Switching Convolutional Neural Network for Crowd Counting

Deepak Babu Sam* Shiv Surya* R. Venkatesh Babu
Indian Institute of Science
Bangalore, INDIA 560012
bsdeepak@grads.cds.iisc.ac.in, shiv.surya314@gmail.com, venky@cds.iisc.ac.in

Abstract

We propose a novel crowd counting model that maps a given crowd scene to its density. Crowd analysis is compounded by myriad of factors like inter-occlusion between people due to extreme crowding, high similarity of appearance between people and background elements, and large variability of camera view-points. Current state-of-the art approaches tackle these factors by using multi-scale CNN architectures, recurrent networks and late fusion of features from multi-column CNN with different receptive fields. We propose switching convolutional neural network that leverages variation of crowd density within an image to improve the accuracy and localization of the predicted crowd count. Patches from a grid within a crowd scene are relayed to independent CNN regressors based on crowd count prediction quality of the CNN established during training. The independent CNN regressors are designed to have different receptive fields and a switch classifier is trained to relay the crowd scene patch to the best CNN regressor. We perform extensive experiments on all major crowd counting datasets and evidence better performance compared to current state-of-the-art methods. We provide interpretable representations of the multichotomy of space of crowd scene patches inferred from the switch. It is observed that the switch relays an image patch to a particular CNN column based on density of crowd.

1. Introduction

Crowd analysis has important geo-political and civic applications. Massive crowd gatherings are commonplace at candle-light vigils, democratic protests, religious gatherings and presidential rallies. Civic agencies and planners rely on crowd estimates to regulate access points and plan disaster contingency for such events. Critical to such analysis is crowd count and density.

In principle, the key idea behind crowd counting is self-evident: density times area. However, crowds are not regular across the scene. They cluster in certain regions and are spread out in others. Typical static crowd scenes from the ShanghaiTech Dataset [19] are shown in Figure 1. We see extreme crowding, high visual resemblance between people and background elements (e.g. Urban facade) in these crowd scenes that factors in further complexity. Different camera view-points in various scenes create perspective effects resulting in large variability of scales of people.

Figure 1. Sample crowd scenes from the ShanghaiTech dataset [19] is shown.

Crowd counting as a computer vision problem has seen drastic changes in the approaches, from early HOG based head detections [6] to CNN regressors [18, 19, 9] predicting the crowd density. CNN based regressors have largely outperformed traditional crowd counting approaches based on weak representations from local features. We build on the performance of CNN based architectures for crowd counting and propose Switching Convolutional Neural Network (**Switch-CNN**) to map a given crowd scene to its density.

Switch-CNN leverages the variation of crowd density within an image to improve the quality and localization of the predicted crowd count. Independent CNN crowd density regressors are trained on patches sampled from a grid in a given crowd scene. The independent CNN regressors are chosen such that they have different receptive fields and field of view. This ensures that the features learned by each CNN regressor are adapted to a particular scale. This ren-

*Equal contribution

ders Switch-CNN robust to large scale and perspective variations of people observed in a typical crowd scene. A particular CNN regressor is trained on a crowd scene patch if the performance of the regressor on the patch is the best. A switch classifier is trained alternately with the training of multiple CNN regressors to correctly relay a patch to a particular regressor. The joint training of the switch and regressors helps augment the ability of the switch to learn the complex multichotomy of space of crowd scenes learnt in the differential training stage.

To summarize, in this paper we present:

- A novel generic CNN architecture, Switch-CNN trained end-to-end to predict crowd density for a crowd scene.
- Switch-CNN maps crowd patches from a crowd scene to independent CNN regressors to minimize count error and improve density localization exploiting the density variation within a scene.
- We evidence state-of-the-art performance on all major crowd counting datasets including ShanghaiTech dataset [19], UCF_CC_50 dataset [6] and WorldExpo'10 dataset [18].

2. Related Work

Crowd counting has been tackled in computer vision by a myriad of techniques. Crowd counting via head detections has been tackled by [17, 16, 14] using motion cues and appearance features to train detectors. Recurrent network framework has been used for head detections in crowd scenes by [12]. They use the deep features from Googlenet [13] in an LSTM framework to regress bounding boxes for heads in a crowd scene. However, crowd counting using head detections has limitations as it fails in dense crowds, which are characterized by high inter-occlusion between people.

In crowd counting from videos, [3] use image features like Tomasi-Kanade features into a motion clustering framework. Video is processed by [10] into a set of trajectories using a KLT tracker. To prevent fragmentation of trajectories, they condition the signal temporally and spatially. Such tracking methods are unlikely to work for single image crowd counting due to lack of temporal information.

Early works in still image crowd counting like [6] employ a combination of handcrafted features, namely HOG based detections, interest points based counting and Fourier analysis. These weak representations based on local features are outperformed by modern deep representations. In [18], CNNs are trained to regress the crowd density map. They retrieve images from the training data similar to a test image using density and perspective information as the similarity metric. The retrieved images are used to fine-tune the trained network for a specific target test scene and the density map is predicted. However, the model's applicability is limited by fine-tuning required for each test scene and perspective maps for train and test sequences which are not readily available. An Alexnet [7] style CNN model is trained by [15] to regress the crowd count. However, the application of such a model is limited for crowd analysis as it does not predict the distribution of the crowd. In [9], a multi-scale CNN architecture is used to tackle the large scale variations in crowd scenes. They use a custom CNN network, trained separately for each scale. Fully-connected layers are used to fuse the maps from each of the CNN trained at a particular scale, and regress the density map. However, the counting performance of this model is sensitive to the number of levels in the image pyramid as indicated by performance across datasets.

Multi-column CNN used by [2, 19] perform late fusion of features from different CNN columns to regress the density map for a crowd scene. In [19], shallow CNN columns with varied receptive fields are used to capture the large variation in scale and perspective in crowd scenes. Transfer learning is employed by [2] using a VGG network employing dilated layers complemented by a shallow network with different receptive field and field of view. Both the model fuse the feature maps from the CNN columns by weighted averaging via a 1×1 convolutional layer to predict the density map of the crowd. However, the weighted averaging technique is global in nature and does not take in to account the intra-scene density variation. We build on the performance of multi-column CNN and incorporate a patch based switching architecture in our proposed architecture, Switch-CNN to exploit local crowd density variation within a scene (see Sec 3.1 for more details of architecture).

3. Our Approach

Convolutional architectures like [18, 19, 9] have learnt effective image representations, which they leverage to perform crowd counting and density prediction in a regression framework. Traditional convolutional architectures have been modified to model the extreme variations in scale induced in dense crowds by using multi-column CNN architectures with feature fusion techniques to regress crowd density.

In this paper, we consider switching CNN architecture (Switch-CNN) that relays patches from a grid within a crowd scene to independent CNN regressors based on a switch classifier. The independent CNN regressors are chosen with different receptive fields and field-of-view as in multi-column CNN networks to augment the ability to model large scale variations. A particular CNN regressor is trained on a crowd scene patch if the performance of the regressor on the patch is the best. A switch classifier is trained alternately with the training of multiple CNN regressors to correctly relay a patch to a particular regressor. The salient properties that make this model excellent for crowd anal-

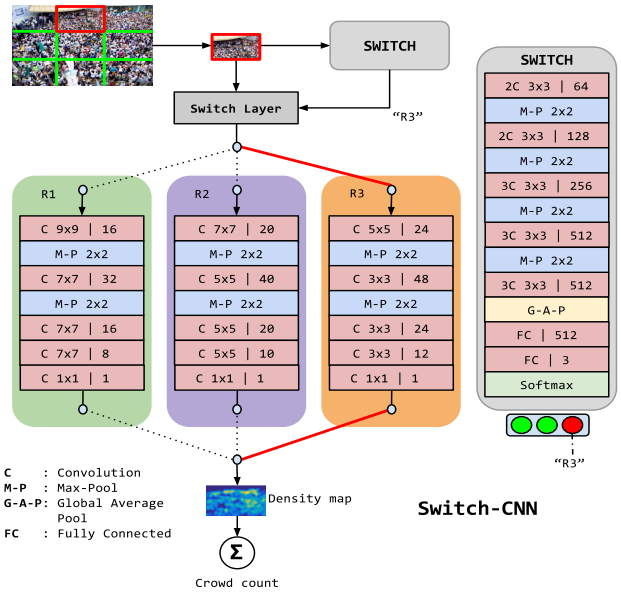

Figure 2. Architecture of the proposed model, **Switch-CNN** is shown. A patch from the crowd scene is highlighted in red. This patch is relayed to one of the three CNN regressor networks based on the CNN label inferred from **Switch**. The highlighted patch is relayed to regressor R_3 which predicts the corresponding crowd density map. The element-wise sum over the entire density map gives the crowd count of the crowd scene patch.

ysis are (1) the ability to model large scale variations (2) the facility to leverage local variations in density within a crowd scene. The ability to leverage local variations in density is important as the weighted averaging technique used in multi-column networks to fuse the features is global in nature.

3.1. Switch-CNN

Our proposed architecture, Switch-CNN consists of three CNN regressors with different architectures and a classifier (switch) to select the optimal regressor for an input crowd scene patch. Figure 2 shows the overall architecture of Switch-CNN. The input image is divided into 9 non-overlapping patches such that each patch is $\frac{1}{3}^{\text{rd}}$ of the image. For such a division of the image, crowd characteristics like density, appearance etc. can be assumed to be consistent in a given patch for a crowd scene. Feeding patches as input to the network helps in regressing different regions of the image independently by a CNN regressor most suited to patch attributes like density, background, scale and perspective variations of crowd in the patch.

We use three CNN regressors introduced in [19], R_1 through R_3, in Switch-CNN to predict the density of crowd. These CNN regressors have varying receptive fields that can capture people at different scales. The architecture of each of the shallow CNN regressor is similar: four convolutional layers with two pooling layers. R_1 has a large initial filter

input : N training image patches $\{X_i\}_{i=1}^{N}$ with ground truth density maps $\{D_{X_i}^{GT}\}_{i=1}^{N}$
output: Trained parameters $\{\Theta_k\}_{k=1}^{3}$ for R_k and Θ_{sw} for the switch

Initialize $\Theta_k \; \forall \; k$ with random Gaussian weights

Pretrain $\{R_k\}_{k=1}^{3}$ for T_p epochs : $R_k \leftarrow f_k(\cdot;\Theta_k)$;

/*Differential Training for T_d epochs*/
/*C_i^k is count predicted by R_k for input X_i*/
/*C_i^{GT} is ground truth count for input X_i*/
for t = *1 to* T_d **do**
  for i = *1 to* N **do**
    $l_i^{best} = \underset{k}{\text{argmin}} |C_i^k - C_i^{GT}|$;
    Backpropagate $R_{l_i^{best}}$ and update $\Theta_{l_i^{best}}$;
  end
end

/*Coupled Training for T_c epochs*/
Initialize Θ_{sw} with VGG-16 weights ;
for t = *1 to* T_c **do**
  /*generate labels for training switch*/
  for i = *1 to* N **do**
    $l_i^{best} = \underset{k}{\text{argmin}} |C_i^k - C_i^{GT}|$;
  end
  $S_{train} = \{(X_i, l_i^{best}) \mid i \in [1, N]\}$
  /*Training switch for 1 epoch*/
  Train switch with S_{train} and update Θ_{sw};
  /*Switched Differential Training*/
  for i = *1 to* N **do**
    /*Infer choice of R_k from switch*/
    $l_i^{sw} = \text{argmax } f_{switch}(X_i; \Theta_{sw})$;
    Backpropagate $R_{l_i^{switch}}$ and update $\Theta_{l_i^{sw}}$;
  end
end

Algorithm 1: Switch-CNN training algorithm is shown. The training algorithm is divided into stages coded by color. **Color code index**: Differential Training, Coupled Training, Switch Training

size of 9×9 which can capture high level abstractions within the scene like faces, urban facade etc. R_2 and R_3 with initial filter sizes 7×7 and 5×5 capture crowds at lower scales detecting blob like abstractions.

Patches are relayed to a regressor using a switch. The switch consists of a switch classifier and a switch layer. The switch classifier infers the label of the regressor to which the patch is to be relayed to. A switch layer takes the label inferred from the switch classifier and relays it to the correct regressor. For example, in Figure 2, the switch classifier relays the patch highlighted in red to regressor R_3. The patch has a very high crowd density. Switch relays it to regressor R_3 which has smaller receptive field: ideal for detecting blob like abstractions characteristic of patches with

high crowd density. We use an adaptation of VGG16 [11] network as the switch classifier to perform 3-way classification. The fully-connected layers in VGG16 are removed. We use global average pool (GAP) on Conv5 features to remove the spatial information and aggregate discriminative features. GAP is followed by a smaller fully connected layer and 3-class softmax classifier corresponding to the three regressor networks in Switch-CNN.

Ground Truth Annotations for crowd images are provided as point annotations at the center of the head of a person. We generate our ground truth by blurring each head annotation with a Gaussian kernel normalized to sum to one to generate a density map. Summing the resultant density map gives the crowd count. Density maps ease the difficulty of regression for the CNN as the task of predicting the exact point of head annotation is reduced to predicting a coarse location. The spread of the Gaussian in the above density map is fixed. However, a density map generated from a fixed spread Gaussian is inappropriate if the variation in crowd density is large. We use geometry-adaptive kernels[19] to vary the spread parameter of the Gaussian depending on the local crowd density. It sets the spread of Gaussian in proportion to the average distance of k-nearest neighboring head annotations. The inter-head distance is a good substitute for perspective maps which are laborious to generate and unavailable for every dataset. This results in lower degree of Gaussian blur for dense crowds and higher degree for region of sparse density in crowd scene. In our experiments, we use both geometry-adaptive kernel method as well as fixed spread Gaussian method to generate ground truth density depending on the dataset. Geometry-adaptive kernel method is used to generate ground truth density maps for datasets with dense crowds and large variation in count across scenes. Datasets that have sparse crowds are trained using density maps generated from fixed spread Gaussian method.

Training of Switch-CNN is done in three stages, namely pretraining, differential training and coupled training described in Sec 3.2–3.5.

3.2. Pretraining

The three CNN regressors R_1 through R_3 are pretrained separately to regress density maps. Pretraining helps in learning good initial features which improves later fine-tuning stages. Individual CNN regressors are trained to minimize the Euclidean distance between the estimated density map and ground truth. Let $D_{X_i}(\cdot;\Theta)$ represent the output of a CNN regressor with parameters Θ for an input image X_i. The l_2 loss function is given by

$$L_{l_2}(\Theta) = \frac{1}{2N}\sum_{i=1}^{N} \|D_{X_i}(\cdot;\Theta) - D_{X_i}^{GT}(\cdot)\|_2^2, \qquad (1)$$

where N is the number of training samples and $D_{X_i}^{GT}(\cdot)$ indicates ground truth density map for image X_i. The loss L_{l_2} is optimized by backpropagating the CNN via stochastic gradient descent (SGD). Here, l_2 loss function acts as a proxy for count error between the regressor estimated count and true count. It indirectly minimizes count error. The regressors R_k are pretrained until the validation accuracy plateaus.

3.3. Differential Training

CNN regressors R_{1-3} are pretrained with the entire training data. The count prediction performance varies due to the inherent difference in network structure of R_{1-3} like receptive field and effective field-of-view. Though we optimize the l_2-loss between the estimated and ground truth density maps for training CNN regressor, factoring in count error during training leads to better crowd counting performance. Hence, we measure CNN performance using count error. Let the count estimated by kth regressor for ith image be $C_i^k = \sum_x D_{X_i}(x;\Theta_k)$. Let the reference count inferred from ground truth be $C_i^{GT} = \sum_x D_{X_i}^{GT}(x)$. Then count error for ith sample evaluated by R_k is

$$E_{C_i}(k) = |C_i^k - C_i^{GT}|, \qquad (2)$$

the absolute count difference between prediction and true count. Patches with particular crowd attributes give lower count error with a regressor having complementary network structure. For example, a CNN regressor with large receptive field capture high level abstractions like background elements and faces. To amplify the network differences, differential training is proposed (shown in blue in Algorithm 1). The key idea in differential training is to backpropagate the regressor R_k with minimum count error for a given training crowd scene patch. For every training patch i, we choose the regressor l_i^{best} such that $E_{C_i}(l_i^{best})$ is lowest across all regressors R_{1-3}. This amounts to greedily choosing the regressor that predicts the most accurate count amongst k regressors. Formally, we define the label of chosen regressor l_i^{best} as:

$$l_i^{best} = \underset{k}{\text{argmin}} |C_i^k - C_i^{GT}| \qquad (3)$$

The count error for ith sample is

$$E_{C_i} = \min_k |C_i^k - C_i^{GT}|. \qquad (4)$$

This training regime encourages a regressor R_k to prefer a particular set of the training data patches with particular patch attribute so as to minimize the loss. While the backpropagation of independent regressor R_k is still done with l_2-loss, the choice of CNN regressor for backpropagation is based on the count error. Differential training indirectly minimizes the mean absolute count error (MAE) over the

training images. For N images, MAE in this case is given by

$$E_C = \frac{1}{N}\sum_{i=1}^{N}\min_k |C_i^k - C_i^{GT}|, \quad (5)$$

which can be thought as the minimum count error achievable if each sample is relayed correctly to the right CNN. However during testing, achieving this full accuracy may not be possible as the switch classifier is not ideal. To summarize, differential training generates three disjoint groups of training patches and each network is finetuned on its own group. The regressors R_k are differentially trained until the validation accuracy plateaus.

3.4. Switch Training

Once the multichotomy of space of patches is inferred via differential training, a patch classifier (switch) is trained to relay a patch to the correct regressor R_k. The manifold that separates the space of crowd scene patches is complex and hence a deep classifier is required to infer the group of patches in the multichotomy. We use VGG16 [11] network as the switch classifier to perform 3-way classification. The classifier is trained on the labels of multichotomy generated from differential training. The number of training patches in each group can be highly skewed, with the majority of patches being relayed to a single regressor depending on the attributes of crowd scene. To alleviate class imbalance during switch classifier training, the labels collected from the differential training are equalized so that the number of samples in each group is the same. This is done by randomly sampling from the smaller group to balance the training set of switch classifier.

3.5. Coupled Training

Differential training on the CNN regressors R_1 through R_3 generates a multichotomy that minimizes the predicted count by choosing the best regressor for a given crowd scene patch. However, the trained switch is not ideal and the manifold separating the space of patches is complex to learn. To mitigate the effect of switch inaccuracy and inherent complexity of task, we co-adapt the patch classifier and the CNN regressors by training the switch and regressors in an alternating fashion. We refer to this stage of training as *Coupled training* (shown in green in Algorithm 1).

The switch classifier is first trained with labels from the multichotomy inferred in differential training for one epoch (shown in red in Algorithm 1). In, the next stage, the three CNN regressors are made to co-adapt with switch classifier (shown in blue in Algorithm 1). We refer to this stage of training enforcing co-adaption of switch and regressor R_{1-3} as *Switched differential training*.

In switched differential training, the individual CNN regressors are trained using crowd scene patches relayed by switch for one epoch. For a given training crowd scene patch X_i, switch is forward propagated on X_i to infer the choice of regressor R_k. The switch layer then relays X_i to the particular regressor and backpropagates R_k using the loss defined in Equation 1 and θ_k is updated. This training regime is executed for an epoch.

In the next epoch, the labels for training the switch classifier are recomputed using criterion in Equation 3 and the switch is again trained as described above. This process of alternating switch training and switched training of CNN regressors is repeated every epoch until the validation accuracy plateaus.

4. Experiments

4.1. Testing

We evaluate the performance of our proposed architecture, Switch-CNN on four major crowd counting datasets At test time, the image patches are fed to the switch classifier which relays the patch to the best CNN regressor R_k. The selected CNN regressor predicts a crowd density map for the relayed crowd scene patch. The generated density maps are assembled into an image to get the final density map for the entire scene. Because of the two pooling layers in the CNN regressors, the predicted density maps are $\frac{1}{4}^{\text{th}}$ size of the input.

Evaluation Metric We use Mean Absolute Error (MAE) and Mean Squared Error (MSE) as the metric for comparing the performance of Switch-CNN against the state-of-the-art crowd counting methods. For a test sequence with N images, MAE is defined as follows:

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N}|C_i - C_i^{GT}|, \quad (6)$$

where C_i is the crowd count predicted by the model being evaluated, and C_i^{GT} is the crowd count from human labelled annotations. MAE is an indicator of the accuracy of the predicted crowd count across the test sequence. MSE is a metric complementary to MAE and indicates the robustness of the predicted count. For a test sequence, MSE is defined as follows:

$$\text{MSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(C_i - C_i^{GT})^2}. \quad (7)$$

4.2. ShanghaiTech dataset

We perform extensive experiments on the ShanghaiTech crowd counting dataset [19] that consists of 1198 annotated images. The dataset is divided into two parts named Part A and Part B. The former contains dense crowd scenes parsed from the internet and the latter is relatively sparse crowd

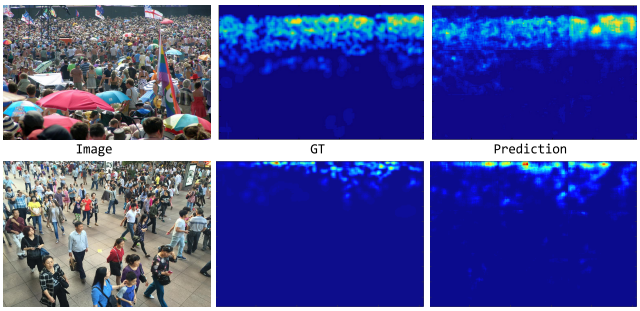

Figure 3. Sample predictions by Switch-CNN for crowd scenes from the ShanghaiTech dataset [19] is shown. The top and bottom rows depict a crowd image, corresponding ground truth and prediction from Part A and Part B of dataset respectively.

scenes captured in urban surface streets. We use the train-test splits provided by the authors for both parts in our experiments. We train Switch-CNN as elucidated by Algorithm 1 on both parts of the dataset. Ground truth is generated using geometry-adaptive kernels method as the variance in crowd density within a scene due to perspective effects is high (See Sec 3.1 for details about ground truth generation). With an ideal switch (100% switching accuracy), Switch-CNN performs with an MAE of 51.4. However, the accuracy of the switch is 73.2% in Part A and 76.3% in Part B of the dataset resulting in a lower MAE.

Table 1 shows that Switch-CNN outperforms all other state-of-the art methods by a significant margin on both the MAE and MSE metric. Switch-CNN shows a 19.8 point improvement in MAE on Part A and 4.8 point improvement in Part B of the dataset over MCNN [19]. Switch-CNN also outperforms all other models on MSE metric indicating that the predictions have a lower variance than MCNN across the dataset. This is an indicator of the robustness of Switch-CNN's predicted crowd count.

We show sample predictions of Switch-CNN for sample test scenes from the ShanghaiTech dataset along with the ground truth in Figure 3. The predicted density maps closely follow the crowd distribution visually. This indicates that Switch-CNN is able to localize the spatial distribution of crowd within a scene accurately.

	Part A		Part B	
Method	MAE	MSE	MAE	MSE
Zhang et al. [18]	181.8	277.7	32.0	49.8
MCNN [19]	110.2	173.2	26.4	41.3
Switch-CNN	**90.4**	**135.0**	**21.6**	**33.4**

Table 1. Comparison of Switch-CNN with other state-of-the-art crowd counting methods on ShanghaiTech dataset [19].

4.3. UCF_CC_50 dataset

UCF_CC_50 [6] is a 50 image collection of annotated crowd scenes. The dataset exhibits a large variance in the crowd count with counts varying between 94 and 4543. The small size of the dataset and large variance in crowd count makes it a very challenging dataset. We follow the approach of other state-of-the-art models [18, 2, 9, 19] and use 5-fold cross-validation to validate the performance of Switch-CNN on UCF_CC_50.

In Table 2, we compare the performance of Switch-CNN with other methods using MAE and MSE as metrics. Switch-CNN outperforms all other methods and evidences a 15.7 point improvement in MAE over Hydra2s [9]. Switch-CNN also gets a competitive MSE score compared to Hydra2s indicating the robustness of the predicted count. The accuracy of the switch is 54.3%. The switch accuracy is relatively low as the dataset has very few training examples and a large variation in crowd density. This limits the ability of the switch to learn the multichotomy of space of crowd scene patches.

Method	MAE	MSE
Lempitsky et al.[8]	493.4	487.1
Idrees et al.[6]	419.5	487.1
Zhang et al. [18]	467.0	498.5
CrowdNet [2]	452.5	–
MCNN [19]	377.6	509.1
Hydra2s [9]	333.73	**425.26**
Switch-CNN	**318.1**	439.2

Table 2. Comparison of Switch-CNN with other state-of-the-art crowd counting methods on UCF_CC_50 dataset [6].

4.4. The UCSD dataset

The UCSD dataset crowd counting dataset consists of 2000 frames from a single scene. The scenes are characterized by sparse crowd with the number of people ranging from 11 to 46 per frame. A region of interest (ROI) is provided for the scene in the dataset. We use the train-test splits used by [4]. Of the 2000 frames, frames 601 through 1400 are used for training while the remaining frames are held out for testing. Following the setting used in [19], we prune the feature maps of the last layer with the ROI provided. Hence, error is backpropagated during training for areas inside the ROI. We use a fixed spread Gaussian to generate ground truth density maps for training Switch-CNN as the crowd is relatively sparse. At test time, MAE is computed only for the specified ROI in test images for benchmarking Switch-CNN against other approaches.

Table 3 reports the MAE and MSE results for Switch-CNN and other state-of-the-art approaches. Switch-CNN performs competitively compared to other approaches with an MAE of 1.62. The switch accuracy in relaying the patches to regressors R_1 through R_3 is 60.9%. However, the dataset is characterized by low variability of crowd density set in a single scene. This limits the performance gain achieved by Switch-CNN from leveraging intra-scene crowd density variation.

Method	MAE	MSE
Kernel Ridge Regression [1]	2.16	7.45
Cumulative Attribute Regression [5]	2.07	6.86
Zhang et al. [18]	1.60	3.31
MCNN [19]	**1.07**	**1.35**
CCNN [9]	1.51	–
Switch-CNN	1.62	2.10

Table 3. Comparison of Switch-CNN with other state-of-the-art crowd counting methods on UCSD crowd-counting dataset [4].

Method	S1	S2	S3	S4	S5	Avg. MAE
Zhang et al. [18]	9.8	**14.1**	14.3	22.2	**3.7**	12.9
MCNN [19]	**3.4**	20.6	12.9	13.0	8.1	11.6
Switch-CNN (GT with perspective map)	4.2	14.9	14.2	18.7	4.3	11.2
Switch-CNN (GT without perspective)	4.4	15.7	**10.0**	**11.0**	5.9	**9.4**

Table 4. Comparison of Switch-CNN with other state-of-the-art crowd counting methods on WorldExpo'10 dataset [18]. Mean Absolute Error (MAE) for individual test scenes and average performance across scenes is shown.

4.5. The WorldExpo'10 dataset

The WorldExpo'10 dateset consists of 1132 video sequences captured with 108 surveillance cameras. Five different video sequence, each from a different scene, are held out for testing. Every test scene sequence has 120 frames. The crowds are relatively sparse in comparison to other datasets with average number of 50 people per image. Region of interest (ROI) is provided for both training and test scenes. In addition, perspective maps are provided for all scenes. The maps specify the number of pixels in the image that cover one square meter at every location in the frame. These maps are used by [19, 18] to adaptively choose the spread of the Gaussian while generating ground truth density maps. We evaluate performance of the Switch-CNN using ground truth generated with and without perspective maps.

We prune the feature maps of the last layer with the ROI provided. Hence, error is backpropagated during training for areas inside the ROI. Similarly at test time, MAE is computed only for the specified ROI in test images for benchmarking Switch-CNN against other approaches.

MAE is computed separately for each test scene and averaged to determine the overall performance of Switch-CNN across test scenes. Table 4 shows that the average MAE of Switch-CNN across scenes is better by a margin of 2.2 point over the performance obtained by the state-of-the-art approach MCNN [19]. The switch accuracy is 52.72%.

5. Analysis

5.1. Effect of number of regressors on Switch-CNN

Differential training makes use of the structural variations across the individual regressors to learn a multichotomy of the training data. To investigate the effect of structural variations of the regressors R_1 through R_3, we train Switch-CNN with combinations of regressors (R_1,R_2), (R_2,R_3), (R_1,R_3) and (R_1,R_2,R_3) on Part A of ShanghaiTech dataset. Table 5 shows the MAE performance of Switch-CNN for different combinations of regressors R_k. Switch-CNN with CNN regressors R_1 and R_3 has lower MAE than Switch-CNN with regressors R_1–R_2 and R_2–R_3. This can be attributed to the former model having a higher switching accuracy than the latter. Switch-CNN with all three regressors outperforms both the models as it is able to model the scale and perspective variations better with three independent CNN regressors R_1, R_2 and R_3 that are structurally distinct. Switch-CNN leverages multiple independent CNN regressors with different receptive fields. In Table 5, we also compare the performance of individual CNN regressors with Switch-CNN. Here each of the individual regressors are trained on the full training data from Part A of Shanghaitech dataset. The higher MAE of the individual CNN regressor is attributed to the inability of a single regressor to model the scale and perspective variations in the crowd scene.

Method	MAE
R_1	157.61
R_2	178.82
R_3	178.10
Switch-CNN with (R_1,R_3)	98.87
Switch-CNN with (R_1,R_2)	110.88
Switch-CNN with (R_2,R_3)	126.65
Switch-CNN with (R_1,R_2,R_3)	**90.41**

Table 5. Comparison of MAE for Switch-CNN variants and CNN regressors R_1 through R_3 on Part A of the ShanghaiTech dataset [19].

5.2. Switch Multichotomy Characteristics

The principal idea of Switch-CNN is to divide the training patches into disjoint groups to train individual CNN regressors so that overall count accuracy is maximized. This multichotomy in space of crowd scene patches is created automatically through differential training. We examine the underlying structure of the patches to understand the correlation between the learnt multichotomy and attributes of the patch like crowd count and density. However, the unavailability of perspective maps renders computation of actual density intractable. We believe inter-head distance between people is a candidate measure of crowd density. In a highly dense crowd, the separation between people is low and hence density is high. On the other hand, for low density scenes, people are far away and mean inter-head distance is large. Thus mean inter-head distance is a proxy for crowd density. This measure of density is robust to scale variations as the inter-head distance naturally subsumes the scale variations.

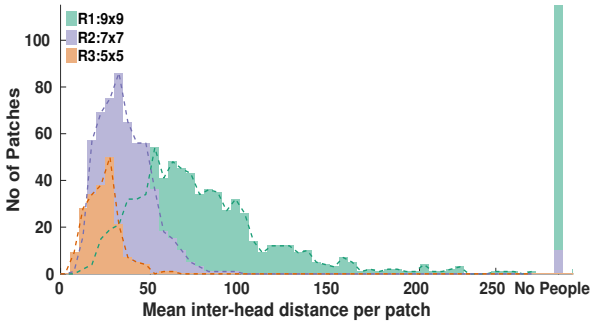

Figure 4. Histogram of average inter-head distance for crowd scene patches from Part A test set of ShanghaiTech dataset [19] is shown in Figure 4. We see that the multichotomy of space of crowd scene patches inferred from the switch separates patches based on latent factors correlated with crowd density.

Figure 5. Sample crowd scene patches from Part A test set of ShanghaiTech dataset [19] are shown. We see that the density of crowd in the patches increases from CNN regressor R_1–R_3.

To analyze the multichotomy in space of patches, we compute the average inter-head distance of each patch in Part A of ShanghaiTech test set. For each head annotation, the average distance to its 10 nearest neighbors is calculated. These distances are averaged over the entire patch representing the density of the patch. We plot a histogram of these distances in Figure 4 and group the patches by color on the basis of the regressor R_k used to infer the count of the patch. A separation of patch space based on crowd density is observed in Figure 4. R_1, which has the largest receptive field of 9×9, evaluates patches of low crowd density (corresponding to large mean inter-head distance). An interesting observation is that patches from the crowd scene that have no people in them (patches in Figure 4 with zero average inter-head distance) are relayed to R_1 by the switch. We believe that the patches with no people are relayed to R_1 as it has a large receptive field that helps capture background attributes in such patches like urban facade and foliage. Figure 5 displays some sample patches that are relayed to each of the CNN regressors R_1 through R_3. The density of crowd in the patches increases from CNN regressor R_1 through R_3.

5.3. Attribute Clustering Vs Differential Training

We saw in Sec 5.2 that differential training approximately divides training set patches into a multichotomy based on density. We investigate the effect of manually clustering the patches based on patch attribute like crowd count or density. We use patch count as metric to cluster patches. Training patches are divided into three groups based on the patch count such that the total number of training patches are equally distributed amongst the three CNN regressors R_{1-3}. R_1, having a large receptive field, is trained on patches with low crowd count. R_2 is trained on medium count patches while high count patches are relayed to R_3. The training procedure for this experiment is identical to Switch-CNN, except for the differential training stage. We repeat this experiment with average inter-head distance of the patches as a metric for grouping the patches. Patches with high mean inter-head distance are relayed to R_1. R_2 is relayed patches with low inter-head distance by the switch while the remaining patches are relayed to R_3.

Method	MAE
Cluster by count	99.56
Cluster by mean inter-head distance	94.93
Switch-CNN	**90.41**

Table 6. Comparison of MAE for Switch-CNN and manual clustering of patches based on patch attributes on Part A of the ShanghaiTech dataset [19].

Table 6 reports MAE performance for the two clustering methods. Both crowd count and average inter-head distance based clustering give a higher MAE than Switch-CNN. Average inter-head distance based clustering performs comparably with Switch-CNN. This evidence reinforces the fact that Switch-CNN learns a multichotomy in the space of patches that is highly correlated with mean inter-head distance of the crowd scene. The differential training regime employed by Switch-CNN is able to infer this grouping automatically, independent of the dataset.

6. Conclusion

In this paper, we propose switching convolutional neural network that leverages intra-image crowd density variation to improve the accuracy and localization of the predicted crowd count. We utilize the inherent structural and functional differences in multiple CNN regressors capable of tackling large scale and perspective variations by enforcing a differential training regime. Extensive experiments on multiple datasets show that our model exhibits state-of-the-art performance on major datasets. Further, we show that our model learns to group crowd patches based on latent factors correlated with crowd density.

7. Acknowledgements

This work was supported by SERB, Department of Science and Technology (DST), Government of India (Proj No. SB/S3/EECE/0127/2015).

References

[1] S. An, W. Liu, and S. Venkatesh. Face recognition using kernel ridge regression. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–7, 2007. 4.4

[2] L. Boominathan, S. S. Kruthiventi, and R. V. Babu. Crowdnet: A deep convolutional network for dense crowd counting. In *Proceedings of the 2016 ACM on Multimedia Conference*, pages 640–644, 2016. 2, 4.3

[3] G. J. Brostow and R. Cipolla. Unsupervised bayesian detection of independent motion in crowds. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 594–601, 2006. 2

[4] A. B. Chan, Z.-S. J. Liang, and N. Vasconcelos. Privacy preserving crowd monitoring: Counting people without people models or tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–7, 2008. 4.4, 3

[5] K. Chen, C. C. Loy, S. Gong, and T. Xiang. Feature mining for localised crowd counting. In *BMVC*, volume 1, page 3, 2012. 4.4

[6] H. Idrees, I. Saleemi, C. Seibert, and M. Shah. Multi-source multi-scale counting in extremely dense crowd images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2547–2554, 2013. 1, 2, 4.3, 2

[7] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, pages 1097–1105, 2012. 2

[8] V. Lempitsky and A. Zisserman. Learning to count objects in images. In *Advances in Neural Information Processing Systems*, pages 1324–1332, 2010. 4.3

[9] D. Onoro-Rubio and R. J. López-Sastre. Towards perspective-free object counting with deep learning. In *European Conference on Computer Vision*, pages 615–629. Springer, 2016. 1, 2, 3, 4.3, 4.4

[10] V. Rabaud and S. Belongie. Counting crowded moving objects. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 705–711, 2006. 2

[11] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 3.1, 3.4

[12] R. Stewart and M. Andriluka. End-to-end people detection in crowded scenes. *arXiv preprint arXiv:1506.04878*, 2015. 2

[13] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–9, 2015. 2

[14] P. Viola, M. J. Jones, and D. Snow. Detecting pedestrians using patterns of motion and appearance. *International Journal of Computer Vision*, 63(2):153–161, 2005. 2

[15] C. Wang, H. Zhang, L. Yang, S. Liu, and X. Cao. Deep people counting in extremely dense crowds. In *Proceedings of the 2015 ACM on Multimedia Conference*, pages 1299–1302, 2015. 2

[16] M. Wang and X. Wang. Automatic adaptation of a generic pedestrian detector to a specific traffic scene. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3401–3408, 2011. 2

[17] B. Wu and R. Nevatia. Detection of multiple, partially occluded humans in a single image by bayesian combination of edgelet part detectors. In *IEEE International Conference on Computer Vision*, volume 1, pages 90–97, 2005. 2

[18] C. Zhang, H. Li, X. Wang, and X. Yang. Cross-scene crowd counting via deep convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 833–841, 2015. 1, 2, 3, 4.2, 4.3, 4.4, 4.5, 4, 4.5

[19] Y. Zhang, D. Zhou, S. Chen, S. Gao, and Y. Ma. Single-image crowd counting via multi-column convolutional neural network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 589–597, 2016. 1, 1, 2, 3, 3.1, 4.2, 3, 4.2, 1, 4.3, 4.4, 4.5, 4.5, 5, 4, 5, 6