Publisher: State and Provincial Joint Engineering Lab. of Advanced Network Monitoring and Control (ANMC)

Cooperate:

Xi'an Technological University (CHINA)
West Virginia University (USA)
Huddersfield University of UK (UK)
Missouri Western State University (USA)
James Cook University of Australia
National University of Singapore (Singapore)

Approval:

Library of Congress of the United States

Shaanxi provincial Bureau of press, Publication, Radio and Television

Address:

4525 Downs Drive, St. Joseph, MO64507, USA

No. 2 XueFu Road, WeiYang District, Xi’an, 710021, China

Telephone: +1-816-2715618 (USA) +86-29-86173290 (CHINA)

Website: www.ijanmc.org

E-mail: ijanmc@ijanmc.org

xxwlcn@163.com

ISSN: 2470-8038

Print No. (China): 61-94101

Publication Date: September 28, 2021

Editor in Chief

Associate Editor-in-Chief

Administrator

Associate Editors

Language Editor

Table of Contents

A Review of Lane Detection Based on Semantic Segmentation

Shi Jiaqi
School of Computer Science and Engineering
Xi'an Technological University
Xi'an, China
E-mail: shijiaqi2019@163.com

Zhao Li
School of Computer Science and Engineering
Xi'an Technological University
Xi'an, China
E-mail: 332099732@qq.com

***Abstract*—With the introduction of full convolutional neural product networks, semantic segmentation networks have also been widely used in the field of deep learning. Most lane detection tasks are currently done on the basis of semantic segmentation networks, so the development of semantic segmentation also directly determines the progress of lane detection. Methods: The development of semantic segmentation networks and the performance comparison between different model frames are used to summarize the improvement points as well as the advantages and disadvantages of each approach. Current lane detection network models with good performance based on semantic segmentation networks are described and the performance between the models is compared. Result: The current development of deep learning-based lane detection methods has been very fruitful, with significant improvements in network performance, but they cannot yet be applied in practice. For example, lightweight networks are not stable enough in extracting features, while deep neural networks are too ineffective in real time. Conclusion: Lane detection is of high research value as a key technology for unmanned driving. However, most of the current neural network methods have not been studied from a practical point of view, and there are few methods that use multiple frames as a basis for research. Therefore, in the future how to efficiently use continuous images for lane detection is a key direction to be researched in the future.**

Keywords-Semantic Segmentation; Lane Detection; Deep Learning; Neural Networks

I. INTRODUCTION

Lane detection is an integral step in the field of driverlessness, allowing cars to identify lanes so that vehicles know which direction they are travelling in and avoid them pulling out of their lanes. Lane detection was first done based on the feature approach, which extracts features and fits them based on lane line image features (e.g. colour, shape). However, feature-based methods are susceptible to poor feature extraction due to factors such as light and obstacle occlusion, and the algorithm for fitting lanes requires a range of parameters based on lane characteristics, often with many limitations. Therefore, feature-based algorithms are not suitable for practical applications.

A. Traditional methods

Due to developments in computer vision, lane detection based on model algorithms has been proposed and this method is mainly divided into straight line detection and curve detection. Most algorithms for straight line detection use the Hough transform to perform this method, which equates straight line detection to coordinate statistics, simplifying detection, but frequent coordinate mapping will increase the complexity of the algorithm and cause a reduction in real-time efficiency. A number of improved algorithms have been subsequently introduced to address this algorithm. For example, the maximum length straight line based lane line detection algorithm proposed by Xie Mei et al. This algorithm connects broken straight lines by setting a maximum straight line gap, selecting the maximum length straight line in the vertical direction on either side of the vertical centre of the image, using the maximum length straight lines on each side as edges, binarising the interior of the edges, and subjecting the interior image to Hough

DOI: 10.21307/ijanmc-2021-021

straight line detection, with the line closest to the vertical centre being the final detected lane line. This method greatly reduces the search area, simplifies the difficulty of the algorithm and speeds up detection efficiency.

There are also many different detection methods for bend detection, most algorithms use different line shapes to fit the lanes and rely on different models, the higher the complexity of their models, the better the fit to the lanes, but taking into account the efficiency of the algorithm also requires a streamlined model. Some of the better known lane modelling methods are the B-spline model and the IPM model (Inverse Perspective Transformation Model)[5]. The IPM model converts the monocular vision image into a bird's eye view by applying an inverse perspective transformation, converting the lanes from far to near into parallel lanes, which reduces the difficulty of lane detection. However, this method requires knowledge of the camera's internal parameters, and then determines the transformation matrix for the inverse perspective based on the specific parameters, so when the camera's internal parameters are not known, the inverse perspective transformation model is not very widely used. The B-spline model uses multiple control points to fit the lane lines, also based on parallel perspective technology, and the algorithm is highly accurate but has poor real-time performance; moreover, the method divides the lane lines into multiple areas for separate detection, especially in the presence of false lane lines or lane wear, and the accuracy of the algorithm is not guaranteed, and the lane line jump is serious.

B. *Deep learning methods*

Research on lane detection based on deep learning neural networks has been conducted in recent years, and the results have been a great improvement compared to traditional algorithms. Due to the variability of the practical situation, most scholars have transformed the lane detection problem into a semantic segmentation problem. Convolutional neural networks have had great success in image detection and recognition, so convolutional-based semantic segmentation networks also have a wide range of applications in lane detection. The laneNet network proposed by Davy [1] et al. converts lane processing into an end-to-end instance segmentation problem, using a lightweight ENet network as the main structure and adding instance segmentation branches to classify different lanes into different categories. XingGang Pan[2] et al. proposed a spatially based deep neural network SCNN (Spacial CNN), which was trained to classify the network for the poorly conditioned dataset CULane, and the network performance was substantially improved in lane detection compared to the traditional convolutional network. The ENet-SAD[3] network is based on the lightweight neural network model ENet incorporating elements of SAD, Self-Attention Knowledge Distillation, which has 20 times fewer parameters, runs and is 10 times faster and more accurate than the state-of-the-art SCNN. While domestic scholars have paid much attention to the diverse road conditions, an improved YOLOv3 model was proposed by Zhang Xiang [4] to improve the adaptive and accuracy problems of lane detection technology in complex road environments, where complex road problems refer to road potholes, rugged mountain roads and other problems. A multi-scale MFCN model was proposed by Shuaihua Wang et al. to solve the lane line sample inhomogeneity problem, using a weighted loss function to solve the lane line inhomogeneity problem. For sharp turns, over curved lanes, CurveLane-NAS, a lane sensitive architecture search framework combining NAS with curved lane detection algorithms proposed by Huawei Noah's Ark Lab and Sun Yat-sen University [6], can automatically capture long-distance coherent and accurate short-distance curve information to solve the problem of curved lane detection.

The neural network methods described above are all based on semantic segmentation networks for end-to-end lane line detection, i.e. the lane detection problem is converted into a multi-category segmentation problem where each lane belongs to one category, which enables the end-to-end training of a well-classified binary graph. This paper therefore focuses on describing the current state of development of lane detection based on semantic segmentation networks.

II. Semantic Segmentation Network

There are many applications of neural networks in the field of computer vision, such as image classification [11], target detection [12], semantic segmentation [14], and instance segmentation [13]. One important problem in computer vision is the semantic segmentation network, as its work is much more complex than the classification and detection tasks. Semantic segmentation of images means that each pixel of the input image is assigned a semantic category to it, thus obtaining a dense classification for each pixel. That is, it requires learning the contour of the object, the location of the object and the class of the object from high-level semantic information and local location information, and thus scholars in general view the semantic segmentation problem as a pixel-level target segmentation.

Traditional semantic segmentation is generally classified into threshold-based segmentation methods [8], region-based segmentation methods [9], edge-based segmentation methods [10] and so on. The threshold segmentation method is one of the commonly used segmentation techniques, which in essence automatically determines the optimal threshold value based on certain criteria and uses these pixels according to the grey level i n order to achieve clustering. Region-based segmentation is a segmentation technique based on the direct search for new regions and can be divided into two basic extraction methods: region growing and region splitting and merging. Region growth is based on individual pixel points, which are aggregated together to form regions with similar features, and is computationally simple and works well for uniformly distributed images. Region splitting and merging starts from the overall image and obtains each sub-region by splitting between pixel points, the quadtree decomposition method is a typical representative method. Edge detection-based segmentation methods segment images by detecting the edges of different regions. The simplest edge detection method is the parallel differential operator method, which uses the nature of discontinuous pixel values in adjacent regions and uses derivatives to detect edge points. Most traditional methods work by extracting low-level semantics of the image, such as size, texture, colour, etc. In complex environments, the response capability and accuracy is far from adequate.

With the development of deep learning, the proposal of convolutional neural networks has allowed significant progress to be made in combining semantic segmentation and neural networks. Because of the powerful generalisation ability of convolutional networks to acquire image features, they have shown excellent performance in different areas of image and video such as image classification, target detection, visual tracking and action recognition. The following subsections describe the development of semantic segmentation networks based on deep learning.

A. Derivation of the semantic segmentation network model

A turning point in the development of semantic segmentation based on deep learning was the FCN, a fully convolutional neural network for end-to-end segmentation, proposed by Jonathan Long [14] et al. in 2014, when a major breakthrough in semantic segmentation was achieved. It upsampling the local information loss caused by the convolutional neural network with a deconvolution operation that restores the feature map to the original image size, hence the current general semantic segmentation network architecture is an encoder-decoder structure. Where the encoder is usually a pre-trained classification network, the task of the encoder is to semantically project the discriminable features learned by the encoder onto the pixel space to obtain dense classification.

A number of scholars have since proposed a number of sophisticated network frameworks, but most have been studied on the basis of fully convolutional networks. In this paper, we only discuss semantic segmentation networks that are applicable to lane detection, and the research in recent years is shown in the following Table 1:

TABLE I. COMPARISON OF IMAGE SEMANTIC SEGMENTATION NETWORKS

Mothods	*Features*	*Advantages*	*Disadvantages*
FCN[14]	Proposes novel end-to-end network architecture ; Encoder-decoder architec-ture ; Fully connected output classification.	Images of any size can be split.	The large number of parameters and the pooling opera-tion caused a loss of spatial information in the images and a low accuracy rate.
SegNet[15]	Symmetrical Encoder-Decoder architecture ; up-sampling to recover im-age size at the decoding stage using unpool-ing; full convolutional layer output classification.	The small number of parameters compared to FCN maintains the integrity of the HF information.	The computational effort is too large to meet the real-time requirements of lane detection. The up-sampling operation also loses adjacent informa-tion.
Unet[16]	Symmetrical structure; co-nnects each stage to the encoder feature map with the upsampled feature map of the decoder.	Can be trained end-to-end from very small data sets; fast.	More suitable for segmentation of medical images
ENet[17]	Consisting of Bottleneck mod-ules; with a large encoder-small decoder st-ructure.	Greatly reduces the nu-mber of parameters and floating point operations, takes up less memory and has high real time performance.	Increases the number of calls to the kernel function; not very precise and unstable results.
PSPNet[18]	Improving ResNet structures using null conv-olution ; A pyramid pooling module has been ad-ded.	The segmentation acc-uracy exceeds that of models such as FCN, DPN and CRF-RNN.	Obscured situations between targets are not handled well and the edges are not seg-mented accurately enough.
ERFNet[19]	ENet network improve-ments; the adoption of factorized convolutions;	Non-bottleneck is more accurate to bottleneck.	High calculation volume compared to Enet.
DeepLab V3+[20]	Uses a modified version of Xception as the base network; uses atrous[19] convolutional kernels.	More accurate segmentation of target edges; considers global information, eliminates noise interference and imp-roves segmentation accuracy.	The model does not run at a high speed and has a high storage space requi-rement.
FPN[21]	Combining FCN and Mask R-CNN[13] using rich multi-scale features.	Semantic segmentation and instance segmentation tasks can be solved simultaneously.	Increased inference time; larger memory footprint; use of image pyramids only in the testing phase.

B. Limitations of semantic segmentation networks

While semantic segmentation web techniques are currently achieving good segmentation results, there is currently no universal algorithm that is applicable to all domains. In practical segmentation tasks, it is necessary to choose the segmentation method flexibly depending on the application scenario, and in some cases it is even necessary to use a combination of segmentation methods to obtain the best results. Therefore semantic segmentation still has some challenges: 1) network training requires a large dataset and pixel-level image quality is difficult to guarantee due to the extensive use of strongly supervised segmentation-based methods that rely on manual data tagging and are less adaptable to unknown scenes; 2) segmentation of small-sized targets is not accurate enough; 3) segmentation algorithms are computationally complex; and 4) interactive segmentation cannot be achieved, which hinders the implementation, application and promotion of segmentation techniques.

III. SEMANTIC SEGMENTATION NETWORK BASED LANE DETECTION METHOD

With the rapid development in the field of unmanned vehicles, scholars have proposed many sophisticated lane detection network models in recent years. The current mainstream lane line detection networks are basically based on semantic segmentation networks to complete the detection task, so this subsection focuses on the current better performance of the semantic segmentation-based lane detection network models.

A. *Related methods*

1) *SCNN*

In a classical CNN, each convolutional layer takes input from the previous layer, applies convolution and nonlinear activation and then passes the output to the subsequent layers. XingGang Pan [2] et al. proposed the SCNN model based on CNN with a spatial attention element. The SCNN model views the rows and columns of the feature map as layers, and also uses convolution plus nonlinear activation to achieve a spatially deep neural network. This allows spatial information to be propagated between different neurons in the same layer, enhancing spatial information and thus being particularly effective for identifying structured objects.

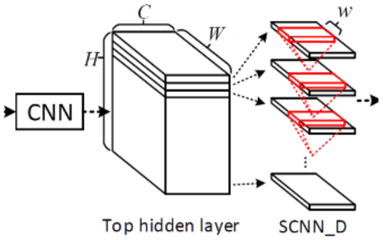


Figure 1. SCNN_D module

As shown in Figure 1, the SCNN_D module, where SCNN is applied to a 3D tensor C×H×W, with C, H, W representing the number of channels, length and width respectively. To achieve spatial information transfer, the tensor is sliced into H slices, the first slice is sent to the convolution layer of size C×W, and the output of this slice is summed to the next slice as a new slice. Then the next slice continues to apply convolution until all slices have been processed and input to the next module. Three similar modules follow, each convolving the feature map from a different direction in three dimensions. SCNN does not acquire global elements when passing information, but passes them sequentially, thus simplifying the structure of information passing and accelerating the efficiency of the model.

2) *LaneNet*

LaneNet [1], on the other hand, transforms the lane detection problem into an instance partitioning problem, where each lane line forms a separate instance, but all belong to the lane line category. The authors propose an end-to-end multitasking network with branching struc-ture, consisting of a lane embedding branch and a lane embedding branch. One of the lane segmentation branches outputs two categories: background and lane lines, repre-sented by a binarised segmentation map; the corresponding pixels of each lane line are concatenated to construct the binarised segmentation map, which has the advantage that the network can predict the lane position even if the lane lines are obscured. The lane embedding branch further separates the segmented lane lines into different lane instances. This branch is based on the one-shot method for distance metric learning, which is easily integrated into standard feedforward networks and can be used for real-time processing. The network structure is shown in Figure 2:

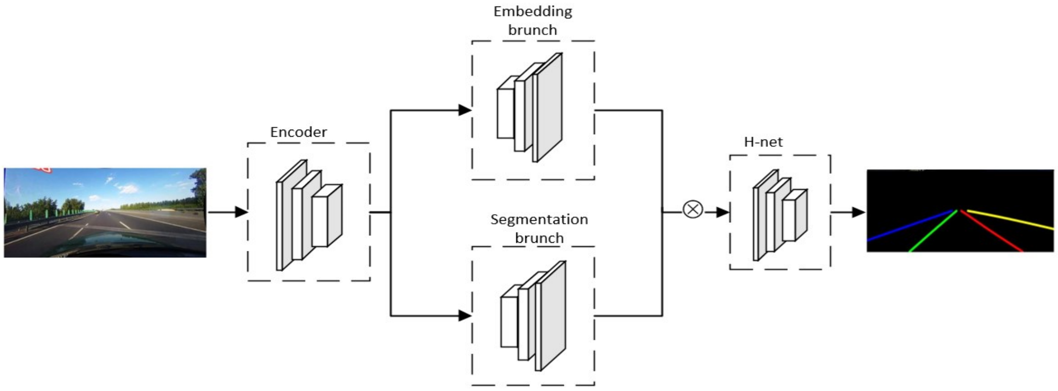


Figure 2. LaneNet network framework

The network then adopts a type of network called H-Net network for predicting the transpose matrix H, which solves the error caused by the traditional transpose matrix in the case of uneven ground planes such as slopes. Finally, the CNN learns the transformation matrix to perform an angular transformation to make the lane lines parallel in order to fit a reliable lane to different pictures or horizon transformations in the pictures.

3) DCNN+DRNN

The lane line features consist of continuous lines and feature extraction by the current frame alone is not sufficient information representation. Therefore QIN Zou [7] et al. proposed lane detection by successive frames, where the information of each frame is extracted by the CNN module and the CNN of multiple successive frames maintains temporal continuity and is fed to the RNN module as feature learning and lane line detection. CNNs have the advantage of being able to process a large number of images, extracting the input image into a small-sized feature map through operations such as convolution and pooling. RNN has the advantage of continuous signal processing, temporal feature extraction and integration, and can be used for lane detection and prediction.

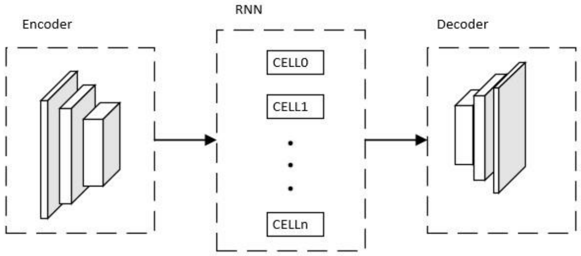


Figure 3. DCNN+DRNN network framework

To fuse CNN and RNN networks into an end-to-end training network, the authors used the classical network structure of semantic segmentation, with the encoding-decoding structure as the main framework. The images are fed into the coding module to obtain a temporal feature map; the feature map is then passed as input to the RNN network to predict lane line information; the output of the RNN is then passed back to the decoding module to obtain a probability map of lane prediction. Experiments have shown that this network performs better than a network based on single-frame feature extraction, and has more stable performance especially under some complex road conditions. And the longer the continuous input sequence the better the performance, further proving that multi-frame images are more helpful than single-frame images.

B. Performance comparison

Deep learning based lane line detection requires a large amount of well labeled lane line training data to train the convolutional neural network model. Early lane line datasets were generally small in size and the scenarios were relatively homogeneous for deep learning lane line detection and the amount of data was too small to achieve a good model. With the rise of lane line detection technology, lane line datasets have evolved rapidly. The CULane dataset contains 133,235 images, of which 88,880 are in the training set, 9,675 in the validation set and 34,680 in the test set. It includes urban, rural and motorway scenarios as well as a variety of weather, heavy lane shading, lane wear and tear, etc. Road conditions are complex and variable, so many networks use the CULane dataset to reflect the performance and strengths and weaknesses of the network.

The key indicator of lane detection is the accuracy rate. Generally, the calculation of the accuracy rate first requires the calculation of the overlap between the true value of the lane T and the predicted value H as a percentage of the true value IoU, and the calculation formula is shown in (1). If IoU is greater than the set threshold, the lane line is considered to be accurately detected and the number of predicted lanes correct TP is added to 1, otherwise the number of predicted lanes incorrect FP is added to 1; the formula for calculating the accuracy rate is shown in (2).

$$\mathrm{IoU} = \frac{H \cap T}{T} \times 100\% \tag{1}$$

$$\mathrm{Precision} = \frac{TP}{TP + FP} \tag{2}$$

To better illustrate the performance comparison between networks, the results of the above network performance are presented in Table 2 below using the CULane dataset the results in the table show that the accuracy of the DCNN+DRNN hybrid neural network is basically better than the other network models in various scenarios. The ability to extract feature information in congestion and bends is slightly weaker than the LaneNet network. As the hybrid neural network model extracts features in multiple consecutive frames, generally the slow movement of vehicles in congestion or large angular shifts in road direction cause long periods of time when road features cannot be extracted or are too different from the features of the previous consecutive frames, which can cause bad results.

TABLE II. ACCURACY COMPARISON OF LANE LINE DETECTION IN DIFFERENT SCENES OF CULANE

Methods	Normal	Crowded	Night	NoLine	Shadow	Arrow	DazzleLight	Curve	Crossroad	Total
SCNN	0.906	0.696	0.661	0.434	0.669	0.841	0.585	0.644	0.532	0.716
LaneNet	0.921	**0.708**	0.714	0.563	0.697	0.850	0.635	**0.746**	0.591	0.742
DCNN+DRNN	**0.984**	0.652	**0.797**	**0.724**	**0.840**	**0.852**	**0.774**	0.731	**0.787**	**0.782**

Accuracy, a key metric for lane detection network performance, is then real-time. Real-time performance is evaluated in terms of processing speed and the amount of memory consumed, but processing speed is relatively more important. As can be seen from the Tabel 3 below, the LaneNet network has the best real time performance, as it uses a lightweight network as the base framework, so the processing speed is significantly better than the other frameworks.

TABLE III. COMPARISON OF DETECTION SPEED OF VARIOUS NETWORK MODELS

Methods	Time(ms)	fps
SCNN	42	23.8
LaneNet	**19**	**52.6**
DCNN+DRNN	58	17.2

IV. CONCLUSION

A. *Summaries*

This paper focuses on the development of lane detection tasks in terms of semantic segmentation. There is still a lot of room for development of semantic segmentation networks, both in terms of the cost of training and the complexity of computation, which are not yet up to the requirements of practical applications. The current lane detection network is basically based on the semantic segmentation network, so the development of the semantic segmentation network has a direct impact on the progress of lane detection. Although deep learning-based lane detection is more adaptable to unknown environments than traditional methods, it is still unable to achieve both accuracy and real-time performance. Although the LaneNet network uses a lightweight network, the detection results are not as good as compared to the DCNN+DRNN hybrid neural network in some poor road conditions. And although the hybrid neural network outperformed the other models in all aspects, the processing speed was clearly not up to the practical requirements.

B. *Prospects*

Although lane detection technology based on deep learning methods is still in the development stage and is still some distance away from practical applications, the trend of development will become faster and faster. There has also been a great deal of progress in deep learning methods, but most of them are based on feature extraction on a single frame basis, and there are still few methods that have been studied on a video basis, while in practical applications the images captured by the camera are often continuous. Therefore, in the future the focus of research on lane detection tasks should be on how to efficiently segment and detect lane lines using continuous frames. In the future deep learning based lane line detection will soon be used in practice.

REFERENCES

[1] Neven D, Brabandere B D, Georgoulis S, et al. Towards End-to-End Lane Detection: an Instance Segmentation Approach [J]. IEEE, 2018.

[2] Pan X, Shi J, Luo P, et al. Spatial As Deep: Spatial CNN for Traffic Scene Understanding. 2017.

[3] Hou Y, Ma Z, Liu C, et al. Learning Lightweight Lane Detection CNNs by Self Attention Distillation [J]. 2019

[4] An improved YOLOv3 model based on skipping connections and spatial pyramid pooling [J]. Systems Science & Control Engineering, 2021, 9(S1).

[5] Chun-yang CHENG, Min LI, Xue-wu ZHANG, Yu-bo XIE, Yan XIANG, Jin-bao SHENG. A Lane Detection Algorithm under Complex Scenes [A]. Advanced Science and Industry Research Center. Proceedings of 2017 2nd International Conference on Computer, Mechatronics and Electronic Engineering(CMEE 2017)[C]. Advanced Science and Industry Research Center: Science and Engineering Research Center, 2017:5.

[6] Xu H, Wang S, Cai X, et al. CurveLane-NAS: Unifying Lane-Sensitive Architecture Search and Adaptive Point Blending [J]. 2020.

[7] Q. Zou, H. Jiang, Q. Dai,et al. Robust Lane Detection From Continuous Driving Scenes Using Deep Neural Networks [J]. 2019.

[8] Zhihuan Wu,Yongming Gao, Lei Li, Junshi Xue,Yuntao Li. Semantic segmentation of high-resolution remote sensing images using fully convolutional network with adaptive threshold [J]. Connection Science, 2019, 31(2).

[9] Vadim Romanuke. A Prototype Model for Semantic Segmentation of Curvilinear Meandering Regions by Deconvolutional Neural Networks [J]. Applied Computer Systems, 2020, 25(1).

[10] Chu He, Shenglin Li, Dehui Xiong, Peizhang Fang, Mingsheng Liao. Remote Sensing Image Semantic Segmentation Based on Edge Information Guidance [J]. Remote Sensing, 2020, 12(9).

[11] Lin T Y, Dollar P, Girshick R, et al. Feature Pyramid Networks for Object Detection[C]// 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE Computer Society, 2017.

[12] Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation[C]// CVPR. IEEE, 2014.

[13] He K, Gkioxari G, P Doll ár, et al. Mask R-CNN [J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2017.

[14] Long J, Shelhamer E, Darrell T. Fully convolutional networks for semantic segmentation[C]// 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2015.

[15] Badrinarayanan V, Kendall A, Cipolla R. SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation [J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2017:1-1.

[16] Ronneberger O, Fischer P, Brox T. U-Net: Convolutional Networks for Biomedical Image Segmentation[C]// International Conference on Medical Image Computing and Computer-Assisted Intervention. Springer International Publishing, 2015.

[17] Paszke A, Chaurasia A, Kim S, et al. ENet: A Deep Neural Network Architecture for Real-Time Semantic Segmentation [J]. 2016.

[18] ZHAO H S, SHI J P, (}I X J, et al. Pyramid network[C]//Proceedings of the 20171EEE Conference on Computer Vision and Pattern Recognition. New York: IEEE Press, 2017:2881-2890. DOI: 10.1109/CVPR.2017.660.

[19] Romera E, Alvarez J M, Bergasa L M, et al. ERFNet: Efficient Residual Factorized ConvNet for Real-Time Semantic Segmentation [J]. IEEE Transactions on Intelligent Transportation Systems, 2017, PP(1):1-10.

[20] CHEN L C, ZHU Y, PAPANDREOU G, et al. Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation [EB/OL]. [2018-08-09].hrrps://arxiv.org/pef/1802.0261v1.pdf.20.FPN

[21] Lin T Y, Dollar P, Girshick R, et al. Feature Pyramid Networks for Object Detection[C]// 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE Computer Society, 2017.

SIS-CNN: Semantic Image Segmentation Using Convolutional Neural Networks

Muhammad Adeel Ahmed Tahir
School of Computer Science and Engineering
Xian Technological University
Xian, China
E-mail: adikhan0313@gmail.com

Zaryab Shaker
School of Computer Science and Engineering
Xian Technological University
Xian, China
E-mail: zaryabkhan0346@gmail.com

Feng Xiao
School of Computer Science and Engineering
Xian Technological University
Xian, China
E-mail: xffriends@163.com

***Abstract*—Semantic image segmentation is a vast area of interest for computer vision which has gained exceptional attention from the research community. It is the process of classifying each pixel in respective category. In this paper, we exploit the problem of scene understanding and perform the segmentation by combining different classification models as a feature encoder and segmentation models as a feature decoder. All of the experiments were performed on Camvid dataset. It covers a wide range of real-world applications such as autonomous driving, virtual/augmented reality, indoor navigation, etc.**

Keywords-Semantic Segmentation; Computer Vision; Scene Understanding; Classification Model; Segmentation Model

I. Introduction

Semantic segmentation [1] is the process of alloting class labels to each pixel in an image. Pixel-wise labels provides us better descriptions of images than bounding box labels. Concluding such labels is a much more challenging task because it involves extremely complex structured prediction problem. Semantic image segmentation [1] (pixel-level classification) is an immense area of interest for computer vision, machine learning [2], and deep learning [3] researchers with many challenges. It has a wide array of practical applications like remote sensing, autonomous driving, indoor navigation, video surveillance and virtual or augmented reality systems etc.

Nowadays Deep Learning techniques [4] provide state-of-the-art performance for image segmentation and classification as well as for detection tasks and captioning using Convolutional Neural Network models and have been mainly accelerating the recent breakthroughs in semantic segmentation using different combinations of CNN models such as VGGNet [5], AlexNet [6], and ResNet [7].

VGG[5] is an advanced object-recognition convolutional neural network model that supports up to 19 layers pre-trained on ImageNet [8] (achieves 92.7% accuracy) and performs efficiently on many datasets outside of ImageNet [8]. ResNet [7] is a deep neural network that has 150+ trainable layers. The modal achieves the highest accuracy in the 2015 ImageNet [8] dataset Challenge. U-Net [9] is a Convolutional Neural architecture designed to deal with biomedical images to solve the problem i-e what and where.

In this paper, we proposed a Segmentation Architecture by combining the two models i-e base model [5], [7] with our segmented model [9], [12] for segmentation. We use our base model as an object feature extractor and use the preceeding

DOI: 10.21307/ijanmc-2021-022

segmentation model to segment the images based on extracted features. We use different models with the implementation of an encoder-decoder [14] having skip architecture [10] for segmenting the boundaries accurately.

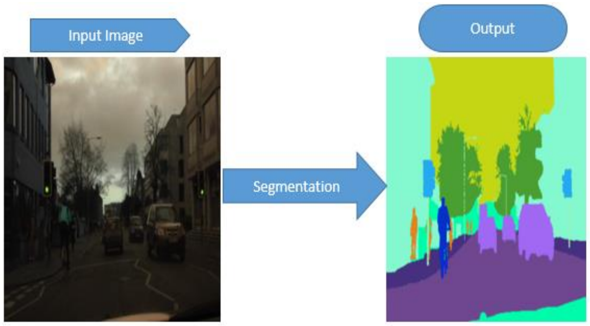


Figure 1. Semantic Image Segmentation

The second part of this paper involves a short survey for segmentation with CNN models. The third part describes the proposed methodology of our framework, the fourth part involves experiments results and graphs. Conclusion is in the fifth part and references are drawn in the last.

II. LITERATURE SURVEY

In recent research of computer vision and pattern recognition, CNN [11] capabilities are highlighted which solve challenging tasks like segmentation [13] and classification [23]. Recent progress in semantic segmentation are mainly enhanced by powerful DNN architectures [9], [12], following by the ideas of FCN's [13]. Different architectures have been developed in this context. Some of the deep learning-based works for semantic segmentation include Fully convolutional networks [13], Encoder-decoder based models [14], Multi-scale and pyramid network-based models [15], Dilated convolutional models [16], and DeepLab family [17], Recurrent neural network-based models [18], Attention-based models [19], etc. All of these approaches have in common that they generally rely on the powerful feature extraction provided by CNN's [5], [6], [7]. Following is a brief study of some of our concerned techniques.

In 2014, Long and Shelhamer et al. [13] presented the novel approach of FCNs for semantic segmentation. The approach represented the state-of-the-art in semantic segmentation and has since set the standard for future directions. FCNs [13] are trained end-to-end, provide a pixel-to-pixel prediction. They also use skip architectures [10] to combine semantic and appearance information. The authors have demonstrated 62.2% mean pixel (IU) on the PASCAL VOC 2011 dataset [24].

The work by Long and Shelhamer et al. [13] builds off of the concept of CNNs pioneered by Matan et al. [20], and the concept of jets pioneered by Koenderink and Van Doorn [21]. In 1991, Matan et al [20]. Used CNNs for recognizing an unconstrained handwritten multi-digit string. They presented a feed-forward network architecture. This is an addition to the work on recognizing isolated digits. In 1987, Koenderink and Van Doorn [21] used local jets to give rich representations of local geometry and semantics with filters on multiple scales. Since the work of Long and Shelhamer et al. [13], several other methods have been explored to improve the performance of semantic segmentation. [1]

In 2017, Chen and Papandreou [17] incorporated probabilistic graphical models in the form of fully Conditional Random Fields (CRF) to overcome poor localization. They proposed “DeepLab” system by applying the ‘atrous convolution’ with upsampled filters trained on image classification to the task of semantic segmentation for dense feature extraction and further extend it to atrous spatial pyramid pooling. They also combine ideas from DCNNs [22] and FCRFs [23] to produce semantically precise predictions and comprehensive segmentation maps. The proposed technique significantly advances the state-of-art in several challenging datasets, including PASCAL VOC 2012 [24] semantic image segmentation benchmark, PASCALContext [25], and Cityscapes [25] dataset.

Later, Zheng and Jayasumana [26] showed that unpacking dense CRFs into individual computations and joining them to the network yields further improvement. They combine the strengths of CNNs and CRFs [26] in a single deep network. Their formulation fully integrates CRF-based probabilistic graphical modeling with emerging deep learning techniques that are capable of passing on error differentials from its outputs to inputs during back-propagation-based training of the deep network while learning CRF [26] parameters. The approach achieves a state-of-the-art on the popular Pascal VOC segmentation benchmark [24].

In 2015, Noh et al [27]. demonstrate a novel semantic segmentation algorithm by learning a deconvolution network that incorporates a learned deconvolution network for even better performance. Since coarse-to-fine structures of an object are reconstructed progressively through a sequence of deconvolution operations, it helps to generate dense and precise object segmentation masks. They further proposed an ensemble approach, which combines the outputs of the proposed algorithm and FCN-based [13] method, and achieved substantially better performance with the help of characteristics of both algorithms.

Losing the context information for images during segmentation was a problem until it was addressed by Yuantao Chen et.al [28] in the paper “improving semantic image segmentation based on feature fusion model”. They proposed a feature fusion model with context features layer-by-layer. Firstly, an image pyramid is formed by pre proceesing the original images. Secondly, an image pyramid is inputted into the network structure by the initialization of feature fusion and expanding receptive fields using Atrous Convolutions. Finally, the score map of the feature fusion model had been calculated and sent to the conditional random field for further processing to optimize results. The approach on the PASCAL VOC 2012 and PASCAL Contex [25] t datasets had achieved better IU than the state-of-the-art works. The method has about 6.3% improved to the conventional methods.

III. PROPOSED METHODOLOGY

We started the problem by taking the camvid dataset [25]. Our segmentation task was carried out by a combining two different models. One is used as a base model and the other one is the segmentation model. Our base model is a feature extractor for a given image and pre-trained on the ImageNet [8] dataset. We fine-tuned our base model on our relative dataset and use it as an encoder [14] part for our segmentation task. We use the skip architecture [10] by taking the output from our concern layers.

The second one is our segmentation model which is used as the decoder [14] part for our architecture. It takes the output from certain layers in our base model through skip architecture [10] as its input. Then the segmentation model segments the image based on the features extracted by the base model.

A. Base Model

CNNs shows a state-of-art for image classification and recognition because of its high accuracy. The CNN follows a hierarchical model [29] that works on building a network, like a funnel, and finally gives out a fully-connected layer where all the neurons are connected to eachother and the output is processed. Our base model is used as a feature extractor having Input size of 224x224 pre-trained on ImageNet [8] with 1000 classes. We are taking classification features by removing fully connected nodes and fine-tune the model on specific layers. We transform the fully connected layers into convolution layers to produce a classification heatmap [13].

We get the image at the input layer and then initialize the weights to avoid layer activation outputs from exploding or vanishing during a feed-forward propagation [30]. After weight initialization, all of the weights ‘w’ multiplied by input ‘x’ are summed up and add a bias of 1 to allow units to learn an appropriate threshold. (1).

$$y = [x_1w_1 + x_2w_2 + . \dots \dots . + x_nw_n] + b$$

$$y = \sum x.w + \beta \qquad (1)$$

We add zero paddings and apply 3x3 kernels with a stride of 2 and apply max-pooling. A trick of ‘shift and stitch’[13] in which the values are being max-pulled after doing the shifting and then we stitch the results into the original image. After that there implies a relu activation function ‘R’(2).

$$R= \max(0,y)$$

$$\hat{y} = P(\psi) \qquad (2)$$

Base model works as an encoder [14] for our architecture. Encoder-Decoder [14] module works as a backbone for semantic segmentation tasks. The encoder extracts features from the input image which is used to produce segmentation output. We get the abstract representations via downsampling. In downsampling, we decrease the number of pixels by getting only the pixels with features. This is done because we are facing memory limits on computer and to reduce processing time. The result of using a pooling layer and creating downsampled or pooled feature maps is a summarized version of the features detected in the input.

B. Skip Connections improve Segmentation Details

We use skip connections [10] in our architecture, skip some layers in the neural network and feed the output of one layer as the input to the other layers instead of just passing to one next layer. By using a skip connection [10], we provide an alternate path for the gradient (with backpropagation). It makes it easier to estimate good weight values for the architecture to obtain better generalization performance. After cascading a set of CNN weights 'w', biase 'b', and non-linear layers to the input 'x', we extract image features 'x_f' from each 'n' layer is defined by

$$x_{fn} = \widehat{y_n} \qquad (3)$$

all outputs 'x_{fn}' are passed to the segmentation architecture through Skip Connections[10].

C. Segmentation Model

Image segmentation with CNNs, involves feeding segments of an image as input to the segmentation model [9], [12], which labels the pixels. Our segmentation module consists of different CONV layers that receive the input from different levels of the base model [5], [7]. It involves upsampling of the images via deconvolution (also known as a transposed layer).

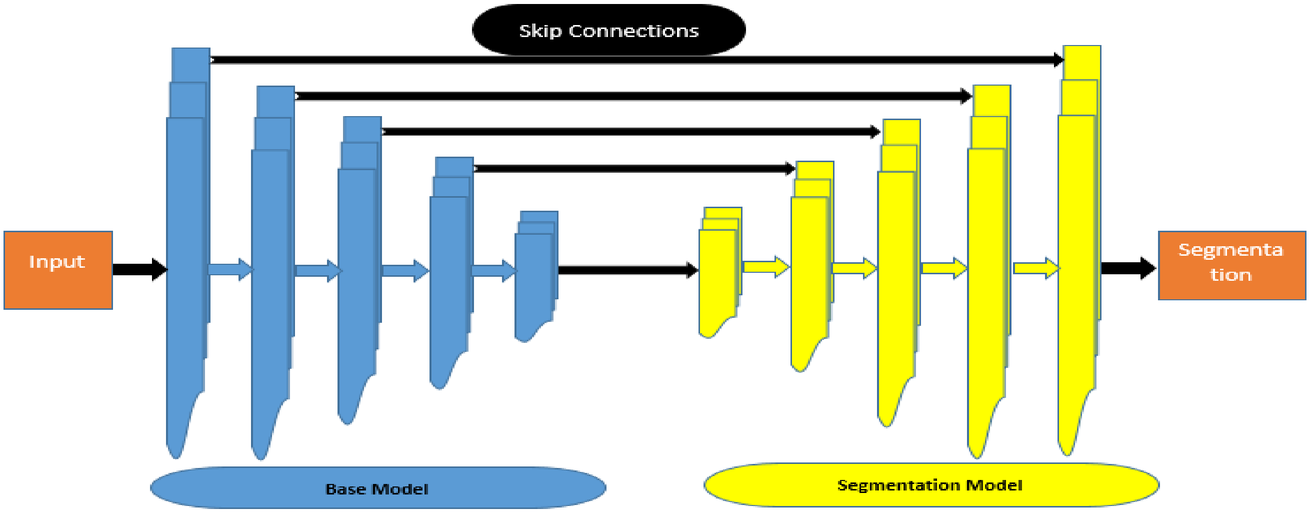


Figure 2. An encoder- decoder based CNN architecture with different combinations of different models for semantic segmentation.

Let 's_n' be the decovolution layer in our segmentation model receiving the inputs from base model. Deconvolution [27] layers need to be stacked very deeply which increases computations and memory allocation. So we use 1x1 conv [31] where the stride is 1 without bias. It gives us faster computation with less information loss by reducing the dimensions of the previous layer and also adds more non-linearity to enhance the potential representation of the network. Input samples 'x_{fn}' are average pooled and passed through the 1x1 conv layer [31]. Applying batch normalization, we regularized our model to avoid the need of dropout. It also reduce the training epochs and get higher accuracy. This is done before utilization of Relu activation function. So,

$$s_n = \chi o\nu\varpi(x_{fn}) \quad (4)$$

After that there is a concatenation layer ‘C’ which concatenates all the inputs receives from previous model in a linear form as well as from skip connections [10] which are then concatenated and pass through 1x1 conv layer with batch normalization and relu function ‘R’.

$$X = \chi o\nu\varpi\left(\sum s_n\right) \quad (5)$$

It passes the results to output layer ‘Z’ having a softmax activation function ‘SOF’.

$$Z = \Sigma O\Phi(X) \quad (6)$$

Where ‘Z’ is the segmented image.

IV. EXPERIMENTS AND RESULTS

1) *Dataset:* We are using the Camvid [25] dataset consists of 701 original images of 360x480p. The images are divided into 3 sets *having* 367 training images, 233 test images and 101 validate image. We make annotations for each image in the original dataset. After that, data augmentation is performed for training set. The images are flipped vertically and horizantly, make 2 more images for each image. So the total number of training images is 1,101 with RGB colors.

2) *Models:* Pre-trained classification and segmentation models are fine tuned and combined which works as an encoder and decoder part for our architecture. By using transfer learning, we adopt VGG[5] and Resnet[7] with pre-trained weights on ImageNet as our base (encoder) module whereas U-net[9] and PSP-net[12] as our segmentation (decoder) module. We are training our model based on the combination of VGG_U-net, VGG_PSP-net, ResNet_U-net, ResNet_ PSP-net.

3) *Training Setup:* We are doing traing against loss and accuracy. Our loss is difference between predicted and actual value defined by ‘L’

$$\Lambda = (\hat{y} - \psi)$$

Weights are updated according to the following relation for backpropogation to minimize the loss

$$w_n = w_o - n \frac{\delta L}{\delta w_o}$$

Where ‘n’ is learning rate, n=$2e^{-4}$, ‘w_n’ is the new weight and ‘w_o’ is the old weight. Batch normalization is used to avoid the need of dropout and serves us to regularize the model. We use adam optimizer to minimize the loss value. We do training for only 5 epochs due to limited resouces with 512 steps on each epoch. After every epoch the model will save its learned weights and it will also validate itself by using the given validation set. It takes around 13 seconds per step (6904s per epoch). During training the model, after each epoch the model will evaluate the performance based on the validation set. To check the overall performance of the model we use the test set with newly images for model prediction.

A. RESULTS

Results shown below describes about the model performance. Due to less resources these results were carried out on simple laptop(core i7, 8gb ram).Results are satisfactory as the model was trained only for 5 epochs.

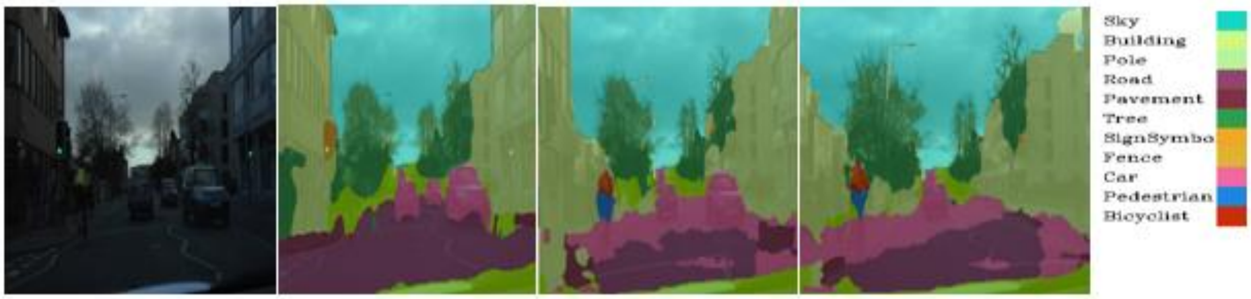


Image 1 PSPNET RESNET-PSPNET RES-UNET Labels

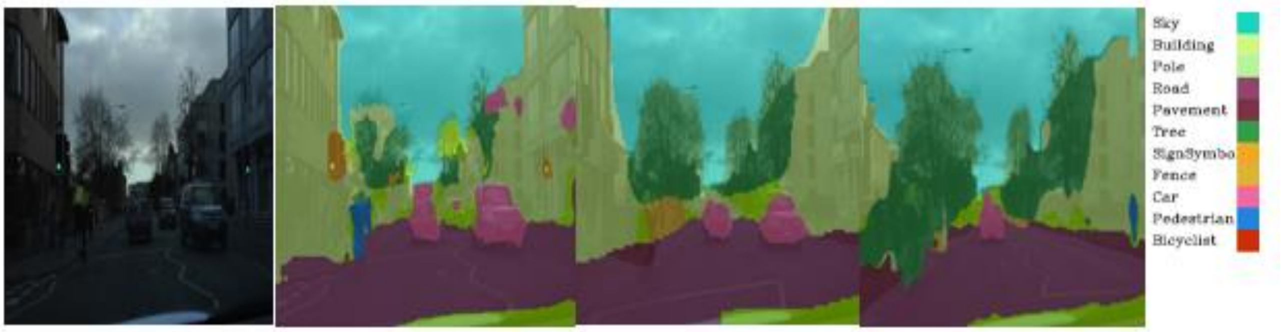


Image 1 UNET VGG-PSPNET VGG-UNET Labels

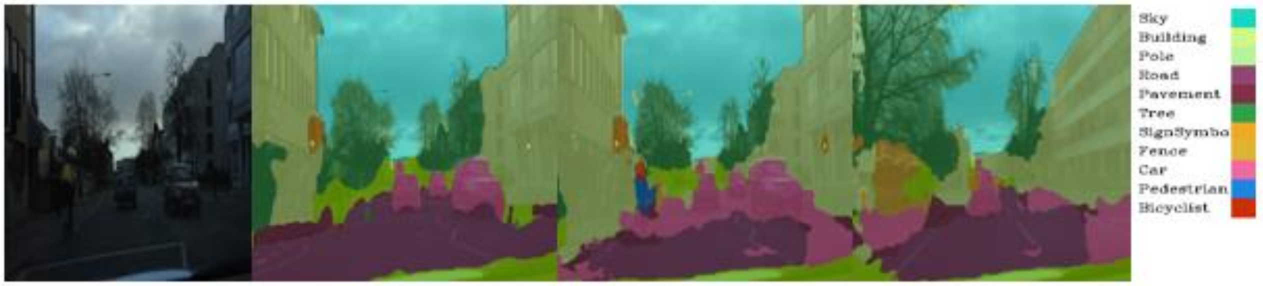


Image 2 PSPNET RESNET-PSPNET RES-UNET Labels

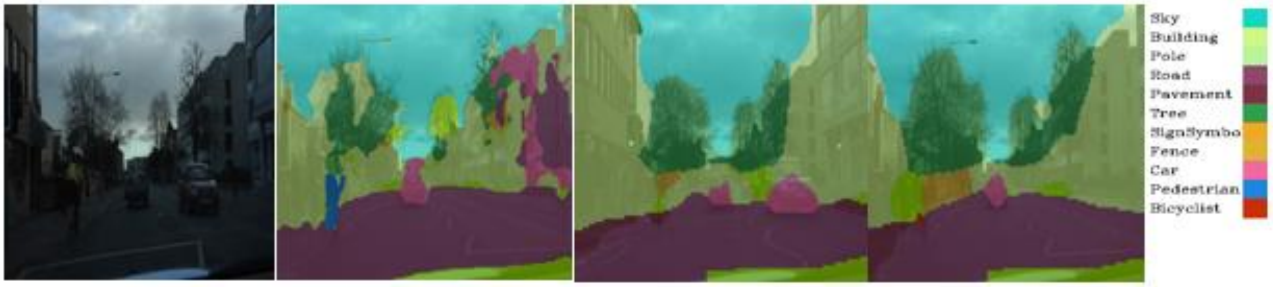
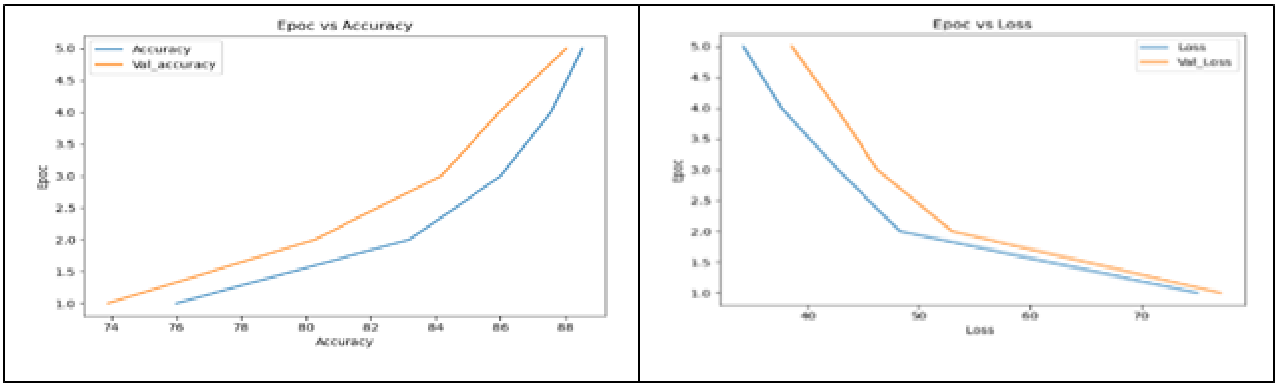


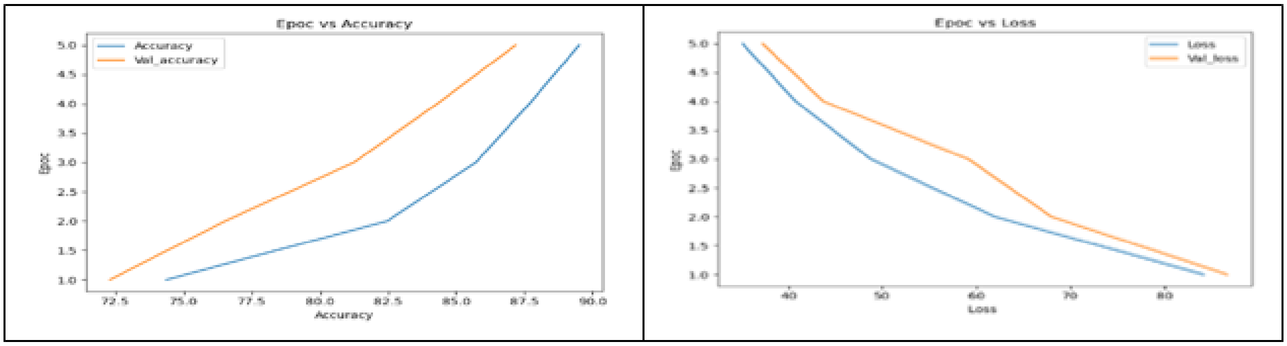
Image 2 UNET VGG-PSPNET VGG-UNET Labels

B. Graphs and Comparisons

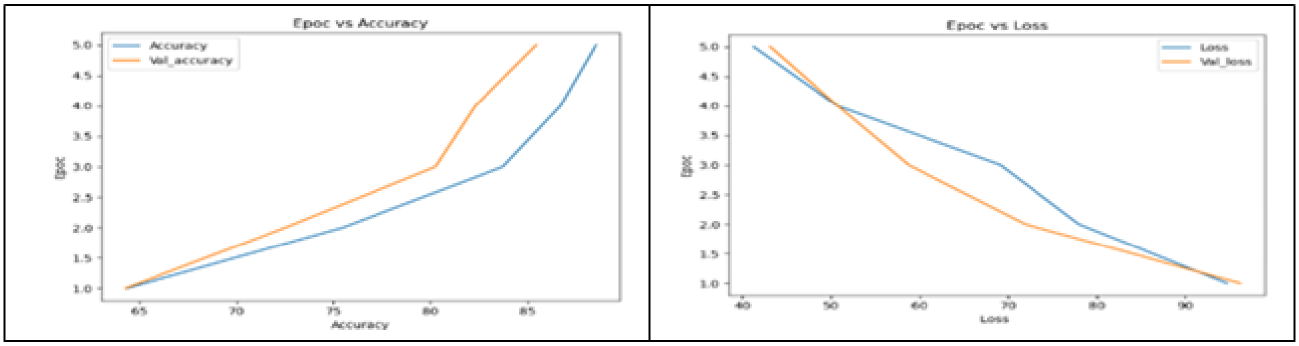
The Loss and Accuracy graphs for each model against each epoch



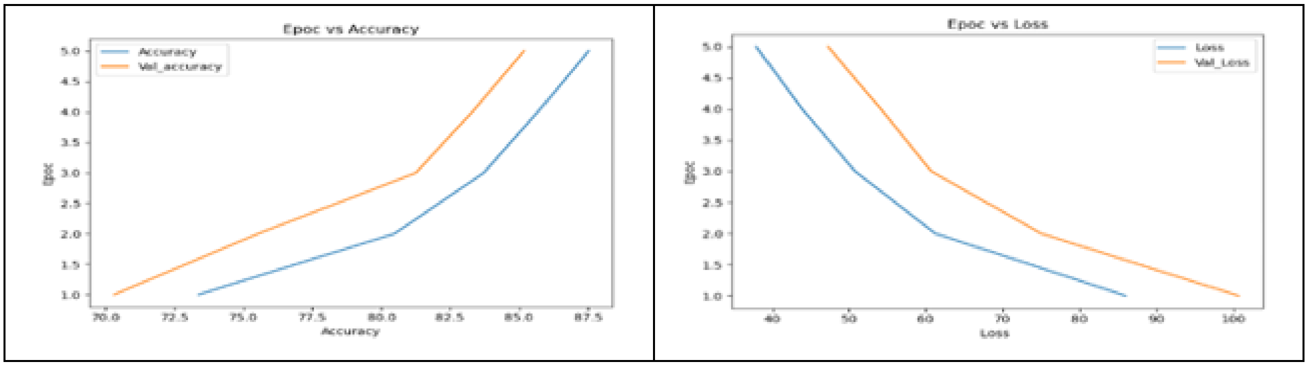
PSPNET



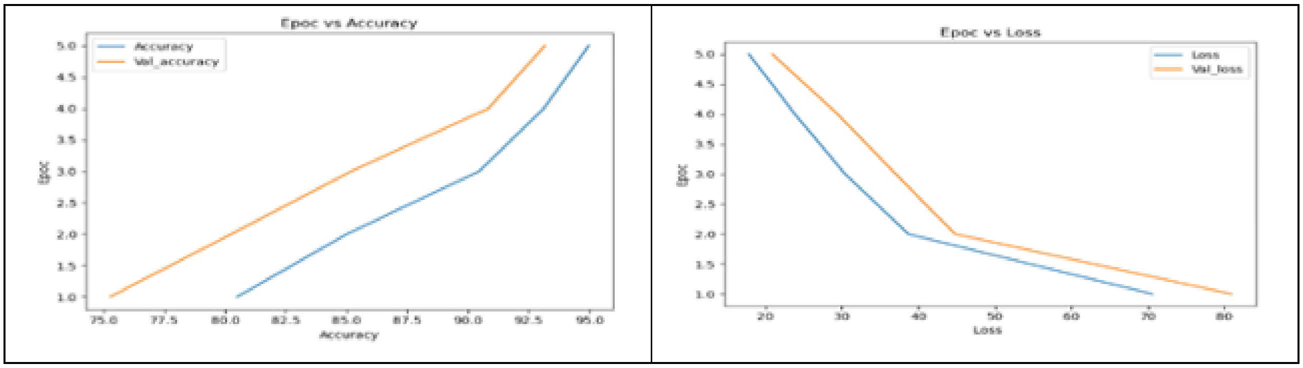
RESNET_UNET



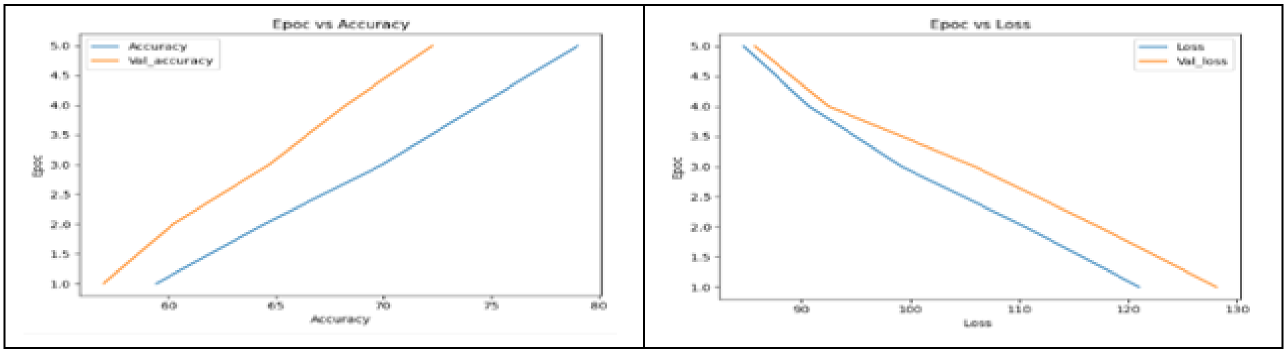
VGG_PSPNET



UNET



RESNET_PSPNET



VGG_UNET

Results and graphs show that when Resnet is combined with pspnet, it gives better results for segmentation in our problem.

V. CONCLUSION

In this work, we have demonstrated the concept of combining different pre-trained classification models and segmentations models for the semantic image segmentation. We developed an end-to-end trainable model that achieved good performance and results on camvid dataset as compared to the level of resources available. We trained our model for only 5 epochs due to limited number of resources. Moreover, if this model is trained on large dataset with a large number of epochs, more accurate and precise results will be achieved that can be used in many real-world applications like autonomous driving.

REFERENCES

[1] P.Wang, P. Chen, Y. Yuan, D. Liu, “Understanding convolution for semantic segmentation,” IEEE, 2018.

[2] MI. Jordan, TM. Mitchell,” Machine learning: Trends, perspectives, and prospects,” Science, 2015.

[3] Y. LeCun, Y Bengio, G.Hinton, “Deep learning,” nature, 2015.

[4] A Garcia-Garcia, S Orts-Escolano, S Oprea, “A review on deep. Learning techniques applied to. Semantic segmentation,” TPAMI, 2017.

[5] Karen. Simonyan, Andrew. Zisserman, “Very deep convolutional networks for large -scale image recognition,” Department of Engineering Science,” University of Oxford, 2015.

[6] A Krizhevsky, I Sutskever, GE. Hinton, “Imagenet classification with deep convolutional neural networks,” Advances in.neural information, 2012.

[7] Kaiming He, Xiangyu Zhang, S Ren, J Sun, “Deep. Residual. Learning for Image Recognition,” Proceedings of the IEEE, 2016.

[8] J. Deng, W. Dong, R. Socher, LJ Li, K Li,” Imagenet: A large-scale hierarchical image database,” IEEE conference, 2009.

[9] Ronneberger, P Fischer, T Brox, “U-net: Convolutional networks for biomedical image segmentation,” International Conference on Medical, 2015.

[10] D. Wu, Y. Wang, ST. Xia, J. Bailey, X. Ma, “Skip connections matter: On the transferability of adversarial examples generated with resnets,” ICLR conference paper, 2020.

[11] R. Yamashita, M. Nishio, RKG. Do, K. Togashi, “Convolutional neural networks: an overview and application in radiology,” Insights into imaging, 2018.

[12] H. Zhao, J. Shi, X. Qi, X. Wang, “Pyramid scene parsing network,” Proceedings of the IEEE, 2017.

[13] J. Long, E. Shelhamer, T. Darrell, “Fully convolutional networks for .semantic segmentation,” Proceedings of the IEEE, 2015.

[14] LC. Chen, Y. Zhu, G. Papandreou, “Encoder-decoder with.atrous separable convolution for semantic.image segmentation,” Proceedings of the IEEE, 2018.

[15] J. Kang, S. Kim, KM. Lee, “Multi - modal/multi-scale convolutional neural network based in-loop filter design for next generation video codec” IEEE (ICIP), 2017.

[16] G. Lin, Q. Wu, L. Qiu, X. Huang, “Image super-resolution using a dilated convolutional neural network,” Neurocomputing, 2018.

[17] LC. Chen, G. Papandreou, I. Kokkinos, “Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs,” zzZZ IEEE, 2017.

[18] T. Mikolov, M. Karafiát, L. Burget, J. Černocký, “Recurrent neural network based language model,” International Speech Communication Association, 2010.

[19] P. Badjatiya, LJ. Kurisinkel, M. Gupta, “Attention-based neural text segmentation,”ECIR, 2018.

[20] O. Matan, C. J. Burges, Y. LeCun, and J. S. Denker, “Multidigit recognition using a space displacement neural network,” NIPS, 1992.

[21] JJ. Koenderink, AJ. Van. Doorn, "Representation of local geometry in the visual system," Biological Cybernetics, 1987.

[22] G. Papandreou, LC. Chen, "Weakly-and semi-supervised learning of a deep convolutional network for semantic image segmentation," Proceedings of the IEEE, 2015.

[23] B. Zhang, C. Wang, Y. Shen, Y. Liu, "Fully connected conditional random fields for high-resolution remote sensing land use/land cover classification with convolutional neural networks," Remote Sensing, 2018.

[24] M. Everingham, L. Van. Gool, CKI. Williams, "The pascal visual object classes (voc) challenge," International Journal of Computer Vision, 2009.

[25] M. Cordts, M. Omran, S. Ramos, "The cityscapes dataset," Future of Datasets in Vision, 2015.

[26] S. Zheng, S. Jayasumana, "Conditional random fields as recurrent neural networks," Proceedings of the IEEE, 2015.

[27] H. Noh, S. Hong, B. Han, "Learning deconvolution network for semantic segmentation," Proceedings of the IEEE, .2015.

[28] Y. Chen, J. Tao, L. Liu, J. Xiong, R. Xia, J. Xie, "Research of improving semantic image segmentation based on a feature fusion model," Journal of Ambient Intelligence and Humanized Computing, 2020.

[29] D-Roy, P-Panda, K-Roy, "Tree-CNN: a hierarchical deep convolutional neural network for incremental learning," Neural Networks, 2020.

[30] PM-Shakeel, S-Baskar, R-Sampath, "Echocardiography image segmentation using feed forward artificial neural network (FFANN) with fuzzy multi-scale edge detection (FMED)," International Journal of Signal and Imaging Systems Engineering," 2019.

[31] DP. Kingma, P. Dhariwal, "Glow: Generative flow with invertible 1x1 convolutions," NIPS 2018.

Road Obstacle Object Detection Based on Improved YOLO V4

Zuo Xiao, Yu Jun
School of Computer Science and Engineering
Xi'an Technological University
Xi'an, 710021, Shaanxi, China
E-mail: 1158198898@qq.com

Hu Yuzhe
Jinan University-University of Birmingham Joint Institute
Jinan University
Guangzhou, 511400, Guangdong, China
E-mail: 18137910896@163.com

Xian Tong
School of Computer Science and Engineering
Xi'an Technological University
Xi'an, 710021, Shaanxi, China

Hu Zhiyi
Engineering Design Institute
Army Research Loboratory
Beijing, 100000, China
E-mail: 763757335@qq.com

***Abstract*—In recent years, as one of the important technical tasks in the field of deep learning, object detection has broad prospects and applications in the field of road obstacle detection. However, in the real driving scene, there are many obstacles, serious occlusion, overlap and other problems, so that the existing obstacle detection algorithm can not effectively detect the obstacles on the road, so it can not guarantee the driving safety. In order to solve the above problems, this paper improves on the basis of Yolo V4 algorithm. Firstly, kmeans + + clustering is used to generate a priori box suitable for the data set to enhance the scale adaptability; Then, the ciou is used as the loss function of coordinate prediction to evaluate the coincidence degree of prediction frame and truth value frame more reasonably. Finally, a suitable target detection data set is constructed by preprocessing the public data set cityccaps. The experimental results show that the improved algorithm can achieve more than 90% accuracy for obstacles with large number of targets in the training set. Compared with the original Yolo V4, the average detection accuracy of the improved algorithm is improved by 2.03%.**

Keywords-YOLO v4 Algorithm; Obstacle; Object Detection；Loss Function

I. INTRODUCTION

As the main means of transportation for travel, cars provide great convenience to our lives, but the problem of safe driving of cars on the road comes with it. According to the National Statistical Yearbook, a total of 247,646 road traffic accidents occurred nationwide in 2019, causing more than 310,000 casualties and property losses of 1,346.1 million yuan. In order to reduce the occurrence of such problems and improve driving safety, object detection technology has been gradually applied to the field of car assisted driving [1]. It can provide vehicles with the perception information of the surrounding environment and automatically detect road obstacles to improve the road. The purpose of driving safety.

In recent years, scholars at home and abroad have gradually applied deep learning technology to obstacle object detection [2]. Prabhakar [3] and others have developed a set of deep learning systems on assisted driving for the detection and classification of road obstacles such as vehicles, pedestrians, animals, etc., suitable for autonomous driving [4] cars driving on highways. Tang Bowen [5] and others used the YOLO v3 algorithm to

DOI: 10.21307/ijanmc-2021-023

complete the UAV obstacle detection. The speed is fast but the accuracy of identifying the position of the object is poor. Guo Jishun [6] and others introduced the dynamic residual network to solve the problem of deep network and poor generalization in object detection, which solved the degradation of deep neural network well, but did not completely solve the performance problem caused by network deepening. The detection speed and accuracy of the above methods need to be improved. In this paper, YOLO v4 [7] of the YOLO series is used for object detection. Compared with YOLO v3, this algorithm lowers the training threshold and uses a single GPU for training more effective. More importantly, it has a significant improvement in detection speed.

Although the yolo v4 algorithm is considerable in terms of accuracy and detection speed, it still has some shortcomings, such as the random initial clustering center of the anchor box a priori box generated by the Kmeans method, resulting in inaccurate clustering results; The too high coincidence of urban environment makes it difficult to predict coordinates, which leads to the low accuracy of the detection results.

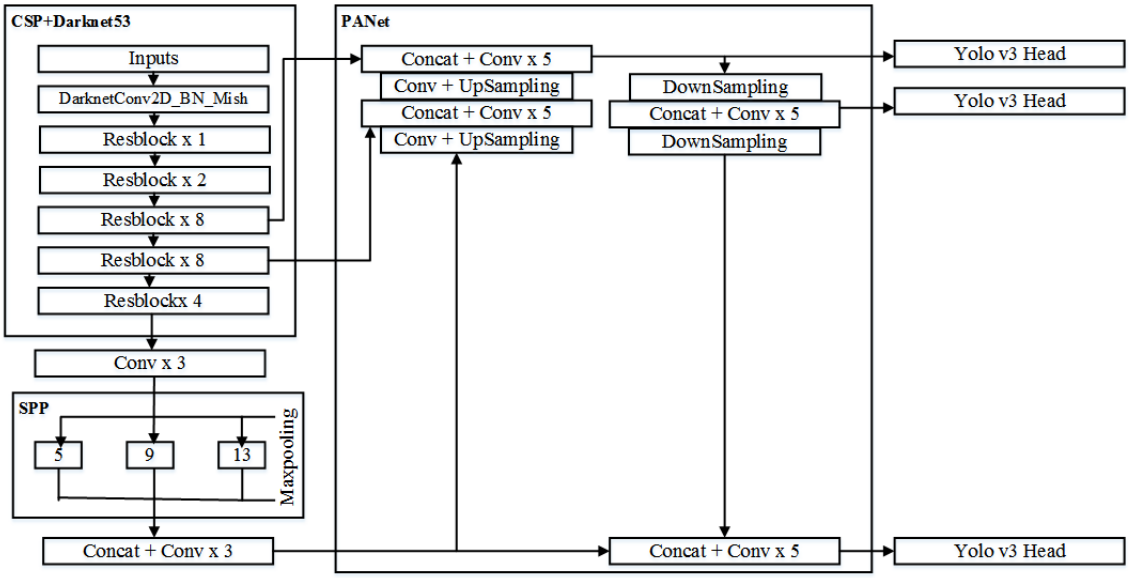


Figure 1. Network structure of YOLO v4 algorithm

TABLE I. ENVIRONMENT CONFIGURATION

Hardware environment	**processor**	**Intel(R) XEON W-2133**
	Graphics card	**Nvidia TITAN XP 12G**
Software Environment	operating system	Ubuntu 16.04
	Deep learning framework	Tensorflow-gpu
	Programming language	Python
	translater	Pycharm2019.1

In order to solve such problems, this paper proposes an improved YOLO V4 object detection algorithm. By improving the network structure of the algorithm, improving the Kmeans clustering and optimizing the coordinate prediction loss function, the improved algorithm is more suitable for object detection of road obstacles.

II. YOLO V4 ALGORITHM PRINCIPLE

YOLO (YouOnlyLookOnce) [8] network is a kind of object detection algorithm based on regression. Its main idea is to divide the image into multiple grids, then use the depth neural network to judge whether the network has a object or not, and then predict the category and position of the object. The network structure of YOLO v4 is shown in Figure 1. Keeping the Head part of YOLO v3, the CSPDarknet53 module selected by the backbone network, introduces spatial pyramid pooling (SPP) as an additional module of the Neck part to expand the receptive field, and PANet's path aggregation module is used as a part of the Neck. Among them, Darknet53 contains 5 residual blocks, and the number of small residual units contained in the residual blocks are 1, 2, 8, 8, and 4 respectively. CSPDarknet53 modifies Darknet53. Each large residual block is added with a CSPNet module and integrated into the feature map through gradient descent. Part of the feature map is convolved, and the other part is combined with the previous convolution result. CSP can improve the ability of convolutional neural networks to extract features and improve computational efficiency. PANet (Path Aggregation Network) makes full use of feature fusion. YOLO v4 also changes the fusion method from addition to multiplication, so that the network can get accurate detection results. YOLO v4 introduces Mosaic data augmentation and SAT for data enhancement, genetic algorithm selects hyperparameters, uses cross-small batch normalization, and uses DropBlock [9] regularization. They lowered the training threshold, allowing the model to get fast and accurate detection results under ordinary GPU conditions.

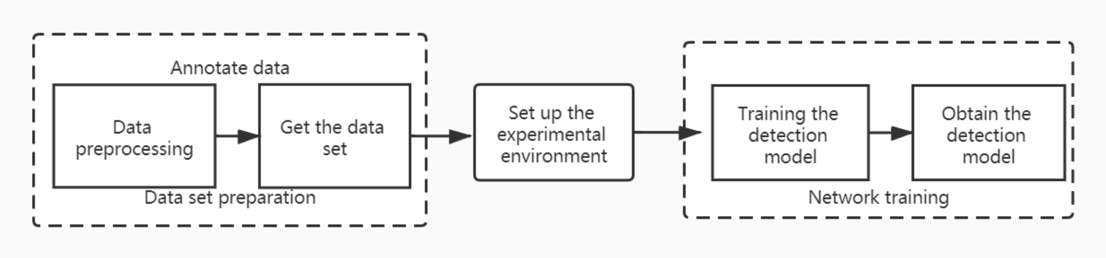

Annotate data
Data preprocessing
Get the data set
Data set preparation
Set up the experimental environment
Training the detection model
Obtain the detection model
Network training

Figure 2. Experimental framework

Although the YOLO v4 algorithm has excellent accuracy and detection speed, there are two problems:

1) The anchor box value (anchor box is a priori box) generated by the Kmeans method, random initial aggregation,the clustering result is not necessarily accurate due to the cluster center, thus affecting the accuracy of the detection result.

2) When the coincidence degree of the object is relatively high, a good coordinate prediction loss function is required to obtain the actual position of the object.

III. IMPROVED YOLO V4 ALGORITHM DESIGN

In response to the above problems, this article has improved the YOLO v4 algorithm. The main work includes:

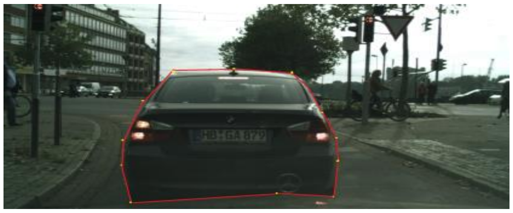
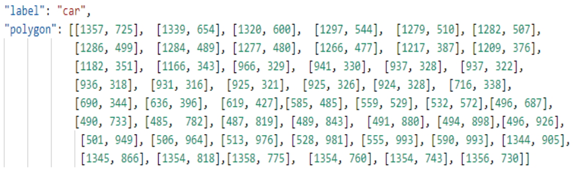
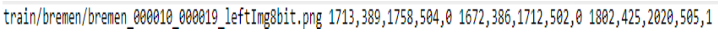


Figure 3. Annotation of car instance segmentation



```
"label": "car",
"polygon": [[1357, 725],  [1339, 654], [1320, 600],  [1297, 544],  [1279, 510], [1282, 507],
            [1286, 499],  [1284, 489], [1277, 480],  [1266, 477],  [1217, 387], [1209, 376],
            [1182, 351],  [1166, 343], [966, 329],  [941, 330],  [937, 328],  [937, 322],
            [936, 318],  [931, 316],  [925, 321],  [925, 326], [924, 328],  [716, 338],
            [690, 344], [636, 396],  [619, 427],[585, 485], [559, 529], [532, 572],[496, 687],
            [490, 733], [485,  782], [487, 819], [489, 843],  [491, 880], [494, 898],[496, 926],
             [501, 949], [506, 964], [513, 976], [528, 981], [555, 993], [590, 993], [1344, 905],
             [1345, 866], [1354, 818],[1358, 775],  [1354, 760], [1354, 743], [1356, 730]]
```

Figure 4. The json file of the car label



```
train/bremen/bremen_000010_000019_leftImg8bit.png 1713,389,1758,504,0 1672,386,1712,502,0 1802,425,2020,505,1
```

Figure 5. The txt file of the image tag

1) Kmeans++ is selected for the generation of anchor box;

2) The coordinate prediction loss function uses CIoU.

A. Generate anchor box with Kmeans++

The YOLO v4 algorithm originally used the Kmeans clustering algorithm to generate the anchor box. Since the initial clustering center of the Kmeans algorithm is randomly selected, the classification results may not be accurate. The selection of the clustering center must be as far away as possible. Therefore, this paper uses the Kmeans++ clustering algorithm to analyze the data set and generate suitable anchor box values. The Kmeans++ algorithm ensures that the latest cluster center is as far away as possible from the previous center. In order to reduce the error caused by the size of the anchor box itself, Intersection over Union (IoU) is selected as the measurement standard, and the calculation formula is shown in formula (1). Among them, box is the object truth box, centroid is the obtained a priori box, and IoU (box, centroid) represents the intersection ratio of the a priori box and the truth box. It can be seen that the smaller the distance d, the larger the intersection ratio, the more the a priori box and the truth box overlap, and the better the clustering effect.

$$d(box, centroid) = 1 - IoU(box, centroid) \quad (1)$$

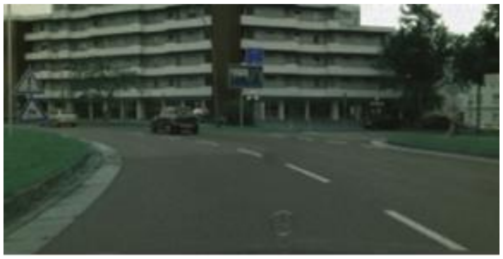


Figure 6. Add Gaussian noise

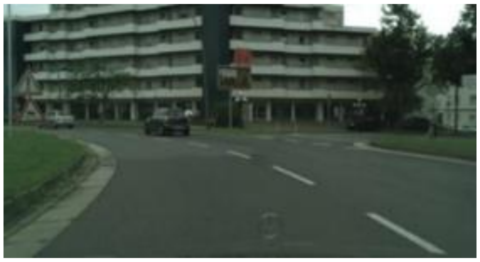
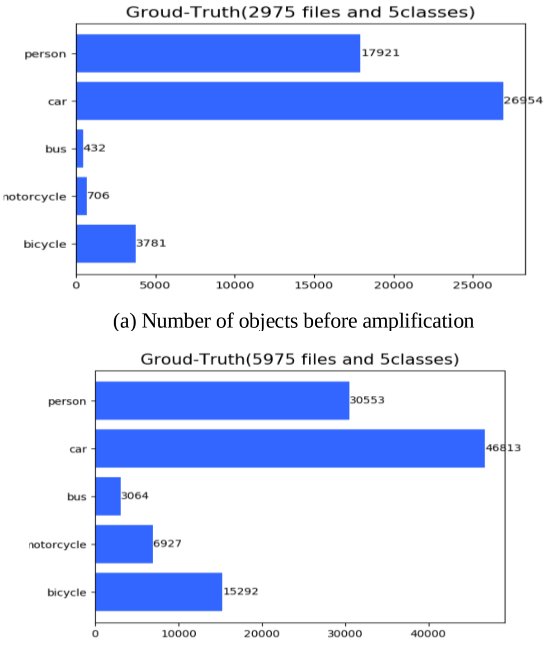


Figure 7. Median fuzzy processing



(a) Number of objects before amplification

(b) Number of objects after amplification

Figure 8. Object number before and after data amplification

TABLE II. MAIN NETWORK PARAMETER VALUES

Parameter	Value	Parameter	Value
LEARN_RATE_INIT	1e-4	MOVING_AVE_DECAY	0.9999
LEARN_RATE_END	1e-6	STAGE_EPOCHS	100

Based on the three output scales, three types of anchor boxes are set, and 9 types of anchor boxes are clustered in this paper. The anchor box values are (54, 56), (93, 89), (207, 161), (60, 109), (133, 125), (145, 257), (85, 167), (254, 188) and (293, 286).

B. Choose CIoU as the loss function of coordinate prediction

In object detection, the method for the model to evaluate the distance between the predicted frame and the true value frame usually adopts IoU, GIoU and DIoU. However, there are the following problems: IoU is a ratio, which is not sensitive to the size of the object, and cannot directly optimize the non-coincident range; GIoU can detect the non-coincident range but does not consider the center distance; DIOU considers the bounding box coincidence and center distance problems but does not Consider the scale ratio. In response to the above problems, this paper uses Complete Intersection over Union (CIoU) as the coordinate loss function, which takes into account the overlap area, center distance and scale ratio, so it can more reasonably evaluate the degree of overlap between the prediction box and the true value box.

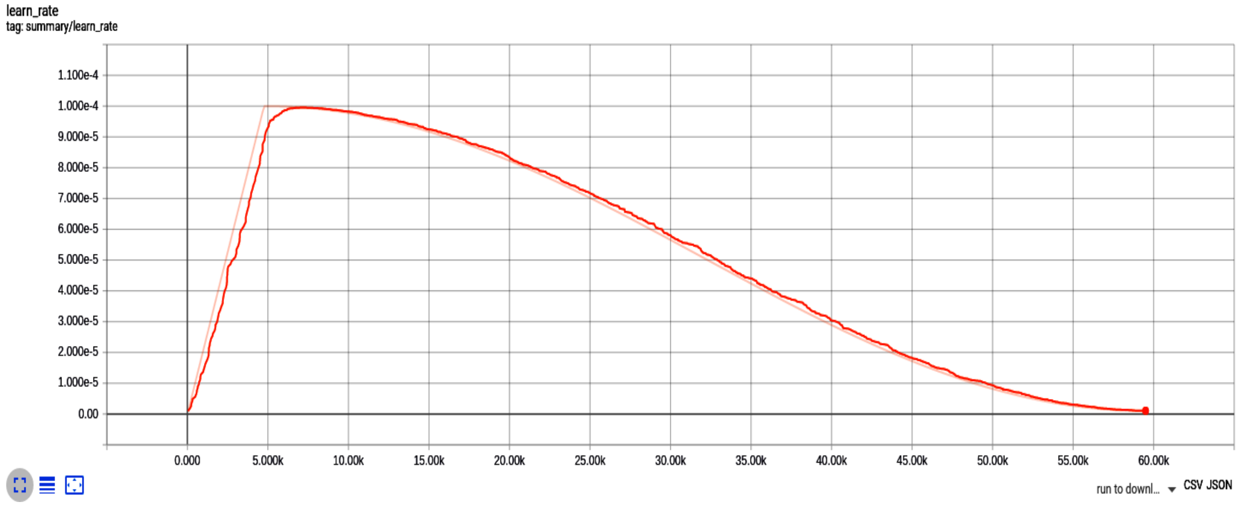


Figure 9. Learning rate change curve

CIoU adds an influence factor on the basis of the penalty item of DIoU, and considers the fitting degree of the aspect ratio of the predicted frame to the aspect ratio of the real frame as the consideration range, as shown in formula (2).Where v is a parameter used to measure the consistency of the aspect ratio， $\alpha$is the trade-off parameter, b represents the center of the prediction box, represents the center of the real box, p^2 () represents the square of the Euclidean distance, and c represents the minimum diagonal distance between the prediction box and the real box in the bounding box. The specific calculation methods of $v$and $\alpha$are shown in formulas (2) and (4), and the loss function of CIoU is shown in formula (5).

CIoU has scale invariance. When the object frame overlaps and contains, the normalized distance between the predicted frame and the real frame is minimized, thereby speeding up the convergence speed, making the regression process more stable, and avoiding divergence problems during the training process.

$$CIoU = IoU - \frac{p^2(b, b^{qt})}{c^2} - \alpha v \tag{2}$$

$$v = \frac{4}{\pi^2}(\arctan\frac{w^g}{h^{8t}} - \arctan\frac{w}{h})^2 \tag{3}$$

$$\alpha = \frac{v}{(1 - IoU) + v} \tag{4}$$

$$L_{CIoU} = 1 - CIoU \quad (5)$$

IV. EXPERIMENT AND RESULT ANALYSIS

In this article the experiment is carried out under the Linux system, and the

The experiment in this article is carried out under the Linux system, and the experiment environment is shown in Table 1. In order to reduce the training time of the deep neural network model and increase the calculation speed, the Nvidia TITAN XP12G graphics card is used, and CUDA9.0 and cuDNN7.0 are configured to call the GPU for acceleration. The deep learning framework chosen is Tensorflow.

The overall framework of the experiment is shown in Figure 2. It mainly includes three parts: the preparation of the data set, the construction of the training environment and the training of the network. The strategy of network training is as follows

A. *Preparation of experimental data set*

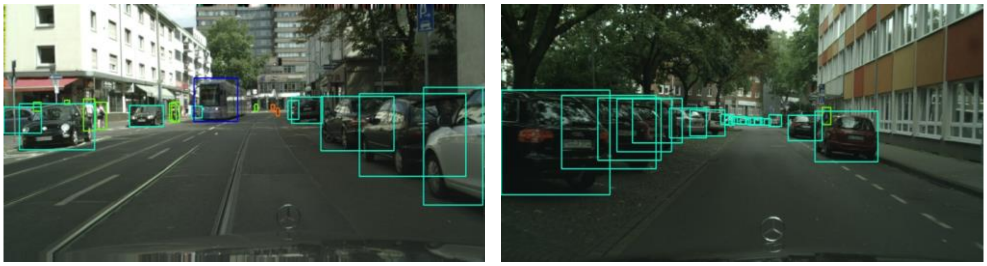
The data in this paper comes from Cityscapes, which mainly contains 5,000 high-quality pixel-level annotated images of driving scenes in urban environments and 20,000 rough-annotated images. Since the official does not provide annotations for the test set images, we used a training set of 2975 images for training and a verification set of 500 images for testing.

We mainly adopt two methods: data annotation and data amplification.

1) Data labeling

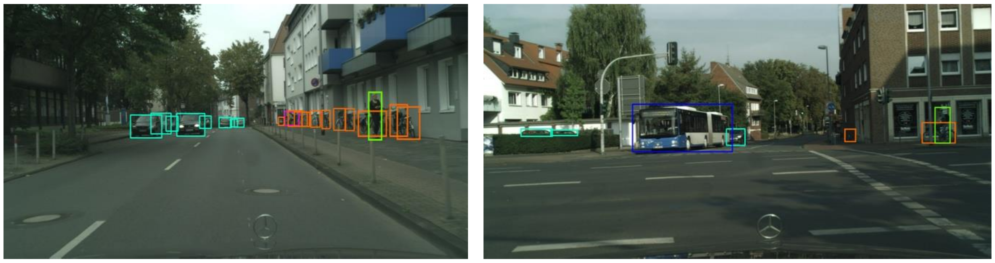
The Cityscapes dataset provides annotation information for semantic segmentation and instance segmentation. Generating a json file to store the outline information of the sample in the labeling method shown in Figure 3. As shown in Figure 4, the json file stores all the coordinate information of the contour points. The txt file shown in Figure 5 saves the coordinate information and target category information of all objects in the image.

2) Data amplification



(a) Images with a large number of targets (b) The image of the target being occluded

Figure 10. Experimental results based on Yolo v4



(a) Images with a large number of objects (b) The image of the objects being occluded

Figure 11. Experimental results based on improved YOLO v4

Data amplification adopts two methods: adding noise and blurring. Figure 6 shows the effect of adding Gaussian noise to the original image. Figure 7 shows the effect of median blur on the original image in the Cityscapes training set, we selected 500 images containing buses, bicycles, and motorcycles for data augmentation. The number of objects in each category before amplification is shown in Figure 8(a), and the number of objects in each category after amplification is shown in Figure 8(b).

B. Model training

In this paper, the detection model is trained according to the parameters shown in Table 2. Among them, BLEARN_RATE_INIT represents the minimum learning rate; LEARN_RATE_END represents the maximum learning rate; MOVING_AVE_DECAY represents the moving average, which is used to estimate the local mean value of some parameters, so that the parameter update is related to the historical value within a period of time. STAGE_EPOCHS represents the iterative training of the data set 100 Table 3 Comparison of test results times. In the actual training process, each training batch will save one model, and it is set to save up to 10 models.

TABLE III. COMPARISON OF TEST RESULTS

AP(%)	Car	Bus	Person	Motorbike	Bicycle	mAP(%)
yolo v4	0.98	0.93	0.92	0.81	0.51	82.95
Improved yolo v4	0.99	0.93	0.92	0.81	0.58	84.98

The strategy of network training during the experiment is as follows.

1) Multi-scale training strategy.

When the detection network is trained, the multi-scale training rules can be used to input different image resolutions, and the model can detect the input small-scale images faster. The specific training method is to modify the input size of the image every few batches so that the model is robust to images of different sizes and can detect images of different scales.

2) Warm up strategy

The data set of the early network is generally small and the network is not deep, so there is no need to adopt the Warm up strategy. As computer vision tasks become more and more complex, setting a fixed learning rate will cause problems in the training process. The best learning rate setting method is to change the learning rate according to the iterative batch, which not only has good training efficiency, but also avoids the instability of the fully connected layer, thereby increasing the deep similarity of the model. During the training process, the learning rate changes with the training batches are shown in Figure 9. The horizontal axis represents the iteration batches, and the vertical axis represents the learning rate. The curve in the figure reaches the highest learning rate at 5000 iterations. The learning rate at different stages is different. First, use a larger learning rate to find the global optimum, and then use a smaller learning rate to find the local optimum to obtain the global optimum solution of the model.

C. Experimental results and analysis

This paper uses the improved YOLO v4 network model before and after the experiment respectively, and the results of the visualization experiment based on YOLO v4 are shown in Figure 10. The visual experiment results based on the improved YOLO v4 are shown in Figure 11. The picture contains urban street scenes under different light levels, including five types of obstacles such as cars, buses, and motorcycles. They are marked by five different color detection boxes. These obstacles are overlapped, exposed to varying degrees of distance and distance.

It can be seen from Figure 10 that most of the obstacles included in the images with many objects and the images with heavy object occlusion have been detected, including the car roof can also be accurately identified. However, there are some problems such as too small positioning box, repeated box and missing detection. The obvious problem is: when people ride motorcycles or bicycles, most of the cases will not detect people, and the detection frame of motorcycles or bicycles is too large, in addition, the detection frame of large buses can not detect the whole car body. It can be seen from Figure 11 that various obstacles can be detected more effectively in the same background. Under different backgrounds, different forms of obstacles can be effectively detected, including obstacles blocked by foreign objects, overlapping objects, incomplete shooting, and blurred pixels. Riders on bicycles and motorcycles can be effectively detected, there is no redundant detection frame, and the coordinate information of obstacles can be predicted more accurately.

The test results of the above-mentioned comparative test are shown in Table 3. It can be seen that the average accuracy of the improved YOLO v4 algorithm is 2.03% higher than the detection result before the improvement.

V. Conclusion

This paper proposes a object detection algorithm based on improved YOLO v4, trains and tests the objects detection network on the Cityscapes dataset. Experimental results show that the improved YOLO v4 algorithm in this paper improves the average recognition accuracy of vehicle objects, and solves the problem of poor accuracy caused by different initial clustering centers in the YOLO v4 algorithm due to different clustering results and the lack of optimized bounding boxes Problem. Compared with YOLO v4, the detection accuracy of the improved algorithm is increased by 2.03%.

References

[1] Zhao Richeng. Research on road obstacle detection technology in assisted driving [D]. Xidian University, 2015.

[2] Wang Tiantao, Zhao Yongguo, Chang Faliang. Obstacle detection based on visual sensor [J]. Computer Engineering and Applications, 2015, 51(4):180-183.

[3] Prabhakar G, Kailath B, Natarajan S, et al. Obstacle detection and classification using deep learning for tracking in high-speed autonomous driving[C]//2017 IEEE region 10 symposium (TENSYMP). IEEE, 2017:1-6.

[4] Zeng Weiliang, Wu Miaosen, Sun Weijun, et al. Overview of Research on Autonomous Taxi Dispatching System [J]. Computer Science, 2020, 47(05):189-197.

[5] Tang Bowen. Research on Obstacle Detection and Obstacle Avoidance Processing During UAV Driving [D]. Guangxi University of Science and Technology, 2019.

[6] Guo Jishun. Semantic segmentation and target detection technology for autonomous driving [D]. University of Electronic Science and Technology of China, 2018.

[7] Zhang Xin, Qi Hua. Research on human abnormal behavior detection algorithm based on yolov4 [J]. Computer and digital engineering, 2021,49 (04): 791-796.

[8] Wong A , Famuori M , Shafi Ee M J , et al. YOLO Nano: a Highly Compact You Only Look Once Convolutional Neural Network for Object Detection [J]. 2019.

[9] Wang J, Gao F, Dong J, et al. Adaptive DropBlock-Enhanced Generative Adversarial Networks for Hyperspectral Image Classification [J]. IEEE Transactions on Geoscience and Remote Sensing, 2020, PP(99):1-14.

Designing Convolutional Neural Network Architecture Using Genetic Algorithms

Ashray Bhandare and Devinder Kaur
Department of EECS, the University of Toledo, Toledo, Ohio, USA

***Abstract*—In this paper, genetic algorithm (GA) is used to optimally determine the architecture of a convolutional neural network (CNN) that is used to classify handwritten numbers. The CNN is a class of deep feed-forward network, which have seen major success in the field of visual image analysis. During training, a good CNN architecture is capable of extracting complex features from the given training data; however, at present, there is no standard way to determine the architecture of a CNN. Domain knowledge and human expertise are required in order to design a CNN architecture. Typically architectures, The GA determine the exact architecture of a CNN by evolving the various hyper parameters of the architecture for a given application. The proposed method was tested on the MNIST dataset. The results show that the genetic algorithm is capable of generating successful CNN architectures. The proposed method performs the entire process of architecture generation without any human intervention.**

Keywords-Convolutional Neural Network; Genetic Algorithm; MNIST Data

I. INTRODUCTION

The idea that programmable computers will become intelligent was conceived over a hundred years before one was built. AI has tackled and solved many problems that are intellectually difficult for human beings but relatively straightforward for computers. Such problems are defined by a set of mathematical rules. The challenge for AI is to transform tasks which are easy and intuitive for humans into formal procedures that a computer can understand. For example, it is easy for humans to recognize a face, a piece of music even when the data is corrupted or incomplete.

With the advancements in big data, Graphical Processing Unit (GPU) technology and algorithms there has been a lot of progress in the field of Deep Learning. Deep Learning is part of machine learning techniques and allows a machine to learn with experience and data. It makes use of artificial neural networks with more than one hidden layer. By implementing more layers and more neurons within a layer, it allows the network to understand complex ideas by building upon simpler ones. For example, a deep network can build the concept of an image of a car by combining simpler concepts, such as edges, corners, contour, and object parts [1].

Convolutional Neural Network (CNN) is one such type of deep networks. Yann LeCan carried out one of the first exercises on CNN. He taught a computer system how to recognize the differences between handwritten digits [2]. When the system chose incorrectly, he would correct it until the program figured out the mathematical operation called convolution. Convolution is a specialized kind of linear operation. Unlike conventional neural networks, CNN’s use this linear operation to obtain an intermediate output (feature) before

DOI: 10.21307/ijanmc-2021-024

using it as an input for the next layer. This is done in at least one of their layers. A typical CNN architecture consists of layers such as convolution layer, pooling layer, and fully connected layer. Each of these layers consists of hyper-parameters that are chosen by researchers using new theoretical insights or intuition gained from experimentation. In this paper, we achieved the following objectives:

Automate the process of CNN architecture selection.

Achieve the architecture by evolving the hyper parameters of CNN using Genetic Algorithm (GA)

Discover CNN architectures without any human intervention that perform well on a given machine-learning task.

GA is inspired by biological evolution, used to find globally optimal solutions and makes use of genetic operators such as selection, crossover, and mutation.

The goal of the proposed algorithms is to discover CNN architectures that perform well on a given machine-learning task with no human intervention. Over the course of many generations, Genetic Algorithm picks out the layers and hyper-parameters to choose from, the algorithm is left with a finite but large space of model architectures to search from. It learns through random exploration and slowly begins to exploit its findings to select higher performing models. It receives the testing accuracy as a means of comparison between architectures and ultimately selects the best architecture. The entire process called an evolutionary experiment goes on for many generations until a fully trained suitable CNN model is generated.

This paper is organized as follows. In Section 2, the dataset used for this analysis is introduced. Section 3 presents the mathematical model of CNN. GA is explained in section 4. Section 5 talks about our proposed method to generate CNN architectures using GA. In Section 6, experiments and results are presented. Conclusions are presented in section 7.

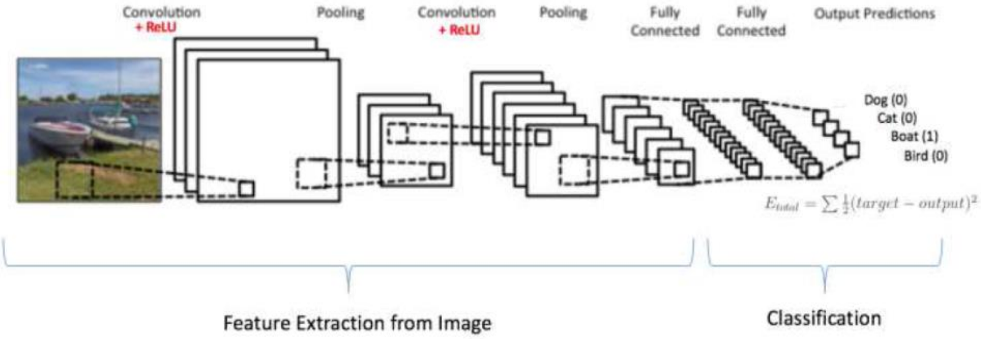


Figure 1. An example of CNN architecture [10]

II. MNIST DATASET

The MNIST Dataset is scanned images of handwritten digits and the associated labels describe which digit 0-9 is contained in each image. The "NIST" stands for National Institute of Standards and Technology, the agency that originally collected this data. The "M" stands for "modified," since the data has been preprocessed for easier use with machine learning algorithms.

The data set consists of 50,000 labeled samples of handwritten digits which is to be used as training data. It also consists of an extra 10,000 images that are unlabeled and used as testing data.

It is one of the popular datasets as it allows researchers to study their proposed methods in a controlled environment. In our case, we will discover the best CNN architecture that classifies this data set using genetic algorithm (GA).

III. Convolutional Neural Network

In machine learning, a CNN is a type of feed-forward artificial neural network in which the connectivity pattern between its neurons is inspired by the organization of the animal visual cortex [3]. This idea was expanded upon by a fascinating experiment by Hubel and Wiesel in 1962 where they showed that some individual neuronal cells in the brain responded (or fired) only in the presence of edges of a certain orientation [4]. For example, some neurons respond when exposed to vertical edges and some respond when shown horizontal or diagonal edges. When all these neurons are arranged in a tilled manner, they were able to produce visual perception. The idea that different neurons in the visual cortex look for different features are the inspiration behind a CNN.

In this paper, we use CNN for the task of image classification. CNN take advantage of the fact that the input consists of images and they constrain the architecture in a more sensible way [5]. In particular, the layers of a CNN have neurons arranged in three dimensions: width, height, depth. (Note that the word depth here refers to the third dimension of an image, not to the depth of a full Neural Network, which can refer to the total number of layers in a network). The neurons in a layer will only be connected to a small region of the layer before it, instead of all of the neurons in a fully-connected manner.

A. Layers in a CNN

A simple convolutional neural network is a sequence of layers. A CNN takes an image and passes it through a series of convolutional, nonlinear (activation), pooling (down sampling), and fully connected layers to get an output [5] [6]. This output can be a single class or a probability of classes that best describe the image. An example of a CNN architecture is shown in Figure 1. Each of these layers is explained in the following subsections.

1) Convolutional layer

A convolutional layer is a core building block of any CNN architecture. It is always the first layer of the architecture. The input of this layers will always be a three-dimensional object (eg. 400x400x3). The best way to understand convolution is to imagine a window of significantly lesser size (eg. 5x5) is being moved across all the areas of the input. In machine learning terms, this window is called a filter or neuron or kernel. Now, this filter is also an array of numbers (the numbers are called weights or parameters).

Figure 2 illustrates the process of convolution on an image of dimensions 5x5x3. The filter used to convolve over this image is of the dimensions 3x3x3. It can be noted that the depth of the filter is same as the depth of the image. The output generated after the process of convolution is calculated by performing elementwise multiplication. These multiplications across the width, height and depth of an image with the filter are all summed up in order to get a single value. Finally, the single value is averaged over the total number of values in the filter. It is computed as follows:

$$Output = \frac{\begin{gathered}[(0*0)+(-1*0)+(1*1)+(0*0)+\\(0*0)+(-1*-1)+(0*2)+(1*0)+\\(1*2)]\ +[(-1*2)+(0*1)+(1*1)+\\(1*2)+(1*1)+(1*0)+(0*0)+\\(0*0)+(0*2)]\ +[(-1*0)+(-1*0)+\\(1*1)+(-1*1)+(0*0)+(1*0)\\+(0*1)+(0*1)+(0*1)]\end{gathered}}{27} = \frac{2+2-1}{27} = 0.111$$

Now, we repeat this process for every location on the input volume. Every unique location on the input volume produces a number. After sliding the filter over all the locations, we are left with a two-dimensional array which is called an activation map or feature map.

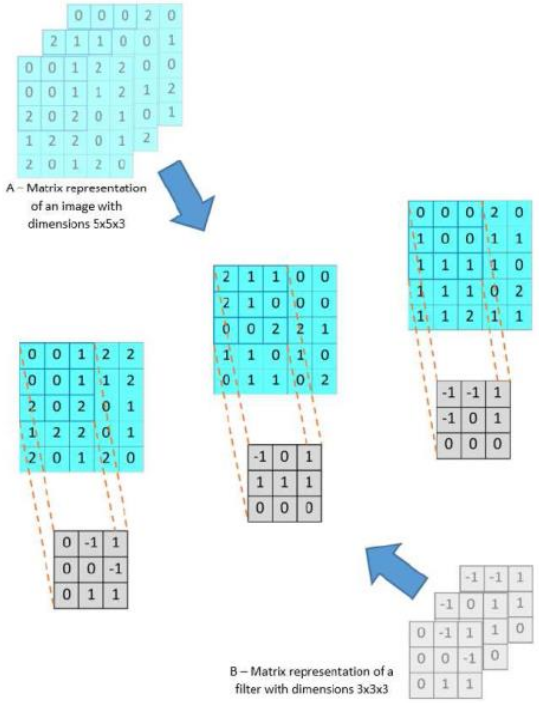

A – Matrix representation of an image with dimensions 5x5x3
B – Matrix representation of a filter with dimensions 3x3x3

Figure 2. The process of convolution on the left-hand corner of an image

The size of this feature map is governed by three parameters.

These parameters need to be decided before convolution is performed. They are as follows:

- **Depth:** The number of filters that convolve over the same image to get different two-dimensional feature maps corresponds to the depth of the output after the convolutional layer.
- **Stride:** The step by which a filter is moved across an image, is called stride.
- **Zero-padding:** The process of padding zeros around an input image is called as zero-padding. This is done so that we can slide the filter to the bordering elements of the input image matrix. This parameter allows us to control the size of the generated feature map.

2) *Non Linearity (Activation Layer)*

Most of the real world datasets are nonlinear in nature. The process of convolution (element-wise matrix multiplication and addition) is a linear operation. Due to this, there are no nonlinear properties in our network. In order to introduce nonlinearity in our network, the output feature maps obtained after the process of convolution in the convolutional layer is passed through a nonlinear function.

One of the most popular nonlinear operation is the ReLU operation. Other nonlinear functions such as tanh or sigmoid can also be used instead of ReLU, but ReLU has been found to perform better in most situations.

3) *Pooling Layer*

Pooling is a way to take large images and shrink them down while preserving the most important information in them. It is common to periodically insert a Pooling layer in-between successive Convolutional layers in a CNN architecture. Maxpooling is one the most popular types of pooling. Maxpooling consists of stepping a small filter across an image and taking the maximum value from the filter at each step. The stride by which the filter steps across the image is usually the same as that of filter size.

The process of pooling has two main advantages. The first is that the amount of parameters or weights is reduced by 75%, thus lessening the computation cost. The second is that it will control overfitting as it reduces the number of parameters and computations in the network.

4) *Fully Connected Layer*

The Fully Connected layer is a traditional Multi-Layer Perceptron layer that uses a softmax activation function in the output layer. The softmax activation function generates the outputs in the range of 0 and 1. The output of the softmax function is equivalent to a categorical probability distribution, it tells you the probability that any of the classes are true.

B. Training using Backpropagation

As we have seen from the subsections above, the convolution and pooling layers act as Feature Extractors from the input image in a CNN architecture while a fully connected layer acts as a classifier generating probabilities for the different classes. This is the forward propagation step.

Now we train the dataset using traditional back-propagation until it has converged and we obtain high classification accuracy. This classification accuracy will be used as a fitness value when the architecture is passed through the GA tuner. This is further explained in the following sections.

IV. Genetic Algorithm

Genetic Algorithms were invented to mimic some of the processes observed in natural evolution. Many people, biologists included, are astonished that life at the level of complexity that we observe could have evolved in the relatively short time suggested by the fossil record [7]. The idea with GA is to use this power of evolution to solve optimization problems. Genetic Algorithms (GAs) are adaptive heuristic search algorithm based on the evolutionary ideas of natural selection and genetics [8]. As such, they represent an intelligent exploitation of a random search used to solve optimization problems. Although randomized, Gas are by no means random, instead, they exploit historical information to direct the search into the region of better performance within the search space.

A. GA Overview

Gas simulates the survival of the fittest among individuals over a consecutive generation for solving a problem. Each generation consists of a population of character strings that are analogous to the chromosome that we see in our DNA. Each individual represents a point in a search space and a possible solution.

The individuals in the population are then made to go through a process of evolution based on the following foundations [8]:

- Individuals in a population compete for resources and mates.
- Those individuals most successful in each ‘competition’ will produce more offspring than those individuals that perform poorly.
- Genes from good individuals propagate throughout the population so that two good parents will sometimes produce offspring that are better than either parent.
- Thus each successive generation will become more suited to their environment.

A population of individuals is maintained within a search space for a GA, each representing a possible solution to a given problem. Each individual is coded as a finite length vector of components, or variables, in terms of some alphabet, usually the binary alphabet {0,1}. These individuals are analogous to chromosomes and the variables are analogous to genes. Thus a chromosome (solution) is composed of several genes (variables). A fitness score is assigned to each solution representing the abilities of an individual to compete. The individual with the abilities of an individual to compete. The individual with the optimal (or generally near optimal) fitness score is sought. The GA aims to use selective breeding of the solutions to produce offspring better than the parents do by combining information from the chromosomes. Figure 3 illustrates a population of chromosomes. In this figure, each chromosome consists of 8 genes.

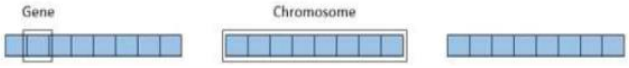


Figure 3. Population of three chromosomes

Parents are selected to mate, on the basis of their fitness, producing offspring via a reproductive plan. Consequently, solutions with higher fitness are given more opportunities to reproduce so that offspring inherit characteristics from each parent. As parents mate and produce offspring, room must be made for the new arrivals since the population is kept at a static size. Individuals in the population die and are replaced by the new solutions, eventually creating a new generation once all mating opportunities in the old population have been exhausted. In this way, it is hoped that over successive generations better solutions will thrive while the least fit solutions die out [7].

New generations of solutions are produced containing, on average, more good genes than a typical solution in a previous generation. Each successive generation will contain better ‘partial solutions’ than previous generations. Eventually, once the population has converged and is not producing offspring noticeably different from those in previous generations, the algorithm itself is said to have converged to a set of solutions to the problem at hand.

B. Genetic Operators

A genetic operator is an operator used in genetic algorithms to guide the algorithm towards a solution to a given problem. There are three main types of operators

- Selection
- Crossover
- mutation

These operators must work in conjunction with one another in order for the algorithm to be successful. Genetic operators are used to creating and maintaining genetic diversity (mutation operator), combine existing solutions (also known as chromosomes) into new solutions (crossover) and select between solutions (selection).

1) Selection

Selection operators give preference to better solutions (chromosomes), allowing them to pass on their ‘genes’ to the next generation of the algorithm. The best solutions are determined using some form of objective function (also known as a ‘fitness function’ in genetic algorithms), before being passed to the crossover operator [8]. Different methods for choosing the best solutions to exist, for example, Roulette wheel selection and tournament selection; different methods may choose different solutions as being ‘best’. The selection operator may also simply pass the best solutions from the current generation directly to the next generation; this is known as elitism or elitist selection.

2) Crossover

Crossover is the process of taking more than one parent solutions (chromosomes) and producing a child solution from them. By recombining portions of good solutions, the genetic algorithm is more likely to create a better solution. As with selection, there are a number of different methods for combining the parent solutions, for example, single-point crossover and two-point crossover [8].

Figure 4 and Figure 5 illustrate the concept of single-point and two-point crossover respectively.

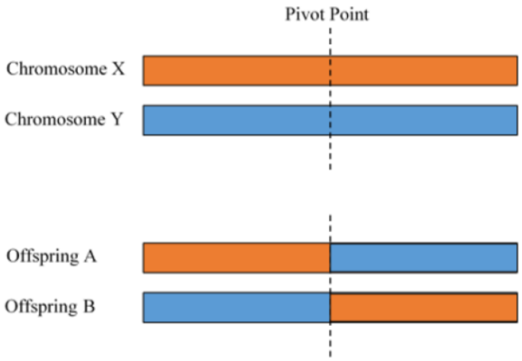


Figure 4. Single-point crossover

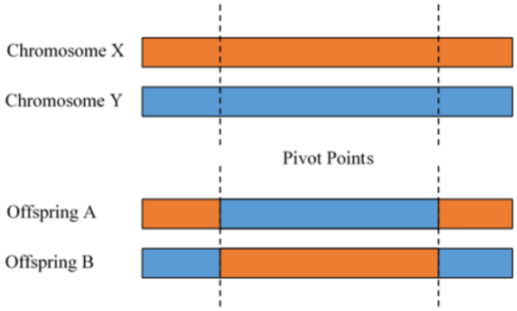


Figure 5. Two-point crossover

3) *Mutation*

The mutation operator encourages genetic diversity amongst chromosomes and attempts to prevent the genetic algorithm converging to a local minimum by stopping the chromosomes becoming too close to one another. In mutating the current pool of chromosomes, a given chromosome may change entirely from the previous chromosome. By mutating the chromosomes, a genetic algorithm can reach an improved solution solely through the mutation operator [8]. Again, different methods of mutation may be used; these range from a simple bit mutation (flipping random bits in a binary string chromosome with some low probability) to more complex mutation methods, which may replace genes in the chromosomes with random values chosen from the uniform distribution or the Gaussian distribution.

V. PROPOSED METHOD

Before training a deep neural network, it is necessary to determine the architecture of that network which is in turn done by choosing the hyper parameters associated with each layer of the network. Usually, the hyper parameters are determined by human intuition, experience or trial, and error.

Table 1, shows all the hyper-parameters associated with the deep neural network and their ranges which are used for experiments in this paper. The focus of this paper is to use genetic algorithm to automatically determine these hyper-parameters for best performance.

TABLE 1. THE VARIOUS HYPER PARAMETERS IN CNN WITH THEIR RANGES

Hyper parameter	Range
No. of Epoch	(0 - 127)
Batch Size	(0 - 256)
No. of Convolution Layers	(0 - 8)
No. of Filters at each Convo layer	(0 - 64)
Convo Filter Size at each Convo layer	(0 - 8)
Activations used at each Convo layer	(sigmoid, tanh, relu, linear)
Maxpool layer after each Convo layer	(true, false)
Maxpool Pool Size for each Maxpool layer	(0 - 8)
No. of Feed-Forward Hidden Layers	(0 - 8)
No. of Feed-Forward Hidden Neurons at each layer	(0 - 64)
Activations used at each Feed-Forward layer	(sigmoid, tanh, softmax, relu)
Optimizer	(Adagrad, Adadelta, RMS, SGD)

When tuning the hyper-parameters of a CNN architecture using genetic algorithm, the most crucial step is the problem representation. In other words, the problem should be formulated in such a way that it is suitable for the genetic algorithm. The variables involved in this tuning process are the various CNN hyper-parameters. Hyper-parameters, which are tuned to their optimal values, will generate the best CNN architecture and provide the highest classification and prediction accuracy for the given MNIST dataset.

For the purposes of experimentation, a direct-encoding representation of the hyper-parameters is performed. Here the value of hyper-parameters is extracted from a GA chromosome, which is in the binary format. Figure 6 shows a random GA architecture in the binary format.

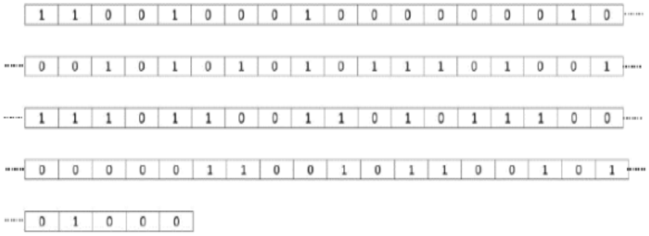


Figure 6. Representation of a GA chromosome

Figure 7 shows an illustration of the problem encoding process. The figure represents a random CNN architecture is obtained from a GA chromosome. Each hyper-parameter is shown with its corresponding binary interpretation.

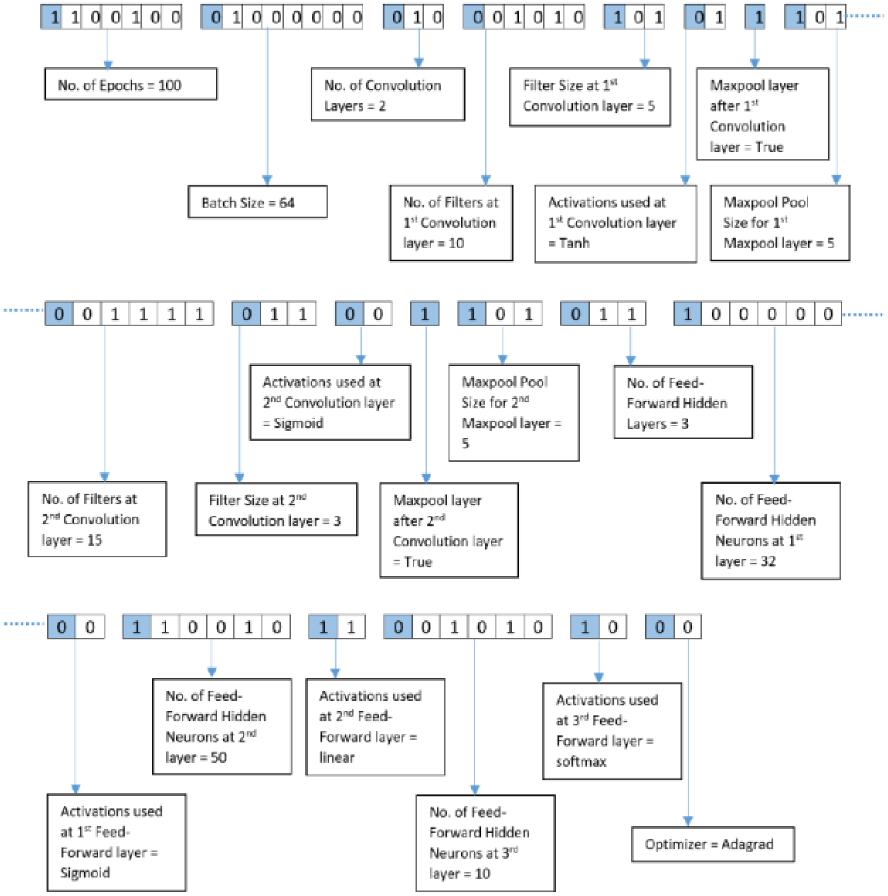

1 1 0 0 1 0 0
0 1 0 0 0 0 0 0
0 1 0
0 0 1 0 1 0
1 0 1
0 1
1
1 0 1
No. of Epochs = 100
No. of Convolution Layers = 2
Filter Size at 1st Convolution layer = 5
Maxpool layer after 1st Convolution layer = True
Batch Size = 64
No. of Filters at 1st Convolution layer = 10
Activations used at 1st Convolution layer = Tanh
Maxpool Pool Size for 1st Maxpool layer = 5
0 0 1 1 1 1
0 1 1
0 0
1
1 0 1
0 1 1
1 0 0 0 0 0
Activations used at 2nd Convolution layer = Sigmoid
Maxpool Pool Size for 2nd Maxpool layer = 5
No. of Feed-Forward Hidden Layers = 3
No. of Filters at 2nd Convolution layer = 15
Filter Size at 2nd Convolution layer = 3
Maxpool layer after 2nd Convolution layer = True
No. of Feed-Forward Hidden Neurons at 1st layer = 32
0 0
1 1 0 0 1 0
1 1
0 0 1 0 1 0
1 0
0 0
No. of Feed-Forward Hidden Neurons at 2nd layer = 50
Activations used at 2nd Feed-Forward layer = linear
Activations used at 3rd Feed-Forward layer = softmax
Activations used at 1st Feed-Forward layer = Sigmoid
No. of Feed-Forward Hidden Neurons at 3rd layer = 10
Optimizer = Adagrad

Figure 7. Representation of the hyper-parameters in binary format

A. Fitness Evaluation

In this study, a population size of 10 chromosomes (CNN architecture) is chosen. The fitness of each chromosome is evaluated using a fitness function.

The fitness function used in this study is the classification accuracy, which determines the number of correctly classified digits in the MNIST dataset. In section 3 of this paper, it is seen how a CNN architecture can be trained to obtain its best classification accuracy. This classification accuracy (ranges between 0 and 1) is the fitness value of a particular CNN architecture.

VI. EVALUATION

In this section, the proposed GA hyper-parameters tuning technique is used to obtain the best CNN architecture for the MNIST dataset.

In this paper tournament selection, single-point crossover and multiple point mutation are the genetic operations performed on the chromosomes. The parameters of the chosen genetic operations are listed in Table 2.

TABLE 2. PARAMETERS OF THE GENETIC OPERATIONS

Parameters	Value
Tournament selection size	2
Crossover Probability	50%
Mutation probability	80%
Genes Mutated	10%

A. Experimental Setup

The Genetic Algorithm based CNN architecture generator has been implemented in python 2.7. Tests were carried on an AWS instance running Ubuntu Server 16.04 LTS.

This instance is powered by an AWS-Specific version of Intel's Broadwell processor, running at 2.7 GHz. It incorporates up to 8 NVIDIA Tesla K80 Accelerators, each running a pair of NVIDLA GK210 GPUs. Each GPU provides 12 GiB of memory (accessible via 240 GB/second of memory bandwidth), and 2,496 parallel processing cores [9].

The Genetic algorithm tuner was implemented with the MNIST dataset with 50,000 images as its training set and another 10,000 images as its testing set. The aim of the tuner was to generate a CNN architecture with the best architecture such that the classification accuracy of the testing set is very high.

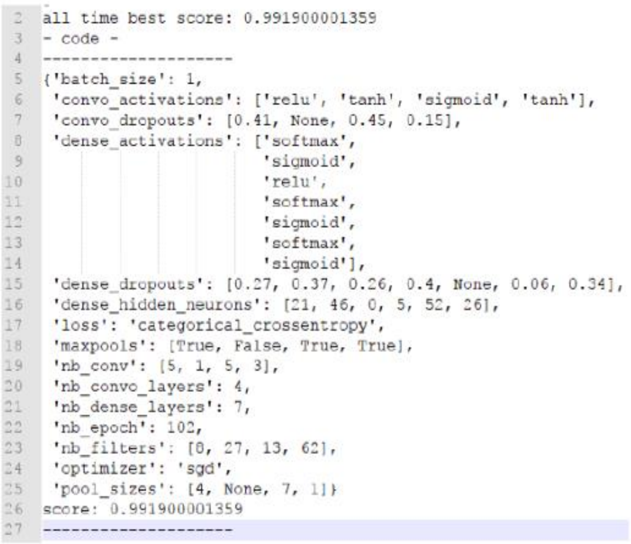
VII. RESULTS

Genetic algorithm with 10 chromosomes generated randomly was executed 10 times, each time with randomly chosen chromosomes. It was noted that each run of GA tuner gave the best chromosome which represented a CNN architecture with an accuracy of greater than 90%. This can be seen in Table 3. However, the best of the 10 runs gave an accuracy of 99.2%. This particular architecture was chosen.

TABLE 3. HIGHEST FITNESS VALUES OBTAINED DURING EACH OF THE 10 EXPERIMENTS

Exp. No.	Highest Fitness Value
1	0.984499992943
2	0.973899998105
3	0.988800008184
4	0.991900001359
5	0.947799991965
6	0.949000005102
7	0.983099997652
8	0.979799999475
9	0.956399999567
10	0.972350000068

The generated output after GA tuning in shown in Figure 8



```
all time best score: 0.991900001359
- code -
--------------------
{'batch_size': 1,
 'convo_activations': ['relu', 'tanh', 'sigmoid', 'tanh'],
 'convo_dropouts': [0.41, None, 0.45, 0.15],
 'dense_activations': ['softmax',
                       'sigmoid',
                       'relu',
                       'softmax',
                       'sigmoid',
                       'softmax',
                       'sigmoid'],
 'dense_dropouts': [0.27, 0.37, 0.26, 0.4, None, 0.06, 0.34],
 'dense_hidden_neurons': [21, 46, 0, 5, 52, 26],
 'loss': 'categorical_crossentropy',
 'maxpools': [True, False, True, True],
 'nb_conv': [5, 1, 5, 3],
 'nb_convo_layers': 4,
 'nb_dense_layers': 7,
 'nb_epoch': 102,
 'nb_filters': [8, 27, 13, 62],
 'optimizer': 'sgd',
 'pool_sizes': [4, None, 7, 1]}
score: 0.991900001359
--------------------
```

Figure 8. Generated CNN architecture after GA tuning

Figure 8 shows the best values for hyper-parameters shown in Table 2 reached by the algorithm.

VIII. CONCLUSION

In this paper, we carried out an experiment to see if genetic algorithm can be used to automatically generate good CNN architectures without any human intervention. The basic techniques of the Gas are designed to simulate processes in natural systems necessary for evolution and; especially those that follow the principles first laid down by Charles Darwin-“survival of the fittest.”

Our simulation results for finding the best CNN architecture using GA to tune the hyper-parameters lead to the generation of architecture which yielded an accuracy rate of more than 90% for the classification of the MNIST dataset. The best of the 10 runs gave an accuracy of 99.2%.

Therefore, it can be concluded that GA have the potential of generating successful CNN architectures automatically.

REFERENCES

[1] Y. B. a. A. C. Ian Goodfellow, Deep Learning, 2016.

[2] Y. J. L. D. B. B. D. J. S. G. H. P. G. I. H. D. H. R. E. a. H. W. LeCun, “Handwritten digit recognition:Applications,” IEEE Communications Magazine, pp. 41-46, 1989.

[3] M. Matusugu, M. Katsuhiko, M. Yusuke and K. Yuji, “Subject independent facial expression recognition with robust face detection using a convolutional neural network,” Neural Networks, vol. 16, pp. 555-559, 2003.

[4] D. H. Hubel and T. N. Wiesel, “Receptive fields and functional architecture of monkey striate cortex,” The Journal of Physiology, vol. 195, pp. 218-243, 1968.

[5] A. karpathy, “CS23In Convolutional Neural Networks for Visual Recognition,” 2016. [Online]. Available: http://cs23In. github.io/convolutional-networks/.

[6] ujjwalkarn, “An Intuitive Explanation of Convolutional Neural Networks,” 11 August 2016. [Online]. Available:https://ujjwalkarn.me/2016/08/11/intuitive-explanation-convnets/.

[7] “Genetic Algorithms,” [Online]. Available: https://www.doc.ic.ac.uk/~nd/surprise_96/journal/voll/hmw/article.html#top.

[8] L. N. d. Castro, “Genetic Algorithms,” in Fundamentals of Natural Computing; Basic Concepts, Algorithms, and Applications, Chapman and Hall/CRC, 2006, pp. 88-91.

[9] J. Barr, “New P2 Instance Type for Amazon EC2-Up to 16 GPUs,” 29 September 2016 [Online]. Available: https://aws.amazon.com/blogs/aws/new-p2-instance-type-for-amazon-ec2-up-to-16-gpus/.

[10] A. Gibiansky, “Convolutional Neural Networks,” 24 February 2014. [Online]. Available: http://andrew.gibiansky.com/blog/machine-learning/convolutional-neural-networks/.

Research on ISPs Selection Technology Based on Network Quality Monitoring

Wang Jian
School of Computer Science and Engineering
Xi'an Technological University
Xi'an, 710021, China
E-mail: 470081729@qq.com

Wang Zhongsheng
School of Computer Science and Engineering
Xi'an Technological University
Xi'an, 710021, China
E-mail: wzhsh1681@163.com

***Abstract*—With the development of the network era, ISP have become more than one. Users can choose ISP according to personal preference when buying broadband, to provide users with the right to choose. However, with the improvement of network speed and bandwidth, people still feel the network congestion during peak hours. In this paper, users can choose different ISPs dynamically based on network quality monitoring. In this paper, the HTTP protocol of the application layer is used to accurately measure the real Internet situation of users, and the PC is used to actively send data to two network links at the same time, and finally the feedback parameter information can be obtained. According to this protocol, the real loading time of data can be accurately measured and the optimal ISP link can be selected. Then, a one-to-one relationship between the client address and different ISPs is established, and this relationship is saved in the routing table. Users will dynamically choose different ISPs by using the network address translation.**

Keywords-network congestion; HTTP protocol; Network quality; Network address translation; Dynamic selection

I. NETWORK QUALITY MONITORING METHOD

Network measurement is obtaining the parameters and performance of the network through quantitative measurement. Usually, the main parameters of network measurement include congestion, delay, bandwidth, packet loss rate, throughput response speed and other application quality of service. Measurements can be divided into different types. According to the network level involved in the measurement, it can be divided into network layer measurement and application layer measurement. The measurement method of this paper uses the means of application layer protocol. Because there are bottlenecks in the network switching interfaces between different ISPs and the use of various filtering technologies, the quality of the network layer is not equal to the quality of the application layer, so choosing application layer measurement can more accurately grasp the network conditions of applications. The application layer measurement method chosen in this paper can be divided into active measurement, probe measurement and proxy measurement according to the difference between measurers and participants [2].

A. *Active Measurement*

The active measurement method relies on sending the measurement traffic to the target node, and then receiving the return traffic to measure each index, to obtain the network quality.

Autonomous measurement method is a kind of the active measurement, which sends a service request to the network by simulating the user's behavior, and then obtains the index parameters of the network by monitoring the time of each step of the request. Its executor can be any network device. The purpose of autonomous measurement is to obtain the quality information of the network where the measuring point is located by measurement.

Passive measurement is the opposite of active measurement. It means that the measurer will not actively send test data to the network, but will measure all traffic passing through the measurer by means of monitoring, and then analyze the traffic data, to obtain the network status data.

DOI: 10.21307/ijanmc-2021-025

B. Probe Measurement

Probe is a kind of network equipment installed at a specific position in the network. It has the functions of receiving instructions, detecting network conditions, collecting data, and reporting results. It is also a network node, but its main function is to collect network quality data. When it is used, it is connected to the subnet to be monitored, and it can obtain the situation of the network through measurement.

Probe measurement is to collect the network traffic passing through the node through the probe, analyze and extract the service characteristics to obtain the performance data. Probe measurement mainly finds and observes the behavior of the network at a special node.

C. Agent Measurement

Agent monitoring mode is a network quality monitoring method that uses agent technology to access target sites through agents. The agent mode solves the problem of how to monitor when there is no connection between the measurement point and the target site. If the measurement point and the target site cannot be directly connected for various reasons, but the measurement point can access the proxy server, and the proxy server can also access the target site, the proxy server can be used as an intermediary to complete the monitoring task between the measurement point and the proxy server.

According to the above measurement methods, this paper mainly adopts the active measurement method to actively send data packets to the network, and then obtains the network quality by monitoring the returned multiple parameter information.

II. APPLICATION LAYER PROTOCOL

Hypertext transfer protocol (HTTP) is an application layer protocol for distributed, cooperative and hypermedia information systems. HTTP is a standard for client-side and server-side requests and responses. By using web browser, web crawler or other tools, the client initiates an HTTP request to the specified port on the server. We call this client the user agent. The answering server stores some resources, such as HTML files and images. We call this reply server the origin server. There may be multiple "middle layers" between the user agent and the source server, such as proxy server, gateway or tunnel.

Although TCP / IP is the most popular application on the Internet, HTTP protocol does not specify that it must be used. In fact, HTTP can be implemented on any internet protocol or other networks. HTTP assumes that its underlying protocol provides reliable transmission. Therefore, any agreement that can provide such guarantee can be used by it. That is, it uses TCP as its transport layer in the TCP / IP protocol family.

Usually, an HTTP client initiates a request to create a TCP connection to the specified port of the server. The HTTP server listens for client requests on that port. Once the request is received, the server will return a status to the client, and the returned content, such as the requested file, error message, or other information.

A HTTP request is mainly divided into four stages: request initiation stage, connection request stage, data transmission stage and request end stage [3]. As shown in Fig. 1.

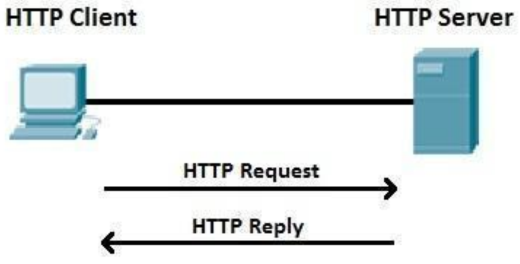


Figure 1. HTTP protocol transmission process.

More specific indicators can be summarized from these four stages, namely: parsing time, connection establishment time, data download time and data download rate. These indicators cover the whole life cycle of a network request, so they can reflect the network quality.

DNS resolution time: refers to the conversion process from domain name to address.

TCP connection establishment time: connection time refers to the time consumed by three handshakes before data transmission between the client and the server.

Data download time: the total time from the first byte of data to the completion of receiving the last byte.

Data download rate: the download rate is obtained by dividing the downloaded data size by the data download time.

A complete website must go through the first three stages from opening to loading. The total time is equal to DNS resolution time, connection establishment time and data download time.

III. NETWORK QUALITY MONITORING PROCESS

We selected several popular websites for monitoring, including 18 websites in search engine, news, shopping, video, blog, CHN domain name website and so on. As shown in Table 1.

TABLE I. LIST OF TEST WEBSITES

Search engine	http://www.baidu.com http://www.sougou.com
News	http://news.sina.com.cn http://www.cnn.com http://www.foxnews.com http://news.qq.com
Shopping	http://www.taobao.com http://www.jd.com http://www.pinduoduo.com
Video	http://www.youku.com http://www.iqiyi.com http://www.mgtv.com
Blog	http://blog.163.com http://www.blogchina.com http://liog.sina.com.cn
.chn	http://www.sjz.chn http://www.ijanmc.chn http://www.iccnea.chn

IPV9 network is an independently developed controllable network in China with the domain name “.chn”. It has a huge address space, with a default address number of 256 bits and a maximum address number of 2048, perfectly compatible with both IPv4 and IPv6 networks. IPV9 features self-controllable, large number of addresses, low latency, fast number of addresses, security, and compatibility with both IPv4 and IPv6 networks. It can also add geospatial information to its domain name to accurately determine its location.

In this paper, the quality of the application layer network is determined through the actual measurement and analysis of the four indicators of HTTP protocol. The measurement process of the indicators is shown in Fig. 2.

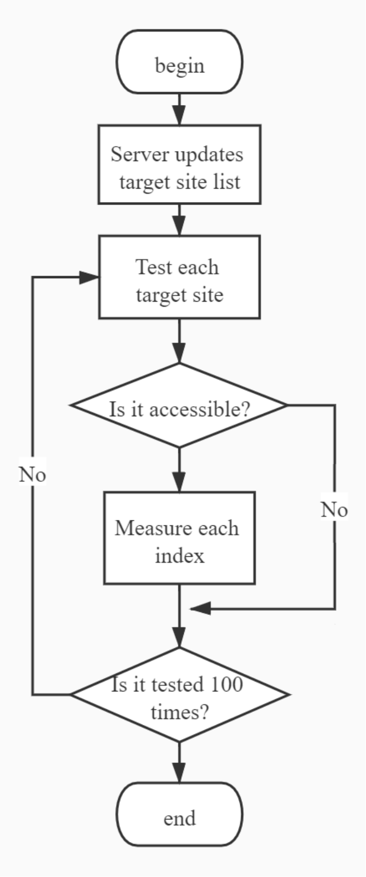


Figure 2. Flow chart of index measurement.

At the beginning, the program updates the list of target sites from the server. For each target site, the measurement program tests its accessibility. If it is accessible, count the index measurement results of this measurement. If it is inaccessible, the measurement results will be recorded as request failure. Each station needs to measure 100 times.

Through two network links, Unicom and Telecom, the Internet speed of these 18 stations is measured. Calculate the average value, and finally get the optimal network link. As shown in formula (1).

$$T_{ave} = \frac{t_{DNS} + t_{TCP} + t_{data}}{N} \qquad (1)$$

T_{ave} represents the final average time, t_{DNS} is the total DNS resolution time, t_{TCP} is the total time to establish TCP transmission channel, t_{data} is the total data download time, and N is the total number of site tests.

Upload the measured total time and average time of each site to the data center, compare which of the two links takes the least time, and get the best link after comparison.

IV. Selecting ISP dynamically

A. The principle of dynamic ISP selection

The router is built between the user and ISP, which can connect two different network links at the same time. The router provides routing and network address translation functions. The user interface is internal IP, and the interface provided by ISP is external IP, which establishes a one-to-one correspondence between internal IP and external IP, and records this relationship in the routing table. Users can dynamically select different networks.

With the development of The Internet and the increase of network applications, IPv4 address exhaustion has become a bottleneck restricting the development of the network. Although IPv6 can fundamentally solve the problem of insufficient IPv4 address space, at present many network devices and network applications are still based on IPv4, so before IPv6 is widely used, the use of some transitional technologies is the main technical means to solve this problem.

Network Address Translation (NAT) is an Internet technology that can convert the private address of the internal network into public IP [8]. It can not only share a public IP address, but also use the bandwidth provided by the local network service provider to access the Internet safely and at high speed, hiding and protecting the computers inside the network.

In this example, when an internal host needs to communicate with a destination host on the public network, the gateway selects an unused public address from the configured public address pool and maps it. Each host is assigned a unique address in the address pool. If the connection is not needed, the address mapping will be deleted and the public address will be restored to the address pool. After receiving the reply packet, the gateway transforms the packet again based on the previous mapping and forwards the packet to the corresponding host. After the IP addresses in the dynamic NAT address pool are used up, other hosts can access the public network only after the occupied public IP addresses are released.

The steps for the user to dynamically select a network are as follows:

Step1: Connect the user interface with the external interfaces of multiple regional interconnection ISPs.

Step2: Set different internal IP addresses on one side of the intermediate router, and set the external IP addresses provided by different regional interconnection ISPs on the other side of the intermediate router.

Step3: Carry out network address conversion, establish a fixed one-to-one correspondence between the internal IP address of the user terminal and the external IP address of the Internet provider, and record this fixed correspondence as a static route in the intermediate routing table.

Step4: The user actively sets the internal IP address of the user terminal in the router list in the user computer according to his preference. Fig. 3 shows the flow chart of the realization of dynamic network selection.

B. Dynamic selection of ISP design

Set up an intermediate router with NAT function between the user terminal and ISP, and the intermediate router assigns different internal IP to users. As shown in Fig. 4, the IP address assigned by ISP1 to the intermediate router port is 200.10.45.253, and the IP address pool assigned to the user terminal interface is 200.10.46.0/24. The IP address assigned by ISP2 to the intermediate router is 200.33.67.253, and the IP address pool assigned to the user interface is 200.33.68.0/24. These addresses are not virtual IP addresses on the Internet.

Users who want to change ISP1 to ISP2 must return the address pool 200.10.46.0/24 assigned by ISP1 to ISP1, and then assign the IP address 200.33.68.0/24 provided by ISP2 to every computer in the community. Obviously, this method is inconvenient, and it also leads to the waste of IP addresses.

We set up a local area network among users, using the internal IP address, such as 192.168.0.0/8. When the user 192.168.0.5 sends a data packet to the 203.120.10.30 on the Internet, the intermediate router first finds the best route in its routing table, and the next hop is to reach ISP1 through the port with the address of 200.10.45.253, so NAT converts 192.168.0.5 into an external address, such as 200.10.46.8. Thus, in 203.120.10.30's view, the datagram was sent from 200.10.46.8. When 203.120.10.30 returned the datagram, it was sent back to 200.10.46.8, and NAT converted it into 192.168.0.5. In this process, some public IP is reserved for private network reuse, which greatly reduces the usage of addresses.

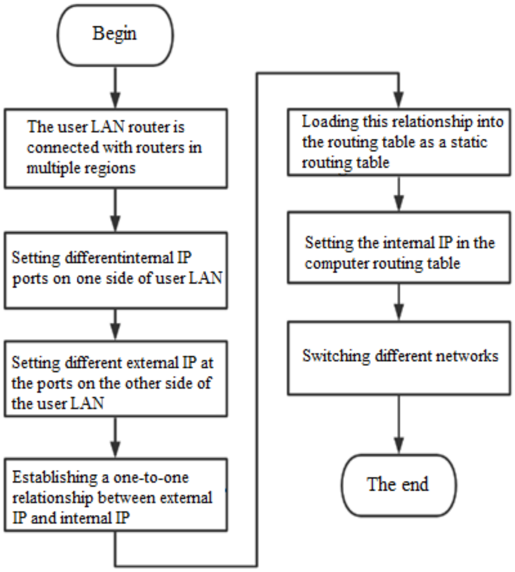

Begin
The user LAN router is connected with routers in multiple regions
Setting differentinternal IP ports on one side of user LAN
Setting different external IP at the ports on the other side of the user LAN
Establishing a one-to-one relationship between external IP and internal IP
Loading this relationship into the routing table as a static routing table
Setting the internal IP in the computer routing table
Switching different networks
The end

Figure 3. Flow chart of user's dynamic selection of network.

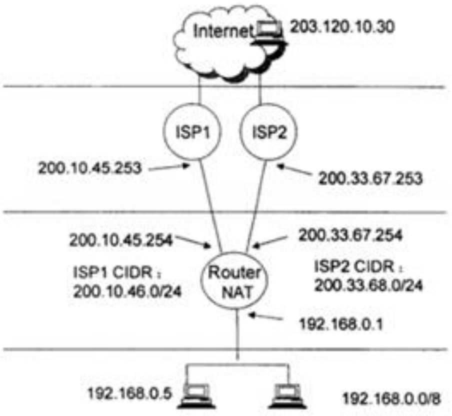


Figure 4. Connection diagram for users to dynamically select ISPs to access the Internet.

C. *Improvement of dynamic ISP selection method*

For many users, at this time, the user LAN can be divided into several subnets, such as LAN1, LAN2 and LAN3 in Figure 4.3. Each subnet relates to two NAT devices through different switches, and then connected to the internet through R1 or R2, and then a one-to-one correspondence between the internal IP address and the external IP address of the ports on both sides of the router is established. When 192.168.1.5 in LAN1 wants to access ISP2, 192.168.1.2 is uniquely set in the router list of its computer. When you want to access ISP1, you will set 192.168.1.1 uniquely in the router list of its computer. Use address translation function to dynamically select different ISPs.

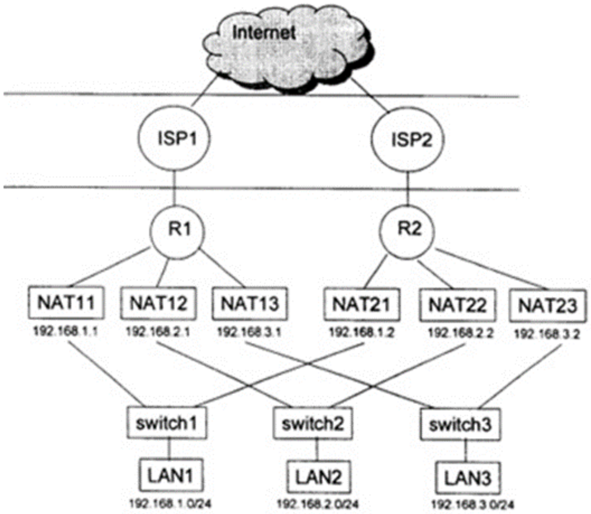


Figure 5. an application example in many users.

The advantage of this design model is that it can accommodate more users to switch between different IPS, and there will be no user waiting when switching ISPs.

V. CONCLUSIONS

The main purpose of this article is in order to be able to let the user the choice of dynamic network, this paper studied from two aspects: according to the source address to specify different ISP link, through active measuring the parameters of the HTTP protocol feedback information can accurately simulate the behavior of the user access to the network, according to user's access to the network consumption time measured real-time network quality data; Statistics on the network quality of different ISPs arriving at a particular site. The one-to-one mapping between internal IP addresses and external IP addresses enables users to dynamically select different ISP links through network address translation.

REFERENCES

[1] G. Eason, B. Noble, and I. N. Sneddon, “On certain integrals of Lipschitz-Hankel type involving products of Bessel functions,” Phil. Trans. Roy. Soc. London, vol. A247, pp. 529–551, April 1955. *(references)*

[2] Cai Zhiping. Network Measurement Technologies, Models and Algorithms based on Active and Passive Measurement[D]. National University of Defense Technology,2005.

[3] LI Kang, CHEN Qinghua, LU Jinxing. A Review of HTTP Protocol Research [J]. Information Systems Engineering,2021(05):126-129.

[4] ZHANG Hongli,FANG Binxing,HU Mingzeng ,JIANG Yu,ZHAN Chunyan,ZHANG Shufeng. Overview of Internet Measurement and Analysis [J]. Journal of Software,2003(01):110-116.

[5] Shigang, Chen, Nahrstedt, et al. An overview of quality of service routing for next-generation high speed networks: Problems and. [J]. IEEE Network, 1998.

[6] Sun Liming. The choice and implementation of multi-route campus network access strategy [J]. Journal of Huaibei Coal Normal University (Natural Science Edition),2004(04):69-73.

[7] LIU Haitao,HUANG Jialin. Application of Policy Routing Technology in Multi-egress Campus Network [J]. Computers and information technology,2002(04):51-53.

[8] LIU Zhan. Application Analysis of NAT Technology in Network Edge Computing [J]. Decision Exploration (Middle),2020(02):81.

[9] HUANG Min,ZHANG Weidong. Network design and practice based on PBR [J]. Computer application,2002(05):72-73.

Using Text and Visual Cues for Fine-Grained Classification

Zaryab Shaker
School of Computer Science and Engineering
Xian Technological University
Xian, China
E-mail: zaryabkhan0346@gmail.com

Muhammad Adeel Ahmed Tahir
School of Computer Science and Engineering
Xian Technological University
Xian, China
E-mail: adikhan0313@gmail.com

Feng Xiao
School of Computer Science and Engineering
Xian Technological University
Xian, China
E-mail: xffriends@163.com

***Abstract*—Text is an important invention of humanity, which plays a key role in human life, so far from dark ages. Text in image is closely related to the scene or a product and is widely used in vision based application. In this paper we are addressing the problem of visual understanding with text. The main focus is combining textual cues and visual cues in deep neural network. First the text is recognized and classified from the image. Then we combine the attended word embedding and visual feature vector which are then optimized by CNN for Fine-grained image classification. We carried out the experiments on soft drink dataset in Pakistan. The results shows that the system achieves significant performance which can be potentially beneficial for real world application e.g. product search.**

Keywords-Scene Text; Product Text; Fine-Grained Classification; Convolution Neural Network; Attention; Product Search

I. INTRODUCTION

Fine-Grained image classification [1, 28] is a real-world emerging problem and it has received great attention from research communities around the globe. In computer vision, fine grained [1] image involves the problem of assigning images to classes where different instances of different classes differ slightly in their appearances e.g., flower species, animal species, product/place types. Therefore, fine-grained image classification [28] is a challenging assignment due to the slight variations among highly-confused categories of instances belonging to various classes of objects, which are hard to distinguish. Further, in some of the cases, human intervention or specific knowledge of a particular domain is also required to perform precise fine-grained image classification [1].

For some years, fine-grained image classification [1] is also being applied for natural scene classification. This involves the natural images of wide and diverse nature. It is witnessed that classification of shops, variety of products in shops, etc. are the considered areas where fine-grained image classification is being used [4]. Using fine-grained image classification [28] techniques on the soft drink dataset is an area that has received limited attention from the researchers, whereas this area has also exciting applications in restaurants and shops, where automated orders could be places once a specific brand of soft drink is going to be out of stock.

In this work, we have exploited text and visual cues in form of features to be used with Convolutional Neural Network for attaining good performance in the fine-grained classification [1, 28] of soft drinks. To the best of our knowledge, it

DOI: 10.21307/ijanmc-2021-026

is a unique application of a classification technique in the domain of fine-grained image classification of soft drink datasets.

The second section of this paper discusses the proposed classification technique, the third section involves results and discussion and conclusions are drawn in the last section.

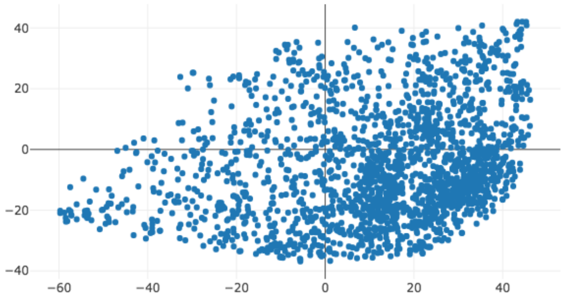


Figure 1. T-SNE Plot for word Embedding

II. Literature Survey

A. Text detection and recognition

Text in images carries a high level of information which makes this property of images very rich in computer vision as well as for humans. The information encoded in the text can be very beneficial for many computer vision applications. Text detection and recognition [7] face some challenges like the diversity of nature, the complexity of background, interference factors, etc. The first novel approach of a real end-to-end model for text detection and recognition in a scene was proposed in 2010 by Neumann et al [8]. It achieved a highly significant increase in recognition rate from 53% to 72% on the Char74k dataset (de Campos et al.) but there is a weakness in this system is that it was only applicable to horizontal texts. Later on Coates et al. [9] 2011 apply large-scale algorithms to build highly effective classifying model for both detection and recognition. Their end-to-end system has high accuracy and performs well on complex natural images but the drawback is that it requires a relatively large volume of training data.

Further in 2014 Jaderberg et al [10] addresses the problem of text detection and recognition by generating text proposals with CNN and provides an end-to-end system for reading text in natural scene images. That system was capable of both text spotting and image retrieval and perform excellently on complex natural images. Yao et al [11] present a unified framework for detecting and recognizing the text in images to handle the texts of different orientations. They also provide a method for 'search dictionary' to correct the recognition errors. The system achieves highly competitive performance, especially on multi oriented texts. Also, the works of Shiva kumara et al. [12] and C Yao et al. [13] realize the significance of multi-oriented text detection and recognition to the research community. In the paper of Yingying Zhu et al. [14], they discussed in detail the recent advances and future trends for scene text detection and recognition.

B. Fine-Grained Classification:

Fine-Grained classification [28] aims for the deep insight into image that is why this problem got a lot of attention from researchers around the globe. Many approaches have been developed to address the particular problem till now with the margin of improvement in the future.

Existing deep learning-based fine-grained image classification approaches [15] could be sub-classified into the following according to the use of additional information or human inference:

1) Approaches that directly use the general deep CNNs for image fine-grained classification [16].

2) Part detection and alignment-based approaches [17].

3) Ensemble of network-based approaches [18].

4) Approaches based on attention mechanisms [19].

The prior work in fine-grained classification [28] can be simply divided into two paths. The first is to detect the discriminative object parts in the image to compensate for nuisance variations such as pose. Many parts-based methods with geometric constraints have been proposed for bird classification [16] and dogs [20].

The second track is to derive discriminative and robust features. Classic hand-crafted feature descriptors such as the Scale Invariant Feature Transform (SIFT) [21], Histogram of Oriented Gradients (HoG) [22], and Color Histogram [23] other methods such as the Part-based One-vs-One Features (POOFs) [24] focus on modeling corresponding parts activation. Deep convolutional neural network (DCNN) approaches for general object classification achieve state-of-the-art performance for fine-grained classification by applying transfer learning [25].

C. Attention Mechanism

The idea of attention is one of the most influential ideas in deep learning allows the network to focus on specific aspects of a complex input. The main idea of the attention t mechanism is to allow the decoder to "look back" at the original input and extracts only the significant information that is important for decoding [27].

Consider we are attempting machine translation on the following sentence: "The cat is beautiful." If you can ask someone to pick out the keywords of the sentence, i.e. which ones describe the most meaning, they would likely say "cat" and "beautiful." Articles like "the" and "is" are not as relevant in translation as the previous words (though they aren't completely insignificant). Therefore, we focus our attention on important words. We use the attention mechanism on texts to get our most relevant words from text features and we put attention on the visual features to get our most relevant features like edges, color, size of object, etc.

The attention mechanism [27] scores each input word (via dot product with attention weights), then to create a distribution scores are passed through softmax function. An attention vector is produce by multiplying distribution with the context vector and then passed to the decoder. The advantage of attention are its ability to identify the information in an input most pertinent to accomplishing a task, increasing performance especially in natural language processing but it increases computations unlike to human.[30].

D. Multimodal Fusion:

Multimodal processing [35] significantly enhances the understanding, modeling, and performance of human-computer interaction. In multimodal fusion [31], user interaction with system is through various input modalities like speech, gesture, and eye gaze. In our context, different multimedia researchers presented different fusion strategies used for combining multiple modalities in order to an encoder-decoder architecture accomplish various tasks [28, 29, 33].

The literature on multimodal fusion [31] research is presented through several classifications based on the fusion methodology. The methods can be described from their advantages, weaknesses, basic concept, and their usage in various analysis tasks but multimodal fusion has several issues that influence the process such as contextual information, confidence level, synchronization between different modalities, etc. In 2016 [32] uses multilayer and multimodal fusion of deep neural networks for video classification. In 2017 [33] uses weakly paired multimodal fusion for object recognition. [34] Uses Multimodal deep networks for image-based document and text classification by introduce an end-to-end learnable multimodal deep network that jointly learns text and image features and performs the final classification based on a fused heterogeneous representation of the document. They validated their approach on the Tobacco3482 and RVL-CDIP datasets. In 2020 [28] did Fine-grained Classification by the Combination of Visual and Locally Pooled Textual Features.

III. PROPOSED METHODOLOGY

Our approach is to classify soft drinks images into their respective classes with the help of text and visual features. We extract textual and visual features from the input image with the help of different models [37, 40] and treat those features as input for our multimodal [31] which combines both the inputs to anticipate the classification of the given image. Textual cues play a key role in fine-grained classification [28], especially in the classification of business places such as bakery, café, bookstore, and daily use products. Multiple models have been developed to address the particular problem. These models assist us to extract the textual information that is highly useful for image classification. We adopt word2vec [40].

Visual features are the second input to our modal. The visual feature is the information about the contents of an image which describes its specific structures such as shapes, edges, objects, patterns, and colors i.e. properties. In our case, we extract the visual features (works as second input) with the help of VGG pertained modal [37] on ImageNet [36] by fine-tuning the modal with our dataset. These visual features work as building blocks with the texture input to give us the desired results.

The 224x224 Input images are transferred to VGG model [37] for visual features extraction. We extract visual features by importing the VGG [37] model from tensor flow keras with the pretrained weights. The top layer is set to false when loading the model. Further, we unfreeze the last five layers to fine tune the modal. After defining the model, PIL object of the image has to be converted in a pixel data NumPy array where we only have one sample and the values are then appropriately scaled to get the features. We get the feature vector 'y_f' from the last max-pooling layer as a 4096 dimensional features. Feature extraction part is from the input layer to the last max pooling layer.

At the same time, the input is send to OCR for character recognition. OCR gives us the classification of text by localizing and recognizing the text. Then OCR saves the recognized text in a file which is then passed to word2vec [40] model for textual representation ' x_f'. Word2Vec [40] is a two-layer neural networks which has been trained for the reconstruction of linguistic contexts of words. It takes a large corpus as its input and produces a vector space, of given dimensions, with each unique word in the corpus being assigned a corresponding vector in the space. The purpose and usefulness of Word2vec [40] is to group similar words vectors together in vector space. That is, it detects similarity between vectors by using the 'cosine similarity' function. Let 'a' be the first vector and 'b' be the second vector,

$$\text{Cos}(a, b) = \frac{a.b}{||a||*||b||}$$

$$x_f = \cos(a, b)(1)$$

After that, we put an attention mechanism [30] on both inputs of multimodal [31] i.e. visual features and textual features. There can be some recognized text that is more relevant than others at the moment of discriminating similar classes. So we need to capture the inner correlation between the textual and visual features. The attention mechanism learns a tensor of weight that is used between the visual features and textual features. Let 'X' be the extracted textual features and 'Y' is visual features and weight is 'W'. We compute the attention mechanism by:

$$w_a = \text{Softmax}\ (\tan h\ (y_{fa}^{t} \,.\, \text{W}.x_f))$$

$$x_{fa} = w_a \,.\, x_f \qquad (2)$$

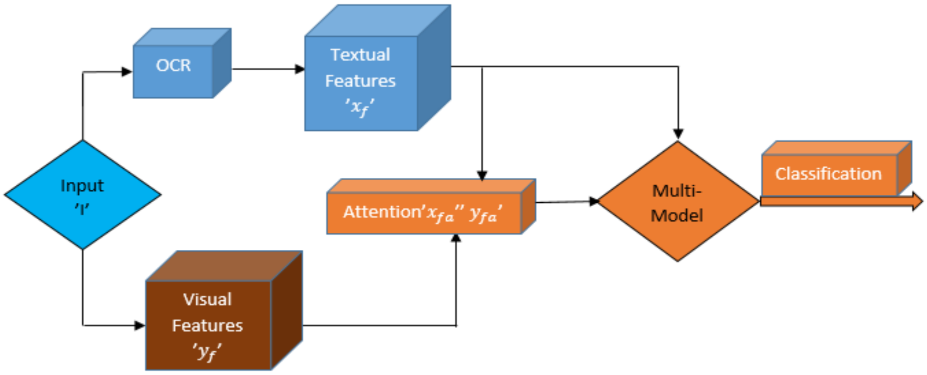


Figure 2. Proposed Model Pipeline_ Text and Visual features are attended and Combined for Fine Grained Classification.

The resulting normalized attended vector'w_a', is multiplied with the textual features 'x_f' to obtain the final attended textual features 'x_{fa} '. The obtained textual features 'x_{fa} ' and the visual features'y_{fa}' are concatenated in the multimodal to form the final features by

$$Z = [x_{fa} + y_{fa}] \qquad (3)$$

Finally, the resulting vector serves as input to a final classification layer that outputs the probability of a given class based on low-rank bilinear pooling operation.

IV. Experiments and results

A. Dataset:

We have collected the new dataset of soft drink bottles of 10 classes in Pakistan with 375 original images. The dataset contains several occluded, rotated, low quality and blurred text instances which increases the difficulty of performing successful text recognition. Due to limited resources, our dataset is not fully organized and has many limitations. The images are divided into 2 sets having 200 training images, 175 test images.

B. Implementation Details:

We start by augmentation the training images flipped vertically and horizontally to make 2 more images of every image to avoid the problem of overfitting for feature extraction mode. So the total number of training images is 600. We load the image and convert it to array using keras preprocessing. We also expand the dimensions and use a preprocess function in keras to fit the image according to our model. Then the predicted output is sent to for attention.

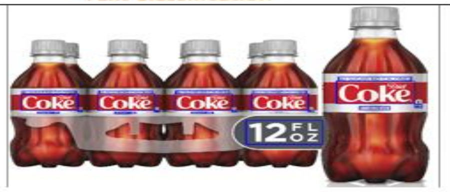
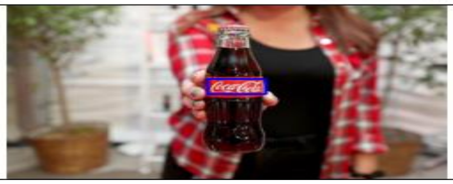
We extract text from all of the images by using the combination of tesseract [39] and easyOCR [38]. EasyOCR is used to localize a text and tesseract is used for the recognition of the text. The extracted text is saved in two files each set of training set text and test set text. Then these sets are loaded for word embedding [40].To recognize context meanings we use two layer shallow neural network to describe word embedding. We import word2vec [40] from genism library.

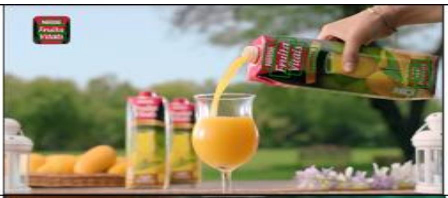
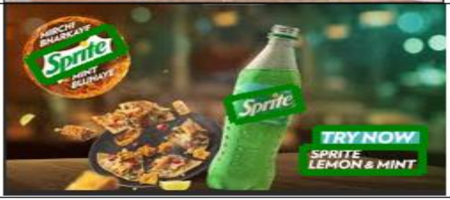
We put the attention on both textual and visual features for the fine-grained classification [28] of our dataset. The network is trained for 5 epochs with Adam optimizer. The batch size employed in all our experiments is pre-defined from a library 'config', with a learning rate of 0.0001, momentum of 0.9.

These experiments were implemented by using tensor flow deep learning framework on a simple laptop of 8 GB ram and 2.7 GHz (i7).

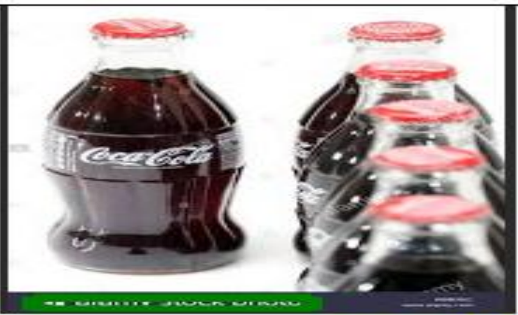
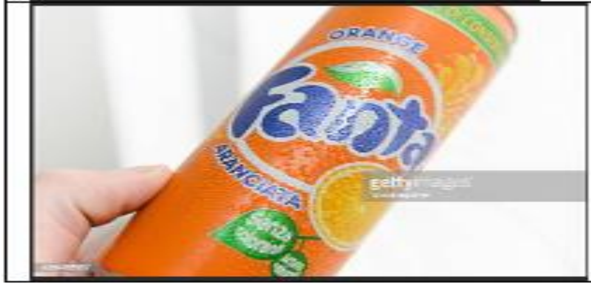
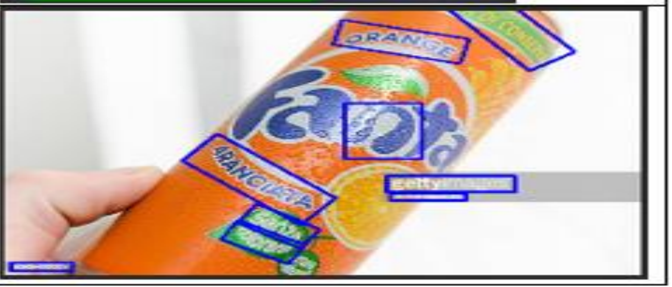
C. Results:

Random results of some classes are shown below.

Original Image	Text Classification	G.T	Prediction
		Coke	Coke (0.972372)
		coca cola	Coca cola (0.776379)

		Nestle juice	Nestle juice (0.896555)
		Sprite	Sprite (0.984277)

Bad results are because of lot of noise or the quality of the image.

Original Image	Results
	
	

D. *Comparison Graphs:*

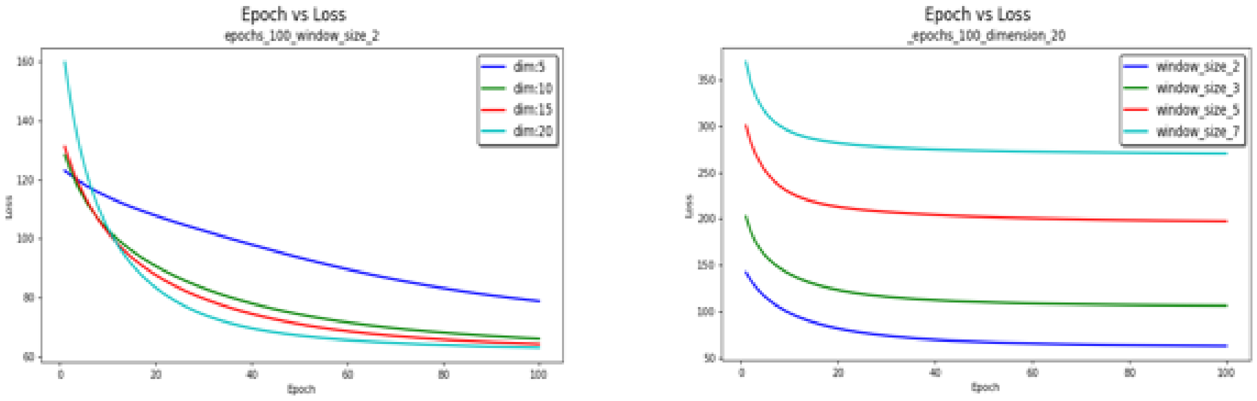


Figure 3. Visualization of Epoch vs Loss with different window size and dimension

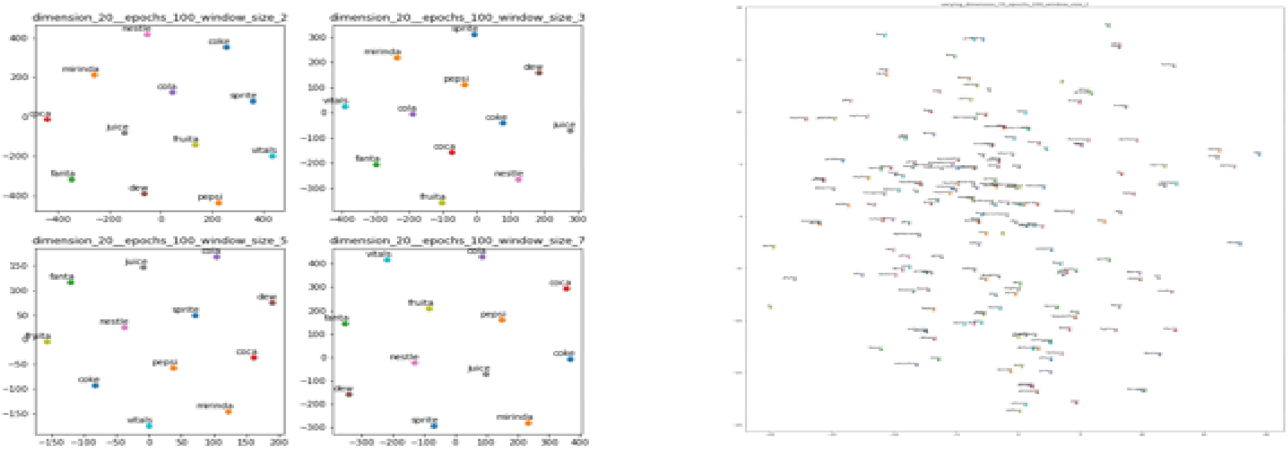


Figure 4. Parameters visualization with different window size and dimension

V. Conclusion

In this paper we demonstrated the importance of textual and visual cues for fine grained classification. We developed a frame work for precise classification of soft drink bottles by combining pre-trained models. The results show that the textual features plays more important role then visual features for classification of real life products.

Furthermore, as the system is created with low resources it can be modified and enhanced for better performance on large datasets and real world classification applications.

References

[1] Z Akata, S Reed, D Walter, H Lee, "Evaluation of output embeddings for fine-grained image classification, " pattern recognition," 2015.

[2] X He, Y Peng, "Fine-grained image classification via combining vision and language," Computer Vision and Pattern Recognition, 2017.

[3] Maron, AL Ratan," Multiple-instance learning for natural scene classification," ICML, 1998.

[4] W Geng, F Han, J Lin, L Zhu, J Bai, S Wang, "Fine-grained grocery product recognition by one-shot learning," Proceedings of the 26th ACM international conference on Multimedia," 2018.

[5] S Albawi, TA Mohammed, "Understanding of a convolutional neural network," IEEE, 2017.

[6] SE Umbaugh," Digital image processing and analysis: human and compute vision applications with CVIPtools," Amazon book, 2010.

[7] Q Ye, D Doermann, "Text detection and recognition in imagery: A survey," IEEE transactions on pattern analysis, 2014.

[8] L Neumann, J Matas, "A method for text localization and recognition in real-world images," Asian conference on computer vision, 2010.

[9] A Coates, B Carpenter, C Case, "Text detection and character recognition in scene images with unsupervised feature learning," IEEE, 2011.

[10] M Jaderberg, A Vedaldi, A Zisserman, "Deep features for text spotting," European conference on computer, 2014.

[11] C Yao, X Bai, W Liu, "A unified framework for multioriented text detection and recognition,"IEEE Transactions on Image Processing, 2014

[12] P Shivakumara, A Dutta, CL Tan, U Pal, "Multi-oriented scene text detection in video based on wavelet and angle projection boundary growing," Multimedia tools and applications, 2014.

[13] Z Zhang, C Zhang, W Shen, C Yao, "Multi-oriented text detection with fully convolutional networks,"pattern recognition, 2016.

[14] Y Zhu, C Yao, X Bai, "Scene text detection and recognition: Recent advances and future trends,"Frontiers of Computer Science, 2016.

[15] B Zhao, J Feng, X Wu, S Yan, "segmentation," International Journal of Automation, 2017.

[16] N Zhang, J Donahue, R Girshick, T Darrell, "Part-based R-CNNs for fine-grained category detection," European conference, 2014.

[17] E Gavves, B Fernando, CGM Snoek, "Fine-grained categorization by alignments," IEEE 2013.

[18] P Baraldi, M Compare, S Sauco, E Zio, "Ensemble neural network-based particle filtering for prognostics,"Mechanical Systems and Signal, 2013.

[19] F Fan, Y Feng, "D Zhao Multi-grained attention network for aspect-level sentiment classification," conference on empirical methods, 2018.

[20] OM Parkhi, A Vedaldi, A Zisserman, "Cats and dogs," IEEE conference, 2012.

[21] G Lowe, "Sift-the scale invariant feature transform," Int. J 2004.

[22] N Dalal, B Triggs, “Histograms of oriented gradients for human detection,” IEEE computer society conference, 2005.

[23] J Van De Weijer, C Schmid, J Verbeek, “Learning color names for real-world applications,” IEEE Transactions, 2009.

[24] T Berg, PN Belhumeur,“Poof: Part-based one-vs.-one features for fine-grained categorization, face verification, and attribute estimation,” Proceedings of the IEEE, 2013.

[25] KC Kamal, Z Yin, B Li, B Ma, “Transfer learning for fine-grained crop disease classification based on leaf images,” IEEE, 2019.

[26] V Badrinarayanan, A Kendall, “Segnet: A deep convolutional encoder-decoder architecture for image segmentation,” IEEE transactions on, 2017.

[27] P Rodr ǵuez, D Velazquez, G Cucurull,” Pay attention to the activations: a modular attention mechanism for fine-grained image recognition,” IEEE Transactions, 2019.

[28] A Mafla, S Dey, AF Biten, L Gomez, “Fine-grained image classification and retrieval by combining visual and locally pooled textual features,” WACV, 2020.

[29] X Bai, M Yang, P Lyu, Y Xu, J Luo, “Integrating scene text and visual appearance for fine-grained image classification,” IEEE Access, 2018.

[30] K Cho, A Courville, Y Bengio, “Describing multimedia content using attention-based encoder-decoder networks,” IEEE Transactions on Multimedia, 2015.

[31] PK Atrey, MA Hossain, A El Saddik, MS Kankanhalli, "Multimodal fusion for multimedia analysis: a survey," Multimedia systems, 2010.

[32] X Yang, P Molchanov, J Kautz, “Multilayer and multimodal fusion of deep neural networks for video classification,” Proceedings of the 24th ACM, 2016.

[33] H Liu, Y Wu, F Sun, B Fang, “Weakly paired multimodal fusion for object recognition,”IEEE, 2017.

[34] N Audebert, C Herold, K Slimani, C Vidal, “Multimodal deep networks for text and image-based document classification,” Joint European Conference, 2019.

[35] P Maragos, A Potamianos, P Gros, “Multimodal processing and interaction: audio, video, text,” IEEE 2008.

[36] J Deng, W Dong, R Socher, LJ Li, K Li, “ImageNet,” IEEE, 2009.

[37] Karen Simonyan, Andrew Zisserman, “Very deep convolutional networks for large-scale image recognition,” Department of Engineering Science, University of Oxford, 2015.

[38] A Karnawat, K More, T Rade, B Rane, M Mulik, “A Survey on Easy OCR Techniques used to build Systems for Visually Impaired People,” ITB, 2016.

[39] R Smith, “An overview of the Tesseract OCR engine,” Ninth international conference on document analysis, 2007.

[40] KW Church, “Word2Vec,” Natural Language Engineering, 2017.

Research and Implementation of Emitter Threat Assessment Based on Distributed Simulation

Feng Xiaojuan
College of Ordnance Science and Technology
Xi’an Technological University
Xi’an, China
E-mail: 1115312460@.qq.com

Liang Xiangyang
School of Computer Science and Engineering
Xi’an Technological University
Xi’an, China
E-mail: liangxy@xatu.edu.cn

***Abstract*—The threat degree of radiation source detected by fighter in penetration operation is the main reference for fighter to make the next operation plan. The research objective of this project is to evaluate the threat of radiation sources detected by aircraft ,based on distributed interactive simulation technology, the system modeling of aircraft flight detection data and ground radar detection warning in distributed system is realized by building a variety of simulation modules such as ground radar model, aircraft radar model, coordinate transformation model and threat level evaluation model, the simulation environment system of airborne emitter warning is completed. In this experimental environment, through simulation, the real-time calculation of the threat factor of the aircraft radiation source can be carried out, and then the dynamic threat assessment of the radiation source target can be realized.**

Keywords-Distributed Simulation; Radar Detection; Coordinate Transformation; Threat Assessment

I. INTRODUCTION

In the modern battlefield, it is very important to accurately evaluate the target's electromagnetic interference and threat level, and reasonably formulate the fighter combat strategy, literature [1] points out that it is very important to improve the survival probability of fighter in battlefield. Literature [2] says that in fact, it is a very ideal state to calculate according to the real battlefield data. In the experimental stage of the algorithm, we need to simulate the flight data and radar data of the fighter in the battlefield through the computer simulation technology, and then carry out the level evaluation according to the simulation data to get the order of the threat coefficient, so as to provide reliable reference data for the formulation of the fighter combat strategy [3-5].

The research of this subject is divided into two modules, one is the realization of simulation model in distributed simulation system, which is realized by MFC in vs environment, the other is Qt development environment, which uses C + + language to realize the assessment of emitter threat.

In the flight trajectory of aircraft in simulated battlefield and the detection of ground radar, the data of combat object in battlefield is simulated by constructing distributed simulation system. Secondly, the modeling and simulation system of Emitter Threat Level Evaluation in penetration operation is constructed. The system can realize the evaluation of the threat level coefficient of the

DOI: 10.21307/ijanmc-2021-027

aircraft to the emitter. The data of the aircraft and the data of each emitter obtained from the data source simulation files are transformed through coordinates and the modeling of aircraft warning radar, The data ranking of the threat coefficient of radiation source to aircraft can be obtained. The interface of the system can see the position of the aircraft in the geographical coordinate system, the attitude angle of the aircraft and the corresponding dashboard display of each attitude angle. At the same time, it can intuitively see the position of the radiation source relative to the aircraft, and the data of the radiation source sorted according to the threat level. This is conducive to the fighter in the battlefield real-time combat strategy, improve the survival probability of the fighter in the battlefield has important practical significance.

II. SIMULATION SYSTEM

A. Construction target

With the development of modern electronic battlefield technology, more and more advanced weapons are put in. The addition of early warning radar, radar jamming, missile attack and other radiation sources leads to the short-term survival probability of fighters in the battlefield [6]. In order to make fighter play an efficient and long-term role in the modern electronic battlefield, it is necessary to calculate the real-time data of fighter and radar, and formulate a reasonable and efficient strategy for fighter [7]. Therefore, in order to ensure the survival probability of the fighter, we use simulation technology to simulate the flight trajectory of the fighter in the battlefield, as well as the data of the radar. Through the propulsion of the unit step, we can get the real-time data of the aircraft and the radar in each step, and then get the real-time data [8].

According to the simulation data, through coordinate transformation and level evaluation model, the order of the threat coefficient of aircraft radiation source at the current time is obtained.

B. The composition of simulation system

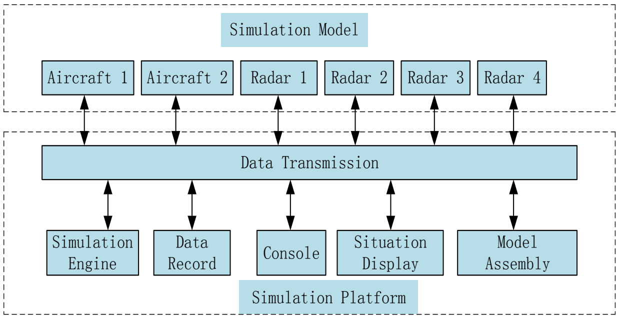


Figure 1. Simulation System Framework

As shown in Figure 1, the simulation system is mainly composed of two parts: simulation model and simulation platform.

The simulation model mainly includes aircraft 1, aircraft 2, radar 1, radar 2, radar 3, radar 4 and other combat objects.

The simulation platform mainly includes simulation engine, data recording, situation display, model assembly, console and so on.

III. SIMULATION MODEL OF SIMULATION SYSTEM

The main models used in the simulation system are aircraft radar model, ground radar model, coordinate transformation model, threat level assessment model, and the results of aircraft operation are displayed.

A. Detection model of ground early warning radar and aircraft radar

The key to the establishment of ground radar model is the radar detection range equation, that is, the calculation of echo signal power. The simplified formula of ground early warning radar and aircraft radar is as follows:

The detection equation of ground early warning radar is as follows：

$$P_{gr} = (P_{gt} G_{gt} \sigma \lambda^2) / (4\pi)^3 R^4 \quad (1)$$

Aircraft radar detection equation:

$$P_r = (P_{gt} G_{gt} G_r \lambda^2) / (4R)^2 \quad (2)$$

The meaning of each parameter in the formula is shown in Table 1.

TABLE I. PARAMETERS OF GROUND TO AIR RADAR DETECTION EQUATION

Parameter	Explain
P_{gt}	Transmitter power of ground early warning radar
G_{gt}	Antenna gain of ground early warning radar
G_r	Aircraft radar antenna gain
λ	wavelength
σ	Radar cross section of target
R	Distance between ground early warning radar and aircraft
P_{gr}	Target echo power received by ground early warning radar
P_r	Signal power of ground early warning radar received by aircraft radar

Assuming that the parameters and sensitivities of a certain type of ground early warning radar and aircraft radar $P_{\min}$ are known, when the ground early warning radar and aircraft situation have been determined, the sum is calculated by the distance between them and the values of other known parameters to judge whether the ground early warning radar and aircraft radar can detect each other and decide whether to alarm. Generally, the detection range of ground early warning radar is wider than that of aircraft radar. The comparison of discrimination results is shown in Table2.

TABLE II. COMPARISON OF GROUND TO AIR RADAR TARGET JUDGMENT RESULTS

Condition	Discriminant results
$P_{gr} < P_{\min地}$	Not detected
$P_{gr} >= P_{\min地}$	Detected
$P_r < P_{\min空}$	Not detected
$P_r >= P_{\min空}$	Detected

B. Coordinate transformation model

Coordinate transformation is an important step in this modeling and simulation. According to the position of the aircraft itself in the geographical coordinate system and the position of the radiation source in the geographical coordinate system, the following coordinate systems should be transformed: geographical coordinate system, geocentric coordinate system, North East coordinate system and carrier coordinate system. The purpose of coordinate transformation is to determine the position of radar in the aircraft coordinate system with the aircraft as the origin. The final position is expressed by azimuth and elevation. These two values are important parameters for the subsequent evaluation of radiation source level.

1) Coordinate transformation

a) Coordinate transformation process

The simulation system obtains the original data of the aircraft and the radiation source in the geographic coordinate system, and then transfers the obtained data into the geocentric rectangular coordinate system and the north sky East coordinate system, and then enters the carrier rectangular coordinate system according to the attitude angle of the read aircraft. After the conversion, the position of the radiation source in the carrier coordinate system can be obtained, thus the azimuth and height angle of the radiation source can be obtained. The flow chart of coordinate transformation is shown in Figure 2

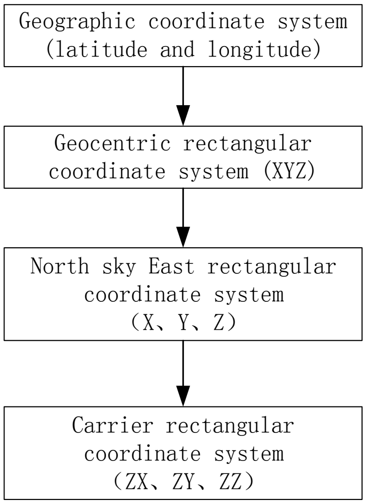


Figure 2. Flow chart of coordinate transformation

b) Geographic coordinate system to geocentric rectangular coordinate system

Latitude is the latitude in the geographical coordinate system, longitude is the longitude in the geographical coordinate system, altitude is the height in the geographical coordinate system. Longitude and latitude need to be converted into radians before entering the conversion formula, as shown below:

Convert latitude to radians:

$$B = \mathrm{la}titude * \pi / 180 \tag{3}$$

Convert longitude to radian:

$$L = longitude * \pi / 180 \tag{4}$$

The height is the directly used H (in meters).

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} (\mathrm{N}+\mathrm{H}) * \mathrm{cosB} * \mathrm{cosL} \\ (\mathrm{N}+\mathrm{H}) * \mathrm{cosB} * \mathrm{sinL} \\ (\mathrm{N} * (1-e^2)+\mathrm{H}) * \mathrm{sinB} \end{bmatrix} \tag{5}$$

As shown below, the corresponding relationship between geographic coordinate system and geocentric rectangular coordinate system is as follows,

c) Geocentric coordinate system to North Tiandong coordinate system

The B'and L' used here are the same as the above explanation. This coordinate system is based on the geocentric coordinate system. After rotating twice, the conversion of the geocentric coordinate system to the North Tian dong coordinate system can be completed. The following is the formula used in the conversion:

$$B^{'} = \begin{bmatrix} 0 & \sin B & \cos B \\ 1 & 0 & 0 \\ 0 & -\cos B & \sin B \end{bmatrix} \tag{6}$$

$$L^{'} = \begin{bmatrix} -\mathrm{sinL} & \mathrm{cosL} & 0 \\ -\mathrm{cosL} & -\mathrm{sinL} & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{7}$$

$$C = B^{'} * L^{'} = \begin{bmatrix} -\sin B \cos L & -\sin B \mathrm{sinL} & \mathrm{cosB} \\ -\sin B & \cos L & 0 \\ \cos B \cos L & \cos B \sin L & \sin B \end{bmatrix} \tag{8}$$

According to the linear transformation corresponding to the matrix, the corresponding relationship between geocentric rectangular coordinate system and North Tian dong rectangular coordinate system can be obtained as follows：

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = C * \begin{bmatrix} \mathrm{x} \\ \mathrm{y} \\ \mathrm{z} \end{bmatrix} \tag{9}$$

d) North Tiandong rectangular coordinate system transfer machine rectangular coordinate system

The north sky East rectangular coordinate system rotates around three axes, and the rotation angle is the attitude angle of the aircraft. According to the transformation of the following formula, the carrier coordinate system can be obtained:

$$\begin{bmatrix} ZX \\ ZY \\ ZZ \end{bmatrix} = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} * \begin{bmatrix} \cos\theta_g * \cos\theta_P - \sin\theta_P * \sin\theta_f * \sin\theta_g & \sin\theta_P * \cos\theta_f + \sin\theta_f * \cos\theta_P * \sin\theta_g & -\cos\theta_f * \sin\theta_g \\ -\cos\theta_f * \sin\theta_P & \cos\theta_P * \cos\theta_f & \sin\theta_f \\ \cos\theta_P * \sin\theta_g + \sin\theta_P * \sin\theta_f * \cos\theta_g & \sin\theta_g * \sin\theta_P - \sin\theta_f * \cos\theta_P * \cos\theta_g & \cos\theta_f * \cos\theta_g \end{bmatrix} \quad (10)$$

1) Ancient forest method

In the ancient forest method, the first step is to compare the importance of all the threat factors from top to bottom, and then quantify the importance of each threat factor, and give its specific value according to experience, so it needs a strong working experience to evaluate, which also determines the quality of the whole evaluation model. The second step is to standardize the importance of threat factors. The process of standardization is to deduce the standardized importance of each factor from bottom to top based on the standardized importance of the last item as 1. The third step is to divide the importance of the standardized threat factor by the sum of the importance of all the standardized threat factors to get the weight of the threat factor [9-10].

Here, the method to determine the weight in Table 3 is described:

a) The importance of threat factor, set as Rj, is determined according to the experience value in the battlefield.

*b) The importance of standardization, set as kJ, is based on the last item, which is set as 1 and calculated from bottom to top Kj=Kj+1*Rj.*

$$Ci = Ki / \sum Ki \quad (11)$$

c) Weight, set to CI, use the formula 11 to get.

TABLE III. WEIGHT VALUE DETERMINATION METHOD

Serial Number	Threat Factor	Importance of Threat Factors Rj	Importance of Standardization Kj	Weight Ci
1	type	4	24	0.72727
2	frequency	3	6	0.18181
3	distance	2	2	0.06061
4	position	1	1	0.03031
total			33	1.000

2) Membership function of each threat factor of radiation source

Membership function of emitter type: in electronic battlefield, the type of target emitter increases the threat degree to a great extent.

$$U\left(x1\right)=\begin{cases}0.1\ , & x1 = Rader \\ 0.9\ , & x1 = Missile\end{cases} \qquad (12)$$

Membership function of emitter frequency: in electronic battlefield, when the frequency of emitter increases, the threat coefficient increases.

$$U\left(x2\right)=\begin{cases}0\ , & 0 <= x2 <= 1000 \\ 1-\mathrm{e}^{-5(x2-0.1)^2}, & x2 > 1000\end{cases} \qquad (13)$$

Membership function of emitter distance: with the increase of the distance, the detection threat of the emitter to the fighter is reduced. For the missile, the navigation distance of the missile is increased, and the warning time of the fighter is prolonged, thus the attack threat to the fighter is reduced. At the same time, with the increase of distance, it increases the probability of the enemy's miss, and further reduces the threat.

$$U\left(x3\right)=\begin{cases}1, & x3 <= 30 \\ 0.5-0.5*\sin(\frac{\pi}{200-30})*(\frac{x3-(200+30)}{2}), & 30 < x3 <= 200 \\ 0, & x3 > 200\end{cases} \qquad (14)$$

Membership function of emitter azimuth: in the process of flight, the direction pointed by the aircraft head is the Direction with the greatest threat degree, and its threat angle has a certain threat degree in azimuth and high and low angles.

$$U\left(x4\right)=\begin{cases}1\ , & -15^0 <= x4 <= 15^0 \\ 0.8, & 170^0 <= x4 <= 190^0 \\ 0.5, & 80^0 <= x4 <= 100^0 \text{或} 260^0 <= x4 <= 280^0 \\ 0.3, & \text{other}\end{cases} \qquad (15)$$

$$W_\mathrm{i} = \sum_{i=1}^{n} C_i * \mathrm{U}(x_i) \qquad (16)$$

Then, formula 16 is used, where n represents the sequence number of the radiation source, CI is the weight of the radiation source threat factor, u (XI) represents the membership function of the radiation source factor, and W represents the threat degree of the radiation source. There are many methods for radiation source grade evaluation, such as AHP [12], fuzzy comprehensive evaluation [13] and fuzzy multi-attribute evaluation [14]. In this paper, the ancient forest method is used to estimate the weight of several threat factors, and then the threat degree is calculated by membership function. Here, the emitter level evaluation modeling and simulation is based on the position of the aircraft itself and the geographical location and frequency of the emitter transmitted by the parameters of the function.

IV. SIMULATION PLATFORM

The simulation tool is composed of a series of modules as follows:

1) Model assembly: the assembly function module of the model ensures that the assembly of the combat object model can be completed quickly and run on the simulation platform;

2) Simulation engine: it is a functional module of simulation engine that parses the XML file of the planned combat object, and drives the combat simulation according to the combat scenario; In the XML file, the initialization data of aircraft and radar as well as the coordinate points and flight speed of aircraft trajectory are stored. Users can change the state of combat objects in the battlefield by changing the data in XML, and then get different data.

3) Situation display: the battlefield situation display function module of the combat scene can realize the intuitive display of the battlefield situation.

4) Data recording: the function module of recording and playing back the simulation process data can record the simulation process data, which is convenient for post analysis and can be played back;

5) Console: the console includes data records. At the same time, after the user clicks the start, pause, continue and stop buttons on the console, MFC completes the operations that the server and client need to perform through the message mapping mechanism [11.15]. Each federate of the simulation system calls the corresponding function to process the data through the message type of the received data, so as to ensure the synchronization of the simulation system [16]. In this simulation system, distributed simulation is used for data transmission, which provides data transmission between simulation engine and model, simplifies the data interface of simulation model, and facilitates the implementation of distributed simulation.

In order to ensure that the simulation time step of the aircraft crew is consistent during the flight, the simulation step is set to 0.01 seconds. The simulator console defines a variable current time with an initial value of 0, which is used to record how many rounds of data the two aircraft have sent. The console defines a class to manage the data of each aircraft, and stores the data in the container. There is a member variable sending time to record the number of flight data sent by each aircraft, and the initial value is 0. When the two aircraft have completed the flight, the console will add one to the current time of the variable, and then check before sending the propulsion command. If the current time is greater than the sending time, it is allowed to send the propulsion command to the aircraft, and add one to the sending time of the corresponding member variable that manages the aircraft. Due to the network delay, there is a problem of how fast the packets of the two aircraft arrive at the console. The aircraft thread with fast packet arrival can sleep for a period of time and detect alternately until the packets of the other two aircraft arrive, and then proceed to the next simulation step. The promotion process of synchronization mechanism is shown in Figure 3.

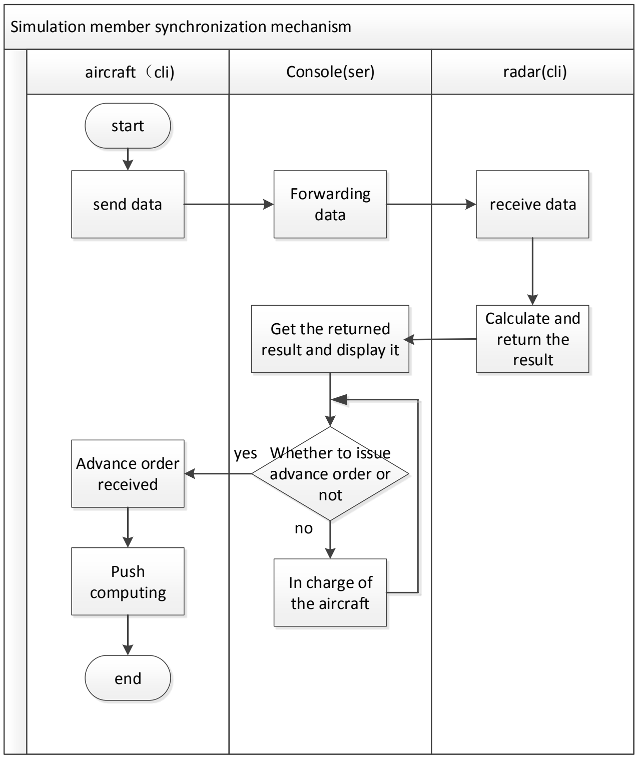

Simulation member synchronization mechanism
aircraft（cli)
Console(ser)
radar(cli)
start
send data
Forwarding data
receive data
Calculate and return the result
Get the returned result and display it
yes
Whether to issue advance order or not
Advance order received
no
Push computing
In charge of the aircraft
end

Figure 3. Flow chart of synchronous propulsion mechanism

The simulation platform uses XML technology to complete the initialization of simulation system parameters and aircraft scenario setting; The console is responsible for the start, pause, continue and stop of the simulation system, for the forwarding of relevant data between ground to air radars, and for the display and data collection of mutual detection results of ground to air radars during aircraft flight; Ground early warning radar and aircraft warning radar are responsible for target detection and warning.

V. THREAT LEVEL ASSESSMENT SYSTEM

According to the data obtained from the simulation platform, the radiation source threat level assessment system is established in the QT environment. When the system starts to run, firstly, the data source is simulated to read the file, and the data is read by line according to the time stamp, and the data at the same time is stored in the container. Secondly, the angle between the radiation source and the aircraft is obtained according to the data in the container through the coordinate transformation model, Then, according to the ancient forest method to determine the weight, and membership function to determine the threat coefficient, namely the emitter threat level evaluation model, and then sort, finally load these data into the interface to display. As shown in Figure 4:

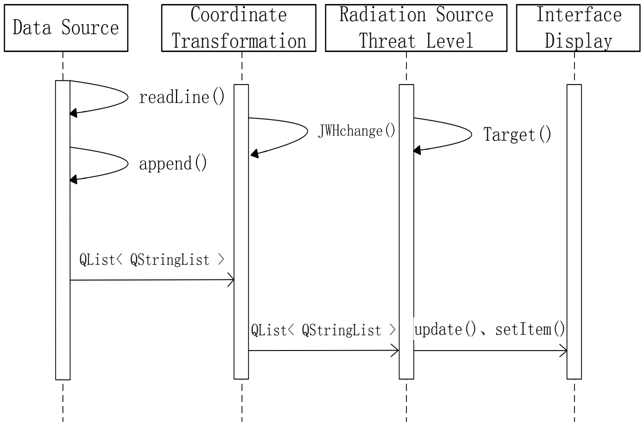
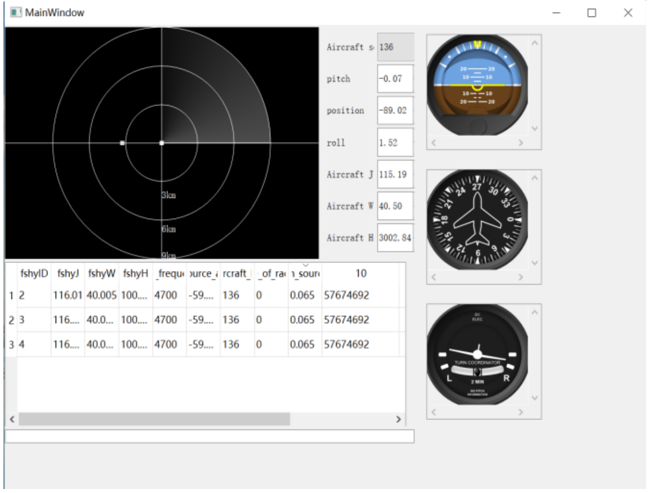


Figure 4. System sequence diagram



Aircraft s 136

pitch -0. 07

position -89. 02

roll 1. 52

Aircraft J 115. 19

Aircraft W 40. 50

Aircraft H 3002. 84

	fshyID	fshyJ	fshyW	fshyH	_frequ	ource_a	rcraft_	_of_ra	n_sour	10
1	2	116.01	40.005	100....	4700	-59....	136	0	0.065	57674692
2	3	116....	40.0...	100....	4700	-59....	136	0	0.065	57674692
3	4	116....	40.0...	100....	4700	-59....	136	0	0.065	57674692

Figure 5. Operation result diagram

The final result of emitter level evaluation based on distributed simulation is the ranking of the emitters detected by the aircraft at a certain time. The final rendering is shown in Figure 5.On the left side is the attitude angle data of the aircraft, and from top to bottom are the pitch angle, heading angle, roll angle, as well as the longitude, dimension and altitude of the aircraft; The scannin image takes the aircraf as the origin and marks the radiation source in the form of points in the two-dimensional coordinate system; The table below is the data of radiation sources, including the longitude, dimension, altitude, frequency, radar area density, aircraft ID, radar working status of radiation sources and the threat degree of radiation sources. The ranking of threat degree can provide effective reference for the formulation of aircraft penetration operation strategy and improve the survival probability of aircraft.

VI. CONCLUSION

In the modern electronic battlefield, it is impossible for fighter to penetrate without being found. When the enemy's air defense missiles and radars begin to fight, how the fighter to penetrate requires the fighter to have a real-time and overall grasp of the data of various radiation sources in the whole battlefield, and have an accurate grasp of the threat of radiation sources, so as to formulate a reasonable and comprehensive plan in the battlefield Some strategies against anti-aircraft such as the flight route that can be realized. Based on the research and implementation of emitter level evaluation in distributed simulation, this paper constructs a simulation system. By simulating the trajectory and state of the combat object in a typical battlefield, the data obtained by step is obtained by comprehensively considering the influence factors of time, space and electromagnetic environment in the battlefield, The order of the threat degree of the fighter's radiation source at a certain time in the battlefield is obtained, which provides a powerful reference for the fighter's next route or attack strategy.

REFERENCES

[1] Ma Hong, bu Wei. Electronic warfare simulation system for operational requirement verification. Electronic information countermeasure technology. 2019. (3). 40-43.

[2] Chen Yongguang, Shao Guopei, Zhang Shunjian. Research on combat effectiveness of air defense weapon in EW simulation system. Firepower and command and control. 1999. (4). 73-78.

[3] Kong Depei, Zhu Yifan, Yang Feng. Function simulation of radar detection. Computer simulation. 2003. (8). 119-122.

[4] Zhao Xingyun, Li Hua, Tang Xuefei. Computer simulation of radar reconnaissance equipment function simulation mathematical model. 2006. (4). 5

[5] Locker J. An introduction to the internet networking environment and SIMNET/DIS[R]. TechnicalReport, M onteney:NavalPostgraduates school, 1993.

[6] SikoraJ.Advanceddistributed(ADS)/distributedinteractivesimulation(DIS)[M]. Phalanx.1- 995.

[7] Zhu Guanlan, Han Yuanjie, Jiang fangting. Research on radiation source threat level assessment technology. Military communication. 2007.23 (262): 10-12.

[8] Sumanta Kumar Das, Modeling Intelligent Decision-Making Command and Control Ag- ents:An Application to Air Defense.Institute for Systems Studies and Analyses, Delhi, India. 2014.

[9] Mark Allen Weiss. Data Structures and Algorithm Analysis in C++[M].3rded. Upper Saddle River, New J ersey: Prentice Hall, 2006.

[10] Liu Bingyan, Liu Xiangwei, Hao Chengmin, et al. Research on threat level evaluation model of electronic air defense target. Ship electronic countermeasures. 2014. 37 (1): 57-61.

[11] Lu Wenzhou. Qt5 development and examples. Beijing: Electronic Industry Press, 2014.

[12] Zhao Xingchen, Wu Jun, Xu Yunshan, et al. Radiation source threat level assessment based on analytic hierarchy process. Modern defense technology. 2012.40 (5): 35-40.

[13] Chen Yawen, Xia Weijie, Wu Lianhui, et al. Assessment of radiation source threat level based on AHP fuzzy comprehensive evaluation method. Modern electronic technology. 2014.37 (19): 21-24.

[14] Jiang Ning, Hu Wei Li, sun Ao. Fuzzy multi attribute method for determining threat level of radiation source. Acta Ordnance Engineering Sinica. 2004.25 (1): 56-59.

[15] Huo Yafei. Practical solution of QT and QT quick development. Beijing: Beijing University of Aeronautics and Astronautics Press. 2012.

[16] Yang Wanhai. Radar system modeling and simulation [M]. Xi'an: Xi'an University of Electronic Science and technology. 2007.

Deep Learning in Product Manufacturing Record System

Wang Wenjing
School of Computer Science and Engineering
Xi'an Technological University
Xi'an, China
E-mail: 1015138082@qq.com

Zhao Li
School of Computer Science and Engineering
Xi'an Technological University
Xi'an, China
E-mail: zhaoli1998@163.com

***Abstract*—Deep learning based data analysis techniques are investigated in the context of product production record systems, using CNN, STACK LSTM, GRU, INCEPTION, ConvLSTM and CasualLSTM techniques to design network models and to study the processing of temporal data. Three network models are proposed for the problem of predicting the pass rate of upcoming product inspection records, namely CNN-STACK LSTM, INCEPTION-GRU and INCEPTION-Casual LSTM, and the structure of each network model follows the learning of local-global features. The experimental results show that the INCEPTION-GRU network model works best among the three models. Based on the prediction results, it is possible to correct in advance the operation of the shop technicians who do not regulate the debugging of the product, so that the initial production efficiency of the product can be improved.**

Keywords-Deep Learning; Local-Global; INCEPTION-GRU

I. INTRODUCTION

Since the 21st century, the new generation of industrial revolution, mainly characterized by intelligence, information technology and automation, namely Industrial Revolution 4.0, has emerged and opened the curtain of the "Industry 4.0" era [1]. With the improvement of product quality and safety in the modern industrial production process, increasing amounts of attention is being paid to the quality control of the whole process of product production. In order to improve product quality and production efficiency, enterprises trace the quality of their products, pay attention to the use of products, and improve and control the existing product quality [2]. In recent years, with the development of manufacturing technology and increased supervision, the probability of accidents in industrial products is also decreasing year by year, and the overall safety level is gradually increasing [3-4].

The product production record system records production record data from processing to packaging, provides comprehensive analysis of product production data, and establishes a production record system that focuses on the production process. In the intelligent manufacturing mode, the operator uses the code scanning gun to enter the relevant data of the product. It greatly improves the efficiency of production data collection and facilitates the improvement of product production efficiency. Deep learning has taken off in recent years, making major breakthroughs not only in medical research, energy consumption in life, finance and communications. Features such as face recognition, speech processing and video object detection are derived based on algorithms and extensive training in deep learning. The learned

DOI: 10.21307/ijanmc-2021-028

features extracted through different deep networks perform well in prediction. Therefore, learning deep features for prediction is becoming Getting more and more popular. In this paper, inspection records in product manufacturing system are used as training samples, CNN, STACK LSTM, GRU, INCEPTION, ConvLSTM and CasualLSTM techniques are used to design network models and to study the processing of temporal data, after comparing and selecting a comprehensive and high network model to predict the results of the product to be inspected. According to the prediction results, the model can correct the operation of the workshop technicians who do not regulate the debugging of products in advance, and largely improve the efficiency of product testing at the early stage of production.

II. RELATED WORK

A wide variety of time series data exists for various industries, financial time series, electricity consumption time series, coal price time series. It is of great value and significance to spend effort on time series data to study and analyze them. Lots of work has been done by domestic and foreign scholars on time series data analysis and forecasting.

RNN is a traditional recurrent neural network [5], which is a model for processing sequential data. The traditional RNN in solving the association between long sequences, through practice, proved that the classical RNN performs poorly, the reason is that when backpropagation is performed, too long sequences lead to abnormal computation of gradients, and gradient disappearance or explosion occurs. LSTM [6] long short-term memory model is a special kind of RNN neural network and effectively deal with long-term dependence problem, it is compared with RNN in two aspects to do improvement The LSTM gating mechanism has three main gates which are forgetting gate, input gate and output gate respectively. The equation of LSTM network structure at moment t is shown below, f_t, i_t, o_t, C_t are forgetting gate, input gate, output gate and cell state respectively, W_t is each gating weight parameter, b_t is each gating bias parameter respectively, σ are sigmoid activation function, and tanh is the hyperbolic tangent activation function.

$$\begin{cases} f_t = \sigma(W_f[h_{t-1}, x_t] + b_f) \\ i_t = \sigma(W_i[h_{t-1}, x_t] + b_i) \\ \tilde{C}_t = \tanh(W_c[h_{t-1}, x_t] + b_c) \\ o_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \\ h_t = o_t \cdot \tanh(C_t) \\ C_t = f_t \cdot C_{t-1} + i_t \cdot \tilde{C} \end{cases} \quad (1)$$

LSTM has the dominant advantage for the prediction of temporal data, but it is not effective in predicting data in spatio-temporal sequences, does not consider spatial correlation and carries redundancy, and cannot portray local features due to the strong local characteristics of spatial data. The ConvLSTM network structure was first proposed in 2015 for precipitation proximity prediction [7], and the experimental results found that ConvLSTM grasps the data spatio-temporal structure, and also proved that ConvLSTM works better than LSTM in obtaining spatio-temporal relationships. The core of ConvLSTM network structure is consistent with LSTM, which also takes the output information of the previous layer as the input information of the next layer. The difference is that the W-weight full link operation is changed to convolution operation, which not only can get the spatio-temporal relationships, but also can extract spatial features like convolution layer thus being able to obtain spatio-temporal features.

GRU [8] is a variant of LSTM network, it inherits the advantages of LSTM network and has a simple structure, LSTM has three gates are forget gate, input gate and output gate, while GRU has only two gates are update gate and reset gate, for memory information transfer, LSTM is passed to the next unit through the output gate, GRU is directly transferred to the next unit without control. t moment GRU unit structure of the equation is shown below, r_t, z_t, $\tilde{h}_t$are reset gate, update gate and memory transfer information respectively.

$$\begin{cases} r_t = \sigma(W_r \cdot [h_{t-1}, x_t]) \\ z_t = \sigma(W_z \cdot [h_{t-1}, x_t]) \\ \tilde{h}_t = \tanh(W \cdot [r_t * h_{t-1}, x_t]) \\ h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t \end{cases} \qquad (2)$$

III. NETWORK MODEL CONSTRUCTION

A. *CNN-STACK LSTM detection record prediction model construction*

1) *Local feature learning*

The convolution layer (Conv) is a structure unique to convolutional neural networks and is used to extract local features of the data, the outline of people on pictures, the shape of cars, etc. Due to the specificity of the dataset, the convolution kernel performs the convolution operation on one-dimensional data according to the left-to-right direction, and the convolution operation is shown in Figure 1.

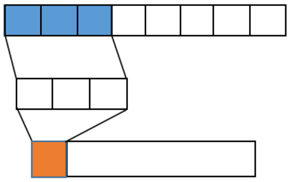


Figure 1. Convolution operation

A BN layer (Batch Normalization) is added after the convolution layer, which sets the increasingly deviated distribution into a normalized distribution by means of normalization, BN layer can make the loss function smoother as well as facilitate gradient descent. to avoid the appearance of "Dead Neuron". The final local feature learning module is (CBRP) shown in Figure 2, which is composed of a convolutional layer (Conv), a BN layer (Batch Normalization), an activation function (Leaky ReLU) and a pooling layer (Pooling).

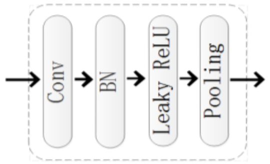


Figure 2. CBRP module structure

2) *Global feature learning*

LSTM has made great breakthroughs in speech recognition, data prediction and financial records, LSTM solves the drawbacks of RNN. Global feature learning is performed using a stacked long and short-term memory model (STACK LSTM), which is a multilayer LSTM network model stacked vertically, thus enhancing the abstraction capability of the network. Article adopt a stacked LSTM network model, as shown in Figure 3, where the first layer is used as the input LSTM network and the output information of the first layer is used as the input information of the second layer.

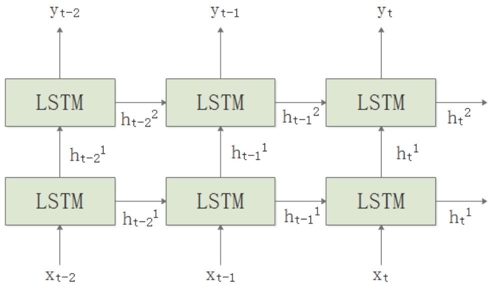


Figure 3. Stacked LSTM structure

3) *CNN-STACK LSTM Network Model*

The CNN-STACK LSTM network model is constructed by three CRPs, a fully connected layer, a STACK LSTM layer and a Softmax layer, where the output results are normalized using the Softmax layer. The overall architecture of the CNN-STACK LSTM network model is shown in Figure 4.

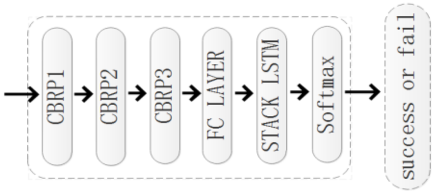


Figure 4. CNN-LSTM network model

B. INCEPTION-GRU detection record prediction model construction

1) Local feature learning

In 2012 AlexNet [10] first used new techniques such as Relu, Dropout in CNN and made a historical breakthrough, the mainstream breakthroughs in network structure are mainly divided into two categories, one is to increase the depth of the network model (number of layers) and the other is to expand the width of the network model (number of neurons). As the number and width of the network model increase, it will bring many negative factors, such as overfitting, gradient disappearance and gradient explosion. Until the emergence of GoogLeNet in 2014, GoogLeNet [11] is composed by inception module, inception is proposed to improve the training results from another perspective, which can use the CPU/GPU computational resources more efficiently and can extract more features with the same computational and weighting parameters, thus improving the training The result.

According to the training data characteristics, the inception basic unit is modified to reduce the convolution of one branch, and the convolution followed by the activation function is modified to add BN (Batch Normalization) layer calculation between the convolution and activation function, and the activation function uses Leaky ReLU, the basic structure of this inception is shown in Figure 5

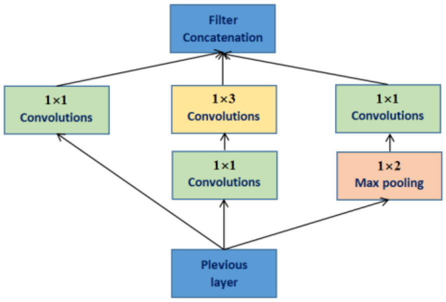


Figure 5. Inception module structure

The use of the inception structure for local feature learning and the expansion and deepening of the local feature network model also increases the local feature network model nonlinearity as well as the fusion of local features of different sizes, both to improve accuracy.

2) INCEPTION-GRU network model

The INCEPTION-GRU network model is constructed by a convolutional layer, two INCEPTIONs, a fully connected layer, a GRU layer and a softmax layer, and the overall architecture of the INCEPTION-GRU network model is shown in Figure 6.

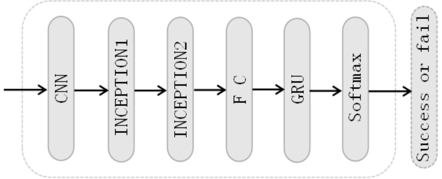


Figure 6. INCEPTION-GRU network model

C. INCEPTION-Casual LSTM Detection Record Prediction Model Construction

1) Global feature learning

Deeper network models can improve nonlinear representation and also learn more complex transformations, which can fit more complex feature inputs [12]. The network layers in the network model each have their own role, and the features learned by each network layer are observed from a classical network ZFNET reverse convolution. The first layer extracts the edges, the second layer extracts the simple shapes, the third layer has extracted the shapes of the targets, and the deeper network layers are able to learn more complex features [13]. If there is only one layer, it means that the transformations to be learned are very complex.

The model uses Casual LSTM as global feature learning. Casual LSTM is a cascade operation of ConvLSTM cells. Casual LSTM adds more nonlinear operations so that the features will be amplified, which is beneficial to capture short-term dynamic changes and emergent situations. The design Casual LSTM is composed of four layers of ConvLSTM units, similar to the four-layer stacked LSTM structure, where the LSTM base unit is replaced with a ConvLSTM.

2) INCEPTION-Casual LSTM Network Model

The INCEPTION-Casual LSTM network model is constructed by a convolutional layer, two INCEPTION, a Casual LSTM layer and a softmax layer. The overall architecture of the network model is shown in Figure 7.

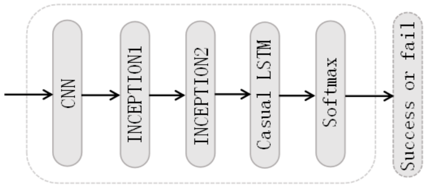


Figure 7. INCEPTION-GRU network model

IV. EXPEEIMENTS

A. Data set structure and processing

The production records of the product are derived from the system's data from the relay shop from November 2018 to April 2019. The data of the production records amount to 260,000 records, of which the inspection records occupy 149,800 records. The data structure of the inspection records is detailed in Table 1.

TABLE I. DATA STRUCTURE OF TEST RECORDS

Field	Annotation	Data Type	Is The Pk
id	*Self-adding id*	*Int (32)*	*yes*
txm	*barcode*	*varchar (40)*	*no*
probbh_id	*Product version id*	*Int (32)*	*no*
cpjcjd_id	*Product process id*	*Int (32)*	*no*
lx	*types*	*Int (2)*	*no*
xh	*Work serial number*	*Int (2)*	*no*
czry	*debuggers*	*varchar (20)*	*no*
jcry	*Testers*	*varchar (20)*	*no*
wlh	*Material number*	*varchar (20)*	*no*
wlms	*Material Description*	*varchar (200)*	*no*
dd	*Order Number*	*varchar (30)*	*no*
bbh	*version number*	*varchar (50)*	*no*
mc	*Name of work process*	*varchar (30)*	*no*
ms	*description*	*varchar (60)*	*no*
create_date	*Creation time*	*date*	*no*
create_username	*Create User*	*varchar (60)*	*no*

Since the data trained by deep learning is the detection record, the fields "primary key identity ", "product process identity ", "product version identity ", " barcode", "order number", "creation time" and "creation user" are not considered, and the inspection record has corresponding redundant fields. The material description is an explanation of the material number, so it is also removed, and the process number, debugger, inspector, material number, version number, process name, and description fields are used as training features.

Since the feature values of the training data are limited, a data mapping approach is adopted to process the string type features in the detection records. First, all the different string values in each feature are counted, and these values are assigned according to the Arabic numerals from smallest to largest, for the feature "description", normal is 0, and fault is 1. The data in Table 2 needs further feature normalization, and the processed data becomes the data with a mean value of 0 and standard deviation of 1. The data is processed in order to allow the model to learn quickly and iteratively optimize to improve the training efficiency.

TABLE II. RESULTS OF DATA PROCESSING OF TEST RECORDS

Xh	Czry	Jcry	Wlh	Bbh	Mc	Ms
5	*0*	*0*	*0*	*0*	*0*	*1*
3	*1*	*1*	*1*	*0*	*1*	*0*
3	*2*	*1*	*0*	*0*	*1*	*0*
5	*3*	*2*	*2*	*0*	*0*	*0*
5	*4*	*0*	*0*	*0*	*0*	*1*
3	*3*	*1*	*1*	*0*	*1*	*0*
3	*5*	*1*	*3*	*0*	*1*	*0*
3	*5*	*1*	*3*	*0*	*1*	*0*
5	*6*	*3*	*3*	*0*	*0*	*0*
5	*5*	*4*	*2*	*0*	*0*	*0*
3	*5*	*1*	*3*	*0*	*1*	*0*
3	*5*	*1*	*3*	*0*	*1*	*0*
5	*7*	*0*	*1*	*0*	*0*	*0*
3	*2*	*1*	*0*	*0*	*1*	*0*
5	*8*	*5*	*2*	*0*	*0*	*1*
3	*6*	*1*	*1*	*0*	*1*	*0*
3	*5*	*1*	*3*	*0*	*1*	*0*

B. Model training

This article uses the Tensorflow framework to build a network model, accompanied by the Adam [14-15] optimization algorithm, which is an optimization algorithm for finding global optima, introducing a quadratic gradient correction. The learning rate is also continuously corrected during training in addition to the correction of the weight parameters and bias parameters using the back propagation algorithm. 140,000 data from the product inspection records were used as training data, 10,000 data were used as test data, and each of the three network models was trained by iterating 500 times.

C. Experimental results

The test data were verified in each of the three network models to verify the performance and prediction effectiveness of the network models. After three prediction experiments with different network models, Table 3 shows the content of the experimental results. The INCEPTION-GRU network model was compared with the other two models in terms of three aspects: time, stability and prediction effect, with short running time, high stability and accurate prediction effect.

TABLE III. COMPARISON OF NETWORK MODEL ACCURACY (%)

Network Model	**1**	**2**	**3**
CNN-STACK LSTM	***92.33***	***92.50***	***92.78***
INCEPTION-GRU	***92.6***	***93.50***	***93.28***
INCEPTION-Casual LSTM	***83.83***	***79.67***	***94.25***

V. SUMMARY AND PROSPECT

In conclusion, this article propose a research method based on deep learning for product inspection record prediction and design three network models to predict the dataset, each model will have different advantages between both the results and the process, and INCEPTION-GRU network model is considered comprehensively from all aspects as a product inspection record prediction model to provide technical support for new product detection. After studying the three network models, it is found that ConvLSTM is not as stable and efficient as LSTM and GRU in processing temporal data. Future work will focus on improving the GRU model to further improve the accuracy of product inspection pass rate.

REFERENCES

[1] Li Wenbin. Research on the essence of industrial revolution 4.0 and its impact [D]. China University of Mining and Technology, 2019.

[2] Zhu Chun Chuan. The use of product traceability system in automated production [J]. Metallurgical Management, 2020(17):75-76.

[3] Dong Yanchao. Research on key technologies of RFID-based industrial product traceability system [D]. Dalian Maritime University, 2017.

[4] Xie W, Xie K.Hang A,Wan G,Qu I,Zhang Q,Tang C J.RFID reseraching:Finding a lot tag rather than only detecting its missing[J]. Journal of Network and Computer Applications, 2014, Volume 41:95-120.

[5] Cheng Zhaolan, Zhang Xiaoqiang, and Liang Yue. Railway freight volume forecasting based on LSTM networks [J]. Journal of Railways, 2020, v.42; No.277(11):19-25.

[6] Wei X, Zhang L , Yang H Q , et al. Machine learning for pore-water pressure time-series prediction: application of recurrent neural networks [J]. Geoscience Frontiers, 2020.

[7] Xingjian Shi, Zhourong Chen, Hao Wang, et al. Convolutional LSTM Network: a Machine Learning Approach for Precipitation Nowcasting [J]. NIPS 2015.

[8] Hu H, Yang Y.A combined GLQP,and DBN-DRF for face recognition in unconstrained environments[C]. 2nd International Conference on Control, Automation and Artificial Intelligence (CAAI 2017), 2017.

[9] LIN M, CHEN Q, YAN S. Network in network [EB/OL]. [2013-12-16]. https: //arxiv. org/abs/1312. 4400.

[10] Goodfellow, I., Bengio, Y., Courville, A.. Deep learning (Vol.1). Cambridge: MIT press, 2016:326-366.

[11] Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., Liu, T., Wang, X., Wang, L., Wang, G. and Cai [J]. 2015. Recent advances in convolutional neural networks. arXiv preprint arXiv:1512.07108.

[12] Raghu M, Poole B, Kleinberg J, et al. On the expressive power of deep neural networks [C]//Proceedings of the 34th International Conference on Machine Learning-Volume 70. jmlr. org, 2017: 2847-2854.

[13] Bianchini M, Scarselli F. On the complexity of neural network classifiers: a comparison between shallow and deep architectures [J]. IEEE transactions on neural networks and learning systems, 2014, 25(8): 1553-1565.

[14] Yang Guanci, Yang Jing, Li Shaobo, et al. Modified CNN algorithm based on Dropout and ADAM optimizer [J]. Journal of Huazhong University of Science and Technology (Natural Science Edition), 2018, 46(7): 122-127.

[15] Chang Zihan. Electricity Price Prediction Based on Hybrid Model of Adam optimized LSTM Neural Network and Wavelet Transform [D]. Lanzhou: Lanzhou University, 2019.

One Way Travel Restriction Device

Zihang Hu
College of mechanical and electrical engineering
Shaanxi University of science and technology
Xi'an, China
E-mail: 1614352048@qq.com

Yuan Mi
College of mechanical and electrical engineering
Shaanxi University of science and technology
Xi'an, China
E-mail: 2576607180@qq.com

Guolong Liu
College of mechanical and electrical engineering
Shaanxi University of science and technology
Xi'an, China
E-mail: 1632040985@qq.com

***Abstract*—A one-way travel restriction device is proposed. This device can be used on the road to limit the one-way traffic of vehicles, avoid retrograde, private car parking space will be occupied sometimes, and the unsafe factors such as occupying the road at the construction site or road in dangerous state. At present, the one-way traffic restriction on the road mainly depends on the use of traffic police, which is not only inefficient, but also wastes a lot of human resources. One way travel restriction device can solve the above problems. The one-way travel restriction device has complete functions and works efficiently, quickly and conveniently, which conforms to the requirements of the times. So how to meet the needs of modern people in the development of intelligence, mechanical and electronic information integration of the control operation structure provides the possibility of practicality, has a very large development value and broad market prospects.**

Keywords-One Way; Restricted; Device; Traffic

I. BACKGROUND OF THE IDEA

In today's society, with the gradual improvement of material living standards, cars have been popularized. According to statistics, as of May 3, 2020, the number of motor vehicles in Xi'an has exceeded 3 million, even 764 circles around the city wall, ranking among the top ten in China. In recent years, although the traffic road has been improving in a good direction, road construction is also being implemented, but the speed is far less than the growth rate of motor vehicles. Traffic safety is also a very important issue. Because of traffic jams, small electric vehicles and motorcycles are often retrograde on the street, which is very dangerous. However, the retrograde of small vehicles is not well contained. At this stage, a large number of traffic police are required to intercept the traffic on the roadside, which is undoubtedly very troublesome, Therefore, we hope to design a one-way travel restriction device.

II. CONTENTS AND KEY PROBLEMS

A. Function description of the project

The device provides a one-way traffic restriction device for vehicles, which can carry out traffic restriction management. It solves the problem that sometimes private cars will be occupied, and when the construction site occupies the road or the road is in a dangerous state, the danger can be avoided through the one-way traffic restriction device of vehicles.

B. Current status of the project

1) Domestic situation

Domestic common warning signs, road signs, road condition reminders, etc. It is used to park cars in the garage or parking space, and in case of

DOI: 10.21307/ijanmc-2021-029

emergency. The one-way travel restriction device has complete functions and works efficiently, quickly and conveniently, which conforms to the requirements of the times. So how to meet the needs of modern people in the development of intelligence, mechanical and electronic information integration of the control operation structure provides the possibility of practicality, has a very large development value and broad market prospects.

2) Current situation abroad

Nowadays, many foreign countries adopt one-way traffic, but in China, because of the large number of vehicles and people, two-way traffic is adopted.

Foreign single traffic can be divided into three types. The first is Manhattan style: long distance, wide range regional one-way traffic mode. The second is the London model, which is mainly based on the one-way road within the plot. The third mode is Singapore mode: the combination of trunk road and branch road.

C. The key problems to be solved in realizing the function of the project

Key problem 1: there will be vehicle retrograde phenomenon: I think the project can play a good role in one-way traffic restriction. We think that solving the retrograde problem has always been a problem. This device can change the retrograde problem from telling people "can't do" to "can't do". The problem of mandatory change.

Key problem 2: unidirectional device can also be used in other aspects. Such as construction restrictions, parking lot entrance, crossroads and other places. It can solve the problem of hidden danger caused by nonstandard driving to a great extent.

III. SPECIFIC SCHEME OF PROJECT IMPLEMENTATION

A. Preliminary planning

At first, we put forward three plans

One is through the transformation of parallelogram mechanism. When the bevel of a parallelogram is tilted, the height of its upper and lower sides will decrease. When the parallelogram is vertical, the height of the upper and lower sides is the maximum. Based on this, when the left and right sides of the quadrilateral tilt, the height can not make the vehicle pass, but the vertical height can pass. According to this principle, the model is put forward. The change of controlling parallelogram mechanism is through the thrust force of bar pair and mechanism. Using the lever principle, when there is no vehicle passing on one side, the parallelogram mechanism is in a non rectangular state because there is no force on one side of the lever, and the height is not enough to make the vehicle pass. When a vehicle passes by, one end of the lever is pressed down and the other end is lifted up. The action of the connecting rod receiving the force and the specific trajectory of the given plane makes the parallelogram receive the right force, the parallelogram deforms into a rectangular state, the height becomes higher, and the vehicle can pass through. If you go retrograde, you can't get through because there is no lever. This scheme can realize the original expectation and can be made into a single line device. However, compared with other schemes, this scheme has the following problems: compared with other schemes, the mechanism is too complex and inefficient, which may restrict some higher vehicles from passing through, and there are too many restrictions, so it is not adopted.

The second scheme: when the vehicle passes through the unidirectional driving device from the reverse direction, the pressure block above the device is pressed down by the wheel, the spring is compressed due to the gravity action of the vehicle, the rod connected with the pressure block is lowered, and the middle of the rod is connected with the frame through the low pair hinge. Therefore, the other end of the rod will be lifted, and the baffle connected with the rod through the hinge will also be lifted, So as to restrict the passing of vehicles. The other end is provided with the same rod assembly. When the vehicle is passing in the positive direction, the lifting of the rod will make the baffle lower, so the vehicle can pass through.

The third scheme: a varistor is set on one side of the road to connect with the gear lever. When the car is passing in the positive direction, the resistance of the varistor decreases due to the increase of pressure, and the gear lever rises, and the car passes. The reverse vehicle cannot lift the lever through the varistor.

B. Determination of final scheme

After discussion, we think that the structure of scheme 1 is more complex, and there are great limitations for the height of the vehicle. However, there is no specific implementation agency to explain his plan. Finally, it was decided to use scheme 2.

IV. SCHEME DESCRIPTION

The device provides a one-way traffic restriction device for vehicles, which can carry out traffic restriction management. It solves the problem that sometimes private cars will be occupied, and when the construction site occupies the road or the road is in a dangerous state, the danger can be avoided through the one-way traffic restriction device of vehicles.

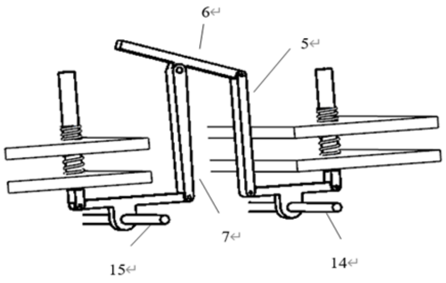


Figure 1. Schematic diagram of the device

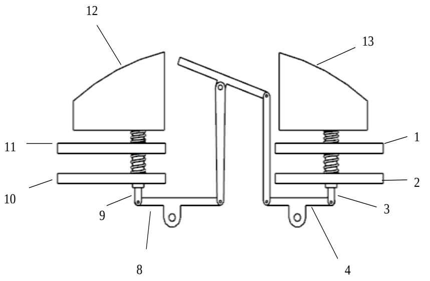


Figure 2. Schematic diagram of the device (main view)

As shown in Figure 1 and Figure 2, the following conclusions can be obtained:the product comprises upward convex pressing blocks 12 and 13, which are respectively connected with the support plate 1 and the support plate 11 through springs. The support plates 1, 2, 10 and 11 are rigidly connected with the shell, and are fixed in the axial direction and transverse direction. The connection among the rods 3, 4, 5, 6, 7, 8 and 9 is hinge connection. Among them, shaft 14, shaft 15 and shell are rigidly connected. When the vehicle enters the device from the right side, the gravity of the vehicle causes the pressure block 13 to move down, and the shaft 3 to move down makes the rod 4 rotate clockwise around the shaft 14, so that the rod 5 moves up, the rod 7 moves down, and the rod 6 rotates counterclockwise, and the vehicle can pass through. When the vehicle enters the device from the left side, the gravity of the vehicle causes the pressure block 12 to move down, the rod 9 to move down causes the rod 8 to rotate anticlockwise, thus the rod 7 to move up, and the rod 6 to rotate clockwise, so the vehicle cannot pass.

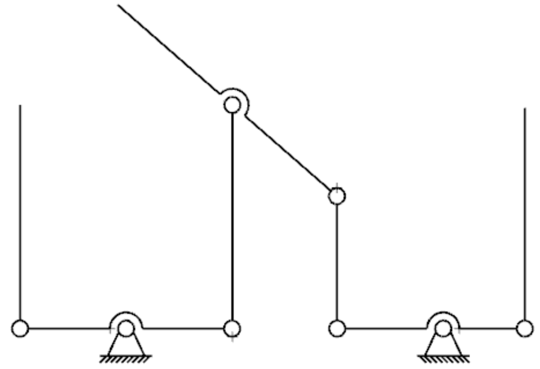


Figure 3. Structure diagram of the device

According to Figure 3, we can know that is the structure diagram of the device. It can be seen from the structure diagram that the main principle of the device is the lever principle. Lever principle is also called "lever balance condition". In order to balance the lever, the two moments (the product of force and arm) acting on the lever must be equal. That is: power × Power arm = resistance × The resistance arm is expressed as F1 ·L1 = F2 ·L2.

Where F1 is power, L1 is power arm, F2 is resistance and L2 is resistance arm.

$$F_1 \times L_1 = F_2 \times L_2 \quad (1)$$

Through (1), we can get the following conclusion: when the right side of the bar moves downward under pressure, the left side of the bar will lift up to form a small slope, which can make the vehicle pass. But the gravity of the car can't act on the left bar directly, so it can't pass from the left. It can be seen from the above formula that in order to balance the lever, the power arm is several times of the resistance arm, and the power is a fraction of the resistance.

When the vehicle passes through the unidirectional driving device from the reverse direction, the pressure block above the device is pressed down by the wheel, the spring is compressed due to the gravity action of the vehicle, the rod connected with the pressure block is lowered, and the middle of the rod is connected with the frame through the low pair hinge. Therefore, the other end of the rod will be lifted, and the baffle connected with the rod through the hinge will also be lifted, So as to restrict the passing of vehicles. The other end is provided with the same rod assembly. When the vehicle is passing in the positive direction, the lifting of the rod will make the baffle lower, so the vehicle can pass through.

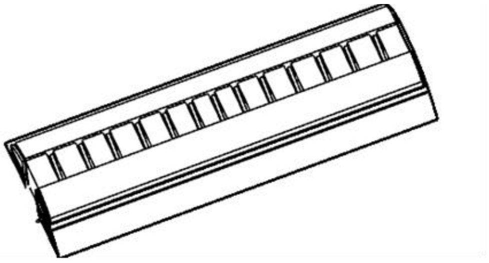


Figure 4. Appearance of one way travel restriction device

As shown in Figure 4 is the appearance of one way travel restriction device. We designed the appearance of the device like this, referring to the appearance of the speed bump on the highway. This device can be more convenient to install on the road, at the same time, the device can also play the role of deceleration belt, can replace the deceleration belt, to achieve double effect.

V. MARKET RESEARCH AND COMPETITION ANALYSIS

A. Market Research and analysis

Domestic common warning signs, road signs, road condition reminders, etc. It is used to park cars in the garage or parking space, and in case of emergency. The one-way travel restriction device has complete functions and works efficiently, quickly and conveniently, which conforms to the requirements of the times. So how to meet the needs of modern people in the development of intelligence, mechanical and electronic information integration of the control operation structure provides the possibility of practicality, has a very large development value and broad market prospects.

B. Competition analysis

PEST analysis is the analysis of macro environment. Macro environment, also known as general environment, refers to the main social forces that bring opportunities or threats to enterprises. They directly or indirectly affect the strategic management of enterprises. An important tool to analyze the macro environment is PEST analysis model, which analyzes the impact of environmental changes on enterprises from the political, economic, social and technological perspectives.

Politics: in 2006, the outline of the national medium and long term science and technology development plan (2006-2020) was published, and the 18th National Congress of the Communist Party of China made the strategic deployment of innovation driven development. In 2015, the opinions of the CPC Central Committee and the State Council on deepening the reform of system and mechanism and accelerating the implementation of innovation driven development strategy were published, In the same year, the implementation plan for deepening the reform of science and technology system was also issued.

Economy: closely linked to development. We should adhere to problem orientation, face the forefront of world science and technology, face the major national needs, and face the main battlefield of the national economy, clarify the main direction of China's innovation and development, make breakthroughs in key areas as soon as possible, and strive to form more competitive advantages.

Deepen reform. We should keep pace with the reform of the science and technology system and the reform of the economic and social fields, strengthen the connection between science and technology and the economy, follow the laws of the socialist market economy and science and technology innovation, get rid of all ideological barriers and institutional barriers that restrict innovation, and build a good environment that supports innovation driven development.

Strengthen incentives. The essence of innovation driven is talent driven. We should put people first, respect the value of innovation and creation, stimulate the enthusiasm and creativity of all kinds of talents, and speed up the gathering of a large-scale, reasonable structure and high-quality innovative talent team.

Open wider to the outside world. We should persist in planning and promoting innovation from a global perspective, make the best use of global innovation resources, comprehensively enhance China's position in the global innovation pattern, and strive to become a leader in several important fields and a participant in important rule making.

Society: develop smart city and digital social technology, and promote the people-oriented new urbanization. Rely on new technology and management innovation to support new urbanization, modern urban development and public services, innovate social governance methods and means, accelerate the informatization process of comprehensive governance of social security, and promote the construction of a safe China. Develop standardized, digital and intelligent technologies for municipal infrastructure such as transportation, electric power, communication and underground pipe network, and promote the large-scale application of key technologies in green building, smart city, ecological city and other fields. We will strengthen key technologies and products in major disaster and public security emergency areas.

Technology: China's innovation driven development has the foundation to accelerate. After years of efforts, the development of science and technology is entering a leap period from quantitative growth to qualitative improvement, the scientific research system is increasingly complete, the talent team is growing, and the independent innovation ability of science, technology, engineering and industry is rapidly improving. Economic transformation and upgrading, continuous improvement of people's livelihood and national defense modernization put forward a huge demand for innovation. The combination of huge market scale, complete industrial system, diversified consumer demand and the improvement of innovation efficiency in the Internet era provides a broad space for innovation. The system of socialism with Chinese characteristics can effectively combine the advantages of concentrating on major affairs and market allocation of resources, which provides a fundamental guarantee for innovation driven development.

VI. CONCLUSIONS

The device provides a one-way traffic restriction device for vehicles, which can carry out traffic restriction management. It solves the problem that sometimes private cars will be occupied, and when the construction site occupies the road or the road is in a dangerous state, the danger can be avoided through the one-way traffic restriction device of vehicles. The utility model relates to a one-way limiting device used for private parking space management, which can control the movement of the support plate through a simple structural design, drive the rod to move up and down, and then control the rotation of the panel, so as to achieve the purpose of one-way limiting. The one-way limiting device of the utility model can be used for the use and management of private parking space, Prevent private cars from going in and out in opposite directions, causing congestion.

REFERENCES

[1] Tang Dayong, graduate thesis of Chang'an University, One way traffic scheme design and application of TransCAD software.

[2] An Tongda,Zhang Mengmeng,Lu Yanwen, School of Aeronautical Engineering, Civil Aviation University of China. Design of directional traffic limiter for one-way traffic road.

[3] Zhen Jiahong,Zhu Jianru,Liu Xiping, China Machine Press,Theory Of Machines And Mechanisms.

[4] Luo Yinzhe,Yang Qi,Shaanxi University of Technology, One way traffic restriction system of intelligent road based on Internet of things.

[5] Pang Dengyu,Southwest Jiaotong University, Performance analysis of urban road network under traffic restriction.

[6] Qimin, Bayannaur vocational and technical school, Application of mechanics in life.

[7] Wen Bangchun, China Machine Press, Mechanical Design Manual.

[8] Wang Ningxia, China Machine Press, machine design.

Design and Analysis of a Robotic Arm for a Commercial Flight Simulator

Haifa El-Sadi
Mechanical Engineering
Wentworth Institute of Technology
Boston, USA
E-mail: elsadih@wit.edu

John Connolly
Mechanical Engineering
Wentworth Institute of Technology
Boston, USA
E-mail: connollyj@wit.edu

***Abstract*—This study is an elementary design and analysis of a novel flight simulator design. The simulator is positioned at the end of a large 5-axis robotic arm. The robot was designed in three configurations for different implementations, the principal application being a Boeing 747 simulator. The main purpose of this design is to give pilots an accurate experience with a wide range of motion up to 2g of acceleration.**

This paper describes the design objectives and the methodology to accomplish the goals. It includes preliminary designs and detailed CAD models. To validate the safety of the design, stress analysis was conducted under gravity loading and maximum dynamic loading.

Keywords-Simulator; CAD; Stress; Robot

I. INTRODUCTION

In the 1950s, companies began building virtual cockpits that simulated the experience of flight for pilots. Flight simulators would replicate each aspect of flying the actual aircraft for the pilots: controls, visuals, and motion. This tool allows for excellent, accurate training without any risk or cost of flying actual aircraft. Changes in the design of software algorithms for generating physical motion in flight simulators have typically been put forward on the grounds of improved motion cueing. Meyer et al. studied the pilot evaluations of algorithms implemented on a six degree of freedom flight simulator simulation a large transport aircraft during low altitude flight [1].

Previous studies attempted to quantify the perceptions of airline pilots about the quality of motion possible when a number of different motion-drive algorithms which were tested on a simulator employing a state-of-art six degrees of freedom motion-base [2]. Eric et al. described a new approach to relate simulator sickness ratings with the main frequency component of the simulator motion mismatch, that is, the computed difference between the time histories of simulator motion and vehicle motion [3]. The simulator motion cueing problem has been considered extensively, some studies showed that a cueing algorithm, that can make better use of the platform workspace whilst ensuring that its bounds are never exceeded [4].

There are simulators with a duplicate cockpit of the Boeing 777 on the top. A trainee sits inside, and screens give the pilot a virtual view of the world. The simulator sits a top six hydraulic legs. To simulate motion during flight, these hydraulic cylinders are finely controlled by electric pumps.

DOI: 10.21307/ijanmc-2021-030

For example, extending the front legs would tilt the simulator backward. The pilots would experience being pulled into their seats. This type of motion could simulate linear acceleration of the aircraft. Conversely, if the back legs were raised and the front lowered, the pilots would be pulled forward. This would simulate braking. On the other hand, a hexapod system allows for six degrees of freedom: linear motion in all axes (x, y, z), as well as rotational motion (roll, pitch, yaw). However, the system is extremely limited in motion. This means simulated movements are short duration and mostly limited to 1G. For example, the acceleration a pilot may feel on the runway in the x-direction might typically be 0.3G. When added to the force of gravity, a pilot will feel a resultant force over a G. These simulators will only be able to produce 1G for the pilot. Although this hydraulic system retains a small footprint, a wider range of motion would allow for more accurately representation of motion.

The objective of this project is to accurately simulate the motion of a Boeing 747 using a robotic arm. The motion of a 747 can be described in five axes of motion: 2 translational, 3 rotational. Thus, a flight simulator built upon a robotic arm requires five degrees of freedom. The system must replicate the g-forces experienced by a pilot during flight of a 747. This paper will detail the design process for this system. After developing a motion profile, the virtual cockpit's weight will be used to develop the system's dynamic and broad electrical requirements. An iterative process of modeling and stress analysis will be used to design the arm. Finally, a motion analysis will validate the system provides the required motion.

II. Robot Arm Configuration

Each of the five degrees of motion (two linear, three rotational) must be simulated with the robot arm (as outlined above, IV. 747 Motion Profile). The robot's configuration is a tandem application of a traditional robot arm and a gyroscopic wrist. The pilot will only experience two translational accelerations at any given instant; therefore, the robot arm only needs to supply two degrees of translational motion. This significantly simplifies the configuration of the arm. The virtual cockpit (simulator box) will be mounted at the end effector of the robot arm. The end effector must be able to supply all three degrees of rotation (roll, pitch, yaw). Figure 1 shows the basic robot configurations. However, Figure 2 shows the shoulder and arm function (simulated translational motion), and Gyroscopic wrist function (rotational motion).

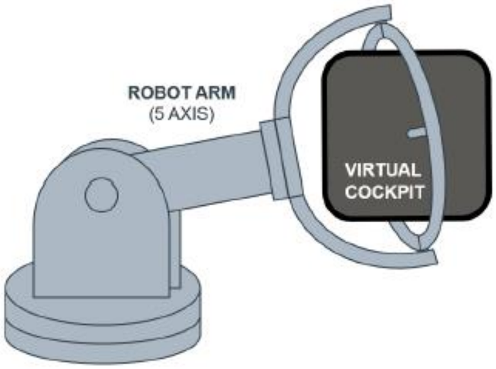


Figure 1. Basic robot configuration

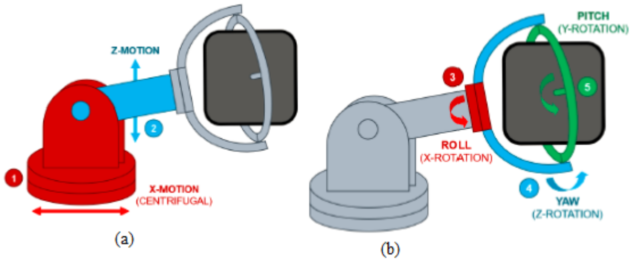


Figure 2. (a) Shoulder and arm function (simulated translational motion), (b) Gyroscopic wrist function (rotational motion)

An objective of the robot's design is to be capable of simulating the complete flight motion of the 747. A statistical study published by the Federal Flight Administration describes the loading conditions of the 747 through flight phases.

To create a maximum resultant load, each of the accelerations will be added as vectors and combined with the system’s weights. Although flight may not ever experience 2g at any one instant, this allows for worst case conditions and inherits a factor of safety. Once each components weight is determined and a factor of safety is applied, the stress simulations can be applied with these loads. As shown in Figure 3, each part of the robot must be able to withstand 2g of acceleration. Using each components weight and a factor of safety of 1.5, the max loading conditions can be calculated. Figure 3 describes the image of each of the major robot components and the system’s joints.

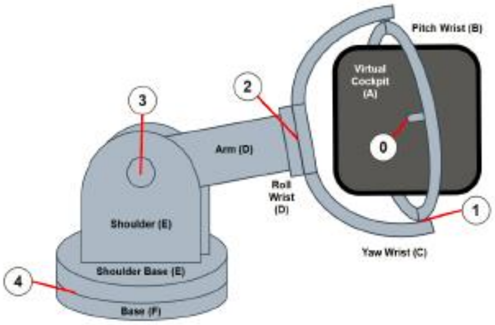


Figure 3. Robot Components (A-F) Robot Joints (0-4)

III. ROBOT MODELS

The simulator model is shown in Figure 4. The model is parametric such that characteristic of the design can easily be changed and updated throughout the model. Each component is designed such that the pin and joint is safe to meet the factor of safety requirement at max loading.

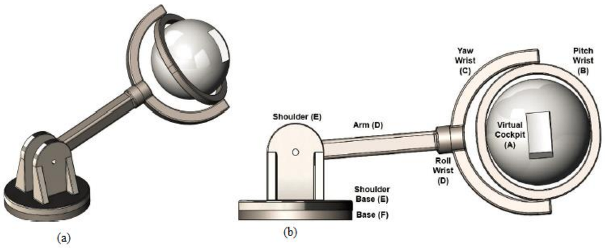


Figure 4. Simulator Isometric view (a) Labeled components (b)

The robot is structurally constructed of 1020 mild steel. This material was selected for its wide manufacturability. The large pins at each joint are machined from 316 stainless steel for its strength as shown in Figure 5.

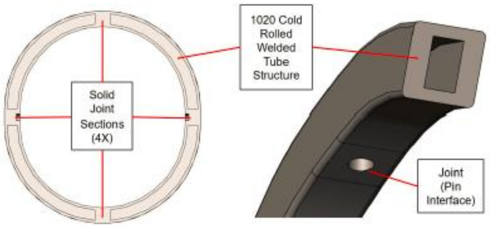

1020 Cold Rolled Welded Tube Structure
Solid Joint Sections (4X)
Joint (Pin Interface)

Figure 5. Pitch wrist cross section and structure

At the pin joints, a solid machined section interfaces with the pin. It is solid because of the stress concentrations at these points. The rest of the structure is created by welding four machined and fabricated plates together to create the tube shown in Figure 6. Shown below is the cross section of the robot which uses this general structure: solid joint sections and structural tubes. This allows for rigid joints at stress concentrations. The tubing structure keeps the weight low but maintains rigidity.

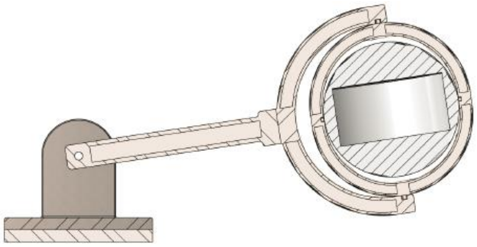


Figure 6. Robot Cross Section

Being a novel design, three configurations of the robot will be designed and simulated. Using a parametric model, each configuration is easily modeled and simulated. The configurations will represent possible implementations. In each version, the radius of the spherical virtual cockpit and the cockpit weight are the principle changes. This cascades to geometry and weight of the pitch and yaw wrist components. However, the tube's cross-sectional dimensions (wall thickness, width) remain constant. The shoulder and arm geometry remain the same through each version as shown in Figure 7.

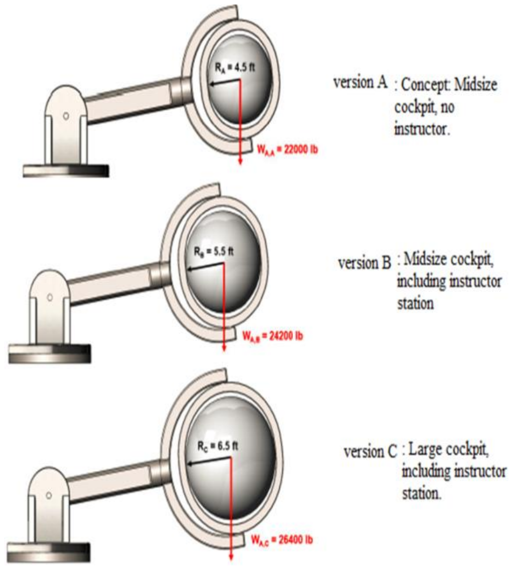

R_A = 4.5 ft
$W_{A,A}$ = 22000 lb
version A : Concept: Midsize cockpit, no instructor.
R_B = 5.5 ft
$W_{A,B}$ = 24200 lb
version B : Midsize cockpit, including instructor station
R_C = 6.5 ft
$W_{A,C}$ = 26400 lb
version C : Large cockpit, including instructor station.

Figure 7. Different versions of cockpit

IV. Stress Analysis

Stress analysis was performed to ensure the safety of each of the robot component. Using SolidWorks static simulation, each component is simulated in its axis which is most susceptible to failure. Each component is designed to be safe for at least 2g of acceleration (and a FOS of 1.15). The static gravity load will also be simulated. Figure 8 shows, ach of the components are meshed using a blended curvature-based mesh. This meshing method is the best available in SolidWorks for the complex geometry of these models. Mesh controls are applied at stress concentration points to increase the accuracy of the simulation results at the failure points. For each component of each version, the meshes and conditions will be tabulated. In these plots, the purple and green arrows represent the loads and the fixtures, respectively. Additionally, the Von Mises stress plots will be tabulated for both the static gravity load and the dynamic 2g load.

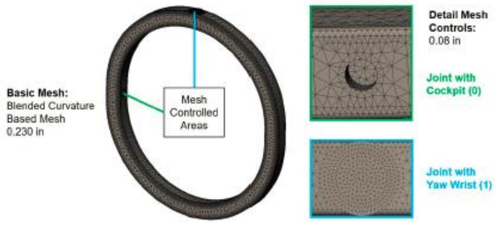
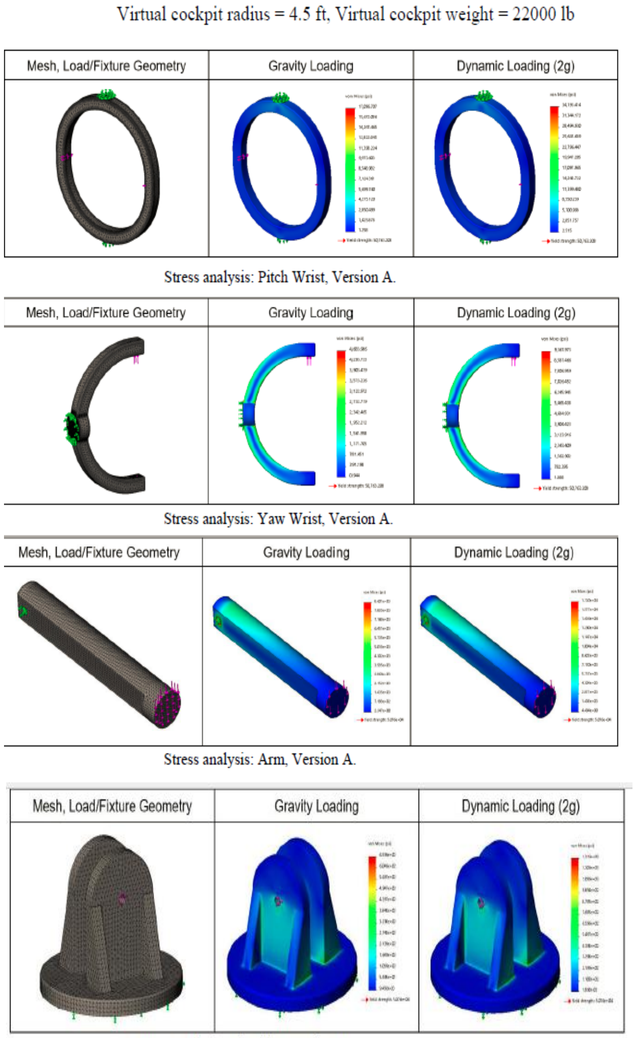


Figure 8. Mesh details for the pitch wrist component (version A)

Figures 9, 10 and 11 show the stress analysis of version A, version B and version C.


Virtual cockpit radius = 4.5 ft, Virtual cockpit weight = 22000 lb
Mesh, Load/Fixture Geometry
Gravity Loading
Dynamic Loading (2g)

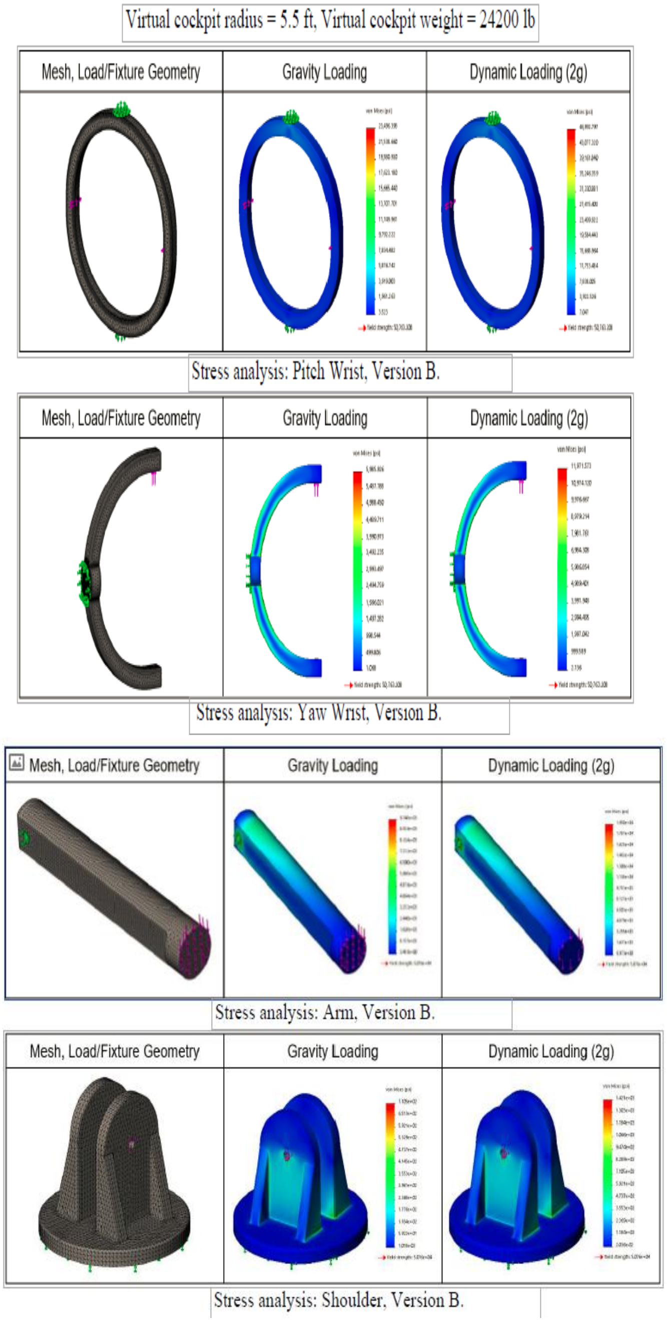
Stress analysis: Pitch Wrist, Version A.

Stress analysis: Yaw Wrist, Version A.

Stress analysis: Arm, Version A.

Stress analysis: Shoulder, Version A.

Figure 9. Stress analysis of version A


Virtual cockpit radius = 5.5 ft, Virtual cockpit weight = 24200 lb
Mesh, Load/Fixture Geometry
Gravity Loading
Dynamic Loading (2g)
von Mises (psi)

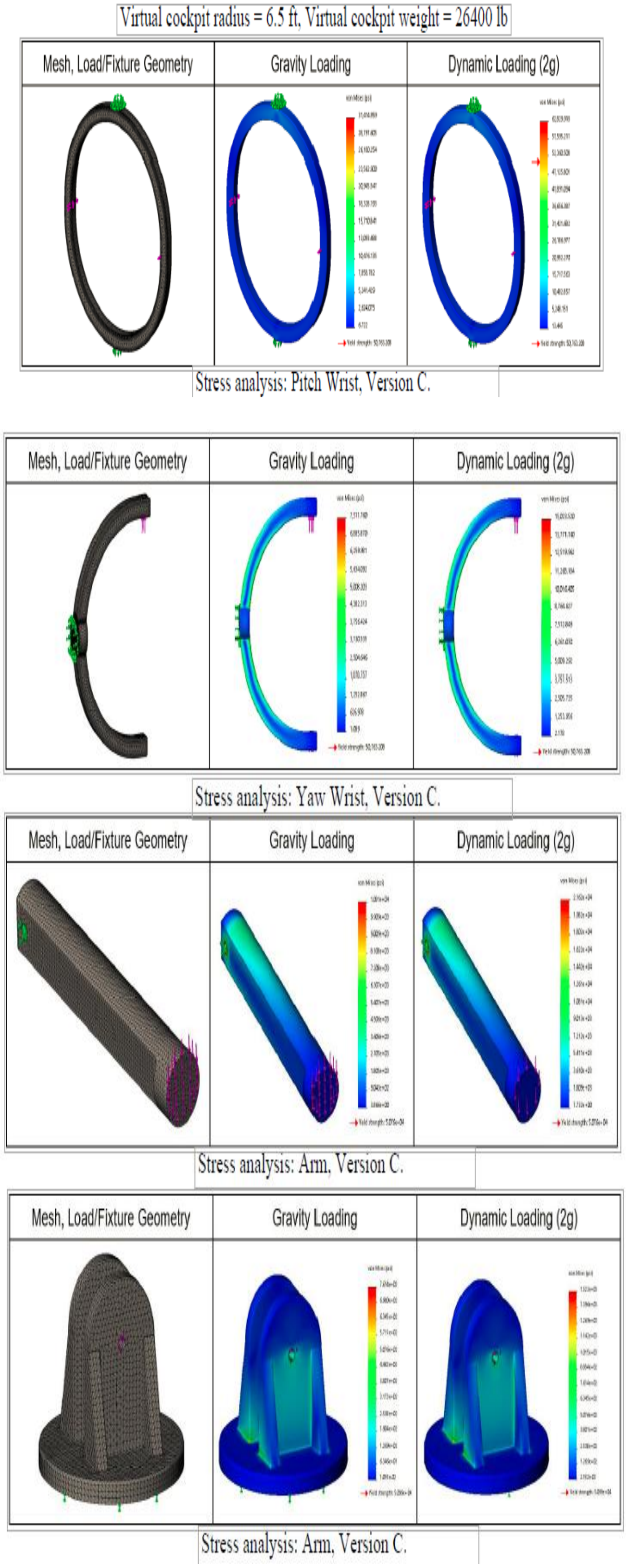
Stress analysis: Pitch Wrist, Version B.

Stress analysis: Yaw Wrist, Version B.

Stress analysis: Arm, Version B.

Stress analysis: Shoulder, Version B.

Figure 10. Stress analysis of version B


Virtual cockpit radius = 6.5 ft, Virtual cockpit weight = 26400 lb
Mesh, Load/Fixture Geometry
Gravity Loading
Dynamic Loading (2g)
von Mises (psi)

Stress analysis: Pitch Wrist, Version C.

Stress analysis: Yaw Wrist, Version C.

Stress analysis: Arm, Version C.

Stress analysis: Arm, Version C.

Figure 11. Stress analysis of version C

Figure 12 shows the minimum factor of safety during gravity loading for each component. The points can be observed as the maximum g-force each component can withstand. Therefore, the lowest point describes the maximum g-force each configuration can withstand. These results are tabulated in table 1 including a 1.15 factor of safety.

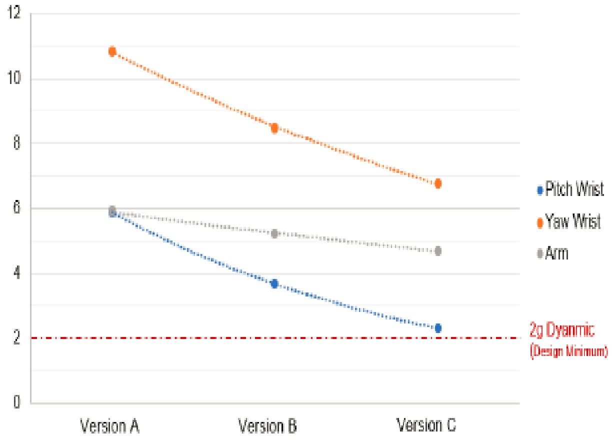


Figure 12. Gravity Loading Minimum Factor of Safety

TABLE 1. MAX SIMULATOR ACCELERATION RESULTS

Simulator Configuration	***Maximum System Acceleration***
Version A	5.11g
Version B	3.21g
Version C	2.00g

V. CONCLUSION

This design and analysis were an interesting experience in a product's design. The design of the robot arm and spherical simulator concept are atypical from conventional simulator designs but may have merit for their wide motion profile. Using the robot's main arm, pilots can experience translational motion over 2g of acceleration for each design concept. The robot's gyroscopic wrist allows the pilots to experience rotational motion in all three axes. These five axes of motion can give pilots a very accurate experience. The robotic arm succeeded in meeting the design objective of accurately simulating the motion of Boeing 747.

Designing three configurations allowed for comparison and opens opportunities for different implementations. By keeping the robots tube structure constant, the different configurations have different performance. The smallest version (A) could withstand up to 5.1g, meaning this simulator could be refitted to simulate a more aggressive motion profile, like that of a fight jet.

There are many areas for improvement in this project. To improve the accuracy of the loading conditions, motion simulation could be conducted to determine dynamic loads (as opposed to the relatively rough hand calculations). The geometry of the robot could also be furthered optimized, including the length of the robot arm, wrist geometry, and tube cross sections dimensions.

REFERENCES

[1] Meyer A. Nahon, and Lloyd D. Reid, "Simulator motion-drive algorithms - A designer's perspective", ARC, VOL., 13, 2012

[2] Lloyd D. Reid and Meyer A. Nahon, " Response of Airline Pilots to Variations in Flight Simulator Motion Algorithms", ARC, VOL., 25, 1988

[3] Eric L. Groen and Jelte E. Bos, "Simulator Sickness Depends on Frequency of the Simulator Motion Mismatch: An Observation", MIT Press, VOL 17, 2008.

[4] Nikhil J.I Garrett and Matthew C. Best, "Model predictive driving simulator motion cueing algorithm with actuator-based constraints", International Journal of Vehicle Mechanics and Mobility, VOL. 51, 2013.