Improvements on head pose estimation algorithm for state classification

Renato Ramos da Silva and Roseli A. Francelin Romero
Department of Computer Science
Institute of Mathematical and Computer Sciences (ICMC)
University of Sao Paulo
Sao Carlos, Sao Paulo, Brazil 13566–590
Email: ramos,rafrance@icmc.usp.br

***Abstract*—An important aspect for a robot to interact with humans is to detect a person in the environment with which it is interacting and to estimate her/him head pose. These aspects made possible to a robot know if it will alone and execute some work or interact socially with a person doing a job or by fun. There are several works of head pose estimation in the literature, but we focus on work that used local histogram of oriented gradient and support vector regression to solve this problem. In this paper, we are proposing an adaptation of this method proposed using only histogram of oriented gradient in order to classify the head pose estimation and the presence of a person in the environment. Then, the new method has been evaluated in a offline and online way. The results showed that the proposed approach has great capability to classify head pose and identify when a person is present in the environment.**

I. Introduction

Human beings are capable to detect the richness of the environment and use it in their favor. The eyes are responsible for acquiring the information which will be evaluated by the human to decide what is important for him/her in a given moment. Some events are simple and processed without learning process, such as, identify when are interacting with other person. But, in other cases, for a more complex event, such as, shared attention, the vision system is an important part of the process of learning it. The vision system can help in the process of identification of an object or the head pose estimation to determine the place of partner‘s focu with which are interacting.

Head pose estimation is an important ability of human beings commonly use to detect a type of non-verbal communication. It can be defined as a process of inferring the orientation of a human head from digital imagery. It is part of a mechanism that helps a person to understand other intentions, confirm ideas or some dangers. Then, it should be considered as an important aspect when the goal is to create an humanoid robot capable to interact socially [13].

In the literature, the area concerned on the development system that emulates a human vision system is known by computer vision. In the context of computer vision, head pose estimation is most commonly interpreted as the ability to infer the orientation of a person’s head relative to the view of sensors [13]. Generally, it has been used as sensors camera(s) or by combining camera(s) with infrared.

In their survey, Chutorian et. all [13] divided in eight categories presented to follow each approach and spans 90 of the most innovative and characteristic papers that have been published on this topic. **Appearance template methods** , in which, a new image of a head is compared to a set of exemplars (each labeled with a discrete pose) in order to find the most similar view. **Detector array methods** that consist in to train a series of head detectors each attuned to a specific pose and to assign a discrete pose to the detector with the greatest support. **Nonlinear regression methods** that make use of nonlinear regression tools to develop a functional mapping from the image or feature data to a head pose measurement. **Manifold embedding methods** that consist in to seek low-dimensional manifolds that model the continuous variation in head pose. New images can be embedded into these manifolds and then used for embedded template matching or regression. **Flexible models** that fit a nonrigid model to the facial structure of each individual in the image plane. Head pose is estimated from feature-level comparisons or from the instantiation of the model parameters. **Geometric methods** that use the location of features such as eyes, mouth, and nose tip to determine pose from their relative configuration. **Tracking methods** recover the global pose change of the head from the observed movement among video frames. **Hybrid methods** , where our method is inserted, that combine one or more of these aforementioned methods to overcome the limitations inherent in any approach.

In order to select an algorithm of one category presented above, the application’s limitations must be known. In our case, the main idea is to use this vision system on NAO robot, which it has support to the Opencv library, as an extension of this work. Then, the estimation needs to be fast, the model needs to recover when a person is not detected and the structure does not GPU programing.

In this paper, we are proposing an adaptation of [12] to classify the head pose estimation and the presence of a person in the environment. For this, we are combining the following methods: cascaded-Adaboost face detection, continuously adaptive mean shift (Camshift) tracking, some image treatment, histogram oriented gradient and support vector classification, to composed our vision system to be inserted in our architecture to control a robotic head. The difference from this propose to the original by adding the Camshift and using classification, where Chutorian use regression.

This article is organized as it follows. In section II, some related works are presented about face detection, tracking, head pose estimation approaches proposed so now. After, in section

III, each part of the computer vision system proposed in this paper is described. Afterward, in Section VI, the experimental results are presented, from a set of experiments carried out to evaluate the performance of the proposed system using offline and online data. Finally, in Section VII, conclusions and future works are presented.

II. Related Work

Face detection is the ability to find the locations and sizes of human faces in arbitrary images. The boosting-based detector, proposed by Viola and Jones [16], using Haar-like features, is most used by the real-time performance [6]. Others works include neural networks [5], [15], the use of three-dimensional properties in the images with curvature analysis [4] and adaboost algorithm based on MB-LBP features with skin color segmentation for face detection [6].

Tracking property is the capability of recovering the global pose change of the head from the observed movement among video frames. There are two kinds of general methods of tracking function. One is the method based on recognition or motion. The other is the tracking method based‘on edges and areas [17].The works include a neural network utilized to learn the skin and nonskin colors to track [15], adaptively builds a target model reflecting changes in appearance [9] and on methods of face bounding box and Convolutional Neural Networks (CNNs) [1].

Head pose estimation is the capacity to estimate the head pose of another person that can be done by a video scenario or using images dataset, such as, the MultiPIE [8] or Head Pose Image Database [7]. In the literature, some works proposed the image treatment using explicit 3D models [11] or 2D view-based models [12], [1].

As far as we know, no previous work addressed jointly the tasks of face detection, tracking, head pose estimation and identification of the human being presence in the environment, as we are addressing in this paper.

III. Computer vision system

In this section, the system that is being proposed by estimating the head pose and the presence of a human being is presented. The process for head pose estimation is constituted by the following stages:

1) **Face detection** - Front facial regions are found by cascaded-Adaboost face detectors.
2) **The process of tracking** - This is done by using Camshift (adopting color) applied to the video images;
3) **Image treatment** - The detected facial region is scale-normalized to a fixed size and converted to grayscale;
4) **Histogram oriented gradient** - A gradient orientation histogram is computed and the descriptors are generated;
5) **Learning method**- The descriptors are passed to Support Vector Classification (SVC) in cascade.

All process are divided in two steps. First, the descriptors are used by the learning method to create a model and in the second step, the learning method uses de descriptor to classify correctly the environment's state.

The states defined by us for the classification are: frontal, right min, right max, down right max, down right min, down frontal, down left min, down left max, left max, left min and out. They are represented in the Figure 1.

All of the stages are described in details to follow.

A. Face detection

To detect the location of the partner'ss head, it was used the Adaboost cascades method to find the frontal face profile [16], [10]. The method uses Haar-like features that encode the existence of oriented contrasts between regions in the image. The name Haar-like features are related with their process, computed in similar way of the coefficients in Haar wavelet transforms. A set of these features can be used to encode the contrasts exhibited by a human face and their spacial relationships. [2]

In the next step, a classifier is trained. It can be applied to a region of interest (of the same size as used during the training phase) in an input image. The classifier outputs "1" if the region likely contains the object (i.e., face), and "0" otherwise [2]. In this point, the algorithm is responsible to define the rectangle of the face, that is, the representation of feature (color) distribution will used by the Camshift algorithm.

Camshift algorithm is a robust method of finding local extrema in the density distribution of a data set. This is an easy process for continuous distributions; in that context, it is essentially just hill climbing applied to a density histogram of the data [2].

The algorithm works as it follows. Firstly, the feature distribution to represent an object is chosen (e.g., color), then the camshift window is started over the feature distribution generated by the object, and finally, the chosen feature distribution over the next video frame is computed. Starting from the current window location, the mean-shift algorithm will find the new peak or mode of the feature distribution, which (presumably) is centered over the object that produced the color and texture in the first place. In this way, the camshift window tracks the movement of the object frame by frame [2].

All this process is responsible for tracking face in different head position, such as, presented in the figure 2, 3 and 4.

B. Image treatment

The process of face detection result of two rectangles because the Camshift process has the property of resize the density distribution. Then, the algorithm merged the rectangles of the face with a limitation of high and width size with the same value of the first detection.

The final rectangle is used to create the region of interest. This region is down-sampled to a fixed size of 32x32 pixels, to ensure that the system is invariant to scale. Furthermore, the image is converted to grayscale.

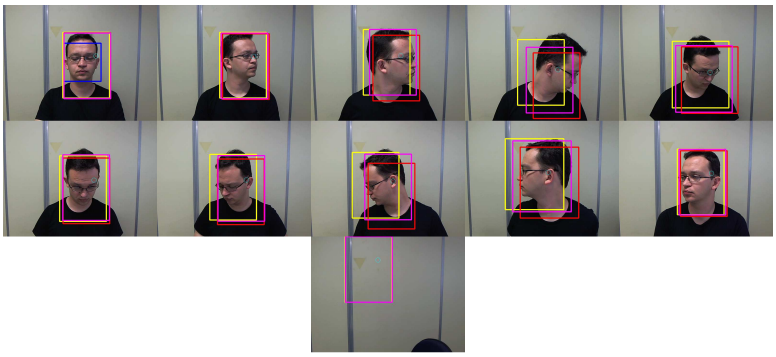

Fig. 1. States defined to be classified by our system. First line: frontal face, right min, right max, down right max, down right min. Second line: down frontal, down left min, down left max, left max, left min. Last state: out

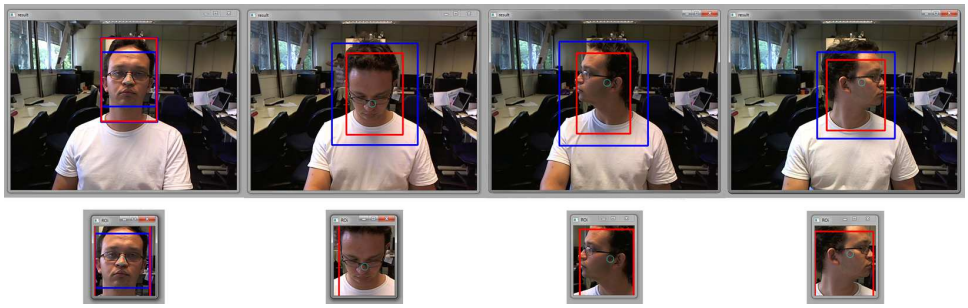

Fig. 2. Tracking head position in the center position. At the top of the figure are the original captured by webcam and the bottom one is down-sampled images of region of interest

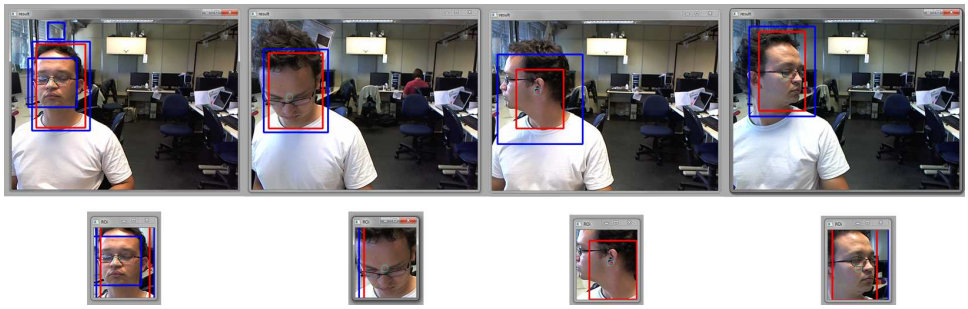

Fig. 3. Tracking head position in the left position of image. At the top of the figure are the original captured by webcam and the bottom one is down-sampled images of region of interest

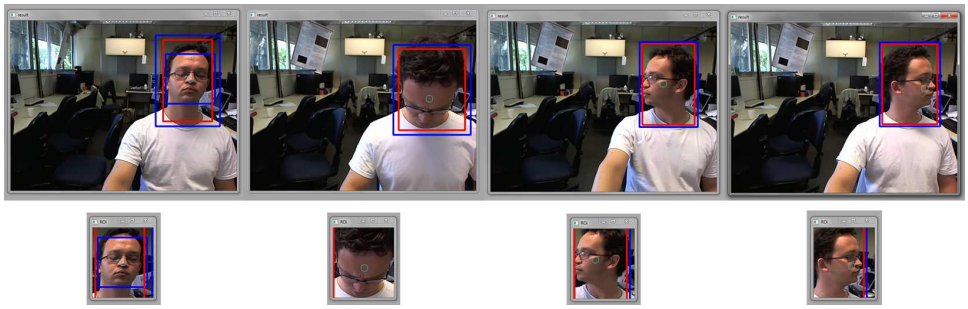

Fig. 4. Tracking head position in the right position of image. At the top of the figure are the original captured by webcam and the bottom one is down-sampled images of region of interest

C. Histogram oriented gradient

To provide a robust description of each facial region, a histogram of gradient orientation is computed. This method is based on evaluating well-normalized local histograms of image gradient orientations in a dense grid.

The basic idea is that local object appearance and its shape can often be characterized rather well by the distribution of local intensity gradients or edge directions, even without precise knowledge of the corresponding gradient or edge positions. In practice, this is implemented by dividing the image window into small spatial regions (*cells*), for each cell accumulating a local 1-D histogram of gradient directions or edge orientations over the pixels of the cell. The combined histogram entries form the representation. For better invariance to illumination, shadowing, etc., it is also useful to contrast-normalize the local responses before using them. This can be done by accumulating a measure of local histogram *energy*, over somewhat larger spatial regions (*blocks*), and using the results to normalize all of the cells in the block. The normalized descriptor blocks are referred by Histogram of Oriented Gradient (HOG) descriptors.

The HOG representation has several advantages. It captures edge or gradient structure that is a characteristic of local shape, and it does so in a local representation with an easily controllable degree of invariance to local geometric and photometric transformations: translations or rotations make little difference if they are much smaller that the local spatial or orientation bin size.

D. Learning method

Support Vector Classification (SVC) is a popular machine learning method for classification. It is looking for the optimal separating hyperplane between the two classes by maximizing the margin between the classes' closest points. It is capable to classify a input data in two-class and multi-class. The basic concept is to use a nonlinear kernel function to project the input data into a high-dimensional space and then to use linear regression to fit a hyperplane, as given by:

$$f(x) = w \cdot \Phi(x) - b \tag{1}$$

where $\cdot$ denotes the dot product and $\mathbf{w}$ the normal vector to the hyperplane. The parameter $\frac{b}{\|\mathbf{w}\|}$ determines the offset of the hyperplane from the origin along the normal vector $\mathbf{w}$.

This is accomplished by simultaneously flattening the hyperplane, e.g. minimizing $\left\|w^2\right\|$, whereas also minimizing the sum of the error from data points which lie outside a margin, ϵ, surrounding the hyperplane. Then, one can use nonlinear kernel functions of the form:

$$\kappa(a, b) = \Phi(a) \cdot \Phi(b) \tag{2}$$

The classification kernel used by the system is known by polynomial kernel 3, it was chosen by better performance in previous comparative analyses with other kernels.

$$\kappa_{polynomial}(a, b) = (gamma * a' * b + coef0)^{degree} \tag{3}$$

where $gamma$, $coef0$ and $degree$ are constants and are empirically defined by 0.00308642, 0 and 3, respectively.

We use ten SVCs in cascade with five level. At first level, the system classifies if there is a person in the environment or if the robot is alone. If there is a person, the system verifies if the head of the person is up or down. In the sequence, the system evaluates if frontal face or left and right face profile, left or right face profile, left 90 degrees or 45 degrees of face profile, and finally, right 90 degrees or 45 degrees of face profile. It works like a decision tree.

For the implementation of the proposed system, it was used OpenCV library and an optimized software package for support vector classification [3].

IV. Results

To validate the proposed system, first was analyzed the performance of the database by using R software [14]. For this, we use a k-fold cross validation, where $k = 10$. In Table I is showed the result of each folder and the mean and standard deviation (S.D.) is showed in the Table II.

The Table I is composed by the accuracies of each classification type done by a SVC in each folder. In general, SVCs using polynomial kernel classify correctly, with the lowest value $78,12$ acquired. Moreover, this table showed also the importance of the database for this problem. This assumption is corroborated by de difference of 12.51 among accuracies in the same SVC.

TABLE II. Mean and Standart Variation of ten cross validation using R software

	Mean	S. D.		Mean	S. D.
Out-In	89,13	1,60	Right Max-Min	94,56	4,26
Up-Down	89,70	1,42	D Frontal - LR	84,27	3,91
Frontal - LR	92,44	1,87	D Left-Right	89,74	3,02
Left-Right	97,74	1,72	D Left Max-Min	91,33	3,22
Left Max-Min	97,10	2,30	D Right Max-Min	89,39	4,10

One can noted from the results obtained in Tables I and II, that the proposed system is appropriated.

After this offline evaluation, a group of tests were prepared in which the system needed to classify new states. For this, four evaluations were executed in front of the camera, where the sequence of states were: frontal, right min, right max, down right max, down right min, down frontal, down left min, down left max, left max, left min and out. Futhermore, in all experiments, the time spend in each position was not concerned. The Table III showed the results of mean of ten images analyzed in the center of each position. In other words, this mean that if for the test, 30 images for frontal face were considered, the images analyzed, in this case were between image 10 and 20. The Table IV presents all images analyzed by the value of mean. A total of 2825 images were analyzed.

The results of Table III showed that the best elements classified were FRONTAL, R-MIN, R-MAX and OUT. In spite of get expressive value in the table, the result of right side is not representing the values of primary test in R software. Moreover, the down right profile does not follow the accuracies of up right profile. The other side profile (left), the situation is worst if compared to the result got with R software.

TABLE I. TEN FOLDER CROSS VALIDATION USING R SOFTWARE

	1	2	3	4	5	6	7	8	9	10
Out-In	89,72	86,96	86,56	89,72	88,93	90,51	91,30	88,14	90,91	88,54
Up-Down	88,54	88,78	92,60	89,26	90,45	89,26	90,21	87,35	88,54	89,02
Frontal-LR	92,59	90,37	89,63	91,85	90,37	94,07	93,33	93,33	95,56	93,33
Left-Right	95,23	100	95,24	97,62	100	97,62	97,62	98,81	96,43	98,81
Left Max-Min	97,37	97,37	94,74	92,10	100	97,37	100	97,37	97,37	97,37
Right Max-Min	97,82	100	89,13	97,83	86,96	91,30	95,65	97,83	95,65	93,48
D Frontal - LR	78,12	84,38	90,625	82,29	79,17	86,46	84,375	87,5	82,29	87,5
D Left-Right	91,03	84,61	92,31	93,59	84,61	89,74	91,03	91,03	91,03	88,46
D Left Max-Min	95,56	91,11	88,89	86,67	88,89	91,11	88,89	95,56	91,11	95,56
D Right Max-Min	96,97	93,94	87,88	87,88	84,85	87,88	93,94	87,88	87,88	84,85

The values seen in Table IV confirmed what was presented in Table III. The diagonal which contains the correct classification get considerable $(21,61)$ with some misrepresentations (three results above $21,61$) what can be considered as an interesting start point of study.

V. CONCLUSION

In this paper, it was presented an ongoing work for the development of a computer vision system to turn able a robot to identify the presence of a person in an environment and in such case, to classify the head pose. This system will be inserted in a social robotic architecture inspired on Behavior Analysis. A complete analyze of data accuracies was presented fo offline data.

The experimental results showed that the proposed system has great capability to classify to identify the presence of a person and her/him head pose. In the real time executions, the system showed have flexibility of the head position, but the classification accuracies need to be improved.

As a future work, we intend to improve the classification accuracy and stability of the system. Moreover, identification of objects will be added to the system. This complex visual system will able to percept all important information from environment to our robotic architecture.

ACKNOWLEDGMENT

The authors would like to thank Bruno Feres de Souza to help us with R software and SVMs. The authors also would like to thank FAPESP with the project number 2008/09744-4, CNPq and CAPES by the support received.

REFERENCES

[1] S. Asteriadis, K. Karpouzis, and S. Kollias. Face tracking and head pose estimation using convolutional neural networks. In *Proceedings of the SSPNET 2nd International Symposium on Facial Analysis and Animation*, FAA '10, pages 19–19, New York, NY, USA, 2010. ACM.

[2] G. Bradski and A. Kaehler. *Learning OpenCV*. O'Reilly Media Inc., 2008.

[3] C.-C. Chang and C.-J. Lin. Libsvm: A library for support vector machines. *ACM Trans. Intell. Syst. Technol.*, 2(3):27:1–27:27, May 2011.

[4] Alessandro Colombo, Claudio Cusano, and Raimondo Schettini. 3d face detection using curvature analysis. *Pattern Recognition*, 39(3):444 – 455, 2006.

[5] R. Feraund, O.J. Bernier, J.-E. Viallet, and M. Collobert. A fast and accurate face detector based on neural networks. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 23(1):42 –53, jan 2001.

[6] K.-B. Ge, J. Wen, and F. Fang. Adaboost algorithm based on mb-lbp features with skin color segmentation for face detection. In *Wavelet Analysis and Pattern Recognition (ICWAPR), 2011 International Conference on*, pages 40–43, july 2011.

[7] N. Gourier, D. Hall, and J. L. Crowley. Estimating face orientation from robust detection of salient facial features. In *Proceedings of Pointing 2004, ICPR, International Workshop on Visual Observation of Deictic Gestures*, 2004.

[8] R. Gross, I. Matthews, J. Cohn, T. Kanade, and S. Baker. Multi-pie. *Image and Vision Computing*, 28(5):807–813, 2010. ¡ce:title¿Best of Automatic Face and Gesture Recognition 2008¡/ce:title¿.

[9] M. Kim, S. Kumar, V. Pavlovic, and H. Rowley. Face tracking and recognition with visual constraints in real-world videos. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1 –8, june 2008.

[10] R. Lienhart and J. Maydt. An extended set of haar-like features for rapid object detection. In *Image Processing. 2002. Proceedings. 2002 International Conference on*, volume 1, pages I–900 – I–903 vol.1, 2002.

[11] L.-P. Morency, J. Whitehill, and J. Movellan. Monocular head pose estimation using generalized adaptive view-based appearance model. *Image and Vision Computing*, 28(5):754–761, 2010. ¡ce:title¿Best of Automatic Face and Gesture Recognition 2008¡/ce:title¿.

[12] E. Murphy-Chutorian, A. Doshi, and M. M. Trivedi. Head Pose Estimation for Driver Assistance Systems: A Robust Algorithm and Experimental Evaluation. In *Intelligent Transportation Systems Conference (ITSC), IEEE*, pages 709–714, September 2007.

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

[14] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2012. ISBN 3-900051-07-0.

[15] P. Vadakkepat, P. Lim, L.C. De Silva, Liu Jing, and Li Li Ling. Multimodal approach to human-face detection and tracking. *Industrial Electronics, IEEE Transactions on*, 55(3):1385 –1393, march 2008.

[16] Paul A. Viola and Michael J. Jones. Rapid object detection using a boosted cascade of simple features. In *CVPR (1)*, pages 511–518. IEEE Computer Society, 2001.

[17] J. I. Yoon, K. K. Ahn, Y. R. Cho, T. H. Nam, D. Q. Truong, and W. K. Jo. A study on face tracking in real-time for robot. In *Control, Automation, Robotics and Vision, 2008. ICARCV 2008. 10th International Conference on*, pages 2182 –2187, dec. 2008.

TABLE III. ACCURACIES OF TEN EXAMPLE CLASSIFICATION

	FRONTAL	R-MIN	R-MAX	D R-MAX	D R-MIN	D FRONTAL	D L-MIN	D L-MAX	L-MAX	L-MIN	OUT
FRONTAL	5,50	0,50	0,00	0,75	0,25	0,50	0,50	1,00	0,00	0,75	0,25
R-MIN	1,75	5,00	0,75	0,00	0,25	0,75	0,75	0,75	0,00	0,00	0,00
R-MAX	0,75	0,75	5,00	0,50	0,75	0,25	0,75	0,25	0,00	0,00	1,00
D R-MAX	1,00	1,00	0,00	2,50	0,25	1,00	0,25	2,75	0,00	0,25	1,00
D R-MIN	0,75	1,25	0,50	1,00	0,75	1,50	0,50	1,50	0,50	0,00	1,75
D FRONTAL	2,25	1,75	0,00	0,75	0,25	2,75	0,75	1,00	0,00	0,00	0,50
D L-MIN	2,25	0,50	0,00	0,25	0,50	2,00	2,75	1,25	0,00	0,00	0,50
D L-MAX	1,50	0,50	0,00	0,25	0,00	1,50	0,75	4,50	0,50	0,25	0,25
L-MAX	3,00	1,25	0,25	0,50	0,50	0,75	0,25	1,25	0,00	0,50	1,75
L-MIN	4,25	0,75	0,00	0,00	0,25	1,50	0,25	1,25	0,25	0,25	1,25
OUT	0,75	0,25	0,25	0,50	0,00	1,25	1,25	0,75	0,00	0,00	5,00

TABLE IV. ACCURACIES OF ALL DATA CLASSIFICATION

	FRONTAL	R-MIN	R-MAX	D R-MAX	D R-MIN	D FRONTAL	D L-MIN	D L-MAX	L-MAX	L-MIN	OUT
FRONTAL	54,59	3,87	0,00	8,08	6,68	3,26	2,46	9,74	0,00	1,69	9,62
R-MIN	17,88	34,81	7,22	2,86	3,54	10,66	3,55	13,86	1,72	0,23	3,67
R-MAX	20,04	6,70	40,41	3,87	3,02	5,40	6,00	7,40	0,66	1,10	5,40
D R-MAX	8,15	7,77	1,00	21,61	3,62	10,58	3,95	19,62	0,00	5,27	18,43
D R-MIN	8,36	7,22	4,80	11,08	9,25	13,13	9,23	20,21	3,00	0,37	13,35
D FRONTAL	15,77	7,91	1,57	4,06	1,82	32,85	6,05	15,86	1,74	1,21	11,16
D L-MIN	15,94	3,33	0,00	0,76	2,57	17,83	26,49	17,72	0,68	3,22	11,47
D L-MAX	13,34	3,26	0,00	1,09	0,83	15,73	5,89	49,55	4,61	2,56	3,14
L-MAX	22,03	8,26	1,48	3,28	1,88	3,44	10,40	22,63	5,35	6,05	15,18
L-MIN	33,89	11,07	1,32	1,77	4,78	16,52	4,54	9,88	1,97	7,15	7,11
OUT	8,83	4,46	2,40	0,56	0,69	10,73	7,30	14,67	1,81	1,87	46,69