Pedestrian Attribute Detection using CNN

Agrim Gupta
Stanford University
agrim@stanford.edu

Jayanth Ramesh
Stanford University
jayanth7@stanford.edu

Abstract

Learning to determine the attributes of pedestrian using their far-view field images is a challenging problem in visual surveillance. Many previous works have focused on the problem of pedestrian detection. Traditionally SVMs have been a popular choice for pedestrian attribute recognition, however recently there has been interest in using CNNs for this task. In this project we implement traditional methods like multi-label SVM and softmax classifiers to predict attributes and compare their performance to ConvNets. We propose two models - ShallowMAR and ShallowSAR, where ShallowSAR treats the attributes independently, while ShallowMAR considers the attributes in a unified framework allowing to exploit the interdependence between attributes. We experiment with different architectures of convolutional neural networks and try to understand their performance on the PETA dataset, which consists of far-view images of pedestrians.

1. Introduction

The capability of recognizing pedestrian attributes such as gender, age, clothing style and such others at far distance is of practical interest in far-view surveillance scenarios where face and body close-shots are hardly available. Thus, attribute recognition has to be performed using full body appearance in the absence of critical face/close-shot body visual information. Such attribute predictions are interesting for a range of applications like image retrieval, querying databases by semantic propositions, tracking-by-detection, re-identification applications, and robotic applications that require semantic information of persons for interaction. We address the problem of pedestrian attribute recognition, where we determine different attributes of a pedestrian by using their full body images.

This problem poses a variety of fundamental challenges. Owing to diverse appearances of pedestrian clothing and uncontrollable multi factor variations such as illumination and camera viewing angle, there exists large intra-class variations among different images for the same attribute. Further, most attributes require fine-grained information which is to be decided based on small sub regions of the input image. In practice, these attributes of interest might be occluded, either by obstacles or by other pedestrians. For example, to determine if a person is 'carrying backpack' the full bag might not be visible due to pedestrian posture. We try to understand the performance of Convolutional Neural Networks and compare it to traditional methods like SVM in predicting the attributes of a pedestrian where the input to the algorithm is a far-view image of a pedestrian and the algorithm predicts the different attributes of a pedestrian. We consider 35 attributes for prediction which consist of the most important attributes in video surveillance [1], [2] and some interesting attributes covering all body parts of pedestrians with varying prevalence [3].

CNNs have been quite impressive when it comes to computer vision tasks, especially image classification tasks, where they have achieved performance comparable to humans [17]. This can be generally attributed to the fact that the different layers in the Convolutional Neural Networks are able to learn the different modes of the images. This success of Convolutional Neural Networks inspired us to use them for the task of pedestrian attribute recognition. We propose two CNN based attribution recognition models - ShallowMAR and ShallowSAR. ShallowMAR treats the pedestrian attribute recognition as a multi-label classification problem, whereas ShallowSAR treats each attribute individually, one at a time for all the attributes. Since the images are of low resolution, we believe that shallow convolutional neural networks will give comparable results to deep convoltuional neural networks.

The major contributions of this project are

- Compare how well ConvNets perform as compared to traditional methods like multi-label SVM and softmax classifiers
- We propose and implement a ShallowSAR model that treats each attribute component individually, and a ShallowMAR model that treats all the attributes in a unified sense, allowing it to explore interesting relationships and dependencies between the attributes

Figure 1. Sample Images form PETA dataset.Positive and negative sample images are indicated by red and blue boxes, respectively.

2. Related Work

Significant work has been done in detecting and recognizing pedestrian attributes from far view images in the past. SVM based individual attribute classifier has been a popular choice, as described in [4]. Novel methods for re-identification using mid-level semantic attributes have been proposed by [1]. Recently, there has been some interest in using Convolutional Neural Networks for this task.[10] explores the aspect of using multi label CNNs for determining pedestrian attributes, where they try to predict multiple attributes together in a unified framework. Deep learning based models which take into account the relationship among pedestrian attributes have been proposed by [5]. They propose the use of convolutional networks for both single attribute classification and multi-attribute classification, where in single attribute classification, the ConvNet is run on the dataset considering only one attribute at a time, while in the multi-attribute method, the ConvNet learns the parameters for all the attributes simultaneously. This recent paper in CVPR 2015 [11] propose a novel deep level model to learn high level features from multiple tasks and multiple data sources. Motion information have also been incorporated in improving detection performance using a sliding window framework [12], which could also be used to improve performance of pedestrian attribute recognition. [13] present baseline methods for binary and multi-class attribute classification. There has been efforts by [14] in detecting attributes of people under large variation of viewpoints, pose, articulation and occlusion. [15] focuses on analyzing gender of a pedestrian and whether the pedestrian is carrying a baggage or not in a public space using top-view camera images. [16] proposes a technique to search through large volumes of surveillance data, keeping the attributes observable at a distance in mind.

3. Methods

We address the challenge of attribute classification using both ConvNets and Non-ConvNets based approaches.

3.1. Non-ConvNets Based Approach

1. **K-Nearest Neighbours:**
   For each image we find its k-nearest neighbours using the Euclidean distance. Then for each of the 35 attributes, if majority of the k-closest images have that attribute present in them, then we predict that the attribute is also present in the test image. The value of k is determined using cross validation.

2. **SVM:**
   We train a binary SVM classifier for each of the 35 binary attributes which is then used to predict the set of attributes for the test images. We formulate a multi-label loss function for the i^{th} training example for the SVM as:

   $$L_i = \Sigma_{j=1}^{c} \max(0, 1 - y_{ij} w_j^T x_i)$$

   where w_j^T denotes the weights of the binary SVM classifier corresponding to the j^{th} attribute and c is the total number of different attributes considered i.e 35. The overall loss is the average loss across all training examples. The labels $y_{ij} \in \{-1, 1\}$. Each attribute is said to be present in the test image, if the score corresponding to that attribute is positive. The final loss with regularization is given by

   $$L = \frac{1}{N} \Sigma_{i=1}^{N} \Sigma_{j=1}^{c} \max(0, 1 - y_{ij} w_j^T x_i) + \frac{\lambda}{2} \Sigma_k \Sigma_l W_{kl}^2 \quad (1)$$

3. **Softmax:**
   In a manner similar to SVM, we train a logistic regression model for each of the attributes, and report our prediction accuracy on the test dataset. The multi-label softmax loss is formulated as

   $$L_i = -\Sigma_{j=1}^{c} y_{ij} \log \sigma(s_{ij}) + (1 - y_{ij}) \log(1 - \sigma(s_{ij})) \quad (2)$$

   where $s_{ij} = w_j^T x_i$. As in the case of SVM, the total loss is the average loss across all the training examples.

σ is the sigmoid function. However one important difference to note here is that the labels $y_{ij} \in \{0, 1\}$. If $\sigma(s_{ij}) > 0.5$, then attribute j is said to present in image i.

4. **Softmax with Modified Loss Fuction:**
We define the modified softmax loss function as

$$L_i = -\Sigma_{j=1}^{c}(y_{ij}\log\sigma(s_{ij}) + (1-y_{ij})\log(1-\sigma(s_{ij})))d_{ij} \quad (3)$$

$$d_{ij} = \exp(p_l) \quad \text{if } y_{ij} = 0$$

$$d_{ij} = \exp(1-p_l) \quad \text{if } y_{ij} = 1$$

where p_l denotes the probability of the presence of attribute l in the training dataset, which can be computed empirically as in [5]. Intuitively, this modified loss function forces the classifier to learn the skewed distributions also. For example, the attribute "V-Neck" is present in very few images. Thus, the probability of occurrence of this attribute in an image is very low. Therefore, if the classifier doesn't detect the attribute on an image which has V-Neck, then it incurs a higher loss, thereby ensuring that the classifier learns the distribution more accurately for the skewed examples. The final loss with regularization is given by

$$L = -\frac{1}{N}\Sigma_{i=1}^{N}\Sigma_{j=1}^{c}(y_{ij}\log\sigma(s_{ij}) + (1-y_{ij})\log(1-\sigma(s_{ij})))d_{ij} + \frac{\lambda}{2}\Sigma_k\Sigma_l W_{kl}^2 \quad (4)$$

5. **Neural Nets:** We implement a four layer fully connected neural network, where the neurons learn the model for each of the attributes simultaneously. Each hidden layer of the neural network has a hidden dimension of 100. We used the SVM loss function with ReLU non-linearity and batch normalization.

3.2. ConvNet Based Approach

In case of training a softmax or SVM based classifier all attributes are treated independently. But most of the attributes are not independent. For example it is more likely that if an image is of a female pedestrian than we have a positive value for the attributes "Long Hair" and "Skirt". We believe that ConvNets would be able to learn these hidden relations among attributes and thus result in better prediction accuracy. In this section two methods are proposed to tackle the problem of pedestrian attribute recognition.

3.2.1 ShallowMAR

Generally, the attributes are connected. In order to learn these dependencies we train a unified multi-attribute ConvNet i.e ShallowMAR: Shallow Multi-Attribute Recognition. We mainly explore two types of ConvNet architecture.

1. (Conv-ReLU-Pool)$\times$ N + Affine $\times$ M

2. (Conv-ReLU-Conv-ReLU)$\times$ N + Affine $\times$ M

In the second type of architecture we see two CONV layers stacked before every POOL layer.This is generally a good idea for larger and deeper networks, because multiple stacked CONV layers can develop more complex features of the input volume before the destructive pooling operation. [6].

3.2.2 ShallowSAR

In order to understand how well the ConvNets understand the inter-dependencies among the attributes, we also train ConvNets for each of the 35 attributes individually and compare it with ShallowMAR. ShallowSAR stands for Shallow Single-Attribute Recognition. We again try out the same types of architectures as mentioned for ShallowMAR.

.

4. Dataset

PEdesTrian Attribute (PETA) [4] is the current biggest challenging pedestrian attributes dataset that has been used for benchmark evaluation. The dataset consists of 19000 pedestrian images, with the image resolution ranging from 17 $\times$ 39 pixels to 169 $\times$ 365 pixels. It includes a total of 8705 persons, each annotated with 61 binary and 4 multi-class attributes. Since the images are taken from real surveillance cameras the dataset contains a burst of image shots of a single pedestrian with varying number of images per pedestrian. Some of the images from the dataset are shown in Figure 1 where a red bounding box for the image indicates a positive example for the attribute, while a blue bounding box indicates a negative example.

We consider only a subset of 35 attributes for prediction from the 61 binary attributes, which consist of the most important attributes in video surveillance [1], [2] and some interesting attributes covering all body parts of pedestrians with varying prevalence [3]. We carefully divide the dataset of 19000 images into 12000 images for training, 2000 for validation and 5000 for testing so as to ensure that the training, validation and the test set data do not have images of the same person in common among them. We do this because, in reality, we won't have images of pedestrians whose attribute we want to determine in our training set.

4.1. Preprocessing

PETA dataset is an amalgamation of 10 different datasets. Due to which the images are of varying sizes and resolutions. We decided to re-size all the images to 64 $\times$64 using Bilinear Interpolation. For all the non-ConvNet based approaches we didn't have access to GPU (at that time) and hence the experiments were performed using images of size 32 $\times$32.

1. **Mean subtraction:** Mean subtraction is the most common form of preprocessing. It involves subtracting the mean across every individual feature in the data, and has the geometric interpretation of centering the cloud of data around the origin along every dimension.

2. **Normalization:** Normalization refers to normalizing the data dimensions so that they are of approximately the same scale. We divide each dimension by its standard deviation, once it has been zero-centered. [6].

3. **Data Augmentation:** One way to improve the performance of the ConvNets is to augment the input data. Data augmentation is especially useful when training data is limited. We experiment with a several kinds of data augmentaion. Specifically we augment our training data by rotation, horizontal flips and shifting the images horizontally and vertically.

5. Experiments

5.1. Setup

We used the assignment framework for all our non-ConvNet based approaches. For ConvNets we used Keras:Deep Learning library for Theano and TensorFlow. We decided to use Keras because it enabled us to do quick prototyping and experimentation. All the experiments were performed using Amazon Web Service on g2.2xlarge (26 ECUs, 8 vCPUs, 2.6 GHz, Intel Xeon E5-2670, 15 GiB memory, 1 x 60 GiB Storage Capacity) instance.

5.2. Implementation Details

We used the binary crossentropy from the Keras library which implements the multi-label softmax loss as discussed in Section 3. All the Conv layers were preceded by appropriate zero padding to preserve the size post convolution. For all our models we used Adam [8] update rule with the default parameters as mentioned in the original paper i.e. we used learning rate = 0.001,$\beta_1 = 0.9$ and $\beta_2 = 0.999$. We did experiment with different learning rates but found that the default parameters worked the best. Figure 2 shows the learning curve for our best model with the parameters mentioned above. Weights for all the Conv-Layers were initialized using Gaussian initialization scaled by $\text{fan}_{\text{in}} + \text{fan}_{\text{out}}$ [9]. In order to prevent over-fitting we add a dropout layer before the first affine layer. In addition to this we use L_2 regularization for weights in all the layers. Hyper parameters were chosen by random grid search by training the model for 5 epochs. We finally choose reg $= .01$ and dropout of 0.2. We trained our models for 100 epochs initially but found that the model was overfitting and subsequently trained the models for 10 epochs.

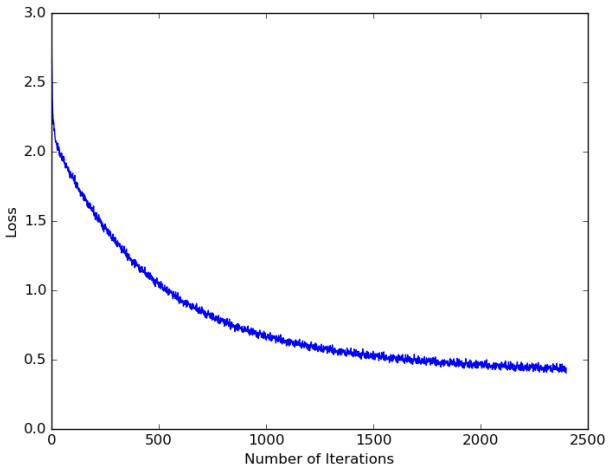

Figure 2. Learning Curve for ShallowMAR model(S6)

5.3. Evaluation Metric

We report the accuracy for each attribute as the fraction of pedestrians for whom it was correctly identified. To compare different models we use the average accuracy across all the attributes. We also report the F1 score. The F1 score can be interpreted as a weighted average of the precision and recall, where an F1 score reaches its best value at 1 and worst score at 0. The relative contribution of precision and recall to the F1 score are equal. The formula for the F1 score is:

$$F1 = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$

where precision and recall are defined as follows:

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive + False Positives}}$$

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive + False Negatives}}$$

6. Results and Discussion

Table 1 compares the accuracy of different methods across all 35 attributes. We also provide the ratio of number of images which had a particular attribute to total number of images.

6.1. Non ConvNet Approaches

6.1.1 K-NN

We expect K-NN to perform very good for attributes which have highly skewed distribution i.e. either a very high ratio or very less ratio.The results presented in the table are for 5-NN. We performed cross validation to determine the value of best k. As we can observe for attributes like "V-Neck" (Very Low Ratio) or "Casual Upper" we have very high prediction accuracy with k-NN. But, this can be attributed to the skewness in data. In attributes with equal distribution like "Male" the classifier does not have a very high prediction accuracy. But this algorithm can't be used in real time due to poor test time (running time) performance.

6.1.2 SVM and Softmax

We present the results obtained for best set of hyper-parameters for SVM and Softmax classifier. In the interest of space we only present the results of our Softmax classifier with modified loss function. Both SVM and Softmax don't perform as well as we had expected. The performance of both the algorithm is significantly less than that of K-NN, indicating that they might not be suitable for this problem.

6.1.3 Neural Network

The table also shows the results obtained for our four layer neural network for the best set of hyper-parameters. We see that the neural network performs significantly better than all the other approaches. We attribute this to the ability of neural network to learn inter-dependencies among different attributes.

6.2. ConvNet Based Approach

6.2.1 ShallowMAR

The Table 2 shows the average training and test accuracy across all 35 attributes for various CNN architectures (CRP - [conv]-[relu]-[maxpool]). We tried different ConvNet architectures with the number of filters at each layer being either 32 or 64, the stride and padding of the filters such that the input dimensions are preserved and experimented with filters of sizes 3 and 5. The best architecture (S6) has CRP-CRP-CRP structure, where the first convolutional layer has 64 filters of size 3, the second convolutional layer has 64 filters of size 5 and the third convolutional layers has 32 filters of size 5. Each convolutional layer is followed by ReLU non-linearity. The maxpool layers are of size 2x2 with stride 2. The last pool layer is followed by dropping out neurons randomly with 20% probability, after which we have the fully connected layer giving the scores for the 35 attributes. Figure 5 shows the architecture of S6.

Some important things we can notice from the table:

Attribute	Ratio	k-NN	SVM	Softmax	NN
Age16-30	0.45	0.50	0.61	0.44	0.61
Age31-45	0.29	0.53	0.48	0.45	0.66
Age46-60	0.09	0.85	0.43	0.54	0.84
AgeAbove61	0.06	0.90	0.43	0.63	0.88
Backpack	0.17	0.75	0.65	0.50	0.84
CarryingOther	0.18	0.71	0.63	0.47	0.80
Casual lower	0.78	0.75	0.42	0.50	0.88
Casual upper	0.77	0.73	0.43	0.53	0.88
Formal lower	0.12	0.75	0.42	0.52	0.88
Formal upper	0.12	0.76	0.42	0.51	0.88
Hat	0.10	0.85	0.37	0.54	0.81
Jacket	0.06	0.90	0.57	0.49	0.93
Jeans	0.28	0.57	0.64	0.61	0.67
Leather Shoes	0.26	0.53	0.48	0.46	0.60
Logo	0.03	0.93	0.69	0.55	0.97
Long hair	0.22	0.66	0.62	0.53	0.73
Male	0.49	0.50	0.55	0.47	0.42
Messenger Bag	0.27	0.66	0.54	0.55	0.61
Muffler	0.08	0.92	0.48	0.50	0.84
No accessory	0.66	0.66	0.50	0.55	0.58
No carrying	0.24	0.54	0.48	0.40	0.78
Plaid	0.02	0.96	0.65	0.60	0.98
PlasticBags	0.06	0.94	0.53	0.50	0.88
Sandals	0.02	0.96	0.67	0.50	0.97
Shoes	0.33	0.55	0.52	0.49	0.63
Shorts	0.03	0.95	0.67	0.55	0.98
Short Sleeve	0.11	0.73	0.60	0.44	0.92
Skirt	0.04	0.96	0.67	0.54	0.96
Sneaker	0.19	0.74	0.70	0.51	0.84
Stripes	0.02	0.98	0.72	0.60	0.99
Sunglasses	0.03	0.90	0.72	0.59	0.94
Trousers	0.47	0.49	0.56	0.56	0.53
Tshirt	0.07	0.79	0.58	0.44	0.93
UpperOther	0.44	0.39	0.67	0.45	0.26
V-Neck	0.01	0.97	0.62	0.60	0.99
Average	*	0.75	0.56	0.53	0.80

Table 1. Attribute Recognition Accuracy on PETA for Different Methods

Model	**Architecture**	**Train Acc**	**Test Acc**	**F1 score**
S1	CRCRCRPx1	0.7947	0.7922	0.42
S2	CRCRCRPx2	0.7542	0.7564	0.43
S3	CRCRPx2	0.7612	0.7621	0.44
S4	CRCRPx3	0.7969	0.797	0.36
S5	CRPx2	0.7902	0.7901	0.43
S6	CRPx3*	0.8043	0.8045	0.42

Table 2. Different ShallowMAR architectures

1. We tried deeper networks like S2,S3 and S4. However, we found that these deeper networks were diffi-

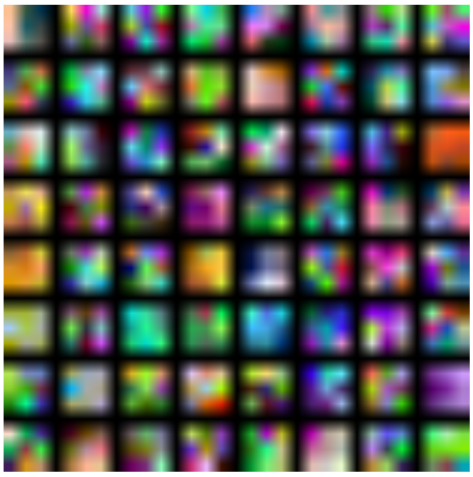

Figure 3. Visualization of first Conv layer weights for best ShallowMAR Architecture

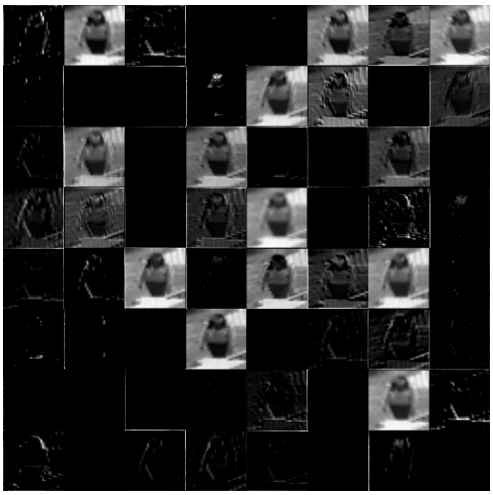

Figure 4. Visualization of activations after first Conv Layer for best ShallowMAR Architecture

cult to train and often resulted in overfitting. So after using regularization techniques like L2 regularization, dropout and early stopping (while training) we were able to prevent overfitting but that didn't translate to better test accuracy.

2. The images of the PETA dataset are the far-view images of pedestrians. Hence, the resolution of images is not that high. Therefore having deeper layers or smaller filter sizes in deeper layers does not improve the performance of the architecture greatly. This further motivated us to choose shallow networks.

3. Also, we can see that the training and test accuracy is nearly identical. This could be attributed to strong regularization. Given, more time we could potentially find an ideal balance of regularization and model strength which would result in better test accuracy.

Figure 5. Architecture of S6 (Best ShallowMAR)

4. Poor F1 scores : This is mainly caused because if we consider attributes which have low ratio like "formal lower" or "V-Neck", the ConvNet learns to predict 0 always which means the absence of these attributes. In this case we would not have any True Positive samples as True positive sample would be when the image has a "V-Neck" and our ConvNet predicted 1. Leading to 0 precision and recall. This causes the average F1 score to be very less.

6.2.2 Visualization

Figure 3 shows the filters in the first convolutional layer of S6. We see a variety of filters that are useful in detecting coarse features of input images. Some of these filters are mainly color blobs and are used by the network to predict the attributes of images that can be easily classified by their colors, and some are more suited for detecting edges, coarse shapes, etc. The most straight-forward visualization technique is to show the activations of the network during the forward pass. Figure 4 shows typical looking activations on the first CONV layer.

6.2.3 ShallowMAR Improvements

Since S6 gave the best validation accuracy (not shown in Table 2) we decided to improve the architecture further. Specifically we tried three modifications (Table3). We can see that adding affine layers gave us an extra 1% accuracy. We also experimented with various channel depths but the accuracies were very close and didn't result in any significant improvement. Further, we also tried adding Gaussian Noise after every Conv Layer. Gaussian Noise layer in Keras adds an additive zero-centred gaussian noise with standard deviation sigma (.01). This is useful to mitigate overfitting (you could see it as a kind of random data augmentation). As it is a regularization layer, it is only active at training time. We didn't notice any significant difference by addition of Gaussian Noise which maybe because we were already using other regularization techniques and data augmentation.

Modification	Channels	Train Acc	Test Acc
FC-32	32-32-32	0.8162	0.8139
FC-32 + GN	32-32-32	0.7941	0.7940
FC-32	64-64-32	0.8062	0.8039
FC-32 + GN	64-64-32	0.7988	0.7988
FC-64	32-32-32	0.8103	0.8110
FC-64 + GN	32-32-32	0.7945	0.7944
FC-64	64-64-32	0.8031	0.8048
FC-64 + GN	64-64-32	0.8078	0.8077

Table 3. Different ShallowMAR architectures

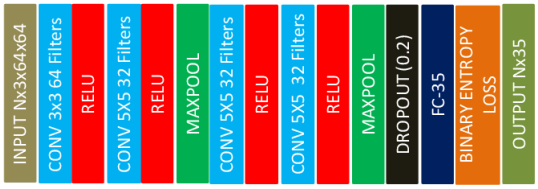

Figure 6. Architecture of Best ShallowSAR

6.2.4 ShallowSAR

We tried different architectures for ShallowSAR in a fashion similar to ShallowMAR. The ShallowSAR network has a CRCRP-CRCRP structure, with the first convolutional layer having 64 filters of size 3 and the other convolutional layers having 32 filters of size 5. Each convolutional layer is followed by ReLU non-linearity and the maxpool layers are of size 2x2 with stride 2. The final pool layer is followed by dropout with the neurons being dropped randomly with probability 0.2. After the dropout layer, we have the fully connected layer, which give the scores for the individual attributes considered. The filters in the first layer of size 3 are able to capture the fine details of the image. Since the images are of low resolution, the images in deeper layers have relatively less information to offer and hence filters of size 5 in deeper layers give similar results as when using filters of size 3. Figure 6 shows the architecture of the best ShallowSAR network.

6.2.5 ShallowSAR vs ShallowMAR

Figure 9 shows the comparison of average accuracy across all 35 attributes. Due to time constraints we only tuned the hyper-parameters for training the ShallowSAR model for the first attribute. We chose the first attribute as it has a balanced distribution. If trained properly ShallowSAR performs significantly better than ShallowMAR which is expected as we have more parameters per attribute. Also, ShallowMAR always tries to learn (memorize) the skewed attributes which result in significant reduction in the loss but tends to do not so good on more balanced attributes.
In our case we can see from Table 4 that on average both the models perform nearly the same. However, ShallowMAR should be preferred as it presents a unified model for all attributes and it can learn the complex relations among the attributes.

Figure 7. Images with lowest number of attribute matches

Model	Average Accuracy
ShallowSAR	0.819
ShallowMAR	0.813

Table 4. ShallowMAR vs ShallowSAR

6.3. ShallowMAR : A Qualitative Study

Figure 8 some of the cherry picked images for which ShallowMAR (S6) correctly predicts all 35 attributes. Some interesting things to notice is that the for images 3 and 4 in spite of lower body features being occluded the ConvNet is able to correctly predict the lower body features. Moreover, for image 6 we have multiple pedestrians in the image but the classifier is still able to recognize the attributes correctly. We can see that in case of far view surveillance images due to poor resolutions it is difficult for humans to make out all the attributes. For example, the person in image 5 could be potentially carrying something. As discussed previously depending on the orientation of the person and image capture, this attribute might be occluded. Further, the attribute "no carrying" is skewed in favor of carrying. In spite of this ShallowMAR is correctly able to predict that the person is not carrying anything.

Figure 7 shows the images for which ShallowMAR (S6) performed the worst. On average for these images it was only able to predict 20 attributes. On closer inspection, most of these 20 attributes which were predicted correctly had a skewed distribution and the correct label was the majority attribute. Since, the PETA data set consists of images with variety of image sizes, some of which are as low 17×39, we can see that these images are noticeably more blurry than the ones shown in Figure 8 making prediction of attributes difficult for the ConvNet.

7. Conclusion And Future Work

We tackled a fairly complicated multi-label classification problem. We implemented both convNet and non-ConvNet

Figure 8. Images with highest number of attribute matches

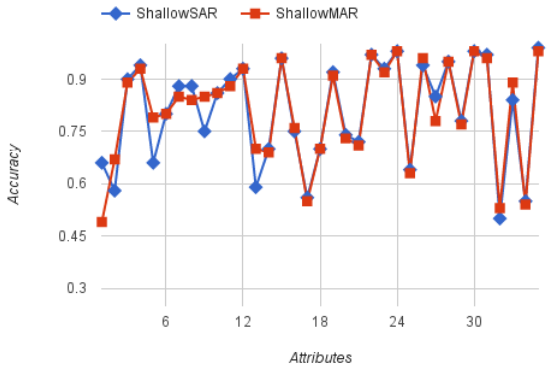

Figure 9. ShallowSAR and ShallowMAR across all 35 attributes

based approaches to address the problem. We see that the ConvNet based approaches were able to outperform traditional methods based on SVM or Softmax. This we attribute to the unique ability of the ConvNets to learn the different components of images like edges, colour blobs and such others. Though we were able to achieve an average accuracy comparable to [5] using ConvNets we would further like to improve the model by improving the the accuracy in case of attributes which have more balanced distribution. [5] uses transfer learning for their DeepMAR architecture. Given more time we would like to use transfer learning for our ShallowMAR architecture.

References

[1] Layne, Ryan, et al. "Person Re-identification by Attributes." BMVC. Vol. 2. No. 3. 2012.

[2] Nortcliffe, Toby. "People analysis cctv investigator handbook." Home Office Centre of Applied Science and Technology 2 (2011): 3.

[3] Deng, Yubin, et al. "Learning to Recognize Pedestrian Attribute." arXiv preprint arXiv:1501.00901 (2015).

[4] Deng, Yubin, et al. "Pedestrian attribute recognition at far distance." Proceedings of the ACM International Conference on Multimedia. ACM, 2014.

[5] Li, Dangwei, Xiaotang Chen, and Kaiqi Huang. "Multi-attribute Learning for Pedestrian Attribute Recognition in Surveillance Scenarios."

[6] CS 231N Lecture Notes

[7] Chollet, Franois, Keras, (2015), GitHub repository, https://github.com/fchollet/keras

[8] Kingma, Diederik, and Jimmy Ba. "Adam: A method for stochastic optimization." arXiv preprint arXiv:1412.6980 (2014).

[9] Glorot, Xavier, and Yoshua Bengio. "Understanding the difficulty of training deep feedforward neural networks." International conference on artificial intelligence and statistics. 2010.

[10] Zhu, Jianqing, et al. "Multi-label CNN based pedestrian attribute learning for soft biometrics." Biometrics (ICB), 2015 International Conference on. IEEE, 2015.

[11] Tian, Yonglong, et al. "Pedestrian detection aided by deep learning semantic tasks." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2015.

[12] Wojek, Christian, Stefan Walk, and Bernt Schiele. "Multi-cue onboard pedestrian detection." Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on. IEEE, 2009.

[13] , Jianqing, et al. "Pedestrian attribute classification in surveillance: Database and evaluation." Proceedings of the IEEE International Conference on Computer Vision Workshops. 2013.

[14] Bourdev, Lubomir, Subhransu Maji, and Jitendra Malik. "Describing people: A poselet-based approach to attribute classification." Computer Vision (ICCV), 2011 IEEE International Conference on. IEEE, 2011.

[15] Yamasaki, Toshihiko, and Tomoaki Matsunami. Pedestrian attribute analysis using a top-view camera in a public space. Springer Berlin Heidelberg, 2012.

[16] Thornton, Jason, et al. "Person attribute search for large-area video surveillance." Technologies for Homeland Security (HST), 2011 IEEE International Conference on. IEEE, 2011.

[17] Andrej Karpathy, Andrej Karpathy blog, http://karpathy.github.io/2014/09/02/what-i-learned-from-competing-against-a-convnet-on-imagenet/