`; files: `{INPUT_FILENAMES}`). HDR+ capture assumptions: HDR+ bursts typically contain 2-10 raw frames from the same scene, generally captured with the same exposure time and gain on mobile rolling-shutter sensors; inter-frame gaps are usually <= 33 ms unless a frame is dropped. Step 1: Analyze the Burst Examine the burst to identify recoverable highlights/shadows after alignment+merge, motion regions that may ghost, and low-SNR areas that require robust multi-frame denoising. Step 2: Generate the Execution Prompt Write a 4-6 sentence instruction for the image editing model. Use highly descriptive and restrictive keywords. Your generated prompt MUST include: - Explicit instruction to align and merge the same-exposure burst into one HDR+ style output. - Specific identification of critical regions: highlight structures, deep-shadow textures, and moving objects requiring de-ghosting. - Strict negative constraints forbidding unnatural HDR halos, glowing shadows, or ghosting artifacts. Goal: Your generated prompt should act as a robust algorithmic blueprint for HDR+ style burst fusion (alignment, merge, denoise, tone-map) into a naturally lit, high-dynamic-range scene.

Written per image by the planner; shown for the row-1 image of the grid on p. 50.

Gemini-3.1-Pro Prompt (Row 1 Image).

Execute a robust alignment and merge of this 4-frame same-exposure burst to synthesize a clean, naturally lit HDR+ style image from a severely underexposed capture. Focus aggressive multi-frame denoising and tone mapping on the deep shadows enveloping the complex machinery to reveal intricate mechanical details, surface textures, and the text on the adjacent informational plaque. Carefully manage the dynamic range to preserve the brighter reflections on the acrylic casing, polished metal components, and the brightly lit background display case without clipping. Ensure precise sub-pixel alignment to maximize the signal-to-noise ratio in this low-light static scene, effectively mitigating any slight camera shake. Strictly avoid unnatural HDR halos around the machine's complex geometry, prevent muddy or artificially glowing shadows during the lifting process, and ensure zero ghosting or alignment artifacts on the sharp mechanical edges.

GPT-5 Prompt (Row 1 Image).

Align and merge the four same-exposure burst frames (N000, N002, N004, N006) using sub-pixel optical-flow with rolling-shutter aware warping, then fuse with robust outlier rejection for HDR+ style output. Protect highlight detail in the illuminated display case at the top-left, the bright wall/shelf at the top-right, and the specular reflections on the acrylic covers and metal rods/knobs of the machine; avoid clipping while maintaining clean edge contrast. Lift and denoise deep-shadow textures in the machine's base, pipes and arm recesses, the pedestal, the radiator area behind, and the text panel on the left, preserving fine machining marks and printed text without waxy smoothing. Apply strong de-ghosting for any micro-parallax or flicker in thin rods, plexiglass edges, and reflective surfaces,

rejecting inconsistent pixels and preventing double edges. Tone-map to a natural museum lighting look with neutral colors and moderate local contrast. Strictly forbid HDR halos, glowing or lifted shadows around bright boundaries, over-sharpening, color shifts, and any ghosting artifacts.

Qwen3.5-35B-A3B Prompt (Row 1 Image).

Align and merge the four burst frames into a single HDR image, focusing on lifting the heavy shadows of the large green mechanical device to reveal intricate gear and lever details while retaining the integrity of the bright overhead lights in the background display case. Pay specific attention to the clear acrylic protection boxes, ensuring their specular highlights are crisp and free of blooming or halo artifacts. Apply aggressive temporal denoising to the dark, underexposed regions of the machine body and floor to reduce noise while preserving sharpness. Strictly enforce natural tone mapping to avoid the 'glowing' shadow look typical of aggressive HDR, and ensure no ghosting artifacts appear from minor inter-frame camera jitter around the static object edges.

FORWARD Generation of a plausible single HDR+ burst frame from a reference HDR image.

Planner Prompt.

You are an expert in computational imaging, working on the image formation model for compressive sensing. Input: one HDR/tone-mapped reference image. Task: generate one plausible HDR+ burst frame (single output image), not the full burst. First infer and write the forward model equation, then write an execution prompt for a downstream image editor. Required response format: 1. 'FORWARD_MODEL:' - Write an exposure and sensor model for one burst frame, for example mapping irradiance to a captured frame under a certain exposure setting (ISO, shutter speed, aperture, gain) with rolling-shutter sampling and sensor noise. - State that HDR+ burst frames are typically same-exposure/same-gain with short inter-frame gaps and slight handheld perturbations. - Describe how one frame should differ from the latent radiance reference (noise, tiny pose shift, rolling-shutter effects) while remaining physically plausible. 2. 'EXECUTION_PROMPT:' - Write a strict 4-6 sentence prompt that instructs the image editor to generate exactly one burst frame from the provided reference. - Require realistic mobile sensor behavior: subtle jitter/misalignment cues, plausible noise, and no dramatic geometry change. - Forbid multi-frame output, exaggerated HDR halos, stylization, text overlays, and unrelated content additions. Hard constraints: - Equation-first ordering is mandatory: 'FORWARD_MODEL' must appear before 'EXECUTION_PROMPT'. - Output must be one representative burst frame only. - Keep padded black regions black if present.

Written per image by the planner; shown for the row-1 image of the grid on p. 50.

Gemini-3.1-Pro Prompt (Row 1 Image).

FORWARD_MODEL: The formation of a single burst frame in an HDR+ pipeline (e.g., constant-exposure burst photography) can be modeled by mapping the dynamic scene irradiance to a discrete, noisy digital signal. For a pixel location x and frame index k, the captured raw signal $Y_k[x]$ is given by: $Y_k[x] = \text{Quantize}\left(g \cdot \int_{t_k + t_{rs}(x)}^{t_k + t_{rs}(x) + \Delta t} E(W_k(x, t), t) dt + \mathcal{P}(\lambda) + \mathcal{N}(0, \sigma^2) \right)$ Where: * $E(W_k(x, t), t)$ is the latent scene irradiance at time t, warped by W_k which models slight handheld camera shake and scene motion relative to the reference frame. * t_k is the base timestamp of the k-th frame. * $t_{rs}(x)$ models the rolling-shutter temporal offset for the row containing pixel x. * Δt is the short exposure time (shutter speed), kept constant across the burst to prevent highlight clipping and simplify alignment. * g is the constant analog/digital gain (ISO). * $\mathcal{P}(\lambda)$ represents signal-dependent Poisson shot noise. * $\mathcal{N}(0, \sigma^2)$ represents signal-independent Gaussian read noise. * $\text{Quantize}(\cdot)$ represents the ADC conversion to digital numbers. In HDR+ systems, all frames in the burst share the exact same short exposure (Δt) and gain (g). Therefore, a single frame from this burst differs from the final tone-mapped reference by being significantly noisier (especially in shadow regions due to the short exposure), exhibiting a flat/linear or unpolished tonal curve before merging, and containing microscopic misalignments (a few pixels of translation/rotation) or slight rolling-shutter skew caused by hand tremor during the short inter-frame gaps. **EXECUTION_PROMPT:** Act as a computational photography simulator and generate exactly one realistic, unmerged burst frame derived from the provided HDR reference image. Degrade the polished HDR appearance by applying characteristic mobile sensor noise, specifically introducing visible luma and chroma noise that heavily impacts the shadow regions to simulate a short, constant-exposure capture. Introduce a very subtle, almost imperceptible global shift or mild rolling-shutter skew to mimic natural handheld camera jitter, without causing dramatic geometric changes. Flatten the localized tone mapping to make the lighting look like a single, slightly underexposed raw-like capture rather than a merged HDR result. Strictly forbid the generation of multi-frame collages, exaggerated HDR halos, artistic stylization, text overlays, or new structural content, and ensure any padded black borders remain pure black.

GPT-5 Prompt (Row 1 Image).

FORWARD_MODEL: - One HDR+ burst frame k is modeled from the latent scene irradiance E(x) via a rolling-shutter sensor with constant exposure: ISO G_iso, shutter T_e, aperture f (effective area A $\propto$ 1/f^2), analog gain g_a and digital gain g_d. The row y starts integrating at t_row(y) = t0 + y·Δt_rs, and each pixel u = (x,y) integrates for T_e. - Expected photons: $\mu_k(u) = \eta \cdot A \cdot s(u) \cdot \int_{t_{row}(y)}^{t_{row}(y)+T_e} E(\pi(u;\theta(t)))\, dt$, where s(u) is lens shading, η is QE, and $\pi(u;\theta(t))$ maps the pixel to the scene under a slight hand-held pose that drifts during readout (tiny translation/rotation). - Measurement (RAW): $z_k(u) = Q\{ g_d [\text{Poisson}(\mu_k(u))] + N(0, \sigma_r^2(G_{iso})) \} + b_{black}$, with Q quantization, Poisson shot noise, Gaussian read noise, and black level b_black. Rolling-shutter induces

subtle vertical skew; demosaic/WB later produces the rendered frame. - HDR+ bursts typically use same exposure/same gain across frames with short inter-frame gaps (≈10--20 ms) and small handheld perturbations. Compared to the tone-mapped reference radiance, a single frame should exhibit slightly higher noise (more visible in shadows), tiny sub-pixel misalignment and minor RS skew, while preserving scene geometry and content. - Plausible parameters for this daylight scene: f/2.0, ISO ≈160, T_e ≈1/120 s, per-frame pose jitter ≤1 px translation and ≤0.3° rotation, row readout Δt_rs such that full-frame readout ≈1/30 s. EXECUTION_PROMPT: Generate exactly one realistic HDR+ burst frame from the provided reference, assuming ISO 160, 1/120 s shutter, f/2.0, same exposure/gain as other (unseen) frames. Apply a sub-pixel handheld shift (≤1 px) with ≤0.3° rotation and mild rolling-shutter vertical skew consistent with a ˜1/30 s full-frame readout; keep global geometry nearly unchanged. Add physically plausible Poisson-Gaussian sensor noise with more visible grain in shadowed regions and very light highlight quantization, without introducing HDR halos. Do not stylize, sharpen aggressively, add text/graphics, or change colors unnaturally; keep any padded black regions pure black. Output a single frame only, aligned closely to the reference but with the subtle jitter and noise cues of a handheld mobile capture.

Qwen3.5-35B-A3B Prompt (Row 1 Image).

FORWARD_MODEL: The forward model for a single HDR+ burst frame I_{burst} can be described as a mapping from the latent scene radiance L through the camera sensor physics, incorporating exposure parameters and noise. The equation is given by: $$I_{burst} = \mathcal{Q} \left(\mathcal{R} \left(L \cdot \frac{g \cdot \Delta t}{\tau} \right) + \eta_{shot} + \eta_{read} \right) \circ \mathcal{T}_{jitter}$$ where $\mathcal{R}$ is the sensor's non-linear response function, g is the ISO gain, Δt is the exposure time, τ is the normalization constant, $\mathcal{Q}$ is the quantization to digital values, η represents Poisson-Gaussian noise (shot and read noise), and $\mathcal{T}_{jitter}$ models the slight affine transformation due to handheld micro-movements. HDR+ burst frames are typically captured with short exposure times (Δt) and moderate gain to prevent saturation in high-dynamic-range scenes, resulting in a frame that is noisier and potentially slightly underexposed compared to the final merged HDR output. The generated frame should differ from the latent radiance reference by introducing realistic photon shot noise, sensor read noise, and a sub-pixel geometric shift to simulate the physical reality of a handheld burst capture, while strictly avoiding the global tone-mapping and sharpening present in the reference. EXECUTION_PROMPT: Generate a single, realistic mobile sensor burst frame derived from the provided reference image. The image should appear slightly underexposed compared to the final HDR result to simulate a short-exposure capture intended for multi-frame merging, with visible high-frequency sensor noise and slight chromatic aberration. Apply a subtle, random affine transformation (jitter) to simulate handheld micro-movements between frames, ensuring the subject's pose remains consistent but slightly misaligned relative to the background geometry. Preserve the original color balance and lighting conditions without applying aggressive tone-mapping, sharpening, or HDR halos. Do not output multiple frames, add text overlays, or alter the scene content, ensuring the output looks like a raw input for an HDR pipeline.

Inpainting

sample grids: p. 50

EXPERT Masked image completion while preserving scene structure and texture continuity.

Shared Prompt.

Task: Perform high-fidelity inpainting to seamlessly fill the missing (blacked-out/masked) regions of the input image. Input: An image with masked (missing) regions and its corresponding binary mask image. Constraints: - Mask Semantics: Treat pixels that are exactly pure black (#000000) in the masked image as missing regions to be inpainted. Use the provided binary mask as the authoritative region map, since non-missing content elsewhere may also contain dark/black pixels. - Strict Mask Adherence: You must not alter a single pixel outside of the explicitly masked regions. The original, unmasked image content must remain 100% identical in structure, color, and sharpness. - Seamless Boundary Blending: Ensure perfect continuity at the edges of the mask. There must be no visible seams, lighting shifts, or texture mismatches where the generated content meets the original pixels. - Semantic & Perspective Consistency: The generated content within the masked region must logically align with the surrounding visual context. Match the existing depth of field, perspective lines, and lighting direction perfectly. - Texture Matching: The newly generated pixels must mimic the exact noise floor, grain, and surface textures of the surrounding original image to avoid looking artificially smooth or digitally inserted. Output: A complete, unmasked version of the image where the missing gaps have been logically filled without any alteration to the original surrounding content.

PLANNER Masked image completion while preserving scene structure and texture continuity.

Planner Prompt.

You are an expert computational imaging assistant. Your task is to analyze an image with missing (blacked-out) regions and write a highly specific "execution prompt" for a downstream image inpainting model. Input: An image with masked (missing) regions and its corresponding binary mask image. Step 1: Analyze the Context Use the provided binary mask as the authoritative missing-region definition. In the masked image, pixels that are exactly pure black (#000000) indicate missing regions, but note that some valid scene

content may also be dark; rely on the mask to disambiguate. Then examine the visual information immediately surrounding the masked region and deduce what logical object, texture, or background is missing (e.g., "The mask covers the lower half of a brick wall and a wooden bench," or "The missing area should contain the continuation of a cloudy sky and the tip of a mountain"). Identify the lighting direction and depth of field. Step 2: Generate the Execution Prompt Write a 4-6 sentence instruction for the image editing model. Use highly descriptive and restrictive keywords. Your generated prompt MUST include: - An explicit description of exactly what objects, textures, or colors need to be generated inside the mask to complete the scene logically. - Specific instructions on lighting, perspective, or focus needed to match the surrounding unmasked areas. - Strict negative constraints forbidding the model from altering any pixels outside the designated mask or introducing logically inconsistent objects. Goal: Your generated prompt should act as a perfect contextual blueprint, allowing the editing model to seamlessly reconstruct the missing data without touching the original image.

Written per image by the planner; shown for the row-1 image of the grid on p. 51.

Gemini-3.1-Pro Prompt (Row 1 Image).

The mask primarily obscures portions of the woman's face, upper body, and the wall behind her, as well as a large section on the left side near the wall. Inpaint the missing sections of the woman's face and upper body, continuing her features, the brown hat, and the grey cardigan. Reconstruct the missing parts of the textured, beige wall behind her, matching the rough surface and lighting. Ensure the inpainted areas align seamlessly with the existing visible parts of the woman and the wall. The lighting should remain consistent, coming from the upper right, casting soft shadows on her face and the wall. Do not alter any areas outside the mask or introduce new objects not implied by the existing context.

GPT-5 Prompt (Row 1 Image).

Execution prompt for inpainting: - Fill the masked regions with the continuation of the scene: a seated vendor against a two-tone stucco wall, a tan felt hat, the person's natural face and dark braided hair, a gray wool sweater, clasped hands resting on a blue skirt with small white dot pattern, and a pile of pink/red onions with tangled roots and green stalks on a dark blue cloth with bits of straw on a cobblestone ground. - Preserve the wall's rough texture and the horizontal color break midway up the wall, and extend the ground's stones and scattered straw consistently under the produce. - Lighting should be soft, overcast daylight with gentle, diffuse highlights; shadows fall subtly to the right, matching the existing scene. - Keep the perspective at eye level and maintain moderate depth of field: the person and onions in sharp focus, background wall slightly softer but not blurred. - Do not alter any pixels outside the mask, do not introduce new objects, people, text, or different clothing styles, and avoid color shifts or changes to the wall seam, blanket edges, or existing contours.

Qwen3.5-35B-A3B Prompt (Row 1 Image).

Reconstruct the scene content obscured by the black shapes defined in the binary mask, specifically filling in the missing portions of the rough beige stucco wall in the upper left and center. Generate the continuation of the person's brown felt hat and grey knitted cardigan in the upper right and center areas, ensuring the fabric texture and folds align with the visible clothing. In the lower central region, restore the pile of red onions with green stalks and the surrounding straw on the ground, matching the organic shapes and colors. On the right side, complete the grey cardigan sleeve and the blue patterned skirt fabric, maintaining the soft focus and natural lighting of the outdoor setting. Strictly forbid any alterations to the pixels outside the mask boundaries, preserving the unmasked hands, the lower skirt, and the ground details exactly as they appear.

Lens Distortion Correction

sample grids: p. 51

EXPERT Geometric rectification of a lens-distorted photograph, straightening curved lines while preserving true perspective.

Shared Prompt.

Task: Correct lens distortion in the input photograph and output a geometrically rectified image. Input: One distorted image. Capture metadata: {LENS_METADATA} Constraints: - Distortion Analysis: Estimate radial and tangential distortion behavior from scene geometry, then perform lens correction conservatively. - Geometry Preservation: Straighten curved lines caused by lens distortion while preserving real perspective and object shapes. - Pad-Region Handling: If black padded borders are present, ignore those padded pixels during distortion estimation and correction decisions. - Output Padding Consistency: Keep padded border regions black in the final output at their corresponding locations. - Photometric Stability: Avoid introducing blur, ringing, or local stretch artifacts during warping. - Zero Hallucination: Do not invent scene structure not supported by the input image. Output: A lens-corrected image with preserved black pad regions.

PLANNER Geometric rectification of a lens-distorted photograph, straightening curved lines while preserving true perspective.

Planner Prompt.

```
You are an expert computational imaging assistant. Your task is to analyze a lens-distorted image and write a precise "execution prompt" for downstream lens correction. Input: One distorted image. Capture metadata: {LENS_METADATA} Step 1: Analyze Distortion Evidence Identify dominant distortion characteristics (radial/tangential tendencies), straight-line deviations, and any black padded border regions that should be excluded from analysis. Step 2: Generate the Execution Prompt Write a 4-6 sentence instruction for the downstream model. Your generated prompt MUST include: - Explicit instruction to estimate and correct lens distortion while preserving true scene perspective. - Specific guidance to ignore black padded border regions during correction inference but preserve those regions as black in the output image. - Constraints against geometric over-correction, boundary tearing, ringing, or texture hallucination. Goal: Produce an execution prompt that yields physically plausible lens correction with stable geometry and proper pad-region handling.
```

Written per image by the planner; shown for the row-1 image of the grid on p. 51.

Gemini-3.1-Pro Prompt (Row 1 Image).

```
Step 1: Analyze Distortion Evidence The image exhibits minor barrel distortion, typical of a 24mm wide-angle lens, though it is relatively well-controlled, likely due to in-camera or lens profile corrections applied to the original RAW file (if this is a JPEG) or inherent optical quality. Straight lines, such as the edge of the elevated highway structure, the railing in the foreground, and the edges of the buildings on the right, show a slight outward bowing, particularly noticeable towards the edges of the frame. The image has significant black padded borders at the top and bottom. Step 2: Generate the Execution Prompt Analyze the input image to estimate and correct the minor barrel distortion inherent to the 24mm focal length, ensuring that straight lines, such as the highway edges, railings, and building facades, are rendered as perfectly straight. It is critical to restrict the distortion analysis exclusively to the central image area and explicitly ignore the black padded borders at the top and bottom during parameter estimation. Apply the calculated correction only to the valid image content, maintaining the black padded regions unaltered in the final output. The correction must preserve the true scene perspective without introducing over-correction, ringing artifacts, texture hallucination, or boundary tearing, yielding a physically plausible and geometrically stable final image.
```

GPT-5 Prompt (Row 1 Image).

```
Step 1: Distortion Evidence - The wide-angle (24 mm) view shows dominant radial barrel distortion: long straight features such as the bridge deck, guardrail, and road edges bow outward near the frame edges. - Vertical elements (utility pole, building edges) exhibit slight outward bending toward the sides, with minor asymmetry suggesting small tangential/decentering components. - There are solid black padded bars at the top and bottom; these are non-image regions and should be excluded from model fitting. Step 2: Execution Prompt Estimate and correct lens distortion in this image, prioritizing removal of radial barrel distortion while preserving the true scene perspective and natural keystone from the camera viewpoint. Use the bridge edges, guardrail, building façades, utility pole, and power lines as straight-line references. Ignore the black padded bars at the top and bottom during distortion estimation, but keep those regions black and unchanged in the output image. Avoid geometric over-correction; do not force perspective rectification or introduce stretching that causes boundary tearing. Prevent ringing artifacts and any texture hallucination; maintain stable, physically plausible geometry across the frame. Keep overall image dimensions and content continuity after correction.
```

Qwen3.5-35B-A3B Prompt (Row 1 Image).

```
Estimate and correct radial lens distortion inherent to the 24mm wide-angle focal length, specifically straightening the utility pole, building edges on the right, and the overhead concrete structure to restore true scene perspective. Explicitly ignore the black padded border regions at the top and bottom during the geometric transformation inference, but ensure the final output image preserves these black regions exactly as they appear in the input without scaling or cropping them. Apply correction constraints to prevent geometric over-correction, boundary tearing, or ringing artifacts, ensuring the foreground railing and textured pavement remain stable and free of hallucination. The correction should maintain the natural depth cues and vanishing points of the urban environment while rectifying the barrel distortion.
```

Light-Field View Extrapolation

sample grids: p. 52

EXPERT Synthesis of the right-side sub-aperture view from the left and center views.

Shared Prompt.

```
Task: Predict a right-side novel view from two known light-field views. Input: Two aligned sub-aperture images ordered as left view and center view. Lenslet convention: 14x14 angular grid, 0-based [row,col] indices. View metadata: {VIEW_METADATA} Constraints: - Extrapolation Direction: Synthesize the target view outside the known pair (for this dataset: infer `R` from `[L, C]`), not interpolation inside the pair. - View Geometry Consistency: Preserve epipolar consistency and disparity geometry when projecting from known angular samples to the target view. - Occlusion Handling: Resolve disocclusions and view-dependent visibility changes using physically plausible scene structure. - Appearance Fidelity: Preserve texture, color, and local contrast without introducing ghosting or duplicated edges. - Zero Hallucination: Do not invent scene content not supported by either input view. Output: One extrapolated target view image that is geometrically and photometrically consistent with the provided left/center inputs and metadata.
```

PLANNER Synthesis of the right-side sub-aperture view from the left and center views.

Planner Prompt.

```
You are an expert computational imaging assistant. Your task is to analyze two light-field
views (left and center) and write a precise "execution prompt" for downstream view
extrapolation. Input: Two aligned sub-aperture views `L` and `C` from the same scene.
Lenslet convention: 14x14 angular grid, 0-based [row,col] indices. View metadata:
{VIEW_METADATA} Step 1: Analyze Angular Cues Infer target-view geometry from disparity,
occlusion boundaries, and view-dependent visibility between `L` and `C`, using the provided
lenslet indices. Step 2: Generate the Execution Prompt Write a 4-6 sentence instruction
for the downstream model. Your generated prompt MUST include: - Explicit instruction to
extrapolate the target right view from `[L, C]` (not interpolate a middle view). - Specific
guidance on disparity-consistent warping beyond observed angular support, occlusion
completion, and preservation of fine structure. - Strict negative constraints against
ghosting, edge duplication, geometric drift, or unsupported hallucinations. Goal: Produce
an execution prompt that yields a physically consistent extrapolated view under light-field
angular geometry.
```

Written per image by the planner; shown for the row-1 image of the grid on p. 52.

Gemini-3.1-Pro Prompt (Row 1 Image).

```
**Execution Prompt:** Extrapolate the target right view at angular position [6, 13] using
the horizontal parallax observed between the left view at [6, 0] and the center view at [6,
6]. Apply disparity-consistent warping to continue shifting the foreground tree trunks to
the left relative to the midground vehicles and background building. Accurately handle the
resulting disocclusions by smoothly completing the newly exposed regions of the dark SUV and
the silver car behind the moving foreground elements. Maintain the fine textures of the
tree bark, grass, and vehicle reflections throughout the extrapolation process. Strictly
avoid ghosting, edge duplication along the occlusion boundaries, geometric drift in the
static background, and any unsupported hallucinated structures in the newly revealed areas.
```

GPT-5 Prompt (Row 1 Image).

```
Extrapolate the target right view at [6,13] from the two given sub-aperture views L@[6,0]
and C@[6,6]; do not interpolate a middle view. Estimate per-pixel disparity from L→C along
the horizontal row and warp C leftward by a factor of (7/6) times the measured L→C
disparity to reach the [6,13] viewpoint, with larger shifts on near tree trunks/branches and
smaller shifts on background cars/houses. Use layered, edge-aware warping with strict
occlusion handling: moving right reveals disocclusions on the right sides of foreground
edges (tree trunks), which must be completed with background continuation inferred from
textures and lines visible in L/C, without inventing unsupported objects. Preserve fine
structures such as thin branches, bark texture, car rims and window edges, and keep straight
lines and brightness/color consistent with C. Hard constraints: no ghosting, no double
edges, no geometric drift, and no hallucinated content beyond physically justified
disocclusions.
```

Qwen3.5-35B-A3B Prompt (Row 1 Image).

```
Generate the target view at angular coordinate `[6,13]` by extrapolating the motion
trajectory established by the left view `[6,0]` and center view `[6,6]`. Apply a
disparity-consistent warp where foreground elements, particularly the central tree trunk,
shift further leftward relative to the background to simulate the camera's continued
rightward movement. Focus on occlusion completion: as the tree trunk moves left, reveal
and reconstruct the previously hidden front bumper and headlight of the silver sedan and the
side of the dark SUV. Ensure high-fidelity preservation of fine textures on the tree bark
and vehicle surfaces while strictly preventing ghosting artifacts at the occlusion
boundaries. Do not hallucinate new objects; rely solely on the geometric projection of
existing scene structures beyond the observed angular support.
```

Light-Field View Interpolation

sample grids: p. 52

EXPERT Synthesis of the missing center sub-aperture view from the left and right views.

Shared Prompt.

```
Task: Predict the missing center view from two neighboring light-field views. Input: Two
aligned sub-aperture images ordered as left view and right view. Lenslet convention: 14x14
angular grid, 0-based [row,col] indices. View metadata: {VIEW_METADATA} Constraints: -
Interpolation Objective: Reconstruct the middle view inside the known angular pair (for
this dataset: infer `C` from `[L, R]`). - View Geometry Consistency: Preserve epipolar
consistency and disparity geometry when synthesizing the center view. - Occlusion Handling:
Resolve view-dependent visibility changes with physically plausible structure. - Appearance
Fidelity: Preserve texture, color, and local contrast without ghosting, edge doubling, or
shimmer artifacts. - Zero Hallucination: Do not invent scene content not supported by the
two input views. Output: One interpolated center-view image consistent with the provided
left/right inputs and metadata.
```

PLANNER Synthesis of the missing center sub-aperture view from the left and right views.

Planner Prompt.

```
You are an expert computational imaging assistant. Your task is to analyze two light-field
views (left and right) and write a precise "execution prompt" for downstream center-view
interpolation. Input: Two aligned sub-aperture views `L` and `R` from the same scene.
Lenslet convention: 14x14 angular grid, 0-based [row,col] indices. View metadata:
{VIEW_METADATA} Step 1: Analyze Angular Cues Infer center-view geometry from disparity,
occlusion boundaries, and view-dependent visibility between `L` and `R`, using the provided
lenslet indices. Step 2: Generate the Execution Prompt Write a 4-6 sentence instruction
for the downstream model. Your generated prompt MUST include: - Explicit instruction to
interpolate the center view from `[L, R]`. - Specific guidance on disparity-consistent
warping, occlusion completion, and preservation of fine structures. - Strict negative
constraints against ghosting, edge duplication, geometric drift, or unsupported
hallucinations. Goal: Produce an execution prompt that yields a physically consistent
center view under light-field angular geometry.
```

Written per image by the planner; shown for the row-1 image of the grid on p. 53.

Gemini-3.1-Pro Prompt (Row 1 Image).

```
Synthesize the center view at angular coordinates [6,6] by interpolating between the left
image [6,0] and the right image [6,13]. The foreground bush exhibits substantial disparity
compared to the midground vehicles (beige and red cars); utilize robust depth-aware warping
to align these layers accurately. Pay close attention to occlusion boundaries, specifically
where the foliage intersects and obscures the rear section and wheels of the beige car,
ensuring seamless completion of these hidden areas. Maintain the sharp structural lines of
the vehicles and the fine, complex details of the leaves. Strictly avoid ghosting artifacts
around the high-disparity foreground elements, prevent edge duplication along vehicle
outlines, and ensure no geometric drift occurs in the background trees and buildings.
```

GPT-5 Prompt (Row 1 Image).

```
Step 1 -- Angular cues: - Baseline is horizontal: left view [6,0], right view [6,13],
target center [6,6]. Foreground shrub in the lower-right shows the largest disparity; the
yellow hatchback has mid-level disparity; distant cars/trees have small disparity.
Occlusion boundaries occur along shrub edges and the car silhouette (roofline, windows,
wheel arches), with one-sided visibility between L and R. Step 2 -- Execution prompt:
Interpolate the center view at [6,6] from the two aligned inputs L=[6,0] and R=[6,13] using
horizontal disparity-consistent warping. Estimate depth and forward-warp L and R halfway
toward the center; fuse them with left--right consistency to select the view that remains
visible at occlusion edges around the shrub and the yellow car, completing disoccluded
pixels via edge-aware inpainting guided by local structure. Preserve fine details such as
window frames, wheel rims, parking lot lines, car panel contours, and thin foliage without
blurring. Enforce coherent geometry across layers (near shrub > car > far background) and
maintain straight lines and texture continuity. Strictly avoid ghosting, double edges,
rubber-sheet distortions, geometric drift, or any unsupported hallucinated objects or
reflections.
```

Qwen3.5-35B-A3B Prompt (Row 1 Image).

```
Generate the center view at angular coordinate [6,6] by interpolating between the left input
[6,0] and right input [6,13]. Apply disparity-consistent warping to align the scene
geometry, ensuring the beige hatchback and red sedan remain correctly positioned relative to
the background despite the minimal parallax observed. Pay close attention to the occlusion
boundaries of the foreground bushes, ensuring no ghosting artifacts appear where the leaves
overlap the pavement. Strictly avoid introducing geometric drift or hallucinating details
that are not present in the source views, maintaining the original noise texture and
lighting consistency.
```

Random Mask Subsampling (10–80% of Pixels)

sample grids: p. 53

EXPERT Reconstruction from randomly masked measurements keeping 10–80% of pixels.

Shared Prompt.

```
Task: Reconstruct the full-resolution RGB image from randomly masked imaging measurements
containing only partial, undersampled information. Input: A randomly masked measurement
image and its corresponding binary mask image. Constraints: - Mask Semantics: Treat
masked-out regions in the measurement as missing data represented by pure black (#000000),
and use the provided binary mask as the authoritative missing-region map. - Sparsity &
Structural Prior: Assume the underlying image is sparse in a natural image basis. Enforce
global structural consistency first. Recover sharp, high-contrast edges while strictly
keeping smooth regions (like skies or blank walls) clean and uniform. - Artifact
Suppression: Actively suppress incoherent noise and sampling artifacts characteristic of
sub-Nyquist sampling. Do not introduce ringing, overshoot, or artificial frequencies. -
Conservative Complexity: Prefer physically plausible, minimal-complexity solutions.
Preserve the correct geometry, structural proportions, and perspective without artistic
enhancement. - Zero Hallucination: Do not hallucinate fine textures, invent new objects,
or add high-frequency details that are not mathematically supported by the underlying sparse
structural data. Output: A photorealistic, clean, artifact-free reconstructed image
featuring natural contrast and lighting, perfectly faithful to the undersampled
measurements.
```

PLANNER Reconstruction from randomly masked measurements keeping 10–80% of pixels.

Planner Prompt.

```
You are an expert computational imaging assistant.  Your task is to analyze an image formed
by random sub-sampling (characterized by partial information and incoherent sampling
artifacts) and write a highly specific "execution prompt" for a downstream reconstruction
model.  Input:  A randomly masked measurement image and its corresponding binary mask image.
Step 1:  Analyze the Measurements Use the provided binary mask as the authoritative map of
missing samples.  In the measurement image, masked-out regions are represented by pure black
(#000000), but valid scene content can also contain dark pixels; rely on the mask for
disambiguation.  Then examine the undersampled image to deduce the global structural
composition.  Identify the primary sharp boundaries that need recovery, and distinguish them
from the naturally smooth regions (e.g., gradients, skies, or flat surfaces) that are
currently corrupted by incoherent sampling noise.  Step 2:  Generate the Execution Prompt
Write a 4-6 sentence instruction for the image editing/reconstruction model.  Use highly
descriptive and restrictive keywords.  Your generated prompt MUST include:  - Explicit
identification of the image formation process as partial, randomly under-sampled
measurements.  - Specific identification of the deduced geometric structures that need sharp
recovery, contrasted with specific smooth regions that need aggressive artifact suppression
(e.g., "Recover the sharp geometric outlines of the bridge cables while completely smoothing
the incoherent sampling noise in the sky").  - Strict negative constraints forbidding the
model from hallucinating fine details, inventing unsupported textures, or altering the
natural lighting.  Goal:  Your generated prompt should act as a perfect algorithmic
blueprint, guiding the reconstruction model to enforce a natural image sparsity prior,
balancing sharp edge recovery with smooth region denoising.
```

Written per image by the planner; shown for the row-1 image of the grid on p. 55.

Gemini-3.1-Pro Prompt (Row 1 Image).

```
Reconstruct the original image from these partial, randomly under-sampled measurements using
the provided binary mask to identify valid samples.  Accurately recover the sharp geometric
boundaries of the prominent arched stone bridge, the structural outlines of the background
buildings, and the distinct silhouette of the skyline.  Concurrently, apply aggressive
artifact suppression to completely smooth the incoherent sampling noise corrupting the dark,
uniform sky and smooth foreground regions.  Strictly avoid hallucinating fine details,
inventing unsupported textures in the foliage or stonework, or altering the natural
low-light scene illumination.  Enforce a strong natural image sparsity prior to perfectly
balance sharp edge fidelity with smooth region denoising.
```

GPT-5 Prompt (Row 1 Image).

```
Use the binary mask to treat only the masked pixels as missing data and reconstruct the
scene from partial, randomly under-sampled measurements.  Recover the sharp geometric
outlines of the stone arch bridge:  the outer arc of the large central arch, the intrados
and keystone region, the straight bridge deck and parapet with regularly spaced openings,
plus the crisp rooflines and facades of the clustered hillside buildings and the riverbank
edges.  Maintain clean, edge-preserving separation between the bridge, buildings, and
vegetation, keeping tree masses as coarse shapes without inventing leaf-level texture.
Aggressively suppress incoherent sampling noise in the smooth sky and on the water surface
and reflections, rendering them as low-frequency gradients with only gentle ripples where
measurements support them.  Preserve the original lighting and color balance; do not
hallucinate fine brickwork, signage, text, windows, people, or any new structures not
supported by the measured samples.  Enforce a natural-image sparsity prior:  strong edge
recovery on architectural boundaries, heavy denoising on flat regions, and no alteration of
scene geometry.
```

Qwen3.5-35B-A3B Prompt (Row 1 Image).

```
The input image consists of partial, randomly under-sampled measurements of a low-light
scene, characterized by incoherent sampling artifacts and missing data regions defined by
the binary mask.  Analyze the faint diagonal structures in the upper right to deduce and
recover the sharp geometric outlines of suspension bridge cables and support towers.
Simultaneously, aggressively smooth the large, naturally dark regions representing the sky
and water to suppress the high-frequency noise without blurring the structural edges.
Strictly forbid the model from hallucinating fine details, inventing unsupported textures,
or altering the natural lighting and color balance of the night scene.
```

Super-Resolution ($\times 4$)

sample grids: p. 56

EXPERT 4x image super-resolution from a lower-resolution observation.

Shared Prompt.

```
Task:  Perform 4x super-resolution on this 256x256 low-resolution image to produce a sharp,
highly detailed 1024x1024 high-resolution output.  Input:  A downsampled low-resolution
image.  Constraints:  - Spatial Scaling:  Strictly upscale the image dimensions by exactly
4x (from 256x256 to 1024x1024 pixels).  - Fidelity & Detail:  Enhance the resolution by
producing a high-resolution version with sharp edges, fine textures, and realistic detail
that is mathematically consistent with the original low-resolution content.  -
Anti-Hallucination:  Do not invent semantic details.  Do not hallucinate legible text from
blurred pixels, invent specific facial features, or add objects not implied by the
low-resolution geometry.  Only interpolate physically plausible high-frequency data.  -
```

```
Artifact Suppression: Avoid the "watercolor," "oil painting," or "plastic" smoothing
artifacts common in generative upsampling. Retain natural image grain and physically
plausible micro-contrast. Output: A photorealistic 1024x1024 image that looks exactly like
a native high-resolution capture of the original scene.
```

PLANNER 4x image super-resolution from a lower-resolution observation.

Planner Prompt.

```
You are an expert computational imaging assistant. Your task is to analyze a 256x256
low-resolution image and write a highly specific "execution prompt" for a downstream 4x
super-resolution model. Input: A downsampled low-resolution image. Step 1: Analyze the
Degradation Examine the low-resolution image to deduce the underlying semantic content.
Identify specific materials, textures, and high-contrast boundaries that have been lost to
the spatial sub-sampling (e.g., "The blurry region in the center is coarse animal fur; the
jagged diagonal line is the edge of a brick wall"). Step 2: Generate the Execution Prompt
Write a 4-6 sentence instruction for the image upscaling model. Use highly descriptive and
restrictive keywords. Your generated prompt MUST include: - Explicit instruction to
perform a 4x super-resolution upscale from 256x256 to 1024x1024. - Specific identification
of the underlying textures that need physically plausible recovery (e.g., "Recover the fine,
distinct strands of the animal fur and the sharp geometric edge of the brick wall"). -
Strict negative constraints forbidding the model from hallucinating recognizable but
incorrect details (like fake text) or applying unnatural "watercolor" smoothing filters.
Goal: Your generated prompt should act as a perfect semantic guide, helping the upscaling
model intelligently interpolate the missing high-frequency data for those specific textures
without inventing alien features.
```

Written per image by the planner; shown for the row-1 image of the grid on p. 57.

Gemini-3.1-Pro Prompt (Row 1 Image).

```
**Degradation Analysis:** The spatial sub-sampling has blurred the individual blades of
grass into soft green patches. High-frequency details on the distant suspension bridge
(cables and towers) are aliased and faint. The woman's tan leather jacket lacks surface
grain, and the vents on her bicycle helmet are pixelated. Background figures and the
distant foggy hills have lost sharp boundary definitions. **Execution Prompt:** Perform a
4x super-resolution upscale from 256x256 to 1024x1024. Recover the individual grass blades
in the field, eliminating the current blurred patches. Reconstruct the fine vertical cables
of the distant bridge and the crisp geometric vents on the bicycle helmet. Restore the
realistic surface grain of the leather jacket and the sharp boundaries of the distant
figures. Do not hallucinate fake text, alien structural details, or phantom faces.
Strictly avoid applying any unnatural watercolor smoothing filters.
```

GPT-5 Prompt (Row 1 Image).

```
Perform a 4x super-resolution upscale from 256x256 to 1024x1024, preserving the original
scene: a person in the foreground on a broad grass field with distant hills, light fog, a
suspension bridge, and a clear blue sky. Recover physically plausible fine textures:
distinct short blades of grass with subtle shadowing, the grain and stitching seams of the
tan leather jacket, the denim weave on the jeans, crisp vents and metallic/plastic ridges on
the bike helmet, and the faint fabric pattern on the shirt. Reconstruct sharp geometric
boundaries: the straight horizon between grass and distant hills/sky, clean vertical towers
and thin diagonal cables of the bridge, and well-defined edges on the person's clothing and
helmet, while keeping the hills and bridge partially softened by fog. Maintain natural
lighting and gradients; avoid oversharpening halos, banding, or noise, and do not invent
detailed facial features. Strictly avoid hallucinating text, logos, vehicles, extra people,
or new architectural elements, and do not apply watercolor-like smoothing or introduce
artificial high-frequency patterns inconsistent with grass, fabric, leather, metal, or
suspension cables.
```

Qwen3.5-35B-A3B Prompt (Row 1 Image).

```
Perform a 4x super-resolution upscale from 256x256 to 1024x1024, focusing on recovering
high-frequency details lost in the initial blur. Specifically, reconstruct the fine,
distinct blades of grass in the foreground to replace the uniform green wash, ensuring
natural variation in height and color. Resolve the texture of the tan jacket, recovering
realistic fabric folds and stitching lines, and clearly define the ventilation slots on the
bicycle helmet. In the distance, sharpen the faint reddish structure of the Golden Gate
Bridge and the hazy mountain range to reveal their geometric forms. Strictly forbid the
application of unnatural "watercolor" smoothing filters or the hallucination of fake text,
ensuring all recovered features remain physically plausible and consistent with the original
scene.
```

Time-of-Flight Depth

sample grids: p. 57

EXPERT Scene reconstruction from time-of-flight imaging measurements.

Shared Prompt.

```
Task: Reconstruct a per-pixel depth map from four raw time-of-flight (ToF) correlation
measurements of the same scene. Input: Four phase-shifted raw ToF correlation images
captured at 0 deg, 90 deg, 180 deg, and 270 deg with {TOF_MODULATION_FREQUENCY} modulation
frequency. Constraints: - Multi-frame consistency: Use all four correlation frames
jointly. Do not treat any single frame as a standalone depth estimate. - Geometric
```

```
fidelity: Preserve object boundaries, thin structures, and depth discontinuities without
staircase artifacts or edge fattening. - Noise robustness: Suppress phase noise and
invalid speckle-like artifacts while keeping physically plausible depth gradients. - MPI
suppression: Remove multi-path interference at corners while preserving true geometric
discontinuities. - Metric integrity: Output a depth image whose values are spatially and
numerically consistent with the measured phase information. Do not hallucinate structures
that are not supported by the four inputs. - Pad-Region Handling: If black padded borders
are present, treat only the non-black valid region as scene content. Output: A clean,
physically consistent depth map for the scene.
```

PLANNER Scene reconstruction from time-of-flight imaging measurements.

Planner Prompt.

```
You are an expert computational imaging assistant. Your task is to analyze four
phase-stepped raw ToF correlation images of the same scene and write a highly specific
"execution prompt" for a downstream depth-reconstruction model. Input: Four phase-shifted
raw ToF correlation images captured at 0 deg, 90 deg, 180 deg, and 270 deg with
{TOF_MODULATION_FREQUENCY} modulation frequency. Step 1: Analyze the Measurements Inspect
the four correlation frames jointly to identify reliable depth boundaries, low-signal
regions, and areas prone to phase noise or ambiguity (for example glossy surfaces, dark
regions, or distant background). Step 2: Generate the Execution Prompt Write a 4-6
sentence instruction for the image model. Use highly descriptive and restrictive keywords.
Your generated prompt MUST include: - Explicit instruction to reconstruct one
metric-consistent depth map from all four phase-shifted ToF inputs. - Specific guidance to
preserve sharp depth discontinuities at object edges while denoising unstable regions. -
Explicit suppression of multi-path interference at corners while keeping physically valid
corner geometry. - Strict negative constraints forbidding geometric hallucinations, edge
blurring, or inconsistent depth layering. - Specific guidance to ignore black padded border
regions during inference but preserve those regions as black in the output image. -
Explicit handling of black padded borders: ignore padded black areas when inferring scene
content, and preserve those padded areas as black in the output. Goal: Your generated
prompt should act as a precise inversion blueprint that converts the four ToF correlation
measurements into a physically plausible depth map.
```

Written per image by the planner; shown for the row-1 image of the grid on p. 57.

Gemini-3.1-Pro Prompt (Row 1 Image).

```
Analyze the four raw ToF correlation images to produce a highly detailed execution prompt
for a depth-reconstruction model. The input correlation frames show a scene captured with a
modulation frequency of 20 MHz. The scene features a flat rectangular block serving as a
base. Resting on top of this block is a complex object composed of smaller blocks and a
wheel-like shape, possibly a toy construction. The correlation images present varying
intensity patterns due to the phase shifts (0, 90, 180, 270 degrees). Key observations: 1.
**Object Boundaries:** Sharp depth discontinuities exist at the edges of the base block and
the object on top. These boundaries are clear in the correlation images, particularly where
foreground objects contrast with the background. 2. **Surfaces and Textures:** The
surfaces of the base block and the objects on top appear relatively diffuse, but some areas
might be darker or have lower reflectivity, leading to weaker correlation signals and
potential noise in those regions. 3. **Potential Multi-path Interference (MPI):** The
corners where the objects on top meet the base block, and the internal corners of the
objects themselves, are susceptible to multi-path interference due to light bouncing between
surfaces. 4. **Low-Signal Areas:** The background behind the objects is darker, indicating
lower signal return and potential phase noise in these distant or less reflective regions.
5. **Padded Borders:** There is a substantial black padded border surrounding the active
image area in all four frames. Execution Prompt formulation: * **Core Task:** Reconstruct
a single, metric-consistent depth map using all four phase-shifted ToF inputs (0, 90, 180,
270 degrees) captured at 20 MHz. * **Edge Preservation and Denoising:** Emphasize
preserving the sharp depth discontinuities at the edges of the base block and the objects
resting on it. Simultaneously, denoise areas with unstable signals, such as the darker
background regions. * **MPI Suppression:** Explicitly instruct the model to suppress
multi-path interference, particularly at the corners and junctions of the objects and the
base block, while maintaining the physical geometry of these corners. * **Negative
Constraints:** Strictly forbid geometric hallucinations, blurring of the sharp edges
separating the objects from the background, and inconsistent depth layering, ensuring a
physically plausible reconstruction. * **Handling Padded Borders:** Specifically guide the
model to ignore the black padded border regions during inference, ensuring they do not
influence the depth estimation of the scene, and mandate that these padded areas are
preserved as black in the final output image. This analysis leads directly to the specific
execution prompt. Execution Prompt: Synthesize a single metric-consistent depth map from
the four provided ToF correlation images captured with phase shifts of 0, 90, 180, and 270
degrees at a 20 MHz modulation frequency. Carefully preserve the sharp depth
discontinuities at the boundaries of the base block and the objects on top, while applying
robust denoising to stabilize the depth estimates in the darker, low-signal background
areas. Explicitly suppress multi-path interference artifacts at the corners and junctions
where the objects meet the base block, ensuring the physical validity of the corner geometry
is maintained. Strictly prohibit the introduction of geometric hallucinations, blurring of
object edges, or inconsistent depth layering in the reconstructed map. Finally, ignore the
prominent black padded border regions surrounding the scene during inference, and ensure
these padded areas are strictly preserved as black in the output depth map.
```

GPT-5 Prompt (Row 1 Image).

Step 1: Measurement analysis - The four correlation frames show a small block-like object on a pedestal in the bottom-center, surrounded by large black padded borders; signal is strong on the object and base, weaker on the rear wall, and very low elsewhere. - Sharp depth boundaries exist at the object edges and pedestal edges; the corner between the two walls and pedestal edges are prone to multi-path and phase instability. - Dark regions and most of the padded borders have near-zero correlation, so they should be treated as invalid/low-SNR to avoid noisy depth. - Preserve crisp geometry on the object and base while denoising the dim background; watch for phase wrapping in low-signal areas. Execution prompt for the depth-reconstruction model - Reconstruct one metric-consistent depth map from all four phase-shifted ToF inputs (0°, 90°, 180°, 270°) captured at {TOF_MODULATION_FREQUENCY_MHZ} MHz by demodulating the complex correlation: phi = atan2(I90−I270, I0−I180), amplitude A = sqrt((I0−I180)^2 + (I90−I270)^2), and convert phase to range R = (phi/2π)·c/(2f). - Use amplitude-gated, edge-preserving processing: keep sharp depth discontinuities at object and pedestal boundaries, apply denoising only within regions of similar phase, and mark very low-A areas (dark background) as invalid rather than smoothing across edges. - Perform robust 2π phase unwrapping with spatial continuity and amplitude support to prevent wrap-induced layering; maintain crisp, planar geometry for the object faces and the pedestal top. - Suppress multi-path interference at corners and occluding edges by favoring the direct-path solution with highest A and local phase consistency, rejecting secondary reflections while keeping physically valid right-angle corner geometry. - Strict negative constraints: no geometric hallucinations, no edge blurring, and no inconsistent front/back depth layering anywhere in the scene. - Explicitly handle the black padded borders: ignore padded black areas when inferring scene content, and preserve those padded areas as black in the output depth image.

Qwen3.5-35B-A3B Prompt (Row 1 Image).

Reconstruct a single, metric-consistent depth map by fusing the four provided phase-shifted ToF correlation frames. Aggressively denoise the low-signal background regions while maintaining the structural integrity of the central object and pedestal. Preserve sharp depth discontinuities at the object boundaries and pedestal edges, explicitly suppressing multi-path interference at corners while retaining physically valid corner geometry. Strictly forbid geometric hallucinations, edge blurring, or inconsistent depth layering; the output must represent a physically plausible 3D geometry. Ignore the black padded border regions during the inference process to focus on the active scene content, but ensure these black padded areas are preserved as black in the final output image.

White Balance

sample grids: p. 58

EXPERT Illumination correction to recover a naturally white-balanced image.

Shared Prompt.

Task: Correct the white balance of this image. Adjust the color temperature and tint so that neutral surfaces appear neutral and the overall color rendition is accurate and natural. Input: An image with a color cast captured under a non-neutral illuminant. Constraints: - Neutral Reference Targeting: Utilize the color picker tool to identify surfaces in the image that should be pure white or 18% gray when available. If neutral references are partially visible or missing due to cropping, infer a physically plausible global correction from scene-wide illumination cues without local recoloring. - Global Shift Only: Apply a mathematically uniform adjustment to the global color temperature (Kelvin) and tint (green/magenta axis) to neutralize the false color cast. - Exposure & Contrast Integrity: Do not alter the image's overall exposure, luminance levels, or local contrast. The brightness of the highlights and shadows must remain identical to the original image. - Zero Hallucination: Do not selectively "paint" or recolor individual objects. The adjustment must act as a global optical filter inversion, preserving the true underlying saturation of non-neutral objects. Output: A perfectly color-balanced image where the lighting appears natural and all neutral tones are rendered completely free of color casts.

PLANNER Illumination correction to recover a naturally white-balanced image.

Planner Prompt.

You are an expert computational imaging assistant. Your task is to analyze an image formed with an incorrect white balance (characterized by an unnatural global color cast) and write a highly specific "execution prompt" for an image restoration model. Input: An image with a color cast captured under a non-neutral illuminant. Step 1: Analyze the Degradation and Scene Examine the image to identify the nature of the color cast (e.g., is it overly warm/amber from tungsten lighting, overly cool/blue from deep shade, or tinted green from fluorescent lights?). Scan the scene to identify a specific, structurally logical object that should act as a neutral reference point when available (e.g., a white piece of paper, a gray concrete wall, or a white shirt). If such neutral references are partially visible or absent due to cropping, infer the global correction from overall illumination and material statistics. Step 2: Generate the Execution Prompt Write a 4-6 sentence instruction for the image editing model. Use highly descriptive and restrictive keywords. Your generated prompt MUST include: - Explicit identification of the specific global color cast that needs to be neutralized (e.g., "Remove the heavy amber/tungsten color cast"). - Specific identification of the object in the scene that the model should use as its neutral reference for the color picker adjustment when available (e.g., "Use the white coffee cup on the table

as the neutral white reference point"), or explicit instruction to use scene-level global cues when no reliable neutral patch exists. - Strict negative constraints forbidding the model from altering the global exposure, painting localized colors, or changing the structural details of the image. Goal: Your generated prompt should act as a perfect algorithmic blueprint, guiding the restoration model to optically invert the specific lighting temperature without relying on blind "auto-color" algorithms.

Written per image by the planner; shown for the row-1 image of the grid on p. 58.

Gemini-3.1-Pro Prompt (Row 1 Image).

Correct the heavy amber/tungsten color cast pervading this image. The scene is significantly tinted orange due to warm artificial lighting. To neutralize this global tint, use the white column visible in the background on the right as your neutral reference point. Adjust the image's overall color temperature to render this column a true, neutral white, thereby restoring accurate colors to the rest of the scene, including the color checker chart in the foreground. Do not alter the image's overall brightness or contrast. Ensure the structural integrity of the scene remains intact; do not paint localized colors or apply any non-global adjustments. Ensure all colors, particularly on the color checker, accurately reflect the new, balanced temperature.

GPT-5 Prompt (Row 1 Image).

Remove the heavy amber/yellow tungsten color cast affecting the entire frame. Use the X-Rite ColorChecker card in the lower-left as the neutral reference; sample one of its gray patches (preferably the 18% mid-gray) to set the white balance. Apply a global correction by shifting temperature toward cooler/blue and adjust the green--magenta tint only as needed until the sampled gray reads neutral (RGB balanced). Do not alter global exposure, contrast, saturation, or add any local color painting; avoid sharpening, noise reduction, or vignette changes. Preserve all structural details, textures, and the original composition while performing only the white-balance recalibration.

Qwen3.5-35B-A3B Prompt (Row 1 Image).

Remove the pervasive heavy amber and orange color cast caused by warm indoor tungsten lighting. Use the grayscale patches located in the bottom row of the X-Rite ColorChecker chart in the bottom-left foreground as the definitive neutral reference point for the white balance adjustment. Specifically, instruct the model to neutralize the color of the medium gray patches in the chart so they appear achromatic (neutral gray) rather than yellowish-orange. Apply this temperature correction globally to all surfaces, ensuring the white pillars and ceiling lights appear white rather than yellow. Strictly prohibit any adjustments to global exposure, brightness, or contrast, and ensure no structural details or background blur are altered during the restoration process.