Exploring Feature Complexity and Regularization in Digit Classification: A Neurocomputational Approach

Christine Wang
Department of Systems Design Engineering
University of Waterloo
Waterloo, ON N2L 3G1, Canada
c752wang@uwaterloo.ca

Jamie Kang
Department of Systems Design Engineering
University of Waterloo
Waterloo, ON N2L 3G1, Canada
j228kang@uwaterloo.ca

Abstract

We investigate how biologically inspired input transformations and model complexity affect classification performance in a simplified vision task. Using the MNIST dataset, we evaluate three model pipelines: a logistic regression baseline trained on raw pixel inputs, a feature-engineered classifier using principal component analysis (PCA) and Sobel edge detection, and a nonlinear regression model leveraging polynomial expansion with regularization. Test accuracy and confusion matrices were used to quantify performance across each setup. The raw pixel model achieved the highest accuracy (92.54%), while PCA and Sobel features yielded lower but competitive results depending on dimensionality and feature type. Polynomial models demonstrated improved performance with degree 2 expansions, with Ridge regularization achieving stable generalization. Our findings highlight the trade-offs between biologically plausible representations, model capacity, and generalization, offering insight into how simplified neural coding strategies influence classification in artificial systems.

1 Introduction

Understanding how biological systems process and classify visual information has been a long-standing objective in computational neuroscience. From early stages in the retina to the primary visual cortex (V1), neural circuits extract meaningful features—such as edges, orientations, and spatial structure—from visual stimuli to enable perception and behaviour [1]. These biological processes have inspired a range of artificial models, which attempt to capture the essence of neural coding and generalization through tractable computational systems.

In this project, we investigate how various input transformations and model complexities affect classification performance in a biologically inspired digit recognition task. We use the MNIST dataset of handwritten digits–a standard benchmark in visual classification—as a simplified proxy for understanding how sensory inputs might be represented and processed by early visual areas [2]. While the task itself is not biologically realistic, the manipulations we explore reflect neurocomputational principles such as dimensionality reduction and non-linear feature interaction [3, 4].

To simulate and evaluate these processes, we implement and compare three model pipelines:

1. A baseline lienar classifier trained on raw pixel inputs;
2. A feature transofrmation pipeling using biologically motivated preprocessing such as PCA and edge detection;
3. A nonlinear regression model using polynomial expansion and regularization to test the role of model complexity in performance and generalization.

For each approach, we vary critical parameters–including the number of PCA components and the degree of polynomial expansion–and measure classification accuracy and robustness. We further evaluate overfitting behaviour and explore how regularization strength impacts model performance in high-dimensional settings.

Our aim is to understand the trade-offs between biological realism, model complexity, and generalization performance in a simplified vision task. By systematically varying both input representations and model structure, we gain insight into how neural-inspired strategies influence classification accuracy—mirroring similar challaneges faced in designing both brain-like system and practical machine learning models.

2 Methods

2.1 Data Preprocessing

We used the MNIST dataset of handwritten digits, which contains 60,000 training images and 10,000 test images, each formatted as a 28×28 greyscale pixel array. All images were flattened into 784-dimensional vectors and normalized to a pixel intensity range of [0, 1] using NumPy. This flattened format was used as input for all models across the experiment. When training time or feature dimensionality posed computational constraints, we used stratified random subsets of 5,000 training and 1,000 test images.

2.2 Baseline Logistic Regression

To establish a performance benchmark, we trained a multinomial logistic regression classifier using the raw pixel inputs and no feature engineering. The model was implemented using scikit-learn's `LogisticRegression` with the `saga` solver and `multi_class='multinomial'` setting. The maximum number of iterations was set to 300 to ensure convergence across all classes, and all training samples were used (60,000 total). Predictions were made on the full 10,000-image test set.

The performance of the model was evaluated using classification accuracy. We also computed a confusion matrix and visualized 10 representative misclassified digits to qualitatively assess which digits were most commonly confused. These results served as the baseline for comparing future experiments with feature transformations and higher-capacity models.

2.3 Feature Transformation via PCA and Edge Detection

We implemented principal component analysis (PCA) and Sobel edge detection on the MNIST data before training to explore the effect of biologically motivated input transformations on classification performance.

PCA was used to reduce the dimensionality of the 784-dimensional flattened input vectors. This linear transformation preserves the directions of greatest variance, which is similar to how early visual neurons encode dominant spatial patterns. We used `scikit-learn`'s `PCA` implementation and evaluated multiple settings for the number of components. We selected PCA component counts of 10, 20, 50, and 100 to represent a range of biologically inspired compression levels. Lower values simulate severe bottlenecks in early vision, while higher values retain more visual detail. This range allowed us to evaluate how much dimensionality is required to preserve discriminative structure for digit classification and assess the trade-off between compression and performance. For each configuration, the transformed training and test data were fed into a Ridge regression classifier. We also visualized reconstructed images from PCA-transformed data using the inverse transform to qualitatively assess the preservation of digit structure.

Sobel edge detection was used to extract spatial gradients that approximate early edge-selective neural responses (e.g., V1 simple cells). Each image was filtered using horizontal and vertical Sobel operators, and the gradient magnitude was computed to produce a single edge-enhanced image. These images were then flattened and used as input features. Then, we once again trained a Ridge regression model on the Sobel features to compare their discriminative power against raw pixel inputs and PCA representations.

2.4 Polynomial Expansion and Regularization

To assess the effect of model complexity and regularization on performance, we expanded the input features using polynomial transformations of degree $d \in \{1, 2\}$. These transformations were applied using scikit-learn's `PolynomialFeatures` with `include_bias=False`. The degree 2 expansion increased the feature space from 784 to over 300,000 dimensions. To manage the computational cost, we trained on a stratified subset of 5,000 training samples and evaluated on 1,000 test samples.

For each degree, we evaluated both Ridge (L2) and Lasso (L1) regularized linear models using scikit-learn's `Ridge` and `Lasso` implementations. The models were configured using `max_iter=1000` and were trained via a pipeline that automatically applied polynomial expansion followed by regularized regression. We tested three regularization strengths: $\alpha \in \{0.01, 0.1, 1.0\}$. For each configuration, we logged training and test accuracy and stored results in a structured DataFrame.

Finally, we generated a line plot showing test accuracy as a function of α, grouped by degree and regularization type. The resulting trends enabled a direct comparison of generalization performance between low-capacity and high-capacity models and highlighted the role of regularization in preventing overfitting in high-dimensional settings.

3 Results

3.1 Baseline Logistic Regression

The baseline multinomial logistic regression classifier trained on raw MNIST pixel inputs achieved a test accuracy of 92.54%. A confusion matrix was generated to visualize class-specific performance, shown in Figure 1. The classifier performed well across most digits but frequently confused digit pairs with similar curvature (e.g., 5 and 6, 3 and 5, 9 and 4).

In addition, we visualized ten representative misclassified digits (Figure 2). These examples highlight ambiguities in handwriting that contribute to model errors, such as unclear digit formation or overlapping strokes.

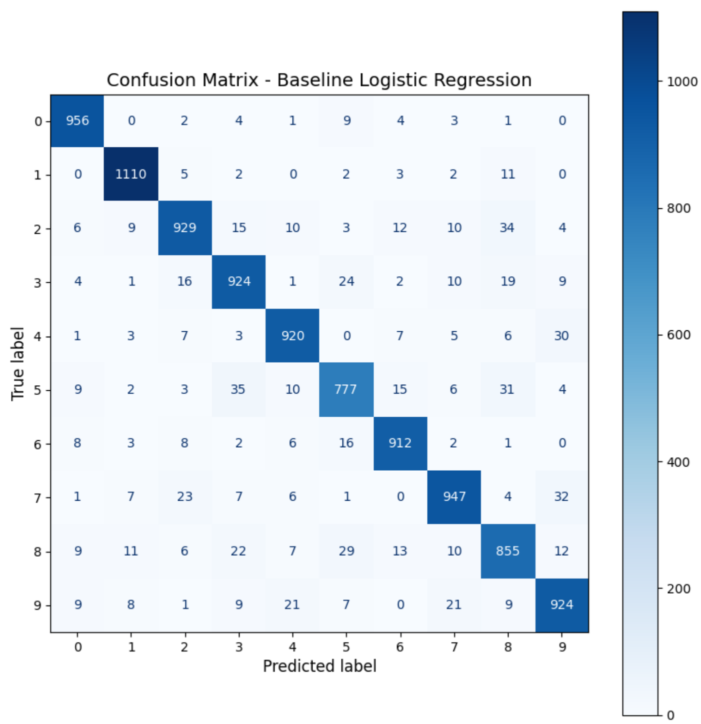

Figure 1: Confusion matrix for baseline logistic regression model. Most predictions fall along the diagonal, with notable confusion between digits with similar shapes.

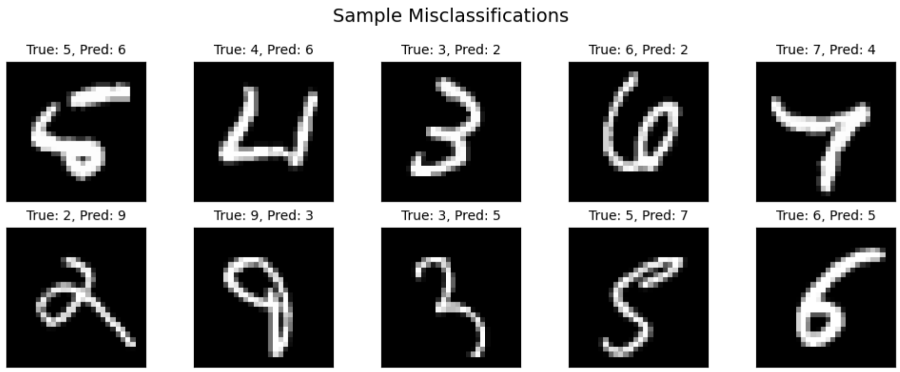

Figure 2: Examples of misclassified digits from the baseline model. Each image shows the true label and the model's prediction. Many errors involve ambiguous handwriting.

3.2 Feature Engineering

We evaluated test accuracy and confusion matrices for PCA-transformed data with 10, 20, 50, and 100 components, and Sobel-filtered images. Accuracy values are listed in Table 1 along with the baseline. The raw pixel baseline achieved the highest accuracy at 92.54%. PCA-based models showed improved performance as dimensionality increased, ranging from 71.74% at 10 components to 86.11% at 100 components. The Sobel feature classifier reached 85.01% accuracy.

Table 1: Test Accuracy by Input Representation

Input Type	Dimensionality	Accuracy (%)
Raw Pixels	784	92.54
PCA (10 components)	10	71.74
PCA (20 components)	20	81.12
PCA (50 components)	50	85.25
PCA (100 components)	100	86.11
Sobel Filter	784	85.01

Figure 3 presents PCA reconstructions of a sample digit, illustrating the effect of dimensionality on input fidelity. Confusion matrices for the PCA-based models are shown in Figure 4. Classification performance improved with increasing component count, particularly for commonly confused digits. Figure 5 shows the confusion matrix for Sobel-based classification. Figure 6 displays Sobel-filtered versions of several example digits.

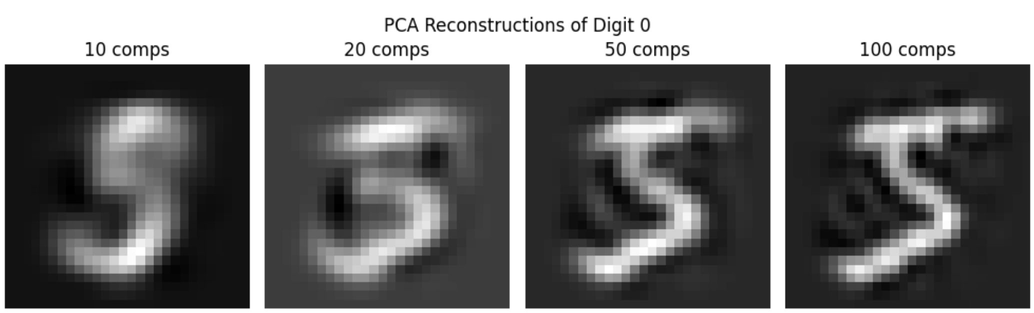

Figure 3: Digit 0 (index 0) reconstructed from PCA-transformed data using 10, 20, 50, and 100 components.

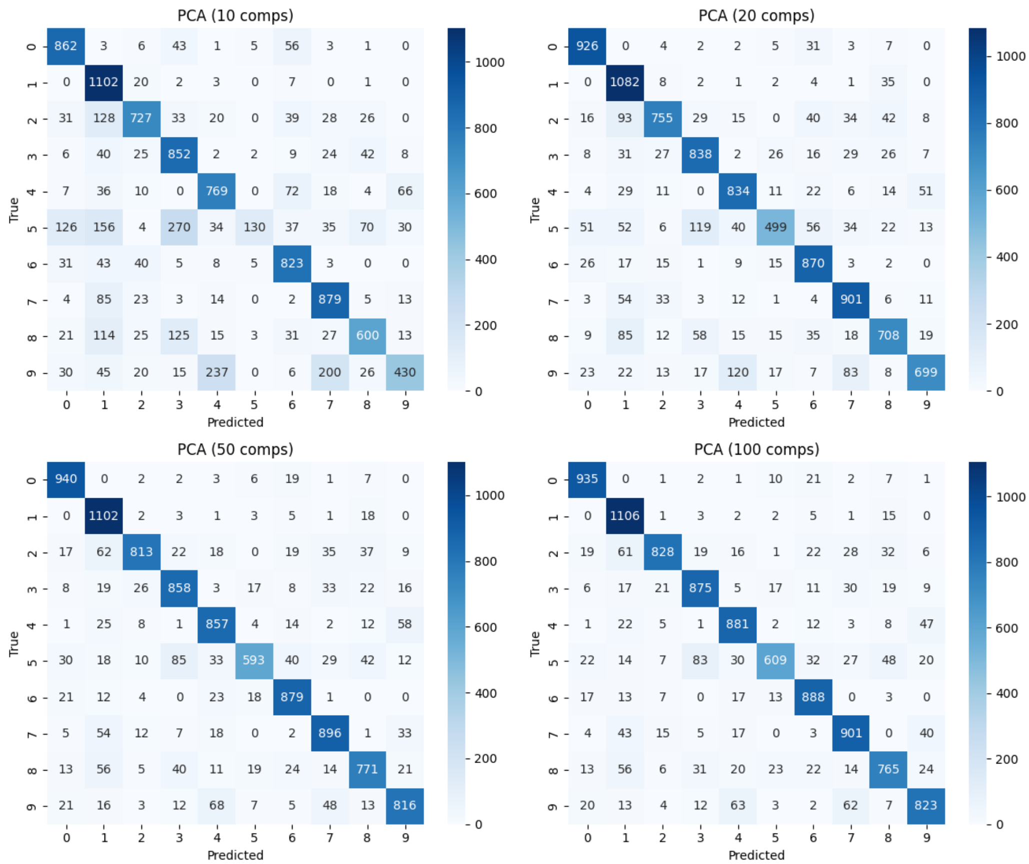

Figure 4: Confusion matrices for Ridge regression classification using PCA-transformed inputs with 10, 20, 50, and 100 components.

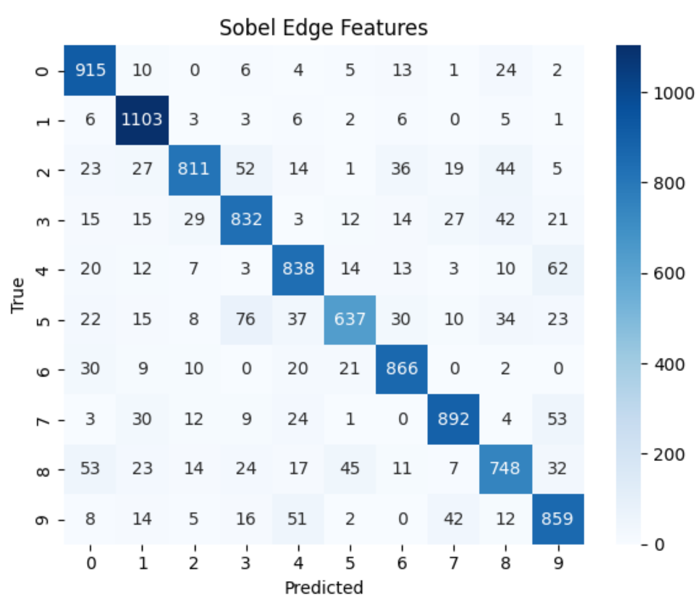

Figure 5: Confusion matrix for Ridge regression classification using Sobel-filtered input images.

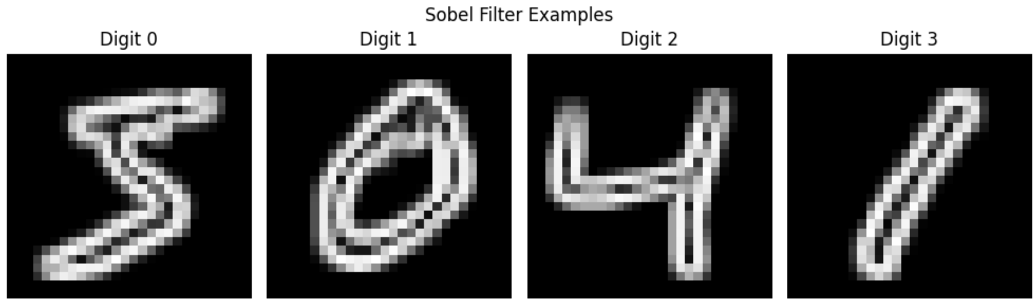

Figure 6: Sobel edge outputs for digits 0 through 3. Each plot shows the gradient magnitude of the input digit after applying the Sobel operator along both axes.

3.3 Polynomial Expansion and Regularization

We evaluated the impact of model complexity and regularization strength by training regression models on polynomial expansions of degree 1 and 2 using Ridge (L2) and Lasso (L1) regularization. Each model was trained on 5,000 samples and evaluated on 1,000 test samples. Table 2 summarizes the results. Figure 7 shows test accuracy as a function of α. Ridge models showed strong and stable generalization, while Lasso models were highly sensitive to regularization strength and prone to underfitting at high α values.

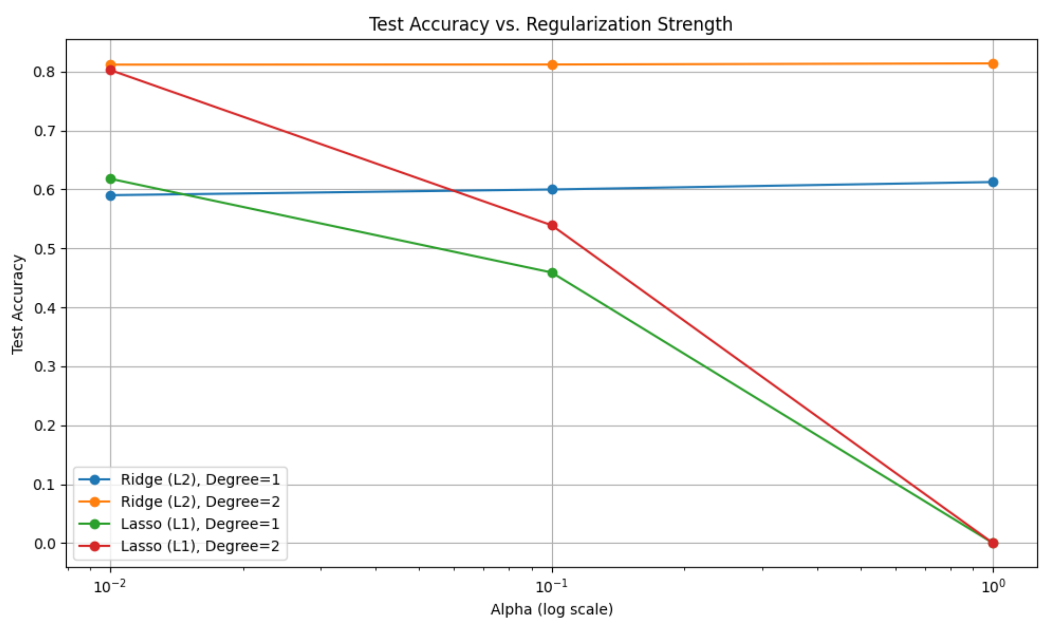

Figure 7: Test accuracy as a function of α for Ridge and Lasso models with degree 1 and 2 polynomial features. Ridge models performed consistently well, while Lasso models degraded at higher regularization strengths.

Table 2: Test accuracy (descending) of polynomial models with Ridge and Lasso regularization. Degree 2 models consistently outperformed degree 1, with Ridge models achieving the highest accuracy.

Degree	Regularizer	α	Train Acc	Test Acc
2	Ridge (L2)	1.00	0.99999	0.81362
2	Ridge (L2)	0.10	1.00000	0.81174
2	Ridge (L2)	0.01	1.00000	0.81152
2	Lasso (L1)	0.01	0.82564	0.80240
1	Lasso (L1)	0.01	0.59364	0.61800
1	Ridge (L2)	1.00	0.65461	0.61243
1	Ridge (L2)	0.10	0.65850	0.59977
1	Ridge (L2)	0.01	0.65956	0.59013
2	Lasso (L1)	0.10	0.50599	0.53918
1	Lasso (L1)	0.10	0.42271	0.45900
1	Lasso (L1)	1.00	0.00000	-0.00002
2	Lasso (L1)	1.00	0.00000	-0.00002

4 Discussion

The classification performance observed across our models provides insight into how different levels of model complexity, input representation, and regularization affect generalization in a simplified visual recognition task. While the baseline logistic regression model performed well with linear decision boundaries, its misclassification of visually similar digits highlights the limitations of linear models in separating overlapping representations in pixel space. This reflects limitations seen in early visual areas such as the retina and lateral geniculate nucleus (LGN), where receptive fields respond to simple spatial features but lack high-level selectivity [5].

Our results after feature engineering processes revealed clear trends in how dimensionality and biologically motivated transformations impact classification accuracy. The PCA-based models demonstrated a strong dependence on the number of retained components (Table 1). At 10 components, test accuracy dropped to 71.74%, indicating that excessive compression discarded important discriminative information. Increasing to 20 and 50 components improved accuracy substantially (81.12% and 85.25% respectively), with 100 components reaching 86.11%. This trend shows that it is necessary to preserve sufficient variance in order to maintain class separability, even when using linear classifiers. Compared to the raw pixel baseline at 92.54%, all PCA models underperformed, but the gap narrowed as dimensionality increased. The loss of detail at low component counts (e.g., PCA-10) may resemble information bottlenecks seen in early sensory pathways where coarse coding limits downstream resolution.

Sobel edge-filtered features achieved 85.01% accuracy, closely matching PCA with 50 components. Unlike PCA, Sobel filters emphasize local gradient information—approximating the behavior of orientation-selective simple cells in V1. This suggests that biologically inspired edge extraction can provide a compact and effective representation for digit classification without requiring statistical learning of transformation parameters. However, edge features alone omit some characteristic shape information, which may explain why Sobel did not outperform the raw pixel baseline or high-dimensional PCA inputs. An interesting observation is made, where the Sobel-based classifier showed strong performance for edge-dominant digits but greater overlap for rounder shapes; this can be observed while comparing Figure 1 and 5. This shows that Sobel features are effective for edge-dominant patterns but less reliable for complex shapes.

One limitation of this step is that we did not combine PCA or Sobel features with nonlinear models or stacking architectures that might better exploit their structure. The Ridge classifier used in all comparisons assumes linear separability in the input space. It is possible that Sobel or PCA inputs would outperform raw pixels if paired with more expressive downstream models. Additionally, we evaluated feature transformations under ideal test conditions; future work could include noise, occlusion, or domain shift to test the advantages of edge-based or compressed representations under stress.

Next, we explored the use of polynomial feature interactions to simulate the hierarchial nonlinear integration observed in later stages of the ventral stream, such as the inferior temporal cortex [6]. These models expanded input dimensionality dramatically but revealed how higher-order interactions, paired with appropriate regularization, can capture discriminative structure even in limited data regimes. Ridge regression proved especially effective, maintaining stable accuracy across all regularization strengths (Figure 7), while Lasso showed instability at higher penalties. This behaviour aligns with the known sparsity-inducing effects of L1 regularization, which can eliminate informative features when over-applied [7]. The difference between Ridge and Lasso is reflected in Table 2, underscoring the importance of choosing the right regularizer based on dimensionality and feature correlation.

That said, the degree 2 Ridge model trained on 5,000 samples performed competitively but did not exceed the accuracy of the baseline model trained on the full 60,000 sample dataset (Figure 1). This highlights a critical principle: increased representational capacity does not guarantee better performance unless accompanied by sufficient data and regularization. It also reinforces that simple linear models, when trained on complete datasets, remain highly competitive.

Despite these insights, our study was constrained by computational limits. Degree 3 polynomial models were excluded due to memory constraints, and Lasso's instability limited our ability to explore sparse representations more deeply. Future work could explore more scalable feature learning approaches—such as neural network-based embeddings, convolutional filters, or dimensionality reduction pipelines that combine PCA with polynomial expansion. It would also be interesting to evaluate performance on noisier or more naturalistic datasets to test model robustness and extend the biological relevance of our findings.

Ultimately, our findings illustrate that computational models inspired by brain-like principles, such as hierarchial feature integration and adaptive regularization, can approximate some of the benefits observed in biological visual systems. However, these approaches require careful tuning to balance flexibility with generalization and efficiency.

References

[1] T. Ichinose and S. Habib. On and off signaling pathways in the retina and the visual system. *Frontiers in Ophthalmology*, 2, 2022. doi: 10.3389/fopht.2022.989002.

[2] Y. Devtulla, R. Raj, and S. Kumar. A comprehensive study for enhancing mnist handwritten digit recognition using convolutional neural networks. *SSRN Electronic Journal*, 2024. doi: 10.2139/ssrn.4931484.

[3] R. Pang, B. Lansdell, and A. L. Fairhall. Dimensionality reduction in neuroscience. *Current Biology*, 26:R656–r660, 2016. doi: 10.1016/j.cub.2016.05.029.

[4] J. Reintjes. Nonlinear optical processes. *Encyclopedia of Physical Science and Technology*, pages 537–581, 2003. doi: 10.1016/b0-12-227410-5/00485-3.

[5] S. Ling, M. S. Pratte, and F. Tong. Attention alters orientation processing in the human lateral geniculate nucleus. *Nature Neuroscience*, 18:496–498, 2015. doi: 10.1038/nn.3967.

[6] J. J. DiCarlo, D. Zoccolan, and N. C. Rust. How does the brain solve visual object recognition? *Neuron*, 73:415–434, 2012. doi: 10.1016/j.neuron.2012.01.010.

[7] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288, 12 2018. ISSN 0035-9246. doi: 10.1111/j.2517-6161.1996.tb02080.x.