

Article

A TEDE Algorithm Studies the Effect of Dataset Grouping on Supervised Learning Accuracy

Xufei Wang^{1,2,*} , Penghui Wang¹, Jeongyoung Song^{3,*}, Taotao Hao¹ and Xinlu Duan¹¹ School of Mechanical Engineering, Shaanxi University of Technology, Hanzhong 723000, China² Key Laboratory of Industrial Automation, Shaanxi University of Technology, Hanzhong 723000, China³ Department of Computer Engineering, Pai Chai University, Daejeon 35345, Republic of Korea

* Correspondence: wxf@snut.edu.cn (X.W.); jysong@pcu.ac.kr (J.S.)

Abstract: Datasets are the basis for research on deep learning methods in computer vision. The impact of the percentage of training sets in a dataset on the performance of neural network models needs to be further explored. In this paper, a twice equal difference enumeration (TEDE) algorithm is proposed to investigate the effect of different training set percentages in the dataset on the performance of the network model, and the optimal training set percentage is determined. By selecting the Pascal VOC dataset and dividing it into six different datasets from largest to smallest, and then dividing each dataset into the datasets to be analyzed according to five different training set percentages, the YOLOv5 convolutional neural network is used to train and test the 30 datasets to determine the optimal neural network model corresponding to the training set percentages. Finally, tests were conducted using the Udacity Self-Driving dataset with a self-made Tire Tread Defects (TTD) dataset. The results show that the network model performance is superior when the training set accounts for between 85% and 90% of the overall dataset. The results of dataset partitioning obtained by the TEDE algorithm can provide a reference for deep learning research.

Keywords: dataset; proportional; supervised learning; accuracy



Citation: Wang, X.; Wang, P.; Song, J.; Hao, T.; Duan, X. A TEDE Algorithm Studies the Effect of Dataset Grouping on Supervised Learning Accuracy. *Electronics* **2023**, *12*, 2546. <https://doi.org/10.3390/electronics12112546>

Academic Editor: Tomasz Trzcinski

Received: 17 April 2023

Revised: 25 May 2023

Accepted: 2 June 2023

Published: 5 June 2023



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

1. Introduction

Deep learning methods are widely used in various fields of research, such as artificial intelligence, machine vision, and speech recognition. The performance of deep learning methods is mainly influenced by three factors: computing power, algorithms, and data. With arithmetic power and algorithms largely established, data play an important role in the process of implementing deep learning tasks in general [1].

Deep learning is only one type of machine learning classification algorithm, and each classification algorithm has its advantages and disadvantages [2]. Taking the medical field as an example, Done Stojanov et al. used a logistic regression algorithm for risk prediction of cardiovascular diseases and achieved the best distinction between heart failure and chronic ischemic heart disease outcomes [3]. Mostafa Langarizadeh et al. used a plain Bayesian network for disease prediction and found that disease prediction based on the plain Bayesian network had the best performance in most diseases [4]. Wenchao Xing et al. proposed an improved KNN classification algorithm based on clustering denoising and density clipping to classify healthcare big data, and the improved KNN algorithm changed the deficiencies of the traditional KNN algorithm in handling large data sets [5]. G Battineni et al. applied SVM to dementia prediction and found that low gamma values (1.0×10^{-4}) and high regularization values ($C = 100$) were better for predicting dementia [6]. Christian Schüldt et al. used the SVM classification algorithm for human action recognition, and their results proved the superiority of the SVM classification algorithm in recognizing complex motion patterns [7].

Usually, deep learning algorithms come in the form of convolutional neural network structures, and data are an important foundation for convolutional neural networks to

form network models with practical analysis capabilities. In machine vision-based object detection tasks, scholars have investigated various aspects of convolutional neural networks and data. Sangwon Kim et al. proposed the Squeeze Vit method to study facial expression recognition with squeezed vision converters. They set the training set (validation set + training set) and the test set of the FER dataset at 80.0% and 20.0%, respectively [8]. Xiangkui Jiang et al. proposed a smoking behavior detection method based on YOLOv5 [9], using a homemade smoking behavior dataset for training and setting the proportion of training set to test set to 7:3. Fei Yang et al. proposed a street image external air conditioner unit detection method [10], setting the homemade air conditioner external unit dataset in the proportion of training set (validation set + training set) and test set as 7.4:2.6. Ziyu Zhao et al. proposed a real-time monitoring method for particleboard surface defects based on an improved YOLOv5 [11], setting the particleboard dataset in the proportion of 8:2 between the training set (validation set + training set) and the test set. Kai Huang et al. used Vision Transform [12,13] deep neural network to improve the automatic classification accuracy of the network model. Using the officially given proportions for the expanded TrashNet dataset, the training set, validation set, and test set were set to 53.0%, 17.0%, and 30.0%, respectively. Xufei Wang et al. proposed to divide the Udacity Self-Driving dataset into seven training sets (validation set + training set) and test sets with different proportions, and they trained in the YOLOv4 algorithm, and they found that the proportion of training and test sets has an impact on the detection accuracy of the network model [14]. Pengfei Wang et al. achieved the detection of complex road objects by improving YOLOv5 [15,16]. The expanded MS COCO dataset is divided according to the proportion of the training set, validation set, and test set as 8:1:1. Bengio Y proposed that better results can be obtained when the dataset is around 10,000, so Train:Val:Test = 6:2:2 can be used. When the dataset reaches a million data, Train:Val:Test = 9.8:0.1:0.1 is used to partition the dataset [17]. Chen H. et al. improved YOLO-v4 network to quickly and effectively identify the students' actions in the smart classroom, and he divided the whole dataset into the training set 80% and the test set 20% [18]. Jia Yao et al. proposed a YOLOv5-based detection algorithm [19] by distributing the homemade Kiwi dataset in the proportions of 8:2 between the training set (validation set + training set) and the test set. S. Zhu et al. used four methods to divide the dataset using the SPXY method, K_S method, duplex method, and equal interval division method, and they finally determined the SPXY method as the optimal dataset division method [20]. The optimal division of the prediction and correction sets was 4.8:5.2 for the blood samples and 9:1 for the imitation body solution samples.

In summary, datasets have a very important role in tasks, such as object detection, and they can affect the accuracy of the analysis results, but there is no uniform method for dividing the proportion of datasets. Sangwon Kim et al. [8,10–12] divided the dataset into a training set, as well as a test set; Jia Yao et al. [9,13,14,18] divided the dataset into a training set, validation set, and test set. Kai Huang et al. [13,14,18] used dataset partitioning proportions recommended by neural network authors, and Sangwon Kim et al. [7–12,18] did not specify the method of dataset grouping proportions. It can be seen that scholars have grouped the datasets in such a way that the proportion of the training set to the dataset as a whole varies between 60% and 98%, and the effect of different percentages of the training set in the dataset on the performance of the network model needs further study. In addition, especially when scholars build a new data set when studying a new problem, inexperienced scholars face confusion when dividing the training set. Therefore, the objective of this paper is to investigate the effect of network models on target detection accuracy when the number of samples in the data set varies and the number of target classes varies. In fact, it is necessary to find a recommended data set partitioning proportionality relationship.

In this paper, we propose a twice equal difference enumeration algorithm to investigate the effect of different training set percentages in the dataset on the performance of the network model and determine the optimal percentage interval of the training set. According to the TEDE algorithm, by selecting the Pascal VOC dataset [21]. It was first randomly

divided into six different datasets from 100% to 50%, and then each dataset was divided into the datasets to be analyzed according to five different training set percentages, respectively, and the optimal neural network model performance, corresponding to the training set percentage interval, was determined by using the YOLOv5 [17] convolutional neural network to train and test the 30 datasets. Finally, the Udacity Self-Driving dataset was tested with the homemade tire tread defect dataset [22].

Briefly, the contributions of this paper include the following two points.

- (1) The same dataset, when divided into training and test sets, has a detection accuracy of the network model that is not sensitive to the number of training set samples, either for all samples or for some samples.
- (2) When the dataset is divided into training and test sets, the detection accuracy of the network model on the test set is higher when the proportion of training set samples is between 0.85 and 0.9.

The remainder of this paper is organized as follows: In Section 2, the work related to the YOLOv5 network, the dataset, and its division and evaluation indicators are briefly described. In Section 3, a twice equal difference enumeration algorithm and its application are briefly introduced. Section 4 describes the experimental procedure and results, and the optimal dataset proportional partitioning interval is tested by generalization experiments. The paper is summarized in Section 5.

2. Related Work

2.1. YOLOv5

YOLO [23] is a deep learning-based convolutional neural network commonly used in the field of object detection. YOLO is capable of end-to-end fast and high-accuracy detection of multiple objects. Based on YOLO, the network performance has been gradually improved through the continuous optimization of network structures, such as YOLOv2 [24], YOLOv3 [25], YOLOv4 [26], and YOLOv5 [17].

The YOLOv5 uses the overall layout of the YOLO family, consisting of Input, Backbone, Neck, and Prediction. Backbone refers to a convolutional neural network that aggregates and forms images at different image fine-grains. The neck is a series of network layers that mix and combine image features and pass the images to the prediction layer. The Neck structure of YoloV5 consists of a FPN + PAN structure and a CSP2 structure [27]. FPN is a top-down, side-by-side connection to build a high-level semantic feature graph at all scales, which is the classical structure of a feature pyramid; PAN is a bottom-up structure to fuse feature graphs from different levels by using a convolutional layer, which can effectively enhance the localization information [28,29]. The CSP2 structure designed using CSPNet can effectively improve the network feature fusion [30]. YOLOv5 uses three enhancement methods, mosaic data enhancement [31], adaptive anchor frame calculation, and adaptive image scaling [32], to improve the detection of tiny objects.

2.2. Dataset and Its Division

In the process of building a deep learning model, the quality of the dataset has an impact on the performance of the network model. Two public datasets and one homemade dataset were selected for this study, and the three datasets are described as follows:

- (a) The Pascal VOC common dataset includes 21,504 images with four major categories, which are subdivided into 20 subcategories on top of that, and the name of the subcategories are aeroplane, bicycle, bird, boat, bottle, bus, car, cat, chair, cow, diningtable, dog, horse, motorbike, person, pottedplant, sheep, sofa, train, and tvmonitor. The dataset comes from the Pascal VOC challenge organized by the Pascal network and is mainly applied to the field of object detection [15].
- (b) The Udacity Self-Driving common dataset includes 24,423 images in four categories, and the names of the five categories are Car, Truck, Pedestrian, trafficLight, and biker.

It is a dataset prepared by Udacity for its self-driving algorithm competition, which is mainly applied to study the object detection area of autonomous driving [16].

- (c) The Tire Tread Defect (TTD) dataset consists of 1487 images in four categories, and the names of the four classes are crack, punctured, scratches, and bulge. The dataset was produced by the authors themselves by collecting the images, which were applied to the study of detecting tire tread defects in automobiles.

According to the introduction, it is known that scholars have two approaches to dividing the dataset training set and test set, and one divides the dataset into a training set, a validator, and a test set. The different division of these two datasets is mainly due to the need for network model building. When using the YOLOv5 for research, the network requires that the data set be divided into a training set and a test set, where the training set contains the validation set within it. The proportions of the dataset divided into training and test sets determine the number of samples in the training set. The number of samples in the training set affects the results of neural network training, which further affects the detection accuracy, speed, and generalization ability of the network model. Therefore, it is necessary to investigate the percentage of the training set in the dataset, and it is meaningful to explore the optimal proportional relationship.

2.3. Model Evaluation Indicators

The study of the effect of different proportions of dataset division on network performance needs to be determined by the performance evaluation parameters of the network model. In this paper, we use the mean average precision (*mAP*) to calculate this. The *mAP* is the average of the mean precision of all classes in the dataset, and the higher the value, the higher the average detection precision and the better the performance. The *map* is calculated by the following, Formula (1).

$$mAP = \frac{1}{N} \sum_{i=1}^N P_i \quad (1)$$

In Formula (1), *N* denotes the number of categories of objects in the dataset, and *P_i* is the average accuracy rate of the *i*th category.

Precision (*P*) represents the proportion of positive cases that are correctly predicted to the data predicted to be positive. *P* is calculated by Formula (2), as follows.

$$P = \frac{TP}{TP + FP} \quad (2)$$

In Formula (2), *TP* indicates the number of correctly detected objects, and *FP* is the number of incorrectly detected non-objects.

3. Twice-Equal Difference Enumeration Grouping Algorithm

3.1. Tede Algorithm

In this paper, we propose a new algorithm that firstly, divides the initial dataset into multiple new datasets according to the equal partition pattern; secondly, it divides each group into two subsets, the training set and the test set, and, in each group, the samples of the training set will be divided according to the equal partition pattern, and the remaining samples in the same group will be used as the test set. Thus, multiple datasets with different sample numbers of the two subsets in each group were constructed. Then, through neural network group learning and testing, select a set of datasets that make neural network detection performance the best, and the relationship between the number of samples in this dataset is the optimal group ratio. The grouping method of the dataset is called the Twice-Equal Difference Enumeration (TEDE) algorithm.

3.2. Principle of the Twice-Equal Difference Enumeration Grouping Algorithm

The complete dataset is denoted by *D*, the proportion of the new dataset in the whole dataset *D* is reduced from 1.0 to 0.0 according to the arithmetic sequence, and the stride

is S , which is recommended to be 0.05, 0.1, 0.2., to obtain D_i ($i = 1, 2, 3, \dots, n$); then, each dataset (D_i) is divided into two subsets, the training set and the test set, which are represented by Training Set (Tr_{ij}) and Test set (Te_{ij}), respectively. The idea for the grouping dataset is shown in Figure 1.

According to the grouping idea in Figure 1, Tr_{11}' and Te_{12}' are used to represent the proportion of Training set (Tr_{ij}) and Test set (Te_{ij}) of each subset in each group. The number and size of the datasets in each group were calculated according to Formula (3).

$$\begin{cases} Tr_{ij}' = \frac{Tr_{ij}}{Tr_{ij} + Te_{ij}} \\ Te_{ij}' = \frac{Te_{ij}}{Tr_{ij} + Te_{ij}} \\ Tr_{ij}' + Te_{ij}' = 1 \\ Tr_{ij}' = [1.0 : s : 0.0] \end{cases} \quad (3)$$

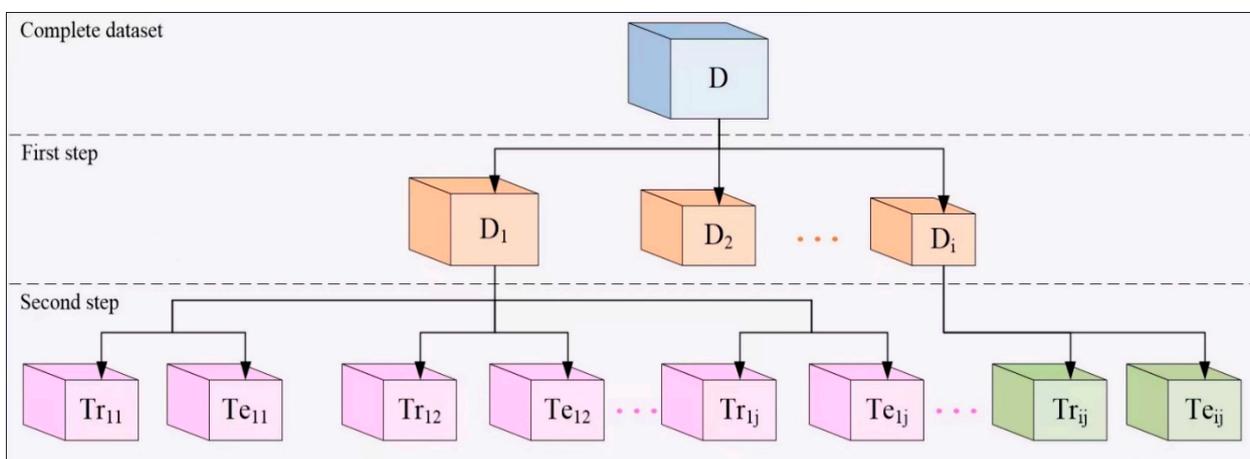


Figure 1. Grouping idea of two equal difference enumeration.

Tr_i' and Te_i' in Formula (3) represent the proportions of Tr_{11} and Te_{12} in the entire dataset, respectively. At the same time, the proportion of the training set in the whole dataset Tr_{11}' is reduced from 1.0 to 0.5 according to the arithmetic sequence, and the stride is S , which is recommended to be 0.05, 0.1, 0.2. According to this algorithm, two subsets of a dataset can be divided into $(1/S + 1)$ groups of different quantities. Then, the $(1/S + 1)$ group is trained and detected, respectively, by using neural network algorithm. By comparing the generalization ability and detection ability of neural network in the $(1/S + 1)$ group, the dataset with the best detection performance is determined as the optimal proportion of the two subsets.

3.3. Grouping of Pascal VOC Dataset

The partitioning of the Pascal VOC dataset is performed in two steps. In the first step, according to Equation (3) of the TEDE algorithm, we set the proportion of the new dataset in the complete dataset D to be reduced from 1.0 to 0.5, and the stride S value of the equal-variance sequence is 0.1, and the complete Pascal VOC dataset is divided into six new datasets with D_i ($i = 1, 2, 3, 4, 5, 6$). The number of images and the number of labels contained in the six new datasets are shown in Table 1.

Table 1. Number of images and labels for the six datasets.

D_i	Proportion	Images	Labels
D_1	100%	21,504	52,576
D_2	90%	19,353	43,946
D_3	80%	17,202	36,719
D_4	70%	15,053	31,114
D_5	60%	12,902	25,847
D_6	50%	10,752	20,350

In the second step, because you cannot use the entire Pascal VOC dataset as a training set, or no training set, the ratio Tr_{11}' cannot be 1.0 and 0. According to the TEDE algorithm, we set the proportion of the training set in D_i to be reduced from 0.9 to 7.0, and the stride S value of the equal-variance sequence is 0.05, as shown in Table 2.

In this way, after a two-step division of the Pascal VOC dataset using the TEDE, 30 sets of datasets D_{ij} including training and test sets awaiting training analysis can be obtained. The full dataset thus obtained using the TEDE is denoted as $D_{ij} = Tr_{ij}/Te_{ij}$ ($i = 1, 2, 3, 4, 5, 6$; $j = 1, 2, 3, 4, 5$). The 30 sets of D_{ij} are trained by the network to obtain the corresponding 30 network models, denoted as N_{ij} .

Table 2. Proportions of the five group divisions.

j	Tr_{ij}'	Te_{ij}'	D_{ij}
1	0.90	0.10	9:1
2	0.85	0.15	8.5:1.5
3	0.80	0.20	8:2
4	0.75	0.25	7.5:2.5
5	0.70	0.30	7:3

4. Experiments

4.1. Experimental Platform

The hardware devices for this experiment include CPU: Inter(R) Core(TM) i9-10900X CPU@2.30 GHz, 64 GB RAM, and GPU: NVIDIA GeForce RTX 3080, 10 GB. The training is performed based on Windows 10 OS and Pytorch1.7 deep learning framework, and the acceleration environment is CUDA11.3, OpenCV4.5.

4.2. Experimental Methods

After establishing the experimental environment, the YOLOv5 network was selected, and the parameters of the network model were set to a total number of 100 iterations, an initial learning rate of 0.01, and a weight decay coefficient of 0.0005. The 30 Tr_{ij} obtained after dividing the Pascal VOC dataset using the TEDE method were trained separately to obtain 30 N_{ij} and then tested on the corresponding test set Te_{ij} separately to obtain the mAP results of 30 N_{ij} . The variation of mAP is analyzed to determine which proportion of the dataset the optimal mAP comes from, respectively, and the optimal division proportion is determined by analyzing the relationship between the number of images in the dataset and mAP.

4.3. Experimental Results Analysis

According to the experimental method, the experimental results are obtained using D_{ij} which has been divided into training and test sets for training and testing. The values of the statistical evaluation index mAP (@.5) are presented side by side in Table 3.

Table 3. mAP (@.5) results for N_{ij} .

D_i	D_{i1}	D_{i2}	D_{i3}	D_{i4}	D_{i5}
D_1	0.874	0.882	0.873	0.852	0.864
D_2	0.871	0.862	0.865	0.845	0.839
D_3	0.853	0.852	0.844	0.840	0.839
D_4	0.831	0.840	0.817	0.827	0.812
D_5	0.817	0.818	0.800	0.802	0.811
D_6	0.795	0.796	0.786	0.790	0.754

As can be seen from Table 3, the size of mAP is different for each row corresponding to D_i . The size of mAP is also different for each column corresponding to D_{ij} . The values of mAP (@.5) are organized into line graphs, as shown in Figure 2.

As seen in Figure 2, when the number of images D_i of the Pascal VOC dataset decreases sequentially with the increase in i , its corresponding mAP (@.5) also decreases sequentially, respectively. The mAP reaches the highest point when the dataset division proportion is taken as D_{i1} or D_{i2} , respectively, and the other mAPs increase and decrease. Specifically, D_1, D_4, D_5 , and D_6 reach the maximum mAP at D_{i2} , respectively, and D_2 and D_3 reach the maximum mAP at D_{i1} .

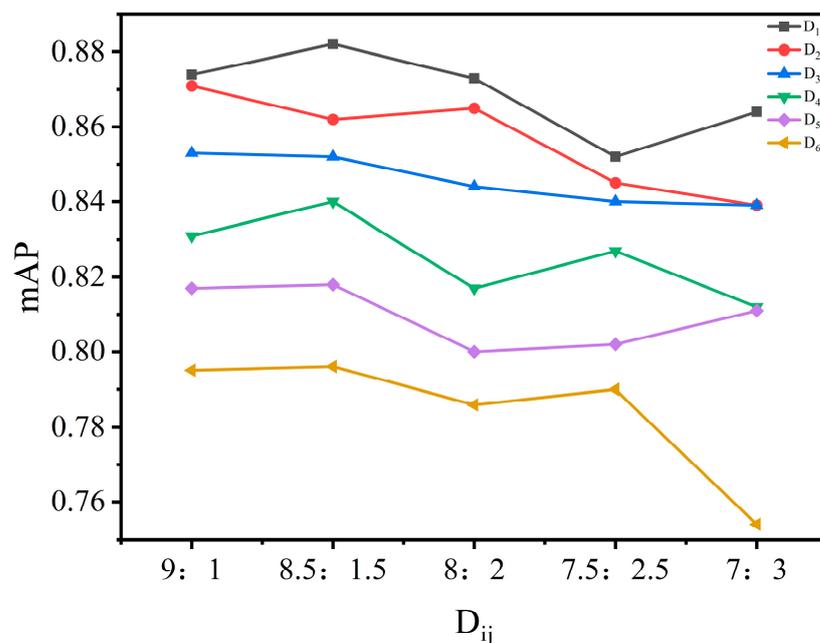


Figure 2. mAP (@.5) variation curve.

To understand the change in mAP (@.5:.95) after the increase in the IoU threshold, mAP (@.5:.95) was counted in Table 4.

Table 4. mAP (@.5:.95) results for N_{ij} .

D_i	D_{i1}	D_{i2}	D_{i3}	D_{i4}	D_{i5}
D_1	0.699	0.700	0.702	0.692	0.688
D_2	0.686	0.680	0.679	0.656	0.650
D_3	0.667	0.677	0.655	0.652	0.651
D_4	0.649	0.648	0.630	0.641	0.621
D_5	0.621	0.630	0.608	0.608	0.612
D_6	0.616	0.622	0.605	0.604	0.567

As can be seen from Table 4, the size of mAP is different for each row corresponding to D_i . The mAP is also different in size for each column corresponding to D_{ij} . The values of mAP (@.5:.95) are organized into line graphs, as shown in Figure 3.

As seen in Figure 3, when the number of images D_i of the PASCAL VOC dataset decreases sequentially with the increase in i , its corresponding mAP (@.5:.95) also decreases sequentially, respectively. The mAP (@.5:.95) reaches the highest point when the dataset division proportion is taken as D_{i1} or D_{i2} , respectively, and the other mAPs have increased and decreased. Specifically, there are D_1 , D_3 , D_5 , and D_6 with maximum mAP at D_{i2} , and D_2 and D_4 with maximum mAP at D_{i1} , respectively.

The analysis of the mAP (@.5) and mAP (@.5:.95) data showed similar trends in the overall mAP values, with the maximum values of mAP occurring at proportions of D_{i1} or D_{i2} , respectively.

The results illustrate that, when experimenting with the Pascal VOC dataset based on the YOLOv5, regardless of the number of dataset images, the mAP of the network model test results can reach a high level when the proportion of D_{i1} or D_{i2} is chosen when dividing the number of training and test sets.

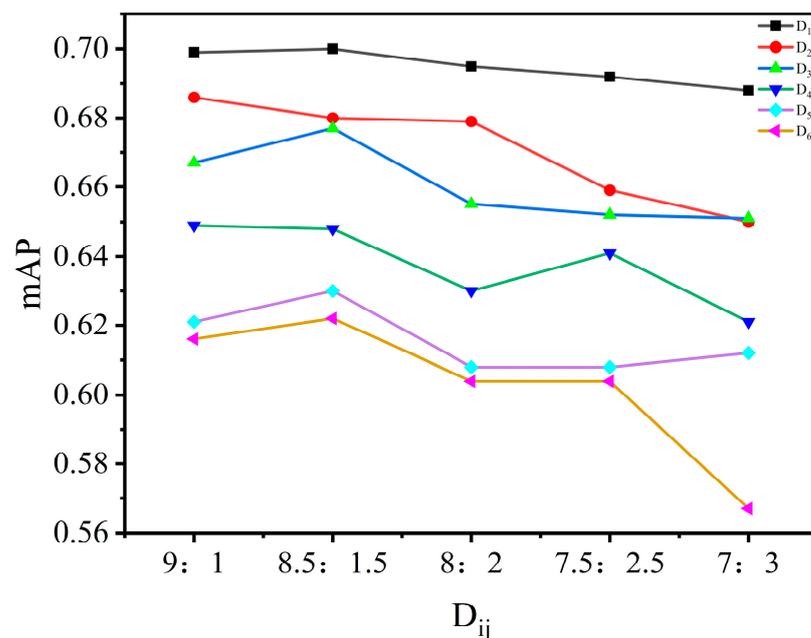


Figure 3. mAP (@.5:.95) variation curve.

To verify the above conclusions in more detail, we choose N_{2i} , obtained after D_{2i} , which is trained by YOLOv5 when mAP (@.5) is used as the reference standard. Then, we choose a typical picture in the Pascal VOC dataset, and the picture is shown in Figure 4a. There are two types of objects in the image, person, and sheep, among which there are six person objects, p1–p6, from the left end to the right end of the image, as well as six sheep objects, s1–s6 from the left end to the right end of the image. Five network models are used to detect the 12 objects in Figure 3a, respectively. The detection results are shown in Figure 4b–f.

In Figure 4b–f, the blue boxes in the detection result pictures indicate the location of the person class object, the purple boxes indicate the location of the sheep class object, and the upper part of the box line is the object class label and the precision rate of the object.

According to the detection results of the pictures in Figure 4, the same number of objects are detected in (b)–(f), and all of them are six persons and five sheep, because the overlapping of sheep objects in the pictures increases the difficulty of detection, resulting in one sheep object not being detected. The detailed data are shown in Table 5.

Table 5. Detailed data on detection results.

N_{2j}	p1	p2	p3	p4	p5	p6	s1	s2	s3	s4	s5
N_{21}	0.95	0.95	0.95	0.95	0.95	0.88	0.93	0.93	0.95	0.93	0.94
N_{22}	0.96	0.95	0.95	0.94	0.94	0.90	0.92	0.93	0.93	0.92	0.90
N_{23}	0.95	0.94	0.94	0.94	0.93	0.93	0.92	0.92	0.94	0.92	0.90
N_{24}	0.94	0.94	0.94	0.93	0.93	0.94	0.91	0.94	0.93	0.93	0.91
N_{25}	0.94	0.94	0.93	0.94	0.88	0.91	0.92	0.92	0.94	0.92	0.91

As can be seen from Table 5, among the object detection results in Figure 3b, p4, p5, s1, and s3–s5 have the highest precision rates, and among the object detection results in Figure 3c, p1 and p6 have the highest precision rates. The two objects, p2 and p3, in Figure 3b,c are tied for the highest accuracy rate. The corresponding dataset proportions in Figure 3b,c are D_{21} and D_{22} , respectively. Therefore, the network model detection obtained when the proportion of dataset D_2 is divided into D_{21} and D_{22} is better.



Figure 4. Detection results of N_{2j} on objects in the Pascal VOC dataset.

In summary, taking Pascal VOC as an example, regardless of the whole or part of Pascal VOC, when the dataset is divided into training and test sets with the proportion set according to D_{21} (9:1) or D_{22} (8.5:1.5), the accuracy rate of both mAP and individual objects of the network model were highest.

4.4. Local Results Analysis

We further analyzed whether there is a better proportional relationship between D_{i1} and D_{i2} , which can lead to higher performance of the network model. In Tables 3 and 4, the mAP, when taking D_{i1} and D_{i2} for each dataset, is selected for subtraction. They are denoted as $|D_{i1} - D_{i2}| \text{mAP} (@.5)$ and $|D_{i1} - D_{i2}| \text{mAP} (@.5; .95)$, respectively, as shown in Table 6.

Table 6. mAP subtraction of D_{i1} and D_{i2} .

D_i	$ D_{i1} - D_{i2} \text{mAP} (@.5)$	$ D_{i1} - D_{i2} \text{mAP} (@.5; .95)$
D_1	0.008	0.003
D_2	0.009	0.006
D_3	0.001	0.01
D_4	0.009	0.001
D_5	0.001	0.009
D_6	0.001	0.006

From Table 6, when mAP (@.5) is used as the reference standard, the largest difference in mAP is 0.009 when the division proportions of D_2 and D_4 are taken as D_{i1} and D_{i2} , and the largest difference in mAP is 0.01 when the division proportions of D_3 are taken as D_{i1} and D_{i2} when mAP (@.5;.95) is used as the reference standard.

When using the two mAP reference standards, the mAP values of the network models obtained from training after D_i was divided using D_{i1} or D_{i2} were similar. Therefore, in future practical applications, there is no need to continue the subdivision of the proportion between D_{i1} and D_{i2} .

4.5. Generalizability Experiments

To increase the reliability of the conclusions obtained with the Pascal VOC dataset only, the Udacity Self-Driving dataset (Udacity) and the TTD dataset (TTD) are then used for their validation. Using their datasets as a whole each divided the training set and test set into five groups, according to the second step of the TEDE algorithm, 10 datasets can be obtained for the experiments and trained using the 4.2 experimental methods, and the statistical experimental results are shown in Table 7.

Table 7. mAP (@.5) results for N_{ij} .

	D_{i1}	D_{i2}	D_{i3}	D_{i4}	D_{i5}
Udacity	0.882	0.89	0.871	0.859	0.870
TTD	0.676	0.67	0.654	0.65	0.651

As shown in the table above, the highest mAP reached 0.89 for the Udacity Self-Driving dataset when the division proportion was taken as D_{i2} , and 0.676 for the TTD dataset when the division proportion was taken as D_{i1} . Therefore, the experimental results are consistent with the findings obtained using the Pascal VOC dataset and are generalizable in practical applications.

To further verify the above conclusions, we select N_{ij} ($i = 1; j = 1, 2, 3, 4, 5$), obtained after the Udacity Self-Driving dataset was trained by YOLOv5, and then we select a typical image from the Udacity Self-Driving dataset, as shown in Figure 5a. There are two types of objects in the figure, person, and car, where there are six person objects and six car objects. The 12 objects in Figure 5a are detected by five network models, respectively. The detection results are shown in Figure 5b–f.

In Figure 5b–f detection results, the blue box in the picture indicates the person class object location, from the left end to the right end of the picture in the order of p1–p3; the yellow box indicates the car class object location, from the left end to the right end of the picture in the order of c1–c4; and, the upper part of the box line is the object class label and the accuracy rate of the object.

Based on the detection results of the images in Figure 5, it can be obtained that the same number of objects are detected in (b)–(e), which are three person objects and four car objects. The overlapping of prediction frames resulting in the accuracy rate of individual objects cannot be seen, three-person objects and two-car objects are missed at the same time, and the detailed data are shown in Table 8.

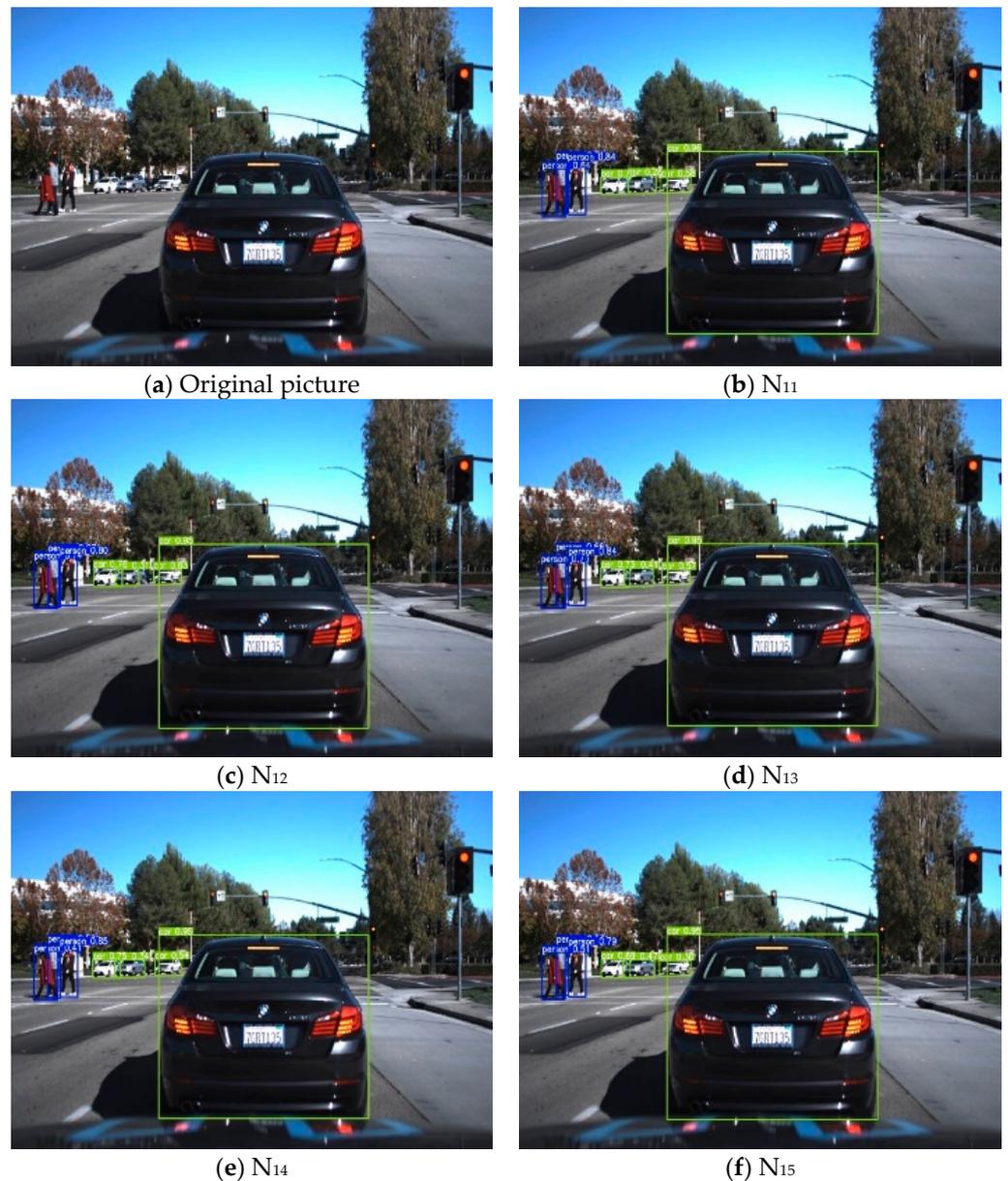


Figure 5. Detection results of N_{1j} on objects in the Udacity Self-Driving dataset.

Table 8. Detailed data of detection results.

N_{1j}	p1	p2	p3	c1	c2	c3	c4
N_{11}	0.64	0.71	0.84	0.70	0.28	0.58	0.96
N_{12}	0.76	0.27	0.80	0.76	0.31	0.63	0.95
N_{13}	0.74	0.66	0.84	0.73	0.41	0.57	0.95
N_{14}	0.51	0.59	0.85	0.75	0.34	0.54	0.95
N_{15}	0.51	0.53	0.79	0.69	0.47	0.30	0.95

According to the results in Table 8, it can be seen that p2 and c4 has the highest accuracy rate among the object detection results in Figure 5b, and p1, c1, and c3 have the highest accuracy rate among the object detection results in Figure 5c. A total of five detected objects has the highest accuracy rates when the dataset is divided using D_{i1} and D_{i2} . Therefore, the network model detection is better when the proportion of the Udacity Self-Driving dataset is divided into D_{i1} or D_{i2} .

5. Conclusions

For how to choose the best division proportion method for datasets, this study proposes a twice-equal difference enumeration (TEDE) algorithm for datasets. Using the YOLOv5, six datasets with different amounts of data were divided based on the common dataset PASCAL VOC, and experiments were conducted on each dataset under five different division proportions to obtain a total of 30 sets of experimental results. The experimental results showed that, when mAP (@.5) and mAP (@.5:.95) were used as evaluation metrics, the highest prediction precision of the network model was achieved when the proportion of the training set station dataset as a whole was set to 0.85–0.9 for datasets with different amounts of data. The generalizability experiments use the Udacity Self-Driving dataset and the TTD dataset and prove that D_{i1} or D_{i2} are the optimal division proportions of the dataset with certain reliability and generalizability. The above conclusion is further supported after testing using real images. Therefore, the TEDE algorithm can provide a reference for researchers engaged in deep learning-based computer vision when working on dataset partitioning, which helps to improve the performance of network models and is important for the establishment of network models.

Author Contributions: Conceptualization, X.W. and J.S.; methodology, X.W. and P.W.; validation, T.H. and X.D.; writing—original draft preparation, writing—review and editing, P.W. and X.W. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the Shaanxi Provincial Key Laboratory of Industrial Automation Research Program under Grant 18JS020.

Data Availability Statement: Data are contained within the article.

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

References

1. So, S.; Badloe, T.; Noh, J.; Bravo-Abad, J.; Rho, J. Deep learning enabled inverse design in nanophotonics. *Nanophotonics* **2020**, *9*, 1041–1057. [[CrossRef](#)]
2. Kotsiantis, S.B.; Zaharakis, I.; Pintelas, P. Supervised machine learning: A review of classification techniques. *Emerg. Artif. Intell. Appl. Comput. Eng.* **2007**, *160*, 3–24.
3. Stojanov, D.; Lazarova, E.; Veljkova, E.; Rubartelli, P.; Giacomini, M. Predicting the outcome of heart failure against chronic-ischemic heart disease in elderly population—Machine learning approach based on logistic regression, case to Villa hospital Genoa, Italy. *J. King Saud Univ. Sci.* **2023**, *35*, 102573. [[CrossRef](#)]
4. Langarizadeh, M.; Moghbeli, F. Applying naive bayesian networks to disease prediction: A systematic review. *Acta Inform. Med.* **2016**, *24*, 364. [[CrossRef](#)]
5. Xing, W.; Bei, Y. Medical Health Big Data Classification Based on KNN Classification Algorithm. *IEEE Access* **2020**, *8*, 28808–28819. [[CrossRef](#)]
6. Battineni, G.; Chintalapudi, N.; Amenta, F. Machine learning in medicine: Performance calculation of dementia prediction by support vector machines (SVM). *Inform. Med. Unlocked* **2019**, *16*, 100200. [[CrossRef](#)]

7. Schuldts, C.; Laptev, I.; Caputo, B. Recognizing human actions: A local SVM approach. In Proceedings of the 17th International Conference on Pattern Recognition, 2004, ICPR 2004, Cambridge, UK, 26 August 2004; Volume 3, pp. 32–36. [CrossRef]
8. Kim, S.; Nam, J.; Ko, B.C. Facial Expression Recognition Based on Squeeze Vision Transformer. *Sensors* **2022**, *22*, 3729. [CrossRef]
9. Jiang, X.; Hu, H.; Liu, X.; Ding, R.; Xu, Y.; Shi, J.; Du, Y.; Da, C. A smoking behavior detection method based on the YOLOv5 network. *J. Phys. Conf. Ser.* **2022**, *2232*, 012001. [CrossRef]
10. Yang, F.; Wang, M. Deep Learning-Based Method for Detection of External Air Conditioner Units from Street View Images. *Remote Sens.* **2021**, *13*, 3691. [CrossRef]
11. Zhao, Z.; Yang, X.; Zhou, Y.; Sun, Q.; Ge, Z.; Liu, D. Real-time detection of particleboard surface defects based on improved YOLOV5 target detection. *Sci. Rep.* **2021**, *11*, 21777. [CrossRef]
12. Huang, K.; Lei, H.; Jiao, Z.; Zhong, Z. Recycling waste classification using vision transformer on portable device. *Sustainability* **2021**, *13*, 11572. [CrossRef]
13. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Polosukhin, I. Attention is all you need. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; Volume 30.
14. Wang, X.; Chen, L.; Li, Q.; Son, J.; Ding, X.; Song, J. Influence of self-driving data set partition on detection performance using YOLOv4 network. *J. Inst. Internet Broadcast. Commun.* **2020**, *20*, 157–165.
15. Jocher, G.; Stoken, A.; Borovec, J.; Changyu, L.; Hogan, A.; Diaconu, L. Ultralytics/yolov5. Github Repository, YOLOv5. 2020. Available online: <https://ui.adsabs.harvard.edu/abs/2020zndo...3983579J/abstract> (accessed on 15 August 2022).
16. Wang, P.; Huang, H.; Wang, M.; Li, B. YOLOv5s-FCG: An improved YOLOv5 method for inspecting Riders' helmet wearing. *J. Phys. Conf. Ser.* **2021**, *2024*, 012059. [CrossRef]
17. Bengio, Y.; Goodfellow, I.; Courville, A. *Deep Learning*; MIT Press: Cambridge, MA, USA, 2017.
18. Chen, H.; Guan, J. Teacher–Student Behavior Recognition in Classroom Teaching Based on Improved YOLO-v4 and Internet of Things Technology. *Electronics* **2022**, *11*, 3998. [CrossRef]
19. Yao, J.; Qi, J.; Zhang, J.; Shao, H.; Yang, J.; Li, X. A real-time detection algorithm for Kiwifruit defects based on YOLOv5. *Electronics* **2021**, *10*, 1711. [CrossRef]
20. Zhu, S.C.; Gao, S.Y.; Ren, C. Study on the division proportion and preprocessing method of the infrared spectral dataset. *Anal. Chem.* **2022**, *50*, 14151429.
21. Everingham, M.; Gool, L.V.A.; Williams, C.K.I.; Winn, J.; Zisserman, A. The Pascal visual object classes (VOC) challenge. *Int. J. Comput. Vis.* **2010**, *88*, 303–338. [CrossRef]
22. An Open Source Self-Driving Car, Udacity, Emeryville, CA, USA. 2017. Available online: <https://github.com/udacity/self-driving-car/tree/master/annotations> (accessed on 30 March 2022).
23. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
24. Redmon, J.; Farhadi, A. YOLO9000: Better, faster, stronger. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 7263–7271.
25. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv Preprint* **2018**, arXiv:1804.02767.
26. Bochkovskiy, A.; Wang, C.Y.; Liao, H.Y.M. Yolov4: Optimal speed and accuracy of object detection. *arXiv Preprint* **2020**, arXiv:2004.10934.
27. Wan, G.; Fang, H.; Wang, D.; Yan, J.; Xie, B. Ceramic tile surface defect detection based on deep learning. *Ceram. Int.* **2022**, *48*, 11085–11093. [CrossRef]
28. Fu, L.; Duan, J.; Zou, X.; Lin, J.; Zhao, L.; Li, J.; Yang, Z. Fast and accurate detection of banana fruits in complex background orchards. *IEEE Access* **2020**, *8*, 196835–196846. [CrossRef]
29. Xue, J.; Zheng, Y.; Dong-Ye, C.; Wang, P.; Yasir, M. Improved YOLOv5 network method for remote sensing image-based ground objects recognition. *Soft Comput.* **2022**, *26*, 10879–10889. [CrossRef]
30. Wang, C.Y.; Liao, H.Y.M.; Wu, Y.H.; Chen, P.Y.; Hsieh, J.W.; Yeh, I.H. CSPNet: A new backbone that can enhance learning capability of CNN. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, Seattle, WA, USA, 13–19 June 2020; pp. 390–391.
31. Irani, M.; Anandan, P.; Hsu, S. Mosaic based representations of video sequences and their applications. In Proceedings of the IEEE International Conference on Computer Vision, Cambridge, MA, USA, 20–23 June 1995; pp. 605–611.
32. Zheng, J.C.; Sun, S.D.; Zhao, S.J. Fast ship detection based on lightweight YOLOv5 network. *IET Image Process.* **2022**, *16*, 1585–1593. [CrossRef]

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