Published in: Pattern Recognition Letters journal.
This version of the article has been accepted for publication, after peer review and is subject to Elsevier terms of use.
The Version of Record is available online at: https://www.sciencedirect.com/science/article/pii/S0167865520303779?via%3Dihub

Head pose estimation by regression algorithm

Andrea F. Abate[a], Paola Barra[a], Chiara Pero[a,⇤⇤], Maurizio Tucci[a]

[a]*Dept. of Computer Science. University of Salerno. Salerno, Italy*

ABSTRACT

Head pose estimation is a very in-depth topic in the context of biometric recognition, especially in video surveillance, because the rotation of the head can affect the recognition of some features of the face. Being able to recognize in advance the pose of the head in pitch, yaw and roll enable frontalization or the extraction of a frame in which a face is frontal in order to allow a more accurate recognition. In this work the Web-Shaped Model algorithm is used for a coding of the pose of the face and then we apply regression algorithms to predict the pose of the face. The presented method is tested on some of the most well-known datasets for the head pose estimation as Biwi, AFLW2000 and Pointing'04 and compared with the various state of the art methods that use these datasets.

1. Introduction

The need to use biometric recognition has become fundamental in recent decades in fields such as access control and video surveillance. In order to face real problems, the research must be directed in challenging directions that foresee situations of uncontrolled lighting, occlusions, facial expressions and head rotations. In particular, these situations are more relevant in conditions of acquisitions not managed by an operator. All that can lead to low face recognition rates or even failures.

Among all the factors that can weaken the quality of recognition, the Head Pose Estimation (HPE) plays a fundamental role. Rotations of the head, see Figure 1, calculated with respect to the x axis, y axis and z axis (respectively *pitch*, *yaw*, and *roll*), can cause occlusions and modify some relationships between elements of the face (De Marsico et al., 2010) making processing operations complicated.

In this paper we use a regression algorithm applied to the Web-Shaped Model (WSM) algorithm (Barra et al., 2020) to predict the pose of the face. We executed our method over some classical datasets as Pointing'04, AFLW2000 and Biwi and then compared results with the various state of the art methods.

⇤⇤Corresponding author:
e-mail: cpero@unisa.it (Chiara Pero)

The method presented makes a rapid estimate of the head pose without training any neural network, therefore without the need for a large amount of data, unlike (Ahn et al., 2018) which uses a deep network to estimate the pose in real time. The main difficulty in approaches using deep learning is that the data provided as input, therefore the images of the face, must be well labeled. In head pose estimation, labeling the images with the degrees of *pitch*, *yaw* and *roll* of the head rotation is a complex and time-consuming process, especially if the process is manual and not supported by specific software or sensors.

The paper is organised as follows: Section 2 presents the state of the art in appearance-based methods for pose estimation; Section 3 shows the WSM; In Section 4 we present the method and experimental results and finally Section 5 concludes the paper.

2. Related work

The classification of existing approaches for HPE into a single taxonomy is a rather difficult task, since there exist many different approaches and classification may result ambiguous. A rough classification, for example, is the following:

- Tracking methods (Demirkus et al., 2014) apply to video sequences and are based on differences between consecutive frames to identify the head pose of subjects;

- Detector methods (Lee et al., 2015) use different classifiers in order to identify relevant poses;
- Methods based on geometric relationships (Papazov et al., 2015) detect the head pose by considering face's anatomical features such as eyes, nose and mouth;
- Methods based on appearance (Smith et al., 2014) try to estimate the pose of an image by matching it against a set of prototypical poses.

For an extended survey, the reader may refer to (Murphy-Chutorian and Trivedi, 2008).

We propose an approach belonging to the latter type of estimation techniques. Figure 1 shows how the head pose is represented in terms of pitch, yaw and roll angles.

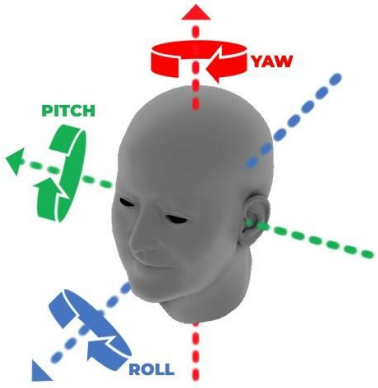

Fig. 1. Head rotation angles

In (Raza et al., 2018) two different convolutional neural networks (CNN) are considered in order to estimate head and body pose respectively. In (Liu and Ma, 2015) an approach is presented which estimates the pose of the whole body using three modules: 1) the first extracts the characteristics relating to the appearance of a subject using the HOG method; 2) the second classify the movements of a subject; 3) the third combines the obtained information to estimate the orientation of the body. In (Pawelczyk and Kawulok, 2014), the authors show that the orientation of the nose has a discriminatory power in classifying the pose and in evaluating head orientation. The video scene information is applied in (Chamveha et al., 2011) to evaluate head orientation using the direction of subject's movement. Below are listed the methods that we considered for comparison in the Section 4 and which are classified according to the use they make of CNN. CNNs are considered an excellent tool for image processing, for this reason there are numerous applications that use CNNs for head pose estimation, in particular we list two interesting approaches. The first, presented in (Ruiz et al., 2018), uses a Multiloss ResNet50 to estimate the pose of the head in pitch, yaw and roll starting directly from the face image; the second (Ranjan et al., 2017) uses a Hyperface network to detect the face, locate the reference points and estimate the head pose in pitch, yaw and roll. In (Zhu et al., 2017) is proposed the 3DDFA (3D Dense Face Alignment) approach, this method uses a CNN to adapt and align a 3D face model to a 2D image. KEPLER (Kumar et al., 2017) uses an H-CNN regressor to solve the problem of face alignment and HPENN presented in (Stiefelhagen, 2004) processes the facial images and estimates the horizontal and vertical alignment of the head (pitch and yaw) through a neural network. In (Abate et al., 2019) a QT-PYR (QuadTree Pitch Yaw and Roll) method is presented which after the extraction of the facial landmarks uses a QuadTree structure to extract a vector that describes the pose of the face, this vector is given as input to a classifier that estimates the pose. hGLLiM (Drouard et al., 2017) propose a mixture of linear regressions in conjuction with partially-latent output which map high dimensional feature vectors into head pose angles and bounding boxes of faces to make predictions in presence of unobservable phenomena. In a similar way, (Drouard et al., 2015) uses face bounding boxes to create HOG-based descriptors that map to head poses. A method for pose estimation on low resolution unconstrained images is presented in (Gourier et al., 2006). A linear auto-associative memory is obtained by training with Widrow-Hoff correction rule. Cosine between source and reconstructed images is used to estimate pose of the head. QuatNet is a multi-regression loss function applied in (Hsu et al., 2018), to train a CNN to process RGB images with no depth information and determine head poses.

FSA-Net is another approach to head pose estimation based on Neural Networks, which relies on regression and feature aggregation (Yang et al., 2019). The method learns a fine-grained structure mapping to spatially group features before aggregation, unlike other existing feature aggregation methods that represent input images in terms of a map of features and their spatial relationships. A novel approach is presented in (Wang et al., 2019). It performs head pose estimation by applying a deep neural network in a Coarse-to-Fine strategy. It jointly trains two subnetworks to classify the input image into four categories according to a given number of GoogleNet blocks and to accurately estimate pose parameters.

3. Web-Shaped Model

The approach in this model is to estimate the head pose, using two approaches in cascade:

- we receive as input the photo of a face and landmark detector extracts the coordinates of 68 face landmarks;
- the WSM draws a virtual spider-web on the face and, based on the position of the landmarks, extracts a descriptive array of the pose of the face.

The first approach is presented in (Kazemi and Sullivan, 2014) and it was chosen due to its robustness. It predicts 68 face landmarks, described as points of coordinates (x, y) on the image, that outline the profile of the face, mouth, nose, eyes and eyebrows. The second approach, presented in (Barra et al., 2020), represents a spider-web shaped model (WSM). The process of features extraction is described in Figure 2-A). Following the lankmark coordinates, a spider-web is "drawn" on the face as input. This spider-web has the center at the tip of the

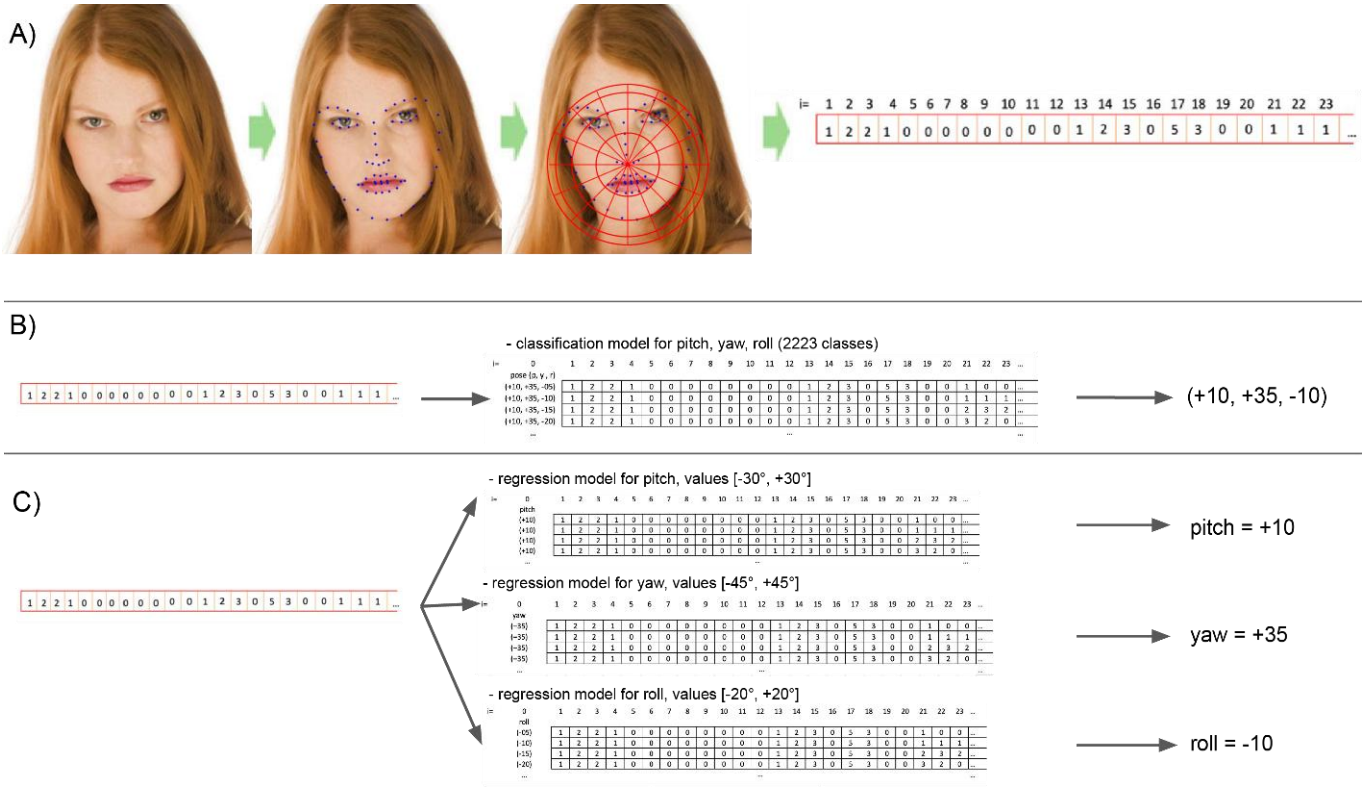

Fig. 2. Representation of the method: A) Summary of the WSM approach; B) Classification; C) Regression.

nose (landmark number 33 of the previous model) and the radius is determined by the furthest landmark from the center. In doing so, this model adapts to the size of the input image.

Being the point O of coordinate $O = (x_{33}, y_{33})$ the center of the model, and being the points $P_j = (x_j, y_j)$, $j = 1,...,68$ except $j = 33$ the remaining landmarks, the Euclidean distance d between O and the farthest landmark, i.e., $r = max\ d(O, P_j)$ where $j = 1,...,68$, is the radius r. The method presented in (Barra et al., 2020) assigns each face landmark in a spider-web ”sector” depending on where it falls. When each reference point has been assigned to a sector, the method extracts a feature vector that describes the coding of the pose of the head.

We define as: ”circles” the concentric circles of the spider-web , ”slices” the spider-web slices delimited by two consecutive rays, ”quarter” identifies a quadrant of a Cartesian plane that has its center in the center of the spider-wed and ”sectors” the section of spider-web delimited by slices and circles. The spider-web configuration used in this work is ”4C 4S var4”, C stays for ”circles” and S stays for ”slices”: so this composition consists in 4 concentric circles and 4 slices for each quarter, as in the third photo in Figure 2-A). The suffix var4 identifies the variant n°4, among those tested, of the ray length: R; 9/10*R; 7/10*R; 4/10*R. The configuration ”4C 4S var4” was chosen because it was the one that showed the best results in (Barra et al., 2020). An array of values is extracted from the spider-web that represents the coding of the pose. The array has a size equivalent to the number of sectors of the spider-web, to be exact $m\ x\ 4\ x\ c$ where m is the number of slices in a quarter and c represents the number of circles, therefore in the case of ”4C 4S var4” the array size is 64.

The sectors in the spider-web are numbered starting from the outermost to the innermost one in clockwise order, and this also determines the order of the elements within the array. Each element contains the number of landmarks falling within that sector, more details on landmark/sector association in the pose vector are given in (Barra et al., 2020).

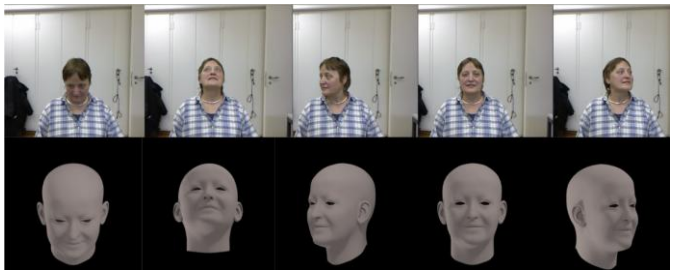

Fig. 3. Samples from Biwi dataset: on the top 5 RGB-images of the subject 01, and on the bottom 5 depth images of the same subject.

3.1. Classification and Regression in WSM

To identify the pose in the extracted array we use the classification method used in (Barra et al., 2020), represented in Figure 2-B), and the regression method. The pose feature array obtained through the classification method is compared with prototypical vectors extracted, using the same method from samples where poses are known, in order to find the most similar one. Each of the arrays corresponds to an encoding of a pose

in terms of pitch, yaw and roll. We make a comparison between the pose feature vector we extracted and those stored in the dataset to perform the pose classification. As output we obtain the pose whose reference vector has the lowest (Euclidean) distance from vector extracted from the incoming image. In this paper, starting from this approach, we use regression to outperform results. So, for each experiment, 3 different regression models were built, for pitch, yaw and roll:

- a regression model for pitch prediction that returns a number in the continuous range [- 30°, +30°];
- a regression model for yaw prediction that returns a number in the continuous range [- 45°, +45°];
- a regression model for roll prediction that returns a number in the continuous range [- 20°, +20°].

The method is represented in Figure 2-C). In doing so, the minimum error can be less than 5° unlike the previous method that uses classification. For further information on the regression methods, refer to the 4.2 section.

4. Experimental results

4.1. Datasets

To compare the results with the state of the art, the described model was trained and tested with different datasets: Biwi Kinect Head Pose dataset (Fanelli et al., 2013), AFLW2000 dataset (Koestinger et al., 2011) and Pointing'04 dataset (Gourier et al., 2004).

Biwi Kinect Head Pose Database (Fanelli et al., 2013) contains 24 sequences of 20 different people (6 females and 14 males - 4 people were recorded twice) from RGB-D cameras. The dataset includes over 15000 frames, including RGB-images and the depth images, captured under indoor environment. Beside this, Biwi dataset includes for each subject an *.obj* file; this file represents the 3D model of the subject. Some samples of the dataset are shown in Figure 3, on the top there are 5 RGB-images of the subject 01, and on the bottom there are 5 depth images of the same subject. The provided data are elaborated with the graphic engine Blender, for each subject the head is positioned in a frontal pose (0° of pitch, yaw and roll). So thanks to a script the following poses were extracted automatically from each 3D head model:

- pitch: range [-30°, +30°];
- yaw: range [-45°, +45°];
- roll: range [-20°, +20°].

The total number of poses extracted from each head amounts to 2223: 13 variations in pitch, 19 in yaw and 9 in roll, with a difference of 5° each, for a total of 44460 poses extracted in the entire dataset. This procedure allows us to annotate each image with pitch, yaw and roll, in order to use the figures as a ground truth for experiments.

AFLW2000 dataset provides face images and corresponding annotation of the groundtruth head poses for the first 2000 images in AFLW (Annotated Facial Landmarks in the Wild) (Koestinger et al., 2011). AFLW contains around 25,000 photos of faces, mostly RGB, collected by the social network Flickr. AFLW samples are very varied for different poses, ethnic traits, ages, facial expressions and environmental conditions. From *AFLW2000* we exclude the images whose angles exceed the interval in pitch [-30°, +30°], in yaw [-45°, +45°] and in roll [-20°, +20°]. Figure 4 shows some samples from this dataset.

Fig. 4. Images selected from AFLW2000 dataset.

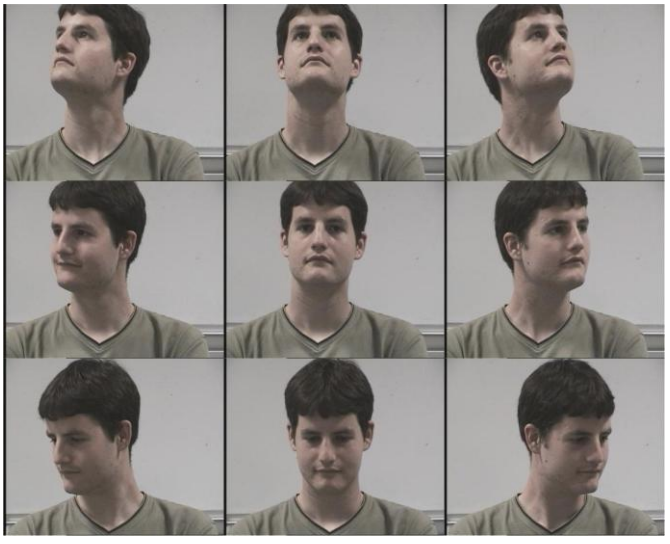

Fig. 5. Images from the Pointing'04 dataset. The pose of the images varies from -30° to +30° for pitch and from -45° to +45° for yaw.

Pointing'04 dataset (Gourier et al., 2004) is a historical dataset, widely used in computer vision for head pose estimation (Drouard et al., 2015; Kong and Mbouna, 2015), as it is composed of faces acquired in the wild and labeled with differences in variety of poses. It uses a difference of 15° and 30° between two consecutive poses in yaw and pitch orientation and does not contain roll information. This dataset includes, for each of the 15 subjects, 2 series of 93 images. Pointing'04 contains 2790 images of subjects. In Figure 5 we can see a subset of 9 images from Pointing'04 dataset.

4.2. Supervised Learning: Regression vs. Classification

Supervised Learning (SL) refers to a class of algorithms that learns a function f that maps an input space X to an output space Y based on a sequence of input-output pairs. There are two main groups of Supervised learning (SL) methods, namely *classification* and *regression*, depending of the nature of the output space. Classification methods predict discrete responses and aim to assign a label y_j e Y at each input element x_i e X. Regression models predict continuous responses (Fiorucci et al., 2020). Relationship between two variables are modeled by linear regression trying to fit linear equation to observed data. Consequently, classification techniques provide the model or function that predicts new data in discrete categories; conversely, regression methods model functions at constant values, which means that it predicts data in continuous numeric data. Our approach stimulates the sensitivity of the regression methods to identify the head pose estimation. The goal is to predict the value of the dependent variable for the three angular values, respectively for pitch, yaw and roll axes associated to head's degrees of freedom, for which some information relating to the explanatory variables is available, in order to estimate the effect on the dependent variable.

4.2.1. Linear Regression

A linear relationship between an independent variable x, usually referred to as a predictor variable and a dependent variable y, i.e. a criterion variable, is expressed by the following equation:

$$y = mx + b \tag{1}$$

where m is the slope of the relationship and b is the y intercept. Linear Regression (LR) is employed to fit a predictive model to the set of training observations (x, y) (Bishop, 2006). The result is the prediction equation that gives the best estimate of y in terms of x. Then, the fitted model is used to make predictions of y for new instances of x.

4.2.2. Bayesian Ridge Regression

Bayesian Regression estimates a probabilistic model using regularization parameters in the procedure (Tipping, 2001). It assumes that the response y results from a probability distribution rather than estimated as a single value. Formally, to obtain a fully probabilistic model, the output y is assumed to be Gaussian distributed around X_w:

$$p(y|X, w, \boldsymbol{\alpha}) = \mathrm{N}(y|0, X_w, \boldsymbol{\alpha}) \tag{2}$$

where $\boldsymbol{\alpha}$ is again treated as a random variable that is to be estimated from the data. A Bayesian view of Ridge Regression (BRR) is obtained in Eq. 3; The spherical Gaussian is adopted for the prior of the coefficient w:

$$p(w|\boldsymbol{h}) = \mathrm{N}(w|0, \boldsymbol{h}^{-1}, \mathbf{I}_p) \tag{3}$$

The priors over $\boldsymbol{\alpha}$ and $\boldsymbol{h}$ represent the gamma distributions. The parameters w, $\boldsymbol{\alpha}$ and $\boldsymbol{h}$ are estimated jointly during the fit of the model, the regularization parameters and being estimated by maximizing the log marginal likelihood (MacKay, 1992).

4.2.3. Logistic Regression

In the existing multiple regression models, Logistic Regression (LgR) represents a particular case of the generalized linear model. It is a regression model applied in cases where the dependent variable y is dichotomous (Tipping, 2001), (Fiorucci et al., 2020). Therefore, LgR allows to analyze the relationship between a dichotomous variable and one or more explanatory variables (both continuous and categorical). In general, the Logistic model can be represented by the following equation:

$$y = \frac{e^{\alpha+\beta x}}{1 + e^{\alpha+\beta x}} \tag{4}$$

where x is the input value, $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ the coefficients of the input value (constant real numbers) and y the predict value. Our implementation fit the model with *L2* regularization. More details about LgR algorithm can be found at (Pedregosa et al., 2011).

4.3. Results and Discussion

Our method is compared with some of the state-of-art appearance-based methods for pose estimation, described in Section 2. We also analyzed our results with respect to different methods like Support Vector Regression - SVR (Smola and Scho¨lkopf, 2004), Gaussian processes - GPR (Rasmussen, 2003), Partial Least Square Regression - PLS (Abdi, 2003), and 3DFM by Kong and Mbouna (2015).

It is possible to divide these strategies refers to the model adopted in the experimental phase; in particular, we distinguish model-based methods and neural network-based methods. The 70% of the datasets images are adopted to training the model and the remaining are used for testing. To evaluate the proposed approach it is used a performance index commonly present as an evaluation criterion in HPE, namely the *Mean Absolute Error* (MAE). The MAE represents the distance between the predicted and the ground truth poses, as defined by the following equation:

$$MAE = \frac{1}{n} \sum_{j=1}^{n} |y_j - \hat{y_j}| \tag{5}$$

where y_j indicates the true angular value poses and $\hat{y_j}$ is the predicted pose. Table 1 and Table 2 compare, respectively, our WSM-regression methods with the approaches Biwi RGB Dataset and Biwi 3D Dataset (Fanelli et al., 2013). All the results show the MAE of head pose estimation obtained for each of three values (yaw, pitch and roll) and also an overall MAE of the error along the three axes. Therefore, the lowest MAE value represents the best model. The Biwi Dataset includes information in different modes, in fact it is the only released dataset that contains RGB and depth images.

As illustrated in Table 1, the comparison results over Biwi RGB Dataset provides a lower MAE value than all other state-of-art approaches, including yaw, pitch and roll. Table 2 shows the comparison results over Biwi 3D dataset and it is possible to appreciate the accuracy of the proposed approach respect to the other models. The pitch angular error of WSM represents the only exception.

Table 1. The Mean Absolute Error (MAE) on the Biwi RGB Dataset

Method	Yaw	Pitch	Roll	MAE
Multi-Loss ResNet50	5.17	6.97	3.39	5.177
hGLLiM	6.06	7.65	5.62	6.44
FSA-Net	2.89	4.29	3.60	3.593
Coarse-to-Fine	4.76	5.48	4.29	4.84
QuatNet	4.01	5.49	2.93	4.14
WSM-LR	3.63	3.44	2.15	3.07
WSM-BRR	3.61	3.35	2.11	3.02
WSM-LgR	**3.12**	**2.31**	**1.88**	**2.43**

Table 2. The Mean Absolute Error (MAE) on the Biwi 3D Dataset

Method	Yaw	Pitch	Roll	MAE
GPR	7.72	9.64	6.01	7.79
PLS	7.35	7.87	6.11	7.11
SVR	6.98	7.77	5.14	6.63
QT-PYR	5.41	12.80	6.33	8.18
FSA-Net	4.27	4.96	2.76	3.996
Coarse-to-Fine	5.16	4.23	5.39	4.926
WSM	6.21	**3.95**	4.16	4.77
WSM-LR	3.06	4.63	2.56	3.41
WSM-BRR	3.06	4.62	2.56	3.41
WSM-LgR	**2.47**	4.56	**2.13**	**3.05**

Table 3 shows the results over AFLW2000. This dataset includes images with large variations, different illumination and occlusion conditions and doesn't have a precise annotation of the poses. The overall MAE calculated through WSM-regression methods are similar to WSM and are better of the other approaches that use neural network-based methods.

Table 3. The Mean Absolute Error (MAE) on the AFLW2000 Dataset

Method	Yaw	Pitch	Roll	MAE
Multi-Loss ResNet50	6.470	6.559	5.436	6.155
Hyperface	7.61	6.13	3.92	5.89
KEPLER	6.45	5.85	8.75	7.01
3DDFA	5.400	8.530	8.250	7.393
FAN	6.358	12.277	8.714	9.116
QT-PYR	7.6	7.6	7.17	7.45
QuatNet	3.973	5.615	3.92	4.503
WSM	**3.11**	4.82	**2.25**	**3.39**
WSM-LR	3.88	**4.66**	2.50	3.68
WSM-BRR	3.82	4.67	2.49	3.66
WSM-LgR	4.31	5.34	2.62	4.09

Finally, in Table 4 are showed the results over Pointing'04. This dataset provides only pitch and yaw angles (roll is absent from the head pose images), therefore none of the state-of-the-art methods reports roll in the experiments evaluation. As happened on Biwi 3D Dataset in Table 2, our methods are the best on predicting the yaw angle error and provide the lowest overall MAE (exception is represented by pitch angular error) and Logistic Regression model (LgR) produces the best results.

Table 4. The Mean Absolute Error (MAE) on the Pointing'04 Dataset

Method	Yaw	Pitch	MAE
HPENN	9.7	9.5	9.6
SFS	12.1	7.3	9.7
SVR	12.82	11.25	12.035
hGLLiM	7.93	8.47	8.2
Probabilistic HDR	8.70	8.85	8.775
3DFM	10.98	9.71	10.345
WSM	10.63	**6.34**	8.4
WSM-LR	5.61	7.73	6.67
WSM-BRR	5.60	7.68	6.64
WSM-LgR	**4.44**	7.55	**5.99**

5. Conclusions

In this work, we presented three Regression methods, combined with Web-Shaped Model, to Head Pose Estimation. The WSM-regression approach stimulates the sensitivity of the regression methods to identify the head pose estimation. The experimental results have been compared with many state-of-art approaches, obtaining good performances in terms of accuracy. In many cases, the proposed fusion methodology applied over several public datasets demonstrates to perform better than many of those based on neural networks. This result is particularly relevant considering that WSM-regression methods do not require any training phase.

References

Abate, A.F., Barra, P., Bisogni, C., Nappi, M., Ricciardi, S., 2019. Near real-time three axis head pose estimation without training. IEEE Access 7, 64256–64265.

Abdi, H., 2003. Partial least square regression (pls regression). Encyclopedia for research methods for the social sciences 6, 792–795.

Ahn, B., Choi, D.G., Park, J., Kweon, I.S., 2018. Real-time head pose estimation using multi-task deep neural network. Robotics and Autonomous Systems 103, 1–12.

Barra, P., Barra, S., Bisogni, C., De Marsico, M., Nappi, M., 2020. Web-shaped model for head pose estimation: An approach for best exemplar selection. IEEE Transactions on Image Processing 29, 5457–5468.

Bishop, C.M., 2006. Pattern recognition and machine learning. Springer.

Chamveha, I., Sugano, Y., Sugimura, D., Siriteerakul, T., Okabe, T., Sato, Y., Sugimoto, A., 2011. Appearance-based head pose estimation with scene-specific adaptation, in: 2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops), IEEE. pp. 1713–1720.

De Marsico, M., Nappi, M., Riccio, D., 2010. Measuring sample distortions in face recognition, in: Proceedings of the 2nd ACM workshop on Multimedia in forensics, security and intelligence, pp. 83–88.

Demirkus, M., Precup, D., Clark, J.J., Arbel, T., 2014. Probabilistic temporal head pose estimation using a hierarchical graphical model, in: European conference on computer vision, Springer. pp. 328–344.

Drouard, V., Ba, S., Evangelidis, G., Deleforge, A., Horaud, R., 2015. Head pose estimation via probabilistic high-dimensional regression, in: 2015 IEEE International Conference on Image Processing (ICIP), IEEE. pp. 4624–4628.

Drouard, V., Horaud, R., Deleforge, A., Ba, S., Evangelidis, G., 2017. Robust head-pose estimation based on partially-latent mixture of linear regressions. IEEE Transactions on Image Processing 26, 1428–1440.

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

Fiorucci, M., Khoroshiltseva, M., Pontil, M., Traviglia, A., Del Bue, A., James, S., 2020. Machine learning for cultural heritage: A survey. Pattern Recognition Letters 133, 102–108.

Gourier, N., Hall, D., Crowley, J.L., 2004. Estimating face orientation from robust detection of salient facial features. ICPR International Workshop on Visual Observation of Deictic Gestures .

Gourier, N., Maisonnasse, J., Hall, D., Crowley, J.L., 2006. Head pose estimation on low resolution images, in: International Evaluation Workshop on Classification of Events, Activities and Relationships, Springer. pp. 270–280.

Hsu, H.W., Wu, T.Y., Wan, S., Wong, W.H., Lee, C.Y., 2018. Quatnet: Quaternion-based head pose estimation with multiregression loss. IEEE Transactions on Multimedia 21, 1035–1046.

Kazemi, V., Sullivan, J., 2014. One millisecond face alignment with an ensemble of regression trees, in: 2014 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1867–1874.

Koestinger, M., Wohlhart, P., Roth, P.M., Bischof, H., 2011. Annotated facial landmark in the wild: A large-scale, real world database for facial landmark localization. Proc. First IEEE International Workshop on Benchmarking Facial Image Analysis Technologies , 2144–2151.

Kong, S.G., Mbouna, R.O., 2015. Head pose estimation from a 2d face image using 3d face morphing with depth parameters. IEEE Transactions on Image Processing 24, 1801–1808.

Kumar, A., Alavi, A., Chellappa, R., 2017. Kepler: Keypoint and pose estimation of unconstrained faces by learning efficient h-cnn regressors, in: 2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017), IEEE. pp. 258–265.

Lee, D., Yang, M.H., Oh, S., 2015. Fast and accurate head pose estimation via random projection forests, in: Proceedings of the IEEE International Conference on Computer Vision, pp. 1958–1966.

Liu, H., Ma, L., 2015. Online person orientation estimation based on classifier update, in: 2015 IEEE International Conference on Image Processing (ICIP), IEEE. pp. 1568–1572.

MacKay, D.J., 1992. Bayesian interpolation. Neural computation 4, 415–447.

Murphy-Chutorian, E., Trivedi, M.M., 2008. Head pose estimation in computer vision: A survey. IEEE transactions on pattern analysis and machine intelligence 31, 607–626.

Papazov, C., Marks, T.K., Jones, M., 2015. Real-time 3d head pose and facial landmark estimation from depth images using triangular surface patch features, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 4722–4730.

Pawelczyk, K., Kawulok, M., 2014. Head pose estimation relying on appearance-based nose region analysis, in: International Conference on Computer Vision and Graphics, Springer. pp. 510–517.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E., 2011. Scikit-learn: Machine learning in Python. Journal of Machine Learning Research 12, 2825–2830.

Ranjan, R., Patel, V.M., Chellappa, R., 2017. Hyperface: A deep multi-task learning framework for face detection, landmark localization, pose estimation, and gender recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence 41, 121–135.

Rasmussen, C.E., 2003. Gaussian processes in machine learning, in: Summer School on Machine Learning, Springer. pp. 63–71.

Raza, M., Chen, Z., Rehman, S.U., Wang, P., Bao, P., 2018. Appearance based pedestrians' head pose and body orientation estimation using deep learning. Neurocomputing 272, 647–659.

Ruiz, N., Chong, E., Rehg, J.M., 2018. Fine-grained head pose estimation without keypoints, in: Proceedings of the IEEE conference on computer vision and pattern recognition workshops, pp. 2074–2083.

Smith, B.M., Brandt, J., Lin, Z., Zhang, L., 2014. Nonparametric context modeling of local appearance for pose-and expression-robust facial landmark localization, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1741–1748.

Smola, A.J., Scho¨lkopf, B., 2004. A tutorial on support vector regression. Statistics and computing 14, 199–222.

Stiefelhagen, R., 2004. Estimating head pose with neural networks-results on the pointing04 icpr workshop evaluation data, in: Proc. Pointing 2004 Workshop: Visual Observation of Deictic Gestures, pp. 21–24.

Tipping, M.E., 2001. Sparse bayesian learning and the relevance vector machine. Journal of machine learning research 1, 211–244.

Wang, Y., Liang, W., Shen, J., Jia, Y., Yu, L.F., 2019. A deep coarse-to-fine network for head pose estimation from synthetic data. Pattern Recognition 94, 196–206.

Yang, T.Y., Chen, Y.T., Lin, Y.Y., Chuang, Y.Y., 2019. Fsa-net: Learning fine-grained structure aggregation for head pose estimation from a single image, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1087–1096.

Zhu, X., Liu, X., Lei, Z., Li, S.Z., 2017. Face alignment in full pose range: A 3d total solution. IEEE transactions on pattern analysis and machine intelligence 41, 78–9