A Method to Gaze Following Detection by Computer Vision Applied to Production Environments

Emannuell Dartora Cenzi, Marcelo Rudek

A Method to Gaze Following Detection by Computer Vision Applied to Production Environments

Emannuell Dartora Cenzi[1[0000-0003-4023-4728]] and Marcelo Rudek[1[0000-0002-6170-3370]]*

[1] Pontifícia Universidade Católica do Paraná PUCPR, Industrial and Systems Engineering Graduate Program – PPGEPS, Curitiba PR 80215-901, Brazil
emannuellcenzi@gmail.com , *marcelo.rudek@pucpr.br
*corresponding author

Abstract. The humans have the natural ability of following objects with the head and eyes and identify the relationship between those objects. This daily activity represents a challenge for computer vision systems. The procedure to identify the relationship between human eye gaze and the trackable objects is complex and demands several details. In this current paper we proposed a review of the main gazing following methods, identified the respective performance of them and also proposed an AI based method to estimate the gaze from 2D images based on head pose estimation. The main important details to be recovered from images are scene depth, head position and alignment and ocular rotation. In this approach we perform a track estimation of the gaze direction without the use of the eye position, and also, the face partial occlusion is considered in the analysis. The proposed approach allows low cost in processing with considerable accuracy at low complexity sceneries, because we don't need to extract the facial features. Gaze tracking is important to evaluate employees' attention to specific tasks in order to prevent accidents and improve work quality. The presented method improves the current knowing workflow by applying the head pose estimation instead face detection for training and inference. The promisors results are presented and open points are also discussed.

Keywords: gaze following, computer vision, artificial intelligence, gaze direction, artificial neural network

1 Introduction

Following the direction of gaze is an important task to understand the behavior in human-human and human-object interaction. The marketing and sales sectors of retail companies seek to understand the consumer behavior in relationship to the acquisition of goods, taking into account cognitive aspects such as visual and behavioral information of the consumer. In factories production lines, for example, the level of attention on different items or parts can be inferred based on human-object eye contact in real time, seeking to understand human interaction in the production process, measure the time spent on tasks and analyze productivity. For safety, in remote lifeproof systems (liveness detection), in education and training approaches to determine the level of attention of students on the teacher.

An analysis of where they direct their gaze is an example of this. In this article we propose an analysis of the datasets and previous approaches to propose an end-to-end solution and we apply a study case with the objective of estimating where people are looking.

In order to avoid any misunderstanding, in this work we use the term "focus of attention" to refer to the direction in which the person is looking (the object of gaze fixation), called gaze following [1].

Shared attention is present in every part in our daily life and it can be observed in almost all social interactions. Human beings have the ability to follow another person's gaze naturally. Although this ability is of vital importance and natural to humans, for computer vision it is extremely challenging for three reasons [2]:

1. Deducting the point of view requires information on the depth of the scene, the pose of the head and the movement of the eyeball. However, inferring the depth of the scene with a monocular image is complex and can have a high computational cost. In addition, estimating the position of the head and the movement of the eyeball is often not possible due to occlusion.

2. There may be ambiguity in the focus of gaze estimates, as in fig. 2 (a).

3. Gaze following involves understanding the geometric relationship between the target person and the other objects in the scene, as well as understanding the content of the scene, which is a difficult task.

2 Related works

Although important, a few works in the field of computer vision have explored the subject by limiting the scope of the problem and restricting situations to scenes of people looking at each other [3], in controlled environments or using multiple image sources to determine the target [4]. The literature review was based on the methodology of [5] where the main identified works works that use eye-tracking techniques and can be applied to controlled scenarios such as human-computer interaction [6, 7].

Recent works have explored the problem of estimating gaze direction in different ways. Some previously explored approaches are highlighted:

- In [3] it was sought to determine whether or not people are looking at each other on television videos.
- An eye-tracking technique that consists of tracking the movements of the eyeball was applied by [8], which predicts the next object of attention in order to improve the recognition of actions.
- Through tracking the eye by monitoring the position of the iris, a set of annotations of facial points is made to perform eye tracking [1].
- Estimating the direction of gaze with only the position of the head, but without the specific target point [9].

- Given an image containing several people, gaze direction is estimated without environmental restrictions, and the target point of the gaze is determined by detecting salient points [4].
- Other works propose approaches for videos in [10, 11].
- 3D images contain information about the depth of the scene and some highlighting contributions are presented in [12, 13, 14].

The problem of gaze following was explored by [15] to infer the attention shared among two or more individuals over another object or human.

The current state of the art for monocular images consists of the use of deep neural networks (deep learning). It is a two-stage neural network to predict the gaze direction of the person selected in the scene, where in the first stage, only the image of the head (crop) and its position in the scene are necessary to perform the gaze direction prediction. Then, possible vectors are generated which are used to characterize the distribution of the points of gaze without considering the content of the scene [2].

Shared attention is present in daily life and can be observed in social interactions [15]. Through videos extracted from public television programs with scenes of social interaction, it was possible to automatically determine shared attention. It is a phenomenon where two or more people simultaneously look at a target in the scene that must be analyzed as a third person (outside the scene). The solution was a proposal for a space-time neural network to detect shared attention in videos.

Different approaches have used eye tracking to successfully determine gaze direction [6, 7, 16, 17]. However, [4] observed that these techniques are severely affected by self-occlusion generated by the individual's positioning in the scene. In this way, they propose a neural network architecture divided into two paths with the objective of determining the focus of people's eyes on everyday images without the restriction of the environment. The detection pipeline is divided into two independent neural networks. The image follows a path that discovers the salient points (Saliency Pathway), assuming that they are targets of attention in order to highlight and emphasize certain objects that people tend to look at. The other path (Gaze Pathway) uses an image of the detected face and its position in a neural network model to determine the direction of gaze from the position of the head [4].

The limitation of the proposal by [4] is that the object in the region of eye focus may not always be salient, revealing a difficulty in finding the end point of the gaze directly through salience algorithms. According to [18], the detection of salient points consists of highlighting regions of an image, and can be applied in the segmentation of images and videos, image compression, action recognition, and video summarization, among others.

Given that the works proposed to use the position of the head and the movement of the eyeball separately to determine gaze direction, [12] proposed an approach with multiple cameras using two separate deep neural networks, one for the prediction of the position of the head and another for the movements of the eyeball without using facial points. To connect the two layers, the “gaze transformation” layer was created. The output presents a vector composed of the starting point (eye coordinates) and the direction of the gaze to the target point in 3D space.

The current state of the art for monocular images [2] consists in the use of deep learning. Advances were made by [4] since 2015 with a new approach to determine the

focus of gaze suggesting a two-step method inspired by the human behavior of gaze following. Especially when a person outside the scene (third person) analyzes it with the objective of estimating the person's gaze direction, the estimation of the target of the gaze is made based on the image of the head. Thus, the authors propose to use the image of the head and its position in the image to determine the gaze direction in the first stage, and then the gaze direction field is "coded" in three different scales. In the second stage, the gaze direction vectors generated in the previous step are linked with the image to generate a heat map where the point with the highest value represents the target of the gaze.

Heatmap regression is a technique used in many applications such as pose estimation [8] that uses regression models to propose a cloud of possible points to determine the pose of people in videos. In the same way, the point of gaze is predicted based on a heatmap of the content of the scene over the multiple estimated gaze direction vectors.

2.1 Gaze following and Head Pose Estimation

Estimating the position of a person's head is a problem that has a wide range of applications, such as assisting in gaze following, defining attention, adjusting 3D models (animations or characterization of characters) to the video, performing face alignment, monitoring driver behavior, or associating with other techniques in view of the fact that it is closely related to everyday problems such as gaze following.

Previous approaches have generally used facial point estimation to make a correspondence from 2D to 3D. However [19, 20] argue that relying entirely on the performance of detecting facial points to estimate the position of the head is a fragile method, which involves some steps constructed in the form of a cascade, so if the first stage fails, consequently the rest will be affected. From the face detection, the markings of the reference points in the 2D image are estimated and adapted in an average 3D model of the human face, and with the camera parameters it is possible to calculate corrections to then make the correspondence between the 2D points and the 3D model.

Large face datasets [21, 22] and efficient methods with different approaches have previously been proposed to solve problems related to facial analysis, such as face detection [23, 24, 25, 26], face recognition [27], age estimation, detection of facial points and estimation of head position [19, 20].

The head pose estimation problem from a simple 2D image is resolved with the ResNet-50 multi-loss neural network architecture [19], where each loss has a classification and a regression corresponding individually to the three angles of yaw, pitch and roll. Given the position of the head, the product of the proposed method is a 3D vector that contains the yaw, pitch and roll angles. Estimating the pose of the head from an image basically requires learning to map between 2D and 3D spaces. Some methods use 3D images that contain depth information not present in 2D images that are the objective of these approaches.

3 Methodology

In this section it is presented the proposed methodology to determine the direction of gaze. This methodology is based on facial detection and head position estimation,

without using the points of the face (markers) in an end-to-end flow. The product of the method consists of a vector connecting the head and the focal point of gaze.

The proposal analyzes the union of face detection techniques and head pose estimation [19] with the gaze following technique proposed by [2]. We joined these methods as presented in fig. 1.

A close relationship was observed between the two techniques, even if for different purposes. Determining gaze direction necessarily goes through the facial detection process, as it does the head pose estimation that uses a cut-out image of the face to determine the position of the head (yaw, pitch, and roll angles).

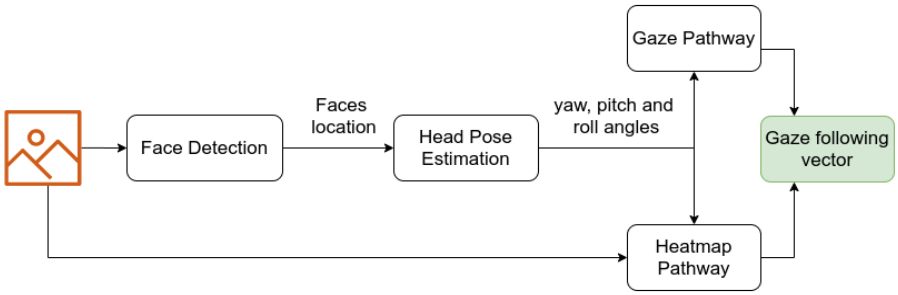

Fig. 1. Proposed detection pipeline.

3.1 Hardware Scheme Configuration

For prototyping, development and training, a computer equipped with Intel Core i5 3.0 Ghz with 4 threads, 16 GB of RAM and 1 NVIDIA 1660 GTX GPU with 6 GB of RAM was used. The operating system used was Ubuntu 16.04 LTS. This setup was powerful enough to run the inference tests and training.

4 Training and Testing Dataset

In order to progress in the problem of predicting the gaze direction of one or more people in images, specific datasets must be used. Proposed in 2015, the GazeFollow dataset [4] contains 122,143 images and 130,339 people, with the annotation from the center of the eyes to where the “annotator” believes that the person under analysis is looking, with up to 10 different possible targets of gaze. For this dataset the authors used only images with the aim of looking within the image.

The dataset consists of a selection of images from different sources, such as SUN [28], MS COCO [29], Actions 40 [30], PASCAL [31], ImageNet [32] and Places [33]. This composition of different sources resulted in a large and challenging dataset of images of people in day-to-day activities with diverse scenarios. In figure 2, three examples of people engaged in activities and their respective annotations can be seen.

Fig. 2. Examples of images from the GazeFollow dataset [4].

In figure 2, examples with one or multiple people can be seen from the GazeFollow dataset, where only one or some but not all have annotations of the direction of gaze (c), even with the target of the gaze within the image or multiple annotations in the same image (b).

Another recent dataset proposal, the Daily GazeFollowing Dataset [2], which analyzes videos of people in everyday scenes interacting with objects and the environment, such as offices and work environments, shared spaces in buildings, interaction between people, etc. The annotations were made by the people in the scene, so they are more reliable according to the authors.

Regarding the dataset, GazeFollowing will be used because it is broader. It is concluded that examples of gaze following in which it is not possible to detect the face are not useful for this study because they are outside the proposed pipeline. In this way, a solution is proposed to repopulate the dataset.

4.1 Repopulation of the GazeFollow Dataset

Divided between training and validation, the dataset proposed by [4] has a large number of images. The annotations maintained the same format, although among the annotations only the gaze vectors originating inside one of the detected face bounding boxes were maintained.

The proposed flow presented in fig. 3 implies the complete re-processing of the GazeFollow dataset. For each image, the faces are detected with the Dual Shot Face Detector [34] and for each face detected it checks if there is any annotation (ground truth) for this image with the point of origin of the gaze inside the bounding box of the face. If the point of origin of the gaze is related to the detected face, the estimation of the head position is made and the result with the pitch, yaw and roll angles is included in the original annotation of the dataset.

Among the images in which it was possible to detect the face and the origin of gaze direction vector annotations, the head pose estimation was processed. Some examples can be seen in figure 8. The result of the head pose estimation has been linked to the dataset as new parameters. These are numbers referring to pitch, yaw and roll angles and table 1 shows the number of modified images in dataset.

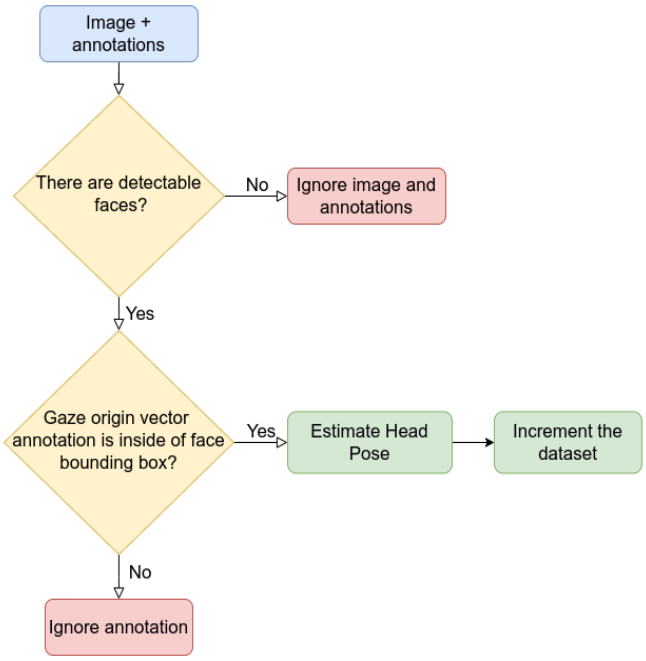

Fig. 3. Flow to repopulate the dataset with head pose estimation. Image elaborated by the author (2020).

Some examples where one or more faces were detected but there are no annotations of the direction of gaze. As presented in Fig. 4, the cyan colored bounding box represent the face annotation and magenta is the not annotated ones.

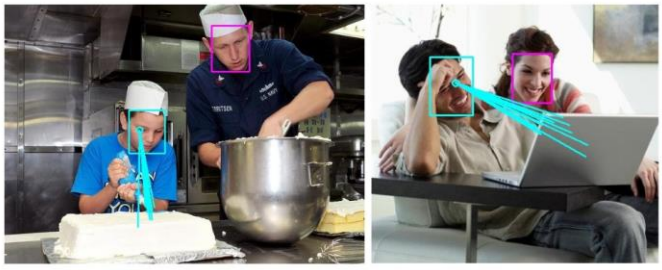

Fig. 4. Example of face detection without annotation of the gaze directions. Adapted from the *GazeFollow Dataset* [4].

Table 1. Number of images in the datasets.

Dataset	Original	Repopulated
Training	119.125	98.508
Test	4.782	3.883

As presented in table 1, from 119,125 images for training, 17.3% were discarded because they did not meet the requirements in the repopulation process, whereas in the test dataset after processing, 18.79% of the images were discarded. The images that met the requirements shown in figure 3 were included in the new dataset with the head position estimation.

5 Training strategy with head pose estimation

The network input parameters are originally divided into three parts: head image, head position and the original image. The head and the original image are scaled to 224 x 224 pixels. After the repopulation of the dataset, there is a new parameter with three positions: pitch, yaw and roll.

Used in training, the new parameter "head_pose" has the three angles and is the product of the process of estimating the head position for each face annotated in the dataset.

Our approach does not use head image to determine the direction of the gaze. We changed the input of the neural network, removing the head image from the training process, and fusion layer. The fusion layer combine the eye position and head pose estimation into a sequential linear operation with ReLU activation function.

The outputs of the network remain with the original implementation and consist of two parts: direction of gaze and visual attention. The direction of gaze is the normalized vector from the head position to the point of gaze and the visual attention is a 56 × 56 pixel heatmap whose values indicate the probability of being the point of gaze.

The function that represents the loss is:

$$\iota \boldsymbol{d} = \mathbf{1} - \frac{\{\boldsymbol{d}, \hat{\boldsymbol{d}}\}}{|\boldsymbol{d}, \hat{\boldsymbol{d}}|} \tag{1}$$

Where d represents the ground truth and $\hat{d}$ is the result of the prediction of the direction of gaze.

To calculate the loss in the heatmap regression, the originally implemented function BCE Loss (Binary Cross Entropy loss) is used, which creates a criterion that measures the binary cross entropy between the ground truth and the output.

Loss functions such as BCE Loss are typically used within the gradient drop, which is an iterative structure for moving parameters (or coefficients) to optimal values. Cross entropy describes the loss between two probability distributions [37].

The implementation is based on the PyTorch framework. To extract the features of the head image, the pre-trained ResNet-50 neural network was used by [2] with the Imagenet dataset [32]. The heatmap is introduced in the training after converging the first training layer, and is generated from the target point of the gaze annotated in the ground truth positioned in the center of the kernel of a Gaussian convolution operation. A sigmoid function was implemented in the output for activation of the heatmap.

Keeping in mind that we are working with the orientation of the head in space, data augmentation functions can be detrimental. Randomly transforming the head pose estimation data (head angles) can cause these to be lost and can confuse the model

during training. Therefore, no techniques were used to expand the dataset during training.

6 Metrics and validation

The classification task can be considered binary when the data input must be classified in only one class [38]. Our assessment compares the annotations (ground truth) of the dataset with the distribution of predictions. Evaluating the performance of different approaches and algorithms to determine gaze direction requires specific metrics for the problem. The following metrics are presented [2]:

- Area Under Curve (AUC) refers to the area under the ROC curve. The higher the value, the better the result.
- L2 or Dist is the Euclidean distance between the focus point of the gaze predicted by the network and the average of the annotation (ground truth).
- Angular error, to be calculated, requires tracing the annotated vectors and the result of the prediction to then calculate the error between them, corresponding to the average between the points of gaze.

Even though the processing cost and time for inference are not the focus of the research, the average time to perform an inference must be taken into consideration when there is a need to embed the solution on IoT devices, or in scenarios with hardware limitation.

7 Results

Hereinafter the results obtained from the implementation of the model to determine gaze direction [2] are presented, as well as the results of the method HopeNet [19] to estimate the position and the angle of the head. Both implementations are available in the Github project repositories [35, 36].

For validation, a 32-second video was recorded at a rate of 30 frames per second (Logitech 12mp webcam, 1080 x 720 resolution) to enable comparison and analyze the performance of the application of [2] without annotating the origin of gaze to validate the efficiency of the proposed change. The video has a total of 960 frames, all with the face exposed, without occlusion.

The objective of this experiment is to understand the importance of the face detector in the first stage of the inference pipeline.

Figure 6 on the left shows a positive result of facial detection and heatmap regression to determine gaze direction. On the right side of Fig.6, the heatmap can be seen linked with the original image to monitor the output of the proposed model. When it is not possible to detect the face (false negative) with the face detector model proposed in this approach, gaze direction cannot be inferred, as in the example in Fig.7. As shown in figure 7 (b), the face detection problem directly affects the next stages of the pipeline

to determine the gaze direction like as in figure 7(a), and exposes the importance of a good face detector to estimate the point of attention in real time.

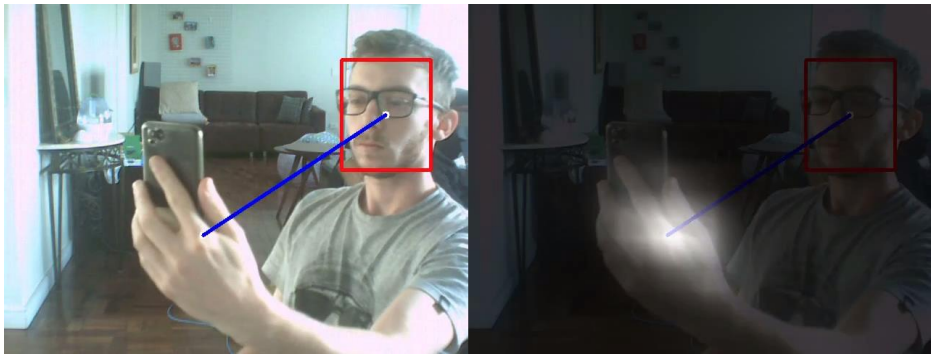

Fig. 6. Example of success in the process of facial detection and gaze following.

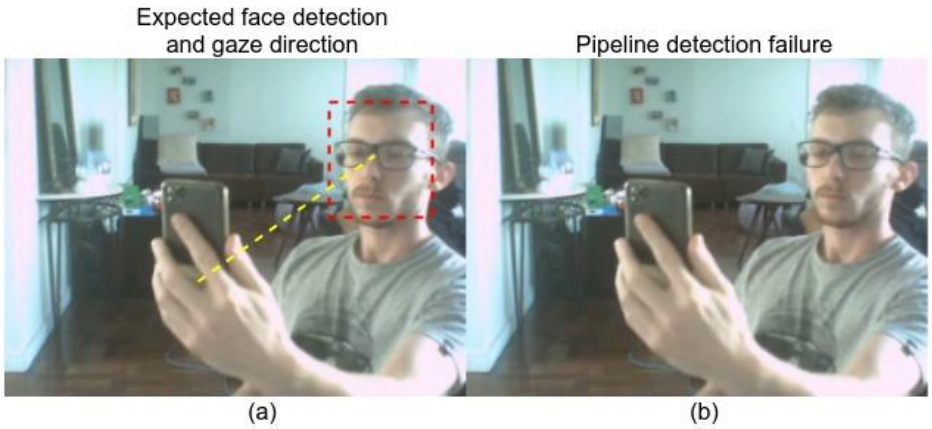

Fig. 7. Example of unsuccessful in face detection and breaking the gaze following frames sequence of the pipeline. (a) expected detection (b) no-detection in consecutive frames.

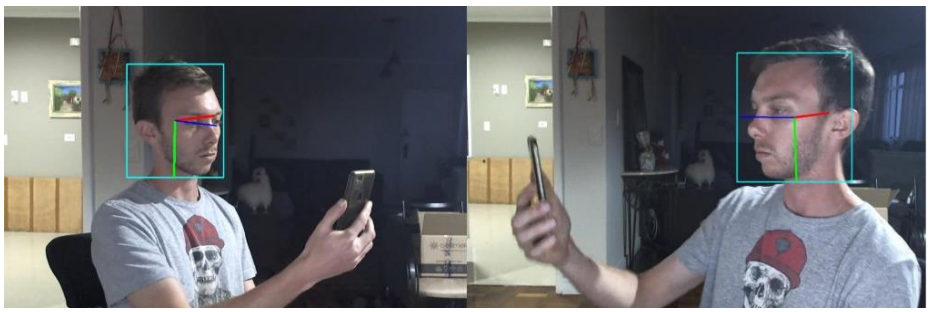

Fig. 8. Application of head pose estimation in the testing video.

Gaze following and head pose estimation are not the same techniques and no studies were found using both techniques directly coupled. The HopeNet head pose estimation [19] partial tests were performed based on the pre-trained model made available by the authors in the Github repository [36].

As presented by [19] and verified in figure 8, HopeNet stood out for its high capacity to determine the head angulation in a wide range of positions and partial occlusion of the face considering the fact that the proposed architecture does not use face points to determine the angulation of the head.

7.1 Results from proposed method

The results of the experiments are listed in table 2. Due to changes in the dataset to include the head position estimate as presented in section 4. The neural network model proposed by [2] was also changed as presented in session 5.

Table 2. Resultados de treinamento com método proposto.

Methods	AUC	Distance L2	Angular Error
Recases *et al.* [4]	0.881	0.175	22.5
Lian *et al.* [2]	0.903	0.156	17.6
Our method	0.911	0.179	20.71

Table 2, shows that our model outperforms [2] and [4] on AUC evaluation metric, and approximates the average of the Euclidean distance between the direction of the gaze direction and ground truth annotation, and average angular error of the direction.
The file with the saved model has an average size 47% smaller (116.5 Mb) than the model made available by [2] (223.4 Mb) for download.

So far, our model does not completely surpass the results obtained by [2], but it presents the model's viability and discusses the use of the gaze direction estimation technique with the head position estimation. Potentially applied in factories environments for production measurement [39], ergonomics systems, classrooms or online teaching softwares, online meetings, or in parallel with other computer vision systems in everyday interactions, web applications and embedded systems.

8 Conclusion

Systems to determine the direction of gaze will be fundamental to improve the human experience when interacting with machines and robots, especially those that seek to imitate human behavior. Based on the results, it is observed that the proposed architecture has the potential to surpass the current state of the art results to determine gaze direction. The proposed method presented in this article has its own characteristics and consists of the connection between the techniques of face detection, head pose estimation and gaze following in a pipeline for inference and training from end to end. Following the proposed flow does not necessarily imply obtaining the same results, given that there is variation in the results of the head position estimate when

repopulating the dataset. This work seeks to determine the gaze direction in 2D images, through a new methodology for training directly with the head pose estimation.

Also, we built a new data structure to the dataset. Changes in the neural network for training will in future be published on github project repository, along with the new dataset. New discussions and experiments, such as training parameters and transformation functions between layers are necessary to evolve the presented methodology to make it more robust.

References

1. Xiong, Xuehan et al. Eye gaze tracking using an RGBD camera: a comparison with a RGB solution. In: Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication. ACM, 2014. p. 1113-1121.
2. Lian, Dongze; Yu, Zehao; Gao, Shenghua. Believe It or Not, We Know What You Are Looking At!. In: Asian Conference on Computer Vision. Springer, Cham, 2018. p. 35-50.
3. Marín-Jiménez, Manuel Jesús et al. Detecting people looking at each other in videos. International Journal of Computer Vision, v. 106, n. 3, p. 282-296, 2014.
4. Recasens, Adria et al. Where are they looking?. In: Advances in Neural Information Processing Systems. 2015. p. 199-207.
5. Reche, A. Y. U. ; Canciglieri Junior, O. ; Rudek, Marcelo ; Estorilio, C. C. A. . Integrated Product Development Process and Green Supply Chain Management: contributions, limitations and applications. JOURNAL OF CLEANER PRODUCTION, p. 119429-119459, 2019
6. Krafka, Kyle et al. Eye tracking for everyone. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2016. p. 2176-2184..
7. Aung, Arkar Min; Ramakrishnan, Anand; Whitehill, Jacob R. Who Are They Looking At? Automatic Eye Gaze Following for Classroom Observation Video Analysis. International Educational Data Mining Society, 2018.
8. Fathi, Alireza; Li, Yin; Rehg, James M. Learning to recognize daily actions using gaze. In: European Conference on Computer Vision. Springer, Berlin, Heidelberg, 2012. p. 314-327.
9. Pfister, Tomas; Charles, James; Zisserman, Andrew. Flowing convnets for human pose estimation in videos. In: Proceedings of the IEEE International Conference on Computer Vision. 2015. p. 1913-1921.
10. Recasens, Adria et al. Following gaze in video. In: Proceedings of the IEEE International Conference on Computer Vision. 2017. p. 1435-1443.
11. Mukherjee, Sankha S.; Robertson, Neil Martin. Deep head pose: Gaze-direction estimation in multimodal video. IEEE Transactions on Multimedia, v. 17, n. 11, p. 2094-2107, 2015.
12. Zhu, Wangjiang; Deng, Haoping. Monocular free-head 3D gaze tracking with deep learning and geometry constraints. In: Proceedings of the IEEE International Conference on Computer Vision. 2017. p. 3143-3152.
13. Parks, Daniel; Borji, Ali; Itti, Laurent. Augmented saliency model using automatic 3d head pose detection and learned gaze following in natural scenes. Vision research, v. 116, p. 113-126, 2015.
14. Mora, Kenneth Alberto Funes; Odobez, Jean-Marc. Person independent 3d gaze estimation from remote rgb-d cameras. In: 2013 IEEE International Conference on Image Processing. IEEE, 2013. p. 2787-2791.
15. Fan, Lifeng et al. Inferring shared attention in social scene videos. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018. p. 6460-6468.

16. Vicente, Francisco et al. Driver gaze tracking and eyes off the road detection system. IEEE Transactions on Intelligent Transportation Systems, v. 16, n. 4, p. 2014-2027, 2015.
17. Wang, Kang; Wang, Shen; JI, Qiang. Deep eye fixation map learning for calibration-free eye gaze tracking. In: Proceedings of the ninth biennial ACM symposium on eye tracking research & applications. ACM, 2016. p. 47-55.
18. Cong, Runmin et al. Review of visual saliency detection with comprehensive information. IEEE Transactions on Circuits and Systems for Video Technology, 2018.
19. Ruiz, Nataniel; Chong, Eunji; Rehg, James M. Fine-grained head pose estimation without keypoints. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. 2018. p. 2074-2083.
20. Yang, Tsun-Yi et al. FSA-Net: Learning Fine-Grained Structure Aggregation for Head Pose Estimation from a Single Image. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2019. p. 1087-1096.
21. Yang, Shuo, et al. "Wider face: A face detection benchmark." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.
22. Vidit Jain and Erik Learned-Miller. Fddb: A benchmark for face detection in unconstrained settings. Technical report, Technical Report UM-CS-2010-009, University of Massachusetts, Amherst, 2010.
23. Yu Chen, Ying Tai, Xiaoming Liu, Chunhua Shen, and Jian Yang. Fsrnet: End-to-end learning face super-resolution with facial priors. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018.
24. Cheng Chi, Shifeng Zhang, Junliang Xing, Zhen Lei, Stan Z Li, and Xudong Zou. Selective refinement network for high performance face detection. In Proceedings of Association for the Advancement of Artificial Intelligence (AAAI), 2019.
25. Haoxiang Li, Zhe Lin, Xiaohui Shen, Jonathan Brandt, and Gang Hua. A convolutional neural network cascade for face detection. In Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015.
26. Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In Proceedings of Advances in Neural Information Processing Systems (NIPS), 2015.
27. Jian Yang, Lei Luo, Jianjun Qian, Ying Tai, Fanlong Zhang, and Yong Xu. Nuclear norm based matrix regression with applications to face recognition with occlusion and illumination changes. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), 39(1):156–171, 2017.
28. Xiao, J., Hays, J., Ehinger, K.A., Oliva, A., Torralba, A.: Sun database: Large-scale scene recognition from abbey to zoo. In: Computer vision and pattern recognition (CVPR), 2010 IEEE conference on. pp. 3485–3492. IEEE (2010).
29. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Doll´ar, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: European conference on computer vision. pp. 740–755. Springer (2014).
30. Yao, B., Jiang, X., Khosla, A., Lin, A.L., Guibas, L., Fei-Fei, L.: Human action recognition by learning bases of action attributes and parts. In: Computer Vision (ICCV), 2011 IEEE International Conference on. pp. 1331–1338. IEEE (2011).
31. Everingham, M., Van Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The pascal visual object classes (voc) challenge. International journal of computer vision 88(2), 303–338 (2010).
32. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al.: Imagenet large scale visual recognition challenge. International Journal of Computer Vision 115(3), 211–252 (2015).
33. Zhou, B., Lapedriza, A., Xiao, J., Torralba, A., Oliva, A.: Learning deep features for scene recognition using places database. In: Advances in neural information processing systems. pp. 487–495 (2014).

34. Li, Jian, et al. "DSFD: dual shot face detector." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2019.
35. GazeFollowing Repository - Github, https://github.com/svip-lab/GazeFollowing, last accessed 2020/03/02.
36. Deep head pose Hopenet - Github, https://github.com/natanielruiz/deep-head-pose, last accessed 2020/03/02.
37. Goodfellow, Ian; Bengio, Yoshua; Courville, Aaron. Deep learning. MIT press, 2016.
38. Sokolova, Marina; Lapalme, Guy. A systematic analysis of performance measures for classification tasks. Information processing & management, v. 45, n. 4, p. 427-437, 2009.
39. Silva, R. L. ; Rudek, M. ; Sjeika, A. ; Canciglieri Junior, O. . Machine Vision Systems for Industrial Quality Control Inspections. IFIP ADVANCES IN INFORMATION AND COMMUNICATION TECHNOLOGY, v. 540, p. 631-641, 2018.