Visualizing Prediction Correctness of Eye Tracking Classifiers

Martin H.U. Prinzler[1,3], Christoph Schröder[2], Sahar Mahdie Klim Al Zaidawi[1], Gabriel Zachmann[2], Sebastian Maneth[1]

[1] Database Lab, University of Bremen; [2] Institute for Computer Graphics and Virtual Reality, University of Bremen; [3] Current Affiliation: Alfred-Wegener-Institut Helmholtz-Zentrum für Polar- und Meeresforschung

Classification via Eye Tracking Trajectories

Eye tracking data is often used for classification tasks. As example, we determine users based on their eye movements (see [1]). Our Method builds on the classification probabilities from fixations and saccades.

Segmentation: To divide the gaze trajectories into fixations and saccades, we implemented the simple *Identification-by-Velocity-Threshold* (IVT) algorithm which is described in slightly different ways in multiple publications (e.g. [1]).

Classification: In our context, classification means to label eye tracking data with the ID of a unique participant. As classifiers we use *Random Decision Forests* (RF) (as implemented in [2]) and *Radial Basis Function Networks* (RBFN) as described in [1].

Prediction Correctness Heatmap (PCH)

In Figure 1 we show the positive histograms for the *Prediction Correctness Value* (PCV) of each fixation of 153 participants on a reading stimulus. The top row (a, b) shows results from the test cases with unseen data. We see that different algorithm perform similar and are highlighting the same areas as beneficial for the prediction (good to differentiate between participants). This leads to further research Questions about what makes this regions special: e.g. "Are there specific words which are read very differently by the participants?"

(a) RF on test case (b) RBFN on test case

(c) RF on training case (d) Fixations heatmap (r) RBFN on training case

Fig. 1: Example for the *Prediction Correctness Heatmap* (PCH) on a reading stimulus.

The bottom row (c, e) (in Figure 1) shows the result on the training data. The overall occurrence of fixation follows a standard density heatmap of the fixations, like shown in (d).

References

[1] George, A., and Routray, A. A score level fusion method for eye movement biometrics. *Pattern Recognition Letters 82* (2016), 207–215.

[2] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al. Scikit-learn: Machine learning in python. *Journal of machine learning research 12*, Oct (2011), 2825–2830.

[3] Rigas, I., and Komogortsev, O. V. Current research in eye movement biometrics: An analysis based on bioeye 2015 competition. *Image Vision Comput. 58* (2017), 129–141.

Prediction Correctness Value (PCV)

For every trajectory segment t of each participant u, our classifier returns a probability $p(t, u)$. The PCV is the difference between the calculated probability of the correct participant $p_c(t)$, and the highest probability from any other participant $p_m(t)$.

$$\mathrm{PCV}(t) = p_c(t) - p_m(t). \quad (1)$$

In case of a correct prediction, the PCV is positive. If the classifier predicted a wrong participant, the PCV will be negative. The greater the difference from the first to the second guess of the algorithm, the greater the absolute value of the PCV.

Participant	01	02	03	04	05	...
Origin	F	F	T	F	F	...
Ranking	2.	...	1.	...	...	...

correct classification: p - p = positive PCV

Participant	01	02	03	04	05	...
Origin	F	F	T	F	F	...
Ranking	...	1.	3.	...	2.	...

wrong classification: p - p = negative PCV

Fig. 2: A simple example for the PCV. **Left**: Participant 03 is correctly predicted. The second plausible prediction would be participant 01. **Right**: Mistakenly, the classifier predicts participant 02.

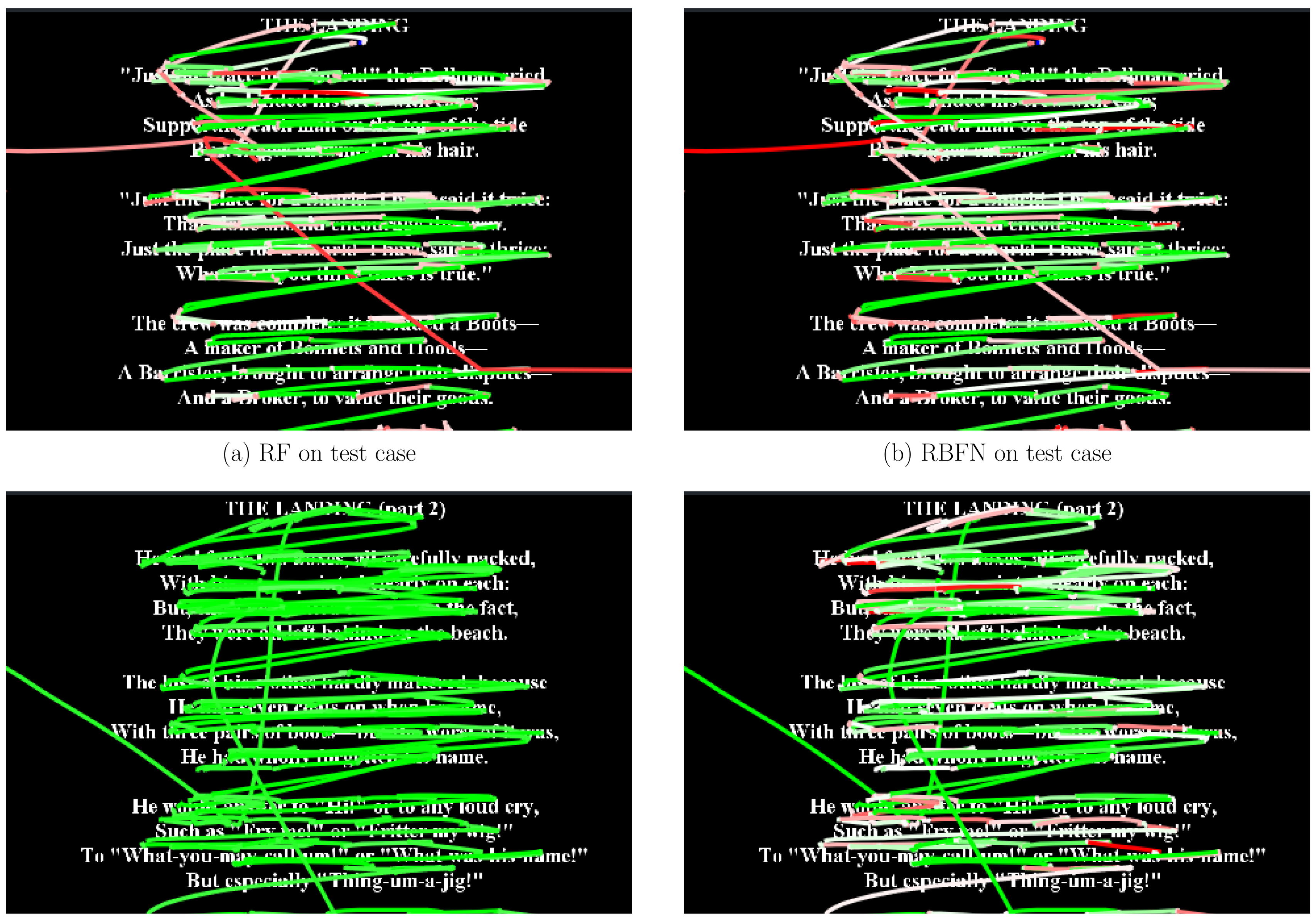
Datasets

We use two datasets from the 2015 *BioEye competition* [3]. Both contain data obtained from 153 participants, whose tasks were to read a poem, and to observe a randomly moving dot (not featured on this poster).

We thank Oleg Komogortsev for providing the used dataset.

Prediction Correctness Trajectory (PCT)

In Figure 3 the PCV is shown for all trajectory segments of a participant reading a poem. (a, b) show the test case, where the algorithms have not seen the data before. (c, d) show the training case. With this visualization it is visible that RF (left) is overfitting (it identifies every segment correctly in the training data (c) but not in the test case (a). RBFN performs similarly on the training (d) and on the test data (b). Beside other things, with this visualization, we can see that the outliers are correctly identified in the training data, while they are mistaken for a different participant by both classifiers in the testing cases. In contrast, eye movements in the region of the text are mostly correctly classified.



(a) RF on test case (b) RBFN on test case

(c) RF on training case (d) RBFN on training case

Fig. 3: Example for the *Prediction Correctness Trajectory* on a reading stimulus.

More Information (paper, source code and more) on our webpage:
`http://wwwdb.informatik.uni-bremen.de/smida_pcv/`

LaTeX TikZposter