International Journal of Networking and Computing – www.ijnc.org, ISSN 2185-2847
Volume 12, Number 2, pages 339–358, July 2022

Single and Ensemble CNN Models with Out-Category Penalty for Image Classification

Yuta Suzuki
Computer Software Laboratory, Kanazawa University
Kakuma, Kanazawa 920-1192, JAPAN

Daiki Kuyoshi
Computer Software Laboratory, Kanazawa University
Kakuma, Kanazawa 920-1192, JAPAN

Satoshi Yamane
Computer Software Laboratory, Kanazawa University
Kakuma, Kanazawa 920-1192, JAPAN

Received: January 26, 2022
Revised: May 4, 2022
Accepted: May 26, 2022
Communicated by Shuichi Ichikawa

Abstract

In recent years, the technology of machine learning has been developing rapidly. Among them, neural network technology has had a great impact on various fields such as image recognition and natural language processing. Among them, CNN or Convolutional Neural Network have been effective in the field of image recognition. However, most of these CNNs learn only the features of the image, and do not learn the meta-information of the image. In this study, we proposed a CNN and its ensemble method that can learn meta-information by using out-category penalties. Experiments were conducted on the CIFAR-10 and CIFAR-100 datasets, and the results show that the proposed method has high accuracy and small out-category error in both single and ensemble models.

Keywords: Convolutional Neural Network, Image Recognition, Ensemble Learning

1 Introduction

In recent years, the technology of machine learning has been developing rapidly. Among them, neural network technology has had a great impact on various fields such as image recognition and natural language processing. There are various types of neural networks, such as deep neural networks(DNN) [1], convolutional neural networks(CNN) [2], and recurrent neural networks(RNN) [3]. Among them, CNNs have been effective in the field of image recognition.

However, most of these CNNs learn only the features of the image, and do not learn the meta-information of the image. For example, when training a dataset of handwritten numeric images, such as MNIST [1], the CNN learns based on the features of each image and the correct answer label, regardless of the concept of numeric values. As a result, it is easy to make mistakes with large numerical errors, such as mistaking "9" for "0", which is close in terms of image feature. A model that overcomes this problem can avoid fatal mistakes, even if there are cases where it makes mistakes. This is a property that is valuable in practical applications.

In 2021, a work was presented on CNNs and their ensemble models that learn to reduce the numerical error as well as the image features in handwritten numeric images [4], [5], [6]. And it was shown that the numerical error was reduced by the method of this work. However, since the methods in these works learn by calculating penalties using numerical information, they could not be applied to image dataset that do not have numerical values such as CIFAR-10 [7].

Therefore, this study aimed to reduce the number of errors beyond acceptable ranges in non-numeric data. In this study, we propose a CNN and its ensemble method that can learn meta-information even for images other than those containing numerical information such as handwritten numeric images. Out-category penalty is a penalty that is categorized in advance with meta-information separate from the correct label, and imposed additionally for misidentification beyond that categorization. The proposed method then learns to reduce these penalties. Our proposed method can be applied to any dataset as long as there is meta-information that can be categorized separately from the correct answer labels.

2 Related Works

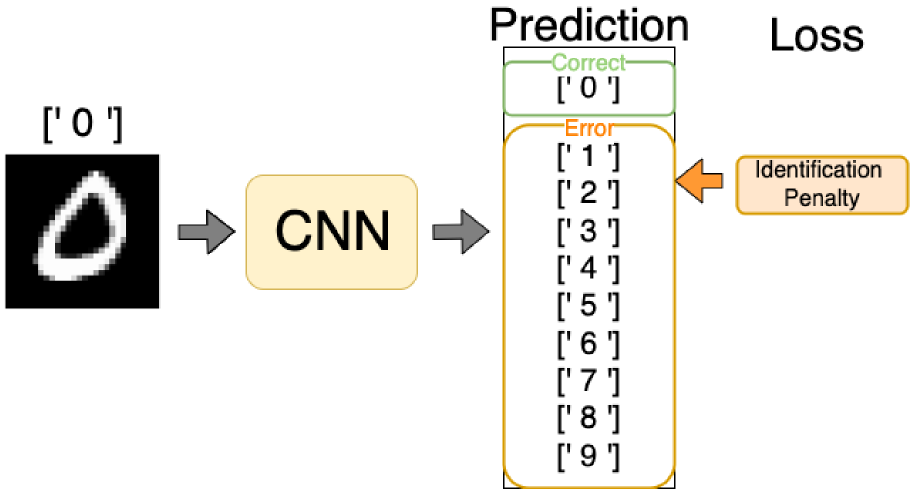
In this section, we present two works on CNN models that learn to reduce numerical errors and the datasets used in this work. Then, this study is an extension of these works.

2.1 An Image Classification Model that Learns Image Features and Numerical Information

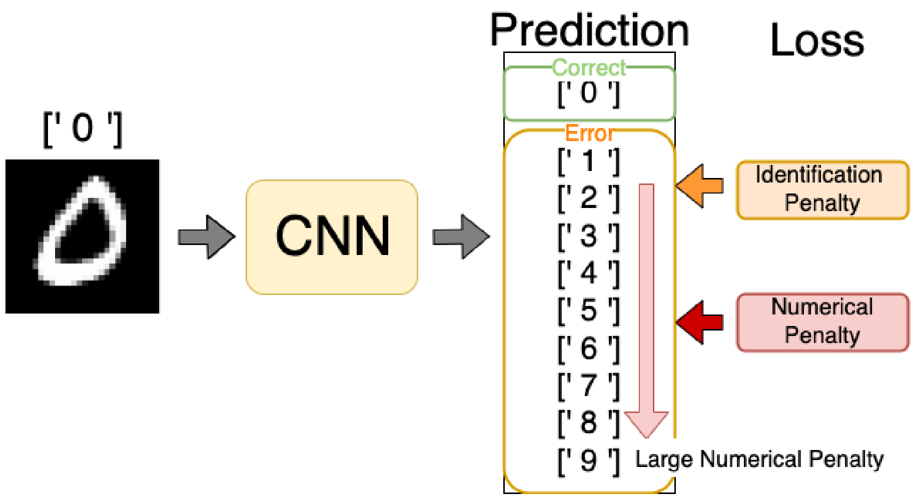
This work, published in 2021, proposed a method for learning not only image features but also numerical information in handwritten numeric recognition [4], [5]. The proposed model in this work calculates penalties for misidentification as well as penalties for the size of numerical error, and learns to reduce these values. For example, Figure 1 shows how a normal CNN and a CNN using this method learn handwritten numeric characters from 0 to 9. Prediction in Figure 1 is the output of the CNN and is in the form of a vector of 1 x the number of correct labels. In Figure 1, consider the case when a "0" image is input. In a normal CNN, identification penalties are applied to predictions other than "0" of the correct label, as shown in Figure 1a. On the other hand, this method adds a numerical penalty in addition to the identification penalty, as shown in Figure 1b. The numerical penalty is calculated as the numerical difference between the correct and predicted labels; the larger the numerical difference, the larger the penalty. In the example in Figure 1b, a large numerical penalty would be applied to predictions such as "8" or "9" that have a numerical difference from the correct label, "0". In this way, this method can learn to be less likely to make predictions with large numerical differences. Experiments were conducted on MNIST, a dataset of handwritten digits from 0 to 9 in Arabic script as shown in Figure 2a and KANNADA-MNIST, a dataset of handwritten digits from 0 to 9 in Kannada script as shown in Figure 2b. The results show that the proposed method in this work shows smaller numerical error than the normal model.

2.2 A Classifier for Reducing Numerical Errors Using Ensemble Method

This work, published in 2021, proposed an ensemble model of CNNs that learns not only image features but also numerical information in handwritten numeric recognition [6]. The ensemble learning method used in this work is a method of combining multiple models to obtain higher accuracy as shown in Figure 3a, and has been shown to be effective in ensembles of CNNs [8]. This study focused on the differences in error patterns that occur between normal CNNs and CNNs that learn numerical information, as proposed in the study in Chapter 2.1. Normal CNNs and CNNs that learn numerical



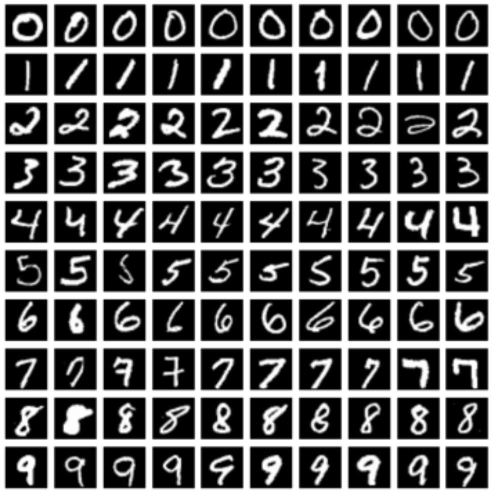
(a) Normal CNN



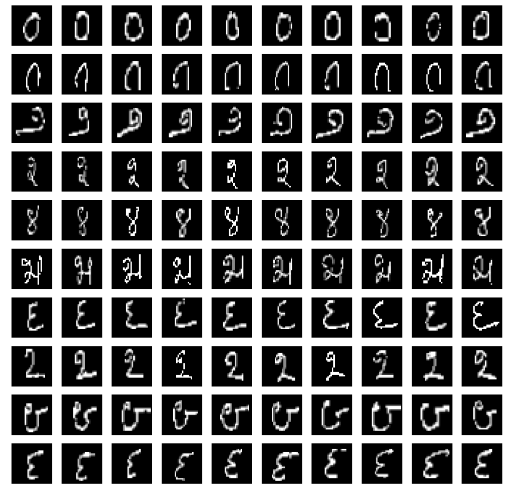
(b) CNN with Numerical Penalty

Figure 1: On the top is normal CNN. On the bottom is CNN with numerical penalty. When the image of "0" is input in the process of learning handwritten numeric characters from 0 to 9.

information differ in whether they impose a numerical penalty or not, as shown in Figure 1, and they differ in the features they learn from each other. They proposed an ensemble model as shown in Figure 3b, based on the idea that ensemble learning is effective in compensating for each other's misidentification. Experiments were conducted on the MNIST and KANNADA-MNIST datasets with an ensemble model of four CNNs. They used a standard CNN with three convolutional layers of 256; 256; 128 channels. The last convolutional layers are followed by two fully connected layers of size 328; 192. Since the dataset used for this was very small data, gray scale images with an image size of 28 x 28, a small-scale CNN was also used. In addition, the ensemble method was conducted by majority vote. As a result, the proposed ensemble model showed smaller numerical error than the normal ensemble model.

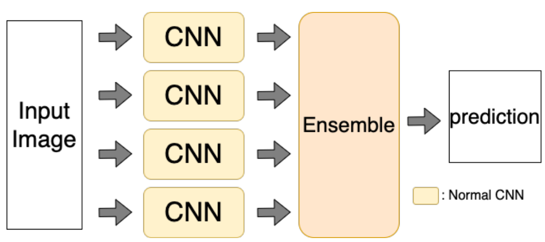


(a) MNIST

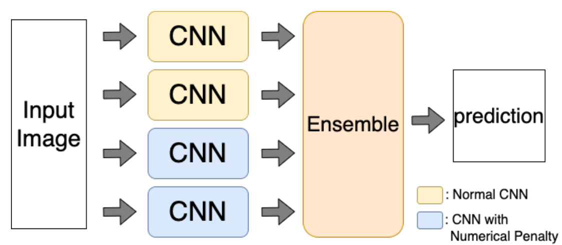


(b) KANNADA-MNIST

Figure 2: On the left are some of the MNIST images. Images with labels "0", "1", "2", "3", "4", "5", "6", "7", "8", and "9" from the top row. On the right are some of the Kannada-MNIST images. Images with labels "0", "1", "2", "3", "4", "5", "6", "7", "8", and "9" from the top row.



(a) Normal ensemble model consisting of normal CNNs



(b) Ensemble models of normal CNNs and CNNs with numerical penalties

Figure 3: On the left is a normal ensemble model with four CNNs. On the right is an ensemble model with two normal CNNs and two CNNs with numerical penalties.

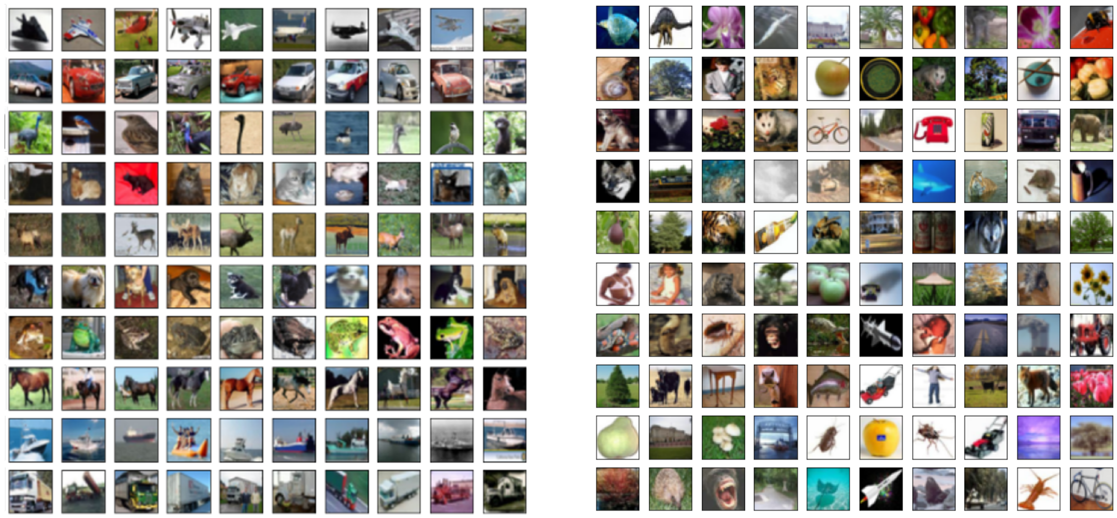
2.3 CIFAR-10 and CIFAR-100

We introduce CIFAR-10 and CIFAR-100 as the datasets used in this study.

CIFAR-10 and CIFAR-100 were published in 2009. As shown in Figure 4a, CIFAR-10 is a dataset of 10 different object color images: "airplane", "automobile", "bird", "cat", "deer", "dog", "frog", "horse", "ship", and "truck". Each image is assigned the corresponding correct label among the 10 types. CIFAR-10 consists of 50,000 training data and 10,000 test data.

CIFAR-100 is an image dataset as shown in Figure 4b, with 100 different object color images and correct labels. In addition to that, CIFAR-100 can also use the super class of 20 classifications instead of 100 classifications as the correct answer label by grouping 5 classifications each by switching the setting. Like CIFAR-10, CIFAR-100 consists of 50,000 training data and 10,000 test data.

CIFAR-10 and CIFAR-100 are among the most major image classification datasets and have been used for validation in various studies.



(a) CIFAR-10 (b) CIFAR-100

Figure 4: On the left are some of the CIFAR-10 images. Images with labels "airplane", "automobile", "bird", "cat", "deer", "dog", "frog", "horse", "ship", and "truck" from the top row. On the right are some of the CIFAR-100 images.

3 Proposed Method

This study aimed to reduce the number of errors beyond acceptable ranges in non-numeric data. In this study, we propose a CNN and its ensemble method that can learn meta-information even in images without numerical information. Our proposed single CNN model imposes an out-category penalty in addition to the penalty for misidentification that is also imposed on normal CNNs. Out-category penalty is a penalty that is categorized in advance with meta-information separate from the correct label, and imposed additionally for misidentification beyond that categorization. The proposed method then learns to reduce these penalties. As a result, we can expect to learn to reduce misidentification beyond categorization by meta-information. Figure 5 shows an example of image classification for "dog," "cat," "boar," and "bear. Prediction in Figure 5 is the output of the CNN and is in the form of a vector of 1 x the number of correct labels. And at this time, let is say that "boar" and "bear" are categorized as dangerous animals. In a normal CNN, when the image of "bear" is input, as shown in Figure 5a, three misidentifications other than the correct one, "bear," are regarded as errors and an identification penalty is applied. On the other hand, as shown in Figure 5b, the proposed method applies only an identification penalty to "boar," which is a misidentification within the dangerous animal category, regarded as an in-category error, while "dog" and "cat," which are outside the dangerous animal category, are regarded as out-of-category errors and an out-of-category penalty is applied in addition to an identification penalty. This is expected to reduce the number of dangerous animals misidentified as non-dangerous animals.

In the proposed CNN model, the loss function L, which calculates the penalty, is expressed as (1) using the hyperparameter α.

$$L = P_i + \alpha P_o \tag{1}$$

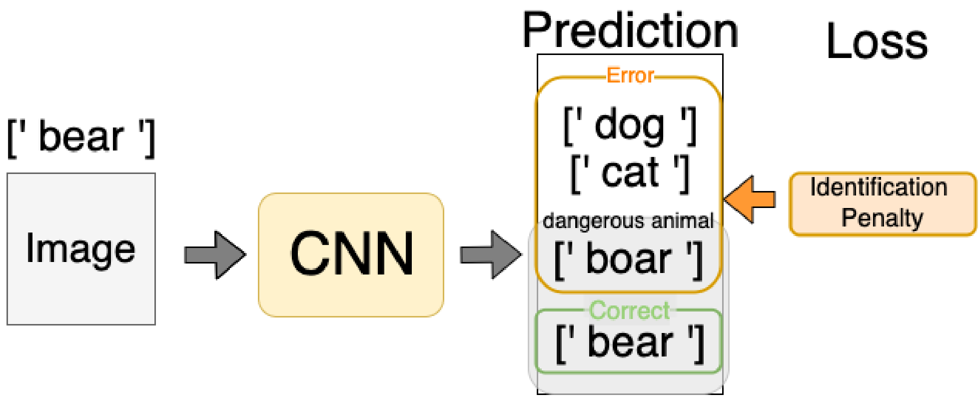
In (1), P_i is the penalty for misidentification, and P_o is the out-category penalty. To calculate P_i and P_o, we use binary cross-entropy (BCE) or categorical cross-entropy (CCE), which is used as a loss function for classification problems [9]. The formula for binary cross-entropy can be expressed as (2) using N input data x and one-hotted correct labels y.

$$BCE(x, y) = \; -\sum_{n=1}^{N}\{y'_n \log\{f(x_n)\} + (1 - y'_n)\log\{1 - f(x_n)\}\} \tag{2}$$

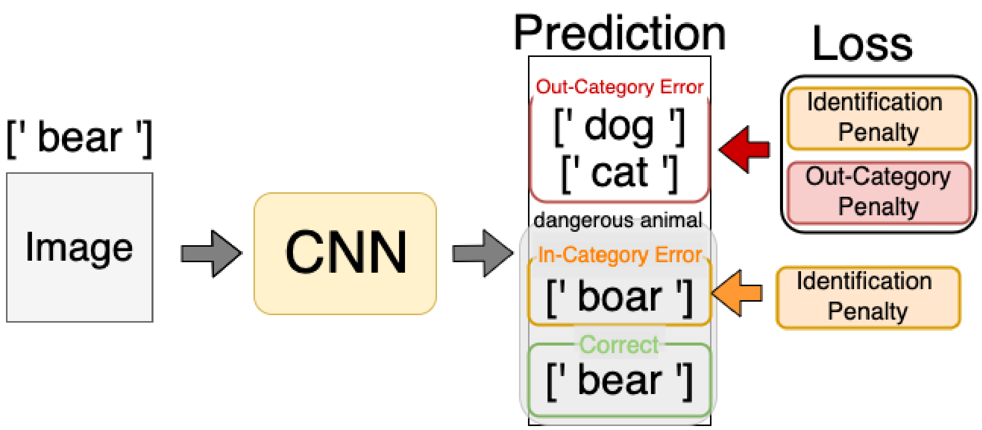
The formula for categorical cross-entropy can be expressed as (3) using N input data x with K correct label types and one-hotted correct labels y.

$$CCE(x,y) = \quad -\sum_{n=1}^{N}\sum_{k=1}^{K}\{y_{n,k}\log\{f(x_n)\} + (1-y_{n,k})\log\{1-f(x_n)\}\} \tag{3}$$

This method makes it possible to learn meta-information even on image datasets that do not have numerical information. As image datasets without numerical information, we consider CIFAR-10 and CIFAR-100 in this study.



(a) Normal CNN



(b) Proposed CNN

Figure 5: On the top is normal CNN. On the bottom is proposed CNN with out-category penalty.

3.1 Proposed Single CNN in CIFAR-10

When we apply the proposed method to CIFAR-10, we categorize it with the meta-information of "animal" and "vehicle" as shown in Figure 6. These two types of animals and vehicles are superclass labels set separately from the CIFAR-10 10-value classification labels. Our proposed single CNN model in CIFAR-10 is shown in Figure 7. In this model, apart from the correct label assigned to the image, we penalize misidentification beyond the categorization of animals and vehicles in the meta-information. In this case, misidentification beyond the categorization in the meta-information is considered an out-category error, and the penalty imposed on it is the out-category penalty. Then it learns to reduce the calculated penalty. As a result, the proposed model is expected to reduce the number of misidentifications beyond the categories of animals and vehicles.

Figure 6: CIFAR-10 categorized by the meta-information of vehicle and animal.

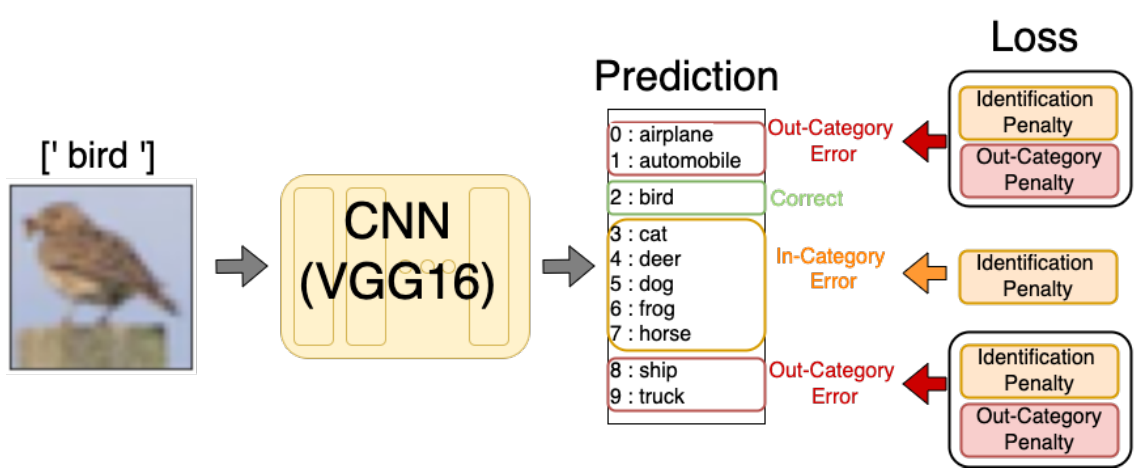


Figure 7: Our Proposed Single Model in CIFAR-10

Algorithm 1 Loss calculation in CIFAR-10

Input: **CorrectLabel**[10], **Predict**[10], **MetaCorrectLabel**[2], **MetaTable**[2][10], α
Output: Loss
1: **MetaPredict**[2] $\leftarrow$ **Predict**[10] $\cdot$ **MetaTable**[2][10]
2: IdentificationPenalty $\leftarrow$ CategoricalCrossEntropy(**CorrectLabel**[10], **Predict**[10])
3: Out-CategoryPenalty $\leftarrow$ BinaryCrossEntropy(**MetaCorrectLabel**[2], **MetaPredict**[2])
4: Loss $\leftarrow$ IdentificationPenalty + αOut-CategoryPenalty
5: **return** Loss

The loss function is calculated as in Algorithm 1. Algorithm 1 requires a one-hot vectorized correct answer label **CorrectLabel**[10] with 10 elements, a prediction result **Predict**[10], a categorized binary correct answer label with additional meta information **MetaCorrectLabel**[2], and a 2×10 array **MetaTable**[2][10] linking the correct and meta-correct labels. α is a hyperparameter that adjusts the impact of the out-category penalty. The explanation of Algorithm 1 is as follows.

1. The first line calculates **MetaPredict**[2], the prediction result in the meta category, from the matrix product of the **Predict**[10] and the **MetaTable**[2][10].

2. In the second line, we calculate the identification penalty by computing the cross entropy between the **CorrectLabel**[10] and the **Predict**[10].

3. In the third line, we calculate the out-category penalty by computing the cross entropy between the **MetaCorrectLabel**[2] and the **MetaPredict**[2].

4. In the fourth line, we calculate the loss as the sum of the identification penalty and the out-category penalty. In this case, we adjust the influence of the out-category penalty by multiplying it by α to ensure good learning.

5. The fifth line returns Loss.

3.2 Proposed Method in CIFAR-100

When we apply the proposed method to CIFAR-100, we categorize it with 20 types of meta-information: ”aquatic mammals”, ”fish”, ”flowers”, ”food containers”, ”fruit and vegetables”, ”household electrical devices”, ”household furniture”, ”insects”, ”large carnivores”, ”large man-made outdoor things”, ”large natural outdoor scenes”, ”large omnivores and herbivores”, ”medium-sized mammals”, ”non-insect invertebrates”, ”people”, ”reptiles”, ”small mammals”, ”trees”, ”vehicles 1”, and ”vehicles 2” as shown in Figure 8. These 20 types are superclass labels that are set separately from the 100-value classification labels of CIFAR-100. Our proposed single CNN model in CIFAR-10 is shown in Figure 9. In this model, apart from the correct label assigned to the image, we apply an out-category penalty for out-category errors that exceed the categorization in the 20 types of meta-information. Then it learns to reduce the calculated penalty. As a result, the proposed model can be expected to reduce the number of misidentifications beyond the 20 categories.

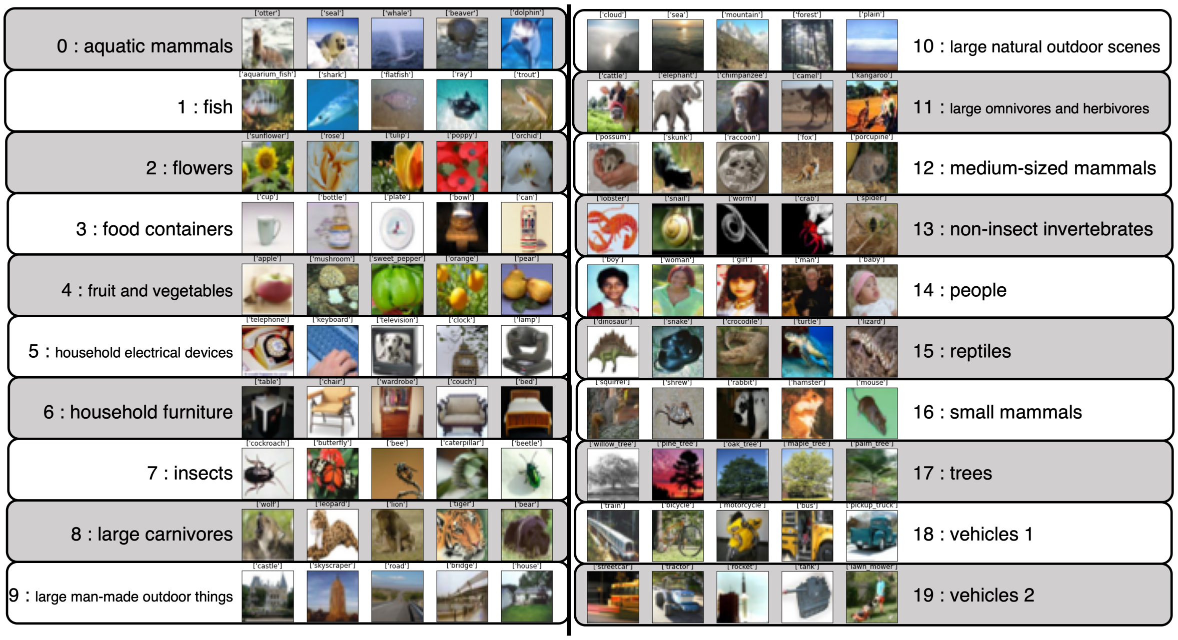


Figure 8: CIFAR-100 with 20 different categorizations by meta-information.

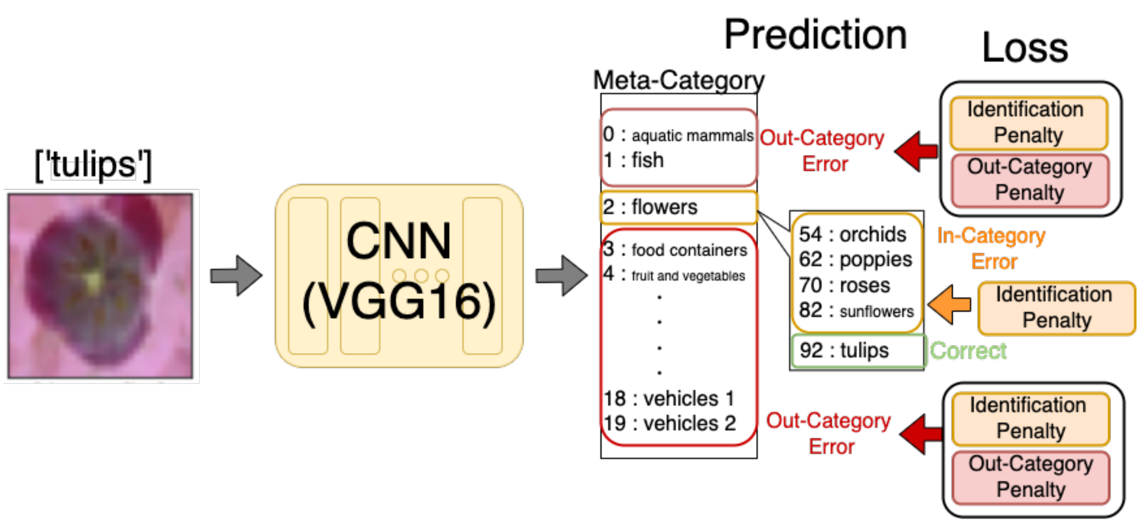


Figure 9: Our Proposed Single Model in CIFAR-100

Algorithm 2 Loss calculation in CIFAR-100

Input: **CorrectLabel**[100], **Predict**[100], **MetaCorrectLabel**[20], **MetaTable**[20][100], α
Output: Loss

```
1: MetaPredict[20] ← Predict[100] · MetaTable[20][100]
2: IdentificationPenalty ← CategoricalCrossEntropy(CorrectLabel[100], Predict[100])
3: Out-CategoryPenalty ← CategoricalCrossEntropy(MetaCorrectLabel[20], MetaPredict[20])
4: Loss ← IdentificationPenalty + αOut-CategoryPenalty
5: return Loss
```

The loss function is calculated as in Algorithm 2. Algorithm 2 requires a one-hot vectorized correct answer label **CorrectLabel**[100] with 100 elements, a prediction result **Predict**[100], a categorical correct answer label **MetaCorrectLabel**[20] with additional 20-valued meta information, and a 20 x 100 array **MetaTable**[20][100] linking the labels and the meta-correct labels. α is a hyperparameter that adjusts the impact of the out-category penalty. The explanation of Algorithm 2 is as follows.

1. The first line calculates **MetaPredict**[20], the prediction result in the meta category, from the matrix product of the **Predict**[100] and the **MetaTable**[20][100].

2. In the second line, we calculate the identification penalty by computing the cross entropy between the **CorrectLabel**[100] and the **Predict**[100].

3. In the third line, we calculate the out-category penalty by computing the cross entropy between the **MetaCorrectLabel**[20] and the **MetaPredict**[20].

4. In the fourth line, we calculate the loss as the sum of the identification penalty and the out-category penalty. In this case, we adjust the influence of the out-category penalty by multiplying it by α to ensure good learning.

5. The fifth line returns Loss.

3.3 Proposed Ensemble Model

We also proposed an ensemble model, as shown in Figure 10. The model is an ensemble model that combines two normal CNNs and two CNNs with out-category penalties as shown in Fig. 10. A normal CNN learns so that the overall identification error is small, while a CNN with an out-category penalty learns so that the out-category error is also small. Since these two types of CNNs learn differently and uniquely, it is expected that ensemble learning, which compensates for each other's misidentification, is effective.

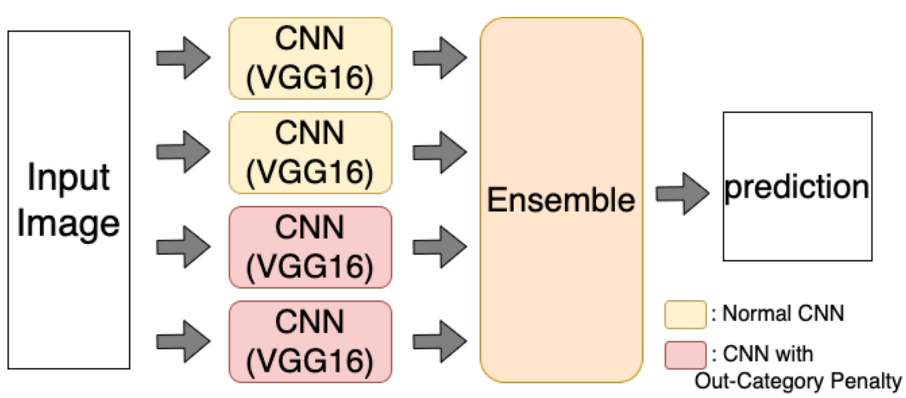


Figure 10: Our Proposed Ensemble Model

4 Experiments

In this study, we use the CIFAR-10 dataset and the CIFAR-100 dataset to compare and verify the results in the normal CNN and the proposed model. Five patterns of hyperparameter α in the proposed model (0.1, 0.3, 0.5, 0.7, and 0.9) are performed to verify its influence. The structure of the CNN used is the VGG16 [10] model as shown in Figure 10. The goal of this study is to reduce the number of unacceptable errors in non-numeric data, and the purpose of the experiment is to confirm that out-category penalties reduce the number of out-category errors. Therefore, although there were various models for CNNs, such as ResNet [11], EfficientNet [12], and Alexnet [13], VGG16 was used based on the ease of performance comparison and the moderate expressivity that would make the ensemble valid. Categorical cross-entropy was used for the identification penalty and binary cross-entropy was used for the out-category penalty in CIFAR-10 and categorical cross-entropy was used in CIFAR-100. Adam [14] was used as the optimization function. The ensemble is based on a simple majority voting method. Under these conditions, we compared each model in terms of accuracy and in-category and out-category error rates. The experiment was also conducted five times with different seeds in each model.

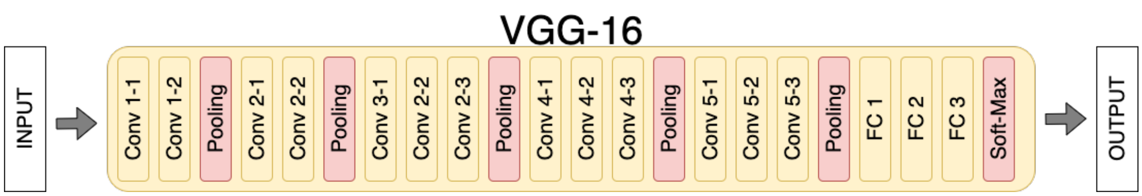


Figure 11: the architecture of VGG16 model

5 Results

In this section, we present the experimental results of this study. The experimental results are divided into four patterns: Results of single models in CIFAR-10, results of ensemble models in CIFAR-10, results of single models in CIFAR-100, and results of ensemble models in CIFAR-100. For each pattern, the results are shown for accuracy, in-category error, out-of-category error, and out-of-category error rate. In-category error is the percentage of misidentifications within a category, and out-of-category error is the percentage of misidentifications outside of a category, and the sum of these is the total percentage of misidentifications. The out-of-category error rate is the percentage of out-of-category error in the total misidentification. Therefore, the out-category error rate provides an indicator that allows us to ignore the influence of the size of the total error rate.

5.1 Results of Single Models in CIFAR-10

The results of the accuracy of the single model in CIFAR-10 are shown in Table 1. From Table 1, we can see that the proposed method shows higher accuracy than normal CNN in all cases with hyperparameters α = 0.1, 0.3, 0.5, 0.7, and 0.9. This shows that the addition of the out-category penalty has made it possible to identify images that could not be identified with the normal penalty for misidentification alone.

The results of in-category and out-category errors for single models in CIFAR-10 are shown in Table 2. Table 2 shows that the in-category error is larger for the proposed method than for the normal CNN in all cases except when $\alpha = 0.1$. This may be because the impact of the out-category penalty made the penalty for in-category errors relatively smaller and the tolerance for in-category errors larger. In contrast, the out-category error is smaller than that of a normal CNN in all cases where $\alpha = 0.1$, 0.3, 0.5, 0.7, and 0.9. Furthermore, we can confirm that the proposed method has smaller out-category error as α increases. From these results, we can see that the addition of the out-category penalty makes the out-of-category errors smaller, and we can also confirm that the impact of the penalty can be controlled by α.

The results of the out-category error rate for single models in CIFAR-10 are shown in Table 3. From Table 3, we can see that the proposed method shows a smaller out-category error rate than the normal CNN in all cases with hyperparameters $\alpha = 0.1$, 0.3, 0.5, 0.7, and 0.9. Furthermore, we can confirm that the out-category error rate of the proposed method becomes smaller as α increases. From these results, we can see that the addition of the out-category penalty makes the out-of-category error rate smaller, and we can also confirm that the impact of the penalty can be controlled by α.

Table 1: Results of the mean and standard deviation of the accuracy for the single models over five trials in CIFAR-10.

Method	**Accuracy(%)**
Baseline	$84.80_{\pm 0.624}$
Proposed Method (+Out-Category Penalty($\alpha = 0.1$))	$\mathbf{85.04}_{\pm 0.617}$
Proposed Method (+Out-Category Penalty($\alpha = 0.3$))	$\mathbf{84.83}_{\pm 0.774}$
Proposed Method (+Out-Category Penalty($\alpha = 0.5$))	$\mathbf{84.88}_{\pm 0.599}$
Proposed Method (+Out-Category Penalty($\alpha = 0.7$))	$\mathbf{84.83}_{\pm 0.520}$
Proposed Method (+Out-Category Penalty($\alpha = 0.9$))	$\mathbf{84.94}_{\pm 0.291}$

Table 2: Results of the mean and standard deviation of in-category and out-of-category error for single models over five trials in CIFAR-10.

Method	**In-Category Error(%)**	**Out-Category Error(%)**
Baseline	$12.47_{\pm 0.556}$	$2.72_{\pm 0.214}$
Proposed Method (+Out-Category Penalty($\alpha = 0.1$))	$\mathbf{12.29}_{\pm 0.484}$	$\mathbf{2.66}_{\pm 0.370}$
Proposed Method (+Out-Category Penalty($\alpha = 0.3$))	$12.59_{\pm 0.830}$	$\mathbf{2.57}_{\pm 0.296}$
Proposed Method (+Out-Category Penalty($\alpha = 0.5$))	$12.66_{\pm 0.624}$	$\mathbf{2.45}_{\pm 0.151}$
Proposed Method (+Out-Category Penalty($\alpha = 0.7$))	$12.72_{\pm 0.546}$	$\mathbf{2.44}_{\pm 0.137}$
Proposed Method (+Out-Category Penalty($\alpha = 0.9$))	$12.65_{\pm 0.493}$	$\mathbf{2.40}_{\pm 0.241}$

Table 3: Results of the mean and standard deviation of the out-category error rate for the single models over eight trials in CIFAR-10.

Method	**Out-Category Error Rate(%)**
Baseline	$17.92_{\pm 1.255}$
Proposed Method (+Out-Category Penalty($\alpha = 0.1$))	$\mathbf{17.75}_{\pm 2.108}$
Proposed Method (+Out-Category Penalty($\alpha = 0.3$))	$\mathbf{16.97}_{\pm 1.727}$
Proposed Method (+Out-Category Penalty($\alpha = 0.5$))	$\mathbf{16.24}_{\pm 0.828}$
Proposed Method (+Out-Category Penalty($\alpha = 0.7$))	$\mathbf{16.13}_{\pm 1.099}$
Proposed Method (+Out-Category Penalty($\alpha = 0.9$))	$\mathbf{15.96}_{\pm 1.847}$

5.2 Results of Ensemble Models in CIFAR-10

The results of the accuracy of the ensemble model in CIFAR-10 are shown in Table 4. From Table 4, we can see that the proposed method shows higher accuracy than the usual CNN ensemble model in all cases with hyperparameters $\alpha = 0.1$, 0.3, 0.5, 0.7, and 0.9. This shows that normal CNNs and CNNs with out-category penalties are able to compensate for each other's misidentification.

The results of in-category and out-of-category errors for the ensemble model in CIFAR-10 are shown in Table 5. Table 5 shows that the out-category error is smaller than the ensemble model of normal CNN in all cases of $\alpha = 0.1$, 0.3, 0.5, 0.7, 0.9. Furthermore, we can confirm that the proposed method has smaller out-category error as α increases. From these results, we can see that even in the ensemble model, the addition of the out-category penalty makes the out-category error smaller, and we can also confirm that its impact can be controlled by α. On the other hand, the in-category errors are smaller than the normal CNN ensemble model at $\alpha = 0.1$, 0.3, and 0.7. The proposed single model with $\alpha = 0.3$, 0.7 showed larger in-category errors than the normal single CNN. Nevertheless, the fact that the ensemble models with $\alpha = 0.3$ and 0.7 showed smaller in-category errors than the normal ensemble models suggests that the ensemble of the proposed method is able to compensate for the misidentification of each other's models more than the normal ensemble.

The results of the out-category error rate of the ensemble model in CIFAR-10 are shown in Table 6. From Table 6, we can see that the proposed method shows a smaller out-category error rate than the normal CNN ensemble model in all cases with hyperparameters $\alpha = 0.1$, 0.3, 0.5, 0.7, and 0.9. Furthermore, we can confirm that the out-category error rate of the proposed method becomes smaller as α increases. From these results, it can be seen that the out-category error rate can be reduced by adding an out-category penalty in the ensemble model, and the impact can be controlled by α.

Table 4: Results of the mean and standard deviation of the accuracy for the ensemble models over five trials in CIFAR-10.

Method	**Accuracy(%)**
Baseline (+Ensemble)	$88.13_{\pm 0.101}$
Proposed Method (+Out-Category Penalty($\alpha = 0.1$) +Ensemble)	$\mathbf{88.18}_{\pm 0.131}$
Proposed Method (+Out-Category Penalty($\alpha = 0.3$) +Ensemble)	$\mathbf{88.25}_{\pm 0.261}$
Proposed Method (+Out-Category Penalty($\alpha = 0.5$) +Ensemble)	$\mathbf{88.16}_{\pm 0.208}$
Proposed Method (+Out-Category Penalty($\alpha = 0.7$) +Ensemble)	$\mathbf{88.25}_{\pm 0.164}$
Proposed Method (+Out-Category Penalty($\alpha = 0.9$) +Ensemble)	$\mathbf{88.24}_{\pm 0.125}$

Table 5: Results of the mean and standard deviation of in-category and out-of-category error for ensemble models over five trials in CIFAR-10.

Method	**In-Category Error(%)**	**Out-Category Error(%)**
Baseline (+Ensemble)	$9.98_{\pm 0.084}$	$1.88_{\pm 0.027}$
Proposed Method (+Out-Category Penalty($\alpha = 0.1$) +Ensemble)	$\mathbf{9.98}_{\pm 0.103}$	$\mathbf{1.83}_{\pm 0.058}$
Proposed Method (+Out-Category Penalty($\alpha = 0.3$) +Ensemble)	$\mathbf{9.95}_{\pm 0.274}$	$\mathbf{1.79}_{\pm 0.077}$
Proposed Method (+Out-Category Penalty($\alpha = 0.5$) +Ensemble)	$10.05_{\pm 0.279}$	$\mathbf{1.78}_{\pm 0.100}$
Proposed Method (+Out-Category Penalty($\alpha = 0.7$) +Ensemble)	$\mathbf{9.98}_{\pm 0.139}$	$\mathbf{1.75}_{\pm 0.040}$
Proposed Method (+Out-Category Penalty($\alpha = 0.9$) +Ensemble)	$10.11_{\pm 0.151}$	$\mathbf{1.64}_{\pm 0.102}$

Table 6: Results of the mean and standard deviation of the out-category error rate for the ensemble models over five trials in CIFAR-10.

Method	Out-Category Error Rate(%)
Baseline (+Ensemble)	$15.87_{\pm 0.168}$
Proposed Method (+Out-Category Penalty($\alpha = 0.1$) +Ensemble)	$\mathbf{15.50}_{\pm 0.401}$
Proposed Method (+Out-Category Penalty($\alpha = 0.3$) +Ensemble)	$\mathbf{15.27}_{\pm 0.744}$
Proposed Method (+Out-Category Penalty($\alpha = 0.5$) +Ensemble)	$\mathbf{15.08}_{\pm 1.025}$
Proposed Method (+Out-Category Penalty($\alpha = 0.7$) +Ensemble)	$\mathbf{14.93}_{\pm 0.248}$
Proposed Method (+Out-Category Penalty($\alpha = 0.9$) +Ensemble)	$\mathbf{13.97}_{\pm 0.863}$

5.3 Results of Single Models in CIFAR-100

The results of the accuracy of the single model in CIFAR-100 are shown in Table 7. From Table 7, we can see that the proposed method shows higher accuracy than normal CNN in the cases of hyperparameter $\alpha = 0.1$, 0.3, and 0.5. This shows that the addition of the out-category penalty has made it possible to identify images that could not be identified with the normal penalty for misidentification alone. On the other hand, the cases of $\alpha = 0.7$ and 0.9 show lower accuracy than the normal CNN. This was probably due to the fact that the out-category penalty was too large. While CIFAR-10 imposed an out-category penalty for misidentifying a classification in only two categories, animals and vehicles, CIFAR-100 imposes an out-category penalty for misidentifying a classification in as many as 20 categories, so the out-category penalty is likely to be large. As a result, we consider that the proposed method shows lower accuracy than the normal CNN when we set a large α such as 0.7, 0.9.

The results of in-category and out-category errors for a single model in CIFAR-100 are shown in Table 8. From Table 8, we can see that the in-category error of the proposed method is larger than that of the normal CNN in all cases of $\alpha = 0.1$, 0.3, 0.5, 0.7, 0.9. This may be because the impact of the out-category penalty made the penalty for in-category errors relatively smaller and the tolerance for in-category errors larger. In contrast, the out-category error is smaller than that of a normal CNN in all cases where $\alpha = 0.1$, 0.3, 0.5, 0.7, and 0.9. In addition, unlike the case of CIFAR-10, the proposed method in CIFAR-100 does not confirm that the out-category error becomes smaller as α increases. This is thought to be because when α was increased, the out-of-category penalty became too large and the proposed model could not be trained properly.

The results of the out-category error rate for a single model in CIFAR-100 are shown in Table 9. From Table 9, we can see that the proposed method shows a smaller out-category error rate than the normal CNN in all cases with hyperparameters $\alpha = 0.1$, 0.3, 0.5, 0.7, and 0.9. Furthermore, we can confirm that the out-category error rate of the proposed method becomes smaller as α increases. From this, it can be confirmed that the influence of the out-category penalty can be controlled by α for the out-category error rate, ignoring the influence of accuracy.

Table 7: Results of the mean and standard deviation of the accuracy for the single models over eight trials in CIFAR-100.

Method	**Accuracy(%)**
Baseline	$45.45_{\pm 3.769}$
Proposed Method (+Out-Category Penalty($\alpha = 0.1$))	$\mathbf{46.66}_{\pm 0.926}$
Proposed Method (+Out-Category Penalty($\alpha = 0.3$))	$\mathbf{46.58}_{\pm 1.890}$
Proposed Method (+Out-Category Penalty($\alpha = 0.5$))	$\mathbf{45.67}_{\pm 3.260}$
Proposed Method (+Out-Category Penalty($\alpha = 0.7$))	$45.42_{\pm 1.858}$
Proposed Method (+Out-Category Penalty($\alpha = 0.9$))	$45.07_{\pm 2.786}$

Table 8: Results of the mean and standard deviation of in-category and out-of-category error for single models over five trials in CIFAR-100.

Method	**In-Category Error(%)**	**Out-Category Error(%)**
Baseline	$\mathbf{14.12}_{\pm 0.217}$	$40.41_{\pm 3.932}$
Proposed Method (+Out-Category Penalty($\alpha = 0.1$))	$14.22_{\pm 0.289}$	$\mathbf{39.10}_{\pm 0.914}$
Proposed Method (+Out-Category Penalty($\alpha = 0.3$))	$15.12_{\pm 0.172}$	$\mathbf{38.29}_{\pm 1.980}$
Proposed Method (+Out-Category Penalty($\alpha = 0.5$))	$15.50_{\pm 0.389}$	$\mathbf{38.82}_{\pm 3.512}$
Proposed Method (+Out-Category Penalty($\alpha = 0.7$))	$15.76_{\pm 0.413}$	$\mathbf{38.68}_{\pm 2.215}$
Proposed Method (+Out-Category Penalty($\alpha = 0.9$))	$16.25_{\pm 0.473}$	$\mathbf{38.67}_{\pm 3.152}$

Table 9: Results of the mean and standard deviation of the out-category error rate for the single models over five trials in CIFAR-100.

Method	**Out-Category Error Rate(%)**
Baseline	$73.98_{\pm 2.032}$
Proposed Method (+Out-Category Penalty($\alpha = 0.1$))	$\mathbf{73.31}_{\pm 0.631}$
Proposed Method (+Out-Category Penalty($\alpha = 0.3$))	$\mathbf{71.65}_{\pm 1.186}$
Proposed Method (+Out-Category Penalty($\alpha = 0.5$))	$\mathbf{71.36}_{\pm 2.118}$
Proposed Method (+Out-Category Penalty($\alpha = 0.7$))	$\mathbf{71.15}_{\pm 1.254}$
Proposed Method (+Out-Category Penalty($\alpha = 0.9$))	$\mathbf{70.32}_{\pm 2.090}$

5.4 Results of Ensemble Models in CIFAR-100

The results of the accuracy of the ensemble model in CIFAR-100 are shown in Table 10. From Table 10, we can see that the proposed method shows higher accuracy than the usual CNN ensemble model in all cases with hyperparameters $\alpha = 0.1$, 0.3, 0.5, 0.7, and 0.9. In the case of $\alpha = 0.7$ and 0.9, where the single model showed lower accuracy than the normal CNN, the proposed ensemble model showed higher accuracy than the normal ensemble model. This shows that normal CNNs and CNNs with out-category penalties are able to compensate each other's misidentification more effectively than normal ensemble methods.

The results of in-category and out-category errors for the ensemble model in CIFAR-100 are shown in Table 11. Table 11 shows that the in-category error is larger than that of the normal CNN ensemble model in all cases with $\alpha = 0.1$, 0.3, 0.5, 0.7, 0.9. This is probably because CIFAR-100, unlike CIFAR-10, considers as many as 20 categories, which is too complex, and the out-of-category penalty was not effective in reducing the in-category errors. Table 11 shows that the out-category error is smaller than the ensemble model of normal CNN in all cases of $\alpha = 0.1$, 0.3, 0.5, 0.7, 0.9. In addition, unlike the case of CIFAR-10, the proposed method in CIFAR-100 does not confirm that the out-category error becomes smaller as α increases. As with the single model, this is thought to be because when α was increased, the out-of-category penalty became too large and the proposed model could not be trained properly.

The results of the out-category error rate of the ensemble model in CIFAR-100 are shown in Table 12. From Table 12, we can see that the proposed method shows a smaller out-category error rate than the normal CNN ensemble model in all cases with hyperparameters $\alpha = 0.1$, 0.3, 0.5, 0.7, and 0.9. Furthermore, we can confirm that the out-category error rate of the proposed method becomes smaller as α increases. From these results, it can be seen that the out-category error rate can be reduced by adding an out-category penalty in the ensemble model, and the impact can be controlled by α.

Table 10: Results of the mean and standard deviation of the accuracy for the ensemble models over five trials in CIFAR-100.

Method	**Accuracy(%)**
Baseline (+Ensemble)	$51.30_{\pm 1.332}$
Proposed Method (+Out-Category Penalty($\alpha = 0.1$) +Ensemble)	$\mathbf{52.08}_{\pm 0.652}$
Proposed Method (+Out-Category Penalty($\alpha = 0.3$) +Ensemble)	$\mathbf{53.14}_{\pm 0.581}$
Proposed Method (+Out-Category Penalty($\alpha = 0.5$) +Ensemble)	$\mathbf{53.22}_{\pm 0.565}$
Proposed Method (+Out-Category Penalty($\alpha = 0.7$) +Ensemble)	$\mathbf{52.81}_{\pm 0.673}$
Proposed Method (+Out-Category Penalty($\alpha = 0.9$) +Ensemble)	$\mathbf{52.29}_{\pm 0.199}$

Table 11: Results of the mean and standard deviation of in-category and out-of-category error for ensemble models over five trials in CIFAR-100.

Method	**In-Category Error(%)**	**Out-Category Error(%)**
Baseline (+Ensemble)	$\mathbf{13.04}_{\pm 0.172}$	$35.65_{\pm 1.478}$
Proposed Method (+Out-Category Penalty($\alpha = 0.1$) +Ensemble)	$13.31_{\pm 0.216}$	$\mathbf{34.60}_{\pm 0.492}$
Proposed Method (+Out-Category Penalty($\alpha = 0.3$) +Ensemble)	$13.88_{\pm 0.119}$	$\mathbf{32.97}_{\pm 0.692}$
Proposed Method (+Out-Category Penalty($\alpha = 0.5$) +Ensemble)	$13.93_{\pm 0.303}$	$\mathbf{32.84}_{\pm 0.703}$
Proposed Method (+Out-Category Penalty($\alpha = 0.7$) +Ensemble)	$14.13_{\pm 0.171}$	$\mathbf{33.05}_{\pm 0.609}$
Proposed Method (+Out-Category Penalty($\alpha = 0.9$) +Ensemble)	$14.57_{\pm 0.181}$	$\mathbf{33.13}_{\pm 0.227}$

Table 12: Results of the mean and standard deviation of the out-category error rate for the ensemble models over five trials in CIFAR-100.

Method	**Out-Category Error Rate(%)**
Baseline (+Ensemble)	$73.19_{\pm 1.074}$
Proposed Method (+Out-Category Penalty($\alpha = 0.1$) +Ensemble)	$\mathbf{72.21}_{\pm 0.259}$
Proposed Method (+Out-Category Penalty($\alpha = 0.3$) +Ensemble)	$\mathbf{70.35}_{\pm 0.607}$
Proposed Method (+Out-Category Penalty($\alpha = 0.5$) +Ensemble)	$\mathbf{70.21}_{\pm 0.814}$
Proposed Method (+Out-Category Penalty($\alpha = 0.7$) +Ensemble)	$\mathbf{70.04}_{\pm 0.402}$
Proposed Method (+Out-Category Penalty($\alpha = 0.9$) +Ensemble)	$\mathbf{69.45}_{\pm 0.364}$

6 Conclusion

This study aimed to reduce the number of errors beyond acceptable ranges in non-numeric data. In this study, we proposed a CNN and its ensemble method that can learn meta-information even in images without numerical information by using out-category penalty. Experiments were conducted on the CIFAR-10 and CIFAR-100 datasets.

As a result, the proposed method shows high accuracy and small out-category error in both single and ensemble models in CIFAR-10. On the other hand, the proposed method showed a larger in-category error. This may be because the impact of the out-category penalty made the penalty for in-category errors relatively smaller and the tolerance for in-category errors larger. Since the decrease in out-category errors is larger than the increase in in-category errors, and the overall accuracy is improved, the proposed method is a better model than the normal CNN model for practical use. Also, since the proposed model is intended to be used in situations where it is acceptable to tolerate some degree of in-category errors, we do not see any problem with it.

Categorization by meta-information is necessary to use the proposed method. Some existing datasets, such as CIFAR-100 and EMNIST [15], also have pre-categorization with meta-information, but if you want to categorize with your own meta-information, you need to be prepared. The proposed method may not show a significant improvement in terms of accuracy, commensurate with the preparation of categorization with meta-information, but there is sufficient improvement in terms of out-category errors. Therefore, in situations where there are out-category errors that you do not want to allow, it is more effective to use the proposed method, even if you are prepared to categorize them with meta-information.

In this study, the hyperparameter α was set to control the impact of the out-category penalty. In the case of simple classifications with binary meta-categories, such as CIFAR-10, there were fewer occasions to impose an out-category penalty, and the larger α was in the range 0.1-0.9, the smaller the out-category error. On the other hand, in CIFAR-100, when the meta-category was a relatively difficult 20-valued classification, the out-category penalty was imposed more often, and when α was 0.3 for the single model and 0.5 for the ensemble model, the out-category error was the smallest. From the above, it can be assumed that the appropriate value of α needs to be set according to the complexity of the meta-category to be prepared. Although it may vary depending on the complexity of the dataset itself, the conclusion of this study is that $\alpha = 0.9$ or higher for binary meta-categories and $\alpha = 0.3$-0.5 for a number of 20-value meta-categories are considered to be the guidelines for optimal training of the proposed method.

References

[1] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[2] K. Fukushima. Neocognitron : A new algorithm for pattern recognition tolerant of deformations and shifts in position. *Pattern Recognition*, 15(6):455–469, 1982.

[3] Zachary C. Lipton, John Berkowitz, and Charles Elkan. A critical review of recurrent neural networks for sequence learning, 2015.

[4] Yuta Suzuki, Toshiki Hatano, Toi Tsuneda, and Satoshi Yamane. An image classification model that learns mnist image features and numerical information. In *2020 Eighth International Symposium on Computing and Networking Workshops (CANDARW)*, pages 483–485, 2020.

[5] Yuta Suzuki, Toshiki Hatano, Toi Tsuneda, Daiki Kuyoshi, and Satoshi Yamane. An image classification model that learns image features and numerical information. *International Journal of Networking and Computing*, 11(2):426–437, 2021.

[6] Yuta Suzuki, Toi Tsuneda, Daiki Kuyoshi, and Satoshi Yamane. A classifier for reducing numerical errors using ensemble method. In *IEEE 10th Global Conference on Consumer Electronics (GCCE)*, 2021.

[7] Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, 2009.

[8] Abdul Wasay and Stratos Idreos. More or less: When and how to build convolutional neural network ensembles. In *International Conference on Learning Representations*, 2021.

[9] Shie Mannor, Dori Peleg, and Reuven Rubinstein. The cross entropy method for classification. pages 561–568, 01 2005.

[10] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition, 2015.

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015.

[12] Mingxing Tan and Quoc V. Le. Efficientnet: Rethinking model scaling for convolutional neural networks, 2020.

[13] Geoffrey E. Hinton Alex Krizhevsky, Ilya Sutskever. Imagenet classification with deep convolutional neural networks. Technical report, 2012.

[14] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017.

[15] Gregory Cohen, Saeed Afshar, Jonathan Tapson, and André van Schaik. Emnist: an extension of mnist to handwritten letters, 2017.