Embedded Recurrent Network for Head Pose Estimation in Car

Guido Borghi, Riccardo Gasparini, Roberto Vezzani, Rita Cucchiara
DIEF - University of Modena and Reggio Emilia, Italy
Email: {guido.borghi,riccardo.gasparini,roberto.vezzani,rita.cucchiara}@unimore.it

***Abstract*— An accurate and fast driver's head pose estimation is a rich source of information, in particular in the automotive context. Head pose is a key element for driver's behavior investigation, pose analysis, attention monitoring and also a useful component to improve the efficacy of Human-Car Interaction systems. In this paper, a Recurrent Neural Network is exploited to tackle the problem of driver head pose estimation, directly and only working on depth images to be more reliable in presence of varying or insufficient illumination. Experimental results, obtained from two public dataset, namely *Biwi Kinect Head Pose* and *ICT-3DHP Database*, prove the efficacy of the proposed method that overcomes state-of-art works. Besides, the entire system is implemented and tested on two embedded boards with real time performance.**

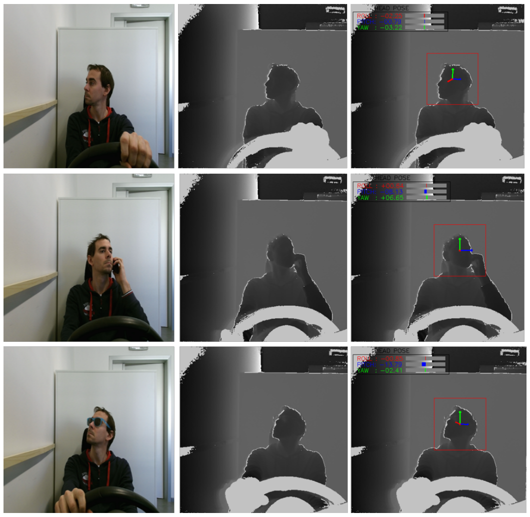


Fig. 1. Graphical results of the proposed system. Starting from depth images (second column), a Recurrent Neural Network is exploited to recover the head pose (third column), expressed with the 3D yaw, pitch and roll.

I. INTRODUCTION

Head pose estimation is a useful source of information in many computer vision fields, such as Human-Computer Interaction [1], Human-Robot Interaction [2], saliency analysis [3] and in particular the automotive context. Indeed, head pose is a concrete tool to investigate many aspects of the driver.

First, a driver monitoring and attention analysis can be conducted through head pose, and driver inattention is one of the most important factor in highway crashes. The *National Highway Traffic Safety Administration* reports that about 25% of police-reported crashes involves driver inattention [4]. Driver inattention can be classified into two main categories [4]: *distraction* and *fatigue*. In turn, distraction is divided in three areas [5]: *Visual Distraction* (*e.g.* driver's gaze is not on the road), *Cognitive Distraction* (*e.g.* driver is not focused into driving activity) and *Manual Distraction*, that can be classified as *Auditory Distraction* (*e.g.* driver is talking at the phone) and *Bio-mechanical Distraction* (*e.g.* driver's hands are not on the steering wheel). Head pose analysis is a key aspect to investigate this kind of driver distractions and fatigue: for example, in [6] driver's head movements and cognitive distraction are correlated during driving activity. Besides, in [7] head pose and eye location information are combined to estimate the driver gaze estimation. Other literature works [8], [9], [10] reveal that driver inspection patterns on the forward view are influenced by cognitive and visual distraction. Furthermore, many car companies have implemented and installed own monitoring systems for driver posture, behavior and attention analysis, proving the increase of interest in this research field.

Second, head pose estimation could be used to detect driver fatigue, that "concerns the inability or disinclination to continue an activity, generally because the activity has been going on for too long" [11]. Four categories of driver fatigue can be considered [4]: *Local Physical Fatigue* (*e.g.* in a skeletal or ocular muscle), *General Physical fatigue* (the consequence of a heavy manual labor), *Central Nervous Fatigue* (drowsiness) and *Mental Fatigue* (low concentration due to bad physical conditions). In [12] head nod after checking the side mirrors, less frequent head motions, the tendency to turn the head to the left are strongly correlated with driver fatigue.

Finally, driver head pose investigation can be exploited for infotainment systems, to make more *user-friendly* the new Human-Car Interaction systems [13], [14] and to increment the velocity and safety of driver operations inside the cockpit. Automotive context is a challenging environment and imposes some requirements on installed systems inside a car cockpit. A fundamental aspect is the *light invariance*: systems have to be reliable even in presence of dramatic light changes (day, night, tunnels, bad weather conditions). Car companies tackle this problem with the adoption of near infrared devices, such as depth cameras, that are less prone to fail in these conditions in respect to standard RGB cameras. Also the *non-invasivity* of the systems is a key aspect: driver's movements and gaze must be not impeded during driving activity and hardware devices have to be integrated inside the limited space of the car cockpit. Several solutions have been proposed [15] and are based

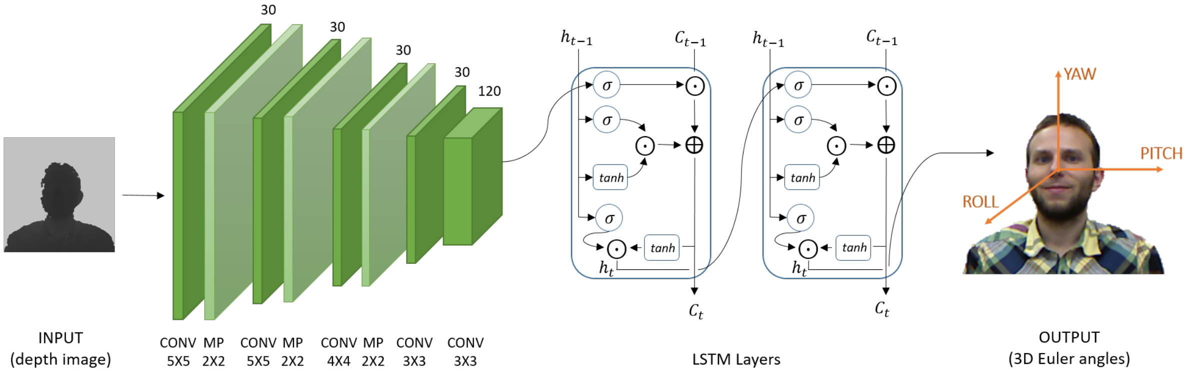


Fig. 2. Overall view of the proposed system. Depth images are the input of the Recurrent Neural Network, composed by 5 convolutional layers and 3 max pooling layers, followed by 2 LSTM modules. The output are continuous values representing the three 3D Euler head angles.

on *Physiological Signals* (electroencephalography, electrocardiography and electromyography) collected with sensors placed usually inside the steering wheel or the car seat, *Vehicle Signals* (parameters acquired from the car bus) and *Physical Signals* based on image acquisition and elaboration: in this paradigm vision-based approaches are probably the best solution. Besides, the presence of several occlusion and the dynamism of the driver body inside the cockpit require robust detection and classification algorithms. Finally, *real time performance* are necessary to detect anomalies and immediately product warning alarms.

In this work, we aim to tackle the head pose estimation problem through a deep approach. In particular, we exploit a Recurrent Neural Network (RNN) in a regression manner, to output continuous 3D angle values (*yaw*, *pitch* and *roll*) with good accuracy and real time performance. The proposed system is based only depth images, obtained through a infrared sensor, to achieve reliability against light changes. Finally, we investigate the possibility to implement the proposed system on two mobile embedded boards (*NVIDIA Jetson TK1* and *NVIDIA Jetson TX1*), with limited computational power and low energy consumption, following a *low-cost* and *plug-in* approach compatible with the automotive context.

II. RELATED WORK

In last decades, several works have tackled the problem of head pose estimation, or rather, the ability to infer the orientation of a person's head relative to the view of the camera [16]. Literature approaches are based on intensity images (2D data), depth images, cloud points (3D data), or a combination of them. Typically, 3D data provide less sensitivity to light changes and partial occlusions, but suffer in term of texture information.

Few works in the literature combine the use of Convolutional Neural Networks (CNN) and depth maps to estimate the head pose directly from images: in [17], [18] for the first time is proposed to use a CNN with depth images acquired with *Microsoft Kinect* camera. Other works that are based on CNN and 3D data concern different tasks, like skeleton body pose estimation [19], action recognition [20], object pose estimation [21] and human body joint identification [22], [23].

A novel triangular surface patch descriptor is encoded in [24] to map 3D head data with the relative head pose learned from a previous training phase. Fanelli et al. [25] proposed a real time framework implemented with Random Regression Forests to detect head position and orientation, directly from depth images. Other approaches tackle the head pose estimation task as a optimization problem: in [26], [27] Particle Swarm Optimization and least-square minimization algorithms are respectively applied on depth images. In [28] is proposed an approach for head pose detection with low quality depth data. Extremely low resolution RGB images are the used in [29] and despite the input quality results close to the state-of-the-art are achieved. [30] tackles the problem of large head pose variations and partial occlusions: after the nose tip detection, geometric features are exploited to extract head pose in real time thanks to the aid of a dedicated graphic unit. The angles of head rotation are predicted exploiting Histograms of Gradients (HOG) feature, extracted from RGB images, in [31]. A CNN and RGB images acquired from a monocular camera are combined in [32]; the deep architecture is exploited in a regression manner with a good accuracy, despite the presence of light changes in input images. Deep learning approaches typically require huge amount of annotated data, the use of synthetic RGB dataset is growing [33].

Several works in literature use a combination of 2D and 3D data: for example, [2] combine stereo camera images and color images through a neural network approach; a initialization to detect the skin color is required at the beginning. Also in [34] are merged depth and intensity data to support a 3D constrained local method for facial feature tracking task. In [35] time-of-flight depth data and color information are combined to perform a real time head pose estimation. The computation work is demanded to a dedicated GPU. A Multi Layer Perceptron and a linear SVM are exploited to combined HOG feature computed on 2D and 3D data, respectively in [36] and in [37]. A 3D morphable model with pose parameters from both RGB and depth data is fitted in [38], exploiting a well-known face detector [39].

TABLE I

RESULTS ON *Biwi Dataset*: PITCH, ROLL AND YAW (EULER ANGLES)

Method	**Data**	**Pitch**	**Roll**	**Yaw**
Saeed et al. [37]	RGB+depth	5.0 ± 5.8	4.3 ± 4.6	3.9 ± 4.2
Fanelli et al.[25]	depth	8.5 ± 9.9	7.9 ± 8.3	8.9 ± 13.0
Yang et al. [36]	RGB+depth	9.1 ± 7.4	7.4 ± 4.9	8.9 ± 8.2
Baltruvsaitis et al.[34]	RGB+depth	5.1	11.2	6.29
Papazov et al. [24]	depth	3.0 ± 9.6	2.5 ± 7.4	3.8 ± 16.0
Venturelli et al. [17]	depth	2.8 ± 3.1	2.3 ± 2.9	3.6 ± 4.1
Our	depth	**2.0 ± 1.9**	**2.1 ± 2.0**	**2.3 ± 2.0**

III. MODEL ARCHITECTURE

The proposed system receives a stream of depth images as input and outputs 3D angles, defining the head orientation. The key elements of the proposal are the following. Firstly, we propose to exploit a Recurrent Neural Network (RNN) to perform head pose estimation, directly and only from depth images, acquired with a frontal and stationary infrared device. No additional features, like nose tip, eyes or mouth localization are required. Second, the pose estimation task has been tackled in a regression manner to produce continuous 3D angles (yaw, pitch and roll) as output. This task is challenging due to the problem of periodicity [40] and the non-continuous nature of Euler angles [41]. Finally, a particular attention is given to the computational requirement of the entire system, which is implemented on embedded boards equipped with a GPU module.

Head detection and localization are out of the scope of this paper and thus we supposed they are available.

A. LSTM module

A *Long Short-Term Memories* (LSTM) network [42] is a type of recurrent network that has achieved good performances on many tasks, like image captioning [43], visual recognition [44] and so on. To the best of our knowledge, very few works in literature use a recurrent approach in a regression manner to perform head pose estimation.

Our LSTM model is updated according to the following equations, that are graphically represented in Figure 2:

$$I_t = \sigma(W_i x_t + U_i H_{t-1} + b_i) \tag{1}$$

$$F_t = \sigma(W_f x_t + U_f H_{t-1} + b_f) \tag{2}$$

$$O_t = \sigma(W_o x_t + U_o H_{t-1} + b_o) \tag{3}$$

$$G_t = tanh(W_c x_t + U_c H_{t-1} + b_c) \tag{4}$$

$$C_t = F_t \odot C_{t-1} + I_t \odot G_t \tag{5}$$

$$H_t = O_t \odot tanh(C_t) \tag{6}$$

where x_t is the input to the memory cell layer at time t, all W and U are weight matrices and b are bias vectors.

Input images of the recurrent network are obtained by a infrared acquisition device. Input values are normalized to set their mean and the variance to 0 and 1, respectively. Only faces with a small part of background are fed into the network: subject's face is cropped using a dynamic window that changes based on the distance of the subject from the camera. Given the head center x_h, y_h a bounding box for each frame is crated with width and height computed as:

$$w, h = \frac{f_{x,y} \cdot R}{Z} \tag{7}$$

where $f_{x,y}$ are the horizontal and vertical focal lengths expressed in pixel of the acquisition device, R is the width of a generic face, 300 mm in our case, and Z is the distance between the subject's face and the device, obtained from the depth images. In this way, it is possible to obtained images with a centered head, with small portion of pixels that belong to the background.

A shallow architecture, as depicted in the left part of Figure 2, is adopted in order to conjugate accuracy in pose estimation task and low computational demand. Three convolutional layers with 30 filters of 5×5 and 4×4 are followed by a max pooling layer of dimensions 2×2. Then, other two convolutional layers are added, with respectively 30 and 120 filters of 3×3. Finally, there are two LSTM layers with a size of 120 and 84. The output is generated by a fully connected layer with 3 neurons, that corresponds to the three 3D angles predicted: yaw, pitch and roll. The activation function is the hyperbolic tangent (*tanh*), which enables the network to generate finite outputs $[-\infty, +\infty] \rightarrow [-1, +1]$. *Adadelta* optimizer [45] is exploited to solve the back-propagation. During the training phase, the L_2 loss is used:

$$L_2 = \sum_{i}^{n} \|y_i - f(x_i)\|_2 \tag{8}$$

where y_i is the ground truth information, expressed in Euler angles and $f(x_i)$ is the network prediction, both normalized between $[-1, +1]$. The RNN has been trained with a batch size of 64, a momentum value of 9^{-1}, a decay value of 5^{-4} and a learning rate initially set to 10^{-1} and its value is automatically updated by Adadelta optimizer. Due to the presence of the recurrent layers, input images have to be organized into temporal sequences. Sequences have a length of 60 frames and are overlapped to each other with 30 frames, each frame has a spatial resolution of 64×64 pixel and one channel, as a gray level image.

Finally, data augmentation is performed: given a depth image, new input images are obtained with a horizontal and vertical translation, a zoom-in and zoom-out operations and the addition of a Gaussian noise. This operation guarantees the increase of the amount of input images, helping to avoid

TABLE II

RESULTS ON *ICT-3DHP Dataset*: PITCH, ROLL AND YAW (EULER ANGLES)

Method	**Data**	**Pitch**	**Roll**	**Yaw**
Saeed et al. [37]	RGB+depth	4.9 ± 5.3	4.4 ± 4.6	**5.1** ± **5.4**
Fanelli et al. [25]	depth	5.9 ± 6.3	-	6.3 ± 6.9
Baltruvsaitis et al.[34]	RGB+depth	7.06	10.48	6.90
Our	depth	**4.9** ± **4.6**	**4.2** ± **4.3**	7.5 ± 6.3

over fitting problems. Moreover, data augmentation creates image with partial occlusions, a key element to have a trained network that could be reliable against head occlusions.

IV. EXPERIMENTAL RESULTS

In this section experimental results are presented. One public dataset, namely *Biwi Kinect Head Pose Dataset*, is used for the training phase. An additional dataset (*ICT-3D database*) is included in the testing phase, conducing a cross-dataset evaluation. The evaluation metric is based on the *Mean Average Error*, the absolute difference between ground truth annotation and network predictions, reported in Euler angles.

A. Dataset

Biwi Dataset has been introduced by Fanelli *et al.* in [46], is one of the few dataset that contains both RGB and depth data, acquired by *Microsoft Kinect* device, and explicitly designed for head pose estimation task. It is composed of about 15k upper body images with a spatial resolution of 640x480, collected with 20 subjects (14 males and 6 females), the head rotation spans about ±50° for roll, ±75° for yaw and ±60° for pitch. The calibration matrix and the head center are given. We use sequences of subjects number 11 and 12 to test our network, even if is not clear the choice of test subjects in the original work.

ICT-3DHP Database [34] is also collected using *Microsoft Kinect* and contains about 14k both RGB and depth frames (640x480 pixel), divided into 10 sequences. All subjects wear a white cap, due to the presence of the *Polhemus Fastrack* used to track the head pose for ground truth annotations. In general, it is easy to note the lack of dataset that contains depth information and oriented to deep approaches. Specifically for our task, to the best of our knowledge, there are no dataset acquired inside vehicle, designed for head pose estimation, that include both intensity and depth images and accurate head pose annotations.

B. Quantitative evaluation

Results of the proposed system are compared with other state-of-art methods. As above mentioned, the testing phase is conducted on two subjects of *Biwi* dataset and the entire *ICT-3DHP* dataset, following the evaluation protocol proposed in [25]. As showed in Table I, our approach overcomes other literature methods with *Biwi* dataset, based on RGB, depth or both data; in particular, our method is more accurate than other recent methods based on a CNN [17], [32] (last paper is not reported in the table because the original evaluation pipeline is not followed; however, results are 3.4 ± 2.9, 2.6 ± 2.5 and 2.8 ± 2.4 for pitch, roll and yaw, respectively). In general, we observe that yaw angle presents an error mean value more accentuated that the other two angles. In Table II are reported results for *ICT-3DHP* dataset. Angle values related to [25] are taken from [19]. Also int his case a good accuracy is achieved, despite the lack of the head center annotation (tip nose is instead provided) that influences the crop of the dynamic window; moreover, no guarantees are provided about the accuracy of the ground truth of training dataset, *Biwi*, and testing dataset: the same angle could be expressed with partially different continuous values.

The good accuracy achieved in both cases allows to implement a driver attention monitoring system or into new infotainment system, all cases in which a precise head pose estimation is required.

C. Reliability against occlusions

In real situations, driver images are usually affected by occlusions, caused by hand movements or objects like smartphones, garments and similar. During the training session, data augmentation produces input depth images with artificial occlusions (as result for example for the horizontal or vertical translation). To test the reliability against occlusions of the proposed system, we have artificially applied masks to the input images of the *Biwi* dataset to partially cover the subject's face. Four types of occlusions are created with a rectangular mask with a size of about 25% of original input image area, that is placed on the top, bottom, left and right part of the image; then, a square of size $60x60$ pixel mask is applied on the tip nose of the subject, covering the central part of the face. Finally, for each test subject five sequences are created, one for each type of occlusion and a final sequence with all the occlusion randomly selected. Table III shows the results of network prediction in presence of these occlusions. As expected, occlusions generally degrade network prediction accuracy. In particular, we note that top occlusions affect in particular pitch prediction, while the center occlusion affects all the angles.

V. EMBEDDED IMPLEMENTATION

The proposed system has been developed and tested using an embedded system. The reasons underlying this implementation are varied: the recent release of cheap GPU board bring up new opportunities to run deep networks in mobile context; then, a plug-in approach is more desirable, in order to have a ready system without a real implementation inside the car

TABLE III

RESULTS OF THE PROPOSED SYSTEM WITH SIMULATED OCCLUSIONS (*Biwi* DATASET)

Type of occlusion	**Head**		
	Pitch	Roll	Yaw
center	11.9 ± 4.8	3.0 ± 3.4	11.0 ±8.2
bottom	8.6 ± 3.8	3.7 ± 2.9	5.0 ±4.2
right	2.5 ± 2.0	2.9 ± 2.8	8.7 ±6.0
top	34.0 ± 18.8	8.4 ± 8.6	8.9 ±6.2
left	2.8 ± 3.1	4.5 ± 3.0	5.5 ±6.1
random	7.8 ± 7.4	3.1 ± 3.5	5.7 ±4.6

cockpit during projecting phase. Besides, low-cost solution is a key element for car companies.

A. Hardware

The use of CNN to perform head pose estimation has driven the choice to two embedded boards equipped with a dedicated graphics processing unit (GPU): a *NVIDIA Jestson TK1* [1] and *TX1* [2]. The *TK1* board is based on a 192 CUDA-cores *Kepler* architecture, has a quad-Core ARM A15 Cortex CPU, 2 GB of RAM memory shared with graphic unit and 16 GB of flash storage. Instead, *TX1* is based on a 256 CUDA-cores *Maxwell* architecture, has a quad-core ARM A57 CPU, 4 GB of shared memory and 16 GB of flash storage. Depth frame are acquired with *Microsoft Kinect One*, based on Time-of-Flight technology, that is able to acquire depth frame with a spatial resolution of 512x424 pixel.

B. Speed evaluation

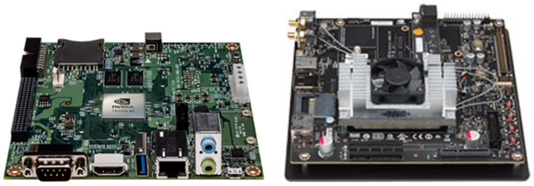
Two logical parts have to be implemented on embedded boards: the acquisition process, to acquire depth frame, and the prediction process, demanded to the RNN described above. Both parts have been developed in *Python*, exploiting the *Keras* framework with *Theano* back-end.
No official libraries have been released to run *Microsoft Kinect SDK* with the Linux platform running on the Jetson boards. Thus, we exploited the C++ library called *Libfreenect* [3], trough which we capture the frames and send to the Python modules by means of a socket communication. Speed evaluation is conducted measuring time spent by the acquisition process and the prediction task. Table V-B reports speed performance, expressed in seconds, for both embedded boards.

TABLE IV

SPEED PERFORMANCE ON JETSON TK1 AND TX1 BOARDS

Boards	**Jetson TK1**		**Jetson TX1**	
	CPU	GPU	CPU	GPU
Acquisition Time	0.0305	0.0277	0.0141	0.0238
Prediction Time	0.0717	0.0256	0.0525	0.0091
Total Time	0.1022	**0.0533**	0.0666	**0.0329**

Results prove the feasibility of the implementation on cheap and embedded boards, with limited hardware resources and low energy consumption. Thanks to the shallow architecture of the RNN and the limited size of the input images ($64x64$ pixels), both CPU and GPU elaborations allow real time or near real time speed performance. On *TK1* the system is able to achieve about 10 fps with CPU and 19 fps with GPU; on *TX1* we have about 15 fps with CPU and 30 fps with the aid of the GPU and cuDNN libraries (these libraries are not supported on *TK1*). As expected, acquisition time is similar in both cases. The speed up is about 2 for both *TK1* and *TX1* boards, with respect to CPU elaboration speed. *TX1* has more CUDA cores than *TK1* and a more powerful processor, that is reflected in our experiments.
With these embedded solutions, a complete *stand-alone* and *plug-in* driver monitor system could be integrated directly inside the car cockpit, without a specific design of the car companies.



(a) (b)

Fig. 3. NVDIA Jetson TK1 and NVIDIA Jetson TX1

VI. CONCLUSIONS

We have presented a RNN to tackle the head pose estimation task, with a good accuracy and real time performance. We focus on the automotive context to set system requirements: light changes in-variance, low computational load, reliability to occlusions and no initialization. The system has been also implemented on two cheap embedded boards to prove the feasibility of the entire system in terms of memory requirements and computational load. A module for head detection or localization is strictly required to create a complete framework that acquires depth images, finds the head center and outputs the three head angles. Finally, a event system could be integrated to produce warnings correlated with the driver head orientation.

ACKNOWLEGMENTS

This work has been carried out within the projects “Citta educante” (CTN01-00034-393801) of the National Technological Cluster on Smart Communities funded by MIUR and “FAR2015 - Monitoring the car drivers attention with multisensory systems, computer vision and machine learning” funded by the University of Modena and Reggio Emilia. We also acknowledge the CINECA award under the ISCRA initiative, for the availability of high performance computing resources and support.

REFERENCES

[1] U. Weidenbacher, G. Layher, P. Bayerl, and H. Neumann, “Detection of head pose and gaze direction for human-computer interaction,” in *International Tutorial and Research Workshop on Perception and Interactive Technologies for Speech-Based Systems*. Springer, 2006, pp. 9–19.

[1] http://www.nvidia.it/object/jetson-tk1-embedded-dev-kit-it.html

[2] http://www.nvidia.com/object/embedded-systems.html

[3] https://github.com/OpenKinect/libfreenect

[2] E. Seemann, K. Nickel, and R. Stiefelhagen, “Head pose estimation using stereo vision for human-robot interaction.” in *FGR*. IEEE Computer Society, 2004, pp. 626–631.

[3] Z. Yücel and A. A. Salah, “Head pose and neural network based gaze direction estimation for joint attention modeling in embodied agents,” in *Proc. Annual Meeting of Cognitive Science Society*, 2009.

[4] Y. Dong, Z. Hu, K. Uchimura, and N. Murayama, “Driver inattention monitoring system for intelligent vehicles: A review,” *IEEE transactions on intelligent transportation systems*, vol. 12, no. 2, pp. 596–614, 2011.

[5] T. A. Ranney, E. Mazzae, R. Garrott, and M. J. Goodman, “Nhtsa driver distraction research: Past, present, and future,” in *Driver distraction internet forum*, 2000.

[6] M. Miyaji, H. Kawanaka, and K. Oguri, “Driver’s cognitive distraction detection using physiological features by the adaboost,” in *12th International IEEE Conference on Intelligent Transportation Systems*. IEEE, 2009, pp. 1–6.

[7] R. Valenti, N. Sebe, and T. Gevers, “Combining head pose and eye location information for gaze estimation,” *IEEE Transactions on Image Processing*, vol. 21, no. 2, pp. 802–815, 2012.

[8] L. S. Angell, J. Auflick, P. Austria, D. S. Kochhar, L. Tijerina, W. Biever, T. Diptiman, J. Hogsett, and S. Kiger, “Driver workload metrics task 2 final report,” Tech. Rep., 2006.

[9] J. L. Harbluk, Y. I. Noy, P. L. Trbovich, and M. Eizenman, “An on-road assessment of cognitive distraction: Impacts on drivers visual behavior and braking performance,” *Accident Analysis & Prevention*, vol. 39, no. 2, pp. 372–379, 2007.

[10] M. A. Recarte and L. M. Nunes, “Mental workload while driving: effects on visual search, discrimination, and decision making.” *Journal of experimental psychology: Applied*, vol. 9, no. 2, p. 119, 2003.

[11] H. D. Croo and M. Bandmann, “The role of driver fatigue in commercial road transport crashes,” in *European Transportation Safety Council*, 2001.

[12] A. Eskandarian, R. Sayed, P. Delaigue, A. Mortazavi, and J. Blum, “Advanced driver fatigue research,” Tech. Rep., 2007.

[13] T. Nawaz, M. S. Mian, and H. A. Habib, “Infotainment devices control by eye gaze and gesture recognition fusion,” *IEEE Transactions on Consumer Electronics*, vol. 54, no. 2, pp. 277–282, 2008.

[14] C. Tran and M. M. Trivedi, “Towards a vision-based system exploring 3d driver posture dynamics for driver assistance: Issues and possibilities,” in *Intelligent Vehicles Symposium*, 2010, pp. 179–184.

[15] N. Alioua, A. Amine, A. Rogozan, A. Bensrhair, and M. Rziza, “Driver head pose estimation using efficient descriptor fusion,” *EURASIP Journal on Image and Video Processing*, no. 1, pp. 1–14, 2016.

[16] E. Murphy-Chutorian and M. M. Trivedi, “Head pose estimation in computer vision: A survey,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 4, pp. 607–626, Apr. 2009.

[17] M. Venturelli, G. Borghi, R. Vezzani, and R. Cucchiara, “”deep head pose estimation from depth data for in-car automotive applications,” *Proceedings of the 2nd International Workshop on Understanding Human Activities through 3D Sensors*, 2016.

[18] ——, “From depth data to head pose estimation: a siamese approach,” in *Proceedings of the 12th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications*, 2016.

[19] B. Crabbe, A. Paiement, S. Hannuna, and M. Mirmehdi, “Skeleton-free body pose estimation from depth images for movement analysis,” in *Proc. of the IEEE International Conference on Computer Vision Workshops*, 2015, pp. 70–78.

[20] S. Ji, W. Xu, M. Yang, and K. Yu, “3d convolutional neural networks for human action recognition,” *IEEE Transactions on pattern analysis and machine intelligence*, vol. 35, no. 1, pp. 221–231, 2013.

[21] A. Doumanoglou, V. Balntas, R. Kouskouridas, and T. Kim, “Siamese regression networks with efficient mid-level feature extraction for 3d object pose estimation,” *CoRR*, 2016. [Online]. Available: http://arxiv.org/abs/1607.02257

[22] E. Hoffer and N. Ailon, “Deep metric learning using triplet network,” in *Proc. of Int’l Workshop on Similarity-Based Pattern Recognition*. Springer, 2015, pp. 84–92.

[23] Y. Sun, Y. Chen, X. Wang, and X. Tang, “Deep learning face representation by joint identification-verification,” in *Advances in Neural Information Processing Systems*, 2014, pp. 1988–1996.

[24] C. Papazov, T. K. Marks, and M. Jones, “Real-time 3d head pose and facial landmark estimation from depth images using triangular surface patch features,” in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 4722–4730.

[25] G. Fanelli, J. Gall, and L. Van Gool, “Real time head pose estimation with random regression forests,” in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE, 2011, pp. 617–624.

[26] P. Padeleris, X. Zabulis, and A. A. Argyros, “Head pose estimation on depth data based on particle swarm optimization,” in *2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*. IEEE, 2012, pp. 42–49.

[27] F. A. Kondori, S. Yousefi, H. Li, S. Sonning, and S. Sonning, “3d head pose estimation using the kinect,” in *Wireless Communications and Signal Processing (WCSP), 2011 International Conference on*. IEEE, 2011, pp. 1–4.

[28] S. Malassiotis and M. G. Strintzis, “Robust real-time 3d head pose estimation from range data,” *Pattern Recognition*, vol. 38, no. 8, pp. 1153–1165, 2005.

[29] J. Chen, J. Wu, K. Richter, J. Konrad, and P. Ishwar, “Estimating head pose orientation using extremely low resolution images,” in *2016 IEEE Southwest Symposium on Image Analysis and Interpretation (SSIAI)*, 2016, pp. 65–68.

[30] M. D. Breitenstein, D. Kuettel, T. Weise, L. Van Gool, and H. Pfister, “Real-time face pose estimation from single range images,” in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE, 2008, pp. 1–8.

[31] V. Drouard, S. Ba, G. Evangelidis, A. Deleforge, and R. Horaud, “Head pose estimation via probabilistic high-dimensional regression,” in *Proc. of IEEE International Conference on Image Processing (ICIP)*, 2015, pp. 4624–4628.

[32] B. Ahn, J. Park, and I. S. Kweon, “Real-time head orientation from a monocular camera using deep neural network,” in *Asian Conference on Computer Vision*. Springer, 2014, pp. 82–96.

[33] X. Liu, W. Liang, Y. Wang, S. Li, and M. Pei, “3d head pose estimation with convolutional neural network trained on synthetic images.”

[34] T. Baltrušaitis, P. Robinson, and L.-P. Morency, “3d constrained local model for rigid and non-rigid facial tracking,” in *Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2012, pp. 2610–2617.

[35] A. Bleiweiss and M. Werman, “Robust head pose estimation by fusing time-of-flight depth and color,” in *IEEE Int’l. Workshop on Multimedia Signal Processing (MMSP)*. IEEE, 2010, pp. 116–121.

[36] J. Yang, W. Liang, and Y. Jia, “Face pose estimation with combined 2d and 3d hog features,” in *21st International Conference on Pattern Recognition (ICPR)*. IEEE, 2012, pp. 2492–2495.

[37] A. Saeed and A. Al-Hamadi, “Boosted human head pose estimation using kinect camera,” in *IEEE International Conference on Image Processing (ICIP)*. IEEE, 2015, pp. 1752–1756.

[38] R. S. Ghiass, O. Arandjelović, and D. Laurendeau, “Highly accurate and fully automatic head pose estimation from a low quality consumer-level rgb-d sensor,” in *Proc. of the 2nd Workshop on Computational Models of Social Interactions: Human-Computer-Media Communication*. ACM, 2015, pp. 25–34.

[39] P. Viola and M. J. Jones, “Robust real-time face detection,” *International journal of computer vision*, vol. 57, no. 2, pp. 137–154, 2004.

[40] K. M. Yi, Y. Verdie, P. Fua, and V. Lepetit, “Learning to assign orientations to feature points,” *arXiv preprint arXiv:1511.04273*, 2015.

[41] A. Kendall, M. Grimes, and R. Cipolla, “Posenet: A convolutional network for real-time 6-dof camera relocalization,” in *Proc. of the IEEE Int’l Conf. on Computer Vision*, 2015, pp. 2938–2946.

[42] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[43] A. Karpathy and L. Fei-Fei, “Deep visual-semantic alignments for generating image descriptions,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3128–3137.

[44] J. Donahue, L. Anne Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell, “Long-term recurrent convolutional networks for visual recognition and description,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 2625–2634.

[45] M. D. Zeiler, “Adadelta: an adaptive learning rate method,” *arXiv preprint arXiv:1212.5701*, 2012.

[46] G. Fanelli, M. Dantone, J. Gall, A. Fossati, and L. Van Gool, “Random forests for real time 3d face analysis,” *International Journal of Computer Vision*, vol. 101, no. 3, pp. 437–458, 2013.