Cross-Domain Multitask Model for Head Detection and Facial Attribute Estimation

SABER MIRZAEE BAFTI, SOTIRIOS CHATZIDIMITRIADIS, AND KONSTANTINOS SIRLANTZIS

Intelligent Interactions Research Group, University of Kent, Canterbury CT2 7NT, U.K.

Corresponding author: Saber Mirzaee Bafti (sm2121@kent.ac.uk)

This work was supported by the European Regional Development Fund within the framework of the INTERREG VA France (Channel) England Programme [Assistive Devices for empowering disAbled People through robotic Technologies (ADAPT) Project].

ABSTRACT Extracting specific attributes of a face within an image, such as emotion, age, or head pose has numerous applications. As one of the most widely used vision-based attribute extraction models, HPE (Head Pose Estimation) models have been extensively explored. In spite of the success of these models, the pre-processing step of cropping the region of interest from the image, before it is fed into the network, is still a challenge. Moreover, a significant portion of the existing models are problem-specific models developed specifically for HPE. In response to the wide application of HPE models and the limitations of existing techniques, we developed a multi-purpose, multi-task model to parallelize face detection and pose estimation (i.e., along both axes of yaw and pitch). This model is based on the Mask-RCNN object detection model, which computes a collection of mid-level shared features in conjunction with some independent neural networks, for the detection of faces and the estimation of poses. We evaluated the proposed model using two publicly available datasets, Prima and BIWI, and obtained MAEs (Mean Absolute Errors) of 8.0 ± 8.6, and 8.2 ± 8.1 for yaw and pitch detection on Prima, and 6.2 ± 4.7, and 6.6 ± 4.9 on BIWI dataset. The generalization capability of the model and its cross-domain effectiveness was assessed on the publicly available dataset of UTKFace for face detection and age estimation, resulting a MAE of 5.3 ± 3.2. A comparison of the proposed model’s performance on the domains it was tested on reveals that it compares favorably with the state-of-the-art models, as demonstrated by their published results. We provide the source code of our model for public use at: https://github.com/kahroba2000/MTL_MRCNN.

INDEX TERMS Head tracking, head pose estimation, multi-task learning, age detection, object detection, mask R-CNN.

I. INTRODUCTION

HPE (Head pose estimation) is an open research area that has drawn the attention of specialists in different domains. The wide applications of HPE in assistive systems, human-computer interface systems, virtual reality etc., have brought it into the center of attention of the research community. For instance, HPE is one of the most efficient UIs (User Interfaces) for paralyzed patients who are suffering from complete quadriplegia [1], [2]. The patients in this group have little control over their four limbs, so head movement is one of the few ways for them to interact with computers and electronic devices. For example, several studies have used head movements in the yaw and pitch directions to control an EPW (Electric Powered Wheelchair) [3]–[5]. Another application of HPEs lies in vehicle-related technologies, where HPEs are implemented to examine the attention of drivers [6]–[8], as well as students’ attention in class [9], [10]. On the other hand, the recent success of VR-related technologies motivated researchers to use HPE for estimating the users’ gaze and FOV (Field of View) via head pose information [11]. Having a fast and reliable HPE model for all the aforementioned applications is critical, and to this end the research community has been focusing on two main HPE approaches; sensor-based and vision-based methods. Though sensor-based (IMU, tilt sensors, etc.) approaches are regarded as promising solutions, they impose an unwelcome level of discomfort and distraction to the users, due to their required attachment to the users’ heads.

In contrast, vision-based techniques enable us to calculate Euler angles from 2D scans of a user’s head, without requiring physical contact. Vision-based HPE is not a new idea, and
various studies with different levels of success have been
carried out to tackle this problem [12]–[18]. Despite the
promising success of these models in pose estimation, they
often suffer from the lack of an integrated head detection
mechanism. Therefore, before feeding the image to the model
for estimation, a preprocessing step needs to be introduced for
cropping the ROI (region of interest; in this study, faces). This
can be achieved either manually or via existing face detection
modules [19]–[21]. Face detection algorithms, used in con-
junction with the HPE, can adversely affect accuracy, speed,
and efficiency [22]. Also, a non-integrated head detection
mechanism would introduce significant processing demands
and delays in a multi-face pose estimation task. Following the
recent success of neural networks in performing concurrent
face detection and landmark detection [23]–[25] through a
set of shared features, some works have proposed the idea
of using multi-task learning models for parallelizing the face
detection and the pose estimation process [22], [26], [27].

Inspired by the wide applications of HPE, but taking into
account the limitations of the existing models and their lack
of extensibility to other domains, we developed a multi-
purpose, multitask object detection model. The proposed
model localizes objects of interest (in this case faces), while
it concurrently estimates attributes of that object. In other
words, the cross-domain, multitask object detection model
can be used for simultaneous face detection and pose estimation,
with adequate generalization capability to be also used
for estimation of other facial attributions (e.g. age). Conse-
quently, we developed an improved version of the current
Mask-RCNN model [19] to detect the face and estimate its
attributes (head pose in this case). Motivated by practical
applications of the HPE, as in assistive technologies for head-
operated wheelchairs [5], we developed our model for esti-
mating the head orientation in the yaw and pitch axes. The
model is built on top of the Mask-RCNN object detection
model and has been tested on two public datasets: BIWI
and Prima. The cross-domain aspect of the model is also
validated on the public dataset UTKFace, for face detection
and age estimation. In the next chapters, we first explore
the existing studies in this area (section II), followed by an
in-depth explanation of the proposed model (section III), and
its testing and evaluation (section IV). Section V discusses
the limitations of the proposed model, while the concluding
remarks summarizing the findings of this study are provided
in section VI.

II. RELATED WORKS

Due to the various applications of HPE, a number of vision-
based techniques for this purpose has been proposed by the
research community. In this section, we first discuss the exist-
ing vision-based HPE techniques, followed by a comprehen-
sive exploration of the multitask learning models developed
for parallelizing several tasks in neural networks.

A. HEAD POSE ESTIMATION

Vision-based techniques for pose estimation have gained
momentum in the computer vision area. Several advantages
over sensor based methods make them more attractive to
developers and end users, including the fact that they require
minimal equipment, being contactless, and ability to be set
up cost-effectively (using just an RGB camera, for example).

Geometrical and learning-based are two approaches have
been used to develop vision-based HPE, in which the
geometry-based ones analyze geometrical features (such
as facial landmarks) to estimate the head pose, while the
learning-based ones estimate it with machine learning tech-
niques. Geometry-based approaches are mainly built upon
two individual modules: i) performing landmark detection
and ii) processing the geometry of the landmarks to esti-
mate the pose. Amongst the first attempts, [28] analyzed the
geometry of five facial landmarks to estimate the head pose.
In [29], the authors analyzed the facial landmark geometry
to estimate the head pose via two cascade steps: first, they
identified the facial landmarks, and then they processed the
landmarks with respect to a virtual web-shaped network for
head pose estimation, in all three axes of yaw, pitch, and roll.

In a more advanced approach, [30] developed a novel
face ellipsoidal model to estimate the yaw pose of drivers’
heads, with the aid of some facial landmarks. Similarly, [31]
utilized a set of modified facial feature extractors, including
adaptive Hough transform [32], template matching, active contour model, and projective geometry properties to detect facial landmarks, and consequently estimate the yaw angles. In line with the geometry-based approaches, some other works attempt to estimate head pose from the correspondence of the features, extracted from a 2D image, and a 3D facial model [11]. This technique analyzes the projection relationship between a 3D facial model and 2D features to calculate the rotation matrix [33]–[36]. Another geometry-based approach [3], estimated the head pose with the aid of a Kinect sensor and three landmarks on the head. Given the high cost of Kinect, other studies have used inexpensive RGB web cameras to capture facial frames that seem to be more cost-effective [4], [37], [38].

On the other hand, we have the learning-based HPE models. Learning-based techniques aim to train a model for estimating the spatial head pose via appearance features. This spatial pose estimation can be either a classification, that classifies the input head images in specific position intervals (discrete), or a regression approach that estimates the head pose continuously. The features in this technique are mainly extracted automatically by convolutional neural networks that need to be trained with a large, annotated face dataset. For instance, [37] deployed a set of Gabor features (i.e., a linear filter used for texture analysis in image processing), along with a machine learning model (i.e., random forest algorithm), for face images classification. Due to the classification nature of this model, it is considered as a discrete HPE model. In another study, [38] proposed a new model for yaw and facial landmark estimation in the wild. Similar to [37], they have classified facial images into several classes of yaw angles with intervals of 15°. Following the great success of CNN in the extraction of features, [39] has trained a deep neural network to learn the mapping function between the visual appearance and the 3D head orientation angles. The authors developed their model as a regression model that finds the correlation of extracted features from a CNN. Similarly, [16] developed an HPE model based on a multi-loss neural network with a function to estimate each of the Euler angles.

### B. MULTITASK LEARNING

For all the HPE models discussed in the previous subsection, the presence of a face in the input image is an assumption. It means in practice the face must be first detected and then cropped from the original image before being fed into the HPE network. Multitask Learning (MTL) models can simplify this process by integrating a head detection step into a HPE model. Given the fact that the early layers of a deep CNN tend to learn generic features of an image, which can be also be useful for other tasks, the idea of sharing learned features for different tasks formed the first multitask learning models [40], [41]. Sharing features in MTL models, does not only lead to an increase in processing speed, but also to less biased features against the data of a particular task [42].

Despite the recent success of MTL models, their application in computer vision and object detection is still in its infancy. One of the few examples is [43], which introduced a multitask learning object detection model for detection of dangerous objects, by detecting an object and estimating its distance from the camera. The authors used a number of convolutional layers for the extraction of features, which were shared for both object detection and distance estimation. Similarly, [44] developed an MTL learning model for object detection and saliency estimation, trained from a non-jointly annotated dataset. In the context of HPE, a lot of efforts have also been put forward to develop an MTL head pose estimation model [22], [26], [27], [42], [45], [46]. Some of them tried to jointly estimate the head pose along with facial landmarks [27], [47], while others tried to detect the head, along with estimating its pose [42]. For instance, [48] used an Mask-RCNN model in a multitask learning setup for joint point estimation, orientation estimation, and body segmentation, by sharing the global features among all tasks. However, the effectiveness of such a network for head pose estimation remains unknown. In a similar way, [42] has developed a multitask learning approach to improve the performance of previous work by integrating a face detection step with feature extraction and pose estimation. Their model, outputs continuous values of head orientation and demonstrated a MAE of less than 4°. One issue with the existing multitask learning HPEs is that most of them are problem-specific models, with the sole goal of face detection and head pose estimation.

The ideas and challenges discussed above, have led us to develop a general purpose MTL model, whose use is not restricted in the HPE domain (hence “cross-domain”), but can be trained to determine an attribute of choice (e.g., other facial attributes, such as age), while performing object detection. In the next section, we present the proposed model architecture, the methodology of its implementation, as well as the training considerations.

### III. APPROACH

This section discusses the architecture of the proposed model, the hyper parameters, and evaluation metrics.

#### A. ARCHITECTURE

The overview of the proposed model is presented in Figure 2. It is important to mention that the backbone of this algorithm is adopted from Mask-RCNN [19]. The whole idea of the proposed network is described as follows. The input images are fed to both a RPN (Region Proposal Network) and a feature descriptor (i.e., Resnet50 for extraction of features from the input image). RPN is a network that identifies the prospective objects (also known as ROIs; Region of Interest) within images. ROIs are coordinates of rectangles (known as bounding boxes) that are likely to contain an object, which would be fed to another classifier to determine the class of the bounded object. In Mask-RCNN [19], the researchers have developed a novel, lightweight neural network that performs a preliminary object detection to extract the ROIs. The RPN network needs to be simultaneously
trained along with the object detection and the attribute estimation model. For training the RPN network, a window slides over the image with a certain stride (sliding steps). For each step, three different windows (called anchors in [20]) with three different aspect ratios (9 anchors in total) are created. \textit{RPN\_ANCHOR\_RATIOS} and \textit{RPN\_ANCHOR\_SCALES} are two hyper parameters of the RPN, representing the width-to-height ratio of the anchors and their sizes, respectively. For instance, for stride of one, in an image with dimensions $w \times h$, the model generates $w \times h \times 9$ anchors. As defined by [20], the anchors with an IOU (Intersection of Union; a metric that measures the overlap between two windows) greater than 70\% with the GT’s (ground truth) bounding box, are flagged as positive (foreground) and the ones with an IOU below 30\% are flagged as negative (background). These positive and negative target anchors are then used to train the RPN network. During the training process, the positive ROIs generated by the RPN with an IOU greater than a threshold (i.e. usually 50\%), are selected for training the object detector (a classifier to identify the class of the object) and other headers (e.g., attribute estimation model); this technique is known as NMS (non-Max Suppression). A certain number of the positive and negative ROIs are then cropped from the feature map and converted to two fixed-size feature maps with a technique called ROIAlign (see [20] for more info). The feature map’s size, which is the input for the bounding box and classifier network, remains at the size of $7 \times 7$, according to [19]. The cropped ROI for the pose estimation is resized to the fixed-size of $28 \times 28$. The feature maps are then connected to three sets of head networks, including a network for classifying objects within the proposals and fine tuning the bounding box coordinates, a network for generating masks, and yet another one for attribute (i.e., pose) estimation. The fixed-size $28 \times 28$ feature map is fed to a network that contains a series of convolutional layers, activation functions, and dense layers (see Figure 3).

In our pose estimation convolutional network, a fixed-size feature map is passed through two sets of Conv. Layer (kernel size: $3 \times 3$) + Max-pooling layer (window size: $2 \times 2$) + Batch Normalization + Activation function (ReLU), followed by one more Conv. Layer and some dense layers as shown in Figure 3. In the last layer, a linear activation function generates the full range of 0 to 1, which is then linearly mapped to the range of 0 to $\varphi_{\text{max}}$.

### B. MULTI-LOSS

The different tasks in neural networks result in different losses, making it necessary for multitask learning algorithms
to have a multi-loss function. For our model, we proposed a multi-loss function that combines the losses of the bounding box, classifier, segmentation, and pose estimation regressors. For the bounding box detection, the L1 loss function is implemented as below:

$$L_{BB} = \sum \text{Smooth L1}(BB_{True} - BB_{prediction}),$$  \hspace{1cm} (1)

where the Smooth L1 is defined as below:

$$\text{Smooth L1}(x) = \begin{cases} 0.5x^2 & \text{if } |x| < 1 \\ |x| - 0.5 & \text{if } |x| \geq 1 \end{cases} \hspace{1cm} (2)$$

Here, $BB_{prediction}$ is the vectorized tensor of the predicted bounding box with a length of 4 ($x, y, w, h$) and $BB_{True}$ is the true bounding box. In [19], the L1 loss function has been implemented to eliminate the malicious effect of potential outliers in bounding boxes. However, due to the restricted pose labels in our datasets, we implemented an L2-loss for training the pose regressor as:

$$L_{Pose} = \sum_{i=0}^{m} (GT_{Pose}^i - f(x)_{Pose}^i),$$  \hspace{1cm} (3)

where $GT_{pose}$ and $f(x)_{pose}$ are the real and predicted pose values of the $i^{th}$ instance. To train the classifier, to distinguish between face and non-face ROIs, the difference between the prediction and the GT is minimized by computing the softmax cross-entropy loss as:

$$L_{Class} = -\sum_{i=0}^{m} y_i \log x_i \hspace{1cm} (4)$$

In Eq. 4, let $y_i$ be the real class of the $i^{th}$ instance, $y_i \in \{0, 1\}$, and $x_i$ be the probability that the proposed region by RPN network contains a face or not. The combination of the individual loss functions, explained in this section, are jointly used for training the model. The next section describes the training process as well as the various hyper parameters.

**C. DATA AUGMENTATION**

Due to some factors like clearance, brightness, resolution, occlusion, etc., images taken in controlled environments are fundamentally different from those taken in the wild. This discrepancy can be detrimental to the performance of a model trained on a controlled-environment dataset in real-world scenarios, due to the lack of generalization. On the other hand, if the training dataset covers a variety of possible imaging conditions (i.e., well-diversified), the trained model will be well-generalized, and automatic translation invariance will be guaranteed [42]. However, the generation of such a diversified dataset is tedious and expensive. For bridging this gap, we have utilized a set of augmentation filters over the input images to enhance the dataset, both in quantity and quality, and improve the resulting model’s generalization capability.

Figure 4 demonstrates the augmentation of an original image with applied contrast, blur, Gaussian noise, pixelation, fog, rain, and snow filters. Varying weather conditions and camera vibration in the wild are among the most prevalent factors that can affect the quality of the captured image.

Vertical and horizontal flipping of images is one of the most common augmentation practices in the computer vision domain. However, this technique does not apply to this study because the datasets already contain the same angle on both sides of yaw, and therefore, flipping the images will add very little to no variability to the dataset due to its symmetric nature. Moreover, a tricky and very important consideration about the flipping augmentation is that the image flipping also requires the GT to be changed accordingly, to account for the reversed angle. See Figure 5 for further clarification.
TABLE 1. Training pipeline of our model.

| Requirement | Description |
|-------------|-------------|
| **Require:** | Head training datasets (BIWI, Prima) |
| | (\{Image, Yaw, Pitch, BB, Mask\})_i=1 |
| **Require:** | Training epoch of \#epoch = 10, \#iteration = 1000 |
| | \#RPN anchors = 500, \#ROIs per Image = 100, and positive ROIs = 30%, learning rate = 0.001 |
| **Require:** | ResNet50 model |

1: For i=0, 1, ..., \#epoch do:
2: For j=0, 1, T do:
3: For k=0, 1, \#iteration do:
4: \( \theta \leftarrow \sum_{i=1}^{N} \text{Smooth L1}(BB_{\text{true}} - BB_{\text{prediction}}) \)
5: \( \theta \leftarrow \sum_{i=1}^{N} \text{pose}_{\text{true}}(\theta) / \text{pose}_{\text{prediction}}(\theta) \)
6: \( \nu_{\text{class}} \leftarrow \sum_{i=1}^{N} \text{log} \nu_{\text{i}} \)
7: \( \theta \leftarrow \nu_{\text{class}} \)
8: End for k:
9: End for j:
10: End for i:
11: End
12: End

For training, we generated a jointly annotated dataset\(^1\) for multitask learning. Generated datasets have been annotated in the COCO format [49] that enables us to train our model, requiring the GT of the faces’ bounding boxes, the heads’ masks, the class label of the instance (face/non-face), and most importantly, the yaw and the pitch.

**B. TRAINING**

In order to evaluate our method, we have trained two individual models for each training dataset (Prima and BIWI). We used 70% of the datasets to train the models, while the rest was equally split between the testing and validation sets (15% for test and 15% for validation). The final global loss function for convergence of the model is declared as below:

\[
L_{\text{Global}} = \lambda_{BB}L_{BB} + \lambda_{\text{Class}}L_{\text{Class}} + \lambda_{\text{Pose}}L_{\text{Pose}},
\]

where \(\lambda\) denotes the weight for each loss term. Throughout the training process, each batch of data, contains the raw images, the GT for the bounding boxes, the segmentations, and the attribute values (i.e. yaw and pitch). The model is trained for 10 epochs with 1000 iteration per epoch. The learning rate is set to 0.001. For achieving a high processing speed, the image meta-size is set to 128 \(\times\) 128. Table 1 summarizes the hyper parameter values and the training pipeline.

\(^1\)www.ai-console.com

Given that not all of the proposed regions by the RPN contain a face, a NMS (non-max suppression) technique is implemented to eliminate negative (non-face) regions as explained in section IIIA. The NMS technique computes the overlap between the ROIs and the GT bounding boxes, as measured by IOU, and removes the proposed regions with an overlap below the threshold. As with most object detection approaches, the threshold of IOU for NMS is 50%. When it comes to joint face detection and pose estimation, the 50% threshold might degrade the performance, since the proposed region with an overlap of more than 50%, might be detected as positive, but a lot of information and face components might be lost on the other 50% [42]. Various techniques have been proposed to overcome this problem. For instance, [16] has used a Kinect depth sensor to detect a face area in the input images as a pre-processing step for the detection of face. In our approach, to avoid this issue, we set the NMS threshold to 80% to ensure that the proposed anchors cover the majority of the face’s characteristics. Then, the models have been trained with 70% of the datasets, while the rest was then used for evaluation. Figure 7 shows some of the learned features from the different layers of the Resnet50 descriptor. For training the model, given the limited number of faces within the image and in order to have a decent training time, we set the hyper parameters as follows. \(\text{Post}_{\text{NMS}}_{\text{ROIS}}_{\text{Training}} = 1000, \text{Train}_{\text{ROIS}}_{\text{Per}}_{\text{Image}} = 100, \text{RPN}_{\text{NMS}}_{\text{Threshold}} = 0.8,\) and \(\text{ROI}_{\text{Positive}}_{\text{Ratio}} = 0.33,\) which means 100 of the 1000 ROIs (generated by RPN), with a score above 0.8, and with a ratio of \(\frac{\text{ROIS}^+}{\text{ROIS}^-} = 0.33\) would be selected for training the headers. We reduced the number of the ROIs \(\text{Train}_{\text{ROIS}}_{\text{Per}}_{\text{Image}}\) from 1000 (i.e. as suggested by [19]) to 100, since we already knew that there is very few number of faces per image in the current datasets. Practitioners might need to increase the values if they want to train the model for crowded images.

As shown in Figure 8, the error for bounding box detection, pose estimation, as well as global loss, has converged exponentially.

The performance of the trained model, in terms of the head pose estimation is presented in the next subsection.
C. RESULTS
The performance of our proposed model is reported in this section. The performance is evaluated by the MAE (Mean Absolute Error) metric as:

$$\text{MAE} = \frac{1}{N} \sum_{i=0}^{N} |\hat{p}_i - p_i|,$$

where $N$ is the number of images, and $\hat{p}_i$ and $p_i$ represent the GT and the predicted pose respectively. Given the importance of real-time inference for such algorithms in real-world scenarios, and the fact that there is only one face per image, to increase the inference time we set the detection parameters as: $\text{Detection Max Instances} = 5$, $\text{Post NMS ROIS Inference} = 5$. In this case, the model selects 5 ROIs (generated by RPN) with the highest confidence score for detection of up to 5 instances. It is important to mention that the $\text{Post NMS ROIS Inference}$ has been set to 5, given that the lower the number, the higher the speed. For detection, also the $\text{Min Detection Confidence}$ was set to 70%, meaning that any detected instances by the model with a confidence score above 70% was considered as positive. The results of our models’ performance on both datasets are shown in Table 2.

D. GENERALIZATION TEST
CNN networks have shown promising results in extracting meaningful features for a wide range of facial attribute estimations, including gender, age, or hairstyle [52]. Therefore, we believe that our proposed model can also be applied to other domains, as its backbone is the standard Resnet-50 feature descriptor. To investigate the generalization capability of the proposed model, we have trained our model on the public dataset UTKFace for age estimation. This dataset contains $\sim$20,000 facial images of people in the range of 0 to 116 years old with wide variation in terms of illumination, pose, facial expression, etc. We used a randomly sampled portion of the dataset ($\sim$30%) for face detection, segmentation, and age estimation. Like the head pose estimation model, we jointly annotate the dataset, where the final annotation file contains the bounding box, masks, class labels, and the corresponding ages. Both output nodes of the attribute estimation header, which were initially developed for yaw and pitch estimation in the HPE problem (see Section III), were now assigned for age prediction. Figure 10 shows some example images of the dataset that are used both in training and in testing.
TABLE 2. Comparison of MAE of our model and different methods on two datasets: Prima and BIWI.

| Model          | Prima Yaw(SD) | Prima Pitch(SD) | BIWI Yaw(SD) | BIWI Pitch(SD) |
|----------------|---------------|-----------------|--------------|----------------|
| Drouard et al. [12] | 7.5(7.28) | 7.3(8.8) | 4.9(4.1) | 5.9(4.8) |
| Fanelli [17]    | -             | -               | 3.8(6.5) | 3.5(5.8) |
| B. Ahn et al [42] | -             | -               | 3.6(3.1) | 4.3(3.7) |
| Wang et al [13] | -             | -               | 8.8(14) | 8.5(11) |
| Valle (MNN) [27] | -             | -               | 3.98 | 4.61 |
| Kepler [50]     | -             | -               | 8.0 | 17.2 |
| HopeNet [16]    | -             | -               | 4.8 | 6.6 |
| Gourier [14]    | 12.1          | 7.3             | -           | - |
| Ricci [15]      | 9.1           | 10.5            | -           | - |
| Lee et al [51]  | 7.1           | 9.0             | 2.47 | 5.4 |
| Andrea et al [29]| 4.4           | 7.5             | -           | - |
| GLLiM [18]      | 7.9(7.9) | 8.4(10.3) | 4.2(5.3) | 5.4(5.4) |
| **Our Model**   | **8.0(8.6)** | **8.2(8.1)** | **6.2(4.7)** | **6.6(4.9)** |

**Face Detection Technique**
- HGK
- Auto
- Manually
- Faster R-CNN
- Skin color model
- Skin color model
- Viola-Jones

V. LIMITATIONS

We have developed a novel multitask cross-domain object detection model and tested its ability to detect faces and estimate facial attributes, including head pose or age. Although the proposed model has shown promising results and good generalization capability, there are still ways in which it can be improved. Practical deployment of the HPE models on an NVidia Xavier development board, when tested on the snapshots of a webcam stream, revealed that the HPE model is very sensitive to several factors, such as the distance between the camera and the face of the user, as well as the background of the snapshots. It is fair to assume that this is a matter of the training datasets’ limited diversity, and we believe that a collective effort is needed to generate a richer dataset to enable training a well-generalized model, suited for real-world applications. In addition, we recommend that future systems implement GANs [57] to generate cost-effective, diversified synthetic images in order to train a well-generalized HPE model for real-world applications.

Apart from the limitation discussed above, the test of the model on the NVidia Xavier also shows some outlier estimations which can be problematic if the system is intended to be used in a sensitive real-world scenarios like head-controlled EPWs [1], [2]. Fortunately, these outliers can be dampened by some techniques like moving average or Kalman filter, however, they can adversely affect the FPS of the system. The FPS of ~4.5 that was achieved on the specific platform may not be fast enough for some real-world applications. Therefore, in future studies optimizing the model speed needs to be a point of focus and further exploration. Using a shallower feature descriptor than Resnet50, or optimization of the headers by reducing their size and complexity, are some potential solutions that can be explored in the forthcoming studies. Furthermore, while determining the yaw and pitch may be adequate for some applications like head-controlled...
The proposed model shows promising results in situations where access to expensive equipment like depth camera sensors is not feasible. This is due to the requirement for manual cropping of the images or the need for having information (pose, age), which eliminates the requirement for monitoring driver awareness.

Moreover, our model’s smaller standard deviation demonstrates better consistency (i.e., less uncertainty) in terms of estimation (see Table 2). We also evaluate the generalization capability of our model by testing it on a different domain problem for age estimation. For this evaluation, our model was trained and tested on the public dataset UTKFace, for head detection and age estimation where we achieved a MAE of 5.3 ± 3.2.

The proposed multitask learning model parallelized the process of the object detection (i.e. head) and attribute estimation (pose, age), which eliminates the requirement for manual cropping of the images or the requirement of having access to expensive equipment like depth camera sensors (e.g. Kinect). The proposed model shows promising results and the potential to be used in various domains, while it maintains an advantage over the problem-specific state-of-the-art models by merging a two-stage process into a single one.

REFERENCES

[1] S. Kumar and N. Dheeraj, “Design and development of head motion controlled wheelchair,” Int. J. Adv. Eng. Technol., vol. 8, no. 5, pp. 816–822, 2015.

[2] Y.-L. Chen, S.-C. Chen, W.-L. Chen, and J.-F. Lin, “A head orientated wheelchair for people with disabilities,” Disability Rehabil., vol. 25, no. 6, pp. 249–253, Jan. 2003. doi: 10.1080/096382802100024979.

[3] F. A. Kondori, S. Yousefi, L. Liu, and H. Li, “Head operated electric wheelchair,” in Proc. Southwest Symp. Image Anal. Interpretation, 2014, pp. 53–56. doi: 10.1109/SSIAI.2014.6806027.

[4] P. Jia, H. H. Hu, T. Lu, and K. Yuan, “Head gesture recognition for hands-free control of an intelligent wheelchair,” Ind. Robot, Int., vol. 34, no. 1, pp. 60–68, Jan. 2007. doi: 10.1016/j.induto.2006.04.003.

[5] J. W. Machangpa and T. S. Chingtham, “Head gesture controlled wheelchair for quadriplegics patients,” Proc. Comput. Sci., vol. 132, pp. 342–351, Jan. 2018. doi: 10.1016/j.procs.2018.05.189.

[6] M. C. G. Quintero, J. O. López, and A. C. C. Pinilla, “Driver behavior classification model based on an intelligent driving diagnosis system,” in Proc. 15th Int. IEEE Conf. Intell. Transp. Syst., Sep. 2012, pp. 894–899. doi: 10.1109/ITSC.2012.6338727.

[7] S. Jha and C.Busso, “Analyzing the relationship between head pose and gaze to model driver visual attention,” in Proc. IEEE 19th Int. Conf. Intell. Transp. Syst. (ITSC), Nov. 2016, pp. 2157–2162. doi: 10.1109/ITSC.2016.7795905.

[8] E. Murphy-Chutorian and M. M. Trivedi, “Head pose estimation and augmented reality tracking: An integrated system and evaluation for monitoring driver awareness,” IEEE Trans. Intell. Transp. Syst., vol. 11, no. 2, pp. 300–311, Jun. 2010. doi: 10.1109/TITS.2010.2044241.

[9] Y. Yan, E. Ricci, R. Subramanian, G. Liu, O. Lanz, and N. Sebe, “A multitask learning framework for head pose estimation under target motion,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 38, no. 6, pp. 1070–1083, Jun. 2016. doi: 10.1109/TPAMI.2015.2477843.

[10] J. Chen, N. Luo, Y. Liu, L. Liu, K. Zhang, and J. Kolodziej, “A hybrid intelligence-aided approach to affect-sensitive e-learning,” Computing, vol. 98, nos. 1–2, pp. 215–233, Jan. 2016. doi: 10.1007/s00122-014-0430-9.

[11] H. Yuan, M. Li, J. Hou, and X. Jiang, “Single image-based head pose estimation with spherical parameterization and 3D morphing,” Pattern Recognit., vol. 103, Jul. 2020, Art. no. 107316. doi: 10.1016/j.patcog.2020.107316.

[12] V. Drouard, S. Ba, G. Evangelidis, A. Deleforge, and R. Horard, “Head pose estimation via high-dimensional regression,” in Proc. IEEE Int. Conf. Image Process. (ICIP), Sept. 2015, pp. 4624–4628. doi: 10.1109/ICIP.2015.7351683.

[13] B. Wang, W. Liang, Y. Wang, and Y. Liang, “Head pose estimation with combined 2D SIFT and 3D HOG features,” in Proc. 7th Int. Conf. Image Graph., Jul. 2013, pp. 650–655. doi: 10.1016/j.icig.2013.133.

[14] N. Gourier, J. Maisonasse, D. Hall, and J. L. Crowley, “Head pose estimation in low resolution images,” in Proc. Int. Eval. Workshop Classification Events, Activities Relationships, in Lecture Notes in Computer Science: Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics, vol. 4122, 2007, pp. 270–280. doi: 10.1007/978-3-540-69568-4_24.

[15] E. Ricci and J.-M. Odobez, “Learning large margin likelihoods for realtime head pose tracking,” in Proc. 16th IEEE Int. Conf. Intell. Process. (ICIP), Nov. 2009, pp. 2593–2596. doi: 10.1109/ICIP.2009.5413994.

[16] N. Ruiz, E. Chong, and J. M. Rehg, “Fine-grained head pose estimation without keypoints,” in Proc. IEEE/CVF Comput. Vis. Pattern Recognit. Workshops (CVPRW), Jun. 2018, pp. 2074–2083. doi: 10.1109/CVPRW.2018.00281.

[17] G. Fanelli, M. Dantone, J. Gall, A. Fossati, and L. Van Gool, “Random forests for real time 3D face analysis,” Int. J. Comput. Vis., vol. 101, no. 3, pp. 437–458, 2013. doi: 10.1007/s11263-012-0540-9.

[18] V. Drouard, R. Horard, A. Deleforge, S. Ba, and G. Evangelidis, “Robust head-pose estimation based on partially-latent mixture of linear regressions,” IEEE Trans. Image Process., vol. 26, no. 3, pp. 1428–1440, Mar. 2017. doi: 10.1109/TIP.2017.2654165.

[19] K. He, G. Gkioxari, P. Dollar, and R. Girshick, “Mask R-CNN,” in Proc. IEEE Int. Conf. Intell. Vis. (ICCV), Oct. 2017, pp. 2961–2969. doi: 10.1109/ICCV.2017.322.

[20] S. Wang, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards real-time object detection with region proposal networks,” in Advances in Neural Information Processing Systems, 2015.

[21] P. Viola and M. Jones, “Rapid object detection using a boosted cascade of simple features,” in Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., vol. 1, Dec. 2001, p. 1. doi: 10.1109/CVPR.2001.990517.

[22] S. Chen, Y. Zhang, B. Yin, and B. Wang, “TRFH: Towards real-time face detection and head pose estimation,” Pattern Anal. Appl., vol. 14, no. 4, pp. 1745–1755, Nov. 2021. doi: 10.1007/s10044-021-01026-3.

[23] J. Deng, J. Guo, Y. Zhou, J. Yu, L. Kotsia, and S. Zafeiriou, “RetinaFace: Single-stage dense face localisation in the wild,” 2019, arXiv:1905.06041.

[24] D. Yoshunin, T. Baydasov, and R. Vlasyuk, “MaskFace: Multi-task face and landmark detector,” 2020, arXiv:2005.09412.

[25] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, “Joint face detection and alignment using multitask cascaded convolutional networks,” IEEE Signal Process. Lett., vol. 23, no. 10, pp. 1499–1503, Oct. 2016. doi: 10.1109/LSP.2016.2603342.

[26] C. Hong, J. Yu, J. Zhang, X. Jin, and K.-H. Lee, “Multimodal face-pose estimation with multitask manifold deep learning,” IEEE Trans. Ind. Informat., vol. 15, no. 7, pp. 3952–3961, Jul. 2019. doi: 10.1109/TII.2018.2884211.

[27] R. Valle, J. M. Buenaposada, and L. Baumela, “Multi-task head pose estimation in-the-wild,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 43, no. 8, pp. 2874–2881, Aug. 2021. doi: 10.1109/TPAMI.2020.3046323.
A. Gee and R. Cipolla, “Determining the gaze of faces in images,” Image Vis. Comput., vol. 12, no. 10, pp. 639–647, 1994, doi: 10.1016/0731-5585(94)90009-6.

A. F. Abate, P. Barra, C. Pero, and M. Tucci, “Head pose estimation by regression algorithm,” Pattern Recognit. Lett., vol. 140, pp. 179–185, Dec. 2020, doi: 10.1016/j.patrec.2020.10.005.

A. Narayanay, R. M. Kaimal, and K. Biljani, “Yaw estimation using cylindrical and ellipsoidal face models,” IEEE Trans. Intell. Transp. Syst., vol. 15, no. 5, pp. 2308–2320, Oct. 2014, doi: 10.1109/TITS.2014.2313371.

A. Nikolaidis and I. Pitas, “Facial feature extraction and pose determination,” Pattern Recognit., vol. 33, no. 11, pp. 1783–1791, 2000, doi: 10.1016/S0031-3203(99)00176-4.

J. Illingworth and J. Kittler, “The adaptive Hough transform,” in Proc. 2nd Int. Conf. 3D Vis., Dec. 2014, pp. 641–648, doi: 10.1109/3DV.2014.54.

G. P. Meyer, S. Gupta, I. Frosio, D. Reddy, and J. Kautz, “Robust model-based 3D head pose estimation,” in Proc. IEEE Int. Conf. Comput. Vis. (ICCV), Dec. 2015, pp. 3649–3657, doi: 10.1109/ICCV.2015.416.

S. G. Kong and R. O. Mbouna, “Head pose estimation from a 2D face image using 3D face morphing with depth parameters,” IEEE Trans. Image Process., vol. 24, no. 6, pp. 1801–1808, June 2015, doi: 10.1109/TIP.2015.2405483.

S. Li, K. N. Nigan, R. Paramesran, and L. Sheng, “Real-time head pose tracking with online face template reconstruction,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 38, no. 9, pp. 1922–1928, Sep. 2016, doi: 10.1109/TPAMI.2015.2500221.

B. Huang, R. Chen, W. Xu, and Q. Zhou, “Improving head pose estimation using two-stage ensembles with top-k regression,” Image Vis. Comput., vol. 93, Jan. 2020, Art. no. 103827, doi: 10.1016/j.imavis.2019.11.005.

X. Zhu and D. Ramanan, “Face detection, pose estimation, and landmark localization in the wild,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2012, pp. 2879–2886, doi: 10.1109/CVPR.2012.6248014.

B. Ahn, J. Park, and I. S. Kweon, “Real-time head orientation from a monocular camera using deep neural network,” in Proc. Asian Conf. Comput. Vis., 2015, pp. 92–96, doi: 10.1007/978-3-319-16811-1_6.

R. Caruana, “Multitask learning,” Mach. Learn., vol. 28, pp. 41–75, Jul. 1997, doi: 10.1023/A:1007379606734.

S. Thrun, “Is learning the n-th thing any easier than learning the first?” in Proc. Adv. Neural Inf. Process. Syst., 1996, pp. 640–646.

B. Ahn, D.-G. Cho, J. Park, and I. S. Kweon, “Real-time head pose estimation using multi-task deep neural network,” Robot. Auto. Syst., vol. 103, pp. 1–12, May 2018, doi: 10.1016/j.robot.2018.01.005.

Y. Chen, D. Zhao, L. Lv, and Q. Zhang, “Multi-task learning for dangerous object detection in autonomous driving,” Inf. Sci., vol. 432, pp. 559–571, Mar. 2018, doi: 10.1016/j.ins.2017.08.035.

A. Khatir, S. Hedege, and R. Hebbalaguppe, “Cross-domain multi-task learning for object detection and saliency estimation,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW), Jun. 2021, pp. 3634–3643, doi: 10.1109/CVPRW5309.2021.00403.

Y. Gu, H. Zang, and S. Kamijo, “Multi-person pose estimation using an orientation and occlusion aware deep learning network,” Sensors, vol. 20, no. 6, pp. 1593, Mar. 2020, doi: 10.3390/s20061593.

R. Ranjan, V. M. Patel, and R. Chellappa, “HyperFace: A deep multi-task learning framework for face detection, landmark localization, pose estimation, and gender recognition,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 41, no. 1, pp. 121–135, Jan. 2019, doi: 10.1109/TPAMI.2017.2781233.

T. Chuan, H. Xinrui, W. Zhicheng, Z. Yu, X. Mingyu, and W. Xin, “Head pose estimation via multi-task cascade CNN,” in Proc. 3rd High Perform. Comput. Cluster Technol. Conf. Jun. 2019, pp. 123–127, doi: 10.1145/3341069.3342979.

Y. Gu, H. Zang, and S. Kamijo, “Multi-person pose estimation using an orientation and occlusion aware deep learning network,” Sensors, vol. 20, no. 6, pp. 1593, Mar. 2020, doi: 10.3390/s20061593.

T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, “Microsoft COCO: Common objects in context,” in Proc. Eur. Conf. Comput. Vis., 2014, pp. 740–755, doi: 10.1007/978-3-319-01519-4.

A. Kumar, A. Alavi, and R. Chellappa, “KEPLER: Simultaneous estimation of keypoints and 3D pose of unconstrained faces in a unified framework by learning efficient H-CNN regressors,” Image Vis. Comput., vol. 79, pp. 49–62, Nov. 2018, doi: 10.1016/j.imavis.2018.09.009.

S. Lee and T. Saito, “Head pose estimation using convolutional neural network,” in IT Convergence and Security 2017 (Lecture Notes in Electrical Engineering), vol. 449. Singapore: Springer, 2017, pp. 164–171, doi: 10.1007/978-981-10-6451-7_20.

A. F. Abate, P. Barra, S. Barra, C. Molinari, M. Nappi, and F. Narducci, “Clustering facial attributes: Narrowing the path from soft to hard biometrics,” IEEE Access, vol. 8, pp. 9037–9045, 2020, doi: 10.1109/ACCESS.2019.2962010.

A. Al-Shamaq and L. Elrefaei, “Age estimation using specific domain transfer learning,” Jordanian J. Comput. Inform. Technol., vol. 6, no. 2, pp. 122–139, 2020.

A. Fariza, M. Arifin, and A. Z. Arifin, “Age estimation system using deep residual network classification method,” in Proc. Int. Electron. Syst. (IES), Sep. 2019, pp. 607–611, doi: 10.1109/ELECSYM.2019.8901521.

W. Cao, V. Mirjalili, and S. Raschka, “Rank consistent ordinal regression for neural networks with application to age estimation,” Pattern Recognit. Lett., vol. 140, pp. 325–331, Dec. 2020, doi: 10.1016/j.patrec.2020.11.008.

Z. Niu, M. Zhou, L. Wang, X. Gao, and G. Hua, “Ordinal regression with multiple output CNN for age estimation,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2016, pp. 4920–4928, doi: 10.1109/CVPR.2016.532.

P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, “Image-to-image translation with conditional adversarial networks,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jul. 2017, pp. 1125–1134, doi: 10.1109/CVPR.2017.632.