Face Detection Using YOLO

Abhishek Rana*
Department of Computer Science and Engineering
KIET Group of Institutions
Ghaziabad, India
Email: *abhishek.1822cs1172@kiet.edu,

***Abstract*—Deep learning is a buzzword these days, and it's a new phase of machine learning that teaches computers to detect patterns in enormous amounts of data. It primarily describes learning at several levels of representation, which aids in making understanding of text, voice, and visual data. Many businesses use a convolutional neural network, a sort of deep learning, to deal with the objects in a video sequence. Deep Convolution Neural Networks (CNNs) have demonstrated excellent performance in terms of object detection, picture classification, and semantic segmentation. Object detection is described as the process of classifying and locating objects. Face detection is one of the most difficult pattern recognition issues. Deep learning is a new phase of machine learning that teaches computers to find patterns in massive volumes of data, and it's a buzzword these days. It mostly refers to learning at several levels of representation, which aids in the comprehension of text, audio, and visual data. To deal with the objects in a video sequence, many firms utilise a convolutional neural network, which is a type of deep learning. In terms of object detection, picture categorization, and semantic segmentation, Deep Convolution Neural Networks (CNNs) have shown to be quite effective. The process of classifying and locating things is referred to as object detection. One of the most difficult pattern recognition problems is face detection.**

***Index Terms*—YOLO,FACE ,Detection**

I. Introduction

Early research focused on several hand-crafted feature extraction methods that were used to train typical machine learning algorithms for detection and recognition. It increases the computational power and time required to extract features, as well as the accuracy of the findings. The same was achieved using neural network models and then deep neural networks to overcome computing time, power, and accuracy. Deep learning [1] models such as convolutional neural networks, recurrent neural networks, and others exist, but deep convolutional neural networks (CNNs) [2] are the best at finding patterns in images. CNN is also capable of accurately classifying, detecting, and labelling the item. R-CNN [3], Fast R-CNN [4], and Region-based CNN (R-CNN),In recent years, object identification networks such as faster R-CNN [5] and YOLO [6] have been popular. Face recognition has numerous applications. Face recognition algorithms rely heavily on it. Face recognition can be used for a variety of purposes, including surveillance and security system authentication. It can also aid in emotion recognition, with subsequent analysis based on the identified emotion being used for emotion-based applications. As a result, it's thought to be a way to convey detailed information about a person, such as age, emotion, gender, and so on. Face detection can also be used to automatically focus on human faces in a camera, assign a tag, and identify different aspects of a face. Face detection software has gained popularity in computer vision and pattern recognition.Deep learning algorithms were used to model diverse circumstances. Face detection remains a difficult topic in computer vision due to huge variations caused by occlusions, illumination, and angles. As a result, accuracy, training time, and processing time for recognising faces in real-time movies are still research topics. The second half of this paper discusses similar work on face detection methods. The third section explains how the YOLO framework detects objects. Section four explains the proposed work. In section five, the experimental setup and dataset details are discussed. Section six examines the findings. Finally, in section seven, the conclusion and future work are discussed.

II. Background and Related Work

Face identification is one of the most difficult tasks in pattern recognition. Early in 1994, Vaillant et al. [7] used a neural network approach to detect faces. They proposed a model that used a neural network to recognise the presence or absence of a face in a picture. The entire image was scanned with the network at all feasible points in this technique. [8] A rotation invariant face detection method was used in 1998, where a "router" network evaluated the face's orientation and an appropriate detector network was deployed. In the year 2002, Gracia built a neural network for recognising the semi-frontal face from a complicated image [9]. Osadchy [10] suggested a convolutional neural network for facial pose estimation and detection. Harcascading for facial feature detection was presented by Wilson et al. [11]. When the face is subjected to diverse illuminations, positions, and expressions, however, limitations occur. Deep learning methods have been used to recognise faces in recent years. CNN (convolutional neural network) [12] is one of the most prominent models for it. Object detection is also improving because to the faster R-CNN. Using the YOLO framework, this study presents an architecture for a convolutional neural network to recognise the face. Hand-crafted features are not used in our architecture. Faces are recognised using a CNN that extracts features on its own. A model's training and testing are both important carried out on two GPU and it detects the faces at a faster rate in real time.

III. Methodology

YOLO is a cutting-edge deep learning framework for object recognition in real time. It beat typical detection datasets like PASCAL VOC [13] and COCO [14]. It is a better model than the region-based detector. In comparison to other detection networks, real-time detection of the item is much faster. This model may run at various resolutions, resulting in high speed and precision. The photos can be adjusted to a random scale to increase scale invariant performance. The detector should be able to learn characteristics for images of various sizes. Object detection should be quick and precise enough to recognise a wide range of objects [15]. The YOLO frameworks are becoming increasingly fast and precise for detection with the help of neural networks. For a small set of items, there is still a limitation. Object detection datasets are currently limited in comparison to classification and tagging datasets. Thousands of photos with tags that are object coordinates in the image make up the object detection databases. Millions of photos with classifications make up the categorization datasets. It costs more to assign an item tag to an image for detection than it does to assign a label for classification. Region-based In a picture, CNN creates bounding boxes and then applies the classifier to these boxes. To eliminate repeated detections, the bounding boxes are adjusted using post-processing techniques such as non-maximum suppression. Multiple bounding boxes and object class probabilities can be predicted by a single CNN. YOLO improves performance by being quick to detect. During the training and testing phases of YOLO (You Only Look Once), a single neural network is applied to the entire image. It stores information about the appearance and classes of people. The bounding box is predicted in our work using features from the image. Parallel bounding boxes are predicted over a picture. As a result, the network analyses the entire image as well as the object within the image. YOLO allows for end-to-end training while maintaining real-time pace. This allows for high average precision to be maintained. The following is how YOLO works: S x S grids are used to split the input image. In the event that the object's centre falls within a grid cell, the grid cell is responsible for detecting the object. The bounding box and confidence score for each cell in the grid are predicted. The precision with which the object is detected in the bounding box is represented by the confidence score. The confidence score is zero if no object is discovered in the cell; otherwise, it is determined using the intersection over union (IOU) between the predicted box and the ground truth. In the bounding box, there are primarily five predictions: x, y, w, h, and confidence.

IV. Proposed Architecture

Our suggested network accepts a 448 by 448 colour image as an input. The design is made up of 7 convolutional layers and a 2 x 2 max pooling layer. Then three fully linked layers are joined, followed by the output layer and the final fully connected layer. The fully connected layer predicts the coordinates and probabilities, whereas the convolutional layers detect simple to complex characteristics in the images. Finally, the output layer employs the NMS (Non-Maximum Suppression) technique to predict both class probabilities and bounding box coordinates.

V. EXPERIMENTAL SETUP DATASET INFORMATION

Two machines are used in the experiment. The first test is run on a Core i5 processor with 8GB of RAM and a 2GB GeForce 820M GPU. The second test is run on a Core i7 processor with 16 GB RAM and a 4 GB NVIDIA GTX 1050 Ti GPU. On the FDDB (Face Detection Dataset and Benchmark) dataset, the proposed convolutional neural network architecture is trained and tested for face detection [16]. In order to train the suggested architecture, we employed the FDDB Dataset. In a series of 2845 photos, there are 5171 faces. This dataset contains regions of people aimed to research the detection problem. This research involves 2667 photos, with a total dataset size of 52.2 MB (in our study).

VI. Result Analysis

The gradient decent optimizer algorithm was used to train the model for 25 epochs. After 20 epochs, accuracy remained practically constant at 92.2 percent, and the optimal value of learning rate was determined after attempting several values, and it was 0.0001, as shown in Fig. 1. For comparison of experimental analysis on CPU and GPU, the same epochs and learning rate are used. On the test dataset, 92.2 percent accuracy was achieved for 20 epochs, as shown in Fig. 2. Different batch sizes were also used to train the network. The batch sizes were maintained at 1, 8, 16, and 32. On a 2 GB 820M Graphics card, it was discovered that when the batch size was 32 or 16, the network was unable to train. It happened because the GPU RAM was insufficient to handle the higher batch size. Figure 3 shows the same thing. The weight file and network configuration file were tested on different resolutions of movies after a network had been trained. Resolutions were also a factor in the FPS (frames per second) rate, as shown in Figure 4. It was discovered that as the resolution was reduced, the FPS increased. Because a low-resolution image includes fewer pixels, the GPU can process it more quickly due to fewer parameter calculations.

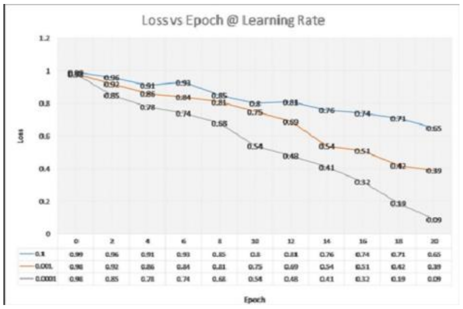

Fig. 1. Loss Vs Learning Rate

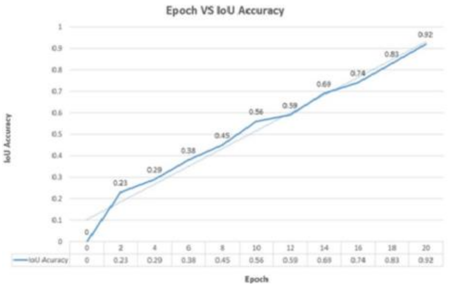

Fig. 2. IoU accuracy vs Epoch

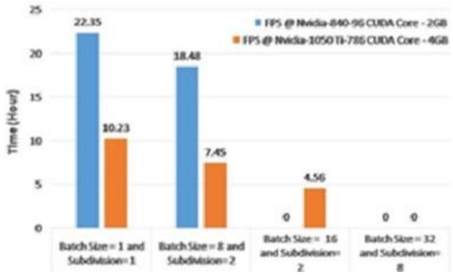

Fig. 3. Batch size vs Training time (hours)

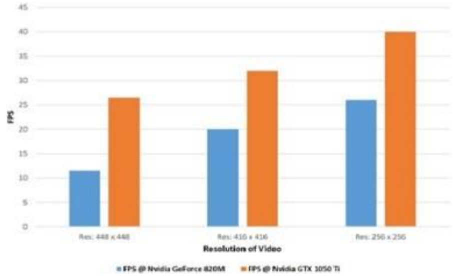

Fig. 4. Resolution of video vs FPS

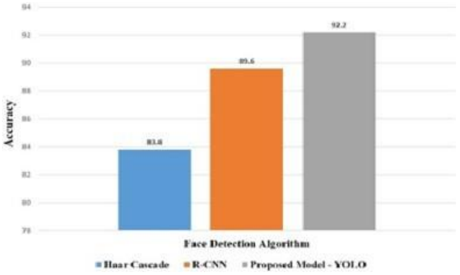

Fig. 5. Comparison of accuracy of proposed model with other face detection algorithm

VII. Conclusion

After fine-tuning all parameters and hyper-parameters of the proposed model, the accuracy of the proposed model was compared to other face identification techniques. The suggested model was shown to be more accurate than the haar cascade technique and the R-CNN based face detection model (Fig. 5).

It may be inferred that deep learning demands a high-configuration NVIDIA graphics card to process a large amount of data (GPU). The task can be computed at a faster rate if the GPU configuration is high. There are several characteristics that are responsible for detecting a face in an image or video. Following are some conclusions that may be drawn from the proposed model's examination. To begin with, the learning rate is affected by the network size as well as the object size. When the network is medium or big, and the object size is tiny, the learning rate should be kept low. The network in our project has 18 layers,0.0001 is the calculated learning rate. Second, the more times the dataset is trained on the network, the better the outcomes. It also causes data overfitting, hence the epoch size should be kept at a level that does not cause network overfitting or underfitting. After 20 epochs, we discovered that the IoU accuracy obtained is the best, at 92.2 percent. In addition, image resolution is critical. The image resolution is inversely related to the frames per second as determined. The proposed model can be improved in the future to recognise very small faces, varied viewpoint modifications, and partial face detection.

References

[1] Y. Lecun, Y. Bengio, and G. Hinton, "Deep learning," Nature, vol. 521, no. 7553, pp. 436–444, 2015.

[2] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," Proceedings of the 25th International Conference on Neural Information Processing Sy

[3] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation," in 2014 IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 580–587.

[4] R. Girshick, "Fast R-CNN," Proc. IEEE International Conference on Computer Vision, ICCV 2015, pp. 1440–1448, 2015.

[5] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: towards realtime object detection with region proposal networks," Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1. MIT Press, pp. 91–99, 2015.

[6] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," 2015.

[7] R. Vaillant, C. Monrocq, and Y. Lecun, "Original approach for the localisation of objects in images," IEEE Proceedings on Vision, Image, and Signal Processing, vol. 4, 1994.

[8] H.A. Rowley, S. Baluja, T. Kanade, “Neural network-based face detection”, IEEE Trans. Pattern Anal. Mach. Intell., vol. 20, no. 1, pp. 23–38, 1998.

[9] C. Garcia and M. Delakis, ”A neural architecture for fast and robust face detection,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 26, no. 11, pp. 1408–1423, 2004.

[10] M. Osadchy, Y. Le Cun, and M. L. Miller, “Synergistic Face Detection and Pose Estimation with Energy-Based Models,” Journal of Machine Learning Research, vol. 8, pp. 1197-1215, 2007.

[11] F. J. Phillip Ian, “Facial feature detection using Haar classifiers,” J. Comput. Sci. Coll., vol. 21, no. 4, pp. 127–133, 2002.

[12] H. Li, Z. Lin, X. Shen, J. Brandt, and G. Hua, “A Convolutional Neural Network Cascade for Face Detection.”, IEEE International Conference on Computer Vision and Pattern Recognition, CVPR 2015, pp. 5325-5334, 2015.

[13] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, “The PASCAL Visual Object Classes (VOC) Challenge”, International Journal of Computer Vision, vol. 88, no. 2, pp. 303-338, 2010.

[14] T.-Y. Lin et al., “Microsoft COCO: Common Objects in Context,” European Conference on Computer Vision, ECCV 2014, Lecture Notes in Computer Science, vol 8693. Springer, Cham, pp. 740-755.

[15] C. Szegedy, A. Toshev, and D. Erhan, “Deep neural networks for object detection,” Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2. Curran Associates Inc., pp. 2553–2561, 2013.

[16] V. Jain and E. Learned-Miller, “FDDB: A Benchmark for Face Detection in Unconstrained Settings.”, Technical Report UM-CS-2010-009, Dept. of Computer Science, University of Massachusetts, Amherst. 2010.