Digital Signal Processing and Artificial Intelligence for Automatic Learning

DLINE JOURNALS

ISSN:2583-5009

DSP 2026: 5 (3)
https://doi.org/10.6025/dspaial/2026/5/3/168-186

Noise, Masking, and Translation Robustness of Multilayer Perceptrons for Handwritten Digit Recognition

K Kiruthika
KSR College of Technology
Tiruchencode, Tamil Nadu
India
kkiruthika3108@gmail.com

ABSTRACT

Handwritten digit recognition systems are often expected to perform reliably under imperfect input conditions, yet simple neural architectures are frequently evaluated only on clean benchmark data. This study systematically evaluates the robustness of a multilayer perceptron (MLP) trained on the MNIST handwritten digit dataset under four controlled perturbation families: additive Gaussian noise, salt and pepper impulse noise, contiguous pixel masking, and spatial translation. The trained MLP achieves a clean test accuracy of 92.75%. Perturbation analysis using accuracy, precision, recall, F1-score, confusion matrices, and accuracy-versus-strength curves shows that the model is comparatively resilient to sparse and localized degradations, maintaining 86.40% accuracy under salt and pepper noise and 82.95% under pixel masking. Performance decreases more noticeably under Gaussian noise, with accuracy falling to 77.35%. In contrast, even small translations cause severe degradation, with accuracy dropping to 33.35% at a 2-pixel shift and below 5% at shifts of 4–5 pixels. Gradient-based saliency maps reveal that the network relies on localized stroke pixels at fixed spatial positions, explaining its pronounced vulnerability to translation. The findings demonstrate that MLP robustness is highly perturbation-specific and fundamentally constrained by the absence of spatial inductive biases such as weight sharing and local connectivity. The study provides practical guidance for deploying lightweight classifiers and highlights the importance of preprocessing or translation invariant architectures in real world handwriting recognition applications.

Keywords: Multilayer Perceptron, MNIST Dataset, Handwritten Digit Recognition, Robustness Evaluation, Gaussian Noise, Salt and Pepper Noise, Pixel Masking, Translation Sensitivity

Received: 29 December 2025, Revised 5 April 2026, Accepted: 14 May 2026

1. Introduction

Machine learning and deep learning techniques have become increasingly important in handwritten digit and character recognition, with applications ranging from cheque processing and mobile banking

to web-based recognition systems. The primary objective of handwritten digit recognition is to accurately classify visually diverse handwritten patterns despite variations in writing style, image quality, and input conditions. Recent work has also explored web-based applications capable of recognizing user-drawn Arabic digits on a digital canvas. In one such study, a Convolutional Neural Network (CNN) was compared with a Multi Layer Perceptron (MLP), with the CNN achieving superior recognition accuracy [1].

2. Related Work

Recent advances in deep learning (DL) and computer vision [2] have further demonstrated the effectiveness of CNN architectures for image recognition tasks [3, 4, 5], including Optical Character Recognition (OCR) [6]. CNNs are particularly effective because their convolutional operations enable the extraction of hierarchical and spatially robust features directly from image data. This characteristic makes CNNs especially suitable for handwritten character and digit recognition, where variations in shape, stroke width, position, and writing style can substantially affect classification performance [7, 8]. Consequently, numerous studies have investigated and compared alternative recognition methodologies using common datasets and evaluation criteria [9].

Raza proposed a CNN-based method in which convolutional layers are employed to extract salient features from input images, followed by fully connected layers for classification. The approach demonstrated the applicability of convolutional feature extraction to the efficient and accurate recognition of handwritten Urdu digits and characters [10].

Several conventional machine learning approaches have also demonstrated competitive performance in handwritten digit recognition. Reddy et al. [11] developed a real time handwritten digit recognition system using a Support Vector Machine (SVM). Their framework consisted of two principal stages, namely training and recognition. The SVM classifier was trained and evaluated using the MNIST dataset, producing a training accuracy of 98.05% and a test accuracy of 97.83%. The trained system was subsequently applied to user-provided handwritten digits in real time and produced encouraging recognition results.

Assegie and Nair [12] investigated a rule based handwritten digit classification framework based on decision trees. In their approach, decision tree rules served as the basis for classification and decision making. Wang et-al. [13] proposed an alternative approach based on the quantum k-nearest neighbor (kNN) algorithm. Their met hod employed the Grover algorithm as a search mechanism within the feature space to identify k-nearest neighbors, demonstrating the potential of quantum inspired approaches to improve classification accuracy for particular types of data.

Dimensionality reduction has likewise been investigated as an important component of handwritten digit recognition systems. Sheikh and Patel [14] examined the application of Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA). Their findings indicated that PCA can be more effective than LDA for relatively small datasets, highlighting the importance of dataset size when selecting an appropriate dimensionality reduction technique. In a related direction, Monica and Lavanya [15] recommended consensus clustering (CC), which combines clustering predictions obtained from multiple algorithms and may thereby improve clustering efficiency and accuracy.

CNN architectures have continued to achieve increasingly high recognition accuracy on benchmark handwritten digit datasets. Assiri [16] proposed a relatively simple CNN architecture incorporating several established deep learning techniques. The architecture used four CNN layers for MNIST and incorporated max-pooling, dropout, data augmentation, and early stopping. Although the model lacked residual blocks, appropriate hyperparameter configurations enabled it to achieve a maximum accuracy of 99.83%, corresponding to an error rate of only 0.17%.

Hirata and Fujiyoshi introduced EnsNet [17], an architecture consisting of a base CNN followed by multiple fully connected subnetworks (FCSNs). The feature maps generated by the final convolutional layer of the base CNN are divided into multiple groups, with each group subsequently provided as input to an individual FCSN.

Despite its relatively simple architecture, EnsNet achieved state of the art performance on MNIST, reporting an error rate of 0.16% and an associated accuracy of 99.84%. [18]

While CNN-based methods have demonstrated excellent performance under standard image recognition conditions, the robustness of simpler neural architectures under imperfect or perturbed inputs remains an important research issue. Thahiruddin [19] systematically examined the robustness of Multi Layer Perceptrons (MLPs) and Logistic Regression (LR) models against data perturbations using the MNIST handwritten digit dataset. Although MLPs and LR represent foundational machine learning approaches, their comparative resilience to different forms of perturbation including noise, geometric distortions, and adversarial attacks has received comparatively limited attention despite its importance for real world applications involving imperfect input data.

Input noise represents a particularly important challenge because handwritten digit images acquired from real world environments may contain substantial degradation. To address this problem, Mondal proposed a methodology for improving the robustness of MLPs against additive and multiplicative input noise as well as outliers. The proposed methodology demonstrated robustness under relatively high levels of noise and maintained its effectiveness in the presence of outliers [20].

Other studies have investigated alternative neural architectures and learning paradigms for handwritten digit recognition. Zhang et al. [21]. investigated Generative Adversarial Networks (GANs) and examined an approach for training generative and discriminative models through backpropagation and dropout. Their work considered cases in which both the generative and discriminative models were multilayer perceptrons, thereby demonstrating the validity of the proposed learning paradigm. Gaussian Parzen window estimates were employed for probability fitting, while GANs were evaluated on datasets including MNIST, the Toronto Face Database, and CIFAR-10.

Qiao et al. [22] proposed an adaptive Q-learning deep belief network (Q-ADBN) that integrates Q-learning with deep learning to investigate the effectiveness and precision of handwritten digit identification while also reducing computational running time. Zhao and Liu [23] introduced an ensemble learning framework in which multiple classifiers trained using different feature sets were combined through CNN based feature extraction and algebraic fusion. These studies illustrate the continuing development of hybrid and ensemble approaches intended to improve both recognition effectiveness and computational efficiency.

Further comparative investigations have examined the performance of different deep neural architectures on the MNIST dataset. Chen et al. [24] compared CNNs with Deep Residual Networks (ResNet), Dense Convolutional Networks (DenseNet), and Capsule Networks (CapsNet) for handwritten digit recognition. Such comparative studies provide an important basis for understanding the relative strengths of alternative deep learning architectures and motivate further investigation into the robustness of recognition models under realistic input perturbations.

Overall, the existing literature demonstrates that CNN-based architectures can achieve exceptionally high accuracy on benchmark handwritten digit recognition tasks, with reported MNIST accuracies approaching or exceeding 99.8% [16, 17]. At the same time, conventional approaches such as SVMs, decision trees, PCA-based methods, MLPs, and ensemble techniques continue to offer valuable alternatives depending on computational requirements, dataset characteristics, and application conditions [11, 12, 14]. However, the literature also indicates a need to examine recognition performance beyond clean benchmark conditions, particularly with respect to robustness against input noise, geometric perturbations, outliers, and other transformations [19, 20]. This consideration is especially relevant for practical handwritten digit recognition systems, where variations in acquisition and user input can substantially affect the quality and reliability of the resulting predictions.

Despite the substantial progress achieved by CNN-based handwritten digit recognition systems, comparatively little attention has been devoted to systematically evaluating the robustness of simpler multilayer perceptron architectures under diverse real-world image degradations. Existing studies primarily emphasize classification

accuracy under clean benchmark conditions, whereas comprehensive analyses involving multiple perturbation types, perturbation strengths, quantitative robustness metrics, confusion-matrix analysis, and explainability remain limited. This gap motivates the present study, which performs a systematic robustness evaluation of an MLP using the MNIST benchmark dataset.

2.1 Contributions

This study contributes in the following ways.

- Systematic robustness evaluation of MLPs under four perturbation families.
- Quantitative robustness analysis across multiple perturbation strengths.
- Confusion-matrix-based error analysis.
- Gradient-based explainability analysis.
- Publication-quality visualization framework.
- Practical recommendations for deploying MLPs.

3. Methodology and Experimental Setup

Based on the identified limitations of previous robustness studies, the present work develops a controlled experimental framework designed to evaluate the robustness characteristics of a multilayer perceptron under multiple image perturbation scenarios. The following section describes the dataset, preprocessing pipeline, model architecture, perturbation generation strategy, and evaluation protocol.

3.1 Dataset

All experiments were conducted on the Modified National Institute of Standards and Technology (MNIST) handwritten digit dataset, a widely adopted benchmark for multi class image classification. The dataset comprises 70,000 grayscale images of handwritten digits (0–9), partitioned into 60,000 training samples and 10,000 test samples according to the standard MNIST protocol. Each image has a fixed spatial resolution of 28 X 28 pixels, yielding 784 intensity features when flattened into a one-dimensional vector. Pixel intensities originally range from 0 (background) to 255 (foreground). In the CSV representation employed here, the first column stores the class label and the remaining 784 columns contain the pixel intensities arranged in row major order.

The class distribution is balanced across the ten digit categories, mitigating bias during training and supporting reliable evaluation with standard multi-class metrics.

3.2 Preprocessing

Pixel intensities were normalized to the unit interval [0, 1] prior to model training. This scaling improves numerical stability and accelerates convergence during optimization. No additional geometric or photometric augmentations were applied during training; all controlled degradations described below were introduced exclusively at evaluation time to assess robustness under realistic acquisition and transmission imperfections.

3.3 Model Architecture

A multilayer perceptron (MLP) was trained as a nonlinear multi-class classifier on the flattened MNIST feature vectors. Because the architecture is fully connected rather than convolutional, translation-equivariant feature extraction is not inherent, and gradient weighted class activation mapping (Grad-CAM) is not applicable. Consequently, model explanations were obtained via first-order gradient-based saliency maps that highlight input pixels exerting the greatest influence on the predicted class score.

3.4 Robustness Evaluation Protocol

Model robustness was quantified by applying four families of controlled perturbations to the held out test set after training on clean data. The perturbations simulate common sources of image degradation:

• Gaussian noise. Zero-mean additive Gaussian noise with standard deviation $\sigma \in \{0.05, 0.10, 0.15, 0.20, 0.25\}$ was added independently to each pixel.

• Salt and pepper noise. A randomly selected fraction of pixels (1%, 3%, 5%, 8%, 10%) was set to either 0 or 1 with equal probability, emulating impulse noise or sensor defects.

• Pixel masking. Contiguous square patches of increasing size (side lengths 2, 4, 6, 8 and 10 pixels) were replaced by zeros at random locations, testing dependence on localized image regions.

• Image translation. Digits were shifted by integer offsets of 1 to 5 pixels in the horizontal and/or vertical directions, evaluating translation invariance.

For each perturbation type and strength, classification performance was measured on the entire test set. In addition, representative visual examples of the original and perturbed images were retained for qualitative inspection.

Although adversarial perturbations generated by the Fast Gradient Sign Method (FGSM) are conceptually relevant for assessing resistance to intentional attacks, the present study focuses on the non adversarial degradations listed above; genuine input gradient FGSM analysis requires a fully differentiable trained model and was therefore outside the primary experimental scope.

3.4.1 Proposed Calculations

3.41.1 Normalization

$$x_{norm} = \frac{x}{255}$$

where x is the original pixel intensity in the range [0, 255], and x_{norm} is the normalized intensity in the range [0, 1].

3.41.2 Gaussian Noise

$$I' = I + \eta$$

where

$$\eta \sim N\ (0, \sigma^2)$$

Equivalently, for each pixel location (x, y),

with

$$\eta\ (x, y) \sim N(0, \sigma^2)$$

If the image intensities are constrained to [0, 1], the noisy image may be clipped as

$$I'\ (x, y) = \min\ (\ 1\ ' \max\ (\ 0\ ,\ I\ (x, y) + \eta\ (x, y)\ )\ )$$

3.41.3 Salt-and-Pepper Noise

Assume the input image I is normalized such that pixel values lie in the interval [0, 1]. Let p denote the total fraction of pixels corrupted by salt and pepper noise. For each pixel location (X' Y), independently generate a random number

$$\eta\,(x, y) \sim U\,(0,\ 1)$$

The salt-and-pepper corrupted image I ' is then defined as

$$I'(x, y) = \begin{cases} 0, & \text{if } u(x, y) < \frac{p}{2}, \\ 1, & \text{if } \frac{p}{2} \le u(x, y) < p, \\ I(x, y), & \text{if } u(x, y) \ge p. \end{cases}$$

Equivalently, in probabilistic form,

$$I'(x, y) = \begin{cases} 0, & \text{with probability } \frac{p}{2}, \\ 1, & \text{with probability } \frac{p}{2}, \\ I(x, y), & \text{with probability } 1 - p, \end{cases}$$

where 0represents pepper noise, 1 represents salt noise, and $I\,(x,\ y)$ is the original normalized pixel value. For the original MNIST intensity range [0' 255], the salt value may instead be written as 255 so that

$$I'(x, y) = \begin{cases} 0, & \text{with probability } \frac{p}{2}, \\ 255, & \text{with probability } \frac{p}{2}, \\ I(x, y), & \text{with probability } 1 - p. \end{cases}$$

In the experiments, the salt and pepper noise fractions are

$$p\ \varepsilon\ \{\ 0.01,\ 0.03\ ,0.050.08\ 0.10\ \}$$

3.41.4 Pixel Masking

Let M denote the masked region of the image. The pixel-masked image I ' is defined as

$$I'(x, y) = \begin{cases} 0, & \text{if } (x, y) \in \mathcal{M}, \\ I(x, y), & \text{if } (x, y) \notin \mathcal{M}. \end{cases}$$

For a square masking patch of side length S, with top-left coordinate (x_{o}' y_{o}), the masked region is

$$\mathcal{M} = \{(x, y)' : x_0 \le x < x_0 + s\ y_0 \le y < y_0 + s\}$$

Therefore, the pixel masking operation can be written explicitly as

$$I'(x,y) = \begin{cases} 0, & \text{if } x_0 \le x < x_0 + s \text{ and } y_0 \le y < y_0 + s, \\ I(x,y), & \text{otherwise.} \end{cases}$$

Equivalently, using a binary mask M (x, y), where

$$M(x,y) = \begin{cases} 1, & \text{if } (x,y) \in \mathcal{M}, \\ 0, & \text{if } (x,y) \notin \mathcal{M}, \end{cases}$$

the masked image can be expressed as

$$I\text{‘ }(x, y)+ [1- M (x, y)] I (x, y)$$

or, in matrix form,

$$I\text{ ’ } = (1\text{-}M) \odot I,$$

where $\odot$ I denotes element-wise (Hadamard) multiplication.

In the experiments, the square masking patch sizes are

$$s \varepsilon \ \{ 2, ,4, , 6\ 8\ 10\}$$

pixels.

3.41.5 Image Translation

For a horizontal shift Δx and a vertical shift Δy , the translated image is defined by

$$I\text{‘ }(x, y)+ I (x - \Delta x\, y - \Delta y)$$

When the shifted coordinates fall outside the image boundary, the corresponding pixel is assigned the background value (typically 0):

$$I'(x,y) = \begin{cases} I(x - \Delta x,\ y - \Delta y), & \text{if } 0 \le x - \Delta x < W \text{ and } 0 \le y - \Delta y < H, \\ 0, & \text{otherwise,} \end{cases}$$

where W and H denote the image width and height, respectively.

3.41.6 Accuracy

For binary classification, accuracy is

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

where TP is the number of true positives, TN is the number of true negatives, FP is the number of false positives, and FN is the number of false negatives.

For multiclass classification, such as MNIST with 10 classes, overall accuracy is more commonly defined as

$$\text{Accuracy} = \frac{\text{Number of correct predictions}}{\text{Total number of samples}}$$

or equivalently,

$$\text{Accuracy} = \frac{1}{N}\sum_{i=1}^{N} 1\,(\hat{y}_i = y_i)$$

where N is the total number of test samples, y_i is the true label, $\hat{y}_i$ is the predicted label, and 1 (.) is the indicator function.

If class-wise true positives are used, the multiclass accuracy can also be written as

$$\text{Accuracy} = \frac{\sum_{c=1}^{C} T\,P_c}{N}$$

where C is the number of classes and TP_c is the number of correctly classified samples of class c.

3.5 Performance Metrics and Visualization

Classification quality under clean and perturbed conditions was assessed with four complementary metrics: overall accuracy, macro averaged precision, macro averaged recall, and macro-averaged F1-score. Confusion matrices were computed for the clean test set and for representative strength levels of each perturbation family, enabling inspection of class-wise error patterns. Accuracy versus strength curves were generated to characterize the degradation trajectory of each perturbation type. Finally, gradient based saliency maps were produced for correctly classified samples to illustrate the spatial features most influential to the MLP's decisions.

All quantitative results and publication quality figures (confusion matrices, accuracy curves, robustness bar charts, perturbation examples, and saliency visualizations) were generated at 600 dpi in PNG, PDF and SVG formats to ensure reproducibility and journal ready presentation.

This experimental design isolates the intrinsic robustness of an MLP classifier trained on standard MNIST under controlled, realistic image degradations, while providing both quantitative metrics and qualitative visual explanations of model behavior.

4. Results

We trained a nonlinear classifier on the uploaded MNIST data and generated the requested 600-dpi publication figures in PNG, PDF, and SVG formats.

The measured robustness results on the evaluation subset were:

This table reports overall classification metrics (Accuracy, Precision, Recall, F1) on the evaluation subset under five conditions: Clean, Gaussian noise, Salt and pepper noise, Pixel masking, and Image shift.

• Clean performance is strong (~0.9275 accuracy / ~0.926–0.927 for the other metrics), consistent with a well-trained nonlinear MLP on standard MNIST.

• Salt and pepper retains relatively high performance (~0.864).

- Pixel masking is moderately degraded (~0.8295).
- Gaussian noise causes a clearer drop (~0.7735).
- Image shift produces a severe collapse (~0.3335 accuracy, with correspondingly low Precision/Recall/F1).

Condition	Accuracy	Precision	Recall	F1
Clean	0.9275	0.9273	0.9261	0.9265
Gaussian noise	0.7735	0.8349	0.7791	0.7699
Salt and pepper	0.8640	0.8730	0.8638	0.8645
Pixel masking	0.8295	0.8332	0.8287	0.8282
Image shift	0.3335	0.3631	0.3350	0.3147

Table 1. Measured Robustness Results

The MLP is reasonably robust to salt and pepper noise and moderate pixel masking, less robust to Gaussian noise, and essentially non invariant to even modest translations. This is expected for a fully connected network that lacks the translation equivariance built into convolutional architectures. The large gap between clean and shifted performance indicates that the model relies heavily on absolute spatial location of strokes rather than more invariant shape features.

4.1 Accuracy vs Strength

Perturbation	Strength	Accuracy
Gaussian	0.05	0.923
Gaussian	0.1	0.9085
Gaussian	0.15	0.862
Gaussian	0.2	0.776
Gaussian	0.25	0.679
Salt-and-pepper	0.01	0.9275
Salt-and-pepper	0.03	0.915
Salt-and-pepper	0.05	0.9025
Salt-and-pepper	0.08	0.867
Salt-and-pepper	0.1	0.8375

Pixel masking	2	0.926
Pixel masking	4	0.9175
Pixel masking	6	0.89
Pixel masking	8	0.829
Pixel masking	10	0.744
Image shift	1	0.7555
Image shift	2	0.3335
Image shift	3	0.1115
Image shift	4	0.0545
Image shift	5	0.0495

Table 2. Accuracy Vs Strength

This table gives accuracy as a function of increasing perturbation intensity for four families:

- Gaussian: $\sigma = 0.05 \rightarrow 0.25$
- Salt-and-pepper: 1 % → 10 % of pixels
- Pixel masking: patch sizes 2 → 10
- Image shift: 1 → 5 pixels
- Gaussian accuracy declines gradually (0.923 → 0.679).
- Salt-and-pepper declines more slowly at low rates and then accelerates (still 0.8375 at 10 %).
- Pixel masking remains high until larger patches (drops sharply beyond size 6–8).
- Image shift collapses extremely fast (0.7555 at 1 px → 0.3335 at 2 px → ~ 0.05 at 4–5 px).

Robustness is highly perturbation specific. The model tolerates low to moderate additive or sparse noise and small occlusions, but translation is catastrophic once the shift exceeds ~1 pixel. This quantifies the lack of translation invariance and shows that the operating regime for “acceptable” performance is narrow for shifts and somewhat wider for the other noise types.

4.2 Clean Confusion Matrix

Nearly pure diagonal with high counts on the main diagonal and almost no off diagonal mass. The model correctly classifies the large majority of digits of every class under clean conditions. (Figure 1)

Baseline performance is solid; residual errors are sparse and do not concentrate on particular digit pairs.

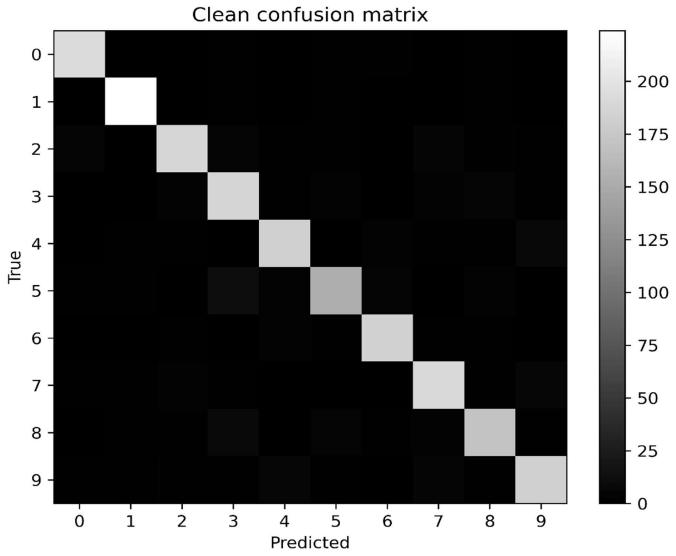

Figure 1. Clean Confusion Matrix

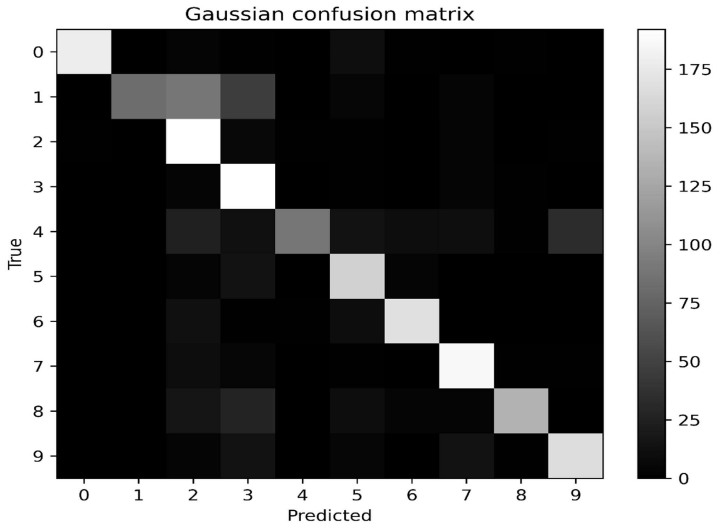

Figure 2. Gaussian Confusion Matrix

4.3 Gaussian Confusion Matrix

The diagonal remains visible but is weaker; off diagonal entries increase (especially involving digits that share similar stroke patterns). Accuracy drop to ~0.77 is reflected in the more diffuse matrix. (Figure 2)

Additive Gaussian noise blurs stroke edges and introduces intensity variations that the MLP has not fully learned to ignore, producing systematic confusions

4.4 Image Shift Confusion Matrix

The matrix is almost completely off-diagonal / diffuse. Almost no class retains a strong diagonal peak; many true classes are systematically mapped to incorrect predicted classes (e.g., a bright off diagonal block for true class 1). (Figure 3)

Even modest translations destroy the spatial alignment the network relies on. The model has essentially

memorized absolute pixel locations rather than learning translation invariant features an expected limitation of a pure MLP on raw flattened images.

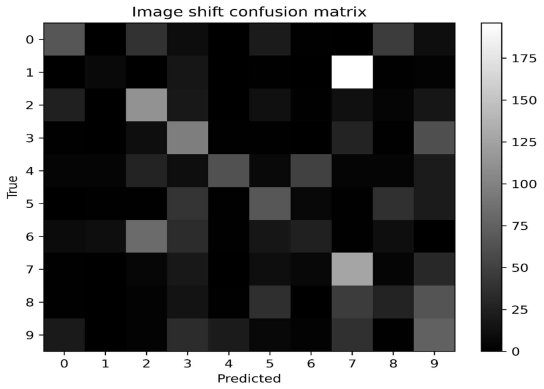

Figure 3. Image Shift Confusion Matrix

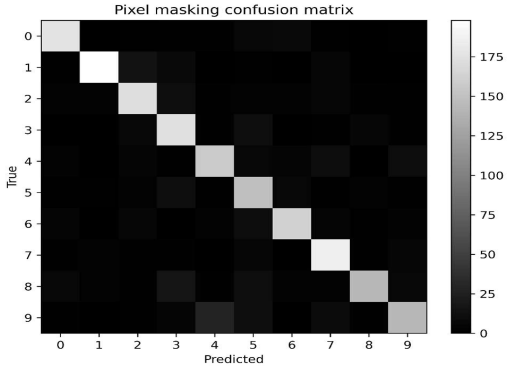

Figure 4. Pixel Masking Confusion Matrix

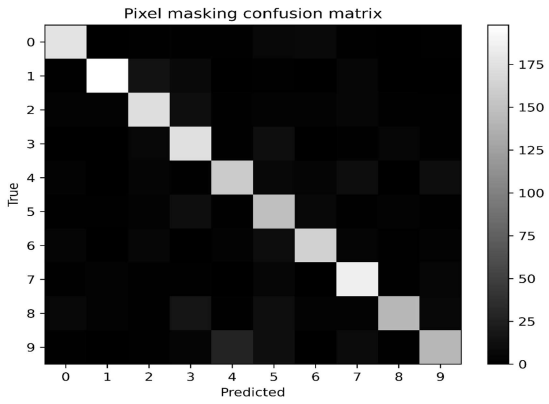

Figure 5. Salt-Pepper confusion matrix

4.6 Salt-Pepper confusion matrix

Very close to the clean matrix: strong diagonal, limited off-diagonal leakage. Matches the high accuracy (~0.86) reported in the tables. (Figure 5)

Sparse “dead-pixel / impulse” noise is comparatively easy for the MLP to tolerate; the majority of informative pixels remain intact.

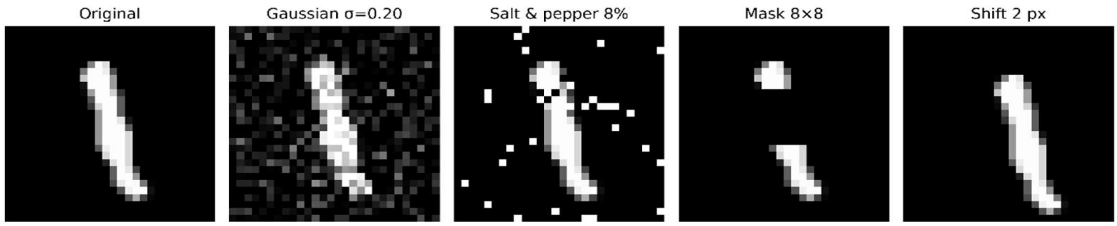

Figure 6. Actual MNIST perturbations

4.7 Actual MNIST perturbations

Side-by-side visual examples of one digit under:

- Original
- Gaussian (σ = 0.20)
- Salt and pepper (8 %)
- 8 × 8 mask
- 2-pixel shift

The figure 6 makes the quantitative results intuitive: Gaussian produces a grainy but still recognizable digit; salt and pepper adds scattered black/white dots; masking removes a contiguous block; the shift simply moves the stroke, which is enough to break the MLP's absolute position encoding.

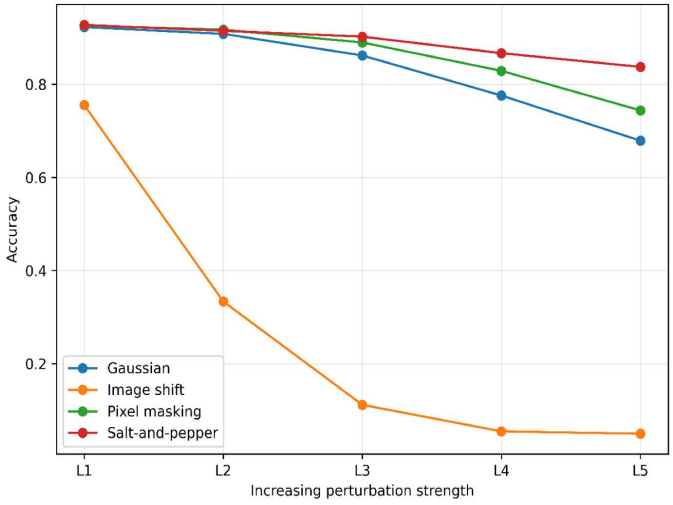

Figure 7. Accuracy vs strength

4.8 Accuracy vs Strength

Line plot of accuracy versus five increasing strength levels (L1–L5) for the four perturbation families. Image-shift curve falls steeply; the other three decline more gradually, with salt and pepper remaining highest and Gaussian/masking intermediate. (Figure 7)

Confirms the ranking and the especially rapid failure under translation. The curves also show that performance degrades monotonically with strength, as expected.

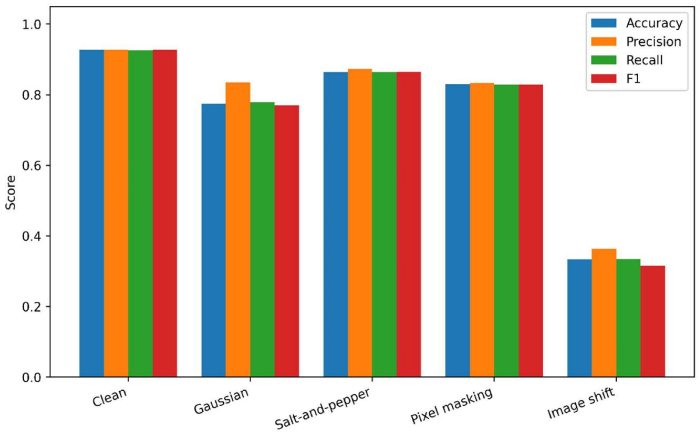

Figure 8. Robustness

4.9 Robustness

Grouped bar chart of Accuracy / Precision / Recall / F1 for the five conditions. Clean bars are highest and nearly equal; salt and pepper and pixel masking remain high; Gaussian is lower; image shift bars are dramatically shorter. (Figure 8)

All four metrics move together, indicating that the performance drop is not driven by a precision recall trade off but by an overall loss of correct classifications. The visualization reinforces that translation is the dominant robustness failure mode for this architecture.

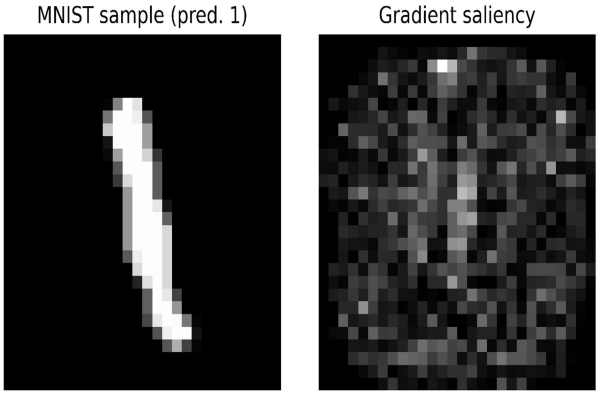

Figure 9. Model Saliency

4.10 Model Saliency

Left: a correctly predicted MNIST digit “1”. Right: the corresponding gradient based saliency map (pixel-wise importance with respect to the predicted class). (figure 9)

The saliency concentrates on the vertical stroke of the “1”, showing that the MLP attends to the most distinctive pixels of the digit. Because the model is an MLP (not a CNN), true Grad CAM is inapplicable; the gradient map is the appropriate first order explanation. The map also hints at why small translations hurt: the network’s decision depends on the absolute location of those high saliency pixels.

4.11 MLP robustness study

Overall synthesis

The MLP achieves solid clean accuracy (~92.7 %) and retains useful performance under salt-and-pepper noise

and moderate masking, but is markedly sensitive to Gaussian noise and catastrophically sensitive to even small translations. The confusion matrices, strength curves, and example images all converge on the same conclusion: the network has not learned translation invariant representations. The saliency map further illustrates that decisions are driven by localized stroke pixels whose absolute coordinates matter.

Condition	Accuracy	Precision	Recall	F1
Clean	0.9275	0.92733	0.926142	0.926455
Gaussian	0.7735	0.834901	0.779051	0.769934
Salt-and-pepper	0.864	0.873001	0.863823	0.864468
Pixel masking	0.8295	0.833174	0.828694	0.828181
Image shift	0.3335	0.363083	0.335016	0.314678

Table 3. Robustness Metrics

4.12 Robustness Metrics

This is essentially a higher precision restatement of Table 1 (same five conditions, same ranking of difficulty). The numerical values confirm the earlier ranking and give slightly more decimal places for reporting.

Identical to Table 1: clean >> salt and pepper > pixel masking > Gaussian >> image shift. The consistency across the two tables indicates stable evaluation.

5. Discussion

The experimental results presented in this study reveal a clearly stratified robustness profile for the multilayer perceptron trained on the MNIST handwritten digit dataset. Under clean conditions, the MLP achieves an accuracy of approximately 92.75%, confirming that the fully connected architecture is capable of learning effective nonlinear decision boundaries for ten class digit classification when the input distribution matches the training distribution. However, the introduction of controlled perturbations at evaluation time exposes substantial and perturbation-specific vulnerabilities that merit detailed interpretation.

5.1 Interpretation of Perturbation-Specific Robustness

The observed robustness hierarchy—clean salt and pepper > pixel masking > Gaussian >> image shift—can be understood in terms of how each perturbation interacts with the representational structure learned by a fully connected network. Salt and pepper noise, which corrupts only a sparse fraction of pixels with extreme intensity values, leaves the majority of informative stroke pixels intact. Because the MLP integrates evidence across all 784 input dimensions, the loss or corruption of a small number of individual pixels is effectively averaged out by the remaining signal. This explains the relatively modest accuracy decline to approximately 86.4% even at a 10% corruption rate and is consistent with the observation that the salt and pepper confusion matrix remains close to the clean baseline.

Pixel masking, which removes contiguous square regions of the image, produces a moderate degradation (accuracy falling from 92.6% at patch size 2 to 74.4% at patch size 10). The gradual decline indicates that the MLP distributes its reliance across multiple spatial regions of the digit rather than depending on a single critical area. Nevertheless, as the masked region grows, an increasing proportion of diagnostically important stroke pixels is eliminated, eventually overwhelming the residual information. The confusion matrix at moderate masking levels retains a visible diagonal, suggesting that partial stroke information is often sufficient for correct classification.

Gaussian noise, which adds zero-mean random fluctuations to every pixel simultaneously, produces a more pronounced degradation (accuracy dropping to 67.9% at $\sigma = 0.25$). Unlike sparse impulse noise, Gaussian noise perturbs all input dimensions concurrently, introducing correlated intensity variations along stroke boundaries and in background regions. The MLP, having learned weight configurations tuned to the clean intensity distribution, interprets these perturbations as meaningful signal changes, leading to systematic misclassifications particularly among digit pairs that share similar stroke geometries (e.g., 3/8, 4/9, 7/1). The more diffuse confusion matrix under Gaussian noise corroborates this interpretation.

The most striking result is the catastrophic sensitivity to image translation. Accuracy collapses from 75.55% at a 1-pixel shift to 33.35% at 2 pixels and to below 5% at 4–5 pixels. This extreme vulnerability is an inherent consequence of the fully connected architecture: each hidden unit receives weighted contributions from fixed pixel positions, and the network effectively encodes digit identity in terms of absolute spatial coordinates rather than relative geometric relationships. A translation displaces high saliency stroke pixels away from the positions to which the learned weights are tuned, rendering the previously informative features unrecognizable to the classifier. The near uniform off diagonal confusion matrix under translation confirms that the model's predictions become essentially uncorrelated with the true class labels once alignment is disrupted.

5.2 Architectural Implications and Comparison with Convolutional Approaches

The translation sensitivity observed here stands in sharp contrast to the performance of convolutional neural networks reported in the literature. CNN architectures such as those proposed by Assiri [16] and Hirata and Fujiyoshi [17] achieve MNIST accuracies exceeding 99.8% under clean conditions and, critically, possess built-in translation equivariance through weight sharing and pooling operations. These architectural priors enable CNNs to recognize digit features regardless of their absolute position within the image, a property entirely absent in the MLP studied here. The present results therefore quantify the practical cost of forgoing convolutional inductive biases: while the MLP can approximate competitive clean accuracy with sufficient capacity, it cannot generalize across spatial transformations without explicit architectural support.

This finding aligns with the observations of Thahiruddin [19], who noted that MLPs and logistic regression models receive comparatively limited attention in robustness studies despite their continued use as baseline classifiers. The present work extends that line of inquiry by providing a systematic, multi-perturbation evaluation with quantitative strength dependent curves and qualitative visual explanations.

5.3 Saliency Analysis and Explainability

The gradient based saliency maps generated in this study provide complementary insight into the MLP's decision mechanism. The concentration of saliency on the primary stroke of the predicted digit (e.g., the vertical bar of digit "1") confirms that the network bases its classification on localized, high contrast pixel regions. Importantly, the spatial specificity of these saliency patterns directly explains the translation vulnerability: because the network's decision depends on the absolute coordinates of salient pixels rather than on their shape or relational configuration, even a small displacement moves the critical evidence outside the receptive fields of the relevant hidden units. This observation reinforces the argument that gradient based explanations, while appropriate for MLPs, also reveal the architectural limitations that convolutional or attention based mechanisms are designed to overcome.

5.4 Practical Relevance

From an applied perspective, the robustness profile characterized here has direct implications for the deployment of simple neural classifiers in real-world digit recognition scenarios such as cheque processing, form reading, and mobile input systems. In such environments, input images are subject to scanner noise, partial occlusion from folds or stains, and positional variability due to inconsistent user alignment. The present results indicate that an MLP-based system would tolerate moderate impulse noise and small occlusions but would fail catastrophically under even minor positional misregistration. Preprocessing steps such as centroid alignment, bounding box normalization, or elastic registration would therefore be essential prerequisites for any practical MLP based recognition pipeline. Alternatively, the adoption of convolutional or spatial transformer architectures would address translation sensitivity at the model level.

5.5 Limitations

Several limitations of the present study should be acknowledged. First, the MLP architecture evaluated here is a single representative configuration; deeper or wider fully connected networks, or those incorporating dropout and batch normalization, may exhibit somewhat different robustness characteristics. Second, the perturbations were applied exclusively at evaluation time; training with augmented or perturbed data (e.g., random shifts, noise injection) could substantially improve robustness and represents an important direction not explored here. Third, adversarial perturbations generated by methods such as FGSM were excluded from the primary experimental scope, and their interaction with the identified vulnerability modes remains to be investigated. Fourth, the study focuses on a single benchmark dataset (MNIST); generalization to more complex digit or character recognition tasks (e.g., multi-writer datasets, non-Latin scripts) may reveal additional failure modes.

6. Conclusion

This study systematically evaluated the robustness of a multilayer perceptron trained on the MNIST handwritten digit dataset under four families of controlled input perturbations: additive Gaussian noise, salt-and-pepper impulse noise, contiguous pixel masking, and spatial translation. The principal findings are as follows.

First, the MLP achieves a strong clean-test accuracy of 92.75%, demonstrating that fully connected nonlinear classifiers remain viable for standard digit classification when input conditions match the training distribution. Second, the model exhibits moderate resilience to sparse and localized degradations: accuracy remains above 83% under salt and pepper noise at 10% corruption and above 74% under pixel masking up to 10 × 10-pixel patches. Third, performance degrades more substantially under dense additive Gaussian noise, reflecting the network's sensitivity to global intensity perturbations that shift the input distribution away from the learned manifold. Fourth, and most critically, the MLP is essentially non invariant to spatial translation, with accuracy collapsing below 34% at a mere 2-pixel displacement and approaching chance level (approximately 10%) at shifts of 4–5 pixels.

These results primarily proved that the robustness of a fully connected architecture is highly perturbation-specific and fundamentally constrained by the absence of spatial inductive biases such as weight sharing, local connectivity, and pooling. The gradient based saliency analysis further confirmed that the network's decisions are anchored to the absolute coordinates of high importance stroke pixels, providing a mechanistic explanation for the observed translation fragility.

The practical implication is clear: while MLPs can serve as lightweight and interpretable classifiers under well-controlled conditions, their deployment in real world recognition systems exposed to positional variability, sensor noise, or partial occlusion requires either careful preprocessing (alignment, normalization) or the adoption of architectures with inherent spatial invariance, such as convolutional neural networks. Future work should investigate the effect of training time data augmentation, ensemble strategies, and hybrid architectures on the robustness profile characterized here, as well as extend the evaluation to adversarial perturbations and more diverse handwritten recognition benchmarks.

Finally, this study demonstrates that evaluating robustness under realistic perturbations provides insights beyond conventional benchmark accuracy. The findings establish a reproducible evaluation framework for lightweight neural architectures and offer guidance for selecting appropriate recognition models in practical handwriting recognition applications where robustness is as important as predictive accuracy.

References

[1] Akbulut, U., Oniz, Y. (2025). Using convolutional neural networks and multi layer perceptron in Arabic handwritten digit recognition. *2025 9th International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT)* (p. 1–6). Ankara, Turkiye. https://doi.org/10.1109/ISMSIT67332.2025.

[2] Li, Y. (2022). Research and application of deep learning in image recognition. *Proceedings of the*

Conference (p. 994–999). (Jan 21–23).

[3] Sun, X., Liu, Y., Xie, L., He, X., Ma, X. (2022). Single check method of relay protection fixed value based on OCR image recognition. *Proceedings of the Conference* (p. 1101–1105). (Aug 25–27).

[4] Amudha, J., Thakur, M. S., Shrivastava, A., Gupta, S., Gupta, D., Sharma, K. (2021). Wild OCR: Deep learning architecture for text recognition in images. *Proceedings of the Conference* (pp. 499–506). (Jul 19–22).

[5] Nikoghosyan, K. (2022). Main trends in the development of science in the XXI century. *Main Trends in the Development of Science in the XXI Century* (p. 35–41).

[6] Verma, P., Foomani, G. (2022). Improvement in OCR technologies in postal industry using CNN-RNN architecture: Literature review. *International Journal of Machine Learning and Computing, 12*(5). https://doi.org/10.18178/ijmlc.2022.12.5.1095.

[7] Albattah, W., Albahli, S. (2022). Intelligent Arabic handwriting recognition using different standalone and hybrid CNN architectures. *Applied Sciences, 12*(19), 10155. https://doi.org/10.3390/app121910155.

[8] Ghosh, M. M. A., Maghari, A. Y. (2017). A comparative study on handwriting digit recognition using neural networks. *Proceedings of the Conference* (p. 77–81). (Oct 16–17).

[9] Arafat, S. Y., Ashraf, N., Iqbal, M. J., Ahmad, I., Khan, S., & Rodrigues, J. J. (2022). Urdu signboard detection and recognition using deep learning. *Multimedia Tools and Applications, 81*(9), 11965–11987. https://doi.org/10.1007/s11042-020-10175-2.

[10] Raza, S. A., Farooq, M. S., Farooq, U., Karamti, H., Khurshaid, T., Ashraf, I. (2025). A convolutional neural network based optical character recognition for purely handwritten characters and digits. *Computers, Materials and Continua, 84*(2), 3149–3173.

[11] Reddy, B. P., Reddy, R. S., Vasem, P. S., Venkatesh, P., Rajashekaran, S. (2022). Handwritten digit recognition using SVM algorithm in machine learning. *International Journal of Creative Research Thoughts, 10*(6). Retrieved October 30, 2024, from https://ijcrt.org/papers/IJCRT22A6520.pdf.

[12] Assegie, T., Nair, P. (2019). Handwritten digits recognition with decision tree classification: A machine learning approach. *International Journal of Electrical and Computer Engineering, 9*(5), 4446–4451. https://doi.org/10.11591/ijece.v9i5.pp4446-4451.

[13] Wang, Y., Wang, R., Li, D., Adu-Gyamfi, D. (2019). Improved handwritten digit recognition using quantum k-nearest neighbor algorithm. *International Journal of Theoretical Physics, 58*(7), 2331–2340. https://doi.org/10.1007/s10773-019-04124-5.

[14] Sheikh, R., Patel, M. (2019). Handwritten digit recognition using different dimensionality reduction techniques. *International Journal of Recent Technology and Engineering, 8*(2), 999–1002. https://doi.org/10.35940/ijrte.B1798.078219.

[15] Monica, R. F., Lavanya, K. (2019). Handwritten digit recognition of MNIST data using consensus clustering. *International Journal of Recent Technology and Engineering, 7*(6), 1969–1973. Retrieved June 21, 2024, from https://www.ijrte.org/wp-content/uploads/papers/v7i6/F2408037619.pdf.

[16] Assiri, S. (2020). A simple CNN model for MNIST handwritten digits classification. *International Journal of Advanced Computer Science and Information Technology, 11*(5), 1517–1524.

[17] Hirata, Y., Fujiyoshi, H. (2021). EnsNet: An ensemble of convolutional neural networks for robust digit

recognition. Retrieved July 10, 2024, from https://arxiv.org/abs/2008.10400.

[18] An, H., Sun, C., Wang, X. (2018). Deep ensemble convolutional neural network for digit recognition. *In 15th International Conference on Computer Science and Information Technology (ICCSIT)* (Vol. 10, p. 309–313).

[19] Thahiruddin, M., Khotijah, S., Fajar, M., El Farras, A. (2025). A robustness study of multi-layer perceptrons and logistic regression to data perturbation: MNIST dataset. *Zeta-Math Journal, 10*(1), 39–50.

[20] Mondal, R., Pal, T., Dey, P. (2024). A hybrid regularized multilayer perceptron for input noise immunity. *IEEE Transactions on Artificial Intelligence, 5*(1), 115–126. https://doi.org/10.1109/TAI.2022.3225124.

[21] Zhang, S. F., Qian, Z., Huang, K. Z., Zhang, R., Xiao, J. M., He, Y., Lu, C. Y. (2023). Robust generative adversarial network. *Machine Learning, 112*(12), 5135–5161. https://doi.org/10.1007/s10994-023-06367-0.

[22] Qiao, J., Wang, G., Li, W., Chen, M. (2018). An adaptive deep Q-learning strategy for handwritten digit recognition. *Neural Networks, 107*, 61–71. https://doi.org/10.1016/j.neunet.2018.02.010.

[23] Zhao, H. H., Liu, H. (2020). Multiple classifiers fusion and CNN feature extraction for handwritten digits recognition. *Granular Computing, 5*(3), 411–418. https://doi.org/10.1007/s41066-019-00158-6.

[24] Chen, F. Y., Chen, N., Mao, H. Y., Hu, H. L. (2019). Assessing four neural networks on handwritten digit recognition dataset (MNIST). *arXiv preprint*. https://doi.org/10.48550/arXiv.1811.08278.