Running Event Visualization using Videos from Multiple Cameras

Y. Napolean*
TU Delft
y.napolean@tudelft.nl

P.T. Wibowo*
TU Delft
priaditeguhwibowo@student.tudelft.nl

J.C. van Gemert
TU Delft
j.c.vangemert@tudelft.nl

ABSTRACT

Visualizing the trajectory of multiple runners with videos collected at different points in a race could be useful for sports performance analysis. The videos and the trajectories can also aid in athlete health monitoring. While the runners unique ID and their appearance are distinct, the task is not straightforward because the video data does not contain explicit information as to which runners appear in each of the videos. There is no direct supervision of the model in tracking athletes, only filtering steps to remove irrelevant detections. Other factors of concern include occlusion of runners and harsh illumination. To this end, we identify two methods for runner identification at different points of the event, for determining their trajectory. One is scene text detection which recognizes the runners by detecting a unique 'bib number' attached to their clothes and the other is person re-identification which detects the runners based on their appearance. We train our method without ground truth but to evaluate the proposed methods, we create a ground truth database which consists of video and frame interval information where the runners appear. The videos in the dataset was recorded by nine cameras at different locations during the a marathon event. This data is annotated with bib numbers of runners appearing in each video. The bib numbers of runners known to occur in the frame are used to filter irrelevant text and numbers detected. Except for this filtering step, no supervisory signal is used. The experimental evidence shows that the scene text recognition method achieves an F1-score of 74. Combining the two methods, that is - using samples collected by text spotter to train the re-identification model yields a higher F1-score of 85.8. Re-training the person re-identification model with identified inliers yields a slight improvement in performance(F1 score of 87.8). This combination of text recognition and person re-identification can be used in conjunction with video metadata to visualize running events.

KEYWORDS

Runners, Visualization, Person re-identification, Text recognition

1 INTRODUCTION

Running events have gained more popularity in recent times due to increasing awareness regarding health and the accessibility of such events for all people regardless of their gender, age, or economic status.

In some races, the athlete's location data is collected using GPS tracker for live tracking [29]. This data is used by event organizers to publish statistics or records of the race, or sometimes for the participants themselves to track their performance. Several vendors for GPS trackers provide a service for data analysis and running race visualization from the race data [1, 3]. However, athletes usually would like to view images/visualizations of themselves retrieved from the event. Also, GPS trackers are personalized to each athlete while cameras can provide an overview for multiple athletes at once. Videos also provide additional information that can be used for athlete health monitoring.

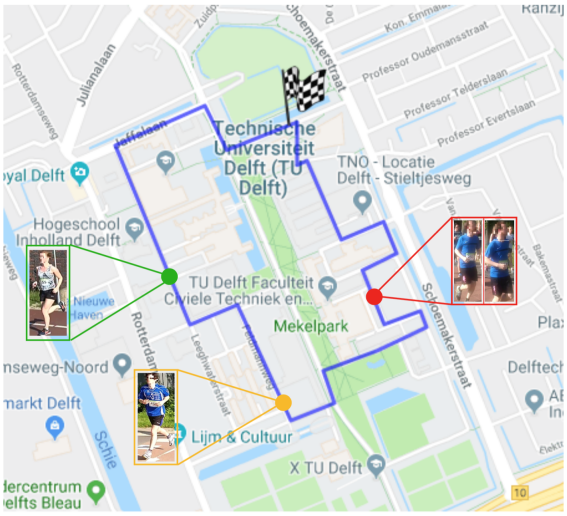


Figure 1: Envisioned result of proposed approach - Shown here is a visualization of a frame of the 2D video with runners represented by markers. The blue line represents the race track where cameras are located at certain points, the flag represents the start and the finish line. The proposed model has the ability to retrieve pictures of runners (at any point of time) given their unique bib number.

While the video streams are time-stamped and also have GPS information on where they are recorded, there is no prior knowledge as to which runner appears in each of the videos. Given that each runner has a unique 'bib number' identifying them, attached to the shirt and with the assumption that any two random people are likely to be dissimilar in appearance, computer vision models can be used for recognizing them. Scene text recognition models can be used to identify the bib number and ultimately the athlete. However, there are potential issues as the bib numbers or the athletes themselves can be fully or partially occluded, as shown in Figure 2. Person re-identification is an approach that utilizes a similarity measure (like Euclidean distance), training a model to retrieve instances of the same person across different cameras [19, 36]. This approach can potentially improve runner identification in scenarios where the bib number is not visible but the runner's features are.

*These authors contributed equally.

(a) (b)

Figure 2: Examples of runners with their bib number tag in image samples. (a) Bib number tags with clear visibility: 405 (green T-shirt), 274 (blue T-shirt), and 8 (white T-shirt). (b) Bib number tag 272 has one of its digits occluded; so it looks like a 27. This along with varying lighting conditions, shadows and variance in pose and camera position makes the task of runner detection non-trivial.

In this work, we use scene text recognition both as a baseline and to collect training samples for a person re-identification method downstream in our pipeline for runner identification. The envisioned result of the model is shown in Figure 1, the race track map with the trajectories of athletes with the ability to retrieve instances of detection of each athlete. The only annotations required are bib numbers that appear in the video to evaluate the scene text recognition model. For the scene text recognition method, we choose deep-learning based text spotter baselines [6, 14]. To evaluate the performance of the proposed approach, we created a ground truth database where the videos are annotated with the bib numbers of appearing runners and the frame interval which they belong.
Our main contributions are:

- A method for visualizing running events using videos recorded at the event. Computer vision based runner detection methods are used with video metadata to find video files and frames where each of the runners appears. The only supervision used in runner identification is the bib numbers known to occur in the video. However, for the whole running event visualization GPS correction and additional filtering steps are used.
- Evaluation of relevant computer vision models for the detection of runners in videos.

2 RELATED WORK

Scene Text Recognition Scene text recognition is the detection, localization and identification of text in images. Recognizing the text in images of scenes can be useful for a wide range of computer vision applications, e.g. robot navigation or industry automation [11, 25]. Traditionally, handcrafted features (e.g. HOG, color, stroke-width, etc.) are utilized to implement a text recognition system [27, 35]. However, because of limited representation ability of handcrafted features, these methods struggle to handle more complex text recognition challenges [22], like the ICDAR 2015 dataset [18]. Deep learning-based methods [6, 14, 21, 24] are more successful in this regard because they can learn features automatically. Lyu et al. [24] modified the Mask R-CNN to have shape-free text recognition ability. Liu et al. [21] and Busta et al. [6] adopted EAST [41] and YOLOv2 [30] as their detection branch and developed CTC-based text recognition branch. He et al. [14] adopted EAST [41] as its text detection branch and developed the attention-based recognition branch. Since in our work, the bib number text appears in unconstrained images of scenes (e.g. dynamic background textures, varying size of text due to distance to camera, illumination variation, etc.) as shown in Figure 2, deep learning based scene text recognition methods would be more well suited to detect the bib numbers because it can perform better with the varying scenarios than traditional (handcrafted features) methods.

Bib Number Detection Using scene text recognition methods is an intuitive choice to detect the bib number and consequently identify the runner who wears it. Ben-Ami et al. [5] proposed a pipeline consisting of face detection and Stroke Width Transform (SWT) to find the location and region of bib number tag, and then applied digit pre-processing and an Optical Character Recognition (OCR) engine [34] to recognize the bib number text. Shivakumara et al. [33] proposed to use torso detection and a Histogram of Oriented Gradient (HOG) based text detector to find the location and region of bib number tag, and then they use text binarization and OCR [34] to recognize the bib number text. Since there is no 'learning' component, the aforementioned models cannot adapt to varying environmental conditions. Existing and publicly available text recognition models [6, 14] are used as our scene text recognition baseline to detect the bib numbers (so as to not reinvent the wheel - focusing on implementing a text recognition system).

Person Re-identification Person re-identification (or re-id) is the task of recognizing an image of a person over a network of video surveillance cameras with possibly non-overlapping fields of view [19]. In previous research, authors [37–39] have investigated classification-based CNN models for person re-id. However, if the dataset lacks training samples per individual, the classification-based model is prone to overfitting [19]. To counter this, the task is cast as a metric learning problem with Siamese networks [9]. Yi et al. [40] uses a Siamese network with pairwise loss that takes a pair of images to train the model and learn the similarity distance

between two images. Hermans et al. [15] showed that using the Siamese network model and triplet loss are a great tool for a person re-identification task because it learns to minimize the distance of a positive pair and maximize the distance of a negative pair at once. Luo et al. [23] improved the previous work (Siamese network model with triplet loss) performance with several training tricks, such as center loss or label smoothing. This work [23], is what we adopt as our person re-id model for runner detection.

3 METHODOLOGY

3.1 Running Event Visualization

This section outlines steps for visualization of the running event.

3.1.1 Running track. To visualize computed trajectories of athletes, a map of the event running track is required. The track is formed by an array of GPS coordinates on an interactive map. The sequence of GPS coordinates are created using *geojson.io*, a web-based tool to create geo-spatial data in GeoJSON file format [2]. It provides an interactive editor to create, view, and share maps [13] similar to Figure 1.

3.1.2 Runner timestamp. Another important element to visualize the athletes trajectory is the timestamp where runners appear at different locations. The individual timestamp is determined by the video timestamp and the frame where the runner appears in the video. From video metadata, we use the date last modified t_V and Duration T_V to get the time the video starts recording. Then we convert the frame f where the runner appears into seconds by dividing it with video frame rate r. To get the individual timestamp, we use linear interpolation, adding the video start time and the converted frame. Individual timestamp is given by:

$$t_R = t_V - T_d + \frac{f}{r}, \tag{1}$$

where t_R is the individual athlete's timestamp, t_V is the video timestamp, f is the frame where the runner appears, r is the frame rate of the video, and T_d is the total duration of the video.
Since runners can appear in multiple frames, we also take into account where the relative position of runners to the camera to determine the best frame to use. Based on our observation, in most videos, the position of the camera and the athletes resembles Figure 3, in which the runners ran towards the camera. So the last frame where a runner appears would be the best choice (closest to the camera), given that the GPS coordinates of the camera will be used to estimate runner location.

3.1.3 Filtering raw GPS data. The Figure 4a shows the trajectory of raw GPS data of videos collected by nine cameras, and it can be seen that the cameras did not stay at one location. Although the videos were taken on the running track, the GPS coordinates seem to stray away. Some of the GPS coordinates deviate too far from the original location; sometimes it is on a highway, a river or a building. The raw GPS data needs to be filtered so that it can be used for visualization.
The first filtering step of raw GPS data is quite simple, i.e. replacing the raw GPS coordinates with the nearest running track points, so the video GPS coordinates stay in the running track. Cosine law is

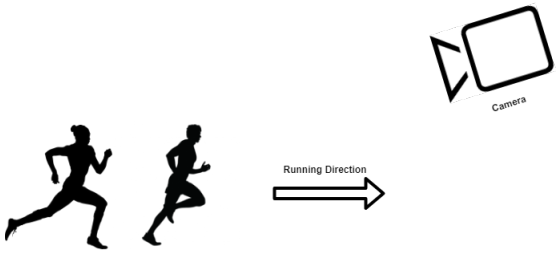
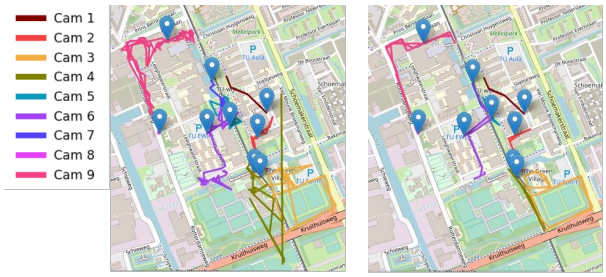


Figure 3: Illustration of how video recording was configured . In most videos, the runners move towards the camera. At first, there is considerable distance between the camera and the runners, but the runners progressively get closer to the camera.

used to filter stray points based on the angle and distance formed by the stray point and two neighbouring ones.
After we apply the filters on raw GPS data, we have a better-looking camera trajectory, as shown in Figure 4b. However, there are still a few stray points from camera 6 that intersects with the trajectory of camera 5 and 7. We manually predict and replace the remaining stray points with new GPS coordinates with estimated points after viewing and analyzing the recorded videos.



(a) **(b)**

Figure 4: Visualization of camera GPS data. (a) Raw camera trajectory. (b) Filtered camera trajectory. There are nine lines with different color representing different cameras. The cameras moved according to the line drawn on the map.

3.1.4 Start and finish location. A piece of information missing from the video data is the individual timestamp and GPS coordinates at the start and finish because there is no video recorded at both locations. So the GPS coordinates and the individual timestamp at the start and finish location are determined by ourselves. The GPS coordinates at the start and finish location are predicted based on the start and finish locations provided by the event organizer. Meanwhile, the individual timestamp of the start location is determined by setting to zero the seconds of the individual timestamp at the location of the first camera. For example, if the individual timestamp at camera 1 is 16:00:10, then the start timestamp is 16:00:00. We assumed that the camera 1 location is only a few meters away (10 - 20 meters) from the start location. The individual timestamp at the finish location is defined by the start time plus the duration required by a runner to finish the race. This duration is obtained from

scraping the Campus Run result website. After the timestamps and the GPS coordinates of the start, finish, and camera locations are obtained, the runner's motion can be interpolated between those locations. Then the javascript library, D3* and Leaflet† are used to create the visualization.

3.2 Runner Detection

The model needs to retrieve the video and frame information where the athletes are detected. We are interested in that information because they are used to visualize the runner's trajectory; the video provides GPS coordinates and video timestamp information, and the frame represents the individual timestamp.

3.2.1 Scene text recognition. The steps to use the text spotter model [6, 14] for our runner detection task are :

- the frames of a video are fed into text spotter model,
- all texts in the image are detected by the text spotter,
- if the bib number we know from scrapping the marathon website is detected, we retrieve the video and frame information for further evaluation.

Text spotter also collects training samples for person re-id which are cropped person images from the video dataset. The cropping occurs around a specified region if the text spotter detects a bib number on the image. Then the bib number is assigned as the label of that cropped image.

Although there are two text spotters, we do not merge the training samples collected by the two text recognition models. Instead, the person re-id model is trained with each training sample separately so as to compare the results of both models.

3.2.2 Person Re-identification. Given the effectiveness of metric learning [15], we choose to adopt the work of Luo et al. [23] which proposes a Siamese network with triplet loss and cross entropy loss.

Since we do not have the ground truth identities of athletes annotated, to evaluate the person re-id model, an object detector is used to detect people in the video dataset and these detections are cropped out. Those cropped images do not have a label because the object detector cannot detect the bib numbers, but the video and information are still stored. The person re-id model is evaluated to see if it can recognize the people detected by the object detector.

It is possible that the object detector results are noisy and might include pedestrians in addition to athletes. So, a *k-NN* is used as an outlier detector, it has competitive performance compared to other outlier detection methods [7].

The idea of using *k-NN* as an outlier detector is that the larger the distance of a query point from its neighbor points, higher the likelihood that it is an outlier [12]. We adopt the definition of an outlier that considers the average of the distance from its k neighbors [4]. If the average distance of an image from its k neighbors is larger than a threshold, then it is considered as an outlier.

3.3 Performance Evaluation

This section describes the mathematical formulation of the evaluation metrics and their implementation in our problems.

3.3.1 Video-wise metric. The video-wise metric is useful to check the number of videos where a runner is detected at least once. Evaluating the performance on the retrieved video information is important because we use the video GPS coordinates to visualize the runners trajectory, so we want to retrieve as many relevant videos as possible. At the same time, it is also undesirable to retrieve irrelevant videos because the irrelevant videos might have GPS coordinates and timestamps that do not agree with GPS coordinates and timestamps that the relevant videos have. Therefore, it could hinder the runner's trajectory visualization.

True positives are when a positive class is correctly predicted to be so, and false positives are when a negative class is detected as positive [28]. In our problem, we collect a ground truth database consisting of the video filename, bib number, and frame interval. So true positive could be defined as correctly retrieved information (video filename, bib number, frame), and the false positive for incorrectly retrieved ones. To evaluate the video-wise metric, the retrieved information we need is only the bib number and its video. We use F1-score for the video-wise metric to evaluate on both relevant and irrelevant videos retrieved by runner detection methods. The video-wise F1-score formula is defined as followed:

$$\text{Recall}_i^v = \frac{|\text{TP}|_i^v}{|\text{GT}|_i^v}\,, \tag{2}$$

$$\text{Precision}_i^v = \frac{|\text{TP}|_i^v}{|\text{TP}|_i^v + |\text{FP}|_i^v}\,, \tag{3}$$

$$\text{F1-score}_i^v = 2\,.\,\frac{\text{Recall}_i^v\,.\,\text{Precision}_i^v}{\text{Recall}_i^v + \text{Precision}_i^v}\,, \tag{4}$$

where i denotes the runner, v denote a video-wise metric, $|GT|$ denotes the number of runner's video in ground truth, $|TP|$ denotes the total true positive (relevant) videos and $|FP|$ denotes the total false positive (irrelevant) video. The final score is defined as the average over M classes, as shown below :

$$\text{F1-score}^v = \frac{1}{M}\sum_{i=1}^{M}\text{F1-score}_i^v\,. \tag{5}$$

3.3.2 Frame-wise metric. Since it is possible that every one of the runners are seen at least once during the event, a naive method that predicts every athlete to be visible in every camera at all times, could achieve a high F1-score. So we need a frame-wise metric. Also, runner detection methods might produce false positives at an irrelevant frame in a relevant video. So, it is necessary to evaluate at a frame level in addition to the video level evaluation.

We use temporal Intersection of Union (IoU) for the frame-wise metric, which measures the relevant interval of frames where the runner appears. The formula of temporal IoU is similar to regional IoU used in object detection [17], except it is only one-dimensional, as shown in Figure 5. The formula for calculating temporal IoU is defined as followed:

$$\text{overlap}_{ij} = \min(E_{ij}^g\,,\,E_{ij}^d) - \max(S_{ij}^g\,,\,S_{ij}^d)\,, \tag{6}$$

$$\text{union}_{ij} = (E_{ij}^g - S_{ij}^g) + (E_{ij}^d - S_{ij}^d) - \text{overlap}_{ij}\,, \tag{7}$$

*https://d3js.org/

†https://leafletjs.com/

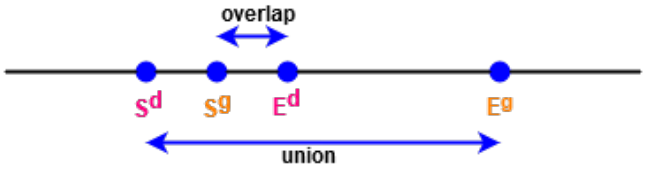


Figure 5: Visualization of frame interval intersection. S denotes the frame start, and E denotes the frame end. And d denotes that the frame interval belongs to detection result and g denotes that the frame interval belongs to the ground truth. The temporal IoU is the ratio between the width of overlap and union.

$$\text{IoU}_{ij} = \begin{cases} 0\,, & \text{if overlap}_{ij} < 0 \\ \frac{\text{overlap}_{ij}}{\text{union}_{ij}}\,, & \text{otherwise} \end{cases} \tag{8}$$

$$\text{IoU}_i = \frac{1}{N_i} \sum_{j=1}^{N_i} \text{IoU}_{ij}\,, \tag{9}$$

where i denotes the runner, j denotes the video file, and N_i is the total of video files where the runner detected. S_{ij}^g and E_{ij}^g are the frame start and the frame end of runner i at video j from ground truth database. S_{ij}^d and E_{ij}^d is the first frame and the last frame which the runner i is detected at video j. The temporal IoU per runner is the average of the temporal IoU over the total of retrieved video. If the detection happens at an irrelevant video j, then IoU_{ij} is zero, so it will lower the IoU_i. The final score is calculated as average over M classes, expressed as :

$$\text{mIoU} = \frac{1}{M} \sum_{i=1}^{M} \text{IoU}_i\,. \tag{10}$$

4 EXPERIMENTS

4.1 Datasets

4.1.1 Ground truth database. Ground truth is necessary for evaluating the performance of the model. It is obtained by manual video analysis, defining a range of frames for each runner where they are recognizable by the human eye. Due to the massive amount of data and time constraints, only runners from the 5 km category are fully annotated. Consequently, only videos with 5 km runners (a total of 127 people) are annotated as ground truth for evaluation. There are a total of 127 runners for the 5km category. 187 videos of these runners were recorded across 9 different locations. The average length of these videos are 35.44 seconds.

The ground truth is established by a range of frames where the runners appear. The range is defined by the "frame start" and "frame end" for every runner in the video. The frame start is annotated when a runner is first recognizable by human-eye. Meanwhile, the frame end is annotated when a runner exits the camera field of view. Figure 6 shows that runner 156 is first seen at the right edge of the screen at frame 98, so this frame is annotated as frame start for runner 156. Then at frame 232, runner 156 is seen for last time at the video, then this frame is annotated as frame end for runner 156. Meanwhile, for runner 155, the frame start is 96, and the frame end is 248.

4.1.2 Evaluation set for person re-id. The person re-id model can not localize and recognize a person simultaneously from a given frame containing multiple objects and the background. It requires cropped images with one person in each image. The training and evaluation set thus must be a collection of cropped images of people, not videos. To create this evaluation set for the person re-id model , images of people from the videos are collected by an object detector and cropped. The difference between the person images collected by the object detector and that collected by text spotters is that the images from the object detector do not have a label. So, the object detector identifies any images with people from a video regardless of whether the person has a bib number or runners with occluded bib number text. The video and frame information of the extracted images of people are also stored for performance evaluation.

There are many object detection methods, such as Faster R-CNN [31] R-FCN [10] and single shot detector [20]. However, experiments comparing them [16] show that Faster R-CNN has better accuracy compared to the other two, so Faster R-CNN is chosen.

4.2 Implementation Details

For text spotter [6, 14] and the object detector Faster R-CNN, available pre-trained models are used. For the person re-id model, we use the same configuration options as the author [23], using the Adam optimizer [32] and the same number of total epochs (120). The output from the object detector (cropped images of people) with the output of scene text recognition as annotation (bib numbers identify runners) is fed into the person re-id model.

Additionally, we collect new training samples to retrain the person re-id model from the images of people obtained using the object detector. We choose the cropped images that are considered as inliers by the previous round of training with the person re-id model.

For *k-NN* model, we choose $k = 5$ and use cosine distance as its distance metric during the inference stage as it is shown to be better than using Euclidean distance for the person re-id task. [23].

The *k-NN* model is trained on the embedding features extracted by the person re-id model. The person re-id model is used only as a feature extractor.

4.3 Performance Analysis

4.3.1 Bib number detection. The model proposed by He et al. [14] is better at detecting the bib numbers than Busta et al. [6]. He et al. [14] has larger recall but lower precision compared to Busta et al. [6], as shown in Table 1. Consequently, although He et al. [14] detects more runners, it produces a larger number of false positives than Busta et al. [6], as shown in Figure 7. Even if a few digits of bib number is occluded, the text spotter [14] will detect it. But because the occluded bib number could look like another number, it will count as false positive. These methods are also compared against naive baselines (Table (1)), that predict all runners to be visible at all times in each video (Baseline (all)) and a random prediction given the number of visible people in the frame (Baseline (random)). Per video, the Baseline(all) has a high recall owing to the fact that all runners are predicted to be visible at all times, this resulting in zero

Figure 6: An example of video annotation. The video is annotated with the runners appearing within a frame interval. The frame interval consists of frame start where the runner start appearing and frame end where the runner is out of frame.

false negatives. However, there are a lot of false positives as not all runners are visible in every video at all times.
The number of detected runners by text spotter [6] from camera 8 is quite low probably because videos recorded by 8 are recorded under harsher illumination, which could interfere with the text spotters [6] performance.

Table 1: The average performance of text spotter on our dataset. He et al. [14] has higher recall, but lower precision compared to Busta et al. [6].

Text Spotter	Recall$^{\upsilon}$	Precision$^{\upsilon}$	F1-score$^{\upsilon}$
Busta et al. [6]	61.03	76.85	66.64
He et al. [14]	86.41	68.41	74.05
Baseline (all)	100	40.52	57.66
Baseline (random)	12.76	14.49	13.57

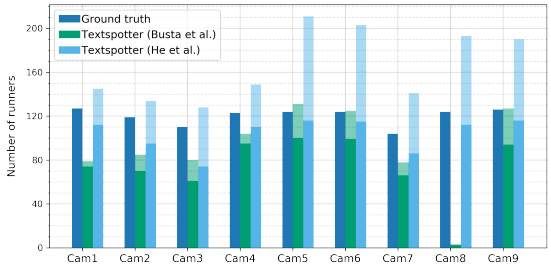


Figure 7: The comparison of the number of runners detected per camera on the event track. The stacked bars with lighter colors are the false positives produced by the text spotter. He et al. [14] has a higher number of true positives and false positives than Busta et al. [6].

4.3.2 Distance threshold for person re-id. A threshold is determined to separate the inliers and outliers for the person re-id methods. We use F1-score$^{\upsilon}$ as a comparison metric between different thresholds. Then we choose a threshold that gives the highest F1-score$^{\upsilon}$. For person re-id with training samples from Busta et al. [6], we choose a threshold of 0.21 that gives an average F1-score$^{\upsilon}$ of 79.00. Also, for person re-id with training samples from He et al. [14], we choose a threshold of 0.22 that gives an average F1-score$^{\upsilon}$ of 85.77.
Since the person re-id with training samples collected by He et al. [14] has higher performance, the detection results from this person re-id model are used to train the model for the second time. For this retrained person re-id model, with 0.05 as the distance threshold an average F1-score$^{\upsilon}$ of 87.76 is obtained.

4.3.3 Comparative results: F1-score$^{\upsilon}$. We also report the F1-score$^{\upsilon}$ between two text recognition models and also the person re-id model trained in three different scenarios. Figure 8 shows that person re-id models generally achieves higher performance compared to the scene text recognition models. It validates the hypothesis that using the whole appearance of a person is better for runner detection as compared to using just the bib number. This could be because of occlusion or blurring of bib numbers (due to athlete motion) hindering the performance of text recognition models. Meanwhile, based on visual inspection of detection results, partial occlusion on runner's body or bib number tag, different runner's pose, or different camera setting (e.g. camera viewpoint or illumination) does not have any significant effect on the performance of the person re-id model. As long as the appearance of runners is distinguishable, the person re-id model can detect them. Person re-id fails when the recording setup is not ideal (e.g. the distance between the camera and runners is too far, the runner facing against the camera), or a considerable part of the runner's body is occluded. Figure 8 also shows a comparison between person re-id models with different training samples. Person re-id with training samples from He et al. [14] has higher performance compared to another with training samples from Busta et al. [6] because He et al. [14] has higher recall so it collects more true positive runners for training. Meanwhile, retrained person re-id only improves a little because most remaining undetected runners are the challenging ones (e.g.

they are recorded farther away from the camera, and not visually distinguishable from one another).
Another important observation is that runners with zero F1-score^{v} at the lower right side of the plot. They are the runners whose bib numbers have less than three digits, which the text spotter struggles to detect. Consequently, the person re-id model does not have training samples and thus it also has a performance of zero for those particular runners.

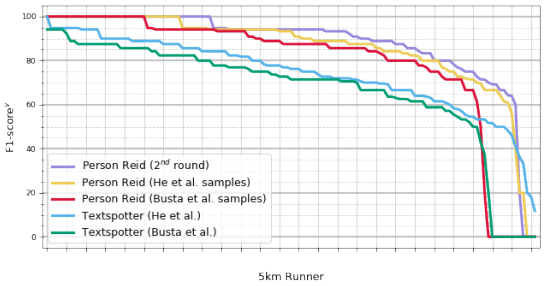


Figure 8: The comparison of F1-scores^{v_i} between person re-identification and text spotter models per runner. The x-axis presents runners sorted on F1-score^{v_i}. The x-axis values are not shown because there are five different axes. Based on F1-score^{v_i}, person re-id models outperform the text spotters.

4.3.4 Comparative results: temporal IoU. To show the performance in retrieving the relevant frames, we present the temporal IoU plot for scene text recognition and three person re-id methods, as shown in Figure 9. The person re-id model still outperforms the scene text recognition ones. It happens because the person re-id uses the training samples from the text spotter, and it can expand the frame interval by detecting runner from earlier frames where the text spotter might find difficulties in detecting the bib number; in the earlier frames, the bib number images are much smaller, but the appearance of the person can be distinct. It is important to notice that some runners have zero temporal IoU. They are the same runners that have zero F1-score^{v}.
The analysis of the comparison between person re-id models with different training samples is more or less similar to the analysis on F1-score^{v}, except it is analyzed on a frame by frame basis. He et al. [14] is better at detecting bib number with a smaller size in the earlier frames; thus person re-id with training samples from this text spotter has a better chance at expanding the frame interval. Meanwhile, the retrained person re-id only improves the average temporal IoU only by 5%. It happens because the runner's images in remaining frames are the hard ones to detect (e.g. the small image at earlier frames due to the distance) and some false positives remain in the second training set.

4.3.5 Outlier detection. In Figure 10, we show the comparison of number of detected runners per camera between He et al. [14] and the corresponding person re-id models. It can be seen that the number of false positives produced by He et al. [14] is quite high. However, the person re-id method can significantly improve producing lesser false positives, although it uses the training samples

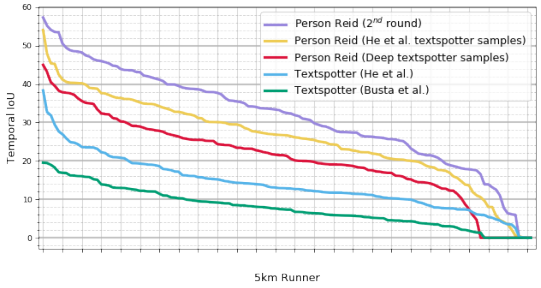


Figure 9: The comparison of temporal IoU between person re-identification and text spotter [14] per runner. The x-axis is runners sorted on its temporal IoU. The x-axis values are not shown because there are five different x-axes. Person re-id models also exceed the performance of text spotters in terms of temporal IoU.

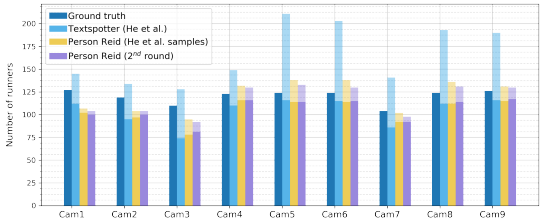


Figure 10: The comparison of the number of detected runners per camera between text spotter [14] and the corresponding person re-id models. The stacked bars with lighter color are false positives. Person re-id models produce less false positives compared to the text spotter [14].

from the text spotter that has many false positives. It validates that *k-NN* in person re-id performs well enough as an outlier detector. Person re-id model with triplet loss minimize the similarity distance between images of the same person and maximize the similarity distance between images of different persons, so the false positives of a runner will have further similarity distance from the training samples of a runner. Then the false positives with larger distances could be rejected.
However, in the second round of training the person re-id model, the reduction in false positives is not significant. This could be because some of sampling more false positives, some of which may be similar in appearance to the runners, and thus are also not rejected as an outlier.

4.3.6 2D timeline visualization. Figure 11 shows the true positives and false positives in our research problem clearly. It can be seen that runners 280, 253, and 456 have all their blue strips aligned with their green strips; using retrained person re-id, they have perfect F1-score^{v} of 100. Meanwhile, using retrained person re-id, runners 336, 150, and 450 still have many false positives. It occurs because of the false positives from text spotter [14] that are used as training

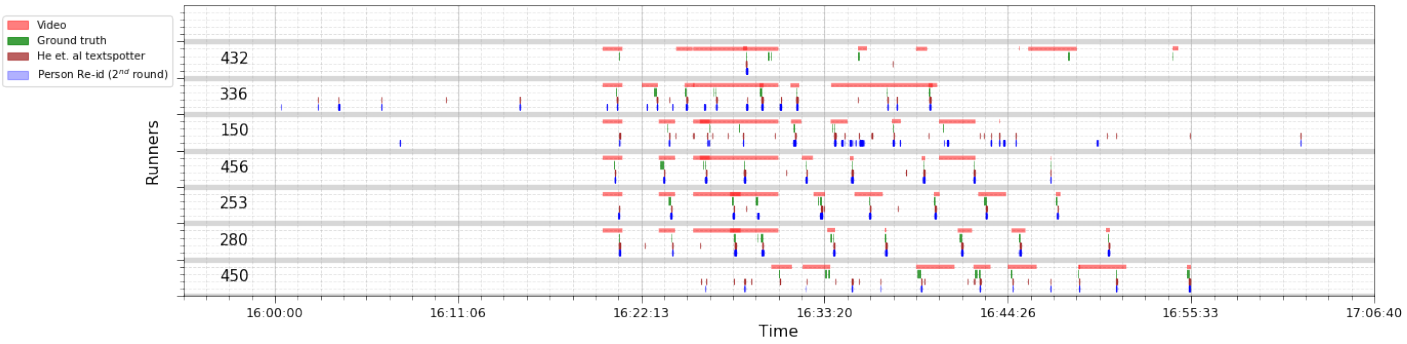


Figure 11: 2D timeline visualization of runner detection. The detection result of a person re-id model retrained for the second time and text spotter [14] are visualized. Not all runners are visualized, three runners most detected are shown (280, 253, 456) and four runners with the lowest (non-zero) detections (432, 450, 336, 150). True positives are the blue strips (person re-id) or brown strips (scene text recognition) that are aligned with green strips (ground truth) and false positives are defined otherwise.

samples; the false positives from blue strips are aligned with the false positive from brown strips.
Another interesting observation is that sometimes the person re-id can detect a runner, although it does not have the training samples at that video. For example, the fourth blue strips at runner 253 do not have aligned brown strips.

5 LIMITATIONS

It is important to note that the performance of the person re-id method depends heavily on the performance of scene text recognition as the latter collects the training samples for the former. For example, person re-id cannot detect the runners that have the bib number with less than three digits. Another thing to note is that there are false positives in the images retrieved by the text spotter. This hinders the performance of person re-id model.

6 CONCLUSIONS AND FUTURE WORK

In this study, we have proposed an automatic approach to create a running event visualization from the video data. We use scene text recognition and person re-identification models to detect the runners and retrieve the videos and frames information where the runners appear so that we can use the individual timestamp and filtered GPS coordinates from the retrieved data for visualization. The experiments show that the scene text recognition models encounter many challenges in runner detection task, which can be mitigated by a person re-id model. The results also show that the performance of the person re-id method outperforms the scene text recognition method. The person re-id method can retrieve the relevant video information almost as good as the ground truth.
This research focused on creating 2D visualizations of athletes with timeline charts and running track visualizations with runners represented by moving markers. This can be further extended to 3D using human pose estimation and spatio-temporal reconstruction [26]. Gait information could also be used additionally for identification. Such reconstruction is not only useful for sports performance analysis and health monitoring, but can also be used for forensic investigations [8]. It would also be interesting to investigate if using another means to collect training samples for person re-id, such as crowdsourcing labeling, can produce a better performance.

7 ACKNOWLEDGEMENT

This study is supported by NWO grant P16-28 project 8 Monitor and prevent thermal injuries in endurance and Paralympic sports of the Perspectief Citius Altius Sanius - Injury-free exercise for everyone program.

REFERENCES

[1] [n. d.]. https://blog.racemap.com/

[2] [n. d.]. GeoJSON. https://geojson.org/

[3] [n. d.]. GPS tracking of outdoor sports events. https://tech4race.com/wordpress/en/home/

[4] Fabrizio Angiulli and Clara Pizzuti. 2002. Fast Outlier Detection in High Dimensional Spaces. In *Proceedings of the 6th European Conference on Principles of Data Mining and Knowledge Discovery (PKDD '02)*. 15–26.

[5] Idan Ben-Ami, Tali Basha, and Shai Avidan. 2012. *Racing Bib Number Recognition*. https://doi.org/10.5244/C.26.19

[6] Michal Busta, Lukas Neumann, and Jiri Matas. 2017. Deep TextSpotter: An End-to-End Trainable Scene Text Localization and Recognition Framework. *2017 IEEE International Conference on Computer Vision (ICCV)* (2017), 2223–2231.

[7] Guilherme O. Campos, Arthur Zimek, Jörg Sander, Ricardo J. Campello, Barbora Micenková, Erich Schubert, Ira Assent, and Michael E. Houle. 2016. On the Evaluation of Unsupervised Outlier Detection: Measures, Datasets, and an Empirical Study. *Data Min. Knowl. Discov.* 30, 4 (July 2016), 891–927.

[8] Jia Chen, Junwei Liang, Han Lu, Shoou-I Yu, and Alexander Hauptmann. 2016. Videos from the 2013 Boston Marathon: An Event Reconstruction Dataset for Synchronization and Localization. (2016).

[9] S. Chopra, R. Hadsell, and Y. LeCun. 2005. Learning a similarity metric discriminatively, with application to face verification. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, Vol. 1. 539–546 vol. 1. https://doi.org/10.1109/CVPR.2005.202

[10] Jifeng Dai, Yi Li, Kaiming He, and Jian Sun. 2016. R-FCN: Object Detection via Region-based Fully Convolutional Networks. *CoRR* abs/1605.06409 (2016).

[11] G. N. Desouza and A. C. Kak. 2002. Vision for mobile robot navigation: a survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24, 2 (2002), 237–267.

[12] S. Garcia, J. Derrac, J. Cano, and F. Herrera. 2012. Prototype Selection for Nearest Neighbor Classification: Taxonomy and Empirical Study. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 34, 3 (March 2012), 417–435. https://doi.org/10.1109/TPAMI.2011.142

[13] Bailey A. Hanson and Christopher J. Seeger. 2016. Creating Geospatial Data with geojson.i.

[14] Tong He, Zhi Tian, Weilin Huang, Chunhua Shen, Yu Qiao, and Changming Sun. 2018. An end-to-end TextSpotter with Explicit Alignment and Attention. *CoRR* abs/1803.03474 (2018).

[15] Alexander Hermans, Lucas Beyer, and Bastian Leibe. 2017. In Defense of the Triplet Loss for Person Re-Identification. *CoRR* abs/1703.07737 (2017).

[16] Jonathan Huang, Vivek Rathod, Chen Sun, Menglong Zhu, Anoop Korattikara, Alireza Fathi, Ian Fischer, Zbigniew Wojna, Yang Song, Sergio Guadarrama, and Kevin Murphy. 2016. Speed/accuracy trade-offs for modern convolutional object detectors. *CoRR* abs/1611.10012 (2016).

[17] Paul Jaccard. 1912. The Distribution of Flora in the Alpine Zone. *New Phytologist* 11 (02 1912), 37 – 50. https://doi.org/10.1111/j.1469-8137.1912.tb05611.x

[18] D. Karatzas, L. Gomez-Bigorda, A. Nicolaou, S. Ghosh, A. Bagdanov, M. Iwamura, J. Matas, L. Neumann, V. R. Chandrasekhar, S. Lu, F. Shafait, S. Uchida, and E. Valveny. 2015. ICDAR 2015 competition on Robust Reading. In *2015 13th International Conference on Document Analysis and Recognition (ICDAR)*. 1156–1160.

[19] Bahram Lavi, Mehdi Fatan Serj, and Ihsan Ullah. 2018. Survey on Deep Learning Techniques for Person Re-Identification Task. *CoRR* abs/1807.05284 (2018). arXiv:1807.05284 http://arxiv.org/abs/1807.05284

[20] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott E. Reed, Cheng-Yang Fu, and Alexander C. Berg. 2015. SSD: Single Shot MultiBox Detector. *CoRR* abs/1512.02325 (2015).

[21] Xuebo Liu, Ding Liang, Shi Yan, Dagui Chen, Yu Qiao, and Junjie Yan. 2018. FOTS: Fast Oriented Text Spotting with a Unified Network. *CoRR* abs/1801.01671 (2018). arXiv:1801.01671 http://arxiv.org/abs/1801.01671

[22] Shangbang Long, Xin He, and Cong Yao. 2018. Scene Text Detection and Recognition: The Deep Learning Era. *CoRR* abs/1811.04256 (2018).

[23] Hao Luo, Youzhi Gu, Xingyu Liao, Shenqi Lai, and Wei Jiang. 2019. Bag of Tricks and A Strong Baseline for Deep Person Re-identification. *CoRR* abs/1903.07071 (2019). arXiv:1903.07071 http://arxiv.org/abs/1903.07071

[24] Pengyuan Lyu, Minghui Liao, Cong Yao, Wenhao Wu, and Xiang Bai. 2018. Mask TextSpotter: An End-to-End Trainable Neural Network for Spotting Text with Arbitrary Shapes. *CoRR* abs/1807.02242 (2018). arXiv:1807.02242 http://arxiv.org/abs/1807.02242

[25] M. M.aftabchowdhury and Kaushik Deb. 2013. Extracting and Segmenting Container Name from Container Images. *International Journal of Computer Applications* 74, 19 (2013), 18âĂŞ22. https://doi.org/10.5120/13001-0039

[26] Armin Mustafa, Hansung Kim, Jean-Yves Guillemaut, and Adrian Hilton. 2016. Temporally coherent 4d reconstruction of complex dynamic scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 4660–4669.

[27] Lukas Neumann and Jiri Matas. 2013. On Combining Multiple Segmentations in Scene Text Recognition. *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, 523–527. https://doi.org/10.1109/ICDAR.2013.110

[28] David Louis. Olson and Dursun Delen. 2008. *Advanced data mining techniques*. Springer.

[29] Hanya Pielichaty. 2017. *Events project management*. Routledge.

[30] Joseph Redmon and Ali Farhadi. 2017. YOLO9000: Better, Faster, Stronger. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2017), 6517–6525.

[31] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster R-CNN: Towards Real-time Object Detection with Region Proposal Networks. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1 (NIPS'15)*. 91–99.

[32] Sebastian Ruder. 2016. An overview of gradient descent optimization algorithms. *CoRR* abs/1609.04747 (2016).

[33] Palaiahnakote Shivakumara, R. Raghavendra, Longfei Qin, Kiran B. Raja, Tong Lu, and Umapada Pal. 2017. A New Multi-modal Approach to Bib Number/Text Detection and Recognition in Marathon Images. *Pattern Recogn.* 61, C (Jan. 2017), 479–491.

[34] R. Smith. 2007. An Overview of the Tesseract OCR Engine. In *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)*, Vol. 2. 629–633. https://doi.org/10.1109/ICDAR.2007.4376991

[35] Kai Wang, Boris Babenko, and Serge Belongie. 2011. End-to-end Scene Text Recognition. In *Proceedings of the 2011 International Conference on Computer Vision (ICCV '11)*. 1457–1464.

[36] Di Wu, Si-Jia Zheng, Xiao-Ping Zhang, Chang-An Yuan, Fei Cheng, Yang Zhao, Yong-Jun Lin, Zhong-Qiu Zhao, Yong-Li Jiang, and De-Shuang Huang. 2019. Deep learning-based methods for person re-identification: A comprehensive review. *Neurocomputing* 337 (2019), 354 – 371. https://doi.org/10.1016/j.neucom.2019.01.079

[37] Lin Wu, Chunhua Shen, and Anton van den Hengel. 2016. Deep Linear Discriminant Analysis on Fisher Networks: A Hybrid Architecture for Person Re-identification. *CoRR* abs/1606.01595 (2016).

[38] Shangxuan Wu, Ying-Cong Chen, Xiang Li, Ancong Wu, Jinjie You, and Wei-Shi Zheng. 2016. An Enhanced Deep Feature Representation for Person Re-identification. *CoRR* abs/1604.07807 (2016).

[39] Tong Xiao, Hongsheng Li, Wanli Ouyang, and Xiaogang Wang. 2016. Learning Deep Feature Representations with Domain Guided Dropout for Person Re-identification. *CoRR* abs/1604.07528 (2016).

[40] D. Yi, Z. Lei, S. Liao, and S. Z. Li. 2014. Deep Metric Learning for Person Re-identification. In *2014 22nd International Conference on Pattern Recognition*. 34–39. https://doi.org/10.1109/ICPR.2014.16

[41] Xinyu Zhou, Cong Yao, He Wen, Yuzhi Wang, Shuchang Zhou, Weiran He, and Jiajun Liang. 2017. EAST: An Efficient and Accurate Scene Text Detector. *CoRR* abs/1704.03155 (2017).