

Article

Enhancing the Automatic Recognition Accuracy of Imprinted Ship Characters by Using Machine Learning

Abdulkabir Abdulraheem, Jamiu T. Suleiman  and Im Y. Jung * 

School of Electronic and Electrical Engineering, Kyungpook National University, Daegu 41566, Republic of Korea; aaoabdul@gmail.com (A.A.); jamiu.suleiman111@gmail.com (J.T.S.)

* Correspondence: iyjung@ee.knu.ac.kr

Abstract: In this paper, we address the challenge of ensuring safe operations and rescue efforts in emergency situations, for the sake of a sustainable marine environment. Our focus is on character recognition, specifically on deciphering characters present on the surface of aged and corroded ships, where the markings may have faded or become unclear over time, in contrast to vessels with clearly visible letters. Imprinted ship characters encompassing engraved, embroidered, and other variants found on ship components serve as vital markers for ship identification, maintenance, and safety in marine technology. The accurate recognition of these characters is essential for ensuring efficient operations and effective decision making. This study presents a machine-learning-based method that markedly improves the recognition accuracy of imprinted ship numbers and characters. This improvement is achieved by enhancing data classification accuracy through data augmentation. The effectiveness of the proposed method was validated by comparing it to State-of-the-Art classification technologies within the imprinted ship character dataset. We started with the originally sourced dataset and then systematically increased the dataset size, using the most suitable generative adversarial networks for our dataset. We compared the effectiveness of classic and convolutional neural network (CNN)-based classifiers to our classifier, a CNN-based classifier for imprinted ship characters (CNN-ISC). Notably, on the augmented dataset, our CNN-ISC model achieved impressive maximum recognition accuracy of 99.85% and 99.7% on alphabet and digit recognition, respectively. Overall, data augmentation markedly improved the recognition accuracy of ship digits and alphabets, with the proposed classification model outperforming other methods.

Keywords: imprinted ship characters; automatic recognition; recognition accuracy; dataset augmentation; machine learning classifiers



Citation: Abdulraheem, A.; Suleiman, J.T.; Jung, I.Y. Enhancing the Automatic Recognition Accuracy of Imprinted Ship Characters by Using Machine Learning. *Sustainability* **2023**, *15*, 14130. <https://doi.org/10.3390/su151914130>

Academic Editors: Mourade Azroul, Azidine Guezzaz, Imad Zeroual, Azeem Irshad, Jamal Mabrouki, Said Benkirane and Shehzad Ashraf Chaudhry

Received: 30 August 2023

Revised: 18 September 2023

Accepted: 22 September 2023

Published: 24 September 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The recognition and identification of imprinted letters and digits on ship components are vital tasks in marine technology applications, including maintenance, identification, and critical operational labels. Accurate recognition of these characters is crucial for both automated systems and human operators, to interpret and understand the information conveyed by engravings. The accurate and swift recognition of these characters is not just a technological pursuit but a fundamental element for upholding the sustainability of marine environments and operations. However, a significant challenge exists within marine technology—recognizing characters on the weathered, corroded surfaces of aged ships and their components. Unlike the clear markings on new vessels, these characters may have blurred or deteriorated over years of exposure to harsh maritime conditions. Even though efforts are made to update characters that have worn out and become blurry over time, there is bound to be a difference in their new state. And there are ships that are constantly updating and those that are not. However, replacing aged components with new ones not only poses environmental concerns but also economic challenges. Therefore, it is essential to find ways to identify and maintain aging components in their current

state, to promote sustainability. Despite advancements in character recognition technology, deciphering these aged and obscured characters presents a unique and demanding task.

In recent years, significant progress has been made in the field of computer vision and pattern recognition. This has enabled the development of robust recognition systems that utilize machine learning techniques, such as classic classifiers and convolutional neural network (CNN)-based classifiers, to achieve high accuracy in character recognition tasks. CNNs have become essential for character recognition, excelling at capturing fine character details for accurate identification. Often, due to dataset characteristics and model design, different CNN architectures exhibit varying performance. However, in the field of ship character recognition, several challenges and limitations persist within the existing methods. These include the scarcity of comprehensive ship character datasets, difficulties in handling the variability of ship characters, the impact of environmental factors on character degradation, the complexity of the backgrounds on which characters are imprinted, the need for real-time recognition, and limited generalization capabilities. Unlike other character recognition datasets, such as handwritten datasets, ship character images are scarce, and this scarcity makes it difficult to obtain sufficient training data for the model to learn the variations in different imprinted ship characters. As a result, the model may overfit, memorizing the training data rather than generalizing effectively to new, unseen data. To address this challenge, we used generative adversarial networks (GANs), which have proven effective in generating synthetic data that closely resemble real samples [1–3]. GANs can incorporate diverse patterns and variations present in imprinted characters, which helps the model generalize better.

Considering the inherent complexity and variability of characters found on ships and their components, as seen in Figure 1, exploring multiple classifiers to identify the most suitable approach to achieving accurate recognition is crucial. This study was motivated by the urgent need for effective solutions that would bridge the gap between maritime safety and sustainability. Specifically, we aimed to develop a robust character recognition system tailored to the complex conditions of weathered ship surfaces and components. By doing so, we would contribute to the broader goals of ensuring safe and environmentally responsible maritime practices. This paper outlines our methodological approach, which involves leveraging machine learning, data augmentation, and State-of-the-Art classification techniques to enhance the accuracy of recognizing ship characters in challenging real-world conditions. Through rigorous evaluation, we demonstrate the effectiveness of our proposed system in addressing this critical maritime challenge. In our study, we conducted evaluations on State-of-the-Art classifiers, by considering relevant and recent works in the domain. By comprehensively assessing various classifiers, we aimed to propose an optimal model that would demonstrate superior performance in recognizing ship characters effectively. Our study includes the evaluation of cutting-edge CNN-based classifiers as well as well-known classic classifiers, such as Gaussian Naive Bayes (GNB), Random Forest (RF), K-Nearest Neighbors (KNNs), Support Vector Machines (SVMs), Stochastic Gradient Descent (SDG), and Decision Trees (DT). Among deep learning methods, CNNs have garnered considerable attention, due to their ability to operate directly on original data without requiring extensive data transformations. This property enables CNNs to preserve the information present in the original data to a greater extent, distinguishing them from other approaches, such as SVMs [4]. CNNs have shown exceptional performance in image recognition tasks, especially when dealing with complex patterns and intricate details. As a result, we included CNN-based models in our evaluation, to determine whether they could achieve near-perfect prediction accuracy on the imprinted digit and alphabet datasets. This study focused on optimizing the model architecture, to enhance recognition accuracy. We achieved this by systematically exploring minor modifications to critical hyperparameters in each CNN model. Specifically, we investigated variations in activation functions, learning rates, optimizers, batch sizes, and epochs, while maintaining consistency in the number of convolutional layers, dense layers, and pool sizes. This meticulous approach enabled the fine-tuning of these selected hyperparameters for each CNN model, leading to the

identification of a suitable architecture with optimal hyperparameters tailored specifically to the dataset, thereby considerably improving recognition accuracy. In addition, our datasets were compared to cutting-edge hybrid classifiers, such as CNN-SVMs [5] and CNN-RF [6]. Furthermore, we developed a CNN-based classifier model for imprinted ship characters (CNN-ISC) and evaluated its performance, by comparing it to other known classifiers, aiming at providing insights into the remarkable effectiveness of our model in recognizing the diverse range of imprinted characters present on ship components.



Figure 1. Sample Images of Source Dataset.

For classifier performance evaluation, we used standard metrics, including the F1 score, precision, recall, and accuracy. The F1 score is the harmonic mean of precision and recall that provides a single number to compare the overall performances of different classifiers. It balances both precision and recall and is often used when both are important. Precision measures how often a classifier correctly identifies positive samples. High precision indicates a low false positive rate. The recall value is a performance metric that measures the percentage of positive instances correctly identified by the model [7–10]. Recall measures how often a classifier correctly identifies positive samples out of all actual positive samples. A high recall indicates a low false negative rate. These metrics provide comprehensive insights into the classifiers' ability to correctly classify the imprinted characters, considering both the precision of positive predictions and the ability to identify true positive instances. By analyzing the performance across these metrics, we could assess the classifiers' overall effectiveness in recognizing the digit and alphabet datasets. By harnessing GAN models, we generated more extensive and diverse datasets. Subsequently, through a careful evaluation process, we compared the performance of the classifiers across these diverse dataset variations.

The successful implementation of this research will markedly advance the maritime industry, by optimizing maintenance and replacement schedules, facilitating part re-use and inventory management, improving accuracy and efficiency, enhancing accident investigation and safety standards, and promoting standardization and interoperability. The precision in character recognition has profound implications for human operators who rely on interpreting and comprehending the information conveyed by these engravings. However, this study goes beyond the realm of character recognition alone, casting a considerable influence on the maritime industry's sustainability. Enhancing ship character recognition extends beyond achieving heightened accuracy in vessel and component identification, to accident prevention, improved regulatory adherence, and, ultimately, a more sustainable and environmentally responsible maritime industry. The following highlights underscore the key findings and contributions of our research:

- We utilized GANs for data augmentation, improving recognition accuracy by incorporating diverse patterns in limited special typed images of imprinted characters on ship components.
- The CNN-ISC model achieved 99.85 and 99.7% accuracy, outperforming other classifiers for both digits and alphabets on ship components.
- The CNN-ISC model's high precision and recall make it valuable for ship character recognition, enhancing maritime safety measures.

The rest of the paper is structured as follows: Section 2 provides an overview of relevant studies on recent recognition models for similar tasks. We introduce a CNN-based approach for recognizing imprinted digit images in Section 3. In Section 4, we present the results of our comprehensive evaluation of the classifiers on the digit and alphabet datasets. We discuss the classifiers' performance, highlighting notable improvements achieved when trained on augmented datasets. We also analyze our findings, in relation to previous works. Finally, we conclude the paper and discuss potential future research directions.

2. Related Work

In the analysis of ship character identification, we conducted an extensive review of the existing literature on ship identification, and we investigated the utilization of data augmentation techniques in this domain. The aim was to understand the advancements made in ship identification methods and to examine how researchers have incorporated data augmentation to enhance the accuracy and robustness of ship recognition systems. By exploring the research conducted on ship identification and the integration of data augmentation, we sought to gain valuable insights into the effectiveness of these approaches and their impact on improving ship recognition systems.

Accumulating research on image recognition [11–17] has enabled the development of novel algorithms and techniques. This progress has facilitated the application of image recognition in various domains. Over time, these algorithms have undergone significant advancements, becoming increasingly sophisticated. To enhance classifier performance, in terms of accuracy, running time, and computational complexity, researchers often employ ensemble methods, which involve combining multiple classifiers. Several studies have explored the use of ensemble methods on well-known datasets, showing their potential for improving classifier outcomes.

The authors of [18] introduced the RMA (ResNet-Multiscale-Attention) model, a fine-grained classification approach for recognizing navigation marks in camera images. This model incorporates an attention mechanism that combines feature maps of three different scales, to capture subtle differences among similar navigation marks. It was trained on a dataset of 10,260 navigation mark images, achieving an accuracy of approximately 96% for classifying 42 types of navigation marks. This value markedly exceeded that of the ResNet-50 model, which achieved around 94%. The authors also provided visualization analyses that demonstrated their model's ability to extract attention regions and essential characteristics of navigation marks, thereby further validating their model effectiveness.

BoxPaste, a powerful data augmentation method tailored for ship detection in SAR imagery, was introduced in [19]. This approach, which involves pasting ship objects from one SAR image onto another, exhibits greater performance on the SAR ship detection dataset than the baseline methods. The authors also proposed a principle for designing SAR ship detectors, highlighting the potential benefits of lighter models. The effectiveness of their data augmentation scheme was further demonstrated by integrating it with RetinaNet and ATSS, resulting in impressive performance gains. In [20–23], the authors addressed the challenge of detecting and tracking ships from video streams in a monitored area. These works focused on developing systems that can effectively identify and track vessels as they enter designated areas.

A comprehensive approach to recognizing vessel plate numbers through an end-to-end method was introduced in [24]. The method comprises two key stages: vessel plate number detection and recognition. In the detection stage, a deep CNN is employed, to

identify the bounding boxes encompassing the vessel plate numbers within the images. Subsequently, in the recognition stage, a Long Short-Term Memory (LSTM) network is utilized, to accurately decipher the text contained within the detected bounding boxes. To assess the effectiveness of their proposed method, the authors conducted evaluations, using a dataset comprising 1000 vessel plate number images. Remarkably, the method achieved impressive text detection and recognition rates of 96.94% and 93.54%, respectively. Comparative analysis to other existing methods revealed that the proposed approach outperformed all other techniques, highlighting its superior performance and efficacy in vessel plate number recognition.

The authors of [25] conducted an extensive survey, to thoroughly examine the current State-of-the-Art methods in scene text detection and recognition. The study provided a comprehensive overview of these two tasks and delved into the significant influence of deep learning techniques on their advancement. The authors further explored recent progress in scene text detection and recognition, encompassing novel deep learning architectures, the utilization of large-scale datasets, and the introduction of improved evaluation metrics. They also discussed the challenges that remain in scene text detection and recognition, including detecting text in low-resolution images, in images with cluttered backgrounds, and in images with non-Latin characters. In [26], the authors proposed a novel rotation-based framework for arbitrary-oriented scene text detection in natural scene images. The framework comprises two main components: a Rotation Region Proposal Network (RRPN) and a Rotation Region of Interest (RRoI) pooling layer. The RRPN generates inclined proposals with text orientation angle information, and the RRoI pooling layer projects arbitrary-oriented proposals to a feature map for a text region classifier. The authors evaluated their framework on three real-world scene text detection datasets and demonstrated its superiority, in terms of effectiveness and efficiency, over previous approaches. In [27], a novel method for omnidirectional scene text detection was proposed, based on a sequential-free box discretization approach that allows for the detection of text in images with arbitrary orientations. The method was evaluated on the ICDAR 2015 omnidirectional scene text detection benchmark, and it achieved State-of-the-Art performance.

In [28], the authors proposed an algorithm for scene text detection using multibox and semantic segmentation. The algorithm first uses a multibox detector to generate a set of text proposals. These proposals are then passed on to a semantic segmentation model, which is used to predict the text region in each proposal. The algorithm was evaluated on the ICDAR 2015 scene text detection benchmark, and it achieved State-of-the-Art performance. A novel end-to-end panoptic segmentation method called Max-DeepLab was proposed in [29]. The method is based on the Mask R-CNN framework, and it uses a novel mask transformer module to improve model performance. The method was evaluated on the COCO panoptic segmentation benchmark, and it achieved State-of-the-Art performance. In addition, in [30], a method for ship plate recognition was introduced, using a Fully Convolutional Network (FCN), a type of deep neural network specialized in image segmentation. The authors argued that FCNs are well suited to ship plate recognition, due to their ability to handle challenges such as occlusion, rotation, and varying illumination in ship plate images. The FCN was trained on a dataset of ship plate images, and its performance was evaluated on a separate test set. The results showed impressive accuracy of 95%. Additionally, performance comparisons with other ship plate recognition methods demonstrated the FCN's superiority.

3. Methodology

This study used datasets containing both digits and alphabets that represented the imprinted characters commonly seen on ship components. These datasets were carefully selected, by considering variations in the shapes, sizes, and styles of the engravings, to make them more realistic. We emphasize the significance of using datasets specifically derived from ship imprints and related components, to ensure the authenticity and precision of the generated digit and alphabet images. In addition to the baseline experiments, we

investigated the impact of dataset augmentation using GANs on classifier performance. GANs are powerful tools for data augmentation, because they can generate synthetic data that closely resemble real-world samples [1–3]. We explored the effectiveness of GAN-based augmentation techniques, such as Wasserstein GAN with a Gradient Penalty (WGAN-GP) and WGAN with Divergence (WGAN-DIV) [31], in improving the recognition performance of classifiers. The WGAN-GP and WGAN-DIV models, among several other GAN models, demonstrated exceptional performance during execution on the engraved digit dataset.

Figure 2 illustrates our system model for identifying imprinted ship characters. The process begins with data preprocessing, which involves preparing the raw images of imprinted ship images by normalizing the data, to ensure accurate classification [32], followed by data augmentation, to enhance the dataset by applying techniques such as the GAN. This augmentation increases diversity and enhances the classification models' robustness. The next phase involves assessing classification models, including various machine learning models designed to recognize imprinted ship images. This phase includes training and fine-tuning the classifier algorithms, to optimize their performance. The classification phase also includes the model validation and evaluation phase. Model validation is the process of testing a trained classifier on a separate dataset, to ensure its performance and generalizability. Model evaluation is the process of assessing a classifier's effectiveness on a dataset, using metrics, such as the F1 score, accuracy, recall, and precision. These metrics provide a comprehensive analysis of the classifier's performance. The recognition phase involves the application of trained classification models, to accurately identify and recognize imprinted ship images. This step enables the system to discern precisely the specific digits represented by the engravings. Finally, the information retrieval system integrates the classification and recognition components. It serves as a cohesive system that allows users to retrieve relevant information based on recognized imprinted ship images. The engraved digit recognition workflow is described in [33].

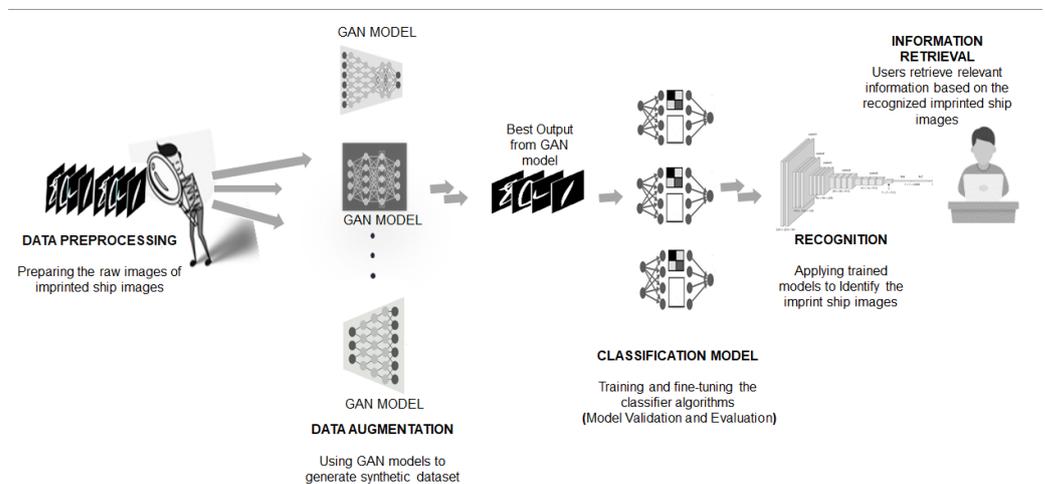


Figure 2. An imprinted Digit and Alphabets Recognition System Model with Data Augmentation Technique.

3.1. Data Collection and Preprocessing

The ship-imprinted character image dataset comprises characters that are etched, engraved, or inscribed onto ship surfaces, such as metal plates or panels (see Figure 1). These characters are typically machine generated and serve various purposes in the maritime industry, including identification, labeling, and signage. However, due to factors such as physical wear, corrosion, exposure to harsh environmental conditions, and the passage of time, the legibility and visibility of these imprinted characters can be significantly compromised. This presents a considerable challenge to accurately identifying and recognizing the characters, particularly in real-world scenarios where the imprinted surfaces may have

undergone extensive deterioration. The ship-imprinted character image dataset exhibits variations in image quality, lighting conditions, and distortion caused by the engraving process itself (see Figure 3). These variations in the appearance and quality of imprinted characters pose significant difficulties for traditional recognition methods, necessitating the development of specialized approaches, to effectively address these challenges.



Figure 3. Collection of Character and Digit Samples from the Source Dataset

The datasets used in this study covered various imprinted characters, including digits 0–9 and 13 alphabets: A, C, D, E, I, L, M, N, O, P, R, S, and T. These characters were obtained from old or poorly maintained ships (Figure 1). These images were carefully selected, to support ship character identification and retrieval systems. Within the context of our research, the presence of a relatively small dataset encompassing only a few alphabets (13) introduces the potential risk of exacerbating mode collapse in the GAN used for data augmentation [33]. This concern informed our strategic decision to concentrate on a specific subset of characters (13 of 26 alphabet characters). Despite this limitation, we hold a strong belief in the broader applicability of our results and conclusions. The images exhibited variations in size and color, but they were preprocessed, to ensure consistency during training and analysis. Specifically, they were normalized to grayscale and resized to 56×56 pixels in width and height. The images were stored in standard formats, such as JPEG or PNG. The dataset encompassed various engraving styles commonly found on ships, including embossed, engraved, and painted characters, either individually or in combination. These characters represented ship identification numbers, hull markings, engine component identifiers, and other characters relevant to ship operations. This comprehensive approach enabled us to capture the diversity and complexity of the characters found in real-world ship components. By evaluating the classifiers' performance on both digit and alphabet datasets, we could identify any variations in recognition capabilities and gain a comprehensive understanding of their effectiveness.

3.2. Data Augmentation with GAN Models

We investigated the impact of dataset augmentation using GANs on classifier performance. GANs are powerful tools for data augmentation, as they can generate synthetic data that closely resemble real-world samples [3,31]. We started with an originally sourced dataset of approximately 100 images per character class, and we then systematically increased the dataset size, to about 200 images per character class, using the most suitable GANs for our dataset, WGAN-GP and WGAN-DIV [33]. This augmentation approach strikingly increased the size and diversity of the dataset, enabling the training of a more robust and improved performance of the classifiers.

3.3. Classifier Selection, Metrics, and Model Performance

The choice of classifiers for evaluation was based on their wide usage and effectiveness in character recognition tasks. We considered well-known classic classifiers, such as GNB, RF, KNNs, SVMs, SDG, and DT. These classifiers have demonstrated their efficacy in various pattern recognition applications, and they provided a solid foundation for our comparative analysis. Additionally, hybrid (CNN-RF and CNN-SVMs) and CNN-based classifiers were employed, to compare the performance of our CNN-ISC against these classifiers. The classifiers were evaluated, using standard evaluation metrics, including precision, recall, and F1 score. Table 1 contains each metric description. We implemented traditional classic algorithms, using Scikit-Learn.

Table 1. Summary of Classification Metrics.

Metric	Description	Formula	Interpretation
Accuracy	Measures the overall correctness of predictions.	$(TPs + TNs) / (TPs + TNs + FPs + FNs)$	High accuracy indicates good overall performance.
Precision	Measures the accuracy of positive predictions.	$TPs / (TPs + FPs)$	High precision indicates fewer false positive errors.
Recall	Measures the proportion of actual positives correctly predicted.	$TPs / (TPs + FNs)$	High recall indicates that most actual positives are correctly predicted.
F1 Score	A harmonic mean of precision and recall.	$2 * (Precision * Recall) / (Precision + Recall)$	Balances precision and recall, which is useful when there is an imbalance between classes.

True positives (TPs) are correctly predicted positive cases, true negatives (TNs) are correctly predicted negative cases, false positives (FPs) are incorrectly predicted positive cases, and false negatives (FNs) are incorrectly predicted negative cases. Classification reports were generated, providing detailed insights into each classifier's performance. The metrics were calculated for both the digit dataset and the selected alphabet characters, allowing for a comprehensive assessment of classifier performance across different imprinted character types. In our experimental setup, we split the dataset into training and testing sets, using a 70:30 ratio. By allocating 70% of the data for training, the model could learn the underlying patterns and relationships present in the data, while the remaining 30% was reserved for testing, to assess the model's performance on new, unseen data. During the training phase, the model optimized its parameters and learned from the training data, to minimize errors and improve accuracy. This iterative process continued until the model converged to a state where further training would not lead to significant improvements. After training, the model was then evaluated on the testing set, where it encountered new samples that were not part of the training data. This evaluation allowed us to gauge the model's performance in a real-world scenario, assessing its ability to make accurate predictions on unseen data.

For each CNN model, we aimed to understand the impact of minor modifications to certain hyperparameters. Specifically, we focused on altering the type of activation function, learning rate, optimizer, batch size, and number of epochs, while keeping the number of convolutional layers, dense layers, and the pool size constant. Through this systematic approach, we carefully adjusted these selected hyperparameters for each CNN model, and we documented the results, to identify the optimal parameter values. Our evaluation encompassed several CNNs [34–38]. Additionally, to explore the impact of network design and depth on dataset performance, we modeled our CNN-ISC as a variant of the CNN [34,38]. The CNN-ISC architecture is designed to learn hierarchical representations of the input data, through a series of convolutional and pooling layers. These layers are responsible for extracting important features from the input images and progressively reducing their spatial dimensions. The convolutional layers use a set of learnable filters, to detect patterns and local features in the images, while the max-pooling layers downsample the feature maps, focusing on the most relevant information.

As shown in Figure 4, the CNN-ISC architecture comprises three Conv2D layers, each followed by a ReLU activation function, to introduce nonlinearity. The first Conv2D layer has 32 filters with a 3×3 kernel, and the subsequent two Conv2D layers have 64 filters with 3×3 kernels. These convolutional layers are responsible for capturing the low-level and high-level features in the input images. After each Conv2D layer, a MaxPooling2D layer with a 2×2 pooling size is applied, to reduce the spatial dimensions of the feature maps. This step helps reduce computational complexity and focuses on the most salient features. A flatten layer is added, to transform the 2D feature maps into a 1D vector, preparing the data for the fully connected layers. Two dropout layers are inserted into the architecture, to prevent overfitting. The first dropout layer randomly drops out 50% of the neurons after the flatten layer, and the second dropout layer has the same dropout rate and comes after the first dense layer. The CNN-ISC architecture also includes two dense layers. The first dense layer has 1024 neurons with a ReLU activation function and is followed by L2 regularization, to further prevent overfitting. Finally, the output layer is a dense layer with 13 neurons using the softmax activation function. This allows the model to perform multi-class classification for the 13 alphabets. For digit recognition, we modified the dense layer to 10, corresponding to digits 0–9. The model was compiled using a learning rate of 0.001.

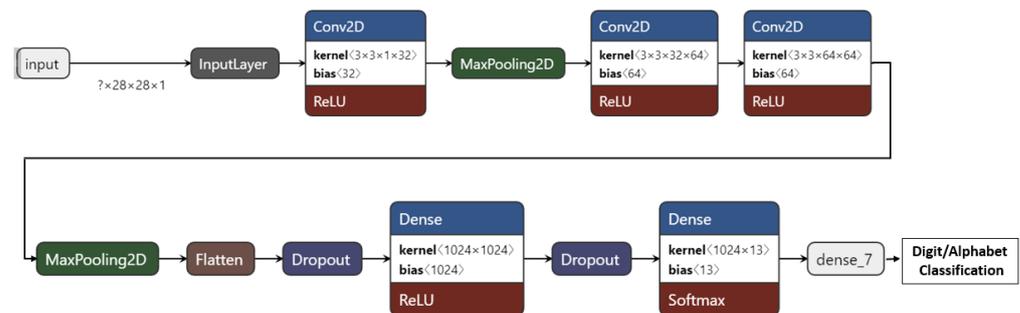


Figure 4. CNN-ISC Architecture.

The hybrid classifiers, CNN-SVMs and CNN-RF, use SVMs and RFs for binary classification, instead of softmax or sigmoid functions. CNNs are used to extract features from input data, capturing hierarchical representations. These extracted features are then fed into the SVMs and RF classifiers, which classify the features into their respective binary classes.

4. Evaluation and Analysis

First, we evaluated the performance of the classic classifiers on two datasets of imprinted ship characters: the imprinted ship digit dataset and the imprinted ship alphabet dataset. We used both the original dataset and a dataset augmented by synthetic data generated using the WGAN-GP and WGAN-DIV models.

We found that the augmented dataset significantly improved the performance of all the classifiers, with the most striking improvements seen for the KNNs and SVMs classifiers. For example, Table 2 shows that the accuracy of the KNNs classifier on the imprinted ship digit dataset increased from 26% to 94% when the augmented dataset was used. The SVMs classifier also showed notable improvement, with its accuracy increasing from 13.8% to 90.6%. Other classifiers also showed improvement, although the gains were not as pronounced. For example, the accuracy of the DT classifier on the imprinted ship digit dataset increased from 13.8% to 66.7%, and the accuracy of the RF classifier increased from 18% to 27.7%. The GNB classifier showed the least improvement, but it still showed some improvement, with its accuracy increasing from 12.7% to 21.6%.

Table 2. Classic Classifiers' Accuracies Across the Digit Datasets.

Classifier	Original Dataset	Augmented Dataset
KNNs	26	94
SVMs	13.8	90.6
DT	13.8	66.7
SDG	18.1	35.8
RF	18	27.7
GNB	12.7	21.6

The augmented dataset also significantly improved the performance of the classifiers on the imprinted ship alphabet dataset (Table 3). The KNNs classifier, for example, had its accuracy increased from 26% to 97%. The SVMs classifier showed notable accuracy improvement (from 20% to 96%). The DT classifier also exhibited substantial accuracy improvement (from 12.8% to 76%). The SDG classifier and RF classifier showed relatively lower improvements after augmentation, but they still showed some improvement. The SDG classifier achieved an initial accuracy of 4.10% on the original dataset, which improved to 51% with augmentation. Similarly, the RF classifier had an accuracy of 2.15% on the original dataset, which increased to 27% after augmentation. The GNB classifier showed the least improvement, but it still showed some improvement (from 2.57% to 40%). These results suggest that the WGAN-DIV and WGAN-GP augmented datasets can notably improve the performance of classifiers for imprinted ship alphabet recognition tasks. The augmentation technique effectively enriches the datasets and provides valuable information for capturing patterns and improving generalization capabilities.

Table 3. Classic Classifiers' Accuracies Across the English Alphabets Dataset.

Classifier	Original Dataset	Augmented Dataset
KNNs	26	97
SVMs	20	96
DT	12.8	76
SDG	4.10	51
RF	2.15	27
GNB	2.57	40

Tables 4 and 5 show the F1 scores of classic classifiers for the digit and alphabet datasets, where A and B represent the original and augmented datasets, respectively. The F1 score is a measure of a classifier's accuracy, incorporating both precision and recall. A high F1 score indicates that the classifier is both accurate and sensitive. Initially, the KNNs and SVMs classifiers had low F1 scores, of 26% and 20%, respectively, for the digit dataset. However, after augmentation by the generated datasets, WGAN-DIV and WGAN-GP, their scores improved significantly, to 97% and 96%, indicating the effectiveness of the augmented dataset. The DT classifier exhibited an initial F1 score of 12.8% for the digit and alphabet datasets, which increased to 76% after augmentation. The SDG and RF classifiers also showed improvements, with F1 scores of 4.10% and 2.15% increasing to 51% and 27%, respectively. The GNB classifier had relatively low F1 scores initially but still showed improvement after augmentation, reaching 40%. The F1 score served as a valuable metric to gauge the classifiers' recognition performance, as it provided a balanced measure of precision and recall. Digits with higher F1 scores demonstrated superior classification, showing a good balance between accurately identifying positive samples (recall) and minimizing FPs (precision). Various factors, including the data distribution and the complexity of the digits, affected classifier performance, leading to variations in their recognition abilities for different digits.

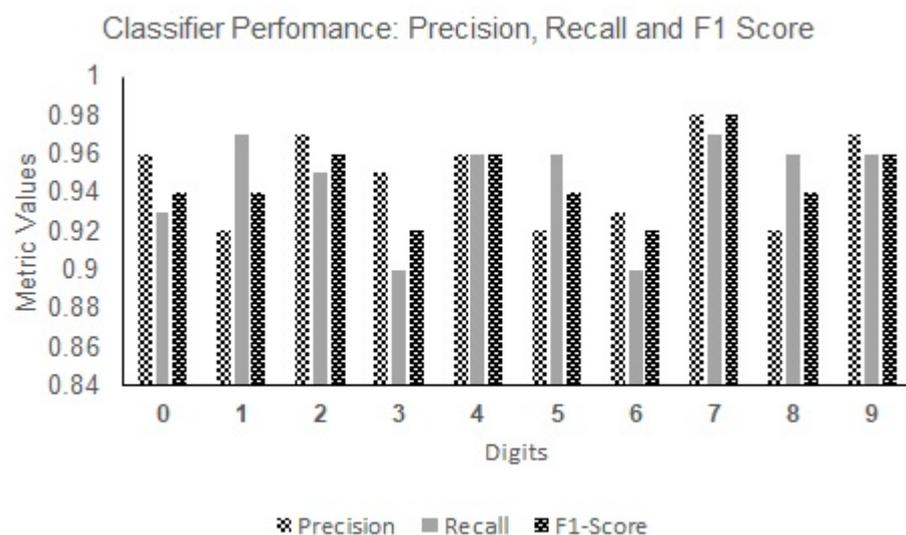
Table 4. Average F1 score (%) for Digits.

Classifier	F1 Score Original Dataset	F1 Score Augmented Dataset
KNNs	22	95
SVMs	06	90
SDG	06	33
RF	05	25
GNB	09	18
DT	13	66

Table 5. Average F1 score (%) for Alphabets.

Classifier	F1 Score Original Dataset	F1 Score Augmented Dataset
KNNs	18	97
SVMs	13	96
SDG	06	28
RF	04	31
GNB	03	29
DT	10	75

Digit 4 consistently demonstrated one of the best performances across various classifiers, with high F1 scores. For instance, the KNNs classifier (Figure 5) excelled in recognizing digit 4, achieving an impressive F1 score of 96%, showcasing its ability to accurately classify positive samples while minimizing FPs. Similarly, in the SVMs classifier (Figure 6), digit 4 achieved the highest F1 score of 94%, indicating accurate classification with well-balanced precision and recall. However, some classifiers faced challenges in recognizing certain digits. Figure 7 shows the performance of the RF classifier. RF struggled with digits, and the SGD classifier in Figure 8, although achieving relatively lower overall F1 scores, performed relatively well for digit 9. These findings provide valuable insights for selecting suitable classifiers and designing effective digit recognition models for various applications.

**Figure 5.** Metric Chart for Digits on the K-Nearest Neighbors Classifier.

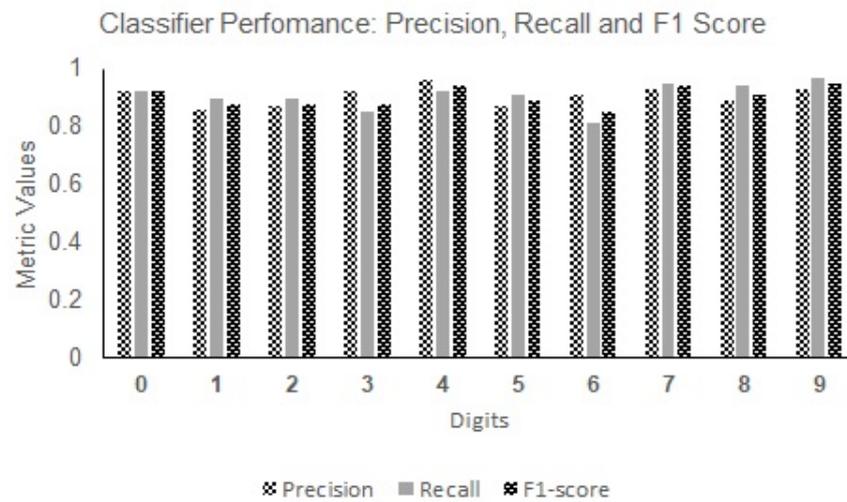


Figure 6. Metric Chart for Digits on the Support Vector Machines Classifier.

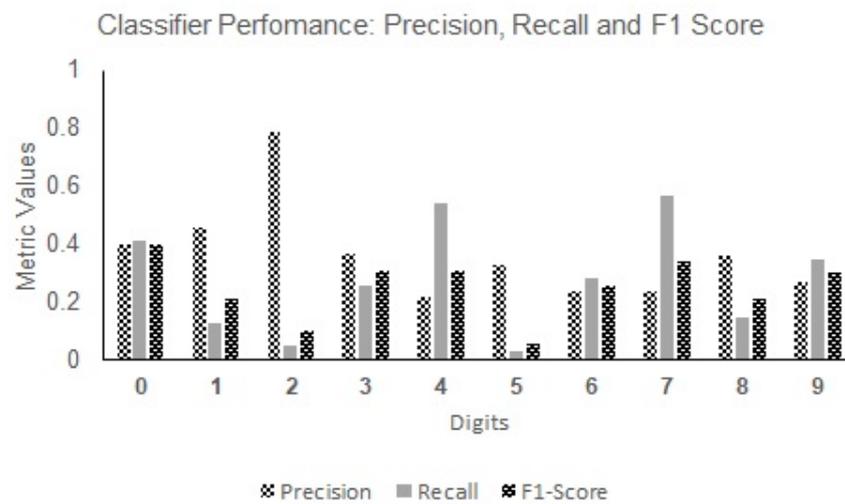


Figure 7. Metric Chart for Digits on the Random Forest Classifier.

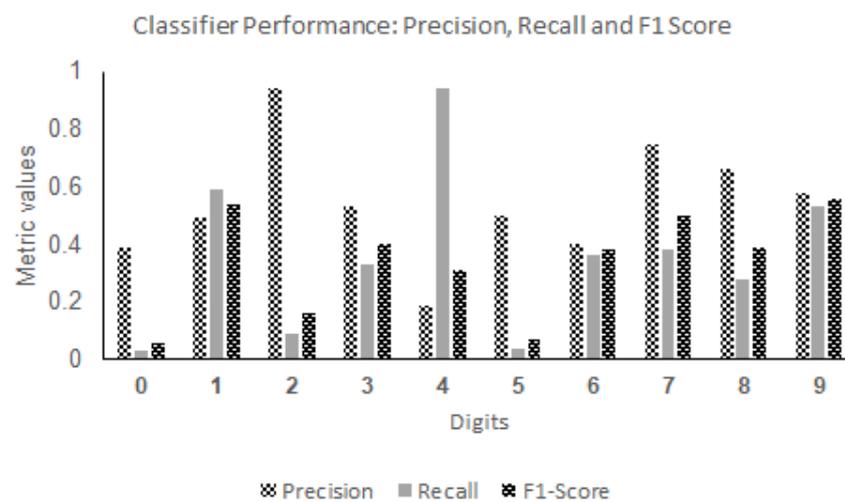


Figure 8. Metric Chart for Digits on the Stochastic Gradient Descent Classifier.

For the alphabets, the letters “C”, “P”, and “I” consistently achieved high F1 scores across various classifiers, including SVMs (Figure 9), KNNs (Figure 10), and GNB (Figure 11). These letters showcased excellent recognition capabilities, with F1 scores ranging from 0.43 to 0.99. Conversely, the letters “E”, “S”, and “R” exhibited relatively lower F1 scores across the classifiers, indicating the need for further optimization in classifier models, to improve their recognition accuracy. The F1 scores for these letters ranged from 0.19 to 0.97 across the KNNs, SVMs, SDG, DT, and GNB classifiers. The variations in performance highlight the impact of different classifiers on recognizing specific alphabet characters.

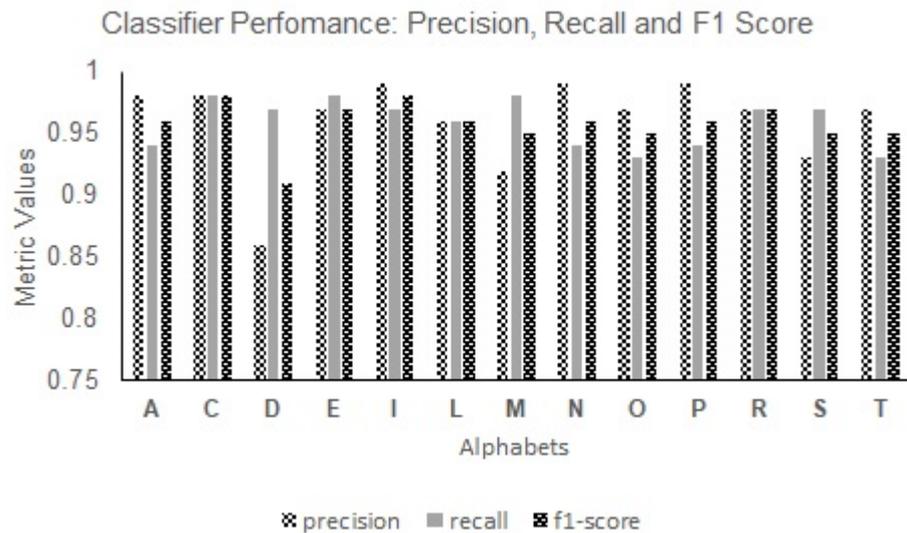


Figure 9. Metric Values for selected Alphabets on the Support Vector Machines Classifier.

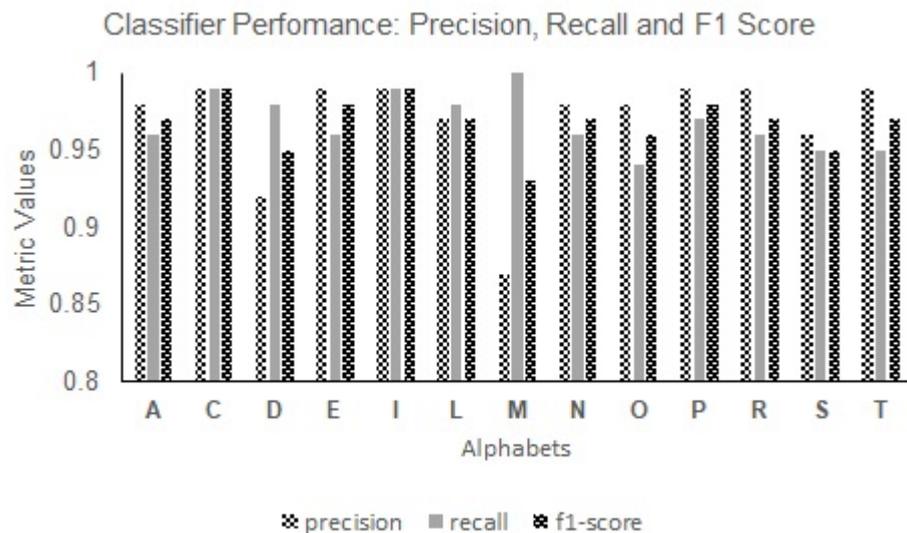


Figure 10. Metric Chart for selected Alphabets on the K-Nearest Neighbors Classifier.

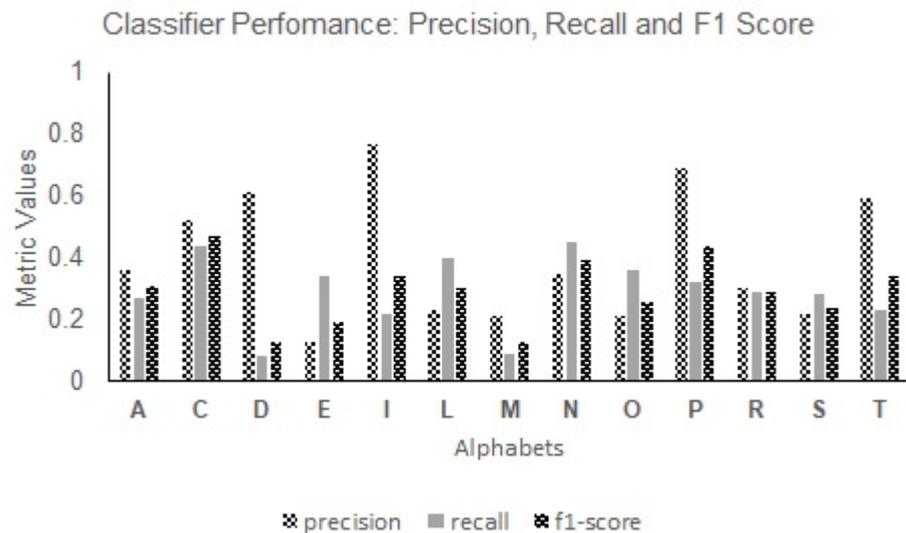


Figure 11. Metric Chart for selected Alphabets on the Gaussian Naive Bayes Classifier.

In our evaluation, we assessed the performance of selected CNN-based models for both digit and alphabet recognition. Tables 6 and 7 present the accuracy results of the CNN models on the original and augmented datasets for digits and alphabets, respectively. All the CNN models demonstrated a notable improvement in performance when trained on the GAN-augmented dataset compared to the original digit dataset. This suggests that the GAN augmentation technique effectively enhanced the models' ability to recognize and classify digits.

Table 6. CNN Accuracies for Digits.

Classifier	Original Dataset	Augmented Dataset
CNN [34]	31.8	99.43
CNN-ISC	10.8	99.7
CNN [35]	47.3	99.0
CNN [36]	43.2	99.5
CNN [38]	48.7	99.2
CNN-SVMs [5]	15.7	97.33
CNN-RF [6]	10.4	96.8

Table 7. CNN Accuracies for Alphabets.

Classifier	Original Dataset	Augmented Dataset
CNN [34]	23.07	99.55
CNN-ISC	26.15	99.85
CNN [35]	24.55	99.61
CNN [36]	32.3	98.93
CNN [38]	18.4	98.6
CNN-SVMs [5]	53	97.16
CNN-RF [6]	44	95.9

Among the evaluated CNN models on the digit datasets, our CNN-ISC model stands out, with the highest accuracy, of 99.7%, making it a top-performing model for digit recognition. The classification plots for both the original and augmented datasets of CNN-ISC are visually represented in Figures 12 and 13, respectively. The accuracy curves for the CNN [35], are shown in Figures 14 and 15. CNN-ISC accomplished the highest accuracy, of 99.85%, on the augmented dataset for the alphabets recognition task. The accuracy curves

for both the original and augmented datasets of CNN-ISC are depicted in Figures 16 and 17, respectively. Figures 18 and 19 display the accuracy curves for the CNN [36] on the original and augmented datasets. We also present the performance of the CNN [34] on the original and augmented datasets in Figures 20 and 21, respectively. Additionally, Figures 22 and 23 showcase the precision, recall, and F1 score performance metrics for both digits and alphabets. The performance metric values of CNN-ISC for both digits and alphabets demonstrate the effectiveness of data augmentation in the recognition task.

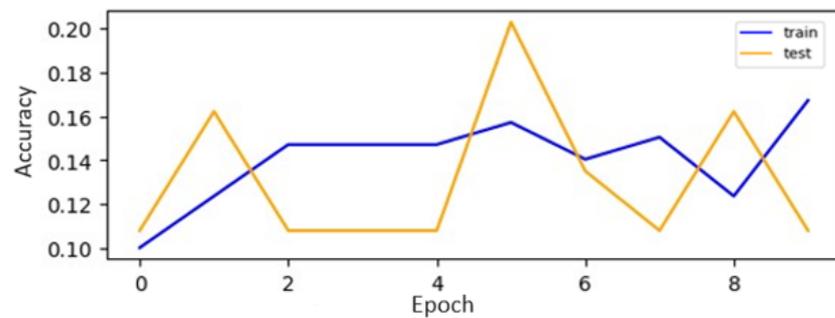


Figure 12. CNN-ISC on the Original Digit Dataset.

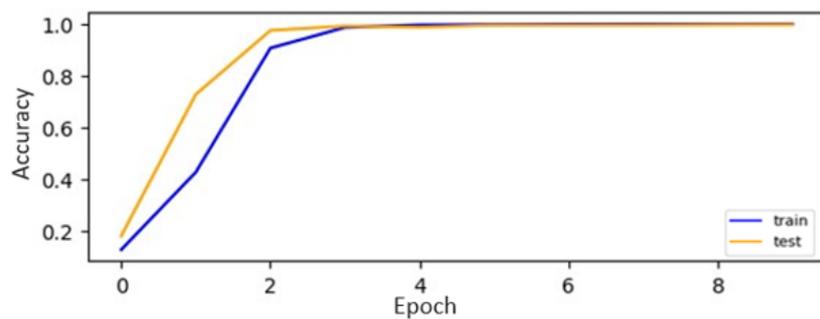


Figure 13. CNN-ISC on the Augmented Digit Dataset.

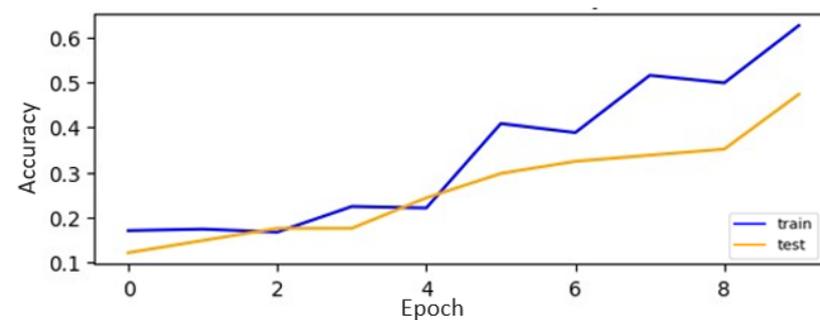


Figure 14. CNN [35] on the Original Digit Dataset.

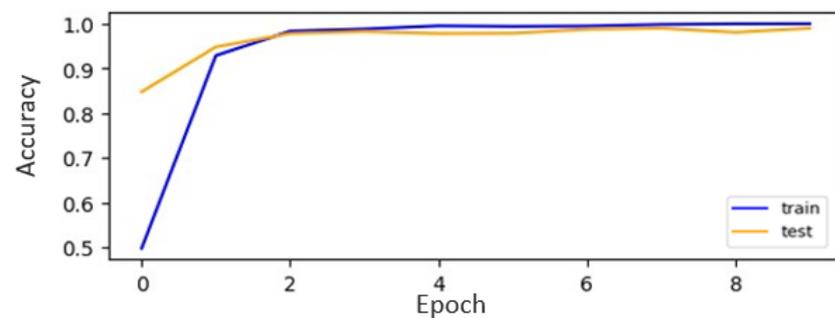


Figure 15. CNN [35] on the Augmented Digit Dataset.

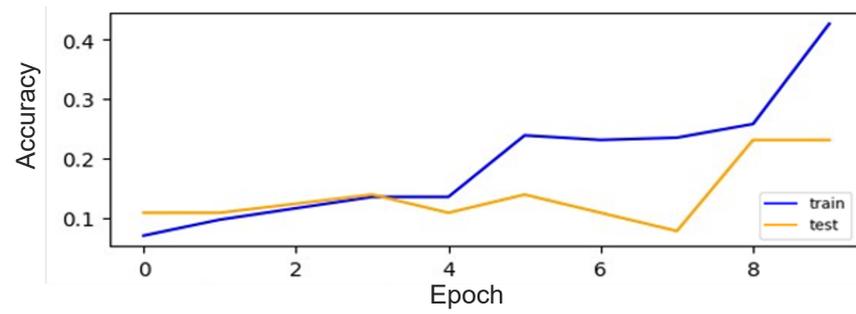


Figure 16. CNN-ISC on the Original Alphabet Dataset.

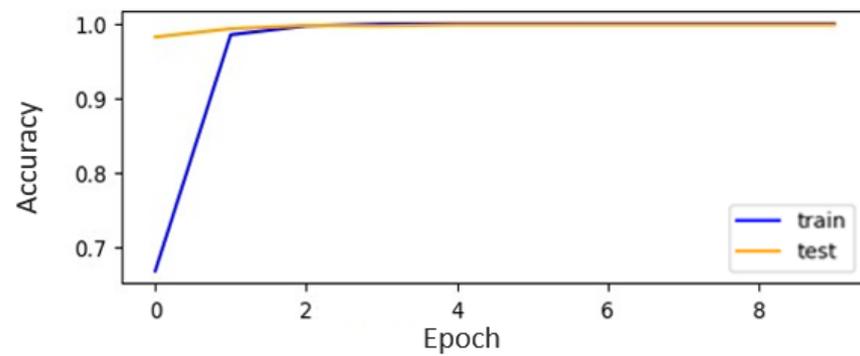


Figure 17. CNN-ISC on the Augmented Alphabet Dataset.

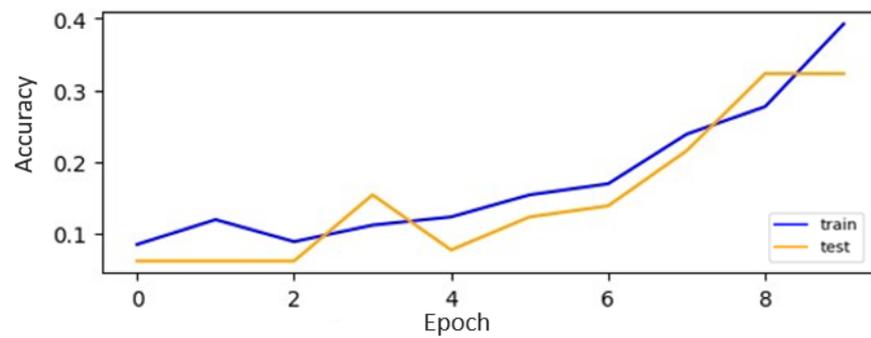


Figure 18. CNN [36] on the Original Alphabet Dataset.

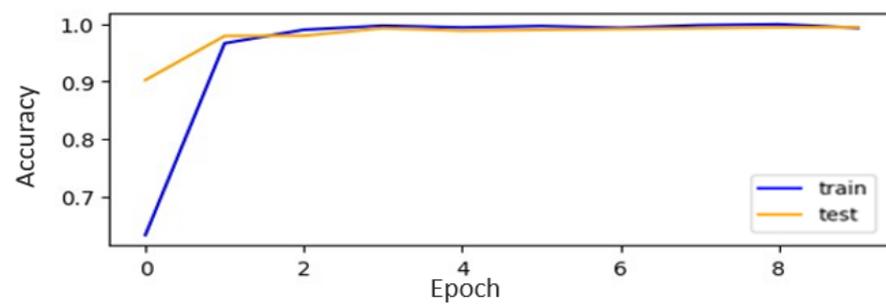


Figure 19. CNN [36] on the Augmented Alphabet Dataset.

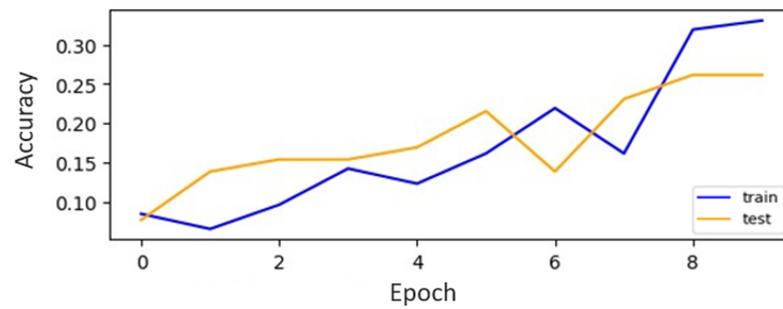


Figure 20. CNN [34] on the Original Alphabet Dataset.

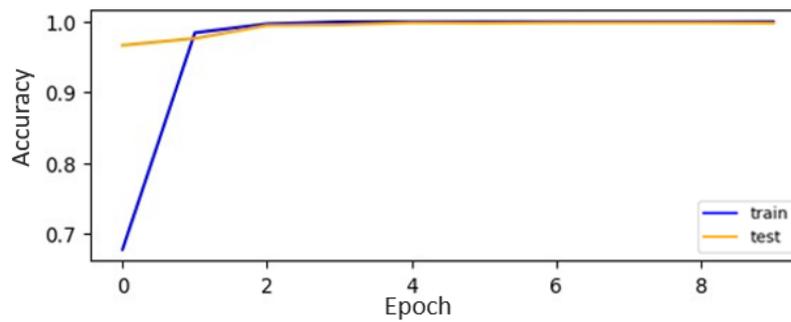


Figure 21. CNN [34] on the Augmented Alphabet Dataset.

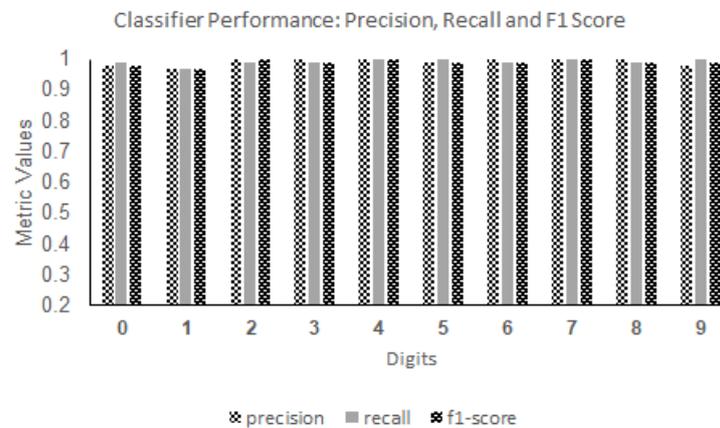


Figure 22. Metric Chart for Digits on CNN-ISC.

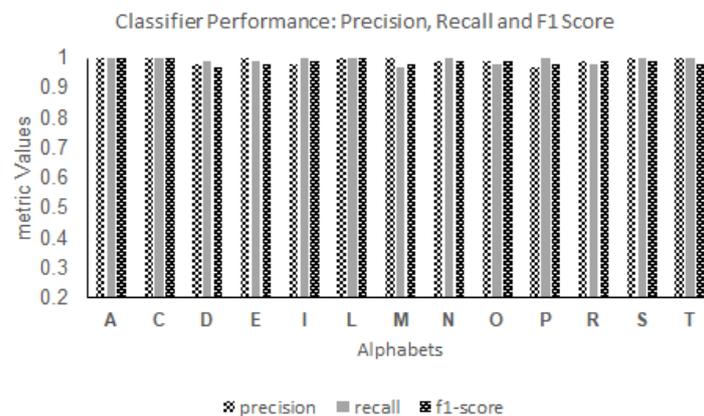


Figure 23. Metric Chart for Alphabets on CNN-ISC.

Alongside the CNN models for the recognition tasks, we also considered hybrid classifiers, namely, CNN-SVMs and CNN-RF. However, the accuracy results presented in Tables 6 and 7 indicate that these hybrid classifiers did not outperform any of the CNN models. Our study underscores the critical role of choosing the right recognition model for character recognition, particularly concerning our dataset. The effectiveness of character recognition models depends on how well they are suited to a specific dataset. We identified CNNs as highly effective for this task, given their exceptional ability to capture intricate character details and use them for precise identification. Notably, different CNN architectures yielded varying results, often due to nuanced differences in their design and hyperparameter configurations. Our finding strongly suggests that the CNN-ISC model is better suited to this specific recognition task, yielding superior performance. This bears significance not only for enhancing character recognition within our dataset but also for adapting our novel model to similar datasets within the same domain. This adaptability is particularly promising, as it suggests broader applicability and potential advancements in character recognition for related applications.

5. Conclusions and Future Work

Overall, CNN-based models, especially CNN-ISC, outperform classic and hybrid classifiers in digit and alphabet character recognition tasks. The remarkable accuracy of CNN-ISC on the dataset underscores its effectiveness in recognizing imprinted characters. The implementation of data augmentation greatly contributes to improved accuracy and F1 scores for both the digit and alphabet datasets, highlighting the importance of this technique in enhancing recognition performance. As part of our future work, we propose expanding the datasets, to include a wider range of ship-component images. This expansion would provide a more diverse and comprehensive representation of imprinted characters, further enhancing classification performance. The efficiency and accuracy enhancements offered by our model have wide-ranging implications for automation and precision across diverse industries, from manufacturing to logistics, agricultural monitoring, and infrastructure maintenance. Additionally, our study emphasizes the importance of considering dataset characteristics when selecting an appropriate model. Different types of datasets may yield varying performance outcomes, potentially revealing a different model as the superior choice over the current best model. As such, further investigations with diverse datasets will shed light on the generalizability and adaptability of different classifiers in various recognition applications.

Author Contributions: A.A., J.T.S. and I.Y.J. conceived and designed the experiments; A.A. and J.T.S. performed the experiments; A.A., J.T.S. and I.Y.J. analyzed the data; A.A. and J.T.S. wrote the paper. I.Y.J. re-organized and corrected the paper. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Research Foundation of Korea (NRF) grant, funded by the Korean government (No. 2021R1F1A1064345), and by the BK21 FOUR project, funded by the Ministry of Education, Korea (No. 4199990113966).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: No new data were created.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Chollet, F. *Deep Learning with Python*; Simon and Schuster: New York, NY, USA, 2021.
2. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial nets. In Proceedings of the Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, Montreal, QC, Canada, 8–13 December 2014.

3. You, A.; Kim, J.K.; Ryu, I.H.; Yoo, T.K. Application of generative adversarial networks (GAN) for ophthalmology image domains: A survey. *Eye Vis.* **2022**, *9*, 6. [[CrossRef](#)]
4. Xu, Y.; Wang, X.; Wang, K.; Shi, J.; Sun, W. Underwater sonar image classification using generative adversarial network and convolutional neural network. *IET Image Process.* **2020**, *14*, 2819–2825. [[CrossRef](#)]
5. Ahlawat, S.; Choudhary, A. Hybrid CNN-SVM Classifier for Handwritten Digit Recognition. *Procedia Comput. Sci.* **2020**, *167*, 2554–2560. [[CrossRef](#)]
6. Ramesh, G.; Tejas, M.; Thakur, R.; Champa, H. Handwritten Kannada Digit Recognition System Using CNN with Random Forest. In Proceedings of the International Conference on Information Processing (ICInPro), Bengaluru, India, 22–24 October 2021; pp. 92–104.
7. Bendib, I.; Gattal, A.; Marouane, G. Handwritten Digit Recognition Using Deep CNN. In Proceedings of the International Conference on Intelligent Systems and Pattern Recognition, Virtual, 16–18 October 2020; pp. 67–70. [[CrossRef](#)]
8. Prabhu, V.U. Kannada-MNIST: A new handwritten digits dataset for the Kannada language. *arXiv* **2019**, arXiv:1908.01242.
9. Rahaman, M.A.; Mahin, M.; Ali, M.H.; Hasanuzzaman, M. BHCDR: Real-Time Bangla Handwritten Characters and Digits Recognition using Adopted Convolutional Neural Network. In Proceedings of the International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT), Dhaka, Bangladesh, 3–5 May 2019; pp. 1–6.
10. Abdulrazzaq, M.B.; Saeed, J.N. A comparison of three classification algorithms for handwritten digit recognition. In Proceedings of the International Conference on Advanced Science and Engineering (ICOASE), Duhok, Iraq, 2–4 April 2019; pp. 58–63.
11. Darapaneni, N.; Krishnamurthy, B.; Paduri, A.R. Convolution Neural Networks: A Comparative Study for Image Classification. In Proceedings of the International Conference on Industrial and Information Systems (ICIIS), Rupnagar, India, 26–28 November 2020; pp. 327–332.
12. Nguyen, V.; Cai, J.; Chu, J. Hybrid CNN-GRU model for high efficient handwritten digit recognition. In Proceedings of the International Conference on Artificial Intelligence and Pattern Recognition, Beijing, China, 10–16 August 2019; pp. 66–71.
13. Shima, Y.; Nakashima, Y.; Yasuda, M. Handwritten digits recognition by using CNN alex-net pre-trained for large-scale object image dataset. In Proceedings of the International Conference on Multimedia Systems and Signal Processing, Shenzhen, China, 28–30 April 2018; pp. 36–40.
14. Ali, S.; Li, J.; Pei, Y.; Aslam, M.S.; Shaukat, Z.; Azeem, M. An Effective and Improved CNN-ELM Classifier for Handwritten Digits Recognition and Classification. *Symmetry* **2020**, *12*, 1742. [[CrossRef](#)]
15. Zhao, H.H.; Liu, H. Multiple classifiers fusion and CNN feature extraction for handwritten digits recognition. *Granul. Comput.* **2020**, *5*, 411–418. [[CrossRef](#)]
16. Ghadekar, P.; Ingole, S.; Sonone, D. Handwritten digit and letter recognition using hybrid dwt-dct with knn and svm classifier. In Proceedings of the International Conference on Computing Communication Control and Automation (ICCUBEA), Pune, India, 16–18 August 2018; pp. 1–6.
17. Khan, S.; Ali, H.; Ullah, Z.; Minallah, N.; Maqsood, S.; Hafeez, A. KNN and ANN-based recognition of handwritten Pashto letters using zoning features. *Int. J. Adv. Comput. Sci. Appl.* **2018**, *9*, 570–577. [[CrossRef](#)]
18. Pan, M.; Liu, Y.; Cao, J.; Li, Y.; Li, C.; Chen, C.H. Visual Recognition Based on Deep Learning for Navigation Mark Classification. *IEEE Access* **2020**, *8*, 32767–32775. [[CrossRef](#)]
19. Suo, Z.; Zhao, Y.; Chen, S.; Hu, Y. BoxPaste: An Effective Data Augmentation Method for SAR Ship Detection. *Remote Sens.* **2022**, *14*, 5761. [[CrossRef](#)]
20. Wawrzyniak, N.; Hyla, T.; Bodus-Olkowska, I. Vessel identification based on automatic hull inscriptions recognition. *PLoS ONE* **2022**, *17*, e0270575. [[CrossRef](#)]
21. Chen, X.; Wang, S.; Shi, C.; Wu, H.; Zhao, J.; Fu, J. Robust ship tracking via multi-view learning and sparse representation. *J. Navig.* **2019**, *72*, 176–192. [[CrossRef](#)]
22. Chen, X.; Ling, J.; Wang, S.; Yang, Y.; Luo, L.; Yan, Y. Ship detection from coastal surveillance videos via an ensemble Canny-Gaussian-morphology framework. *J. Navig.* **2021**, *74*, 1252–1266. [[CrossRef](#)]
23. Liu, R.W.; Yuan, W.; Chen, X.; Lu, Y. An enhanced CNN-enabled learning method for promoting ship detection in maritime surveillance system. *Ocean Eng.* **2021**, *235*, 109435. [[CrossRef](#)]
24. Huang, S.; Xu, H.; Xia, X.; Zhang, Y. End-to-end vessel plate number detection and recognition using deep convolutional neural networks and LSTMs. In Proceedings of the International Symposium on Computational Intelligence and Design (ISCID), Hangzhou, China, 8–9 December 2018; Volume 1, pp. 195–199.
25. Long, S.; He, X.; Yao, C. Scene text detection and recognition: The deep learning era. *Int. J. Comput. Vis.* **2021**, *129*, 161–184. [[CrossRef](#)]
26. Ma, J.; Shao, W.; Ye, H.; Wang, L.; Wang, H.; Zheng, Y.; Xue, X. Arbitrary-oriented scene text detection via rotation proposals. *IEEE Trans. Multimed.* **2018**, *20*, 3111–3122. [[CrossRef](#)]
27. Liu, Y.; Zhang, S.; Jin, L.; Xie, L.; Wu, Y.; Wang, Z. Omnidirectional scene text detection with sequential-free box discretization. *arXiv* **2019**, arXiv:1906.02371.
28. Zhang, M.; Yan, Y.; Wang, H.; Zhao, W. An Algorithm for Natural Images Text Recognition Using Four Direction Features. *Electronics* **2019**, *8*, 971. [[CrossRef](#)]

29. Wang, H.; Zhu, Y.; Adam, H.; Yuille, A.; Chen, L.C. Max-deeplab: End-to-end panoptic segmentation with mask transformers. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 5463–5474.
30. Zhang, W.; Sun, H.; Zhou, J.; Liu, X.; Zhang, Z.; Min, G. Fully convolutional network based ship plate recognition. In Proceedings of the International Conference on Systems, Man, and Cybernetics (SMC), Miyazaki, Japan, 7–10 October 2018; pp. 1803–1808.
31. Abdulraheem, A.; Jung, I.Y. A Comparative Study of Engraved-Digit Data Augmentation by Generative Adversarial Networks. *Sustainability* **2022**, *14*, 12479. [[CrossRef](#)]
32. Kassani, S.H.; Kassani, P.H.; Wesolowski, M.J.; Schneider, K.A.; Deters, R. Classification of Histopathological Biopsy Images Using Ensemble of Deep Learning Networks. *arXiv* **2019**, arXiv:1909.11870.
33. Abdulraheem, A.; Suleiman, J.T.; Jung, I.Y. Generative Adversarial Network Models for Augmenting Digit and Character Datasets Embedded in Standard Markings on Ship Bodies. *Electronics* **2023**, *12*, 3668. [[CrossRef](#)]
34. Ramprasath, M.; Anand, M.V.; Hariharan, S. Image classification using convolutional neural networks. *Int. J. Pure Appl. Math.* **2018**, *119*, 1307–1319.
35. Ge, D.Y.; Yao, X.F.; Xiang, W.J.; Wen, X.J.; Liu, E.C. Design of high accuracy detector for MNIST handwritten digit recognition based on convolutional neural network. In Proceedings of the International Conference on Intelligent Computation Technology and Automation (ICICTA), Xiangtan, China, 26–27 October 2019; pp. 658–662.
36. Ali, S.; Shaukat, Z.; Azeem, M.; Sakhawat, Z.; Mahmood, T.; ur Rehman, K. An efficient and improved scheme for handwritten digit recognition based on convolutional neural network. *SN Appl. Sci.* **2019**, *1*, 1125. [[CrossRef](#)]
37. Siddique, F.; Sakib, S.; Siddique, M.A.B. Recognition of handwritten digit using convolutional neural network in python with tensorflow and comparison of performance for various hidden layers. In Proceedings of the International Conference on Advances in Electrical Engineering (ICAEE), Dhaka, Bangladesh, 26–28 September 2019; pp. 541–546.
38. Jana, R.; Bhattacharyya, S. Character Recognition from Handwritten Image Using Convolutional Neural Networks. In *Advances in Intelligent Systems and Computing*; Springer: Singapore, 2019; p. 922. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.