From Depth Data to Head Pose Estimation: a Siamese approach

Marco Venturelli, Guido Borghi, Roberto Vezzani, Rita Cucchiara
University of Modena and Reggio Emilia, DIEF
Via Vivarelli 10, Modena, Italy
{*marco.venturelli, guido.borghi, roberto.vezzani, rita.cucchiara*}*@unimore.it*

Keywords: Head Pose Estimation, Deep Learning, Depth Maps, Automotive

Abstract: The correct estimation of the head pose is a problem of the great importance for many applications. For instance, it is an enabling technology in automotive for driver attention monitoring. In this paper, we tackle the pose estimation problem through a deep learning network working in regression manner. Traditional methods usually rely on visual facial features, such as facial landmarks or nose tip position. In contrast, we exploit a Convolutional Neural Network (CNN) to perform head pose estimation directly from depth data. We exploit a Siamese architecture and we propose a novel loss function to improve the learning of the regression network layer. The system has been tested on two public datasets, *Biwi Kinect Head Pose* and *ICT-3DHP database*. The reported results demonstrate the improvement in accuracy with respect to current state-of-the-art approaches and the real time capabilities of the overall framework.

1 INTRODUCTION

Head pose estimation provides a rich source of information that can be used in several fields of computer vision, like attention and behavior analysis, saliency prediction and so on. In this work, we focus in particular on the automotive field: several works in literature show that head pose estimation is one of the key elements for driver behavior and attention monitoring analysis. Moreover, the recent introduction of semi-autonomous and autonomous driving vehicles and their coexistence with traditional cars is going to increase the already high interest on driver attention studies. In this cases, human drivers have to take driving algorithms under controls, also for legal related issues (Rahman et al., 2015).

Driver's hypo-vigilance is one of the most principal cause of road crashes (Alioua et al., 2016). As reported by the official US government website[1], distracting driving is responsible for 20-30% of road deaths: it is reported that about 18% of injury crashes were caused by distraction, more than 3000 people were killed in 2011 in a crash involving a distracted driver, and distraction is responsible for 11% of fatal crashes of drivers under the age of twenty. Distraction during driving activity is defined by *National Safety Administration* (NHTSA) as "*an activity that could divert a person's attention away from the primary task of driving*". (Craye and Karray, 2015) defines three classes of driving distractions: 1) *manual distraction*: driver's hands are not on the wheel; examples of this kind of activity are incorrect use of infotainment system (radio, GPS navigation device and others) or text messaging. 2) *visual distraction*: driver's eyes are not looking at the road, but, for example, at the smartphone screen or a newspaper. 3) *cognitive distraction*: driver's attention is not focused on driving activity; this could occur due to torpor, stress, and bad physical conditions in general or, for example, if talking with passengers. Smartphone abuse during driving activity leads to all of the three distraction categories mentioned above; in fact, that is one of the most important cause of fatal driving distraction, with about 18% of fatal driver accidents in North America, as reported by NHTSA.

Several works have been proposed for in-car safety and they can be divided by the type of signal used (Alioua et al., 2016).

1) *Physiological signals*: special sensors as electroencephalography (EEG), electrocardiography (ECG) or electromyography (EMG) are places inside the cockpit to acquire signals from driver's body; this kind of solution is very intrusive and a body-sensor contact is strictly required;

2) *Vehicle signals*: vehicle parameters like velocity changes, steering wheel motion, acquired from car bus, can reveal abnormal driver actions;

[1] http://www.distraction.gov/index.html

3) *Physical signals*: image processing techniques are exploited to investigate driver vigilance through facial features, eye state, head pose or mouth state; these methods are non-intrusive, thus image are acquired from inside cockpit cameras.
Taking into account the above exposed elements, some characteristics can be elected as crucial for a reliable and implementable head pose estimation framework, also related to the placement and the choice of the most suitable sensing device:

- **Light invariance**: the framework should be reliable on each weather condition that could dramatically changes the type of illumination inside the car (shining sun and clouds, in addition to sunrises, sunsets, nights etc.). Depth cameras are proven to be less prone to fail in these conditions than classical RGB or stereo sensors;
- **Non invasive**: it is fundamental that acquisition devices do not impede driver's movements during driving activity; in this regard, recently many car industries have placed sensors inside steering wheel or seats to passively monitor driver's physiological conditions;
- **Direct estimation**: the presence of severe occlusions or the high variability of driver's body pose could make facial feature detection extremely challenging and prone to failure; besides, no initialization phase is welcome.
- **Real time performances**: in automotive context an attention monitoring system is useful only if can immediately detect anomalies in driver's behavior;
- **Small size**: acquisition device has to been integrated inside cockpit, often in a particular position (like next rear-view mirror): recently, the release of several low cost, accurate and small sized 3D sensors open new scenarios.

In this work, we aim at exploiting a deep architecture to perform in real time head pose regression, directly from single-frame depth data. In particular, we use a *Siamese* network to improve our training phase by learning more discriminative features, and optimize our regression layer network loss function.

2 RELATED WORK

(Murphy-Chutorian and Trivedi, 2009) shows that head pose estimation is the goal of several works in the literature. Current approaches can be divided depending on the type of data they rely on, RGB images (2D information), depth maps data (3D information), or both. In general, methods for head pose estimation relying solely on RGB images are sensitive to illumination, partial occlusions and lack of features (Fanelli et al., 2011), while depth-based approaches are lacking of texture and color information.
Several works in the literature proposed to use Convolutional Neural Networks with depth data, but especially in skeleton body pose estimation (Crabbe et al., 2015) or action recognition tasks (Ji et al., 2013). These works reveal how techniques like background subtraction, depth maps normalization and data augmentation could influence deep architectures performance. Recently, (Doumanoglou et al., 2016) exploits Siamese Networks to perform object pose estimation, applying a novel loss function that can boost the performance of a regression network layer. Other works, like (Hoffer and Ailon, 2015; Sun et al., 2014), exploit a Siamese approach in deep architecture to improve network learning capabilities and to perform human body joint identification.
Several works rely only on depth data. As Figure 1 shows, the global quality of depth images strictly depends by the technology of the acquisition device. In (Papazov et al., 2015) shapes of 3D surfaces are encoded in a novel triangular surface patch descriptor to map an input depth with the most similar ones that were computed from synthetic head models, during a precedent a training phase. (Kondori et al., 2011) exploits a least-square minimization of the difference between the rate prediction and the measured rate of change of input depth. Usually, depth data are characterized by low quality. Starting from this assumption, in (Malassiotis and Strintzis, 2005) is proposed a method designed to work on low quality depth data to perform head localization and pose estimation; this method relies on an accurate nose localization. In (Fanelli et al., 2011) a real time framework based on Random Regression Forests is proposed, to perform head pose estimation directly from depth images, without exploiting any facial features. In (Padeleris et al., 2012) Particle Swarm Optimization is used to tackle the head pose estimation, treated as an optimization problem. This method requires an initial frame to construct the reference head pose from depth data; limited real time performance are obtained thanks to a GPU.
(Breitenstein et al., 2008) tackles the problem of large head pose variations, partial occlusions and facial expressions from depth images: several methods present in the literature have poor performance with these factors. In the work of Bretenstein et al. the main issue is that the nose must be always visible, due to this method uses geometric features to generate nose candidates which suggest head position hypothesis. The

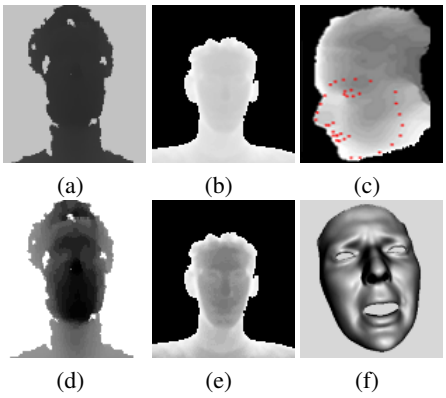

Figure 1: Examples of depth images taken by different acquisition devices. (a) is acquired by *Microsoft Kinect* based on structured-light technology (BIWI dataset (Fanelli et al., 2011)). (b) is obtained thanks to *Microsoft Kinect One*, a time-of-flight 3D scanner; (d)-(e) are the correspondent images, after contrast stretching elaboration to enhance facial clues. Images (c)-(f) come from synthetic dataset (Fanelli et al., 2010; Baltrušaitis et al., 2012).

alignment error computation is demanded to a dedicated GPU in order to work in real time.

(Chen et al., 2016) achieves results very close to state-of-art results, even if in this work the problem of head pose estimation is token on extremely low resolution RGB images. HOG features and a Gaussian locally-linear mapping model are used in (Drouard et al., 2015). These models are learned using training data, to map the face descriptor onto the space of head poses and to predict angles of head rotation.
A Convolutional Neural Network (CNN) is used in (Ahn et al., 2014) to perform head pose estimation from RGB images. This work shows that a CNN properly works even in challenging light conditions. The network inputs are RGB images acquired from a monocular camera: this work is one of the first attempt to use deep learning techniques in head pose estimation problem. This architecture is exploited in a data regression manner to learn the mapping function between visual appearance and three dimensional head estimation angles. Despite the use of deep learning techniques, system working real time with the aid of a GPU. A CNN trained on synthetic RGB images is used also in (Liu et al.,). Recently, the use of synthetic dataset is increasing to support deep learning approaches that basically require huge amount of data. A large part of works relies on both 2D and 3D data. In (Seemann et al., 2004) a neural network is used to combine depth information, acquired by a stereo camera, and skin color histograms derived from RGB images. The user face has to be detected in frontal pose at the beginning of framework pipeline to initialize the color skin histograms. In (Baltrušaitis et al., 2012) a 3D constrained local method for robust facial feature tracking under varying poses is proposed. It is based on the integration both depth and intensity information.
(Bleiweiss and Werman, 2010) used time-of-flight depth data to perform a real time head pose estimation, combined with color information. The computation work is demanded to a dedicated GPU. (Yang et al., 2012) elaborated HOG features both on 2D and 3D data: a Multi Layer Perceptron is then used for feature classification. Also the method presented in (Saeed and Al-Hamadi, 2015) is based on RGB and depth HOG, but a linear SVM is used for classification task. Ghiass et al. (Ghiass et al., 2015) performed pose estimation by fitting a 3D morphable model which included pose parameter, starting both from RGB and depth data. This method relies on face detector of Viola and Jones (Viola and Jones, 2004).

3 HEAD POSE ESTIMATION

The described approach aims at estimating pitch, roll and yaw angles of the head/face with respect to the camera reference frame. A depth image is provided as input and a Siamese CNN is used to build an additional loss function which improves the strength of the training phase. Head detection and localization are supposed to be available. No additional information such as facial landmarks, nose tip position, skin color and so on are taken into account, differently from other methods like (Seemann et al., 2004; Malassiotis and Strintzis, 2005; Breitenstein et al., 2008). The network prediction is given in terms of Euler angles, even if the task is challenging due to problems such periodicity (Yi et al., 2015) and the non-continuous nature of Euler angles (Kendall et al., 2015).

3.1 Head acquisition

First of all, face images are cropped using a dynamic window. Given the center x_c, y_c of the face, each image is cropped at a rectangular box centered in x_c, y_c, with width and height computed as:

$$w,h = \frac{f_{x,y} \cdot R}{Z}, \tag{1}$$

where $f_{x,y}$ are the horizontal and vertical focal lengths (in pixels) of the acquisition device, R is the width of a generic face (300 mm in our experiments) and Z is the distance between the acquisition device and the user obtained from the depth image. The output is an image which contains a partially centered face and some part of background. Then, the cropped images

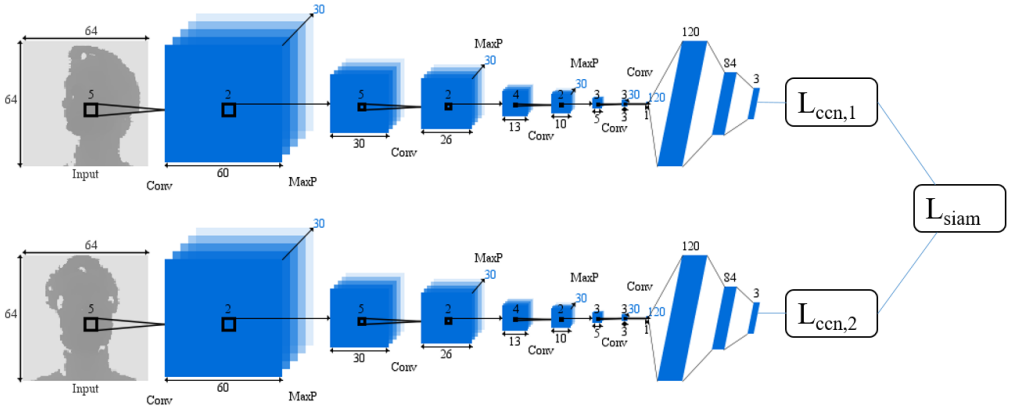

Figure 2: The Siamese architecture proposed for training phase.

are resized to 64x64 pixels. Input image values are normalized to set their mean and the variance to 0 and 1, respectively. This normalization is also required by the specific activation function of the network layers.

3.2 Training phase

The proposed architecture is depicted in Figure 2. A Siamese architecture consists in two or more separate networks, that could be identical — as in our case— and are simultaneously trained. It is important to note that this Siamese architecture is used only during training phase, while a single network is used during the testing. Inspired by (Ahn et al., 2014), each single neural network has a shallow deep architecture in order to obtain real time performance and good accuracy. Each network takes images of 64x64 pixels as input and it is composed of 5 convolutional layers. The first four layers have 30 filters each, whereas the last one has 120 filters. Max-pooling is conducted only three times, due to the relative small size of input images. At the end of the network there are three fully connected layers, with 120, 84 and 3 neurons, respectively. The last 3 neurons correspond to the three angles (*yaw*, *pitch* and *roll*) of the head. The last fully connected layer works in regression. The size of the convolution filters are 5x5, 4x4, 3x3, depending on the layer. The activation function is the hyperbolic tangent (*tanh*): in this way, the network can map output $[-\infty,+\infty] \rightarrow [-1,+1]$, even if ReLU tends to train faster that other activation functions (Krizhevsky et al., 2012). The network is able to output continuous instead of discrete values. We adopt the Stochastic Gradient Descent (SGD) as in (Krizhevsky et al., 2012) to solve the back-propagation.
Each single neural network has a L2 loss:

$$L_{cnn} = \sum_{i}^{n} \|y_i - f(x_i)\|_2^2, \qquad (2)$$

where y_i is the ground truth information (expressed in roll, pitch and yaw Euler angles) and $f(x_i)$ is the network prediction.
Siamese network takes in input pair of images: considering a dataset with about N frames, a huge number $\binom{N}{2}$ of possible pairs can be used. Only pairs with at least 30 degrees of difference between all head angles are selected.
Exploiting Siamese architecture, an additional loss function based on both network outputs can be defined. This loss combines each of the two regression losses and it is the L2 distance between the prediction difference and the ground truth difference:

$$\begin{array}{rcl} L_{siam} & = & \sum_i^n \|d_{cnn}(x_i) - d_{gt}(x_i)\|_2^2 \\ d_{cnn}(x_i) & = & f_1(x) - f_2(x) \\ d_{gt}(x_i) & = & y_1 - y_2 \end{array} , \qquad (3)$$

where $d_{cnn}(x_i)_k$ is the difference between the outputs $f_i(x)$of the two single networks and $d_{gt}(x_i)$ the difference between the ground truth values of the pair.
The final loss is a combination of the losses function of the 2 single networks $L_{cnn,1}, L_{cnn,2}$ and the loss of the Siamese match L_{siam}:

$$L = L_{cnn,1} + L_{cnn,2} + L_{siam} \qquad (4)$$

Each single network has been trained with a batch size of 64, a decay value of 5^{-4}, a momentum value of 9^{-1} and a learning rate set to 10^{-1}, decreased up to 10^{-3} in the final epochs (Krizhevsky et al., 2012). Ground truth angles are normalized to $[-1,+1]$.
We performed data augmentation to increment the size of training input images and to avoid over fitting. Additional patches are randomly cropped from each corner of the input images and from the head center; besides, patches are also extracted by cropping input images starting from the bottom, upper, left and right and adding Gaussian noise. Other additional input samples are created thanks to the pair input system:

Table 1: Results on *Biwi Dataset*: pitch, roll and yaw are reported in Euler angles.

Method	Data	Pitch	Roll	Yaw
(Saeed and Al-Hamadi, 2015)	RGB+RGB-D	5.0 ± 5.8	4.3 ± 4.6	3.9 ± 4.2
(Fanelli et al., 2011)	RGB-D	8.5 ± 9.9	7.9 ± 8.3	8.9 ± 13.0
(Yang et al., 2012)	RGB+RGB-D	9.1 ± 7.4	7.4 ± 4.9	8.9 ± 8.2
(Baltrušaitis et al., 2012)	RGB+RGB-D	5.1	11.2	6.29
(Papazov et al., 2015)	RGB-D	3.0 ± 9.6	2.5 ± 7.4	3.8 ± 16.0
Our	RGB-D	2.8 ± 3.2	2.3 ± 2.9	3.6 ± 4.1
Our+Siamese	RGB-D	$\mathbf{2.3 \pm 2.7}$	$\mathbf{2.1 \pm 2.2}$	$\mathbf{2.8 \pm 3.3}$

different pairs are different inputs for the Siamese architecture, and so also for each single network. Besides, data augmentation conducted in this manner produces samples with occlusion, and thus our method could be reliable against head occlusions.

4 EXPERIMENTAL RESULTS

Experimental results of the proposed approach are given using two public Kinect datasets for head pose estimation, namely *Biwi Kinect Head Pose Database* and *ICT-3DHP database*. Both of them contains RGB and depth data. To check the reliability of proposed method we performed a cross-dataset validation, training the network on the first dataset and testing on the second one. The evaluation metric is based on the *Mean Average Error* (MAE) between the absolute difference in angle between network predictions and ground truth.

4.1 Biwi Kinect Head Pose Database

Introduced in (Fanelli et al., 2013), it is explicitly designed for head pose estimation from depth data. About 15000 upper body images of 20 people (14 males and 6 females; 4 people were recorded twice) are present. The head rotation spans about ± 75 deg for yaw, ± 60 deg for pitch and ± 50 deg for roll. Both RGB and depth images are acquired sitting in front a stationary *Microsoft Kinect*, with a resolution of 640x480. Besides ground truth pose angles, calibration matrix and head center - the position of the nose tip - are given. Depth images are characterized by visual artifacts, like holes (invalid values in depth map). In the original work (Fanelli et al., 2011), the total number of samples used for training and testing and the subject selection is not clear. We use sequences 11 and 12 to test our network, which correspond to not repeated subjects. Some papers use own method to collect results (e.g. (Ahn et al., 2014)), so their results are not reported and analyzed.

4.2 ICT-3DHP Database

ICT-3DHP Dataset (Baltrušaitis et al., 2012) is a head pose dataset, collected using *Microsoft Kinect* sensor. It contains about 14000 frames (both intensity and depth), divided into 10 sequences. The resolution is 640x480. The ground truth is annotated using a *Polhemus Fastrack* flock of birds tracker, that require a showy white cap, well visible in both RGB and RGB-D frames. This dataset is not oriented for deep learning, because of its small size and the presence of few subjects.

4.3 Quantitative evaluation

The performance of the proposed head pose estimation are compared with a baseline system. To this aim, we trained a single network with the structure of one Siamese component. Input data and data augmentation are the same on both cases. In addition, the results are also compared with other state-of-the-art techniques. As above mentioned, the training has been done on *Biwi dataset* (2 subjects used for test), while the testing phases also exploited the *ICT-3DHP dataset*.

Table 1 reports the experimental results obtained on *Biwi Kinect Head Pose Dataset*. The evaluation protocol is the same proposed in (Fanelli et al., 2011). Results reported in Table 1 show that our method overcomes other state-of-the-art techniques, even those working on both RGB and depth data.

Table 2 reports the results on *ICT-3DHP Dataset*; the values related to (Fanelli et al., 2011) were taken from (Crabbe et al., 2015). On this dataset, the dynamic face crop algorithm is degraded due to an imprecise head center location provided in the available ground truth. The authors published the position of the device exploited to capture the head angle instead of the head itself. Thus, part of the head center locations are inaccurate. To highlight this problem, we report that (Baltrušaitis et al., 2012) had a substantial improvement of using GAVAM (Morency et al., 2008), an adaptive key frame based differential tracker, over all other trackers. Their method in this

Table 2: Results on *ICT-3DHP Dataset*: pitch, roll and yaw are reported in Euler angles.

Method	Data	Pitch	Roll	Yaw
(Saeed and Al-Hamadi, 2015)	RGB+RGB-D	4.9 ± 5.3	$\mathbf{4.4} \pm 4.6$	$\mathbf{5.1} \pm \mathbf{5.4}$
(Fanelli et al., 2011)	RGB-D	5.9 ± 6.3	-	6.3 ± 6.9
(Baltrušaitis et al., 2012)	RGB+RGB-D	7.06	10.48	6.90
Our	RGB-D	5.5 ± 6.5	4.9 ± 5.0	10.8 ± 11.0
Our+Siamese	RGB-D	$\mathbf{4.5} \pm \mathbf{4.6}$	$\mathbf{4.4} \pm \mathbf{4.5}$	9.8 ± 10.1

case reports an absolute error of 2.9 for yaw, 3.14 for pitch and 3.17 for roll. Finally, we highlight the benefit of Siamese training phase. In fact, the proposed approach perform better than the single network as well as the other competitors, even those which rely on both RGB and RGB-D data. The prediction of roll angles is accurate, even if in the second dataset there is a lack of training data images with roll angles.
Figure 3 and Figure 4 report angle frame per error and errors at specific angles for both dataset.
Figure 5 shows an example of working framework for head pose estimation in real time: head center is taken thanks to ground truth data; the face is cropped from raw depth map (in the center image, the blue rectangle) and in the right frame yaw, pitch and roll angles are shown. Total time of processing on a CPU (*Core i7-4790* 3.60GHz) is 11.8 s and on a GPU (*NVidia Quadro k2200*) is 0.146 s, computed on 250 frames from *Biwi*.

5 Conclusion

We present a innovative method to directly extract head angles from depth images in real time, exploiting a deep learning approach. Our technique aim to deal with two main issue of deep architectures in general, and CNNs in particular: the difficulty to solve regression problems and the traditional heavy computational load that compromise real time performance for deep architectures. Our approach is based on Convolutional Neural Network with shallow deep architecture, to preserve time performance, and is designed to resolve a regression task.
There is rich possibility for extensions thanks to the flexibility of our approach: in future work we plan to integrate temporal coherence and stabilization in the deep learning architecture, maintaining real time performance, incorporate RGB or infrared data to investigate the possibility to have a light invariant approach even in particular conditions (e.g. automotive context). Head localization through deep approach could be studied in order to develop a complete framework that can detect, localize and estimate head pose inside a cockpit.

Table 3: Performance evaluation (*fps*)

Method	Time	GPU
(Fanelli et al., 2011)	40 ms/frame	x
(Papazov et al., 2015)	76 ms/frame	
(Yang et al., 2012)	100 ms/frame	
Our	**10 ms/frame**	x

REFERENCES

Ahn, B., Park, J., and Kweon, I. S. (2014). Real-time head orientation from a monocular camera using deep neural network. In *Asian Conference on Computer Vision*, pages 82–96. Springer.

Alioua, N., Amine, A., Rogozan, A., Bensrhair, A., and Rziza, M. (2016). Driver head pose estimation using efficient descriptor fusion. *EURASIP Journal on Image and Video Processing*, 2016(1):1–14.

Baltrušaitis, T., Robinson, P., and Morency, L.-P. (2012). 3d constrained local model for rigid and non-rigid facial tracking. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 2610–2617. IEEE.

Bleiweiss, A. and Werman, M. (2010). Robust head pose estimation by fusing time-of-flight depth and color. In *Multimedia Signal Processing (MMSP), 2010 IEEE International Workshop on*, pages 116–121. IEEE.

Breitenstein, M. D., Kuettel, D., Weise, T., Van Gool, L., and Pfister, H. (2008). Real-time face pose estimation from single range images. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE.

Chen, J., Wu, J., Richter, K., Konrad, J., and Ishwar, P. (2016). Estimating head pose orientation using extremely low resolution images. In *2016 IEEE Southwest Symposium on Image Analysis and Interpretation (SSIAI)*, pages 65–68.

Crabbe, B., Paiement, A., Hannuna, S., and Mirmehdi, M. (2015). Skeleton-free body pose estimation from depth images for movement analysis. In *Proc. of the IEEE International Conference on Computer Vision Workshops*, pages 70–78.

Craye, C. and Karray, F. (2015). Driver distraction detection and recognition using RGB-D sensor. *CoRR*, abs/1502.00250.

Doumanoglou, A., Balntas, V., Kouskouridas, R., and Kim, T. (2016). Siamese regression networks with efficient mid-level feature extraction for 3d object pose estimation. *CoRR*, abs/1607.02257.

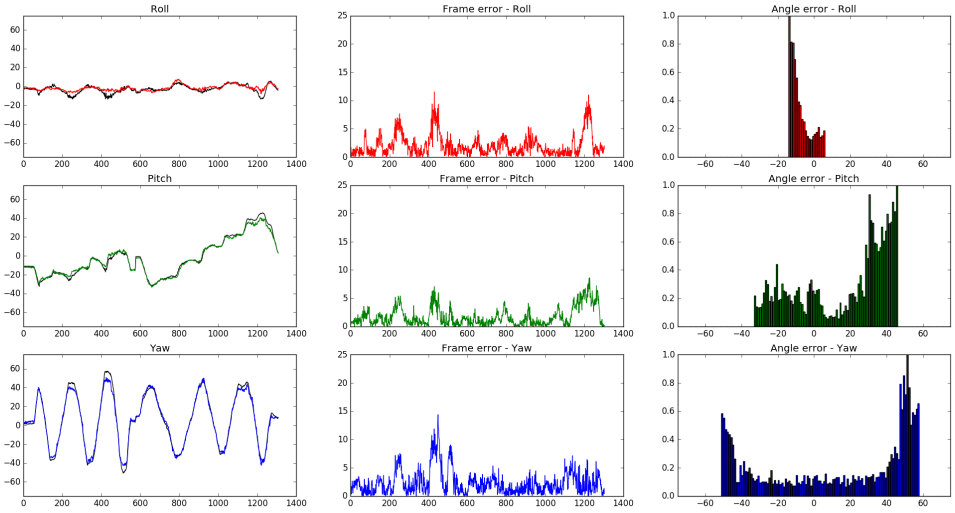

Figure 3: Experimental results on *Biwi dataset*: ground truth is black. The second column reports the angle error per frame, while the last column reports histograms that highlight the errors at specific angles.

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

Fanelli, G., Dantone, M., Gall, J., Fossati, A., and Van Gool, L. (2013). Random forests for real time 3d face analysis. *International Journal of Computer Vision*, 101(3):437–458.

Fanelli, G., Gall, J., Romsdorfer, H., Weise, T., and Gool, L. V. (2010). A 3-d audio-visual corpus of affective communication. *IEEE Transactions on Multimedia*, 12(6):591 – 598.

Fanelli, G., Gall, J., and Van Gool, L. (2011). Real time head pose estimation with random regression forests. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 617–624. IEEE.

Ghiass, R. S., Arandjelović, O., and Laurendeau, D. (2015). 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*, pages 25–34. ACM.

Hoffer, E. and Ailon, N. (2015). Deep metric learning using triplet network. In *Proc. of Int'l Workshop on Similarity-Based Pattern Recognition*, pages 84–92. Springer.

Ji, S., Xu, W., Yang, M., and Yu, K. (2013). 3d convolutional neural networks for human action recognition. *IEEE Transactions on pattern analysis and machine intelligence*, 35(1):221–231.

Kendall, A., Grimes, M., and Cipolla, R. (2015). Posenet: A convolutional network for real-time 6-dof camera relocalization. In *Proc. of the IEEE Int'l Conf. on Computer Vision*, pages 2938–2946.

Kondori, F. A., Yousefi, S., Li, H., Sonning, S., and Sonning, S. (2011). 3d head pose estimation using the kinect. In *Wireless Communications and Signal Processing (WCSP), 2011 International Conference on*, pages 1–4. IEEE.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105.

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

Malassiotis, S. and Strintzis, M. G. (2005). Robust real-time 3d head pose estimation from range data. *Pattern Recognition*, 38(8):1153–1165.

Morency, L.-P., Whitehill, J., and Movellan, J. (2008). Generalized adaptive view-based appearance model: Integrated framework for monocular head pose estimation. In *Proc. of 8th IEEE Int'l Conf. on Automatic Face & Gesture Recognition, 2008. FG'08.*, pages 1–8. IEEE.

Murphy-Chutorian, E. and Trivedi, M. M. (2009). Head pose estimation in computer vision: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.*, 31(4):607–626.

Padeleris, P., Zabulis, X., and Argyros, A. A. (2012). Head pose estimation on depth data based on particle swarm optimization. In *2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pages 42–49. IEEE.

Papazov, C., Marks, T. K., and Jones, M. (2015). 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*, pages 4722–4730.

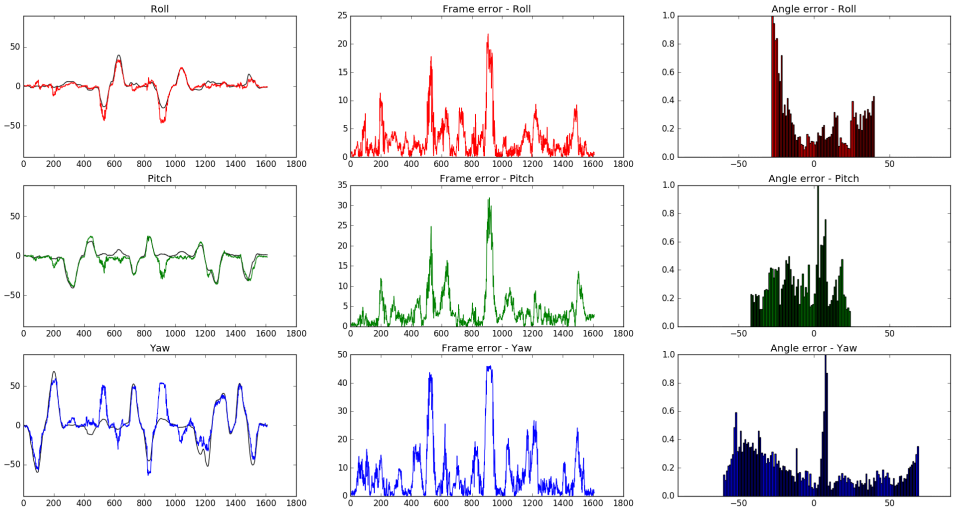

Figure 4: Experimental results on *ICT-3DHP dataset*: see Figure 3 for explanation.

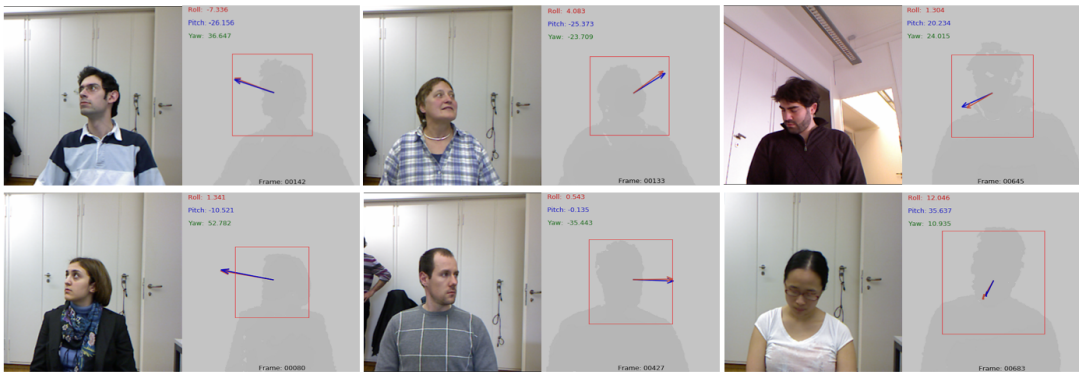

Figure 5: The first and the third columns show RGB frames, the second and the fourth the correspondent depth map frame with a red rectangle that reveals the crop for the face extraction (see Section 3.1). The blue arrow is ground truth, while the red one is our prediction. Numerical angle prediction is reported on the top. Images taken from *Biwi Dataset*.

Rahman, H., Begum, S., and Ahmed, M. U. (2015). Driver monitoring in the context of autonomous vehicle.

Saeed, A. and Al-Hamadi, A. (2015). Boosted human head pose estimation using kinect camera. In *Image Processing (ICIP), 2015 IEEE International Conference on*, pages 1752–1756. IEEE.

Seemann, E., Nickel, K., and Stiefelhagen, R. (2004). Head pose estimation using stereo vision for human-robot interaction. In *FGR*, pages 626–631. IEEE Computer Society.

Sun, Y., Chen, Y., Wang, X., and Tang, X. (2014). Deep learning face representation by joint identification-verification. In *Advances in Neural Information Processing Systems*, pages 1988–1996.

Viola, P. and Jones, M. J. (2004). Robust real-time face detection. *International journal of computer vision*, 57(2):137–154.

Yang, J., Liang, W., and Jia, Y. (2012). Face pose estimation with combined 2d and 3d hog features. In *Pattern Recognition (ICPR), 2012 21st International Conference on*, pages 2492–2495. IEEE.

Yi, K. M., Verdie, Y., Fua, P., and Lepetit, V. (2015). Learning to assign orientations to feature points. *arXiv preprint arXiv:1511.04273*.