Supervised Descent Method (SDM) applied to accurate pupil detection in off-the-shelf eye tracking systems

Andoni Larumbe
Public University of Navarre
Pamplona, Spain
andoni.larumbe@unavarra.es

Rafael Cabeza
Public University of Navarre
Pamplona, Spain
rcabeza@unavarra.es

Arantxa Villanueva
Public University of Navarre
Pamplona, Spain
avilla@unavarra.es

ABSTRACT

The precise detection of pupil/iris center is key to estimate gaze accurately. This fact becomes specially challenging in low cost frameworks in which the algorithms employed for high performance systems fail. In the last years an outstanding effort has been made in order to apply training-based methods to low resolution images. In this paper, Supervised Descent Method (SDM) is applied to GI4E database. The 2D landmarks employed for training are the corners of the eyes and the pupil centers. In order to validate the algorithm proposed, a cross validation procedure is performed. The strategy employed for the training allows us to affirm that our method can potentially outperform the state of the art algorithms applied to the same dataset in terms of 2D accuracy. The promising results encourage to carry on in the study of training-based methods for eye tracking.

CCS CONCEPTS

• **Computing methodologies** → **Tracking**; **Supervised learning by regression**;

KEYWORDS

Eye tracking, Supervised Descent Method, SDM, Cascaded Regressors, 2D iris center estimation

ACM Reference Format:
Andoni Larumbe, Rafael Cabeza, and Arantxa Villanueva. 2018. Supervised Descent Method (SDM) applied to accurate pupil detection in off-the-shelf eye tracking systems. In *ETRA '18: 2018 Symposium on Eye Tracking Research and Applications, June 14–17, 2018, Warsaw, Poland*. ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3204493.3204551

1 INTRODUCTION

In the last years the research about off-the-shelf eye tracking has attracted the attention of a significant number of researchers. The possibility of doing eye tracking using a frontal webcam or the camera of a mobile gadget would potentially open the application field of the technology. The knowledge regarding high resolution and infrared eye tracking can be partially applied to the lower resolution scenarios but it cannot overcome all the new challenges showing up in the new working framework.

When talking about eye tracking technology it is usual to differentiate between the stages corresponding to the image processing part, i.e. eye tracking and the method employed to connect the image with the Point of Regard (PoR) or the Line of Sight (LoS) named generally as gaze estimation. Basically, the eye tracking algorithms have as input the image or images acquired by the vision system and process the data in order to extract valid features, such as pupil center among others. The gaze estimation stage calculates gaze as a function of image features. This type of eye/gaze tracking systems are named as feature based when they perform a mapping, e.g. by means of a polynomial, to estimate the PoR. Model based methods aim to extract a geometrical model of the setup including the subject to achieve the estimation of gaze but in general they employ features extracted from the image to estimate gaze too. In contrast, appearance based methods do not extract features from the image but they employ the whole image or a rasterized vector of the image in order to estimate gaze as a result of a learning procedure such as neural networks. In fact, they can be described as methods based on learning or training processes. In other words, they require a set of data considered as training data representing the variability of the problem in order to get adapted to the solution. Interesting reviews about the gaze estimation methodology can be found in the literature focused in both, high and low resolutions systems [Ferhat and Vilariño 2016; Hansen and Ji 2010].

An analysis of the works devoted to high resolution and infrared gaze estimation shows that feature and model based eye tracking systems have demonstrated to be the consensus solution [Cerrolaza et al. 2012b; Guestrin and Eizenman 2006]. In general, these methods have demonstrated to be simpler and more accurate approaches for high performance systems. The impact of training based methods has not a big impact in high resolution systems to date, except for remarkable works such as [Fuhl et al. 2016] in which training based procedures are used for robust pupil detection using high resolution images in wild working conditions. However, when moving to lower resolution systems the influence of learning and training methodologies begins to be more relevant and show up as a promising tool in many aspects related to gaze estimation using off-the-shelf components. We still find relevant works presenting features and model based approaches. One of the first works regarding low resolution is the one presented by Valenti et al. [Valenti et al. 2012]. An iterative process is carried out in which "normalized" eye images are obtained from head position and eye position is then employed to correct head information. The paper clearly demonstrates that the combination of both elements improves gaze estimation. In the work by Zhang et al. [Zhang et al. 2016] isophotes and gradient features are employed to estimate the eye center locations. Gradient information is also used in the work

ETRA '18, June 14–17, 2018, Warsaw, Poland

ACM ISBN 978-1-4503-5706-7/18/06.
https://doi.org/10.1145/3204493.3204551

by Timm and Barth [Timm and Barth 2011]. The fast radial symmetry transform takes advantage of the pupil shape assumed to be circular in order to detect iris center [Skodras and Fakotakis 2015]. Image topography is also employed in [Villanueva et al. 2013].

In principle, training methodologies could be applied to both, eye tracking and gaze estimation stages with a variable level of overlapping between them. A clear example of total overlapping between the two stages are those works employing deep learning in which the input is the image and the output is gaze and no processing of the image is performed. Those kind of systems are based on large scale training procedures in which a huge number of images representing the variability of the problem feed a Convolutional Neural Network (CNN) that is adapted to the problem and ideally should be able to find a solution successfully for a new image of the same problem [Krafka et al. 2016]. Other approaches take advantage of the power of CNN based methods but they consider it interesting to reinforce the method by employing knowledge based information, such as head pose or facial landmarks [Zhang et al. 2015].

Pursuing in the area of learning techniques, in the work by Kacete et al. [Kacete et al. 2016b] random forest techniques are used to estimate head pose and 2D pupil center [Kacete et al. 2016a]. Both outputs are them used to estimate gaze. Random forest-based techniques [Breiman 2001] used randomized trees to be applied in classification and regression problems. Cascaded regressors methods have demonstrated to be highly accurate and robust in facial landmark tracking [Feng et al. 2015b; Xiong and De la Torre 2013]. In this manner, works applying cascaded regressors for pupil center detection and eye tracking can be found in recent publications [Gou et al. 2016, 2017].

We present an accurate iris center detection method based on cascaded regressors for low resolution scenarios using GI4E database[1] as evaluation framework. GI4E is a database containing images of 103 users gazing at different points on the screen in a standard desktop scenario, i.e. twelve images per user have been recorded [Villanueva et al. 2013]. One of the outstanding characteristics of the database is the accuracy of the labelling procedure. The images contain labels for the center of the iris and the eye corners. Each image has been marked by three independent users, and the final label has been calculated as the mean value among the three assuring high accurate data. GI4E has been selected as evaluation framework not only because its accuracy but due to the fact that it is a well-known standard in the field. In fact, many of the works mentioned previously evaluate their results using GI4E.

In the next section the methods employed by our method are presented in which the cascaded regressor is described. A two stage procedure has been designed as it will be explained in the method. The experiments carried out permit us to measure the robustness and accuracy of our pupil center detection method to be compared with state of the art works using GI4E. Finally, the conclusions and further work are presented.

2 METHODS

The main idea is to train two cascaded regressors (**CR**) based on Supervised Descent Method (SDM) and Random Cascaded-Regression Copse (R-CR-C) by which to detect the pupil center. SDM was proposed by X. Xiong and F. de la Torre [Xiong and De la Torre 2013] for minimizing Nonlinear Least Squares (NLS) problems in the context of computer vision without using second order descent methods which have some drawbacks in this field. On the one hand, the function might not be analytically differentiable and numerical approximations are impractical, on the other, the Hessian may be large and not positive definite. R-CR-C was proposed by Z. Feng et al. [Feng et al. 2015b] as a modification of the SDM algorithm, improving it by proposing among others, an adaptive scheme for scale-invariant updates and the addition of a regularization term.

Given an image $\mathbf{d} \in \mathbb{R}^{m\times 1}$ of m pixels, $\mathbf{d}(\mathbf{x}) \in \mathbb{R}^{p\times 1}$ indexes p landmarks in the image. $\mathbf{h}$ is a nonlinear feature extraction function, $\mathbf{h}(\mathbf{d}(\mathbf{x})) \in \mathbb{R}^{128p\times 1}$ in the case of extracting Histogram of Oriented Gradients (HOG) features. In this setting, during training SDM learns a cascaded regressor (**CR**) from a set of L images labeled with a group of ground-truth landmarks $\{\mathbf{x}_*^i\}(i = 1, \cdots, L)$. A **CR** is formed by K weak regressors in cascade as:

$$\mathbf{CR} = \mathbf{r}_1 \circ \mathbf{r}_2 \circ \cdots \circ \mathbf{r}_K. \tag{1}$$

Each weak regressor $\mathbf{r}_k$ is represented by $\{\mathbf{R}_k, \mathbf{b}_k\}$ $(k = 1, \cdots, K)$, where $\mathbf{R}_k$ is the descent map and $\mathbf{b}_k$ is the bias term of the k^{th} regressor. The bias term represents the average of $\mathbf{R}_k\mathbf{h}(\mathbf{d}^i(\mathbf{x}_*^i))$, where $\mathbf{h}(\mathbf{d}^i(\mathbf{x}_*^i))$ represents the HOG values computed on the local patches extracted from the ground truth landmarks for the i^{th} image. The bias term is required because $\mathbf{h}(\mathbf{d}^i(\mathbf{x}_*^i))$ is parametrized not only by $\mathbf{x}$, but also by the images (i.e., $\mathbf{d}^i$) and the **CR** has to learn to generalize detection for new images (e.g., different subjects).

In training, images are normalized by the inter-eye distance (IED) in order to perform a scale-invariant strategy. For each iteration k, $\mathbf{R}_k$ and $\mathbf{b}_k$ are computed by minimizing the expected loss between the true state $\mathbf{x}_*^i$ and the previous iteration predicted state $\mathbf{x}_{k-1}^i$, given by

$$\sum_{i=1}^{L} \|\mathbf{x}_*^i - \mathbf{x}_{k-1}^i + \mathbf{R}_k\mathbf{h}(\mathbf{d}^i(\mathbf{x}_{k-1}^i)) - \mathbf{b}_k\|_2^2 + \lambda\|\mathbf{R}_k\|_F^2, \tag{2}$$

where λ is the weight of the regularization term. Once $\mathbf{R}_k$ and $\mathbf{b}_k$ have been estimated, each sample $\mathbf{x}_{k-1}^i$ is updated to its new location $\mathbf{x}_k^i$ as follows:

$$\mathbf{x}_k^i = \mathbf{x}_{k-1}^i - \mathbf{R}_k\mathbf{h}(\mathbf{d}^i(\mathbf{x}_{k-1}^i)) + \mathbf{b}_k. \tag{3}$$

After an update, we recompute a new descent map $\mathbf{R}_{k+1}$ and a new location $\mathbf{x}_{k+1}^i$.

In testing, images are normalized by the the IED and landmarks locations $\mathbf{x}_k^i$ are updated by Equation 3 starting from the initialization (i.e., $\mathbf{x}_0^i$) and using the learned **CR**. Figure 1 shows the simplified flow diagram for training (left) and testing (right) SDM. A more detailed description can be found in the original papers [Feng et al. 2015b; Xiong and De la Torre 2013].

Regarding the two **CR** trained in this work, both have the same architecture. Using a set of mean landmarks calculated from a database ground-truth values and a bounding box, $\mathbf{x}_0$ is generated, a HOG nonlinear feature extraction is performed ($\mathbf{h}(\mathbf{d}(\mathbf{x}_0))$) and $\mathbf{x}_k$ is updated recursively using Equation 3. The main difference between the two regressors is the set of landmarks for which they

[1] http://gi4e.unavarra.es/databases/gi4e/

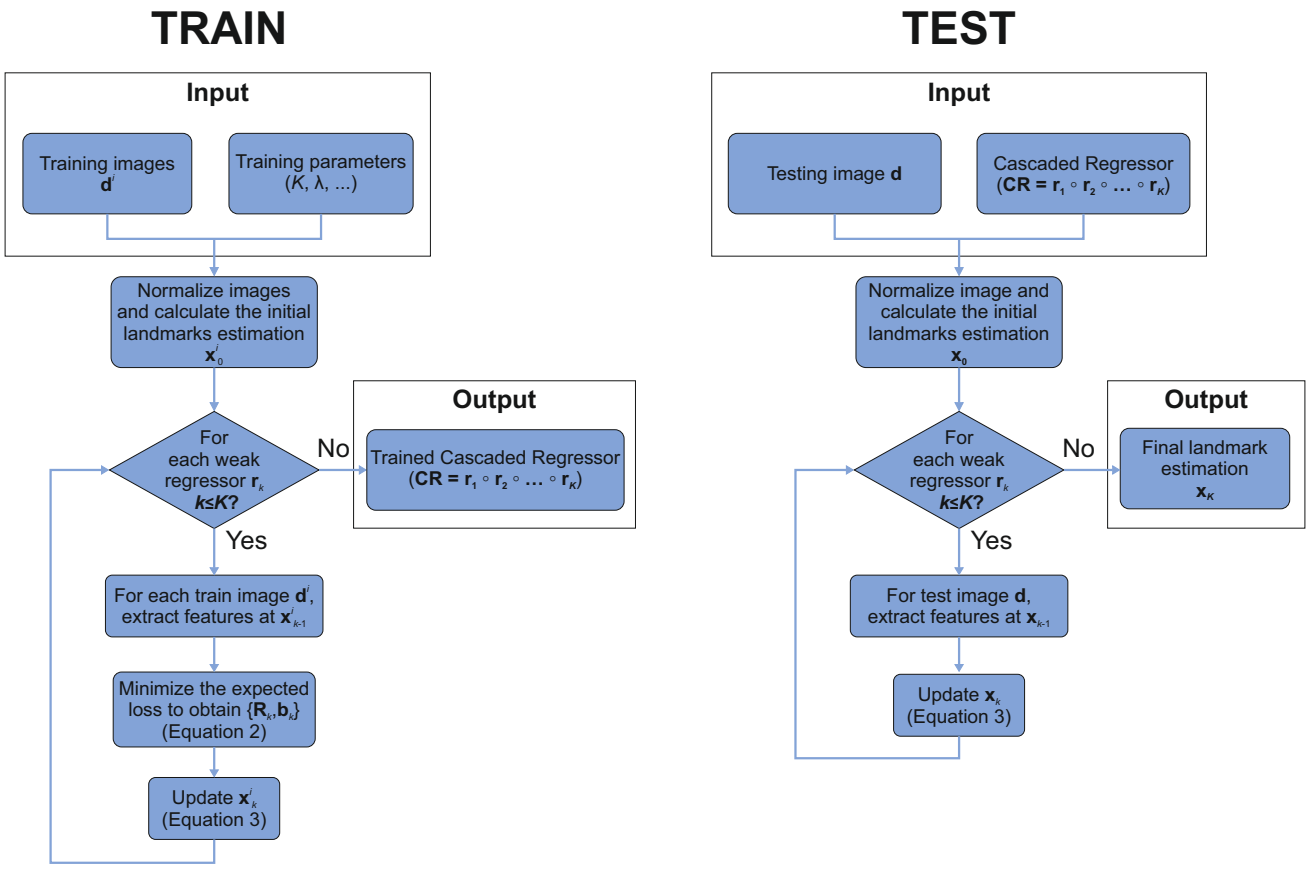

Figure 1: Flow diagram for training (left) and testing (right) SDM.

have been trained. The first **CR** (named as Face Cascaded Regressor, F-CR) is trained to detect a set of 32 landmarks which corresponds to the facial points while the second one (named as Eye Cascaded Regressor, E-CR) is trained to detect the inner and outer eye corners and the pupil centers (6 landmarks).

The mean landmarks for each regressor have been calculated aligning the ground-truth landmarks of all images into a common coordinate frame using the Generalized Procrustes Alignment (GPA) [Cerrolaza et al. 2012a; Goodall 1991]. Figure 2 shows the original data aligned using GPA (green) and the mean landmarks (red).

Figure 3 shows an overview of the method proposed. Firstly, a face bounding box is detected and the mean landmarks calculated for F-CR ($\mathbf{f}_0 := \mathbf{x}_0$) are initialized (Figure 3a). Then, $\mathbf{f}_k$ is updated recursively as many times as the number of weak regressors composing the F-CR according to Equation 3 (Figure 3b). Using the landmarks detected by F-CR, the roll angle of the user in the image can be estimated and, in order to get a nearly 0-roll pose and prevent the tracker from getting lost, the image is rotated. In this manner the eye bounding box is created and the mean landmarks are calculated for E-CR ($\mathbf{e}_0 := \mathbf{x}_0$) as shown in Figure 3c. Finally $\mathbf{e}_k$

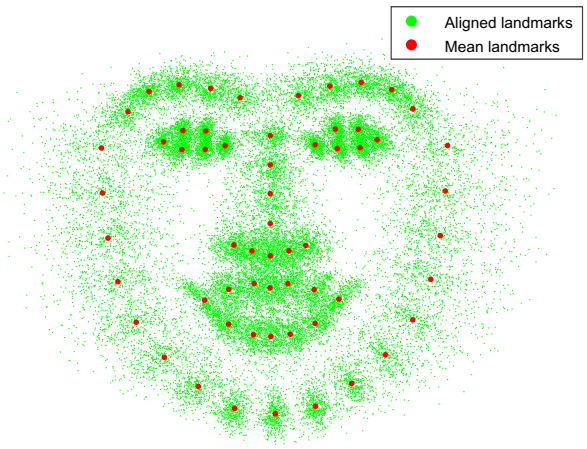

Figure 2: Mean landmarks calculated from the alignment of the original landmarks. Alignment has been performed using GPA.

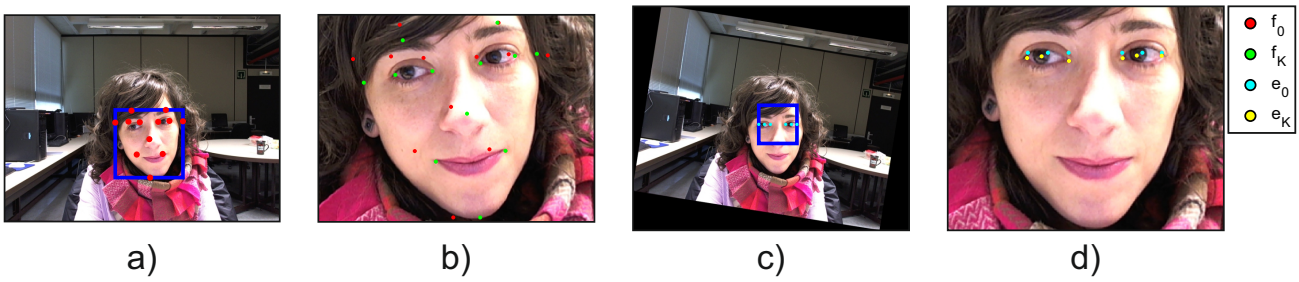

Figure 3: Method overview. Firstly, a face bounding box is detected and $\mathbf{f}_0$ is initialized (a). Then, $\mathbf{f}_k$ is updated recursively (b). Using the landmarks detected by F-CR, the image is rotated, the eye bounding box is created and $\mathbf{e}_0$ is initialized (c). Finally $\mathbf{e}_k$ is updated recursively (d) and the output is corrected according to the estimated rotation value.

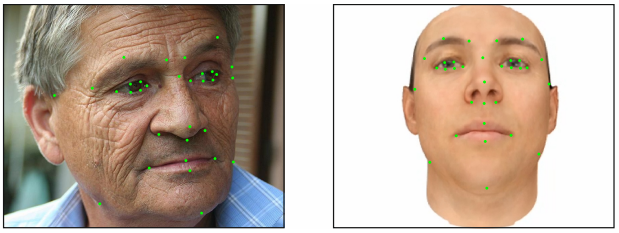

Figure 4: Example of F-CR training data. On the left, a real image from HELEN dataset and on the right, synthetic image from UPNA Synthetic Head Pose Database. Both images are represented with the 32 ground-truth landmarks used to train F-CR.

Figure 5: Data augmentation example. On the left the original image and on the right the flipped image. It is worth noting that when flipping landmarks, the index numbers must be changed to make them correspond to the opposite eye.

is updated recursively as many times as the number of weak regressors composing the E-CR (Figure 3d) and the output is corrected according to the estimated rotation value.

Implementing image rotation before E-CR is very important to compensate for the lack of databases that have the center of the eye marked with the required precision. This normalization procedure by means of a rotation reduces the variability of images and allows E-CR to work better with fewer training images. In the case of the F-CR, there are many more databases (no pupil center is required) and the regressor can learn more variability.

2.1 Face Cascaded Regressor

The use of this regressor is based on the fact that, as we have already said, to the best of our knowledge there is a lack of databases that have the center of the eye marked with the required precision and it is hard to perform a single step procedure which detects well in a set of highly variable images. The F-CR training and tracking is performed by using a tracking software written in C++ by Patrik Huber and available on GitHub [Huber 2015; Huber et al. 2015]. The specific details that have been implemented in the present work for this first regressor are detailed below.

The face bounding box is detected using Viola-Jones algorithm [Viola and Jones 2001] and the landmarks chosen to track are a subset of 32 landmarks from the 68 points used by the Intelligent Behaviour Understanding Group (iBUG) [Sagonas et al. 2013b]. A comparison using different subsets of landmarks is made and it will be presented in evaluation section. Regarding the training process, F-CR has been trained using both real and synthetic images: a total of 3283 real and 1200 synthetic images have been used. The aim of adding synthetic images to the regressor training process is to increase the training data to make it more robust in high rotation situations with a set of images that shows high rotation values [Feng et al. 2015a; Larumbe et al. 2017]. Real images has been chosen from AFW [Zhu and Ramanan 2012], HELEN [Le et al. 2012], iBUG [Sagonas et al. 2013a] and LFPW [Belhumeur et al. 2013] databases. Nevertheless, landmarks used are re-annotated by using the iBUG semi-automatic annotation methodology followed by an additional manual correction [Sagonas et al. 2016, 2013b]. The synthetic images have been obtained by using the UPNA Synthetic Head Pose Database [Larumbe et al. 2017]. Figure 4 shows an example of the two types of images used, real (left) and synthetic (right). Both images are represented with the 32 ground-truth landmarks used to train F-CR.

2.2 Eye Cascaded Regressor

As regard the second regressor, E-CR training and tracking is performed by using a tracking software written in MATLAB by Zhenhua Feng and available on GitHub [Feng 2016]. The specific details of according to the present work this regressor are as follows:

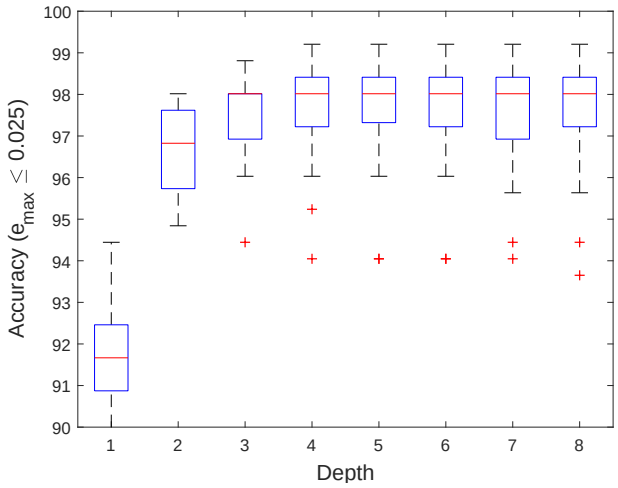

Figure 6: Accuracy comparison of all parameter combinations with varying values of K. It can be observed that error does not improve from $K = 4$.

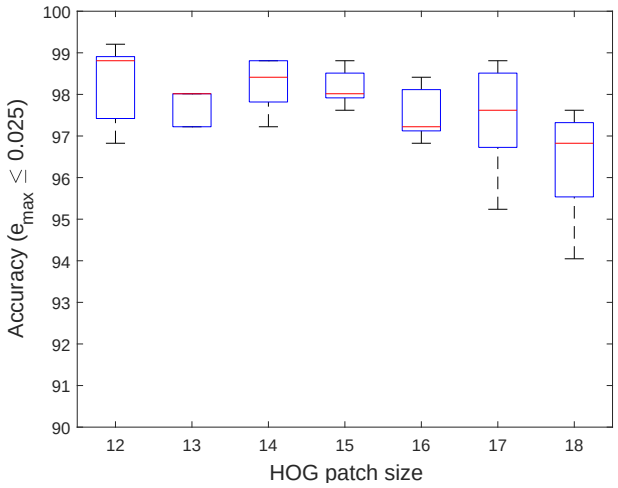

Figure 7: HOG patch size effect. It can be observed that $P = 12$ provides the best accuracy results and increasing P above 15 worsens results.

The eye bounding box is generated using the output of the first regressor (the obtained eye corners landmarks). Once the image is rotated the IED is calculated and determines both eye bounding box width and height (i.e. it is a square bounding box). Images have been chosen from GI4E database [Villanueva et al. 2013] which as it is said before is a database for eye-tracking that consists of a set of more than 1300 images acquired with a standard webcam, corresponding to different subjects gazing at different points in the screen. In order to improve the tracking and increase robustness, we decided to perform a data augmentation in training by flipping images horizontally. Figure 5 shows an example of the implementation of this data augmentation, on the left the original image and on the right the flipped image are presented. It is worth noting that when flipping landmarks, the index must be changed to make them correspond to the opposite eye. Tracking two landmarks (pupil center ones) would be enough but it is decided to track six (inner and outer eye corners and the pupil centers) because in the case of having a video, these can replace the output of the first regressor to rotate the next-frame image and generate the eye bounding box saving computation time. However, there is no video database that have the center of the eye marked to get results with.

Looking at equations 1 and 2, it can be observed that we can set parameters as the regularization weight (λ) or depth (K). Other parameter that can be settled is the size of the local patch around the landmarks where the HOG features are extracted. An analysis of the optimal parameters will be discussed in the evaluation section.

3 EVALUATION

In order to apply the proposed method, we consider it important to carry out a preliminary study of the algorithm and to evaluate its behaviour according to the alternative parameters of the regressor. The cascaded regressor as it has been previously described is based on specific parameters such as the depth, i.e. the number of weak regressors K, the regularization parameter λ and the radius of the HOG feature P. It is important to evaluate the sensitivity of the results according to these parameters. This study is carried out for E-CR regressor for which the accuracy is highly relevant in this work. In the case of F-CR is not the accuracy but the robustness that is important in our study. The objective of the F-CR regressor is to obtain a bounding box in which the eye area is contained hence, it is decided to carry out a study about the robustness of the regressor as a function of the number of landmarks employed in the detection of the face.

This section is organized as follows, in subsections 3.1 and 3.2 the study regarding E-CR parameters and F-CR robustness is carried out. Once both regressors have been optimized, the results of the method applied to GI4E are presented and discussed in the subsection 3.3.

3.1 E-CR optimal parameters

As stated above, an analysis of the optimal parameters for E-CR has been made. For this purpose, a data partition has been performed splitting data into three groups: 60% for training, 20% for cross validation and 20% for testing. It is worth noting that this splitting has been performed by users and not by images because this ensures independence between cross validation, test and train images. Thus, since GI4E has 103 users, 61 are used for training, 21 for cross validation and 21 for testing. The value ranges of the parameters are as follows: depth $K = \{1, \cdots, 8\}$, HOG patch size $P = \{12, \cdots, 18\}$ and regularization weight $\lambda = \{1, 3, 5, 10, 30\}$, resulting in a total of 280 training combinations.

To evaluate the accuracy of the proposed algorithm and to compare it with state of art, the relative error measure proposed by [Jesorsky et al. 2001] is used. It first calculates the absolute error as the Euclidean distance between pupil center estimates and ground-truth values provided by the database and it is normalized relative to the inter-pupillary distance (ground-truth). This is formulated by Equation 4:

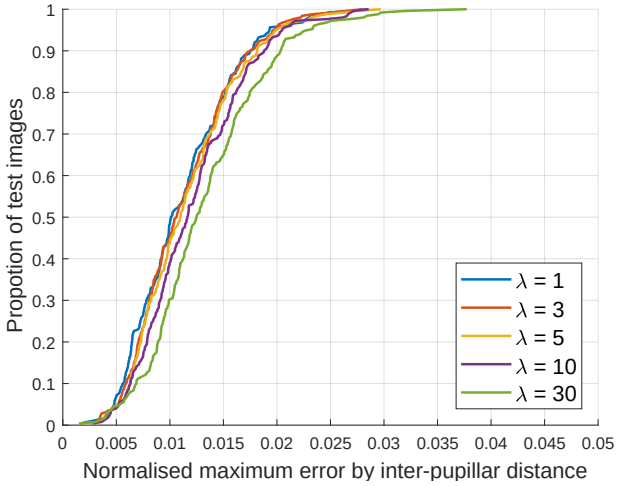

Figure 8: Regularization weight λ analysis using cumulative error distribution normalized by inter-pupillar distance. It is observed that $\lambda = 1$, $\lambda = 3$ and $\lambda = 5$ present similar results. Increasing λ above 10 worsens results.

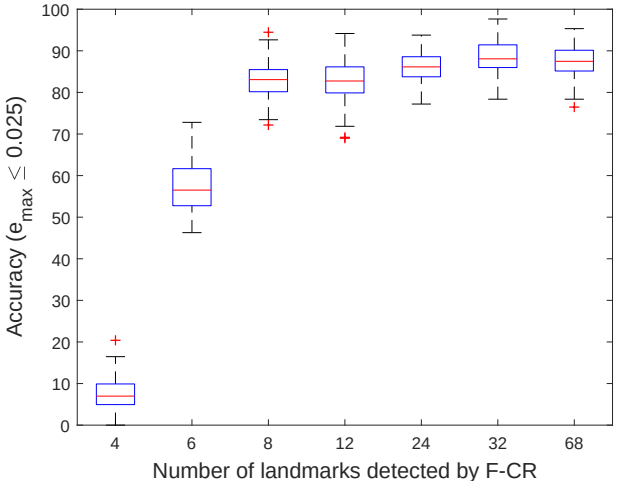

Figure 9: Accuracy comparison of all training with varying values of N. It can be observed that error does not improve from $N = 32$.

$$e_{max} = \frac{max(d_{left}, d_{right})}{\omega}, \tag{4}$$

where d_{left} and d_{right} are the absolute errors for the eye pair, and ω is the inter-pupillary distance. The maximum of d_{left} and d_{right} after normalization is defined as *maximum normalized error* e_{max}. The accuracy is calculated as the percentage of images for which the error is below specific e_{max} values.

Firstly, a depth (K) analysis has been made on cross validation subjects in order to determine a threshold value for K from which the improvement is not significant. Depth is the first parameter to be optimized because it affects the computation time, and it is important to know if an increase of K (with its correspondent decrease in frames per second) leads to an improvement in terms of accuracy. Figure 6 shows a box-plot comparing the accuracy $e_{max} \leq 0.025$ of all parameter combinations with varying values of K. It can be observed that from $K = 4$ the accuracy $e_{max} \leq 0.025$ does not improve significantly. Once K is set to 4, the next parameter to optimize is the HOG patch size, i.e. P, which also has an effect on the computation time of the algorithm. Figure 7 shows a box-plot comparing the accuracy $e_{max} \leq 0.025$ of all parameter combinations with varying values of P. It can be observed that P equal to 12 provides the best accuracy results and increasing P above 15 worsens results. The last parameter to optimize is the regularization weight, i.e. λ. It is selected using the cumulative error distribution of parameter combinations with $K = 4$ and $P = 12$ as it is shown in Figure 8. It is observed that $\lambda = 1$, $\lambda = 3$ and $\lambda = 5$ present similar results, but $\lambda = 1$ is selected because it presents a higher percentage of images with $e_{max} \leq 0.025$.

Finally, according to the previous analysis the E-CR with $K = 4$, $P = 12$ and $\lambda = 1$ values is selected.

3.2 F-CR robustness

In this section the effect of the number of landmarks detected by the F-CR is studied. As well as the depth parameter, the number of landmarks detected by the F-CR affects the computation time, and it is important to know if an increase of the number of landmarks detected leads to an improvement of final result. For this purpose a different data partition is performed: data have been divided into two groups, 80% for training and 20% for testing (split by users). However to avoid the possible bias derived from the specific set of training users in the result, a hundred random splits have been performed and the accuracy of result has been calculated from the average of all of them.

The number of landmarks N selected for the study are {4, 6, 8, 12, 24, 32, 68} and results are shown in Figure 9. It can be observed that error does not improve from $N = 32$.

3.3 Results and discussion

Using the optimal parameters of E-CR and the optimal number of points detected by F-CR obtained in previous sections, i.e. $N = 32$ for F-CR and $K = 4$, $P = 12$, $\lambda = 1$ for E-CR, a new thousand random splits have been performed (80% for training and 20% for testing) and the final result has been calculated from the average of all of them. The evaluation of the proposed method on the GI4E database and his comparison with state of the art methods are summarized in Table 1. In order to compare the results, three accuracy levels have been defined: $e_{max} \leq 0.05$, $e_{max} \leq 0.1$ and $e_{max} \leq 0.25$.

The proposed method gains the best results for all accuracy measures $e_{max} \leq 0.05$, $e_{max} \leq 0.1$ and $e_{max} \leq 0.25$. Most of the methods in the comparison do not use training strategies but they employ alternative ad-hoc designed algorithms. More specifically, the second best $e_{max} \leq 0.05$ of them [Gou et al. 2016] is based on a similar cascaded regression strategy trying to use eye synthetic images in order to augment the training data. This method presents

Table 1: Accuracy comparison for pupil center localization on the GI4E Database

Method	$e_{max} \leq 0.05$	$e_{max} \leq 0.1$	$e_{max} \leq 0.25$
Timm2011[Timm and Barth 2011]	92.40%	96.00%	97.50%
Baek2013[Baek et al. 2013]	79.50%	88.00%	-
Villanueva2013[Villanueva et al. 2013]	93.90%	97.30%	98.50%
Zhang2016[Zhang et al. 2016]	97.90%	99.60%	99.99%
Gou2016[Gou et al. 2016]	98.20%	99.80%	99.80%
Gou2017[Gou et al. 2017]	94.20%	99.10%	99.80%
Ours	**99.14%**	**99.99%**	**100%**

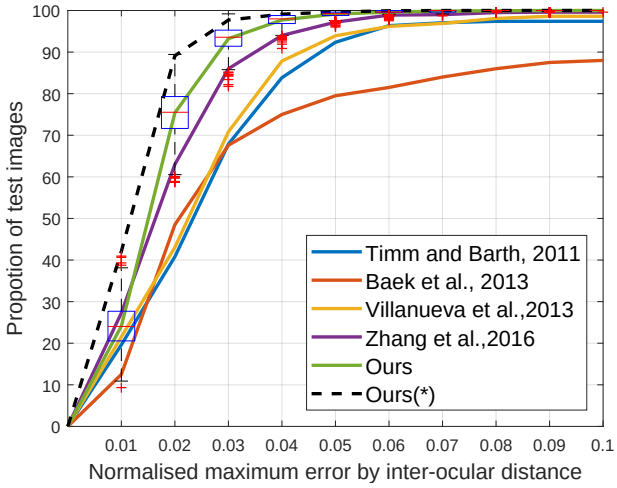

Figure 10: Accuracy curves of the proposed and the state of the art methods on the GI4E database. Box-plots representing the distribution of the thousand splits used in our method are included.

the second best results for $e_{max} \leq 0.05$ in their first version however the accuracy decreases for a later work of the same authors [Gou et al. 2017]. The third best $e_{max} \leq 0.05$ method [Zhang et al. 2016] consists in calculating the isophote curves, i.e. curves of equal intensity, of the gradient image assuming that the large contrast in the pupil or iris area will permit a rough estimation of the center by using a voting procedure. Additional stages are required in the method in order to achieve a more accurate detection of the center for the GI4E such as a selective oriented gradient filter, i.e. SOG filter, energy maps post processing and iris radius constraints among others.

However, a further accuracy analysis is made as shown in Figure 10 in which accuracy curves of the proposed and the state of the art methods are compared on the GI4E database. Gou2016[Gou et al. 2016] and Gou2017[Gou et al. 2017] results are not shown because the data are not publicly available. Furthermore, box-plots representing the distribution of the thousand splits used in our method are included, showing that our method has an outstanding improvement capability. The curve ours(*) shows the case in which E-CR with $K = 4$, $P = 12$ and $\lambda = 1$ is selected and the eye corners are ideally detected, i.e. F-CR is obviated and ground-truth is used instead. From the graph the potential benefits of the proposed method in terms of accuracy can be easily appreciated.

4 CONCLUSIONS

We have presented a two stage procedure based on SDM and R-CR-C methods by which to detect the pupil center. Two cascaded regressors have been trained, tested and optimized. The first one, F-CR is optimized to detect a set of 32 landmarks which corresponds to the facial points while the second one, E-CR is optimized to detect the inner and outer eye corners and the pupil centers. The proposed method is successfully applied to one of the state of art databases such as GI4E.

The results achieved by our method using SDM show promising outcomes as well as great capacity for improvement as it has been shown in our preliminary results. Since it is based on training, our method presents a significant adaptability power to alternative databases and robustness. As future work we propose to pursue in alternative training ways in order to improve the accuracy and robustness of the face detection stage. More specifically, a careful study of different face detectors used to initialize the first cascaded regressor is proposed.

ACKNOWLEDGMENTS

We would like to acknowledge the Spanish Ministry of Economy, Industry and Competitiveness for their support under Contract TIN2014-52897-R and TIN2017-84388-R.

We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan X Pascal GPU used for this research.

REFERENCES

Seung-Jin Baek, Kang-A Choi, Chunfei Ma, Young-Hyun Kim, and Sung-Jea Ko. 2013. Eyeball model-based iris center localization for visible image-based eye-gaze tracking systems. *IEEE Transactions on Consumer Electronics* 59, 2 (2013), 415–421.

Peter N Belhumeur, David W Jacobs, David J Kriegman, and Neeraj Kumar. 2013. Localizing parts of faces using a consensus of exemplars. *IEEE transactions on pattern analysis and machine intelligence* 35, 12 (2013), 2930–2940.

Leo Breiman. 2001. Random Forests. *Mach. Learn.* 45, 1 (Oct. 2001), 5–32. https://doi.org/10.1023/A:1010933404324

J.J. Cerrolaza, A. Villanueva, and R. Cabeza. 2012a. Hierarchical statistical shape models of multiobject anatomical structures: Application to brain MRI. *IEEE Transactions on Medical Imaging* 31, 3 (2012), 713–724. https://doi.org/10.1109/TMI.2011.2175940

Juan J. Cerrolaza, Arantxa Villanueva, and Rafael Cabeza. 2012b. Study of Polynomial Mapping Functions in Video-Oculography Eye Trackers. *ACM Trans. Comput.-Hum. Interact.* 19, 2, Article 10 (July 2012), 25 pages. https://doi.org/10.1145/2240156.2240158

Zhenhua Feng. 2016. SDM: A Matlab implementation of Supervised Descent Method for facial landmark detection and tracking. https://github.com/FengZhenhua/Supervised-Descent-Method. (2016).

Zhen-Hua Feng, Guosheng Hu, Josef Kittler, William Christmas, and Xiao-Jun Wu. 2015a. Cascaded collaborative regression for robust facial landmark detection trained using a mixture of synthetic and real images with dynamic weighting. *IEEE Transactions on Image Processing* 24, 11 (2015), 3425–3440.

Zhen-Hua Feng, Patrik Huber, Josef Kittler, William Christmas, and Xiao-Jun Wu. 2015b. Random cascaded-regression copse for robust facial landmark detection. *IEEE Signal Processing Letters* 22, 1 (2015), 76–80.

Onur Ferhat and Fernando Vilariño. 2016. Low Cost Eye Tracking: The Current Panorama. *Computational Intelligence and Neuroscience* 2016, 5 (2016), 2–14.

Wolfgang Fuhl, Thiago Santini, Gjergji Kasneci, and Enkelejda Kasneci. 2016. PupilNet: Convolutional Neural Networks for Robust Pupil Detection. *preprint* abs/1601.04902 (2016). arXiv:1601.04902 http://arxiv.org/abs/1601.04902

Colin Goodall. 1991. Procrustes methods in the statistical analysis of shape. *Journal of the Royal Statistical Society. Series B (Methodological)* (1991), 285–339.

Chao Gou, Y. Wu, Kang Wang, Fei-Yue Wang, and Q. Ji. 2016. Learning-by-synthesis for accurate eye detection. In *2016 23rd International Conference on Pattern Recognition (ICPR)*. 3362–3367. https://doi.org/10.1109/ICPR.2016.7900153

Chao Gou, Yue Wu, Kang Wang, Kunfeng Wang, Fei-Yue Wang, and Qiang Ji. 2017. A joint cascaded framework for simultaneous eye detection and eye state estimation. *Pattern Recognition* 67 (2017), 23–31. https://doi.org/10.1016/j.patcog.2017.01.023

Elias Daniel Guestrin and Moshe Eizenman. 2006. General theory of remote gaze estimation using the pupil center and corneal reflections. *IEEE Trans. Biomed. Engineering* 53, 6 (2006), 1124–1133. https://doi.org/10.1109/TBME.2005.863952

Dan Witzner Hansen and Qiang Ji. 2010. In the Eye of the Beholder: A Survey of Models for Eyes and Gaze. *IEEE Trans. Pattern Anal. Mach. Intell.* 32, 3 (March 2010), 478–500. https://doi.org/10.1109/TPAMI.2009.30

Patrik Huber. 2015. superviseddescent: A C++11 implementation of the supervised descent optimisation method. https://github.com/patrikhuber/superviseddescent. (2015).

Patrik Huber, Zhen-Hua Feng, William Christmas, Josef Kittler, and Matthias Ratsch. 2015. Fitting 3d morphable face models using local features. In *Image Processing (ICIP), 2015 IEEE International Conference on*. IEEE, 1195–1199.

Oliver Jesorsky, Klaus J Kirchberg, and Robert W Frischholz. 2001. Robust face detection using the hausdorff distance. In *International Conference on Audio-and Video-Based Biometric Person Authentication*. Springer, 90–95.

A. Kacete, J. Royan, R. Seguier, M. Collobert, and C. Soladie. 2016a. Real-time eye pupil localization using Hough regression forest. In *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, Vol. 00. 1–8. https://doi.org/10.1109/WACV.2016.7477666

Amine Kacete, Renaud Séguier, Michel Collobert, and Jérôme Royan. 2016b. Head Pose Free 3D Gaze Estimation Using RGB-D Camera. In *ICGIP*, SPIE (Ed.). Tokyo, Japan. https://hal.archives-ouvertes.fr/hal-01393594

Kyle Krafka, Aditya Khosla, Petr Kellnhofer, Harini Kannan, Suchendra Bhandarkar, Wojciech Matusik, and Antonio Torralba. 2016. Eye Tracking for Everyone. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Andoni Larumbe, Mikel Ariz, Jose J Bengoechea, Ruben Segura, Rafael Cabeza, and Arantxa Villanueva. 2017. Improved Strategies for HPE Employing Learning-By-Synthesis Approaches. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1545–1554.

Vuong Le, Jonathan Brandt, Zhe Lin, Lubomir Bourdev, and Thomas Huang. 2012. Interactive facial feature localization. *Computer Vision–ECCV 2012* (2012), 679–692.

Christos Sagonas, Epameinondas Antonakos, Georgios Tzimiropoulos, Stefanos Zafeiriou, and Maja Pantic. 2016. 300 faces in-the-wild challenge: Database and results. *Image and Vision Computing* 47 (2016), 3–18.

Christos Sagonas, Georgios Tzimiropoulos, Stefanos Zafeiriou, and Maja Pantic. 2013a. 300 faces in-the-wild challenge: The first facial landmark localization challenge. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*. 397–403.

Christos Sagonas, Georgios Tzimiropoulos, Stefanos Zafeiriou, and Maja Pantic. 2013b. A semi-automatic methodology for facial landmark annotation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*. 896–903.

Evangelos Skodras and Nikos Fakotakis. 2015. Precise Localization of Eye Centers in Low Resolution Color Images. *Image Vision Comput.* 36, C (April 2015), 51–60. https://doi.org/10.1016/j.imavis.2015.01.006

Fabian Timm and Erhardt Barth. 2011. Accurate Eye Centre Localisation by Means of Gradients. In *VISAPP*.

R. Valenti, N. Sebe, and T. Gevers. 2012. Combining Head Pose and Eye Location Information for Gaze Estimation. *IEEE Transactions on Image Processing* 21, 2 (2012), 802–815. https://ivi.fnwi.uva.nl/isis/publications/2012/ValentiTIP2012

Arantxa Villanueva, Victoria Ponz, Laura Sesma-Sanchez, Mikel Ariz, Sonia Porta, and Rafael Cabeza. 2013. Hybrid method based on topography for robust detection of iris center and eye corners. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)* 9, 4 (2013), 25.

Paul Viola and Michael Jones. 2001. Rapid object detection using a boosted cascade of simple features. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, Vol. 1. IEEE, I–I.

Xuehan Xiong and Fernando De la Torre. 2013. Supervised descent method and its applications to face alignment. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 532–539.

Wenhao Zhang, Melvyn L Smith, Lyndon N Smith, and Abdul Farooq. 2016. Eye center localization and gaze gesture recognition for human-computer interaction. *Journal of the Optical Society of America* 33, 3 (2016), 314–325.

Xucong Zhang, Yusuke Sugano, Mario Fritz, and Andreas Bulling. 2015. Appearance-Based Gaze Estimation in the Wild. *CoRR* abs/1504.02863 (2015).

Xiangxin Zhu and Deva Ramanan. 2012. Face detection, pose estimation, and landmark localization in the wild. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2879–2886.