SHARED TRANSFORM LEARNING AND HALF THRESHOLDING FOR TENSOR RESTORATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Low-rank tensor models provide a global reference for removing sparse corruption, but a compact representation can omit genuine signal retained in the observations. We propose ST-HALFNET, an unfolded network with a learned shared tensor transform and Half thresholding for sparse corruption estimation. An orthogonal transform and nonuniform slice ranks are learned jointly under a total rank budget. Scaled factor updates estimate a reference for each input, and robust local regression predicts the signal omitted by that reference. Half thresholding then estimates corruption from the corrected residual. The restored image is obtained by subtracting this estimate from the observation, preserving signal outside the reference family. We establish optimal rank allocation for a fixed transform and derive a restoration error bound separating missed corruption, prediction error, and shrinkage. On full-resolution CAVE and ICVL images, ST-HalfNet achieves the highest mean PSNR across the tested salt-and-pepper densities and transfers to random-valued and sparse Gaussian corruption with unchanged weights, exceeding the strongest baseline by 2.94 to 13.14 dB in the transfer settings. Comparisons with the best clean approximation show that a learned reference can guide restoration beyond its own approximation accuracy.

1 INTRODUCTION

Low-rank tensor restoration can discard image detail that corruption has left intact. Low-rank models exploit spatial and spectral redundancy (Candès et al., 2011; Lu et al., 2020), but fine textures and narrow boundaries can fall outside their representation. Figure 1 shows thin branches that remain in the observation but are blurred in the low-rank estimate. Even an optimally fitted reference remains limited by its representation. The challenge is to use global low-rank structure to identify corruption while preserving reliable measurements of detail beyond the model's capacity.

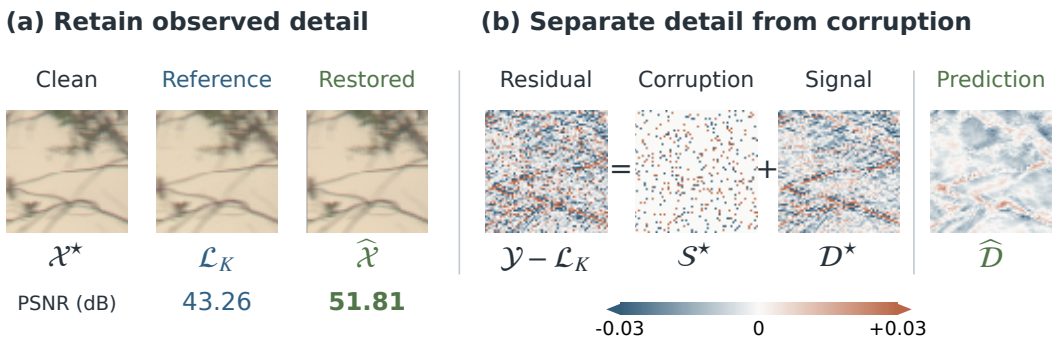


Figure 1: **Preserving detail omitted by a low-rank reference.** (a) Sparse correction retains fine branches blurred in the reference. (b) The observed residual contains both corruption and signal; local regression predicts the signal component. ICVL at 10% salt-and-pepper corruption; scores use the full image. Residuals at 550 nm share a scale of ± 0.03.

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.

Restoring the observation by sparse subtraction requires distinguishing corruption from omitted signal. For $\mathcal{Y} = \mathcal{X}^\star + \mathcal{S}^\star$, a reference $\mathcal{L}$ leaves

$$\mathcal{Y} - \mathcal{L} = \underbrace{\mathcal{X}^\star - \mathcal{L}}_{\text{signal residual } \mathcal{D}^\star} + \mathcal{S}^\star. \tag{1}$$

Here $\mathcal{X}^\star$ is the clean tensor and $\mathcal{S}^\star$ is sparse corruption. Direct thresholding can mistake the signal residual $\mathcal{D}^\star$ for corruption; raising the threshold protects more signal but leaves more corruption undetected. Predicting the omitted signal can reduce this ambiguity before thresholding.

ST-HALFNET learns a shared representation for the reference and predicts signal outside it. Changing spectral coordinates changes the spatial singular values and the benefit of each rank unit. We jointly learn an orthogonal transform and nonuniform slice ranks from clean training tensors, allocating a fixed total budget to structure shared across images. Scaled factor updates then estimate a reference for each corrupted input.

Robust local regression predicts the omitted signal from neighboring residuals. The corrected residual contains corruption perturbed by prediction error, which can be small even when the omitted signal is substantial. Half thresholding limits shrinkage of active corruption coefficients, since underestimation would leave corruption in the restored image. Subtracting this estimate from $\mathcal{Y}$ preserves observations outside its support. The reference guides the correction without constraining the output to its representational capacity.

We establish optimal rank allocation for a fixed transform and derive a restoration bound separating missed corruption, prediction error, and shrinkage. Trained ablations on CAVE and ICVL support the shared representation and Half estimator; paired outputs isolate the effect of local prediction. At 10% salt-and-pepper corruption, mean restoration PSNR exceeds that of the best clean approximation by 4.27 dB on CAVE and 11.77 dB on ICVL. A compact reference can thus guide recovery of details beyond its own representational capacity.

2 RELATED WORK

The t-product and invertible transforms express tensor operations through matrix slices (Kilmer & Martin, 2011; Kilmer et al., 2013; Kernfeld et al., 2015), providing a basis for tensor RPCA (Lu et al., 2020; Lu, 2021). Q-rank and learned transforms adapt the coordinates (Kong et al., 2021; Wu et al., 2024; Wang et al., 2025), while multirank formulations allow distinct slice ranks (Zhang et al., 2014; Song et al., 2023). Variable projection jointly optimizes orthogonal coordinates and their tensor representations (Newman & Keegan, 2025). Our representation is shared across clean training tensors and couples these coordinates with an exact allocation of total slice rank.

Unfolding learns the parameters of a finite optimization sequence. Alternating projections and scaled gradient methods provide efficient tensor RPCA updates (Qiu et al., 2022; Dong et al., 2023b; Wu, 2026); unfolded matrix, Tucker, and t-SVD methods learn steps and thresholds (Cai et al., 2021; Dong et al., 2023a; Feng et al., 2025). Li et al. (2025) couple thresholded t-SVD with a sparse transformer. ST-HalfNet uses scaled updates to construct a reference and Half thresholding to estimate corruption with limited shrinkage of active coefficients (Xu et al., 2012).

Spatial information complements global low rank through total variation (Wang et al., 2018; Peng et al., 2023) or learned spatial and spectral features (Bodrito et al., 2021; Lai et al., 2023; Ye et al., 2024; Wu et al., 2025). Our local polynomial regression (Takeda et al., 2007) predicts signal omitted by the global reference; the resulting sparse estimate determines which observations to change.

3 LEARNING THE SHARED REPRESENTATION

The shared representation determines which spatial structure the reference can retain under a fixed rank budget. Let $\{\mathcal{X}_i\}_{i=1}^N$ be nonzero clean training tensors with $\mathcal{X}_i \in \mathbb{R}^{n_{1,i} \times n_{2,i} \times d}$. Write $X^{(b)} = \mathcal{X}(:,:,b)$ for the bth matrix slice. An orthogonal matrix $Q \in \mathbb{R}^{d \times d}$, $Q^\top Q = I$, defines spectral coordinates through a mode-3 transform (Kernfeld et al., 2015):

$$[\mathcal{T}_Q(\mathcal{X})]^{(s)} = \sum_{b=1}^{d} Q_{sb} X^{(b)}, \qquad \mathcal{T}_{Q^\top}\mathcal{T}_Q(\mathcal{X}) = \mathcal{X}. \tag{2}$$

This common change of spectral coordinates preserves the Frobenius norm.

The vector $\mathbf{r} = (r_1, \ldots, r_d)$ assigns a matrix rank to each transformed slice and defines

$$\mathcal{M}(Q, \mathbf{r}) = \{\mathcal{L} : \operatorname{rank}([\mathcal{T}_Q(\mathcal{L})]^{(s)}) \le r_s,\ s = 1, \ldots, d\},$$
$$\mathcal{A}_R = \{\mathbf{r} \in \mathbb{N}_0^d : \textstyle\sum_s r_s = R,\ r_s \le r_{\max}\}. \tag{3}$$

Here $r_{\max} \le \min_i \min(n_{1,i}, n_{2,i})$ and $R \in \{0, \ldots, dr_{\max}\}$. The pair $(Q, \mathbf{r})$ is shared across inputs; the matrix factors within this family are estimated for each observation.

By orthogonality and the Eckart–Young theorem (Eckart & Young, 1936), truncating each transformed SVD gives the exact best approximation in this family. With singular values in decreasing order, its relative squared error and training objective are

$$j(\mathcal{X}; Q, \mathbf{r}) = \frac{\sum_s \sum_{a > r_s} \sigma_a^2([\mathcal{T}_Q(\mathcal{X})]^{(s)})}{\|\mathcal{X}\|_{\mathrm{F}}^2}, \qquad J(Q, \mathbf{r}) = \sum_i w_i j(\mathcal{X}_i; Q, \mathbf{r}). \tag{4}$$

We use $w_i = 1/N$; the analysis holds for $w_i \ge 0$ with $\sum_i w_i = 1$. Normalization weights approximation relative to each image's energy. Minimizing J couples coordinates and ranks: changing Q changes the spatial singular values, and hence which slices benefit most from additional rank. The objective measures the structure retained within slices, beyond their total spectral energy.

For fixed Q, the gain from adding the ath rank unit to slice s is

$$g_{s,a}(Q) = \sum_i w_i \frac{\sigma_a^2([\mathcal{T}_Q(\mathcal{X}_i)]^{(s)})}{\|\mathcal{X}_i\|_{\mathrm{F}}^2}, \qquad 1 \le a \le r_{\max}. \tag{5}$$

Proposition 1 (Optimal rank allocation)**.** *For fixed Q, select the largest R values among the $dr_{\max}$ gains in Eq.* (5)*, resolving ties within a slice by increasing a. Let r_s be the number selected from slice s. Then $\mathbf{r} \in \mathcal{A}_R$ and it globally minimizes $J(Q, \mathbf{r})$ over $\mathcal{A}_R$.*

Nonincreasing gains ensure that each slice retains its first r_s singular directions. Allocation therefore maximizes retained training energy without searching over rank vectors (Appendix A.1). The transform determines the gains to which this exact allocation is applied.

For transform learning, write $J_R(Q) = \min_{\mathbf{r} \in \mathcal{A}_R} J(Q, \mathbf{r})$. We initialize with the spectral eigenbasis Q_0 of the normalized training references and parameterize $Q = \exp(A)Q_0$, $A^\top = -A$. Each iteration takes a gradient step with ranks fixed, accepts it by Armijo decrease in J, and refits the ranks exactly. Every accepted nonzero step therefore satisfies

$$J_R(Q_{t+1}) \le J(Q_{t+1}, \mathbf{r}_t) < J(Q_t, \mathbf{r}_t) = J_R(Q_t). \tag{6}$$

Ranks are allocated on the training references. Validation evaluates each resulting pair and selects one shared representation for restoration. Appendix A.2 specifies the gradient and step acceptance. The selected Q and $\mathbf{r}$ remain fixed while the restoration parameters are learned.

4 ST-HalfNet

ST-HalfNet alternates sparse correction with reference fitting, then predicts signal in the final residual (Figure 2). With $Q, \mathbf{r}$ shared across inputs, each observation has factors $U_k^{(s)} \in \mathbb{R}^{n_1 \times r_s}$ and $V_k^{(s)} \in \mathbb{R}^{n_2 \times r_s}$, giving $\mathcal{L}_k = \mathcal{T}_{Q^\top}(\{U_k^{(s)} V_k^{(s)\top}\}_{s=1}^d)$. A truncated SVD of the transformed, clipped observation initializes balanced factors. Subsequent residuals use the original observation. The $K = 24$ factor updates share $T = 12$ parameter groups through $j(k) = \lfloor k/2 \rfloor$, with a separate terminal threshold τ_T.

4.1 Estimating the low-rank reference

At update k, the residual $\mathcal{R}_k = \mathcal{Y} - \mathcal{L}_k$ gives the entrywise sparse estimate $\mathcal{S}_k = H_{t_k}(\mathcal{R}_k)$. For scalar input v and threshold $t > 0$, Half thresholding minimizes $\frac{1}{2}(x - v)^2 + (2t/3)^{3/2}|x|^{1/2}$, with zero selected when $|v| \le t$ (Xu et al., 2012). An active coefficient keeps the sign of v and satisfies

$$0 < |v| - |H_t(v)| < t/3, \qquad |v| > t. \tag{7}$$

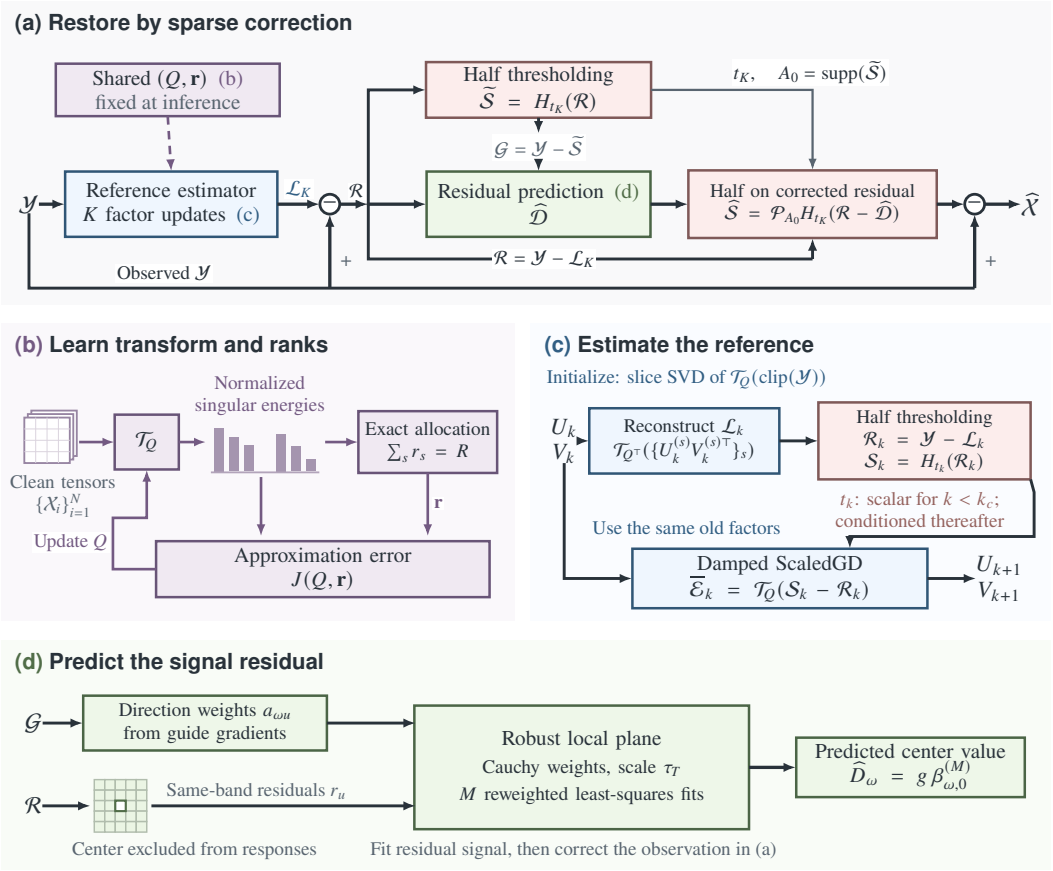


Figure 2: **ST-HalfNet.** (a) Restore the observation by sparse correction. (b) Clean training tensors determine a shared transform and rank allocation. (c) Coupled factor and sparse updates estimate a reference for each input. (d) Local prediction separates residual signal from corruption before the final correction.

Soft thresholding at the same activation threshold removes t from every active magnitude. Half therefore contributes less shrinkage to restoration when the estimated corruption is subtracted. Appendix A.3 derives the map and this inequality.

Thresholds use local structure and residual energy to adapt the corruption estimate. From $z = \tanh(\mathcal{R}_k/\tau_{j(k)})$, we measure agreement with spectral and spatial neighbors, local energy relative to its image mean, and global energy. These four features form $f_\omega(\mathcal{R}_k)$ at entry ω. Shared coefficients $a \in \mathbb{R}^4$ modulate a positive base threshold:

$$t_{k,\omega} = \tau_{j(k)} \exp\big(a^\top f_\omega(\mathcal{R}_k)\big), \qquad \mathcal{S}_k = H_{t_k}(\mathcal{R}_k). \tag{8}$$

Conditioning begins at update k_c; earlier updates use $\tau_{j(k)}$. Feature values depend on the input, while their coefficients are shared (Appendix B.2).

With $\mathcal{S}_k$ fixed, the factors fit the corrected observation through the objective $\frac{1}{2}\|\mathcal{L} + \mathcal{S}_k - \mathcal{Y}\|_{\mathrm{F}}^2$. Let $\overline{\mathcal{E}}_k = \mathcal{T}_Q(\mathcal{L}_k + \mathcal{S}_k - \mathcal{Y})$. Each nonzero-rank slice takes a simultaneous damped ScaledGD step (Dong et al., 2023b; Wu, 2026)

$$U_{k+1}^{(s)} = U_k^{(s)} - \eta_{j(k)} \overline{E}_k^{(s)} V_k^{(s)} (G_{V,k}^{(s)})^{-1}, \tag{9}$$

$$V_{k+1}^{(s)} = V_k^{(s)} - \eta_{j(k)} \overline{E}_k^{(s)\top} U_k^{(s)} (G_{U,k}^{(s)})^{-1}, \tag{10}$$

where $G_V = V^\top V + (\gamma \operatorname{tr}(V^\top V)/r_s + \epsilon) I$, with $\gamma \geq 0$, $\epsilon > 0$; G_U is defined analogously. Both updates use the current factors. Gram rescaling accounts for their magnitudes; Table 5 gives the numerical settings.

4.2 PREDICTING SIGNAL LEFT OUTSIDE THE REFERENCE

The estimated reference provides a global signal model, but its residual still contains the two components in Eq. (1). We now predict the signal component $\mathcal{D}^\star = \mathcal{X}^\star - \mathcal{L}_K$ before the final corruption estimate. Set $\mathcal{R} = \mathcal{Y} - \mathcal{L}_K$, compute $\widetilde{\mathcal{S}} = H_{t_K}(\mathcal{R})$ using the terminal base threshold τ_T, and form the guide $\mathcal{G} = \mathcal{Y} - \widetilde{\mathcal{S}}$.

We predict each residual from a local plane fitted to neighbors in the same band (Takeda et al., 2007), excluding the center from the responses. Gradients of the guide define a local structure tensor J_ω and weights

$$a_{\omega u} = \exp\left(-\kappa \frac{u^\top J_\omega u}{\operatorname{tr} J_\omega + \tau_T^2}\right), \tag{11}$$

where u is a window offset normalized by its radius. Directional weights favor neighbors along edges, while Cauchy residual weights of scale τ_T reduce the influence of contaminated responses. After M reweighted plane fits, a learned gain g multiplies the fitted intercept to obtain $\widehat{D}_\omega$. All responses come from the observed residual.

The provisional nonzero locations define $A_0 = \operatorname{supp}(\widetilde{\mathcal{S}})$. Let $\mathcal{P}_B$ retain entries in an index set B and set all others to zero. Subtracting the predicted signal from the residual gives the final estimates

$$\widehat{\mathcal{S}} = \mathcal{P}_{A_0} H_{t_K}(\mathcal{R} - \widehat{\mathcal{D}}), \qquad \widehat{\mathcal{X}} = \mathcal{Y} - \widehat{\mathcal{S}}. \tag{12}$$

The computed thresholds and A_0 remain fixed during this step. Local prediction can reject candidate corrections or revise their magnitude; entries outside the final support keep their observed values. The output therefore draws on both the reference and the measurements that survive sparse corruption.

Training. The objectives follow the two roles of the network: fit the reference, then optimize the corrected image. After learning $(Q, \mathbf{r})$ from clean tensors, we hold them fixed and train through all $K = 24$ updates. A loss on $\mathcal{L}_K$ is followed by a loss on $\widehat{\mathcal{X}}$, both using per-image log MSE. Steps, thresholds, and conditioning are fitted in both phases; the second also fits g. The shared transform accompanies 30 scalar restoration parameters (Appendix C.3).

5 RESTORATION ERROR ANALYSIS

Restoration depends on prediction and shrinkage at modified entries, while unmodified entries keep their measurements. Write $e = \mathcal{L}_K - \mathcal{X}^\star$ and $\mathcal{D}^\star = -e$. The residual prediction error is $q = \widehat{\mathcal{D}} - \mathcal{D}^\star = e + \widehat{\mathcal{D}}$. The final thresholding acts on $\mathcal{V} = \mathcal{S}^\star - q$.

Let $t = t_K > 0$ entrywise, let $\Omega = \operatorname{supp}(\mathcal{S}^\star)$, and define $A = A_0 \cap \{\omega : |V_\omega| > t_\omega\}$. Here Ω contains the true corruption locations and A contains the entries actually modified by the estimator.

Proposition 2 (Restoration error bound)**.** *For the estimator in Eq.* (12)*, let* $b_\omega = |V_\omega| - |H_{t_\omega}(V_\omega)|$ *on* A*. Its entrywise error is*

$$\widehat{X}_\omega - X^\star_\omega = \begin{cases} S^\star_\omega, & \omega \notin A, \\ q_\omega + \operatorname{sign}(V_\omega) b_\omega, & \omega \in A. \end{cases} \tag{13}$$

Consequently,

$$\|\widehat{\mathcal{X}} - \mathcal{X}^\star\|_{\mathrm{F}}^2 \le \underbrace{\|\mathcal{P}_{\Omega \setminus A} \mathcal{S}^\star\|_{\mathrm{F}}^2}_{\textit{missed corruption}} + \left(\underbrace{\|\mathcal{P}_A q\|_{\mathrm{F}}}_{\textit{prediction error}} + \underbrace{\tfrac{1}{3}\|\mathcal{P}_A t\|_{\mathrm{F}}}_{\textit{shrinkage}} \right)^2. \tag{14}$$

The bound follows from Eq. (7) and the triangle inequality. It shows where reference accuracy matters: on modified entries, its error enters through $q = e + \widehat{\mathcal{D}}$; at unmodified intact entries, the observation is already correct. Local prediction can therefore compensate for reference error instead of forcing a higher-rank fit. At the same corrected residual, thresholds, and candidate support, Soft has shrinkage term $\|\mathcal{P}_A t\|_{\mathrm{F}}$, compared with $\frac{1}{3}\|\mathcal{P}_A t\|_{\mathrm{F}}$ for Half. Appendix A.4 proves the bound and gives sufficient conditions for activating corrupted entries.

Local prediction error. A local plane fit with a design matrix of rank three reproduces affine signal residuals when neighboring responses are uncorrupted and $g = 1$. More generally, q separates into gain bias, local approximation error, and weighted neighbor contamination. Appendix A.5 derives this decomposition from the equivalent regression weights. It explains how a structured residual can be large while its prediction error is small.

Restoration beyond the reference family. The best clean approximation has squared error

$$\delta^2 = \min_{\mathcal{L} \in \mathcal{M}(Q,\mathbf{r})} \|\mathcal{L} - \mathcal{X}^\star\|_{\mathrm{F}}^2 = \|\mathcal{X}^\star\|_{\mathrm{F}}^2 j(\mathcal{X}^\star; Q, \mathbf{r}). \tag{15}$$

Whenever the bound in Eq. (14) is below δ^2, restoration is more accurate than every member of the reference family. Learning $(Q, \mathbf{r})$ improves the reference available for corruption estimation; preserved measurements and residual prediction let the output exceed its capacity. The experiments measure approximation and restoration separately to test this distinction.

6 Experiments

6.1 Image restoration and transfer

We use CAVE with 20/4/8 training/validation/test cubes and ICVL with 162/20/20 (Yasuma et al., 2010; Arad & Ben-Shahar, 2016). All test images retain their native spatial dimensions and 31 bands. We train one ST-HalfNet model per dataset with a salt-and-pepper corruption rate $p \sim \mathrm{Uniform}[0.05, 0.30]$ for each training image. For transfer, selected entries are replaced by independent $\mathrm{Uniform}[0, 1]$ values or receive additive Gaussian noise with $\sigma = 50/255$. The weights and solver settings remain fixed. Each method receives identical corrupted inputs. PSNR uses full-cube MSE with peak one; SSIM is averaged over bands, then each scene is weighted equally. Appendix C gives data scaling and training settings; Appendix E.1 reports all PSNR/SSIM results.

Table 1: **Salt-and-pepper restoration.** Mean PSNR (dB) on both complete test sets. Bold and underline denote the two highest values.

	CAVE			ICVL		
Method	10%	30%	50%	10%	30%	50%
Matrix PCP (Candès et al., 2011)	30.27	25.11	19.71	39.50	32.73	23.58
DFT-TRPCA (Lu et al., 2020)	39.79	37.34	32.89	42.06	39.66	32.30
EAPT-DCT (Qiu et al., 2022)	42.81	38.32	31.91	29.23	28.28	25.14
t-SVD ScaledGD (Wu, 2026)	32.64	31.52	26.95	30.27	29.78	27.11
Tucker unfolding (Dong et al., 2023a)	45.32	39.95	29.42	40.93	40.52	<u>34.78</u>
HSDT-M (Lai et al., 2023)	<u>45.81</u>	40.77	31.97	<u>45.19</u>	<u>40.75</u>	32.14
ILRNet (Ye et al., 2024)	45.56	<u>41.25</u>	29.77	35.37	30.19	22.73
MP-HSIR (Wu et al., 2025)	43.01	40.20	<u>34.23</u>	30.48	27.15	21.54
ST-HalfNet	**50.19**	**45.13**	**37.07**	**55.81**	**50.67**	**44.30**

ST-HalfNet has the highest mean PSNR in all six settings in Table 1. At 50% corruption, its advantage over the strongest comparator is 2.84 dB on CAVE and 9.52 dB on ICVL. Each dataset uses one model across densities, and inference receives only the corrupted cube. The following analyses distinguish improvements in the reference from those due to the final correction.

Transfer changes the corrupted values while preserving entrywise sparsity. With unchanged weights, ST-HalfNet exceeds the strongest competitor by 2.94 to 13.14 dB across the eight conditions (Table 2). Its advantage extends beyond the binary replacement used in training.

Figure 3 shows preserved contours and thin structures; Appendix E.2 adds scenes and spatial error maps across all 31 bands.

Table 2: **Transfer to other corruption distributions.** Mean PSNR (dB), using the same weights and solver settings as Table 1.

	CAVE				ICVL			
	Random-valued		Gaussian		Random-valued		Gaussian	
Method	10%	30%	10%	30%	10%	30%	10%	30%
Matrix PCP	30.46	25.98	31.62	29.46	38.98	31.54	40.16	36.39
DFT-TRPCA	39.78	36.66	40.19	38.69	<u>41.76</u>	38.09	42.33	40.68
EAPT-DCT	39.38	<u>37.53</u>	43.44	40.82	30.25	24.38	<u>45.80</u>	40.23
t-SVD ScaledGD	32.73	30.42	32.92	32.40	30.28	29.34	30.33	30.13
Tucker unfolding	<u>45.11</u>	30.83	<u>45.42</u>	<u>43.04</u>	40.92	<u>40.14</u>	40.99	<u>40.93</u>
HSDT-M	29.89	21.45	32.34	27.29	34.33	25.08	37.38	30.44
ILRNet	35.87	24.01	41.78	34.67	33.20	25.93	31.47	31.44
MP-HSIR	32.89	21.71	39.13	35.10	29.65	23.91	31.41	30.90
ST-HALFNET	**50.93**	**44.58**	**50.64**	**45.98**	**54.90**	**49.40**	**54.25**	**49.29**

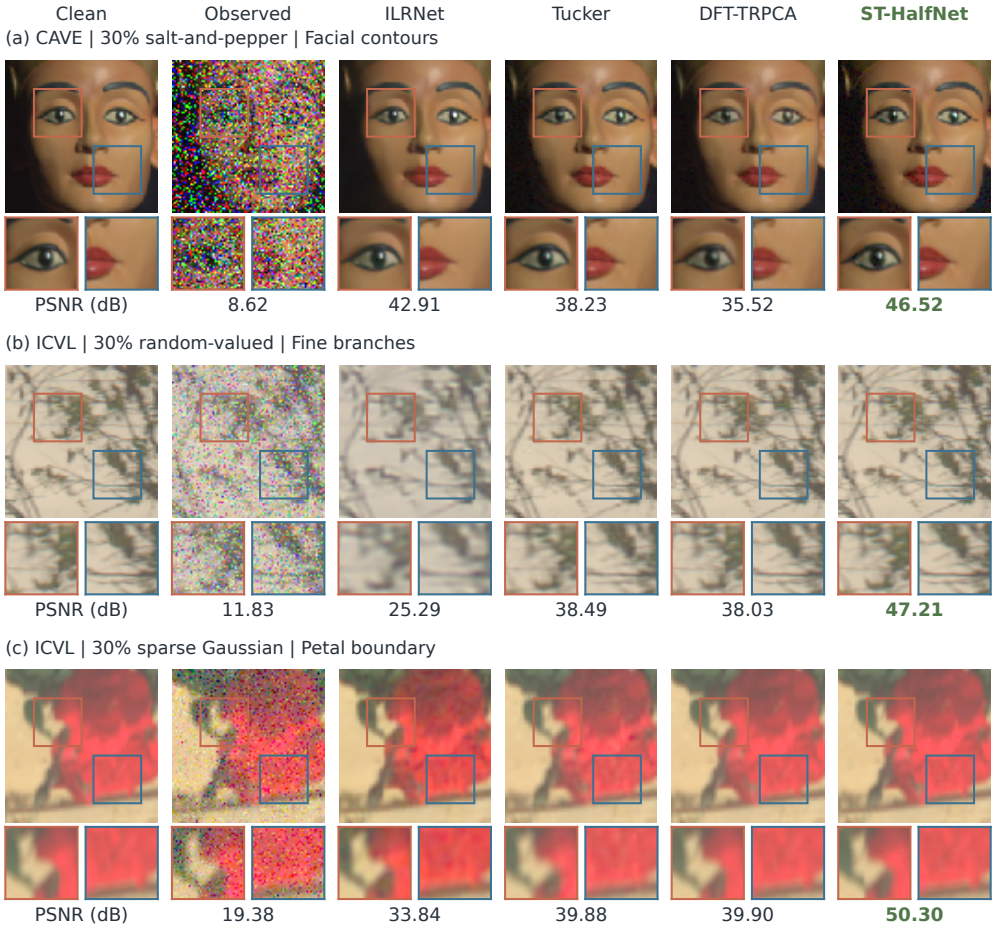


Figure 3: **Detail preservation across corruption distributions.** Facial contours, thin branches, and petal boundaries at 30% salt-and-pepper, random-valued, and sparse Gaussian corruption. Boxes mark enlarged details; scores are full-image PSNR (dB).

6.2 SIGNAL PRESERVATION AND CORRUPTION CORRECTION

Table 3 compares ST-HalfNet with residual thresholding ($\widehat{\mathcal{D}} = 0$) using the same trained network, estimated reference, thresholds, and candidate support.

Table 3: **Reference accuracy and residual prediction.** (a) Mean PSNR (dB). (b) MSE reduction relative to residual thresholding.

	CAVE			ICVL		
	10%	30%	50%	10%	30%	50%
(a) Reconstruction accuracy: mean PSNR (dB)↑						
Best clean approximation	45.917	45.917	45.917	44.041	44.041	44.041
Reference $\mathcal{L}_K$	45.257	42.843	36.610	43.919	43.375	40.915
Residual thresholding	48.950	44.238	36.958	51.566	46.982	42.403
ST-HalfNet	**50.186**	**45.131**	**37.072**	**55.807**	**50.672**	**44.298**
(b) MSE reduction from residual prediction (%)↑						
Uncorrupted entries (Ω^c)	36.6	29.1	9.8	69.0	68.2	55.7
Corrupted entries (Ω)	21.9	16.7	1.3	57.2	53.4	29.4

The best clean approximation bounds the PSNR attainable within the learned family for each image. At 10% corruption, residual thresholding already exceeds its mean PSNR on both datasets. Full restoration exceeds these means by 4.27 dB on CAVE and 11.77 dB on ICVL. The ICVL reference is only 0.12 dB below the approximation limit on average, so more accurate factor fitting within the same family cannot explain the restoration gain. Retaining and correcting observations recovers information excluded by the rank constraint.

Local prediction raises mean PSNR in all six paired comparisons while leaving the reference unchanged. On ICVL at 30%, the error contributions from uncorrupted and corrupted entries fall by 68.2% and 53.4%, respectively (Table 3(b)). The first reduction measures preserved signal at intact entries; the second measures more accurate correction where corruption is present. Figure 4 locates these improvements along fabric stripes and fine branches.

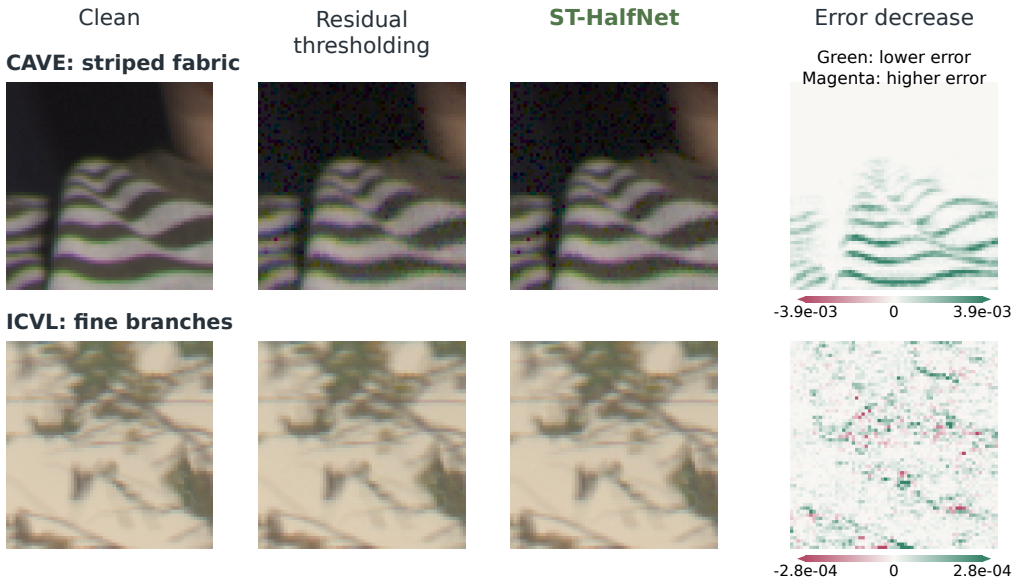


Figure 4: **Effect of local residual prediction.** Matched regions at 30% salt-and-pepper corruption. The right column shows squared error before prediction minus squared error after prediction, averaged over all bands. Green indicates reduced error. Both outputs use the same trained network.

For controlled tensors $\mathcal{X} = \mathcal{L}_0 + \alpha\mathcal{D}$, the more accurate output changes from L to $Y - S$ as signal is added outside the rank constraint, using the same estimated L, S without local prediction (Table 7).

6.3 SHARED REPRESENTATION AND HALF THRESHOLDING

Table 4 compares learned and fixed representations, and Half and Soft thresholding. The Soft variant retains the learned pair. All variants share the training data, rank budget $R = 512$, 24 factor updates, optimization budget, and validation rule.

Table 4: **Representation and thresholding ablations.** Mean test PSNR (dB); each variant is trained separately.

Dataset	Model	10%	30%	50%
CAVE	ST-HALFNET (full)	**50.19**	**45.13**	**37.07**
	Fixed transform and ranks	41.63	37.78	31.58
	Soft thresholding	46.83	41.30	32.81
ICVL	ST-HALFNET (full)	**55.81**	**50.67**	**44.30**
	Fixed transform and ranks	44.11	39.80	33.91
	Soft thresholding	48.88	44.47	39.29

Learning $(Q, \mathbf{r})$ improves mean PSNR over fixed DCT with uniform ranks by 5.49 to 11.70 dB across the six conditions. The approximation curves in Figure 5 explain how the reference benefits: at $R = 512$, the learned family's relative squared error is about $1/12$ and $1/36$ of the fixed representation's error on CAVE and ICVL. Adapting coordinates and allocation lets the same rank budget retain more spatial structure.

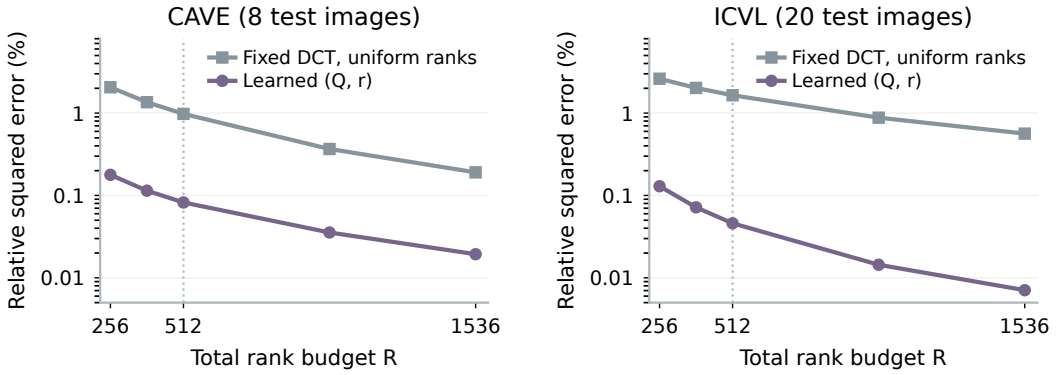


Figure 5: **Approximation under a rank budget.** Mean relative squared error, shown as a percentage; lower is better. The learned transform is fixed across budgets; its ranks use training references. The dotted line marks the model budget $R = 512$.

Replacing Half by Soft reduces mean PSNR by 3.36 to 6.93 dB across the six conditions, with the representation fixed and restoration parameters fitted separately. Thus the sparse estimator contributes beyond representation learning. This network comparison complements the smaller shrinkage term in Section 5; controls with matched activation thresholds in Appendix D examine the operator's effect directly.

7 CONCLUSION

ST-HalfNet uses a compact tensor reference to estimate sparse corruption while preserving signal outside the reference family. Learning shared coordinates and ranks improves the reference; local residual prediction and Half thresholding determine the correction applied to the observations. The error analysis explains their distinct roles, and comparisons with the best clean approximation demonstrate recovery beyond the rank constraint. These results support learning low-rank representations for the corrections they enable as well as the signal they can represent.

AI USE STATEMENT

We used generative AI tools for language editing, literature search and summarization, and limited feedback on research framing and experimental design. All AI-assisted suggestions were critically reviewed and independently verified by the authors; final research decisions, analyses, interpretations, and manuscript content were made by the authors. We take full responsibility for the final content of this work.

REPRODUCIBILITY STATEMENT

The appendix specifies the model, proves the theoretical results, and details the data splits, corruption generation, training settings, and evaluation metrics. The accompanying code provides training and evaluation entry points and generates the reported tables and figures from measurements for each image.

REFERENCES

Boaz Arad and Ohad Ben-Shahar. Sparse recovery of hyperspectral signal from natural RGB images. In *European Conference on Computer Vision*, pp. 19–34. Springer, 2016.

Théo Bodrito, Alexandre Zouaoui, Jocelyn Chanussot, and Julien Mairal. A trainable spectral-spatial sparse coding model for hyperspectral image restoration. In *Advances in Neural Information Processing Systems*, volume 34, pp. 5430–5442, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/2b515e2bdd63b7f034269ad747c93a42-Abstract.html.

HanQin Cai, Jialin Liu, and Wotao Yin. Learned robust PCA: A scalable deep unfolding approach for high-dimensional outlier detection. In *Advances in Neural Information Processing Systems*, volume 34, 2021.

Emmanuel J. Candès, Xiaodong Li, Yi Ma, and John Wright. Robust principal component analysis? *Journal of the ACM*, 58(3):11:1–11:37, 2011. doi: 10.1145/1970392.1970395.

Harry Dong, Megna Shah, Sean Donegan, and Yuejie Chi. Deep unfolded tensor robust PCA with self-supervised learning. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 1–5, 2023a. doi: 10.1109/ICASSP49357.2023.10095485.

Harry Dong, Tian Tong, Cong Ma, and Yuejie Chi. Fast and provable tensor robust principal component analysis via scaled gradient descent. *Information and Inference: A Journal of the IMA*, 12(3):1716–1758, 2023b. doi: 10.1093/imaiai/iaad019.

Carl Eckart and Gale Young. The approximation of one matrix by another of lower rank. *Psychometrika*, 1(3):211–218, 1936. doi: 10.1007/BF02288367.

Lanlan Feng, Ce Zhu, Yipeng Liu, Saiprasad Ravishankar, and Longxiu Huang. Learnable scaled gradient descent for guaranteed robust tensor PCA. *arXiv preprint arXiv:2501.04565*, 2025.

Eric Kernfeld, Misha Kilmer, and Shuchin Aeron. Tensor–tensor products with invertible linear transforms. *Linear Algebra and its Applications*, 485:545–570, 2015. doi: 10.1016/j.laa.2015.07.021.

Misha E. Kilmer and Carla D. Martin. Factorization strategies for third-order tensors. *Linear Algebra and its Applications*, 435(3):641–658, 2011. doi: 10.1016/j.laa.2010.09.020.

Misha E. Kilmer, Karen Braman, Ning Hao, and Randy C. Hoover. Third-order tensors as operators on matrices: A theoretical and computational framework with applications in imaging. *SIAM Journal on Matrix Analysis and Applications*, 34(1):148–172, 2013. doi: 10.1137/110837711.

Hao Kong, Canyi Lu, and Zhouchen Lin. Tensor Q-rank: New data dependent definition of tensor rank. *Machine Learning*, 110(7):1867–1900, 2021. doi: 10.1007/s10994-021-05987-8.

Zeqiang Lai, Chenggang Yan, and Ying Fu. Hybrid spectral denoising transformer with guided attention. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 13019–13029, 2023. doi: 10.1109/ICCV51070.2023.01201.

Liang Li, Jianli Zhao, Sheng Fang, Siyu Chen, and Hui Sun. A TRPCA-inspired deep unfolding network for hyperspectral image denoising via thresholded t-SVD and top-K sparse transformer. *arXiv preprint arXiv:2506.02364*, 2025.

Canyi Lu. Transforms based tensor robust PCA: Corrupted low-rank tensors recovery via convex optimization. In *IEEE/CVF International Conference on Computer Vision*, pp. 1145–1152, 2021.

Canyi Lu, Jiashi Feng, Yudong Chen, Wei Liu, Zhouchen Lin, and Shuicheng Yan. Tensor robust principal component analysis with a new tensor nuclear norm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(4):925–938, 2020. doi: 10.1109/TPAMI.2019.2891760.

Elizabeth Newman and Katherine Keegan. Optimal matrix-mimetic tensor algebras via variable projection. *SIAM Journal on Matrix Analysis and Applications*, 46(3):1764–1790, 2025. doi: 10.1137/24M1702635.

Jiangjun Peng, Yao Wang, Hongying Zhang, Jianjun Wang, and Deyu Meng. Exact decomposition of joint low rankness and local smoothness plus sparse matrices. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(5):5766–5781, 2023. doi: 10.1109/TPAMI.2022.3204203.

Haiquan Qiu, Yao Wang, Shaojie Tang, Deyu Meng, and Quanming Yao. Fast and provable nonconvex tensor RPCA. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 18211–18249. PMLR, 2022.

Guang-Jing Song, Michael K. Ng, and Xiongjun Zhang. Tensor completion by multi-rank via unitary transformation. *Applied and Computational Harmonic Analysis*, 65:348–373, 2023. doi: 10.1016/j.acha.2023.03.007.

Hiroyuki Takeda, Sina Farsiu, and Peyman Milanfar. Kernel regression for image processing and reconstruction. *IEEE Transactions on Image Processing*, 16(2):349–366, 2007. doi: 10.1109/TIP.2006.888330.

Xiangming Wang, Haijin Zeng, Jiaoyang Chen, Sheng Liu, Yongyong Chen, and Guoqing Chao. OTLRM: Orthogonal learning-based low-rank metric for multi-dimensional inverse problems. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39(20):21278–21286, 2025. doi: 10.1609/aaai.v39i20.35427.

Yao Wang, Jiangjun Peng, Qian Zhao, Yee Leung, Xi-Le Zhao, and Deyu Meng. Hyperspectral image restoration via total variation regularized low-rank tensor decomposition. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 11(4):1227–1243, 2018. doi: 10.1109/JSTARS.2017.2779539.

Tong Wu. Guaranteed nonconvex low-rank tensor estimation via scaled gradient descent. *Journal of Machine Learning Research*, 27(9):1–90, 2026. URL https://www.jmlr.org/papers/v27/25-0012.html.

Tongle Wu, Bin Gao, Jicong Fan, Jize Xue, and Wai Lok Woo. Low-rank tensor completion based on self-adaptive learnable transforms. *IEEE Transactions on Neural Networks and Learning Systems*, 35(7):8826–8838, 2024. doi: 10.1109/TNNLS.2022.3215974.

Zhehui Wu, Yong Chen, Naoto Yokoya, and Wei He. Mp-hsir: A multi-prompt framework for universal hyperspectral image restoration. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 13009–13020, 2025.

Zongben Xu, Xiangyu Chang, Fengmin Xu, and Hai Zhang. $l_{1/2}$ regularization: A thresholding representation theory and a fast solver. *IEEE Transactions on Neural Networks and Learning Systems*, 23(7):1013–1027, 2012. doi: 10.1109/TNNLS.2012.2197412.

Fumihito Yasuma, Tomoo Mitsunaga, Daisuke Iso, and Shree K. Nayar. Generalized assorted pixel camera: Postcapture control of resolution, dynamic range, and spectrum. *IEEE Transactions on Image Processing*, 19(9):2241–2253, 2010. doi: 10.1109/TIP.2010.2046811.

Jin Ye, Fengchao Xiong, Jun Zhou, and Yuntao Qian. Iterative low-rank network for hyperspectral image denoising. *IEEE Transactions on Geoscience and Remote Sensing*, 62:1–15, 2024. doi: 10.1109/TGRS.2024.3449130.

Zemin Zhang, Gregory Ely, Shuchin Aeron, Ning Hao, and Misha E. Kilmer. Novel methods for multilinear data completion and de-noising based on tensor-SVD. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3842–3849, 2014. doi: 10.1109/CVPR.2014.485.

A PROOFS

A.1 OPTIMAL RANK ALLOCATION

For an orthogonal Q, applying Q to each mode-3 fiber preserves its Euclidean norm. Summing over spatial locations gives $\|\mathcal{T}_Q(\mathcal{Z})\|_{\mathrm{F}}^2 = \|\mathcal{Z}\|_{\mathrm{F}}^2$. The transformed slices can be approximated independently. The Eckart–Young theorem (Eckart & Young, 1936) therefore gives Eq. (4).

Proof of Proposition 1. The total squared singular energy equals the tensor energy. Consequently,

$$J(Q, \mathbf{r}) = \sum_i w_i \frac{\sum_s \sum_{a>r_s} \sigma_a^2([\mathcal{T}_Q(\mathcal{X}_i)]^{(s)})}{\|\mathcal{X}_i\|_{\mathrm{F}}^2} \tag{16}$$

$$= 1 - \sum_{s=1}^{d} \sum_{a=1}^{r_s} g_{s,a}(Q). \tag{17}$$

Each sequence $g_{s,1}, \ldots, g_{s,r_{\max}}$ is nonnegative and nonincreasing. Order the $dr_{\max}$ gains by decreasing value, placing an earlier index before a later tied index in the same slice. If (s, a) is among the first R entries, so is every (s, b) with $b < a$. Thus the selected indices in each slice are exactly $1, \ldots, r_s$ for some $r_s \le r_{\max}$, and $\sum_s r_s = R$.

Every feasible rank vector selects R gains from this same list. Its retained energy is at most the sum of the largest R gains, which the constructed vector attains. Equation (17) proves optimality. The same argument covers $R = 0$, $R = dr_{\max}$, and zero gains. □

Once singular values are available, sorting the gains costs $\mathcal{O}(dr_{\max} \log(dr_{\max}))$ time and $\mathcal{O}(dr_{\max})$ memory. The transform and slice SVDs dominate fitting cost.

A.2 TRANSFORM DERIVATIVES AND DESCENT

For a matrix Z, define $\varphi_r(Z) = \min_{\mathrm{rank}(P)\le r} \|Z - P\|_{\mathrm{F}}^2$. When $0 < r < \min(n_1, n_2)$ and $\sigma_r(Z) > \sigma_{r+1}(Z)$, the truncated SVD $P_r(Z)$ is the unique minimizer. Differentiating the minimized residual gives

$$\nabla_Z \varphi_r(Z) = 2\big(Z - P_r(Z)\big). \tag{18}$$

This is the derivative of the optimized approximation value, with the optimizing projection held fixed in the partial derivative. For $r = 0$, it reduces to $2Z$; for full matrix rank, it is zero.

At fixed $\mathbf{r}$, let $\overline{X}_{is} = [\mathcal{T}_Q(\mathcal{X}_i)]^{(s)}$ and $E_{is} = \overline{X}_{is} - P_{r_s}(\overline{X}_{is})$. The chain rule yields

$$[\nabla_Q J]_{sb} = 2 \sum_i w_i \frac{\langle E_{is}, X_i^{(b)} \rangle_{\mathrm{F}}}{\|\mathcal{X}_i\|_{\mathrm{F}}^2}. \tag{19}$$

The denominator is independent of Q. Let θ contain the independent entries of a skew-symmetric matrix $A(\theta)$. Differentiating $Q(\theta) = \exp(A(\theta))Q_0$ gives the parameter gradient used by the algorithm. Skew symmetry implies $\exp(A)^\top \exp(A) = I$, so every iterate remains orthogonal.

The step direction is the parameter gradient, scaled down when its norm exceeds the specified clipping level. For $d_\theta \neq 0$, the accepted step $\theta^+ = \theta - \beta d_\theta$ obeys

$$J(Q(\theta^+), \mathbf{r}) \le J(Q(\theta), \mathbf{r}) - c\beta \|d_\theta\|_2^2, \qquad c > 0, \quad \beta > 0. \tag{20}$$

At a singular-value tie, a selected truncated SVD supplies the residual direction; acceptance is still evaluated with the exact approximation value. Thus every accepted nonzero step decreases the fixed-rank objective. Refitting the training ranks can only decrease it further:

$$J_R(Q_{t+1}) \le J(Q_{t+1}, \mathbf{r}_t) < J(Q_t, \mathbf{r}_t) = J_R(Q_t).$$

This proves Eq. (6). The validation score selects among the resulting pairs, with each rank vector retained from its training allocation.

A.3 Half thresholding and shrinkage

For $z \ge 0$ and $\lambda > 0$, consider

$$\phi(x) = \tfrac{1}{2}(x - z)^2 + \lambda x^{1/2}, \qquad x \ge 0. \tag{21}$$

The objective is continuous and tends to infinity with x, so it attains a global minimum. Every positive minimizer is stationary and satisfies

$$z - x = \frac{\lambda}{2\sqrt{x}}. \tag{22}$$

The zero and active objectives are equal at $x = \lambda^{2/3}$ and $z = 3\lambda^{2/3}/2$. Choosing $\lambda_\tau = (2\tau/3)^{3/2}$ sets the activation boundary to τ. For $|z| > \tau$, the selected minimizer is

$$H_\tau(z) = \operatorname{sign}(z)\frac{4|z|}{3}\cos^2\left[\frac{1}{3}\arccos\left(-\frac{3\sqrt{3}\lambda_\tau}{4|z|^{3/2}}\right)\right]. \tag{23}$$

To identify the global minimum, write the stationary equation as $z = h(x) = x + \lambda/(2\sqrt{x})$. The function h decreases up to $x = (\lambda/4)^{2/3}$ and increases thereafter. Thus an interior local minimum, when it exists, is the larger stationary solution. At a stationary point,

$$\phi(x) - \phi(0) = \tfrac{1}{2}\sqrt{x}\big(\lambda - x^{3/2}\big).$$

It improves on zero exactly when $x > \lambda^{2/3}$, which on the increasing branch is equivalent to $z > 3\lambda^{2/3}/2$. At equality both are global minima and our map selects zero. For signed input, an optimal nonzero value has the input's sign, since this minimizes the squared-distance term. The active solution satisfies $x > 2\tau/3$. Substituting λ_τ into Eq. (22) gives

$$0 < z - x < \frac{(2\tau/3)^{3/2}}{2\sqrt{2\tau/3}} = \frac{\tau}{3}, \tag{24}$$

which proves Eq. (7). Symmetry covers negative inputs.

A.4 Proof of Proposition 2: restoration error

Proof. On A^c, $\widehat{S}_\omega = 0$, so the error equals $S^\star_\omega$ and vanishes on $A^c \cap \Omega^c$. On A, $H_{t_\omega}(V_\omega) = V_\omega - \operatorname{sign}(V_\omega)b_\omega$. Substituting $V_\omega = Y_\omega - L_{K,\omega} - \widehat{D}_\omega$ gives $\widehat{X}_\omega = L_{K,\omega} + \widehat{D}_\omega + \operatorname{sign}(V_\omega)b_\omega$. The two regions are disjoint, so their squared errors add. The triangle inequality on A and $0 < b_\omega < t_\omega/3$ give Eq. (14). The argument applies to the computed positive thresholds and predictions. □

When $A = \Omega$, the condition

$$\|\mathcal{P}_\Omega(e + \widehat{\mathcal{D}})\|_{\mathrm{F}} + \tfrac{1}{3}\|\mathcal{P}_\Omega t\|_{\mathrm{F}} < \|e\|_{\mathrm{F}}$$

is sufficient for restoration to improve on $\mathcal{L}_K$. It compares the cost of correcting corrupted entries with the reference error removed from intact ones and permits approximately low-rank clean signals.

The improvement over the low-rank reference also has an exact expression. Let $C_\omega = \widehat{D}_\omega + \operatorname{sign}(V_\omega)b_\omega$ on A, and $C_\omega = 0$ elsewhere. The same disjoint decomposition yields

$$\|\widehat{\mathcal{X}} - \mathcal{X}^\star\|_{\mathrm{F}}^2 = \|\mathcal{P}_{\Omega\setminus A}\mathcal{S}^\star\|_{\mathrm{F}}^2 + \|\mathcal{P}_A e + C\|_{\mathrm{F}}^2.$$

Subtracting this expression from $\|e\|_{\mathrm{F}}^2 = \|\mathcal{P}_A e\|_{\mathrm{F}}^2 + \|\mathcal{P}_{A^c} e\|_{\mathrm{F}}^2$ and expanding the square gives

$$\begin{aligned}\|e\|_{\mathrm{F}}^2 - \|\widehat{\mathcal{X}} - \mathcal{X}^\star\|_{\mathrm{F}}^2 = \|\mathcal{P}_{A^c} e\|_{\mathrm{F}}^2 - \|\mathcal{P}_{\Omega\setminus A}\mathcal{S}^\star\|_{\mathrm{F}}^2 \\ - 2\langle \mathcal{P}_A e, C\rangle_{\mathrm{F}} - \|C\|_{\mathrm{F}}^2.\end{aligned} \tag{25}$$

The first term is the reference error removed at inactive entries; the second is missed corruption. The last two terms describe the combined effect of local prediction and shrinkage on activated entries. This identity applies to each computed iterate and its actual activation set. In particular, the interaction term can have either sign, so the effect of changing the local detail estimate together with shrinkage depends on their alignment with the low-rank error.

Selection of entries for correction. On an uncorrupted entry, the provisional residual is $-e_\omega$ and the corrected residual is $-q_\omega$. Thus $|e_\omega| \le t_\omega$ or $|q_\omega| \le t_\omega$ is sufficient for the entry to remain unchanged. At a corrupted entry, the inequality

$$|S_\omega^\star| > t_\omega + \max(|e_\omega|, |q_\omega|)$$

implies that both $S_\omega^\star - e_\omega$ and $S_\omega^\star - q_\omega$ exceed the threshold in magnitude and have the sign of $S_\omega^\star$. The entry then belongs to A, and its estimated corruption has the correct sign.

A.5 Local regression and prediction error

For the local fit in Appendix B.3, let V_ω contain rows $v_u^\top$ from the local plane design, and let W_ω contain the positive weights used in its last solve. When this design has column rank three, the intercept is a weighted sum of neighboring residuals, with coefficients

$$\ell_\omega^\top = e_1^\top (V_\omega^\top W_\omega V_\omega)^{-1} V_\omega^\top W_\omega, \qquad e_1 = (1,0,0)^\top. \tag{26}$$

Because $\ell_\omega^\top V_\omega = e_1^\top$, these coefficients reproduce constants and remove linear offsets: $\sum_u \ell_{\omega u} = 1$ and $\sum_u \ell_{\omega u} u = 0$. Writing $D_{\omega,u}^\star$ for the signal residual at the neighbor indexed by u, decompose it for any slope ν_ω as $D_{\omega,u}^\star = D_\omega^\star + \nu_\omega^\top u + \xi_{\omega u}$. Substituting $r_u = D_{\omega,u}^\star + S_{\omega,u}^\star$ into $\widehat{D}_\omega = g\sum_u \ell_{\omega u} r_u$ and subtracting $D_\omega^\star$ gives Eq. (27). The constant term uses $\sum_u \ell_{\omega u} = 1$, and the linear term vanishes because $\sum_u \ell_{\omega u} u = 0$:

$$q_\omega = (g-1) D_\omega^\star + g\sum_u \ell_{\omega u}\xi_{\omega u} + g\sum_u \ell_{\omega u} S_{\omega,u}^\star. \tag{27}$$

Thus

$$|\widehat{D}_\omega - D_\omega^\star| \le |g-1|\,|D_\omega^\star| + |g|\,\|\ell_\omega\|_1 \|\xi_\omega\|_\infty + |g|\left|\sum_u \ell_{\omega u} S_{\omega,u}^\star\right|. \tag{28}$$

This separates gain bias, departure from a local affine residual, and contamination retained by the fitted regression. The identities use the fitted, observation-dependent weights and describe the prediction term in Eq. (14). For $g = 1$ and uncorrupted affine responses, the intercept reproduces the center residual exactly.

B Model specification

Algorithm 1 separates the K coupled factor and sparse updates from the final local prediction. The transform, ranks, and learned scalar parameters are shared across observations; the factors and predictions are computed from each input. Appendix C gives the numerical settings.

B.1 Initialization and learned schedules

For training reference i, let $M_i \in \mathbb{R}^{(H_i W_i)\times d}$ contain its spectra as rows. The rows of Q_0 are the eigenvectors, in decreasing eigenvalue order, of

$$C = \frac{1}{N}\sum_{i=1}^{N} \frac{M_i^\top M_i}{\|\mathcal{X}_i\|_{\mathrm{F}}^2}.$$

Algorithm 1 ST-HalfNet restoration

Require: observation $\mathcal{Y}$; learned representation $(Q, \mathbf{r})$; steps $\{\eta_j\}_{j=0}^{T-1}$, thresholds $\{\tau_j\}_{j=0}^{T}$, and coefficients a, g
Require: depth K; sharing map j; conditioning onset k_c; local radius h, iterations M, and direction strength κ
Estimate the global reference
initialize balanced slice factors from the SVD of the transformed, clipped observation; reconstruct $\mathcal{L}_0$
for $k = 0, \dots, K-1$ **do**
  $\mathcal{R}_k \leftarrow \mathcal{Y} - \mathcal{L}_k$
  $t_k \leftarrow \tau_{j(k)}$ if $k < k_c$, otherwise Eq. (8)
  $\mathcal{S}_k \leftarrow H_{t_k}(\mathcal{R}_k)$
  transform $\mathcal{S}_k - \mathcal{R}_k$ and update both factors by Eq. (10)
  reconstruct $\mathcal{L}_{k+1}$
end for
Predict residual signal and correct the observation
$\mathcal{R} \leftarrow \mathcal{Y} - \mathcal{L}_K$; compute t_K from Eq. (8) with τ_T
$\widetilde{\mathcal{S}} \leftarrow H_{t_K}(\mathcal{R})$; $\mathcal{G} \leftarrow \mathcal{Y} - \widetilde{\mathcal{S}}$; $A_0 \leftarrow \mathrm{supp}(\widetilde{\mathcal{S}})$
compute $\widehat{\mathcal{D}}$ from $\mathcal{R}$ and $\mathcal{G}$ by Eqs. (31)–(33)
$\widehat{\mathcal{S}} \leftarrow \mathcal{P}_{A_0} H_{t_K}(\mathcal{R} - \widehat{\mathcal{D}})$
$\widehat{\mathcal{X}} \leftarrow \mathcal{Y} - \widehat{\mathcal{S}}$
return $\widehat{\mathcal{X}}$

This uncentered second moment initializes the optimization of J_R.

For each corrupted input, apply an economy SVD to each active slice of $\mathcal{T}_Q(\mathrm{clip}(\mathcal{Y}, -c, c))$. If the leading singular triplets are P_r, Σ_r, W_r, set $U_0 = P_r \Sigma_r^{1/2}$ and $V_0 = W_r \Sigma_r^{1/2}$. The clipping level c limits the influence of large observations on initialization; all subsequent residuals use the original observation.

The learned schedules are parameterized by

$$\eta_j = \eta_{\max} \,\mathrm{sigmoid}(c_j), \quad j = 0, \dots, T-1, \qquad \tau_j = \tau_{\min} + \sum_{a=j}^{T} \mathrm{softplus}(b_a), \quad j = 0, \dots, T.$$

Here $\eta_{\max} > 0$ and $\tau_{\min} > 0$. Base thresholds are positive and decreasing, while residual conditioning can vary the entrywise boundaries between updates that share a base threshold. The Gram systems in Eq. (10) are solved directly.

B.2 Residual features

Let $z = \tanh(\mathcal{R}_k / \tau_{j(k)})$. For spatial coordinates (i, j) and band b, set

$$c_{ijb}^{\mathrm{spec}} = z_{ijb} \,\mathrm{mean}_{b' \in \mathcal{N}_b} z_{ijb'}, \qquad c_{ijb}^{\mathrm{spat}} = z_{ijb} \,\mathrm{mean}_{(i',j') \in \mathcal{N}_{ij}} z_{i'j'b}, \tag{29}$$

$$e_{ij} = d^{-1} \sum_b z_{ijb}^2, \qquad \bar{e} = (n_1 n_2)^{-1} \sum_{ij} e_{ij}. \tag{30}$$

The feature vector in Eq. (8) is $(c_{ijb}^{\mathrm{spec}}, c_{ijb}^{\mathrm{spat}}, e_{ij} - \bar{e}, \bar{e})$. Spectral neighbors are valid bands at offsets $\{-2, -1, 1, 2\}$, and spatial neighbors are the eight-connected pixels. Valid-neighbor averaging handles the image and spectral boundaries. The four coefficients are shared by the conditioning updates and the terminal estimate. Base thresholds remain learned positive scalars; conditioning produces the entrywise activation boundaries.

B.3 Direction-weighted local regression

At entry $\omega = (i, j, b)$, a window of integer radius $h \geq 1$ contains valid same-band offsets $(\delta_i, \delta_j) \in \{-h, \dots, h\}^2 \setminus \{(0, 0)\}$. Define $u = (\delta_i / h, \delta_j / h)$, $v_u = (1, u^\top)^\top$, and $r_u = R_{i+\delta_i, j+\delta_j, b}$. Central

differences of $\mathcal{G}$, with replicated boundary values, give its spatial gradient. Averaging $\nabla G \nabla G^\top$ over valid pixels in a 3×3 window gives J_ω. With $\beta_\omega^{(0)} = 0$, perform $M \geq 1$ updates:

$$w_{\omega u}^{(m)} = \frac{a_{\omega u}}{1 + [(r_u - v_u^\top \beta_\omega^{(m)})/\tau_T]^2}, \qquad m = 0, \ldots, M-1, \tag{31}$$

$$\beta_\omega^{(m+1)} = \left(\sum_{u \in \mathcal{N}_\omega} w_{\omega u}^{(m)} v_u v_u^\top \right)^{-1} \sum_{u \in \mathcal{N}_\omega} w_{\omega u}^{(m)} v_u r_u, \tag{32}$$

$$\widehat{D}_\omega = g\, \beta_{\omega,0}^{(M)}. \tag{33}$$

Each update solves a 3×3 system for the intercept and two slopes. Only observed neighbors enter the fit; padding has zero weight. Positive weights and a spanning local design give a unique solution. Given the guide, these local solves are parallel across pixels and bands. Thresholds and the provisional candidate set are held fixed during this terminal estimation.

C Experimental details

Implementation. Experiments use PyTorch 2.5.1 with CUDA 12.1 on Ubuntu 24.04 and one NVIDIA RTX A6000 GPU with 48 GB memory. ST-HalfNet restores each test cube at its original spatial resolution; the remaining training and evaluation settings are given below.

C.1 Datasets and metrics

CAVE uses 20 training, four validation, and eight test cubes, each of size $512 \times 512 \times 31$. ICVL uses 162/20/20 cubes and retains the native image sizes. Acquisition groups are separated by the recorded scene grouping. The data loader divides a scene by its 99.9th percentile before corruption; values above one are retained. All methods receive the same normalized target and corrupted input. Representation learning uses clean training references, and validation selects fitted parameters. Test cubes do not enter either training objective.

For a cube containing n elements, $\mathrm{PSNR} = -10 \log_{10}(\|\widehat{\mathcal{X}} - \mathcal{X}^\star\|_{\mathrm{F}}^2/n)$. SSIM uses an 11×11 Gaussian window with standard deviation 1.5 and is averaged over bands. Dataset means weight cubes equally. A mean PSNR is therefore the average of individual image PSNRs, not the PSNR obtained after averaging their MSEs. Each baseline uses the output definition and intensity mapping specified below.

C.2 Corruption generation

For each entry, let B_ω be Bernoulli with mean p, C_ω Bernoulli with mean $1/2$, U_ω uniform on $[0, 1]$, and Z_ω standard normal. Independent draws give

$$\text{salt-and-pepper:} \quad \mathcal{Y} = (1 - B) \odot \mathcal{X}^\star + B \odot C, \tag{34}$$

$$\text{random-valued:} \quad \mathcal{Y} = (1 - B) \odot \mathcal{X}^\star + B \odot U, \tag{35}$$

$$\text{sparse Gaussian:} \quad \mathcal{Y} = \mathcal{X}^\star + (\sigma/255) B \odot Z, \quad Z_\omega \sim \mathcal{N}(0, 1). \tag{36}$$

Here σ is the Gaussian standard deviation on the 0–255 scale; the normalized standard deviation is $\sigma/255$. The mask selects individual tensor entries indexed by spatial position and spectral band. Random-valued noise replaces, rather than adds to, those entries. Gaussian noise is added only on its mask. All methods and ablations receive the same corrupted observation for each image and condition. Nested density masks and common standardized draws support density and amplitude comparisons.

The reported grid contains salt-and-pepper and sparse Gaussian at 10%, 30%, and 50%, random-valued replacement at 10% and 30%, and Gaussian amplitudes $\sigma \in \{30, 50, 70\}$ at 30%.

C.3 Representation and recovery training

The dataset models and trained ablations use total rank $R = 512$. Twenty CAVE or five ICVL clean references are selected from distinct acquisition groups first. The initial spectral eigenbasis is

followed by at most twelve transform updates. The gradient of the transform parameters is clipped to norm 5. Armijo backtracking starts at step size 10, halves the step at most six times, and uses sufficient-decrease constant 10^{-4}. The selected iterate minimizes relative approximation error on the validation set. Fitting stops on a rejected step or after four iterations without validation improvement.

Table 5: Model and training settings.

Quantity	Setting
Rank budget / factor updates	$R = 512$ / $K = 24$
Parameter sharing / conditioning onset	$T = 12$, $j(k) = \lfloor k/2 \rfloor$ / $k_c = 12$
Local regression	5×5 window; $M = 3$ reweighting iterations; $\kappa = 4$
Input clipping for factor initialization	$c = .8$
Gram damping	$\gamma = .01$, $\epsilon = 10^{-6}$
Initial steps / threshold sequence	.45 / linearly spaced from .30 to .02
Threshold floor / maximum step	.002 / 1
Recovery training corruption	Salt-and-pepper, $p \sim \mathrm{Uniform}[0.05, 0.30]$
Effective batch / optimizer	Two complete images / Adam
Reference / image objective updates	160 / 40

The shared representation remains fixed during both recovery phases. The first objective is

$$\mathcal{L}_{\mathrm{ref}} = \mathbb{E}_{i,p,Y} \log\left(\frac{\|\mathcal{L}_K(\mathcal{Y}_{i,p}) - \mathcal{X}_i\|_{\mathrm{F}}^2}{n_i} \right), \tag{37}$$

and the second objective replaces $\mathcal{L}_K$ by $\widehat{\mathcal{X}}$:

$$\mathcal{L}_{\mathrm{image}} = \mathbb{E}_{i,p,Y} \log\left(\frac{\|\widehat{\mathcal{X}}(\mathcal{Y}_{i,p}) - \mathcal{X}_i\|_{\mathrm{F}}^2}{n_i} \right). \tag{38}$$

The expectations denote scene and corruption sampling from the declared training distributions. The implementation averages per-image log MSE within each effective batch. Image order is shuffled by seeded passes. Each training image draws its corruption rate independently from $\mathrm{Uniform}[0.05, 0.30]$. The reference phase uses learning rate .005 for step, threshold, and conditional parameters. The final-image phase uses .001 for steps and thresholds, .005 for conditional coefficients, and .01 for local gain. Gradients are clipped to norm 5; g starts at one. CAVE batches two images directly; ICVL accumulates gradients from two single-image microbatches before one optimizer update.

Validation uses fixed 10%, 30%, and 50% observations, with equal weight for each scene and rate. CAVE validates every 20 updates and ICVL every 40. Each phase retains its best validation iterate, including its starting point. The reference phase selects by reference PSNR and the final-image phase by restored-image PSNR. The final parameters are fixed during all reported test conditions.

C.4 Comparison methods

The four optimization methods return their estimated low-rank component. Matrix PCP (Candès et al., 2011) operates on the $(HW) \times d$ unfolding with $\lambda = 1/\sqrt{\max(HW, d)}$, at most 200 augmented-Lagrangian iterations, and tolerance 10^{-6}. DFT-TRPCA (Lu et al., 2020) uses the Fourier tensor nuclear norm, $\lambda = 1/\sqrt{d \max(H, W)}$, at most 500 ADMM iterations, and tolerance 10^{-5}. Its spectral shrinkage is evaluated on the GPU with Fourier conjugacy. EAPT-DCT (Qiu et al., 2022) uses the released HSI rank pattern $5/3/1/0$, $\mu = 500$, $\gamma = .85$, and 24/18 iterations for CAVE/ICVL, chosen using validation. The spatial boundaries scale with image width. The non-learned t-SVD ScaledGD (Wu, 2026) uses slice rank 16/32 for CAVE/ICVL, 60 simultaneous updates, step .5, and Soft threshold $.5(.95)^{k+1}$ from initialization threshold .5.

We use the author architectures for Tucker unfolding (Dong et al., 2023a), HSDT-M (Lai et al., 2023), ILRNet (Ye et al., 2024), and MP-HSIR (Wu et al., 2025). Tucker uses rank $(192, 192, 8)$ and 48 updates. HSDT-M contains 521,158 parameters; its full-image prediction uses 512^2 tiles with overlap 64. ILRNet performs complete-cube inference without output clipping. MP-HSIR uses 31 bands, width 64, and impulse prompt 1. The prepared image intensities are used for training and inference, and the network returns its restored image in those units.

Tucker and HSDT-M are fitted for this task. On CAVE, ILRNet and MP-HSIR start from author checkpoints and are adapted on the training split. On ICVL, both methods use the 162/20/20 split; MP-HSIR starts from random weights with fixed CLIP text features.

The learned baselines use salt-and-pepper corruption rates sampled uniformly from 5% to 30%. HSDT and Tucker select checkpoints at 20% corruption; ILRNet and MP-HSIR use 10% and 30%. The ST-HalfNet checkpoints use 10%, 30%, and 50% as specified above. HSDT uses 64^2 patches, batch 8, and an 8,000-update initial fit followed by 2,000-update calibration candidates. ILRNet uses 64^2 patches, batch 8, and Adam at initial rate 10^{-4}. CAVE uses ten adaptation epochs with the author multiscale patches; ICVL uses 4,000 updates and 64 sampled patches per training scene. MP-HSIR uses 64^2 patches, batch 32, AdamW at 10^{-4}, 100 warmup updates, and cosine decay to 10^{-6} over 1,000 CAVE adaptation updates. Its ICVL fit uses 4,000 updates at initial rate 2×10^{-4}, 400 warmup updates, and validation every 1,000 updates. MP-HSIR minimizes L1 error between its raw restored image and the prepared clean target; HSDT and ILRNet use clean-image MSE. Tucker uses its self-supervised residual objective, complete CAVE images, and 512^2 ICVL crops. Validation selects weights and solver settings before testing.

D Additional analysis

These controls examine how the spatial singular structure guides transform learning, why observations can carry signal beyond a rank constraint, and how local prediction changes the final correction.

D.1 Controlled representation learning

Each synthetic dataset contains 8/4/4 training/validation/test tensors of size $18 \times 18 \times 6$. Transformed slices have ranks $(1, 1, 2, 2, 3, 3)$, equal Frobenius norm, and disjoint row and column subspaces. Consequently their normalized spectral second moment is $I/6$, while the matrix-slice approximation depends on coordinates. Five datasets use a random orthogonal generating transform; a second family uses the DCT. Representation fitting uses total rank 12.

Recovery selects 5% of entries uniformly without replacement and assigns independent random signs and magnitudes in $[5, 7]$ to them. Eight factor updates share the initial factors, steps, and activation thresholds across Soft and Half controls. No local predictor is used. The known generating transform measures the cost of estimating coordinates from finite training references. NMSE is squared error divided by squared target norm. Table 6 reports the full combinations of representations and thresholding operators.

Table 6: **Controlled sparse recovery.** Mean errors with fixed representations and matched activation thresholds.

Data	Representation / map	L-NMSE	S-NMSE	F1	Active MAE
Random Q	Spectral basis / Soft	0.5473	0.1801	0.2155	1.6211
	Spectral basis / Half	0.3883	0.1590	0.2534	0.7915
	Learned / Soft	0.1907	0.0669	0.4520	1.1965
	Learned / Half	0.0919	0.0287	0.6498	0.3365
DCT	Generating DCT / Soft	0.0836	0.0415	0.7576	1.0654
	Generating DCT / Half	0.0096	0.0029	0.9891	0.1757
	Learned / Soft	0.1019	0.0422	0.6606	1.0591
	Learned / Half	0.0286	0.0065	0.8800	0.2019

The isotropic spectral second moment leaves the spectral eigenbasis without a preferred orientation, whereas the spatial singular values still distinguish representations. On the random-transform tensors, learning the representation lowers Half's reference NMSE from 0.3883 to 0.0919 and improves support F1 from 0.2534 to 0.6498. The objective therefore captures structure that spectral energy alone does not identify. Within each representation, Half also reduces sparse recovery error relative to Soft at matched activation thresholds.

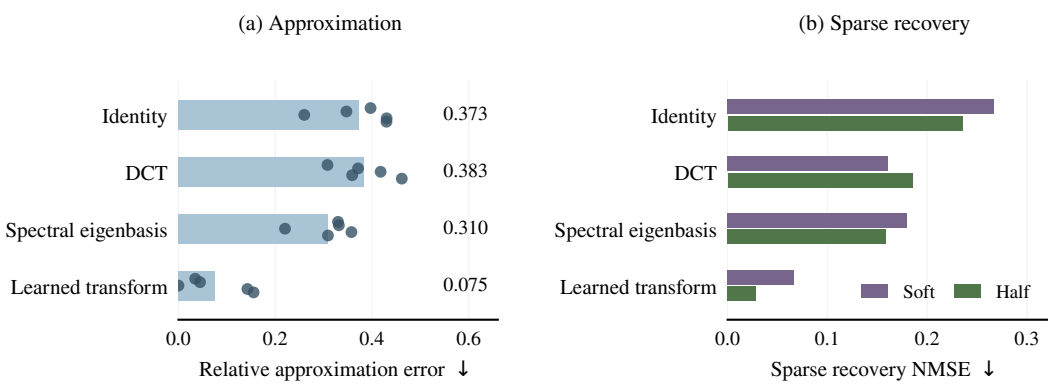


Figure 6: **Transform and thresholding controls.** Left: approximation error; bars are means and points are five independent datasets. Right: sparse recovery error using Soft or Half. All tensors share the same normalized spectral second moment.

D.2 Restoring approximately low-rank tensors

We generate six tensors $\mathcal{X} = \mathcal{L}_0 + \alpha\mathcal{D}$ of size $64 \times 64 \times 8$, with $\alpha \in \{0, 0.03, 0.10\}$. In the generating coordinates, $\mathcal{L}_0$ has slice rank three and $\mathcal{D}$ contains higher spatial frequencies in orthogonal row and column subspaces. Both components have unit RMS. We corrupt 10% of tensor entries and apply the same 24 Half updates to every pair.

Each run produces a low-rank estimate L and a corruption estimate S. We compare the two restored outputs, L and $Y - S$, using these same estimates and no local residual predictor. Table 7 reports $\|\widehat{\mathcal{X}} - \mathcal{X}\|_{\mathrm{F}}^2 / \|\mathcal{X}\|_{\mathrm{F}}^2$. The nonzero singular values of each unnormalized $\mathcal{L}_0$ slice are $(1, 0.6, 0.3)$; each $\mathcal{D}$ slice has twelve singular values decreasing linearly from 1 to 0.3. After separate RMS normalization, every added singular value remains below the third singular value of $\mathcal{L}_0$ for the reported α. Hence $\mathcal{L}_0$ is the best rank-three approximation, and orthogonality gives error $\alpha^2/(1+\alpha^2)$. At $\alpha = 0$, the low-rank output is more accurate; with added signal, subtracting the corruption preserves information excluded by the rank constraint.

Table 7: **Low-rank output or sparse correction.** Mean normalized squared error over six tensors; bold marks the more accurate output. Both outputs use the same factors and sparse estimate.

Added signal α	Best rank-three approximation	Return L	Return $Y - S$
0.00	0	$\mathbf{2.40 \times 10^{-5}}$	5.55×10^{-5}
0.03	8.99×10^{-4}	9.34×10^{-4}	$\mathbf{1.65 \times 10^{-4}}$
0.10	9.90×10^{-3}	1.01×10^{-2}	$\mathbf{1.47 \times 10^{-3}}$

D.3 Residual prediction

We compare the approximation error of the shared representation with the restoration error of ST-HalfNet. For each clean test image, truncated SVDs of its transformed slices give the closest tensor in $\mathcal{M}(Q, \mathbf{r})$. This quantity measures what the chosen representation can express. The network estimates $\mathcal{L}_K$ from the corrupted observation.

Table 3 compares two restoration outputs from the same network. Residual thresholding returns $Y - \widetilde{S}$ by setting $\widehat{\mathcal{D}} = 0$; ST-HalfNet predicts the signal residual before estimating the final corruption. Both outputs use the same reference, thresholds, candidate support, and learned parameters. The comparison isolates the effect of local prediction from the quality of the reference estimate.

Error contributions. Restoration MSE is the sum of contributions from falsely modified uncorrupted entries, corrected corrupted entries, and missed corruption. Each contribution is divided by

the number of elements in the complete image. Table 3(b) groups these errors into uncorrupted and corrupted entries, averages over scenes, and compares the outputs with and without residual prediction using the same trained model.

Figure 4 compares residual thresholding, $\widehat{X}^{\mathrm{thr}} = Y - H_{t_K}(Y - L_K)$, with the full estimate $\widehat{X}^{\mathrm{ST}}$. The signed error map is

$$\Delta(i,j) = \frac{1}{31} \sum_{b} \left[(\widehat{X}^{\mathrm{thr}}_{ijb} - X^{\star}_{ijb})^2 - (\widehat{X}^{\mathrm{ST}}_{ijb} - X^{\star}_{ijb})^2 \right].$$

Green marks lower error after residual prediction; magenta marks higher error. Each row uses a symmetric linear scale at the 99.5th percentile of $|\Delta|$. The displayed windows are 64×64 pixels. Table 3(b) separately measures error changes on uncorrupted and corrupted entries over the complete test sets.

D.4 Ablations

The variants in Table 4 share the dataset split, rank budget $R = 512$, 24 factor updates, and 160/40 reference/image training updates. Restoration parameters are fitted separately and selected by validation. The fixed representation uses an orthogonal DCT and uniform ranks: 17 for 16 slices and 16 for the remaining 15.

D.5 Rank allocation and parameter response

Allocation of representational capacity. Figure 7 orders the 31 transformed slices by assigned rank and accumulates their share of the total budget. The four largest ranks hold 71.7% of the CAVE budget and 91.4% of the ICVL budget. This concentration reflects the unequal gains in Eq. (5): a small number of learned spectral coordinates retain most of the rank budget. Uniform allocation spreads the same 512 units nearly equally over all slices.

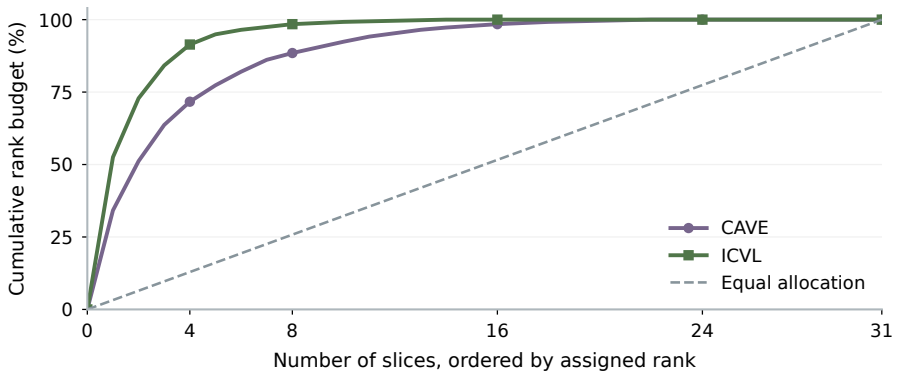


Figure 7: **Distribution of the learned rank budget.** Slices are sorted by assigned rank. Curves show the cumulative fraction of the total budget.

Approximation over a range of budgets. Figure 5 uses the transform learned at $R = 512$ and allocates ranks from clean training references for $R \in \{256, 384, 512, 1024, 1536\}$. Fixed DCT uses uniform ranks. The learned family has lower test approximation error throughout this range, showing that its efficient use of rank extends beyond the budget used to fit the transform.

Direction strength. Figure 8 varies $\kappa \in \{0, 1, 2, 4, 8\}$ on the CAVE and ICVL validation sets at 10% and 30% salt-and-pepper corruption, using the same observations and fitted model. Directional weighting improves over uniform weights in all four comparisons. At the model setting $\kappa = 4$, the gains range from 0.49 to 1.09 dB, supporting the use of local orientation in predicting the residual signal.

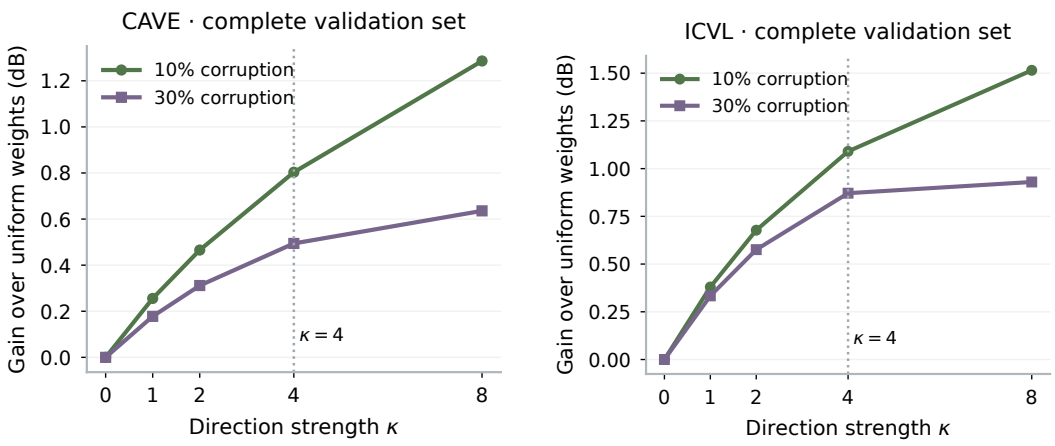


Figure 8: **Sensitivity to direction strength.** Mean validation PSNR gain over uniform weights. All other quantities are fixed; the dotted line marks $\kappa = 4$.

Restoration over iterations. Figure 9 follows the reference and the output of direct residual thresholding through the 24 factor updates at 30% corruption. Their different trajectories distinguish reference fitting from sparse correction. The final local prediction uses the reference after update 24, improving the restored image without a further factor update.

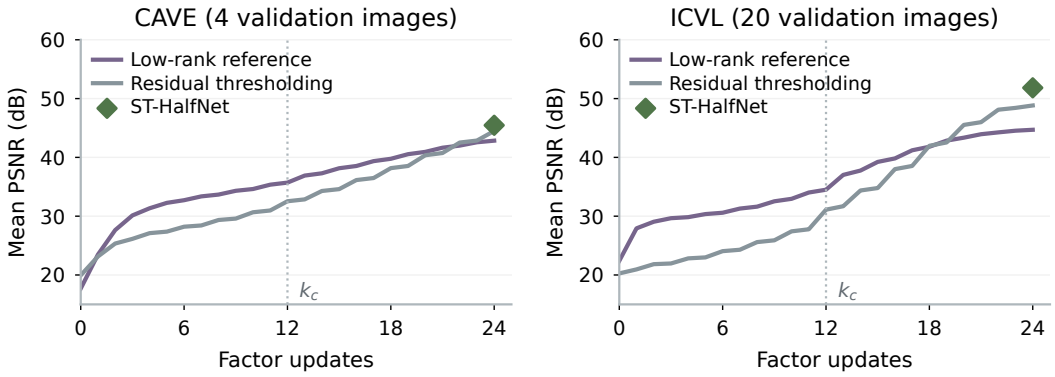


Figure 9: **Restoration over iterations.** Mean validation PSNR at 30% corruption. Diamonds mark the final restoration; dotted lines mark the start of residual conditioning.

E Additional restoration results

E.1 PSNR and SSIM

Tables 8–10 give mean PSNR and SSIM on the complete test sets, complementing the PSNR comparisons in Section 6. Bold and underlining identify the two highest values.

The SSIM results reinforce the PSNR comparisons: ST-HalfNet leads at all three salt-and-pepper densities on both datasets. At 50% corruption, its mean SSIM is 0.9280 on CAVE and 0.9835 on ICVL, so the gain extends to spatial structure.

Table 8: **Salt-and-pepper restoration.** Mean PSNR (dB) and SSIM.

	10%		30%		50%	
Method	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
CAVE						
Matrix PCP	30.27	0.9681	25.11	0.9035	19.71	0.7807
DFT-TRPCA	39.79	0.9907	37.34	0.9840	32.89	0.8791
EAPT-DCT	42.81	0.9781	38.32	0.9515	31.91	0.8267
t-SVD ScaledGD	32.64	0.9020	31.52	0.8572	26.95	0.6755
Tucker unfolding	45.32	0.9883	39.95	0.9773	29.42	0.8533
HSDT-M	45.81	0.9895	40.77	0.9616	31.97	0.7927
ILRNet	45.56	0.9894	41.25	0.9753	29.77	0.6711
MP-HSIR	43.01	0.9875	40.20	0.9787	34.23	0.8525
ST-HALFNET	**50.19**	**0.9952**	**45.13**	**0.9844**	**37.07**	**0.9280**
ICVL						
Matrix PCP	39.50	0.9915	32.73	0.9405	23.58	0.7074
DFT-TRPCA	42.06	0.9940	39.66	0.9890	32.30	0.7640
EAPT-DCT	29.23	0.9507	28.28	0.9195	25.14	0.7122
t-SVD ScaledGD	30.27	0.8812	29.78	0.8316	27.11	0.6176
Tucker unfolding	40.93	0.9746	40.52	0.9737	34.78	0.9313
HSDT-M	45.19	0.9867	40.75	0.9567	32.14	0.8080
ILRNet	35.37	0.9545	30.19	0.9244	22.73	0.8253
MP-HSIR	30.48	0.7966	27.15	0.5620	21.54	0.3836
ST-HALFNET	**55.81**	**0.9989**	**50.67**	**0.9960**	**44.30**	**0.9835**

Random-valued corruption uses the same sparse support draws as salt-and-pepper corruption, but replaces selected entries by independent uniform values rather than zero or one. The trained weights remain fixed.

Table 9: **Random-valued corruption.** Mean PSNR (dB) and SSIM with fixed weights.

	CAVE				ICVL			
	10%		30%		10%		30%	
Method	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Matrix PCP	30.46	0.9707	25.98	0.8803	38.98	0.9906	31.54	0.9353
DFT-TRPCA	39.78	0.9909	36.66	0.9710	41.76	0.9939	38.09	0.9851
EAPT-DCT	39.38	0.9709	37.53	0.9562	30.25	0.9426	24.38	0.8863
t-SVD ScaledGD	32.73	0.8972	30.42	0.7487	30.28	0.8814	29.34	0.8210
Tucker unfolding	45.11	0.9883	30.83	0.8774	40.92	0.9747	40.14	0.9728
HSDT-M	29.89	0.6021	21.45	0.2845	34.33	0.8715	25.08	0.6279
ILRNet	35.87	0.8494	24.01	0.5278	33.20	0.9565	25.93	0.9073
MP-HSIR	32.89	0.7673	21.71	0.4475	29.65	0.8645	23.91	0.7062
ST-HALFNET	**50.93**	**0.9965**	**44.58**	**0.9849**	**54.90**	**0.9985**	**49.40**	**0.9945**

At fixed 30% density, increasing Gaussian amplitude from $\sigma = 30$ to 70 raises ST-HalfNet's mean PSNR from 45.71 to 46.23 dB on CAVE and from 48.62 to 49.71 dB on ICVL. This is consistent with the activation condition in Appendix A.4: larger nonzero corruption can be easier to distinguish from residual signal.

Table 10: **Sparse Gaussian restoration.** Mean PSNR (dB) and SSIM over density and amplitude.

	Density ($\sigma = 50/255$)						Amplitude ($p = 30\%$)			
	10%		30%		50%		30/255		70/255	
Method	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
CAVE										
Matrix PCP	31.62	0.9757	29.46	0.9462	27.39	0.8936	30.46	0.9564	28.78	0.9395
DFT-TRPCA	40.19	0.9910	38.69	0.9859	35.31	0.9020	39.14	**0.9868**	38.42	0.9855
EAPT-DCT	43.44	0.9798	40.82	0.9608	36.71	0.8923	40.85	0.9546	40.85	0.9653
t-SVD ScaledGD	32.92	0.8999	32.40	0.8640	31.05	0.7648	32.58	0.8724	32.28	0.8594
Tucker unfolding	45.42	0.9883	43.04	0.9823	36.31	0.9249	44.24	0.9852	42.12	0.9794
HSDT-M	32.34	0.7122	27.29	0.4629	24.67	0.3260	30.90	0.6320	24.47	0.3353
ILRNet	41.78	0.9599	34.67	0.8081	30.93	0.6914	38.65	0.9047	31.96	0.7148
MP-HSIR	39.13	0.9362	35.10	0.8425	32.59	0.7616	36.28	0.8752	34.75	0.8307
ST-HALFNET	**50.64**	**0.9961**	**45.98**	**0.9882**	**40.73**	**0.9566**	**45.71**	0.9867	**46.23**	**0.9892**
ICVL										
Matrix PCP	40.16	0.9930	36.39	0.9729	32.58	0.9187	37.26	0.9783	35.89	0.9694
DFT-TRPCA	42.33	0.9941	40.68	0.9900	36.58	0.9167	41.05	0.9904	40.47	0.9897
EAPT-DCT	45.80	0.9827	40.23	0.9305	34.83	0.8020	39.96	0.9259	40.76	0.9400
t-SVD ScaledGD	30.33	0.8836	30.13	0.8523	29.49	0.7705	30.23	0.8609	30.07	0.8475
Tucker unfolding	40.99	0.9747	40.93	0.9740	40.35	0.9715	41.04	0.9742	40.87	0.9740
HSDT-M	37.38	0.9087	30.44	0.7537	25.68	0.6243	34.60	0.8522	27.01	0.6435
ILRNet	31.47	0.9463	31.44	0.9171	29.67	0.8806	31.51	0.9451	29.28	0.8659
MP-HSIR	31.41	0.9332	30.90	0.8926	29.85	0.8467	31.39	0.9319	29.31	0.8241
ST-HALFNET	**54.25**	**0.9981**	**49.29**	**0.9936**	**44.83**	**0.9822**	**48.62**	**0.9922**	**49.71**	**0.9944**

Figure 10 shows the spectral distribution of reconstruction error.

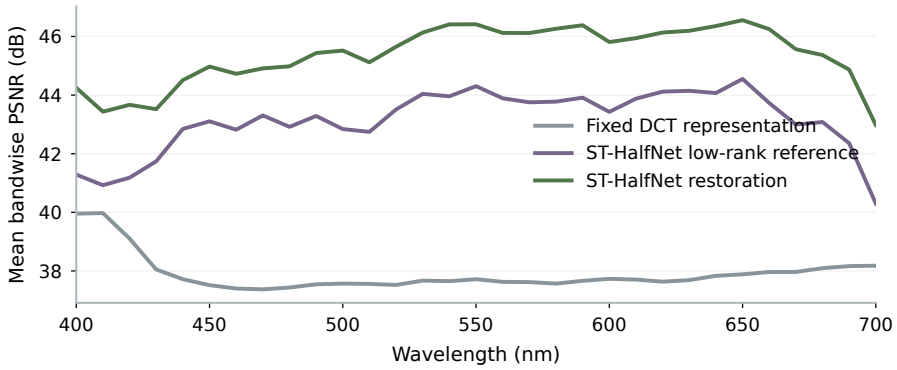


Figure 10: **Spectral reconstruction accuracy.** Mean per-band PSNR on all eight CAVE images at 30% salt-and-pepper corruption.

E.2 SPATIAL COMPARISONS

Figures 11–13 show six additional test scenes. Methods share 128×128 regions, two 40×40 detail windows, and RGB rendering at 650/550/450 nm with gamma 2.2. Each score uses the complete 31-band image.

Spatial reconstruction error. Figure 14 shows the per-pixel RMSE over all 31 bands in the same regions as Figure 3. One linear scale is shared by all methods, exposing errors at contours and thin branches as well as isolated remaining corruption.

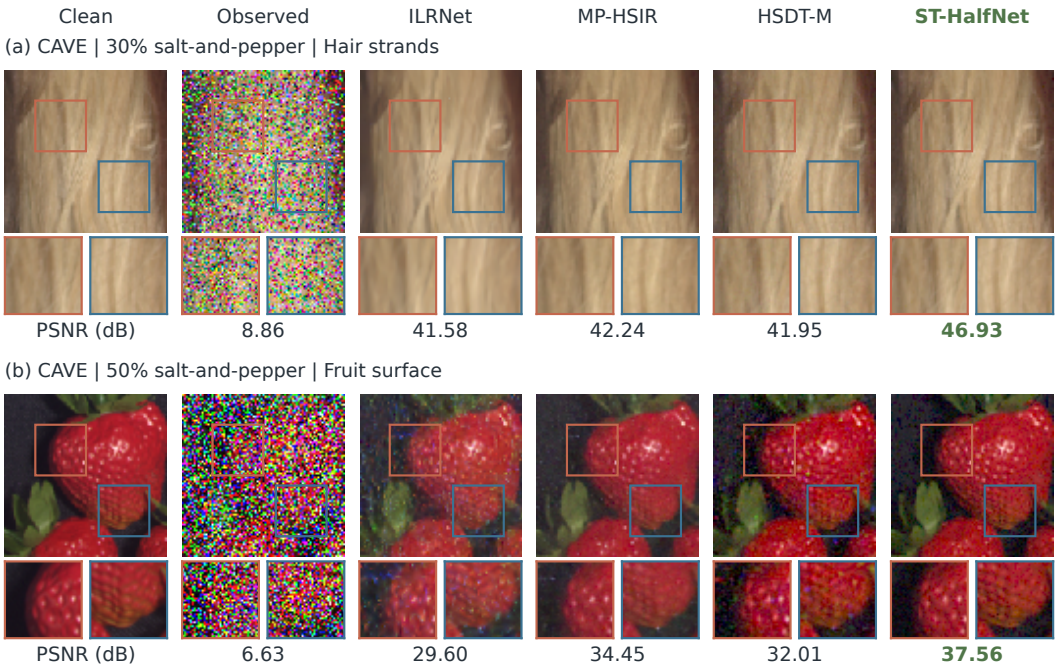


Figure 11: **Texture restoration.** Hair strands and fruit surfaces. Scores are full-image PSNR (dB).

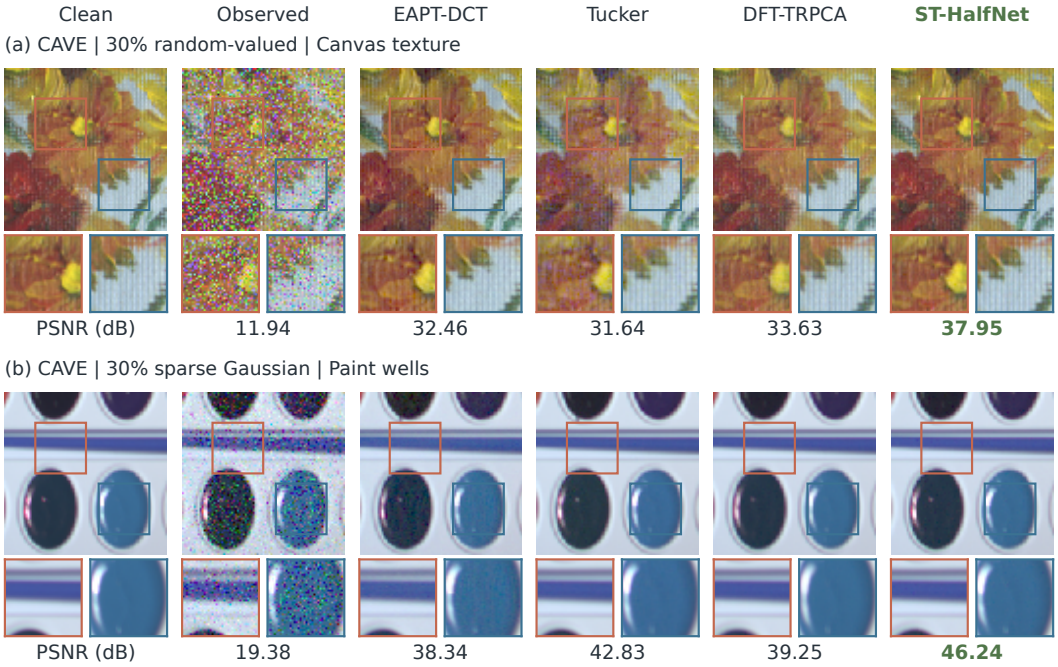


Figure 12: **Painting and paint wells.** Random-valued and sparse Gaussian corruption. Scores are full-image PSNR (dB).

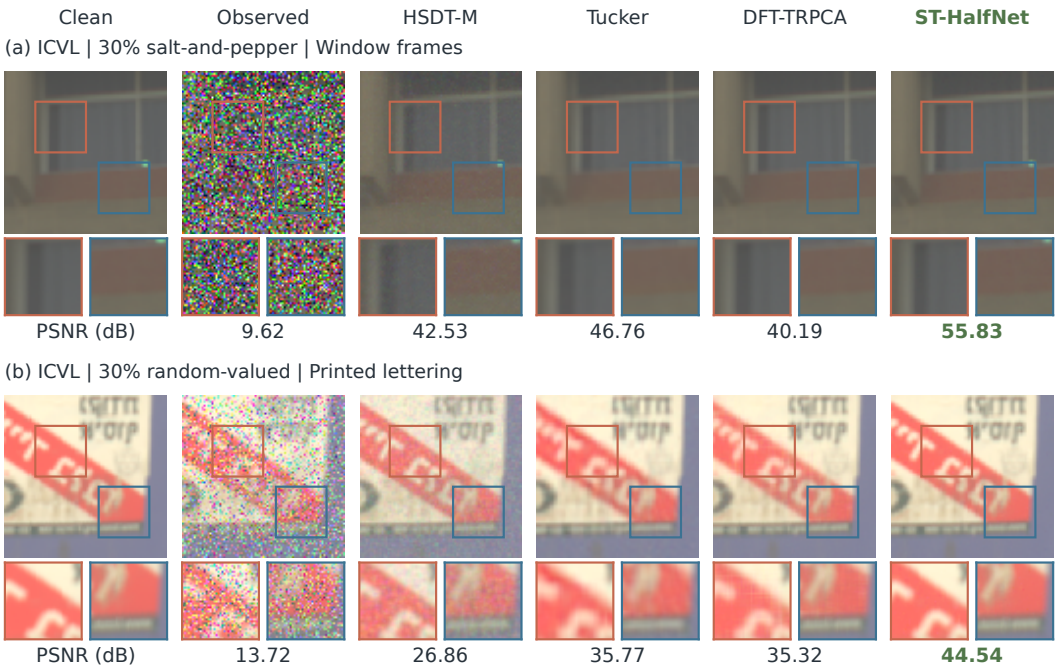


Figure 13: **Edges and lettering.** Two ICVL scenes; full-image PSNR (dB).

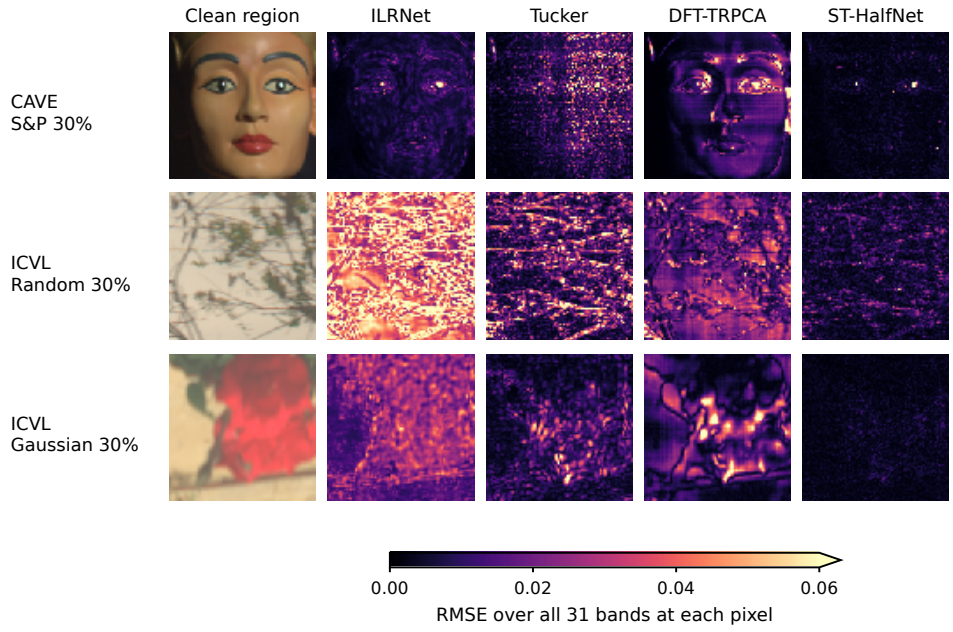


Figure 14: **Spatial reconstruction error.** Brighter values indicate larger RMSE; the common scale is capped at 0.06.