Isomorphic Loss Function for Head Pose Estimation

Iulian Felea, Corneliu Florea, Constantin Vertan, Laura Florea
Image Processing and Analysis Laboratory,
University Politehnica of Bucharest
Spaliul Independentei 313, Bucharest, Romania
{iulian.felea; corneliu.florea; laura.florea; constantin.vertan}@upb.ro

ABSTRACT

Accurate head pose estimation is a key step in many practical applications involving face analysis tasks, such as emotion recognition. We address the problem of head pose estimation in still color images acquired with a standard camera with limited resolution details. To achieve the proposed goal, we make use of the recent advances of Deep Convolutional Neural Networks. As head angles with respect on yaw and pitch are continuous, the problem is one of regression. Typical loss function for regression are based on L_1 and L_2 distances which are notorious for susceptibility to outliers. To address this aspect we introduce an isomorphic transformation which maps the initially infinite space into a closed space compressed at the ends and thus significantly down–weighting the significance of outliers. We have thoroughly evaluated the proposed approach on multiple publicly head pose databases.

Keywords

Head Pose Estimation; Loss function; L_1 distance; Isomorphic transformation.

1 INTRODUCTION

The head pose estimation is the process of inferring the orientation of a human head from digital images. Robust and efficient algorithms for head pose estimation are necessary in a plethora of applications such as human-computer interfaces (where the orientation and head trajectory may be encoding a message by itself, as is the case of nodding), emotion where the facial expression is complemented by head pose (such as for stress) or face recognition analysis. The problem is made more difficult by the variation of the illumination angle and intensity, the variability of the human faces and those of the environment. Another aspect which is lacking somehow is the existence of annotation for large, "in–the–wild" databases.

In computer vision, the estimation of the head pose refers to inference of the orientation of a person's head, relative to the view of a camera. Following the seminal review on the topic by Murphy-Chutorian and Trivedi [MCT09], a more rigorous definition of the head pose estimation is the "ability to infer the orientation of a head relative to a global coordinate system, but this subtle difference requires knowledge of the intrinsic camera parameters to undo the perceptual bias from the perspective distortion". Typically, the head pose is modelled in a three-dimensional space by three Euler angles of rotation around three axis orthogonal to each other: the yaw, the pitch and the roll, as showed in figure 1.

To give an anatomical background let us recall the work of Ferrario et al. [FSS+02] who measure the range of head motion for an average adult male. The following values have been determined: the sagittal flexion and extension (i.e., forward to backward movement of the neck) from -60.4^0 to $+69.6^0$, a frontal lateral bending (i.e., right to left bending of the neck) from -40.9^0 to $+36.3^0$ and a horizontal axial rotation (i.e., right to left rotation of the head) from -79.8^0 to $+75.3^0$.

Due to its importance, many proposals for the head pose problem exists [MCT09]. Yet, the strong majority of methods approach the problem using as input sequences of frames (videos), or by 3D data (i.e. including depth information).

Using only standard image sequence, without depth information, with tracking (and thus working on the frame correlation to diminish the error), one should note the work of Asteriadis et al [ASKK09] which use Distance Vector Fields to locate face features and follow by head pose regression; the same authors continued, [AKK14], by showing that building models oriented per person's, better performance is at hand. Saragih et al. [SLC11] used a constrained local model, with parameter correlated over consecutive frames and improved convergence of the model by mean shift. Valenti et al. [VSG12] showed that combining independently extracted head pose information and eye pupil

(a) (b) (c)

Figure 1: Illustration of the three head rotations: yaw (a - image from Columbia database), the pitch (b) and roll with pitch (as images from HPEG database) .

localization data, can boost the performance in either problem.

In the case of independently considered frames, we note the method proposed by Kumano et al. [KOY+09] who used a variable-intensity template in a Bayesian framework for a combined head pose and face expression analysis. Vincente et al. [VHX+15] Jeni and Cohn [JC16] used various face fiducial point locator to infer the 3D head pose as a preamble for gaze estimation. Preda et al. [PFSF15], relied on HoG and Local Binary Pattern (LBP) to describe input data and on Multi-Layer-Perceptron for regression.

Lately, following the winning of the ILSVRC competition by the Alexnet architecture [KSH12], the dominant solution in various sub–aspects of image classification became the Deep Convolutional Neural Networks (DCNN). Recently, the residual network architecture (ResNet) [HZRS16] have gained the upper hand. The DCNN have been used, previously, in head pose estimation too. For instance, in [MR15] a classifier corrects the indication of a regressor for gathering head pose data as input for human–robot interaction. Recently Venturelli et al. [VBVC17] used RGB-D data from and RGB-Kinect camera to regress, by means of DCNN, the head angles.

Our approach assumes that the head orientation is given in conjunction with faces detected by a specifically trained detector. In the proposed solution the DCNN based detector proposed by Zhang et al. [ZZLQ16] is used. Furthermore, as the authors have released a second version of the algorithm, named Multi-task Cascaded Convolutional Networks -2 (MTCCN-2), which showed improved accuracy, we have adopted too.

The remainder of this paper is organized as follows: Section 2 presents the Deep Neural Network architecture and the adaptation of the loss function proposed; Section 3 summarizes the main features of the databases, while Section 4 presents the achieved results. The paper ends with discussions and proposes further developments.

2 METHOD

2.1 Architecture and Regression Loss

As mentioned in the introductory section, we have relied on DCNN for the actual task of head pose estimation. We have experimented with two standard architectures, namely AlexNet [KSH12] and ResNet50 (Residual network with 50 layers as described in [HZRS16]). In all cases we have used the fine-tuning paradigm: both networks have been previously trained on ImageNet and the found weights initialize the training here. Since we aim to predict head-pose for yaw and pitch, the output dimensions of the last fully connected layer, initially set at 1000 (number of classes in ImageNet), was resized at only 2: one predicts the horizontal angle and the other the vertical angle of the head. Finally, we replaced the last Softmax layer with a custom regression loss layer based on the L_1 distance between ground truth information and the predicted angles.

The loss function is in the simplest case the mean absolute angular error of the head pose estimation for the forward pass:

$$\begin{aligned} \mathscr{L}^{(1)}_{MAE} &= \frac{1}{2N}\sum_{i=1}^{N}\left(|x_i^h - y_i^h| + |x_i^v - y_i^v|\right) \\ &= \frac{1}{2N}\sum_{i=1}^{N}\left(L_1(x_i^h; y_i^h) + L_1(x_i^v; y_i^v)\right) \end{aligned} \tag{1}$$

where N denotes the number of images used by the neural network at training, x_i - the ground truth value (the true horizontal-h/vertical-v angle of the head) and y_i is the estimated angle. $L_1(a,b)$ is the absolute error between a and b, $|a-b|$. The backward pass assumes the derivatives of the $\mathscr{L}$ distance with respect to the given axis.

$$\begin{aligned} \frac{d\mathscr{L}}{dx} &= \sum_{i=1}^{N}\frac{dL_1}{dx_i^h} + \frac{dL_1}{dx_i^v}; \\ \frac{dL_1}{dx_i^k} &= \frac{x_i^k - y_i^k}{|x_i^k - y_i^k|} = sgn(x_i^k - y_i^k). \end{aligned} \tag{2}$$

2.2 Linear label normalization

In regression problems, the training procedure usually optimizes a low value Minkovski distance (typically L_1 or L_2). The loss function is often formed

by adding a regularization term, where the goal is the distance between the estimated values of the network and the ground-truth. However, it is generally known that L_1 and L_2 norm minimization is sensitive to outliers, which can result in poor generalization depending on the amount of outliers present during training [Hub11]. We recall that outliers are rare samples that sometimes appear in the training data, generated, for instance, by facial makeup, rare illumination conditions etc. The outliers can be associated with samples with noisy ground-truth annotation.

In the presence of outliers, the main issue of using L_1 loss in regression problems is that outliers can have a disproportionately high weight and thus, will influence the training procedure by reducing the generalization ability. Previously, the problems was approached by using more elaborated loss functions such as based on Tukey biweight function [BRCN15].

An intuitive solution is to transform the initial set of labels into a more compact one. Under such auspices, eq. (1) becomes:

$$\mathscr{L}^{(2)} = \frac{1}{2N}\sum_{i=1}^{N}\Big(F(L_1(x_i^h;y_i^h)) + F(L_1(x_i^v;y_i^v))\Big) \quad (3)$$

where F is a 1D mapping function that aims to intrinsically reduce the weight of outliers.

Assuming that the given labels range into $[Y_{min};Y_{max}]$ a simple implementation of F is linear mapping into [-1,1]:

$$F(y) = 2\frac{y - Y_{min}}{Y_{max} - Y_{min}} - 1 \quad (4)$$

2.3 Isomorphic mapping

Although simple and intuitive, eq. (4) does not redistribute the weight of the outliers with respect to the weight of normal data. Its positive effect, when exists, can only be attributed to mapping the labels into a domain that is more accessible to a neural networks.

A truly positive mapping would re-weight outliers, which should be in minority with respect to normal data. Assuming that initial data is given into the real number space, and errors close to infinity may exist due network producing completely wrong predictions, one wishes to map them into a closed algebraic space, where predictions have an upper bound. Such a mapping is:

$$F(y) = \frac{y}{1+|y|} \quad (5)$$

given that the input label, $y = y_i$ is in a symmetrical range (i.e. $-Y_{min} = Y_{max}$). The mapping compresses the higher domain, thus under–weighting large errors, which should be in minority when compared to smaller ones that form the majority.

We note that the function defined by eq. (5) is the generative function of the symmetrized version of the logarithmic-like image processing model [VOFF08, NDC13]. In that application the function acts as an isomorphism between $(-1,1)$ range and the image definition range $[0;D_{max} = 255]$. As it is an isomorphism, it also preserves the topology of the initial space [Opp67]. In other words, it means:

$$F(L_1(x;y)) = L_1(F(x);F(y)) \quad (6)$$

The practical advantage is that instead of replacing eq. (5) into eq. (3) and obtaining a more complicate loss function (which implies also more complicated derivatives for the backward pass), one can simply apply the function to all labels and follow by training and testing into $(-1;1)$ space; next one will simply transform the prediction by the inverse of the isomorphic mapping, that is:

$$F^{-1}(x) = \frac{x}{x-1} \quad (7)$$

3 DATABASES

For our experiments we used Boston University Head Tracking [lSA00], Columbia Gaze [SYFN13], Head Pose and Eye Gaze (HPEG) [ASKK09], INRIA [GL04] and UPNA [ABVC16] datasets for head pose estimation. A set of illustrative images from the databases are in Figure 2.

Boston University Head Tracking Dataset. The Boston database [lSA00] contains short video sequences at a resolution of 320×240 pixels with uniform and varying illuminations. Each type of illumination videos was recorded with different subjects. For example, five participants were recorded in 45 videos with normal illumination and only three subjects recorded 27 videos with varying illuminations. To achieve head pose and orientation ground truth information, the subjects wear a magnetic tracker that collected head movement. Each one of them was instructed to perform various head movements, such as rotations, translations.

Columbia Gaze Dataset. The Columbia database [SYFN13] contains 5880 de images of 64 persons with 21 gaze directions and 5 head positions from ethnically diverse subjects. The subjects ranged from 18 to 36 years of age and 21 of them wore prescription glasses. The main purpose of this database is estimation of the point of gaze and thus the images have high-resolution: 5184×3456 pixels. Each of the 64 de persons was recorded with yaw head angle of 0, 15^0 and 30^0 left

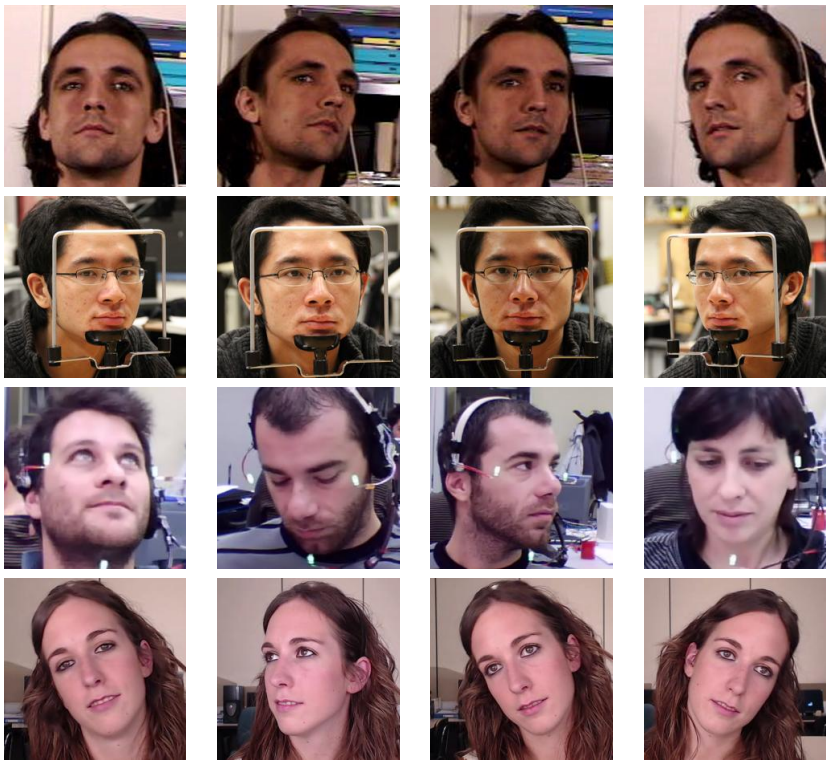

Figure 2: Various head poses from the used databases. Rows from top to bottom: Boston, Columbia, HPEG, UPNA. Note the diversity inside and between databases.

and right. The images were taken in a laboratory and various head angles were obtained by moving the recoding camera, while the subjects stood still.

Head Pose and Eye Gaze (HPEG) Dataset. The HPEG database [ASKK09] is given as videos of 640 × 480 resolution. It is separated in two section differing by the subject to camera distance. It contains data from 10 persons and we extracted all the frames from the given sequences. The head position includes yaw and pitch variations from -30^0 to $+30^0$. The database comes with annotation on head angle obtained from a magnetic sensor placed on top of the subject head.

INRIA Head Pose Dataset The INRIA (also named Pointing04) database [GL04] consists of 15 sets of images acquired with different people. Subjects have between 20 to 40 years old, some have facial hair and seven are wearing glasses. Head pose variation includes yaw and pitch and ranges between -90 and +90 degrees, with a step of 15 degrees for yaw, and 30 and 15 for pitch. Negative values for pitch correspond to bottom poses and positive values correspond to top poses.

UPNA Dataset. The UPNA database [ABVC16] contains a set of 120 videos, acquired from 10 different subjects (6 males and 4 females) of 12 videos each. Every set of 12 videos is composed of six guided-movement sequences and six free-movement sequences. In the guided sequences, the user follows a specific pattern of movement. In the free sequences, the user moves the head at will. In order to provide the database with more uniformity, it has been considered convenient that every video begins and ends with the head in a frontal position, at a working distance from the camera (55–60 cm). Movement ranges include translations going up to more than 200 mm in any axis from the starting point, and rotations up to 30 degrees.

4 IMPLEMENTATION AND RESULTS

4.1 Implementation details

The various alternatives described in this paper were implemented in Matlab. The DCNN architecture support is based on MatConvNet library [VL15].

We ran various tests scenarios. We started with a pretrained AlexNet on ImageNet which was altered as described in section 2.1; for both training and testing there were used Boston, Columbia, HPEG and INRIA databases. The training contains 70% while testing 30%, randomly selected from each of the databases. The results are: $MAE_{Pitch} = 1.00$ and $MAE_{Yaw} = 1.27$. The performance is extremely accurate thus, given the size of the DCNN and the correlation between individual frames, the network memorized the databases.

Given the later observation the train set is formed by joining all images from Columbia, HPEG and INRIA

Table 1: Head pose estimation on Boston dataset when various loss functions and normalization scenarios are envisaged in conjunction with ResNet-50.

Architect.	Normaliz.	Pitch	Yaw
AlexNet	None	4.44	9.53
ResNet-50	None	3.65	9.02
ResNet-50	DB-EQ	2.60	6.32
ResNet-50	DB-EQ + LIN	2.53	5.11
ResNet-50	DB-EQ + ISO	2.32	4.61

while Boston forms the test. In this case the performance dropped to $MAE_{Pitch} = 4.44$ and $MAE_{Yaw} = 9.02$. It shows clearly that generalization is poor over database as there is insufficient variation in data.

Database equalization

At this step we added UPNA database in the training set and still leave Boston completely only in the testing. The problem in this scenario is that UPNA has considerable more images and during training it dominates and the network learns it. Consequently, we balance the contribution of each dataset, by randomly selecting only 6000 images from UPNA.

4.2 Results

The various results, can be seen in Table 1. We tested with both mentioned architectures: AlexNet and ResNet-50. We have experimented with various types of normalization and loss function as follows: no normalization (marked as "None"), database equalization ("DB-EQ") by dropping the contribution of UPNA, linear loss normalization ("LIN") and isomorphic loss normalization marked by ("ISO").

Regarding the results, the first note is that ResNet-50 architecture dominates AlexNet. Database equalization helps, as the system does no learn UPNA any longer. Also the isomorphic loss function shows better performance than linear as, indeed, large errors are in minority.

4.3 Comparison with State of the Art

Multiple methods have reported mean absolute accuracy on the Boston database. None (us included) have trained on Boston but only tested. The comparative results may be followed in table 2. Also on our case, due to the fact that the inverse of of the isomorphism, at the end is susceptible to augment values, sometimes we get prediction far outside the given range. We have bounded these values to range of possible head angles.

The results showed that we obtain the smallest error for the pitch angle variation and the second average error, in the condition that we do not take into consideration any temporal correlation.

Table 2: Comparative performance on the Boston database. Please note that all other methods do head tracking thus reducing the error by enforcing correlation among consecutive frames. In contrast, we report prediction on independent frames, without temporal correlation. With bold letter we have marked the best result for each category.

Method	Pitch	Yaw	Mean
La Cascia et al. [lSA00]	6.1	**3.3**	4.7
Asteriadis et al. [ASKK09]	3.82	4.56	4.7
Kumano et al. [KOY+09]	4.2	7.1	11.3
Valenti et al. [VSG12]	5.26	6.10	5.68
Saragih et al. [SLC11]	4.5	5.2	4.85
Vincente et al. [VHX+15]	6.2	4.3	5.25
Jeni&Cohn [JC16]	2.66	3.93	**3.3**
Proposed	**2.32**	4.61	3.46

5 CONCLUSIONS AND FUTURE WORK

In this paper, we proposed a simple method to alter the regression loss so to impose more uniform error so to obtain an accurate head pose angle on individual frames. The algorithm consist in based on Deep Convolutional Neural Networks for both face detection where we have relied on the MTCCN-2 solution and for head pose estimation.

Further developments may envisage domain adaptation between database to be able to relevantly train on various databases and transferred the learned knowledge on the test one. Also the process of adjusting the loss function may assume further optimization with respect to a given error.

6 ACKNOWLEDGMENTS

The authors would like to thank NVidia Corporation for donating the Tesla K40c GPU that helped us run experimental setup for this research. The work is partially financed by University Politehnica of Bucharest,through the "Excellence Research Grants" Program, UPB – GEX. Identifier: UPB–EXCELENTA–2016, Contract no. 95/26.09.2016 (aFAST).

7 REFERENCES

[ABVC16] M. Ariz, J. Bengoechea, A. Villanueva, and R. Cabeza, *A novel 2d/3d database with automatic face annotation for head tracking and pose estimation*, CVIU **148** (2016), 201–210.

[AKK14] S Asteriadis, K Karpouzis, and S Kollias, *Visual focus of attention in non-calibrated environments using gaze estimation*, IJCV **107(3)** (2014), 293–316.

[ASKK09] S. Asteriadis, D. Soufleros, K. Karpouzis, and S. Kollias, *A natural head pose and eye gaze dataset*, ACM Workshop on Affective Interaction in Natural Environments, 2009, pp. 1–4.

[BRCN15] V. Belagiannis, C. Rupprecht, G. Carneiro, and N. Navab, *Robust optimization for deep regression*, ICCV, 2015, pp. 2830–2838.

[FSS+02] V. F. Ferrario, C. Sforza, G. Serrao, G. Grassi, and E. Mossi, *Active range of motion of the head and cervical spine: a three-dimensional investigation in healthy young adults*, J. Orthopaedic Research **20(1)** (2002), 122–129.

[GL04] N. Gourier and J. Letessier, *The pointing 04 data sets*, ICPR, Visual Observation of Deictic Gestures, 2004.

[Hub11] P. Huber, *Robust statistics*, Springer, 2011.

[HZRS16] K. He, X. Zhang, S. Ren, and J. Sun, *Deep residual learning for image recognition*, CVPR, 2016.

[JC16] L. A. Jeni and J. F. Cohn, *Person-independent 3d gaze estimation using face frontalization*, CVPR Workshops, 2016, pp. 792–800.

[KOY+09] S. Kumano, K. Otsuka, J. Yamato, E. Maeda, and Y. Sato, *Pose-invariant facial expression recognition using variable-intensity templates*, IJCV **83** (2009), no. 2, 178–194.

[KSH12] A. Krizhevsky, I. Sutskever, and G. Hinton, *Imagenet classification with deep convolutional neural networks*, NIPS (F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, eds.), 2012, pp. 1097–1105.

[lSA00] M. laCascia, S. Sclaroff, and V. Athitsos, *Fast, reliable head tracking under varying illumination: An approach based on registration of texture-mapped 3d models*, IEEE T. PAMI **22(4)** (2000), 322–336.

[MCT09] E. Murphy-Chutorian and M. Trivedi, *Head pose estimation in computer vision: A survey*, IEEE T. PAMI **31(4)** (2009), 607–626.

[MR15] S. Mukherjee and N. Robertson, *Deep head pose: Gaze-direction estimation in multimodal video*, IEEE Trans. on Multimedia **17** (2015), no. 11, 2094–2107.

[NDC13] Laurent Navarro, Guang Deng, and Guy Courbebaisse, *The symmetric logarithmic image processing model*, Digital Signal Processing **23** (2013), no. 5, 1337–1343.

[Opp67] A. V. Oppenheim, *Generalized superposition*, Information and Control **11** (1967), no. 5,6, 528 – 536.

[PFSF15] V. Preda, C. Florea, A. Sima, and L. Florea, *Simple head pose estimation in still images*, ISSCS), 2015, pp. 1–4.

[SLC11] J. Saragih, S. Lucey, and J. Cohn, *Deformable model fitting by regularized landmark mean-shift*, IJCV **91** (2011), no. 2, 200–215.

[SYFN13] B.A. Smith, Q. Yin, S.K. Feiner, and S.K. Nayar, *Gaze locking: Passive eye contact detection for human–object interaction*, ACM Symposium on User Interface Software and Technology, 2013, pp. 271 – 280.

[VBVC17] M. Venturelli, G. Borghi, R. Vezzani, and R. Cucchiara, *Deep Head Pose Estimation from Depth Data for In-car Automotive Applications*, ArXiv e-prints **1703.01883** (2017).

[VHX+15] F. Vicente, Z. Huang, X. Xiong, F. De la Torre, W. Zhang, and D. Levi, *Driver gaze tracking and eyes off the road detection system*, IEEE Trans. on Intelligent Transportation Systems **16** (2015), no. 4, 2014–2027.

[VL15] A. Vedaldi and K. Lenc, *Matconvnet – convolutional neural networks for matlab*, ACM-MM, 2015.

[VOFF08] C. Vertan, A. Oprea, C. Florea, and L. Florea, *A pseudo-logarithmic framework for edge detection*, Advances in Computer Vision (J. Blanc-Talon et al, ed.), LNCS, vol. 5259, Springer Verlag, 2008, pp. 637 – 644.

[VSG12] R. Valenti, N. Sebe, and T. Gevers, *Combining head pose and eye location information for gaze estimation*, IEEE Trans. on Image Processing **21** (2012), no. 2, 802–815.

[ZZLQ16] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, *Joint face detection and alignment using multitask cascaded convolutional networks*, IEEE Signal Processing Letters **23** (2016), no. 10, 1499–1503.