2016 23rd International Conference on Pattern Recognition (ICPR)
Cancún Center, Cancún, México, December 4-8, 2016

Robust Hand Detection in Vehicles

T. Hoang Ngan Le, Chenchen Zhu, Yutong Zheng, Khoa Luu and Marios Savvides
CyLab Biometrics Center and the Department of Electrical and Computer Engineering,
Carnegie Mellon University, Pittsburgh, PA, USA
Email: {thihoanl, chenchez, yutongzh, kluu }@andrew.cmu.edu, msavvid@ri.cmu.edu

***Abstract*—The problems of hand detection have been widely addressed in many areas, e.g. human computer interaction environment, driver behaviors monitoring, etc. However, the detection accuracy in recent hand detection systems are still far away from the demands in practice due to a number of challenges, e.g. hand variations, highly occlusions, low-resolution and strong lighting conditions. This paper presents the Multiple Scale Faster Region-based Convolutional Neural Network (MS-FRCNN) to handle the problems of hand detection in given digital images collected under challenging conditions. Our proposed method introduces a multiple scale deep feature extraction approach in order to handle the challenging factors to provide a robust hand detection algorithm. The method is evaluated on the challenging hand database, i.e. the Vision for Intelligent Vehicles and Applications (VIVA) Challenge, and compared against various recent hand detection methods. Our proposed method achieves the state-of-the-art results with 20% of the detection accuracy higher than the second best one in the VIVA challenge.**

I. Introduction

The problems of hand detection have been studied for years with the aim of ensuring the generalization of robust unconstrained hand detection algorithms to unseen images. However, the detection accuracy in recent hand detection systems [1], [2] are still far away from the demands in practice due to a number of challenges. Particularly, the hand variations, highly occlusions, low-resolution and strong lighting conditions, as shown in Figure 1, are the important factors that need to be considered. Meanwhile, blurring of colors due to hand movement, skin tone variation in recorded videos due to camera quality are also the other difficulties in this problem.

This paper presents a Convolutional Neural Network (ConvNet) based approach named Multiple Scale Faster Region-based Convolutional Neural Network (MS-FRCNN) to handle the problems of hand detection in given digital images collected under challenging conditions, e.g. hand variations, strong lighting, occlusions, low-resolution, etc. Our proposed method extends the framework of Faster-RCNN [3] with the significant modification in the multiple scale deep feature extraction in both the Regional Proposal Network (RPN) and the detection network in order to handle the challenging factors to provide a robust hand detection algorithm in the wild. The method takes the advantages of the Multiple Scale Regional Proposal Network (MS-RPN) to introduce a set of region proposals and the Multiple Scale Region-based Convolutional Neural Network (MS-RNN) to extract the regions of interest (RoI), i.e. regions of hands. Each RoI is then assigned to a confidence score.

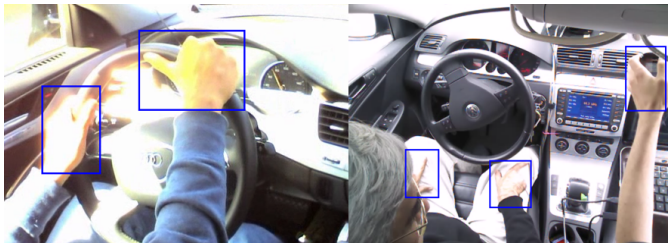


Fig. 1. Some examples of hand detection results using our proposed MS-FRCNN method on VIVA database [5]. Our proposed method can robustly detect hands across variations, occlusions, strong illumination and low resolution conditions.

The design of the proposed deep network can be seen in Figure 2. The deep learning Caffe framework [4] is employed in our implementation. The experiments are presented on the challenging hand database, i.e. the Vision for Intelligent Vehicles and Applications (VIVA) Challenge [5]. Our proposed method achieves the state-of-the-art results[1] in the problem of hand detection on VIVA database.

The rest of this paper is organized as follows. In section II, we review prior work on hand detection, the standard Faster-RCNN network in object detection and its limitations in the problem of hand detection. In section III, we present our proposed approach to detect hands from given input images in the wild. Section IV presents experimental results obtained using our proposed approach on the challenging hand detection database, i.e. VIVA challenge database. Finally, our conclusions on this work are presented in Section V.

II. Related Work

In this section, we first review prior approaches in hand detection. Then, we summarize a general deep learning framework and the Faster-RCNN method. Finally, we present the limitations of the Faster R-CNN method in the defined hand detection problem.

A. Hand Detection

Detecting and tracking of human hands have been widely addressed in many areas, such as: virtual reality, human computer interaction environment, driver behavior monitoring. In this paper, we focus on hands in vehicles [5] (of a driver) detection. Indeed, a robust hand detection system not only helps to study driver behavior and alertness but also provides

[1]Submission date: Apr. 3rd, 2016, the VIVA hand detection ranking can be seen at http://cvrr.ucsd.edu/vivachallenge/index.php/hands/hand-detection/

978-1-5090-4846-5/16/$31.00 ©2016 IEEE

document and human-machine interaction features. One of the first well performing approaches to detect the human hands was proposed by Mittal et al. [6]. They presented a two-stage approach to detect hands in unconstrained images. Three complementary detectors are employed to propose hand bounding boxes. These proposal regions are then used as inputs to train a classifier to compute a final confidence score. In their method, the context-based and skin-based proposals with a sliding window shape based detector are used to increase the recall. However, these skin-based features cannot contribute in our presented problem since all videos are recorded under poor illumination and gray-scale level. Later, Ohn-Bar et al. [7] introduced a vision-based system that employs a combined RGB and depth descriptor in order to classify hand gestures. The method employs various modifications of HOG features with the combination of both RGB and depth images to achieve a high classification accuracy. Ohn-Bar et al. [8] also introduced the multimodal vision method to characterize driver activities based on head, eye and hand cues. The fused cues from these three inputs using hierarchical Support vector Machines (SVM) enrich the descriptions of the driver's state allowing for evaluation of driver performance captured in on-road settings. However, this method with a linear kernel SVM for detection focuses more on analyzing the activities of the driver correlated among these three cues. It does not emphasize the accuracy of hand detection of drivers in challenging conditions, e.g. shadow, low resolution, phone usage, etc. Meanwhile, these proposed methods [9], [10], [11] for hand tracking and analysis are only applicable in depth images with high resolution. They are therefore unusable in the types of videos used in this work.

Unlike all the previous approaches that select a feature extractor beforehand and incorporate a linear classifier with the depth descriptor beside RGB channels, our method solves the problem under a deep learning framework where the global and the local context features, i.e. multi scaling, are synchronized to Faster Region-based Convolutional Neural Networks in order to robustly achieve semantic detection.

B. Deep Learning Framework

Convolutional Neural Networks, one of the most successful approaches to object detection, can be seen as a variant of multilayer perceptrons. The key ideas of ConvNet based methods aim to simulate the animal visual cortex system containing a complex arrangement of cells sensitive to receptive fields. In the defined models, the implemented filters are designed as human visual cells to spatially explore local correlations in an observed image. It efficiently presents the sparse connectivity and the shared weights since these kernel filters are replicated over the entire image with the same parameters in each layer. The pooling step, a form of down-sampling, has important role in a defined ConvNet network. Indeed, max-pooling is one of the most well-known pooling methods for object detection and classification because it reduces the computational complexity in upper layers. This step is processed by eliminating non-maximal values and provides a small amount of translation

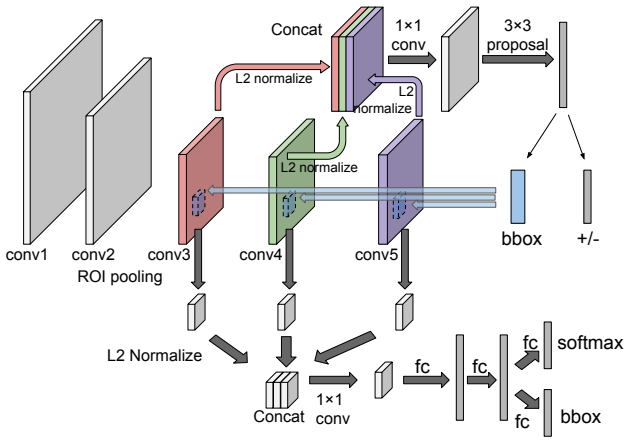


Fig. 2. Our proposed Multiple Scale Faster RCNN approach to robust hand detection. The Multiple Scale Regional Proposal Network and Multiple Scale Region-based CNN share the first five convolutional layers. Then, the multiple scale features from convolutional layers 3, 4 and 5 are used to enhance the receptive field representation.

invariance in each level. ConvNet is efficiently to explore highly discriminative deep features. However, this method is computationally expensive. The computational cost in the algorithm can be qualified when it is implemented in a Graphics Processing Unit (GPU). The Caffe framework [4] is a rapid deep learning implementation using CUDA C++ for GPU computation. This framework also has a capability to binding to Python/Numpy and MATLAB environment.

C. Faster R-CNN

Given a set of object proposals extracted from an image, the Region-based Convolutional Neural Network (R-CNN) method [12] employs a deep ConvNet to classify these proposals. This method is able to achieve high accuracy in the problem of object detection. However, it costs computational time. Firstly, the object proposals in an image are generated using Selective Search algorithm or Multiscale Combinatorial Grouping (MCG) method [13]. These object proposals are then used as inputs to train the deep ConvNet network and fine-tunes it with a softmax regression layer in the final step. By swapping the last layer with the Support Vector Machines (SVM) and using the features from fine-tuned ConvNet, the system is further trained for object detection. Finally, it performs bounding-box regression. However, given a large-scale image database, this system usually takes a lot of time to extract features from each image and physically store those extracted features in a hard disk, taking a large amount of space. At test-time, the detection process takes 47s for one image (with VGG16, on a GPU) due to the highly computational time in the feature extraction steps.

Fast R-CNN method aims to reduce the computational time in the detection network using the ROI-pooling layer. However, the computational steps in the region proposal are still beyond the network. Therefore, it remains a bottleneck,

resulting in sub-optimal solution and dependence on the external region proposal methods. Recently, Faster R-CNN [3] has been introduced to address this problem by using the Region Proposal Network (RPN). An RPN can be considered as a fully convolutional network to predict the object bounds and the objectness scores. It uses anchors with different scales and shapes to achieve translation invariance. The total computational time for both the proposal and the detection steps are within 0.2 seconds using very deep VGG-16 model. It is because the detection network shares the full-image convolution features with the RPN network.

D. Drawbacks of Faster R-CNN in Hand Detection

Faster R-CNN has been employed in object detection, e.g. persons, animals, vehicles, etc., on PASCAL VOC dataset with the state-of-the-art detection accuracy. However, the objects in this database usually occupy the majority of an image, i.e. these objects have considerable numbers of pixels. However, in our problem, we are interested in detecting human hands in the wild that are usually small and have low resolution as shown in Fig.1. Unfortunately, the detection network in Faster R-CNN has trouble to detect such small objects since generally it cannot find the human hands with small sizes. It is because the ROI-pooling layer builds features only from one single high level feature map. Indeed, the VGG-16 model employs ROI-pooling from the 'conv5' layer with an overall stride of 16. Therefore, given an object with the sizes smaller than 16 pixels, the region of the projected ROI-pooling will be less than 1 pixel in the 'conv5' layer, although the region provided by the RPN is correct. In this case, the detection network will be challenged to estimate the object class and regress the bounding box location based on these limited extracted features.

III. Our Approach to Robust Hand Detection

Our proposed Multiple Scale Faster-RCNN approach is presented to robustly detect hands in challenging hand databases. Our approach utilizes both the global and the local deep features to encode human hands in images. However, the scaling ranges of the filter responses are in different from layer to layer. Therefore, there is a process to further calibrate these values. The average features for layers in Faster-RCNN are employed to augment features at each location.

A. Multiple Scale Faster-RCNN

In the defined problem, human hands in observed images are usually collected under variations, low-resolution, highly occlusion and strong lighting conditions. Therefore, the standard Faster R-CNN is very hard to robustly detect these hand objects. The receptive fields in the last convolution layer (conv5) in the standard Faster R-CNN is quite large. For example, given a hand ROI region of sizes of 64×64 pixels in an image, its output in conv5 only contains 4×4 pixels, which is insufficient to encode informative features. To make it even worse, as the convolution layers go deeper, each pixel in the corresponding feature map gather more and more convolutional information outside the ROI region. Thus, it contains higher proportion of information outside the ROI region if the ROI is really small. The two problems together, make the feature map of the last convolution layer less representative for small ROI regions. Thus, an approach to combine both global and local features to enhance the robustness of the deep features can help to detect hands in images in the wild. In order to enhance this capability of the network, the feature maps from shallower convolution feature maps, i.e. conv3 and conv4, are incorporated to the deeper one, i.e. conv5, in both RPN and ROI pooling, as shown in Figure 2. Therefore, in the ROI regions, the network can detect lower level features containing higher proportion of human hand features.

In our implementation, both Regional Proposal Network and R-CNN are employed in multiple scales in order to train these hand proposals at various scales. Our network includes five sharing convolution layers, i.e. conv1, conv2, conv3, conv4 and conv5 as defined in [3]. In the first two convolution layers of the network, there are an ReLU layer, an LRN layer and a Max-pooling layer designed respectively after each convolution layer. In the last three convolution layers, each convolution layer is followed with only one ReLU layer. Especially, in the last three convolution layers, i.e. conv3, conv4 and conv5, their outputs are also used as the input to three corresponding ROI pooling layers and normalization layers as shown in Fig. 2. These weight normalization outputs are concatenated and shrunk to use as the input for the next two fully connected layers. In the final steps, there are both a softmax layer for object classification and a regression function to take care of bounding box refinement.

B. Weight Normalization

The limitations revealed in single layer feature extraction triggered our implementation of the combination of multiple convolution layers. However, without weight normalization layers, the naive concatenation of the three feature map tensors turns out to be problematic.

Convolution feature maps are generally different in terms of their numbers of channels, scale of values and norm of feature map pixels. In practice, our observation concludes that the smaller-scaled values often appear in deeper layers while larger-scaled values dominate the shallower layers. As a result, by naively concatenating the feature map tensors, the result often turns out to be less robust since the system fails to tune the downstream parameters for each feature map tensor, rendering the "larger" features dominate the "smaller" ones.

The normalization steps in each feature map tensor are straightforward as the issue discussed above [14]. Additionally, the number of channels of these tensors is able to be changed. Therefore, a scaling factor for each tensor is employed in order to scale the output and increase the robustness of the system. In this implementation, each concatenated feature map tensor passes through the normalization layer. Then,

weight normalization is implemented within each pixel, i.e. normalizing along the channel axis, in the tensor:

$$\hat{\mathbf{x}} = \frac{\mathbf{x}}{\|\mathbf{x}\|_2}$$
$$\|\mathbf{x}\|_2 = (\sum_{i=1}^{d} |x_i|)^{\frac{1}{2}}$$

where the $\mathbf{x}$ and $\hat{\mathbf{x}}$ stand for the original pixel vector and the normalized pixel vector respectively. d stands for the number of channels in each feature map tensor.

Then each channel in the tensor will multiply a scaling factor γ_i:

$$y_i = \gamma_i \hat{x}_i$$

In the training steps, the scaling factor γ are recalculated. In addition, the input $\mathbf{x}$ is also updated using back-propagation and chain rule:

$$\frac{\partial l}{\partial \hat{\mathbf{x}}} = \frac{\partial l}{\partial \mathbf{y}} \cdot \gamma$$
$$\frac{\partial l}{\partial \mathbf{x}} = \frac{\partial l}{\partial \hat{\mathbf{x}}} \left(\frac{\mathbf{I}}{\|\mathbf{x}\|_2} - \frac{\mathbf{x}\mathbf{x}^T}{\|\mathbf{x}\|_2^3} \right)$$
$$\frac{\partial l}{\partial \gamma_i} = \sum_{y_i} \frac{\partial l}{\partial y_i} \hat{x}_i$$

where $\mathbf{y} = [y_1, y_2, ..., y_d]^T$.

C. New Layer in Deep Learning Caffe Framework

In RPN, normalization layers are added to perform weight normalization to each pooled feature map, i.e. from the third, forth and fifth convolution layer. After that, the data are scaled according to the scaling factor, which is initialized carefully to ensure the downstream features in a similar scale as the original work in Faster-RCNN.

In detection network, similar to RPN, two more ROI pooling layers are used for extracting features from the third and forth convolution feature maps. Normalization layers are then added to perform weight normalization to each ROI pooling tensor, which then scaled and concatenated the same manner as the RPN. In order to return the same channel size as the Faster-RCNN feature map tensors, an additional 1×1 convolution layer is applied after concatenation layers both in RPN and detection network, as shown in Fig. 2.

IV. Our Experimental Results

This section is organized as follows. Firstly, the experimental databases are introduced in subsection IV-A. Then, subsection IV-B describes the evaluation protocols using in this work. Finally, subsection IV-C, we present the experimental results on VIVA hand database.

A. Database Collection

The Vision for Intelligent Vehicles and Applications Challenge [5] consists of 2D bounding boxes around hands of drivers and passengers from 54 videos collected in naturalistic driving settings of illumination variation, large hand movements, and common occlusion. There are 7 possible viewpoints, including first person view. Some of the data was captured in test beds, while some was kindly provided by YouTube. In the challenging evaluation protocol, the standard evaluation set consists of 5,500 training and 5,500 testing images.

B. Evaluation Methods

To evaluate the performance on VIVA database, we compute the average precision (AP), average recall (AR) rate, and frame per section (PFS). AP is the area under the Precision-Recall curve whereas AR is calculated over 9 evenly sampled points in log space between 10^{-2} and 10^{0} false positives per image. A hand detection is considered true or false according to its overlap with the ground-truth bounding box. A box is positive if the overlap score is more than 0.5. The overlap score between two boxes is defined as $\frac{GT \cap DET}{GT \cup DET}$, where GT is the axis aligned bounding rectangle around area ground-truth bounding box and DET is the axis aligned rectangle around detected bounding box. The hand detection challenge is evaluated on two levels: Level-1 (L1): hand instances with minimum height of 70 pixels, only over the shoulder (back) camera view. Level-2 (L2): hand instances with minimum height of 25 pixels, all camera views. The proposed method is evaluated on a 64 bits Ubuntu 14.04 computer with CPU Intel(R) Core(TM) i7-4770K CPU@ 3.50GHz and Matlab 2014a.

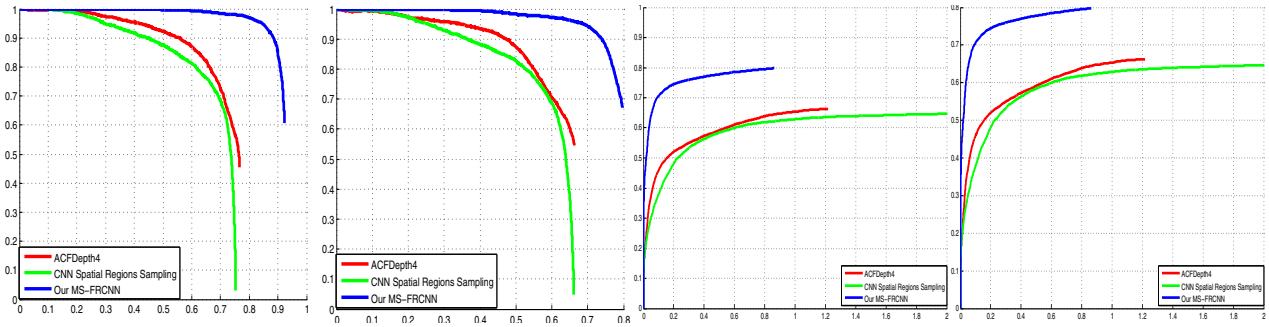
C. Hand Detection on VIVA database

Table I summaries the performance of our proposed approach, Das et al. [1], and Bambach et al. [2] using the measurements of AP, AR and FPS at both levels. Compare to the state-of-the-art methods, our proposed approach is higher 17.5% on L2-AP and higher than 20.7% on L1-AP whereas the AR obtained by our system is better from 24.7% to 30.3% on L2-AR, L1-AR, respectively. Processing time by [1] was not reported yet while the one by [2] takes 0.783 FPS on GPU environment and our testing time is 0.234 FPS on GPU.

Fig. 3 visualizes the AP and the AR rates at both levels (L1 and L2). From Table I and Fig. 3, we can see that the proposed MS-FRCNN outperforms others in higher AP, AR and less processing time. From Fig. 3(a, b), our AP is almost [1], [2] when Recall less than 0.2. As Recall increases from 0.2 to 1.0, Precision obtained by [1], [2] dramatically regraded while our Precision still remains at high scores. Some illustrations of hand detection by our proposed method on VIVA database is given in Fig.4

V. Conclusion

This paper has presented the MS-FRCNN approach to handle the problems of hand detection in images collected in vehicles under challenging conditions. The proposed method employs a multiple scale deep feature to provide a robust hand detection system. The method is evaluated on the challenging hand databases, i.e. VIVA Challenge, and compared against



(a) (b) (c) (d)

Fig. 3. ROC curves on AP and AR obtained by [1] (green), [2] (red) and our proposed MS-FRCNN (blue) on VIVA database. (a): L1-AP, (b): L2-AP, (c): L1-AR, (d): L2-AR. Our method achieves the state-of-the-art results on this database.

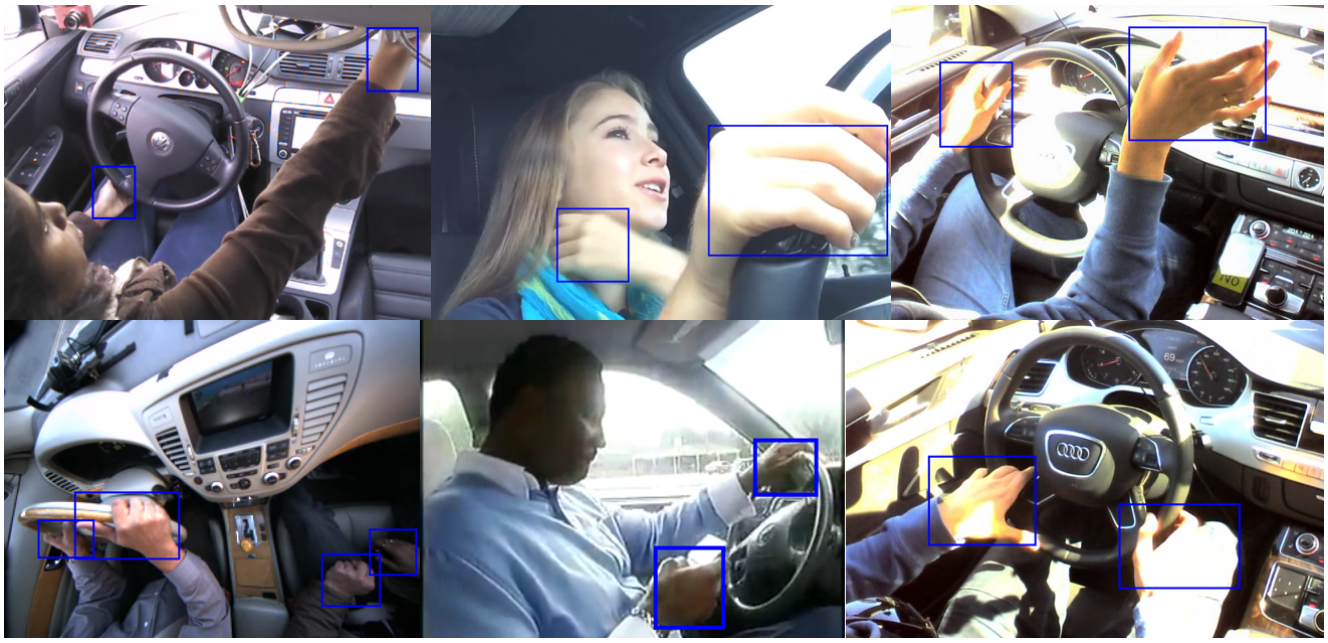


Fig. 4. Some examples of hand detection result using our proposed MS-FRCNN method on VIVA database [5].

TABLE I
PERFORMANCE OF DAS ET AL. [1], BAMBACH ET AL. [2], AND OUR PROPOSED MS-FRCNN ON PRECISION-RECALL CURVE (AP), AVERAGE RECALL (AR) RATE AND FRAME PER SECOND (FPS) AT BOTH LEVELS (L1 AND L2)

Methods	L1-AP	L2-AP	L1-AR	L2-AR	FPS
Das et al. [1]	70.1	60.1	53.8	40.4	
Bambach et al. [2]	66.8	57.8	48.1	36.6	0.783
MS-FRCNN	**90.8**	**77.6**	**84.1**	**65.1**	**0.234**

various recent hand detection methods. Our proposed MS-FRCNN is able to achieve the state-of-the-art results on VIVA with 20% of the detection accuracy higher than the second best one in the challenge.

REFERENCES

[1] N. Das, E. Ohn-Bar, and M. Trivedi, "On performance evaluation of driver hand detection algorithms: Challenges, dataset, and metrics," in *In IEEE Conf. Intelligent Transportation Systems*, 2015, pp. 2953 – 2958.

[2] S. Bambach, D. Crandall, and C.Yu, "Viewpoint integration for hand-based recognition of social interactions from a first-person view," in *Proceedings of the 17th ACM International Conference on Multimodal Interaction (ICMI)*, 2015, pp. 351–354.

[3] S. Ren, K. He, R. B. Girshick, and J. Sun, "Faster R-CNN: towards real-time object detection with region proposal networks," *CoRR*, vol. abs/1506.01497, 2015. [Online]. Available: http://arxiv.org/abs/1506.01497

[4] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," *arXiv preprint arXiv:1408.5093*, 2014.

[5] *The Vision for Intelligent Vehicles and Applications (VIVA) Challenge, Laboratory for Intelligent and Safe Automobiles, UCSD*, http://cvrr.ucsd.edu/vivachallenge/.

[6] A. Mittal, A. Zisserman, and P. H. S. Torr, "Hand detection using multiple proposals," in *British Machine Vision Conf.*, 2011, pp. 1–11.

[7] E. Ohn-Bar and M. M. Trivedi, "Hand gesture recognition in real time for automotive interfaces: A multimodal vision-based approach and evaluations," *IEEE Transactions on ITS*, vol. 15, no. 6, pp. 2368–2377, 2014.

[8] E. Ohn-Bar, S. Martin, A. Tawari, and M. M. Trivedi, "Head, eye, and hand patterns for driver activity recognition," in *ICPR*, 2014, pp. 660–665.

[9] X. Sun, Y. Wei, S. Liang, X. Tang, and J. Sun, "Cascaded hand pose regression," in *CVPR*, 2015, pp. 824–832.
[10] C. Qian, X. Sun, Y. Wei, X. Tang, and J. Sun, "Realtime and robust hand tracking from depth," in *CVPR*, 2015, pp. 1106–1113.
[11] S. Sridhar, F. Mueller, A. Oulasvirta, and C. Theobalt, "Fast and robust hand tracking using detection-guided optimization," in *CVPR*, 2015, pp. 3213–3221.
[12] R. Girshick, J. Donahue, and J. M. T. Darrell, "Region-based convolutional networks for accurate object detection and semantic segmentation," *IEEE Transactions on PAMI*, Accepted may 2015.
[13] P. Arbeláez, J. Pont-Tuset, J. Barron, F. Marques, and J. Malik, "Multiscale combinatorial grouping," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 328–335.
[14] W. Liu, A. Rabinovich, and A. C. Berg, "Parsenet: Looking wider to see better," *arXiv preprint arXiv:1506.04579*, 2015.