International Journal of Pattern Recognition and Artificial Intelligence

Face Occlusion Detection Using Deep Convolutional Neural Networks

Yizhang Xia*, Bailing Zhang

Department of Computer Science and Software Engineering, Xian Jiaotong-Liverpool University, SIP, Suzhou 215123, China

yizhang.xia@xjtlu.edu.cn

Frans Coenen

Department of Computer Science, University of Liverpool, Liverpool L69 3BX, UK

Coenen@liverpool.ac.uk

With the rise of crimes associated with Automated Teller Machines (ATMs), security reinforcement by surveillance techniques has been a hot topic on the security agenda. As a result, cameras are frequently installed with ATMs, so as to capture the facial images of users. The main objective is to support follow-up criminal investigations in the event of an incident. However, in the case of miss-use, the user's face is often occluded. Therefore, face occlusion detection has become very important to prevent crimes connected with ATM usage. Traditional approaches to solving the problem typically comprise a succession of steps: localization, segmentation, feature extraction and recognition. This paper proposes an end-to-end facial occlusion detection framework, which is robust and effective by combining region proposal algorithm and Convolutional Neural Networks (CNN). The framework utilizes a coarse-to-fine strategy, which consists of two CNNs. The first CNN detects the head element within an upper body image while the second distinguishes which facial part is occluded from the head image. In comparison with previous approaches, the usage of CNN is optimal from a system point of view as the design is based on the end-to-end principle and the model operates directly on image pixels. For evaluation purposes, a face occlusion database consisting of over fifty thousand images, with annotated facial parts, was used. Experimental results revealed that the proposed framework is very effective. Using the bespoke face occlusion dataset, Aleix and Robert (AR) face dataset and the Labeled Face in the Wild (LFW) database, we achieved over 85.61%, 97.58% and 100% accuracies for head detection when the Intersection over Union-section (IoU) is larger than 0.5, and 94.55%, 98.58% and 95.41% accuracies for occlusion discrimination, respectively.

Keywords: Automated Teller Machine (ATM); Convolutional Neural Network (CNN); Face occlusion detection; Multi-Task Learning (MTL)

1. Introduction

Automated Teller Machines (ATMs) have always been the targets of criminal activity since their widespread introduction in the 1970s. For example, fraudsters can obtain card details and PINs using a wide range of tactics. Among the possible

*Corresponding author. Tel:+86 512 88161502.

techniques to defend against ATM crime, real time automatic alarm systems seem to be the most straightforward technical solution to maximize protection. This is because the surveillance cameras are installed in nearly all ATMs. However, current video surveillance for ATM requires constant staff monitoring, which has the obvious disadvantages of human error caused by fatigue or distraction.

Face occlusion detection has been studied for several years with a number of methods published [26, 31], many of which aim to reinforce ATM security. The published approaches can be roughly categorized into two categories: face or head detection approaches and occlusion classification approaches.

In the first category, the objective is robust face detection algorithms in the presence of partial occlusions, with two common practices, namely, facial component-based approaches and shape-based approaches. Facial component-based approach, such as that presented [20], detected facial components such as eyes, nose and mouth, and determines a face area based on the component detection result. For example, the method proposed in [20] combined seven AdaBoost-based classifiers for whole face with individual face-part classifiers trained on non-occluded face sample sets, and a decision tree and Linear Discriminant Analysis (LDA) to classify non-occluded faces and various types of occluded faces. Inspired by discriminatively trained part-based models, Ahmed [7] proposed a Selective Part Model (SPM) to detect faces under certain types of partial occlusions. Gul [15] applied what is known as the Viola-Jones approach [38], with free rectangular features, to detect left half faces, right half faces and the holistic faces. AdaBoost-based face detection was also improved upon in [5] in order to detect partially occluded faces, which, however, only worked for frontal faces with sufficient resolutions.

Shape-based approaches [3, 4, 17, 21, 28] detect faces based on the prior knowledge of head, neck and shoulder shapes. In the scenario of ATM video surveillance, [28] proposed to compute the lower boundary of the head by moving object edge extraction and head tracking. Motion information was also exploited in [21] to detect the head and shoulder shape with the aid of B-spline active contouring. Color has also been applied as a major clue to detect the head or face, following appropriate template fitting strategies [3, 4, 17]. These approaches, however, are limited to constrained poses and well-controlled illumination conditions. In [3, 4, 17] the head or shoulder were detected by using ellipse or what are known as “omega templates“, which, however, will fail when a face is severely occluded and/or the shapes are severely changed with different face poses.

Some of the previous published researches have tried to solve the face occlusion detection problem by straightforward classification [22]. For example, by separating a face area into upper and lower parts, Principal Component Analysis (PCA) and Support Vector Machine (SVM) were combined to distinguish between normal faces and partially occluded faces in [22].

Until recently, the most successful approaches to object detection utilized the well-known sliding window paradigm [10], in which a computationally efficient classifier tests for object presence in every candidate image window. The steady increase in complexity of the core classifiers has led to improved detection quality, but at the cost of significantly increased computation time per window [6, 11, 16, 35, 40]. One approach for overcoming the tension between computational tractability and high detection quality is through the use of "detection proposals" [8, 37]. If high object recall can be reached with considerably fewer windows than used by sliding window detectors, significant performance improvement can be achieved. Current top performing object detectors, when applied to PASCAL benchmark image datasets [9] and ImageNet [30], all used detection proposals [6, 11, 13, 16, 35, 40]. According to [18], approaches for generating object proposals can be divided into four types: grouping methods, window scoring methods, alternative methods and baseline methods. Grouping methods attempt to generate multiple (possibly overlapping) segments that are likely to correspond to objects. Window scoring methods are used to score each candidate window according to how likely it is to contain an object. Inspired by the success of applying object proposal approaches in different object detections, this paper proposes a face occlusion detection system using the highly ranked object proposal technique, EdgeBoxes [47].

Over the last several years, there has been increasing interests in deep neural network models for solving various vision problems. One of the most successful deep learning frameworks is the CNN architecture [24], which is a bio-inspired hierarchical multilayered neural network that can learn visual representations directly from raw images. CNN possesses some key properties, namely translation invariance and spatially local connections (receptive fields). Pre-trained CNN models can be exploited as generic feature extractors for different vision tasks [24]. Among the various advantages of deep neural networks over classical machine learning techniques, the most frequently cited examples include the conveniences for the implementation of knowledge transfer, Multi-Task Learning (MTL), attribute learning, multi label classification, and weakly supervised learning.

In this paper, a novel CNN based approach to face occlusion detection is proposed. A CNN cascade paradigm is adopted, which tackles the occlusion detection problem in a coarse-to-fine manner. The first CNN implements head/shoulder detection by taking a person's upper body image as input. The second CNN takes the output of the previous CNN as input and locates and classifies different facial parts. To facilitate the study of various face occlusion problems, a database directed at different kind of facial occlusions was created. The database consists of over fifty thousand images which are demarcated with four facial parts: two eyes, nose and mouth.

To the best of our knowledge, this is the first work directed at analyzing how face occlusion can be detected using multi-task CNN. The approach was verified on our face occlusion dataset, AR dataset [29] and LFW dataset [19], obtaining 94.55%, 95.58% and 95.41% accuracies, respectively.

The rest of the paper is organized as follows. Section 2 provides the problem descriptions with the introduction of our face occlusion dataset. Section 3 overviews the proposed method and elaborates on the details of the coarse-to-fine framework.

(a) (b) (c)

Fig. 1. Face occlusions examples for the withdrawing cash from an ATM scenario.

Section 4 reports the experiment results, followed by conclusion in Section 5.

2. Face Occlusion Dataset

A normal face image consists of two eyes, a nose and a mouth. The geometrical and textual information from these components is critical to the recognition or identification of a person. If any of these components is blocked or covered, the face image is considered to be occluded. Commonly found face occlusions include the faces being partially covered by a hat, sunglasses, mask or muffler. Some examples are given in Fig. 1.

To facilitate research on face occlusion detection, a database of face images which has different facial regions intentionally covered or occluded was created. The images were taken using the Microsoft webcam studio camera and AMCap 9.20 edition. Some example images are illustrated in Figs. 2 and Fig. 3. The video was filmed with a white background. There were 220 people involved, including 140 males and 80 females.

During the photography, a subject was asked to stand in front of the camera with a set of different specified poses, including looking right ahead, up and down roughly 45 degree, and right and left roughly 45 degree. In addition to taking face images without any occlusions, a subject was also asked to wear sunglasses, hat (in yellow and black), white mask and black helmet, again in five different poses. Each subject has 6 video clips recorded, with 30 seconds for each clip and 25 frames per second. The illumination was normal office lighting condition. Other conditions, for example, clothing, make-up, hair style and expression, have not been strictly controlled.

Although the created dataset is comprehensive, it cannot become public currently due to legal considerations. To alleviate the problem, we also utilized the AR face database [29], which is one of the earliest and most popular benchmark face databases. AR faces have been often used for robust face recognition. It includes a number of different types occlusions: faces with sunglasses and face partially cov-

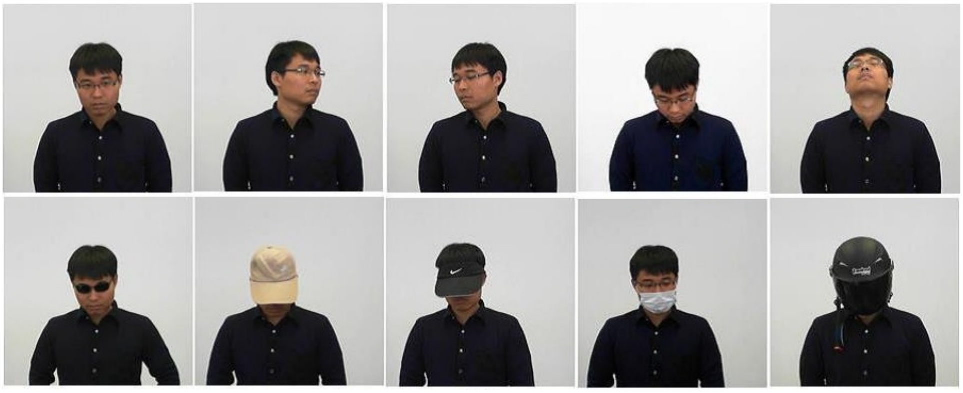


Fig. 2. Upper body of our created dataset

Fig. 3. Head of our created dataset

ered by a scarf. The AR faces dataset consists of over 3200 frontal face images taken from 126 subjects, with some examples given in Fig. 4.

The LFW dataset [19] was also employed to further evaluate our approach. There are 13000 images and 1680 people in the LFW dataset collected from the Internet. The faces were detected by the OpenCV implementation of the Viola-Jones [38] face detector. The cropped region returned by the detector was then automatically enlarged by a factor of 2.2 in each dimension to capture more of the head and then scaled to a uniform size, 250*250. For the evaluation presented in this paper, 1000 images were selected and the heads manually cropped. Occlusions were created using black rectangles and facial landmark localization [45]. Some examples are given in Fig. 5.

(a) (b)

Fig. 4. Examples of face from the AR face Database. (a) head and shoulder, (b) head.

Fig. 5. Examples of images from the LFW database with occlusions superimposed

3. System Overview

Though deep neural networks have achieved remarkable performance, it is still difficult to solve many real-world problems by a single CNN model. With the current technology, the resolution of an input to CNN must be relatively small. This will cause some details of an image lost. To get better performance, a common practice of coarse-to-fine paradigm has been applied with CNN design [34, 48]. Following the same line of thought, we proposed a two-stage convolutional neural network for face occlusion detection, as illustrated in Fig. 6. The first CNN detects the head from a person's upper body image while the second CNN distinguishes which facial part is occluded from the head image.

3.1. *Head Detection*

To identify the locations of the head in an image, advantages were taken of Region with Convolutional Neural Networks (R-CNN) [12], which is the state-of-the-art

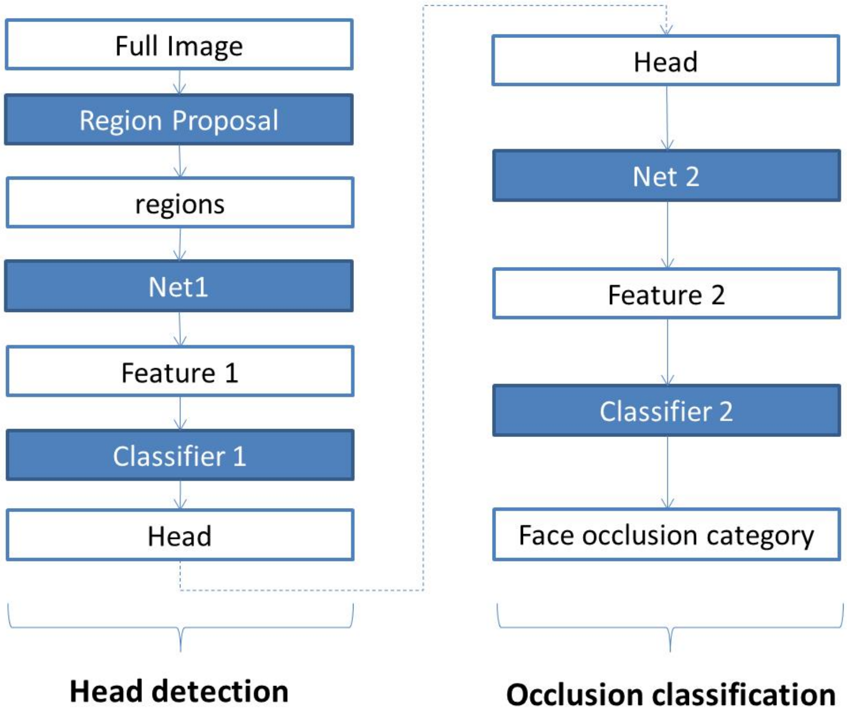


Fig. 6. The flow diagram of whole system

object detector that classifies candidate object hypothesis generated by appropriate region proposal algorithms. The R-CNN leverages several advantages of computer vision development and CNN, including the superb feature expression capability from a pre-trained CNN, fine-tuning flexibility for specific objects to be detected and the ever-increasing efficiency of object proposal generation schemes.

Among the off-the-shelf object proposal generation algorithms, the EdgeBoxes technique [47] was chose which has attracted much interest in recent years. EdgeBoxes is built on the Structural Edge Map to locate object boundaries and find object proposals. The number of enclosed edges inside a bounding box is used to rank the likelihood of the box containing an object.

The overall convolutional net architecture is shown in Fig. 7. The network consists of three convolution stages followed by three fully connected layers. A convolution stage includes a convolution layer, a non-linear activation layer, a local response normalization layer and a max pooling layer. The non-linear activation layer and local response normalization layers are not included in Fig. 7 and Fig. 8 as data size was not changed. Using shorthand notation, the full architecture is C(8,5,1)-$\tilde{A}$-N-P-C(16,5,1)-$\tilde{A}$-N-P-C(32,5,1)-$\tilde{A}$-N-P-FC(1024)-$\tilde{A}$-FC(128)-$\tilde{A}$-FC(4)-$\tilde{A}$, where C(d,f,s) indicates a convolutional layer with d filters of spatial size f×f, applied to the input with stride s. A is the non-linear activation function, which uses the ReLU activation function[14]. FC(n) is a fully connected layer with n output nodes. All pooling layers P use max-pooling in non-overlapping 2×2 regions and all normalization layers N are defined as described in Krizhevsky et al. [24]. The final layer is connected to a soft-max layer with dense connections. The structure of the networks and the hyper-parameters were empirically initialized based on previous works using CNNs.

The well-known overfitting problem has been taken into account in our design with the following considerations. Firstly, we empirically compared a set of different CNN architectures with varying number of kernels and selected the one which is deemed as the most effective with regard to the trade-off between network complexity and performance. Secondly, the overfitting has been avoided to a large extent as the CNN size is very moderate, which compares sharply with some published CNN models for large-scale image classifications [24].

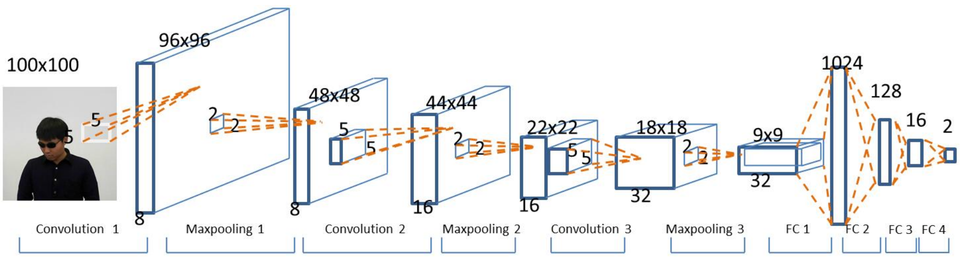


Fig. 7. Architecture of head detector CNN

The adopted CNN used the shared weight neural network architecture [25], in which the local receptive field (kernel or filter) is replicated across the entire visual field to form a feature map, which is known as convolution operation. The sharing of weights reduces the number of free variables, and increases the generalization performance of the network. Weights (kernels or filters) are initialized at random and will learn to be edge, color or specific pattern detectors.

In deep CNN, the classical sigmoidal function has been replaced by a Rectifier Linear Unit(ReLu) to accelerate training speed. Recent CNN-based approaches [23, 24, 33, 36, 41, 43] applied the ReLU as the nonlinear activation function for both the convolution layer and the full connection layer, often with faster training speed as reported in [24].

Typical pooling functions include average-pooling and max-pooling layers. Average pooling takes the arithmetic mean of the elements in each pooling region while max-pooling selects the largest element from the input.

The four layers of convolution, nonlinear activation, pooling and normalization, are combined hierarchically to form a convolution stage (block). Generally, an input image will be passed through several convolution stages for extracting complex descriptive features. In the output of the topmost convolution stage, all small-sized feature maps are concatenated into a long vector. Such a vector plays the same role as hand-coded features and it is fed to a full connection layer. A standard full connection operation can be either the conventional Multi-Layer Perceptron (MLP) or SVM.

As the fully connected layers receive feature vector from the topmost convolution stage, the output layer can generate a probability distribution over the output classes. Toward this purpose, the output of the last fully-connected layer is fed to a K-way softmax (where K is the number of classes) layer, which is the same as a multi-class logistic regression.

3.2. *Face Occlusion Classification*

The second stage CNN takes the output from head detector as input and implements the face occlusion classification. The CNN trains the classifier with the implicit, highly discriminative features to differentiate facial parts and distinguish whether a facial part is occluded or not at the same time. This is aided by a multi-task learning paradigm described in more detail below.

The intuition of multi-task learning is to jointly learn multiple tasks by exploiting a shared structural representation and improving the generalization by using related tasks. Deep neural networks such as CNN have been proven advantageous in multi-task learning due to their powerful representation learning capability and the knowledge transferability across similar tasks [32, 42, 44].

Inspired by the successes of CNN multi-task learning, we configured the second stage CNN to enable its shared representation learned in the feature layers for two independent MLP classifiers in the final layer. Specifically, we jointly trained the facial parts (left eye, right eye, nose and mouth) classification and occlusion/non-occlusion decision simultaneously.

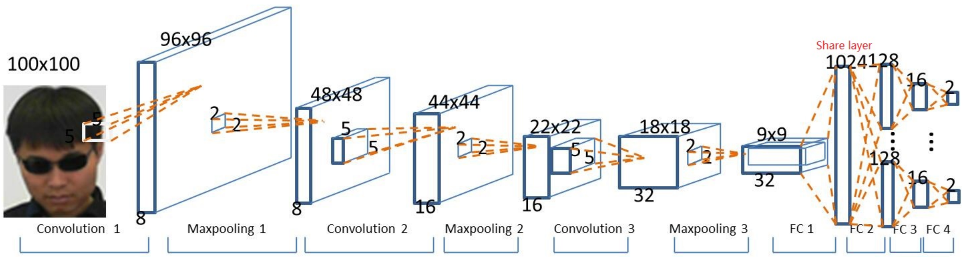


Fig. 8. Architecture of face occlusion classifier CNN

In our experiments, we adopt a multi-task CNN as shown in Fig. 8. The architecture is same as for the CNN for head detector. When multi-task learning is performed, we minimize the linear combination of individual task loss [42] as:

$$L_{joint} = \sum_{i=1}^{N} \alpha_i L_i \tag{1}$$

where N is the total number of tasks, α_i is weighting factor for the i-th task and L_i is the i-th task loss. When one of α_is takes 1, it will degenerate to classical single task learning.

3.3. *Bounding Box Regression*

A bounding box regression module is employed to improve the detection accuracy. Many bounding boxes generated from the EdgeBoxes algorithm are not close to the object ground truth, but might be judged as positive samples. We can regard detection as a regression problem to find the location of an object. This formulation can work well for localizing a single object. Based on the error analysis, we implemented a simple method to reduce localization errors. Inspired by the bounding-box regression employed in the Deformable Parts Model (DPM) [10], we trained a linear regression model to predict a new detection window given by the last pooled features for the region proposals produced by the EdgeBoxes. This simple approach can fix a large number of mislocalized detections, thus substantially boosting the accuracy.

3.4. *Pre-train*

By supervised learning with sufficient annotated training data, CNNs that contain millions of parameters have demonstrated competitive performance for visual recognition tasks [23, 24, 33, 36, 41, 43] when starting from a random initialization. However, CNN architecture has a property that is strongly dependent on large amounts of training data for good generalization. When the amount of labeled data is limited, directly training a high capacitor CNN may become problematic. Researches [39] have shown an alternative solution to compensate the problem by choosing an optimised starting point which can be pre-trained by transferring parameters from either supervised learning or unsupervised learning, as opposite to a random initialized start. We first trained the CNN model in the supervised mode using the ImageNet data and then fine-tuned it on the domain-specific labeled images as the head detector. To be specific, the pre-trained model is designed to recognize objects in natural images. The leveraged knowledge from the source task could reflect some common characteristics shared in these two types of images such as corners or edges.

4. Experiment Results

The proposed approach was analyzed using our face occlusion dataset, the AR face dataset and the LFW dataset with pre-training and fine-tuning. For head detector, the pre-training stage employed 10% of images from the ILSVRC2012 [30] for the classification task. For the occlusion classification, the pre-training was implemented by the face recognition.

4.1. *Implementation Details*

The experiments were conducted on a computer Dell Tower 5810 with Intel Xeon E5-1650 v3 and 64G of memory. In order to speed up CNN training, a GPU, NVIDIA GeForce GTX TITAN, is plugged on the board. The program operated under 64-bit windows 7 Ultimate with Matlab 2013b, Microsoft visual studio 2012 and CUDA7.0[46].

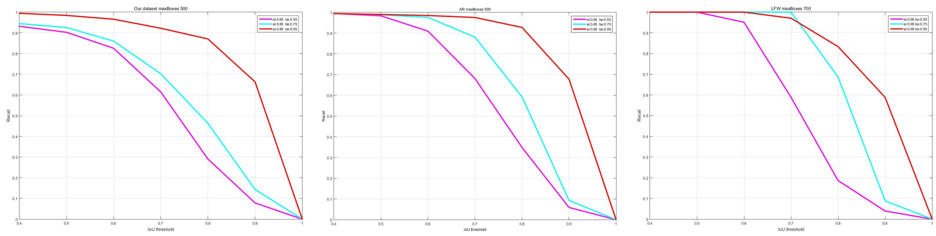
We trained our models using stochastic gradient decent with a batch size of 128 examples and momentum of 0.01. The learning rate was initialized as 0.01 and adapted during training. More specifically, we monitored the overall loss function. If the loss was not reduced for 5 epochs continually, the learning rate was dropped by 50%. We deem the network converged if the loss is stabilized.

In the training procedure, firstly, 10% of images from the ILSVRC2012 are used to train Net1 from random initialization. Then, the fine tuning on Net1 is implemented by head detection dataset in the AR face dataset, our dataset and the LFW dataset. Thirdly, Net2 is initialized from the Net1 after fine turning. Finally, the fine tuning on Net2 is trained by face occlusion classification dataset in the AR face dataset, our dataset and the LFW dataset. And the test procedure is described in Fig. 6.

4.2. *Head Detector*

As the state-of-the-art object proposal method with regard to the trade-off between the speed and recall, EdgeBoxes method needs to tune the parameters at each desired overlap threshold [47]. We estimated three pairs of α and β parameters to evaluate the object proposal corresponding to the head hypothesis, following the standard object proposal evaluation framework. The detection recall is calculated as the ratio of ground truth bounding boxes that have been predicted among the EdgeBoxes proposals with an Intersection Over Union (IoU) larger than a given threshold.

Three useful pairs of α and β values in EdgeBoxes [47] were estimated to provide the head candidates from our dataset, the AR face dataset and the LFW dataset, as shown in Fig. 9. The parameters α and β control the step size of the sliding window search and the NMS threshold, respectively. The two figures in Fig. 9 illustrate the algorithm's behavior with varying α and β when the same number of object proposals are generated. More specifically, three pairs of α and β are 0.65/0.55, 0.65/0.75, and 0.85/0.95 respectively. They were tested when the max region proposal was fixed at 500 with respect to our database and the AR database and 700 on the LFW database. It is obvious that the biggest value obtain the best recall, such that we were able to achieve 97.53% on our face occlusion database,



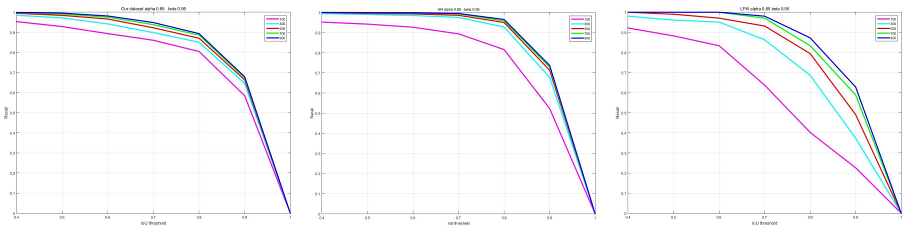
(a) (b) (c)

Fig. 9. The three useful variants of EdgeBoxes when the max number of region proposal is same.

99.35% on the AR face database and 100% on the LFW database when IoU is larger than 0.5. In conclusion, when the density of the sampling rate increases, we will get higher recall, but the run time becomes longer.

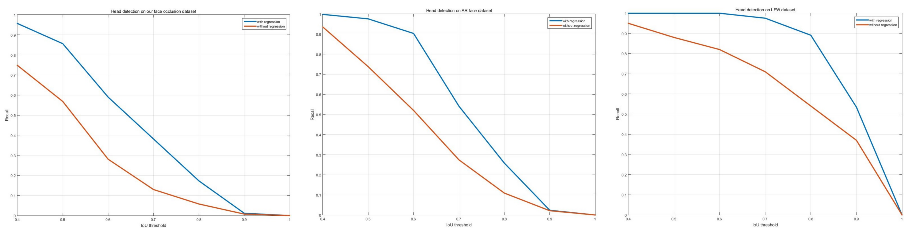
The second experiment compared the different number of bounding boxes produced by EdgeBoxes according to ranking score. A suite of maximal number of boxes, 100, 300, 500, 700, 900, was evaluated when the density of the sampling rate and NMS threshold are fixed to 0.85 and 0.95 respectively, as shown in Fig. 10. In order to make sure the face occlusion classification is valid, the IoU between region and ground truth is larger than 0.5. The recall is approximately 100% on our face occlusion database, the AR face database and the LFW database (Fig. 10). The recall does not increase when the maximum number of region proposal is larger than 500 on our dataset and the AR dataset and 700 on the LFW dataset. In consideration of the tolerance of Net1, the α, β and max number of region proposal were set to 0.85, 0.95 and 500 on our dataset and AR dataset respectively. By taking account of the complex background of LFW dataset, the maximal number of boxes was selected as 700 and the others parameters were the same for our dataset and AR dataset. The same parameters were selected for our face occlusion database and the AR face database. However, the recall obatined with repcet to the AR dataset is lower due to the greater variability of our database. The recall curves (Fig. 9 and Fig. 10) demonstrate that almost all of the head has been selected as proposals by the EdgeBoxes.

After obtaining the candidate regions from EdgeBoxes, the regions are classified into head and non-head by the trained CNN module. Before applying the CNN for the classification, a hypothesis object proposal will be discarded if the proposal can be simply judged as head or useless patch based on the following reasoning: a region is head if the overlap with the whole image is less than 5% on AR face database, a region is head or useless patch if the overlapping with the full image is between 2% and 30% on our face occlusion database. Then, the features are extracted via Net1 from normalized regions, 100×100 gray images. Next, the subsequent MLP predicts whether the region is a head region or not. After MLP, several positive regions are merged into a box by the non-max suppression with 0.3 overlap threshold



(a) (b) (c)

Fig. 10. The difference maximal number of region proposal when the α is 0.85 and β is 0.95.



(a) (b) (c)

Fig. 11. The recall of head detection, the red line is the performance of head detector without regression and the blue line indicate the performance of head detector with regression.

sorted by the reliability of the MLP, which indicated the accuracy of detection. The performance of the approach was tested on the AR face database and our face occlusion database. The accuracies using our face occlusion database, the AR face database and the LFW database were 56.83%, 73.79% and 88.52% with 0.5 IoU, respectively (Fig. 10). The performance can be further improved by bounding box regression, as expounded in the following.

We use a simple bounding-box regression stage to improve the localization performance. After scoring each object proposal by MLP, we predict a new bounding box for detection using a class-specific bounding-box regressor. This is similar in spirit to the approach used in deformable part models [12]. The better location of head is regressed from features computed by the CNN (Fig. 11). After regressing the positive features from CNN, the performance is improved, with increases of 28.78%, 23.79% and 11.48% on our face occlusion database, AR face database and LFW database respectively with IoU 0.5. The reason why the regressor improves the performance is that the head is centered in the images. However, the regressor only improves when the classification result from CNN is correct. Examples of the head detection from our face occlusion database, AR face database and LFW database are shown in the Fig. 12, Fig. 13 and Fig. 14, respectively.

We created a head detector based on HoG feature extraction [1] and SVM classification [2] as a baseline approach. The parameters for the HOG was set as orientations in the range [0, 360], 40 orientations bins and 3 spatial levels, which means that the inputs for SVM have a dimension of 840, $(1 + 4 + 16) \times 40$. For the SVM, we employed C-SVC and radial basis kernel function [2]. The comparison performance is illustrated in Table 1, which demonstrates that our approach is robust and effective in complex scenes. The computational complexity is showed Table 2. The major part of the computation of our approach is from the region proposal algorithm, EdgeBoxes.

Table 1. Accuracy of head detection when IoU is larger than 0.5

	LFW	AR	Our dataset
HOG+SVM	91.87%	97.44%	72.41%
Our method	100%	97.58%	85.61%

Table 2. Computational complexity on head detection(fps)

	LFW	AR	Our created dataset
HOG+SVM	32.48	24.06	419.68
Our method	0.94	2.02	1.76

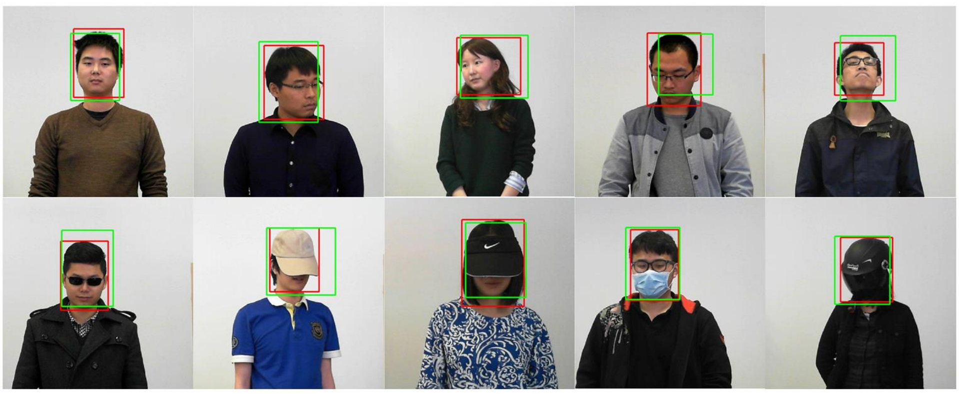


Fig. 12. Examples of head detection results from our face occlusion database, red rectangle is ground truth and the green rectangle is detection location.

4.3. *Face Occlusion Classification*

After obtaining the head position by the head detector, the face occlusion classifier was used to classify the type of face occlusion.

Fig. 13. Examples of head detection results from the AR face database, the red rectangle is the ground truth and the green rectangle is the detection location.

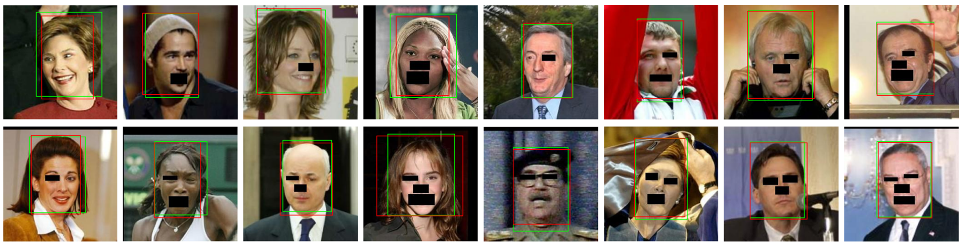
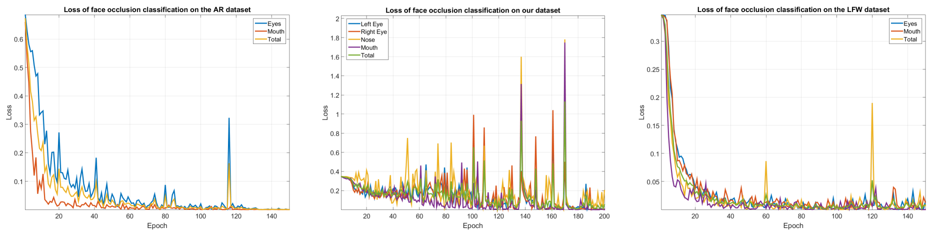


Fig. 14. Examples of head detection results from the LFW database, the red rectangle is the ground truth and the green rectangle is the detection location.

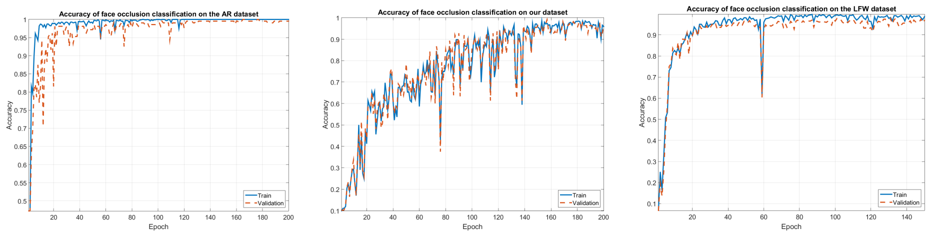
The AR face dataset [29] is a popular dataset for face recognition. It contains over 4,000 color images from 126 people's faces (70 males and 56 females). Images cover the frontal view faces with occlusions (sun glasses and scarf), different facial expressions and illumination conditions (Fig. 1). For the face occlusion classifier, the dataset is categorized into two occlusion conditions: face with eyes occluded by sunglasses and face with mouth occluded by scarf.Half of the un-occluded faces were use to pre-train the CNN model, following a general face recognition methodology. More specifically, the pre-train dataset consisted of 31 male and 25 female faces (frontal view face with different expressions and illumination).

After pre-training, the fully connected layers were replaced by a new MLP, initialized from random connection values for the fine-tuning, by applying the face



(a) (b) (c)

Fig. 15. Comparison of loss function values during training



(a) (b) (c)

Fig. 16. Comparison of accuracy values during training and validation

occlusion dataset. The CNN with multi-tasks learning was designed to predict the presence or absence of different facial parts, thus indicating occlusion or not. The structure of CNN is illustrated in Fig. 8.

The loss function values during training is illustrated in Fig. 15. After the loss becomes stabilized, the converged model is applied to test a testing sample with accuracy as shown in Table 3.

The accuracy values during training and validation are given in Fig. 16. Cross validation and early stopping are employed to avoid overfitting. Fig. 16 illustrates that the accuracies during training and validation continually rise. Early stopping is used to select the appropriate trained model during training and avoid overfitting.

Table 3. Face occlusion classification on AR dataset

	eyes	Mouth	Total
Accuracy	98.58%	100%	98.58%

The system performance for face occlusion classification was also evaluated using our face occlusion dataset and the LFW dataset. The experiment procedures are similar to the AR face dataset. The difference is that there is a variety of occlusions from different facial parts, namely, left eye, right eye, nose and mouth. Accordingly, there are four MLPs following the shared layer to verify whether each facial part is occluded or not, as explained in Fig. 8.

The training loss is shown in Fig. 15, and the occlusion accuracies on our dataset and LFW are 94.55% and 95.41% respectively, with further details provided in Table 4. Our model is a multi-task framework that shares the front layers. The summed loss changes with the updating of CNN parameters. And it can be seen that the loss fluctuates more obviously on our face dateset because it is much more complex compared with AR dataset and LFW dataset.

The face detector combined with Haar feature [27] extractor and Viola-Jones [38] classifier is used as the base line of face occlusion classification. The experiments result is summarized in Table 5, which indicates that our method outperforms Haar-based face detection. And the corresponding computational complexity is reported in Table 6 showing that our method is faster than the classical approach.

Table 4. The Accuracy on LFW and our face occlusion dataset

The Accuracy on our face occlusion dataset					
	Left eye	Right eye	Nose	Mouth	Total
Our dataset	98.15%	99.07%	98.15%	99.07%	94.55%
LFW	97.79%	98.91%	99.63%	99.02%	95.41%

Table 5. Accuracy on face occlusion classification

	LFW	AR	Our dataset
Haar+VJ	45.22%	47.98%	21.20%
Our method	95.41%	98.58%	94.55%

Table 6. Computational complexity on face occlusion classification(ms)

	LFW	AR	Our created dataset
Haar+VJ	16.29	58.23	22.35
Our method	13.10	20.67	17.31

4.4. *Error Analysis*

There are two factors that may impact the effectiveness of the proposed approach. Firstly, the region proposal algorithm should be robust with regard to the generated candidate bounding boxes. However, as the EdgeBoxes produce the candidate

(a) (b) (c)

Fig. 17. Examples of wrong region proposal

(a) (b) (c)

Fig. 18. Error prediction image sample

regions by the edge, it has several potential problems: (a) negative data will be generated when a person wears clothing with complex texture (Fig. 17(a)); (b) a head will appear larger with certain hairstyles (Fig. 17(b)); (c) the head will not be segmented when a person has long hair and wears a black dust coat (Fig. 17(c)). Secondly, there are three factors that may influence the performance of face occlusion classifier, including the illumination variations, occlusions and the partial occlusions. Fig. 18 further explains the difficult situations.

5. Conclusion

This paper proposed an approach for face occlusion detection to enhance the surveillance security for ATM. The coarse-to-fine approach consists of a head detector and a face occlusion classifier. The head detector is implemented with EdgeBoxes — region proposal, CNN and MLP. The method achieved detection accuracies of 97.58%, 85.61% and 100% on the AR face database, our face occlusion database and LFW dataset. For face occlusion classification, CNN is applied with a pre-training strategy via usual face recognition task, followed by fine-tuning with the face occlusion classification based on MTL to verify whether a facial part is occluded or not. Our approach is evaluated on the AR face dataset, our dataset and LFW dataset, achieving 98.58%, 94.55% and 95.41% accuracies, respectively. Further work is being made toward the improvement of the model by more robust and accurate region proposal, which will render it more realistic for real-world applications.

6. Acknowledgement

The first author thanks Chao Yan and Rongqiang Qian for their valuable helps. The constructive comments and suggestions from the anonymous reviewers are much appreciated.

References

1. A. Bosch, A. Zisserman and X. Munoz, Representing shape with a spatial pyramid kernel, *the 6th ACM International Conference on Image and Video Retrieval*, pp. 401–408, New York, 2007.
2. C.-C. Chang and C.-J. Lin, Libsvm: A library for support vector machines, *ACM Trans. Intell. Syst. Technol.*, Vol. 2, pp. 27:1–27:27, 2011.
3. T. Charoenpong, C. Nuthong and U. Watchareeruetai, A new method for occluded face detection from single viewpoint of head, *2014 11th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*, pp. 1-5, 2014.
4. T. Charoenpong, Face occlusion detection by using ellipse fitting and skin color ratio, *Burapha University International Conference (BUU)*, pp. 1145-1151, 2013.
5. J. Chen, S. Shan, S. Yang, X. Chen and W. Gao, Modification of the adaboost-based detector for partially occluded faces, *18th International Conference on Pattern Recognition*, Vol. 2, pp. 516-519, 2006.
6. R. Cinbis, J. Verbeek and C. Schmid, Segmentation driven object detection with fisher vectors, *2013 IEEE International Conference on Computer Vision (ICCV)*, pp. 2968-2975, 2013.
7. A. El-Barkouky, A. Shalaby, A. Mahmoud and A. Farag, Selective part models for detecting partially occluded faces in the wild, *2014 IEEE International Conference on Image Processing (ICIP)*, pp. 268-272, 2014.
8. I. Endres and D. Hoiem, Category-independent object proposals with diverse ranking, *IEEE Transactions on Pattern Analysis and Machine Intelligence*,Vol. 36, pp. 222–234, 2014.
9. M. Everingham, S. Eslami, L. Van Gool, C. Williams, J. Winn and A. Zisserman, The pascal visual object classes challenge: A retrospective, *International Journal of Computer Vision*, Vol. 111, pp. 98-136, 2015.
10. P. Felzenszwalb, R. Girshick, D. McAllester and D. Ramanan, Object detection

with discriminatively trained part-based models, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 32, pp. 1627-1645, 2010.

11. R. Girshick, J. Donahue, T. Darrell and J. Malik, Rich feature hierarchies for accurate object detection and semantic segmentation, *2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 580-587, 2014.
12. R. Girshick, J. Donahue, T. Darrell and J. Malik, Region-based convolutional networks for accurate object detection and segmentation, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 38, PP. 142-158, 2016.
13. R. Girshick, Fast r-cnn, *The IEEE International Conference on Computer Vision (ICCV)*, pp. 1440-1448, 2015.
14. X. Glorot, A. Bordes and Y. Bengio, Deep sparse rectifier neural networks, *Journal of Machine Learning Research*, Vol. 15, pp. 315-323, 2011.
15. S. Gul and H. Farooq, A machine learning approach to detect occluded faces in unconstrained crowd scene, *2015 IEEE 14th International Conference onCognitive Informatics Cognitive Computing (ICCI*CC)*, pp. 149-155, 2015.
16. K. He, X. Zhang, S. Ren and J. Sun, Spatial pyramid pooling in deep convolutional networks for visual recognition, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 37, pp. 1904-1916, 2015.
17. H. Sun, J. Wang, P. Sun and X. Zou, Facial area forecast and occluded face detection based on the ycbcr elliptical model, *International Conference on Mechatronic Sciences, Electric Engineering and Computer (MEC)*, pp. 1199-1202, 2013.
18. J. Hosang, R. Benenson, P. Dollár and B. Schiele, What makes for effective detection proposals?, *IEEE transactions on pattern analysis and machine intelligence* Vol. 38, pp. 814-830, 2016.
19. G. B. Huang, M. Ramesh, T. Berg and E. Learned-Miller, Labeled faces in the wild: A database for studying face recognition in unconstrained environments, *Technical Report*, University of Massachusetts, Amherst, pp. 07-49, 2007.
20. K. Ichikawa, T. Mita, O. Hori and T. Kobayashi, Component-based face detection method for various types of occluded faces, *3rd International Symposium on Communications, Control and Signal Processing*, pp. 538-543, 2008.
21. G. Kim, J. K. Suhr, H. G. Jung and J. Kim, Face occlusion detection by using b-spline active contour and skin color information, *11th International Conference on Control Automation Robotics Vision (ICARCV)*, pp. 627-632, 2010.
22. J. Kim, Y. Sung, S. Yoon and B. Park, A new video surveillance system employing occluded face detection, *Innovations in Applied Artificial Intelligence*, pp. 65-68, 2005.
23. J. Krause, T. Gebru, J. Deng, L.-J. Li and L. Fei-Fei, Learning features and parts for fine-grained recognition, *2014 22nd International Conference on Pattern Recognition (ICPR)*, pp. 26-33, 2014.
24. A. Krizhevsky, I. Sutskever and G. E. Hinton, Imagenet classification with deep convolutional neural networks, *Advances in Neural Information Processing Systems*, pp. 1097-1105, 2012.

25. Y. LeCun, Y. Bengio and G. Hinton, Deep learning, *Nature*, pp. 436-444, 2015.
26. S. Liao, A. K. Jain and S. Z. Li, Partial face recognition: Alignment-free approach, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 35, pp. 1193-1205, 2013.
27. R. Lienhart and J. Maydt, An extended set of haar-like features for rapid object detection, *International Conference on Image Processing*, Vol. 1, pp. 900-903, 2002.
28. D.-T. Lin and M.-J. Liu, Face occlusion detection for automated teller machine surveillance, *Advances in Image and Video Technology*, pp. 641-651, 2006.
29. A. Martinez and R. Benavente, The ar face database, *CVC Technical Report*, 1998.
30. O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg and L. Fei-Fei, ImageNet Large Scale Visual Recognition Challenge, *International Journal of Computer Vision (IJCV)* Vol. 115, pp. 211-252, 2015.
31. J. Shermina and V. Vasudevan, Recognition of the face images with occlusion and expression, *International Journal of Pattern Recognition and Artificial Intelligence* Vol. 26, 2012.
32. L. Sijin, L. Zhi-Qiang and A. Chan, Heterogeneous multi-task learning for human pose estimation with deep convolutional neural network, *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*,pp. 488-495, 2014.
33. K. Simonyan and A. Zisserman, Two-stream convolutional networks for action recognition in videos, *Advances in Neural Information Processing Systems*, pp. 568-576, 2014.
34. Y. Sun, X. Wang and X. Tang, Deep convolutional network cascade for facial point detection, *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Oregon, pp. 3476-3483, 2013.
35. C. Szegedy, S. Reed, D. Erhan and D. Anguelov, Scalable, high-quality object detection, *arXiv preprint*, 2014.
36. Y. Taigman, M. Yang, M. Ranzato and L. Wolf, Deepface: Closing the gap to human-level performance in face verification, *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1701-1708, 2014.
37. K. van de Sande, J. Uijlings, T. Gevers and A. Smeulders, Segmentation as selective search for object recognition, *IEEE International Conference on Computer Vision (ICCV)*, pp. 1879-1886, 2011.
38. P. Viola and M. Jones, Rapid object detection using a boosted cascade of simple features, *Computer Vision and Pattern Recognition*, Vol. 1, pp. 511-518, 2001.
39. R. Wagner, M. Thom, R. Schweiger, G. Palm and A. Rothermel, Learning convolutional neural networks from few samples, *The 2013 International Joint Conference on Neural Networks (IJCNN)*, pp. 1-7, 2013.
40. X. Wang, M. Yang, S. Zhu and Y. Lin, Regionlets for generic object detection, *IEEE International Conference on Computer Vision (ICCV)*, pp. 17-24, 2013.

41. S. Yi, W. Xiaogang and T. Xiaoou, Deep learning face representation from predicting 10,000 classes, *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1891-1898, 2014.
42. C. Zhang and Z. Zhang, Improving multiview face detection with multi-task deep convolutional neural networks, *IEEE Winter Conference on Applications of Computer Vision (WACV)* , pp. 1036-1041, 2014.
43. N. Zhang, M. Paluri, M. Ranzato, T. Darrell and L. Bourdev, Panda: Pose aligned networks for deep attribute modeling, *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1637-1644, 2014.
44. Z. Zhang, P. Luo, C. Loy and X. Tang, Facial landmark detection by deep multi-task learning, *European Conference on Computer Vision (ECCV)*, pp. 94-108, 2014.
45. E. Zhou, H. Fan, Z. Cao, Y. Jiang and Q. Yin, Extensive facial landmark localization with coarse-to-fine convolutional network cascade, *IEEE International Conference on Computer Vision Workshops (ICCVW)*, pp. 386-391, 2013.
46. X. Zhu, K. Li, A. Salah, L. Shi and K. Li, Parallel implementation of mafft on cuda-enabled graphics hardware, *IEEE/ACM Trans. Comput. Biol. Bioinformatics* Vol. 12, pp. 205-218, 2015.
47. C. Zitnick and P. Dollar, Edge boxes: Locating object proposals from edges, *European Conference on Computer Vision (ECCV)*, pp. 391-405, 2014.
48. Z. Liu, P. Luo, X. Wang and X. Tang, Deep learning face attributes in the wild, *The IEEE International Conference on Computer Vision (ICCV)*, pp. 3730-3738, 2015.