Optimizing Driving Completeness Prediction Models: A Comparative Study of YOLOv7 and Naïve Bayes at Institut Teknologi Sumatera

Muhammad Habib Algifari [1*], Ilham Firman Ashari [2*], Eko Dwi Nugroho [3*], Aidil Afriansyah [4*], Mario Vebriyanto[5*]
* Teknik Informatika, Institut Teknologi Sumatera
muhammad.algifari@if.itera.ac.id [1], firman.ashari@if.itera.ac.id [2], eko.nugroho@if.itera.ac.id [3], aidil.afriansyah@if.itera.ac.id [4], mario.119140179@student.itera.ac.id [5]

Article Info

Article history:

Received 2023-11-05
Revised 2023-11-13
Accepted 2023-11-15

Keyword:

Model,
YOLOv7,
Naïve Bayes,
Object Detection,
Driving Equipment.

ABSTRACT

The number of vehicles in Indonesia is increasing every year. The number of motor vehicle accidents in 2022 will be more than 100,000. It is hoped that several regulations regarding motorbike rider equipment will increase awareness of rider safety. By utilizing image recognition technology developed with artificial intelligence, it is possible to create digital image processing models or images that are fast and accurate for detecting driving equipment. The object detection model developed uses a dataset in the form of images of motorists who want to enter ITERA through the main gate. The object detection model will also be integrated with the classification model to create a program that can detect motorbike rider equipment, such as mirrors, helmets, not wearing a helmet, shoes, not wearing shoes, open clothes, and closed clothes. After detecting motorized rider equipment in the classification area, the results will be transferred to a classification model to determine the level of safety for motorized riders, either insufficient or sufficient safety. The test results show that the optimal object detection model was found at an epoch value of 70 with a batch-size of 16, producing a mAP value of 0.8914. The optimal classification model uses the naive Bayes method which has been trained with a dataset of 62 data and achieves an accuracy of 94%.

I. Introduction

The equipment of motorized riders and passengers has been regulated by the Indonesian government in Law No. 22 of 2009 Article 57 Paragraphs 1 and 2, and Article 285[1][2]. Furthermore, a circular issued by ITERA with Number B/188/IT9.A2/RT.08/2022 regulates access in and out of ITERA. Several regulations regarding the ITERA gate state that the main ITERA gate will open at 05:30 WIB and close at 21:00 WIB, while the west gate will close at 17:00 WIB. ITERA's main gate can be accessed by employees, students and guests, while ITERA's west gate can only be accessed by employees and project vehicles [3][4].

The hope is that the driving equipment detection model can increase drivers' awareness of obeying the rules and ultimately reduce the risk of accidents. This research will develop digital image processing with deep learning algorithms to detect factors that can reduce the risk of accidents. Digital image processing will include detecting the presence of rear-view mirrors on motorbikes, the use of helmets, shoes and closed clothing. Each detected element will be evaluated using the Naive Bayes algorithm to conclude the level of driver safety on the road [5].

The use of the YOLOv7 (You Only Look Once version 7) method in digital image processing is a significant step in the development of an object detection system. YOLOv7 is a Convolutional Neural Network based algorithm that has been proven to have extraordinary performance. One important aspect that differentiates YOLOv7 is its extremely fast processing time, enabling real-time object detection with high accuracy. This high speed and accuracy make YOLOv7 superior to most available object detectors [6][7][8]. By utilizing this technology, a system can be designed that is able to recognize objects quickly and accurately, making a major contribution to improving the performance of various applications, such as traffic monitoring, security and image processing.

The data that will be used in this research is a number of images of motorbike riders taken from CCTV footage of the ITERA main gate [9][10]. At the object detection stage, a labeling process will be carried out to identify several driver safety attributes, including helmet use (H), non-use of helmet (TPH), presence of rear-view mirrors (SL), use of closed clothing (PT), use of open clothing (PB), use of shoes (PS), and non-use of shoes (TPS). This process will make it possible to understand and analyze motorcyclist behavior in the context of traffic safety in more detail.

Each label has an important role in the classification of the naive Bayes method, which involves seven key parameters, namely helmet, not wearing a helmet, shoes, not wearing shoes, open clothes, closed clothes, and mirrors, each with a value of 1 or 0. However, The mirror parameter has three different values, namely 0, 1, and 2. The Naive Bayes algorithm operates based on the concepts of statistics and probability, so that for each parameter, probability calculations are carried out to estimate the possibility of similarity in characteristics with other samples. [11][12][13][14]. The results of this classification will produce two outputs, namely a level of safety that is considered sufficient and a level of safety that is considered insufficient.

Testing this research will use the dataset testing method, which has information that will be used to match the results with the information in the dataset. The information in the dataset includes name labels and seven parameters and values that will be used in naive Bayes classification. Apart from that, testing will also utilize several parameters, such as Mean Average Precision (mAP), Intersection over Union (IoU), Precision, and Recall, to evaluate the optimal level of object detection [15][16][17].

This research aims to develop a classification model for motorbike driver safety equipment which can later be implemented in front of the main gate of ITERA. With this system, if the driver's safety is deemed complete, the driver will be allowed to enter the area. This application has the potential to be the beginning of the development of automatic gates that can be used at ITERA main gates.

II. METHODS

The modelling has been developed with the aim of classifying motorbike rider equipment that wishes to enter ITERA based on safety levels. The attributes included in this modelling are the use of helmets (H), the non-use of helmets (TPH), the presence of rear-view mirrors (SL), the use of closed clothing (PT), the use of open clothing (PB), the use of shoes (PS), and the non-use of shoes (TPS). The results of this method will produce an assessment of the level of safety, which can be sufficient or insufficient safety. Evaluation and testing of this model will involve the use of (confusion matrix) and Intersection over Union to measure the level of accuracy of the model.

A. Development of Model YOLOv7

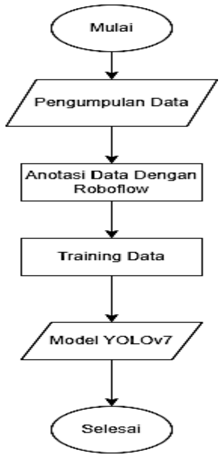

Figure 1. Yolo V7 Model Training

YOLO modelling will be carried out with 2 main processes including a training process and a testing process, each process has several stages and inputs so that it can produce output according to wishes. The stages in each process can be seen from the flowchart in Figure 1 which is an illustration of the training process which begins with the data collection process which is an image with objects of motorbike riders who do not use or use several attributes such as motorbike mirrors, helmets, closed clothing and shoes. Each image will be annotated using RoboFlow to carry out labelling and bounding boxes. After the annotation is carried out, a training dataset will be carried out so that YOLOv7 can detect objects. The results of this process are images that have been detected and labelled H (Helmet), TPH (Not Wearing Helmet), S (Rear-view Mirror), PT (Closed Clothes), PB (Open Clothes), PS (Wearing Shoes), and TPS (Not Wear shoes). Labelling is marked with a rectangular line or bounding box.

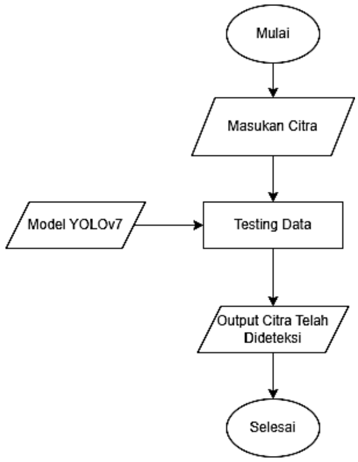

Figure 2. Yolo V7 Model Testing

Meanwhile in Figure 2 is a picture of the testing process which begins by entering the image you want to test into Python for testing with YOLOv7 which has been trained with the output in Figure 1. So, this process will produce an image of a motorbike rider with the equipment that has been detected.

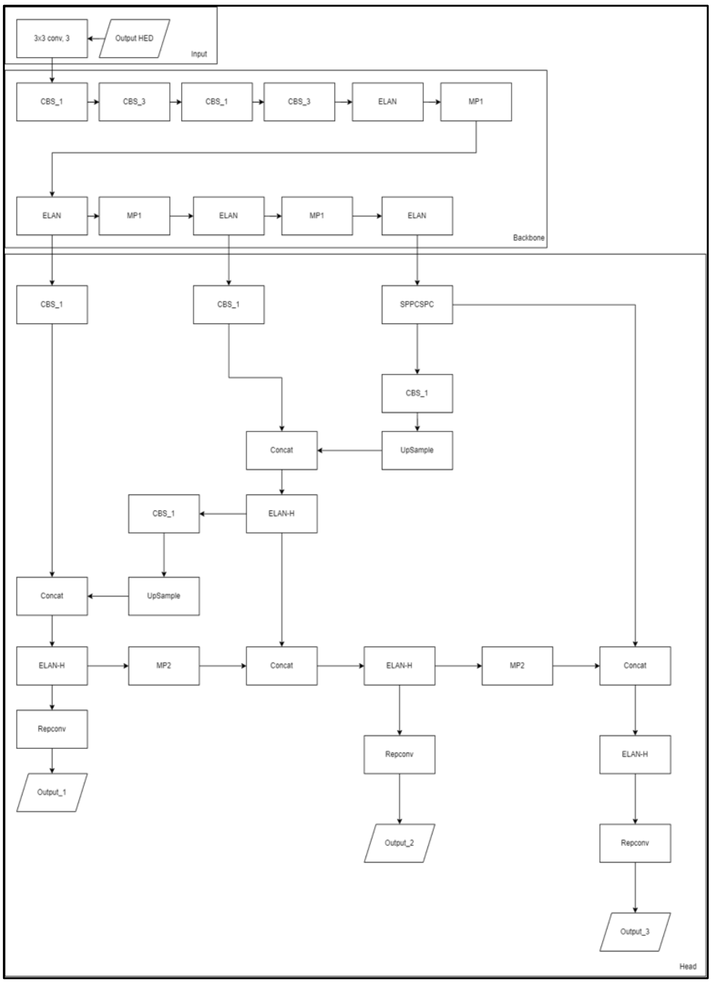

Figure 3. YoloV7 Model Architecture

1) Input

The input image will be conv 3x3 with numbering of kernel stage 3 so that the output image will be (input image size) x (input image size) x 3.

2) Backbone

In this process several modules are used including CBS (Cross-Stage Partial Network) which uses several combinations of convolution which is useful for extracting important features from the input image, BN (batch normalization) which is useful for improving network stability and performance and SILU (Sigmoid- Weighted Linear Unit) which is useful for increasing the network's ability to learn more complex and abstract features. The combination used can be seen in the picture below where CBS_1 only uses convolution, CBS_2 uses SILU and convolution, and CBS_3 uses SILU and BN. The ELAN (Efficient Layer Aggregation Network) module is a process that is filled with several CBS_1 and CBS_2 processes which will be collected to carry out a concat process which is useful for combining 2 or more tensors with the same dimensions and then twinning CBS_1, and the MP (Multi) module. - Prediction) is a process that contains a maxpool process which is useful for changing the dimensions of the input to half the initial dimensions which will be continued with CBS_1 and combined with the processes below CBS_1 and CBS_2. In the backbone process, MP1 or multi-prediction with 1 channel is used. Every output from the ELAN process will be continued back to the head process.

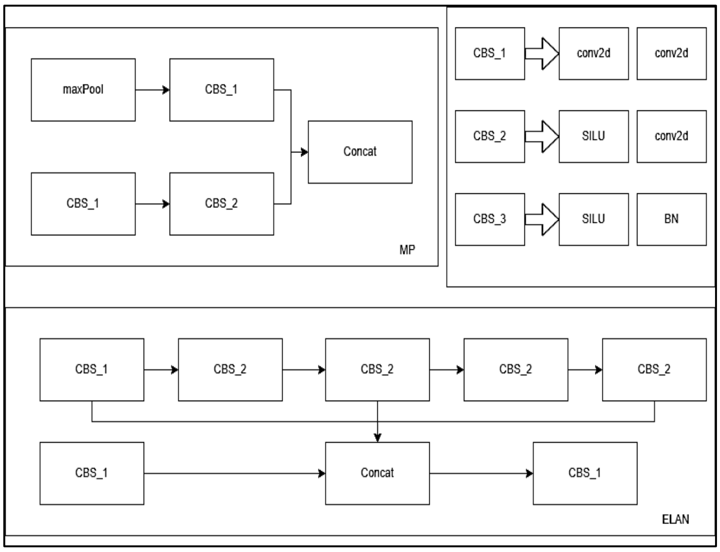

Figure 4. MP and ELAN Model Structure

3) Head

In the backbone process there are 3 outputs produced by the ELAN process with 3 different image dimensions because 3 MP processes have occurred to be continued in the head process. The head process will do a lot, all of which are interconnected to get 3 outputs that have obtained the results of object detection, where the first output from the backbone will be processed with CBS_1 and then combined with an upsample from the second output of the backbone so that the tensors of the two outputs are the same so that the concat process can be carried out. After the concat process is carried out again, ELAN-H is almost the same as ELAN, but each output from CBS_2 will be combined in the cancat process. The process continues again with the repconv process to increase training time and increase the inference effect so that the output has maximum results. At the 2nd output of the backbone, the same process is carried out, where concat will be carried out with an upsample from the 3rd output of the backbone and the ELAN-H process will be carried out again. After that, ELAN-H will concat again with MP from the output of 1 backbone with 2 channels. ELAN-H and repconv are carried out to get the original results as the second output. Almost the same process is carried out on the 3rd backbone output but there is a SPPCSPC process in the initial process of the 3rd backbone output. The SPPCSPC process is carried out to obtain multi-scale object information while maintaining the feature map size unchanged.

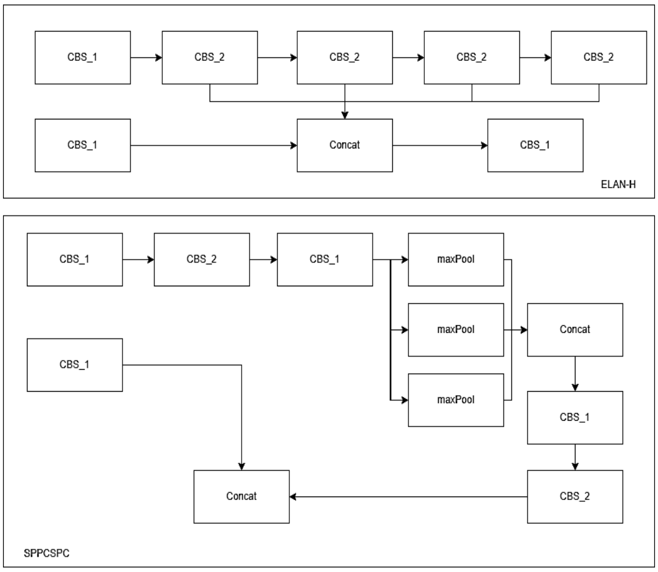

Figure 5. Structure of the ELAN-H and SPCCSPC models

B. Development of Naive Bayes

Naive Bayes modelling will be carried out with 2 main processes including a training process and a testing process, each process has several stages and inputs so that it can produce output according to wishes.

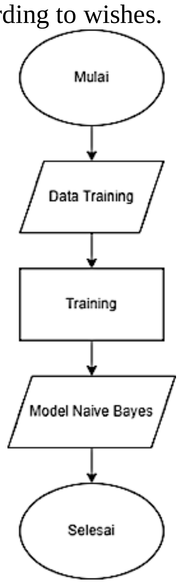

Figure 6. Naïve Bayes Model Training

The stages in each process can be seen from the flowchart in Figure 6, a description of the training process which begins with the process of entering training data in the form of a table containing 4 parameters including helmets, shoes and clothing with respective values YES and NO, meanwhile, the mirror parameters will be has 3 values, namely 0,1, and 2. The table is also equipped with conclusions that have Safe and Unsafe levels. The dataset will be trained so that it will get output in the form of a naive Bayes model to get a calcification. Classification is obtained from the naive Bayes model which has calculated probabilities and opportunities.

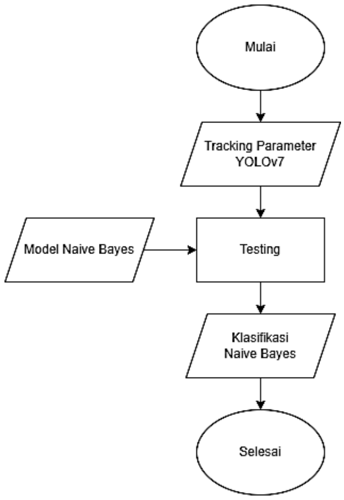

Figure 7. Naïve Bayes Model Testing

Meanwhile, in figure 7 depicts the testing process which begins by entering the tracking results into YOLO to obtain the parameters that will be used in the naive Bayes classification. After the data is obtained, testing is carried out with a previously created model to calculate the probability and opportunity to obtain output in this process. The output of this process is the final classification result that motorists can be declared safe or unsafe.

III. Result and Discussion

The results of the research that has been carried out consist of a dataset, dataset annotation, YOLOv7 model, naive Bayes model, YOLOv7 testing results, naive Bayes testing results, object detection classification model testing, and confusion matrix evaluation. The dataset used is divided into 3 types and several Hyperparameter configurations have been changed to obtain optimal object detection results.

A. Data Collection

Modelling was carried out with the following data distribution: 184 train sets, 62 valid sets and 27 valid sets. with distribution between classes 225 helmets (153 train, 54 valid, 18 test), 311 wearing shoes (217 train, 64 valid, 30 test), 107 open clothing (73 train, 24 valid, 10 test), 304 closed clothing (203 train, 70 valid, 31 test), 502 mirrors (340 train, 113 valid, 49 test), 218 not wearing a helmet (125 train, 40 valid, 23 test), 97 and not wearing shoes (56 train, 30 valid, 11 test).

The second data uses CCTV footage for 2 days at the main gate of ITERA on 3 and 4 September 2023 at 07:00 – 17:00 by taking frames every 210 seconds so that more than 30,000+ images are obtained and then checked manually so that 1598 images are obtained. there is a picture of a motorbike rider.

with the following data distribution 1116 train sets, 402 valid sets and 80 valid sets with distribution between classes 1805 helmets (1251 train, 473 valid, 81 test), 2304 wearing shoes (1603 train, 588 valid, 113 test), 725 open clothes (513 train, 179 valid, 33 test), 2455 closed clothing (1712 train, 626 valid, 117 test), 3787 mirrors (2637 train, 966 valid, 184 test), 1405 not wearing a helmet (994 train, 339 valid, 72 test), and 564 did not wear shoes (399 train, 141 valid, 24 test). The 4th dataset is an updated dataset from the 3rd dataset which is added with vehicle bounding boxes.

B. *Data Annotation*

Data that has been taken from several methods described above will be annotated and labeled according to the attributes in the digital image. Image annotation uses an AI platform, namely Roboflow datasets that have been annotated can be accessed using the API or export the dataset. The dataset that has been annotated is public on RoboFlow, but if you want the dataset to be non-public then you can subscribe to RoboFlow. An example of an annotated image can be seen in Figure 8.

C. *Dataset Naive bayes*

The naive Bayes dataset was created from the YOLOv7 dataset which has been annotated to obtain a conclusion where each attribute has its own value in accordance with the regulations applicable in the ITERA Region.

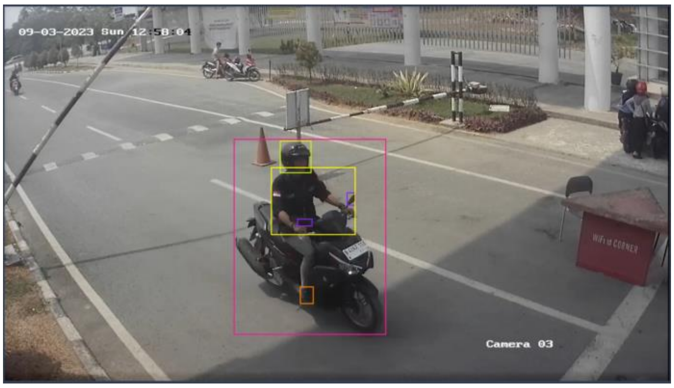

Figure 8. Annotated Image

By taking an annotated dataset into CSV output in RoboFlow. Each final attribute result contained in the body of the conclusion will be divided into the number of people in the annotated dataset and will be added to the final mirror result which has been divided by the number of motorbikes. The final result will be divided into 2, including sufficient safety if the final value is more than 70 and less safety if the final result is less than 70. The calculations that occur can be seen from dataset attribute value weighting.

```
import pandas as pd
data = pd.read_csv('NB_Train.csv')
data['Total Helm'] = data['Helm'] + data['Tidak Pakai
        Helm']
print(data)
data['Total Persentase'] = (
    data['Helm'] * 40 +
    data['Pakaian Tertutup'] * 15 +
    data['Pakaian Terbuka'] * 7 +
    data['Pakai Sepatu'] * 15 +
    data['Tidak Pakai Sepatu'] * 7 +
    data['Tidak Pakai Helm'] * 20
)
data['Total Persentase'] = (
    data['Total Persentase'] / data['Total Helm']
    + data['Spion'] * 15
)
print(data['Total Persentase'])
data['Kesimpulan']          =          data['Total
Persentase'].apply(lambda x: 1 if 101 > x > 70 else 0)
```

D. *Load Model YoloV7*

The model that has been obtained in the train process will be taken and formed based on the YOLOv7 architecture in carrying out detection which can be seen in Figure 3. The results of the function will create a YOLOv7 architecture that can detect classes that have been previously trained. Formation of the Yolo V7 model.

```
def attempt_load(weights, map_location=None):
    model = Ensemble()
    for w in weights if isinstance(weights, list)
else [weights]:
        attempt_download(w)
        ckpt = torch.load(w,
map_location=map_location)
        model.append(ckpt['ema' if ckpt.get('ema')
else 'model'].float().fuse().eval())

    for m in model.modules():
        if type(m) in [nn.Hardswish, nn.LeakyReLU,
nn.ReLU, nn.ReLU6, nn.SiLU]:
            m.inplace = True
        elif type(m) is nn.Upsample:
            m.recompute_scale_factor = None
        elif type(m) is Conv:
            m._non_persistent_buffers_set = set()

    if len(model) == 1:
        return model[-1]
    else:
        print('Ensemble created with %s\n' %
weights)
        for k in ['names', 'stride']:
            setattr(model, k, getattr(model[-1],
k))
        return model
```

E. *Load Model Resnet-101*

Using the functions in torch vision with the resnet-101 model to help and convince the YOLOv7 model that has been trained in determining detection results. The resnet model is a part of CNN that will obtain information from images with different dimensions. The process of creating the resnet-101 model.

```
def load_classifier(name='resnet101', n=2):
    model =
torchvision.models.__dict__[name](pretrained=True)

    filters = model.fc.weight.shape[1]
    model.fc.bias = nn.Parameter(torch.zeros(n),
requires_grad=True)
    model.fc.weight = nn.Parameter(torch.zeros(n,
filters), requires_grad=True)
    model.fc.out_features = n
    return model
```

F. Detection of Objects in the Detection Area Using Naïve Bayes

This process will begin by checking each existing prediction result to obtain the bounding box value for the vehicle class to be included in the area list. The next process obtains the overall middle value of x and y in each detection result to check if the x and y values are within the bounding box of the vehicle that has been included in the area list. And will be stored in saveNB to be classified by naive bayes. The program code for detection in classified areas.

```
jumlahkend = 0
for kendaraan in det:
    if kendaraan[5] == 1:
        area.append(kendaraan[0])
        area.append(kendaraan[2])
        area.append(kendaraan[1])
        area.append(kendaraan[3])
        jumlahkend +=1

saveNB = []
for kotak in det:
    print(kotak)
    center_x = (int(kotak[0])+int(kotak[2])) / 2
    center_y = (int(kotak[1])+int(kotak[3])) / 2

    for i in range(jumlahkend):
        x1 = int(area[(i*4)+0])
        y1 = int(area[(i*4)+2])
        x2 = int(area[(i*4)+1])
        y2 = int(area[(i*4)+3])
        print(x1,y1,x2,y2,center_x,center_y,i)
        if x1 < center_x < x2 and y1 < center_y <
y2:
            saveNB.append(kotak.tolist())
            saveNB.append(i)
            break
```

G. Program Detection and Classification Results

With the integrated program, you will be able to detect the attributes of motorized driving equipment that wants to enter the ITERA area on video or images.

The detection results that will be carried out are also equipped with a classification area which will change colour according to the attribute classification results in that area. Figure 9 is an output image that detects that the driver's equipment is categorized as adequate safety and Figure 10 is an output image that detects that the driver's equipment is categorized as inadequate safety.

And there are also several conditions that are taken into consideration, including the presence of more than 1 vehicle, the output of which can be seen in Figure 11, where each vehicle has its own classification area, namely the vehicle detection bounding box. The presence of more than 2 people in 1 motorized vehicle which can be seen in Figure 12 is a special classification which will change the color of the classification area to blue.

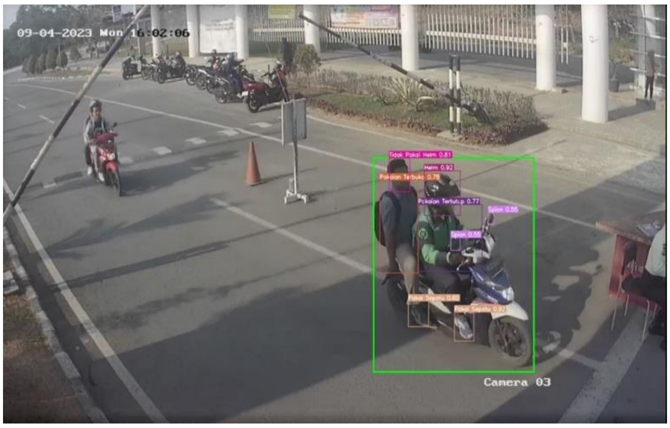

Figure 9. Sufficient Safety Output

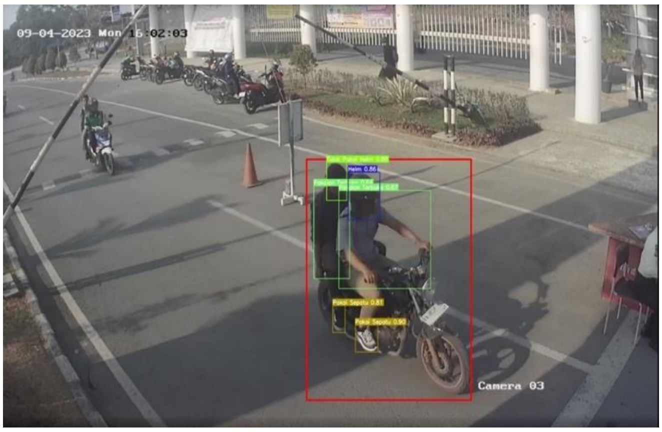

Figure 10. Poor Safety Output

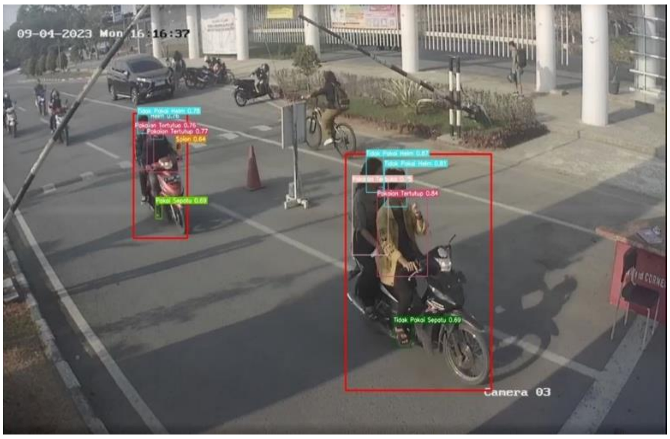

Figure 11. Output from more than one vehicle

Figure 12. Output of more than 2 people in 1 vehicle.

H. YOLOv7 Testing

YOLOv7 modelling has provided calculated test parameter values which have been calculated. Several models that have been carried out have test values which can be seen in Table 1 for several modelling and test results that have been carried out.

TABLE I
YOLOV7 MODEL TESTING7

Data	Batch-*size*	*Epoch*	mAP	Precision	Recall
1	16	100	0.7216	0.741	0.7541
1	16	125	0.7737	0.8165	0.7538
1	16	165	0.7128	0.6952	0.7623
1	16	200	0.07143	0.07835	0.194
2	16	70	0.5877	0.5258	0.6449
2	16	110	0.8426	0.7475	0.8821
2	16	125	0.9104	0.8559	0.8897
2	16	150	0.3646	0.5626	0.4231
3	16	55	0.6286	0.5471	0.7284
3	16	75	0.7242	0.6618	0.7575
3	16	90	0.8459	0.8393	0.7782
3	16	100	0.4523	0.4499	0.4866

I. Naïve Bayes Testing

Naive Bayes testing will be carried out by manual calculations using matrix confusion to obtain accuracy values using equation 0.2, precision using equation 0.3, and recall using equation 0.4. The fusion matrix obtained during modelling can be seen in the following image.

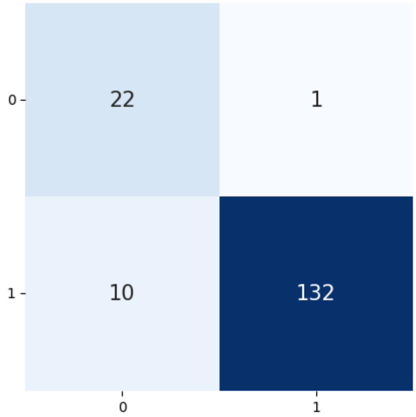

Figure 13. Confusion Matrix from 184 Datasets

With the confusion matrix with a dataset of 184 obtained in Figure 13, calculations were carried out to obtain accuracy, precision and recall as follows.

Accuracy = $\frac{22+132}{22+1+132+10}$ = 0,93 = 93%
Precision = $\frac{22}{22+1}$ = 0,956 = 96%
Recall = $\frac{22}{22+10}$ = 0,687 = 69%

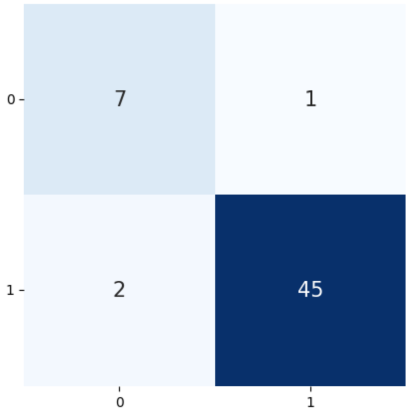

Figure 14. Confusion Matrix from 62 Datasets

With the confusion matrix with a dataset of 62 obtained in Figure 14, calculations are carried out to obtain accuracy, precision and recall as follows.

Accuracy = $\frac{7+45}{7+1+45+2}$ = 0,945 = 94%
Precision = $\frac{7}{7+1}$ = 0,875 = 87%
Recall = $\frac{7}{7+2}$ = 0,78 = 78%

J. Evaluation

Evaluation will be carried out by changing several values from the Hyperparameter configuration including threshold, filtering, batch-size, Intersection over Union (IoU) and number of epochs. The changes will make the object detection model more optimal in carrying out detection and classification. After making several changes to the Hyperparameter configuration used in creating the object detection model, it was found that the batch-size value was influenced by the type of GPU used in order to speed up the process by using the built-in cloud GPU on Google Collab, 16 was found to be the optimal value using the Google Collab cloud GPU.

The best mAP value of 0.9104 was at epoch 125 using the 2nd data taken using your own camera, but when testing with CCTV footage there were objects that were not detected due to differences in the data collection point of view. By using the same data and being able to detect vehicles when testing, the best epoch value is 70, namely 0.8914. Changing epochs affects the occurrence of underfitting and overfitting because epochs are the number of steps in 1 batch size so that if the number of epochs is too few or too many then the model will experience underfitting or overfitting due to less or more iterations.

The values of filtering, Intersection over Union (IoU) and threshold will be used in model testing so that when testing is carried out by changing these values to make an unexpected prediction is displayed. The filtering used in this research is to change the image to a resolution of 640. The IoU used is 0.2, where if there is an IoU value that exceeds 0.2 then the prediction is not displayed. And the threshold used in this research is 0.5, where if there is a prediction below 0.5 then the prediction is not displayed.

Evaluation of the Classification Model by changing the number of training datasets results in an optimal classification model using a dataset of 62 data which gets an accuracy value of 94%. So from the integration of the two models, we get a program that can carry out detection and classification well.

IV. Conclusion

Using the YOLOv7 method as an object detection algorithm in detecting equipment objects for motorists who want to enter the ITERA area through the main gate can detect equipment attributes well. Because the YOLOv7 model used is the most optimal model from several trials in model training. As well as using a dataset that is adapted to the placement of tools that can take images at any time at the main ITERA gate.

Using the naive Bayes method in carrying out classification to obtain a conclusion about the level of safety of motorists who want to enter the ITERA area through the main gate. This can be seen from the many trials of training the naive Bayes model to produce an optimal model. The naive Bayes model can also be integrated well with the YOLOv7 model.

The tests carried out in this research aim to obtain the most optimal model when used in detecting and classifying motorbike rider equipment attributes based on their level of safety. Tests carried out show that the YOLOv7 model used is a model that uses CCTV recording data which can detect vehicles with batch-size 16 and epoch 70 with a mAP value obtained in testing of 89%. The naive Bayes model used is a model with a dataset of 62 data which gets an accuracy value of 94%. So, from the integration of the two models, a program can be obtained that can carry out detection and classification.

Acknowledgment

Thank you to the Faculty of Industrial Technology at Institut Teknologi Sumatera for helping to fund this research.

References

[1] I. G. K. S. Yasa and N. M. R. Sukardi, "Kajian Yuridis Terhadap Ketentuan Wajib Helm Bagi Pengendara Sepeda Motor Berdasarkan Undang-Undang RI No. 22 Tahun 2009 Tentang Lalu Lintas Dan Angkutan Jalan," *J. Ilm. Raad Kertha*, vol. 3, no. 1, pp. 28–47, 2020, doi: 10.47532/jirk.v3i1.167.

[2] F. Feronika, U. Gunadarma, Y. Ratnasari, U. Gunadarma, H. Prabowo, and U. Gunadarma, "Keselamatan Berkendara Sepeda Motor," no. November, pp. 4–6, 2021.

[3] I. F. Ashari, M. Idris, and M. A. Nasrulah, "Analysis of Combination of Parking System with Face Recognition and QR Code using Histogram of Oriented Gradient Method," *IT J. Res. Dev.*, vol. 7, no. 1, pp. 94–110, 2022, doi: 10.25299/itjrd.2022.9958.

[4] I. F. Ashari, M. D. Satria, and M. Idris, "Parking System Optimization Based on IoT using Face and Vehicle Plat Recognition via Amazon Web Service and ESP-32 CAM," *Comput. Eng. Appl. J.*, vol. 11, no. 2, pp. 137–153, 2022, doi: 10.18495/comengapp.v11i2.409.

[5] Muttaqin Muttaqin, Y. Yuswardi, A. Maulidinnawati, A. Parewe, I. F. Ashari, and M. Munsarif, *Pengantar Sistem Cerdas*. 2023.

[6] J. H. Lo, L. K. Lin, and C. C. Hung, "Real-Time Personal Protective Equipment Compliance Detection Based on Deep Learning Algorithm," *Sustain.*, vol. 15, no. 1, 2023, doi: 10.3390/su15010391.

[7] Y. EĞİ, "YOLO V7 and Computer Vision-Based Mask-Wearing Warning System for Congested Public Areas," *Iğdır Üniversitesi Fen Bilim. Enstitüsü Derg.*, vol. 13, no. 1, pp. 22–32, 2023, doi: 10.21597/jist.1243233.

[8] F. Rofii, G. Priyandoko, M. I. Fanani, and A. Suraji, "Vehicle Counting Accuracy Improvement By Identity Sequences Detection Based on Yolov4 Deep Neural Networks," *Teknik*, vol. 42, no. 2, pp. 169–177, 2021, doi: 10.14710/teknik.v42i2.37019.

[9] M. Rizal *et al.*, *Konsep dan Implementasi Internet of Things*, no. Mi. 2023.

[10] Muttaqin, J. Simarmata, M. A. Suryawan, J. Antares, M. N. A. Nur, and I. F. Ashari, *Internet of Things (IOT) Teori dan Implementasi*. 2023.

[11] H. Chen, S. Hu, R. Hua, and X. Zhao, "Improved naive Bayes classification algorithm for traffic risk management," *EURASIP J. Adv. Signal Process.*, vol. 2021, no. 1, 2021, doi: 10.1186/s13634-021-00742-6.

[12] I. F. Ashari, F. A, Daffa, and S. A, "Sentiment Analysis of Tweets About Allowing Outdoor Mask Wear Using Naïve Bayes and TextBlob," *Indones. J. Comput. Sci.*, vol. 12, no. 3, pp. 1092–1103, 2023, doi: 10.33022/ijcs.v12i3.3238.

[13] I. F. Ashari, M. C. Untoro, E. Maharani Sutowo, D. Salsabila, and D. Athiyah Zhabiyan, "Hyperparameter Tuning Feature Selection with Genetic Algorithm and Gaussian Naïve Bayes for Diabetes Disease Prediction," *J. Telemat.*, vol. 17, no. 1, 2022, [Online]. Available: https://www.researchgate.net/publication/365036216.

[14] M. C. Untoro, M. Praseptiawan, I. F. Ashari, and A. Afriansyah, "Evaluation of Decision Tree, K-NN, Naive Bayes and SVM with MWMOTE on UCI Dataset," *J. Phys. Conf. Ser.*, vol. 1477, no. 3, 2020, doi: 10.1088/1742-6596/1477/3/032005.

[15] Z. Zou, K. Chen, Z. Shi, Y. Guo, and J. Ye, “Object Detection in 20 Years: A Survey,” *Proc. IEEE*, vol. 111, no. 3, pp. 257–276, 2023, doi: 10.1109/JPROC.2023.3238524.
[16] C. N. Liunanda, S. Rostianingsih, and A. N. Purbowo, “Implementasi Algoritma YOLO pada Aplikasi Pendeteksi Senjata Tajam di Android,” *J. Infra*, vol. Vol 8, No., pp. 1–7, 2020.
[17] A. Ardiansyah and N. F. Hasan, “Deteksi dan Klasifikasi Penyakit Pada Daun Kopi Menggunakan Yolov7,” *J. Sisfokom (Sistem Inf. dan Komputer)*, vol. 12, no. 1, pp. 30–35, 2023, doi: 10.32736/sisfokom.v12i1.1545.