SETSUNET: SHARED CONSENSUS GEOMETRY FOR FEW-STEP RECOVERY

Anonymous authors
Paper under double-blind review

ABSTRACT

SETSUNET is a shared learned optimizer for few-step geometric and visual recovery. It learns search directions from local gradient and curvature geometry, projects them to complement a measured base solve, and coordinates their contribution across observation views. This construction allows the same graph network to operate on image variables, transformations and graph states. We establish whitening equivariance for the learned directions and numerical solve, and use it to accelerate the complete solver with Cholesky whitening while preserving physical updates and trained weights. Experiments show better recovery than HSLM, Ceres LM, Dogbox and VeLO on multiframe alignment, stereo and biased-graph estimation, with competitive results on pairwise alignment and image restoration. Learned directions also improve recovery over equal-rank random completion. Together, these results demonstrate that learned search geometry can serve as a reusable component for accurate recovery under small update budgets.

1 INTRODUCTION

Image alignment estimates transformations, stereo estimates disparity, and image restoration estimates intensities. In each case, an iterative solver must decide how to update its current estimate from the available observations. Combining physical models with learned components is a central approach to data-driven inverse problems (Arridge et al., 2019). Learned optimizers learn and reuse update rules across objectives (Andrychowicz et al., 2016; Metz et al., 2022; Moudgil et al., 2026). Extending this idea to visual inverse problems requires a network that can handle different numbers of unknowns and different relationships between them.

SETSUNET learns a shared rule for extending a numerical solver's search space. A graph network predicts mixing coefficients for the current gradient and curvature vectors from the geometry within each variable block and messages from its neighbors. Each input vector has one entry per state variable; the network predicts coefficients for combining these vectors. One 49,096-parameter model can therefore construct directions for image variables, transformations and graph states.

The architecture combines learned direction generation with analytic subspace solves (Figure 1). Learned directions complement a base step already solved in a measured space. Curvature-orthogonal projection preserves the base solution's stationary condition, and two observation models determine the correction jointly. Refinement and a bounded curved step then produce a finite update. The same geometric representation also allows us to change the internal whitening coordinates while preserving the physical update. We use this property to replace spectral whitening with a faster Cholesky implementation, without retraining the network.

The experiments test reuse of the shared checkpoint on multiframe alignment, stereo, restoration and graph estimation. To assess recovery for local deployment, we use five outer steps to limit sequential update rounds and a separate Jacobian budget to limit derivative evaluations.

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.

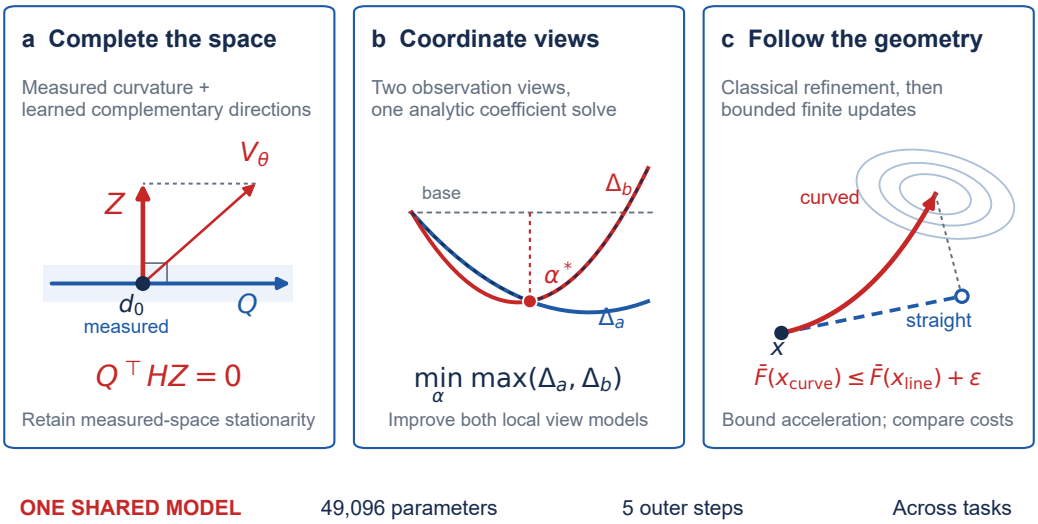

Figure 1. One SETSUNET update. (a) The shared graph network adds search directions to the measured space. Curvature-orthogonal projection makes the added directions complement the base solution. (b) The two observation views jointly determine how to combine them. (c) Refinement and a bounded curved step produce the next estimate. All tasks use the same 49,096-parameter network for five outer steps.

Our contributions are:

1. **A shared rule for learning search spaces.** A graph network constructs blockwise directions that complement a measured base solve; two observation models determine the correction within this learned space.
2. **Whitening equivariance and equivalent acceleration.** The learned directions and numerical solve preserve the physical update under a change of block whitening. This permits a paired Cholesky implementation with the same weights, giving 2.89–3.17× complete-solve acceleration on the tested full-block stereo cases.
3. **Accurate recovery in five steps.** Comparisons with learned optimizers and numerical solvers demonstrate gains in visual and geometric tasks, with further experiments on new scenes and larger states.

2 Related Work

Learned optimizers and priors. Early recurrent optimizers learned update rules from optimization trajectories (Andrychowicz et al., 2016). VeLO, Celo, Celo2 and μLO learn update rules that transfer across neural-network training problems (Metz et al., 2022; Moudgil et al., 2025; 2026; Thérien et al., 2026). Fixed-depth sparse-code predictors (Gregor & LeCun, 2010) and learned primal-dual reconstruction (Adler & Öktem, 2018) combine iterative structure with trainable components. Learned proximal networks encode reconstruction priors, including equivariant versions (Fang et al., 2024; Savir et al., 2026). SETSUNET learns a rule for constructing search directions and uses measured curvature to choose an update within them.

Subspace solvers. HSLM constructs its space from gradient, memory, Krylov and randomized curvature information (Hoang & Lewis, 2026). SETSUNET recomputes its learned space from the current blockwise gradient and curvature carriers, then projects it to complement a measured base solve. Two observation models determine the correction in this space.

Geometric structure and coordination. Scalar constructions yield equivariant vector maps (Villar et al., 2021), and whitening admits an orthogonal freedom (Kessy et al., 2018). Gauge-equivariant learned preconditioners have also been developed for lattice QCD (Lehner & Wettig, 2023). SETSUNET carries equivariance through its learned directions and numerical solve, enabling equiva-

lent whitening implementations. Its projection builds on augmented subspace methods (Gaul et al., 2013), while base-relative coordination relates to multi-objective Newton and conflict-aware updates (Fliege et al., 2009; Liu et al., 2021; Chen et al., 2024).

Finite updates. Damped least-squares and geometric acceleration provide foundations for curved updates (Marquardt, 1963; Transtrum & Sethna, 2012; Liu & Zhang, 2026). Safeguards and generalization bounds have also been studied for learned optimization (Heaton et al., 2023; Sucker et al., 2025), and differentiable optimization layers and nonlinear solvers support learning through optimization (Amos & Kolter, 2017; Pineda et al., 2022). Early stopping can affect recovery (Raskutti et al., 2014); our main recovery comparisons fix the number of outer steps.

3 SETSUNET ARCHITECTURE

3.1 THE BASE UPDATE

Let $x \in \mathbb{R}^n$ be the unknown state, such as image intensities, transformations or graph variables. Each problem supplies an initial estimate, variable blocks, their adjacency, and two observation objectives F_a, F_b. A view is a group of objective factors on this shared state; separate acquisitions and statistical independence are not required.

For known-blur recovery, convolution residuals use a fixed checkerboard split and spatial regularizers use edge-parity groups. SIDD uses even/odd indices of multiscale regional data, noise-model and spatial-regularization factors. Convolution and local differences are evaluated on the full image tile before grouping, preserving their spatial couplings. All factors are retained, so $F_a + F_b = E$ for the original tile objective E. Stereo likewise groups factors formed jointly from the image pair. The partition is fixed by the task adapter; it can affect the update.

We first rescale the variable blocks using their mean local curvature. This puts the search directions in whitened coordinates. Let g_a, g_b denote the gradients and $H_a, H_b \succ 0$ the damped curvature models in these coordinates. We use Gauss–Newton curvature for residual objectives and shifted Hessians for scalar objectives. Their averages are $g = (g_a + g_b)/2$ and $H = (H_a + H_b)/2$, and each view has the quadratic model $q_\nu(d) = g_\nu^\top d + \frac{1}{2} d^\top H_\nu d$, where $\nu \in \{a, b\}$. The solver maps the final direction back to physical coordinates before changing x.

The solver begins with a small basis Q built from the mean and difference gradients, the previous accepted displacement, graph structure and Krylov responses. After removing dependent directions, it minimizes the mean quadratic within this space:

$$\begin{aligned} A &= Q^\top H Q, \qquad \underbrace{d_0 = -Q A^{-1} Q^\top g}_{\text{measured base step}}, \\ e &= g + H d_0, \qquad Q^\top e = 0. \end{aligned} \tag{1}$$

The resulting step d_0 is optimal within Q, but the remaining gradient e can still point outside that space. The network uses e and the cached curvature responses to decide how to extend this measured space at the current state.

3.2 LEARNING ADDITIONAL SEARCH DIRECTIONS

Inputs. The network works with 20 vectors: the remaining gradient e, the view-gradient difference, the mean gradient, the previous displacement, and up to eight curvature responses per view. We call these vectors *carriers*; each has one component per state variable. Within each block, normalized Gram entries describe their relative directions, and block/probe summaries describe their scale. Together they provide 436 features per block. The two views share a response scale for each probe, so their relative magnitudes are preserved.

Graph computation. Two embedding layers map the input features to width 48. Two residual message rounds then exchange information between neighboring variable blocks. A shared output head predicts coefficients for eight mixtures of the carriers. Adding variables changes the number or size of the blocks, while all blocks use the same network weights.

Search directions. Let $U_i \in \mathbb{R}^{n_i \times 20}$ contain the normalized carriers for block i. For each direction j, the network predicts local coefficients ℓ_{ij}, global coefficients whose block average is $\bar{c}_j$, and a residual coefficient ρ_{ij}. Mixing the carriers gives a direction with n_i components, even though the number of predicted coefficients is fixed. We stack the block outputs, normalize each direction, and project it to complement the base solution:

$$\begin{aligned} \widetilde{V}_{ij} &= \underbrace{U_i(\ell_{ij} + \bar{c}_j)}_{\text{local and global carrier mixing}} + \operatorname{softplus}(\rho_{ij}) U_i[:, 1], \\ V &= \operatorname{colnorm}\Big(\operatorname{stack}_i \widetilde{V}_i\Big), \qquad Z = \operatorname{orth}\big(\underbrace{(I - QA^{-1}Q^\top H)}_{\text{curvature-orthogonal projector}} \; V\big). \end{aligned} \tag{2}$$

Here $U_i[:, 1]$ is the residual carrier. The head predicts 328 values per block, and differing block coefficients let an output leave the span of the original global carriers. The solver requests eight measured and eight learned directions; orthogonalization removes dependent columns.

Projection gives $Q^\top H Z = 0$, so every correction $d = d_0 - Z\alpha$ preserves $Q^\top(g + Hd) = 0$.

3.3 Combining the two observation views

The network determines the additional space Z; the two observation models determine the update within it. Since the views may favor different combinations, we choose coefficients that minimize the larger quadratic change relative to the base step:

$$\begin{aligned} C_\nu &= Z^\top H_\nu Z, \qquad b_\nu = Z^\top(g_\nu + H_\nu d_0), \\ \Delta_\nu(\alpha) &= q_\nu(d_0 - Z\alpha) - q_\nu(d_0) = \tfrac{1}{2}\alpha^\top C_\nu \alpha - b_\nu^\top \alpha, \\ \alpha^* &\in \arg\min_\alpha \underbrace{\max_{\nu \in \{a,b\}} \Delta_\nu(\alpha)}_{\text{worst change from the base step}}, \qquad d = d_0 - Z\alpha^*. \end{aligned} \tag{3}$$

For two views, the maximum is the mean change plus half their absolute difference. The rule therefore balances progress with disagreement. Appendix A derives its scalar dual.

The feasible choice $\alpha = 0$ leaves the base step unchanged, so exact minimax completion satisfies $q_\nu(d) \le q_\nu(d_0)$ for both views. The task configuration then applies two mean-objective conjugate-gradient corrections (CG2) or none (CG0). Section 4.5 tests their effect on recovery.

3.4 Taking a finite step

A direction derived from a local model may need to bend as the residual changes. Let v be the refined direction in physical coordinates. Following geodesic acceleration (Transtrum & Sethna, 2012; Liu & Zhang, 2026), we estimate the residual's second derivative along v by finite differences and solve $(J^\top J + \lambda D)a = -J^\top r_{vv}$. Here r and J are the residual and Jacobian, and D is the floored diagonal of $J^\top J$. We bound the acceleration a and compare a straight step with a curved one:

$$\begin{aligned} \widehat{a} &= \min\{1, 2\|v\|/\|a\|\}\, a, \\ x_{\text{line}}(t) &= x + tv, \qquad x_{\text{curve}}(t) = x + tv + \underbrace{\tfrac{1}{2}t^2\widehat{a}}_{\text{bounded curvature correction}}, \\ x_{\text{trial}}(t) &= \begin{cases} x_{\text{curve}}(t), & \bar{F}(x_{\text{curve}}(t)) \le \bar{F}(x_{\text{line}}(t)) + \epsilon(t), \\ x_{\text{line}}(t), & \text{otherwise}, \end{cases} \end{aligned} \tag{4}$$

where $\bar{F} = (F_a + F_b)/2$ and $\epsilon(t)$ is a numerical tolerance. The curvature correction has norm at most $t\|v\|$ for $0 \le t \le 1$. Armijo acceptance and the actual-to-predicted reduction ratio determine the accepted step and subsequent damping. The accepted displacement supplies the next history direction. Appendix B gives the constants.

3.5 Equivalent whitening, unchanged physical updates

The network's geometric inputs also make its updates independent of the whitening basis. Let $N \succ 0$ be the fixed physical block metric and let W, W' be two blockwise whitening maps: $W^\top N W = W'^\top N W' = I$. Then $W' = WO$ for a block-diagonal orthogonal O.

Proposition 1 (whitening equivariance). With consistent gradient, curvature and history transformations, SETSUNET's carrier features and network coefficients are unchanged. Its directions and reduced solve transform so that $d' = O^\top d$ and $W'd' = Wd$. In exact arithmetic, with the same rank decisions and physical finite-update rule, the solver takes the same physical step.

The key identity is $U_i'^\top U_i' = U_i^\top U_i$: block rotations preserve the features, while carrier mixing rotates the output. The measured basis, completion and CG transform consistently (Appendix C). Consequently, symmetric whitening $N^{-1/2}$ can be replaced by Cholesky whitening $L^{-\top}$, where $N = LL^\top$, without changing the trained weights. Section 4.7 measures the resulting acceleration.

3.6 Training and transfer

Local subspace pretraining combines synthetic geometric problems with radiometric calibration. Trajectory training then uses 60 small synthetic SE3 and bundle-adjustment problems and 12 separate validation problems. We select one checkpoint by the mean five-step validation objective over two fixed training-order seeds and four checkpoints per run, then reuse it across all applications. Appendix B gives the data sources and selection details.

4 Experiments

4.1 Experimental setup

We compare SETSUNET with HSLM (Hoang & Lewis, 2026), Ceres LM (Agarwal et al., 2023), SciPy Dogbox (SciPy contributors, 2026) and official-weight VeLO (Metz et al., 2022). HSLM tests subspace construction, Ceres LM and Dogbox provide established numerical-library implementations, and VeLO supplies a learned-update comparison. Every retained case is evaluated with all five methods with the same observations and initialization. Appendix D describes their implementations.

Two complementary budgets for local deployment use five outer steps and, separately, at most 21 joint Jacobian evaluations, with matched observations and initialization and one shared checkpoint. The 18-task heterogeneous benchmark covers multiframe alignment, image recovery, stereo and graph estimation. Additional studies evaluate new synthetic instances and public images, transfer across graph sizes, larger multiframe scenes and a 1,518-case benchmark; Appendices E and F give their coverage.

We report each task's physical error or PSNR. Five-method comparisons use a 2% relative-error margin for practical ties, equivalent to 0.172 dB in PSNR via RMSE. Larger advantages are wins; Table 1 bolds every method tied with the best mean. This descriptive tolerance is not a significance test. Common preparation failures remain in the coverage totals.

4.2 Recovery across 18 heterogeneous tasks

SETSUNET wins or ties against each comparator on all 18 tasks under the declared margin (Figure 2). The entries span seven task groups and different state dimensions, using one shared network. Appendix E reports every endpoint in its task units.

4.3 Recovery on additional scenes and larger benchmarks

The complementary comparison combines 32 problems—twelve synthetic cases and twenty public-image problems—with 32 additional biased graphs spanning 48–256 variables. The two cohorts retain their original inputs and selection histories. SETSUNET wins or ties on 61, 60, 60, 61 of these 64 cases against HSLM, Ceres LM, Dogbox and VeLO, respectively. All 32 additional biased graphs favor SETSUNET beyond the 2% margin. Separate 20-case, five-target multiframe and eleven-case transfer studies also use the same five methods (Appendix F).

The larger study uses 580 HPatches pairs and 116 multiframe sequences (Balntas et al., 2017), four synthetic groups of 64 instances, 60 TUM windows (Sturm et al., 2012), 66 GoPro known-blur regions (Nah et al., 2017), 100 LOL-v2 regions (Yang et al., 2021), 160 SIDD regions (Abdelhamed et al., 2018), and 180 Middlebury stereo crops (Scharstein et al., 2014). Shared preparation succeeds

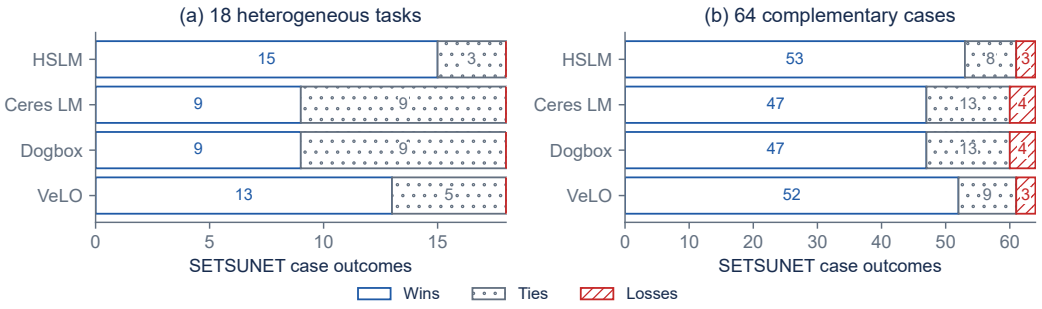

Figure 2. Five-step recovery with one shared SETSUNET model. Bars count wins, practical ties and losses under the 2% error margin: (a) 18 heterogeneous tasks and (b) 64 complementary cases. All five methods cover the same cases in each panel.

on 1,503 of the 1,518 cases. Table 1 reports all five methods on every group; Appendix F gives paired intervals.

Table 1. Five-step recovery on common prepared cases. Bold marks the best mean and practical ties within 2% in error (equivalent RMSE for PSNR). Arrows indicate the preferred direction.

Group	Scored/reg.	SETSUNET	HSLM	Ceres LM	Dogbox	VeLO
HPatches pairs ↓	579/580	**46.2240**	56.5578	**46.1859**	**46.1879**	70.8503
Multiframe ↓	115/116	**15.2157**	17.9581	18.7085	16.4495	27.5571
Balanced graph ↓	64/64	0.1090	0.5561	**0.0101**	**0.0101**	0.7295
Biased graph ↓	64/64	**0.2387**	0.5813	0.3522	0.3522	0.7338
Broyden ↓	64/64	5.37e-06	0.1822	**7.02e-16**	**4.93e-16**	0.5223
SE3 ↓	64/64	0.0049	0.0089	**0.0034**	**0.0034**	0.0093
TUM ↓	47/60	0.0592	0.0662	**0.0368**	**0.0366**	0.0678
Known blur ↑	66/66	**41.2231**	39.4490	**41.2238**	**41.2238**	36.9533
Low light ↑	100/100	**10.3752**	**10.2626**	**10.2577**	**10.3087**	**10.3169**
SIDD ↑	160/160	**23.9106**	**23.8230**	**23.8927**	**23.9030**	**23.9602**
Stereo ↓	180/180	**2.2550**	2.5275	2.6018	2.5328	2.6018

SETSUNET has the lowest mean multiframe error, 15.2157 pixels, followed by Dogbox at 16.4495. Its mean stereo error is 2.2550 pixels, followed by HSLM at 2.5275. It also has the best biased-graph mean. Under the 2% margin, SETSUNET is best or tied for best in seven of the eleven groups, including HPatches, known blur, low light and SIDD. Ceres LM and Dogbox obtain lower errors on balanced graphs, Broyden and pose estimation. These results identify task regimes in which the shared search-space architecture provides effective few-step recovery.

Complementing the five-step results, we test ten small cases in each of six task types under a separate cap of 21 joint Jacobian evaluations. Internal derivative recomputations count, and recovery is measured at the latest completed endpoint within the cap. VeLO uses a separately identified allowance of 21 joint gradients. SETSUNET has the best mean low-light and stereo recovery in this study; other task rankings differ from the five-step comparison (Appendix H).

4.4 Learned versus random search directions

We compare learned and random completion while retaining the same measured solve, projection, minimax coordination and CG2 refinement. Sixteen new 96-variable graph scenes each supply one balanced condition and three strengths of bias or noise, giving 112 conditions. Scene geometry, initialization and Gaussian draws stay fixed across strengths. The random reference averages recovery errors over three fixed random seeds with the same direction budget. Descriptive 95% intervals resample scene seeds; Appendix G gives the construction.

Relative to this random reference, learned completion reduces RMSE by 48.73% [47.01, 50.52] under bias and 76.30% [72.99, 79.45] under noise (Figure 3 and Table 2). At most 16 search directions

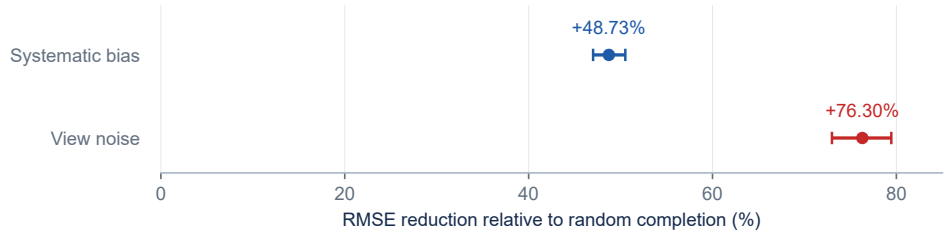

Figure 3. Learned versus random completion under the same minimax coordination and CG2 refinement. Points show relative RMSE reduction against the mean result of three equal-rank random completions. Error bars are descriptive 95% intervals over 16 scene seeds. Each effect averages the three nonzero strengths within a seed. Positive values favor learned completion.

are retained for these 96-variable problems. This comparison evaluates the complete learned direction generator against random subspaces; it does not isolate training from the network architecture.

Table 2. Direction and coordination comparisons with CG2. Entries are relative RMSE reductions with descriptive 95% seed-cluster intervals; each column contains 48 conditions from 16 seeds. The first row averages recovery errors over three equal-rank random completions before comparison with learned directions. The second compares minimax with mean coordination using the same learned directions.

Reference	Systematic bias	View noise
Random completion (3 seeds)	+48.73% [+47.01, +50.52]	+76.30% [+72.99, +79.45]
Mean coordination	+9.45% [+6.57, +12.37]	-10.43% [-15.71, -5.18]

4.5 Consensus and Refinement under Observation Bias

The two-view solve controls how disagreement between observations affects the update. Its action is especially simple when the reduced curvature is the same in both views, $C_a = C_b = C \succ 0$. Writing $m = (b_a + b_b)/2$ and $\delta = (b_a - b_b)/2$, Eq. (3) gives

$$\alpha^* = \underbrace{C^{-1}m}_{\text{mean coefficients}} - \underbrace{\eta C^{-1}\delta}_{\text{disagreement correction}}, \qquad \eta = \mathrm{clip}_{[-1,1]}\left(\frac{\delta^\top C^{-1} m}{\delta^\top C^{-1}\delta}\right), \tag{5}$$

with $\eta = 0$ when $\delta = 0$. The first term follows the mean gradient; the second corrects it along the view difference. Appendix A derives this expression. We test its effect over complete five-step runs.

Under systematic bias, minimax coordination with CG2 reduces error by 9.45% [6.57, 12.37] relative to mean coordination with the same network, winning 41 of 48 conditions. At bias strengths $2/3$ and 1, the reductions are 14.36% [10.98, 17.87] and 12.08% [10.10, 14.26]. Removing the two CG corrections changes these to -3.23% [-5.24, -1.26] and -0.72% [-2.62, 1.14] (Figure 4). The benefit at these bias strengths depends on retaining CG refinement.

Under asymmetric noise, mean coordination gives lower error. The bias and noise sweeps therefore favor different coordination rules (Figure 4).

4.6 Curved Updates and Larger States

We compare straight and bounded curved updates with the tangent direction fixed. The mean-cost rule requires the curved trial to have no higher average view cost than the straight trial. The per-view rule imposes this requirement on each view separately; both allow numerical tolerance. On six nonlinear instances, the mean-cost rule improves two results and ties four relative to straight updates. On the 18-task benchmark, it improves two multiframe cases over the per-view rule.

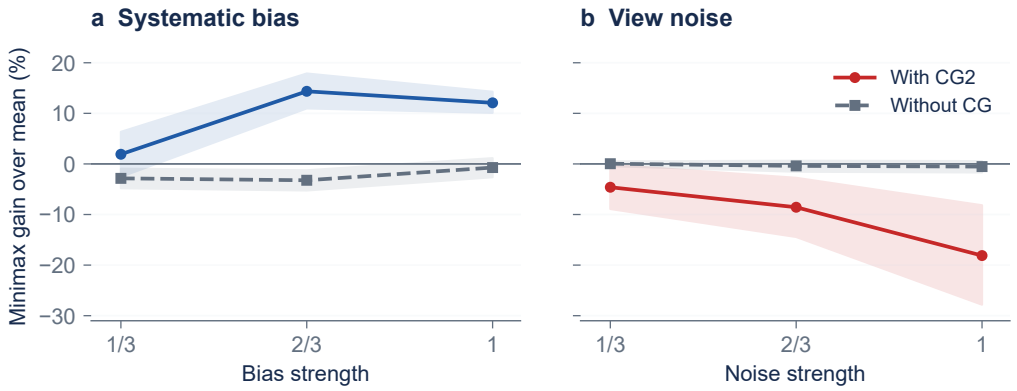

Figure 4. Consensus with and without refinement. Each point compares minimax with mean coordination using the same learned proposal and CG setting. Lines show the mean effect over 16 paired scenes, with descriptive 95% seed-cluster intervals. CG2 enables the positive bias effect, while asymmetric noise favors mean coordination.

With the same checkpoint and $8 + 8$ direction budget, learned completion outperforms equal-rank random completion on all six tested instances at 48, 192 and 768 variables. Mean relative error reductions are 16.40%, 33.09% and 24.85%, respectively. Appendix G gives the per-size results and transformation controls.

4.7 Equivalent whitening implementations

Paired Cholesky whitening gives 2.89–$3.17\times$ complete five-step acceleration on three full-block stereo cases, preserving recorded step-control decisions with relative endpoint differences below 1.35×10^{-11} (Appendix C). The exact minimax correction vanishes in this setting: the experiment isolates execution equivalence and whitening cost, not acceleration of an active learned correction. A further check on 30 crops from 15 scenes preserves all 150 paired-step decisions, with relative endpoint differences below 2.19×10^{-10}. Other block layouts and eigenvalue-clipped metrics retain spectral whitening.

5 Discussion

Fixed-size coefficient prediction lets one checkpoint operate across image, transformation and graph states. Multiframe and stereo gains persist on larger collections, and biased-graph gains extend across state dimensions.

The cross-method evaluation uses only two budgets: five outer steps bound sequential updates, and 21 joint Jacobians bound derivative evaluations, including recomputation. Current implementations lack comparable low-level optimization and backend adaptation, so their elapsed times are not representative measures of algorithmic efficiency.

Multi-block Cholesky whitening and fused operations remain implementation targets. FPGA/ASIC latency, area and energy remain to be measured.

6 Conclusion

SETSUNET provides a shared search-space architecture for few-step inverse problems. A graph network constructs directions from gradient and curvature geometry, while projection and two-view coordination turn them into corrections to a numerical base solve. The same learned rule transfers from synthetic geometric training to visual and graph recovery across different state dimensions. Whitening equivariance further connects this architecture to equivalent, faster numerical implementations without retraining. Together, these contributions establish a practical way to combine learned search geometry with analytic optimization for accurate recovery under limited update budgets.

AI USE STATEMENT

Generative AI tools assisted with English-language polishing and literature retrieval. The authors take responsibility for the final content.

DATA AND CODE AVAILABILITY

The supplementary ZIP provides the minimal SETSUNET inference implementation, shared checkpoint, a bounded installation check, and compact results for Tables 1–9 and Figure 2 with their paper mappings. Full processed experimental data and detailed records are available from the authors upon request. Public datasets are available from their original providers, cited in the text.

REFERENCES

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*, pp. 1692–1700, 2018. URL https://openaccess.thecvf.com/content_cvpr_2018/html/Abdelhamed_A_High-Quality_Denoising_CVPR_2018_paper.html.

Jonas Adler and Ozan Öktem. Learned primal-dual reconstruction. *IEEE Transactions on Medical Imaging*, 37(6):1322–1332, 2018. doi: 10.1109/TMI.2018.2799231. URL https://doi.org/10.1109/TMI.2018.2799231.

Sameer Agarwal, Keir Mierle, and The Ceres Solver Team. Ceres Solver, 2023. URL https://github.com/ceres-solver/ceres-solver. Version 2.2.

Brandon Amos and J. Zico Kolter. OptNet: Differentiable optimization as a layer in neural networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 136–145, 2017. URL https://proceedings.mlr.press/v70/amos17a.html.

Marcin Andrychowicz, Misha Denil, Sergio Gómez, Matthew W. Hoffman, David Pfau, Tom Schaul, Brendan Shillingford, and Nando de Freitas. Learning to learn by gradient descent by gradient descent. In *Advances in Neural Information Processing Systems*, volume 29, 2016. URL https://proceedings.neurips.cc/paper/2016/hash/fb87582825f9d28a8d42c5e5e5e8b23d-Abstract.html.

Simon Arridge, Peter Maass, Ozan Öktem, and Carola-Bibiane Schönlieb. Solving inverse problems using data-driven models. *Acta Numerica*, 28:1–174, 2019. doi: 10.1017/S0962492919000059. URL https://doi.org/10.1017/S0962492919000059.

Vassileios Balntas, Karel Lenc, Andrea Vedaldi, and Krystian Mikolajczyk. HPatches: A benchmark and evaluation of handcrafted and learned local descriptors. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5173–5182, 2017. URL https://openaccess.thecvf.com/content_cvpr_2017/html/Balntas_HPatches_A_Benchmark_CVPR_2017_paper.html.

Lisha Chen, Heshan Fernando, Yiming Ying, and Tianyi Chen. Three-way trade-off in multi-objective learning: Optimization, generalization and conflict-avoidance. *Journal of Machine Learning Research*, 25(193):1–53, 2024. URL https://www.jmlr.org/papers/v25/23-1287.html.

Zhenghan Fang, Sam Buchanan, and Jeremias Sulam. What's in a prior? learned proximal networks for inverse problems. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/f4757db82a02eea015670ecca605d5cc-Abstract-Conference.html.

Jörg Fliege, L. M. Graña Drummond, and Benar F. Svaiter. Newton's method for multiobjective optimization. *SIAM Journal on Optimization*, 20(2):602–626, 2009. doi: 10.1137/08071692X. URL https://doi.org/10.1137/08071692X.

André Gaul, Martin H. Gutknecht, Jörg Liesen, and Reinhard Nabben. A framework for deflated and augmented Krylov subspace methods. *SIAM Journal on Matrix Analysis and Applications*, 34(2): 495–518, 2013. doi: 10.1137/110820713. URL https://arxiv.org/abs/1206.1506.

Karol Gregor and Yann LeCun. Learning fast approximations of sparse coding. In *Proceedings of the 27th International Conference on Machine Learning*, 2010. URL https://icml.cc/2010/papers/449.pdf.

Howard Heaton, Xiaohan Chen, Zhangyang Wang, and Wotao Yin. Safeguarded learned convex optimization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 7848–7855, 2023. doi: 10.1609/aaai.v37i6.25950. URL https://ojs.aaai.org/index.php/AAAI/article/view/25950.

M. Duc Hoang and Timothy J. Lewis. Adaptive hybrid subspace Levenberg Marquardt algorithm with adequacy monitor for large scale least squares problems, 2026. URL https://arxiv.org/abs/2608.25524. Preprint, v1.

Agnan Kessy, Alex Lewin, and Korbinian Strimmer. Optimal whitening and decorrelation. *The American Statistician*, 72(4):309–314, 2018. doi: 10.1080/00031305.2016.1277159. URL https://arxiv.org/abs/1512.00809.

Christoph Lehner and Tilo Wettig. Gauge-equivariant neural networks as preconditioners in lattice QCD. *Physical Review D*, 108:034503, 2023. doi: 10.1103/PhysRevD.108.034503. URL https://arxiv.org/abs/2302.05419.

Bo Liu, Xingchao Liu, Xiaojie Jin, Peter Stone, and Qiang Liu. Conflict-averse gradient descent for multi-task learning. In *Advances in Neural Information Processing Systems*, volume 34, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/9d27fdf2477ffbff837d73ef7ae23db9-Abstract.html.

Jianing Liu and Dong H. Zhang. Higher-order geometric updates for Levenberg–Marquardt method via Riemann normal coordinates, 2026. URL https://arxiv.org/abs/2607.07623. Preprint, v1.

Donald W. Marquardt. An algorithm for least-squares estimation of nonlinear parameters. *Journal of the Society for Industrial and Applied Mathematics*, 11(2):431–441, 1963. doi: 10.1137/0111030. URL https://doi.org/10.1137/0111030.

Luke Metz, James Harrison, C. Daniel Freeman, Amil Merchant, Lucas Beyer, James Bradbury, Naman Agrawal, Ben Poole, Igor Mordatch, Adam Roberts, and Jascha Sohl-Dickstein. VeLO: Training versatile learned optimizers by scaling up, 2022. URL https://arxiv.org/abs/2211.09760.

Abhinav Moudgil, Boris Knyazev, Guillaume Lajoie, and Eugene Belilovsky. Celo: Training versatile learned optimizers on a compute diet. *Transactions on Machine Learning Research*, 2025. URL https://openreview.net/forum?id=SLqJbt4emY.

Abhinav Moudgil, Boris Knyazev, and Eugene Belilovsky. Celo2: Towards learned optimization free lunch. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/12271b64c483ad8f6192eb6aaa102044-Abstract-Conference.html.

Seungjun Nah, Tae Hyun Kim, and Kyoung Mu Lee. Deep multi-scale convolutional neural network for dynamic scene deblurring. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3883–3891, 2017. URL https://openaccess.thecvf.com/content_cvpr_2017/html/Nah_Deep_Multi-Scale_Convolutional_CVPR_2017_paper.html.

Luis Pineda, Taosha Fan, Maurizio Monge, Shobha Venkataraman, Paloma Sodhi, Ricky T. Q. Chen, Joseph Ortiz, Daniel DeTone, Austin Wang, Stuart Anderson, Jing Dong, Brandon Amos, and Mustafa Mukadam. Theseus: A library for differentiable nonlinear optimization. In *Advances in Neural Information Processing Systems*, volume 35, 2022. URL https://arxiv.org/abs/2207.09442.

Garvesh Raskutti, Martin J. Wainwright, and Bin Yu. Early stopping and non-parametric regression: An optimal data-dependent stopping rule. *Journal of Machine Learning Research*, 15(11):335–366, 2014. URL https://www.jmlr.org/papers/v15/raskutti14a.html.

Oriel Savir, Zhenghan Fang, and Jeremias Sulam. Learning affine-equivariant proximal operators. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2026. URL https://arxiv.org/abs/2604.15556.

Daniel Scharstein, Heiko Hirschmüller, York Kitajima, Greg Krathwohl, Nera Nešić, Xi Wang, and Porter Westling. High-resolution stereo datasets with subpixel-accurate ground truth. In *German Conference on Pattern Recognition*, 2014. URL https://www.cs.middlebury.edu/~schar/papers/datasets-gcpr2014.pdf.

SciPy contributors. Scipy least_squares: nonlinear least-squares optimization, 2026. URL https://docs.scipy.org/doc/scipy/reference/generated/scipy.optimize.least_squares.html.

Jürgen Sturm, Nikolas Engelhard, Felix Endres, Wolfram Burgard, and Daniel Cremers. A benchmark for the evaluation of RGB-D SLAM systems. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 573–580, 2012. doi: 10.1109/IROS.2012.6385773. URL https://doi.org/10.1109/IROS.2012.6385773.

Michael Sucker, Jalal Fadili, and Peter Ochs. Learning-to-optimize with PAC-Bayesian guarantees: Theoretical considerations and practical implementation. *Journal of Machine Learning Research*, 26(211):1–53, 2025. URL https://www.jmlr.org/papers/v26/24-0486.html.

Benjamin Thérien, Charles-Étienne Joseph, Boris Knyazev, Edouard Oyallon, Irina Rish, and Eugene Belilovsky. μLO: Compute-efficient meta-generalization of learned optimizers. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/66fdfe692a360dfbad1fba32f64694da-Abstract-Conference.html.

Mark K. Transtrum and James P. Sethna. Improvements to the Levenberg–Marquardt algorithm for nonlinear least-squares minimization, 2012. URL https://arxiv.org/abs/1201.5885.

Soledad Villar, David W. Hogg, Kate Storey-Fisher, Weichi Yao, and Ben Blum-Smith. Scalars are universal: Equivariant machine learning, structured like classical physics. In *Advances in Neural Information Processing Systems*, volume 34, pp. 28848–28863, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/f1b0775946bc0329b35b823b86eeb5f5-Abstract.html.

Wenhan Yang, Wenjing Wang, Haofeng Huang, Shiqi Wang, and Jiaying Liu. Sparse gradient regularized deep retinex network for robust low-light image enhancement. *IEEE Transactions on Image Processing*, 30:2072–2086, 2021. doi: 10.1109/TIP.2021.3050850. URL https://scholars.cityu.edu.hk/en/publications/sparse-gradient-regularized-deep-retinex-network-for-robust-low-l/.

SUPPLEMENTARY MATERIAL

A COMPLETION: ALGORITHM AND DERIVATION

One outer step. Both views are linearized at the current state. Their mean block curvature defines the whitening metric. The solver constructs a measured basis Q, solves its mean quadratic, and asks the network for additional directions. The projected directions define a small two-view problem; its solution is refined and mapped back to physical coordinates before taking a finite step:

1. Form $g = (g_a + g_b)/2$ and $H = (H_a + H_b)/2$. Build Q from gradient, history, graph-weighted and Krylov directions, caching $Y_\nu = H_\nu Q$.
2. Compute $A = Q^\top HQ$, $c = A^{-1}Q^\top g$, $d_0 = -Qc$ and $e = g - HQc$.
3. Generate V with the shared network and set $Z = \mathrm{orth}((I - QA^{-1}Q^\top H)V)$, removing dependent columns.
4. Evaluate H_aZ, H_bZ and solve Eq. (3). If its numerical checks fail, retain d_0.
5. Apply the configured mean-CG refinement. Map the resulting direction to physical coordinates and form the bounded acceleration in Eq. (4).
6. Compare straight and curved trials at each backtracking time, then accept using the mean observation objective. Store the accepted displacement as history.

Preserving the base solve. Since $A = Q^\top HQ$,

$$Q^\top H(I - QA^{-1}Q^\top H) = 0. \tag{6}$$

Orthogonalizing the projected columns preserves this nullspace. Hence $Q^\top HZ = 0$, and $Q^\top(g + H(d_0 - Z\alpha)) = 0$ for every α. The implementation additionally removes roundoff in the span of HQ. Each learned proposal is thus used to extend a stationary base solution.

Scalar dual. Let $C_w = wC_a + (1-w)C_b$ and $b_w = wb_a + (1-w)b_b$. Introducing an epigraph for the two quadratic changes gives nonnegative multipliers summing to one. Eliminating α yields

$$\max_{0\le w\le 1} \phi(w), \qquad \phi(w) = -\tfrac{1}{2} b_w^\top C_w^{-1} b_w, \qquad \alpha_w = C_w^{-1} b_w. \tag{7}$$

The positive-definite reduced matrices make the primal convex, and strict epigraph feasibility gives strong duality. With $D = C_a - C_b$ and $b_\Delta = b_a - b_b$,

$$\phi'(w) = -b_\Delta^\top \alpha_w + \tfrac{1}{2}\alpha_w^\top D\alpha_w. \tag{8}$$

The solver checks the endpoints and bisects an interior root. Choosing $\alpha = 0$ gives zero change in both views, so the exact solution satisfies $\Delta_a(\alpha^*), \Delta_b(\alpha^*) \le 0$. This property applies to completion before the subsequent mean-CG refinement.

Common curvature. If $C_a = C_b = C$, write $m = (b_a + b_b)/2$ and $\delta = (b_a - b_b)/2$. The objective becomes

$$\tfrac{1}{2}\alpha^\top C\alpha - m^\top \alpha + |\delta^\top \alpha|. \tag{9}$$

Its optimality condition is $C\alpha - m + \eta\delta = 0$, with $\eta \in [-1, 1]$ in the subgradient of the absolute value. Substitution gives Eq. (5). At an interior dual optimum, training differentiates the stationarity equation implicitly; endpoint solutions follow their active branch.

B NETWORK, TRAINING AND SOLVER CONFIGURATION

Features. The 20 carriers contain e, $g_a - g_b$, $(g_a + g_b)/2$, the previous accepted displacement, and eight cached curvature responses from each view. Missing responses and initial history are zero. Each carrier is normalized by its full-state Euclidean norm. For every block, the network receives 400 entries of the carrier Gram matrix, four block descriptors and 32 probe summaries.

The block descriptors are its fraction of the state dimension, residual-carrier energy, normalized graph degree and an indicator for nonzero degree. For probe q_j, define

$$s_j = \sqrt{\tfrac{1}{2}(\|Y_a[:,j]\|^2 + \|Y_b[:,j]\|^2)}, \qquad y_{\nu j} = Y_\nu[:,j]/s_j. \tag{10}$$

The four summaries per probe are the blockwise sums of $q_j \odot (y_{aj} + y_{bj})/2$, y_{aj}^2, y_{bj}^2 and $y_{aj} \odot y_{bj}$. Both views use the same s_j, preserving their relative response magnitude. Norm denominators have a numerical floor.

Network. Two width-48 linear–LayerNorm–SiLU layers embed the 436 features. Two residual message rounds concatenate each block's hidden state with the mean of its graph neighbors, then apply a width-48 linear–LayerNorm–SiLU map. The head emits $8(1 + 2 \cdot 20) = 328$ values per block: residual coefficients, local carrier coefficients and global coefficients averaged over blocks. All blocks use the same 49,096 parameters. Eq. (2) gives the output construction.

Training. Local subspace pretraining uses synthetic matrix, sensor, SE3 and bundle-adjustment problems, plus radiometric calibration on three HPatches sequences: `i_autannes`, `i_bologna` and `i_books`. It evaluates initialization seeds 17, 29 and 43 using separate validation problems. The selected local checkpoint (seed 17, update 120) initializes both trajectory runs. Trajectory training uses 30 synthetic SE3 and 30 bundle-adjustment instances, with six separate validation instances from each family. Its objective is

$$\mathcal{L}_T = \frac{\bar{F}(x_T) - \bar{F}(x_{\mathrm{LM},5})}{\max\{\bar{F}(x_0) - \bar{F}(x_{\mathrm{LM},5}),\, 10^{-3}\bar{F}(x_0),\, 10^{-30}\}}. \tag{11}$$

Training uses AdamW with learning rate 10^{-4}, weight decay 10^{-5} and gradient clipping at norm 2. The trajectory protocol fixes two training-order seeds, 17 and 29, each starting from the same pretrained weights. Each run receives 60 updates: the first 40 unroll two steps and the last 20 unroll five. Both runs are evaluated at updates 0, 20, 40 and 60 on the same 12 validation problems, always using five-step inference. Selection minimizes the mean normalized observation loss across these eight evaluations. Seed 29 at update 40 attains the minimum, 0.003151903, versus 0.003220161 at the common initialization. This checkpoint is still in the two-step training phase; all reported inference uses five steps. The seed pair defines the trajectory-stage selection budget, rather than a test-time average over models. Every paper result uses this one checkpoint. Recovery truth does not enter its training or selection; the supplementary package records all eight validation scores.

Numerics and refinement. Subspace arithmetic uses float64. The measured and completion budgets are eight directions each, with rank tolerance 10^{-9} and completion residual tolerance 10^{-7}. A twice-reorthogonalized basis removes dependent directions. Measurement graphs, Broyden and TUM use two mean-CG corrections; the remaining main configurations use none. Curvature is Gauss–Newton for residual objectives and a positive-definite shifted Hessian for scalar objectives. The block metric is formed from mean curvature, with eigenvalue floor $32\epsilon_{\mathrm{mach}} \max_i |N_{ii}|$.

Finite step. The acceleration uses

$$r_{vv} \approx \frac{2[r(x + hv) - r(x) - hJv]}{h^2}, \quad h = 0.02, \qquad (J^\top J + \lambda D)a = -J^\top r_{vv}. \tag{12}$$

Here D is the floored diagonal of $J^\top J$. Backtracking tests $t = 1, 1/2, \ldots, 1/32$ with Armijo coefficient 10^{-4}. Acceptance requires positive predicted reduction and an actual-to-predicted ratio above 10^{-3}. Relative damping starts at 10^{-3}, decreases by a factor 0.3 after a full step with ratio above 0.75, increases by 3 below 0.25, and by 10 on rejection, within $[10^{-9}, 10^{6}]$. The curve envelope uses tolerance 10^{-14} times the mean absolute straight-trial cost, floored at 10^{-30}.

C WHITENING EQUIVARIANCE AND EQUIVALENT EXECUTION

Proof of Proposition 1. Let W and W' whiten the same positive-definite block metric N, with $W' = WO$ and block-diagonal orthogonal O. A physical gradient γ_ν, curvature B_ν and displacement history h are represented as

$$g_\nu = W^\top \gamma_\nu, \qquad H_\nu = W^\top B_\nu W, \qquad h_W = W^{-1} h. \tag{13}$$

Replacing W gives $g'_\nu = O^\top g_\nu$, $H'_\nu = O^\top H_\nu O$ and $h_{W'} = O^\top h_W$. Block energies and graph-weighted scalar seeds are unchanged. Ordered orthogonalization and Krylov actions therefore give $Q' = O^\top Q$ and $Y'_\nu = O^\top Y_\nu$. Carrier norms, block Gram entries and probe summaries are invariant, so the hidden states and predicted scalar coefficients are identical. Carrier mixing gives $V' = O^\top V$.

The curvature-orthogonal projector transforms by conjugation. Thus $Z' = O^\top Z$, the reduced matrices and right-hand sides are unchanged, and the same α gives $d' = O^\top d$. Mean-CG refinement obeys the same transformation. Mapping back yields $W'd' = Wd$. Physical acceleration, trial costs and damping consequently receive the same inputs; induction gives the same trajectory. The argument fixes the physical metric, block partition and graph, and assumes unchanged rank decisions. Floating-point branches are checked separately in the experiment below.

Cholesky execution. Symmetric whitening takes $W = N^{-1/2}$. For $N = LL^\top$, the equivalent factor is $W' = L^{-\top}$. Gradients use $L^{-1}\gamma$, physical displacements use $L^{-\top}d$, and history uses $L^\top h$. This distinction between gradient and displacement maps is essential. The implementation enables Cholesky for a single full block when $N - \tau I$ is positive definite, using the original floor τ. Other layouts and clipped metrics keep the symmetric implementation.

For a full mean block $N = (B_a + B_b)/2$, the whitened views satisfy $H_a + H_b = 2I$. After forming one view, the second is obtained by subtraction. Paired eigen and paired Cholesky controls separate this reuse from the factorization change.

Table 3. Full five-step time in seconds, medians of three rotated serial repeats after one warmup. All routes share the original network and recorded host/backend. The last two cases confirm a frozen implementation. Ratios divide original time by Cholesky time. Non-stereo cases take the original fallback. The full-block stereo cases have zero exact minimax correction; these timings isolate equivalent execution and whitening cost.

Case	Original	Paired eigen	Cholesky	Ratio
Biased graph	0.0251	0.0247	0.0253	0.99×
Multiframe	3.0780	3.0190	3.2319	0.95×
Stereo A	7.9020	6.6296	2.7307	2.89×
Low light	2.4179	2.4865	2.4280	1.00×
SIDD	0.8425	0.8729	0.8691	0.97×
Stereo B	9.3961	7.5215	2.9686	3.17×
Stereo C	9.0197	7.9213	2.9613	3.05×

Verification and timing. The pilot uses five cases; two further stereo cases test the frozen implementation. Each route receives one warmup and three rotated serial repeats on an Intel Core Ultra 7 155H, Windows 11, Python 3.13.7 and PyTorch 2.7.0.dev20250310, with one numerical thread. Timings cover the complete five-step solve with prepared problems and models. The 84 endpoints are rescored with zero discrepancy; the largest relative endpoint difference is 1.35×10^{-11}. All 420 step records preserve acceptance, step scale, damping, measured rank and curve selection.

The three stereo cases show 2.89–3.17× acceleration. In this full-block setting the mean whitened curvature is I and the base already contains its optimum $d_0 = -g$. Any nonzero increment raises the mean quadratic by $\|u\|^2/2$, so exact minimax adds no further correction. This experiment isolates execution equivalence and whitening cost; it is not intended to measure acceleration of the learned-completion component. Learned-direction gains are measured separately against random completion in Section 4.4.

Additional equivalence cases. We compare one five-step solve per implementation on 30 stereo crops from 15 Middlebury scenes, using two fixed crops per scene and the same weights. Relative endpoint differences stay below 2.19×10^{-10} and recovery errors differ by less than 9.86×10^{-12} pixels. All 150 paired steps preserve the five control decisions above. These runs check numerical equivalence only.

D TASKS AND COMPARISON PROTOCOL

Shared problems. Within each comparison, methods receive the same observations, masks, variable parameterization and initial state. The primary quality budget is five outer steps. Internal rejection, damping retries and line searches follow each method's implementation. The separate Jacobian-budget study is specified in Appendix H. Every SETSUNET run uses the same checkpoint.

Task metrics and inputs. Measurement graphs use state RMSE, Broyden uses residual RMS, and SE3/TUM use aligned translation RMSE in meters. HPatches pair and multiframe alignment use corner/reprojection error in pixels. Stereo uses disparity error in pixels. Restoration uses PSNR in dB. Results retain their task units; PSNR differences are additive, while reported relative error reductions specify whether they average per-instance ratios or compare group means.

The expanded visual protocols use HPatches, GoPro, LOL-v2, SIDD and Middlebury (Balntas et al., 2017; Nah et al., 2017; Yang et al., 2021; Abdelhamed et al., 2018; Scharstein et al., 2014):

- HPatches uses all 116 sequences and 580 pairs with a fixed feature front end; multiframe problems share this preparation.
- Known-blur recovery uses 64×64 regions from 66 frames in 11 GoPro videos, with a controlled blur operator. Low light uses 32×32 regions from 100 LOL-v2 test images.
- SIDD uses all 40 validation source images, four predefined standard blocks per image and one fixed 64×64 region per block. These three restoration groups use 16×16 tiled inference.
- Stereo uses 15 Middlebury scenes and 12 predefined crop positions per scene. These image experiments compare recovery optimizers on the stated regions, rather than full-resolution dataset leaderboards.

The expansion also includes 64 instances in each of four synthetic groups and 60 TUM windows from three sequences (Sturm et al., 2012). Shared preparation fails for one HPatches pair, its multiframe sequence and 13 TUM windows, leaving 1,503 cases; no replacements are added.

Baseline implementations. HSLM follows the authors' formulation with the small-dimension extension used in these experiments. Ceres LM executes the native library trust-region solver with DENSE_QR, a five-iteration limit and one numerical thread. Its function tolerance is 10^{-14} and gradient/parameter tolerances are 10^{-12}. Dogbox calls SciPy's rectangular trust-region solver with the exact shared residual/Jacobian interface, Jacobian-based scaling and 10^{-13} stopping tolerances; an iteration callback stops after five updates. Sparse Jacobians retain their sparse representation. Both libraries retain natural early termination.

VeLO uses the released HyperV2 numerical code and unchanged official meta-weights. The primary transfer flattens the physical state into one tensor, declares horizon five and executes five updates. It receives the shared scalar objective and gradient without task-fitted normalization or learning rates. The same configuration is used across all fixed-step comparisons. This evaluates transfer to inverse problems.

Reporting. The five comparison methods are SETSUNET, HSLM, Ceres LM, Dogbox and VeLO. Each scored case has all five endpoints. For error metrics $a, b \geq 0$, a practical tie satisfies $|a - b| \leq \max\{0.02 \min(a, b), 10^{-10} + 10^{-6} \max(a, b)\}$. The numerical floor prevents roundoff near zero from producing artificial wins. For PSNR scores p_a, p_b, a tie satisfies $|p_a - p_b| \leq 20 \log_{10}(1.02)$, corresponding to a 2% RMSE ratio at a common intensity range. Table 1 applies the same margin to each group mean and its best value; for mean PSNR this is a geometric-mean RMSE ratio. Larger advantages count as wins. This is a descriptive convention, not a test of statistical or perceptual significance. The two 32-case cohorts retain their distinct construction and selection histories.

E Per-Task Results and Comparison Coverage

Table 4 reports the 18-task heterogeneous benchmark. The same shared network is used for each formulation and state dimension. Table 5 identifies the additional studies; all five methods are evaluated on every retained case.

Table 4. Five-step recovery on the 18-task heterogeneous benchmark. Parentheses give state dimensions. RMSE and pixel errors are lower-is-better; PSNR is higher-is-better.

Task	Problem (dimension)	Metric	SETSUNET	HSLM	Ceres LM	Dogbox	VeLO
T01	Biased graph (48)	RMSE	0.1165	0.6537	0.2068	0.2068	0.7400
T02	Biased graph (96)	RMSE	0.2321	0.7267	0.2857	0.2857	0.7643
T03	SE3 (18)	m	0.0014	0.0026	0.0014	0.0014	0.0039
T04	Multiframe (16)	px	2.5350	3.6085	3.5863	3.7153	3.5862
T05	Multiframe (16)	px	0.1637	0.2540	0.2116	0.1849	0.2113
T06	Known blur (256)	dB	40.1678	34.6637	40.1625	40.1625	33.0457
T07	Low light (786)	dB	8.2994	8.2896	8.2192	8.2480	8.2079
T08	Low light (786)	dB	8.3023	8.2751	8.2210	8.2841	8.2091
T09	SIDD (770)	dB	34.1144	33.2233	34.1158	34.1152	34.2352
T10	Stereo (1536)	px	0.1884	0.3941	0.4047	0.3501	0.4047
T11	Biased graph (48)	RMSE	0.1572	0.7118	0.1978	0.1978	0.7740
T12	Biased graph (96)	RMSE	0.2363	0.6827	0.2910	0.2910	0.7244
T13	SE3 (30)	m	0.0038	0.0046	0.0038	0.0038	0.0040
T14	Multiframe (40)	px	11.7885	14.7951	15.2581	16.6917	28.6592
T15	Known blur (256)	dB	53.4392	53.0480	53.4322	53.4322	47.2617
T16	Low light (786)	dB	8.0832	8.0402	8.0253	8.0547	8.0252
T17	SIDD (770)	dB	35.4657	35.0917	35.4655	35.4651	35.4356
T18	Stereo (1536)	px	0.2503	0.2933	0.3222	0.3020	0.3222

Table 5. Comparison coverage. Common preparation failures affect every method. The complementary suite combines 32 extension cases and 32 biased graphs, retaining cohort identities. The derivative study counts VeLO gradients separately from Jacobians.

Study	Scored/reg.	Budget	Methods
Heterogeneous tasks	18/18	5 steps	All five
Complementary cases	64/64	5 steps	All five
Large collections	1,503/1,518	5 steps	All five
Graph and multiframe	20/20	5 steps	All five
Five-target multiframe	7/8	5 steps	All five
Optimizer transfer	11/11	5 steps	All five
Derivative budget	60/60	21 derivatives	All five

The 64-case complementary suite combines twelve additional synthetic cases, twenty public-image problems and 32 further biased graphs. Figure 2 uses the full common 64-case denominator for each comparison and the 2% practical-tie rule.

F Expanded Benchmarks and Transfer

Main-text Table 1 reports all 11 groups and all five methods on the same 1,503 prepared cases. Table 6 gives paired differences against the four comparators. The 10,000 bootstrap draws resample source sequences, images, scenes or synthetic seeds, preserving correlated crops within each source. Intervals are descriptive and have no simultaneous multiple-comparison correction.

Table 6. Mean recovery improvement and descriptive 95% source-cluster intervals. Positive favors SETSUNET. Units follow Table 1; restoration differences are in dB.

Group	HSLM	Ceres LM	Dogbox	VeLO
HPatches pairs	+10.3 [-0.111, +30.9]	-0.0381 [-0.175, +0.0456]	-0.0361 [-0.169, +0.0434]	+24.6 [-0.0399, +73.7]
Multiframe	+2.74 [-0.276, +7.17]	+3.49 [+0.17, +8.2]	+1.23 [-1.14, +4.41]	+12.3 [+3.72, +23]
Balanced graph	+0.447 [+0.399, +0.494]	-0.0989 [-0.116, -0.0819]	-0.0989 [-0.116, -0.0822]	+0.621 [+0.603, +0.637]
Biased graph	+0.343 [+0.291, +0.391]	+0.113 [+0.103, +0.124]	+0.113 [+0.103, +0.124]	+0.495 [+0.476, +0.515]
Broyden	+0.182 [+0.166, +0.201]	-5.37e-06 [-5.81e-06, -4.92e-06]	-5.37e-06 [-5.81e-06, -4.91e-06]	+0.522 [+0.492, +0.554]
SE3	+0.00406 [+0.00334, +0.00482]	-0.0015 [-0.00208, -0.000968]	-0.0015 [-0.00209, -0.000973]	+0.00446 [+0.00336, +0.00566]
TUM	+0.00696 [+0.00119, +0.0165]	-0.0225 [-0.0621, -0.00318]	-0.0226 [-0.0625, -0.00318]	+0.00859 [+0.00428, +0.0143]
Known blur	+1.77 [+1.67, +1.88]	-0.00067 [-0.00189, +0.000726]	-0.00067 [-0.00191, +0.00073]	+4.27 [+3.83, +4.75]
Low light	+0.113 [+0.0938, +0.133]	+0.118 [+0.0942, +0.141]	+0.0665 [+0.0549, +0.0787]	+0.0583 [+0.0472, +0.0693]
SIDD	+0.0876 [+0.0187, +0.206]	+0.0179 [+0.0139, +0.0222]	+0.00763 [+0.00577, +0.0097]	-0.0496 [-0.0836, +0.00719]
Stereo	+0.273 [+0.178, +0.388]	+0.347 [+0.233, +0.48]	+0.278 [+0.18, +0.396]	+0.347 [+0.234, +0.479]

The multiframe and stereo group means favor SETSUNET over all four comparators. The multiframe intervals against HSLM and Dogbox include zero, so these mean differences alone do not establish stable advantages across source sequences. The balanced-graph and pose results favor the classical library solvers. SIDD means are practically tied under the declared margin. Table 1 reports each group separately; the intervals retain their original metric differences.

Additional comparisons. The 20-case study contains fourteen synthetic problems and six public multiframe scenes. SETSUNET has win/tie/loss counts of 18/0/2, 12/2/6, 12/1/7, 16/1/3 against HSLM, Ceres LM, Dogbox and VeLO, respectively. The eleven-case transfer study gives 8/2/1, 3/6/2, 3/6/2, 8/2/1 in the same order. Both use the 2% margin and the same primary VeLO configuration.

Seven of the eight five-target multiframe scenes pass shared preparation. Mean errors are 8.5793, 8.3740, 8.3206, 7.9846, 180.5865 pixels for SETSUNET, HSLM, Ceres LM, Dogbox and VeLO, respectively. This larger observation graph preserves the same method definitions and five-step budget; all five methods use the same seven scenes.

G Direction Comparisons, Coordination and Size Transfer

Paired graph construction. The paired direction study uses 16 new seeds, 91914001–91914016, with 96 variables. Edges connect $(i, i+1)$ and $(i, i+3)$ where valid, with anchors $0.1I$. Truth is a sinusoid with Gaussian perturbations of standard deviation 0.2. Two Gaussian observation draws are generated once per scene and reused across strengths. View 0 has noise standard deviation 0.01. The noisy-view sweep gives view 1 standard deviation $0.01 + 0.09s$; the bias sweep keeps both at 0.01 and adds $0.12s\sin(t)$ to view 1, where measurement rows span $t \in [0, 4\pi]$. Strengths are $s = 1/3, 2/3, 1$, plus one common balanced condition, producing 112 conditions.

Direction and coordination controls. The reported direction-source comparison uses learned completion and three equal-rank random completions with the same minimax coefficients and CG2 refinement. The coordination comparison uses the same learned directions with mean or minimax coefficients, separately with CG2 and without CG. All reported configurations use five steps and the same observations. The straight realization isolates coordination on these linear residuals.

The three random seeds are 17011–17013. Their recovery errors are averaged before comparison with the learned model. Each scene contributes one effect averaged over its three nonzero strengths. Descriptive intervals use 20,000 scene-seed bootstrap draws, keeping strengths together. Main-text Table 2 reports learned-versus-random and coordination effects under CG2.

Table 7. Minimax versus mean coordination under systematic bias, with the same learned directions. Positive relative RMSE reduction favors minimax.

Bias strength	CG2	CG0
$2/3$	+14.36% [10.98, 17.87]	-3.23% [-5.24, -1.26]
1	+12.08% [10.10, 14.26]	-0.72% [-2.62, 1.14]

This comparison supports the joint effect of consensus and refinement under bias. With asymmetric noise, mean coordination has lower recovery error. Local quadratic protection describes the completion subproblem; recovery is measured after the whole trajectory.

Size transfer. Two new biased-graph seeds are evaluated at each dimension with the same checkpoint and $8 + 8$ direction budget. Learned directions beat equal-rank random directions in all six cases.

Table 8. Mean relative recovery gain of learned completion over equal-rank random completion on the size-transfer instances.

Variables	Versus random
48	+16.401%
192	+33.093%
768	+24.845%

On the same 192-variable instance, graph-isomorphic relabeling changes the mapped-back endpoint by 4.958×10^{-13}. Scaling both view objectives by ten changes it by 4.197×10^{-13}. Scaling only one view changes the objective balance and the endpoint, as expected.

H JACOBIAN BUDGETS

Jacobian budget. Six task types contribute ten small cases each: biased graphs, SE3, multiframe alignment, stereo, known blur and low light. Each reported endpoint comes from one budgeted solve. A joint two-view Jacobian evaluation costs one unit; an individual view costs one half. Internal recomputations count, and cached reuse does not. Recovery is taken from the latest completed iterate within the 21-unit cap, without selecting an iterate by its recovery score.

Table 9. Recovery with a cap of 21 joint Jacobians; VeLO uses 21 joint gradients, reported separately. Ten cases per task.

Task	SETSUNET	HSLM	Ceres LM	Dogbox	VeLO
Biased graph	0.2478	0.1862	0.2587	0.2587	0.7334
SE3	0.0035	0.0039	0.0035	0.0035	0.0061
Multiframe	2.4655	2.0605	2.0399	3.7360	9.5249
Stereo	3.7066	3.9112	3.9361	3.8892	3.9438
Known blur	40.9827	40.9825	40.9825	40.9825	40.3676
Low light	9.9075	8.0177	8.0344	7.9807	8.0846

All five methods have 60 usable records. SETSUNET consumes 21 joint-Jacobian units per case; its graph, SE3 and blur endpoints are at unit 20, while the remaining endpoints are at 21. The inputs use small synthetic problems, stereo crops, three-target multiframe graphs and 16×16 restoration tiles. Solvers returning below the cap retain their returned endpoint and actual count. VeLO receives 21 joint-gradient evaluations with declared horizon 21; its column is not an equal-Jacobian-cost comparison. Incomplete endpoints receive no substitute quality score.

Among these five routes, SETSUNET has the best mean low-light and stereo quality. HSLM has lower biased-graph error and Ceres LM has lower multiframe error. Together with the five-step study, the derivative study evaluates two budgets relevant to local deployment.

Evaluation scope. Cross-method recovery is compared using outer-step and derivative budgets only. The current implementations lack comparable low-level optimization and backend adaptation, so elapsed times are not representative of algorithmic efficiency. The within-SETSUNET whitening comparison in Appendix C separately tests equivalent implementations of the same solver.

The supplementary ZIP includes the minimal inference implementation and compact result tables. Detailed per-case results and derivative counts are available from the authors upon request.