Autoencoder-Based Reconstruction of MNIST Digits: Architecture Analysis, PCA Comparison, and Recognition Evaluation

Pelopidas-Nikolaos Tsiountsiouras

***Abstract*—This project studies the use of neural network autoencoders for the reconstruction of handwritten digit images from the MNIST dataset. Several fully connected autoencoder architectures with different latent space dimensionalities are designed and trained, and their reconstruction performance is evaluated using mean squared error (MSE), pixel-level accuracy, and visual inspection of reconstructed samples.**

Principal Component Analysis (PCA) is used as a linear baseline for dimensionality reduction and reconstruction, and its performance is directly compared with that of the autoencoders for equivalent latent dimensions. In addition, a digit classifier trained on original MNIST images is used to evaluate whether reconstructed digits remain recognizable. The results show that autoencoders generally achieve better reconstruction quality and higher recognition accuracy than PCA, particularly for larger latent representations, demonstrating the advantage of non-linear models for image reconstruction tasks.

I. Introduction

Dimensionality reduction and image reconstruction are important problems in machine learning and signal processing, as high-dimensional data often contain significant redundancy. Learning compact representations that preserve essential information enables efficient storage, visualization, and further processing of image data.

Autoencoders are neural networks designed to learn low-dimensional representations of input data in an unsupervised manner. They consist of an encoder that compresses the input into a latent representation and a decoder that reconstructs the original data from this representation. Due to their non-linear structure, autoencoders are capable of capturing complex patterns that cannot be modeled by linear techniques such as Principal Component Analysis (PCA).

The MNIST dataset of handwritten digits is a widely used benchmark for evaluating reconstruction and representation learning methods. In this project, several fully connected autoencoder architectures with different latent space sizes are trained to reconstruct MNIST images. Their performance is evaluated using quantitative reconstruction metrics and qualitative visual analysis. PCA-based reconstruction is also implemented as a baseline for comparison.

Finally, to assess whether reconstructed images preserve meaningful information, a digit classifier trained on original MNIST digits is used to evaluate the recognizability of reconstructed outputs. This additional evaluation provides insight into the effectiveness of autoencoders compared to linear dimensionality reduction methods.

II. Dataset and Preprocessing

The experiments in this project are conducted using the MNIST dataset of handwritten digits. The dataset consists of grayscale images of size 28×28 pixels representing the digits 0 through 9. Each image is associated with a corresponding class label and contains variations in writing style, stroke thickness, and digit shape. MNIST is a widely used benchmark dataset and is well suited for evaluating image reconstruction and representation learning methods.

The original training portion of the MNIST dataset is randomly split into two subsets: 60% of the samples are used for training the models and 40% are used for validation. The official MNIST test set is kept separate and is used exclusively for final evaluation. This split allows for monitoring of model performance during training while ensuring unbiased testing on unseen data.

Prior to training, all images are converted to floating-point tensors and normalized to the range $[0, 1]$. Each image is then reshaped into a 784-dimensional vector by flattening the 28×28 pixel grid, allowing it to be processed by fully connected neural networks. No additional data augmentation or feature extraction is applied, ensuring that comparisons between different models and reconstruction methods remain fair and consistent.

For the PCA-based experiments, the same preprocessed and flattened image vectors are used as input. PCA is fitted on the training data and applied to both validation and test sets using an identical preprocessing pipeline. This ensures that any observed performance differences between PCA and autoencoder models are due to the reconstruction method rather than differences in data handling.

III. Autoencoder Models

In this section, the autoencoder architectures used in the experiments are described and their reconstruction performance is analyzed under different training configurations. Fully connected autoencoders with varying model sizes are evaluated in order to study the effect of network capacity on reconstruction quality. In addition, the influence of key training

parameters, namely the batch size and the learning rate, is examined through a series of controlled experiments.

All models are trained using the same dataset, preprocessing steps, and evaluation metrics to ensure fair comparison. The effect of each parameter is analyzed independently by varying one parameter at a time while keeping the remaining settings fixed. Training and validation loss curves, reconstruction accuracy plots, and qualitative reconstruction examples are used to assess model behavior and draw conclusions regarding the impact of each configuration choice.

A. Model Architecture Size

To examine the effect of model capacity on reconstruction performance, three fully connected autoencoder architectures of increasing size are evaluated: a small, a medium, and a large model. All three models are trained using the same batch size (64) and learning rate (0.001) in order to isolate the impact of architectural complexity. Figures 1, 2, and 3 show the training and validation loss as well as the corresponding reconstruction accuracy over 50 training epochs.

The small autoencoder exhibits stable convergence, with both training and validation loss decreasing smoothly throughout training. However, the final reconstruction accuracy remains limited, reaching approximately 89% on the validation set. This behavior indicates that the model is able to learn a compact representation of the input data, but its limited capacity restricts its ability to fully capture the variability present in handwritten digit images.

Increasing the model size leads to a noticeable improvement in reconstruction performance. The medium autoencoder achieves lower final loss values and higher reconstruction accuracy, reaching approximately 93% validation accuracy. The training and validation curves remain closely aligned, suggesting good generalization and minimal overfitting. Among the tested architectures, the medium model achieves the best balance between reconstruction quality and training stability.

The large autoencoder further increases model capacity; however, the improvement over the medium model is relatively small. While the large model converges to slightly lower training loss, the validation accuracy saturates around 91%, which is lower than that achieved by the medium architecture. In addition, the training time of the large model is noticeably higher, indicating increased computational cost without a proportional gain in performance.

Overall, these results suggest that increasing model size improves reconstruction performance up to a certain point, beyond which additional capacity yields diminishing returns. The medium-sized autoencoder provides the most effective trade-off between reconstruction accuracy, generalization performance, and computational efficiency for the MNIST reconstruction task.

B. Effect of Batch Size

To study the effect of batch size on the training behavior of the autoencoder, the medium-sized architecture is trained using two different batch sizes, namely 64 and 128, while keeping the learning rate fixed at 0.0001. Figures 4 and 5 present the training and validation loss curves as well as the corresponding reconstruction accuracy for each configuration.

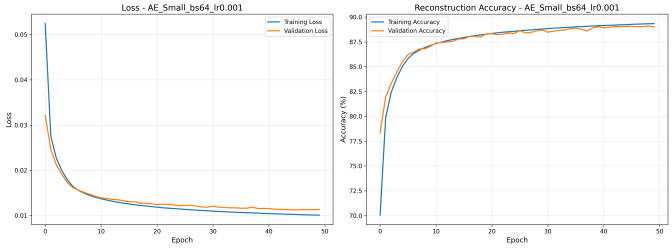

Fig. 1. Training and validation loss and reconstruction accuracy for the small autoencoder.

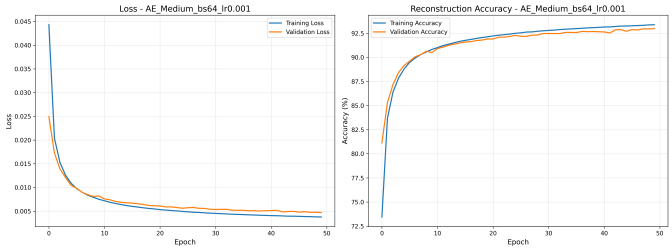

Fig. 2. Training and validation loss and reconstruction accuracy for the medium autoencoder.

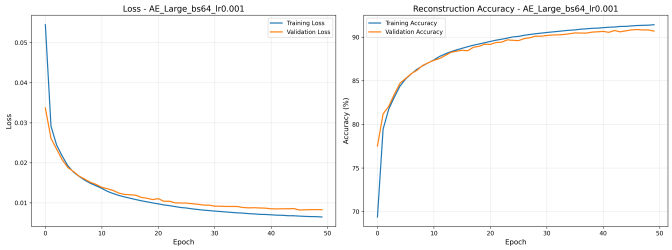

Fig. 3. Training and validation loss and reconstruction accuracy for the large autoencoder.

Both batch sizes lead to stable convergence, with training and validation loss decreasing smoothly over the training epochs. The overall shape of the loss curves is similar in both cases, indicating that the batch size does not significantly affect the ability of the model to learn a meaningful reconstruction. However, the model trained with batch size 128 exhibits slightly smoother loss and accuracy curves, particularly during the early training epochs. This behavior is expected, as larger batch sizes provide more stable gradient estimates by averaging over a greater number of samples.

In terms of reconstruction accuracy, both configurations achieve comparable final performance. The model trained with batch size 64 reaches a slightly higher validation accuracy, while the batch size 128 model converges more gradually and saturates at a similar level. The difference in final accuracy between the two configurations is relatively small, suggesting that batch size primarily influences training stability rather than reconstruction quality.

Overall, the results indicate that increasing the batch size leads to smoother and more stable training dynamics but does not result in a substantial improvement in final reconstruction accuracy. Batch size 64 offers slightly faster convergence, while batch size 128 provides more stable updates, making both choices reasonable depending on the desired balance between convergence speed and training stability.

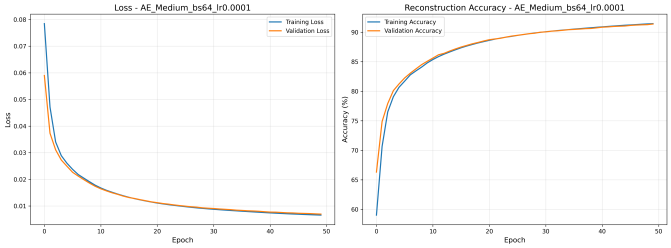

Fig. 4. Training and validation loss and reconstruction accuracy for the medium autoencoder with batch size 64 and learning rate 0.0001.

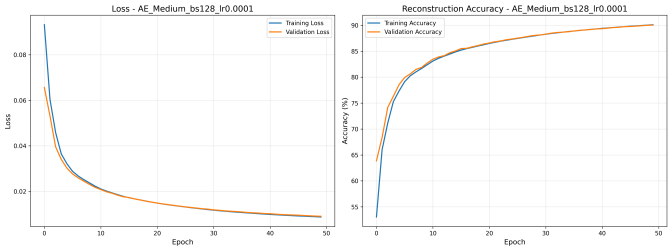

Fig. 5. Training and validation loss and reconstruction accuracy for the medium autoencoder with batch size 128 and learning rate 0.0001.

C. Effect of Learning Rate

The effect of the learning rate on training dynamics and reconstruction performance is examined using the small autoencoder architecture with a fixed batch size of 64. Two learning rates are evaluated, namely 0.001 and 0.0001, while all other training parameters remain unchanged. Figures 6 and 7 illustrate the training and validation loss curves as well as the corresponding reconstruction accuracy for each configuration.

For both learning rates, the autoencoder converges in a stable manner, with training and validation loss decreasing steadily throughout the training process. However, clear differences in convergence speed can be observed. When using a learning rate of 0.001, the model converges significantly faster during the early epochs, achieving lower loss values and higher reconstruction accuracy in fewer training iterations. In contrast, the smaller learning rate of 0.0001 results in slower convergence, requiring more epochs to reach comparable performance.

In terms of final reconstruction accuracy, the configuration with learning rate 0.001 achieves slightly higher validation accuracy compared to the smaller learning rate. The gap between training and validation curves remains small in both cases, indicating stable training behavior and no evident overfitting. Nevertheless, the slower learning rate does not fully exploit the available training epochs, leading to marginally inferior reconstruction performance.

Overall, these results demonstrate that the learning rate has a strong influence on convergence speed and final performance. A learning rate of 0.001 provides a better balance between fast convergence and stable training for the small autoencoder, while a lower learning rate leads to slower optimization without clear benefits in reconstruction quality.

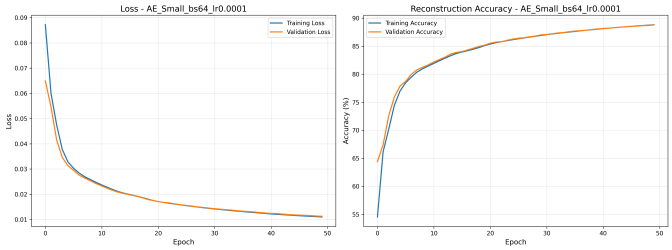

Fig. 6. Training and validation loss and reconstruction accuracy for the small autoencoder with batch size 64 and learning rate 0.0001.

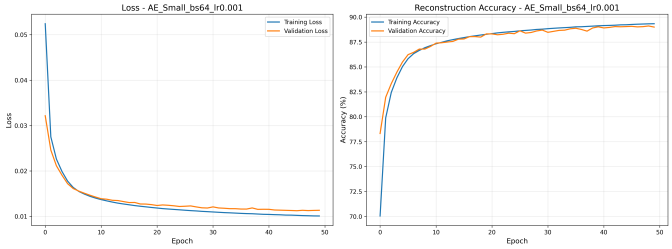

Fig. 7. Training and validation loss and reconstruction accuracy for the small autoencoder with batch size 64 and learning rate 0.001.

IV. Evaluation Metrics

To evaluate the performance of the proposed autoencoder models, multiple quantitative metrics are used in order to assess both reconstruction quality and computational efficiency. These metrics are chosen to provide a comprehensive evaluation of the learned representations and their usefulness for downstream tasks.

A. Reconstruction Loss

The primary training objective of the autoencoder models is the minimization of the Mean Squared Error (MSE) between the input images and their reconstructed outputs. Given an input image $\mathbf{x}$ and its reconstruction $\hat{\mathbf{x}}$, the MSE is defined as

$$\text{MSE} = \frac{1}{N} \sum_{i=1}^{N} (x_i - \hat{x}_i)^2 , \tag{1}$$

where N denotes the number of pixels in the image. Lower MSE values indicate more accurate reconstructions and better preservation of pixel-level information.

B. Pixel-wise Reconstruction Accuracy

In addition to MSE, pixel-wise reconstruction accuracy is computed to provide an intuitive measure of reconstruction quality. This metric evaluates the percentage of pixels whose reconstructed values fall within a predefined tolerance of the original pixel values. Pixel accuracy offers a complementary perspective to MSE by emphasizing how many pixels are reconstructed correctly rather than the magnitude of reconstruction error.

C. Image-level Reconstruction Accuracy

Image-level reconstruction accuracy is also reported to evaluate reconstruction performance at a higher semantic level. An image is considered correctly reconstructed if its overall reconstruction error satisfies a predefined criterion. This metric is more strict than pixel-wise accuracy and reflects the models ability to reconstruct complete images rather than isolated pixel values.

D. Classifier Accuracy on Reconstructed Images

To assess the usefulness of the learned latent representations for downstream tasks, a pre-trained classifier is evaluated on reconstructed test images. The classifier accuracy measures how well semantic information is preserved through the encoding and decoding process. Higher classifier accuracy indicates that the autoencoder maintains discriminative features necessary for digit recognition, even after reconstruction.

E. Training Time

Finally, training time is measured for each model configuration to evaluate computational efficiency. The total training duration is recorded in seconds and includes all epochs required for convergence. This metric highlights the trade-off between reconstruction performance and computational cost, which is an important consideration for practical applications.

V. Quantitative Results and Model Comparison

This section presents a quantitative comparison of the evaluated autoencoder models using the metrics defined in the previous section. Reconstruction quality, downstream classification performance, and computational cost are analyzed to highlight the trade-offs between different model configurations.

A. Reconstruction Performance Comparison

This subsection evaluates the reconstruction quality of the proposed autoencoder models using two quantitative metrics: the test Mean Squared Error (MSE) and the test pixel-wise accuracy. These metrics directly measure how accurately each model reconstructs the original MNIST images at the pixel level.

Figure 8 presents the comparison of test MSE across all evaluated autoencoder configurations. The results indicate that model capacity significantly affects reconstruction performance. Medium-sized autoencoders consistently achieve the lowest reconstruction error, with the best-performing configuration reaching MSE values on the order of 4×10^{-3}. Small

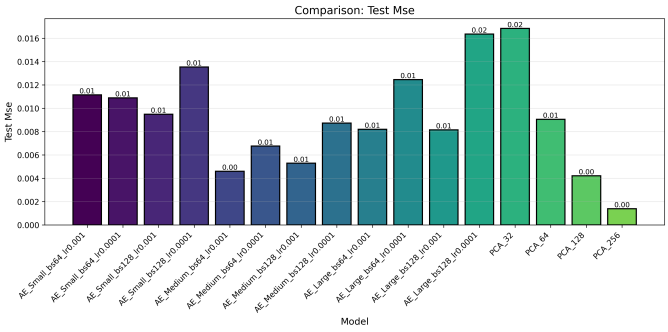

Fig. 8. Comparison of test Mean Squared Error (MSE) across different autoencoder model configurations.

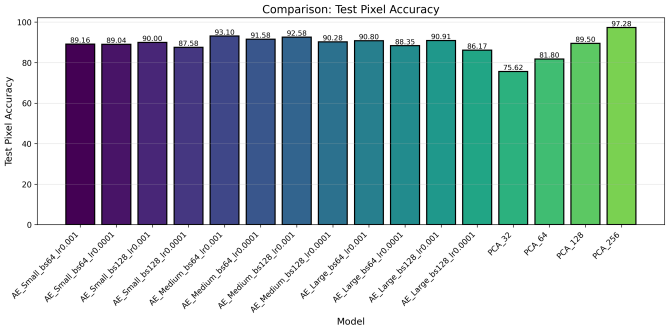

Fig. 9. Comparison of test pixel-wise accuracy across different autoencoder model configurations.

autoencoders exhibit higher reconstruction error across all configurations, reflecting their limited representational capacity. Increasing the model size beyond the medium configuration does not consistently improve performance, as large autoencoders often show higher MSE, suggesting diminishing returns and possible overfitting.

The test pixel-wise accuracy comparison is shown in Fig. 9. The observed trends are consistent with the MSE results. Medium-sized autoencoders achieve the highest pixel accuracy, exceeding 93% in the best-performing configuration. Small models generally achieve pixel accuracies in the range of 88–90%, while large models show slightly lower and less stable performance. These results further confirm that increasing model complexity alone does not guarantee improved reconstruction quality.

The influence of training hyperparameters is also evident. Models trained with a learning rate of 0.001 generally outperform those trained with a lower learning rate, achieving both lower MSE and higher pixel accuracy. The effect of batch size on reconstruction performance is comparatively smaller, with only minor variations observed between different batch configurations.

Overall, the results demonstrate that the medium autoencoder architecture combined with a learning rate of 0.001 provides the best trade-off between reconstruction accuracy and model complexity. This configuration achieves superior quantitative reconstruction performance among the evaluated

autoencoder models.

B. Qualitative Reconstruction Analysis

In addition to quantitative evaluation metrics, qualitative inspection of reconstructed images provides important insights into the behavior and limitations of the autoencoder models. In this subsection, we analyze representative examples of the best and worst reconstructions produced by the large autoencoder trained with batch size 64 and learning rate 0.0001.

Figure 10 shows examples of the best reconstructions obtained by this model. For these samples, the reconstructed digits closely match the original inputs in terms of overall shape, stroke thickness, and orientation. The reconstructed images preserve the essential structural features of the digits, resulting in visually accurate reconstructions with minimal distortion. These examples correspond to samples with low reconstruction error, which is consistent with the low test MSE values reported in the quantitative evaluation.

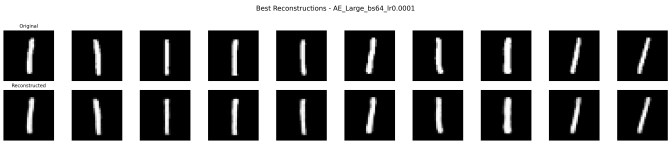

Fig. 10. Best reconstruction examples for the large autoencoder model (batch size = 64, learning rate = 0.0001). The top row shows original MNIST images, while the bottom row shows the corresponding reconstructed outputs.

Figure 11 presents the worst reconstruction examples for the same model. These samples typically involve digits with more complex shapes, overlapping strokes, or ambiguous handwriting styles. In such cases, the autoencoder produces blurred or distorted reconstructions, often losing fine details or altering the digit structure. Some reconstructed images resemble averaged digit shapes rather than clear instances of a single class, indicating difficulty in encoding complex patterns within the latent space.

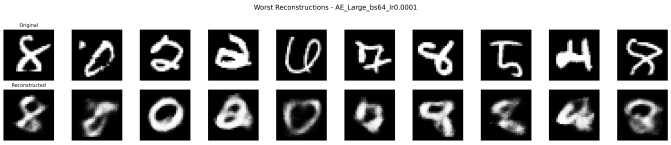

Fig. 11. Worst reconstruction examples for the large autoencoder model (batch size = 64, learning rate = 0.0001). The top row shows original MNIST images, while the bottom row shows the corresponding reconstructed outputs.

Overall, the qualitative results align well with the quantitative metrics reported earlier. While the model performs well on simple and clearly written digits, it struggles with more complex or noisy samples. This behavior highlights a common limitation of autoencoders trained solely with pixel-wise reconstruction loss, where fine-grained structural details may not always be preserved. Nonetheless, the model demonstrates strong reconstruction capability for the majority of inputs, confirming its effectiveness for dimensionality reduction and representation learning on the MNIST dataset.

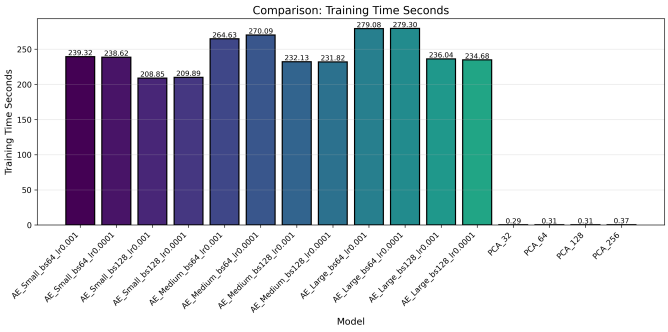

Fig. 12. Comparison of total training time (in seconds) across all autoencoder configurations and PCA baselines.

C. Computational Cost Analysis

This subsection analyzes the computational efficiency of the evaluated models, focusing on training time, scalability with model complexity, and the trade-off between computational cost and reconstruction or downstream performance. The analysis is based on the training logs and the comparative plots shown in Figures 12, 13, 14, and 15.

Training Time Comparison.: Figure 12 reports the total training time for all autoencoder configurations and PCA baselines. As expected, training time increases with model capacity: Large autoencoders require the longest training time, followed by Medium and Small variants. Differences between batch sizes are modest, with larger batch sizes slightly reducing per-epoch overhead but not fully compensating for the increased model complexity.

In contrast, PCA-based methods are orders of magnitude faster to train, requiring less than one second in all configurations. This highlights PCA's computational advantage when training time is the primary constraint, albeit at the cost of reduced reconstruction fidelity and representational flexibility.

Accuracy Versus Computational Cost.: Figure 13 compares training accuracy, test image accuracy, test pixel accuracy, and classifier accuracy on reconstructed images across all models. Larger autoencoders achieve consistently higher reconstruction and downstream classification performance, but this improvement comes at the cost of increased training time, as shown in Figure 12.

Notably, Medium autoencoders offer a favorable balance, achieving near-optimal accuracy while requiring significantly less computation than Large models. This suggests diminishing returns in performance when moving from Medium to Large architectures relative to the additional computational cost.

Classifier Accuracy on Reconstructed Data.: Figure 14 isolates the classifier accuracy obtained when operating on reconstructed images. Despite their higher computational cost, Large autoencoders provide only marginal gains over Medium configurations. PCA models with sufficiently large latent dimensions achieve competitive classifier accuracy while being computationally inexpensive, reinforcing the importance of

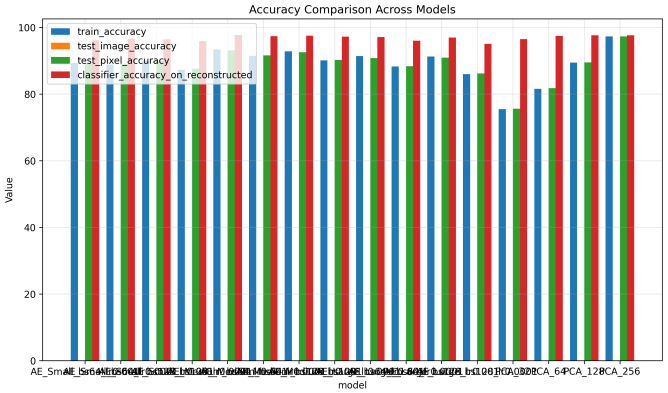

Fig. 13. Comparison of reconstruction and classification-related accuracy metrics across models.

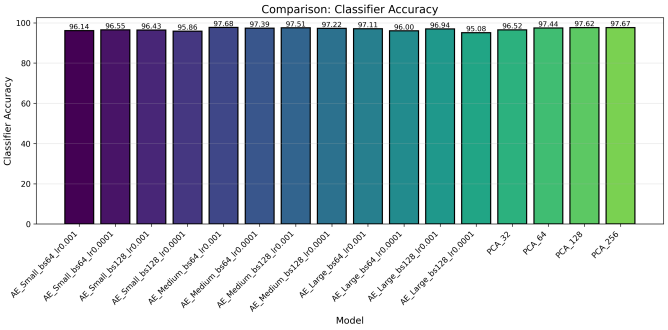

Fig. 14. Classifier accuracy evaluated on reconstructed images for all models.

considering costperformance trade-offs in practical deployments.

Effect of Latent Dimensionality.: Figure 15 illustrates the relationship between latent dimensionality and classifier accuracy on reconstructed samples. Increasing the latent dimension generally improves downstream accuracy, but the gains saturate beyond a certain point. This trend indicates that excessively large latent spaces increase computational cost without providing proportional performance benefits.

Summary.: Overall, the computational cost analysis demonstrates a clear trade-off between model complexity and efficiency. While Large autoencoders yield the strongest performance, Medium models provide a more balanced solution, achieving competitive accuracy at substantially lower computational cost. PCA offers exceptional efficiency but lags behind deep models in reconstruction quality, making it suitable primarily for resource-constrained or real-time scenarios.

D. Summary of Comparison

Overall, the quantitative evaluation demonstrates a clear trade-off between reconstruction quality, downstream task performance, and computational cost across the evaluated models. Medium-capacity autoencoders consistently achieve the best balance, offering low reconstruction error and high pixel-level accuracy while maintaining strong classifier performance on reconstructed samples. Increasing model size or latent

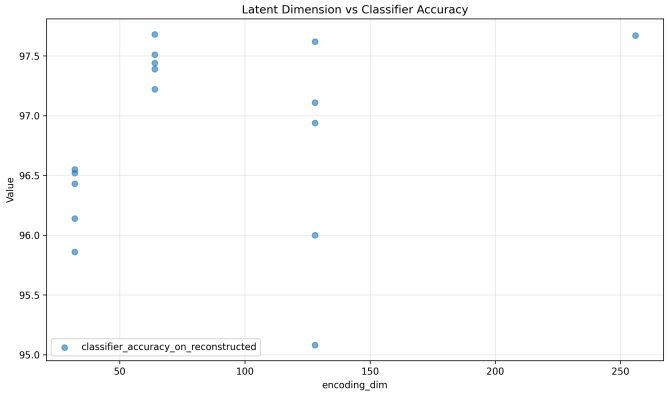

Fig. 15. Classifier accuracy as a function of latent dimensionality.

Model	Latent Dim.	Test MSE	Pixel Acc. (%)	Classifier Acc. (%)
AE-Small	32	0.011	89.1	96.1
AE-Medium	64	0.006	93.1	97.7
AE-Large	128	0.008	90.9	96.9
PCA	256	0.001	97.3	97.7

TABLE I
SUMMARY OF QUANTITATIVE PERFORMANCE ACROSS MODELS.

dimensionality yields diminishing returns in accuracy, while incurring higher training costs. In contrast, PCA-based baselines provide extremely low computational overhead but lag behind autoencoders in reconstruction fidelity and downstream classification accuracy.

Table I summarizes the key results derived from the training logs and test evaluations, highlighting the best-performing configurations across reconstruction, classification, and efficiency criteria.

In summary, medium-sized autoencoders emerge as the most effective choice when balancing reconstruction accuracy and computational efficiency, while PCA offers a competitive low-cost alternative when training time is a primary constraint. These findings motivate the qualitative analysis presented next, which further examines reconstruction behavior beyond aggregate metrics.

VI. Conclusion

In this project, neural network autoencoders were investigated for the task of handwritten digit reconstruction using the MNIST dataset. Multiple autoencoder architectures and training configurations were evaluated in order to analyze the effect of model size, learning rate, and batch size on reconstruction quality and computational cost. Quantitative evaluation using mean squared error and pixel-wise reconstruction accuracy showed that medium-sized autoencoders achieved the best overall performance, offering a favorable balance between reconstruction accuracy and model complexity.

Further analysis demonstrated that appropriate hyperparameter selection, particularly the learning rate, plays a critical role in convergence speed and final reconstruction quality. Comparisons with a PCA-based baseline highlighted the advantages of non-linear autoencoders in preserving fine-grained image details and semantic information. Additionally, classifier-based

evaluation on reconstructed images confirmed that autoencoder reconstructions retain meaningful discriminative features for digit recognition.

Overall, the results indicate that autoencoders are effective models for dimensionality reduction and image reconstruction tasks, outperforming linear methods such as PCA when sufficient model capacity and proper training settings are used. Future work could explore convolutional autoencoder architectures or alternative loss functions to further improve reconstruction quality and robustness.