www.ietdl.org

Published in IET Computer Vision
Received on 9th December 2013
Revised on 3rd March 2014
Accepted on 16th March 2014
doi: 10.1049/iet-cvi.2013.0320

ISSN 1751-9632

Multimodal random forest based tensor regression

Sertan Kaymak, Ioannis Patras

School of Electronic Engineering and Computer Science, Queen Mary University of London, UK
E-mail: s.kaymak@qmul.ac.uk

Abstract: This study presents a method, called random forest based tensor regression, for real-time head pose estimation using both depth and intensity data. The method builds on random forests and proposes to train and use tensor regressors at each leaf node of the trees of the forest. The tensor regressors are trained using both intensity and depth data and their votes are fused. The proposed method is shown to outperform current state of the art approaches in terms of accuracy when applied to the publicly available Biwi Kinect head pose dataset.

1 Introduction

Several applications, including human-centred user interfaces, can greatly benefit from fast and accurate three-dimensional (3D) head pose estimation. Until recently, the majority of the work on head pose estimation used data that contained little head pose variations and utilised intensity information alone, [1–5]. However, the accuracy of such works can be affected by several factors, such as illumination changes. Facilitated by the availability of cheap depth cameras, there has been a growing interest in methods that use depth data.

In this paper, we address the problem of head pose estimation by extending previous methods on random forest based head pose estimation in two ways: (i) by using tensor-based regression at each leaf node; and (ii) by fusing depth and intensity data. This approach differs to classical random forests, in which the prediction models at each leaf node disregard the data of the sample that arrives at the leaf in question. Indeed, typical random forest methods employ leaf node prediction models that rely on the statistics of the training samples that arrive at the leaf in question. Typically the mean, and sometimes also the covariance matrix are used. Here, we propose to use stronger regression models in order to estimate parameters more accurately. We specifically propose to use tensor regression models as they have shown good generalisation properties when the availability of data is sparse. This is often the case in our work, where the number of the samples that arrive at several leaf nodes can be small.

In Fig. 1, we give an outline of the proposed method. First, a regression forest is learned by passing randomly extracted depth patches with head pose parameters into the tree. At each non-leaf node, each patch is sent either to the left or to the right branch according to the result of a test on features extracted from the patch in question.

The test is chosen at training time as the one that maximises the information gain on the head pose parameters that will be achieved if the test in question is be applied on the training patches that arrive at the internal node in question. Once a test is chosen, it is applied on the training patches that arrive at that internal node. This is repeated recursively until a stopping criterion is met.

At the leaf nodes, a typical regression forests model calculate simple statistics of targets, in our case the head pose parameters, that are associated with the training patches that arrive at them. Typically, the mean and the (co)covariance are estimated, so that when a test patch arrives at the leaf in question a probabilistic vote is cast. In most cases, a single vote at the position determined by the estimated mean. This is a rather crude regression model that ignores the appearance of the patch. In this paper, we propose to learn a tensor regression model at each leaf node, and report results on three variations of tensor regression models. The first one, termed random forest based tensor regression employing only depth data (RF-TR-D), uses patches extracted from depth images. The second one, termed random forest based tensor regression employing only intensity data (RF-TR-I), uses patches extracted from greyscale images, and the third one, termed RF-TR-ID, fuses the information from RF-TR-D and RF-TR-I.

The early version of this work appears in [6]. Here we extended [6] by adapting a joint classification and regression scheme that classifies patches into head and not head ones by fusing the output of intensity and depth tensor regressors, and by performing experiments to additional datasets.

In summary, this paper contributes two novel approaches to the problem of real-time head pose estimation. Firstly, a new regression method based on random forests and tensor models is proposed. The proposed method instead of modelling the votes at each leaf node with a Gaussian, employs a tensor-based regression model. Secondly, we proposed fusion of depth and intensity data in the random forest regression framework.

The remainder of this paper is as follows. Related work is discussed in Section 2. In Section 3, we introduce the multimodal random forest based tensor regression method. In Section 4, we show how the proposed method is applied to the problem of head pose estimation. Results are given in Section 5 and conclusions are drawn in Section 6.

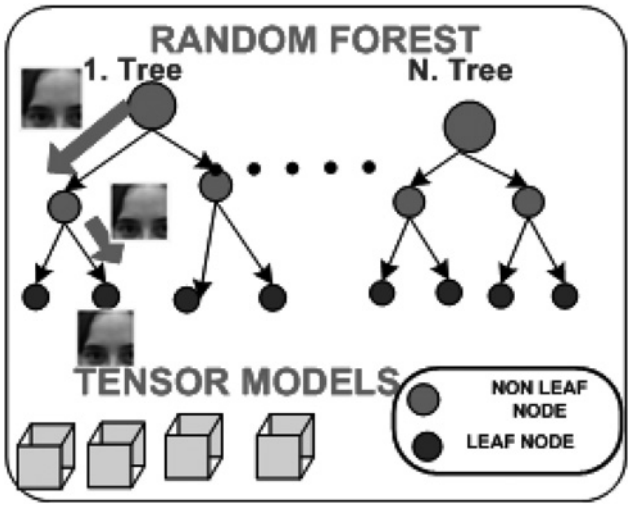

Fig. 1 *Multimodal random forest based tensor regression*

Fixed size patches are extracted from depth and intensity patches and passed through the forest. The patches which reach the leaf nodes are used as inputs to the tensor based regression models to obtain the estimates of the head angles

2 Related work

In this section, we review works in the area of head pose estimation, organising them into three main categories: those that employ two-dimensional (2D) images, those that employ 3D depth data, and those that combine 2D images and 3D depth data.

Head pose estimation approaches that employ 2D images can be further separated into two groups: 2D appearance based methods and 2D feature based methods. The 2D appearance based techniques analyse the entire head region. Osadchy *et al.* [7] proposed a convolutional neural network system which allowed the mapping of facial images to head pose parameters and achieved a near real-time performance (5 fps). Other methods use statistical modelling of facial regions, such as active appearance models (AAMs) [8], multi-view AAMs [9], 3D morphable models [10, 11] and constrained local models (CLM) [12]. Baltrusaitis *et al.* [13], extended the CLM method and proposed a method (CLM-Z) that allows the fusion of intensity and depth information in a single framework. The combination of CLM-Z with Generalised adaptive view-based appearance model (GAVAM) was proposed in [14]. The 2D feature based methods rely on the detection of facial features Vatahska *et al.* [15]. Whitehill *et al.* [16] proposed a method in which the orientation of a head was calculated using the locations of the tip of the nose and both eyes.

The second category of head pose estimation systems uses 3D depth data. The system proposed by Breitenstein *et al.* [17] is robust to large head pose variations, facial variations, and partial occlusions. In an offline process, 3D average face models are rendered in different head pose and stored at graphics processing unit (GPU) memory. Then, a new range images are captured and shape signatures for each pixel are calculated. The shape signature results in high curvature around nose tip pixel locations and contains information about the nose position and orientation. This leads to a set of head pose hypotheses. For each head pose hypotheses, the input range image was compared with each of the average 3D face models stored in the GPU memory by calculating matching cost. The model which leads to the smallest cost was then used for parameter estimation. This cost is the alignment cost between the reference and the input frame and is calculated using the difference in depth values between the corresponding frames. Fanelli *et al.* [18, 19] proposed a real-time head pose estimation technique using random regression forests. In [18], a large set of 3D synthetic faces is generated and used to train random forests for continuous head pose estimation. This technique is extended to a joint classification and regression scheme in [2]. This system allowed the extraction of patches from the upper body region of a person from depth data and only patches which belonged to the head region were used to estimate the head pose in real time. The estimation was performed using low-resolution data captured by a Microsoft Kinect Camera.

The third category involves the combination of 2D and 3D data. Seemann *et al.* [20] presented a head pose estimation system based on neural networks. Greyscale and depth data were used as inputs to the neural networks to calculate the head pose. Morency *et al.* [21] proposed to estimate head pose using intensity and depth data. Their method is based on intensity and depth view-based eigenspaces. These Eigenspaces are generated using different views of the same object. A prior model is reconstructed in order to represent views in this eigenspace. Then, pose variation changes are estimated between each of the frame in the prior model and the new frame. These difference values are used in Kalman filter update for tracking the pose parameters.

This paper builds upon works on random forests and tensor regressors. Random forests [22] are a collection of a certain type of decision trees that have proved very efficient in addressing both classification and regression problems. They are capable of dealing with large training data sets and are very easy to implement. Tensor based learning approaches, including tensor-based regressors, have certain advantages over vector-based approaches [23] – methods that arbitrarily vectorise data that is naturally organised as 2D arrays (e.g. images), 3D tensors (e.g. image sequences or colour images) or higher order tensors. Firstly, vectorisation leads to loss of the neighbourhood structure information. Secondly, since tensor-based methods reduce the dimensionality of the parameter space by performing the analysis along the tensor modes (two modes for a 2D image), they are less prone to over fitting when small training sets are used. In [23], tensor learning for regression was proposed. Two mapping functions were learned using Canonical (CANDECOMP)/Parallel factors (PARAFAC) decomposition, [24]. The square loss and e-sensitive loss functions were studied using Frobenius norm regularisation. The reported results showed that tensors resulted in improved accuracy in terms of angular error for head pose estimation.

3 Multimodal random forest based tensor regression

In this section we describe the proposed methods. Firstly, an RF-TR-D is described. Secondly, integration of intensity and depth at each leaf node of a random forest using tensor models (RF-TR-ID) is introduced.

A number of aligned grey scale images and depth data with head location and orientation parameters are used for the construction of a forest. An example of the aligned greyscale images and depth data is given in Fig. 2. For the construction of each tree, a subset of aligned intensity and

Fig. 2 *Aligned*

a Depth data
b Corresponding grey scale image
The background is suppressed on the greyscale image using a segmentation of the depth data
The bounding box shows the head/face region on both grey scale and depth data
During training, fixed size depth and grey scale patches belonging to head/face region are extracted within bounding box
Fixed size patches are also extracted from torso, arms and hair

depth data is selected and several fixed size patches belonging to head/face region are extracted from both the intensity image and the depth data. Fixed size patches are also extracted from the other image areas depicting the torso, the arms and the hair.

We assume that images are annotated in terms of the face bounding box. Given that we proceed in building a classification and regression forest. The classification of the patches in terms of whether they belong to the background or the face region is performed at the leaf nods of the forest. The ratio of positive (i.e. belonging to the face area) and negative (i.e. not belonging to the face area) patches that arrive at each leaf node during testing is calculated and stored. During testing, when a patch reaches the leaf note, it is classified as positive or negative, depending on whether this ratio is greater that one or not.

3.1 Tree Construction

Let us denote a random forest by $\boldsymbol{T}=\{T_t\}$, where a tree in the forest is denoted by T_t. Each tree, T_t, is built using a set of patches, $\{P_i\}$, which are randomly chosen from the training data. A patch is denoted by $P_i = (I_i^f, c_i, \theta_i)$ where I_i^f are the extracted features, c_i is a class label that reveals whether the patch belongs to the face/head region ($c_i = 1$) or not ($c_i = 0$) and $\theta = \{\theta_x, \theta_y, \theta_z, \theta_{\text{yaw}}, \theta_{\text{pitch}}, \theta_{\text{roll}}\}$ is a vector which contains the head pose parameters. The values θ_x, θ_y, θ_z are offsets between the patch centre and the head centre in 3D and θ_{yaw}, θ_{pitch}, θ_{roll} are the Euler angles of the head pose parameters.

The tree is constructed using the method proposed in [19, 22]. The parameters of a tree in each internal node are selected by generating a number of binary tests and by selecting the best test according to the optimisation criterion. In this work, we use stumps tests with parameters $t_{F_1, F_2, \tau}$. That is

$$\frac{1}{|F_1|}\sum_{q\epsilon F_1} I^f(q) - \frac{1}{|F_2|}\sum_{q\epsilon F_2} I^f(q) > \tau \tag{1}$$

where I^f is a feature channel and F_1 and F_2 are rectangular regions determined randomly within the depth patch. τ denotes the threshold. Similarly to [18, 25], our tests use the difference the differences between the average values of rectangular regions. The optimisation function is defined using both classification and regression measure. The function is optimised by randomly choosing between a measure based on the classification performance (face/not face) and a measure based on the regression performance (i.e. related to the estimation of the head pose) at each non-leaf node.

More specifically, the classification measure is defined as

$$U_C(P\backslash t^k) = \sum_{i\in\{L,R\}} w_i H(P\backslash t^k) \tag{2}$$

where

$$H(P\backslash t^k) = p(c|P)\ln(p(c|P)) \tag{3}$$

The set of patches at the parent node i is denoted by $P_{i\epsilon\{L,R\}}$ and the sets of patches at the child nodes are denoted by P_L and P_R. The ratio of patches is denoted by $w_i = |P_i|/|P|$.

The regression measure is defined as the differential entropy of the set of patches P at the internal node minus the weighted sum of differential entropies at the left child node P_L and the right child node P_R, which are defined after the slitting process. That is

$$U_R(P\backslash t^k) = H(P) - (w_L H(P_L) + w_R H(P_R)) \tag{4}$$

where $H(P)$ is the differential entropy of the set $P_{i\in L,R}$ and $w_{i=L,R}$ is the ratio of patches sent to each child node. The equation then becomes

$$U_R(P\backslash t^k) = \log\left(\overset{v}{\Sigma} + \overset{a}{\Sigma}\right) - \sum_{i=\{L,R\}} w_i \log\left(\overset{v}{\underset{i}{\Sigma}} + \overset{a}{\underset{i}{\Sigma}}\right) \tag{5}$$

where Σ^v and Σ^a are the covariance matrices of the offset vectors and rotation angles.

The splitting process stops and a leaf node is declared when the number of patches is below a threshold or when the tree reaches a predefined depth level.

In this paper, fixed sized patches (60×60) are extracted from the input data. Scale variation is not considered, since we assume that the subjects are approximately at the same distance to the camera. This is a valid assumption in the Biwi head pose dataset, [19] which we use for performance

doi: 10.1049/iet-cvi.2013.0320

evaluation. Some example images from this dataset can be seen in Fig. 3.

3.2 Tensor regression at the leaf nodes

Typically, in random regression forests, a multivariate Gaussian distribution models the distribution of the head pose parameters that corresponds to the patches that arrive at the leaf in question. The parameters of each Gaussian are then stored at each leaf node.

In this paper, we propose to use tensor regression modes [23] instead of a multivariate Gaussian distribution for regression. More specifically, higher rank linear support tensor regression models are trained at each leaf node. These models provide a mapping between patches and the corresponding head orientation parameters. A linear tensor regression model is defined as

$$y = f(\boldsymbol{\mathcal{X}}; \boldsymbol{\mathcal{W}}, b) = \langle \boldsymbol{\mathcal{X}}, \boldsymbol{\mathcal{W}} \rangle + b \tag{6}$$

where in our case $\boldsymbol{\mathcal{X}} \in \Re^{M \times N}$ denotes the feature channel patch tensor and $\boldsymbol{\mathcal{W}}$ is the weight tensor. M and N are the dimensions of the patches. The scalar b denotes the bias. As in [23] the weight tensor is constrained to be the sum of R rank-one tensors using the CANDECOMP/PARAFAC decomposition, that is

$$\begin{aligned} \boldsymbol{\mathcal{W}} &= \sum_{r=1}^{R} \vec{u}_r^{(1)} \circ \vec{u}_r^{(2)} \circ, \ \ldots, \ \circ \vec{u}_r^{(M)} \\ &\triangleq [[\, \vec{U}^{(1)}, \ \vec{U}^{(2)}, \ \ldots, \ \vec{U}^{(M)} \,]] \end{aligned} \tag{7}$$

where $\vec{U}^{(j)} = [\, \vec{u}_1^{(j)}, \ \ldots, \ \vec{u}_R^{(j)} \,]$. The CANDECOMP/PARAFAC decomposition of a two-way array can be seen in Fig. 4.

The model parameters ($\Theta = \{\boldsymbol{U}^{(1)}, \boldsymbol{U}^{(2)}, \ldots, (U)^M, b\}$) at each leaf node are learned by minimising the regularised empirical risk function. This function is minimised by using a set of labelled two-mode feature channel patch tensors $\{\boldsymbol{\mathcal{X}}_i, y_i\}_{i=1}^N$ and the associated pose angles, y_i. The risk function is given by

$$L(\Theta) = \frac{1}{2} \sum_{i=1}^{N} l(y_i, f(\boldsymbol{\mathcal{X}}_i; \Theta)) + \frac{\lambda}{2} \psi(\Theta) \tag{8}$$

where $l(\cdot)$ is the ϵ-insensitive loss function and $\psi(\cdot)$ is the Frobenius norm regularisation term. We select the rank R of the tensor by performing cross validation on the data at each leaf node.

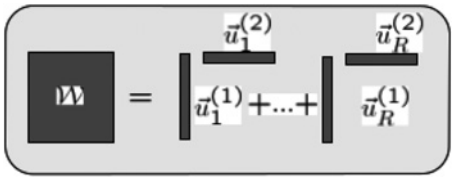

Fig. 4 *The CANDECOMP/PARAFAC decomposition of a two-way array*

At each leaf node of a forest two tensor regression models are trained. One using depth information, and the second using intensity information.

4 Head pose estimation

During testing, fixed size patches are densely extracted from the depth data and passed through the forest. Each extracted patch is directed via binary tests performed at the internal nodes towards leaf nodes.

When a test patch reaches a leaf node, the patch is classified according to whether it comes from the area depicting the head or other body parts. The classification is made by examining the ratio of head to non-head patches that reach the leaf in question during training ($p(c = 1/P)$). At leaves for which the ratio is greater than one, the trained regress or is applied, an estimate for the head pose parameters is obtained and a vote is cast in the corresponding Hough space. Leaf nodes with high variance ($\Sigma^v > 800$) do not cast votes. Then, the mean shift algorithm is applied in order to remove outliers.

For head centre estimation, an approach similar to that proposed in [19] is followed. At each leaf node a vote is cast at the mean of a Gaussian estimated at training time. First, the votes are grouped together. Then, a mean shift with a sphere radius equal to the average face model of [26] is applied in order to remove outliers. Then the votes are averaged to obtain the final head centre estimate.

5 Experimental results

In this section, we evaluate the proposed method by conducting experiments on the publicly available Biwi Kinect head pose database [19] and the ICT-3DHP [27]. We report the performance of the variants of the random forest tensor regression, i.e. a) using only depth data

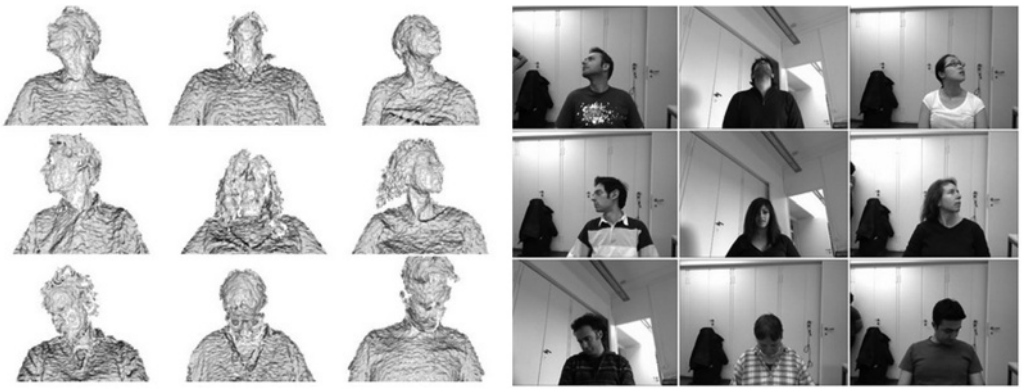

Fig. 3 *RGB images and depth data from the Biwi Kinect Head Pose Database, [19]*

(RF-TR-D) and b) by fusing depth and intensity data (RF-TR-ID). For comparison, we provide results obtained with a baseline random forest.

5.1 Biwi dataset

For the evaluation of the proposed methods we use two publicly available datasets, namely the Biwi Kinect [19] and the ICT-3DHP [27] datasets. The Biwi Kinect dataset contains both depth data and RGB images of the upper body region of 20 different people (14 men and 6 women) that turn their heads in different directions. The data was captured using a Kinect sensor. Twenty-four sequences were generated, since some people were recorded twice. All images are annotated with the location of the head centre and with the head rotation angles. The range of the rotation angles is approximately between ±75° for yaw, ±60° for pitch and ±50° for roll. The approximate rotation and translation errors of this dataset are reported as roughly 1 mm and 1 degree, respectively. Example images can be seen in Fig. 3.

5.2 ICT-3DHP dataset

The ICT-3DHP dataset also contains both depth data and RGB images of the upper body region of different people that turn their heads in different directions. The data were captured by a Kinect sensor. Ten sequences were generated. Each sequence contained approximately 1400 frames. All images are annotated with the location of the head centre and with the head rotation angles using Polhemus FASTRAK folck of birs tracker.

5.3 Parameter setting

For our evaluation, we partitioned the Biwi dataset and used 18 sequences of 18 subjects as training set and two sequences of two subjects as test set. Each forest contained seven trees. Each tree was generated using 3000 sample images. The values of the parameters used to train the random forest were set as follows: the sizes of the patches were set to 60×60, and the maximum size of sub patches to 30×30 pixels; the maximum tree depth was set to 15; the minimum number of patches that arrived at an internal node during training was set to 20 and the number of tests at each internal node was set to 10 000. The stride was set equal to 5 and the maximum variance to 500. The tensor model's training is performed for each leaf node after the random forest is constructed. Their performance depends on the rank (R) and the regularisation parameters (C) used. Their values were selected by cross validation.

Fig. 5 shows head pose parameter computation cost for an average frame in millisecond for the proposed methods, RF-TR-D and RF-TR-ID. Fig. 5*a* depicts the average run time as a function of the stride value. Fig. 5*b* depicts the average run time as a function of the number of trees when the stride value is 15. As can be seen in Fig. 5, the proposed method RF-TR-D processes 26 frames per second or higher when the stride value is set to 12 or higher. Similarly, the proposed method RF-TR-ID also processes 19 frames per second or higher when the stride value is set to 12 or higher. Although the proposed intensity and depth fusion at each leaf node (RF-TR-ID) results in a slightly higher computation cost than the proposed RF-TR-D method, both methods work in real-time.

For the evaluation of the proposed methods on the ICT-3DHP dataset, we used the forest, which was generated using the Biwi dataset was used. The parameter setting which was used during the forest training is described in Section 5.3.

5.4 Random forest with tensor models

In this section, we compare the performance of the proposed method with plain random forests. The results can be seen in Fig. 6. Figs. 6*a* and *b* reports the percentage of correctly estimated head poses on depth images for different thresholds for head centre localisation and angular error. In Fig. 6*c*, we report the mean angle error (MAE) against the percentage of leaves.

As can be seen in Fig. 6*b* the proposed intensity and depth fusion at each leaf node perform similarly for different thresholds. The proposed method, RF-TR-D is slightly more accurate than the RF-TR-ID. However, the proposed method RF-TR-ID resulted in lower MAEs for different angle thresholds (Fig. 6*c*).

Table 1 presents mean and standard errors calculated using two-fold cross validation. The mean and standard errors are given for RF-TR-D and RF-TR-ID. In Fig. 7 estimates of head pose parameters using the proposed methods can be seen on the subject's data. The cylinder shows the estimate head pose parameters.

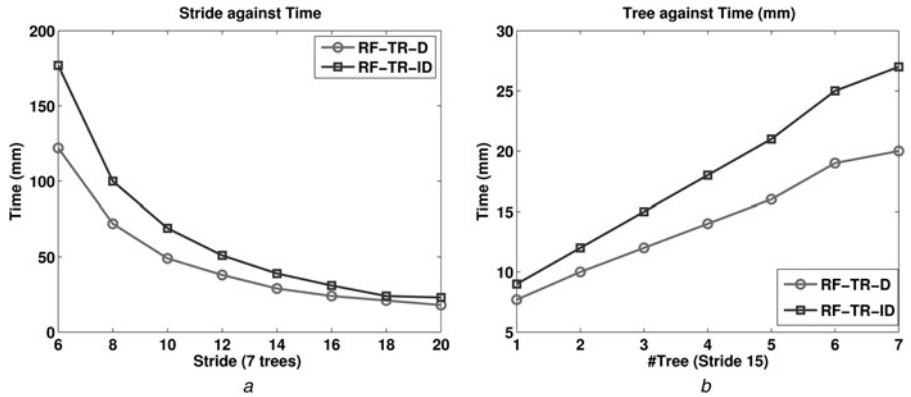

Fig. 5 *Head pose parameter computation time of a given depth and intensity data in milliseconds for the proposed methods, RF-TR-D and RF-TR-ID*

a Computation time using seven trees as a function of the stride parameter
b Computation time as a function of the number of the trees when the stride value is set to 15

doi: 10.1049/iet-cvi.2013.0320

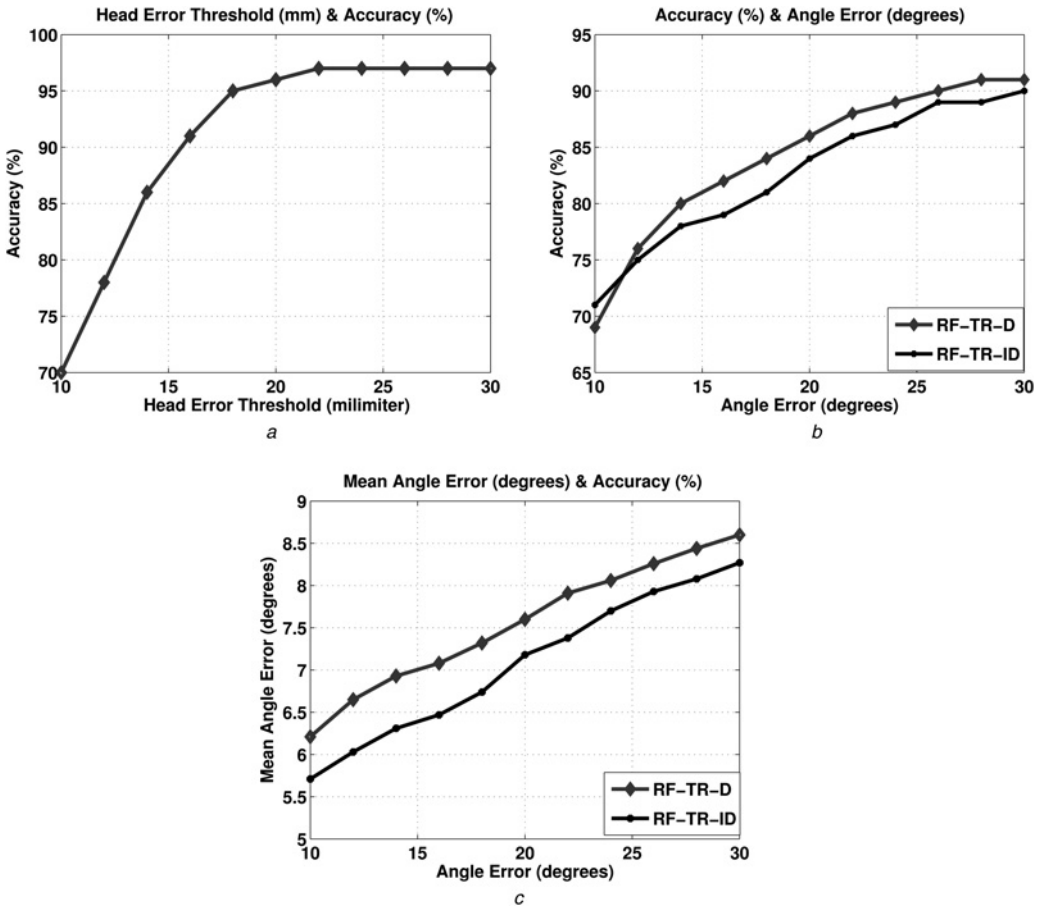

Fig. 6 *Performance of the proposed methods, RF-TR-D and RF-TR-ID*

a Accuracy of head centre estimation against different angle thresholds
b Accuracy of head orientation estimation against different angle thresholds
c Mean Angular Error against different angle thresholds

5.5 Comparison with other methods

The performance of the proposed methods was compared with the performance of the CLM [28], the CLM that employs both depth and intensity data (CLM-Z) [27] and the CLM-Z with a GAVAM [14]. Results are reported in Tables 2 and 3. In Table 2, the results for (RF-TR-D and RF-TR-ID) are obtained as the average values of two runs on the Biwi Dataset. In Table 3, the results for (RF-TR-D and RF-TR-ID) are obtained from the ICT-3DHP Dataset. The results of the other methods are reported in [27]. The random forest with Gaussian models is also implemented and the results are provided in the fourth column of Tables 2 and 3 for both Biwi and ICT-3DHP dataset, respectively. We can conclude that the proposed models perform better than the other methods.

As can be seen in Table 3 the GAVAM and CLM-Z with GAVAM methods clearly outperform the proposed methods. GAVAM is an integration of differential tracking and keyframe based approaches. This approach is strengthened by using keyframes to avoid drifting. The keyframes depict the face of a certain subject in different poses and scales. These keyframes are acquired and adapted online during testing time. As a tracking method, the performance of the CLM-Z method depends on the initialisation of the pose parameters. Once good head pose parameter estimates can be provided by GAVAM, accurate landmark positions could be obtained for the CLM-Z method. Then the tracking provides high precision during parameter estimation. In contrast, the proposed methods estimate the head pose at each frame independently. As a result, the tracking based methods result in accurate parameter estimation on the ICT-3DHP dataset which is not very challenging, since the subjects move their head slowly in front of a Kinect camera and all frames are captured and stored. However, as can be seen in Table 2, the performance of the proposed methods are higher than the tracking based approaches when the Biwi data is used. This is expected because this dataset is more challenging. There are some cases when the subjects move their head fast in front of a Kinect camera. There are also some cases when the frames are lost during the recording of this dataset. To conclude, fast motion and missing frames leaded tracking based algorithms to fail. Therefore the proposed method can be more useful in application where the available data is more challenging, such as that in the Biwi dataset.

Table 1 Mean and standard deviation of the head orientation error

Method	Pitch, °	Yaw, °	Roll, °	Stride
RF-TR-D	5.15 ± 0.59	7.8 ± 0.70	4.8 ± 0.48	5
RF-TR-ID	5.08 ± 0.46	7.83 ± 0.41	5.07 ± 0.19	5
RF-TR-I	4.67 ± 0.46	8.52 ± 0.41	5.57 ± 0.19	5
RF-TR-D	5.08 ± 0.14	8.2 ± 0.14	4.77 ± 0.10	10
RF-TR-ID	4.97 ± 0.01	8.17 ± 0.18	4.49 ± 0.28	10

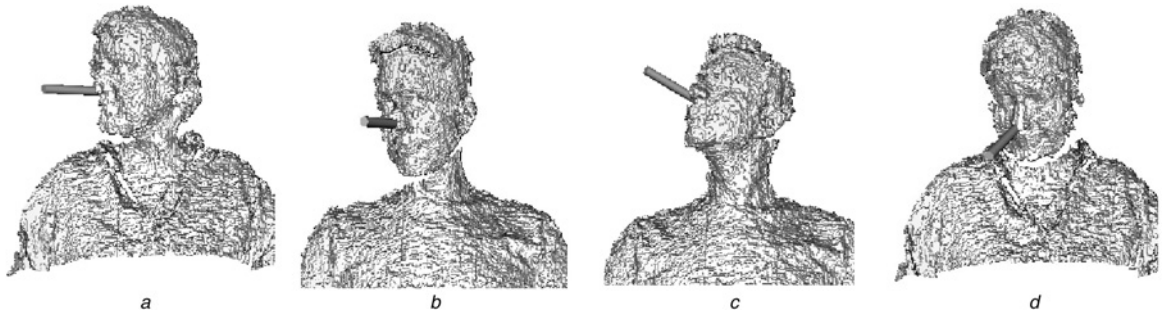

Fig. 7 *Several examples of head orientation estimation results on the depth data of two subjects from the Biwi Kinect Dataset*

Table 2 Comparison of head estimation results on the Biwi Kinect dataset

Method	Pitch	Yaw	Roll	Mean
proposed 1 RF-TR-D	5.15	7.8	4.8	5.91
proposed 2 RF-TR-ID	5.08	7.83	5.07	5.99
proposed 3 RF-TR-I	4.67	8.52	5.57	6.25
random forests	6.71	7.95	5.67	6.78
random forests [19]	8.5	9.2	8.0	8.6
CLM [28]	18.30	28.30	28.49	25.21
CLM-Z [27]	12.03	14.80	23.26	16.69
CLM with GAVAM[27]	5.10	6.29	11.29	7.56

We report the mean absolute error

Table 3 Comparison of head estimation results on the ICT-3DHP dataset

Method	Pitch	Yaw	Roll	Mean
proposed 1 RF-TR-D	5.85	8.53	7.64	7.34
proposed 2 RF-TR-ID	5.85	9.32	7.76	7.64
proposed 3 RF-TR-I	5.73	9.81	7.80	7.78
random forests	6.27	9.81	7.81	7.96
random forests [19]	9.40	7.17	7.53	8.03
GAVAM [14]	3.50	3.00	3.50	3.34
CLM [28]	9.92	11.10	7.30	9.44
CLM-Z [27]	7.06	6.90	10.48	8.15
CLM-Z with GAVAM [27]	3.14	2.90	3.17	3.07

We report the mean absolute error

6 Conclusion

In this paper, we have presented a novel framework for head pose estimation, called multimodal random forest based tensor regression. More precisely, we create random trees and extend them so that they include a tensor regressor at each leaf. In this way we combine the advantages of both methods, thus being able to process large sets of training data by generating strong predictions at each leaf node. We also study the effect that fusion of multiple sources of information has on the performance of a head pose estimation system. The efficacy of our method was demonstrated on the publicly available Biwi database. The experiments showed that our proposed framework outperforms typical random forests.

7 Acknowledgment

We would like to thank Gabriele Fanelli for helpful clarifications and Tadas Baltruaitis for useful code and clarifications.

8 References

1 Murphy-Chutorian, E., Trivedi, M.M.: 'Head pose estimation in computer vision: A Survey', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2009, **31**, pp. 607–626
2 Ma, B., Zhang, W., Shan, S., Chen, L., Gao, W.: 'Robust head pose estimation using LGBP'. Proc. IAPR Int. Conf. on Pattern Recognition, 2006, pp. 512–515
3 Raytchev, B., Yoda, I., Sakaue, K.: 'Head pose estimation by nonlinear manifold learning'. Proc. IAPR Int. Conf. on Pattern Recognition, 2004, pp. 462–466
4 Heinzmann, J., Zelinsky, A.: '3-D Facial pose and gaze point estimation using a robust real-time tracking paradigm'. Proc. IEEE Int. Conf. on Automatic Face and Gesture Recognition, 1998, pp. 142–147
5 Hu, Y., Chen, L., Zhou, Y., Zhang, H.: 'Estimating face pose by facial asymmetry and geometry'. Proc. IEEE Int. Conf. on Automatic Face and Gesture Recognition, 2004, pp. 651–656
6 Kaymak, S., Patras, I.: 'Exploiting depth and intensity information for head pose estimation with random forests and tensor models'. Proc. Asian Conf. on Computer Vision Workshops, 2012, pp. 160–170
7 Osadchy, M., Cun, Y.L., Miller, M.L.: 'Synergistic face detection and pose estimation with energy-based models'. Proc. Neural Information Processing Systems, 2005
8 Cootes, T.F., Edwards, G.J., Taylor, C.J.: 'Active appearance models', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2001, **23**, pp. 681–685
9 Ramnath, K., Koterba, S., Xiao, J., *et al.*: 'Multi-view AAM fitting and construction', *Int. J. Comput. Vis.*, 2008, **76**, (2), pp. 183–204
10 Blanz, V., Vetter, T.: 'A morphable model for the synthesis of 3D faces', *Comput. Graph. Inter. Tech.*, 1999, pp. 187–194
11 Storer, M., Urschler, M., Bischof, H.: '3D-MAM: 3D morphable appearance model for efficient fine head pose estimation from still images'. Proc. IEEE Int. Conf. Computer Vision Workshops, 2009, pp. 192–199
12 Cristinacce, D., Cootes, T.: 'Feature detection and tracking with constrained local models'. Proc. British Machine Vision Conf., 2006, pp. 929–938
13 Baltrusaitis, T., Robinson, P., Morency, L.: '3D Constrained local model for rigid and non-rigid facial tracking'. Proc. IEEE Conf. on Computer Vision and Pattern Recognition, 2012, pp. 2610–2617
14 Morency, L., Whitehill, J., Movellan, J.: 'Generalized adaptive view-based appearance model: integrated framework for monocular head pose estimation'. Proc. IEEE Int. Conf. Automatic Face and Gesture Recognition, 2008, pp. 1–8
15 Vatahska, T., Bennewitz, M., Behnke, S.: 'Feature-based head pose estimation from images'. Proc. Conf. on Humanoid Robots, 2007, pp. 330–335
16 Whitehill, J., Movellan, J.R.: 'A discriminative approach to frame-by-frame head pose tracking'. Proc. IEEE Int. Conf. on Automatic Face and Gesture Recognition, 2008, pp. 1–7
17 Breitenstein, M.D., Kuettel, D., Weise, T., Gool, V.L., Pfister, H.: 'Real-time face pose estimation from single range images'. Proc. IEEE Conf. on Computer Vision and Pattern Recognition, 2008
18 Fanelli, G., Gall, J., Gool, L.V.: 'Real time head pose estimation with random regression forests'. Proc. IEEE Conf. on Computer Vision and Pattern Recognition, 2011, pp. 617–624
19 Fanelli, G., Weise, T., Gall, J., Gool, L.V.: 'Real time head pose estimation from consumer depth cameras', Symp. German Assoc. Pattern Recognit., 2011, pp. 101–110
20 Seemann, E., Nickel, K., Stiefelhagen, R.: 'Head pose estimation using stereo vision for human-robot interaction'. Proc. IEEE Int. Conf. on Automatic Face and Gesture Recognition, 2004, pp. 626–631

doi: 10.1049/iet-cvi.2013.0320

21 Morency, L.P., Sundberg, P., Darrell, T.: 'Pose estimation using 3D view-based eigenspaces'. Proc. IEEE Int. Conf. on Analysis and Modeling of Faces and Gestures Workshops, 2003, pp. 45–52
22 Breiman, L.: 'Random Forests', *Mach. Learn.*, 2001, **45**, (1), pp. 5–32
23 Guo, W., Kotsia, I., Patras, I.: 'Tensor learning for regression', *IEEE Trans. Image Process.*, 2012, **21**, (2), pp. 816–827
24 Kolda, T.G., Bader, B.W.: 'Tensor decompositions and applications', *SIAM Rev.*, 2009, **51**, (3), pp. 455–500
25 Criminisi, A., Robertson, D., Konukoglu, E., *et al.*: 'Regression forests for efficient anatomy detection and localization in computed tomography scans', *Med. Image Anal.*, 2013, **17**, (8), pp. 1293–1303
26 Paysan, P., Knothe, R., Amberg, B., Romdhani, S., Vetter, T.: 'A 3D face model for pose and illumination invariant face recognition'. Proc. IEEE Int. Conf. on Advanced Video and Signal Based Surveillance, 2009, pp. 296–301
27 Baltrusaitis, T., Robinson, P., Morency, L.: '3D constrained local model for rigid and non-rigid facial tracking'. Proc. IEEE Conf. Computer Vision and Pattern Recognition, 2012
28 Saragih, J.M., Lucey, S., Cohn, J.F.: 'Deformable model fitting by regularized landmark mean-shift', *Int. J. Comput. Vis.*, 2011, **91**, (2), pp. 200–215