An Information Fusion Framework for Person Localization Via Body Pose in Spectator Crowds

Muhammad Shaban[a,b], Arif Mahmood[b,c,*], Somaya Ali Al-Maadeed[b], Nasir Rajpoot[a]

[a]*Department of Computer Science, University of Warwick, Coventry, UK.*
[b]*Department of Computer Science and Engineering, College of Engineering, Qatar University, Doha*
[c]*Department of Computer Science, Information Technology University (ITU), Lahore, Pakistan*

Abstract

Person localization or segmentation in low resolution crowded scenes is important for person tracking and recognition, action detection and anomaly identification. Due to occlusion and lack of inter-person space, person localization becomes a difficult task. In this work, we propose a novel information fusion framework to integrate a deep head detector and a body pose detector. A more accurate body pose showing limb positions will result in more accurate person localization. We propose a novel *Deep Head Detector* (DHD) to detect person heads in crowds. The proposed DHD is a fully convolutional neural network and it has shown improved head detection performance in crowds. We modify Deformable Parts Model (DPM) pose detector to detect multiple upper body poses in crowds. We efficiently fuse the information obtained by the proposed DHD and the modified DPM to obtain a more accurate person pose detector. The proposed framework is named as Fusion DPM (FDPM) and it has exhibited improved body pose detection performance on spectator crowds. The detected body poses are then used for more accurate person localization by segmenting each person in the crowd.

Keywords: Crowd Analysis; Person Segmentation; Person Localization; Information Fusion; Body Pose Detection; Upper Body Detection

1. Introduction

Vision based algorithms have achieved significant progress for scenes containing single or few persons for human detection, person tracking, localization and recognition [1–5]. However, automated analysis of relatively dense crowds is significantly more difficult [6]. It is due to a number of challenges posed by the crowded environment such as severe occlusion, low resolution, and perspective distortions. Though dense crowd analysis is more complex, it offers better solution to the real-world applications. It is important for surveillance and security, space and infrastructure management of large events such as political, religious, social, and sports gatherings. In dense crowds, person pose estimation may be considered as the first step towards person localization, recognition, anomaly identification as well as action assignment and recognition.

Deformable Parts based Model (DPM) [2] person pose detector has shown excellent performance for single or few person pose detection. The DPM separately detects positions of different body parts and then joins them using Dynamic Programming (DP). However, DPM was originally developed for scenes containing isolated or non-overlapping persons. For the case of crowded environments, the performance of DPM degrades. We modify the DPM algorithm to efficiently detect upper body poses in crowded scenes. We also propose a deep learning based novel head detector for low resolution crowded scenes. We propose fusion of information obtained by the deep head detector and modified DPM, resulting in a Fusion DPM (FDPM) algorithm which has shown better performance on crowded scenes.

Figure 1: Some sample head/face images from the SHOCK dataset. All images are of quite low resolution. (a-d) Examples of different viewing angles of head. (e) An instance with lack of facial features. (f-j) Examples of partially occluded faces.

In crowd scenes, full person detection becomes extremely challenging due to high occlusion and low person resolution. For most of the persons only upper body remains visible and

*Corresponding author
Email addresses: m.shaban@warwick.ac.uk (Muhammad Shaban), arif.mahmood@itu.edu.pk (Arif Mahmood), s_alali@qu.edu.qa (Somaya Ali Al-Maadeed), n.m.rajpoot@warwick.ac.uk (Nasir Rajpoot)
URL: itu.edu.pk/faculty-itu/dr-arif-mahmood/ (Arif Mahmood)

lower body becomes occluded therefore full person detectors cannot be applied on dense crowds. Due to close vicinity of persons and wide variation in person poses, upper body detection as a whole becomes challenging. In contrast, person heads remain visible most of the times and therefore fusion of head information obtained using deep learning with a part based upper body pose detector significantly increases the performance of the pose detector.

As discussed earlier, person heads have higher probability of being visible compared to upper bodies. Therefore, we may assume that if there is a head with high confidence there is a person, though the opposite may not be true. In order to localize persons, we experimented with different existing head/face detectors. Most of these have shown low accuracy on the densely crowded scenes because of low resolution of face/heads, varying viewing angles and absence of facial features in many cases as shown in Fig. 1.

In order to obtain an accurate head detector in crowded scenes, we propose a Deep Head Detector (DHD) which is a fully convolutional neural network. We consider the problem of head detection as a segmentation problem. The proposed DHD assigns each image pixel a probability of being a head pixel. Probability of a pixel exponentially decays as distance increases from the head center. The proposed network consists of multiple convolutional and de-convolutional layers. We compared head detection results of our proposed Deep Head Detector (DHD) with LBP, HOG and Haar based detectors. We find that even after retraining these detectors on the same dataset, these algorithms still exhibit lower accuracy than the proposed DHD algorithm. Preliminary results in this direction are recently presented in a workshop [7].

Once heads in an image are detected with high confidence, we then consider fusion of head information with FDPM which detects all upper-body candidate skeletons with certain confidence using appearance model and head segmentation probability map. The proposed refinement reduces the confidence of skeletons without a clear head and thus boosts the confidence of true detection. Then we select the skeleton with maximum confidence score from each overlapping group of skeletons, assuming the group belongs to the same person in real world. We compare the results of the FDPM with the existing methods and observe that our method performed better.

Once we select high confidence skeletons we consider the pose of the corresponding person to be encoded by that skeleton. For person localization by segmentation, we learn multiple color based Gaussian distributions for each limb of a specific person. Using these Gaussian distributions, we then perform pixel assignment to each limb. Then all limbs are integrated to yield a full person segment. The same process is repeated for all persons in the crowd. In contrast to some existing algorithms which yield rectangular body boxes [10], our algorithm produces exact person segment. The rectangular body boxes may contain more than one persons in dense crowds making action assignment very difficult. Fig. 2 shows the flow of our proposed algorithm.

The organization of the rest of the paper is as follows. Related work is reviewed in the following Section 2. Sections 3 and 4 describe our approach in detail. In section 5, we discuss experiments and results. Finally, we conclude our work and give suggestions for the possible extensions in Section 6.

2. Related Work

Significant research efforts have been devoted to the crowd analysis during the past decade. However, most of these research works have only addressed quite high level problems such as crowd counting [4, 8, 9], crowd flow classification, segmentation and stability analysis [11, 12]. Low level crowd analysis such as person pose estimation and person segmentation in crowded scenes have not been well investigated [13]. Moreover, crowd datasets used in these investigations are quite sparse in terms of number of persons. For high level tasks, research community has used extremely dense crowd datasets where hardly person heads are visible for example UCF crowd segmentation and counting datasets. Due to lack of person visibility, low level analysis cannot be performed on these datasets. On the other hand, action recognition datasets mostly have only one person or few persons performing some action. Therefore, in terms of number of persons there is a big gap between extremely dense crowd datasets and very sparse action recognition datasets which is yet to be filled. In this work, we focus on a complex spectator crowd dataset (SHOCK) [6] which consists of up to 150 persons per frame/image. Using this dataset, we perform low level analysis through head localization. Some preliminary results on SHOCK dataset were presented by the original authors [6]. We go a step forward and propose algorithms for person pose estimation using head positions and person segmentation through the predicted pose.

One can relate head segmentation/localization with face detection but in crowded scenes it is significantly different. Most of the face detectors [5, 14, 15] rely on face specific features such as skin color or structure of eyes, nose and mouth etc. But in crowds, these features are not strong enough due to low resolution of a person in an image. Additionally, face pose varies a lot from frontal to left, right, down and rear poses. Recent successes of deep neural networks for image classification [16, 17] and segmentation [18] tasks motivated us to use deep NN to address head segmentation problem where hand crafted feature based methods [5] failed on even head detection task. Our head segmentation method is inspired by the fully convolutional neural network in [19]. Initial results on the proposed head segmentation method have recently been presented in [7].

Yang *et al*. [20] proposed a face detection method in crowds where they used an object detector to generate bounding box proposal for faces and then validate each proposal with their proposed Faceness net. Whereas, our proposed method predicts all faces in a given image at once. Qin *et al*. [21] and Chen *et al*. [22] proposed a cascaded CNN to predict faces. They used multiple CNNs to predict a single face whereas we used one CNN to predict multiple heads in a given image. Recently Faster RCNN has also been employed for face detection [23]. Hu *et al*. [24] has trained CNN at multiple resolutions and the final detection scores are integrated to detect faces with large

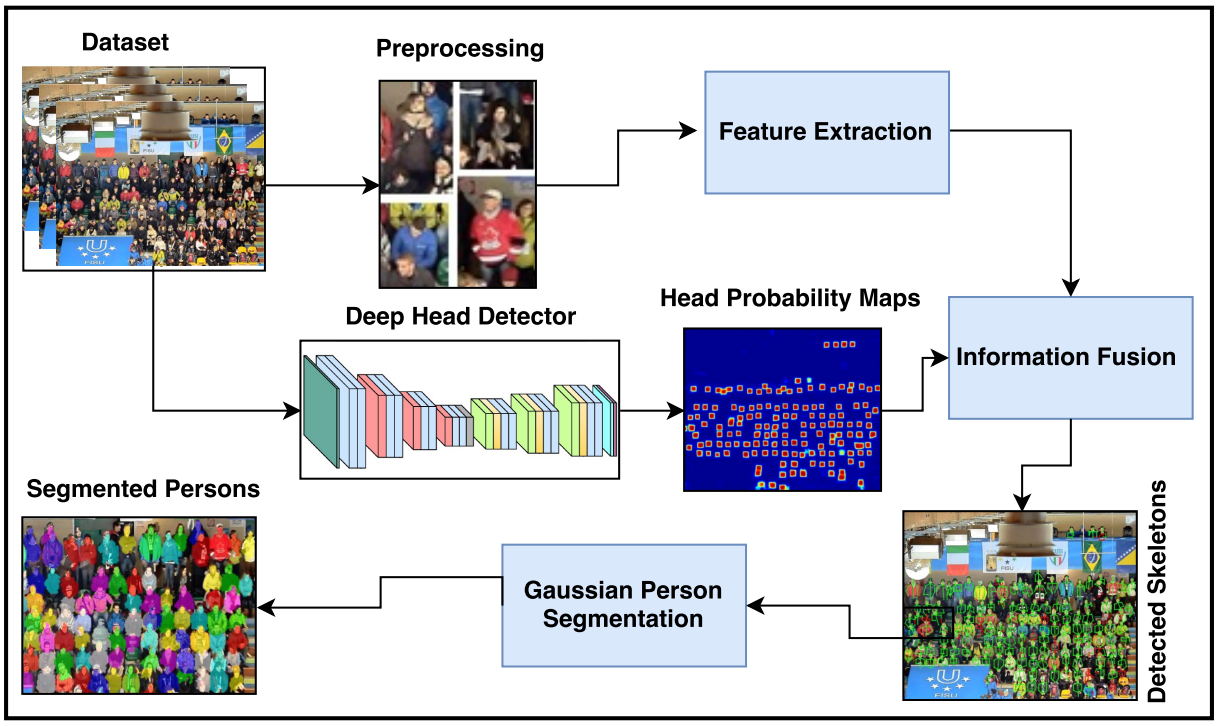

Figure 2: Flow diagram of the proposed Fusion DPM (FDPM) algorithm. During training, preprocessing phase extracts each person from raw images. Feature extraction step extracts the features of each individual to feed them into FDPM. Deep Head Detector (DHD) processes raw images to predict the head probability maps. During testing, FDPM uses each person's features and head probability map to predict person skeleton. Gaussian Person Segmentation (GPS) module uses a raw image along with predicted skeletons to generate segmentation masks for each person.

scale variations. The computational complexity will linearly increase with the number of resolutions being used.

In recent years, significant work has been done on person pose estimation in images and videos [25–28]. Deformable part based models [2, 29–31] and approaches based on deep convolutional neural network [32–35] have achieved quite good performance. The main focus of these methods was person pose estimation in scenes with mostly single person or very few persons. Performance of these methods in crowded scenes degrades because they are designed to capture appearance and geometric relation between the body parts of a single person. Hence, they are less robust to person-person occlusions. Newell *et al*.proposed a stacked network [35] for the detection of person pose. Their network was specifically designed for single high resolution person whereas our method can detect many overlapping low resolution persons. Cao *et al*. [28] has recently proposed a very fast multi-person pose detection algorithm, however their performance degrades on low resolution images as often required in crowd scenes.

Background subtraction is a very simple case of segmentation and extensively addressed in literature [36–39]. However, segmentation of person body parts is a very challenging problem due to limb articulation, diverse appearance and occlusions. Existing methods address the segmentation of one or few persons using color and motion based approaches [40, 41]. Some methods have been proposed for multiple person segmentation and occlusion handling but each method has its own limitation. For instance, the approach proposed in [42] used stereo disparity cues along with color and motion, while performance of [43] depends on pedestrian detector, and the method in [44] used information from multiple cameras to segment people. The authors of [45] used a pose and super pixel based approach to segment single person in a video. We address this problem for scenes with spectator crowds by fusing the person pose with color based Gaussian mixture models and handle the occlusion problem by introducing scene level constraints.

3. Deep Head Detector (DHD) Algorithm

In dense crowds, head/face of a person has relatively high probability of being visible as compared to the rest of the body. Therefore, presence of a head/face with high confidence means presence of a person. However, head detection in crowded scenes is a difficult problem due to significant noise and outliers. We pose the head detection as a segmentation problem by estimating a probability for each pixel being the head pixel. Our approach is different from the typical head/face or object detectors, which is to slide a filter on the image and classify each image location as an object or a non-object. We propose a deep learning based solution that segments full image at once by assigning a probability of belonging to a head to each pixel. Fig. 3a & b show an input image and head labels used for training. The output of the proposed network is shown in Fig. 3c. For each pixel probability of being a head pixel is shown using pseudo color. Since the training labels are bounding boxes for each head, therefore detection also appears as rectangles. We

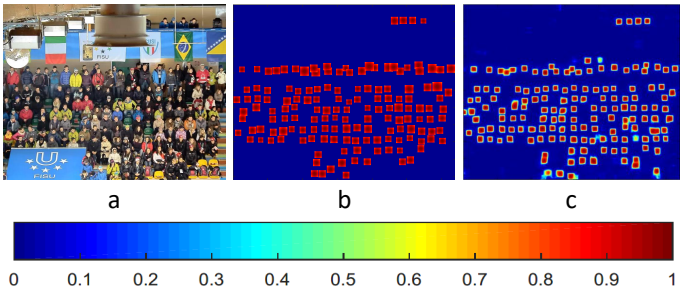

Figure 3: Training and testing of DHD: (a) Original Image. (b) The ground truth head bounding boxes with Gaussian probability weights. (c) Head probability map generated by the proposed DHD network.

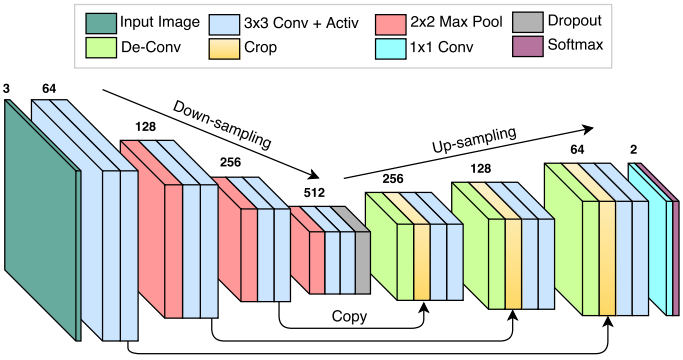

Figure 4: Deep Head Detector (DHD) network architecture: each block consists of a set of layers and each layer represents a specific operation as shown on the top of image. Within a block each layer has same depth written on the top of each block. Left to right arrows show that last layer of each down-sampling block is also used during up-sampling.

use this probability map as an input to the person pose detector as later explained in Section 4.1.

3.1. Network Architecture

The deep convolutional neural network employed for head/face detection is motivated by the success of UNET [19] for cell segmentation. A block diagram of the proposed network is shown in Fig. 4. We employ multiple down-sampling and up-sampling blocks where down-sampling blocks are used to shrink the image size and capture the context information whereas symmetric up-sampling blocks expand the down-sampled image to get better localization. Down-sampling blocks in the proposed network capture the context information whereas up-sampling blocks localize the head using features from the last layer and high resolution features from the symmetric down-sampling layers using the copy steps. Use of high resolution features from down-sampling blocks has also been proved useful to achieve better performance in many other studies, for example DCAN [59] and DenseNet [60] architectures.

To avoid overfitting and to improve learning, we empirically select the number of down-sampling blocks to be four and up-sampling blocks to be three as explained in Section 5.2. We used 2×2 max pooling with no overlap, 3×3 unpadded convolution filter to avoid boundary effect and tanh activation function. A dropout layer with 0.5 dropout rate is used in the last down-sampling block. The proposed design of a down-sampling block in the DHD network as shown in Fig. 4 is given by

$$y_i = f_{max}(y_{i-1}) \circ f_{conv}(\cdot, W_{2i-1}) \circ f_a(\cdot) \circ f_{conv}(\cdot, W_{2i}) \circ f_a(\cdot), \tag{1}$$

where f_{max}, f_{conv}, and f_a are the max-pooling, convolution and activation functions respectively, y_{i-1} is the output of previous block, $i \in \{1, 2, \ldots, D\}$, D is the total number of down-sampling blocks, y_0 represents the input image, and W is the trainable weight matrix for down-sampling block. The operator ($\circ$) provides the output of preceding function to the superseding function and operator ($\cdot$) represents the output of the preceding function.

In each up-sampling block, we concatenate the feature maps of de-convolution layer with the feature maps of last convolution layer of respective down-sampling block. The proposed design of an up-sampling block in the DHD Network is shown in Fig. 4 and given by

$$z_j = f_{dconv}(z_{j-1}, W'_{3j-2}) \circ f_{cat}(\cdot, y_{D-j}) \circ f_{conv}(\cdot, W'_{3j-1}) \circ f_a(\cdot) \circ f_{conv}(\cdot, W'_{3j}) \circ f_a(\cdot), \tag{2}$$

where f_{dconv} and f_{cat} are deconvolution and concatenation functions, z_{j-1} is the output of the previous up-sampling block and y_{D-j} is the output of the $(D-j)^{th}$ down-sampling block. W' is the trainable weight matrix for up-sampling block, $j \in \{1, 2, \ldots, U\}$ and U is total number of up-sampling blocks. The output of the last down-sampling block is the input of the first up-sampling block, $z_0 = y_D$. In case of unpadded convolution, input image size becomes crucial because each input of 2×2 max pooling layer must have even dimensions to be down-sampled perfectly. Therefore, only a set of image sizes work for given number of down-sampling blocks. The validity of an image size for a given number of down-sampling layers can be verified by Eqs. 3 and 4. This relationship depends on reduction in image size after a convolution, number of convolutions in a down-sampling block, down-sampling ratio and number of down-sampling blocks. The reduction in image size after one convolution depends on convolution filter size and stride size. Let a be the reduction in image size

$$a = n_o - \frac{n_o - f_s + 1}{S}, \tag{3}$$

where n_o, f_s and S are the sizes of image, convolution filter and stride, respectively. During down-sampling, we are reducing image size with an integer factor (max pool). Therefore, if the output size of the last down-sampling block is an integer value then output sizes of all intermediate blocks will also be integer values. Let n_D be the output size of the last down-sampling block. Using Eq.(3), n_D is given by

$$n_D = (r_d)^{D-1} \times n_o - 2a \left(\frac{1 - (r_d)^D}{1 - r_d} \right), \tag{4}$$

where r_d is the down-sampling ratio. Using (4), we ensure the input image size n_o is the one which results in an integer n_D. Our network has unpadded convolutions in both down-sampling and up-sampling paths. Therefore, output of the last up-sampling block is much smaller than the input image especially for small input images. The output dimensions are important for better selection of input image size and amount of overlapping. Let n_U be the output size of last up-sampling block. Using Eqs. (3) and (4), n_U is given by

$$n_U = (r_u)^U \times n_D - 2a\left(\frac{(r_u)^U - 1}{r_u - 1}\right) \tag{5}$$

where r_u is the up-sampling ratio. D and U are the number of down-sampling and up-sampling blocks. For a given input image size $n_o \times n_o$ the output probability map size will be $n_U \times n_U$. Since $n_U < n_o$, the output probability map will correspond to central portion of the input image, leaving the border with no probability values. To solve this problem we divide the large input image into smaller overlapping blocks of size $n_o \times n_o$. We use alpha blending to fuse the probability values of the overlapped region.

3.2. Network Output and Optimization

After the last up-sampling block, a convolution filter of size $1 \times 1 \times C$ is used to map output to the required number of classes(C). Let $f(\mathbf{u}, c)$ be the activation score at a pixel position $\mathbf{u}$ for class c. We convert these scores into probabilities by using a pixel-wise softmax. Let $p(\mathbf{u}, c)$ be the probability of pixel $\mathbf{u}$ belonging to class c

$$p(\mathbf{u}, c) = \frac{\exp(f(\mathbf{u}, c))}{\sum_{c'=1}^{C} \exp(f(\mathbf{u}, c'))}. \tag{6}$$

To measure the deviation of the predicted probability map p from the ground-truth l, we use cross entropy as criterion. We minimize following cost function

$$cost = -\sum_{\mathbf{u}} \sum_{c=1}^{C} l(\mathbf{u}, c) \log_2(p(\mathbf{u}, c)), \tag{7}$$

where $l(\mathbf{u}, c)$ represents the actual probability at pixel $\mathbf{u}$ belonging to class c. We use Adam stochastic optimization method [46] for optimization of our model with 0.001 initial learning rate. The probability map p after convergence of the algorithm is used as an input to the person pose detector as explained in Section 4.1. The probability map p can also be used for the localization of heads in crowed images as discussed in Section 5.2.

4. Information Fusion for Person Pose Detection in Dense Crowds

Person pose estimation is a problem of localization of person body parts such as head, torso, upper and lower arms (Fig. 5 a). These parts have different sizes. We divide each body part in to multiple keypoints (Fig. 5b). We use a rectangular window to extract appearance feature of each keypoint (Fig. 5c). Then we train a model that learns appearance of each portion, their co-occurrences and spatial relations.

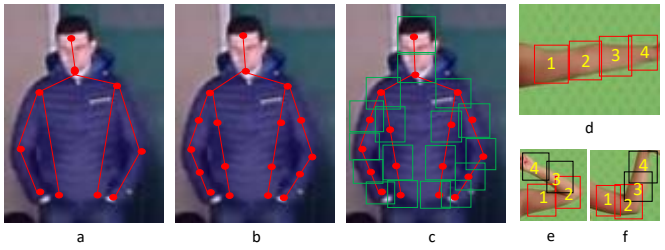

Figure 5: (a) Upper body skeleton overlaid on a person. (b) Extended skeleton used for training. (c) A rectangular patch is used to capture appearance information of each of the 18 sub-parts. (d-f) Each rectangular box represents a state. Three different images represent different forms of each state where same colored states (red or black) share the same orientation.

Figure 6: (a) Upper body pose detection results of DPM algorithm with training by the original authors on Buffy dataset. (b-c) Zoomed in versions of two different areas in (a). Green and red lines represent true and false detections. Large number of false detection are clearly visible. To avoid these errors, we have to retrain DPM on SHOCK training dataset.

In order to estimate the position of different limbs and body pose of an individual in an image containing a crowd we consider Deformable Parts Model (DPM) [2]. The performance of the DPM algorithm is excellent on the images containing single or few persons which are well separated. However, we practically observed that the performance of DPM degraded significantly on images containing crowded scenes with large number of persons close to each other. One such example is shown in Fig. 6 in which one can see false detections, miss detections and incorrect poses and limb positions. It is because person pose estimation in densely crowded images is more challenging task due to low resolution of persons and severe occlusions. In order to handle these challenges we fuse the information extracted by DPM algorithm and the information extracted by the deep head detector and obtained good performance on crowd images. We named this technique as Fusion DPM (FDPM) algorithm.

4.1. Model for the Crowded Scenes

In classical part models [47] all possible pose articulation of a body part are generated by scaling and rotation of a base sate of that body part e.g. upward and horizontal pointed lower arm can be generated by rotation of its downwards pointing state. Linear transformations of a base state cannot generate all possible real world states of a limb, especially out of plane rotations. A better approach is to represent the articulated pose of a person using a mixture-of-states for each body part. A state represents a sub-part in a particular appearance, orientation and scale (Fig. 5d-f). Therefore, we do not use manually rotated and foreshortened states of each body part to model an articulated pose. Instead, a fixed set of possible rotated and foreshortened states for each body part are inferred from training dataset using clustering to model an articulated pose. Each limb can appear in different orientations but in a particular orientation all states on a rigid limb will share the same orientation due to the rigidity constraint, Fig. 5d-f. This constraint motivates us to learn a prior from the co-occurrence of states with specific orientation and scale. This kind of prior will favor the realistic poses over non-realistic ones.

We divide a person into K sub-parts (keypoints), for each sub-part we have S unique states thus we have $S \times K$ states in total. Let M_s be the state matrix of size $S \times K$ where each column corresponds to a particular sub-part and each value in that column corresponds to a unique orientation and scale of that sub-part. We compute a co-occurrence matrix (M_{co}) of size $KS \times KS$ which encodes co-occurrence score of any two states in the training data. We also compute an occurrence score matrix (M_{os}) of size $S \times K$, using training data. For this purpose K-means clustering is applied on the normalized position of each sub-part with respect to its parent sub-part. M_{os} encodes the occurrence score of a particular state for a particular sub-part which is size of the corresponding cluster. A pose P is a set of states such that exactly one state is selected from the each column of the state matrix, $P = \{s_1, s_2, \cdots s_K\}$. The sum of occurrence scores of these states is the occurrence score of P and sum of co-occurrence score of these states is the co-occurrence score of P. The overall score of the pose is the sum of these two scores [2].

$$C(P) = \sum_{i=1}^{K} M_{os}\{s_i\} + \sum_{i=1}^{K}\sum_{j=1}^{K} M_{co}\{s_i, s_j\}, \qquad (8)$$

where $M_{os}\{s_i\}$ is the occurrence score of state s_i in the matrix M_{os} and $M_{co}\{s_i, s_j\}$ is the co-occurrence score of states $\{s_i, s_j\} \in P$ as given by the matrix M_{co}.

Another model is used to capture the appearance of each state. It primarily learns color and illumination invariant template of that state. Let M_w be the matrix such that each row corresponds to a particular state in the state matrix. Each row contains state template representing significance of a particular dimension of the feature of that state. Thus a state template partially encodes the global geometry of the pose because each state has different appearance model. The appearance cost [2] for pose P is given by

$$A(I, P) = \sum_{i=1}^{K} M_w\{s_i\} \cdot \phi(I, s_i), \qquad (9)$$

where I is an image, $M_w\{s_i\}$ is the weight vector for the state s_i and $\phi(I, s_i)$ is HOG feature vector computed for state s_i in image I.

Along with variability in orientation and appearance, body parts can also deform in many ways while being a valid pose. However, this deformation is limited due to kinematic constraints. Let M_d be the state deformation weight matrix of size $K^2S^2 \times 4$ where each row corresponds to the deformation weight for a pair of states. The deformation cost [2] for pose P is given by

$$D(I, P) = \sum_{i=1}^{K}\sum_{j=1}^{K} M_d\{s_i, s_j\} \cdot \Phi(s_i, s_j), \;\; \{s_i, s_j\} \in P, \qquad (10)$$

where $\Phi(s_i, s_j) = [dx \;\; dx^2 \;\; dy \;\; dy^2]$ encodes the deformation between state s_i and s_j, and $dx = x_i - x_j$ and $dy = y_i - y_j$ denote the relative location of each state s_i with respect to each state s_j. $M_d\{s_i, s_j\}$ is a row in the state deformation weight matrix corresponding to state s_i and s_j.

In crowded environments often head has low resolution and lot of pose variations. Therefore HOG based appearance model in DPM exhibits degraded performance. We propose the output of DHD algorithm to be fused with HOG based score. For this purpose, we convert the output of DHD algorithm (head probability map) into head confidence score by subtracting a threshold value. A positive score will favour the head detection whereas a negative score will disfavour. We scale the HOG score by the head presence score at each location.

$$H(I, P) = \alpha \cdot \sum_{s_h \in P} A(I, s_h) \cdot (\psi(s_h) - \beta), \qquad (11)$$

where s_h are the states corresponding to the two head sub-parts including forehead and chin/neck, $\psi(s_h)$ represents the probability of s_h being a head, given by Eq. (6). $A(I, s_h)$ is the appearance score of each state in s_h, and α and β are hyper parameters.The α is the relative weight of head prior and the β is the threshold. If the head probability is larger than β then $H(I, P)$ is positive otherwise it will be negative. Depending upon the value of $\psi(s_h)$, overall score of a pose will increase or decrease. Both α and β are empirically learned from the training dataset.

The complete fusion model, consisting of co-occurrence and head priors, appearance and deformation models is given by

$$M(I, P) = C(P) + A(I, P) + H(I, P) + D(I, P). \qquad (12)$$

The term $H(I, P)$ may turn out to be negative in case of poor head appearance. Similarly $D(I, P)$ may evaluate negative if sub-part positions in a test image I are away from the positions in the model for pose P. In such cases, both of these terms will

cause reduction in the overall value of $M(I, P)$. During training process the weight matrices M_w and M_d are learned while maximizing $M(I, P)$ using Structured SVM. The learned matrices are then used during testing. Poses with $M(I, P)$ larger than a threshold are considered as candidate predictions while poses with smaller $M(I, P)$ are discarded. In crowded scenes, mostly only upper body parts including heads, shoulders, arms and torso remain visible. Due to occlusion caused by the surrounding persons and other objects such as seats in the stadium, lower body parts become invisible. Therefore, in this work we trained only an upper body pose detector for persons in crowded environments.

4.2. A New Accuracy Measure for Pose Detection in Crowds

Current evaluation metrics include Probability of a Correct Pose (PCP) [26] , Probability of Correct Key-point (PCK) [2] and Average Precision of Keypoints (APK) [2]. Both PCP and PCK only penalize miss-detection while do not incorporate false positives, resulting in high accuracy in case of multiple detection of the same person. The common drawback of these measures is that these only report the detection accuracy for each body part independently. In multiple detection, individual keypoints may gain good accuracy while the overall pose detection remains poor. It is because the overall accuracy of a pose is not efficiently captured by the individual keypoint accuracy. In order to handle this problem, we propose a new evaluation metric for accuracy of pose estimation which measures the correct *Probability of Full Pose (PFP)*. In case of multiple detection of the same person, in the proposed PFP, only the best matching skeleton is considered true positive while the remaining skeletons are false positives. The PFP score depends on the percent correctly detected keypoints in the true positive skeleton, with a penalty induced by the false positive skeletons.

For a crowded image containing multiple subjects we compare the set of predicted skeletons S_p and the set of ground truth skeletons S_{gt}. For each ground truth skeleton, we select the best matching predicted skeleton. While comparing two skeletons each corresponding keypoint is compared. We used the same criterion as used for PCK and APK in [2] to evaluate the correctness of an individual key-point. A key-point is considered correct if it falls within $\gamma \times max(h, w)$ pixels of ground truth key-point, where h and w are the height and width of the person bounding box. We consider $\gamma = 0.20$ which is not too strict nor too loose for evaluation of predicted keypoints and it is also used in previously published works [2] for PCK and APK evaluation measures. We experimented with different values of γ for APK measure to show the consistency in prediction results (Table2). For a given ground truth skeleton, a predicted skeleton with maximum number of correct keypoints is the best match. Let S_{tp} be the set of best matching skeletons or true positives, $S_{tp} = S_p \cap S_{gt}$. If no predicted skeleton with at least one correct keypoint is found then that ground truth skeleton has remained undetected and constitutes a false negative. Let S_{fn} be the set of false negatives, $S_{fn} = S_{gt} - S_{tp}$. All predicted skeletons which are not in the set S_{tp} are false positives S_{fp}, $S_{fp} = S_p - S_{tp}$. For each false negative and false positive skeleton we assign a zero PFP score. For the true positive

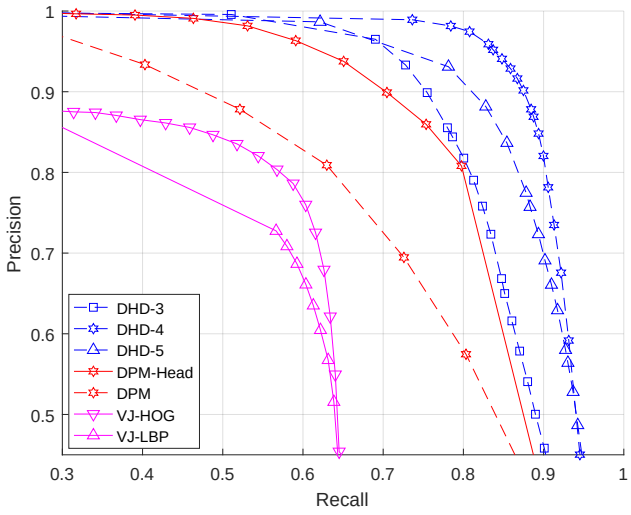

Figure 7: Precision recall curve based comparison of different methods trained on SHOCK dataset.

skeletons, PFP score is the ratio of correct keypoints to the total keypoints. Let $PFP(i)$ be the PFP score of i^{th} skeleton.

$$PFP(i) = \begin{cases} \frac{Number\ of\ Correct\ Keypoints}{Total\ Number\ of\ Keypoints}, & i \in S_{tp} \\ 0, & i \in S_{fp} \\ 0, & i \in S_{fn} \end{cases} \quad (13)$$

For a given crowd image containing large number of persons, an image PFP score is defined as

$$I_{PFP} = \frac{\sum_{i \in S_{tp}} PFP(i)}{|S_{tp}| + |S_{fp}| + |S_{fn}|}, \quad (14)$$

where $|\cdot|$ represents the cardinality of the set. For a dataset of crowd images, average I_{PFP} over all test images is used for comparison. The proposed measure *Image Probability of Full Pose* I_{PFP} more accurately captures accuracy of a pose detection method in crowded scenes.

5. Experiments and Results

The proposed algorithms are tested on two publicly available datasets including a crowd dataset SHOCK [6] and Buffy [1] dataset. The DHD algorithm is compared with four existing methods including Viola-Jones HOG [5], Viola-Jones LBP [5], DPM head [2] , and DPM upper body [2] detectors. The FDPM algorithm is compared with six existing methods including Tran [48], Andriluka [49], Eichner [50], Sapp-1 [51], Sapp-2 [52] and DPM [2]. The proposed DHD and FDPM algorithms have shown excellent performance in all experiments.

5.1. The Datasets

In our experiments we used a recently published spectator crowd dataset by Davide *et al*. [6]. This dataset captured spectators of an ice hockey match in Trento, Italy. It consists of 60

videos of 30 seconds each, 15 videos have different types of annotation such as body and face bounding boxes of each person. Each frame is of 1024×1280 pixels and around 75 to 150 spectators are visible in each frame. We randomly divided annotated videos into training dataset consisting of 10 videos and testing dataset consisting of 5 videos. For the training of head localization method we used head bounding boxes as ground truth. We convert these bounding boxes into probability maps where probability at the centers of each box is 1 and probability of neighbouring pixels exponentially decays as distance increases from the head box center using Gaussian distribution. The minimum probability within the bounding box is clipped to 0.75. For pose estimation we manually annotated upper body pose of 10 thousand persons in training videos. Note that SHOCK is the only crowd dataset which has person annotations. In other very dense crowd datasets [9, 11, 53] individual persons are not visible, therefore these datasets cannot be used for person pose detection or person segmentation.

The Buffy dataset consists of unconstrained images with associated ground-truth stick-men annotations. It is very challenging in term of person appearance, scale variability, highly cluttered background, and different kinds of person clothing. A line segment is provided as annotation indicating location, size and orientation of the upper body parts (head, torso, upper/lower right/left arms). Exactly one person is annotated in each frame and there are 748 annotated frames from 5 episodes of the fifth season of the TV show *Buffy the Vampire Slayer*. We report pose detection results on a test subset from this dataset which consists of three episodes, in total 276 frames.

5.2. Experiments on Head Detection

For the training of the proposed DHD, we randomly cropped 10 patches of size 428×428 pixels from each frame of the training videos with frame rate reduced to one frame per second. On-line data augmentation is done by random horizontal flipping of training patches during each epoch. We train our network for 30 epochs of data. Beyond that error reduction is not significant. The down-sampling ratio is fixed to 0.50 and the up-sampling ratio was fixed to 2.00.

We train our network by varying the down-sampling blocks to three (DHD-3), four (DHD-4) and five (DHD-5). The corresponding up-sampling blocks are 2, 3, and 4 respectively. We observed that the deeper network DHD-5 started learning undesired low level details such as pixel color values for positive class, instead of learning a useful mixture of high level and low level information. On the other hand, the shallow network DHD-3 was unable to segment all heads in the given images. It accounted more false negatives while the deeper network DHD-5 exhibited lot of false positives especially on patches containing skin color such as hands. The DHD-4 network learned a good representation of data and hence performed the best as compare to DHD-3 and DHD-5 (Fig. 7).

We compared the results of DHD algorithm with two implementations of Viola-Jones (VJ) object detector [5] and two variation of Deformable Part Models (DPM). For a fair comparison we retrained VJ with LBP and VJ with HOG features on SHOCK training dataset. The generic trained VJ by the original

Figure 8: VJ-LBP and VJ-HOG are the head detection results of Viola-Jones detector trained using LBP and HOG features. DHD is the propose head detector. Green is true positive, red is false positive and black is false negative.

Table 1: Head detection results of different methods trained and tested on the SHOCK dataset.

Algorithm	Precision	Recall	F1 Score
DHD-3	0.8989	0.7549	0.8206
DHD-4	**0.9287**	**0.8588**	**0.8924**
DHD-5	0.8816	0.8279	0.8539
VJ-HOG [5]	0.7600	0.6035	0.6727
VJ-LBP [5]	0.7187	0.5734	0.6379
DPM-Head [2]	0.8596	0.7535	0.8031
DPM [2]	0.6946	0.7259	0.7099

authors showed degraded performance on the SHOCK dataset. We also retrained DPM model with three parts (head, face, neck) and 18 parts (upper body) on SHOCK training dataset. The DPM trained on Buffy by the original authors has shown degraded performance on the SHOCK dataset. Retraining has significantly improved performance of these existing algorithms. Results of the retrained methods are reported in Table 1 which is organized according to the maximum F_1 scores achieved by each algorithm

$$F_1 = 2(\text{Precision} * \text{Recall})/(\text{Precision+Recall}).$$

Corresponding precision and recall values are also given in Table 1. Precision-recall curves for all methods are shown in Fig. 7. These results demonstrate that the proposed DHD-4 algorithm has outperformed the other methods with a significant margin. The proposed DHD-4 can process 2.66 frames per second on Intel Xeon CPU E5-2650 with 128GB RAM. Note that DHD-4 is significantly faster than the DPM based head detector which takes 2.32 seconds to process one frame on the same machine.

5.3. Experiments on Pose Detection

In order to train a pose detector on crowd dataset we manually annotated 10 joints in the upper body of each person in

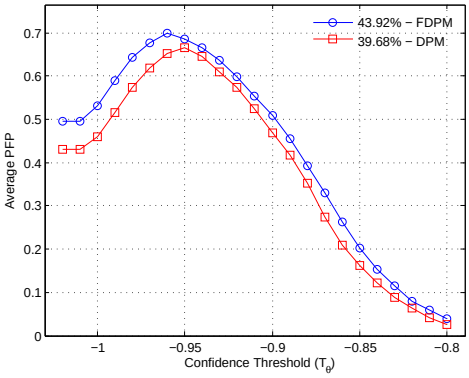

Figure 9: Comparison of the proposed FDPM with the existing DPM algorithm using average PFP score on SHOCK dataset.

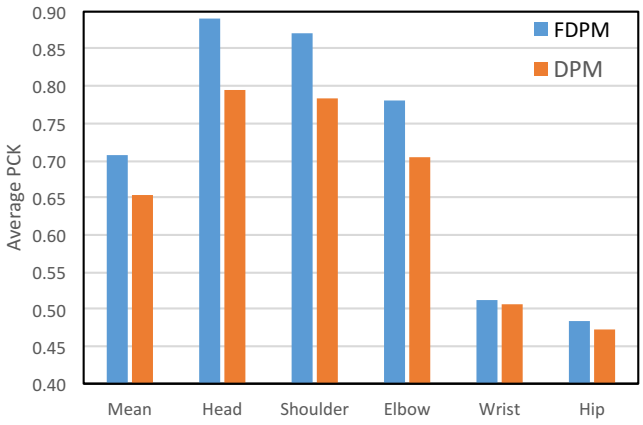

Figure 10: Comparison of FDPM and DPM algorithms using average Probability of Correct Keypoint (PCK) criterion on SHOCK dataset.

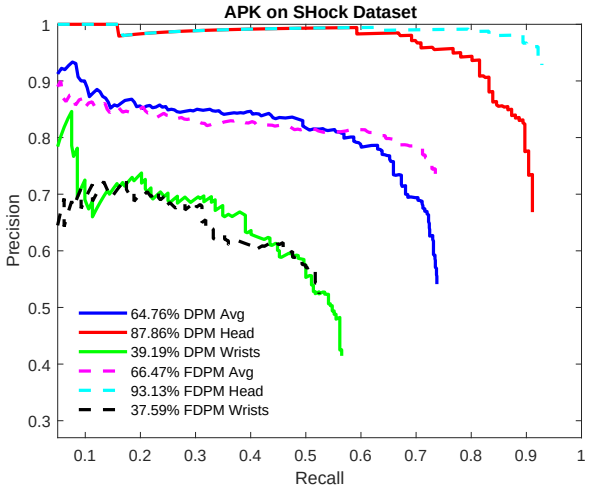

Figure 11: Comparison of FDPM and DPM algorithms using Average Precision of Keypoints (APK) criterion on SHOCK dataset.

training (10 videos) and test datasets (5 videos). Annotations include one point on head, neck, each shoulder, each elbow, each wrist, and two on torso (Fig. 5a). Each limb is divided into multiple parts to learn multiple templates (Fig. 5b). In training data, considering each person skeleton independently we get 10 thousand skeletons. However, there are significant redundancies in skeletons due to minor person motion across many frames. We prune these redundant skeletons to get only highly articulated skeletons using K-mean clustering with $K = 300$. Only fully visible skeletons are considered for training. The selected skeletons are flipped left-right because of equal probability of each instance to happen in real world scenarios. Negative training images are taken from the INRIA Person database [54]. For fair comparison, DPM algorithm is also retrained on the same dataset.

During test for each skeleton confidence score $M(I, P)$ is computed using Eq. (12). Skeletons with $M(I, P) \geq T_\theta$ are considered as candidate skeletons where T_θ is the confidence threshold. A small value of T_θ will result in large number of skeletons and vice verse. Fig. 9 shows variation of average PFP with the variation of T_θ from minimum confidence threshold to a maximum value. In order to compare the proposed FDPM with the existing DPM we computed Area Under the Curve (AUC) in Fig. 9 using mean of the average PFP scores. Overall, our method has higher AUC score as compared to DPM. Our method not only perform better in term of AUC but it is also quite robust to false positives as it has higher average PFP score on lower thresholds.

In order to compare our method with the existing methods we also evaluate it on PCK, PCP and APK criteria in addition to our proposed PFP metric. For individual keypoint analysis we use PCK and APK evaluation criteria for SHOCK dataset. PCK is a false positive invariant criterion and only penalize miss detections. Therefore, its best score is at lower confidence threshold ($T_\theta = -1.02$) as compared to PFP. Our proposed FDPM performed much better than the DPM in terms of keypoint evaluation. FDPM has higher average PCK for all keypoints. Fig. 10 shows that head prior not only helps to prune false detections but also improves localization of head and its neighboring keypoint.

To evaluate our method using APK criterion, we compute precision of all keypoints at different recall values. Then we average all these precision values for each keypoint across dataset. Which represents the APK score of a keypoint. To report an APK for pose on whole dataset we again average the APK scores across keypoints. Our algorithm has shown higher mean APK score on SHOCK dataset as compared to DPM. Comparison of both methods is presented in Table 3 and summarized in Fig. 11.)

In SHOCK dataset, people are in stadium seating arrangement, therefore wrists and hips have lower probability of being visible as compared to heads, shoulders and elbows. Wrists also become invisible if hands are inside pockets. Moreover, annotation of occluded limbs on a low resolution dataset is also a challenging task and it is not as precise as heads, shoulders and elbows annotation. Therefore the accuracy of all compared methods has significantly degraded on wrist and hip compared to head, shoulder and elbow as shown in Table 3.

We also evaluated our method on a non-crowd dataset where only few persons are visible. We used Buffy dataset for this

Table 2: Average Precision of Keypoints (APK) Score for different values of alpha γ on SHOCK Dataset

Keypts	**gamma**	**0.10**	**0.15**	**0.20**	**0.25**	**0.30**
Average	FDPM	34.26	52.29	66.47	74.97	82.78
	DPM	32.81	51.45	64.76	74.91	81.97
Head	FDPM	81.42	90.60	93.13	93.29	93.68
	DPM	75.42	86.01	87.86	90.91	91.90
Wrist	FDPM	9.42	23.58	37.59	50.73	62.11
	DPM	11.40	25.20	39.19	52.72	62.77

Table 3: Comparison of Average APK Scores on SHOCK Dataset between the original DPM, DPM with False Positive Removal (DPM-FPR) and Fusion DPM (FDPM)

Method	Mean	Head	Sholdr	Elbow	Wrist	Hip
DPM	64.76	87.86	86.69	72.21	**39.19**	**37.83**
DHDFPR	64.99	89.26	87.35	72.24	38.85	37.21
FDPM	**66.47**	**93.13**	**91.21**	**75.61**	37.59	34.81

purpose because many previous pose detection methods have reported their results on it. Test dataset images also have a subset of bounding boxes which are detected by a rigid HOG upper-body detector. To make a fair comparison we also detect skeletons within these bounding boxes. On this dataset our method has obtained similar performance as compared to the other state of the art methods including Tran [48], Andriluka [49], Eichner [50], Sapp 1 [51], Sapp 2 [52], and DPM [2] pose detectors as shown in Table 4. We also compared our results with the DPM in term of APK on Buffy dataset (Table 5). Note that, these results are generated using Buffy ground truth, where an image has only one annotated person.

We observe that FDPM algorithm is more robust to partial occlusions and self-occlusions, especially when the lower body gets occluded due to surrounding persons in the spectator crowd. Occlusions are handled during pose detection phase which is done partially, only on the upper body. Pose detection model is based on four different costs as discussed in Section 4. Three of these costs, including appearance, co-occurrence and deformation help to predict the most probable location of an occluded limb. However, similar to the other pose detectors, the FDPM cannot find the body pose of a person if only head is visible and everything else gets occluded.

Table 4: PCP on Subset of Buffy Test Set

Method	Torso	Head	U.arms	L.arms	Avg
Andriluka [49]	90.7	95.5	79.3	41.2	76.7
Eichner [50]	98.7	97.9	82.8	59.8	84.8
Sapp 1 [51]	**100**	**100**	91.1	65.7	89.2
Sapp 2 [52]	**100**	96.2	95.3	63	88.6
DPM [2]	98.8	99.2	94.8	**68.6**	90.3
FDPM	**100**	99.5	**97.0**	68.3	**91.2**

Table 5: Average Precision of Keypoints (APK) Score on Buffy Dataset.

Method	Avg	Head	Shoulder	Elbow	Wrist	Hip
DPM	77.48	86.48	87.03	80.32	**57.27**	76.31
FDPM	**78.24**	**87.44**	**88.02**	**81.17**	56.30	**78.29**

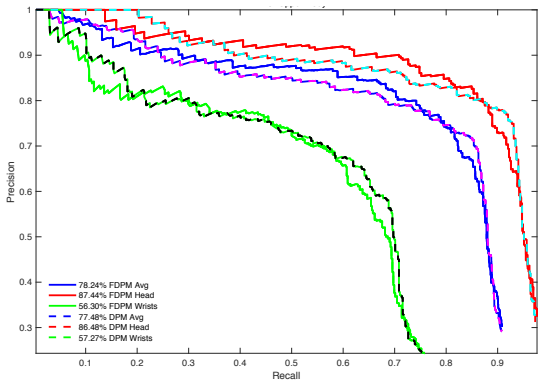

Figure 12: Comparison of FDPM and DPM algorithms using Average Precision of Keypoints (APK) criterion on Buffy dataset.

5.4. Person Localization by Segmentation

On a given image, after pose detection, we apply Gaussian Person Segmentation (GPS) to segment each person. GPS is a color based classification of pixels belonging to a particular person-limb. We use 10 joints from the upper body pose as shown in Figure 5a for upper body segmentation. Two adjacent joints $(\mathbf{v}_i, \mathbf{v}_j)$ represent a limb. Let $\mathbf{m}_{ij}$ be the midpoint of the limb computed as average of the two joints. Let d_{ij} be the distance of $\mathbf{m}_{ij}$ from each of its end. For each limb we consider an ellipse shaped region assumed definitely belonging to that limb (Figure 15a). All pixels within this ellipse are considered as ground truth.

$$\frac{\mathbf{m}_{ij}(x)^2}{s_x d_{ij}} + \frac{\mathbf{m}_{ij}(y)^2}{s_y d_{ij}} = 1, \tag{15}$$

where $\mathbf{m}_{ij}(x)$ and $\mathbf{m}_{ij}(y)$ are the x and y components of $\mathbf{m}_{ij}$. Scale factors (s_x, s_y) are used to control ellipse height and width. In all of our experiments we used $(s_x, s_y) = (0.6, 1)$ for heads, $(0.65, 0.75)$ for torsos, and $(0.5, 1)$ for the remaining limbs. In order to align an ellipse with a particular limb we rotate the ellipse by $\theta_{ij} = \cos^{-1}\{(\mathbf{v}_i^\top \mathbf{v}_j)/(|\mathbf{v}_i||\mathbf{v}_j|)\}$. The number of colored regions may vary across limbs. For example, for face the color of hairs is significantly different from the color of forehead. For each of these regions we need to learn a different color model. In our experiments we learn three color models for head and two for all other limbs using EM algorithm. For each model we learn its weight w_i, mean μ_i, and covariance Σ_i. Given these parameters probability of a pixel $\mathbf{x} = [x_r,\ x_g,\ x_b]^\top$ belonging to a particular model is given by

$$g(\mathbf{x}|\mu_i, \Sigma_i) = \frac{1}{(2\pi)^{D/2}\sqrt{|\Sigma_i|}} \exp\Big(-\frac{1}{2}(\mathbf{x} - \mu_i)^t \Sigma_i^{-1}(\mathbf{x} - \mu_i)\Big), \tag{16}$$

where $D = 3$ is the color channels of the image. The probability of a pixel belonging to a particular limb with model param-

Figure 13: Comparison of the proposed FDPM algorithm with the existing DPM technique. FDPM has mostly performed better than DPM algorithm.

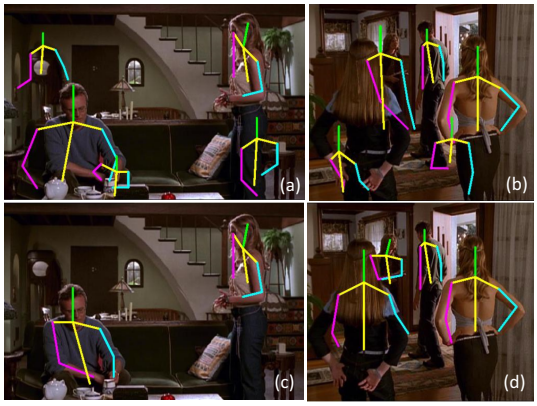

Figure 14: Pose detection results comparison on Buffy dataset: (a-b) Pose detection by DPM algorithm (c-d) Pose detection by the proposed FDPM algorithm. It can be observed that quality of detected poses has significantly improved.

eters λ is given by

$$p(\mathbf{x}|\lambda) = \sum_{i=1}^{M} w_i g(\mathbf{x}|\mu_i, \Sigma_i), \tag{17}$$

where M is the number of the models learned for that limb.

In order to find all pixels that belong to a particular limb an extended ellipse is drawn which is 1.5 times bigger than the ellipse used for the learning parameters. For each limb the extended ellipses are shown in Figure 15b. We assume that all pixels belonging to a particular limb are contained within the extended ellipse. For each pixel $\mathbf{x}$ within the extended ellipse, probability $p(\mathbf{x}|\lambda)$ is computed using Eq. 17. If the probability $p(\mathbf{x}|\lambda)$ is larger than a threshold the pixel $\mathbf{x}$ is assigned label of that person otherwise the pixel remains unlabeled. If a pixel remains unlabeled after considering all skeletons detected by the pose detector then that pixel belongs to the background. For a particular person, pixel regions surrounded by the labeled pixels are also assigned the same label. A morphological closing

Table 6: Proposed Gaussian Person Segmentation (GPS) algorithm compared with Background/Foreground Segmentation Algorithms

Method	Precison	Recall	F1 Score	IoU
DCOLOR [55]	0.69	0.50	0.58	0.41
FPCP [56]	0.40	0.99	0.57	0.40
GODECK [57]	0.40	0.99	0.56	0.40
RMAMR [58]	0.42	0.99	0.58	0.41
Proposed GPS	**0.90**	**0.72**	**0.80**	**0.67**

Figure 15: (a) regions used for training of the color model of respective body parts are represented by small green ellipses whereas (b) represents the regions with large orange ellipses that are used to segment each body part.

operation (erosion then dilation) is applied to smooth out the labeled region boundaries.

In crowd scenes most persons are partially occluded by the neighboring persons. Therefore, before learning a color model for a person-limb, we need to verify that limb must be visible, not occluded. In SHOCK dataset we observe that mostly a person is occluded by the persons sitting in front of him. To resolve this type of occlusions we segment persons in a specific order. From the set of given persons, a person with lowest head position is processed first. This person is assumed to be in front of all of the remaining persons in the set.

GPS algorithm is evaluated using F1-score and intersection over union (IoU). GPS outperformed the foreground-background segmentation algorithms as well as moving object detection methods with a significant margin on SHOCK dataset (Table 6). In this experiment our proposed algorithm obtained F1-score of 0.80 and IoU of 0.67. An example frame from SHOCK with each person as one segment automatically detected by the GPS algorithm is shown in Fig. 16. In our experiments, the GPS algorithm was able to obtain significantly more accuracy than current state-of-the-art algorithms. Note that the existing algorithms in Table 6 were executed on a batch of 15 frames while the proposed GPS algorithm was executed on a single frame. Table 6 shows average results computed on 31 frames from each of the five test videos.

6. Conclusion

A person localization method via improved body pose is proposed for crowded scenes. The improvement in body pose is

Figure 16: Person segmentation estimated by Gaussian Person Segmentation (GPS) algorithm in one frame of SHOCK dataset.

based on fusion of information obtained by modified DPM and a novel Deep Head Detector (DHD) which is a deep learning based head/face detector. The DHD has shown better performance than current head/face detectors in low resolution crowded scenes. The proposed fusion based person pose detection algorithm has achieved better performance on crowded scenes in SHOCK dataset and also on multi-person Buffy dataset, compared to the original DPM and other existing algorithms. A Gaussian Person Segmentation (GPS) algorithm is used to segment all pixels belonging to a single person using the detected pose by FDPM algorithm as spatial prior. A Gaussian mixture model is learned for each limb, which is then used to decide which pixels actually belong to that limb. The experiments demonstrate better performance of the proposed methods over current state-of-the-art algorithms. In the proposed fusion framework both DHD and DPM capture different information resulting in improved performance. The proposed fusion framework is generic and may be used with other head and pose detection methods as well. An interesting future direction may be an early fusion by jointly optimizing both detectors. The proposed FDPM algorithm learns the body pose from the annotations in the training dataset. Given enough examples of rotated persons in the training data, FDPM will be able to detect pose of rotated persons. Training on a dataset containing large number of rotated persons is also an important future direction.

References

[1] M. Eichner, M. Marin-Jimenez, A. Zisserman, V. Ferrari, 2D articulated human pose estimation and retrieval in (almost) unconstrained still images, Int. J. Computer Vision 99 (2) (2012) 190–214.

[2] Y. Yang, D. Ramanan, Articulated human detection with flexible mixtures of parts, IEEE Transactions on Pattern Analysis and Machine Intelligence 35 (12) (2013) 2878–2890.

[3] L. Pishchulin, E. Insafutdinov, S. Tang, B. Andres, M. Andriluka, P. V. Gehler, B. Schiele, Deepcut: Joint subset partition and labeling for multi person pose estimation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 4929–4937.

[4] M. Rodriguez, I. Laptev, J. Sivic, J.-Y. Audibert, Density-aware person detection and tracking in crowds, in: ICCV, IEEE, 2011, pp. 2423–2430.

[5] P. Viola, M. J. Jones, Robust real-time face detection, Int. J. Computer Vision 57 (2) (2004) 137–154.

[6] D. Conigliaro, P. Rota, F. Setti, C. Bassetti, N. Conci, N. Sebe, M. Cristani, The SHOCK dataset: Analyzing crowds at the stadium, in: CVPR, 2015, pp. 2039–2047.

[7] M. Shaban, A. Mahmood, S. Al-maadeed, N. Rajpoot, Multi-person head segmentation in low resolution crowd scenes using convolutional encoder-decoder framework, in: International Workshop on Representation, analysis and recognition of shape and motion FroM Image data (RFMI 2017), 2017.

[8] V. Rabaud, S. Belongie, Counting crowded moving objects, in: CVPR, Vol. 1, IEEE, 2006, pp. 705–711.

[9] H. Idrees, I. Saleemi, C. Seibert, M. Shah, Multi-source multi-scale counting in extremely dense crowd images, in: CVPR, 2013, pp. 2547–2554.

[10] H. Idrees, K. Soomro, M. Shah, Detecting Humans in Dense Crowds Using Locally-Consistent Scale Prior and Global Occlusion Reasoning, in: IEEE Transactions on Pattern Analysis and Machine Intelligence, (37) 10, (2015) 1986–1998.

[11] S. Ali, M. Shah, A Lagrangian particle dynamics approach for crowd flow segmentation and stability analysis, in: CVPR, IEEE, 2007, pp. 1–6.

[12] E. L. Andrade, S. Blunsden, R. B. Fisher, Hidden Markov models for optical flow analysis in crowds, in: ICPR, Vol. 1, IEEE, 2006, pp. 460–463.

[13] S. Gong, T. Xiang, S. Hongeng, Learning human pose in crowd, in: ACM IWMPVA, ACM, 2010, pp. 47–52.

[14] R.-L. Hsu, M. Abdel-Mottaleb, A. K. Jain, Face detection in color images, IEEE Trans. Patt. Anal. Mach. Intel. 24 (5) (2002) 696–706.

[15] R. Lienhart, J. Maydt, An extended set of Haar-like features for rapid object detection, in: ICIP, Vol. 1, IEEE, 2002, pp. I–900.

[16] A. Krizhevsky, I. Sutskever, G. E. Hinton, ImageNet classification with deep convolutional neural networks, in: Advances in NIPS, 2012, pp. 1097–1105.

[17] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, arXiv preprint arXiv:1409.1556.

[18] J. Long, E. Shelhamer, T. Darrell, Fully convolutional networks for semantic segmentation, in: CVPR, 2015, pp. 3431–3440.

[19] O. Ronneberger, P. Fischer, T. Brox, U-net: Convolutional networks for biomedical image segmentation, in: MICCAI, Springer, 2015, pp. 234–241.

[20] S. Yang, P. Luo, C.-C. Loy, X. Tang, From facial parts responses to face detection: A deep learning approach, in: The IEEE International Conference on Computer Vision (ICCV), 2015.

[21] H. Qin, J. Yan, X. Li, X. Hu, Joint training of cascaded cnn for face detection, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.

[22] D. Chen, G. Hua, F. Wen, J. Sun, Supervised transformer network for efficient face detection, in: European Conference on Computer Vision, Springer, 2016, pp. 122–138.

[23] H. Jiang, E. Learned-Miller, Face detection with the faster r-cnn, in: 2017 12th IEEE International Conference on Automatic Face Gesture Recognition (FG 2017), 2017, pp. 650–657. doi:10.1109/FG.2017.82.

[24] P. Hu, D. Ramanan, Finding tiny faces, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2017, pp. 1522–1530.

[25] L. He, G. Wang, Q. Liao, J.-H. Xue, Depth-images-based pose estimation using regression forests and graphical models, Neurocomputing 164 (2015) 210–219.

[26] V. Ferrari, M. Marin-Jimenez, A. Zisserman, Progressive search space reduction for human pose estimation, in: CVPR, IEEE, 2008, pp. 1–8.

[27] M. Fergie, A. Galata, Mixtures of Gaussian process models for human pose estimation, Image and Vision Computing 31 (12) (2013) 949–957.

[28] Z. Cao, T. Simon, S.-E. Wei, Y. Sheikh, Realtime multi-person 2d pose estimation using part affinity fields, in: CVPR, 2017.

[29] B. Epshtein, S. Ullman, Semantic hierarchies for recognizing objects and parts, in: CVPR, IEEE, 2007, pp. 1–8.

[30] P. F. Felzenszwalb, D. P. Huttenlocher, Pictorial structures for object recognition, Int. J Computer Vision 61 (1) (2005) 55–79.

[31] L. He, G. Wang, Q. Liao, J.-H. Xue, Latent variable pictorial structure for human pose estimation on depth images, Neurocomputing 203 (2016) 52–61.

[32] A. Toshev, C. Szegedy, Deeppose: Human pose estimation via deep neural networks, in: CVPR, Vol. 203, 2014, pp. 1653–1660.

[33] Z. Liu, C. Zhang, Y. Tian, 3D-based deep convolutional neural network for action recognition with depth sequences, Image and Vision Computing 00 (2016) 1–10.
[34] A. Jain, J. Tompson, Y. LeCun, C. Bregler, Modeep: A deep learning framework using motion features for human pose estimation, in: ACCV, Springer, 2014, pp. 302–315.
[35] A. Newell, K. Yang, J. Deng, Stacked hourglass networks for human pose estimation, in: European Conference on Computer Vision, Springer, 2016, pp. 483–499.
[36] A. Elgammal, D. Harwood, L. Davis, Non-parametric model for background subtraction, in: ECCV, Springer, 2000, pp. 751–767.
[37] S. Javed, A. Mahmood, T. Bouwmans, S. K. Jung, Motion aware graph regularized rpca for background modeling of complex scenes, in: International Conference on Pattern Recognition (ICPR 2016), 2016.
[38] M. Piccardi, Background subtraction techniques: a review, in: Int. Conf. SMC, Vol. 4, IEEE, 2004, pp. 3099–3104.
[39] S. Javed, A. Mahmood, T. Bouwmans, K. J. Soon, Superpixels based manifold structured sparse rpca for moving object detection, in: International Workshop on Activity Monitoring by Multiple Distributed Sensing, BMVC 2017, 2017.
[40] G. Mori, X. Ren, A. A. Efros, J. Malik, Recovering human body configurations: Combining segmentation and recognition, in: CVPR, Vol. 2, IEEE, 2004, pp. II–326.
[41] C. Bhole, C. Pal, Fully automatic person segmentation in unconstrained video using spatio-temporal conditional random fields, Image and Vision Computing 51 (2016) 58–68.
[42] G. Seguin, K. Alahari, J. Sivic, I. Laptev, Pose estimation and segmentation of multiple people in stereoscopic movies, IEEE Trans. Patt. Anal. Mach. Intel. 37 (8) (2015) 1643–1655.
[43] D. Mitzel, E. Horbert, A. Ess, B. Leibe, Multi-person tracking with sparse detection and continuous segmentation, ECCV (2010) 397–410.
[44] K. Kim, L. S. Davis, Multi-camera tracking and segmentation of occluded people on ground plane using search-guided particle filtering, in: ECCV, Springer, 2006, pp. 98–109.
[45] C. Bhole, C. Pal, Automated person segmentation in videos, in: ICPR, IEEE, 2012, pp. 3672–3675.
[46] D. Kingma, J. Ba, Adam: A method for stochastic optimization, arXiv preprint arXiv:1412.6980.
[47] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, D. Ramanan, Object detection with discriminatively trained part-based models, IEEE Trans. Patt. Anal. Mach. Intel. 32 (9) (2010) 1627–1645.
[48] D. Tran, D. Forsyth, Improved human parsing with a full relational model, in: ECCV, Springer, 2010, pp. 227–240.
[49] M. Andriluka, S. Roth, B. Schiele, Pictorial structures revisited: People detection and articulated pose estimation, in: CVPR, IEEE, 2009, pp. 1014–1021.
[50] M. Eichner, V. Ferrari, S. Zurich, Better appearance models for pictorial structures., in: BMVC, Vol. 2, 2009, p. 5.
[51] B. Sapp, C. Jordan, B. Taskar, Adaptive pose priors for pictorial structures, in: CVPR, IEEE, 2010, pp. 422–429.
[52] B. Sapp, A. Toshev, B. Taskar, Cascaded models for articulated pose estimation, in: ECCV, Springer, 2010, pp. 406–420.
[53] R. Mehran, A. Oyama, M. Shah, Abnormal crowd behavior detection using social force model, in: CVPR, IEEE, 2009, pp. 935–942.
[54] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in: CVPR, Vol. 1, IEEE, 2005, pp. 886–893.
[55] X. Zhou, C. Yang, W. Yu, Moving object detection by detecting contiguous outliers in the low-rank representation, IEEE Trans. Patt. Anal. Mach. Intel. 35 (3) (2013) 597–610.
[56] P. Rodriguez, B. Wohlberg, Fast principal component pursuit via alternating minimization, in: ICIP, IEEE, 2013, pp. 69–73.
[57] T. Zhou, D. Tao, Godec: Randomized low-rank & sparse matrix decomposition in noisy case, in: ICML, 2011, pp. 33–40.
[58] X. Ye, J. Yang, X. Sun, K. Li, C. Hou, Y. Wang, Foreground–background separation from video clips via motion-assisted matrix restoration, IEEE Trans. CSVT 25 (11) (2015) 1721–1734.
[59] H. Chen, X. Qi, L. Yu, P. Heng, DCAN: deep contour-aware networks for accurate gland segmentation, Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, (2016) 2487–2496.
[60] K. He, X. Z. Kaiming, S. Ren, J. Sun, Deep residual learning for image recognition, Proceedings of the IEEE conference on computer vision and pattern recognition, (2016), 770–778.