Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Appearance-based gaze estimation using deep features and random forest regression

Yafei Wang[a,b], Tianyi Shen[b], Guoliang Yuan[b], Jiming Bian[a], Xianping Fu[b,*]

[a] *School of Physics and Optoelectronic Engineering, Dalian University of Technology, Dalian 116024, China*
[b] *Information Science and Technology College, Dalian Maritime University, Dalian 116026, China*

ARTICLE INFO

Article history:
Received 22 March 2016
Revised 26 July 2016
Accepted 29 July 2016
Available online xxx

Keywords:
Appearance
Gaze estimation
Deep features
Random forest
CNN

ABSTRACT

Conventional appearance-based gaze estimation methods employ local or global features as eye gaze appearance descriptor. But these methods don't work well under natural light with free head movement. To solve this problem, we present an appearance-based gaze estimation method using deep feature representation and feature forest regression. The deep feature is learned through hierarchical extraction of deep Convolutional Neural Network (CNN). And random forest regression with cluster-to-classify node splitting rules is used to take advantage of data distribution in sparse feature space. Experimental results demonstrate that the deep feature has a better performance than local features on calibrated gaze regression. The combination of deep features and random forest regression provides an effective solution for gaze estimation in a natural environment.

1. Introduction

The eye gaze plays a significant role in understanding human attention, feeling and mind. It is essential for many multimedia applications such as cognitive processes analysis and human-computer interaction [1]. Although many non-intrusive gaze tracking systems are proposed [2], it is still hard to get accurate gaze estimation when using one single web camera under natural light with free head movement.

The gaze estimation methods are divided into two categories, feature-based methods and appearance-based methods. Feature-based methods typically depend on pupil detection with the light sources' reflections on the cornea. A map from pupil center of geometry model to the gaze calibration points is established by calculating eye movement features. These methods could achieve very high accuracy (error less than 1°), but the function of calibrated regression fluctuate strongly due to free head movement [3]. Instead of focusing on geometric mapping, appearance-based methods treat the cropped eye image as a point representation in high dimensional space and learn the mapping relation from this point in given feature space to screen coordinates. Thus, the gaze estimation of new input eye images can be acquired by regression model trained by all exited image data, therefore, some latent gaze features can be implicitly modeled.

Traditional appearance-based gaze estimation methods were proposed using intensity feature, histogram information or gradient information of the whole eye image as input data. Recently, there have been a lot of novel approaches to appearance description. Lu et al. [4] introduced Grid feature, which is a 15-dimension feature calculating the sum ratio of pixel intensity in separated blocks. It has a high computational complexity combining with adaptive linear regression. But the method requires a fixed head pose. Building on Lu's work, Mora and Odobez [5] took both left and right eye images into consideration and put forward couple adaptive linear regression to eliminate gaze error between both eyes. Wang et al. [6] proposed a coarse-to-fine gaze estimation with TOP (Topology-preserving) feature, which is a sparse feature selection learning based on dense local feature. The well topological structure is kept in manifold space,which improves the features representative ability of eye image. However, these proposals are not effective with free head movement.

To solve this problem, Mora and Odobez [7] estimated gaze angle with head pose estimation using RGB-D data. In their method, a 3D face model was built upon the multimodal Kinect data, and gaze direction in the head coordinate system was determined by [4] method on the eye appearance. Although their method provides robust and accurate head pose tracking, the overall gaze estimation accuracy is relatively low (around 10°).] Lai et al. [8] took advantage of approximate random forest to build a

[*] Corresponding author.
E-mail addresses: wangyafei@dlmu.edu.cn (Y. Wang), fxp@dlmu.edu.cn (X. Fu).

http://dx.doi.org/10.1016/j.knosys.2016.07.038
0950-7051/

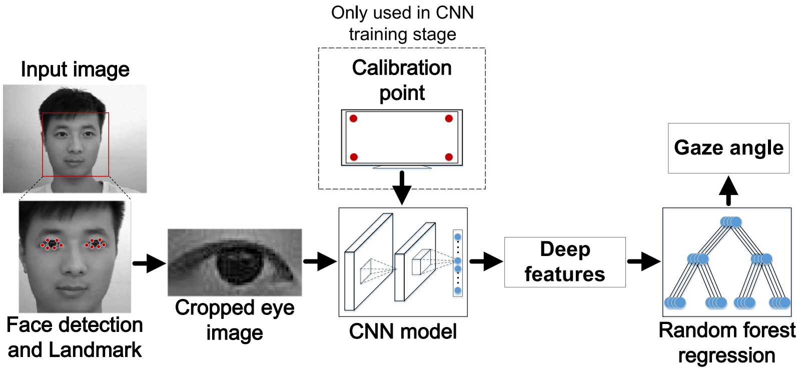

Fig. 1. Overview of proposed gaze estimation method.

5-dimension joint head pose and eye gaze space model, which performed well in frontal head motion. But this method relies too much on the accuracy of head pose estimation and needs fine configuration of head pose and eye gaze. Lu and Chen [9] employed auto-encoder to learn a sparse codebook of eye images and used spatial pyramid pooling to get eye features which are sparse coefficients reconstructing on the basis of codebook. This patch-based features improve the gaze estimation accuracy without training by the same person during the different sessions. Zhang et al. [10] took no account of head pose in gaze estimation and realized the appearance-based gaze estimation using CNN immediately to learn the mapping between eye image and gaze coordinate. They concatenated three-dimensional head pose in the last hidden layer with the output of fully connected layer. The change of original LeNet slightly improves the performance of CNN on gaze estimation with free head movement.

We focus on the gaze tracking system using one single web camera under natural light and allowing free head movement. To address this issue, we present a novel gaze estimation method combining deep feature extraction and feature forest regression. Unlike previous appearance-based gaze estimation methods, the proposed method extracts deep features from deep CNN to predict eye gaze direction. The CNN is used to classify eye images into calibrated fixations index through multi-scale convolutions and pooling with last hidden layer as deep features. The deep features have spare representation in characterize image and shown significant improvement on classification method.

Having extracted the deep features, regression forest is used to find the direct correlation between feature space and gaze direction. During forest node splitting, we firstly minimize the squared error loss by cluster and then split the node by binary classification for better partitions in deep feature space. The applying of fine-tuned cluster algorithm and SVM (Support Vector Machine) classification has a positive impact on the error reduction of the child nodes in gaze regression since it averages the predicted gaze cluster. The proposed method works well on eye images dataset under the condition of natural light and free head motion. Deep features regression forest model shows certain robustness to occlusion.

The paper is structured as follows: Section 2 gives the deep feature learning method and random forest regression for deep feature. Experimental results are given in Section 3, while traditional methods are used as a contrast. Additionally, the test result within our images dataset are also analyzed. Finally, Section 4 draws conclusions and gives some directions for future work.

2. Proposed method

2.1. Overall model

The overall framework proposed is shown in Fig. 1. In order to verify our method on individual images, an offline training and testing eye images dataset is built by cropping eye region from its localization of facial landmarks. First, the subject's face which always appears fully in the field of view is detected using modified version of Viola-Jones method [11]. Once the face region which takes the bounding box (Fig. 1) is localized, next step is to obtain the eye region. In our pipeline, a SDM (Supervised Descent Method) [12] facial landmark detector is used to find the eye corners and other fiducial points. The selected landmarks designating the eye region are shown as red dots in Fig. 1. This algorithm satisfies the eye region localization due to its two important characteristics which have been evaluated on "face in the wild" datasets [12]: (1) it is robust under natural light with different illumination conditions and (2) its outcome is significantly fast and accurate for face align in the wild with large head rotary movement.

Afterwards, all images in which the cropped eye images contain background region are discarded, which happened in about 4% of all cases. In this manner, the images dataset delivers eye images without background noise for the evaluation of appearance-based estimation function.

In this paper, the appearance variation of appearance-based estimation function to inferring gaze is handled in two degrees of freedom, in other words, our gaze estimator outputs two dimension gaze angle vector. General framework of our method contains cascaded stages, deep feature representation and feature regression.

Firstly, learning deep feature for eye gaze appearance is to set up a CNN deep feature space. Based on sparse calibration [6] which means using less training points during gaze calibration, the eye images can be divided into n classes which predict same value with calibration points. The CNN with several convolution layers and max-pooling layers is built to indicate classes by last soft-max layer. Deep feature is learned as the last hidden layer constitute the sparse feature space after extraction cascade of network.

Secondly, training regression forest model for deep feature is utilized to predict gaze direction. While regression forest refers to utilize many decision trees with respective random drawn training samples at leaf node, regression forest benefits by its ensemble voting output. The exert of regression forest seems more like partitions in the given feature space. In order to reduce the error loss, at the split of each node, it is necessary to find clusters of training data at current node and determine its classification of the found clusters. The cluster algorithm preserves that the found clusters are