International Journal of Computational Intelligence Systems
Vol. **14(1)**, 2021, *pp.* 1217–1228
DOI: https://doi.org/10.2991/ijcis.d.210326.001; ISSN: 1875-6891; eISSN: 1875-6883
https://www.atlantis-press.com/journals/ijcis/

Research Article

Multi-Scale Person Localization With Multi-Stage Deep Sequential Framework

Sultan Daud Khan[1,*], Saleh Basalamah[2]

[1]*National University of Science and Technology, Islamabad, Pakistan*
[2]*Umm Al-Qura University, Mecca, Saudi Arabia*

ARTICLE INFO

Article History
Received 22 Aug 2020
Accepted 19 Feb 2021

Keywords
Scale estimation
Deep learning
Head detection
Crowd analysis

ABSTRACT

Person detection in real videos and images is a classical research problem in computer vision. Person detection is a nontrivial problem that offers many challenges due to several nuisances that commonly observed in natural videos. Among these, scale is the main challenging problem in various object detection tasks. To solve the scale problem, we propose a framework that estimates the scales of person's heads, as we argue that head is the only visible part in complex scenes. we propose a head detection framework that explicitly handles head scales. The framework consists of two sequential networks: (1) scale estimation network (SENet) and (2) head detection network. SENet predicts the distribution of scales from the input image in the form of histogram. Then the scale histogram adjust anchor scale set of region proposal network that generates object proposals. These objects proposals are then classified into two classes, that is, head and background by the detection network. We evaluate proposed framework on three challenging benchmark datasets. Experiment results show that proposed framework achieves state-of-the-art performance.

1. INTRODUCTION

Automated face and pedestrian detection has gained much attention from research community. During recent years, several efforts have been made toward this direction and significant performance has been achieved. For understanding crowd dynamics, most of the state-of-the-art methods focus on face and pedestrian detection [1,2], the general task of localizing people in complex and unconstrained environment is still a challenging problem.

Face detector relies on facial features that cannot be extracted from occluded face or in a case when the person turns his back to camera. Similarly, pedestrian detector relies on different body part features that are not visible in complex scenes due to occlusion. On the other hand, head does not suffer from aforementioned limitations due to unique installation of camera in natural scenes. For safety and security, surveillance cameras are mounted overhead. In this unique settings, human head is the only visible and least occluded part of human body.

Head detection as a preprocessing task, has played an important role in many video surveillance applications, for example, person identification [3,4], action recognition [5,6], tracking [7], autonomous driving, behaviors understanding [8,9].

Generally, we treat head detection problem as a special case of object detection. A reliable head detector must precisely detect human heads in complex and unconstrained environment. Although, several advancements have been made in this direction, yet head detection in complex scenes is still a challenging problem. Due to cluttered background and high variations in appearances of heads, poses and scales, head detection is a challenging task. The variation in appearances and poses can be addressed by deep neural networks, however, it requires a specific strategy to solve scale problem.

The problem of scale is a major problem of every object detector [10]. Several methods reported in literature that aimed to solve the scale problem. Traditional methods [11,12] used hand-craft features to detect object at multiple scales by resizing the image several times and generate an image pyramid. This strategy increase computational complexity which limits their application in real time.

The hand-crafted features are replaced by hierarchical features computed by CNNs, provide great performance boost to object detectors. Liu *et al.* [13] trained a single detector by resizing the image using scale-aware network. Similary, Sign and Davis [14] proposed a multiple-scale detectors, where each detector independently handles specific scale of objects. Wang *et al.* [15] proposed cascaded mask network that combines multi-scale inputs for detecting multiple scale objects. These methods work fine in specific situations, however incur high computation cost and require high inference time and memory consumption.

Most recent detection methods [16–18], solve scale problem by generating object proposals with different scales and aspect ratios.

*Corresponding author. Email: sultandaud@nutech.edu.pk

These detectors utilize feature maps of the top layers to generate object proposals. These detector achieve significant performance in detecting large objects, however, performance degrades in detecting small objects. Due to large receptive field and low resolution, feature maps of top layers are more beneficial to capture semantic information but do not contain information about small objects. To detect multi-scale objects, Cai *et al.* [19] improves the resolution of feature maps by employing convolutional layers. Lin *et al.* [20] proposes Feature Pyramid Network (FPN) that produces object proposals by generating multi-scales feature maps. Cao *et al.* [21] uses single-shot detector (SSD) as backbone and adopts fusion strategy to combine multi-scale features from different layers. However, it is hard to combine features from different layers in SSD method. Fu *et al.* [22] proposed deconvolutional single-shot detector (DSSD). The method utilized skip connections and deconvolutional layers to detect multi-scale objects. The deconvolutional layers of both DSSD and FPN leverage feature maps of the top-most layer that lost details of small objects. Cui *et al.* [23] proposed the multi-scale deconvolutional singleshot detector (MDSSD) that merged top-most layer with shallow layers to achieve more semantic feature maps.

Aforementioned methods adopt different multi-scale strategies to detect multi-scale objects in natural images. However, these methods cannot explicitly determine object scales. Thus, we ask a question whether there is a "one model method" that explicitly predict the object scales in an image. In order to answer this question, we need to understand the underlying factors of scale problem. From empirical studies, we confirm that drastic perspective distortions in the images cause scale variations in the image. The perspective distortion is related to camera calibration and indicates the scale change from near to far end of image. Perspective information has been widely incorporated in many crowd counting problems. However, acquisition of perspective information is hard and requires human efforts [24].

In order to predict wide range of head scales, we propose a scale estimation network (SENet), that explicitly determine the distribution of scales in the form of histogram, namely, histogram of scales (HoS). With this prior knowledge about the scales, we can re-size the input image to best fit the detector's. Generally, proposed framework consists of two sequential stages. The first is the SENet, which estimates the distribution of scales of human heads in the input image and the second is detection, which uses the predicted scales and detects human head by re-scaling the image according to predicted scales.

Contribution. Proposed framework has the following contributions compare to other state-of-the-art methods.

1 We adopt divide and conquer strategy and divide the head detection problem into scale estimation and head detection sub-problems.

2 We propose a novel SENet that generates HoS for each input image. HoS is then utilized by region proposal network (RPN) to generate object proposals.

3 Proposed SENet is different from SharMask [25] in terms of handling scales. SharpMark search over all predefined scales and generate scale-aware object proposals while SENet estimates limited number but most appropriate scales that best describes the characteristics of input image.

4 We perform rigorous evaluation and compare proposed method with other state-of-the-art methods on three challenging benchmark datasets. Experiments results show that proposed method supersedes other state-of-the-art methods by a great margin.

The rest of paper is organized as follows: Section 2 discusses related work. We provide details of proposed methodology in Section 3. Experiment results discussed in Section 4 and Section 7 concludes the paper.

2. RELATED WORK

The task of head detection is similar to face detection. Both face detection and head detection are the specific forms of generic object detection. In other words, head detection provide an aid to face recognition by providing a bounding box in real-time applications. In this section, we review some methods on face detection, generic object detection and head detection.

2.1. *Generic Object Detection*

Most of existing methods focus on generic object detection convolutional neural network (CNN)-based methods. We categorize these models into two groups: (1) region-based frameworks and (2) region-free frameworks. Region-based are two-stage frameworks, consist of three important steps: (1) generate region candidate proposals via predefined anchor boxes; (2) extract features from each proposals; and (3) predict the class score for each proposal. Girshick *et al.* [26] proposed the first CNN-based framework for object classification. To address the shortcomings of regions with CNN (RCNN), Girshick [27] proposed Fast RCNN that improves the detection accuracy. Fast RCNN, although improves the detection performance, however, the framework still depends on separate module for generating region proposals that hampers the application of the framework in real time. Ren *et al.* proposed Faster RCNN [17] that employs RPN to efficiently compute accurate object proposals. These region-based frameworks perform well in generic object detection tasks, however, the computation of region proposals require high computation capable hardware resource and storage. Single-stage frameworks, You-Only-Look-Once (YOLO) [28] and its variants YOLOv2 [29], YOLOv3 [30], YOLOv4 [31], and SSD [32], directly computes the class probabilities and bounding boxes for each input image in a single forward pass.

The abovementioned generic object detection models perform low in head detection task. This attributes to the smaller size of human head. These frameworks utilize feature maps from the last convolution layer that contains inadequate information about the small objects.

2.2. Face Detection

Face detection and recognition is well studied topic. A considerable amount of work is reported in literature to detect faces in natural images. Inspired by the performance of CNN in generic object detection tasks, current face detection methods are also based on CNN [33–36]. To detect face in complex scenes, various

context-based methods [37–39] proposed during recent years. Since scale of face varies in natural images, Hao *et al.* [40] proposed scale-aware detector to detect faces with different scales. Hu and Ramanan [38] proposed an effective framework to detect tiny faces by contextual reasoning. Yang *et al.* [41] proposed Faceness-Net that detects faces by refining proposals using faceness score.

The above face detectors achieve high accuracy on face detection dataset, however, these models cannot perform well when applied to head detection tasks. Although, the task of face detection is similar to head detection, the size and appearance of head changes drastically with camera view point and direction. For example, there is considerable difference between the shape and appearance of front and back head. Therefore, face detectors cannot be employed to detect heads in complex scenes.

2.3. Head Detection

Earlier head detection models use hand-craft features, for example, Local binary patterns, Scale-invariant feature transform (SIFT), and Histogram of Oriented Gradients (HOG) features and learn a nonlinear classifier. One of the classical method proposed by Viola and Jones [42] employed cascade classifier using Haar-like features. Ren [43] proposed conditional random field (CRF)-based temporal model for head detection that leverages temporal information to improve the detection performance. Yan *et al.* [44] propose deformable part model that used widely adopted Histogram of oriented gradients to detect objects. A new model for head detection proposed by [45] by exploiting contextual information. They extended R-CNN-based object detector with two type of contextual cues. First, they leveraged person–scene relationship to build global model that predict positions and coarser scale of heads from the image. Second, pairwise relation among the objects were explicitly exploited to train a CNN model by using a structured-output structure loss. Later on, they combined the output of each model into a joint CNN framework.

The abovementioned models have achieved impressive performance compared to traditional methods, such as DPM [43,46]. However, these methods suffer from the following weakness in one way or another: (1) the early R-CNN rely on region proposals which were typically generated using hand-craft features [47]. (2) Most of previous methods [20,28,32,48] set anchor boxes to various sizes and aspect ratios to make use of multi-scale features extracted from different levels of deep CNN; (3) these methods do not address the multi-scale problem thoroughly. Although, multi-scale problem was addressed by [49,50] by modelling different filters to capture object of different sizes, however these methods suffers from computational cost.

3. OVERVIEW OF PROPOSED FRAMEWORK

In this section, we briefly discuss the pipeline of proposed framework. Proposed framework has two sequential parts: scale estimation and detection. The first part estimate the distribution of scales (in terms of histogram) by looking only once at the input image and detection network utilizes this information to detect human heads by looking at the input image multiple times at multiple scales according the scales estimated by SENet. The pipeline of overall framework is shown in Figure 1.

3.1. SENet: Scale Estimation Network

In this section, we discuss proposed SENet that predicts scale distribution of heads appeared in an input image. The goal of SeNet is to learn the distribution of head's sizes appear in the image. Our

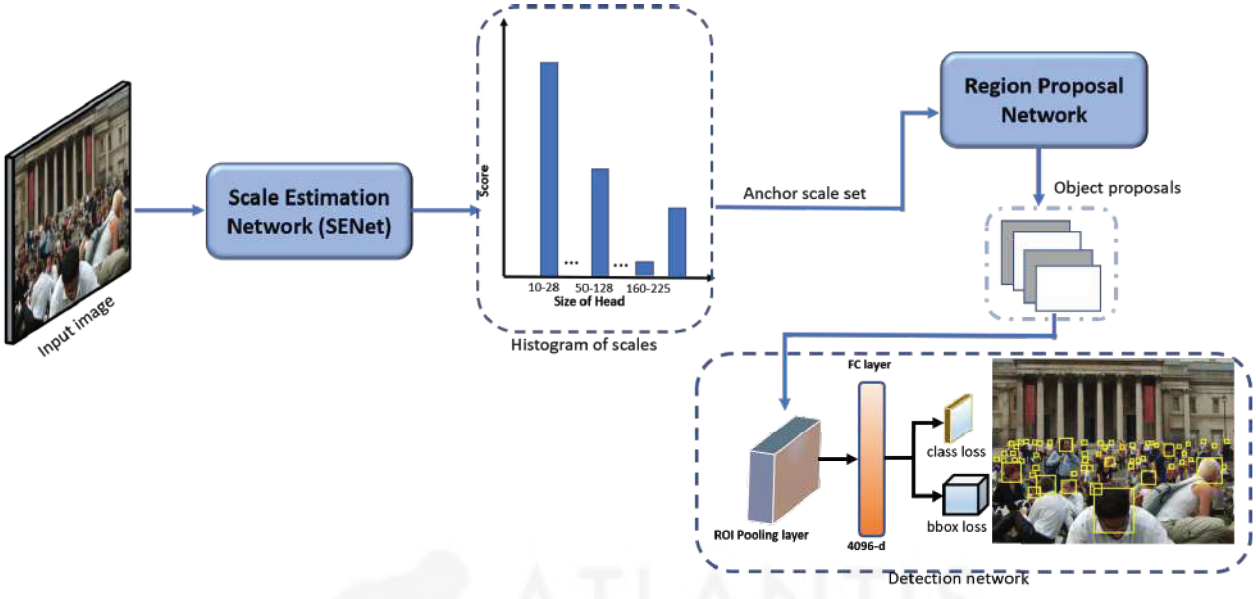

Figure 1 Pipeline of proposed framework. The input to framework is an arbitrary size image. The scale estimation network (SENet) takes the input image and outputs distribution of scales of heads (histogram of scales). The anchor scale set of region proposal network (RPN) is set according to estimated scales. RPN outputs object proposals (bounding boxes) with class probabilities. Object proposal are then applied as input to the detection network that detects heads in the input image.

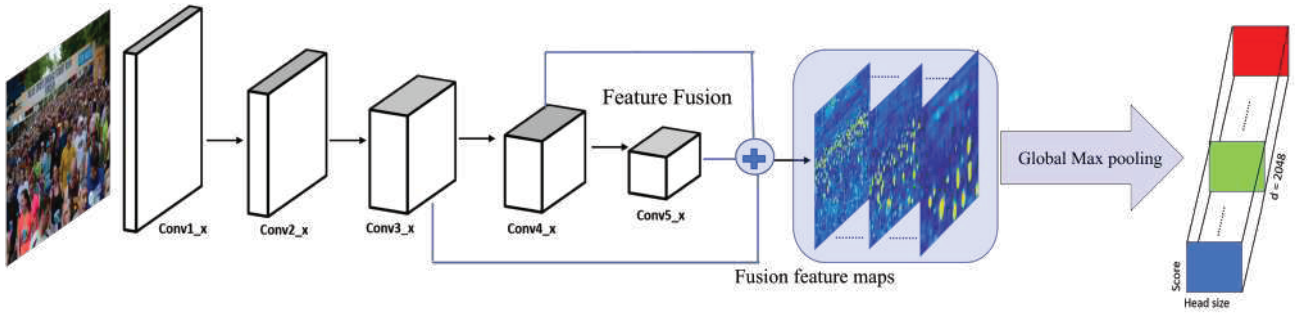

Figure 2 Scale estimation network (SENet). The input of the network is an arbitrary size image. Hierarchical features from different layers are fused and applied to global max pooling layer that outputs fixed size vector. The vector captures the distribution of scales of heads appear in the input image.

proposed SeNet is similar to scale proposal network (SPN) [40] and takes input image and outputs fixed size vector. Figure 2 illustrates proposed SENet for scale estimation. Our SeNet follows the backbone architecture of ResNet-101 [51]. ResNet-101 consists of 101 layers and overcomes the problem of vanishing gradient. The architecture of ResNet is divided into five stages. The first stage contains one convolution layer with filter size of 7×7 and stride of 2 with 64 channels followed by pooling layer with filter size of 3×3 and stride of 2. The first stage is followed four stages, namely, conv2_x, conv3_x, conv4_x and conv5_x. Each stage contains a set of three convolution layers, that is, $[\ 1 \times 1,\ 3 \times 3,\ 1 \times 1\ ]$. In RestNet-101, conv2_x contains 3 sets with total of 9 layers. Similarly, conv3_x contains 4 sets that makes 12 layers, conv4_x contains $3 \times 23 = 64$ layers, and last stage conv5_x contains $3 \times 3 = 9$ layers. The resolution of feature map is reduced by half after passing through each stage. However, the resolution of feature map is same within stage.

Due to its unique architecture, ResNet has enjoyed success in image classification and object detection tasks [52–54]. Object detection methods usually utilize the last convolution layer (conv5_x) for generating region proposals. Since the receptive field of last covolutional layer is large, therefore, it contains more information about large objects than small objects. Therefore, these detectors are not suitable to detect small objects.

As discussed in Section 1, human heads in natural images lie in wide range of scales. Moreover, the size of heads is relatively small compared to other objects in natural image. According to definition in [55], objects with size smaller than 32×32 pixels are considered as small objects. Current state-of-the-art *scale-invariant* [17,56–58] and scale-variant [32,38,59] models cannot handle such wide range of scales and small sizes of human heads.

To detect small human heads and capture wide range of scale variations, unlike other methods, that use features from the last convolutional layer for object detection, we adopt fusion strategy of integrating features from both shallow and top layers of the network. It is well established fact in [21,60,61] that shallow layers, with small receptive field size, capture fine grained details, while top layers capture the context due to large receptive fields.

Proposed fusion strategy is illustrated in Figure 3. As shown in Figure, we use conv3_3, conv4_3, and conv5_3 layers during fusion process. Shallow layer, that is, conv3_3 is assumed to contain more information about the small objects, while top layers, that is, conv4_3, and conv5_3 contain features of large objects. The feature

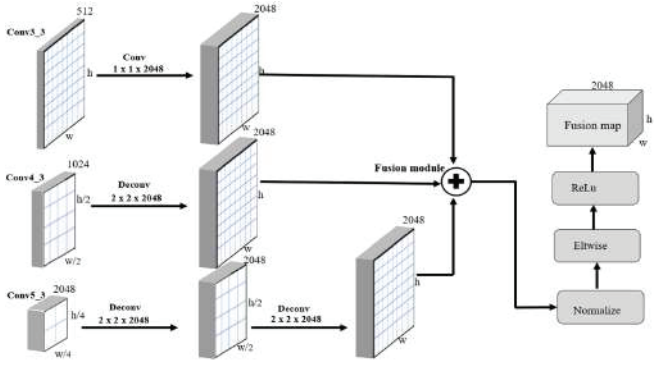

Figure 3 Generation of fused feature map by fusing feature maps from multiple convolutional layers.

maps of these layers are different in sizes and number of channels. The resolution of feature map of top layer (conv5_x) is half of the size of feature map of conv4_3 layer. Similary, the size of feature map of conv4_3 layer is half of feature map of conv3_3 layer. In order to make these features maps suitable for fusion, we up-sample feature maps of conv4_3 and conv5_3 to match the size of feature maps of conv3_3 layer. We use deconvolution layer to up-sample the feature maps. In order to make conv4_3 and conv5_3 layer equal to the size of conv3_3, we apply one 2×2 deconvolution with 2048 channels to conv4_3 layer and two 2×2 decovolution layers (one after the other) to conv5_3 layer. Moreover, we employ 1×1 convolution layer to conv3_3 layer with 2048 channels to match channel dimension. We use 1×1 convolution layer to suppress aliasing and generate final fused feature map.

The network then provides final fusion map as input to global max pooling layer. This pooling layer sets the pool size equal to the size of fusion map and computes maximum value for each input channel of fusion map. The output of global max pooling layer is a vector $N = \{s_1, s_2, ..., s_n\}$ of size n, where each element $s_i \in N$ represents the probability of having certain size of head in image. It is to be noted that $n = 2048$ equal to the number of channels in fusion map.

For capturing the local statistics of feature map and scale distribution of human heads, we utilize vector N and generate a HoS denoted as H. Let S_h represents the size of head and C_h represents the confidence of head. We define the size of head S_h as the length or width of square bounding box. We then sort N based on sizes of

heads, where S_1 represents the smallest head and S_n represents the largest head.

Each bin b_i of histogram represents the cumulative confidence score of having head of sizes in the image. In other words, for each i_{th} bin, we accumulate all confidence scores related to head sizes that fall within a certain range. The horizontal axis of the histogram represents head size and vertical axis shows the cumulative score.

We then compute histogram $H = \{b_1, b_2, ...b_n\}$, where each bin b_i is computed as follows:

$$b_i = \sum_{h=1}^{N} C_h, \quad \forall S_h \mid 2^{s_1+(i-1)k} \leqslant S_h < 2^{s_1+ik}$$

where k represents the bin's width in logarithmic scale of base 2 and is defined as $k = \frac{s_n - s_1}{n}$.

With the integration of histogram layer, the network essentially becomes a score accumulator and which avoids location information and accumulates confidence values from all locations. With the avoidance of location information, the network become more responsive to features that best describe the human head, even if the size of head is smaller or larger than the receptive filed of the network.

During training, we formulate the problem of estimating scale as a minimizing Kullback–Leibler divergence. Let $\hat{H} = \left\{\widehat{h_1}, \widehat{h_2}, \ldots, \widehat{h_n}\right\}$ is the ground truth histogram. Then the cross entropy loss is calculated as

$$D_{KL}(\hat{H}, H) = \sum_{1 \leqslant i \leqslant n} \widehat{h_i} \cdot (\log \widehat{h_i} - (\log h_i) \tag{1}$$

3.2. Optimum Scale Selection

The estimated histogram H is of high resolution and describes the statistics of image by covering all possible scales. However, prediction using all estimated scales is computationally expensive. Therefore, we reduce the resolution of histogram by selecting optimal scales that is utilized by object detector to precisely locate the heads in image. We first smooth the histogram by using moving average filter. With this step, high frequency components and noise are removed. For finding the optimal scales, we employ non-maxima suppression (NMS) based on threshold value that finds peaks in histogram. The peaks of the histogram represent head size with it corresponding score. We find threshold value during validation process.

After employing NMS, we achieve optimum scales that can be utilized by object detector by resizing the image accordingly. NMS is based on fixed threshold that does not guarantee optimum number of scales. Changing the threshold value changes the results, therefore, finding best optimization strategy is a potential research direction. However, from empirical evidences, we observed that such sub-optimal strategy still achieved high precision and recall rate.

3.3. Detection Network

After achieving optimum scales, the next step is to employ detection network to perform detection by rescaling the image multiple times according to predicted scales. We employ RPN as head detector, however any detector can be used. There are couple of reasons for selecting RPN. (1) RPN serve as best competitor for two-class object detectors. (2) Most of existing object detectors [17,62] employ RPN for generating object proposals.

The RPN takes an arbitrary size image and provide a set of bounding boxes (called region proposals) as output. RPN is introduced by Ren *et al.* [17] and in original settings, RPN utilizes feature map of the last convolutional layer (5_{th}) for object proposals generation. RPN uses pre-defined set of scales (anchor scale set) to generates several object proposals with different sizes and aspect ratios. RPN predicts class probabilities and bounding boxes for different objects by utilizing feature map (from last convolutional layer) and object proposals. In original settings, RPN uses anchor scale set that consists of three predefined scales $\{128, 256, 512\}$. Although with these predefined scales, RPN easily detects large objects, yet misses small objects in natural images and videos. Since, we are interested in detecting human head in natural images, the size of which is usually very small (15–20 pixels). Therefore, detectors with predefined scales get trouble in detecting small heads.

To detect small objects, the intuitive way is to include small scales to the anchor set. Although this will improve the performance of a detector in detecting in small object, however, the accuracy will not be sufficient. Therefore, instead of using predefined scales, we utilize the scales predicted by proposed SENet as discussed in section 3.1.

For RPN training, we use output fusion map that consists of w × h locations. For each location, we generate n anchors with aspect ratio of 1:1. Here n represents the number of scales predicted by SENet. Since we are detecting human heads, therefore we keep square like aspect ratio. We assign each anchor a binary label to represent whether the anchor belongs to the head or background. We assign positive value to the anchor belongs to head. We compute intersection-over-union (IoU) between the ground truth and anchor, and assign positive value if IoU is above than 0.7. It is to be noted that we assign positive value to all of those anchors that achieves higher overlap with the same ground truth bounding box. Usually we delete the anchors that exceed the boundary of image and do not contribute to the loss. We assign negative value to anchors if IoU is less than 0.3. We generate positive and negative anchors in a mini-batch generated from a single image. We observed that the number of negative anchors are large than positive anchors This will bias the loss function more toward negative. To address this problem, we compute the loss of mini-batch by randomly choosing 256 anchors with sample ratio of 1:1 from a single image. In case the number of samples from positive class is less than 128 samples, we use negative samples complete the batch of 256 anchors. We minimize the loss function for each anchor belongs to mini-batch in the following way same in [17].

$$L(c_i, b_i) = \frac{1}{M_{cls}} \sum_{i=1}^{K} L_c(c_i, \widehat{c_i}) + \omega \frac{1}{M_{reg}} \sum_{i=1}^{K} L_b(b_i, \widehat{b_i}) \tag{2}$$

where M represents the number of samples in mini-batch. i is the index of an anchor and c_i is the predicted class probability of anchor i, $\widehat{c_i}$ is the ground truth and value is either 1 or zero. $\widehat{c_i} = 1$ represents the positive class and 0 represents the negative class or background. L_c is the log class loss and L_b is log regression loss. b_i is predicted bounding box that has four parameters of location, that is, [x, y, w, h], where x and y are the horizontal and vertical co-ordinates of bounding box and w and h represent the width and height of bounding box. Similarly, $\widehat{b_i}$ represents ground truth bounding box generated manually. These two terms in Equation (2) are normalized by M_{cls} and M_{reg} respectively, and ω is a balancing parameter.

Xavier initialization [63] is employed to initialize layers of the network, with learning rate of 0.001 which decreases by rate of 10 after every 10k iterations.

4. EXPERIMENT RESULTS AND ANALYSIS

In this section, we briefly discuss the datasets used for evaluating and comparing different state-of-the-art head detection models. We use three publicly available benchmark datasets: HollywoodHeads [45], Casablanca [43], and SCUT-HEAD [64].

HollywoodHeads dataset is considered as largest dataset that contains 224,740 images sampled from 21 different Hollywood movies. The dataset provides 369,846 annotations (bounding boxes) that covers the human heads of different scales, orientations, and appearances. The dataset is divided into three sets. The set part provides 216,719 images for training the models that covers 15 different movies. The second set provides 6,719 images from 3 movies for validation and third set contains the remaining 1,302 images from the remaining 3 movies for testing. The dataset is publicly available and can be downloaded from the following link: *https://www.di.ens.fr/willow/research/headdetection/*

SCUT-HEAD is recently proposed by Peng *et al.* [64]. The dataset is divided into two parts: *PartA* and *PartB*. *PartA* consists of 2000 images samples from a zoom-out camera installed in the corner of a classroom of the university. *PartA* contains 67,321 annotations of human heads. Usually, inside the classroom, human heads have similar poses and orientations, therefore, the images are carefully chosen to minimize the similarity and gain the variance among the images. The density of people in *PartA* varies from 0 to 90 people per image with average count of 51.8. On the other hand, *PartB* contains 2,405 images that covers different indoor scenes and collected from various sources over the internet. The dataset provides 43,930 annotations. The scenes in *PartB* covers relatively low dense scenes where the density varies from 20 to 70 people per image. The dataset can be downloaded from the following link: *https://github.com/HCIILAB/SCUT-HEAD-Dataset-Release*

Casablanca dataset is first proposed in [43] to evaluate head detection model. This dataset consists of 1,47,000 frames sampled from an old Hollywood movie "Casablanca" released in 1942. The dataset consists of monochromic images with the resolution of 464 × 640 pixels. The dataset is very challenging due to cluttered background, significant variations in scales, poses and appearances of human heads. The dataset can be downloaded from the following link: *https://www.di.ens.fr/willow/research/headdetection/*.

Evaluation process: To quantitatively evaluate the performance of different methods, we use mean Average Precision (mAP), a widely adopted evaluation metric for object detection tasks. Generally, the value mAP is computed from Precision-Recall curve that shows the performance of a model over a fixed value of IoU. We keep the value of IoU = 0.5 in all experiments. From empirical evidences, we observed that using fixed threshold value cannot provide conclusive information about the performance of models. Therefore, to further verify the performance and evaluate location bias, we also use IoU = 0.7 similar to [65]. In short, we use two evaluation metrics, that is, mAP and Precision-Recall curves to evaluate the performance of different head detection models.

We now evaluate and compare our method with other reference methods on Casablanca, HollywoodHeads and SCUT-HEAD datasets. For comparison, we use eight most related methods, namely, DPM-Head [66], Context-CNN [45], FCHD [67], Faster-RCNN [17], SSD [32], VJ-CRF [43], Reinspect [68], and YOLO9000 [29]. The performance of these methods on HollywoodHeads, Casablanca, and SCUT-HEADS dataset is reported in Tables 1–3, respectively.

From experiment results, we observe VJ-CRF [43] achieves lower performance on all benchmark datasets. VJ-CRF method employs traditional Viola-Jones algorithm that uses Haar like features for object detection. These features are very sensitive to illumination and affected by different poses of human heads and faces. Furthermore, haar-like features are calculated from input image in a sliding windows fashion with a fixed size. This strategy misses small objects and accumulates many false positives over the location of an object that further decreases the performance. DPM-Head [66] also

Table 1 Performance evaluation of different methods on HollywoodHeads dataset.

Methods	mAP@0.5	mAP@0.7
DPM-Head [66]	0.62	0.31
Context-CNN [45]	0.75	0.42
FCHD [67]	0.64	0.34
Reinspect [68]	0.79	0.49
Faster-RCNN [17]	0.76	0.43
SSD [32]	0.43	0.29
VJ-CRF [43]	0.37	0.21
YOLO9000 [29]	0.82	0.57
Proposed	0.86	0.63

Table 2 Performance evaluation of different methods on Casablanca dataset.

Methods	mAP@0.5	mAP@0.7
DPM-Head [66]	0.54	0.34
Context-CNN [45]	0.67	0.54
FCHD [67]	0.62	0.46
Reinspect [68]	0.69	0.57
Faster-RCNN [17]	0.48	0.28
SSD [32]	0.38	0.21
VJ-CRF [43]	0.29	0.15
YOLO9000 [29]	0.71	0.56
Proposed	0.75	0.67

shows lower performance compare to other state-of-the-art methods on all datasets as obvious from Tables 1–3. However, as shown in Table 2, the performance of DPM-Head on Casablanca dataset is lower than its performance on other datasets. This reduce performance of DPM-Head attributes to the smaller size of the head. Moreover, human heads in Casablanca dataset lie in wide range of scales that becomes challenging for DPM-Head to precisely localize heads. FCHD [67] employs fully covolutional network (FCN) for head detection. The image is resized to a fixed size, that is, 224×224 before feeding into the network. Since the size of natural images is larger than 224×224, resizing the large image into smaller size may cause loss of information about the small objects.

From Table 2, it is obvious that FCHD performs lower on Casablanca dataset compare to its performance on other datasets. Context-CNN [45] utilizes R-CNN [26] and exploits rich context of the scene to detect human heads. R-CNN uses Selective Search method for generating object proposals. These proposal are later on resized and feed into the network for classification. Since Selective Search strategy is based on greedy method and avoids learning process, therefore the method cannot generate high-quality object proposals that cover the wide range of scales of objects. As shown in Tables 1 and 2, SSD [32] suffers setback on both Casablanca and HollywoodHeads dataset, however, it shows improved performance on SCUT-HEAD dataset as obvious in Table 3. Its reduce performance attributes to the way it deals the scale problem. SSD utilizes shallow layers to detect small objects, however, shallow layers have lower discriminating power and do not contain contextual information. Faster-RCNN [17] achieves comparable performance on HollywoodHeads and SCUT-HEAD dataset, however, the performance decreases on Casablanca dataset.

This is due to the fact that Faster-RCNN utilizes the feature map of the last convolutional layer for object detection. The resolution of feature map of the last convolutional layer reduces and losses information about the small objects. On the other hand, YOLO9000 [29] achieves comparable results on all three datasets. We observed that YOLO9000 faces difficulty in detecting small objects. This is due to reason that the model down sample the feature (13×13) map of the last convolutional layer down sampled by 32. We argue that down sampling low level features will lose the information about small objects. To conserve information about the small as well as large object, proposed method adopts a feature fusion strategy, where the feature map of high level layers are up-sampled and then combined with low level feature maps. We also report the performance in terms of precision-recall curves on all datasets in Figure 4. From the experiments results, we observe that by adopting such feature fusion strategy, proposed framework beats other state-of-the-art methods by a significant margin.

Generally, the performance of state-of-the-art-methods on SCUT-HEAD dataset is relatively higher than on Casablanca and HollywoodHeads datasets. It may attribute to larger sizes (average size $\geq$ 80 pixels) of human heads in SCUT-HEAD dataset and human heads have limited range of scales. Therefore, it is not challenging even for a single and fixed scale detector to precisely detect human heads in SCUT-HEAD dataset. As obvious from Table 1, state-of-the-art-methods achieve high performance on HollywoodHeads dataset. This is due to reason that the dataset covers less crowded scenes with limited occlusion. On the other hand, state-of-the-art detectors perform low on Casablanca dataset as obvious from Table 2. Casalanca dataset is challenging due to monochromic images, clutter background, wide range of scale of heads, and extremely small size of heads.

We also report qualitative results of different methods on all three dataset in Figures 5–7. Figure 5 shows qualitative results of top four methods on HollywoodHeads datasets on samples of two different scenes. From the Figure, it is obvious that proposed method precisely detects heads in both scenes. On the other hand, Faster-RCNN localizes human heads in both scenes, but accumulates multiple positive bounding boxes around the heads that results in lower precision and recall rates. Reinspect [68] produce comparable results in both scenes, however, it accumulate false positives that results in lower performance. Yolo9000 [29] performs well in the first scene (first row of Figure 5), but accumulates multiple false positives in the second scene.

In Figure 6, we report qualitative results of different methods on sample frames from two different scenes of Casablanca dataset. From the Figure, it is obvious that both scenes are challenging for head detection task. In both scenes, human heads lie in wide range of scales, that is, the size of near head is large while the size of far heads is small. Moreover, with clutter background in both scenes, small size heads are not visible. We compare the results of top four methods. In the both scenes, Context-CNN [45] detects two heads that are clearly visible while accumulates many false positives. Similarly, Reinspect [68] detect two heads in the first scene with no false positive while in second scene, it accumulates many false positives. Yolo9000 [29] on the other hand, achieve comparable performance by detecting heads in both scenes, however, it faces challenges in detecting small heads. On contrary, our proposed method with explicit scales (determined by SANet), achieves better performance by detecting heads with significant scale variations. Moreover, proposed method also able to detect small heads in both scenes. However, proposed methods could not detect occluded heads and heads in regions with clutter background.

Figure 7 illustrates the performance of different methods on SCUT-HEAD dataset. The first row in Figure 7 shows the results of top four methods on one of the scene from Part A while the second row shows the results on a sample from Part B. From the Figure, it is obvious that Faster-RCNN [17] accumulates multiple positive detections around a single head. Reinspect [68] produces better results than Faster-RCNN by detecting only one bounding box for each person. However, Reinspect could not detect small heads at the rear end of the image in the second scene. Yolo9000 [29], on the other hand, performs well than Faster-RCNN and Reinspect by detecting small heads at the rear end of the second scene as well as overlays unique bounding box for each person in both scenes. However, Yolo9000 accumulates false positives in both cases. From the qualitative results in Figure, it is is obvious that our proposed method out performs other state-of-the-art methods by precisely detecting all persons in both scenes.

5. ABLATION STUDY

In this section, we perform ablation study to understand the effect of different layers on detection performance. Moreover, we also

Table 3 Performance evaluation of different methods on SCUT-HEAD dataset.

Methods	Part A		Part B	
	mAP@0.5	mAP@0.7	mAP@0.5	mAP@0.7
DPM-Head [66]	0.75	0.43	0.78	0.47
Context-CNN [45]	0.84	0.54	0.86	0.53
FCHD [67]	0.78	0.49	0.81	0.51
Reinspect [68]	0.87	0.65	0.89	0.68
Faster-RCNN [17]	0.86	0.59	0.87	0.61
SSD [32]	0.85	0.63	0.80	0.64
VJ-CRF [43]	0.62	0.37	0.68	0.42
YOLO9000 [29]	0.91	0.75	0.92	0.74
Proposed	0.92	0.78	0.94	0.77

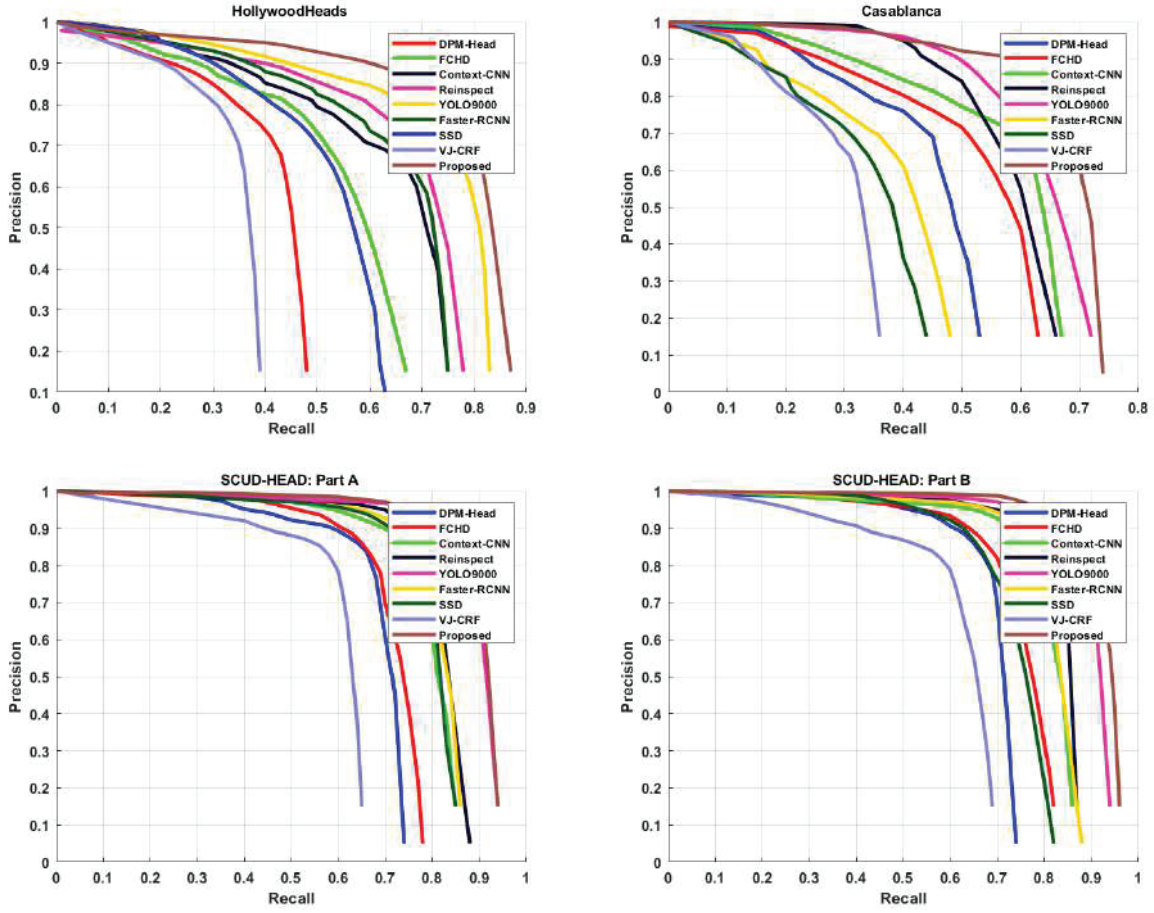

Figure 4 Performance comparison (in terms of precision-recall curves) of different methods using different datasets.

understand the effect of feature fusion of various layers. For ablation study, we choose Casablanca dataset, since this dataset is challenging due to significant variation in head scales, smaller head size, clutter background, and occlusion. We train and evaluate each of the following configurations:

1 *Basic model1*: In this configuration, ResNet-101 is used as

2 *Basic model2*: ResNet with

3 *Basic model3*: ResNet with

4 *Fusion model1*: ResNet with fusion of feature maps of Conv_5x and

5 *Fusion model2*: ResNet with fusion of feature maps of Conv_5x and

6 *Fusion model3*: ResNet with fusion of feature maps of Conv_4x and Conv_3x layers.

7 *Fusion model4 (proposed)*: ResNet with fusion of feature maps of Conv_5x, Conv_4x and Conv_3x layers.

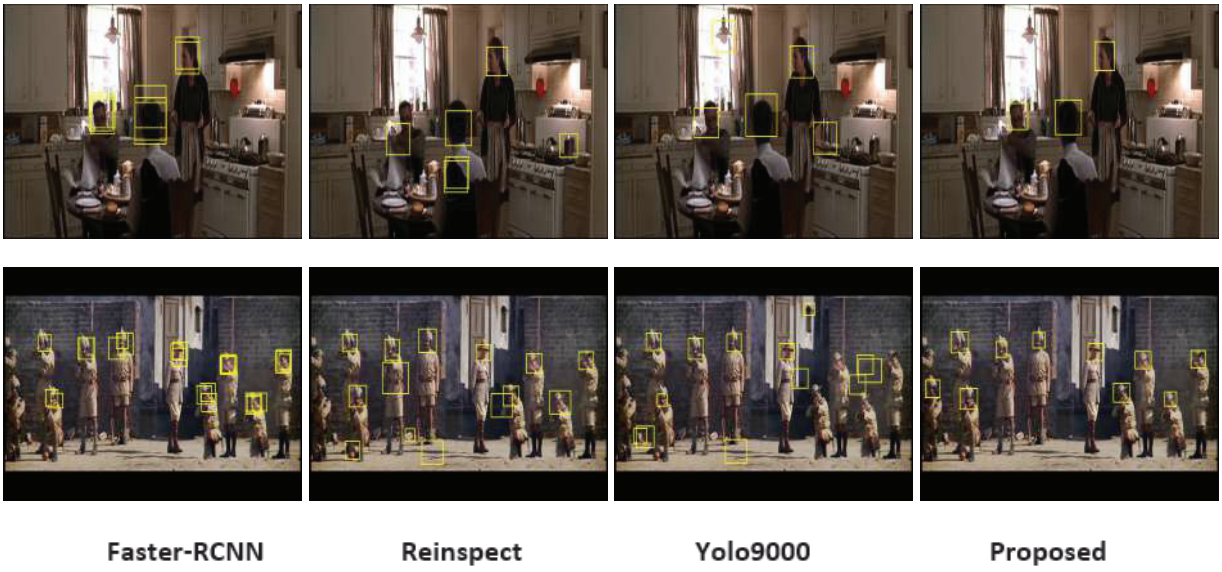

Figure 5 Performance of different methods on two sample images of HollywoodHeads dataset.

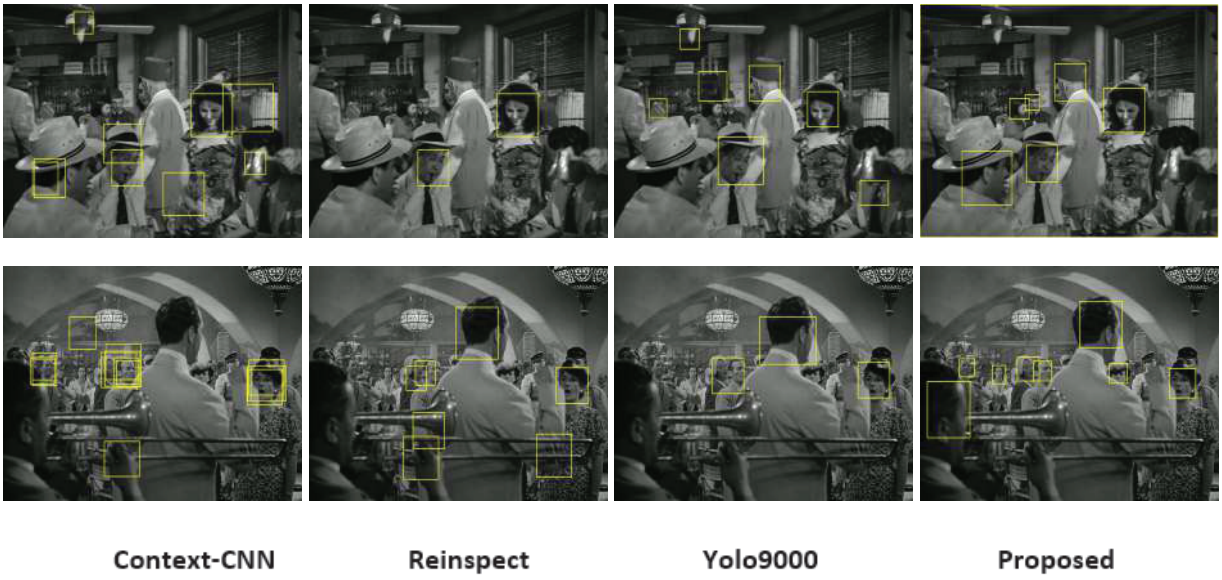

Figure 6 Performance of different methods on two sample images of Casablanca dataset.

In the first three configurations, we do not adopt any fusion strategy, while in remaining configurations we adopt different fusion strategies by combining different layers. The quantitative results of different configurations are reported in Table 4. As it is obvious from Table, all three basic models could not yield good results compare to fusion models. This is due to reason that *Basic model1* uses last convolutional layer. The receptive field of the last convolutional layer is large and suitable for detecting large objects and helpful in extracting contextual information. However, this configuration could not detect heads of small size. *Basic model3* uses Conv_3x layer that has small receptive field, which is suitable for detecting small objects but accumulates much more background noise. However, the performance improves considerably, when feature maps from multiple layers are fused.

From this study, we observe that feature fusion from multiple layers achieves detection performance beyond other state-of-the-art methods that do not exploit complementary relationship of different layers.

6. FAILURE SCENARIOS

In this section, we discuss the failure cases caused by the crowded and challenging nature of the analyzed datasets. From experiments, we found that there are two cases, where proposed framework fail to detect humans. The first scenario is related to the low resolution of images. In low-resolution crowded images, humans are blurred and do not have sharp boundaries/edges that causes sever clutter in the scene. Due to this reason, the proposed framework fails

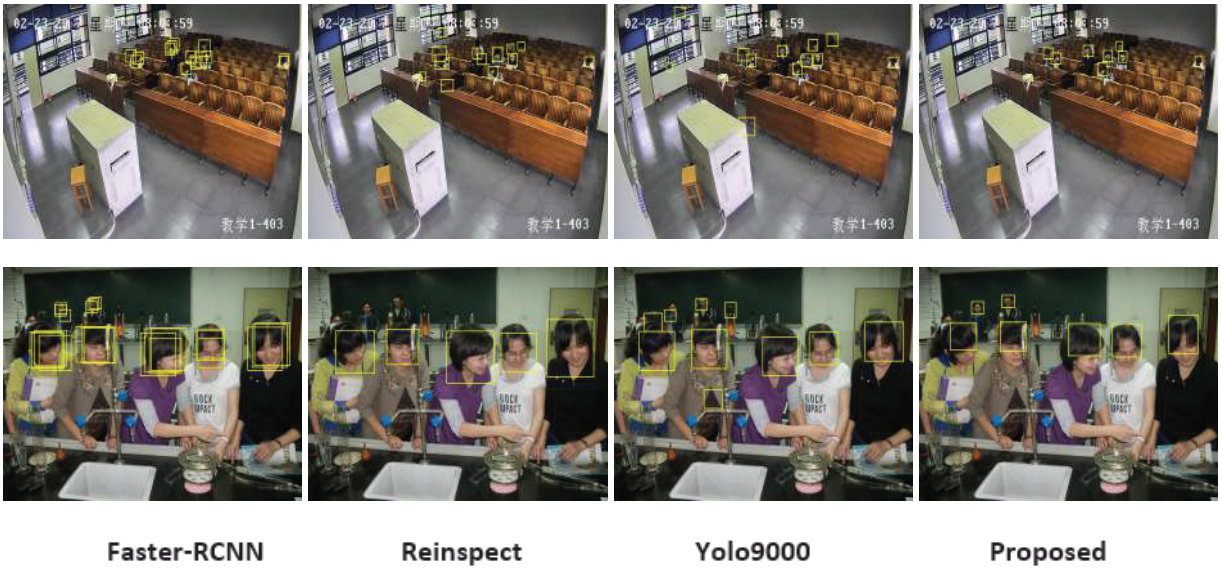

Figure 7 | Performance of different methods on SCUT-HEAD dataset. The first row represents the performance of different methods on a sample of a scene from Part A while the second row shows the performance of different methods on a sample of scene from Part B.

to distinguish between human and background. The second scenario is related to high-density crowds. In high-density crowds, the size of head is extremely small and occluded and proposed framework fails to detect human heads in these situations. It is to be noted that in high-density crowds, the humans are standing close to each other in a constrained environment and position of camera relative to humans causes occlusions. Due to partial and full occlusions, human heads are not visible and unable to be detected. We observe that in both these situations, images are hard to be annotated by humans.

7. CONCLUSION

In this paper, we propose a framework that detects human heads in challenging scenes. The framework consists of two sequential networks: (1) SENet and (2) head detection network. The SENet takes the input image and predicts the distribution of scales of all heads in the input image. The anchor scale set of RPN is then adjusted according to the predicted scales. The RPN generates object proposals that are later on classified by a detection network. We evaluate and compare proposed method with different state-of- the-art methods on different challenging benchmark datasets. The experiment results show the effectiveness of proposed framework. Since our proposed method is designed to generate object proposals, therefore, this framework can also be adopted for generic object detection tasks.

Table 4 | Results of Ablation study with different configurations on Casablanca dataset.

Configurations	mAP
Basic model 1	0.58
Basic model 2	0.54
Basic Model 3	0.57
Fusion model 1	0.71
Fusion model 2	0.68
Fusion model 3	0.69
Fusion model 4	0.75

CONFLICTS OF INTEREST

The authors have no conflict of interest.

AUTHORS' CONTRIBUTIONS

The authors confirm contribution to the paper as follows: study conception and design: S.D.K; data collection: S.D.K; analysis and interpretation of results: S.D.K & S.B; draft manuscript preparation: S.D.K & S.B. All authors reviewed the manuscript and agreed to the published version of the manuscript.

ACKNOWLEDGMENTS

This research work is jointly supported by Umm Al-Qura University, Saudi Arabia and National University of Science and Technology, Pakistan.

REFERENCES

[1] R. Ranjan,V.M. Patel, R. Chellappa, Hyperface: adeep multi-task learning framework for face detection, landmark localization, pose estimation, and gender recognition, IEEE Trans. Pattern Anal. Mach. Intell. 41 (2019), 121–135.

[2] X. Wang, T. Xiao, Y. Jiang, S. Shao, J. Sunand, C. Shen, Repulsion loss: detecting pedestrians in a crowd, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 2018, pp. 7774–7783.

[3] L. Zheng, Y. Huang, H. Lu, Y. Yang, Pose-invariant embedding for deep person re-identification, IEEE Trans. Image Process. 28 (2019), pp. 4500–4509.
[4] H. Yao, S. Zhang, R. Hong, Y. Zhang, C. Xu, Q. Tian, Deep representation learning with part loss for person reidentification, IEEE Trans. Image Process. 28 (2019), 2860–2871.
[5] K. Simonyan, A. Zisserman, Twostream convolutional networks for action recognition in videos, in Advances in Neural Information processing systems, Montreal, Quebec, Canada, 2014, pp. 568–576.
[6] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, L. Fei-Fei, Large-scale video classification with convolutional neural networks, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 2014, pp. 1725–1732.
[7] Y. Tian, A. Dehghan, M. Shah, On detection, data association and segmentation for multi-target tracking, IEEE Trans. Pattern Anal. Mach. Intell. 41 (2018), 2146–2160.
[8] S.D. Khan, S. Bandini, S. Basalamah, G. Vizzari, Analyzing crowd behavior in naturalistic conditions: Identifying sources and sinks and characterizing main flows, Neurocomputing. 177 (2016), 543–563.
[9] S. Khan, G. Vizzari, S. Bandini, S. Basalamah, Detecting Dominant Motion Flows and People Counting in High Density Crowds. J. WSCG 22 (2014), pp. 21–30.
[10] P. Zhou, B. Ni, C. Geng, J. Hu, Y. Xu, Scale-transferrable object detection, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 2018, pp. 528–537.
[11] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), IEEE, San Diego, CA, USA, 2005, vol. 1, pp. 886–893.
[12] D.G. Lowe, Distinctive image features from scaleinvariant keypoints, Int. J. Comput. Vision. 60 (2004), 91–110.
[13] Y. Liu, H. Li, J. Yan, F. Wei, X. Wang, X. Tang, Recurrent scale approximation for object detection in CNN, in Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 2017, pp. 571–579.
[14] B. Singh, L.S. Davis, An analysis of scale invariance in object detection snip, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 2018, pp. 3578–3587.
[15] G. Wang, Z. Xiong, D. Liu, C. Luo, Cascade mask generation framework for fast small object detection, in 2018 IEEE International Conference on Multimedia and Expo (ICME), IEEE, San Diego, CA, USA, 2018, pp. 1–6.
[16] X. Wang, A. Shrivastava, A. Gupta, A-fast-RCNN: hard positive generation via adversary for object detection, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 2017, pp. 2606–2615.
[17] S. Ren, K. He, R. Girshick, J. Sun, Faster R-CNN: towards real-time object detection with region proposal networks, in Advances in Neura Information Processing Systems, Montreal, Quebec, Canada, 2015, pp. 91–99.
[18] P. Purkait, C. Zhao, C. Zach, SPP-net: deep absolute pose regression with synthetic views, arXiv preprint arXiv: 1712.03452, 2017.
[19] Z. Cai, Q. Fan, R.S. Feris, N. Vasconcelos, A unified multi-scale deep convolutional neural network for fast object detection, in European Conference on Computer Vision, Amsterdam, The Netherlands, 2016, pp. 354–370.
[20] T.-Y. Lin, Feature pyramid networks for object detection, in 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 2017, vol. 7, p. 4.
[21] G. Cao, X. Xie, W. Yang, Q. Liao, G. Shi, J. Wu, Feature-fused SSD: fast detection for small objects, in Ninth International Conference on Graphic and Image Processing (ICGIP 2017), International Society for Optics and Photonics, Qingdao, China, 2018, p. 106151E.
[22] C.-Y. Fu, W. Liu, A. Ranga, A. Tyagi, A.C. Berg, DSSD: deconvolutional single shot detector, arXiv preprint arXiv:1701.06659, 2017.
[23] L. Cui, R. Ma, P. Lv, X. Jiang, Z. Gao, B. Zhou, M. Xu, MDSSD: multiscale deconvolutional single shot detector for small objects, arXiv preprint arXiv: 1805.07009, 2018.
[24] M. Shi, Z. Yang, C. Xu, Q. Chen, Revisiting perspective information for efficient crowd counting, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 2019, pp. 7279–7288.
[25] P.O. Pinheiro, T.-Y. Lin, R. Collobert, P. Dollár, Learning to refine object segments, in European Conference on Computer Vision, Amsterdam, The Netherlands, 2016, pp. 75–91.
[26] R. Girshick, J. Donahue, T. Darrell, J. Malik, Rich feature hierarchies for accurate object detection and semantic segmentation, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 2014, pp. 580–587.
[27] R. Girshick, Fast R-CNN, in Proceedings of the IEE International Conference on Computer Vision, Santiago, Chile, 2015, pp. 1440–1448.
[28] J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You only look once: unified, real-time object detection, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 2016, pp. 779–788.
[29] J. Redmon, A. Farhadi, Yolo9000: better, faster, stronger, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 2017, pp. 7263–7271.
[30] J. Redmon, A. Farhadi, Yolov3: an incremental improvement, arXiv preprint arXiv: 1804-02767, 2018.
[31] A. Bochkovskiy, C.-Y. Wang, H.-Y. Mark Liao, Yolov4: optimal speed and accuracy of object detection, arXiv preprint arXiv: 2004.10934, 2020.
[32] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, A.C. Berg, SSD: single shot multibox detector, SSD: single shot multibox detector in European Conference on Computer Vision, Amsterdam, The Netherlands, 2016, pp. 21–37.
[33] H. Li, Z. Lin, X. Shen, J. Brandt, G. Hua, A convolutional neural network cascade for face detection, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 2015, pp. 5325–5334.
[34] B. Yang, J. Yan, Z. Lei, S.Z. Li, Convolutional channel features, in Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 2015, pp. 82–90.
[35] S. Yang, P. Luo, From facial parts responses to face detection: a deep learning approach, in Proceedings of the IEEE International Conference On Computer Vision, Santiago, Chile, 2015, pp. 3676–3684.

[36] S. Zhang, C. Chi, Z. Lei, S.Z. Li, Refineface: refinement neural network for high performance face detection, IEEE Trans. Pattern Anal. Mach. Intell. (2020).
[37] K. Zhang, Z. Zhang, H. Wang, Z. Li, Y. Qiao, W. Liu, Detecting faces using inside cascaded contextual CNN, in Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 2017, pp. 3171–3179.
[38] P. Hu, D. Ramanan, Finding tiny faces, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 2017, pp. 951–959.
[39] C. Zhu, Y. Zheng, K. Luu, M. Savvides, CMS-RCNN: contextual multi-scale regionbased CNN for unconstrained face detection, in: B. Bhanu, A. Kumar (Eds.), Deep Learning for Biometrics, Springer, Cham, Switzerland, 2017, pp. 57–79.
[40] Z. Hao, Y. Liu, H. Qin, J. Yan, X. Li, X. Hu, Scale-aware face detection, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, Hawaii, 2017, pp. 6186–6195.
[41] S. Yang, P. Luo, C.C. Loy, X. Tang, Faceness-net: Face detection through deep facial part responses, IEEE Trans. Pattern Anal. Mach. Intell. 40 (2017), 1845–1859.
[42] P. Viola, M.J. Jones, Robust real-time face detection, Int. J. Comput. Vision. 57 (2004), 137–154.
[43] X. Ren, Finding people in archive films through tracking, in 2008 IEEE Conference on Computer Vision and Pattern Recognition, IEEE, Anchorage, AK, USA, 2008, pp. 1–8.
[44] J. Yan, Z. Lei, L. Wen, The fastest deformable part model for object detection, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 2014, pp. 2497–2504.
[45] T.-H. Vu, A. Osokin, Context-aware CNNs for person head detection, in Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 2015, pp. 2893–2901.
[46] P.F. Felzenszwalb, R.B. Girshick, D. McAllester, D. Ramanan, Object detection with discriminatively trained part-based models, IEEE Trans. Pattern Anal. Mach. Intell. 32 2010.1627–1645.
[47] J.R.R. Uijlings, K.E.A. Van De Sande, T. Gevers, A.W.M. Smeulders, Selective search for object recognition, Int. J. Comput. Vision. 104 (2013), 154–171.
[48] J. Dai, Y. Li, K. He, J. Sun, R-FCN: object detection via region-based fully convolutional networks, in Advances in Neural Information Processing Systems, Long Beach, CA, USA, 2016, pp. 379–387.
[49] S. Yang, Y. Xiong, C.C. Loy, Face detection through scalefriendly deep convolutional networks, arXiv preprint arXiv: 1706-02863, 2017.
[50] Y. Zhang, D. Zhou, S. Chen, S. Gao, Y. Ma, Single-image crowd counting via multi-column convolutional neural network, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 2016, pp. 589–597.
[51] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 2016, pp. 770–778.
[52] Z. Bai, D. Jiang, On the multi-scale real-time object detection using resnet, in Chinese Conference on Pattern Recognition and Computer Vision (PRCV), Xi'an, China, 2019, pp. 63–73.
[53] F. Haque, H.-Y. Lim, D.-S. Kang, Object detection based on VGG with resnet network, in 2019 International Conference on Electronics In formation, and Communication (ICEIC), IEEE, Auckland, New Zealand, 2019.
[54] S. Saha, K.M. Khabir, S.S. Abir, A. Islam, A newly proposed object detection method using faster R-CNN inception with resnet based on tensorflow, in Real-Time Image Processing and Deep Learning 2019, International Society for Optics and Photonics, Baltimore, MD, USA, 2019, vol. 10996, p. 109960X.
[55] K. Tong, Y. Wu, Recent advance In small object detection based on deep learning: a review, Image Vision Comput. 97 (2020), 103910.
[56] Y. Li, B. Sun, T. Wu, Y. Wang, Face detection with end-to-end integration of a convnet and a 3d model, in European Conference on Computer Vision, Amsterdam, The Netherlands, 2016, pp. 420–436.
[57] M. Jin, H. Li, Feature-enhanced onestage face detector for multi-scale faces, J. Electron. Imaging. 29 (2020), 013006.
[58] H. Mliki, S. Dammak, E. Fendri, An improved multi-scale face detection using convolutional neural network, Signal Image Video Process. 14 (2020), 1345–1353.
[59] H. Qin, J. Yan, X. Li, X. Hu, Joint training of cascaded CNN for face detection, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 2016, pp. 3456–3465.
[60] S. Zhang, G. He, H.-B. Chen, N. Jing, Q. Wang, Scale adaptive proposal network for object detectio in remote sensing images, IEEE Geosci. Remote Sensing Lett. 16 (2019), 864–868.
[61] C. Lisha, P. Lv, J. Xiaoheng, G. Zhimin, Z. Bing, X. Mingliang, *et al.*, MDSSD: multi-scale deconvolutional single shot detector for small objects, Sci. China Inf. Sci. 63 (2020), 120113.
[62] Q. Fan, W. Zhuo, Few-shot object detection with attention-RPN and multi-relation detector, in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 2020, pp. 4013–4022.
[63] X. Glorot, Y. Bengio, Understanding the difficulty of training deep feedforward neural networks, in Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, 2010, pp. 249–256.
[64] D. Peng, Z. Sun, Z. Chen, Z. Cai, L. Xie, L. Jin, Detecting heads using feature refine net and cascaded multi-scale architecture, in International Conference on Pattern Recognition (ICPR), IEEE, Beijing, China, 2018, pp. 2528–2533.
[65] S. Gidaris, N. Komodakis, Locnet: improving localization accuracy for object detection, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 2016, pp. 789–798.
[66] Y. Yang, D. Ramanan, Articulated human detection with flexible mixtures of parts, IEEE Trans. Pattern Anal. Mach. Intell. 35 (2012), 2878–2890.
[67] A. Vora, V. Chilaka, FCHD: fast and accurate head detection in crowded scenes, arXiv preprint arXiv: 1809-08766, 2018.
[68] R. Stewart, M. Andriluka, End-to-end people detection in crowded scenes, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 2016, pp. 2325–2333.