A Hybrid System for On-line Blink Detection

Yijia Sun
Department of Computing
Imperial College London
yijia.sun@imperial.ac.uk

Stefanos Zafeiriou
Department of Computing
Imperial College London
s.zafeiriou@imperial.ac.uk

Maja Pantic
Department of Computing
Imperial College London
Faculty of EEMCS
University of Twente
m.pantic@imperial.ac.uk

Abstract

Eye blinking behaviour has been shown to be one of the most informative non-verbal behavioural cues for indicating deceptive behaviour. Traditional blink detection methods tend to use a tracker to extract static eye region images and classify those images as open and closed eyes in order to detect blinks. However, those recognition systems are frame based and do not incorporate temporal information. For this reason, they perform poorly as the tracker fails to detect eyes due to rapid head movement or occlusion. In this paper, we present an approach which combines Hidden Markov Models and Support Vector Machines to model the temporal dynamics of eye blinks and improve the blink detection accuracy.

1. Introduction

Deception detection and determination of concealment-of-intent are very important areas of research as they have a wide range of applications. These include airport security checkpoints, border crossing stations, and other security screening points. Also thousands of people are treated on a daily basis for suicidal depression, schizophrenia, and eating disorders. The ability to deceive is considered to have been part of the evolution process, as being able to conceal your intent improved an individual's survival chances. For this reason, deception is an inevitable aspect of human interaction [9]. Given its pervasiveness, one might expect that humans would be adept at spotting deceit. However, several recent meta-analytic studies [5] have shown that professionals and lay people alike perform poorly at detecting deceit or concealed intent to do others harm, achieving detection accuracy averages that hover slightly above chance (54%). In addition, several studies in experimental psychology suggest that some of the visual behavioural signals cannot be identified or tracked by the human eye because they are too subtle and fleeting to measure [1]. Hence, automated, unobtrusive monitoring and assessment of concealment-of-intent behaviours could form a valuable tool for all involved professionals. In addition to several popular non-verbal behavioural cues which have been adopted for deception detection (such as facial expression, body gestures, voice and verbal style), there is conclusive evidence showing that eye blinking behaviour is also related [3].

In recent decades, research regarding eye blink detection has been conducted. In [17], a method based on dual state tracking was presented which used intensity and edge information to distinguish between closed and open eyes. An improved version of their techniques was later proposed in [18] in order to detect more states, including open, narrow and closed. In particular, the system fed Gabor filter response coefficients into a neural network and analysed eye-relevant action units (AU41, AU42 and AU43) to recognize eye state. In [4][8], the method localized eyes by applying motion analysis and it calculated the normalized cross correlation with pre-trained open-eye and closed-eye templates in each frame. The method detected blinks by observing the waveform of the correlation scores and helped to classify voluntary and spontaneous blinks based on blink duration. This method was applied in a real-time vision-based HCI system which was designed to help disabled people to interact with computer using voluntary blinks. Another technique, introduced in [7], clustered upper and lower eyelids after processing point-based motion. The blink waveform was computed through calculating the space between the upper and the lower eyelid. A driver drowsiness detection system was developed based on this technique for security. In [6], an approach using frame differencing and optical flow was introduced. It was shown that image flow analysis, which contains both the magnitude and the direction of eyelid movement, was more reliable than static appearance. In [10], an eye-blinking detection system was designed based on the analysis of the deformation of active contours that captured the eye. In [11], features produced by the application of a Gabor filter bank were used for eye-blink detection. In [14], a detailed eye region model was used for blinking detection. Finally, a comparison between different features for open and closed eye detection was performed in [13]. Noticeably, all these systems require robust algorithms

for eye tracking and may not perform well when the tracker cannot detect the eye region correctly. A robust eye blink recognition system in unconstrained environment remains a challenging problem. Particularly, the aforementioned methods do not take temporal information into consideration. To solve this problem, we propose a hybrid blink detection method, which combines Hidden Markov Models (HMMs) and Support Vector Machines (SVMs). Our methodology is built upon recent developments on temporal modeling of Facial Action Units (FAUs) [20,21,22]. Introduction of the temporal dynamics in blink modeling enables our method to distinguish between the states of closed and half-open eyes. In particular, our method detects four states according to four temporal segments of blinking: neutral (open-eye), onset (closing eye), apex (closed-eye) and offset (half-open eye). Several features were extracted from each frame and their performances were compared in testing process. Two hybrid models, the blink model and the non-blink model, were trained on image sequences with and without blinks respectively. Blinks were detected by comparing the two models' likelihoods, and durations were obtained through calculating the number of the apex states after decoding. A sliding window was further applied to enable on-line blink detection in real-time.

2. A Hybrid System

This paper introduces an approach which models full temporal dynamics of eye blinking. The method comprises of four main steps: extracting features from each eye image/frame, classifying those features for each frame to one of the states, training hybrid temporal models and applying a sliding window on the testing segment to enable the detection system to work in real-time. HMMs [20,21] can represent the temporal dynamics of eye blinking efficiently. The emission probabilities are usually estimated using mixtures of Gaussian probability density functions. These Gaussian mixtures suffer from poor discrimination because they are trained by likelihood maximization which assumes the model is correct. In contrast, SVM discriminates one class from the other one extremely well. Thus, a hybrid SVM-HMM model was exploited by our blink detection system. Previous works were revolving around capturing transitions between open eyes and closed eyes [6,7,13]. However, these approaches are not able to fully describe eye blinking, since blinks are usually more subtle and complex. Therefore, we employed four temporal states of blinking in our system: neutral (open-eye), onset (closing eye), apex (closed-eye) and offset (half-open eye). All the frames were labelled as 1, 2, 3 and 4 according to which state was appropriate.

2.1 Feature Extraction

We have exploited several popular features independently and compared their performance in our blink detection system: HOG (Histogram of oriented gradients) [23], Gabor filter responses [13], LBP (Local Binary Pattern) [24], optical flow [13] and pixel intensity: HOG compute the distribution of gradient direction in predefined cells of a static image. It can be used as a spatial descriptor which can described eyelid position. Another method which can also be used to extract features is by the use of Gabor Filter responses. A Gabor filter is a complex exponential modulated by a Gaussian function in the spatial domain. Generally, a Gabor filter bank is created by filters of five scales and four orientations. Another feature extraction method is the so-called LBPs. LBP are histograms which are computed by comparing each pixel with its neighbours in the cell of an image and gets an eight binary digit output for each cell. Additionally, optical flow is employed, which captures the relative motion between consecutive frames . Optical flow can describe both magnitude and direction of the eyelid movement.

2.2 Pairwise SVM Classification

Once the features for all sets of image sequences have been extracted, a group of pairwise SVM classifiers were trained for the multi-class classification problem (i.e., classify the image to one of the states). For each pair of temporal segments, we computed a pairwise classifier. Thus, there were $C_4^2 = 4 \times (4-1)/2 = 6$ classifiers trained to distinguish four states, which were μ_{12}/μ_{21} (neutral and onset), μ_{13}/μ_{31} (neutral and apex), μ_{14}/μ_{41} (neutral and offset), μ_{23}/μ_{32} (onset and apex) μ_{24}/μ_{42} (onset and offset) and μ_{34}/μ_{43} (apex and offset). The output of the SVM is the distance between a test pattern and the hyperplane defined by the support vectors. In [15], a method was proposed to estimate posterior probability by feeding the output in a sigmoid function. The model is shown in Equation (1), where h(x) is the distance between the testing data x with the decision boundary of the SVMs. Consequently, thrse SVMs produced predicted labels for each frame, along with the confidence levels of these labels.

$$p(y=+1 \mid x) = g(h(x), w^T, b) = \frac{1}{1+\exp(w^T h(x)+b)} \quad (1)$$

2.3 Temporal Modelling

HMMs are well-known robust machine learning tools and have been applied successfully in speech recognition and analysis of facial expression dynamics. Besides emission probabilities, as previously mentioned, there are two other parameters which are used to define an HMM, i.e., transition probabilities between states and initial probabilities. Transition probabilities (TP) are the probabilities of different transitions between the latent states of the model. Emission probabilities (EP) (also known as output probabilities which govern the distribution of the observed variable at a particular time given the state of the latent variable at that time. Finally, initial probability (IP) is the probability distribution of the initial frame in each image sequence. Among them, TP and IP are estimated from the distribution of training data directly, while EP are set equal to the emission output from the SVM classifiers (i.e, using (1)). Efficient algorithms exist for computing posterior probabilities of each state given the observations and vise-versa to compute the emission probabilities of the whole sequence.

We modeled two kind sequences in our system (two HMMs): blink sequences, which contains blinks, and non-blink episodes, which does not contain any blink. In modeling non-blink sequence, only one of the four segments is present: the neutral state. Hence, there was only one kind of transition in this model: neutral to itself, which is shown in Figure 1. Blink sequences are modeled as sequences containing a complete blink using four different temporal states: neutral, onset, apex and offset. The general form of this blink model is shown in the Figure 2. The blink model allows transitions from every state to its next state, as well as, to itself (besides neutral), but also from offset back to neutral. We assumed that the blinking started from neutral, progressed through the rest of the states and finally returned to the neutral state. Since the only state transition in the non-blink model is from neutral to neutral, we avoided having the same transition in the blink model in order to better discriminate between the two models. Therefore, we kept only the first and the last frame as neural state during the blink sequence pre-segmentation so that there was no transition from neutral to itself in blink sequences.

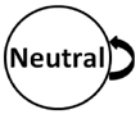

Fig. 1. Transitions of Non-blink Model

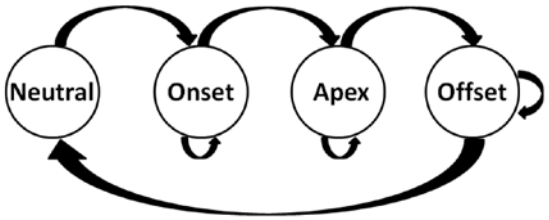

Fig. 2. Transitions of Blink Model

2.4. Real-time Processing Using Sliding Window

In order to detect blinks in real-time video streams (instead of pre-segmented sequences), we exploited a sliding window on the testing image sequence. The principle of the sliding window is explained in Figure 3 below: It starts sliding from the first frame of the raw sequence from time T and after N steps it stops when the window reaches the end of the raw sequence. Once the segment was extracted, we evaluated the probability containing a blink or not. In addition, we were able to decode each frame so that we obtained the frame predicted labels and could then estimate the blink duration by calculating the number of apex states (labelled as 3). In our system, the Viterbi algorithm was used to solve the decoding problem.

Once the window begins to slide and return the frames within it, we evaluate the segment using the two pre-trained hybrid models, namely the blink model and non-blink model. The likelihoods, which describe how probable it is that this segment was generated by each model, were compared in order to determine the existence of a blink or not. The segment is labeled as "blink" if the blink model's likelihood is higher and vice versa. Finally, for every segment, we also decode each frame using the model whose likelihood was higher during the comparison.

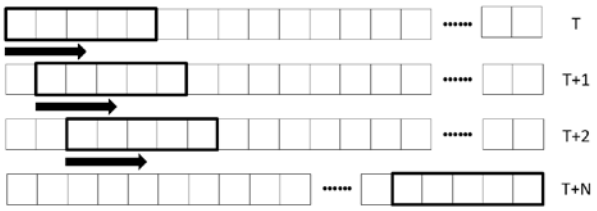

Fig. 3. Real time processing using sliding windows

3. Experiment and Discussion

For the presented work, we conducted experiments using the data recordings from the MAHNOB-Implicit Tagging Database [16]. These audio-visual data consist of spontaneous behaviour of a person watching a video. The recordings were made in a lab setting, using six cameras (in 61 fps), a uniform background and constant lighting conditions. The front-view camera recording was used in our experiments. We used the data recorded from 12 subjects (out of 27 in the dataset). For each subject, blink sequences varied much more than non-blink sequence. Hence, we pre-segmented 40 video clips with blinks and 4 video clips without blinks for each subject. Additionally, in order to apply on-line testing, a separate 30 second video clip was extracted randomly for each subject as well (excluding the above 44 clips). DIKT [12], an on-line tracker, was applied for eye region tracking accompanied with EyeAPI, an eye centre localization tool [19], to automate the extraction of eye region for each frame. Figure 4 shows an entire blink sequence. Every frame of the dataset was annotated as neutral, onset, apex and offset state.

Once we extracted features frame by frame for all sets of image sequences using methods described in Section 2.1. Those features were fed into classifiers and six pairwise SVMs were trained. We then fully trained two hybrid models, the blink and non-blink model, based on these SVMs through estimating three parameters, as described in Section 2.3. While the transformation from pairwise probability to posterior probability was implemented by Libsvm[2].

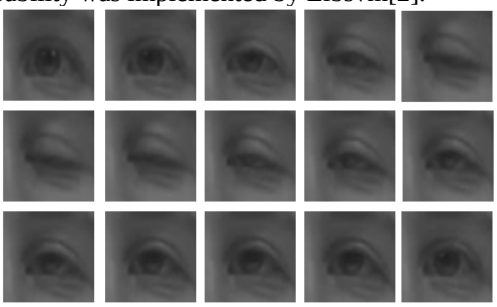

Fig. 4. Full Blinking Behaviour

The layout of the rest of the experiments is as follows: (3.1) experiment adopting leave one subject out cross validation and classifying pre-segmented sequences as blink sequences or non-blink sequences. (3.2) experiment using leave one subject out cross validation and decoding each frame as one of the temporal states. (3.3) experiment exploiting a sliding window.

3.1 Pre-segmented Sequences Classification

In each cycle of this experiment, we left one subject out (40 blink sequences and 4 non-blink sequences) for test and trained on the other 11 subjects. Each pre-trained hybrid model for blink and non-blink was applied on every test sequence and likelihood was estimated. We classified each sequence through comparing their likelihoods.

We conducted testing using five different features and compared the results which are shown in Table I. In this table, we display the precisions, recalls and F1 measures for the five feature sets employed. As can be seen HOG, Gabor and intensities performed almost equally well in this task. We believe that intensity features performed quite well in this task since the recording were collected under well-controlled illumination conditions.

3.2 Frame By Frame Classification

There is only one state in the non-blink model so that this model always classifies every frame as neutral. Thus, we conducted this experiment using only blink model and blink sequences. In each iteration, we left one subject out (40 blink sequences) for test and trained on the other 11 subjects. The pre-trained blink model classified each frame of the testing sequences as one of the temporal states: neutral, onset, apex and offset. In order to show the advantage of the temporal model, we employed a four-class SVM using Libsvm [2] as a classifier without temporal information.

Table I. Full Blinking Behaviour Classification results of Pre-Segmented Sequences (IN = pixel intensity, Gabor = Gabor filter, LBP = Local Binary Patterns, OF = Optical Flow)

	HOG	INT	Gabor	LBP	OF
Precision	94.58%	**99.38%**	98.75%	68.13%	97.50%
Recall	**100.00%**	99.58%	**100.00%**	**100.00%**	**100.00%**
F1 measure	97.22%	**99.48%**	99.38%	81.04%	98.74%

Table II displays the classification accuracy of two different approaches and five different features exploited. The introduction of temporal information increased the accuracy independently of the type of features used. As can be seen, optical flow displays the best discriminative ability. Using image based descriptors such as HOG and Gabor responses it is quite hard to discriminate between onset and offset, while in case of optical flow the motion fields of onset and offset have opposite directions.

Table II. Frame-by-frame Classification results

	HOG	INT	Gabor	LBP	OF
4-class SVM	52.45%	50.19%	54.45%	48.00%	**62.83%**
Hybrid Model	78.9%	69.96%	72.99%	66.06%	**79.36%**

3.3. Experimenting with a Sliding Window

In this experiment, a sliding window was applied on the testing sequence in order to detect blinks and calculate blink durations in real-time. In each iteration, we left one subject (randomly segmented 30 second video clip) out for test and trained two models on the pre-segmented sequences (40 blink sequences and 4 non-blink sequences) of the other 11 subjects. During testing, for each segment extracted by the window, the two pre-trained hybrid models evaluated and classified it as a sequence with or without blinks. The blink model was exploited to decode each frame if the segment was detected to contain blink. Otherwise, we used non-blink model to decode each frame as neutral state. After applying majority voting on those frames which were decoded for many times, we calculated the number of apex states and estimated blink durations.

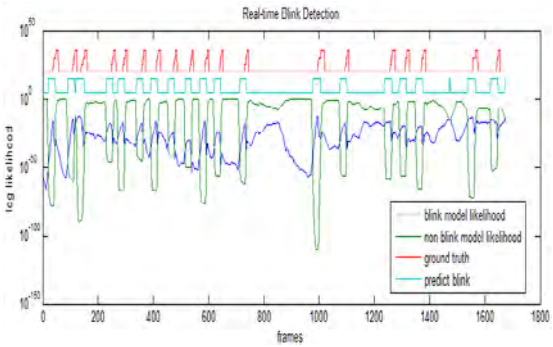

Fig. 5. Real time-blink detection

Among all the testing results, 90.99% of the blinks were recognized successfully while 7.21% of the patterns were misclassified as blinks. Even all estimations of blink durations were longer than the ground truth; we could still spot spontaneous blinks from voluntary blinks. A real-time detection result for one of the subjects is shown in Figure 5.

4. Conclusion

In this paper, we have presented a hybrid system combining HMM and SVM for automatic eye blink detection and blink duration calculation. Several popular blink detection features were extracted and their performances were compared. We modelled blink temporal dynamics into our system. As a result, the temporal model works significantly better than multi-class SVM when classifying each frame. Even though we demonstrated almost 100% accuracy in detecting blinks, frame-by-frame classification, in one of the temporal models, remains a challenging problem.

5. Acknowledgments

This work has been supported by the European Research Council under the ERC Starting Grant agreement no. ERC-2007-StG-203143 (MAHNOB).

6. References

[1] J. Burgoon. Nonverbal measurement of deceit. The sourcebook of nonverbal measures: Going beyond words, pages 237–250, 2005.

[2] C. Chang and C. Lin, “Libsvm: a library for support vector machines”, ACM Transactions on Intelligent Systems and Technology (TIST), volume 2, number 3, pages 27, 2011.

[3] K. Fukuda, “Eye blinks: new indices for the detection of deception”, International Journal of Psychophysiology, volume 40, number 3, pages 239–245, 2001.

[4] K. Grauman, M. Betke, J. Gips, and G. Bradski, "Communication via eye blinks-detection and duration analysis in real time", CVPR 2001, volume 1, 2001.

[5] M. Hartwig and C. Bond Jr, "Why do lie-catchers fail? a lens model meta analysis of human lie judgments", Psychological bulletin, volume 137, number 4, 2011.

[6] R. Heishman and Z. Duric, "Using image flow to detect eye blinks in color videos", WACV’07, 2007.

[7] T. Ito, S. Mita, K. Kozuka, T. Nakano, and S. Yamamoto, "Driver blink measurement by the motion picture processing and its application to drowsiness detection", In 5th International Conference on Intelligent Transportation Systems, 2002, pages 168– 173, 2002.

[8] M. Khan and A. Mansoor, "Real time eyes tracking and classification for driver fatigue detection", Image Analysis and Recognition, pages 729–738, 2008.

[9] M. Knapp, Lying and deception in human interaction. Allyn and Bacon, 2008.

[10] A. Krolak and P. Strumillo, "Vision-based eye blink monitoring system for human-computer interfacing". In Conference on Human System Interactions, pages 994–998, 2008.

[11] J. Li, "Eye blink detection based on multiple gabor response waves", In International Conference on Machine Learning and Cybernetics, volume 5, pages 2852–2856, 2008.

[12] S. Liwicki, S. Zafeiriou, G. Tzimiropoulos, and M. Pantic, "Fast and robust appearance-based tracking", In Proceedings of IEEE International Conference on Automatic Face and Gesture Recognition (FG'11), pages 507–513, Santa Barbara, CA, USA, March 2011.

[13] K. Minkov, S. Zafeiriou, and M. Pantic, "A comparison of different features for automatic eye blinking detection with an application to analysis of deceptive behavior", In International Symposium on Communications Control and Signal Processing (ISCCSP), pages 1–4, 2012

[14] T. Moriyama, T. Kanade, J. Xiao, and J. Cohn. "Meticulously detailed eye region model and its application to analysis of facial images", IEEE Transactions on Pattern Analysis and Machine Intelligence, volume 28, number 5, pages 738–752, 2006.

[15] J. Platt et al., "Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods", Advances in large margin classifiers, pages 61–74, 1999.

[16] M. Soleymani, J. Lichtenauer, T. Pun, and M. Pantic, "A multi-modal database for affect recognition and implicit tagging", IEEE Transactions on Affective Computing, in press.

[17] Y. Tian, T. Kanade, and J. Cohn, "Dual-state parametric eye tracking", In Proceedings of Fourth IEEE International Conference on Automatic Face and Gesture Recognition, pages 110–115, 2000.

[18] Y. Tian, T. Kanade, and J. Cohn, "Eye-state action unit detection by gabor wavelets", Advances in Multimodal Interfaces ICMI 2000, pages 143–150, 2000.

[19] R. Valenti and T. Gevers, "Accurate eye center location and tracking using isophote curvature", CVPR, 2008

[20] M. F. Valstar, M. Pantic, "Fully Automatic Recognition of the Temporal Phases of Facial Actions", IEEE Transactions on Systems, Man and Cybernetics, volume 42, pages 28 - 43, 2012.

[21] S. Koelstra, M. Pantic, I. Patras, "A dynamic texture based approach to recognition of facial actions and their temporal models", IEEE Transactions on Pattern Analysis and Machine Intelligence, volume 32, number 11, pages 1940 - 1954, 2010.

[22] O. Rudovic, V. Pavlovic, M. Pantic, "Kernel Conditional Ordinal Random Fields for Temporal Segmentation of Facial Action Units", Proceedings of the 12th European Conference on Computer Vision (ECCV-W'12), Florence, Italy. October 2012.

[23] Navneet Dalal and Bill Triggs, "Histograms of oriented gradients for human detection", International Conference on Computer Vision and Pattern Recognition, volume 2, pages 886–893

[24] T. Ojala, M. Pietikäinen, and D. Harwood, "A Comparative Study of Texture Measures with Classification Based on Feature Distributions", Pattern Recognition, volume 29, pages 51-59, 1996