Regular Article

Development of a Multi-Constraint Loss Function for Image Recovery from Pruned Features

Van-Phi Hoang, Thi-Nga Dao

Faculty of Radio-Electronic Engineering, Le Quy Don Technical University, Hanoi, Vietnam

Correspondence: Thi-Nga Dao, daothinga@lqdtu.edu.vn
Communication: received 27 May 2025, revised 27 June 2025, accepted 29 June 2025
Online publication: 30 June 2025, Digital Object Identifier: 10.21553/rev-jec.408

Abstract– **Existing image recovery methods have demonstrated that original images can be reconstructed from full features. This work considers a more challenging problem of recovering images from pruned features learned by deep neural networks. The problem is addressed in this study by introducing a multi-constraint loss function that integrates L2 distance, sixth-power summation, and total variation regularization to enhance reconstruction quality. This function enhances image smoothness and fidelity while ensuring that reconstructed images are encoded as vectors closely aligned with the pruned feature. The proposed loss function enables robust image recovery, preserving key visual features even at high pruning ratios. Additionally, this study investigates the impact of different pruning levels on reconstruction fidelity, highlighting the trade-off between pruning efficiency and recoverability. These findings provide valuable insights into inverse problems in deep learning and image processing, with implications for security risk assessment and feature redundancy analysis.**

Keywords– **Deep neural network, image recovery, inverse problem, optimization, pruned features.**

1 INTRODUCTION

Feature extraction is a fundamental technique for learning essential patterns from input data while reducing dimensionality [1]. Deep learning models, in particular, effectively extract hierarchical features from images, encoding essential visual information compactly through hidden layers [2]. In this context, the inverse problem of reconstructing input images from these extracted features has received significant attention. The feasibility of reconstructing images from full (unpruned) deep features has been established in previous studies. For instance, Mahendran *et al.* [3] demonstrated that such features retain sufficient information to allow image inversion through optimization techniques, laying the groundwork for understanding what is encoded in neural network representations. However, this work considers a more challenging and less explored problem: recovering images from pruned features learned by deep neural networks. Pruning is a widely adopted technique to create more efficient models by removing less salient neurons or connections, thereby reducing computational and storage overhead [4], [5]. While effective for model compression, pruning inherently introduces information loss, making image recovery significantly more difficult compared to unpruned cases. Structural information within pruned feature spaces is often diminished, presenting unique challenges for reconstruction algorithms.

Addressing this problem is crucial for several reasons. First, it provides insight into feature redundancy within neural networks and evaluates how much meaningful information remains post-pruning. Second, it supports the development of improved pruning strategies by identifying recoverability limits [6]. Third, it has significant implications for privacy and security, as it relates to model inversion attacks where adversaries attempt to reconstruct sensitive data from compressed or pruned components [7], a concern for data collected by ubiquitous Internet of Things (IoT) devices. Despite its importance, the problem of reconstructing images from pruned features remains largely underexplored. Existing feature extraction tools, such as autoencoders [1], are not specifically optimized for this task, especially when dealing with structurally incomplete features produced by aggressive pruning.

To address these limitations, a multi-constraint optimization framework is introduced for recovering images from pruned features. The proposed approach employs a unified loss function that integrates L2 distance, sixth-power summation, and total variation regularization to mitigate information loss and enhance reconstruction fidelity. At the beginning of the recovery process, a random dummy image is initialized. The L2 distance is then computed by comparing the feature extracted from the dummy image with the original pruned feature. The sixth-power summation term is used to constrain the range of reconstructed images. Finally, the total variation regularizer ensures image smoothness by comparing values of neighboring pixels in reconstructed images. The dummy image can be found by solving an optimization problem with the weighted loss function using the Adam optimizer.

Comprehensive experiments on MNIST, CIFAR-100, and LFWPeople datasets using several deep learning models to generate pruned features demonstrate significantly superior performance, achieving lower Mean Squared Error (MSE), Higher Structural Simi-

larity Index (SSIM), and Peak Signal-to-Noise Ratio (PSNR) across all datasets. Furthermore, an in-depth analysis of the impact of pruning ratios on recovery quality reveals a fundamental trade-off between pruning efficiency and reconstruction fidelity, providing insights into the structural importance of retained features [8]. The key contributions include the development of a novel framework for recovering original training data from pruned features, experimental validation across diverse datasets, and a systematic evaluation of the trade-offs between pruning and reconstruction quality. These findings advance the fields of model compression, secure data handling, and feature redundancy analysis.

The remainder of this paper is structured as follows. Section 2 reviews related work, section 3 presents the proposed method, section 4 discusses experimental results, and section 5 concludes with future directions.

2 RELATED WORK

This section examines image recovery methods, emphasizing regularization techniques, full feature limitations, and challenges in reconstructing pruned features.

2.1 Regularization techniques in image recovery

Regularization techniques are critical for addressing noise suppression, detail preservation, and ambiguity in image recovery. Total Variation (TV) regularization, introduced by Rudin *et al.*[9] has been widely used to promote smoothness while preserving edges in image reconstruction tasks. Building on this, higher-order norms, such as the sum of sixth powers proposed in this study, extended the concept of sparsity-inducing norms. While L1 regularization (Lasso) is a commonly adopted approach, researchers like Osher *et al.*[10] have demonstrated the advantages of using higher-order norms in specific scenarios, achieving improved performance in various image processing applications, including medical imaging and computer vision. The integration of multiple regularization techniques has emerged as a powerful strategy, with studies showing that well-designed combinations often outperform single-method approaches.

2.2 Image recovery from full and pruned features

Image recovery from deep neural network features has advanced significantly. Mahendran *et al.* [3] pioneered inversion of full features via optimization, proving deep features retain recoverable visual information. Their framework laid the foundation for subsequent advancements in image reconstruction. Following this, Nash *et al.* [11], introduced hierarchical inversion techniques that refine coarse-to-fine details, improving sharpness and perceptual quality. Yang *et al.* [12] further enhanced inversion techniques by leveraging adversarial alignment, demonstrating improved recovery performance for complex feature spaces. While most prior studies have focused on full features, pruned features introduce additional, particularly challenging issues due to the substantial information loss during neuron removal. Despite their efficiency for storage and privacy [4], achieved through pruning techniques like structured pruning which are designed to create compact models [5], the inherent complexity of pruned networks requires dedicated techniques to compensate for missing feature information [8].

This gap motivates the present study, which specifically focuses on recovering images from pruned features. By analyzing the redundancy in retained features and assessing the feasibility of reconstruction under aggressive pruning, this work introduces a specialized framework aimed at improving image recovery in constrained environments.

3 PROPOSED METHOD

This section presents a new multi-constraint loss framework for image recovery from pruned features, explaining its methodology, optimization, and theoretical basis for reliable validation.

3.1 Method overview

The proposed method aims to recover an original image x_o from its pruned feature $\boldsymbol{h}_o^{pruned}$, where x_o denotes the input image and $\boldsymbol{h}_o^{pruned}$ is a pruned version of the full feature vector $\boldsymbol{h}_o$. Here, $\boldsymbol{h}_o$ represents the unpruned feature vector extracted from the penultimate layer of a convolutional neural network, encapsulating high-level visual information of x_o. The pruned feature $\boldsymbol{h}_o^{pruned}$ is derived by removing a fraction of neurons from $\boldsymbol{h}_o$ based on a pruning ratio, which quantifies the proportion of discarded features. The ratio ranges from 0.0, indicating no pruning, to 0.9, corresponding to 90% neuron removal. The relationship between $\boldsymbol{h}_o$ and $\boldsymbol{h}_o^{pruned}$ directly impacts reconstruction fidelity, as higher pruning ratios reduce feature dimensionality and increase information loss.

Figure 1 presents the overall workflow, where a dummy input x_d is iteratively refined using a multi-constraint loss function to produce the reconstructed image x_*. Although not part of the core recovery procedure, a separate evaluation step is applied to measure reconstruction quality. The process begins by initializing x_d with uniformly sampled pixel values. Through optimization, x_d is adjusted such that its pruned feature representation $\boldsymbol{h}_d^{pruned}$, obtained via the network, closely matches the target $\boldsymbol{h}_o^{pruned}$. This is achieved by minimizing a multi-constraint loss function $L(x)$, formulated as

$$L(x) = \alpha_{L2} L_{L2} + \alpha_{sixnorm} L_{sixnorm} + \alpha_{TV} L_{TV}, \quad (1)$$

where α_{L2}, $\alpha_{sixnorm}$, and α_{TV} are weights coefficients balancing three complementary components:

- L2 Distance: Measures the normalized Euclidean distance between the dummy and target pruned features.

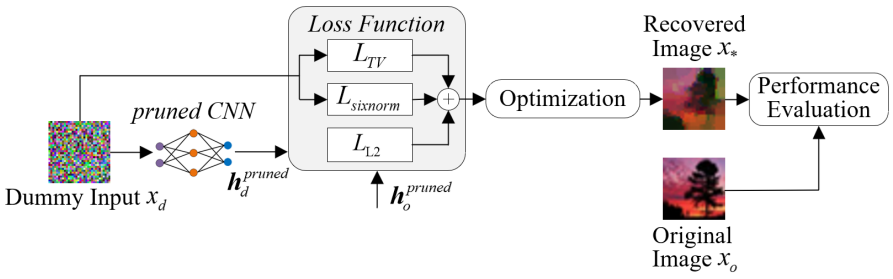

Figure 1. Overview of the proposed image recovery framework and its evaluation.

$$L_{L2} = \frac{\left\| h_o^{pruned} - h_d^{pruned} \right\|_2^2}{\left\| h_o^{pruned} \right\|_2^2}. \quad (2)$$

- Sixth-power Summation: This term limits pixel values to prevent excessive growth, enhance features, avoid flat regions, and reduce noise for smoother reconstructed images.

$$L_{sixnorm} = \sum_i (x_d^i)^6. \quad (3)$$

- Total Variation (TV): This promotes smoothness by penalizing high variations between neighboring pixels [9].

$$L_{TV} = \sum_i (x_d^i - x_d^{i+1})^2. \quad (4)$$

A grid search methodology was employed to determine optimal values for α_{L2}, $\alpha_{sixnorm}$, and α_{TV}. This process evaluated diverse parameter combinations across datasets to ensure robustness and generalizability. The selected weights balance reconstruction accuracy, noise suppression, and spatial coherence while avoiding overfitting to specific datasets.

The image recovery process follows an iterative approach described in Algorithm 1. This algorithm gradually refines an initial dummy image to match the original pruned feature. The process begins with the pruned feature, denoted as h^o_{pruned}, along with a pre-trained neural network model and the loss function weights α_{L2}, $\alpha_{sixnorm}$, and α_{TV}. To ensure adaptability across diverse recovery scenarios, key operational parameters such as the learning rate and maximum iteration count are predefined. At the start, a dummy image x_d is initialized with pixel values randomly sampled from a uniform distribution within a specified range $[r_1, r_2]$. In this work, the parameters are set as $r_1 = 0$ and $r_2 = 1$. In each iteration, the algorithm performs three main steps. First, forward propagation generates the pruned feature h_d^{pruned} for the dummy image. Next, the multi-constraint loss function $L(x)$ is evaluated to quantify the discrepancy between h_d^{pruned} and h_o^{pruned}. In the update phase, the Adam optimizer is employed with the specified learning rate to iteratively refine x_d in the direction that minimizes $L(x)$. To maintain the reconstructed image within the valid pixel range [0, 1], a clipping operation is performed after each update. The process continues until the loss difference between consecutive iterations falls below a predefined threshold ϵ, indicating convergence, or the maximum number of iterations is reached. Once completed, the algorithm outputs the optimized image x_*, which serves as the recovered visual content based on the pruned features.

Algorithm 1 Multi-constraint loss optimization for image recovery from pruned features

Input:
  Pruned original feature h_o^{pruned}
  Neural network model
  Loss weights $\{\alpha_{L2}, \alpha_{sixnorm}, \alpha_{TV}\}$
Parameters:
  Learning rate lr
  Maximum iterations max_{iters}
Initialization:
  Initialize dummy image $x_d \sim \mathcal{U}[0,1]$ ▷ Uniform distribution
Optimization:
for $iter = 1$ **to** max_{iters} **do**
  Forward pass:
    Compute pruned dummy feature h_d^{pruned}
  Loss computation:
    $L_{L2} \leftarrow \frac{\left\| h_o^{pruned} - h_d^{pruned} \right\|_2^2}{\left\| h_o^{pruned} \right\|_2^2}$ ▷ L2 distance
    $L_{sixnorm} \leftarrow \sum_i (x_d^i)^6$ ▷ Sixth power term
    $L_{TV} \leftarrow \sum_i (x_d^i - x_d^{i+1})^2$ ▷ Total variation
    $L(x) \leftarrow \alpha_{L2} L_{L2} + \alpha_{sixnorm} L_{sixnorm} + \alpha_{TV} L_{TV}$
  Update step:
    Update x_d using Adam optimizer with lr
    $x_d \leftarrow clip(x_d, 0, 1)$ ▷ Ensure valid pixel range
  if $\|L(x)_{iter} - L(x)_{iter-1}\| < \epsilon$ **then** ▷ Check convergence
    break
  end if
end for
Output: Recovered image x_*

3.2 Theoretical analysis

The proposed method addresses the challenging task of image recovery from pruned neural network features, requiring a specialized approach due to the inherent complexity of deep networks and the information loss caused by pruning. To achieve this, the method employs a unified loss function that integrates multiple constraints to enhance reconstruction quality. The normalized L2 distance term is introduced to minimize discrepancies between the reconstructed

image and the pruned feature, aligning with previous findings on effective loss functions in image recovery [3]. Additionally, the sum of sixth powers effectively regulates pixel intensity distribution, improving image stability and detail retention, as supported by Osher *et al.* [10]. Total variation regularization further enhances spatial consistency by suppressing noise while preserving sharp edges, a technique widely adopted in image processing since its introduction by Rudin *et al.* [9]. The combination of these constraints ensures improved reconstruction fidelity and robustness in recovering pruned features.

Unlike autoencoder-based methods that require extensive pre-training on large datasets, the proposed approach leverages a direct optimization strategy, eliminating dependency on dataset-specific models. This design enhances computational efficiency and adaptability across various pruning conditions, aligning with recent trends in lightweight neural network recovery frameworks suitable for IoT and edge computing paradigms. Such an approach proves advantageous when compared to generative adversarial networks, which often require significant training resources and may face instability issues during the learning process [13]. By integrating multiple constraints in a unified framework, the proposed method effectively reconstructs key image features from heavily pruned data while maintaining efficiency. This targeted approach offers a robust solution for image recovery in applications such as model compression, privacy-preserving networks, and resource-constrained environments. The method's capacity to address aggressive pruning scenarios highlights its practical relevance in real-world computer vision tasks.

4 PERFORMANCE EVALUATION

This section assesses the proposed method's ability, covering experimental setup, visual analysis, and quantitative evaluation of pruning effects.

4.1 Experimental setup

Experiments were conducted on three benchmark datasets: MNIST (grayscale handwritten digits), CIFAR-100 (color object images), and LFWPeople (facial photographs). A LeNet-based CNN was trained on each dataset, and pruned feature vectors were extracted from its penultimate layer. Neuron pruning was performed using a magnitude-based approach, where neurons with the lowest activation values were progressively removed. The pruning ratio ranged from 0.0, indicating no pruning, to 0.9, representing the removal of 90 percent of neurons, with increments of 0.1. It should be noted that the no pruning condition (i.e., a pruning ratio of 0.0), where images are reconstructed from the full set of features, is comparable to scenarios examined in foundational studies on gradient inversion attacks, such as the work of Zhu *et al.* [14]. This setting serves as a standard baseline for evaluating attacks on complete, unpruned representations, and has been similarly explored in prior work, including [15]. Optimization was performed using the Adam optimizer with a learning rate of 0.01 over 2,000 iterations, starting from dummy images uniformly initialized in the range $[0, 1]$. Hyperparameters α were tuned via grid search: For MNIST and CIFAR-100 datasets, values were $\alpha_{L2} = 10^3$, $\alpha_{sixnorm} = 10^{-7}$, $\alpha_{TV} = 10^{-4}$, while for the LFW dataset, the corresponding values were $\alpha_{L2} = 10^3$, $\alpha_{sixnorm} = 10^{-9}$, $\alpha_{TV} = 5.10^{-7}$. These hyperparameters were selected to balance data fidelity, regularization constraints, and smoothness, ensuring optimal reconstruction performance across all datasets. As an illustrative example, the tuning process for the CIFAR-100 dataset is detailed in the Appendix. Performance was assessed using 20 randomly sampled images per dataset and three metrics: MSE for pixel-wise accuracy, SSIM for structural similarity, and PSNR for perceptual quality.

4.2 Visualization of data reconstruction

The qualitative effectiveness of the proposed method was assessed through visual reconstruction on three datasets: MNIST, CIFAR-100, and LFWPeople, using a representative pruning ratio of 0.3. Figure 3 illustrates the iterative reconstruction process, starting from a randomly initialized image and progressing through multiple optimization steps to the final output at 2,000 iterations. The iterative process demonstrated adaptability to varying image complexities and the capacity to recover semantically meaningful features despite pruning-induced data loss.

For MNIST, digit structures begin to emerge within the first 100 iterations, with clear contours forming early in the process. By iteration 2,000, the reconstructed digits closely resemble the ground truth, accurately preserving stroke patterns and spatial layout. In the CIFAR-100 dataset, the method effectively reconstructs color-rich images, with object boundaries and textures becoming recognizable in early stages and progressively refined over time. Although slight smoothing is observed in high-frequency regions, color gradients and structural details are largely retained. On the LFWPeople dataset, the method successfully reconstructs facial features, such as eye contours and expressions, with increasing clarity. The outputs maintain structural coherence and spatial fidelity despite the loss of some feature-level information due to pruning.

Across all datasets, four consistent trends underscore the method's robustness. Critical visual features emerge early in the optimization process, often within 100 to 500 iterations, and refine progressively. Fine-grained details, including digit strokes in MNIST, object textures in CIFAR-100, and facial landmarks in LFWPeople, are preserved with high precision. High-frequency noise and artifacts diminish significantly in later iterations, particularly beyond 1,500 iterations, ensuring perceptual clarity. Furthermore, the method maintains sharp object boundaries and faithful color reproduction, crucial for applications requiring accurate visual recovery.

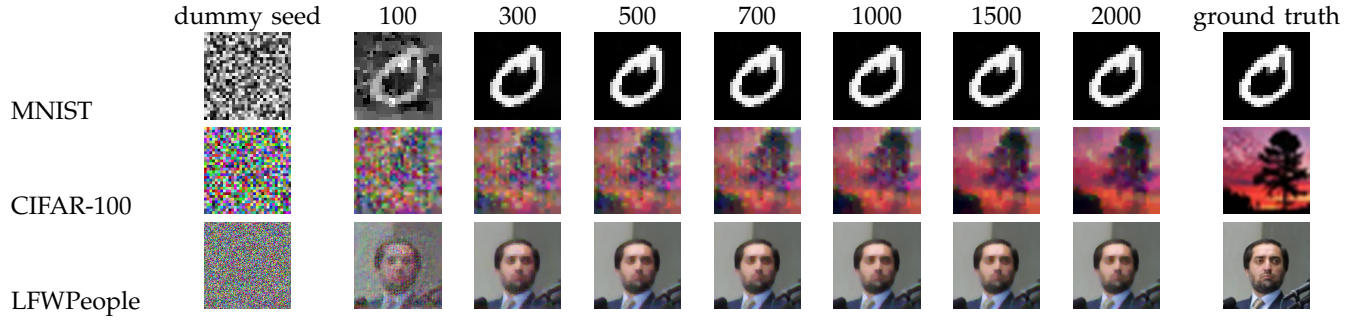

Figure 2. Progression of image generation across different datasets and iterations using L1 distance with a pruning ratio of 0.3.

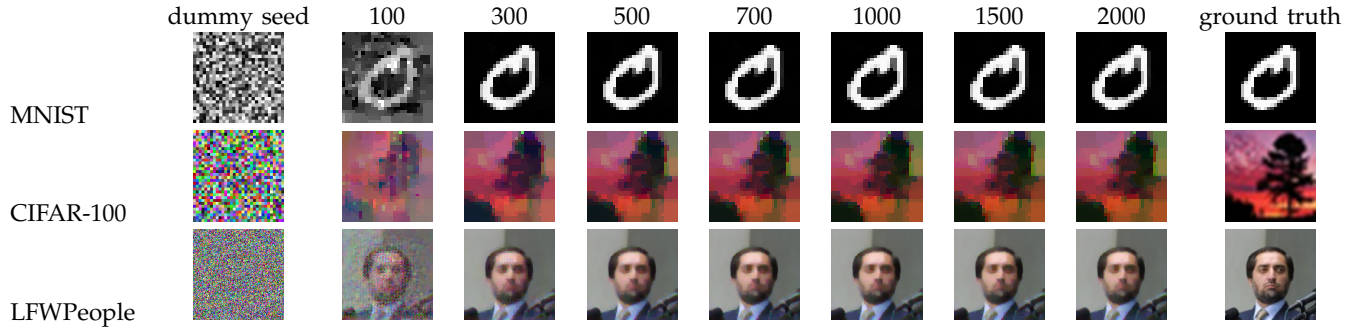

Figure 3. Progression of image generation across different datasets and iterations using L2 distance with a pruning ratio of 0.3.

These results validate the efficacy of the multi-constraint loss framework, which integrates L2 feature alignment, sixth-power intensity regularization, and total variation-based spatial smoothing. The method consistently achieves high-quality reconstructions across datasets within 2,000 iterations, demonstrating robustness to pruning-induced information loss and adaptability to varying image complexities.

4.3 Impact of pruning ratios

This study examines how different pruning levels affect image reconstruction quality across the datasets. To provide a statistically robust evaluation of the proposed method, experiments were conducted across a range of pruning ratios. Figures 4 and 5 summarize the performance metrics for reconstruction using L1 and L2 distance, respectively, on the MNIST, CIFAR-100, and LFWPeople datasets. For each pruning ratio, the results were averaged over 20 distinct test samples. In the figures, each data point represents the mean of the performance metric (MSE, SSIM, and PSNR), while the corresponding error bars depict the standard deviation. This visualization allows for a clear assessment of both the performance and the consistency of our method. The results demonstrate that the proposed approach maintains strong reconstruction quality (low MSE, high SSIM/PSNR) with low variance, even as the information loss from pruning increases. This underscores the robustness and reliability of the proposed multi-constraint loss function. Notably, Figure 4 presents the results of the L1-based approach, which is also shown visually in Figure 2 while Figure 5 illustrates the L2-based method described in this study. The use of an L2 distance term in the loss function consistently yields slight improvements in terms of MSE, SSIM, and PSNR values across most pruning ratios and datasets. This observation supports its integration into the proposed multi-constraint loss framework to enhance reconstruction fidelity. Higher MSE reflects reduced pixel-level accuracy, while declining SSIM and PSNR values indicate diminished structural and perceptual fidelity. Performance degradation exhibits nonlinear behavior. At moderate pruning levels, specifically in the range from 0.0 to 0.6, mean squared error gradually increases by approximately 5 to 30%, while structural similarity, and peak signal-to-noise ratio decrease by around 3 to 15%, and 1 to 5 dB, respectively. These trends suggest that some lost information remains recoverable. Beyond a critical threshold, ranging from 0.7 to 0.9, metrics deteriorate abruptly, with MSE escalating by 80 to 120% and PSNR collapsing below 22 dB for complex datasets like LFWPeople. This signifies irreversible feature loss due to excessive pruning.

Dataset complexity profoundly influences robustness to pruning. MNIST, characterized by simple grayscale digit structures, demonstrates superior resilience, maintaining PSNR above 28 dB even at a 0.7 pruning ratio. In contrast, CIFAR-100 and LFWPeople, which feature intricate color variations and textures, experience accelerated quality degradation. For instance, LFWPeople's PSNR drops to 21.5 dB under 90% pruning, underscoring the challenge of reconstructing nuanced facial features from highly pruned features.

These findings highlight a fundamental trade-off between pruning efficiency and reconstruction fidelity. Notably, as shown in Table I, the average runtime per sample remains stable across different datasets and pruning ratios. Specifically, the runtime ranges from 11.5 to 11.9 seconds across datasets, with MNIST achieving 11.549 seconds, CIFAR-100 achieving 11.815 seconds, and LFWPeople achieving 11.935 seconds.

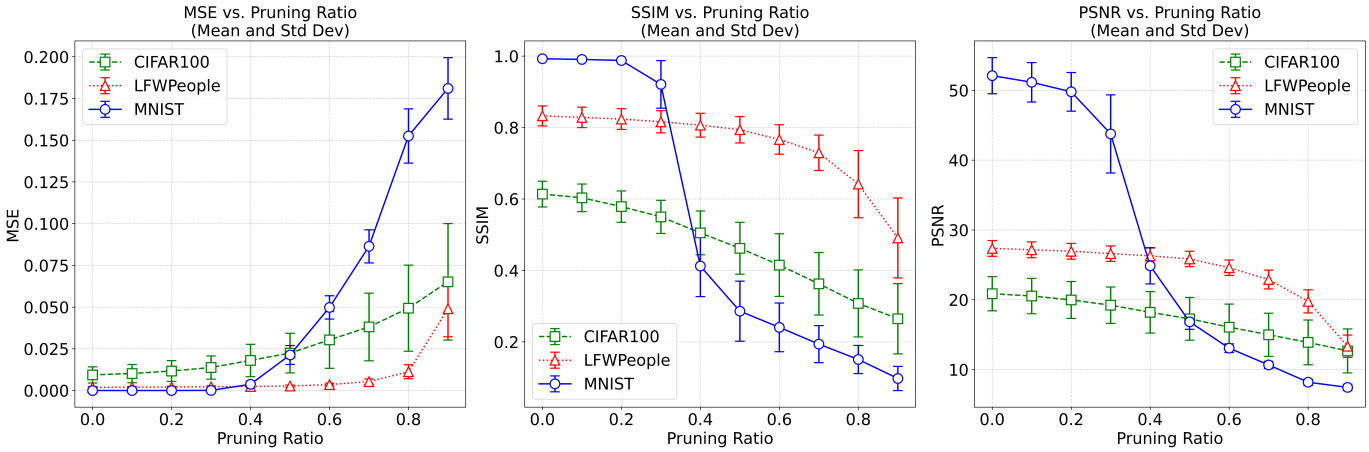

Figure 4. Performance metrics using L1 distance versus pruning ratios for the datasets.

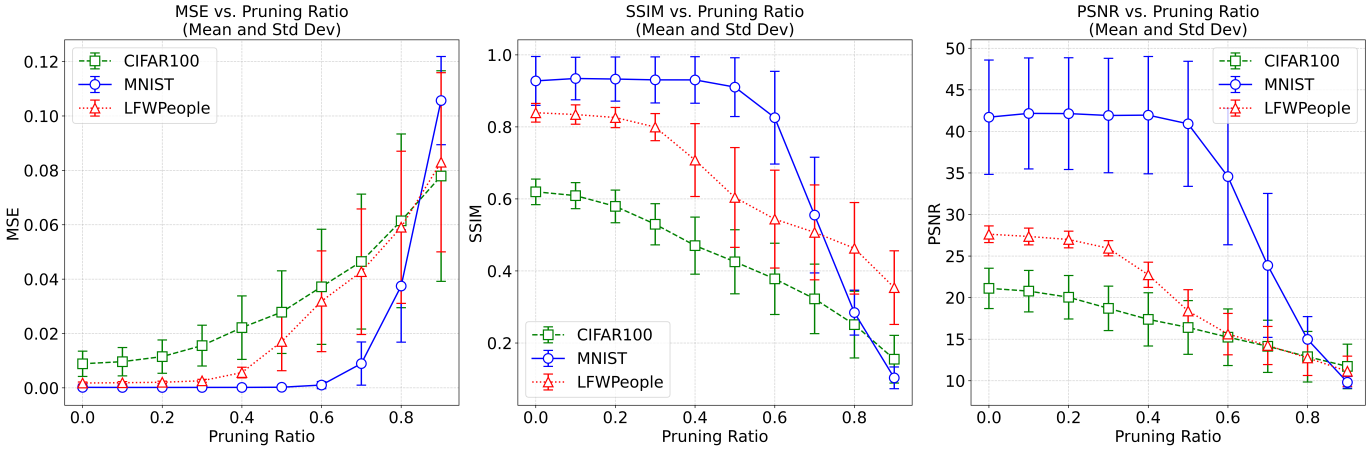

Figure 5. Performance metrics using L2 distance versus pruning ratios for the datasets.

This consistency indicates that the computational cost of the method is largely independent of the input data complexity or the degree of feature pruning, making this approach feasible and efficient for various applications. While aggressive pruning reduces computational demands, it imposes strict limits on recoverable image quality, especially for datasets with high visual complexity. The proposed multi-constraint framework achieves high SSIM and PSNR values, thereby validating its effectiveness in mitigating information loss across diverse pruning scenarios. This balance positions the method as a practical solution for applications, such as edge AI in IoT systems, that prioritize both computational efficiency and high-quality image recovery.

Table I
Computational efficiency across datasets on the LeNet model

Datasets	Image Resolution	Real-time tasks	Feature Size	Runtime (s)
MNIST	28×28	Handwritten digits	588	11.549
CIFAR-100	32×32	Object recognition	768	11.815
LFWPeople	250×250	Facial recognition	47628	11.935

A further investigation into the data reconstruction capability of the proposed method was conducted using the ResNet-14 model. Some images from CIFAR-100 are selected for the experiment. As can be seen in Figure 6, images can be reconstructed with clear patterns like original images. The collected results show that our method can recover training inputs even when using pruned representations of images.

5 CONCLUSION

This study presents a multi-constraint loss function for image recovery from pruned features. The proposed approach balances pruning efficiency and reconstruction quality, enabling accurate image recovery while preserving essential visual features. The results demonstrate the effectiveness of this method under varying pruning conditions. However, certain limitations remain, particularly at high pruning ratios, where recovery accuracy declines, and sensitivity to perturbations increases. Future research could focus on improving robustness to transmission errors, optimizing adaptive parameter tuning, and integrating advanced deep learning models to enhance recovery accuracy and efficiency.

The presented research provides valuable insights into inverse problems in deep learning by introducing a notably flexible approach to image recovery. The method is particularly beneficial in scenarios with limited training data, such as privacy-sensitive environments or resource-constrained settings like those found

Figure 6. Performance of the proposed method with the ResNet architecture.

in distributed IoT deployments. While requiring further validation and domain-specific investigation, there is potential for this approach to be explored in applications such as medical imaging, where efficient feature and information recovery are crucial, and where understanding the impact of feature pruning on diagnostic information could be valuable. Additionally, it may assist in security risk assessments by demonstrating the potential to recover key image features from pruned data, thereby highlighting privacy vulnerabilities in cyber-physical systems.

References

[1] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *science*, vol. 313, no. 5786, pp. 504–507, 2006.

[2] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[3] A. Mahendran and A. Vedaldi, "Understanding deep image representations by inverting them," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 5188–5196.

[4] S. Han, J. Pool, J. Tran, and W. Dally, "Learning both weights and connections for efficient neural network," *Advances in neural information processing systems*, vol. 28, 2015.

[5] Y. He and L. Xiao, "Structured pruning for deep convolutional neural networks: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 46, pp. 2900–2919, 2023.

[6] P. Molchanov, S. Tyree, T. Karras, T. Aila, and J. Kautz, "Pruning convolutional neural networks for resource efficient inference," *arXiv preprint arXiv:1611.06440*, 2016.

[7] M. Fredrikson, S. Jha, and T. Ristenpart, "Model inversion attacks that exploit confidence information and basic countermeasures," *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, pp. 1322–1333, 2015.

[8] Z. Liu, M. Sun, T. Zhou, G. Huang, and T. Darrell, "Rethinking the value of network pruning," *ArXiv*, vol. abs/1810.05270, 2018.

[9] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D: nonlinear phenomena*, vol. 60, no. 1-4, pp. 259–268, 1992.

[10] S. Osher, M. Burger, D. Goldfarb, J. Xu, and W. Yin, "An iterative regularization method for total variation-based image restoration," *Multiscale Modeling and Simulation*, vol. 4, no. 2, pp. 460–489, 2005.

[11] C. Nash, N. Kushman, and C. K. Williams, "Inverting supervised representations with autoregressive neural density models," in *Proceedings of the 22nd International Conference on Artificial Intelligence and Statistics*. PMLR, 2019, pp. 1620–1629.

[12] Z. Yang, J. Zhang, E.-C. Chang, and Z. Liang, "Neural network inversion in adversarial setting via background knowledge alignment," in *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, 2019, pp. 225–240.

[13] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," *Communications of the ACM*, vol. 63, no. 11, pp. 139–144, 2020.

[14] L. Zhu, Z. Liu, and S. Han, "Deep leakage from gradients," *Advances in neural information processing systems*, vol. 32, 2019.

[15] T.-N. Dao and H. Lee, "Encgradinversion: Image encoding and gradient inversion-based batch attack in federated learning," *IEEE Internet of Things Journal*, vol. 12, no. 4, pp. 3858–3870, 2024.

Van-Phi Hoang received a Bachelor of Engineering in Electrical and Electronic Engineering from Le Quy Don Technical University, Vietnam, in 2017. He is currently pursuing a Master of Engineering degree in Electronic Engineering at Le Quy Don Technical University, Vietnam. His research interests include federated learning, network security, and IoT systems.

Thi-Nga Dao received a B.S. degree in Electrical and Communication Engineering from the Le Quy Don Technical University, Vietnam in 2013, an M.S. degree in Computer Engineering from University of Ulsan in 2016, and a Ph.D. degree in Computer Engineering from University of Ulsan, South Korea in 2019. She was a postdoctoral fellow with the Computer Science and Engineering Department, Ewha Womans University. Since July 2019, she has been a lecturer in the Faculty of Radio-Electronic Engineering, Le Quy Don Technical University, Hanoi, Vietnam. Her research interests include security in federated learning, machine learning-based applications in network security, network intrusion detection and prevention systems.

APPENDIX

This appendix details the systematic grid search performed to determine the optimal hyperparameters α_{L2}, $\alpha_{sixnorm}$, and α_{TV} for the CIFAR-100 dataset. Metrics MSE, SSIM, and PSNR are employed to find the optimal hyperparameters. The results of this process are summarized below:

Step 1: Optimize the α_{L2} coefficient

In this step, we adjust the value α_{L2} within the set [10^2, 10^3, 10^4]. The other two parameters are fixed: $\alpha_{sixnorm} = 10^{-7}$ and $\alpha_{TV} = 10^{-4}$. The experimental results are presented in Table II. Based on the experimental results, $\alpha_{\mathrm{L2}} = 10^3$ yields the best results, demonstrated by an optimal balance across MSE, SSIM, and PSNR metrics.

Step 2: Optimize the $\alpha_{sixnorm}$ coefficient

In this step, we adjust the $\alpha_{sixnorm}$ value within the set [10^{-6}, 10^{-7}, 10^{-8}]. The other two parameters are fixed: $\alpha_{L2} = 10^3$ and $\alpha_{TV} = 10^{-4}$. The experimental results are presented in Table III. Based on the experimental results, $\alpha_{sixnorm} = 10^{-7}$ yields the best results, demonstrated by an optimal balance across MSE, SSIM, and PSNR metrics.

Step 3: Optimize the α_{TV} coefficient

In this step, we adjust the α_{TV} value within the set [10^{-3}, 10^{-4}, 10^{-5}]. The other two parameters are fixed: $\alpha_{L2} = 10^3$ and $\alpha_{sixnorm} = 10^{-7}$. The experimental results are presented in Table IV. Based on the experimental results, $\alpha_{TV} = 10^{-4}$ yields the best results, demonstrated by an optimal balance across MSE, SSIM, and PSNR metrics.

Through systematic experimentation on the CIFAR-100 dataset, the optimal parameter set was identified as $\alpha_{L2} = 10^3$, $\alpha_{sixnorm} = 10^{-7}$, and $\alpha_{TV} = 10^{-4}$, which achieves a robust balance among the evaluation metrics (MSE, SSIM, and PSNR), with its effectiveness demonstrated by the results presented in Tables II, III, and IV.

Table II
Evaluation metric (MSE, SSIM, and PSNR) across pruning ratios with different α_{L2} values

Metrics		MSE			SSIM			PSNR		
α_{L2}		10^2	10^3	10^4	10^2	10^3	10^4	10^2	10^3	10^4
Pruning Ratios	0	0.01977	0.00646	**0.00438**	0.28084	0.71023	**0.80127**	17.03968	21.89671	**23.58292**
	0.1	0.01917	0.00617	**0.00443**	0.29243	0.72542	**0.78765**	17.17418	22.09693	**23.54023**
	0.2	0.01817	0.00587	**0.00526**	0.30695	0.73737	**0.74699**	17.40751	22.31720	**22.78693**
	0.3	0.01732	0.00599	**0.00593**	0.32579	**0.74087**	0.71179	17.61335	22.22286	**22.26769**
	0.4	0.01692	**0.00629**	0.00780	0.32271	**0.72766**	0.63284	17.71657	**22.01028**	21.07911
	0.5	0.01589	**0.00699**	0.00928	0.34897	**0.71710**	0.59176	17.99007	**21.55603**	20.32503
	0.6	0.01502	**0.00819**	0.01297	0.37478	**0.64537**	0.41745	18.23325	**20.86679**	18.87135
	0.7	0.01429	**0.00924**	0.01458	0.41676	**0.59048**	0.41548	18.44948	**20.34145**	18.36176
	0.8	0.01306	**0.01141**	0.02249	**0.45313**	0.44769	0.19775	18.84055	**19.42673**	16.47938
	0.9	0.01450	**0.01806**	0.03679	**0.40301**	0.27491	0.04310	**18.3867**	17.43318	14.34270

Table III
Evaluation metrics (MSE, SSIM, and PSNR) across pruning ratios with different $\alpha_{sixnorm}$ values

Metrics		MSE			SSIM			PSNR		
$\alpha_{sixnorm}$		10^{-6}	10^{-7}	10^{-8}	10^{-6}	10^{-7}	10^{-8}	10^{-6}	10^{-7}	10^{-8}
Pruning Ratios	0	0.01488	0.00646	**0.00645**	0.69062	0.71023	**0.71155**	18.27299	21.89671	**21.90682**
	0.1	0.01413	**0.00617**	0.00618	0.69673	**0.72542**	0.72540	18.50011	**22.09693**	22.09091
	0.2	0.01496	**0.00587**	0.00588	0.68851	**0.73737**	0.73675	18.25095	**22.3172**	22.30419
	0.3	0.01407	**0.00599**	0.00601	0.69844	0.74087	**0.74099**	18.51692	**22.22286**	22.21410
	0.4	0.01432	**0.00629**	0.00632	0.68980	**0.72766**	0.72723	18.44140	**22.01028**	21.99596
	0.5	0.01565	**0.00699**	0.00701	0.66715	0.71710	**0.71726**	18.05527	**21.55603**	21.54554
	0.6	0.01870	**0.00819**	0.00821	0.60850	0.64537	**0.64550**	17.28153	**20.86679**	20.85906
	0.7	0.02457	**0.00924**	0.00925	0.56638	0.59048	**0.59093**	16.09623	**20.34145**	20.33901
	0.8	0.02805	**0.01141**	0.01154	**0.50102**	0.44769	0.44106	15.52082	**19.42673**	19.37803
	0.9	0.04543	0.01806	**0.01796**	**0.29414**	0.27491	0.27807	13.42620	17.43318	**17.45678**

Table IV
Evaluation metric (MSE, SSIM, and PSNR) across pruning ratios with different α_{TV} values

Metrics		MSE			SSIM			PSNR		
α_{TV}		10^{-3}	10^{-4}	10^{-5}	10^{-3}	10^{-4}	10^{-5}	10^{-3}	10^{-4}	10^{-5}
Pruning Ratios	0	0.01975	0.00646	**0.00438**	0.28275	0.71023	**0.80041**	17.04509	21.89671	**23.58492**
	0.1	0.01915	0.00617	**0.00443**	0.29485	0.72542	**0.78654**	17.1779	22.09693	**23.53643**
	0.2	0.01818	0.00587	**0.00528**	0.30793	0.73737	**0.74570**	17.40364	22.31720	**22.77572**
	0.3	0.01734	0.00599	**0.00594**	0.32706	**0.74087**	0.71064	17.61054	22.22286	**22.26297**
	0.4	0.01694	**0.00629**	0.00774	0.32342	**0.72766**	0.63383	17.71136	**22.01028**	21.11119
	0.5	0.01589	**0.00699**	0.00929	0.35045	**0.71710**	0.59617	17.98756	**21.55603**	20.31924
	0.6	0.01503	**0.00819**	0.01302	0.37600	**0.64537**	0.40982	18.23019	**20.86679**	18.85362
	0.7	0.01434	**0.00924**	0.01436	0.41844	**0.59048**	0.42573	18.43308	**20.34145**	18.42738
	0.8	0.01311	**0.01141**	0.02231	**0.45592**	0.44769	0.19318	18.82423	**19.42673**	16.51504
	0.9	0.01450	**0.01806**	0.03710	**0.40387**	0.27491	0.04613	**18.38535**	17.43318	14.30677