JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

People Counting in Crowd: Faster R-CNN

[1]Shreya Varkuti, [2]SaikumarThumma, [3]PranavAddala [4]Muni Sekhar Velpuru, [5]H Venkateswara Reddy
[1,2,3]Student, [4,5]Associate Professor, [1,2,3,4]Department of IT & [5]CSE
Vardhaman College of Engineering Hyderabad, India
munisek@gmail.com, pranavaddala13@gmail.com, varkutishreyareddy@gmail.com

Abstract— People counting in a crowd is a significant challenge in the field of computer vision. Head detection-based approaches are utilized instead of density map-based crowd counting techniques to get more trustworthy crowd counting findings. This is because, in the case of density maps, the right location does not necessarily contribute to the final crowd count. This leads to untrustworthy results, particularly in the case of false positives. As a result, solving the problem of head detection in cluttered settings is a difficult issue. A population count may be required for statistical purposes that aid in the development of marketing plans, or it may be utilized for crowd control in various scenarios. Image processing is a technique of improving or extracting information from a photograph by performing operations on it. In our project, the system's input is a surveillance system's picture/video, which is then separated into image frames. Our proposed system calculates the number of people in the scene using the Faster R-CNN object detection algorithm.

Keywords – R-CNN, Untrustworthy, False Positives, Surveillance

I. INTRODUCTION

People Counting is the process of computing the people in specified area. In general, we use an electrical instrument to count the number [1] of persons passing through a corridor or entry for finding any specific patterns or customer visiting pattern in an organization etc. Estimating the number of people in a given region may be incredibly important information for both security and safetyreasons (for example, an unusual shift in the number of people could indicate the cause or result of a deadly incident)as well as economic ones (for instance, optimizing the schedule of a public transportation system on the basis of the number of passengers). As a result, this topic has been tackled in various studies in the domains of video analysis and intelligent video surveillance.

Two ways have been used to address the problem of people counting. People in the scene are first individually recognized, using some sort of segmentation and object detection, and then counted in the direct technique (also known as detection based). Instead [2], in the indirect technique (also known as map based or measurement based), counting is done by measuring some attribute that does not need the identification of each individual in the scene separately. Because accurate segmentation of persons in a picture is a complicated problem that cannot be handled consistently, especially in crowded settings, the indirect technique is thought to be more resilient.

Aside from the video processing methods that are routinely employed for surveillance in public locations, audio analysisis a valuable addition. However, when a huge group of individuals arrives, the majority of image processing systems, which frequently employ object detection and tracking, find it difficult to calculate their number. Background extraction-based techniques, such as those developed at Gdansk University of Technology's Multimedia Systems Department, fail to separate items adequately when individuals move at close distances or when their hands are linked. Other ways deal with the segmentation problem by using numerous cameras or using models of human forms derived from studying the foreground of a picture. Furthermore, given the structure under consideration, installing a large number of cameras would be impractical.

Fig .1: Crowd at different places

Many factors play a role in determining the best approach for counting things. Apart from the challenges that any image processing using Neural Networks faces, such as the size of the training data, its quality, and so on.

Counting Objects Problem Specific Challenges:

- Description of the counted items
- Overlapping
- A view from different angles
- The identified items smallest size
- Speed testing and training.

II. ITERATURE SURVEY

A. *Existing System*

In existing system, we count things by calculating a density map. The initial step is to create training samples so that a density map may be generated for each image. Annotations in the positions of pedestrians' [3] heads have been added to the image. Convolution using a Gaussian kernel is used to create a density map and normalized so that integrating it gives the number of objects. The next step is totrain a fully Convolutional network to map an image to a density map, which can then be combined to determine the number of objects. So far, we've looked at U-Net and Fully Convolutional Regression Network (FCRN) as FCN designs. U-Net is a popular FCN for picture segmentation that is frequently used with biological data. Its structure is similar to that of an auto- encoder.

A block of convolutional layers processes an input picture, followed by a pooling layer (down sampling). This method is repeated numerous times on the outputs of following blocks. The essential elements of an input image are encoded (and compressed) inthis way by the network. The second half of U-Net is symmetric, but instead of pooling layers, up sampling is used to ensure that the output dimensions match those of the input picture.[4] suggested the Fully Convolutional Regression Network (FCRN). The architecture resembles that of U-Net. The key distinction is that in the down sampling section, information from higher resolution levels is not transmitted straight to the equivalent layers in the up- sampling half. The research proposes two networks, FCRN- A and FCRN-B, with different down sampling intensities. FCRN-A pools every convolutional layer, whereas FCRN-Bpools every second layer.

Limitations

The majority of the head counting algorithms presented above employed SSD for foreground extraction and LBP feature-based Ada boost for head recognition. When compared to Faster R-CNN, SSD has a slower computing performance. These approaches are extremely light-sensitive additional restrictions or conditions If the photos are of poor quality, SSD will not deliver accurate results.

The aforesaid restrictions are an issue of the existing projects thus the main goal is to design a people counting system that can count the number of people in real time at a low cost with accurate results. The image is obtained at a vertical perspective from a video clip of a live event, and thenumber of heads present in the image is counted.

III. PROPOSED METHODOLOGY

The suggested method takes real-time video from IP cameras as input, [5] turns it into multiple frames, and feeds it to our model as training data. Even though the training images were of poor quality, we employed the Faster R- CNN method to improve the accuracy. R-CNN is faster because it generates region suggestions using a novel region proposal network (RPN), which takes less time than standard methods like Selective Search. The system is extremely efficient and has a high prediction rate. This project is straightforward, cost-effective, and simple to set up and manage.

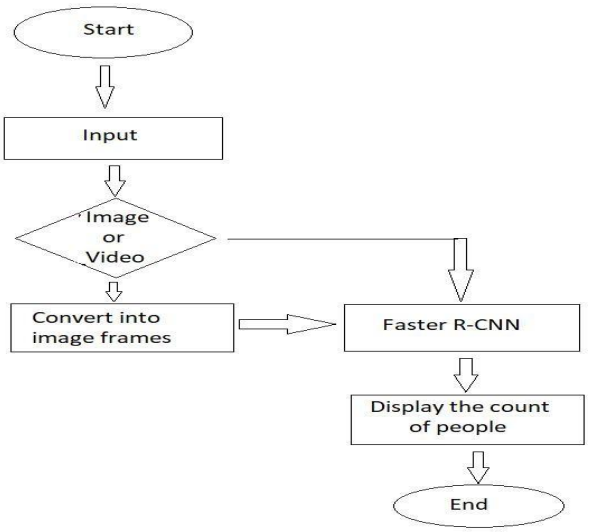

Fig. 2: Flowchart

We utilized Open CV in our project to convert real- time video into frames. The Faster R-CNN uses these images as training images. The original color image was transformed to a size of 1024x1024, before being supplied to the network input. It is supplied to the Faster R-CNN, which uses the input to generate a set of proposals, each of which has a score indicating its likelihood of being a Head as well as the Head's class/label. When estimating Head positions for the RPN, anchor boxes give a predeterminedset of bounding boxes of various sizes and ratios that are utilized as a reference. These boxes are often chosen based on object sizes in the training dataset to capture the scaleand aspect ratio of a certain head class to detect.

Anchor Boxes are usually placed in the center of the sliding glass. They aid in the detecting process by speeding it up and increasing efficiency. The anchor's initial FC layer (i.e., binary classifier) [6] has two outputs. The first is used to classify the region as a backdrop, and the second is used to classify it as an object. Each anchor is given an objectless score, which is then utilized to generate the categorization label. For each region proposal, the first layer generates a two-element vector. The region proposal is categorized as background if the first element is 1 and the second elementis 0. The area symbolizes ahead if the second element is 1 and the first element is 0. Counting can then be done.

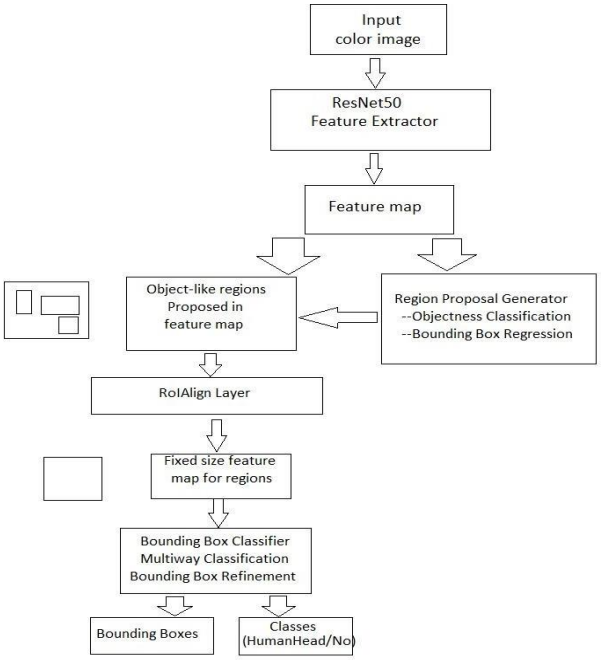

Fig. 3: Construction Diagram

ResNet50 Feature Extract

The initial phase in our method is feature extraction. The image is the input to the deep convolutional neural network, and the extracted feature map is the output. Subsequent modules will use these features as well. Resnet-50 is the foundation of our model's fundamental feature extraction network.

In 2015, He et al. presented this deep residual learning method to correctly train the deepest network. In our model, the original Resnet-50 network is divided into two parts: part one, which includes layers conv1 to conv4 x, is used to extract common characteristics, and part two, whichincludes layer conv5 x and above layers, extracts feature of proposals for final classification and regression. A region proposal network follows the feature extraction network (RPN). The features in the window are mapped to a low- dimensional vector, which will be utilized for object- background classification and proposal regression, and awindow of size n n glides onto the feature map and stays at each point. At the same time, according to k anchors, which are rectangular boxes of various shapes and sizes, k region suggestions centered on the sliding window in the original image are retrieved.

ROI Align Layer

The Region proposal network is utilized in Faster RCNN to predict objectless and regression box differences (w.r.t to anchors). To produce proposals,[7] these offsets aremerged with the anchors. Rather than the feature layer, theseapproaches are frequently the size of the input image. As a result, the recommendations must be scaled down to the level of feature maps. Furthermore, the ideas can have a variety of widths, heights, and aspect ratios.

For a downstream CNN layer to extract features, these must be standardized. Both of these issues are addressed with ROI Pool. From the feature map, ROI pooling extracts a fixed-length feature vector. The hxw Roi window is divided into a HxW grid of approximately size h/H x w/W, and the values in each sub-window are then max-pooled. Each channel of the feature map is pooled separately. However, in order to transfer the generated proposal to exactx,y indices, we perform a lot of quantization (i.e. ceiling, floor) operations. These restrictions cause a misalignment between the ROI and the extracted features. This may not affect detection/classification, which is resistant to minor perturbations, but it has a significant detrimental influence on pixel-accurate mask prediction. To remedy this, the ROI Align solution was presented, which eliminates all quantization procedures. Instead, each proposal's

exact values are computed using bilinear interpolation. The proposal is divided into a pre-determined number of smaller regions, similar to ROI Pool.

Four points are sampled inside each smaller region. Bilinear interpolation is used to calculate the feature value for each sampled point. The final output is obtained byperforming a max or average operation.

IV. IMPLEMENTATION

Faster R-CNN architecture contains 2 networks:

A. Region Proposal Network (RPN)

B. Object Detection Network

A. Region Proposal Network (RPN)

The anchors formed by sliding window convolution applied to the input feature map are output by this area proposal network, which uses a convolution feature map generated by the backbone layer as input.

Anchors

The network generates the maximum number of k- anchor boxes for each sliding window. For each of the different sliding positions in the image, [8] the default value of k=9 (3 scales of (128*128, 256*256, and 512*512) and 3 aspect ratios of (1:1, 1:2, and 2:1) is used. As a result, we get N = W* H* k anchor boxes for a convolution featuremap of W * H. These region proposals are then transmitted through an intermediate layer with 3*3 convolution and 1 padding, as well as 256 or 512 output channels (for ZF or

VGG-16). This layer's output is sent through two 1*1 convolution layers, the classification layer, and the regression layer. the regression layer has 4*N (W * H * (4*k)) output parameters (denoting the coordinates of bounding boxes) and the classification layer has 2*N (W * H * (2*k)) output parameters (denoting the probability of object or not object).

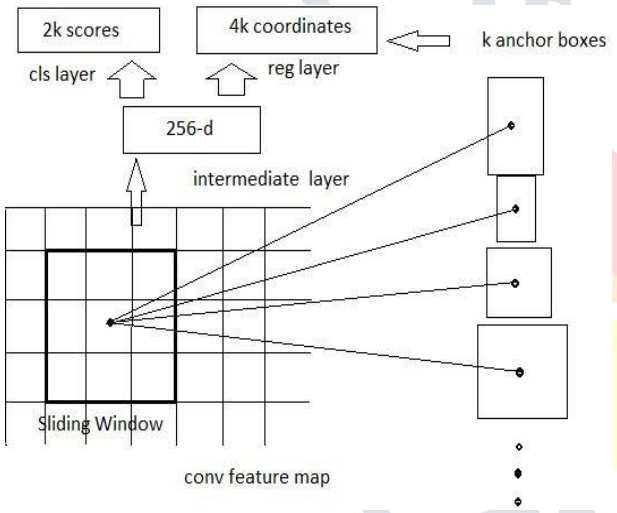

Fig. 4: Anchor Variations

Training and Loss Function (RPN)

To begin, we remove any cross-boundary anchors so that the loss function does not rise. There are around 20000(60*40*9) anchors in a typical 1000*600 image. Whenwe remove the cross-boundary anchors, [9] we're left with about 6000 anchors per image. Based on their classification and IoU, the paper also employs Non-Maximum Suppression. They employ a fixed IoU of 0.7 in this case. This brings the total number of anchors down to 2000. The benefit of employing non-Maximum suppression is that it does not degrade accuracy. Back-propagation and stochastic gradient descent can be used to train RPNs from start to finish. Each mini-batch is created from the anchors of a single image.

It selects 256 random anchors with positive and negative samples in a 1:1 ratio rather than training a loss function on each one. If there are 128 positives in an image, additional negative samples are used. We need to assign a binary class label to RPNs before we can train them (whether the concerned anchor contains an object or background). To apply a positive label to an anchor in the R-CNN article, tworequirements are used.

$$L(\{p_i\},\{t_i\}) = 1 \div N_{cls}\left(\sum_i L_{cls}\left(p_i, p_i^*\right)\right) + \lambda \div N_{reg}\left(\sum_i p_i^* \times L_{reg}\left(t_i, t_i^*\right)\right)$$

were,

Pi= is the likelihood that an anchor will include an object ornot.

Pi*= is the ground truth value of anchors that determineswhether or not they contain an object.

ti = predicted anchor coordinates

ti*= is the ground truth coordinate for bounding boxes. Lcls= stands for Classifier Loss (binary log loss over twoclasses).

Lreg = Loss of Regression (Where R is smooth L1 loss,Lreg = R(ti-ti*))
Ncls = Mini-batch size (256) normalization parameter Nreg = Regression's normalization parameter (equivalent to2400 anchor locations). To ensure that the n=btoh loss parameter is equally weighted,

B. Object Detection Network

Faster R-CNN uses an object detection network that is quite similar to that of Fast R-CNN. It can also be used as a backbone network with VGG-16. It also employs the RoI pooling layer for creating fixed-size area proposals, as well as twin softmax classifier layers and the bounding box regressor for object and bounding box prediction.

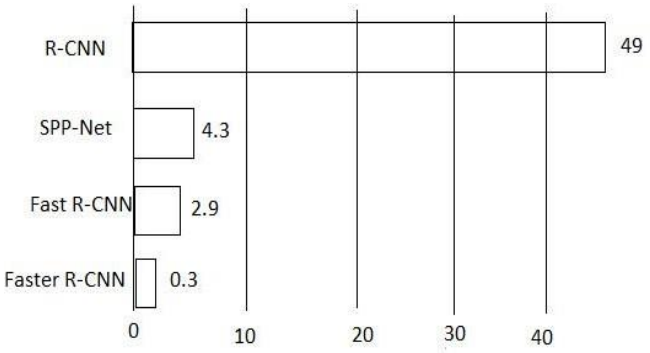

Fig. 5: Comparison of test-time speed of Object Detection

RoI Pooling

We transmit the output created by region proposal onto the Roi pooling layer, which performs the same purpose as Fast R-CNN in converting [10] different-size RPN regionproposals into a fixed-size feature map. In this article, we went through RoI pooling in great detail. The output of this RoI pooling layer is of size (7*7*D) (where D =256 for ZF).

SoftMax and Bounding Box Regression Layer

The RoI pooled feature map of size (7 * 7 * D) is then transmitted to two completely connected layers, which flatten the feature maps and then send the output to two parallel fully connected layers, [11][12] each with a distinct task assigned to them: The first layer predicts the objects in the region proposal using a SoftMax layer with N+1 output parameters (N is the number of class labels and background). A bounding box regression layer with 4* N output parameters is the second layer. This layer regresses the object's bounding box location in the image.

V. EXPERIMENTAL RESULTS

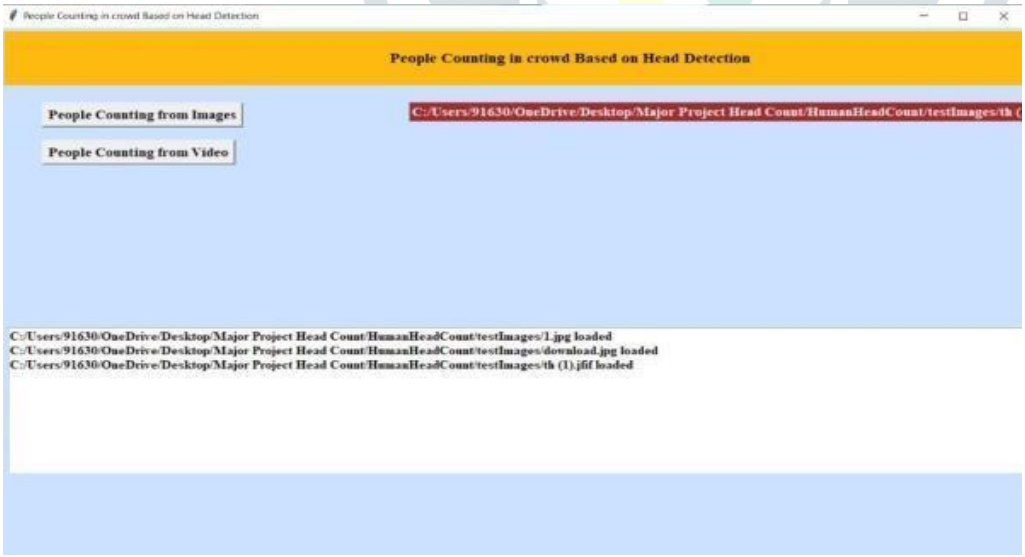

Fig 6: User Interface of our System

Fig 7: Output Screen

VI. Conclusion And Future Work

The results of the tests reveal that using deep convolutional neural networks, we can successfully detect the human head on 2D images regardless of the observer's turn. Even on low-quality images, the Faster R- CNN architecture displays acceptable accuracy rates and provides significant speed. However, a vast number of computer vision applications do not require real-time object detection. Our system uses IP cameras, which ensures its scalability for detecting waits in buffets, monitoring visitors in retail, recognizing pedestrians on roadways using outdoor video cameras, determining the workload of public transportation stations, and so on, as well as displaying the number of people.

We now use Faster R-CNN to create our model, but we will experiment with additional methods in the future, such as YOLO, to enhance accuracy.

We will try to propose a better approach to the problem by comparing the two models in terms of accuracy and computing performance.

References

[1] A. KUMAR SINGH, D. SINGH and M. GOYAL, "People Counting System Using Python," 2021 5th International Conference on Computing Methodologies and Communication (ICCMC), 2021, pp. 1750-1754, doi: 10.1109/ICCMC51019.2021.9418290.

[2] X. Shi, X. Li, C. Wu, S. Kong, J. Yang and L. He, "A Real-Time Deep Network for Crowd Counting," ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2020, pp. 2328- 2332, doi: 10.1109/ICASSP40776.2020.9053780.

[3] S. Thasveen M. and L. Mredhula, "Real Time Crowd Counting: A Review," 2020 International Conference on Futuristic Technologies in Control Systems & Renewable Energy (ICFCR), 2020, pp. 1-5, doi: 10.1109/ICFCR50903.2020.9249984.

[4] M. Ahmad, I. Ahmed and A. Adnan, "Overhead View Person Detection Using YOLO," 2019 IEEE 10th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON), 2019, pp. 0627- 0633, doi: 10.1109/UEMCON47517.2019.8992980.

[5] V. H. Roldão Reis, S. J. F. Guimarães and Z. K. Gonçalves do Patrocínio, "Dense Crowd Counting with Capsule Networks," 2020 International Conference on Systems, Signals and Image Processing (IWSSIP), 2020, pp. 267-272, doi: 10.1109/IWSSIP48289.2020.9145163.

[6] P. Zhao, K. A. Adnan, X. Lyu, S. Wei and R. O. Sinnott,"Estimating the Size of Crowds through Deep Learning," 2020 IEEE Asia-Pacific Conference on Computer Science and Data Engineering (CSDE), 2020, pp. 1-8, doi: 10.1109/CSDE50874.2020.9411377.

[7] X. Wu, Y. Zheng, H. Ye, W. Hu, J. Yang and L. He, "Adaptive Scenario Discovery for Crowd Counting," ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2019, pp. 2382-2386, doi: 10.1109/ICASSP.2019.8683744.

[8] Z. Shen, Y. Xu, B. Ni, M. Wang, J. Hu and X. Yang, "Crowd Counting via Adversarial Cross-Scale Consistency Pursuit," 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018, pp. 5245-5254, doi: 10.1109/CVPR.2018.00550.

[9] M. C. Le, M. -H. Le and M. -T. Duong, "Vision-based People Counting for Attendance Monitoring System," 2020 5th International Conference on Green Technology and Sustainable Development (GTSD), 2020, pp. 349- 352, doi: 10.1109/GTSD50082.2020.9303117.

[10] S. Gong, E. Bourennane and J. Gao, "Multi-feature Counting of Dense Crowd Image Based on Multi-column Convolutional Neural Network," 2020 5th International Conference on Computer and Communication Systems (ICCCS), 2020, pp. 215-219, doi:10.1109/ICCCS49078.2020.9118564.

[11] J. Zong, B. Huang, L. He, B. Yang and X. Cheng, "Device-Free Crowd Counting Based on the Phase Difference of Channel State Information," 2020 IEEE International Conference on Information Technology,Big Data and Artificial Intelligence (ICIBA), 2020, pp. 1343- 1347, doi: 10.1109/ICIBA50161.2020.9276804.

[12] S. Wang, R. Li, X. Lv, X. Zhang, J. Zhu and J. Dong, "People Counting Based on Head Detection and Reidentification in Overlapping Cameras System," 2018 International Conference on Security, Pattern Analysis, and Cybernetics (SPAC), 2018, pp. 47-51, doi: 10.1109/SPAC46244.2018.8965468.