Crowd Monitoring for Pandemic using Mask R-CNN

Ding-Bang Chen, Hooman Samani, Chan-Yun Yang and Gene-Eu Jan

***Abstract*— In this research, a vision system based on pixel-level deep learning image recognition algorithm is proposed. This system can be used to identify whether people entering and exiting a building are wearing a mask, and to calculate the cumulative number of people in a specific space. The system monitors and analyses crowd entering and exiting and as soon as finds that they are not wearing mask or the number of people in a specific space has reached the upper limit, the system sends a warning notification to the administrator. Through advanced detection systems, personnel can be controlled, especially for the prevention of infectious diseases during pandemics. The proposed system could be significant module for artificial intelligent systems developed for pandemics**

I. Introduction

Coronavirus is a highly contagious virus which may cause pandemics. Several coronaviruses are known to cause respiratory infections, from the common cold to more serious diseases such as Middle East Respiratory Syndrome (MERS) [1] and Severe Acute Respiratory Syndrome (SARS) [2]. Using deep learning technology we could develop systems to automate several tasks specifically for this special situation. By employing the state of the art in deep vision system [3], routine and simple tasks can be handed over to the robot to be completed and reduce the pressure of human resources.

II. Research Applications and Methods

A. *Mask R-CNN*

1. Mask R-CNN structure

Mask R-CNN[4] uses a two-stage process where the first stage is RPN and the second stage predicts the bounding box and the target category in parallel, and then uses RoIAlign to output an additional binary mask. Different from other frameworks, the classification of objects depends on mask prediction. In addition, the FPN method uses feature maps with high feature levels in different dimensions for prediction, and also improves the shortcomings of ROI Pooling in Faster R-CNN [5], so that the longitude of the bounding box and object positioning can truly reach the pixel level. It could increase the accuracy rate by 10~50%. For the improvement of object positioning accuracy, Mask R-CNN, which includes the concept of FCN [6], can achieve very good instance segmentation for objects. The overall architecture of Mask R-CNN in presented in Fig. 1.

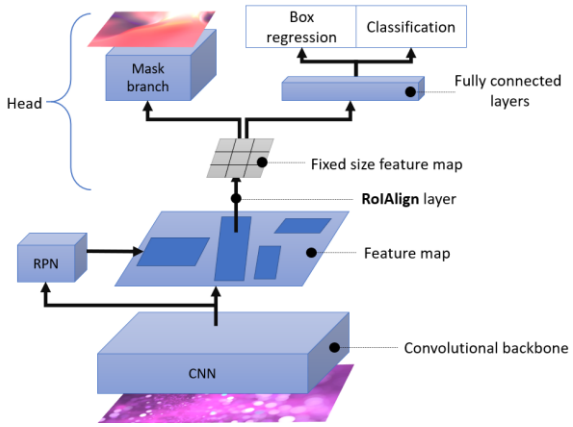

Fig. 1: Mask R-CNN architecture diagram

2. RPN(Region Proposal Network)

In Mask R-CNN, the RPN is first used to find the proposal, and then the RPN results are further screened through Mask R-CNN. The type and location of each target in the picture can be found in the RPN. It's structure is shown in Fig.2.

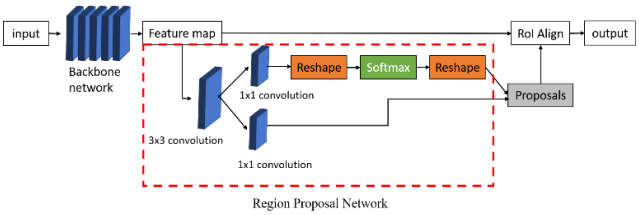

Fig. 2: RPN structure

This system performs another convolution on the previously output feature map and then divides it into two branches. The first branch is used to determine whether the anchors belong to the foreground or the background through softmax function. The results of the two classifications are sent to the proposal layer. The other branch is bounding box regression to correct anchors to obtain accurate proposals, and similar to the other branch it also sends the results to the proposal layer. After the above steps, the final proposal layer adjusts the position of each anchor according to the regression result of bounding box regression and excludes anchors that exceed the boundary and have a high degree of overlap. The next step is to sort according to the foreground score, take out a small number of anchors as proposals, calculate the coordinates of each proposal as the coordinates on the original

Ding-Bang Chen, is with the Department of Electrical Engineering, National Taipei University, New Taipei City, 23741 Taiwan (e-mail: a9585756555@gmail.com).

Hooman Samani, is with the School of Engineering, Computing and Mathematics, University of Plymouth, UK (e-mail: hooman.samani@plymouth.ac.uk).

Chan-Yun Yang, is with the Department of Electrical Engineering, National Taipei University, New Taipei City, 23741 Taiwan (e-mail: cyyang@mail.ntpu.edu.tw).

Gene-Eu Jan, is with the Department of Electrical Engineering, National Taipei University, New Taipei City, 23741 Taiwan (e-mail:gejan@mail.ntpu.edu.tw).

image, and finally send it to the RoI Align layer to complete the overall process of RPN.

3. ResNet (Residual Network)

ResNet [7] (residual neural network) is an artificial neural network (ANN) of a kind that builds on constructs known from pyramidal cells in the cerebral cortex. ResNet proposed that deep network training using the conceptual history of residual learning is easier and opens the door to deep networks.

The network design of ResNet is simply to add a route for simple addition, as shown in Fig.3. The convolution layer combined in this way is called a block. The method is simple but makes deep network training much easier. When stacking deeper networks, ResNet designed a bottleneck block to reduce the width of the 3x3 convolution and greatly reduce the amount of computation required.

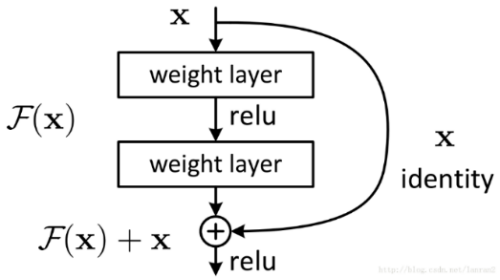

Fig. 3: ResNet residual learning unit

In Fig.3 we define a new term called Residual. The Residual operation value can be expressed as:

$$\text{Residual}=H(x)-x$$

That is, the difference output of Residual can be obtained through the above formula. Therefore, Residual is also a function of x, so it is also written as F(x). This can be expressed as:

$$F(x)=H(x)-x$$

It can be inferred from the above formula that we can shift the term of the expression to express the output of Residual as:

$$H(x)=F(x)+x$$

The above formula calculates Residual and sends the final result to the relu layer for calculation. Therefore, assuming that if the residual value is equal to zero, the initial input and the final input are the same. This unit is called identity mapping. The intention is not to degrade the model. When the residual value is not equal to 0, it is through residual mapping, so that the weight layer of each layer can learn some new and more complex features.

In the ResNet network structure diagram (Fig. 4), for the input part, we use general convolution, and use a large stride to lower the resolution. In the stage block section, ResNet has a total of 4 stage blocks, and each stage block is made up of a stack of several building blocks. Whether using stride or pooling, each stage generally reduces the resolution and enlarges the channel first, and then does a series of residual learning. In the final output part, different outputs are designed according to the task.

layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112	7×7, 64, stride 2				
		3×3 max pool, stride 2				
conv2_x	56×56	$\begin{bmatrix} 3\times3, 64 \\ 3\times3, 64 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 64 \\ 3\times3, 64 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$
conv3_x	28×28	$\begin{bmatrix} 3\times3, 128 \\ 3\times3, 128 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 128 \\ 3\times3, 128 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times8$
conv4_x	14×14	$\begin{bmatrix} 3\times3, 256 \\ 3\times3, 256 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 256 \\ 3\times3, 256 \end{bmatrix}\times6$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times6$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times23$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times36$
conv5_x	7×7	$\begin{bmatrix} 3\times3, 512 \\ 3\times3, 512 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 512 \\ 3\times3, 512 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$
	1×1	average pool, 1000-d fc, softmax				
FLOPs		1.8×10^9	3.6×10^9	3.8×10^9	7.6×10^9	11.3×10^9

Fig. 4: ResNet structure diagram

4. FPN (Feature Pyramid Networks)

FPN (Feature Pyramid Networks) [8] (Fig. 5) is a method that uses models to efficiently extract features of each dimension in a picture. In computer vision, multi-dimensional target detection has always been to generate feature combinations that reflect different dimensional information by taking reduced or enlarged images of different dimensions as input.

In the structure of FPN, in order to improve the feature extraction method of the CNN network, the final output feature can better represent the information of each dimension of the input image. Its basic process has three distinctions:

1. The path from bottom to top is the generation of different dimensional features from bottom to top.
2. The top-to-bottom path is the feature supplement and enhancement.
3. The association expression between the features of the CNN network layer and the final output features each dimension.

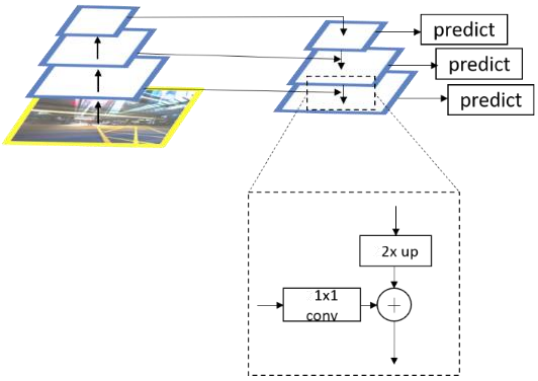

Fig. 5 FPN (Feature Pyramid Networks) structure diagram

B. Our proposed system

When people enter the gates of public places, we can use the vision system on the epidemic prevention robot to detect them. The system architecture is shown in Fig. 6. The visual system can detect the number of people passing through and whether they wear a mask through the trained weights and the Mask R-CNN model. Wearing a mask is an important task to prevent the spread of the virus, and through the human numerical control module, the number of people in a specific space can be prevented from being excessive, resulting in the inability to maintain social distance. The integration of the above-mentioned functions can send out notifications and warnings when someone does not wear a mask or there are

too many people in the community, so that measures can obtain information in real time to prevent epidemic prevention problems. The above-mentioned vision system can be applied to the entrances of schools and public spaces or to automatic surveillance systems to achieve epidemic prevention and reduce manpower requirements.

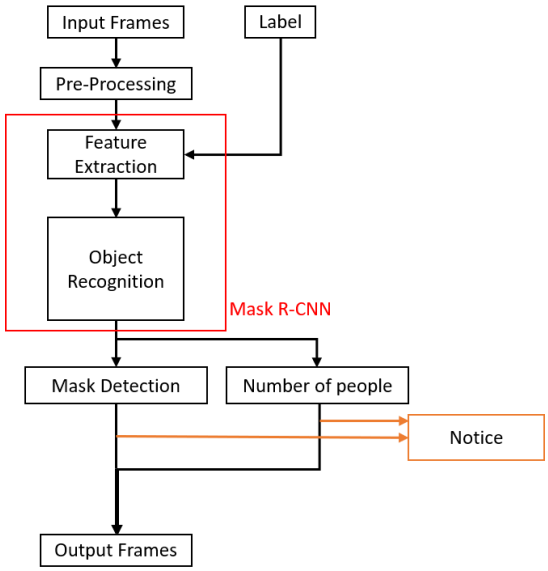

Fig. 6 Proposed system block diagram

III. IMPLEMENTATION

A. *Mask Detection*

1.1 Mask Detection

Regarding the training part of the mask data, we use the Kaggle dataset of multi-person and environment-intensive photos for data annotation. The dataset contains many different scenes, including complex multiplayer scenes and a single environment, including 700 photos without and wearing masks. In order to strengthen the accuracy of the model training, we also collected 300 pieces of data for training. In each photo, the facial information of each person is manually tagged, and divided into two categories: masked and non-masked. After completing the marked information, we set the training parameters, and set its category to 3 categories, including wearing a mask, not wearing a mask, background, etc. The training epoch is set to 200, that is, 100 steps are done in each epoch, a total of 20,000 iterations. Then after the path setting of the training set and the test set is completed, the training can be started. The training loss function is shown in Fig.7.

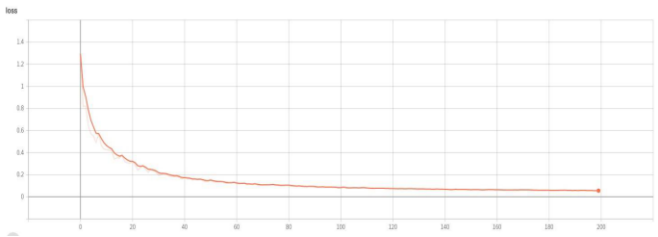

Fig. 7 Total loss figure of training

From Fig. 7, it can be found that through 20,000 iterations of training, the total loss of the first epoch is about 1.3 at the beginning, and it has converged to about 0.3 at about 20 epochs, and finally at 200 epochs too about 0.05. It can be observed that as the number of iterations slowly increases, loss also slowly drops to a position close to 0, and there is no overfitting.

In this research, 800 of the 1000 pictures are used as the training set, and the remaining 200 pictures are used as the verification set. In the process of training, repeated verifications are continuously carried out. Verification was performed every 5 epochs, and the mAP calculation of the above process was performed. Then the calculated result has been sent to tensorboard to draw the chart, as shown in Fig. 8.

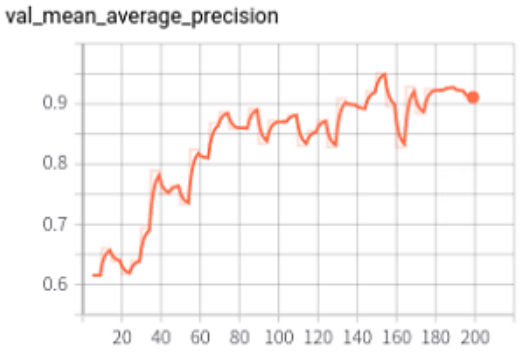

Fig. 8 Trained mAP line chart

From the above figure, it can be known that after 200 epochs of training are completed, 40 times of mAP calculation line graphs can be drawn from the graph. From the 5th epoch, about 62% of the mAP is counted every 5 times. At about 100 epochs, its mAP grows to about 86%.

Finally, at 200 epochs, it is about 91% mAP. It can be observed that as the number of iterations gradually increases, the result calculated by mAP also rises gradually. It is proved that the accuracy of this model has been significantly increased after repeated learning.

After training, in a simple environment, people who wear a mask can be clearly distinguished from those who don't, as shown in Fig. 9. The system could detect that the mask of the left image and also could detect the person without a mask on the right hand side.

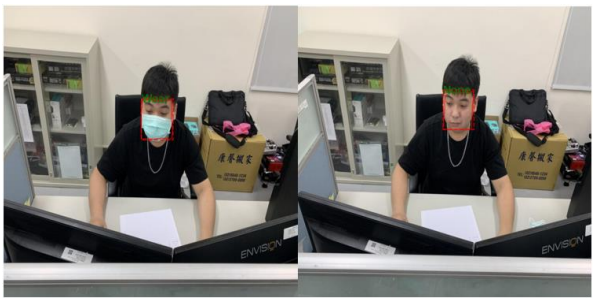

Fig. 9: Mask detection system testing

In single person and multi people scenes, there will be many targets to be detected in one screen. The mask detection part aims to be used in surveillance systems or mobile robots. Generally, in the process of detecting a target, the accuracy of the recognition will be affected by many factors, such as the occlusion of the detected target, the angle of detection, and the size of the detected target. Therefore, this research divided the detection angles into front, 45 degrees and 90 degrees for testing, as shown in Fig. 10.

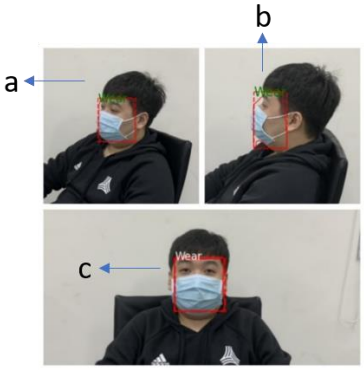

Fig. 10 Mask detection with various angles

a. the upper left is rotated by 45 degrees
b. the upper right is rotated by 90 degrees
c. the lower image is the front

B. Operation control

Group infection could cause many problems in epidemic prevention. In order to prevent too many people from gathering in a specific space where the air in the space is not circulated and it is impossible to maintain a certain social distance we have designed a cumulative system that can detect the number of people entering the area in a fixed field, to observe the number of people entering the field, and to limit the number of people in the field, In order to achieve the effect of preventing crowds.

In terms of data collection, because COCO's dataset has already defined tags that can identify people, there is no need to recollect data and manual tags. In order to effectively control the number of people entering and leaving the field, this research used Mask R-CNN with a trained model to mark objects that appear as "people" on the screen, and check them frame by frame through changes in the coordinates. Then, the people judged as entering the field are accumulated, and finally the accumulated result is compared with the upper limit of the number of people in the field set by the admin. If it exceeds, a notification will be issued (Fig. 11).

Fig. 11: Counting the number of people

IV. CONCLUSION

In this research, we have investigated the use of the deep learning vision system for detection to prevent the spread of the epidemic in order to solve the social and safety problems caused by diseases. We have developed a system for mask detection as well as early warning notifications through the detection of the number of people to prevent the inability to maintain social distance caused by high number people in a specific space.

In the future, it is expected that this system can be widely used in applications such as robotics and surveillance systems. In addition to improving the training and recognition accuracy of the system, improving occlusion and light problems are important issues to be addressed for the improvement of the system.

REFERENCES

[1] Al Hajjar, S., Memish, Z. A., & McIntosh, K. (2013). Middle East respiratory syndrome coronavirus (MERS-CoV): a perpetual challenge. Annals of Saudi medicine, 33(5), 427-436

[2] Bchetnia, M., Girard, C., Duchaine, C., & Laprise, C. (2020). The outbreak of the novel severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2): A review of the current global status. Journal of infection and public health.

[3] Chung, C. L., Chen, D. B., & Samani, H. (2020, November). Action Detection and Anomaly Analysis Visual System using Deep Learning for Robots in Pandemic Situation. In 2020 International Automatic Control Conference (CACS) (pp. 1-6). IEEE.

[4] He, K., Gkioxari, G., Dollár, P., & Girshick, R. (2017). Mask r-cnn. In Proceedings of the IEEE international conference on computer vision (pp. 2961-2969).

[5] Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. arXiv preprint arXiv:1506.01497.

[6] Long, J., Shelhamer, E., & Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 3431-3440).

[7] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 770-778).

[8] Lin, T. Y., Dollár, P., Girshick, R., He, K., Hariharan, B., & Belongie, S. (2017). Feature pyramid networks for object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 2117-2125)