Real-time Multi-person Eyeblink Detection in the Wild for Untrimmed Video

Wenzheng Zeng¹, Yang Xiao†¹, Sicheng Wei¹, Jinfang Gan¹, Xintao Zhang¹, Zhiguo Cao¹, Zhiwen Fang²,³ and Joey Tianyi Zhou⁴,⁵

¹Key Laboratory of Image Processing and Intelligent Control, Ministry of Education, School of Artificial Intelligence and Automation, Huazhong University of Science and Technology, Wuhan 430074, China
²School of Biomedical Engineering, Southern Medical University, Guangzhou 510515, China
³Department of Rehabilitation Medicine, Zhujiang Hospital, Southern Medical University, Guangzhou 510280, China
⁴Centre for Frontier AI Research, Agency for Science, Technology and Research (A*STAR), Singapore
⁵Institute of High Performance Computing, Agency for Science, Technology and Research (A*STAR), Singapore

{wenzhengzeng, Yang_Xiao, sichengwei, jinfangan, zgcao}@hust.edu.cn, fzw310@smu.edu.cn, zhouty@cfar.a-star.edu.sg

https://github.com/wenzhengzeng/MPEblink

Abstract

Real-time eyeblink detection in the wild can widely serve for fatigue detection, face anti-spoofing, emotion analysis, etc. The existing research efforts generally focus on single-person cases towards trimmed video. However, multi-person scenario within untrimmed videos is also important for practical applications, which has not been well concerned yet. To address this, we shed light on this research field for the first time with essential contributions on dataset, theory, and practices. In particular, a large-scale dataset termed MPEblink that involves 686 untrimmed videos with 8748 eyeblink events is proposed under multi-person conditions. The samples are captured from unconstrained films to reveal “in the wild” characteristics. Meanwhile, a real-time multi-person eyeblink detection method is also proposed. Being different from the existing counterparts, our proposition runs in a one-stage spatio-temporal way with end-to-end learning capacity. Specifically, it simultaneously addresses the sub-tasks of face detection, face tracking, and human instance-level eyeblink detection. This paradigm holds 2 main advantages: (1) eyeblink features can be facilitated via the face’s global context (e.g., head pose and illumination condition) with joint optimization and interaction, and (2) addressing these sub-tasks in parallel instead of sequential manner can save time remarkably to meet the real-time running requirement. Experiments on MPEblink verify the essential challenges of real-time multi-person eyeblink detection in the wild for untrimmed video. Our method also outperforms existing approaches by large margins and with a high inference speed.

†Yang Xiao is corresponding author (Yang_Xiao@hust.edu.cn).

1. Introduction

Real-time eyeblink detection in the wild is a recently emerged challenging research task [19] that can widely serve for fatigue detection [2], face anti-spoofing [34], affective analysis [7], etc. Although remarkable progress has been made [9, 10, 19], the existing methods generally focus on single-person cases within trimmed videos. Multi-person scenario within untrimmed videos has not been well concerned yet. However, detecting long-term eyeblink behaviors at multi-instance level is more preferred for some practical application scenarios. For example, it can be used to estimate attendees’ attention level and emotional state change during social interaction [9, 33]. Thus, effective and real-time multi-person eyeblink detection in the wild for untrimmed video is indeed required.

To this end, we shed the light on this research problem with essential contributions on dataset, theory, and practices. First, a challenging labeled multi-person eyeblink de-
tection benchmark termed MPEblink is built under in-the-wild conditions. It consists of 686 untrimmed long videos captured from unconstrained movies to reveal the “in the wild” characteristics. The contained scenarios are realistic and diverse, such as social interactions and group activities. To our knowledge, MPEblink is the first multi-person eyeblink detection dataset that focuses on in-the-wild long videos. Fig. 1 illustrates a sample video with ground truth annotations within it. Different from the existing eyeblink detection benchmarks [10, 12, 19], the proposed benchmark aims at being aware of all the attendees along the whole video and detecting their eyeblinks at instance level. In summary, MPEblink is featured with multi-instance, unconstrained, and untrimmed, which makes it more challenging and realistic than previous formulations.

To perform eyeblink detection, previous methods [9, 10, 12, 19, 40] generally take a sequential approach containing face detection, face tracking, and classification on the pre-extracted local eye clues within a temporal window. Whereas such a pipeline seems reasonable, it has several critical drawbacks. First, the use of isolated components may lead to sub-optimal results as they are not jointly optimized and it is inefficient due to the inability to reuse the features of each stage. Second, the eyeblink features only contain the information from the pre-extracted local eye clues (i.e., a small part of face), lacking the useful information from global face context such as head pose and illumination condition that are crucial for detecting eyeblinks in the wild. Moreover, the pre-extracted local eye clues are also unreliable due to the localization challenge towards unconstrained scenarios. Third, the sequential approach leads to computational cost being sensitive to the subject amount, which is hard to meet the real-time running requirement.

To tackle these issues, we propose a one-stage multi-person eyeblink detection framework called InstBlink, which can simultaneously detect human faces, track them over time, and do eyeblink detection at instance level. InstBlink takes inspiration from the existing query-based methods [41, 45, 47, 48] and models the spatio-temporal face as well as eyeblink representations at instance level within each query. The insight is that the features can be effectively shared among these sub-tasks and the eyeblink features can be facilitated via face’s global contexts (e.g., head pose and illumination condition) with joint optimization and interaction, especially in unconstrained in-the-wild cases. Experiments verify the superiority of InstBlink in both effectiveness and efficiency, while also highlighting the critical challenges of real-time multi-person eyeblink detection in the wild for untrimmed video.

The main contributions of this work lie in 3 folders:

- To our knowledge, it is the first time that instance-level multi-person eyeblink detection in untrimmed videos is formally defined and explored;
- We introduce an unconstrained multi-person eyeblink detection dataset MPEblink that contains 686 untrimmed videos with 8748 eyeblink events, featured with more realistic and challenging;
- We propose a one-stage multi-person eyeblink detection method that can jointly perform face detection, tracking, and instance-level eyeblink detection. Such a task-joint paradigm can benefit the sub-tasks uniformly.

2. Related Work

Eyeblink detection dataset. Existing eyeblink detection datasets [1, 10, 12, 13, 34, 36] generally focus on constrained indoor cases with a consistent environmental setup. HUST-LEBW [19] extends the scenarios to the unconstrained outdoor cases. From then on, the community starts to pay more attention to eyeblink detection in the wild. Nevertheless, HUST-LEBW mainly focuses on single-person scenarios towards trimmed videos, which limits the application of methods in more realistic scenarios such as group activities and social interactions. To address such limitation, we introduce a new dataset termed MPEblink. The proposed dataset is featured with multi-person, unconstrained, and untrimmed, thus being more realistic and challenging.

Eyeblink detection method. Existing methods generally judge eyeblinks from the pre-extracted local eye clues (e.g., local eye region [6, 9, 10, 19, 25] or landmarks around eyes [8, 14, 35, 40]). To obtain the local eye clues, the existing works generally run in a sequential way including face detection, face tracking, and landmark detection. Finally, the eyeblink result is classified from the pre-extracted local eye clues within a temporal window. Such a pipeline is inefficient due to the isolated optimization among different sub-tasks. Meanwhile, the eyeblink features only contain the information from the pre-extracted local eye clues, lacking useful global information such as head pose and illumination, and the pre-extracted eye clues are also unreliable due to the localization challenge toward unconstrained cases. Besides, due to the multi-stage characteristic, existing methods do not scale well with the number of people in the scene to meet the real-time running requirement. In this paper, we propose a one-stage eyeblink detection framework that can simultaneously detect human faces, track them along the time, and do eyeblink detection at instance level. Within it, eyeblink features can be facilitated via global face context, and the features can be effectively shared among the sub-task to achieve high inference speed.

Spatio-temporal action detection. Spatio-temporal action detection [15, 20, 22, 26, 27, 39, 48] aims at simultaneously obtaining the spatial and temporal location of an action, which is defined as an action tube [22]. The task only focuses on the isolated action tubes without instance awareness and thus can not make an instance-level analysis along the whole video. Different from that, our proposed multi-
person eyeblink detection task aims at simultaneously being aware of the presence of people and analyzing their eyeblink behaviors at instance level.

**Query-based methods.** Query-based methods are impacting many computer vision tasks. After DETR [4], the pioneer that introduces the query-based framework to address object detection, numerous efforts have been subsequently paid to facilitate visual recognition under the query concept in the fields of object detection [24, 32, 41, 49], pose estimation [38], action detection [30, 48], and segmentation [5, 43, 45, 47]. Inspired by these, we build the first one-stage multi-person eyeblink detection framework for untrimmed video. We propose a unified solution to simultaneously model the instance-level face and eyeblink representations in the whole video.

### 3. The MPEblink Benchmark

Existing eyeblink detection datasets generally focus on single-person scenarios, and are also limited in aspects of constrained conditions or trimmed short videos. To explore unconstrained eyeblink detection under multi-person and untrimmed scenarios, we construct a large-scale multi-person eyeblink detection dataset termed MPEblink to shed light on this research topic that has not been well studied before. The distinguishing characteristics of MPEblink lie in 3 aspects: multi-person, unconstrained, and untrimmed long video, which makes it more realistic and challenging than previous datasets.

#### 3.1. Task Formulation

We think that a good multi-person eyeblink detection algorithm should be able to (1) detect and track human instances’ faces reliably to ensure the instance-level analysis ability along the whole video, and (2) detect eyeblink boundaries accurately within each human instance to ensure the precise awareness of their eyeblink behaviors.

Formally, for an untrimmed video with $T$ frames and $N_{gt}$ people in it, towards the $j$-th person, let $\mathbf{u}_j \in \mathbb{R}^{T \times 4}$ denote its face bounding boxes across the video. $\mathbf{c}_j \in \mathbb{R}^T$ is given as the face labels to reflect the face existence of this instance in each frame (i.e., $\mathbf{c}_j^t \in \{1, 0\}$, where $\mathbf{c}_j^t = 0$ indicates that the face is not visible due to serious occlusion or departure at time $t$). $\mathbf{P}_j$ is also set to $\emptyset$ when the person is not visible. We use $\mathbf{B}_j$ and $\mathbf{c}_j$ to represent the instance-level information of each person. Suppose person $j$ blinks $K_j$ times in the video. Let $\mathbf{B}_j = [\mathbf{s}_j, \mathbf{e}_j]$ denote the $k$-th eyeblink event within this person, where $\mathbf{s}_j^k$ and $\mathbf{e}_j^k$ denote its starting and ending time. Suppose a multi-person eyeblink detection algorithm produces $H$ human instance hypotheses. Within each human instance hypothesis $i$, it needs to produce a sequence of predicted face bounding boxes $\mathbf{u}_i$, face classification scores $\mathbf{c}_i$, and $K_i$ eyeblink proposals $\mathbf{B}_i = [\mathbf{s}_i, \mathbf{e}_i]$. Better performance can be acquired when predictions get closer to ground truths.

#### 3.2. Data Collection

We collect 686 untrimmed video clips with various lengths (i.e., 7.1-85.9s) from 86 different unconstrained movies that involve the “in the wild” characteristics. Some snapshots of the acquired samples are shown in Fig. 2. It can be observed that the samples are with variational indoor and outdoor scenarios for different themes. Thus, the
acquired samples from these movies are much more realistic and closer to practical applications. Meanwhile, due to the high divergence of the selected movie data source, the captured multi-person eyeblink samples are also of great challenges for effectively detecting eyeblinks. For instance, the pose, illumination condition, expression, and face size among instances differ a lot, even within the same video. Moreover, because of the various types of interaction among instances, there may exist severe occlusion or people in and out scenarios as shown in Fig. 3, making it harder for both instance localization and their eyeblink event detection. Besides, different from the existing unconstrained counterpart [19] that mainly focuses on trimmed short videos, the samples from MPEblink are untrimmed. As a result, we not only need to recognize eyeblink but also need to locate the starting and ending time of each eyeblink event, which is more challenging especially when rapid and consecutive eyeblinks occur. An example is shown in Fig. 4. It can be observed that the appearance difference among eye statuses is small when consecutive rapid eyeblinks happen, making it hard to distinguish the boundary of each eyeblink. Considering the above challenges together, it can be summarized that unconstrained multi-person eyeblink detection in untrimmed videos is indeed a challenging research task.

3.3. Data Annotation

For human instances in each video, we labeled their face bounding boxes exhaustively across the whole video. To facilitate the research of landmark-based methods and include a broader range of applications, 68 facial landmark positions [44] are also annotated. Technically, this part of the annotation is under a semi-supervised manner: we use the state-of-the-art face analysis engine InsightFace [21] to achieve face bounding box [16] and landmark [44] detection. To track instances across frames, a matching strategy that considers bounding box IoU and similarity among deep face features [11] is designed. Besides, human annotators carefully check the annotation quality and correct the wrong bounding boxes and tracking results generated by the algorithm. For eyeblink events within each instance, human annotators carefully define their start and end frame. Finally, 8748 eyeblink events are labeled.

3.4. Data Statistics

The dataset statistics comparison among MPEblink and the other eyeblink detection datasets are given in Table 1. Actually, MPEblink provides the largest number of eyeblink events (i.e., 8748) within 686 untrimmed videos. The biggest difference from the previous datasets is that the videos in our dataset contain various number of human instances (i.e., 1-8) with exhaustive annotation. The videos are captured under unconstrained in-the-wild conditions with high diversity as shown in Fig. 2. Compared with the existing in-the-wild counterpart [19] that generally focuses on trimmed short videos, the length of videos from our dataset is much longer and more diverse. In summary, MPEblink is featured with more realistic and challenging and thus has broader application values.

3.5. Evaluation Metrics

Existing metrics for eyeblink detection only focus on single-person cases without multi-instance awareness. To address this limitation we introduce 2 new metrics termed Inst-AP and Blink-AP to give attention to both instance awareness quality and eyeblink detection quality.

Inst-AP. It aims to evaluate the instance detection and tracking ability of methods, which is the basis for analyzing the eyeblink behaviors of each instance. We modify the standard evaluation metric video-AP in spatio-temporal action detection [22] to fit our task. Different from the original implementation, the 3D IoU is calculated between each instance proposal (i.e., a sequence of face bounding boxes) and ground truth across the whole video, rather than the action tube only. As a result, the proposed metric can reveal the algorithm’s ability for detection and tracking instances, while video-AP can just reveal the localization ability of the isolated action tube without instance-level awareness. We report the averaged Inst-AP under the IoU from 50% 95% with step 5%. During calculating Inst-AP, each obtained true positive (TP) prediction has matched with one ground-truth instance, and thus, we can calculate the eyeblink detection accuracy of these TP predictions using the proposed Blink-AP. The TP predictions under IoU of 50% are used to calculate the subsequent Blink-AP.

Blink-AP. This metric is used to reflect the model’s eyeblink detection ability within each instance. We tailor the standard AP metric in temporal action detection [3] into our task. Only the matched (i.e., true positive) predictions when calculating Inst-AP are taken into consideration in this metric as the blink accuracy of these predictions is rarely affected by the face detection and tracking accuracy. We report the Blink-AP under the temporal IoU of 50% and 75%.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Num. of videos</td>
<td>1</td>
<td>80</td>
<td>8</td>
<td>5</td>
<td>107</td>
<td>6000</td>
<td>673</td>
</tr>
<tr>
<td>Video length</td>
<td>3.3m</td>
<td>4.1-4.7s</td>
<td>2.7-8.9m</td>
<td>1.4-2.7m</td>
<td>0.3-3.4m</td>
<td>0.6s</td>
<td>0.5s</td>
</tr>
<tr>
<td>Unconstrained</td>
<td>✓</td>
<td>×</td>
<td>×</td>
<td>✓</td>
<td>×</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Instances/video</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Blink count</td>
<td>61</td>
<td>255</td>
<td>353</td>
<td>300</td>
<td>1867</td>
<td>3000</td>
<td>381</td>
</tr>
</tbody>
</table>
4. Method

In this section, we present our InstBlink that takes a video clip as input and directly outputs the face positions of each instance and their eyeblink intervals in the whole video clip. The InstBlink design takes inspiration from the existing query-based methods \cite{4,41,45,47,48} but formulates the architecture to model both face and eyeblink representations at instance level along the video clip. The overall architecture is illustrated in Fig. 5.

Given a video clip \( I \in \mathbb{R}^{T \times 3 \times H \times W} \) where \( T \) denotes the number of frames and \( H \times W \) is the spatial size of each frame, InstBlink first applies a backbone network to extract video feature \( F_v \in \mathbb{R}^{T \times C \times H' \times W'} \) where \( C \) is the channel number and \( H' \times W' \) is the spatial size of the feature. Afterwards, a query-based architecture iterates \( M \) times, which consists of 3 components: the Query Interaction Module (QIM), Video Interaction Module (VIM), and task-specific heads (i.e., face head and blink head). At the end of each interaction, the queries will be updated and the instance-level face and eyeblink predictions will be output by the task-specific heads. During inference, the output from the last iteration will be used as the final prediction results.

4.1. Instance Query

Within InstBlink, the spatio-temporal instance queries \( \{ q^i \}_{i=1}^N \) are responsible for characterizing every human instance’s joint face and eyeblink features in a video. Each query contains \( T \) embeddings (i.e., \( q^i \in \mathbb{R}^{T \times C} \)), where \( C \) is the feature dimension. Each embedding generally focuses on the instance’s face and eyeblink representations in the corresponding frame. Each query is also paired with a proposal tube \( p^i \in \mathbb{R}^{T \times 4} \) (i.e., spatio-temporal bounding boxes across the time), which aims at indicating the face location of \( i \)-th instance across the entire video clip. At the first iteration of each complete forward propagation, \( q^i \) and \( p^i \) are initialized by copying 2 learnable parameters \( \tilde{q}^i \in \mathbb{R}^{1 \times C} \) and \( \tilde{p}^i \in \mathbb{R}^{1 \times 4} \) \( T \) times along the temporal dimension.

4.2. Query Interaction Module (QIM)

QIM targets at (1) enhancing the association between the specific query and its corresponding human instance and (2) modeling spatio-temporal face and eyeblink representations of the associated instance. Specifically, we first adopt a spatial self-attention layer to allow spatial interaction among queries within each frame:

\[
\{ q^i \}_{i=1}^N = \text{MHSA} \left( \{ q^i \}_{i=1}^N \right), \quad t \in [1, T],
\]

where MHSA is the multi-head self-attention \cite{42}. Spatial interaction can build strong communication among queries to better model the instance features under complex circumstances such as occlusion due to human interactions. Then, temporal self-attention is used within each query along the temporal dimension to realize temporal interaction:

\[
\{ q^i \}_{i=1}^T = \text{MHSA} \left( \{ q^i \}_{i=1}^T \right), \quad i \in [1, N].
\]

Applying temporal interaction within each query allows the embeddings from different frames to communicate with each other to facilitate instance tracking and model temporal eyeblink representations of the corresponding instance.

4.3. Video Interaction Module (VIM)

VIM aims at collecting the face and eyeblink information of the target instance from the video feature. Particularly, the dynamic filters \( m^i \) are first generated from each query embeddings \( q^i \). Then, the filters will filter an RoI feature by dynamic convolution to extract highly related features for both face and eyeblink. The RoI feature is obtained by applying RoI align \cite{17} on the video feature using the proposal tube \( p^i \). After obtaining the filtered feature, a linear projection is applied to form the updated query feature \( q^i \). The updated query feature will be used to make both face and eyeblink predictions by the task-specific heads.

4.4. Task-specific Heads

The instance-level face and eyeblink predictions can be obtained simultaneously by applying task-specific heads on the query features. The heads are shared among queries. **Face head.** Given the updated query feature \( \tilde{q}^i \), a Multilayer Perceptron (MLP) layer with Sigmoid normalization is used to indicate the existence of human faces as

\[
e^f = \text{Sigmoid} \left( \text{MLP}_c \left( \tilde{q}^i \right) \right),
\]

\[
e^b = \text{Sigmoid} \left( \text{MLP}_b \left( \tilde{q}^i \right) \right),
\]

\[
e^e = \text{Sigmoid} \left( \text{MLP}_e \left( \tilde{q}^i \right) \right),
\]

**Blink head.**
where \( c_i \in \mathbb{R}^T \) denotes the face classification scores across frames. Face localization is achieved in a similar way as

\[
l^i = MLP_{l} (\tilde{q}^i),
\]

where \( l^i \in \mathbb{R}^{T \times 4} \) denotes the face bounding boxes across frames. It will also be used to renew the proposal tube \( p^i \).

**Blink head.** Eyeblink prediction towards \( \tilde{q}^i \) is achieved by an MLP layer with Sigmoid normalization as

\[
b^i = \text{Sigmoid} \left( MLP_{b} (\tilde{q}^i) \right),
\]

where \( b^i \in \mathbb{R}^T \) is the frame-level eyeblink scores. Each element \( b_i^k \) in \( b^i \) indicates the possibility that the \( t \)-th frame is inside an eyeblink event. During inference, we simply merge the adjacent frame-level eyeblink predictions based on a threshold (e.g., 0.3 in our implementation) to form the final interval-level eyeblink predictions \( B_t^k = [s_k^t, e_k^t] \) where \( k \in K \) is the \( k \)-th eyeblink interval within the total \( K \) eyeblink predictions. An intuitive example of the merging procedure is shown in Fig. 6.

Different from the existing works that only extract features from the local eye region, our blink head also considers useful global information such as global face appearance, head pose, and illumination condition to aid the implicit eye localization and eyeblink characterization. The reason is that the blink head directly filters global features from the query embeddings where the global face context is already stored. During training, the eyeblink clues will not only flow back to the RoI feature, but also to the instance query to enhance the eyeblink-related feature representation ability of queries. Moreover, because of the shared feature among blink head and face head, a kind of interaction and synergy is built where the information flow from blink head can also benefit the face representations to boost the face detection and tracking ability as verified in Sec. 5.3.

As we have \( N \) queries, we finally obtain the predictions from them in parallel:

\[
\{ y^j \}_{j=1}^{N} = \left\{ (c^i, l^i, b^i) \right\}_{i=1}^{N}.
\]

**4.5. Training**

To optimize InstBlink, one-to-one assignment between instance-level predictions \( \{ y^j \}_{j=1}^{N} \) and ground truths is first conducted based on their instance-level face tracklet similarity. Then, the loss is calculated concerning face and eyeblink items jointly for optimization. As in Sec. 3.1, the ground truth annotations can be summarized as

\[
\{ \tilde{y}^j \}_{j=1}^{N_{gt}} = \left\{ \left( \tilde{c}^j, \tilde{l}^j, \tilde{b}^j \right) \right\}_{j=1}^{N_{gt}},
\]

where \( \tilde{c}^j \in \mathbb{R}^T, \tilde{l}^j \in \mathbb{R}^{T \times 4}, \) and \( \tilde{b}^j \in \mathbb{R}^T \) indicate the frame-level face existence, face position (i.e., bounding box) and eyeblink existence (i.e., \( \tilde{b}^j_i = 1 \) indicates frame \( i \) is inside an eyeblink event and conversely for \( \tilde{b}^j_i = 0 \), which can be derived from \( B_t^j \) using an inverse process in Fig. 6) respectively. We use Hungarian algorithm [23] to perform instance-level bipartite matching between predictions and ground truths. The matching cost is given as

\[
L_{\text{Hung}} (y^j, \tilde{y}^j) = \sum_{t=1}^{T} \left[ L_{\text{cls}} (\tilde{c}^j_t, c^j_t) + 1_{\{\tilde{c}^j_t \neq 0\}} \left( L_{\text{box}} (\tilde{l}^j_t, l^j_t) \right) \right],
\]

where \( L_{\text{cls}} \) indicates the focal loss [29] for face existence classification, \( L_{\text{box}} \) is the combination of L1 loss and Glou loss [37] for face localization. The loss weights are the same as [41, 47]. After obtaining the optimal assignment \( \hat{\sigma} \) between predictions and ground truths, the network is optimized by the loss as

\[
L = \sum_{j=1}^{N_{gt}} \left( L_{\text{face}} \left( \tilde{y}^j, \tilde{y}^j_{\hat{\sigma}(j)} \right) + \lambda L_{\text{blink}} \left( \tilde{y}^j, \tilde{y}^j_{\hat{\sigma}(j)} \right) \right),
\]

where \( \tilde{y}^j_{\hat{\sigma}(j)} \) is the prediction that has been matched with the ground truth instance \( \tilde{y}^j \). \( L_{\text{face}} \) is the face tracklet detection loss that shares the same form as \( L_{\text{Hung}} \) (i.e., \( L_{\text{face}} \left( \tilde{y}^j, \tilde{y}^j_{\hat{\sigma}(j)} \right) = L_{\text{Hung}} \left( \tilde{y}^j, \tilde{y}^j_{\hat{\sigma}(j)} \right) \)), and \( L_{\text{blink}} \) denotes instance-level eyeblink detection loss as \( L_{\text{blink}} \left( \tilde{y}^j, \tilde{y}^j_{\hat{\sigma}(j)} \right) = \sum_{t=1}^{T} L_{\text{cls}} \left( c^j_t, \tilde{c}^j_t \right) \) with \( \lambda = 5 \). For unmatched predictions, only \( \sum_{t=1}^{T} L_{\text{cls}} \left( c^j_t, \tilde{c}^j_t \right) = 0 \) is used to supervise their face existence classification output \( c^j \) to be close to 0 (i.e., no face).

**5. Experiment**

**Dataset and evaluation metrics.** We first conduct experiments on the proposed MPEblinks dataset. We split the dataset into 423 training videos, 128 validation videos, and 135 test videos. The videos from the same movie will only appear in one set. We train the network on the training set, validate it on the validation set and report the performance on the test set. We report Inst-AP to reveal the instance localization ability and Blink-AP to reveal the eyeblink detection ability within instances, as introduced in Sec. 3.5. Meanwhile, experiments are also conducted on HUST-LEBW for single-person and trimmed cases. Recall, Precision, and F1 score are reported following [19].
Implementation details. We use ResNet-50-FPN [18, 28] backbone. The network is pre-trained on YouTube-VIS [46] under [47] for a general instance representation ability. The query number $N$ and iteration time $M$ are set to 50 and 4 respectively. AdamW [31] optimizer with a batch size of 8 is used to train the model. The initial learning rate is set to 2.5e-5 for the backbone and 2.5e-4 for the other parts. At the training stage, for the sake of memory efficiency, we use the frame rate of 12 FPS for image sampling and the input clip length is set to 11 that is longer than most eyeblink events (i.e., 0.2-0.4s). The frames are also resized to 640 × 360 before sending into the network. The whole training procedure lasts for 10,000 iterations and the learning rate is multiplied by 0.1 at iteration 6000 and 9000. During test, the original frame rate is used and the input clip length is set to 36 with a stride of 18. The predictions within adjacent clips are linked via concerning face bounding box IoU.

5.1. Benchmark Results on MPEblink Dataset

Baselines. Multi-person eyeblink detection in untrimmed videos is a new task that has not been well concerned before. Therefore, we tailor the existing eyeblink detection approaches with a unified instance detection and tracking method. Specifically, we use the state-of-the-art face analysis toolbox InsightFace [21] to achieve face [16] and landmark [44] detection. To track each instance, we link the face bounding boxes from the adjacent frames based on their similarity on box IoU. After obtaining the instance tracklets (i.e., a sequence of face bounding boxes and landmarks), we employ 4 representative eyeblink detection method [9, 19, 35, 40] within each instance tracklets. Such a sequential pipeline is commonly used in the existing approaches under the single-person assumption.

Main results. From Table 2, it can be summarized that:

- All of the multi-person eyeblink detection algorithms cannot achieve a satisfactory performance (i.e., Blink-AP$_{50}$ lower than 30% and Blink-AP$_{75}$ lower than 10%). This indicates that eyeblink detection under multi-person, untrimmed, and unconstrained scenarios is indeed challenging and has not been well solved yet.
- For Blink-AP, InstBlink significantly outperforms the others by large margins (i.e., 21% at least of Blink-AP$_{50}$ and 6% at least of Blink-AP$_{75}$), which verifies the superiority of the proposed framework. We argue that one essential reason is that our framework can model a better long-term temporal eyeblink representation than the frame-based method [9] and the sliding window based methods [19, 35, 40]. Moreover, under the proposed framework, eyeblink features can be facilitated via face’s global context (e.g., head pose and illumination condition) with joint optimization and interaction, while previous works that utilize a sequential manner cannot. From Table 2, it can also be summarized that the landmark-based methods [35, 40] perform poorer than the region-based counterparts. We think that one essential reason is that landmark detection is unreliable under unconstrained conditions. A similar conclusion has already been made in [19] but the property of multi-person and long video in MPEblink make it worse for landmark-based eyeblink detection methods. Our method is region-based, but localizes eye region in an implicit way where global face context is included and thus becomes more robust.
- InstBlink also outperforms others on Inst-AP. We think that is because our framework can better model the long-term spatio-temporal instance representations, while the counterparts achieve tracking under a tracking-by-detection framework, which contains limited spatio-temporal modeling and may suffer from heavy occlusion.

Inference speed analysis. The result is listed in Table 3, assuming the 4 compared methods use InsightFace for face & landmark detection and InstBlink inferences within a clip length of 36. It can be seen that InstBlink is also of high inference speed (i.e., 112 FPS for network forwarding) while the real-time capacity of other methods is not superior. As the number of instances increases, their running time also increases while our method is not sensitive to the number of people because of the one-stage inference property.

5.2. Benchmark Results on HUST-LEBW Dataset

Experiments are also conducted on the HUST-LEBW dataset to explore the generalizability of InstBlink towards single-person and trimmed in-the-wild cases. The results are given in Table 4. For models trained on the HUST-LEBW dataset, our method still outperforms others by a large margin (i.e., 3.78% on F1 score), even with limited training data (less than 450 trimmed samples) to train our one-stage model for multiple tasks (face detection, tracking, and eyeblink detection). Meanwhile, the model trained on MPEblink can obtain 83.45% F1 score on HUST-LEBW. This indeed verifies the strong generalization ability of InstBlink and MPEblink towards eyeblink detection task.

5.3. Ablation Study

Spatial and temporal modeling in QIM. From Table 5, we can see that (1) without applying temporal interaction, the performance of Blink-AP drops from 27.19% to 4.58%. This indicates that temporal modeling is critical for eyeblink detection in untrimmed videos (i.e., only using appearance features can not accurately localize eyeblinks). It can also be observed that temporal interaction facilitates...
Table 3. The inference speed comparison on a single NVIDIA 3090 GPU, assuming the compared methods use InsightFace for face detection (time consumption $T=9.3$ ms including pre-processing) and landmark detection. #faces denotes the face amount in the scene.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Time/image (ms)</td>
<td>8.9 + 2.6 for data processing</td>
<td>$T=9.3$ + 5.4 × #faces</td>
<td>$T=5.4$ × #faces</td>
<td>$T=5.7$ × #faces</td>
<td>$T=9.1$ × #faces</td>
</tr>
</tbody>
</table>

Table 4. Performance comparison on the HUST-LEBW dataset, where InstBlink_cross indicates InstBlink trained on MPEblink and tested on HUST-LEBW.

<table>
<thead>
<tr>
<th>Training set</th>
<th>Method</th>
<th>Eye Recall</th>
<th>Precision</th>
<th>F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>HUST-LEBW [19]</td>
<td>Soukupová and Cech [40]</td>
<td>Left 36.07</td>
<td>64.71</td>
<td>46.32</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Right 30.16</td>
<td>57.58</td>
<td>39.58</td>
</tr>
<tr>
<td>mEBAL [10]</td>
<td>Hu et al. [19]</td>
<td>Left 54.10</td>
<td>69.19</td>
<td>67.35</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Right 44.44</td>
<td>76.71</td>
<td>56.28</td>
</tr>
<tr>
<td></td>
<td>Blink detection+ [35]</td>
<td>Both 58.99</td>
<td>80.05</td>
<td>67.90</td>
</tr>
<tr>
<td></td>
<td>InstBlink (Ours)</td>
<td>Both 97.64</td>
<td>56.62</td>
<td>71.68</td>
</tr>
<tr>
<td></td>
<td>Daza et al. [9]</td>
<td>Left 96.03</td>
<td>60.80</td>
<td>74.46</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Right 79.50</td>
<td>73.48</td>
<td>76.37</td>
</tr>
<tr>
<td>MPEblink</td>
<td>InstBlink_cross (Ours)</td>
<td>Both 91.34</td>
<td>76.82</td>
<td>83.45</td>
</tr>
</tbody>
</table>

Table 5. Ablation studies on QIM and VIM.

<table>
<thead>
<tr>
<th>Method</th>
<th>Blink-AP$_50$</th>
<th>Blink-AP$_75$</th>
<th>Inst-AP</th>
</tr>
</thead>
<tbody>
<tr>
<td>w/o QIM</td>
<td>3.20</td>
<td>0.39</td>
<td>58.93</td>
</tr>
<tr>
<td>w/o temporal interaction in QIM</td>
<td>4.58</td>
<td>0.55</td>
<td>63.61</td>
</tr>
<tr>
<td>w/o spatial interaction in QIM</td>
<td>26.65</td>
<td>6.18</td>
<td>62.45</td>
</tr>
<tr>
<td>w/o filter operation in VIM</td>
<td>22.27</td>
<td>4.59</td>
<td>65.01</td>
</tr>
<tr>
<td>Full model</td>
<td>27.19</td>
<td>7.16</td>
<td>67.89</td>
</tr>
</tbody>
</table>

Table 6. The performance comparison with and without blink head on the MPEblink dataset.

<table>
<thead>
<tr>
<th>Blink head</th>
<th>Inst-AP</th>
<th>Inst-AP$_50$</th>
<th>Inst-AP$_75$</th>
</tr>
</thead>
<tbody>
<tr>
<td>×</td>
<td>65.86</td>
<td>81.73</td>
<td>71.23</td>
</tr>
<tr>
<td>✓</td>
<td>67.89</td>
<td>84.51</td>
<td>73.76</td>
</tr>
</tbody>
</table>

the performance of Inst-AP. We argue that this is because it can also build instance-level association among the features across frames to facilitate the instance tracking ability.

(2) Spatial interaction can boost Inst-AP by a large margin (i.e., 5.44%) and slightly boost Blink-AP, as it may build strong communication among queries to better model the instance features under complex circumstances such as occlusion due to human interactions.

Filter operation in VIM. As in Table 5, compared with directly using RoI features to update query, using filtered RoI features can boost 4.92% on Blink-AP$_50$ and 2.88% on Inst-AP. We speculate that the filter operation can activate task-specific RoI features and resist background. Thus, finer face and eyeblink clues can be collected to update queries.

Multi-task learning mechanism. Here we study the effect of the blink head from a multi-task learning perspective. As shown in Table 6, adding blink head can also boost 2.03% on Inst-AP, which demonstrates that features for face detection and tracking can also benefit from eyeblink clues.

5.4. Qualitative Analysis

We visualize the prediction results of InstBlink in Fig. 7. The examples show that our model can aware the attendees robustly and detect their eyeblinks under different facial appearance and head pose. Nevertheless, it can not distinguish the rapid consecutive eyeblinks, which also reveals the challenge of eyeblink detection in untrimmed videos.

6. Conclusions and Limitations

In this work, we shed the light on a new research task termed multi-person eyeblink detection in the wild for untrimmed video. We formally define the task and introduce a multi-person eyeblink detection dataset named MPEblink. To perform multi-person eyeblink detection in untrimmed videos, we introduce a one-stage multi-person eyeblink detection framework InstBlink. Experiment results demonstrate the superiority of InstBlink in both effectiveness and efficiency. However, our SOTA performance is still unsatisfactory (i.e., Blink-AP$_50$ < 30%). This verifies the challenges of unconstrained multi-person eyeblink detection in long videos. Besides, although the proposed dataset focuses on multi-person, unconstrained, and untrimmed scenarios that are more realistic and have a broader application value, there are lack of crowd scenes (e.g., >10 instances) within it. In the future, we will pay more attention to the crowd scenes and enrich the dataset.

Acknowledgment

This work is jointly supported by the National Natural Science Foundation of China (Grant No. 62271221 and U1913602), and Natural Science Foundation of Guangdong Province (Grant No. 2023A1515011260). Joey Tianyi Zhou is funded by the SERC (Science and Engineering Research Council) Central Research Fund (Use-Inspired Basic Research), and the Singapore Government’s Research, and Innovation and Enterprise 2020 Plan (Advanced Manufacturing and Engineering Domain) under programmatic Grant A18A1b0045.
References


