Analysis of Lying through an Eye Blink Detection Algorithm

Walter Genchi, Riccardo Menoli, Davide Parolo,
Marco Baggio, Andrea Della Vecchia

Contents

1. Introduction

Lie detection using physiological methods has been utilized for the past 75 years, starting with the work of Benussi, with varying rates of success in identification of the guilty. More recently, eye blink measures have attracted considerable attention, since they have been related to cognitive processes. They can be recorded easily without the subject's awareness and without the application of electrodes.

The first part of our project consisted of acquiring the background knowledge in the domain of cognitive and behavioral sciences with a focus on the relevant researches on eye blink and lie detection. The second part was more technical and regarded the research of the free tools[1] already available for eye blink detection with a good performance in terms of accuracy. To our knowledge, the only freely-available software for blink detection is built on the OpenCV Python modules and is based on a metric called *eye aspect ratio* (EAR). Despite the acceptable results in live blink detection, this ready-solution performed poorly in different video conditions. Therefore, the third part of the project consisted of building, testing and validating a Support Vector Machine (SVM) classifier using the EAR features. Finally, we created an interface for the video to analyse from a lie detection point of view, together with some useful descriptive plots of blinking measures.

2. Background knowledge

2.1. Eye blinking

Human eye blinking has been studied for decades by psychologists, psychiatrists, ophthalmologists and neurophysiologists, because of its widespread applications.

From a physiological point of view, blinking not only helps to moist the outer eye and prevents evaporation from the ocular surface, but also defends the eye from environmental exposure and mechanically removes deposited particles and cellular debris (see Craig [2] and Lemp and Wolfley [9]).

Having said that, we can divide blinking into reflex, voluntary and involuntary. The difference between reflex and involuntary blink stands in the stimulus: while reflex blink is a response to an external stimulus (auditory, cognitive, trigeminal or visual origin), involuntary blink occurs without external stimuli.

Causes of Blinking

Several parameters influence involuntary blinking, including the temperature, relative humidity, illumination, draft, eye diseases, and working with video display units. Moreover, there is evidence that the mental state and activity are of crucial importance: for example, mental stress, cognitive awareness and task performed can significantly influence the

[1] Smart Eye Pro is a commercial tool offering an advanced remote eye-tracking system.

blink rate (BR), i.e. the number of blinks per time. Bentivoglio et al. [1] measured BR in three common behavioral conditions: resting, reading and conversation. The mean BR at rest was 17 blinks/minute, during conversation it increased to 26 blinks/minute and it decreased to 4.5 blinks/minute while reading[2]. In the same experiment, they excluded differences in BR related to eye color, eyeglass wearing and age.

Measures of Blinking

As seen from the above, an intuitive and actually very used measure to quantify eye blinking is blink rate (BR). When comparing BR measures of different periods for the same subject, it is useful to compute the change in BR during the baseline and the target period. Another possible measure is the inter-blink time, together with its mean and standard deviation.

2.2. Lying

Generally speaking, lying is defined as "a psychological process by which one individual deliberately attempts to convince another person to accept as true what the liar knows to be false, to gain some type of benefit or to avoid loss" (Abe, 2011).

Cognitive Demand

Besides the possible benefits or losses that a lie might imply, there is a large body of evidence to suggest that lying can be more congnitively demanding than truthtelling and several aspects of lying (see Leal and Vrij [8]) contribute to this increased mental load. The process of lying involves cognitive effort not only while producing a false statement, but also by inhibiting the true statement (which often activates automatically), by monitoring the interlocutor's reactions and by adjusting the behavior congruently to the lie. However, lying is not always more demanding than truthtelling[3]: distinct types of lies may differ in their cognitive complexity and may require different levels of cognitive effort. For example, the cognitive effort may be minimal when the subject is simply denying a fact that actually happened or when the lie has been overlearned. To overcome this issue, authors in literature have proposed different strategies, which are usually used in police interrogations, and which allow increasing cognitive load of liars, keeping unaltered the cognitive load of truth-tellers. Among such strategies, the *unexpected questions* (see Vrij et al. [17]) are questions to which liars could not prepare their response in advance and need to fabricate plausible responses in the case of unexpected questions, and this yields

[2]Women had higher BR than men just while reading

[3]As suggested by Leal and Vrij [8], in an experimental design interviewees have to be motivated to be believed and the target event has to be easily retrieved.

an increase in the cognitive load, which is reflected in a number of indices of mental effort, such as reaction time (RT) (see Monaro et al. [10]).

People's Performance at Lie Detection

People lie twice a day on average (see DePaulo et al. [4]). However, it goes without saying that there are certain circumstances (the so-called *high-stakes situations*) where lies are more likely to give rise to heavy consequences to both an individual and society, such as in forensic settings. In such situations, researchers have shown that interrogators rarely have the capacity to see the truth behind lies, regardless of the fact that they occur on a regular basis. If we consider now the lie detection performance of other human observers (see Su and Levine [16]), it has been shown that it is only slightly above chance for ordinary people.

Different Approaches to Lie Detection

There have been proposed different approaches to discriminate liars and truth-tellers in real world situations, each with its own evidence-based measure.

The emotional-based approach has its roots in the Darwin's book, *The Expression of Emotions in Man and Animals*, where he proposed a famous theory called the inhibition hypothesis: some facial actions that are the most difficult to create voluntarily are also the hardest to be inhibited (see Darwin and Prodger [3]). Further development in this direction have been made in 1980 by Ekman et al. [7], who reported a list of Action Units (AUs)[4]. Each AU is related to the movement of a single facial muscle and can produce motion of a facial part or appearance changes in a facial region. Based on Darwin's inhibition hypothesis, AUs such as sadness (AU1+AU4) are hard to inhibit. Therefore, AUs have been shown to be potential indicators for distinguishing liars from truth-tellers in *high-stakes situations*, based on the existence of *micro-expressions* (see Ekman and Friesen [5]).

Physiological measures, such as thermal body imaging, voice stress analysis and the polygraph are also part of the emotional-based approach to lie detection. The drawback of these physiological approaches is that they only focus on measures emanating from the peripheral rather than central nervous system. In this case, the interpretation of the measurements is not directly related to the emotional states of humans, which are controlled by the central nervous system and are detected by the fMRI. The other disadvantage is that they are easily degraded by certain countermeasures, such as tranquillizers or repeated practice. For example, silently counting back-ward or pinching a finger, are effective in altering the results in detection of deception via fRMI.

A psychological linguistic measure is another deceit detection approach which has been

[4] Ekman and Friesen [6] proposed the Facial Action Coding System (FACS) to encode facial movements during facial expressions using Action Units (AUs).

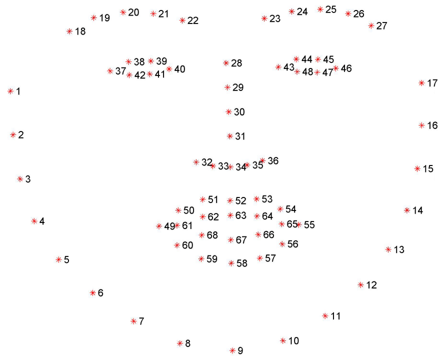


Figure 1: The full set of 68 facial landmarks that can be detected via dlib.

widely studied by psychologists in the last few decades. Word frequency, grammar usage are tentative word frequency are only some of the verbal cues, which have recently started to be combined with other behavioral elements. One of the most recent examples in this sense regarded the Yelp fake review detector, which is supposed to guarantee quality and reliability of reviews written by active users. To sum up, great caution is required when applying such algorithms to *high-stakes situations* because verbal countermeasures are easier to adopt than non-verbal ones.

Finally, the behavioral approach is based on the observation of the nonverbal behavior of the suspect while he is producing a deceptive or truthful response. A typical experimental set-up would be the online navigation space, where it could be possible to monitor the user's activity by observing for example his reaction time (RT) or the keystroke and mouse dynamics (see Monaro et al. [10]), which have all the advantage of being implicit measures.

3. Project

After the acquisition of the background knowledge and the research of already done eye blinking detectors, the project mainly consisted of four parts: evaluation of the Python modules `OpenCV` and `dlib` for facial landmarks blink counter and detector, implementation of the live blink detector, improvement of blink detector using SVM and, finally, interpretation of the results from a lie detection point of view.

3.1. OpenCV and Facial Landmarks

The implementation which follows is based on the tutorial from the blog Pyimagesearch (see Rosebrock [11]) which, in turn, takes as its starting point the work of Soukupova and Cech [14] for the the metric called *eye aspect ratio* (EAR). The 68 facial landmarks are based on the `iBUG 300-W` dataset (see Sagonas et al. [12] and Figure 1), which the `dlib` facial landmark predictor was trained on.

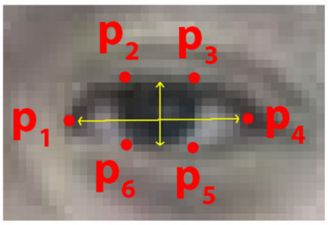


Figure 2: The 6 facial landmarks associated with the eye.

EAR

For every video frame, the eye landmarks (see in Figure 1, [37,42] and [43,48] landmarks for left and rigth eye respectively) are detected. The *eye aspect ratio* (EAR) between height and width of the eye is computed, following the work of Soukupova and Cech [14], as

$$EAR = \frac{\| p_2 - p_6 \| + \| p_3 - p_5 \|}{2 \| p_1 - p_4 \|},$$

where $p_1, \ldots, p_6$ are the corresponding 6 eye landmark locations, shown in Figure 2, and $\| p_2 - p_6 \|$ is the Euclidean distance between the locations p_2 and p_6.

The EAR is mostly constant when an eye is open and is getting close to zero while closing an eye. Soukupova and Cech [14] report that EAR is partially person and head pose insensitive; moreover they claim that the open eye has a small variance among individuals and it is fully invariant to a uniform scaling of the image and inplane rotation of the face. Since eye blinking is performed by both eyes synchronously, the EAR of both eyes is averaged. The weight of the denominator is appropriate, since there is only one set of horizontal points but two sets of vertical points.

The advantage of using EAR instead of traditional *image processing* methods should be evident: firstly, we avoid both eye localization and thresholding to find the white of the eyes and, secondly, we do not need to determine whether the "white" region of the eyes disappear for a period of time (indicating a blink) or not.

Tuning Parameters

In order to determine whether a blink has taken place in a video stream or not, we need two tuning parameters: the algorithm will detect a blink if the EAR falls below a certain threshold `EYE_AR_THRESH` for a certain number of consecutive frames `EYE_AR_CONSEC_FRAMES` and then rises above the threshold. Following a trial and error procedure on some testing videos, it was observed that the best strategy was to assign a value of 0.28 to `EYE_AR_THRESH` and 2 to `EYE_AR_CONSEC_FRAMES`.

These parameters are definitevely person and head pose sensitive, but from our analysis we have come up that these two values could at best describe the average eye blink behaviour.

3.2. Live Blink Detector

The implementation of the live blink detector straightly followed from the Rosebrock [11]'s guidelines. The problems encountered were related mainly with the low level of control of blink detection accuracy, provinding that it was not algorithmically easy to produce an EAR running plot, which is the minimum requirement to build a simple, but still useful lie detection tool. Moreover, given the overall inaccuracy of this first-stage blink detector (see table 5) in particular when the subject was moving, we decided to make the blink detector more robust using SVM, which is mathematically not feasible when dealing with real-time video streams.

3.3. Blink Detector using SVM

It generally does not hold that low value of the EAR means that a person is blinking. A low value of the EAR may occur when a subject closes his/her eyes intentionally for a longer time or performs a facial expression, yawning, etc., or the EAR captures a short random fluctuation of the landmarks.

Therefore, following what has been proposed by Soukupova and Cech [14], we built a classifier that takes as an input the EAR values belonging to a larger temporal window of a frame.

Training and Testing

The training dataset is called *Eyeblink8*[5] which consists of 8 videos with 4 individuals (1 wearing glasses). Videos are recorded under different conditions with faces mostly oriented directly to the camera. It contains many natural face movements, facial mimics and other intensive non-blink moves. The dataset contains over 82600 frames (640×480) with 353 blinks. All videos were recorded using Logitech C905 camera with 30fps acquisition. The authors of the dataset recognize 3 states: open, half and close. When the blink starts, half tags are assigned to individual frames until is fully closed. Fully closed eye is defined as 90–100% of the eye is covered with the eyelid. Fully closed eyes are tagged with close, while opened eye is again tagged with half until it is fully open. Also not fully closed eye blinks can be annotated this way (consisting only from "halfs"). If only one eye is visible (or blinks) the tag Left/Right is added to the eye state, based on the location of the visible blink.

For each frame, a 7-dimensional feature[6] is gathered by concatenating the EARs of its ±3 neighboring frames. The dataframe structure obtained for a video of n frames is

[5] Freely available at http://www2.fiit.stuba.sk/~fogelton/acvr2014/index.html

[6] We tried also to use a 13-dimensional window, as suggested by Soukupova and Cech [14], but we noticed that its performance was poor when dealing with quick consecutive blinks.

Frame	$\mathbf{x_1}$	$\mathbf{x_2}$	$\dots$	$\mathbf{x_5}$	$\mathbf{x_6}$	$\mathbf{x_7}$	$\mathbf{y}$
4	EAR_1	EAR_2	$\dots$	EAR_5	EAR_6	EAR_7	0
5	EAR_2	EAR_3	$\dots$	EAR_6	EAR_7	EAR_8	0
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$.
k	EAR_k	EAR_{k+1}	$\dots$	EAR_{k+4}	EAR_{k+5}	EAR_{k+6}	1
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$n-3$	EAR_{n-6}	EAR_{n-5}	$\dots$	EAR_{n-2}	EAR_{n-1}	EAR_n	0

Table 1: Dataframe structure given as input to SVM.

	Video 1	Video 2	Video 3	Video 4	Video 5	Video 6	Video 7	Video 8
# of 0's	15410	10253	8760	5040	10423	4740	8523	4550
# of 1's	282	461	435	268	233	387	547	338

Table 2: The number of 0's and 1's computed for each of the 8 video of *Eyeblink8*, based on the table 1 data transformation.

represented in table 1: EAR_i is the value of EAR at the i-th frame, $y_k = 1$ if the eye is fully closed in at least one of the frames belonging to the 7-dimensional window, while $y_k = 0$ if the eye is half or open in all the frames belonging to the same 7-dimensional window.

The linear SVM classifier was trained from manually annotated sequences $\mathbf{y}$ (constructed by the tag already given in the *Eyeblink8* dataset). A 7-dimensional feature $\mathbf{x_1}, \dots, \mathbf{x_7}$ is computed (using the EAR measure done by Rosebrock [11]) and classified by SVM for each frame except the three frames at the beginning and at the ending of a video sequence. The training dataset was obtained using the following procedure:

- Given the much larger number of 0's than 1's in $\mathbf{y}$ (see table 2), the strategy was to balance the dataset by taking, for each video, a random sample of 0's of size equal to the number of 1's. The final result was a dataset of 5900 rows, which was then randomly splitted into training (80% of the observations) and test (20% of the observations) set. The distribution of 0's and 1's in the two sets is balanced: the training set has 2348 1's over 4720 units, while the test set has 602 1's over 1180 units.

- In order to avoid different scales of EAR when comparing different videos, a normalization was done firstly for each video (before sampling the 0's and 1's) and secondly a standardization in the training set[7].

[7]The training set standardization is then applied to the previsions.

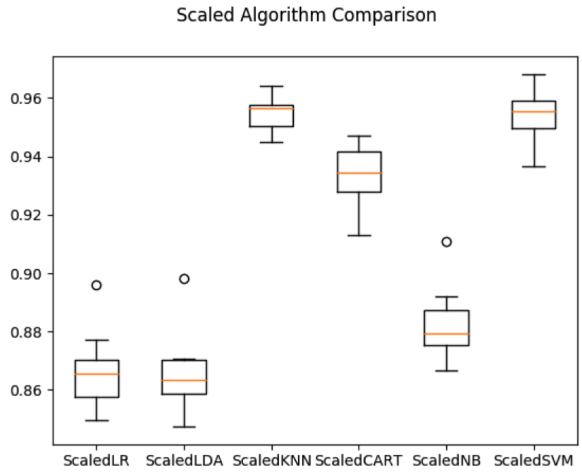


Table 3: Boxplot comparing the accuracy of the standardized ML models.

The choice of the SVM, instead of other machine learning classifiers, is due to his consistency when dealing with dependent observations (see Steinwart et al. [15]). In our case, rows are highly correlated with each other. Moreover, in SVM it was actually observed an higher accuracy than other machine learning models (see the boxplot in table 3), which were all scaled first and then trained using a 10-fold cross validation. In order to obtain the highest accuracy on the test set, the chosen values for the SVM kernel and parameter C were `kernel='rbf'` (i.e. *radial basis function kernel*) and `C=1.7`. Moreover, the prediction performance was evaluated on the test set, obtaining a 95% accuracy and other measures of goodness (see table 4).

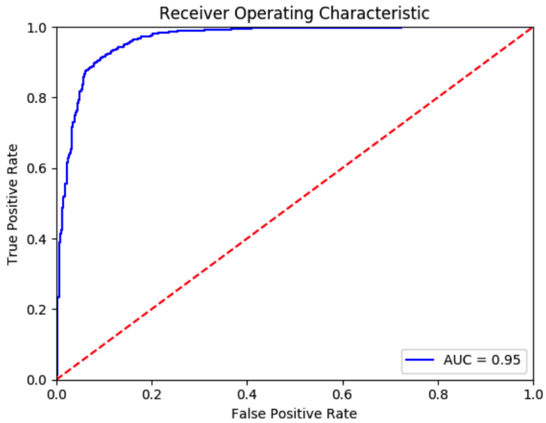
It should be noted that SVM performs 95% of accuracy on detecting the presence of eyes closed in the 7-dimensional window. For this reason when a blink occurs, assuming that the classifier is never wrong and that the eyes are closed for only one frame, the SVM prevision is made by a sequence of seven 1's between 0's. This sequence has to be converted into one blink. It is important to consider that SVM previsions could be wrong both with false positives and false negatives.

The first algorithm we tested to convert SVM prevision into blinks was a majority based algorithm: we looped the list of the SVM previsions, when a 1 was found the six following elements were checked. If their sum was greater than 4, a blink was detected and all the six following elements were put to 0.
The main problem of the majority based algorithm are "long blinks", i.e. blinks in which the eyes are closed for many frames. For example, think about a blink in which the subject stays with the eyes closed for 8 frames, assuming that the classifier is never wrong, there is a sequence of 14 consecutive 1's. The majority based algorithm detects 2 blinks.

After some empirical tests we noticed that the best rule was to create only sequences of

Prevision	Truth 0	1	Total
0	539	39	578
1	22	580	602
Total	561	619	1180

	precision	**recall**	**f1-score**	**support**
0.0	0.96	0.93	0.95	578
1.0	0.94	0.96	0.95	602
avg / total	0.95	0.95	0.95	1180



Area Under the Curve (AUC) = 0.948

Table 4: Measures of goodness for the SVM blink detector.

consecutives 1's and 0's and then turn the only 1's sequences into blinks. The consecutives 1's and 0's sequences are created as follows:

- 0's: single $(\ldots,1,1,0,1,1,\ldots)$, double $(\ldots,1,1,0,0,1,1,\ldots)$ and triple $(\ldots,1,1,0,0,0,1,1,\ldots)$ 0's between series of consecutives 1's were recognized as misclassified and changed to 1's.
- 1's: single $(\ldots,0,0,1,0,0,\ldots)$, double $(\ldots,0,0,1,1,0,0,\ldots)$ 1's between series of consecutives 0's were recognized as misclassified and changed to 0's.

After this transformation, when a sequence of 1's was found, a blink is detected.

Validation

The performance has been evaluated not only in terms of the numbers of correctly classified blinks (*true positives*), but also considering the important cases when a video frame was misclassified as a blink (*false positives*) or when a blink was missed (*false negatives*). Based on these data, two measures have been computed

$$\text{Precision} = \frac{true\ positives}{true\ positives + false\ positives} \quad \text{and}$$

$$\text{Recall} = \frac{true\ positives}{true\ positives + false\ negatives}.$$

SVM Precision = 43% Recall = 81%		**OpenCV** Precision = 7% Recall = 12%
SVM Precision = 100% Recall = 23%		**OpenCV** Precision = 0% Recall = 0%
SVM Precision = 58% Recall = 89%		**OpenCV** Precision = 7% Recall = 33%

Table 5: Precision and Recall computed for Scenario 1 (top), Scenario 2 (middle) and Scenario 3 (bottom).

The SVM blink detector was firstly validated on some videos from the `iBUG 300-W` dataset and its performance was therefore compared with the OpenCV blink detector (see Rosebrock [11]). The aim of this stage was to test the two system's blink detection ability, independently of variations in pose, expression, illumination, background, occlusion, and image quality. Following what has been done in Sagonas et al. [12], the following three scenarios have been considered:

- Scenario 1: A number of testing videos will be of people recorded in well-lit conditions displaying arbitrary expressions in various head poses. This scenario aims to evaluate algorithms that could be suitable for facial motion analysis in laboratory and naturalistic well-lit conditions.

- Scenario 2: A number of testing videos will be of people recorded in unconstrained conditions, displaying arbitrary expressions in various head poses but without large occlusions. This scenario aims to evaluate algorithms that could be suitable for facial motion analysis in real-world human-computer interaction applications.

- Scenario 3: A number of testing videos will be of people recorded in completely unconstrained conditions including the illumination conditions, occlusions, make-up, expression, head pose, etc. This scenario aims to assess the performance of facial landmark tracking in arbitrary conditions.

There has been taken one video for each scenario. Looking at the results in table 5, the OpenCV blink detector performs poorly under all challenging conditions. The OpenCV detector encounters several problems when dealing with facial expressions and pose variations (Scenario 1) and also heavily misclassifies blinks under arbitrary conditions (Scenario 3). Both SVM and OpenCV blink detector have difficulties under challenging illumination conditions (Scenario 2), due to face landmarks detection problems, which is the

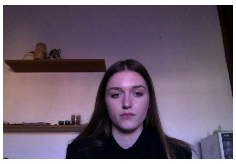
SVM Precision = 70% Recall = 98%		**OpenCV** Precision = 13% Recall = 63%
SVM Precision = 89% Recall = 93%		**OpenCV** Precision = 33% Recall = 4%

Table 6: Precision and Recall computed for Bianca's (top) and Filippo's (bottom) video.

common ground for both algorithms. From one hand, the SVM classifier never misclassifies the blinks (Precision=100%) and misses around 80% of the blinks (Recall=23%). On the other hand, the OpenCV blink detector misses and misclassifies all blinks.

Secondly, the validation procedure was done also for videos of friends, relatives and acquaintances; the results for two videos[8] are summarised in 6. We can notice that also for new and longer videos (the lenghts are 3:45 and 7:30 minutes, respectively for Bianca and Filippo) the validation procedure shows a significant improvement in the SVM blink detector in terms of both precision and recall, especially when occlusions occur (i.e. Filippo wears glasses). Moreover, the overall SVM classifier performance is better in Filippo, due to the *frame rate* the video was recorded at: Bianca was recorded at 14 fps, while Filippo at 29 fps. The SVM blink detector was trained with 30 fps videos (see 3.3), so this difference in performances was expected, but unavoidable[9].
The objective of this second validation was to determine whether these videos could be used or not for a further analysis of some empirical measures of blinking (e.g. blink rate and EAR) from a lie detection point of view. The overall performance of the SVM classifier was good enough to proceed to the last step of our project.

3.4. Is the Person Lying?

In order to determine whether there could be some empirical measures (e.g. blink rate and EAR) which could be helpful for lie detection, at first, we built an interface containing the video, some descriptive statistics and two real-time plots. Secondly, we decided an experimental setting that could significantly highlight differences in those empirical measures among different periods (i.e. baseline, target period and target offset), taking the work of Leal and Vrij [8] as reference.

[8]SVM blink detector performs well also on other videos.
[9]Every member of the group recorded some videos with the camera of his own laptop, which could not be set to a standard frame rate.

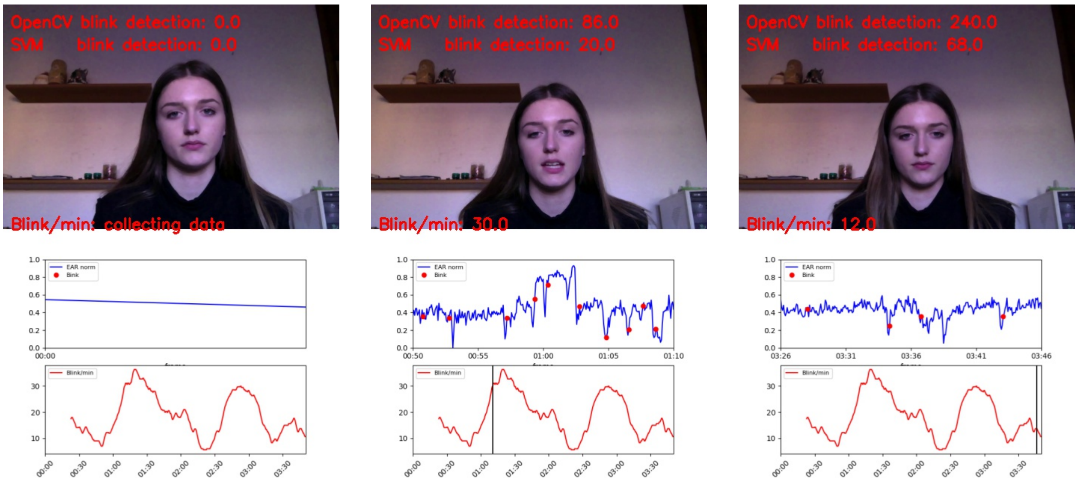


Table 7: Frame 1, 985 and 3168 of the interface from Bianca's video.

Video Interface

Given a video as an input, firstly the program performs a pre-processing of the raw-data (i.e. for each frame, it detects facial landmarks, computes and normalizes the EAR values and arranges data in the form of table 1), secondly the already-trained SVM classifier computes the previsions (0 = opened eye, 1 = closed eye), then the sequence of 0's and 1's is converted into blinks / no blinks based on the empirical rule describe in 3.3, and finally the program returns, for each frame, an output like those showed in table 7.

On the top, there is a real-time counter of blinks detected by the OpenCV and the SVM detectors up to that frame. Lower, it is shown the instant blink rate, which for frame x at instant t (in seconds) is computed considering the frames belonging to the interval $(t - 20, t)$. This 20 seconds time window[10] is then plotted in the graph below, together with the normalized EAR value (blue line) for each frame belonging to the window; a red dot is shown corresponding to the frame at which the blink was detected. The last graph keeps track of the blink rate[11] throughout the whole video and denotes the current frame by a vertical black line. This feature is very useful when looking at these empirical measures from a lie detection point of view.[12]

Experimental Setting

The structure of the recorded videos is organized in 5 periods of the same length, where the person follows the instructions[13] reported on the screen:

1. Baseline: the subject is at rest, the instruction is "Look at the screen".

[10]Note that for the first 20 seconds the blink rate is not available, so the output is "Blink/min: collecting data".
[11]A smoothed value is displayed for aesthetic reasons.
[12]See the appendix for program's details.
[13]The text is black and the background is white. The language used is italian.

2. Target Period: the subject is telling the truth, the istruction is “We ask you to tell in detail what you did last week”.
3. Baseline: the subject is at rest, the instruction is “Look at the screen”.
4. Target Period: the subject is lying[14], the istruction is “We ask you to tell in detail what you did last week, telling a lie (without any truth[15])”.
5. Baseline: the subject is at rest, the instruction is “Look at the screen”.

Due to some adjustments during the lie detection analysis, we performed the experiments following two slightly different settings, in terms of length of the video and interaction with the person, keeping the same structure described above. In the first setting we recorded two *short* videos (3:45 minutes) and one *long* video (7:30 minutes), without any interaction with the subject. In the second setting we recorded nine *long* videos (7:30), where the subject was asked some unexpected questions in both target periods.

Results

Here we report the two most significant results obtained not only from a lie detection point of view, but also in terms of accuracy of blink detection[16]. One video belongs to the first setting (i.e. no interaction with the subject) and the subject tells the truth first and then lies. The other video belongs to the second setting (i.e. unexpected questions during target periods) and the subject first lies and then tells the truth.

In both Bianca’s video (first setting) and Filippo’s video (second setting) it was observed a significant difference both in the blink rate and in the average inter-blink interval during the 5 periods (see table 8 and table 9). The blink rate during both target periods is higher than in the baseline periods, due to the fact that the mean blink rate at rest is 17 blinks/minute, while during conversation it increases to 26 blinks/minute (see Bentivoglio et al. [1]). However, for both videos there is a significant difference in the blink rate while the person is lying and while he/she is telling the truth, independently from the order of the target periods: not only the mean blink rate is lower while the person is lying, but also the mean inter-blink interval is lower when the subject is telling the truth.

In order to test the effect of unexpected questions on a subject’s blink rate, we can have a look at what happens in Filippo’s video in the interval (1:58 - 2:35), when he’s talking about his fake skying experience last Sunday (see 3). In the interval (1:58 - 2:25) his mean blink rate is 39 and after the unexpected question (2:25 - 2:35) his blink rate decreases to 32, which in terms of percentages is -36% from the average blink rate during

[14]The order in which the two target periods were shown was counterbalanced.
[15]The subjects were instructed to tell full lies.
[16]Seven videos from the second setting where recored at either 7 fps or 12 fps, which led to a weak blink detection performance.

	Baseline	**Truth**	**Baseline**	**Lie**	**Baseline**
Blink Rate (blink/min)	12	29	13	25	12
MEAN Inter-Blink (sec)	1.04	0.46	0.9	0.54	1
SD Inter-Blink (sec)	0.5	0.2	0.8	0.28	0.46

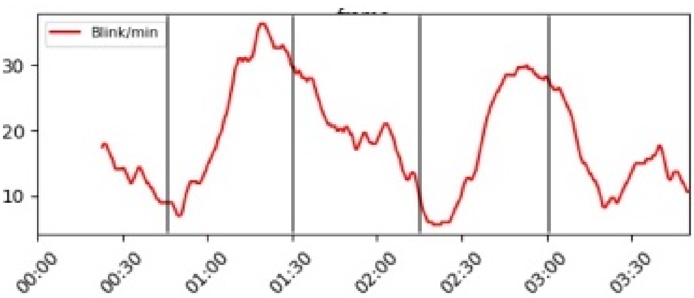


Table 8: Descriptive statistics during the 5 periods for Bianca's video.

	Baseline	**Lie**	**Baseline**	**Truth**	**Baseline**
Blink Rate (blink/min)	43	50	42	57	43
MEAN Inter-Blink (sec)	1.40	1.2	1.5	1.1	1.2
SD Inter-Blink (sec)	0.8	0.8	1.1	0.6	0.6

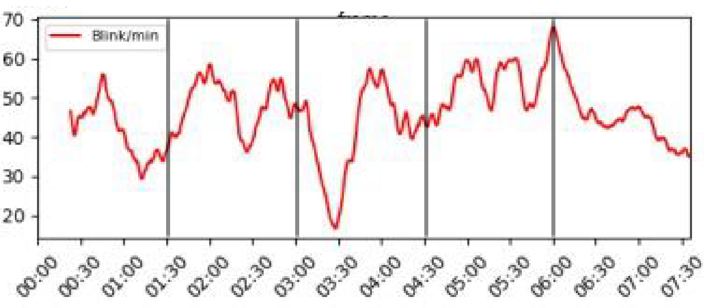


Table 9: Descriptive statistics during the 5 periods for Filippo's video.

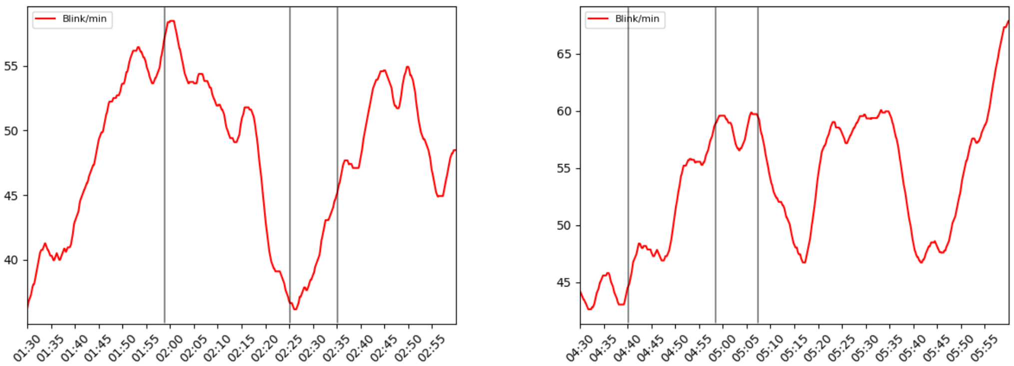


Figure 3: The effect of unexpected question in Filippo during lie (left) and truth (right).

the whole lie period. The same decrease can be shown for interval (4:40 - 5:07) while he is telling about his last Sunday experience in Padova: before the unexpected question (4:40 - 4:58) his average blink rate is 67, while during the answer (4:58 - 5:07) it decreases to 47. However, in the second target period the average blink rate is 57, which indicates that the unexpected question makes the subject decrease his blink rate by only 18%.

4. Conclusion

This work has firstly developed a robust and efficient blink detector, based on the *eye-aspect ratio* (EAR) value given as input to a fully-trained and validated SVM machine learning model. The performance was much higher than the existing OpenCV blink detector, under many video qualities and conditions. Secondly we used this blink detector to analyse all possible variations of some empirical measures in different subjects, during the baseline and the target period. In order to do this, we have developed a video interface embedded with some live statistical measures.

Finally, we have tried to analyse from a descriptive point of view the data displayed, with the intent to find out some analogies with the current research in the topic of lie detection and eye blinking. Due to the type of experimental design, it was not feasible to compare our results with those obtained for example in Leal and Vrij [8]: in our experiment, the subject's activity was not the same during the baseline and the target period. Talking is a factor that definitly influences eye blinking. However, it was found a significant effect of the unexpected questions on the blink rate, especially when lying.

We believe that a more accurate experimental design and a possible live blink detection implementation could lead to a complete lie detector, which classifies a subject or a part of his speech as a lie.

References

[1] Anna Rita Bentivoglio, Susan B Bressman, Emanuele Cassetta, Donatella Carretta, Pietro Tonali, and Alberto Albanese. Analysis of blink rate patterns in normal subjects. *Movement Disorders*, 12(6):1028–1034, 1997.

[2] J Craig. Structure and function of the preocular tear film. *The Tear Film: structure, function and clinical examination*, pages 18–50, 2002.

[3] Charles Darwin and Phillip Prodger. *The expression of the emotions in man and animals*. Oxford University Press, USA, 1998.

[4] Bella M DePaulo, Deborah A Kashy, Susan E Kirkendol, Melissa M Wyer, and Jennifer A Epstein. Lying in everyday life. *Journal of personality and social psychology*, 70(5):979, 1996.

[5] Paul Ekman and Wallace V Friesen. Detecting deception from the body or face. *Journal of personality and Social Psychology*, 29(3):288, 1974.

[6] Paul Ekman and Wallace V Friesen. Facial action coding system. 1977.

[7] Paul Ekman, Gowen Roper, and Joseph C Hager. Deliberate facial movement. *Child development*, pages 886–891, 1980.

[8] Sharon Leal and Aldert Vrij. Blinking during and after lying. *Journal of Nonverbal Behavior*, 32(4):187–194, 2008.

[9] MA Lemp and DE Wolfley. The lacrimal apparatus. *Adler's Physiology of the Eye, edited by Hart WM Jr. St. Louis, MO: Mosby*, pages 18–28, 1992.

[10] Merylin Monaro, Luciano Gamberini, and Giuseppe Sartori. The detection of faked identity using unexpected questions and mouse dynamics. *PloS one*, 12(5):e0177851, 2017.

[11] Adrian Rosebrock. Eye blink detection with opencv, python, and dlib. URL `https://www.pyimagesearch.com/2017/04/24/eye-blink-detection-opencv-python-dlib/`.

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

[13] Jie Shen, Stefanos Zafeiriou, Grigoris G Chrysos, Jean Kossaifi, Georgios Tzimiropoulos, and Maja Pantic. The first facial landmark tracking in-the-wild challenge: Benchmark and results. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 50–58, 2015.

[14] T Soukupova and Jan Cech. Real-time eye blink detection using facial landmarks. In *21st Computer Vision Winter Workshop (CVWW 2016)*, pages 1–8, 2016.

[15] Ingo Steinwart, Don Hush, and Clint Scovel. Learning from dependent observations. *Journal of Multivariate Analysis*, 100(1):175–194, 2009.

[16] Lin Su and Martin D Levine. High-stakes deception detection based on facial expressions. In *Pattern Recognition (ICPR), 2014 22nd International Conference on*, pages 2519–2524. IEEE, 2014.

[17] Aldert Vrij, Sharon Leal, Pär Anders Granhag, Samantha Mann, Ronald P Fisher, Jackie Hillman, and Kathryn Sperry. Outsmarting the liars: The benefit of asking unanticipated questions. *Law and human behavior*, 33(2):159–166, 2009.

A. Appendix

We built two programs: `video_to_plot` and `video_to_doublevideo`. Both programs take as input a video[17] and estimate blinks as described in 3.3. The outputs of the two programs are different:

- `video_to_plot` saves a `jpg` file called `plot.jpg` in the program folder. The file `plot.jpg` contains the blink rate series, where for every frame x at instant t (in seconds) the value of the blink rate is computed considering the number of blinks belonging to the frames in the interval $(t-20, t)$, i.e. the 20 seconds time window.

- `video_to_doublevideo` creates a folder called `final_video` in the program folder. This folder contains as many `jpg` files as the number of the frames of the video given as input. The `jpg` files are called `frame*.jpg`, where `*` is the number of the frame to which they relate. The content of these images is described in 3.4. The images generated by this program can be combined to create a video using a video editing program. For this aim, we used *ImageJ*[18].

The main reason why we built two different programs is their clear difference in computational time. The first program is lighter: a Macbook Pro with an Intel i5 Processor and 8 GB RAM takes 3 minutes for processing a minute of a 30 fps video. The second one is heavier: a Macbook Pro with an Intel i5 Processor and 8 GB RAM takes 35 minutes for processing a minute of a 30 fps video.

We decided to build programs that store the obtained results and so, do not produce a "real-time" interface: firstly, the computational effort to build a real-time interface is much bigger than the effort required to generate a plot; secondly, again from a computational point of view, a real-time output is not much less expensive than generating frames to build up videos. However, we considered that an output that can be examined many times without the use of the program is worth the extra computational effort done.

In addition, both these programs can be parallelized due to the fact that:

- Finding face landamarks and computing the EAR value is an independent operation, for every frame.

- Given the blink's estimate, creating `frame*.jpg` is independent, for every frame.

These are definitely the most computationally expensive operations. If we parallelize them, the computation time will be highly reduced.

[17] Note that if the video given as input is less than 20 seconds, the program outputs error. Moreover, if the video is either too short (less than 1 minute) or too long (more than 15 minutes) the plot could present some aesthetic problems.

[18] Freely available at https://imagej.nih.gov/ij/