

Received 23 October 2023, accepted 14 November 2023, date of publication 16 November 2023,  
date of current version 22 November 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3333904

## RESEARCH ARTICLE

# Two-Stage Billet Identification Number Recognition Using Label Distribution

**HYEONAH JANG<sup>1</sup>, HYEYEON CHOI<sup>ID1</sup>, BUM JUN KIM<sup>1</sup>, SANG WOO KIM<sup>ID1</sup>, (Member, IEEE),  
AND GYOGWON KOO<sup>2</sup>**

<sup>1</sup>Department of Electrical Engineering, Pohang University of Science and Technology, Pohang 37673, South Korea

<sup>2</sup>Division of Intelligent Robot, Daegu Gyeongbuk Institute of Science and Technology, Daegu 42988, Republic of Korea

Corresponding authors: Sang Woo Kim (swkim@postech.edu) and Gyogwon Koo (gkoo@dgist.ac.kr)

This work was supported by DGIST Research and Development Program of the Ministry of Science and ICT under Grant 23-DPIC-14 and Grant 23-IT-10-03.

**ABSTRACT** With the progressive automation of factories, the demand for deep learning methods capable of recognizing characters is rising. A billet identification number (BIN) is a string of characters that contains all information about the billet, but it is often oriented arbitrarily. Because each plant has different features of data, it requires time and effort to secure enough data to train the model that can be applied to each plant. In addition, the existing BIN recognition model confuses characters with similar shapes when rotated because it shares a feature extractor for angle estimation and character recognition. In this study, we propose a method to solve the problems and improve the BIN recognition performance. We separate the two parts of extracting angles and characters, allowing each module to independently focus on the features of the data. Label distribution is used to enhance the angle estimation accuracy with a small dataset, and the triangular distribution results in the highest accuracy. Finally, to train rotated characters, a large amount of data that are randomly rotated is required, but by separating the angle and character module, the variation within classes is reduced, resulting in high BIN recognition performance even with a small dataset.

**INDEX TERMS** Angle estimation, billet identification, character recognition, label distribution.

## I. INTRODUCTION

Deep learning (DL) has been applied to tasks in various fields [1], [2], [3], including factory automation in the steel industry [4], [5]. Text recognition is imperative in this industry, and DL models are used to recognize billet identification numbers (BINs). BINs contain information related to the material of the billet, which is tailored to the customers' needs [6]. As the characteristics of the billet depend on its material, the incorrect recognition and entry of even a single BIN can cause significant financial loss. Therefore, the character recognition accuracy of BINs should ideally be close to 100%. Character recognition is performed at various industrial sites, and each factory has a different environment; a variety of cameras are used to extract data. Thus, the data become noisy and complicated. In the case of BIN data, the characteristics of

data, such as font and steel type, vary from factory to factory, and the location of the characters is not fixed. Therefore, to achieve high accuracy, it is essential to acquire and train on new data. However, obtaining and labeling data from the factories can be a time-consuming and costly process, so it is crucial to quickly label a small amount of data and train a model to ensure satisfactory performance.

Various algorithms from traditional computer vision methods to neural networks have been proposed [7], [8], [9]. Processing factory data in a traditional method is necessary to modify the algorithm several times to handle new input data. However, the algorithm based on a neural network is easy and simple because a model can be applied directly to other data once developed. In addition, the development of an actively studied DL model has far exceeded the performance of the traditional method. Several DL models have also been developed, increasing detection accuracy. Even DL-based methods have been proposed to identify

The associate editor coordinating the review of this manuscript and approving it for publication was Zhipeng Cai<sup>ID</sup>.

steel products without additional location information of characters required in vision tasks, pre-processing to leave only the area of characters, or noise canceling [10], [11]. This has improved character recognition performance for slabs [12]. An end-to-end model capable of arbitrarily oriented BINs has also been proposed [13]. It simultaneously handles the classification and positioning of extracted features, and achieves high performance on randomly rotated BINs. The area to be recognized in the billet images is a portion of the overall image. If the entire billet image is input into the character recognition model, noise from the background causes performance degradation. Therefore, by extracting and inputting only the area of characters into the model, the performance can be enhanced [14]. Furthermore, various studies are being conducted, including methods that extract individual region of each character [15] and increase the resolution of the extracted characters [16] to improve character recognition performance.

DL method is being applied to character recognition with various datasets, not limited to BIN data. For instance, there are studies focused on recognizing handwritten text in historical documents. This data has low-quality annotations, and a single sentence has multiple annotations due to simultaneous labeling by multiple people and models [17]. Additionally, due to the characteristics of the handwritten dataset, a method for page-level text recognition, along with its associated evaluation metrics, has been proposed [18]. Another data is scene text data, which is the most well-known publicly available dataset in character recognition. Inspired by characters forming sequences to create meaningful words, the DL model extracts additional contextual features which are useful for predicting words [19]. The introduction of an attention module and the utilization of character position features improve character recognition performance [20]. After the introduction of Transformers in the field of image processing, which achieved high performance, the Vision Transformer (ViT) model has been used as the base network for feature extraction [21]. Afterward, several approaches, including modifying the ViT to mix and merge image patches [22], and new frameworks that combine attention mechanisms and transformers [23], have appeared.

Scene text and handwritten data are not rotated, so the proposed methods do not take the angle of the characters into consideration. Furthermore, the two datasets consist of meaningful words, whereas the BIN data is a random sequence of alphabets and numbers, making it impossible to extract contextual features. Finally, the former datasets contain a large amount of trainable data, whereas the BIN data requires training the model with limited data. Therefore, achieving high performance when applying the aforementioned models to BIN data is difficult. The end-to-end BIN recognition model [13] exhibited high performance for BIN data rotated in four directions but performed poorly for BIN data rotated through arbitrary angles. Recognizing characters that are all inclined at the same angle is easier

than recognizing ones that are rotated at any angle. The background and other neighboring characters must also be taken into account when recognizing randomly rotated characters to distinguish two different pairs of characters that appear to be identical when rotated. This prevents the character recognition algorithm from focusing only on the character itself, causing performance degradation. Once all data are oriented similarly, high recognition accuracy is expected as the models can concentrate solely on character recognition. Furthermore, sufficient data for each angle class are required during model training, but obtaining enough data corresponding to each angle evenly takes time and effort. Therefore, algorithms that can be applied to real data must be efficiently trained with fewer data. If the orientations of all data are homogenized, high accuracy is achieved based on training with significantly fewer data. Therefore, we propose a method of homogenizing the orientations of images and then using the image for character recognition to ensure high character recognition performance.

Ambiguous data problems are related to data containing incorrect labels based on subjective judgment. These include a person's age [24], the number and severity of acne [25], the angle of rotation of objects, among others. These data have ambiguous parts to judge; thus, labels can vary from person to person, and labeling requires time and effort. Finally, this labeling load increases label errors. Instead of giving a single value to a label, there is an efficient way to give label weights to surrounding classes that may be confused with ground truth. By giving the margin for the error, the algorithm can learn the information that the prediction is not the ground truth, but the close value around it. The label distribution method is adopted in this study to reduce the loss of predictions close to ground truth so that the model can be trained efficiently.

The proposed method involves two stages—an angle estimation algorithm to homogenize the orientations of images and a character recognition module. We apply a method of making data have homogenized angles using an angle prediction algorithm to increase the character recognition performance. We use label distribution learning to improve the accuracy of the angle estimation algorithm and propose a suitable distribution. The existing label distribution learning method commonly uses the Gaussian distribution with a mean of 0 and a standard deviation of 1, and more suitable distributions may be available for the angle estimation problem. The Gaussian distribution assigns some values to 3–4 classes close to the ground truth, but the values decrease rapidly as one moves further from the ground truth. Thus, it is only effective in corresponding to a few classes and is ineffective on data with many classes. In addition, unlike ordinal data, which are assigned values in ascending order without setting a range for assigning weights around a ground truth, data with periodical classes like BIN angle data require a range for weight assignment. This selected range affects performance; thus, we propose an appropriate method to assign weights to the desired label range.

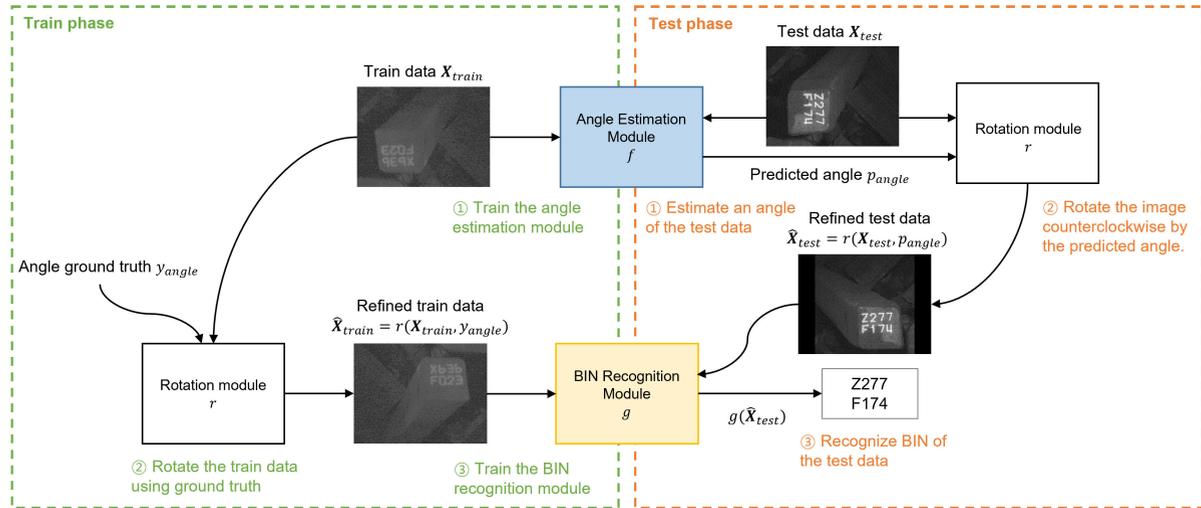


FIGURE 1. The overall structure of two-stage BIN recognition.

The contributions of this paper are as follows:

- 1) The proposed model comprises two dedicated stages, enabling each stage to focus on their respective role to achieve high performance.
- 2) An effective distribution for label distribution learning is identified and applied to ambiguous angle labels to improve angle estimation performance.
- 3) Good performance is achieved using a small dataset. Furthermore, the recognition model can be trained and tested quickly for altered data.

The remainder of the paper is organized as follows. In Section II, we introduce the BIN dataset and describe the proposed method. Section III compares the performance of the proposed algorithm with that of an existing character recognition model. Finally, Sections IV and V discuss and summarize the conclusions of the paper.

## II. PROPOSED METHOD

Recognition of BIN data of homogenized orientation is easier than that of BIN data rotated through arbitrary angles. Plenty of training data are required corresponding to each angle likely to appear during detection to train a model to recognize rotated characters accurately [26], [27]. However, homogenized BIN data has a single angle label; thus, there is no need for a lot of data to be evenly distributed [28]. In this context, we propose a framework that predicts the rotated angles of BIN data, makes the rotation angles of BIN data consistent, and then recognizes characters.

### A. OVERALL FRAMEWORK

Fig. 1 illustrates the overall structure of the proposed algorithm. The framework consists of two stages: the angle prediction module and the BIN recognition module, and the training process and the testing process are slightly different. The modules are trained separately, and they operate sequentially in the testing process.

**Train** In the angle estimation module, we train on the train data  $\mathbf{X}_{train} \in \mathbb{R}^{H \times W}$ . In the BIN recognition module, we train on the refined train data  $\hat{\mathbf{X}}_{train}$ . Here,  $\hat{\mathbf{X}}_{train}$  is the result of rotating  $\mathbf{X}_{train}$  counterclockwise by the angle ground truth  $y_{angle} \in \mathbb{R}$  using the rotation module  $r$ . The train phase involves separately training the angle estimation module and the BIN recognition module in this manner.

**Test** In the test phase, we initiate the process by inputting the test data  $\mathbf{X}_{test}$ , which lacks angle information, into the angle estimation module  $f$  to predict the angle  $p_{angle} \in \mathbb{R}$ . Subsequently, we rotate  $\mathbf{X}_{test}$  using the rotation module  $r$  to obtain  $\hat{\mathbf{X}}_{test}$ . Finally, the BIN is extracted from  $\hat{\mathbf{X}}_{test}$  using the BIN recognition module  $g$ .

### B. ANGLE ESTIMATION MODULE

The angle estimation module utilizes a pre-trained ResNet-18 as the base model (Fig. 2). The output of the last fully connected layer in ResNet-18 is modified to have a dimension of  $C_{angle} = 360$ , where  $C_{angle}$  is the number of angle class. The module is trained to minimize the multi-class cross-entropy loss as follows.

$$L_{angle} = - \sum_{i=0}^{C_{angle}-1} y_{angle,i} \log(p_{angle,i}). \quad (1)$$

The label distribution and prediction of the module corresponds to  $\mathbf{y}_{angle} \in \mathbb{R}^{C_{angle}}$  and  $\mathbf{p}_{angle} = f(\mathbf{X}) \in \mathbb{R}^{C_{angle}}$ .

The accuracy is determined based on the highest probability predicted angle. Predicted angles within  $\pm 10$  degrees from the ground truth are considered correct. The values of  $y_{angle}$  and  $p_{angle}$  range between  $0^\circ$  and  $359^\circ$ . We assume that  $0^\circ$  follows  $359^\circ$ ; thus, the difference between  $359^\circ$  and  $0^\circ$  is  $1^\circ$ , not  $359^\circ$ . In this paper,  $e_{angle}(x)$  denotes the minimum difference between two angles and cannot be negative (2). The parameter  $\alpha$  denotes the maximum absolute error that the angle estimation algorithm treats as the correct answer

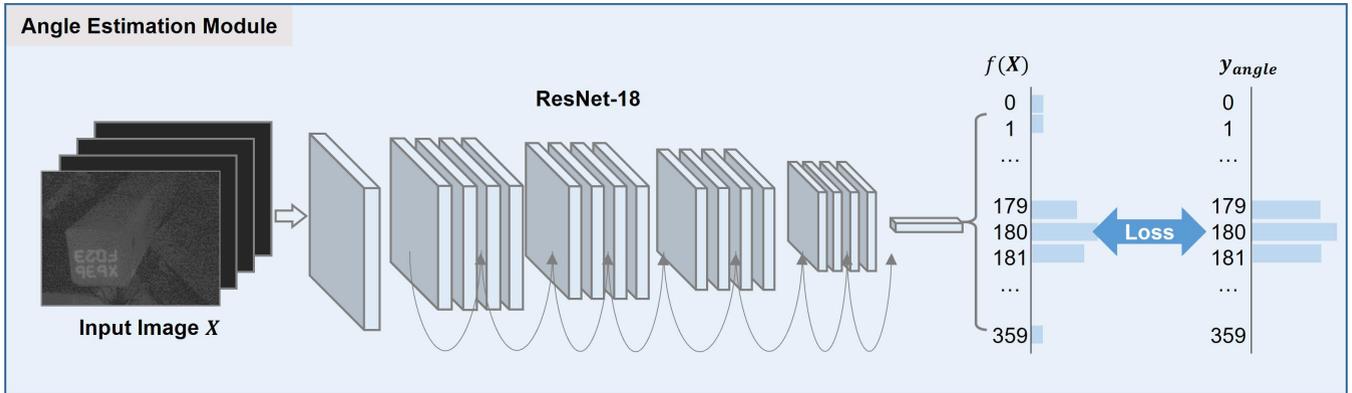


FIGURE 2. Angle estimation module.

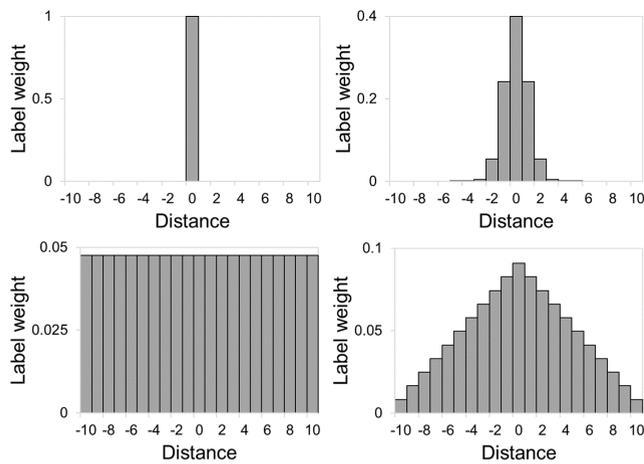


FIGURE 3. Label distribution graph: (a) One-hot encoded graph, (b) Gaussian distribution graph, (c) Uniform distribution graph, and (d) Triangular distribution graph.

within the range of the predicted angle away from the ground truth. The larger the tilted angle of the BIN image, the more difficult it is to recognize the character. Therefore,  $\alpha$  was set to  $10^\circ$  empirically.

$$e_{angle}(x) = \min(|y_{angle} - x|, 360 - |y_{angle} - x|). \quad (2)$$

Label distribution is adopted to increase detection accuracy and consider the periodicity of the labels. The rotation angle of BIN data must be predicted to homogenize the orientations of randomly rotated data. An angle estimation algorithm is proposed in this study for this purpose. However, the labels themselves may be erroneous [29] owing to manual labeling. Moreover, angle labels are periodic. By using label distribution, values of label are distributed around the ground truth based on one-hot encoded labels; This enables the consideration of variable loss based on the distance from the ground truth during the prediction of surrounding values. Therefore, label distributions are more effective than squared error loss at reflecting the periodicity of angle labels.

The distribution should be applied to certain labels within the desired range of classes because the BIN angle data

comprises 360 classes, and  $\alpha$  was used for the class margin of label distribution. We consider three distributions—the Gaussian distribution [24], uniform distribution, and triangular distribution. The Gaussian distribution assigns values to the neighboring classes following the Gaussian distribution with the mean value assigned to the ground truth (3). The uniform distribution assigns the same value to all neighboring classes, including the ground truth (4). The triangular distribution assigns values to the surrounding classes following the triangular distribution, with the maximum value assigned to the ground truth (5). Fig. 3 depicts graphs of four types of label distributions. The X-axis represents the error with respect to the ground truth, and the Y-axis represents the value assigned to the label.

$$h_{Gauss}(x) = \begin{cases} \frac{1}{\sqrt{2\pi}} e^{-\frac{e_{angle}(x)^2}{2}}, & \text{if } e_{angle}(x) \leq \alpha. \\ 0, & \text{otherwise.} \end{cases} \quad (3)$$

$$h_{Uni}(x) = \begin{cases} \frac{1}{2\alpha + 1}, & \text{if } e_{angle}(x) \leq \alpha. \\ 0, & \text{otherwise.} \end{cases} \quad (4)$$

$$h_{Tri}(x) = \begin{cases} -\frac{1}{(\alpha + 1)^2} e_{angle}(x) + \frac{1}{\alpha + 1}, & \text{if } e_{angle}(x) \leq \alpha. \\ 0, & \text{otherwise.} \end{cases} \quad (5)$$

### C. BIN RECOGNITION MODULE

The BIN recognition module [13] consists of one encoder and two decoders (Fig. 4). The encoder serves as a feature extractor, while the decoders play the role of generators. The feature extractor utilizes a modified VGG-19 model, in which the fully connected layers are removed. The two decoders are composed of a positioning part and a classification part. The positioning part generates the order of characters, while the classification part generates the classes of characters. The decoder uses transposed convolution to increase the size, followed by an elementwise sum of the output feature from the decoder and encoder, both of which have the same size. The two decoders have a similar structure,

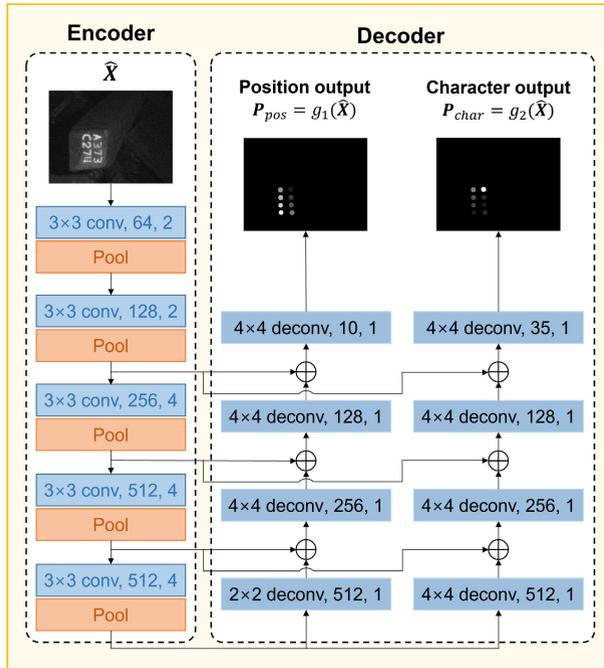


FIGURE 4. BIN recognition module.

but they are trained to predict different ground truths. As a result, the number of output channels differs between the positioning part (10 channels) and the classification part (35 channels). In the Fig. 4, the blue rectangles represent convolutional layers. For example, “ $3 \times 3$  conv, 64, 2” indicates that a  $3 \times 3$  convolution with an output channel of 64 is applied twice. Similarly, “ $4 \times 4$  deconv, 10, 1” indicates that a  $4 \times 4$  deconvolution with an output channel of 10 is executed once.

The loss is calculated by computing the pixel-wise multi-class cross-entropy between the output obtained from each decoder and the corresponding ground truth. The BIN recognition module is trained to minimize the sum of the position loss from the position class part and the character loss from the character class part.

$$L_{BIN} = L_{pos} + L_{char}, \quad (6)$$

$$L_{pos} = -\frac{1}{H \cdot W} \times \left( \sum_{y=1}^H \sum_{x=1}^W \sum_{i=0}^{C_{pos}-1} \mathbf{Y}_{pos,i}(x, y) \log(\mathbf{P}_{pos,i}(x, y)) \right), \quad (7)$$

$$L_{char} = -\frac{1}{H \cdot W} \times \left( \sum_{y=1}^H \sum_{x=1}^W \sum_{i=0}^{C_{char}-1} \mathbf{Y}_{char,i}(x, y) \log(\mathbf{P}_{char,i}(x, y)) \right), \quad (8)$$

where  $C_{pos}$  and  $C_{char}$  are the number of position classes and the number of character classes, respectively, with  $C_{pos}$  being 10 and  $C_{char}$  35,  $\mathbf{Y}_{pos} \in \mathbb{R}^{H \times W \times C_{pos}}$  and

$\mathbf{Y}_{char} \in \mathbb{R}^{H \times W \times C_{char}}$  are the ground truth of BIN position and character, respectively, and  $\mathbf{P}_{pos} \in \mathbb{R}^{H \times W \times C_{pos}}$  and  $\mathbf{P}_{char} \in \mathbb{R}^{H \times W \times C_{char}}$  are the output of the position decoder part and character decoder part of BIN recognition module, respectively. The final string is extracted using the position output and character output generated by the decoders. The position output represents the pixel-wise order information of characters, and the character output the pixel-wise class information. By selecting the most frequent character class among the pixels in each position class and outputting them in sequential order, we obtain the recognition value of BIN.

### III. EXPERIMENTS

#### A. DATASETS

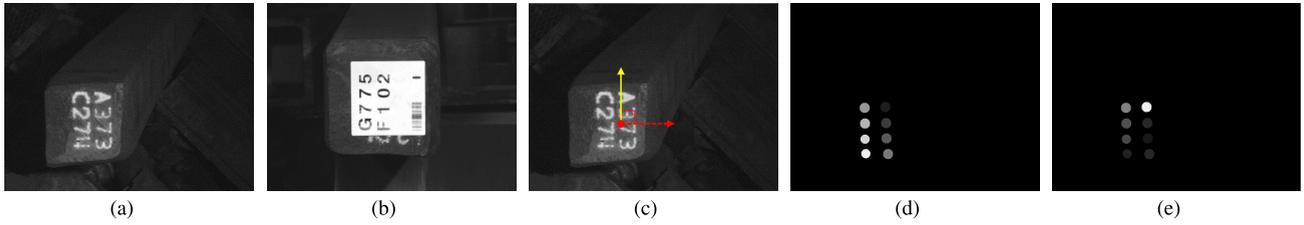
BIN data is a rectangular gray image of  $480 \times 640$  size on the cross-section of a long bar shaped billet. There are two types of BIN painting: paint-type (Fig. 5(a)) and sticker-type (Fig. 5(b)). Paint-type BINs are directly painted in white on the surface of gray billets at the end, and they consist of two or three lines of eight or nine capital letters and numbers. The first and second lines include four characters, and an optional third line comprises one character. Sticker-type BINs are printed in black on white square paper, and they comprise two lines of seven or eight characters. The font and style differ from the paint-type and the size of characters is smaller than that of the paint-type. There are few characters that are erased or poorly visible in sticker-type BINs, but there are cases where characters from the paint-type are not fully concealed and protrude slightly behind the sticker.

In BIN, the first character in the first row is an alphabet. In addition, the alphabets O and I are not used because they are difficult to distinguish from the numbers 0 and 1, respectively. The billet may be rotated through arbitrary angles, and thus, so may the BIN. Therefore, BIN recognition methods should be capable of handling data rotated through various angles. In this paper, 5,719 paint-type BIN data are considered, with most of the data corresponding to labels of  $0^\circ$ ,  $90^\circ$ ,  $180^\circ$ ,  $270^\circ$ , and some having other angles. Additionally, 2,415 sticker-type BIN data are included, with most of the data concentrated on surrounding labels of  $270^\circ$ .

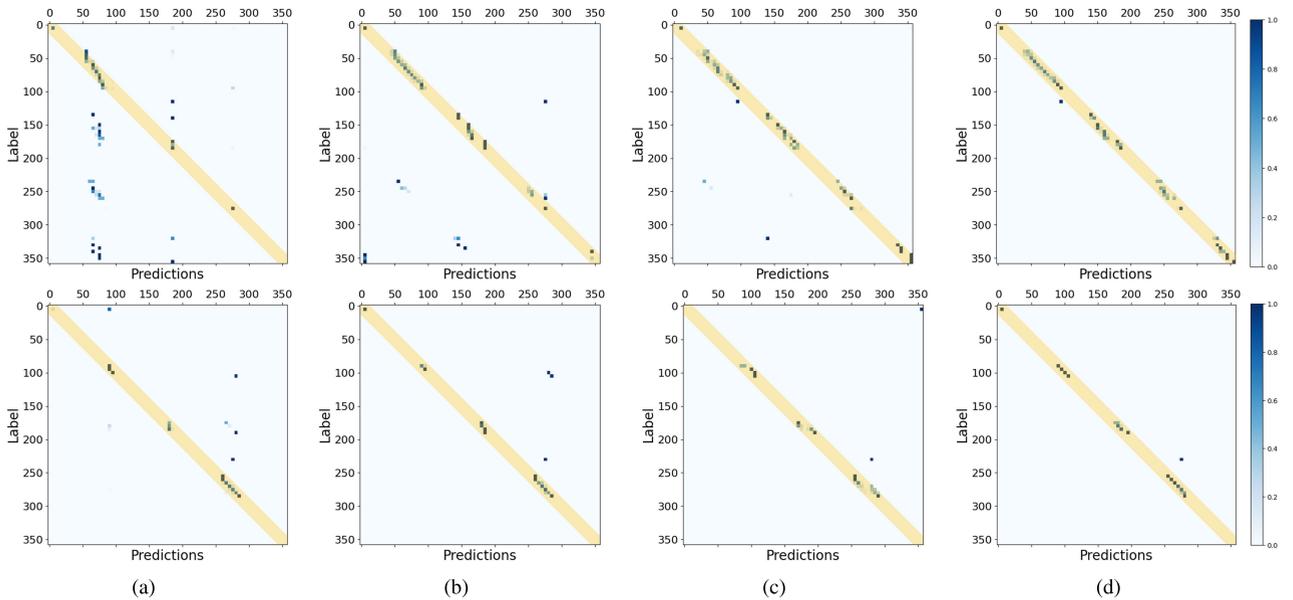
Each BIN image has a label of an angle and a string to be used for angle estimation module and BIN recognition module. In Fig. 5(c), a red dot is placed in the center of the string, the direction of the string is drawn with a red line, and the angle difference from the yellow line is used as an angle label for the angle estimation module. The order and class of the characters are generated from the center of each character. Fig. 5(d), 5(e) demonstrates the order label on which characters are listed and the class label of the character. These two labels are used to train the BIN recognition module.

#### B. IMPLEMENTATION DETAILS

In the experiment, a total of 5,719 BIN data were used, including 3,811 for training, 638 for validation, and 1,270



**FIGURE 5.** Example of rotated BIN data of (a) paint-type and (b) sticker-type. Our method uses three types of labels: (c) angle label, (d) order label, and (e) character class label, and these are visualizations of each label.



**FIGURE 6.** Confusion matrices of the detection performances based on (a) One-hot, (b) Gaussian, (c) Uniform, and (d) Triangular distributions. The upper figures represent the results for paint-type BIN data, while the lower figures sticker-type. The points in the yellow area are correct answers within the tolerance range, and the other points are the incorrectly predicted samples. In (d) triangular, most of the points are in the yellow area, and the incorrectly predicted samples also do not deviate significantly from the yellow area.

**TABLE 1.** Comparative accuracies achieved using different label distributions with respect to the one-hot label. Acc denotes the ratio of the correct predictions and the total number of testing data.

Method	Acc.	
	Paint-type	Sticker-type
One-hot	77.3	91.7
Gaussian	97.5	98.8
Uniform	99.1	97.9
Triangular (ours)	<b>99.8</b>	<b>99.6</b>

for testing. Additionally, 2,414 sticker-type BIN data were used, with 1,932 for training, 241 for validation, and 241 for testing. Both datasets are used in the label distribution and BIN recognition experiments. The angle estimation algorithm is trained using a learning rate of 0.01, a weight decay of 0.0005, and a batch size of 32. The model is optimized using a stochastic gradient descent optimizer and a cosine annealing schedule [30]. The BIN recognition model is trained using a learning rate of 0.0001, a batch size of 1, and the Adam optimizer. The learning rate is scheduled by multiplying it

by 0.95 in each epoch, and dropout with a keep probability of 0.2 is utilized. An early stopping point for training after 10 epochs is used based on validation loss in the absence of continuous updates. The two-stage model is trained on a computer running Windows 10 with Intel Core i7-10700k CPU @ 3.80GHz, 32 GB RAM, and NVIDIA GeForce RTX 3090.

**C. LABEL DISTRIBUTION**

The network correctly recognizes BIN characters even when the BIN image is slightly tilted. Thus, a margin is set for the calculation of accuracy. Predicted results within an error of 10 ° with respect to the ground truth are treated as correct answers.

Table. 1 summarizes the accuracy of angle estimation. In the case of one-hot label, the accuracy is low, and several observations lie outside the diagonal in Fig. 6 because the distance between wrong predictions and the ground truth is not reflected in the calculation of loss in this case. For example, the predicted loss corresponding to 180° and 179°, and 180° and 0°, are identical, even though the former

**TABLE 2.** Performance comparison of the BIN recognition model based on the label distribution. Accuracy (Acc.) represents the percentage of correct predictions out of the total number of testing data. "Two" means our proposed 2-stage framework. "-O," "-G," "-U," and "-T" represent different label distributions used in the angle estimation module: one-hot encoded label, Gaussian distribution, uniform distribution, and triangular distribution.

Method	Paint-type				Sticker-type			
	Precision	Recall	F <sub>1</sub> -score	Acc.	Precision	Recall	F <sub>1</sub> -score	Acc.
Koo et al. [13]	0.9877	0.9873	0.9875	91.7	0.9764	0.9748	0.9756	87.1
Two-O	0.9147	0.8721	0.8929	92.1	0.9470	0.9230	0.9349	89.6
Two-G	0.9833	0.9803	0.9818	96.8	0.9875	0.9858	0.9866	95.9
Two-U	0.9928	0.9912	0.9920	97.8	0.9901	0.9890	0.9896	92.1
Two-T	<b>0.9982</b>	<b>0.9980</b>	<b>0.9981</b>	<b>99.1</b>	<b>0.9923</b>	<b>0.9912</b>	<b>0.9917</b>	<b>96.3</b>

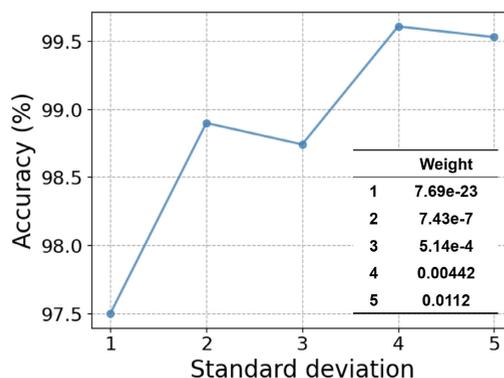
loss should be much smaller. Ideally, the loss should be directly proportional to the proximity between the angles. The Gaussian and uniform distributions improved the model accuracy in relation to the accuracy of the one-hot label, but wrong samples are still observed. The triangular distribution yields the highest accuracy, and the corresponding error is small as well.

#### D. COMPARISON WITH STATE-OF-THE-ART

The two-stage model exhibits higher accuracy compared to using the BIN recognition module alone. The highest BIN recognition accuracy is achieved when images with homogenized angles are used to train the model. The data presented in Table 2 indicate that the BIN recognition model performs better when the angle estimation model predicts rotational angles accurately. The angle estimation accuracy is the highest when the triangular distribution is used as the label distribution. Thus, the highest BIN recognition accuracy is achieved using the triangular distribution as the label distribution.

The BIN is incorrectly predicted in some cases, e.g., letters are detected to be identical to ambient letters and those with similar shapes, with the first type being more common. Two-stage BIN recognition alleviates this problem and predicts each BIN character independently without referring to surrounding characters. Furthermore, the proposed network does not confuse letters with those with similar shapes after being rotated.

Due to the infrequency of recognizing non-existent characters or detecting missing characters, the experimental results, excluding the one-hot encoded labels of the two-stage model, achieve an  $F_1$  score of 0.98 or higher. In particular, the two-stage model with the triangular distribution achieves the highest  $F_1$  score, precision, and recall values. The two-stage model, except for the triangular distribution, shows lower precision, recall, and  $F_1$  score compared to using the BIN recognition module independently. This is due to the two-stage model's BIN recognition module being trained only on characters with the same angle, resulting in incomplete character recognition if the angle prediction module fails to predict the angle correctly. Using only the BIN recognition module allows recognition of rotated characters, but the overall

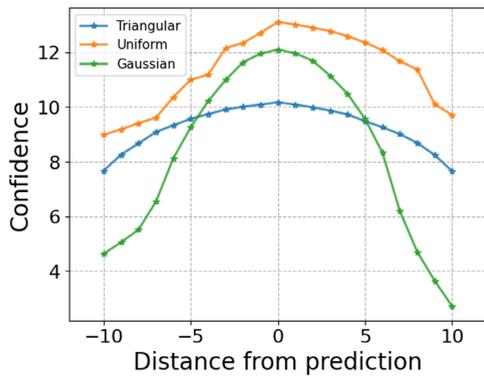


**FIGURE 7.** Accuracy of angle prediction based on the standard deviation of the Gaussian distribution.

performance of character recognition is compromised. The two-stage model, which enhances performance by focusing on character learning with consistent angles, is suitable for practical industrial applications as it treats single incorrect characters and completely wrong BIN strings equally.

#### IV. DISCUSSION

When cross-entropy loss is used, the original single-label distribution exhibits identical losses corresponding to the pair,  $10^\circ$  and  $20^\circ$ , and the pair,  $10^\circ$  and  $100^\circ$ . However, ideally, small errors should correspond to small losses. The proposed model exhibits high accuracy when appropriate weights are assigned to classes that are close to the ground truth. Model accuracy also depends on the type of distribution used. Assigning smaller weights with progressive distance from the ground truth yields higher accuracy than assigning constant weights to all labels. Moreover, moderately small weights yield better results than very small weights for labels that are very far from the ground truth. We conducted additional experiments to investigate the impact of varying the size of the standard deviation in the Gaussian distribution and assigning values to classes that have maximum allowable error from the ground truth. In Fig. 7, as the standard deviation decreases, there is a tendency for the accuracy to decrease. The weights in the table correspond to the smallest weight value for each distribution, which is set based on the respective standard deviation. When comparing the values



**FIGURE 8.** The average confidence values of the predicted classes by the angle estimation module varied based on the label distribution.

of the smallest weights, the minimum weight value from the triangular distribution lies between the smallest weight values of Gaussian distributions weight standard deviations of 4 and 5. Thus, the value of the smallest weight should not be smaller than 0.00442, which is a point where the accuracy begins to decrease.

The Gaussian distribution assigns high values to labels belonging to the correct class, but the value is too high considering the values assigned to surrounding classes. The uniform distribution assigns equal values to all labels within  $10^\circ$ , but does not emphasize the correct class. The triangular distribution is observed to be an optimal combination of the two—the highest value is assigned to the ground truth class, but the value is not excessive in relation to those assigned to surrounding classes. Thus, the triangular distribution yields the highest angle estimation accuracy among the three aforementioned distributions.

The dataset used in this paper contains a large number of classes, and it is possible that there are inaccuracies in the assigned labels, so we consider the classes surrounding a single ground truth as correct answers. In contrast to the triangular distribution, the Gaussian distribution requires parameter adjustment to find an appropriate standard deviation as the number of classes and the tolerance range changes. Therefore, when the data has such characteristics, it is advantageous to apply the triangular distribution. In addition, we verify that the confidence predicted by the angle estimation module has a distribution similar to that of the label. Fig. 8 demonstrates that using the triangular distribution yields the most symmetric outcomes. However, the other graphs deviate from symmetry. This indicates that using the triangular distribution ensures that the assigned weights for each class are effectively applied.

The 2-stage framework method with the addition of the angle estimation module significantly improved the BIN recognition accuracy on a limited dataset. The previously high-performing BIN recognition model [13] encountered a decline in accuracy from 99% to 92% when the amount of training data reduced from 15,763 images to 3,811 images, which is approximately a 24% decrease. To address the 129318

performance decline due to the lack of data, we opted for reducing feature variance as a solution. When all characters have the same orientation, the feature variance to be learned is reduced, allowing for sufficient feature learning with a smaller training dataset. Therefore, we constructed a 2-stage framework with the addition of an angle estimation module, resulting in an improvement in BIN recognition accuracy from 91.7% to 99.1% for paint-type BIN and from 87.1% to 96.3% for sticker-type BIN. The results indicate that by reducing the feature variance of the training data in situations with limited data, the model's performance could be enhanced.

## V. CONCLUSION

In this paper, we proposed a method for estimating the rotation angle of BIN data using label distribution and the 2-stage BIN recognition framework. This method achieved a high BIN recognition performance with a randomly rotated and limited training data.

In the 2-stage framework, features for angles and characters are separately trained in the angle estimation module and the character recognition module, respectively. Each module focuses on its respective features independently, and the reduction in the variation of the features being trained leads to improved performance in each module. Consequently, the overall framework's performance is enhanced. In addition, employing triangular distribution labels in the angle estimation module resulted in a higher accuracy of 22.5%p for paint-type BIN data and 7.9%p for sticker-type BIN data compared to one-hot encoded labels.

In future work, we will validate whether the triangular distribution labels used in the angle estimation module represent the most optimized distribution and whether aligning the range for treating correct answers with the range that assigns values to labels results in the highest performance.

## REFERENCES

- [1] J. Xu, M. Kovatsch, D. Mattern, F. Mazza, M. Harasic, A. Paschke, and S. Lucia, "A review on AI for smart manufacturing: Deep learning challenges and solutions," *Appl. Sci.*, vol. 12, no. 16, p. 8239, Aug. 2022.
- [2] J. Wang, Y. Ma, L. Zhang, R. X. Gao, and D. Wu, "Deep learning for smart manufacturing: Methods and applications," *J. Manuf. Syst.*, vol. 48, pp. 144–156, Jul. 2018.
- [3] T. Kotsiopoulos, P. Sarigiannidis, D. Ioannidis, and D. Tzovaras, "Machine learning and deep learning in smart manufacturing: The smart grid paradigm," *Comput. Sci. Rev.*, vol. 40, May 2021, Art. no. 100341.
- [4] Z. Fan, J. Lu, B. Qiu, T. Jiang, K. An, A. N. Josephraj, and C. Wei, "Automated steel bar counting and center localization with convolutional neural networks," 2019, *arXiv:1906.00891*.
- [5] D. Yang, Y. Cui, Z. Yu, and H. Yuan, "Deep learning based steel pipe weld defect detection," *Appl. Artif. Intell.*, vol. 35, no. 15, pp. 1237–1249, Sep. 2021.
- [6] G. Koo, J. P. Yun, S. J. Lee, H. Choi, and S. W. Kim, "End-to-end billet identification number recognition system," *ISIJ Int.*, vol. 59, no. 1, pp. 98–103, Jan. 2019.
- [7] A. D. O. Riordan, D. Toal, T. Newe, and G. Dooly, "Object recognition within smart manufacturing," *Proc. Manuf.*, vol. 38, pp. 408–414, Jan. 2019.
- [8] T.-C. Hsu, Y.-H. Tsai, and D.-M. Chang, "The vision-based data reader in IoT system for smart factory," *Appl. Sci.*, vol. 12, no. 13, p. 6586, Jun. 2022.

- [9] X. Zhang and Y. Wang, "Industrial character recognition based on improved CRNN in complex environments," *Comput. Ind.*, vol. 142, Nov. 2022, Art. no. 103732.
- [10] S. J. Lee, J. Ban, H. Choi, and S. W. Kim, "Localization of slab identification numbers using deep learning," in *Proc. 16th Int. Conf. Control, Autom. Syst. (ICCAS)*, Gyeongju, South Korea, Oct. 2016, pp. 1174–1176.
- [11] L. Kang, Y. Chen, W. Jhong, and C. Hsu, "Deep learning-based identification of steel products," in *Proc. Int. Conf. Intell. Inf. Hiding Multimedia Signal Process.*, Sendai, Japan, 2018, pp. 315–323.
- [12] S. J. Lee, W. Kwon, G. Koo, H. Choi, and S. W. Kim, "Recognition of slab identification numbers using a fully convolutional network," *ISIJ Int.*, vol. 58, no. 4, pp. 696–703, 2018.
- [13] G. Koo, J. P. Yun, H. Choi, and S. W. Kim, "Unified deep neural networks for end-to-end recognition of multi-oriented billet identification number," *Expert Syst. Appl.*, vol. 168, Apr. 2021, Art. no. 114377.
- [14] W. Jeong, H. Choi, B. J. Kim, H. Jang, D. G. Lee, D. Lee, and S. W. Kim, "Adapting masking network for Bloom identification number recognition to different domains," in *Proc. 22nd Int. Conf. Control, Autom. Syst. (ICCAS)*, Busan, South Korea, Nov. 2022, pp. 257–261.
- [15] L. Q. Xiao, C.-C. Lin, C.-H. Chen, and J.-C. Lee, "Steel billet serial number detection based on deep learning," in *Proc. IET Int. Conf. Eng. Technol. Appl. (IET-ICETA)*, High Tatras, Slovakia, Oct. 2022, pp. 1–2.
- [16] Z. Wang, Y. Dong, D. Niu, M. Liu, Q. Li, and X. Chen, "Billet number recognition based on ESRGAN and improved YOLOv5," in *Proc. 37th Youth Academic Annu. Conf. Chin. Assoc. Autom. (YAC)*, Beijing, China, Nov. 2022, pp. 1384–1389.
- [17] S. Tarride, T. Faine, M. Boillet, H. Mouchère, and C. Kermorvant, "Handwritten text recognition from crowdsourced annotations," in *Proc. 7th Int. Workshop Historical Document Imag. Process.*, San Jose, CA, USA, Aug. 2023, pp. 1–6.
- [18] E. Vidal, A. H. Toselli, A. Ríos-Vila, and J. Calvo-Zaragoza, "End-to-end page-level assessment of handwritten text recognition," *Pattern Recognit.*, vol. 142, Oct. 2023, Art. no. 109695.
- [19] J. Baek, G. Kim, J. Lee, S. Park, D. Han, S. Yun, S. J. Oh, and H. Lee, "What is wrong with scene text recognition model comparisons? Dataset and model analysis," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 4715–4723.
- [20] X. Yue, Z. Kuang, C. Lin, H. Sun, and W. Zhang, "RobustScanner: Dynamically enhancing positional clues for robust text recognition," in *Proc. Eur. Conf. Comput. Vis.*, Aug. 2020, pp. 135–151.
- [21] R. Atienza, "Vision transformer for fast and efficient scene text recognition," in *Proc. Int. Conf. Document Anal. Recognit.*, Lausanne, Switzerland, 2021, pp. 319–334.
- [22] Y. Du, Z. Chen, C. Jia, X. Yin, T. Zheng, C. Li, Y. Du, and Y.-G. Jiang, "SVTR: Scene text recognition with a single visual model," 2022, *arXiv:2205.00159*.
- [23] H. Zhang, G. Luo, J. Kang, S. Huang, X. Wang, and F.-Y. Wang, "GLaLT: Global-local attention-augmented light transformer for scene text recognition," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Mar. 6, 2023, doi: [10.1109/TNNLS.2023.3239696](https://doi.org/10.1109/TNNLS.2023.3239696).
- [24] B.-B. Gao, C. Xing, C.-W. Xie, J. Wu, and X. Geng, "Deep label distribution learning with label ambiguity," *IEEE Trans. Image Process.*, vol. 26, no. 6, pp. 2825–2838, Jun. 2017.
- [25] X. Wu, N. Wen, J. Liang, Y.-K. Lai, D. She, M.-M. Cheng, and J. Yang, "Joint acne image grading and counting via label distribution learning," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Seoul, South Korea, Oct. 2019, pp. 10642–10651.
- [26] F. Abramovich and M. Pensky, "Classification with many classes: Challenges and pluses," *J. Multivariate Anal.*, vol. 174, Nov. 2019, Art. no. 104536.
- [27] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *J. Big Data*, vol. 6, no. 1, pp. 1–48, Jul. 2019.
- [28] R. L. Figueroa, Q. Zeng-Treitler, S. Kandula, and L. H. Ngo, "Predicting sample size required for classification performance," *BMC Med. Inform. Decis. Making*, vol. 12, no. 1, pp. 1–10, Feb. 2012.
- [29] H. Song, M. Kim, D. Park, Y. Shin, and J.-G. Lee, "Learning from noisy labels with deep neural networks: A survey," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 34, no. 11, pp. 8135–8153, Nov. 2022, doi: [10.1109/TNNLS.2022.3152527](https://doi.org/10.1109/TNNLS.2022.3152527).
- [30] I. Loshchilov and F. Hutter, "SGDR: Stochastic gradient descent with warm restarts," in *Proc. Int. Conf. Learn. Represent. (ICLR)* Toulon, France, 2017, pp. 1–16.



**HYEONAH JANG** received the B.S. and M.S. degrees in electrical engineering from the Pohang University of Science and Technology, Pohang, Republic of Korea, in 2020 and 2022, respectively, where she is currently pursuing the Ph.D. degree in electrical engineering.

Her current research interests include computer vision and deep learning.



**HYEYEON CHOI** received the B.S. and M.S. degrees in electrical engineering from the Pohang University of Science and Technology, Pohang, South Korea, in 2017 and 2019, respectively, where she is currently pursuing the Ph.D. degree in electrical engineering.

Her current research interests include defect detection, semantic segmentation, image fusion, and deep neural networks.



**BUM JUN KIM** received the B.S. and M.S. degrees in electrical engineering from the Pohang University of Science and Technology, Pohang, South Korea, in 2018 and 2020, respectively, where he is currently pursuing the Ph.D. degree in electrical engineering.

His research interest includes deep learning for computer vision.



**SANG WOO KIM** (Member, IEEE) was born in Pyeongtaek, South Korea, in 1961. He received the B.S., M.S., and Ph.D. degrees in control and instrumentation engineering from Seoul National University, Seoul, South Korea, in 1983, 1985, and 1990, respectively.

In 1992, he joined the Pohang University of Science and Technology, Pohang, South Korea, as an Assistant Professor, where he is currently a Full Professor with the Department of Electrical Engineering. In 1993, he was a Visiting Fellow with the Department of System Engineering, The Australian National University, Canberra, ACT, Australia. His research interests include intelligent control, process automation, fault diagnosis, battery management systems, and deep neural networks.



**GYOGWON KOO** received the B.S. degree from the Department of Electrical Engineering, Kyungpook National University, Daegu, South Korea, in 2013, and the M.S. and Ph.D. degrees from the Department of Electrical Engineering, Pohang University of Science and Technology, Pohang, South Korea, in 2015 and 2020, respectively.

Since 2020, he has been a Senior Researcher with the Daegu Gyeongbuk Institute of Science and Technology. His current research interests include computer vision and deep learning.

...