On the Sparsity-Storage-Accuracy Tradeoff in Parsimoniously Activated Dictionary Learning

Zihui Zhao[1] **Yang Li**[2*]
[1]Institute of Data and Information, Tsinghua University
[2]School of Artificial Intelligence, Chinese University of Hong Kong (Shenzhen)

Abstract

Dictionary learning has long been studied from both optimization and probabilistic perspectives. While formulations with element-wise sparsity regularization (e.g., L1-based sparse coding) admit well-established probabilistic interpretations, many structured variants that impose global constraints lack a clear and tractable generative view. In this paper, we revisit a class of practically effective yet theoretically under-explored dictionary learning methods that impose a simple global regularization on the number of activated dictionary atoms, which we term parsimoniously activated dictionary learning (PADL). We show that PADL admits a formulation that is approximately equivalent to maximum a posteriori estimation under a structured generative model, with auxiliary latent variables that govern global activation patterns. This formulation allows us to derive generalization guaranties that are difficult to obtain under the original formulation. More importantly, it yields an analytical characterization of the tradeoff between sparsity, storage cost, and reconstruction accuracy, enabling data-driven estimation of optimal hyperparameters. Based on this connection, we develop an efficient and interpretable PADL algorithm that eliminates manual hyperparameter tuning, achieving improved reconstruction performance under comparable sparsity levels on visual benchmarks. We further demonstrate its practical utility in accelerating inference for vision-language models.

1 Introduction

Dictionary learning has long been a fundamental tool in signal processing and machine learning, aiming to find sparse representations of data through an optimization framework [29]. From the optimization perspective, classical sparse coding-based methods, such as K-SVD, formulate dictionary and code estimation as regularized reconstruction problems, yielding scalable algorithms with direct control over reconstruction error and coefficient sparsity [29, 2, 27]. From the probabilistic perspective, dictionary learning can be described by generative models over observations and latent coefficients, providing generalization analysis and uncertainty estimates [30, 50, 10, 7, 33]. These two views are closely connected for element-wise sparsity constraints. For example, maximum a posteriori (MAP) estimation with Laplace coefficient priors yields the classic ℓ_1 sparse-coding formulation [29, 13], which makes classical sparse dictionary learning both computationally practical and statistically interpretable.

However, this optimization–probabilistic bridge is much less developed for structured dictionary-learning objectives that impose global constraints on atom usage. These global constraints are particularly important [12], as they naturally introduce a form of model selection, leading to improved robustness and statistical efficiency. Group-sparse methods encourage block-level selection. For example, simultaneous sparse approximation [42] enforces that multiple sparse signals share the same

*Thanks to National Natural Science Foundation of China (62371270).

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

support; activation-constrained formulations [19, 38] impose constraints on the number of activated atoms; and recent dictionary-learning variants are increasingly used for scalable representation learning [42, 19, 38]. Such global constraints inherently call for hierarchical modeling. While hierarchical probabilistic approaches to dictionary learning, such as Beta–Bernoulli [30, 50] and Indian Buffet Process models [10], have been studied, they are generally not directly connected to the optimization formulations used in practice.

Motivated by this gap, we revisit a class of practically effective yet theoretically under-explored dictionary learning methods that impose a simple global regularization on the number of activated dictionary atoms [19, 38], which we term parsimoniously activated dictionary learning (PADL). Its objective can be written in the unified form as:

$$\min_{D,R} \|X - DR\|_F^2 + \lambda_1 \|R\|_1 + \lambda_2 \sum_{k=1}^{K} \|r_k\|_\infty, \tag{1}$$

where X is the data matrix, D is the dictionary, R is the coefficient matrix, and r_k is the k-th coefficient row. The $\|\cdot\|_F^2$ term controls the reconstruction error and the $\|\cdot\|_1$ term controls sample-level representation sparsity. The row-wise ℓ_∞ term constrains global atom activation by penalizing whether a dictionary atom is used by any sample. This formulation is simple, easy to optimize with alternating optimization, and highly effective in fields including trajectory generation and DNA compression [37, 38, 1]. However, it requires hyperparameter grid search to select optimal λ_1 and λ_2, which hinders practical stability and efficiency. Our goal is to provide a generative probabilistic view of this formulation that enables principled optimal hyperparameter computation by characterizing the sparsity–storage–accuracy tradeoff.

We derive this PADL formulation from a structured auxiliary-latent z formulation through MAP estimation with $P(X, R, z; D)$. The coefficient prior $p(R)$ induces per-sample sparsity, while the latent variable $P(z \mid r)$ models whether an atom is activated by the dataset.

This dual formulation retains the computational simplicity of the optimization view while enabling several analytical results from its generative view. In particular, we derive a sufficient condition under which non-essential atoms are suppressed and establish a high-probability reconstruction-error bound. These results make explicit how the activation threshold and regularization weights govern the tradeoff among sparsity, storage, and reconstruction accuracy.

Based on this formulation, we develop an efficient PADL algorithm with analytical data-driven hyperparameter estimation, reducing reliance on exhaustive hyperparameter search. With a smaller activated dictionary size, our method achieves better reconstruction quality and significantly reduces the time required for parameter search. We evaluate the framework on both classical image reconstruction benchmarks and Vision–Language Model (VLM) inference acceleration via token compression. On image reconstruction tasks, our method achieves superior reconstruction performance and empirically aligns with the theoretical predictions on optimal hyperparameter selection and dictionary activation thresholds. On VLM inference acceleration tasks, our method consistently attains higher token compression ratios while preserving downstream task performance.

Our contributions can be summarized as follows:

- We revisit PADL, a storage-aware dictionary-learning framework that regularizes cross-sample atom activation and directly targets the sparsity–storage–accuracy tradeoff.
- We introduce a structured MAP-induced interpretation of PADL, linking the row-wise activation penalty to an atom-level auxiliary latent activation model.
- We theoretically characterize the sparsity–storage–accuracy tradeoff via guarantees on atom suppression and reconstruction error, leading to an efficient hyperparameter estimation rule.
- We validate the framework on controlled image reconstruction benchmarks and further demonstrate its applicability through image and video VLM compression experiments.

2 Method

2.1 Problem Formulation

Given a data set $X = \{X_i\}_{i=1}^N$, where each $X_i \in \mathbb{R}^d$ corresponds to the feature vector of a sample, we introduce a dictionary $D = [d_1, \ldots, d_K] \in \mathbb{R}^{d \times K}$. Each sample is generated as

$$X_i = DR_i + \epsilon_i,$$

where $R_i \in [0,1]^{K \times 1}$ and ϵ_i is isotropic Gaussian noise, yielding the likelihood

$$p(X_i \mid R_i; D) \propto \exp\!\big(-\tfrac{1}{2}\|X_i - DR_i\|_2^2\big)\,.$$

Specifically, we assume each coefficient follows an independent Beta distribution,

$$R_{ki} \sim \mathrm{Beta}(1, \beta), \qquad \beta > 1, \tag{2}$$

where $1 \le k \le K$ with density

$$p(R_{ki}) = \beta(1 - R_{ki})^{\beta - 1}, \qquad R_{ki} \in [0,1]. \tag{3}$$

This prior concentrates mass near zero and naturally favors sparse activations. To model the storage cost, we introduce a latent activation variable $z_k \in \{0,1\}$ for each dictionary atom d_k, indicating whether the atom is effectively used across the samples. Let $r_k \in \mathbb{R}^N$ denote the k-th row of R, i.e., the coefficients of atom k across all samples. Conditioned on r_k, we define the activation prior

$$z_k \mid r_k \sim \mathrm{Bernoulli}\left(\frac{1}{1 + \exp\big(-\gamma(\max_i R_{ki} - \delta)\big)}\right), \tag{4}$$

where $\delta \in [0,1]$ denotes an activation threshold and $\gamma > 0$ controls the sharpness of the decision. The quantity $\max_i R_{ki}$ summarizes whether atom k contributes significantly to at least one sample. An atom d_k is considered active only if r_k attains a sufficiently large coefficient for some sample (above the threshold). The threshold δ therefore acts as a direct control on activation, while the Beta prior governs per-sample sparsity. We can therefore define a joint distribution

$$P(X, R, z; D) = \prod_{i=1}^N P(X_i \mid R_i; D) \cdot \prod_{k=1}^K \prod_{i=1}^N p(R_{ki}) \cdot \prod_{k=1}^K P(z_k \mid r_k). \tag{5}$$

2.2 MAP Objective and Algorithm

Under the probabilistic model introduced above, we apply MAP estimation to learn model parameters given data set X. Taking the negative log-posterior yields an energy of the form

$$-\log P(R, z \mid X; D) = \sum_{i=1}^N \frac{1}{2}\|X_i - DR_i\|_2^2 - \sum_{k,i} \log p(R_{ki}) - \sum_k \log P(z_k = 0 \mid r_k) + C, \tag{6}$$

where C is a constant independent of (R, z).

Proposition 2.1 (MAP-Induced Objective)**.** *Under the local approximation of the Beta prior, MAP estimation of $P(X, R, z; D)$ in Eq. 5 leads to the following objective:*

$$\min_{D \in R^{d \times K}, R \in [0,1]^{K \times N}} \quad \|X - DR\|_F^2 + \lambda_1 \|R\|_1 + \lambda_2 \sum_{k=1}^K \phi_\gamma\big(\max_i R_{ki} - \delta\big), \tag{7}$$

where $\phi_\gamma(u) = \log(1 + e^{\gamma u})$, with $\lambda_1 = \beta - 1$ and $\lambda_2 = 1$. For practical optimization, we use a row-wise ℓ_∞ surrogate for this peak-activation cost:

$$\min_{D \in R^{d \times K}, R \in [0,1]^{K \times N}} \quad \|X - DR\|_F^2 + \lambda_1 \|R\|_1 + \lambda_2 \sum_{k=1}^K \|r_k\|_\infty, \tag{8}$$

where $\|r_k\|_\infty = \max_i |R_{ki}|$.

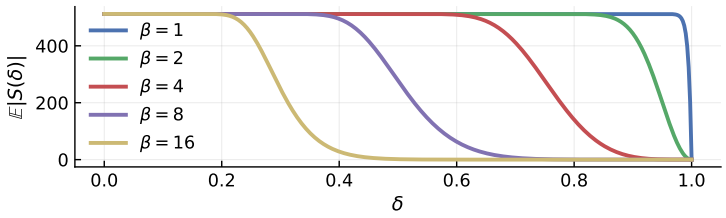


Figure 1: The estimated dictionary activation size as a function of δ under different values of β, estimated on CIFAR-100.

Proof. See Appendix B. □

Intuitively, the Beta prior $p(R_{ki})$ induces an ℓ_1-type penalty near zero that promotes representation sparsity. Numerical diagnostics on real data (Appendix B.5) further confirm that the local approximation error remains small at the observed operating points. Meanwhile, the activation prior $p(z_k = 0|r_k)$ gives rise to a soft barrier on $\max_i R_{ki}$, for which the row-wise ℓ_∞ term serves as a convex surrogate in the practical PADL objective.

Unlike classical sparse coding objectives, which balance only reconstruction and coefficient sparsity, the PADL formulation explicitly models dictionary size as an optimization objective. The activation threshold δ and the associated ℓ_∞ term act as direct control parameters for dictionary size, enabling explicit regulation of the sparsity-storage-accuracy tradeoff.

Proposition 2.1 further connects the hyperparameters λ_1 and λ_2 to the prior parameter β. In the next subsection, we illustrate how the error bound is affected by the hyperparameters and the activation threshold δ.

2.3 Tradeoff Characterization

To characterize the tradeoff between dictionary size and reconstruction error, we first establish the condition when non-essential atoms are suppressed (Proposition 2.2), and then derive the expected activated dictionary size under the Beta model (Proposition 2.3). These results lead to a reconstruction bound in Theorem 2.4 and the threshold-calibration rule in Corollary 2.5.

Proposition 2.2 (Suppression of Non-essential Atoms)**.** *Assume that a subset of atoms S can reconstruct the data with bounded error ϵ. For any additional atom $k \notin S$ with nonzero coefficient row r_k, let $e = X - D_S R_S$ and suppose $|e_i^\top d_k| \le \eta$ for all i. If*

$$\lambda_1 + \lambda_2 \frac{\|r_k\|_\infty}{\|r_k\|_1} \;>\; 2\eta, \tag{9}$$

then setting the coefficient vector r_k to zero strictly decreases the objective. In particular, for atoms whose peak activation exceeds the threshold, $\|r_k\|_\infty \ge \delta$, it is sufficient that

$$\lambda_1 + \lambda_2 \frac{\delta}{\|r_k\|_1} \;>\; 2\eta. \tag{10}$$

Proof. See Appendix C.1 □

This result formalizes the role of the activation penalty: atoms that only exhibit occasional or weak activations are eliminated, even if their coefficients satisfy the ℓ_1 sparsity constraint. Such atoms correspond to spurious patterns that are likely caused by noise. The model therefore favors visual primitives that are consistently useful, yielding a compact and reusable dictionary.

Proposition 2.3 (Expected Activated Dictionary Size)**.** *Assume that the activation coefficients satisfy $R_{ki} \overset{i.i.d.}{\sim} \mathrm{Beta}(1, \beta)$ for $k = 1, \dots, K$ and $i = 1, \dots, N$. Define the active atom set induced by threshold δ as*

$$S(\delta) \;:=\; \{\, k \;:\; \max_i R_{ki} > \delta \,\}.$$

Let $F(r) = 1-(1-r)^\beta$ denote the cumulative distribution function of $\mathrm{Beta}(1,\beta)$*. Then the expected size of the active set is*

$$\mathbb{E}\,|S(\delta)| \;=\; K\big(1-F(\delta)^N\big) \;= K\left[1-\big(1-(1-\delta)^\beta\big)^N\right]. \tag{11}$$

In particular, $\mathbb{E}|S(\delta)|$ is a strictly decreasing and smooth function of δ, with

$$\mathbb{E}|S(0)| = K, \qquad \mathbb{E}|S(1)| = 0.$$

Proof. See Appendix C.2 □

The activated dictionary size $\mathbb{E}|S(\delta)|$ is strictly decreasing in δ, as shown in Figure 1, which indicates that a larger activation threshold decreases the dictionary size by activating fewer atoms.

Theorem 2.4 (High-Probability Reconstruction Bound)**.** *Under the active-span restricted-eigenvalue condition stated in Appendix C.3, the average reconstruction error satisfies, with probability at least $1-\tau$,*

$$\frac{\|e\|_F^2}{N} \;\le\; C(|S|)\left(\lambda_1 + \frac{\lambda_2\delta}{U(N,\beta,\tau)}\right)^2, \tag{12}$$

where $C(|S|)$ grows linearly with $|S|$ and

$$U(N,\beta,\tau) := \frac{N}{1+\beta} + \sqrt{\frac{N}{2}\log\frac{1}{\tau}}$$

grows linearly with N.

Proof. See Appendix C.3 □

Theorem 2.4 shows that the average reconstruction error bound depends on both the active dictionary size $|S|$ and the activation threshold δ. Since $|S|$ is also governed by δ according to Proposition 2.3, the threshold influences the bound through both factors. In particular, as sample size N increases, $U(N,\beta,\tau)$ grows linearly, causing the threshold-dependent term $\lambda_2\delta/U(N,\beta,\tau)$ to decay at a rate of $\mathcal{O}(1/N)$ for fixed δ. These findings further enable a model-based estimation of the activation threshold, as characterized by the following corollary.

Corollary 2.5 (Model-Based Activation Threshold Estimation)**.** *For fixed data size, data distribution, and confidence level, the model-based threshold estimate δ^* satisfies the following interior stationarity condition:*

$$\frac{N\,F(\delta)^{N-1}\,f(\delta)}{1-F(\delta)^N} = \frac{2\lambda_2}{U(N,\beta,\tau)\left(\lambda_1 + \frac{\lambda_2\delta}{U(N,\beta,\tau)}\right)}. \tag{13}$$

Here,

$$F(\delta) := \Pr(R_{ki} \le \delta) = 1-(1-\delta)^\beta, f(\delta) := F'(\delta) = \beta(1-\delta)^{\beta-1}.$$

Proof. See Appendix D.4 □

2.4 Dictionary Learning Algorithm

Using the above theoretical results, we proposed a modified PADL algorithm with analytical hyperparameter estimation from data, as shown in Algorithm 1. This algorithm adopts alternating optimization [27] with projected gradient descent to jointly learn the dictionary and the sparse representation while ensuring convergence, a standard technique in previous literature [43, 3, 6]. To estimate the hyperparameters, we use a one-shot plug-in empirical Bayes procedure (Lines 6–8). After $t_s = 50$ warm-up steps in the main experiments, we fit $\mathrm{Beta}(1,\widehat{\beta})$ to entries of the coefficient matrix R via maximum likelihood estimation and use the fitted parameter to calibrate λ_1, λ_2, and δ^*. The resulting calibration is data-adaptive under the chosen warm-up objective and initialization.

During testing, the dictionary remains fixed, and the hyperparameters and activation threshold are set to their estimated values; only the sparse representation R is updated. Note that, while the hyperparameters could be updated iteratively during alternating optimization to obtain a more precise estimate of the prior distribution, we empirically find that one-shot warm-up estimation is sufficient in our experiments, as shown in Appendix F.5.

Algorithm 1 Parsimoniously Activated Dictionary Learning

Require: Data matrix $\mathbf{X}$; dictionary size K; outer iterations T; inner steps k_r, k_d; warm-up steps t_s
Ensure: Learned dictionary $\mathbf{D} \in \mathbb{R}^{d \times K}$

Use preprocessed data: $\mathbf{X}_{\text{norm}} \leftarrow \mathbf{X}$
Initialize $\mathbf{D} \sim \mathcal{U}[0,1]$, $\mathbf{R} \sim \text{Beta}(1,1)$
Initialize $(\lambda_1, \lambda_2, \delta) \leftarrow (0, 1, 0)$ as warm-up defaults
for $t = 1$ to T **do**
  if $t = t_s$ **then**
    Fit empirical $\mathbf{R}$ with $\text{Beta}(1, \widehat{\beta})$
    Set $\lambda_1 \leftarrow \widehat{\beta} - 1$, $\lambda_2 \leftarrow 1$, and compute δ^* using Eq. 13
    Re-initialize $\mathbf{D} \sim \mathcal{U}[0,1]$, $\mathbf{R} \sim \text{Beta}(1, \widehat{\beta})$
  end if
  for $\ell = 1$ to k_r **do**
    Update $\mathbf{R}$ by solving $\min_{\mathbf{R}} \ \|\mathbf{X}_{\text{norm}} - \mathbf{DR}\|_F^2 + \lambda_1 \|\mathbf{R}\|_1 + \lambda_2 \sum_{k=1}^{K} \|r_k\|_\infty$.
    Project coefficients: $\mathbf{R} \leftarrow \Pi_{[0,1]}(\mathbf{R})$.
  end for
  for $\ell = 1$ to k_d **do**
    Update $\mathbf{D}$ by solving $\min_{\mathbf{D}} \ \|\mathbf{X}_{\text{norm}} - \mathbf{DR}\|_F^2$ s.t. $\|d_k\|_2 = 1, \ \forall k$.
  end for
end for
return $\mathbf{D}$

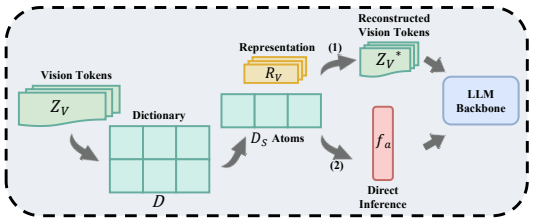


Figure 2: Two implementations on VLMs are considered: reconstructing visual tokens and performing direct inference via a fine-tuned adapter.

2.5 VLM Inference Compression

PADL can also be used as a structured visual compression module for VLM inference. Given visual tokens $\mathbf{Z}_V$ produced by a vision encoder, we learn a visual dictionary $\mathbf{D}_V$ and sparse codes $\mathbf{R}_V$ such that $\mathbf{Z}_V \approx (\mathbf{D}_V \mathbf{R}_V)^T$. The row-wise activation regularizer selects a compact active sub-dictionary $\mathbf{D}_S$, while the sparse codes can be quantized for transmission or storage. As shown in Figure 2, the compressed representation can be used in two ways: reconstructing visual tokens before feeding them to the LLM, or passing the sparse representation through a lightweight adapter for direct inference. The detailed VLM formulation, compression protocol, and adapter setting are provided in Appendix E.

3 Experiment

Our experimental evaluation focuses on validating the theoretical contributions of our framework, particularly the Bayesian interpretation and the parameter selection guidelines. We conduct experiments on CIFAR-100 [20] and SVHN [28] to demonstrate the practical implications of our theoretical framework. Each dataset contains color images of size 32×32. All images are converted to grayscale, and non-overlapping 8×8 patches are extracted for training and evaluation. The dictionary size is fixed at 128 atoms across all experiments. We use 20,000 random patches for training and reserve 2,000 patches for testing and performance evaluation. Unless otherwise specified, the number of activated atoms denotes the number of dictionary atoms satisfying $\max_i |R_{ki}| > \delta^*$ on the evaluation set, using the same thresholding rule across methods.

Table 1: Model-predicted and grid-selected parameters and final reconstruction performance on CIFAR-100. Cost includes both warm-up and training duration. Grid search is conducted on 10^3-configuration grid. See Appendix F.10 for timing details.

Selection	λ_1	λ_2	δ	RMSE	PSNR	SSIM	Activated Atoms	Cost
Model-predicted	1.31	1.00	0.44	0.099	20.56	0.568	19	52.56 s
Grid-selected	1.33	1.01	0.49	0.097	20.71	0.571	19	10.98 h

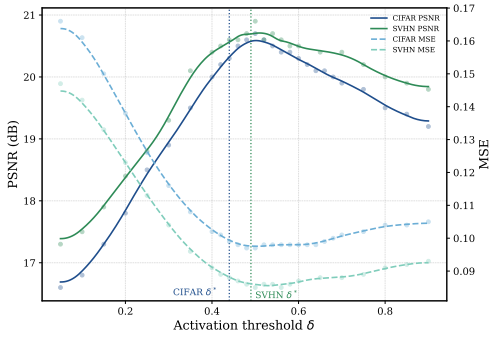


Figure 3: Reconstruction error under varying activation thresholds δ on CIFAR-100 and SVHN; dashed lines indicate the predicted δ^*.

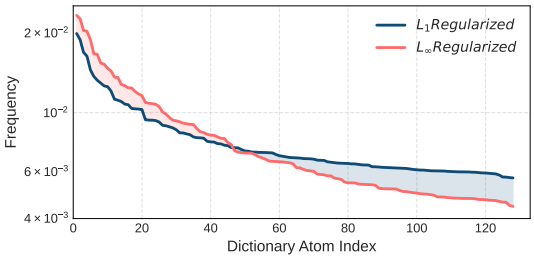


Figure 4: Atom utilization frequency on CIFAR-100 under L_∞- and L_1-regularized dictionaries.

For VLM applications, we conduct experiments with LLaVA-1.5 [23] on VQA tasks for image inputs and Video-LLaVA [22] for video inputs. The dictionary size is fixed at 1024 atoms. For image inputs, our evaluation includes VQAv2 [16], ScienceQA [25], TextVQA [36], POPE hallucination benchmark [21], MME [15] and MMBench [24]. We apply $4\times$ dimensional compression together with $2\times$ quantization. The adapter layer is fine-tuned on the LLaVA-1.5 instruction-tuning dataset [23]. For video inputs, our evaluation is conducted on MSVD-QA [8], MSRVTT-QA [45], TGIF-QA [18] and ActivityNet-QA [46]. We apply $8\times$ dimensional compression together with $2\times$ quantization. The adapter layer is fine-tuned on a dataset provided by Valley [26]. We compare our method with two SOTA token reduction methods: LLaVA-PruMerge [35] and freePruner [44]. The variant without fine-tuning is denoted as **PADL**, while the variant with fine-tuning is denoted as **PADL+**. For image benchmarks, we follow the standard evaluation metrics of each benchmark. For video QA, we report accuracy (%) and GPT-based answer-quality scores assessed using GPT-3.5-Turbo, following prior work.

Experiments were run on Ubuntu 22.04.5 with two Intel Xeon Gold 6348 CPUs, and a single NVIDIA A800 80GB PCIe GPU. The CUDA version was 12.2.

3.1 Hyperparameter Estimation Validation

Table 1 compares the model-predicted and grid-selected parameters and their final reconstruction performance. The predicted parameters achieve nearly identical reconstruction quality and the same activated dictionary size, while requiring only approximately 0.133% of the full-grid search time.

To validate the error trend under different activation thresholds δ, we conduct experiments with varying δ values on both datasets. As shown in Figure 3, the theoretically predicted optimal δ^* is indicated by the vertical dashed lines. These results show that our theoretical predictions closely align with the empirical optima, supporting the derived threshold-selection rule and the associated theoretical characterization.

3.2 Parsimonious Activation Validation

Figure 4 presents a comparison of dictionary atom usage frequencies between the L_∞- and L_1-regularized dictionaries, evaluated under identical parameter configurations and averaged over the CIFAR-100 test set. The usage in the PADL model is concentrated on a smaller subset of the

Table 2: Reconstruction performance comparison on CIFAR-100 and SVHN. For recent baselines and PADL, results are reported as mean ± standard deviation over 10 repeated runs.

Method	CIFAR-100			SVHN		
	RMSE	PSNR	SSIM	RMSE	PSNR	SSIM
KSVD [2]	0.152	17.87	0.472	0.147	16.34	0.449
DDL [39]	0.132	18.07	0.514	0.124	18.01	0.527
GDDL [40]	0.123	18.32	0.519	0.118	19.33	0.539
BPFA [30]	0.127	18.14	0.520	0.121	18.72	0.529
IBP [10]	0.119	18.58	0.535	0.119	19.57	0.545
CRsAE [41]	0.106 ± 0.007	20.13 ± 0.56	0.558 ± 0.035	0.092 ± 0.006	20.30 ± 0.60	0.565 ± 0.025
JumpReLU SAE [34]	0.104 ± 0.016	19.58 ± 0.97	0.520 ± 0.041	0.091 ± 0.010	20.01 ± 1.19	0.518 ± 0.042
L_1-Regularized DL	0.112 ± 0.006	18.98 ± 0.54	0.545 ± 0.030	0.094 ± 0.012	20.46 ± 1.03	0.559 ± 0.037
PADL	$\mathbf{0.099 \pm 0.009}$	$\mathbf{20.56 \pm 0.55}$	$\mathbf{0.568 \pm 0.030}$	$\mathbf{0.090 \pm 0.006}$	$\mathbf{20.88 \pm 1.03}$	$\mathbf{0.583 \pm 0.037}$

Table 3: Computational-cost summary on CIFAR-100 using 2,000 training patches.

Method	Cost	Outcome
K-SVD [2]	661.4s	20 atoms / 0.152 RMSE
CRsAE [41]	32.0s	56 atoms / 0.106 RMSE
JumpReLU SAE [34]	3.1s	61 atoms / 0.104 RMSE
PADL	44.2s	19 atoms / 0.099 RMSE

Table 4: Ablation study on CIFAR-100.

Configuration	RMSE	PSNR	SSIM
PADL	0.099	20.56	0.568
w/o L_∞	0.112	18.98	0.545
w/o L_1	0.108	19.17	0.549
w/o L_1 & L_∞	0.117	18.49	0.525

original dictionary, suggesting that the regularization on R encourages a more compact and efficient representation.

3.3 Comparative Reconstruction Performance

While our primary focus is theoretical validation, we compare PADL with existing dictionary learning and sparse auto-encoder methods to assess its practical utility. Tables 2 and 3 show that PADL achieves competitive reconstruction quality with substantially fewer activated atoms.

We further evaluate the computational cost associated with these improvements. Using a smaller subset of 2,000 training patches, Table 3 summarizes training-time cost on the controlled dictionary-learning benchmark. Although PADL is not the fastest training procedure among all baselines, it remains within a practical wall-clock range while producing the smallest activated dictionary and the lowest RMSE in this comparison. Detailed training-time and warm-up-cost studies are provided in Appendices F.2 and F.10.

3.4 Ablation Studies

We conducted ablation studies to evaluate the contribution of each component in our framework. Table 4 shows the performance of different configurations. Figure 5 shows the sensitivity of recon-

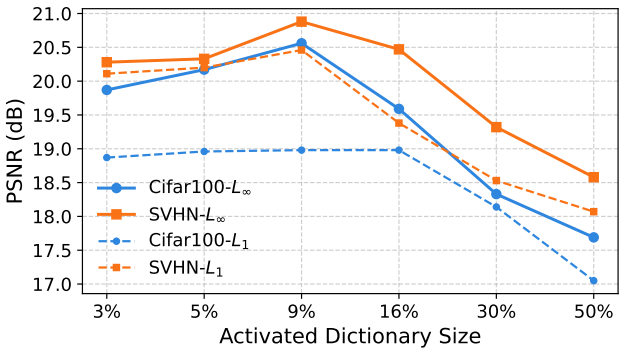


Figure 5: PSNR comparison on CIFAR-100 and SVHN under varying dictionary activation levels.

Table 5: Performance comparison on image VQA benchmarks. Payload reduction is measured relative to the uncompressed visual representation; for PADL, it includes dimensional compression and quantization. PADL denotes the version without adapter fine-tuning, while PADL+ and LLaVA-PruMerge+ use adapter fine-tuning.

Method	LLM	Payload reduction	VQA^{v2}	SQA^{I}	VQA^{T}	POPE	MME	MMB
InstructBLIP	Vicuna-7B	–	-	60.5	50.1	-	-	36.0
InstructBLIP	Vicuna-13B	–	-	63.1	50.7	78.9	1212.8	-
Shikra	Vicuna-13B	–	77.4	-	-	-	-	58.8
IDEFICS-9B	LLaMA-7B	–	50.9	-	25.9	-	-	48.2
IDEFICS-80B	LLaMA-65B	–	-	60.0	30.9	-	-	54.5
Qwen-VL	Qwen-7B	–	**78.8**	67.1	**63.8**	-	-	38.2
LLaVA-1.5	Vicuna-7B	1×	78.5	66.8	58.2	85.9	**1510.7**	64.3
LLaVA-1.5 + PruMerge+	Vicuna-7B	4×	76.6	67.5	55.6	86.5	1414.0	62.9
LLaVA-1.5 + FreePruner	Vicuna-7B	2×	77.6	68.6	60.0	87.7	1485.2	63.8
LLaVA-PruMerge+	Vicuna-7B	4×	76.8	68.3	57.1	84.0	1462.4	**64.9**
LLaVA-1.5 + **PADL**	Vicuna-7B	8×	76.9	65.5	57.7	84.6	1473.1	64.1
LLaVA-1.5 + **PADL+**	Vicuna-7B	8×	78.2	**68.8**	61.0	**88.5**	1485.1	64.7

Table 6: Performance comparison on video reasoning benchmarks. For PADL, payload reduction includes dimensional compression and quantization.

Methods	**LLM size**	**Payload reduction**	**MSVD-QA**		**MSRVTT-QA**		**ActivityNet-QA**		**TGIF-QA**	
			Accuracy	**Score**	**Accuracy**	**Score**	**Accuracy**	**Score**	**Accuracy**	**Score**
VideoChat	7B	–	56.3	2.8	45.0	2.5	–	2.2	34.4	2.3
LLaMA-Adapter	7B	–	54.9	3.1	43.8	2.7	34.2	2.7	-	-
Video-ChatGPT	7B	–	64.9	3.3	49.3	2.8	35.2	2.7	51.4	3.0
Video-LLaVA	7B	1×	70.7	3.9	59.2	3.5	45.3	3.3	70.0	4.0
Video-LLaVA + PruMerge	7B	∼ 14×	71.1	3.9	58.4	3.5	48.3	3.4	-	-
Video-LLaVA + PruMerge+	7B	4×	71.1	–	59.3	–	47.7	–	–	–
Video-LLaVA + FreePruner	7B	8×	71.3	3.9	59.5	3.5	48.4	3.4	-	-
Video-LLaVA + **PADL**	7B	16×	72.3	3.9	61.1	3.5	48.1	3.4	69.9	4.1
Video-LLaVA + **PADL+**	7B	16×	**75.2**	**4.1**	**64.1**	**3.7**	**49.7**	**3.4**	**75.3**	**4.3**

struction error to sparsity variations. Compared with the L_1-regularized approach, PADL achieves better reconstruction performance while using fewer activated dictionary atoms. The results show that both regularization terms contribute significantly to the performance, with the ℓ_∞ term providing particularly important theoretical properties.

3.5 VLM Compression Results

We further evaluate PADL as a visual compression module for image and video VLMs. Tables 5 and 6 report downstream performance together with the reduction in visual communication payload. PADL achieves $8\times$ image and $16\times$ video payload reduction while maintaining competitive performance. With adapter fine-tuning, PADL+ further improves performance on several benchmarks. These results illustrate the practical payload–accuracy tradeoff for edge–cloud inference. Detailed formulation, compression settings, and threshold validation are provided in Appendix E.

4 Conclusion

In this work, we revisit PADL, a dictionary-learning formulation for balancing sparsity, storage, and accuracy, through a structured probabilistic perspective. By connecting the PADL objective to an auxiliary latent-variable MAP-induced formulation, we obtain interpretable parameter-calibration rules, suppression conditions, activated-dictionary-size estimates, reconstruction-error bounds, and a solvable activation-threshold condition. Compared with dictionary-learning analyses that focus primarily on reconstruction accuracy and coefficient sparsity, this view explicitly accounts for the activated dictionary size, which directly corresponds to storage cost. The resulting calibration procedure is simple to implement and reduces reliance on exhaustive hyperparameter search. Experiments

on classical image reconstruction benchmarks validate the predicted sparsity–storage–accuracy behavior, and additional VLM experiments suggest that the same principle can be useful for sparse visual-representation compression.

5 Acknowledgment

This work is supported by the Natural Science Foundation of China (Grant 62371270). We thank Prof. Yu Wang for insights motivating the trade-off analysis and Yuanbo Tang for guidance on dictionary validation.

References

[1] Don Adjeroh, Yong Zhang, Amar Mukherjee, Matt Powell, and Tim Bell. Dna sequence compression using the burrows-wheeler transform. In *Proceedings. IEEE computer society bioinformatics conference*, pages 303–313. IEEE, 2002.

[2] Michal Aharon, Michael Elad, and Alfred Bruckstein. K-svd: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Transactions on signal processing*, 54(11):4311–4322, 2006.

[3] Hedy Attouch, Jérôme Bolte, and Benar Fux Svaiter. Convergence of descent methods for semi-algebraic and tame problems: proximal algorithms, forward–backward splitting, and regularized gauss–seidel methods. *Mathematical programming*, 137(1):91–129, 2013.

[4] S Derin Babacan, Shinichi Nakajima, and Minh N Do. Bayesian group-sparse modeling and variational inference. *IEEE transactions on signal processing*, 62(11):2906–2921, 2014.

[5] Alberto Bocchinfuso, Daniela Calvetti, and Erkki Somersalo. Bayesian sparsity and class sparsity priors for dictionary learning and coding. *arXiv preprint arXiv:2309.00999*, 2023.

[6] Jérôme Bolte, Shoham Sabach, and Marc Teboulle. Proximal alternating linearized minimization for nonconvex and nonsmooth problems. *Mathematical Programming*, 146(1):459–494, 2014.

[7] Trevor Campbell and Boyan Beronov. Sparse variational inference: Bayesian coresets from scratch. *Advances in Neural Information Processing Systems*, 32, 2019.

[8] Feilong Chen, Minglun Han, Haozhi Zhao, Qingyang Zhang, Jing Shi, Shuang Xu, and Bo Xu. X-llm: Bootstrapping advanced large language models by treating multi-modalities as foreign languages. *arXiv preprint arXiv:2305.04160*, 2023.

[9] Scott Shaobing Chen, David L Donoho, and Michael A Saunders. Atomic decomposition by basis pursuit. *SIAM review*, 43(1):129–159, 2001.

[10] Hong-Phuong Dang and Pierre Chainais. Indian buffet process dictionary learning: algorithms and applications to image processing. *International Journal of Approximate Reasoning*, 83: 1–20, 2017.

[11] Hong Phuong Dang and Pierre Chainais. Towards dictionaries of optimal size: A bayesian non parametric approach. *Journal of Signal Processing Systems*, 90(2):221–232, 2018.

[12] Bogdan Dumitrescu and Ciprian Doru Giurcăneanu. Adaptive-size dictionary learning using information theoretic criteria. *Algorithms*, 12(9):178, 2019.

[13] Michael Elad. *Sparse and redundant representations: from theory to applications in signal and image processing*. Springer Science & Business Media, 2010.

[14] Georgios Exarchakis and Jorg Lucke. Discrete sparse coding. *Neural computation*, 29(11): 2979–3013, 2017.

[15] Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Jinrui Yang, Xiawu Zheng, Ke Li, Xing Sun, et al. Mme: A comprehensive evaluation benchmark for multimodal large language models. *arXiv preprint arXiv:2306.13394*, 2023.

[16] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6904–6913, 2017.

[17] Marc Henniges, Gervasio Puertas, Jörg Bornschein, Julian Eggert, and Jörg Lücke. Binary sparse coding. In *International Conference on Latent Variable Analysis and Signal Separation*, pages 450–457. Springer, 2010.

[18] Yunseok Jang, Yale Song, Youngjae Yu, Youngjin Kim, and Gunhee Kim. Tgif-qa: Toward spatio-temporal reasoning in visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2758–2766, 2017.

[19] Andreas Krause and Volkan Cevher. Submodular dictionary selection for sparse representation. In *International Conference on Machine Learning (ICML)*, 2010.

[20] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[21] Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. *arXiv preprint arXiv:2305.10355*, 2023.

[22] Bin Lin, Yang Ye, Bin Zhu, Jiaxi Cui, Munan Ning, Peng Jin, and Li Yuan. Video-llava: Learning united visual representation by alignment before projection. In *Proceedings of the 2024 conference on empirical methods in natural language processing*, pages 5971–5984, 2024.

[23] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 26296–26306, 2024.

[24] Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, et al. Mmbench: Is your multi-modal model an all-around player? In *European conference on computer vision*, pages 216–233. Springer, 2024.

[25] Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. Learn to explain: Multimodal reasoning via thought chains for science question answering. *Advances in Neural Information Processing Systems*, 35: 2507–2521, 2022.

[26] Ruipu Luo, Ziwang Zhao, Min Yang, Junwei Dong, Da Li, Pengcheng Lu, Tao Wang, Linmei Hu, Minghui Qiu, and Zhongyu Wei. Valley: Video assistant with large language model enhanced ability. *arXiv preprint arXiv:2306.07207*, 2023.

[27] Julien Mairal, Francis Bach, Jean Ponce, and Guillermo Sapiro. Online dictionary learning for sparse coding. In *Proceedings of the 26th annual international conference on machine learning*, pages 689–696, 2009.

[28] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Baolin Wu, Andrew Y Ng, et al. Reading digits in natural images with unsupervised feature learning. In *NIPS workshop on deep learning and unsupervised feature learning*, volume 2011, page 7. Granada, 2011.

[29] Bruno A Olshausen and David J Field. Sparse coding with an overcomplete basis set: A strategy employed by v1? *Vision research*, 37(23):3311–3325, 1997.

[30] John Paisley and Lawrence Carin. Nonparametric factor analysis with beta process priors. In *Proceedings of the 26th annual international conference on machine learning*, pages 777–784, 2009.

[31] Vardan Papyan, Jeremias Sulam, and Michael Elad. Working locally thinking globally: Theoretical guarantees for convolutional sparse coding. *IEEE Transactions on Signal Processing*, 65 (21):5687–5701, 2017.

[32] Vardan Papyan, Yaniv Romano, Jeremias Sulam, and Michael Elad. Theoretical foundations of deep learning via sparse representations: A multilayer sparse model and its connection to convolutional neural networks. *IEEE Signal Processing Magazine*, 35(4):72–89, 2018.

[33] Sangha Park, Seungryong Yoo, Jisoo Mok, and Sungroh Yoon. Save: Sparse autoencoder-driven visual information enhancement for mitigating object hallucination. *arXiv preprint arXiv:2512.07730*, 2025.

[34] Senthooran Rajamanoharan, Tom Lieberum, Nicolas Sonnerat, Arthur Conmy, Vikrant Varma, János Kramár, and Neel Nanda. Jumping ahead: Improving reconstruction fidelity with jumprelu sparse autoencoders. *arXiv preprint arXiv:2407.14435*, 2024.

[35] Yuzhang Shang, Mu Cai, Bingxin Xu, Yong Jae Lee, and Yan Yan. Llava-prumrge: Adaptive token reduction for efficient large multimodal models. *arXiv preprint arXiv:2403.15388*, 2024.

[36] Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. Towards vqa models that can read. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8317–8326, 2019.

[37] Yan Tang, Zihui Zhao, Zixuan Zhang, Yang Li, et al. Data-efficient and robust trajectory generation through pathlet dictionary learning. In *The Third Conference on Parsimony and Learning (Proceedings Track)*.

[38] Yuanbo Tang, Zhiyuan Peng, and Yang Li. Explainable trajectory representation through dictionary learning. In *Proceedings of the 31st ACM International Conference on Advances in Geographic Information Systems*, pages 1–4, 2023.

[39] Snigdha Tariyal, Angshul Majumdar, Richa Singh, and Mayank Vatsa. Deep dictionary learning. *IEEE Access*, 4:10096–10109, 2016.

[40] Snigdha Tariyal, Angshul Majumdar, Richa Singh, and Mayank Vatsa. Greedy deep dictionary learning. *arXiv preprint arXiv:1602.00203*, 2016.

[41] Bahareh Tolooshams, Sourav Dey, and Demba Ba. Deep residual autoencoders for expectation maximization-inspired dictionary learning. *IEEE Transactions on neural networks and learning systems*, 32(6):2415–2429, 2020.

[42] Joel A Tropp, Anna C Gilbert, and Martin J Strauss. Algorithms for simultaneous sparse approximation. part i: Greedy pursuit. *Signal processing*, 86(3):572–588, 2006.

[43] Paul Tseng. Convergence of a block coordinate descent method for nondifferentiable minimization. *Journal of optimization theory and applications*, 109(3):475–494, 2001.

[44] Bingxin Xu, Yuzhang Shang, Yunhao Ge, Qian Lou, and Yan Yan. freepruner: A training-free approach for large multimodal model acceleration. *arXiv preprint arXiv:2411.15446*, 2024.

[45] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. Msr-vtt: A large video description dataset for bridging video and language. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5288–5296, 2016.

[46] Zhou Yu, Dejing Xu, Jun Yu, Ting Yu, Zhou Zhao, Yueting Zhuang, and Dacheng Tao. Activitynet-qa: A dataset for understanding complex web videos via question answering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 9127–9134, 2019.

[47] Ming Yuan and Yi Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 68(1):49–67, 2006.

[48] Jusheng Zhang, Xiaoyang Guo, Kaitong Cai, Qinhan Lv, Yijia Fan, Wenhao Chai, Jian Wang, and Keze Wang. Hybridtoken-vlm: Hybrid token compression for vision-language models. *arXiv preprint arXiv:2512.08240*, 2025.

[49] Sipeng Zheng, Bohan Zhou, Yicheng Feng, Ye Wang, and Zongqing Lu. Unicode: Learning a unified codebook for multimodal large language models. In *European Conference on Computer Vision*, pages 426–443. Springer, 2024.

[50] Mingyuan Zhou, Haojun Chen, Lu Ren, Guillermo Sapiro, Lawrence Carin, and John Paisley. Non-parametric bayesian dictionary learning for sparse image representations. *Advances in neural information processing systems*, 22, 2009.

A Related Works

A.1 Dictionary Learning and Sparse Representation

Traditional dictionary learning aims to represent data using a small number of dictionary atoms while preserving reconstruction fidelity. Sparse coding and the seminal work of Olshausen and Field [29] established this formulation as a central model for sparse representation, and subsequent algorithms such as K-SVD [2] and online dictionary learning [27] made dictionary learning scalable to larger datasets. These classical methods primarily control sparsity at the coefficient level, typically through explicit sparsity constraints or ℓ_1-type regularization. In this paper, we focus on a different but related quantity: the number of dictionary atoms that are effectively activated across a dataset, which is directly tied to storage and deployment cost.

From the probabilistic perspective, dictionary learning is formulated through generative models over observations, latent coefficients, and sometimes latent usage variables. Bayesian sparse coding and nonparametric dictionary-learning models use such variables to infer sparsity patterns and dictionary size from data [30, 50, 10, 11]. Closely related models such as binary sparse coding [17] and discrete sparse coding [14] introduce discrete latent variables to represent sparse activations. These formulations provide a natural way to reason about posterior uncertainty, latent dimensionality, sample complexity, and generalization [30, 50, 10]. In these probabilistic formulations, the latent usage variables are commonly part of the reconstruction mechanism, for example by gating sample-level coefficients or factors [30, 50, 10, 17, 14]. Our formulation instead introduces an auxiliary atom-level activation variable that does not directly participate in reconstruction, but induces a row-wise activation prior corresponding to cross-sample atom usage.

From the optimization perspective, dictionary learning is commonly posed as a regularized reconstruction problem. This view leads to efficient objectives and algorithms, and it supports theoretical analyses of sparse recovery, stability, identifiability, and convergence [9, 31, 32]. Classical sparse coding, K-SVD, and online dictionary learning fall into this category [29, 2, 27]. Their regularizers are usually imposed on individual coefficients or sample-level sparse codes, making the learned representation compact but not explicitly controlling how many atoms are activated across the full dataset. As a result, the regularization weights that govern sparsity and dictionary usage are typically selected as external hyperparameters.

Structured sparse modeling extends this optimization view by introducing group or joint sparsity constraints. Group Lasso and simultaneous sparse approximation encourage structured selection across predefined groups or multiple related signals [47, 42]. Bayesian group-sparse modeling provides probabilistic interpretations of group selection [4], and recent dictionary-coding work further explores sparsity and class-sparsity priors for discarding irrelevant dictionary components [5]. These works are closely related in that they impose structure beyond element-wise sparsity. The constraint considered here is positioned differently: it is atom-wise but defined across samples, so that the row of coefficients associated with an atom determines whether that atom contributes to the effective dictionary size.

PADL is designed to connect these two perspectives. On the optimization side, it gives a tractable objective with reconstruction, coefficient sparsity, and cross-sample atom activation terms. On the probabilistic side, it admits an auxiliary-latent MAP formulation in which the activation variable induces the row-wise regularizer without gating the reconstruction likelihood. This connection allows us to retain the computational form of regularized dictionary learning while deriving quantities that are more naturally obtained from a probabilistic view, including expected activated dictionary size, generalization bounds, and a data-driven activation-threshold rule.

A.2 Compositional Learning Methods on VLM

As a broader interpretation of dictionary learning, recent work on VLMs has increasingly emphasized compositional representations, where complex visual or multimodal inputs are decomposed into a small set of reusable primitives. UniCode [49] proposes a unified codebook that maps heterogeneous modalities into a shared discrete space, enabling cross-modal compositionality and efficient reuse. SAVE [33] leverages sparse autoencoders to extract semantically meaningful visual factors, showing that structured sparsity can mitigate object hallucination by enforcing interpretable latent components. HybridToken-VLM and HTC-VLM [48] compress visual prefixes through hybrid token represen-

Table 7: High-level comparison of representative dictionary-learning formulations.

Perspective	Representative works	Relation to PADL
Optimization-based	Sparse coding, K-SVD, online DL [29, 2, 27]	Optimizes reconstruction with coefficient sparsity; PADL further regularizes atom activation across samples.
Probabilistic	Bayesian sparse coding, IBP/Beta-process DL, binary/discrete sparse coding [30, 50, 10, 11, 17, 14]	Provides generative views of sparsity and dictionary size; PADL uses an auxiliary activation variable to induce a row-wise MAP regularizer.
Global-constraint	Group Lasso, simultaneous sparse approximation, Bayesian group sparsity [47, 42, 4, 5]	Models group or shared-support structure; PADL focuses on atom-wise activation across the dataset.
PADL	This work	Connects a storage-aware optimization objective with an auxiliary-latent probabilistic formulation.

tations, combining semantic abstraction with token-level efficiency. These methods demonstrate that replacing dense visual streams with structured, compositional units can significantly improve both efficiency and robustness in VLMs. However, existing approaches typically introduce such structure heuristically or implicitly, without a principled mechanism to control the tradeoff among representation sparsity, dictionary size, and reconstruction fidelity. Our work complements these efforts by providing a probabilistically grounded dictionary learning framework that makes this tradeoff explicit and optimizable, enabling compositional compression in VLMs with theoretical guarantees.

B Proof of MAP Objective

This appendix derives the first-order MAP-induced objective in Proposition 2.1 under the coefficient-prior approximation in Lemma B.1. It also explains the row-wise ℓ_∞ activation surrogate used in the practical PADL objective.

B.1 Posterior Energy

Recall the joint distribution

$$P(X, R, z; D) = \prod_{i=1}^{N} P(X_i \mid R_i; D) \cdot \prod_{k=1}^{K} \prod_{i=1}^{N} p(R_{ki}) \cdot \prod_{k=1}^{K} P(z_k \mid r_k),$$

where $r_k = (R_{k1}, \ldots, R_{kN})$ is the k-th row of R. Taking the negative log-posterior and using the inactive event $z_k = 0$ for the storage term gives

$$-\log P(R, z \mid X; D) = \sum_{i=1}^{N} \frac{1}{2} \|X_i - DR_i\|_2^2 - \sum_{k,i} \log p(R_{ki}) - \sum_{k} \log P(z_k = 0 \mid r_k) + C, \tag{14}$$

where C is a constant independent of (R, z).

In the following, we analyze the two prior-induced terms and show that they correspond to ℓ_1 and ℓ_∞-type regularization.

B.2 Beta Prior Induces ℓ_1 Regularization

Lemma B.1 (Beta Prior and ℓ_1 Regularization)**.** *Suppose $R_{ki} \sim \mathrm{Beta}(1, \beta)$ with $\beta > 1$ and that R_{ki} are in the small-coefficient regime ($R_{ki} \ll 1$) for all k, i. Then*

$$-\sum_{k,i} \log p(R_{ki}) \;\approx\; \text{const} + (\beta - 1)\, \|R\|_1, \tag{15}$$

where $\|R\|_1 = \sum_{k,i} |R_{ki}|$ and the constant term does not depend on R.

Proof. For each coefficient,

$$p(R_{ki}) = \beta(1 - R_{ki})^{\beta-1},$$

and hence

$$-\log p(R_{ki}) = -\log \beta - (\beta - 1)\log(1 - R_{ki}).$$

For $R_{ki} \ll 1$, we use the first-order Taylor approximation $\log(1 - R_{ki}) \approx -R_{ki}$, yielding

$$-\log p(R_{ki}) \approx -\log \beta + (\beta - 1)R_{ki}.$$

Summing over all (k, i) gives

$$\begin{aligned} -\sum_{k,i} \log p(R_{ki}) &\approx -KN \log \beta + (\beta - 1) \sum_{k,i} R_{ki} \\ &= \text{const} + (\beta - 1)\|R\|_1, \end{aligned}$$

which completes the proof. □

Lemma B.1 shows that the Beta prior asymptotically induces an ℓ_1-type penalty with weight $\lambda_1 = \beta - 1$ near zero. This encourages per-sample sparsity while imposing stronger penalties on large coefficients.

B.3 Activation Prior Induces a Row-wise ℓ_∞ Barrier

Recall the activation model

$$z_k \mid r_k \sim \text{Bernoulli}\left(\frac{1}{1 + \exp(-\gamma(\max_i R_{ki} - \delta))}\right).$$

Define the softplus function

$$\phi_\gamma(u) = \log(1 + e^{\gamma u}).$$

Lemma B.2 (Activation Prior and ℓ_∞ Barrier)**.** *Up to an additive constant independent of R,*

$$-\log P(z_k = 0 \mid r_k) \simeq \phi_\gamma(\max_i R_{ki} - \delta).$$

Moreover, as $\gamma \to \infty$, ϕ_γ converges to a hard barrier:

$$\lim_{\gamma \to \infty} \phi_\gamma(u) = \begin{cases} 0, & u < 0, \\ \infty, & u > 0. \end{cases}$$

Proof. We have

$$P(z_k = 1 \mid r_k) = \frac{1}{1 + \exp(-\gamma(\max_i R_{ki} - \delta))},$$

and hence

$$P(z_k = 0 \mid r_k) = \frac{1}{1 + \exp(\gamma(\max_i R_{ki} - \delta))}.$$

Therefore,

$$-\log P(z_k = 0 \mid r_k) = \log\big(1 + \exp(\gamma(\max_i R_{ki} - \delta))\big) = \phi_\gamma(\max_i R_{ki} - \delta).$$

For $u < 0$, $\gamma u \to -\infty$ as $\gamma \to \infty$, so $\phi_\gamma(u) \to 0$. For $u > 0$, $\gamma u \to +\infty$ and $\phi_\gamma(u) \to \infty$. This establishes the barrier behavior. □

Lemma B.2 shows that the activation prior penalizes the row-wise peak $\max_i R_{ki}$ and converges to a hard constraint $\max_i R_{ki} \le \delta$ as $\gamma \to \infty$. This motivates the row-wise ℓ_∞ activation surrogate used in Eq. 8.

B.4 Derivation of the MAP Objective

Substituting Lemmas B.1 and B.2 into the posterior energy 14, and setting the noise scale to $\sigma = 1$ by normalization, we obtain the first-order MAP-induced objective

$$\min_{D,R} \ \|X - DR\|_F^2 + (\beta - 1)\|R\|_1 + \sum_{k=1}^{K} \phi_\gamma(\max_i R_{ki} - \delta).$$

Writing $\lambda_1 = \beta - 1$ and introducing a scaling constant $\lambda_2 > 0$ for the activation prior yields

$$\min_{D,R} \ \|X - DR\|_F^2 + \lambda_1\|R\|_1 + \lambda_2 \sum_{k=1}^{K} \phi_\gamma(\max_i R_{ki} - \delta),$$

which corresponds to Eq. 7 in the main text. The soft barrier penalizes the row-wise peak $\max_i R_{ki}$ and, for large γ, approaches a hard threshold constraint at δ. The computational objective in Eq. 8 uses the row-wise ℓ_∞ term as a convex activation surrogate for this peak-activation cost, thereby completing the proof of Proposition 2.1.

B.5 Numerical Diagnostic of the First-Order Approximation

We examine supplementary direct-threshold reconstruction experiments on CIFAR-100 and SVHN using 2,000 training patches, 500 held-out patches, and $K = 128$. For each dataset, five coefficient initializations share one learned dictionary and the same evaluation samples. These experiments are reported separately from Algorithm 1.

Let $h(r) = -\log(1 - r)$ and $C_{ki} = R_{ki}\mathbf{1}\{R_{ki} \geq \delta\}$, with $\delta = 0.40$ for CIFAR-100 and $\delta = 0.49$ for SVHN. We report the mean-coefficient diagnostic

$$\varepsilon_{\text{mean}} = \frac{h(\bar{c}) - \bar{c}}{h(\bar{c})}, \qquad \bar{c} = \frac{1}{KN}\sum_{k,i} C_{ki}.$$

The mean includes all coefficient locations, including zeros. Since h is nonlinear, this diagnostic differs from the aggregate approximation error of the coefficient-prior term.

Table 8: Mean-coefficient diagnostic across five coefficient initializations, reported in percent.

Dataset	Mean (%)	Std. (%)	Range (%)
CIFAR-100	3.7523	0.0016	$[3.7506, 3.7545]$
SVHN	3.8248	0.0049	$[3.8167, 3.8295]$

C Technical Lemmas for Tradeoff Characterization

This appendix provides the technical foundations for the tradeoff analysis. We establish sufficient conditions under which non-essential atoms are suppressed, derive a high-probability upper bound on the aggregate activation mass of an atom, and combine these results to obtain a reconstruction error bound.

Throughout this section, let $r_k = (R_{k1}, \ldots, R_{kN})$ denote the coefficient row corresponding to atom d_k, and define the objective

$$f(D, R) = \|X - DR\|_F^2 + \lambda_1\|R\|_1 + \lambda_2 \sum_{k=1}^{K} \|r_k\|_\infty,$$

where $\|r_k\|_\infty = \max_i |R_{ki}|$.

C.1 Sufficient Condition for Atom Suppression

Lemma C.1 (Sufficient Condition for Atom Suppression)**.** *Assume that a subset of atoms S satisfies*

$$\|X - D_S R_S\|_F^2 \leq \varepsilon,$$

where D_S and R_S denote the dictionary and coefficients restricted to S. For an atom $k \notin S$, let r_k be its coefficient row and define the residual $e = X - D_S R_S$. Suppose $\|d_k\|_2 = 1$ and that

$$|e_i^\top d_k| \le \eta \quad \textit{for all } i.$$

If $r_k \neq 0$ and

$$\lambda_1 + \lambda_2 \frac{\|r_k\|_\infty}{\|r_k\|_1} \; > \; 2\eta, \tag{16}$$

then setting $r_k = 0$ strictly decreases the objective, i.e., $f(D, R_{-k}) - f(D, R) < 0$. In particular, for an atom whose peak activation satisfies $\|r_k\|_\infty \ge \delta$, it is sufficient that

$$\lambda_1 + \lambda_2 \frac{\delta}{\|r_k\|_1} \; > \; 2\eta. \tag{17}$$

Proof. Let R_{-k} denote the coefficient matrix obtained by zeroing the k-th row of R, and define $\Delta f := f(D, R_{-k}) - f(D, R)$.

Reconstruction term. Using $X - DR = e - d_k r_k^\top$ and $\|d_k\|_2 = 1$, we have

$$\|X - DR\|_F^2 = \|e\|_F^2 - 2\sum_{i=1}^{N} e_i^\top d_k \, R_{ki} + \|r_k\|_2^2.$$

Therefore,

$$\Delta_{\text{rec}} = \|e\|_F^2 - \|X - DR\|_F^2 = 2\sum_{i=1}^{N} e_i^\top d_k \, R_{ki} - \|r_k\|_2^2 \le 2\eta \, \|r_k\|_1,$$

where the negative quadratic term has been dropped to obtain a valid upper bound.

Regularization term. Zeroing r_k reduces the ℓ_1 penalty by $\lambda_1\|r_k\|_1$ and reduces the row-wise ℓ_∞ penalty by $\lambda_2\|r_k\|_\infty$. Hence

$$\Delta_{\text{reg}} = -\lambda_1\|r_k\|_1 - \lambda_2\|r_k\|_\infty.$$

Combining terms. Combining the two bounds gives

$$\Delta f \le (2\eta - \lambda_1)\|r_k\|_1 - \lambda_2\|r_k\|_\infty.$$

Thus $\Delta f < 0$ whenever Eq. 16 holds. If $\|r_k\|_\infty \ge \delta$, then $\|r_k\|_\infty / \|r_k\|_1 \ge \delta / \|r_k\|_1$, and the simpler threshold-dependent condition in Eq. 17 implies Eq. 16. □

C.2 Expected Dictionary Activation Size

For each atom k, define the indicator of being active under threshold δ:

$$\mathbf{1}_k(\delta) \; := \; \mathbb{I}\Big\{ \max_{1 \le i \le N} R_{ki} > \delta \Big\}.$$

Then the active set size satisfies

$$|S(\delta)| = \sum_{k=1}^{K} \mathbf{1}_k(\delta), \qquad \Rightarrow \quad \mathbb{E}|S(\delta)| = \sum_{k=1}^{K} \mathbb{E}\,\mathbf{1}_k(\delta).$$

By identical distribution across k, it suffices to compute $\mathbb{E}\,\mathbf{1}_1(\delta) = \Pr(\max_i R_{1i} > \delta)$.

Since $\{R_{1i}\}_{i=1}^N$ are i.i.d. with CDF F, we have

$$\Pr\Big(\max_{1 \le i \le N} R_{1i} \le \delta \Big) = \prod_{i=1}^{N} \Pr(R_{1i} \le \delta) = F(\delta)^N.$$

Therefore,

$$\Pr\Big(\max_{1 \le i \le N} R_{1i} > \delta \Big) = 1 - F(\delta)^N,$$

and hence

$$\mathbb{E}|S(\delta)| = \sum_{k=1}^{K} \Pr\left(\max_{1 \le i \le N} R_{ki} > \delta \right) = K\big(1 - F(\delta)^N\big).$$

Substituting $F(\delta) = 1 - (1-\delta)^\beta$ yields

$$\mathbb{E}|S(\delta)| = K\left[1 - \big(1 - (1-\delta)^\beta\big)^N\right],$$

which proves 24.

To show monotonicity and smoothness, note that for $\beta > 0$, $F(\delta)$ is smooth and strictly increasing on $(0,1)$ with density

$$f(\delta) = F'(\delta) = \beta(1-\delta)^{\beta-1} > 0.$$

Thus, $\mathbb{E}|S(\delta)|$ is smooth on $(0,1)$ and

$$\frac{d}{d\delta}\mathbb{E}|S(\delta)| = -K\,N\,F(\delta)^{N-1} f(\delta) < 0,$$

so it is strictly decreasing in δ.

Finally, the endpoint values follow immediately: $F(0) = 0 \Rightarrow \mathbb{E}|S(0)| = K(1-0) = K$, and $F(1) = 1 \Rightarrow \mathbb{E}|S(1)| = K(1-1) = 0$.

At test time, the training-calibrated threshold is applied unchanged. Since Eq. 11 also depends on the sample count N, activated dictionary sizes can differ between training and test sets even under matched distributions.

C.3 High-Probability Reconstruction Error Bound

Lemma C.2 (High-Probability Upper Bound on $\|r_k\|_1$)**.** *Assume $R_{ki} \overset{\text{i.i.d.}}{\sim} \mathrm{Beta}(1,\beta)$ with $\beta > 1$. Then for any $\tau \in (0,1)$, with probability at least $1-\tau$,*

$$\|r_k\|_1 \le U(N,\beta,\tau) := \frac{N}{1+\beta} + \sqrt{\frac{N}{2}\log\frac{1}{\tau}}. \tag{18}$$

Proof. Each R_{ki} lies in $[0,1]$ and satisfies $\mathbb{E}[R_{ki}] = \frac{1}{1+\beta}$. Let $S_k = \sum_{i=1}^N R_{ki} = \|r_k\|_1$. By Hoeffding's inequality,

$$\Pr(S_k - \mathbb{E}[S_k] \ge t) \le \exp\left(-\frac{2t^2}{N}\right).$$

Setting $t = \sqrt{\frac{N}{2}\log\frac{1}{\tau}}$ yields

$$\Pr\left(S_k \ge \frac{N}{1+\beta} + \sqrt{\frac{N}{2}\log\frac{1}{\tau}} \right) \le \tau,$$

which proves the claim. □

Lemma C.3 (High-Probability Reconstruction Bound)**.** *Let $e = X - D_S R_S$ be the residual using the active atom set S, and define*

$$\eta := \max_{i,k} |e_i^\top d_k|.$$

Assume that every non-essential active candidate $k \notin S$ satisfies $\|r_k\|_\infty \ge \delta$. Assume also that the active sub-dictionary satisfies the restricted active-span condition

$$\|D_S^\top e_i\|_2^2 \ge \alpha_S \|e_i\|_2^2 \qquad \text{for all } i, \tag{19}$$

for some $\alpha_S > 0$. Then, for any $\tau \in (0,1)$, with probability at least $1-\tau$,a sufficient condition for suppressing each fixed non-essential atom is

$$2\eta \le \lambda_1 + \frac{\lambda_2 \delta}{U(N,\beta,\tau)}. \tag{20}$$

Under this condition, the reconstruction error satisfies

$$\|e\|_F^2 \le \frac{N}{4}\frac{|S|}{\alpha_S}\left(\lambda_1 + \frac{\lambda_2\delta}{U(N,\beta,\tau)}\right)^2 \quad \text{with probability at least } 1-\tau. \tag{21}$$

Proof. From Lemma C.1, if $\|r_k\|_\infty \geq \delta$, a sufficient condition for suppressing a non-essential atom $k \notin S$ is

$$2\eta \leq \lambda_1 + \lambda_2 \frac{\delta}{\|r_k\|_1}.$$

By Lemma C.2, with probability at least $1-\tau$, $\|r_k\|_1 \leq U(N,\beta,\tau)$. Therefore $\delta/\|r_k\|_1 \geq \delta/U(N,\beta,\tau)$, and Eq. 20 is a sufficient condition.

For the geometric part, Eq. 19 gives

$$\alpha_S \|e_i\|_2^2 \leq \|D_S^\top e_i\|_2^2 = \sum_{k\in S} (e_i^\top d_k)^2.$$

Since $(e_i^\top d_k)^2 \leq \eta^2$, we have $\sum_{k\in S}(e_i^\top d_k)^2 \leq |S|\eta^2$, hence

$$\|e_i\|_2^2 \leq \frac{|S|}{\alpha_S}\eta^2.$$

Summing over all i yields

$$\|e\|_F^2 \leq N\,\frac{|S|}{\alpha_S}\,\eta^2.$$

Substituting the bound on η from 20 gives 21. □

Remark C.4 (Geometry of the active-span condition). The condition in Eq. 19 avoids requiring $D_S D_S^\top$ to be full rank. If residuals are restricted to $\mathrm{span}(D_S)$ and $D_S^\top D_S \succeq \alpha_S I$, then Eq. 19 holds. For arbitrary residuals, the same argument controls the component projected onto the active span.

C.4 Numerical Examination of the Active-Span Condition

Using the experiments in Appendix B.5, we evaluate Eq. 19 with the matched dictionary D, normalized data X, and thresholded coefficients C. Let $e = X - DC$ and let S contain the nonzero rows of C. For each nonzero residual, we compute

$$\alpha_i = \frac{\|D_S^\top e_i\|_2^2}{\|e_i\|_2^2}, \qquad \alpha_{\min} = \min_{i:\|e_i\|_2>0} \alpha_i.$$

Table 9 summarizes $\alpha_{\min}$ across coefficient initializations. The minima remain separated from zero at the evaluated operating points.

Table 9: Statistics of the minimum active-span ratio across five coefficient initializations in the supplementary direct-threshold experiments.

Dataset	Mean	Std.	Range
CIFAR-100	2.1408	0.0177	[2.1254, 2.1710]
SVHN	2.7652	0.0975	[2.6750, 2.9113]

D Scaling Behavior and Interpretation

Recall from Lemma C.3 that, with probability at least $1-\tau$, the reconstruction error satisfies

$$\|e\|_F^2 \leq \frac{N}{4}\,\frac{|S|}{\alpha_S}\left(\lambda_1 + \frac{\lambda_2\delta}{U(N,\beta,\tau)}\right)^2, \tag{22}$$

where S is the active atom set, $|S|$ its cardinality, and α_S is the restricted active-span constant from Eq. 19.

D.1 Dependence on the Number of Samples

For fixed β and confidence level τ,

$$U(N, \beta, \tau) = \frac{N}{1+\beta} + \sqrt{\frac{N}{2} \log \frac{1}{\tau}}$$

grows linearly with N as $N \to \infty$. Consequently, the contribution of the activation term inside the parentheses in Eq. 22 decays as $\mathcal{O}(1/N)$, and the bound asymptotically simplifies to

$$\frac{\|e\|_F^2}{N} = \mathcal{O}\left(\frac{|S|}{\alpha_S}\,\lambda_1^2\right). \tag{23}$$

Thus, the reconstruction error *per sample* remains bounded in the large-sample regime.

Intuitively, as more visual samples become available, the aggregate activation mass $\|r_k\|_1$ of each atom concentrates around its expectation under the Beta prior. The threshold contribution $\lambda_2\delta/U(N, \beta, \tau)$ therefore vanishes in the large-sample regime, while the active set size is still controlled by the threshold through $\mathbb{E}|S(\delta)|$.

D.2 Roles of ℓ_1 and ℓ_∞

The two regularizers in the objective contribute in complementary ways.

The ℓ_1 term determines the asymptotic error floor through λ_1. It controls the average magnitude and count of coefficients within each sample, thereby governing per-sample representation complexity. In the large-N regime, λ_1 becomes the dominant factor shaping the reconstruction error.

In contrast, the row-wise ℓ_∞ term enters the suppression condition through

$$\frac{\|r_k\|_\infty}{\|r_k\|_1},$$

and, for atoms whose peak exceeds the threshold, through the conservative quantity $\delta/U(N, \beta, \tau)$. This ratio is large for atoms whose usage is concentrated in a small number of samples, so such atoms are more easily suppressed when they are weakly aligned with the residual. Conversely, atoms that contribute frequently across samples incur only a vanishing activation penalty as N grows.

This mechanism formalizes the intended behavior of the combined ℓ_1–ℓ_∞ regularization: it favors atoms that provide stable, compositional structure while eliminating peak-only or idiosyncratic components. In the context of visual representations, the model retains primitives that recur across many samples and discards those that encode transient or sample-specific artifacts.

D.3 Dependence on Dictionary Geometry

The factor $|S|/\alpha_S$ in Eq. 22 captures the effective model complexity. The cardinality $|S|$ measures how many atoms are actually used, while α_S quantifies how well the active span captures the residual directions being bounded.

Even with a large nominal dictionary, the bound depends only on the *active* subset and its restricted active-span conditioning. Poorly conditioned or overly redundant selections (small α_S) can inflate the bound, whereas compact and well-spread subsets yield tighter guarantees. This reveals that reconstruction accuracy is governed jointly by statistical regularization and the geometry of the selected representation.

D.4 Optimal Activation Threshold $\delta^\star$

We can further obtain an explicit characterization of the optimal activation threshold δ by minimizing a proxy of the theoretical upper bound. The key observation is that δ controls the expected active dictionary size through the Beta prior, while simultaneously entering the reconstruction bound through the activation penalty term.

Active set size as a function of δ. Define the active atom set induced by thresholding

$$S(\delta) := \{k \in [K] : \max_{1 \le i \le N} R_{ki} > \delta\}.$$

Under $R_{ki} \overset{i.i.d.}{\sim} \mathrm{Beta}(1, \beta)$, let

$$F(\delta) := \Pr(R_{ki} \le \delta) = 1 - (1-\delta)^{\beta}, \qquad f(\delta) := F'(\delta) = \beta(1-\delta)^{\beta-1}.$$

Then

$$\mathbb{E}|S(\delta)| = K\Big(1 - F(\delta)^N\Big), \tag{24}$$

since $\Pr(\max_i R_{ki} \le \delta) = F(\delta)^N$.

A bound-driven objective. From Lemma C.3, the reconstruction error bound takes the form

$$\|e\|_F^2 \;\lesssim\; \frac{N}{4}\,\frac{|S|}{\alpha_S}\left(\lambda_1 + \frac{\lambda_2 \delta}{U(N,\beta,\tau)}\right)^2.$$

Treating α_S as approximately constant with respect to δ (standard in such proxy optimization), and replacing $|S|$ by its expectation in 24, we minimize the scalar objective

$$\Phi(\delta) := \Big(1 - F(\delta)^N\Big)\left(\lambda_1 + \frac{\lambda_2 \delta}{U(N,\beta,\tau)}\right)^2, \qquad \delta \in [0,1]. \tag{25}$$

This objective explicitly captures the sparsity–storage–accuracy tradeoff: $1 - F(\delta)^N$ decreases with δ (fewer active atoms, lower storage), while the factor $\lambda_1 + \lambda_2\delta/U(\cdot)$ increases with δ (stronger activation constraint, potentially higher error).

Closed-form optimality condition. A stationary point $\delta^\star \in (0,1)$ of 25 satisfies

$$\frac{N\,F(\delta)^{N-1}\,f(\delta)}{1 - F(\delta)^N} = \frac{2\lambda_2}{U(N,\beta,\tau)\left(\lambda_1 + \frac{\lambda_2\delta}{U(N,\beta,\tau)}\right)}. \tag{26}$$

The left-hand side is determined solely by the Beta prior and sample count N, while the right-hand side depends on the regularization scale λ_1, activation weight λ_2, and concentration term $U(N,\beta,\tau)$. In practice, $\delta^\star$ can be obtained efficiently by one-dimensional root finding (e.g., bisection or Newton's method), yielding a stable and deterministic hyperparameter selection rule.

Remark D.1 (Non-degenerate regime). The bound in Theorem 2.4 is derived under the restricted active-span condition in Eq. 19. When $\delta \to 1$, the active set $S(\delta)$ becomes empty and this condition is violated, so the bound no longer applies. Therefore, the endpoint $\delta = 1$ corresponds to a degenerate regime and is excluded from the feasible domain. The meaningful optimum lies in the interior $\delta \in (0,1)$, where the stationary condition in Eq. 26 characterizes the trade-off between sparsity, storage, and accuracy.

D.5 Interpretation from an MDL Perspective

From a Minimum Description Length (MDL) viewpoint, the objective

$$\|X - DR\|_F^2 + \lambda_1\|R\|_1 + \sum_{k=1}^{K}\|r_k\|_\infty$$

can be interpreted as trading off three code lengths: the cost of encoding the residual, the cost of encoding sparse coefficients, and the cost of instantiating dictionary atoms.

The ℓ_1 term penalizes the number and magnitude of nonzero coefficients, corresponding to the description length of per-sample codes. The row-wise ℓ_∞ term penalizes whether an atom is used at all, corresponding to the cost of introducing a new primitive into the model. In this sense, the activation penalty performs model selection over atoms, while the sparsity penalty performs compression within the selected model.

The theory therefore shows that the PADL formulation can be interpreted as Bayesian model selection under an auxiliary activation prior.

E VLM Inference Compression

This section provides the detailed VLM formulation, compression protocol, and supplementary validation for PADL-based visual-representation compression. The main image and video results are reported in the main text, while the appendix focuses on implementation details and threshold validation.

E.1 VLM Integration

The parsimoniously activated dictionary learning paradigm can be naturally integrated into VLM inference as a structured visual compression module. In modern multimodal systems, images and videos are encoded into long sequences of visual tokens that often contain substantial redundancy. These dense prefixes dominate both computation and communication cost, especially in edge-cloud deployment scenarios, where visual features must be transmitted to a remote language model.

Consider a standard VLM pipeline. An image $\mathbf{I} \in \mathbb{R}^{C\times H\times W}$ or a video $\mathbf{I}_{\text{vid}} \in \mathbb{R}^{N_t\times C\times H\times W}$ is processed by a vision encoder that partitions each frame into $N_s = \frac{H}{p} \times \frac{W}{p}$ non-overlapping patches and maps each patch to a d-dimensional embedding. For frame i, this yields

$$\mathbf{Z}_V^{(i)} = [\mathbf{z}_{i,1}, \ldots, \mathbf{z}_{i,N_s}] \in \mathbb{R}^{N_s\times d}, \quad \mathbf{z}_{i,j} \in \mathbb{R}^d.$$

Aggregating all frames gives a visual token matrix

$$\mathbf{Z}_V \in \mathbb{R}^{N_V\times d}, \qquad N_V = N_t N_s,$$

which is concatenated with the textual embedding $\mathbf{Z}_T \in \mathbb{R}^{L_T\times d}$ and fed into the LLM for autoregressive generation.

Our method inserts a dictionary-based compression layer on $\mathbf{Z}_V$. We learn a visual dictionary $\mathbf{D}_V \in \mathbb{R}^{d\times K}$ and compute a sparse representation

$$\mathbf{Z}_V \approx (\mathbf{D}_V\mathbf{R}_V)^T, \qquad \mathbf{R}_V \in \mathbb{R}^{K\times N_V},$$

where each column of $\mathbf{R}_V$ is sparse and only a small subset of atoms in $\mathbf{D}_V$ are globally activated. In an edge-cloud setting, the sparse codes can be quantized and transmitted:

$$\widetilde{\mathbf{R}}_V = \text{Quantize}(\mathbf{R}_V),$$

together with the activated sub-dictionary $\mathbf{D}_S \subset \mathbf{D}_V$. As shown in Figure 2, on the cloud side, visual tokens can be reconstructed by

$$\mathbf{Z}_V^* = (\mathbf{D}_S\widetilde{\mathbf{R}}_V)^T,$$

and then concatenated with $\mathbf{Z}_T$ for standard VLM inference. Alternatively, a lightweight adapter f_a can be trained to directly process the sparse representation:

$$\widetilde{\mathbf{Z}}_V^* = f_a(\widetilde{\mathbf{R}}_V, \mathbf{D}_S),$$

bypassing explicit reconstruction. This enables the language model to operate on compressed visual features, yielding further acceleration. In both modes, our framework reduces the visual representation size and controls the number of stored primitives through a principled sparsity-storage-accuracy tradeoff.

E.2 Compression Setting and Evaluation Protocol

We evaluate PADL as a sparse visual-representation compression module. In this setting, the visual features produced by the vision encoder are projected onto a learned dictionary, and the resulting sparse codes are used as compact visual representations. The reported compression ratio refers to the reduction in visual representation size induced by dimensional compression and quantization. Specifically, for image inputs we apply $4\times$ dimensional compression together with $2\times$ quantization, resulting in an overall $8\times$ representation-size reduction. For video inputs, we apply $8\times$ dimensional compression together with $2\times$ quantization, resulting in an overall $16\times$ representation-size reduction.

For image VQA experiments, we use LLaVA-1.5 [23] as the backbone and evaluate on VQAv2 [16], ScienceQA-IMG [25], TextVQA [36], POPE [21], MME [15], and MMBench [24]. The dictionary

contains 1024 atoms. We compare the original LLaVA-1.5 model with LLaVA-PruMerge [35], FreePruner [44], PADL without adapter fine-tuning, and PADL with a lightweight adapter. We denote the version without adapter fine-tuning as **PADL**, and the version with adapter fine-tuning as **PADL+**. The adapter is trained on the LLaVA-1.5 instruction-tuning dataset.

For video VQA experiments, we use Video-LLaVA as the backbone and evaluate on MSVD-QA [8], MSRVTT-QA [45], ActivityNet-QA [46], and TGIF-QA [18]. The adapter is fine-tuned on the Valley dataset [26]. The evaluation follows the same protocol as prior video VQA work, reporting both accuracy and GPT-based answer-quality scores where applicable.

E.3 Computational Overhead

Table 10 reports the parameter and FLOP overhead introduced by the PADL-based VLM compression module. The dictionary encoding and adapter introduce additional computation, but this overhead is substantially smaller than the downstream LLM-side computation saved by compressing visual representations. This supports the practical motivation of applying PADL to VLM inference, where the visual prefix can contribute significantly to the total computational cost. In the direct-adapter mode, the adapter maps the compressed sparse representation to a shorter visual prefix before LLM decoding, which accounts for the reported LLM-side FLOP saving.

Table 10: Parameter and FLOP overhead of the PADL-based VLM compression module. Dictionary encoding excludes final reconstruction. The LLM-side computation saving is reported for video inputs.

Item	Parameters	FLOPs
Dictionary encoding	8.91M	+1.725T
Adapter	38.05M	+13.0G
LLM-side computation saving	–	-25.64T

For video inputs, the projected visual tensor has shape $[8, 257, 4096]$, giving $N_v = 2056$ tokens. With $d = 4096$, $K = 1024$, and $s = 50$ coefficient updates, the matrix-multiplication cost of dictionary encoding is

$$F_{\text{enc}} = 4sdKN_v = 1.725 \text{ TFLOPs},$$

counting one multiply-add as two FLOPs. Final reconstruction adds $2dKN_v$ FLOPs, giving 1.742 TFLOPs in total. For 576 image tokens, the corresponding encoding and total costs are 0.483 and 0.488 TFLOPs. These counts exclude offline dictionary fitting, adapter training, and quantization.

E.4 Threshold Validation on VLM Compression

To examine whether the threshold predicted by our theory remains meaningful in the VLM setting, we perform a threshold sweep on VQAv2 using PADL without adapter fine-tuning. We vary the activation threshold δ from 0.38 to 0.46 with step size 0.02. The theoretically predicted optimal threshold is $\delta = 0.412$, which is close to the empirical optimum at $\delta = 0.42$.

Table 11: Threshold validation for PADL-based VLM compression on VQAv2 without adapter fine-tuning. The theoretically predicted threshold $\delta = 0.412$ closely matches the empirical optimum around $\delta = 0.42$.

δ	0.38	0.40	0.42	0.44	0.46
VQAv2 Accuracy	71.5	74.8	**77.0**	76.8	76.7

These results suggest that the proposed data-driven threshold estimation can provide a useful operating point even in the VLM compression setting. We emphasize that this experiment is not intended to replace the controlled reconstruction experiments in the main text, but rather to show that the threshold-selection principle transfers to a larger multimodal application scenario.

E.5 Main VLM Results

The main image and video VQA results are reported in Tables 5 and 6. This appendix focuses on the detailed compression setup, computational overhead, and threshold validation.

E.6 Discussion

The VLM experiments support three observations. First, PADL can be used as a plug-in visual compression module for both image and video VLMs. Second, the predicted activation threshold provides a useful operating point in the VLM setting, as shown by the VQAv2 threshold sweep in Table 11. Third, adapter fine-tuning can further align compressed sparse representations with the downstream language model, improving the performance of PADL+ over the no-adapter variant.

We emphasize that these experiments are not intended to claim that PADL universally dominates all VLM token-pruning or token-merging methods. Instead, they show that this SSA-based dictionary learning framework can be integrated into multimodal inference pipelines and can achieve favorable compression–performance tradeoffs. The main theoretical and empirical claims of the paper are supported by the controlled reconstruction experiments in the main text, where reconstruction accuracy, coefficient sparsity, and activated dictionary size can be directly measured.

F Additional Dictionary Learning Experiments

This section reports additional dictionary-learning experiments that further support the empirical claims in the main paper. These experiments are designed to evaluate modern sparse-representation baselines, optimization stability, robustness to warm-up length and nominal dictionary size, mini-batch approximation, post-training pruning, and the effect of patch-boundary artifacts on SSIM. They are placed in the appendix because the main text focuses on the core sparsity–storage–accuracy (SSA) tradeoff and the corresponding theoretical analysis.

F.1 Comparison with Recent Sparse-Representation Baselines

In addition to classical dictionary-learning and Bayesian sparse coding baselines, we compare PADL with recent sparse-representation methods, including CRsAE and JumpReLU SAE. We also report an L_1-regularized dictionary learning baseline to isolate the effect of the proposed row-wise activation regularization. All results are averaged over 10 repeated runs.

Table 12: Comparison with recent sparse-representation baselines on CIFAR-100 and SVHN. Results are reported as mean and standard deviation over 10 repeated runs. Lower RMSE is better; higher PSNR and SSIM are better.

Dataset	Method	RMSE	PSNR	SSIM
CIFAR-100	CRsAE [41]	0.106 ± 0.007	20.128 ± 0.561	0.557 ± 0.035
	JumpReLU SAE [34]	0.104 ± 0.016	19.584 ± 0.971	0.520 ± 0.041
	L_1-Regularized DL	0.112 ± 0.006	18.980 ± 0.540	0.545 ± 0.030
	PADL	$\mathbf{0.099 \pm 0.009}$	$\mathbf{20.560 \pm 0.554}$	$\mathbf{0.568 \pm 0.030}$
SVHN	CRsAE [41]	0.092 ± 0.006	20.302 ± 0.599	0.565 ± 0.025
	JumpReLU SAE [34]	0.091 ± 0.010	20.008 ± 1.194	0.518 ± 0.042
	L_1-Regularized DL	0.094 ± 0.012	20.465 ± 1.031	0.559 ± 0.037
	PADL	$\mathbf{0.090 \pm 0.006}$	$\mathbf{20.880 \pm 1.030}$	$\mathbf{0.583 \pm 0.037}$

These results show that PADL is competitive not only with classical sparse coding methods, but also with more recent sparse-representation baselines. The improvement over the L_1-regularized baseline suggests that the row-wise activation-aware term provides benefits beyond ordinary element-wise sparsity.

F.2 Training Time and Activated Dictionary Size

We further compare training time, activated atoms, and reconstruction error on CIFAR-100 using 2000 samples, 8×8 patches, and 400 training iterations. The number of activated atoms measures the effective dictionary size after training.

Table 13: Training time, activated atoms, and reconstruction error on CIFAR-100. PADL is not the fastest method, but it achieves the lowest RMSE while using the fewest activated atoms among the recent sparse-representation baselines.

Method	Train Time	Activated Atoms	RMSE
K-SVD	661.4s	20	0.152
CRsAE [41]	32.0s	56	0.106
JumpReLU SAE [34]	3.1s	61	0.104
PADL	44.2s	**19**	**0.099**

The results indicate that PADL remains within a practical training-time range while learning a more compact effective dictionary. Compared with CRsAE and JumpReLU SAE, PADL uses fewer activated atoms and achieves lower reconstruction error. This supports the claim that PADL improves the sparsity–storage–accuracy tradeoff rather than merely improving reconstruction accuracy.

F.3 Effect of Patch-Boundary Artifacts on SSIM

The relatively low SSIM values in the main patch-based reconstruction protocol are partly caused by patch-boundary artifacts. To verify this, we conduct an additional experiment on CIFAR-100 using overlapping reconstruction. The overlap reduces stitching artifacts between neighboring patches.

Table 14: Reconstruction performance on CIFAR-100 with overlapping reconstruction. The improvement in SSIM confirms that the lower SSIM in the original patch-based protocol is partly caused by patch-boundary artifacts.

Method	SSIM	RMSE	PSNR
CRsAE [41]	0.8337	0.1012	23.0179
JumpReLU SAE [34]	0.7366	0.0959	23.9402
PADL	**0.8402**	**0.0937**	**24.7478**

With overlapping reconstruction, PADL achieves an SSIM of 0.8402, substantially higher than the value obtained under the non-overlapping patch protocol. This suggests that the original SSIM scores should be interpreted together with the patch-based evaluation setting, and that PADL remains competitive when patch-boundary artifacts are mitigated.

F.4 Optimization Stability

Although the PADL objective is non-convex and contains a non-smooth row-wise ℓ_∞-type regularization term, the alternating optimization procedure is empirically stable. Table 15 reports the evolution of the reconstruction term, the element-wise sparsity term, and the row-wise activation term on CIFAR-100 and SVHN. Each term is averaged by sample size.

The reconstruction and regularization terms show clear descent and stabilization behavior on both datasets. On SVHN, the L_1 term slightly increases from iteration 300 to 400, but the overall optimization remains stable and the reconstruction term continues to decrease. These results provide empirical evidence that the non-smooth activation-aware regularization does not cause severe oscillation or divergence under our training settings.

Table 15: Optimization behavior of the three objective terms on CIFAR-100 and SVHN. The objective terms generally decrease and stabilize during training, suggesting stable empirical optimization behavior.

Iter	CIFAR-100			SVHN		
	Recon	L_1	L_∞	Recon	L_1	L_∞
20	0.774834	0.252	0.748664	1.344806	0.251	0.701239
100	0.658850	0.186	0.701099	1.084268	0.224	0.507329
200	0.649933	0.165	0.594202	0.737957	0.214	0.451594
300	0.647558	0.151	0.518092	0.583525	0.203	0.447624
400	0.647077	0.141	0.507625	0.545520	0.209	0.439489

F.5 Robustness to Warm-Up Length

PADL estimates prior-related parameters during a warm-up stage. To evaluate whether the method is sensitive to the exact warm-up length, we vary the number of warm-up steps t_s on CIFAR-100 and report results over 10 repeated runs.

Table 16: Effect of warm-up steps t_s on CIFAR-100. Results are reported over 10 repeated runs.

t_s	10	30	50	70	100
RMSE	0.1074 ± 0.0127	0.1015 ± 0.0115	0.0994 ± 0.0093	0.0992 ± 0.0101	0.0989 ± 0.0088

When t_s increases from 10 to 100, RMSE improves from 0.1074 to 0.0989. More importantly, once the warm-up length reaches a moderate range, performance becomes stable: the difference between $t_s = 50$, $t_s = 70$, and $t_s = 100$ is small. This indicates that PADL is not highly sensitive to the exact warm-up length as long as a reasonable warm-up stage is used.

F.6 Robustness to Nominal Dictionary Size

For image reconstruction, $K = 128$ gives a $2\times$ overcomplete dictionary for $d = 64$ grayscale patches. For VLM compression, $K = 1024$ provides an undercomplete bottleneck for $d = 4096$ visual tokens.

We next vary the nominal dictionary size K on CIFAR-100. Since PADL is designed to control the effective activated dictionary size, the number of activated atoms should grow much more slowly than the nominal dictionary size.

Table 17: Effect of nominal dictionary size K on CIFAR-100. Results are reported over 10 repeated runs. Although K increases by $16\times$, the number of activated atoms grows only moderately.

K	32	64	128	256	512
RMSE	0.1013 ± 0.0119	0.0997 ± 0.0113	0.0994 ± 0.0093	0.1002 ± 0.0102	0.1030 ± 0.0129
Activated Atoms	13	14	19	21	24
Activated Ratio	41%	22%	15%	8%	5%

Although the nominal dictionary size increases from 32 to 512, the number of activated atoms only increases from 13 to 24. This demonstrates that PADL does not simply activate proportionally more atoms as the initial dictionary becomes larger. Instead, it maintains a relatively compact effective active subset, which is consistent with the intended role of the row-wise activation regularization.

F.7 Mini-Batch Approximation

The row-wise activation term introduces cross-sample dependence at the latent-code level. For large-scale datasets, full-batch optimization may be impractical, so we use mini-batch approximation in large-scale settings. Table 18 compares full-batch and mini-batch optimization on CIFAR-100.

Table 18: Comparison between full-batch and mini-batch optimization on CIFAR-100. Mini-batch optimization closely matches the full-batch result.

Setting	RMSE	PSNR	SSIM
Full-batch, $N = 2000$	0.099	20.56	0.568
Mini-batch, $B = 256$	0.101	20.54	0.567

The difference between full-batch and mini-batch optimization is small under this setting. This supports the use of mini-batch approximation when scaling PADL to larger datasets or higher-dimensional visual features.

F.8 Post-Training Dictionary Pruning

PADL does not dynamically shrink the nominal dictionary size during training. Instead, it encourages the data to concentrate usage on a compact subset of atoms, which makes post-training pruning possible. To evaluate this effect, we prune low-utilization atoms after training and measure reconstruction performance on CIFAR-100.

Table 19: Post-training pruning on CIFAR-100. PADL supports substantial atom pruning with limited degradation, indicating that it learns a compact effective dictionary.

Pruning Ratio	30%	40%	50%	60%	70%
RMSE	0.098	0.100	0.101	0.101	0.102
PSNR	20.05	20.03	19.98	19.91	19.81
SSIM	0.561	0.557	0.551	0.544	0.536

Even when 60% of atoms are pruned, RMSE only changes to 0.101. This suggests that many low-utilization atoms are not essential for reconstruction and that PADL concentrates representation usage onto a compact active subset. Therefore, the main efficiency benefit of PADL should be understood as effective dictionary compression and deployment-time pruning, rather than training-time reduction of the nominal dictionary size.

F.9 Empirical Reasonableness of the Beta Surrogate

The probabilistic derivation of PADL uses a Beta-inspired surrogate prior over coefficients. We do not interpret optimized coefficients as literal i.i.d. Beta samples. Instead, the Beta prior is used as a tractable surrogate that induces an interpretable objective and enables data-driven hyperparameter estimation. To assess the empirical reasonableness of this surrogate, we fit a Beta distribution to the warm-up coefficients on CIFAR-100.

Table 20: Empirical Beta fit of warm-up coefficients on CIFAR-100. The fitted distribution is close to the assumed $\mathrm{Beta}(1, \beta)$ family.

$\hat{\alpha}$	$\hat{\beta}$	KS Statistic	Q-Q RMSE
0.956	2.31	0.0115	0.005366

The fitted value $\hat{\alpha} = 0.956$ is close to 1, and the discrepancy measures are small. This supports the use of the $\mathrm{Beta}(1, \beta)$ family as a practical approximation for deriving a tractable MAP surrogate and estimating hyperparameters from warm-up coefficients.

F.10 Warm-Up Cost Compared with Grid Search

The warm-up stage is used for one-shot data-driven parameter estimation. It is not a repeated hyperparameter search procedure. To illustrate the difference, we compare the cost of our warm-up-based estimation with repeated parameter search on 2000 CIFAR-100 samples.

Table 21: Cost comparison between warm-up-based parameter estimation and repeated parameter search on 2000 CIFAR-100 samples.

Procedure	Time Cost
PADL warm-up + training	52.5651s
10 rounds of parameter search	395.2022s
Estimated $10 \times 10 \times 10$ grid search	10.98h

The full-grid time is estimated by scaling the measured cost of 10 search rounds to 10^3 configurations. This estimate is approximately $752\times$ the reported warm-up-and-training cost. The corresponding final-performance comparison is reported in Table 1.

F.11 Additional Baseline: Group-Sparse Regularization

The row-wise activation regularization in PADL is related to structured sparsity, but it is not equivalent to a standard group-sparse penalty. To examine this distinction, we compare with an $\ell_{2,1}$-regularized baseline on CIFAR-100.

Table 22: Comparison with an $\ell_{2,1}$-regularized baseline on CIFAR-100. The $\ell_{2,1}$ baseline uses 23% of atoms but gives worse reconstruction than PADL.

Method	Activated Atom Ratio	RMSE
$\ell_{2,1}$-Regularized Baseline	23%	0.115
PADL	15%	**0.099**

This result suggests that PADL is not merely replacing the element-wise sparsity penalty with a generic group-sparse regularizer. The row-wise activation-aware term is designed to control the effective activated dictionary size across samples, which leads to a better reconstruction–storage tradeoff under this setting.

F.12 Summary

The additional experiments support the main claims from several perspectives. First, PADL remains competitive against recent sparse-representation baselines while using fewer activated atoms. Second, the optimization process is empirically stable despite the non-convex and non-smooth objective. Third, the method is robust to moderate changes in warm-up length and nominal dictionary size. Fourth, mini-batch approximation closely matches full-batch optimization in the controlled setting. Fifth, post-training pruning confirms that PADL concentrates usage on a compact effective dictionary. Finally, the Beta-fit experiment and warm-up cost comparison support the practical role of the probabilistic surrogate and the one-shot parameter-estimation procedure.

G Limitations

This work focuses on revisiting PADL through a probabilistic lens and deriving practical calibration rules, rather than exhaustively covering all dictionary-learning variants. The analysis uses a Beta-inspired surrogate prior and an active-span condition to obtain tractable characterizations of atom suppression, activated dictionary size, and reconstruction error. These assumptions are used to guide data-driven hyperparameter estimation, and our empirical results suggest that the resulting estimates provide useful operating points. The current implementation adopts a one-shot warm-up stage for efficiency; iterative parameter re-estimation may further refine the calibration.

The resulting estimates depend on the warm-up protocol, and an overly short warm-up can degrade performance. The reported reconstruction comparisons are descriptive and do not include formal statistical significance tests.

The VLM experiments serve as application-level validation for visual-representation compression, and broader evaluation across additional backbones and deployment settings is left for future work.

H Broader Impacts

This work primarily contributes to the academic study of dictionary learning, sparse representation, and structured compression. By revisiting PADL through a probabilistic perspective, the paper provides a bridge between optimization-based dictionary-learning objectives and generative modeling interpretations. We expect this connection to be useful for future research on theoretically grounded sparse coding, interpretable representation learning, and data-driven hyperparameter calibration.

A potential positive impact of this work is that it encourages more systematic analysis of the tradeoff among sparsity, storage, and accuracy. This tradeoff is increasingly important as machine learning models are deployed in settings where memory, communication, and computational resources are constrained. The proposed analysis may help researchers design more compact representations without relying solely on exhaustive hyperparameter search, thereby improving the accessibility and efficiency of experimental research.

The VLM experiments further suggest that storage-aware dictionary learning may be a useful tool for studying visual-representation compression in multimodal systems. In this sense, the work may support future academic research on efficient VLM inference, edge-cloud collaboration, and compositional visual representations. At the same time, the proposed method is a general representation-compression technique and does not by itself address the broader risks of downstream multimodal models, such as hallucination, bias, privacy leakage, or misuse. When integrated into deployed VLM systems, it may inherit the risks and limitations of those systems. We therefore view the VLM results as application-level validation of the sparsity–storage–accuracy principle rather than as a standalone deployment-ready solution.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction state the scope of the proposed PADL framework, including its probabilistic motivation, activation-aware regularization, theoretical analysis, and empirical validation. The claims are aligned with the theoretical results and experimental evidence presented in the main text and appendix.

   Guidelines: In the introduction we highlighted the theoretical contributions with experimental validations.

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper discusses limitations related to computational cost, parameter sensitivity, storage–accuracy trade-offs, and the experimental scope. Additional runtime, parameter, and ablation results are provided in the experiments and appendix.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: The main theoretical statements explicitly state the required assumptions, including the probabilistic model, activation thresholding rule, and restricted active-span condition. Complete derivations and proofs are provided in the appendix, with the main text presenting the key statements and intuition.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The paper reports the datasets, preprocessing procedures, model configurations, dictionary sizes, optimization settings, hyperparameter calibration procedure, and baseline configurations needed to reproduce the main experimental results. Additional details are included in the appendix.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: The current submission does not include a public code release. Nevertheless, the paper provides algorithmic details, dataset descriptions, preprocessing procedures, and experimental settings to support independent reimplementation; an anonymized code release is planned.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: The experimental section and appendix specify the data splits, preprocessing, dictionary sizes, training schedules, hyperparameters, hyperparameter selection procedure, optimizer settings, and baseline implementations used in the reported experiments.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

Justification: The appendix reports standard deviations for repeated experimental runs where applicable. These error estimates quantify variability across repeated runs under the same experimental settings.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The paper reports the computational resources used for the experiments, including the relevant hardware setting and runtime or overhead measurements. Additional compute-related details are provided in the appendix.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research follows the NeurIPS Code of Ethics. The work uses standard public datasets and pretrained models, does not involve human-subject studies, and is reported in an anonymized form for submission.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.

- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [N/A]

    Justification: This is a fundamental research and not tied to particular applications.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper poses no such risks.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

Justification: The paper cites the public datasets, pretrained models, and baseline methods used in the experiments, and respects their stated licenses and terms of use. No existing assets are redistributed as part of this submission.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [N/A]

Justification: The paper does not release new assets.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [N/A]

Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.