University of Guelph

Assignment

Course No: ENGG*6100

Course Title: Machine Vision

Mini-Project No: 02

Date of Handout: 28.01.2025

Submission Deadline: 10.02.2025

Submitted by	Submitted to
Pramit Dutta *ID: 1319376* *MASC (Engg + AI)* *University of Guelph*	**Medhat Moussa** *Professor,* *School of Engineering,* *University of Guelph.*

Project Title: Enhancing Industrial Digit Recognition with lightweight MobileNetV2: Geometric Transformations and GAN-Based Upsampling.

1. Introduction

In industrial settings, serial number recognition is crucial for tracking and quality control. This project focuses on developing an efficient algorithm to detect and recognize serial numbers from industrial part images. Given the inconsistent digit sizes, style and class imbalance, robust preprocessing and recognition techniques are essential.

In consideration of these challenges, this project utilizes a deep learning-based approach using **MobileNetV2**, a lightweight neural network architecture. To further enhance recognition accuracy, **geometric transformations** are applied for **minority class upsampling**, increasing dataset diversity. Additionally, a **Generative Adversarial Network (GAN)-based approach** is explored to generate synthetic samples, improving dataset balance and overall robustness.

1.1 Problem Definition

Industrial Digit Recognition is a crucial approach in industrial automation, and it provides efficient tracking, quality control, and inventory management. The challenge lies in accurately extracting and recognizing serial numbers from industrial part images, where variations in digit appearance, image quality, and dataset distribution create significant obstacles.

The provided dataset consists of 1885 images across 12 different part types, captured under consistent camera, lighting, and object distance. However, despite these controlled conditions, several challenges affect part recognition. The challenges are as follow:

1.1.1 Class Imbalance in Dataset Representation

The dataset contains an uneven distribution of serial numbers, with some classes appearing significantly more than others, as shown in the Total Sample per Category plot (figure 1). The plot highlights disparities where certain serial numbers (for example- 20871905, 70085119-0300) have far less samples than others. This imbalance can lead to biased models that favor dominant classes while misclassifying rare ones.

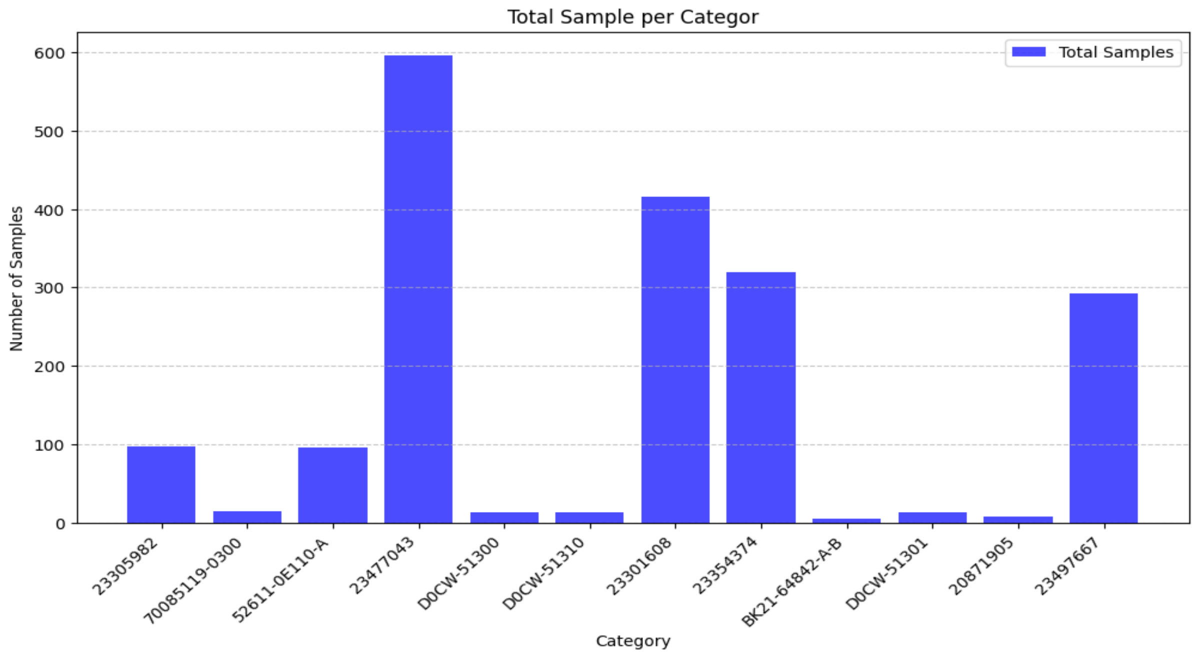

Figure 1: Class Imbalance in Dataset: Uneven Sample Distribution Across Categories

1.1.2 Varying Digit Size

As showed in figure 2, serial numbers are not uniform across all parts, leading to variations in font style, thickness, and character spacing. This inconsistency makes it difficult to apply a single image processing technique across all parts.

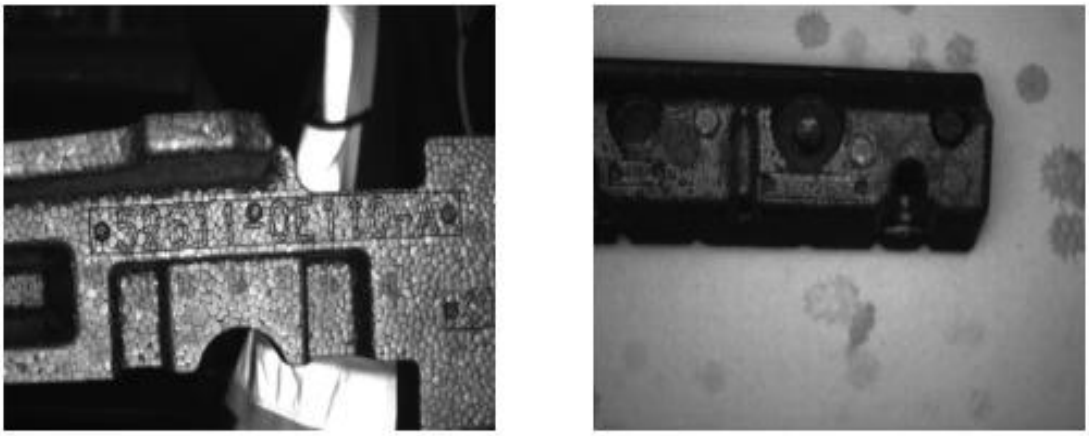

Figure 2: Variability in Digit Appearance Due to Differences in Size, Font, and Engraving Style

1.1.3 Mislabeling and Ground Truth Errors

Incorrect labeling is a significant challenge in serial number recognition. In figure 3 (File: 70085119-0300_3149), the image is wrongly classified as '70085119', despite displaying 'BK21-64842-A-B'. Such errors introduce noise in the dataset, leading to biased training, inaccurate predictions, and inconsistencies in model performance evaluation.

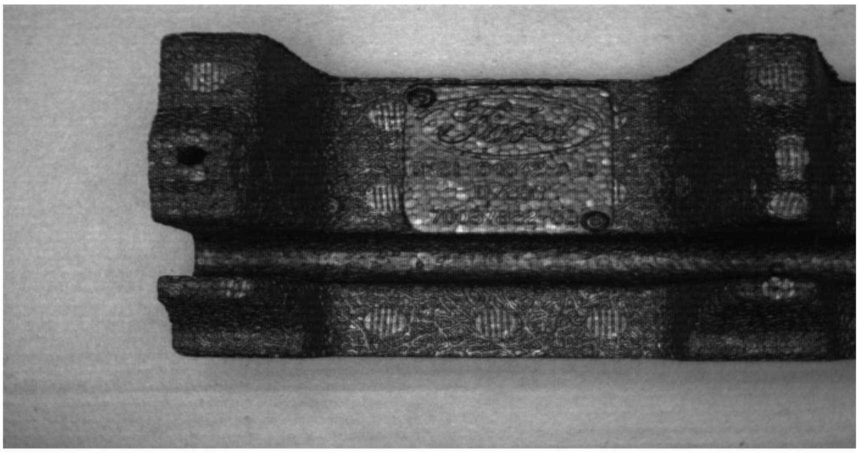

Figure 3: Example of Incorrect Labeling – The part is misclassified as '70085119.'

1.1.4 Blurriness and Distortions

Some of the images have blurry serial number affecting the clarity of digits (figure 4). These distortions reduce the effectiveness of the proposed algorithm and increase the likelihood of misclassification.

Moreover, Certain digits, such as 0 vs. O, 1 vs. I, and 8 vs. B, have similar shapes, increasing the risk of confusion in recognition models. Differentiating between these characters is essential for high accuracy.

Figure 4: Image Quality Challenge – Blurred and Angled View Affecting Serial Number Visibility

1.2 Objectives

The primary objectives of this project are to:

1. Develop an algorithm for accurate part recognition using lightweight MobileNetV2.
2. Address class imbalance in the dataset by implementing minority class upsampling via geometric transformation techniques.
3. Explore the use of Generative Adversarial Networks (GANs) to generate synthetic samples and enhance dataset diversity for improved recognition performance.

1.3 Rationale for Chosen Methods

Industrial Digit Recognition can be performed in many ways that includes digit-by-digit recognition or end-to-end recognition. However, the variability in digit size and inconsistent positioning makes it harder to extract the test using traditional image processing techniques. Considering all this circumstances, in this project, a deep learning-based approach was implemented where the other features of the parts were used to identify the part and serial number.

Automatic Feature Extraction: Traditional image processing techniques, such as edge detection, thresholding, and contour analysis, struggle with the inherent challenges of industrial digit recognition such as varying font styles, surface textures that sometimes resemble digits, occlusions, and distortions. Automatic feature extraction via deep learning provides greater flexibility and robustness by learning hierarchical features directly from data. Unlike conventional methods that rely on handcrafted features, this approach can adapt to diverse font variations, engraving depths, and noise, making it more effective in real-world scenarios. The ability to automatically extract discriminative features without manual intervention ensures better accuracy, especially in complex industrial settings.

MobileNetV2 for Serial Number Recognition: MobileNetV2 was selected due to its lightweight architecture and high efficiency in feature extraction, making it well-suited for real-time industrial applications. Compared to heavier convolutional neural networks (CNNs), MobileNetV2 maintains a balance between computational efficiency and accuracy, which is essential for handling large-scale datasets while ensuring fast inference times.

Minority Class Upsampling for Dataset Balance: This dataset exhibits severe class imbalance, where certain serial numbers are overrepresented while others have significantly fewer samples. To address this, minority class upsampling techniques, such as several geometric transformations are employed to artificially increase the representation of underrepresented classes. This helps reduce model bias and ensures better generalization across all serial numbers.

GAN as An Alternative of Geometric Transformation: Generative Adversarial Networks (GANs) have the potential to increase dataset diversity by generating synthetic images with unique variations. A notable phenomenon in GAN-generated samples is the presence of noisy or gibberish-like artifacts, which still contain latent patterns that can improve model robustness. Training on such diverse data forces the model to learn generalized feature representations, making it better at handling real-world inconsistencies. Additionally, testing on test data augmented with different transformations from those used in training allows a better evaluation of the model's generalization to unseen distortions.

2. Literature Review

Convolutional Neural Networks (CNN) with its superior performance through strong feature extraction and generalization capabilities, have become one of the top choices for digit recognition. D. Shii [1] examined whether a CNN trained on single-digit images could recognize overlapping digits without additional training. Their findings indicated that the network could partially distinguish overlapping digits, achieving a recognition rate of 55% for one of two digits, which improved to 46% for both digits when considering the top three predictions. This implies the potential of CNNs in handling occluded character recognition and other challenging vision tasks.

An imbalanced dataset can significantly impact deep learning models by causing bias toward majority classes. P. Dutta et al. [2] highlighted this issue in retinal disease detection, where underrepresented classes were harder to classify, leading to lower recognition accuracy and generalization errors. This imbalance can result in misclassification and reduced model reliability, particularly in real-world applications where accurate detection of all classes is crucial.

Upsampling plays a role tackling class imbalance in which ensure fair representation of all categories. J. R. Barr et al. [3] emphasized its importance in imbalanced datasets, where models tend to favor majority classes, leading to poor generalization. By increasing the presence of minority samples, upsampling helps balance class distributions, enhancing model performance and reliability. The study highlighted various upsampling techniques, such as bootstrapping and SMOTE, demonstrating their effectiveness in improving classification accuracy by increasing the presence of minority samples.

Upsampling is essential for improving classification accuracy in datasets with limited real samples. A. R. Oh [4] demonstrated the effectiveness of GAN-based augmentation, particularly with CycleGAN and StyleGAN2-ada, to generate synthetic vessel images. Their study showed that combining real and synthetic data significantly enhanced model performance, with the best classification accuracy reaching 87.01%.

GANs have been used to generate realistic synthetic noise that improves deep-learning model performance. L. D. Tran et al. [5] proposed a GAN-based noise model that learns real noise distributions, enhancing the effectiveness of denoising networks. Their approach outperformed traditional methods by improving robustness on real-world noisy images.

In a separate Study, N. M. Saad et al. [6] proposed an automated LCD digit recognition system using an IP webcam and neural network classification. The system processes grayscale images, applies adaptive thresholding, and segments digits for recognition. It achieved 90% accuracy on 50 test images, demonstrating efficiency in industrial data collection.

In another study, N. Qin et al. [7] introduced LeanNet, a pruned VGG-16 model designed for recognizing digits on industrial products. By reducing redundant filters, LeanNet achieved a 25× smaller model size and 4.5× fewer FLOPs while maintaining accuracy. It outperformed MobileNet and SqueezeNet, making it ideal for resource-constrained applications.

3. Methodology and Experimentation

This section outlines the methodology used for serial number recognition in industrial part images. The proposed approach involves data preprocessing, feature extraction, and model training, with a focus on addressing dataset imbalance and image variability. The experimental setup and evaluation metrics are also outlined that will be used to assess the effectiveness of the chosen methods.

3.1 Algorithm for Industrial Digit Recognition using The Part's Features

This algorithm describes the steps for recognizing industrial digits using a MobileNetV2-based deep learning model with data augmentation strategies.

Step 1: Load the Dataset: Load the dataset and resize them to (128,128) to match the model's input. Split the dataset randomly into 80% training and 20% testing for evaluation.

Step 2: Upsampling Strategies: To improve model generalization and address class imbalance, two upsampling approaches are explored.

Approach 1: Geometric Transformation: Apply standard image transformations on the training dataset, including rescaling, rotation, width and height shifts, shearing, zooming, horizontal flipping, and filling new pixels using the nearest neighbor method.

Approach 2: GAN-based Augmentation: Train a Generative Adversarial Network (GAN) to generate synthetic images for minority classes. However, due to RAM limitations, only two of the six minority classes were upsampled using GAN-generated synthetic images to enhance their representation and so this approach was used for classification among eight of the twelve classes.
For Test Data in Both Approaches: Use only geometric transformations such as rescaling, rotation, flipping, and zooming to simulate real-world variations to increase dataset diversity without altering the fundamental image characteristics.

Step 3: Load and Modify MobileNetV2 Model as a Classifier: Load MobileNetV2 with pre-trained ImageNet weights, excluding the top classification layer, and set the input shape to (128,128). Apply global average pooling to reduce dimensionality, add a fully connected layer with 1024 neurons using ReLU activation, and include a final classification layer with softmax activation, where the number of neurons equals the number of classes.

Step 4: Compile and Train the Model: Compile the model with Adam optimizer, categorical cross-entropy loss, and track accuracy as the evaluation metric. Train the model using augmented real images (Approach 1) or a combination of real and GAN-generated images (Approach 2).

Step 5: Evaluate the Model: Predict the class labels for test images (upsampled via geometric transformation) and evaluate the model with precision, recall, and F1-score for each class.

3.1.1 Geometric Transformation based Upsampling Strategies

The data distribution, as shown in Figure 1, highlights the need for minority class upsampling. In this project, classes with fewer than 50 samples are considered minority classes, and among the 12 classes, six fall into this category, while the remaining six are majority classes with significantly more samples.

Geometric transformation techniques were applied to upsample six minority classes, enhancing dataset diversity while preserving essential features. The image illustrates different transformations used, including width shift, height shift, shear transformation, zooming, and rotation, which generate additional variations for these classes. Systematic application of these transformations enables the model to recognize objects from multiple perspectives, improving its ability to generalize and classify underrepresented classes accurately.

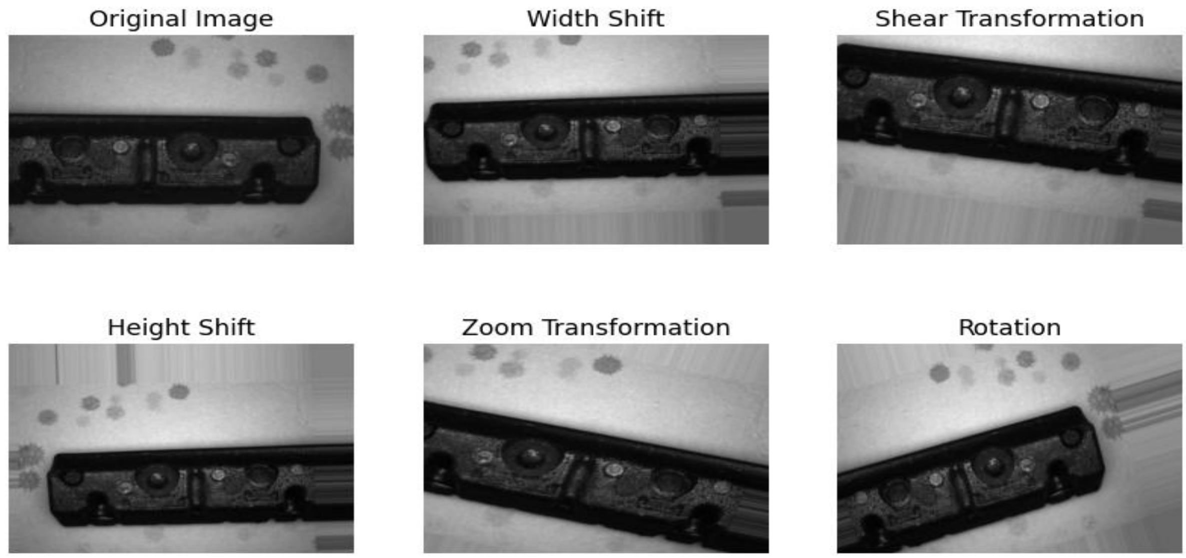

Figure 5: Geometric Transformations for Minority Class Upsampling: Width Shift, Height Shift, Shear, Zoom, and Rotation

The test set was also upsampled using geometric transformations to ensure the model captures minority class features robustly, even in the presence of variations. This augmentation helps improve generalization, allowing the model to recognize and classify underrepresented classes more effectively in real-world scenarios. Figure 6 shows the data distribution after minority class upsampling which shows more minority class samples. In this project 20 images were produced per original image.

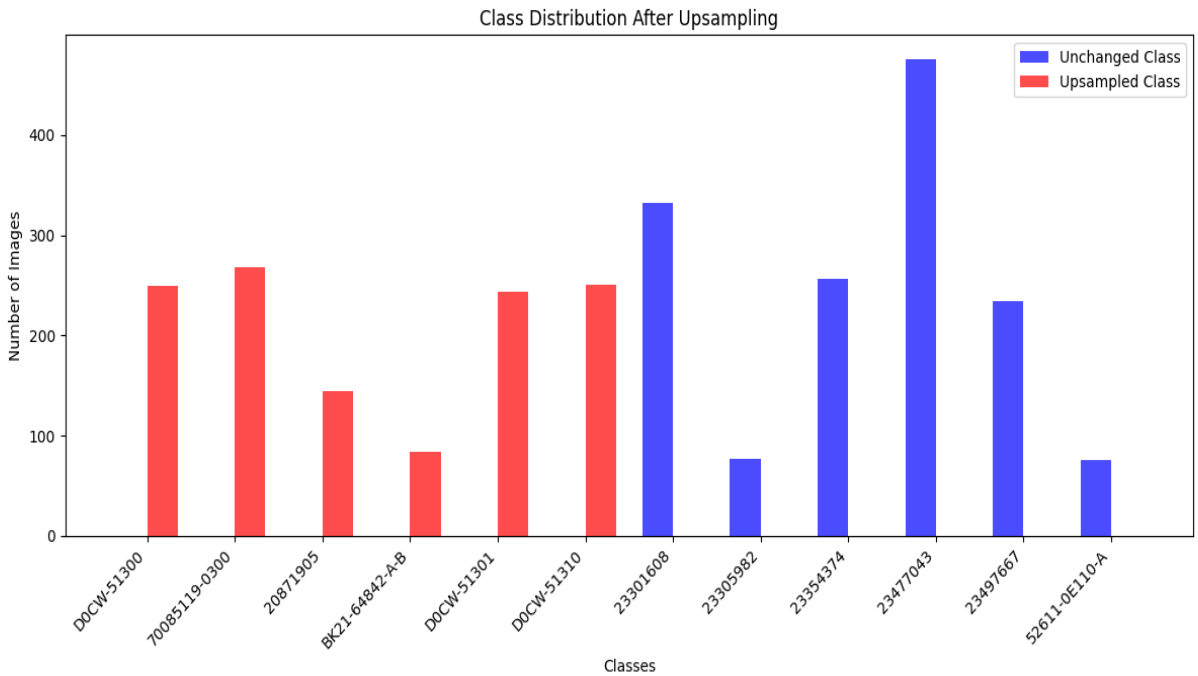

Figure 6: Class Distribution After Upsampling: Red bars represent upsampled minority classes, while blue bars indicate unchanged majority classes.

3.1.2 GAN based Upsampling Strategy

Generative Adversarial Networks (GANs) are utilized for upsampling minority classes by generating synthetic images that mimic real samples. This approach helps balance the dataset by increasing the representation of under-sampled classes, improving the model's ability to generalize across all categories.

3.1.2.1 GAN Architecture:

The architecture of Generative Adversarial Networks (GAN) consists of two key components: a Generator and a Discriminator, both engaged in an adversarial process to improve data synthesis.

The Generator takes in random noise as input and produces synthetic images that attempt to mimic real images. These generated images are then evaluated by the Discriminator, which is trained to distinguish between real images from the dataset and fake images created by the Generator. The Discriminator provides feedback by classifying images as real or fake, which helps the Generator refine its output to produce more realistic samples over successive iterations.

Figure 7 illustrates the GAN architecture used in this study, highlighting the interaction between the Generator and Discriminator. The Generator learns to create realistic images, while the Discriminator improves its classification ability, leading to progressively higher-quality synthetic samples that enhance the dataset diversity for minority classes.

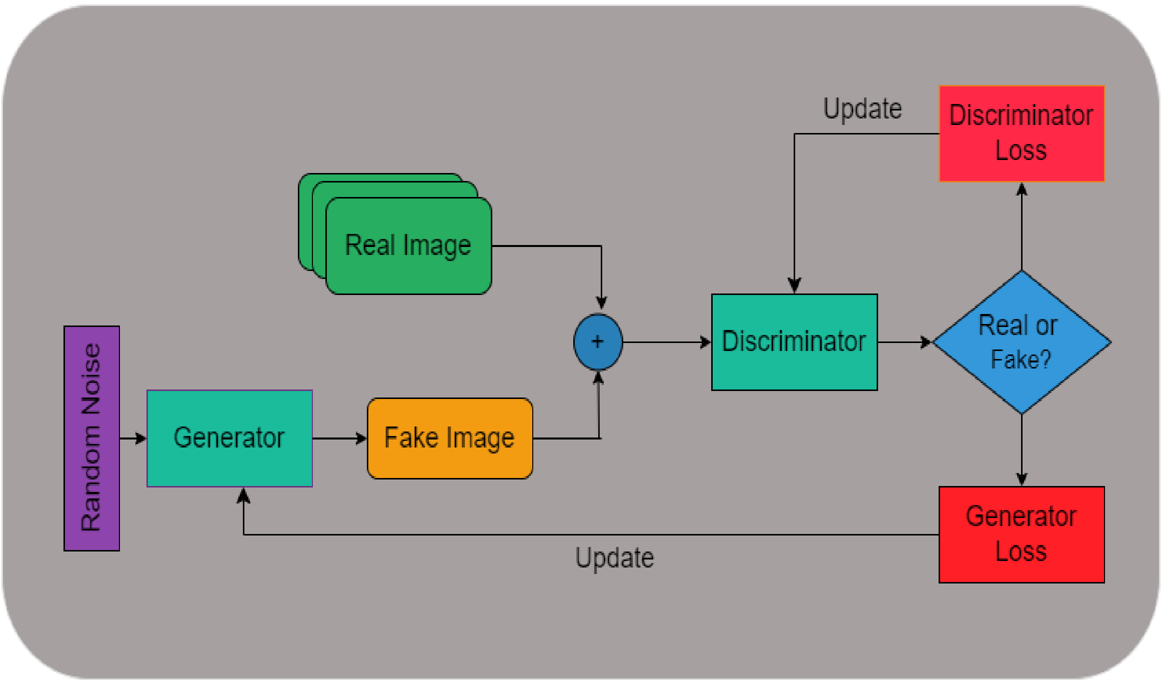

Figure 7: GAN Architecture for Minority Class Upsampling

3.1.3 Pretrained Modified MobileNetV2 Architecture

MobileNetV2 is a lightweight deep learning model optimized for efficient feature extraction while maintaining high accuracy, making it ideal for mobile and embedded applications [8]. Figure 8 illustrates the modified MobileNetV2 architecture, which processes 128×128 input images through an initial convolution layer before extracting hierarchical features across low, mid, and high-level stages.

In the early stages, fundamental spatial details are captured, with the spatial resolution progressively decreasing. As the network deepens, the extracted features become more abstract and semantically meaningful, refining the ability to distinguish complex patterns. The core structure relies on inverted residual blocks (IRBs), where features are expanded, transformed through computationally efficient operations, and then projected to a compact representation. Residual connections are applied in layers where input and output dimensions match, improving gradient flow and feature retention.

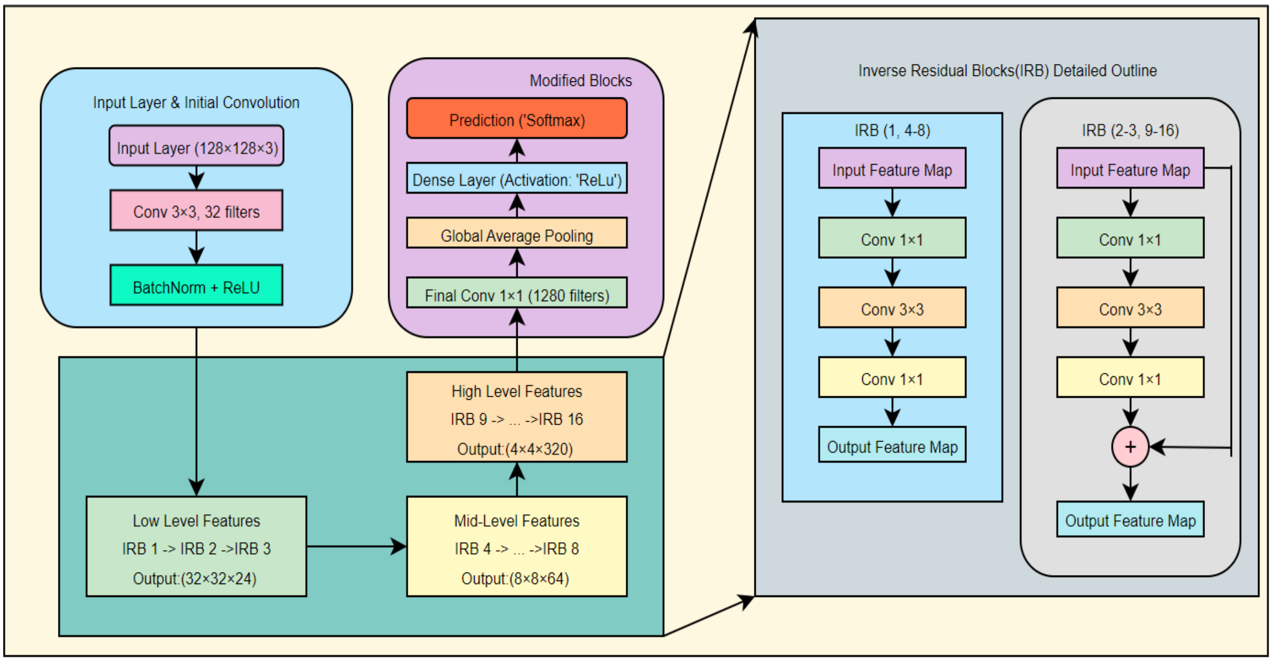

Figure 8: Modified MobileNetV2 Architecture: A Lightweight Feature Extractor Model

To tailor the model for our desired classification, a Global Average Pooling (GAP) layer condenses spatial information, followed by a fully connected dense layer with 1024 neurons (ReLU activation) to enhance learning capacity. A SoftMax classification layer maps features to target classes, enabling multi-class recognition. The model is optimized using Adam with categorical cross-entropy loss, ensuring effective classification for industrial parts.

3.2 Experimentation

The project was implemented on the Google Colab platform, utilizing a T4 GPU for accelerated training. The model was developed using TensorFlow for deep learning, while OpenCV was used for image processing. The model was compiled using the Adam optimizer, with categorical cross-entropy loss to manage multi-class predictions and accuracy as the primary evaluation metric, ensuring effective classification performance.

3.3 Evaluation Metrices

Apart from accuracy, precision recall and F1 score was used to evaluate the model's performance as in an imbalance dataset, accuracy sometime may not reflect the accurate performance. The accuracy, precision, recall and F1 score can be defined as follow:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{3.1}$$

$$Precision = \frac{TP}{TP+FP} \tag{3.2}$$

$$Recall = \frac{TP}{TP+FP} \tag{3.3}$$

$$F1\ Score = 2 \times (\frac{Precision \times Recall}{Precision + Recall}) \tag{3.4}$$

where TP, TN, FP, and FN represent true positive, true negative, false positive and false negative respectively. Accuracy measures how well the model correctly identifies a class as positive among all available samples. In contrast, precision evaluates how reliable the model is in classifying a sample as positive by considering the ratio of true positives to the total number of true positives and false positives. Additionally, recall justifies the model's ability to identify a class by calculating the proportion of true positives relative to the total number of true positives and false negatives. The F1 score, a harmonic mean of precision and recall, provides a balanced assessment of the model’s performance in terms of precision and recall.

4. Result and Observation

This section presents the quantitative and qualitative analysis of the results obtained using a MobileNetV2 model trained with two different minority class upsampling approaches. In all cases, the upsampled augmented test dataset is used to evaluate the model's performance. Assessing performance on the augmented dataset provides insights into how well the model generalizes to variable industrial conditions, where images may vary in appearance due to real-world factors such as lighting, scale, and perspective changes.

In this project, the label was assumed as the ground truth, which considered as the correct reference value assigned to each image. In this context, the label is also serves as the serial number that needs to be accurately extracted from the images for validation and analysis.

4.1 Quantitative Analysis

The following Section 4.1.1 presents the results of the Geometric Transformation-based Upsampling Technique, while Section 4.1.2 evaluates the GAN-based Upsampling approach and compares their effectiveness in improving model’s performance.

4.1.1 Geometric Transformation based Upsampling Technique

The model, trained with geometric transformation-based upsampling, achieved an impressive **97% accuracy** across **twelve classes**. With a **macro-averaged precision, recall, and F1-score of 0.97**, it performed consistently well, even for the less frequent classes. The table below provides class-wise results, in terms of Precision, Recall and F1 Score.

Table 1: Class-wise Performance Metrics of the Model

Class \ Metrices	Precision	Recall	F1 Score	Samples
20871905	1.00	1.00	1.00	21
23301608	0.98	0.98	0.98	84
23305982	1.00	1.00	1.00	20
23354374	1.00	1.00	1.00	64
23477043	0.98	0.98	0.98	120
23497667	1.00	1.00	1.00	58
52611-0E110-A	1.00	0.95	0.97	20
70085119-0300	0.95	1.00	0.98	42
BK21-64842-A-B	1.00	0.90	0.95	21
D0CW-51300	0.88	0.88	0.88	42
D0CW-51301	0.86	0.88	0.87	42
D0CW-51310	1.00	0.98	0.99	42

The table-1 shows classes 20871905, 23305982, 23354374, and 23497667 attained a perfect 1.00 score across all metrics. Other classes, such as D0CW-51300 and D0CW-51301, showed slightly lower scores around 0.88, which indicates minor misclassifications. Overall, the results indicate that the model generalizes well across different categories, with minimal performance drops in certain cases.

Figure 10 presents the confusion matrix illustrating the model's classification performance across multiple classes. The diagonal elements represent correctly predicted labels, while off-diagonal elements indicate misclassifications. The model achieves perfect classification for 23330582, 23497667, and 52611-0E110-A, while slight misclassification is observed in 23301608, BK21-64842-A-B, and DOCW-51310**,** where a few samples were incorrectly classified.

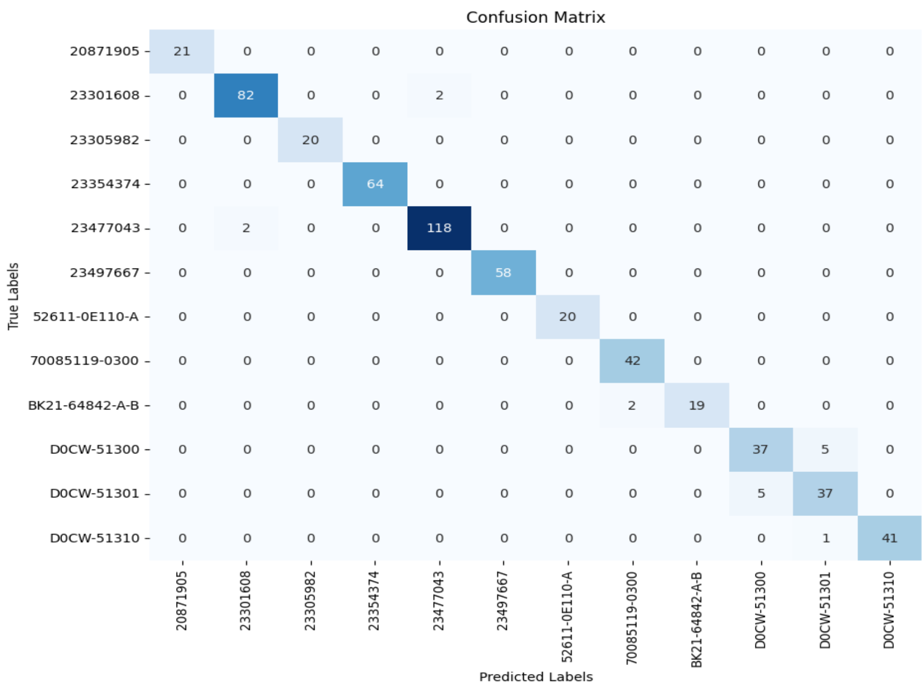

Figure 9: Confusion Matrix for MobileNetV2 Model with Minority Class Upsampling via Geometric transformation

4.1.2 GAN based Upsampling Technique

Due to RAM limitations, implementing the GAN-based upsampling technique for all minority classes was not feasible. To meet the project requirement of evaluating the model on at least eight classes, the geometric transformation-based model was trained on eight classes, while GAN-based upsampling was applied to two of the six minority classes. Table-2 presents the class-wise performance of both models, where training was conducted on upsampled data—one using geometric transformations and the other incorporating GAN-based upsampling. In both cases, the test set was augmented using geometric transformations.

The table-2 compares the classification performance of Geometric Transformation-based Upsampling and Generative Adversarial Network (GAN)-based Upsampling using precision, recall, and F1-score across eight classes. Both techniques improved model performance, with Geometric Transformation-based upsampling achieving high scores, including perfect 1.00 precision, recall, and F1-scores for most classes. However, minor drops were observed in 23477043 (0.97 recall) and 23301608 (0.97 F1-score), indicating slight misclassifications. GAN-based upsampling, while effective, showed slightly lower recall and F1-scores for some classes, such as 20871905 (0.86 recall, 0.92 F1-score) and 70085119-0300 (0.95 recall, 0.98 F1-score), suggesting that although it introduces diverse synthetic variations, it may also create subtle inconsistencies affecting recall. On average, GAN-based upsampling achieved 0.99 precision, 0.99 recall, and 0.99 F1-score, while Geometric Transformation-based upsampling had slightly lower scores of 0.98 precision, 0.97 recall, and 0.98 F1-score.

Table 2: Comparison of Class-wise Performance Metrics of Two Upsampling Techniques

Metrices / Class	Geometric Transformation based Upsampling			Generative Adversarial Network (GAN) based Upsampling		
	Precision	**Recall**	**F1 Score**	**Precision**	**Recall**	**F1 Score**
20871905	1.00	1.00	1.00	1.00	0.86	0.92
23301608	0.96	0.98	0.97	0.94	0.98	0.96
23305982	1.00	1.00	1.00	1.00	1.00	1.00
23354374	1.00	1.00	1.00	1.00	1.00	1.00
23477043	0.98	0.97	0.98	0.98	0.98	0.98
23497667	1.00	1.00	1.00	1.00	1.00	1.00
52611-0E110-A	1.00	1.00	1.00	0.95	1.00	0.98
70085119-0300	1.00	1.00	1.00	1.00	0.95	0.98
Average	0.99	0.99	0.99	0.98	0.97	0.98

Overall, both techniques performed well, but Geometric Transformation-based upsampling demonstrated slightly better recall and F1-scores in certain cases. While GAN-based upsampling enhances diversity, geometric transformations help maintain structural consistency, which is particularly important in industrial digit recognition, where preserving the original shape and orientation of digits ensures more reliable classification. Moreover, GAN-based upsampling generally requires a larger dataset for optimal results, as training on limited samples can introduce inconsistencies in synthetic image generation.

4.2 Qualitative Analysis

In deep learning, visualizing a model's decision-making process provides insights into how it learns and identifies important features. For this reason, instead of using the proposed algorithm as a black-box, Grad-CAM (Gradient-weighted Class Activation Mapping) was used to highlight the areas where the model places the most attention. The figure 11 shows an original image alongside feature maps extracted from models trained using geometric transformation-based upsampling and GAN-based upsampling.

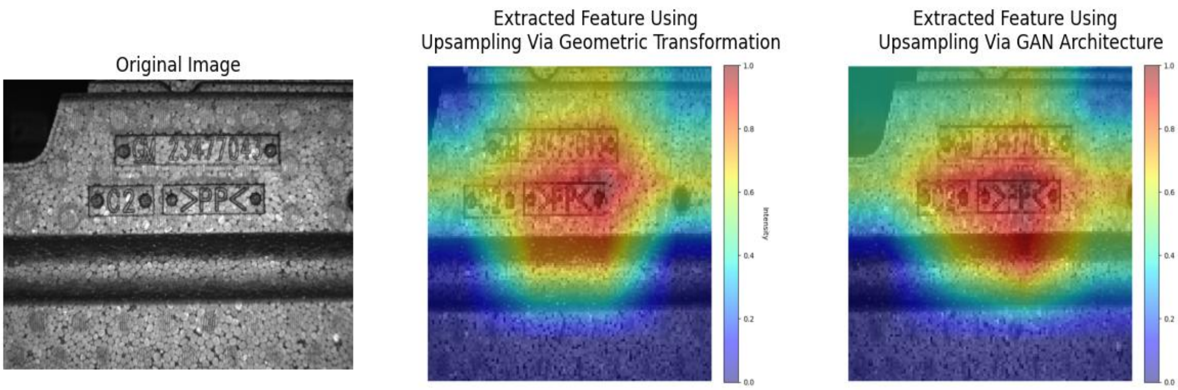

Figure 10: Grad-CAM visualizations showing the extracted features from models- showing focus on the serial number region, indicating it as an effective feature.

Even though these models learn automatically, both are consistently focusing on the region containing the serial number, indicating that they are giving more focus on the serial number of the object. While there are slight differences in how attention is distributed, the focus remains on the area of interest, suggesting that the upsampling approaches enable the model to learn meaningful representations.

4.3 Ablation Study

This section examines the impact of minority class upsampling on the proposed algorithm. In this study, the modified MobileNetV2 was trained and tested on the original dataset, and its performance was compared to that of the same model trained and tested on the upsampled dataset. The results of this comparison are presented in Table 3.

Table 3: Comparison of the modified MobileNetV2 trained on the original dataset versus the upsampled dataset.

Metrices / Class	Original Dataset			Upsampled Dataset		
	Precision	**Recall**	**F1 Score**	**Precision**	**Recall**	**F1 Score**
20871905	0.50	0.67	0.57	1.00	1.00	1.00
23301608	0.93	0.98	0.95	0.98	0.98	0.98
23305982	0.65	1.00	0.78	1.00	1.00	1.00
23354374	1.00	0.84	0.92	1.00	1.00	1.00
23477043	0.97	0.98	0.98	0.98	0.98	0.98
23497667	1.00	0.97	0.98	1.00	1.00	1.00
52611-0E110-A	0.95	0.90	0.92	1.00	0.95	0.97
70085119-0300	1.00	0.50	0.67	0.95	1.00	0.98
BK21-64842-A-B	0.43	0.62	0.51	1.00	0.90	0.95
D0CW-51300	0.33	0.50	0.40	0.88	0.88	0.88
D0CW-51301	0.60	0.50	0.54	0.86	0.88	0.87
D0CW-51310	0.50	0.60	0.54	1.00	0.98	0.99

The results in Table 3 demonstrate how minority class upsampling enhances model performance, particularly for underrepresented classes. For the minority classes- 20871905, 70085119-0300, BK21-64842-A-B, D0CW-51300, D0CW-51301, and D0CW-51310 the F1 score improved by 0.43, 0.32, 0.44, 0.48, 0.37, and 0.45 respectively, showing that upsampling effectively addresses class imbalance. Overall, both precision and recall increased across most classes, leading to more balanced predictions. All of the minority classes saw improvements in recall and F1 score, reinforcing the role of upsampling in enhancing minority class detection. These findings highlight the effectiveness of upsampling in reducing bias towards majority classes and improving model generalization.

4.4 Analysis of the Final Prediction Layer

In the industrial digit recognition model, the prediction layer which uses a softmax activation function generates a probability vector, assigning confidence scores to various possible serial numbers. The serial number with the highest probability is selected as the predicted label. This setup can be directly utilized in real-world implementation, such as automation and smart inventory management. Figure 12 is an example of a model (trained using geometric transformation based upsampled dataset) inference, where the model successfully identified the serial number 52611-0E110-A with 99.99% confidence.

Figure 11: Example of Model Inference: The system accurately predicts the serial number "52611-0E110-A " with a confidence of 99.99%

5. Conclusion

This study explored industrial digit recognition using MobileNetV2, incorporating Geometric Transformation-based Upsampling and GAN-based Upsampling to address class imbalance. The results indicate that both techniques effectively improved classification performance, with Geometric Transformation-based Upsampling achieving slightly higher recall and F1-scores in some cases, ensuring structural consistency in digit recognition. Meanwhile, GAN-based Upsampling introduced diversity, helping the model generalize better but occasionally leading to subtle inconsistencies in recall.

Overall, both techniques performed well, but Geometric Transformation-based Upsampling demonstrated slightly better recall and F1-scores in certain cases. While GAN-based Upsampling enhances diversity, geometric transformations help maintain structural consistency, which is particularly important in industrial digit recognition, where preserving the original shape and orientation of digits ensures more reliable classification. Moreover, GAN-based Upsampling generally requires a larger dataset for optimal results, as training on limited samples can introduce inconsistencies in synthetic image generation. In the qualitative analysis, Grad-CAM visualizations confirmed that both models primarily focused on the serial number regions, validating their feature-learning capability. This study also analyzed the effect of using upsampling and showed how upsampling technique can provide robust output even in a highly imbalanced dataset.

Further improvements can be made by incorporating Few-Shot GANs, which allow for higher-quality synthetic data generation with minimal training samples. Optimizing the training process with adaptive learning rate techniques could also enhance convergence, leading to better generalization across different industrial parts.

6. Tools and Resources

- Industrial Digit Recognition Dataset
- Google Colab
- TensorFlow (Including MobileNetV2 with Imagenet weights)
- Keras, OpenCV, NumPy, Pandas, Scikit-learn.
- Large Language Model (LLM) for writing refinement

References

[1] Daigo Shii, Ryosuke Miyoshi, and Kazuyuki Hara, "Performance of Pre-Learned Convolution Neural Networks Applied to Recognition of Overlapping Digits," *2020 IEEE International Conference on Big Data and Smart Computing (BigComp)*, pp. 113-116, DOI: 10.1109/BigComp48618.2020.00-90.

[2] P. Dutta, K. A. Sathi, M. A. Hossain, and M. A. A. Dewan, "Conv-ViT: A Convolution and Vision Transformer-Based Hybrid Feature Extraction Method for Retinal Disease Detection," *J. Imaging*, vol. 9, no. 7, p. 140, Jul. 2023, doi: 10.3390/jimaging9070140.

[3] **J. R. Barr, M. Sobel, and T. Thatcher**, "Upsampling, a Comparative Study with New Ideas," *2022 IEEE 16th International Conference on Semantic Computing (ICSC)*, pp. 318-321, 2022. DOI: 10.1109/ICSC52841.2022.00059.

[4] A. **R. Oh, J. S. Lee, J. Lee, D. W. Nam, S. W. Moon, and W. Yoo**, "On Constructing Vessel Dataset Structure Using GAN-based Data Augmentation," *2021 International Conference on Information and Communication Technology Convergence (ICTC)*, pp. 1701-1702, 2021. DOI: 10.1109/ICTC52510.2021.9620827.

[5] **L. D. Tran, S. M. Nguyen, and M. Arai**, "GAN-based Noise Model for Denoising Real Images," *Proceedings of the International Conference on Machine Learning and Computing (ICMLC)*, pp. 1-9, 2021. DOI: 10.1109/ICMLC.2021.9620827.

[6] N. M. Saad, N. S. M. Noor, A. R. Abdullah, O. Y. Fong, and N. N. S. A. Rahman, "Real-time LCD digit recognition system," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 6, no. 2, pp. 402-411, May 2017. doi: 10.11591/ijeecs.v6.i2.pp402-411

[7] N. Qin, L. Liu, D. Huang, B. Wu, and Z. Zhang, "LeanNet: An efficient convolutional neural network for digital number recognition in industrial products," *Sensors*, vol. 21, no. 3620, pp. 1-16, May 2021. doi: 10.3390/s21113620

[8] S.-H. Tsang, "Review: MobileNetV2 — Light Weight Model (Image Classification)," *Towards Data Science*, Medium, [Online]. Available: https://medium.com/towards-data-science/review-mobilenetv2-light-weight-model-image-classification-8febb490e61c. Accessed: Feb. 9, 2025.