POLITECNICO MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE E DELL'INFORMAZIONE

Executive Summary of the Thesis

Detection of life-threatening situations by Visual Motion Analysis

Laurea Magistrale in Computer Science and Engineering - Ingegneria Informatica

Author: Marco Costanzo

Advisor: Prof. Vincenzo Caglioti

Academic year: 2023-24

1. Introduction

The extensive use of cameras in public areas and the development of Computer Vision techniques have been crucial for the deployment of systems able to automatically detect life-threatening situations in real-time. This work analyzes some of those situations and proposes a series of methodologies for the detection of such life-treating events. A vision-based heuristic methodology for the detection of a fall is developed. This methodology can be beneficial for the population, especially for elderly people, allowing minimization of the damage and ensuring fast and effective medical assistance. The introduced fall detection framework has been adapted to recognize the action of climbing over turnstiles. A study on the trajectory and a procedure for the recognition of passive behaviours during a jump has been developed. Identifying such behaviours allows us to recognize potentially dangerous situations which occur when a person does not oppose any resistance to the inertial motion, thus resulting in a potentially harmful impact on the ground. Finally, a methodology to identify in real-time potentially life-threatening situations in railways has also been developed. This methodology allows a fast response during those events, even in scenarios in which few people are close to the scene.

2. Fall detection

The proposed heuristic approach for fall detection is composed of the following stages: preprocessing, person detection, person tracking, extraction of the 3d landmarks and fall detection. The person detection stage, whose output is a collection of windows, each one containing a single person in the scene, allows the usage of techniques for single-person pose estimation. Furthermore, the usage of a landmark model composed of 33 different landmarks allows one to recognize a wide variety of movements. The developed fall detection criterion is based on the fact that, when a fall occurs, the angle between the vector crossing the center of the head and the center of the hips of a person, i.e. the central vector, and the ground plane is small [1]. Furthermore, the center of gravity reaches a descending speed that is greater than a given threshold value, because the gravity acceleration makes the person accelerate downward [2]. If a person stands up within a short time after a fall, the alarm is not triggered. The developed procedure is shown in Figure 1. Several types of falls have been successfully recognized using the presented procedure, as shown in Table 1.

3. Detection of climbing over turnstiles in subways

The introduced fall detection framework has been adapted to recognize the action of climbing over the turnstile. In this case, the proposed heuristic is based on the variation of the ascending speed, which should be greater than a given threshold value, since the person must accelerate upward to win the gravity acceleration during the jump. Furthermore, the distances between the center of the hips of a person and the left and right heels should be lower than a threshold, since when a person is climbing over a turnstile, their legs get closer to their hips. This action should happen in a short time after a strong upward movement has been detected and the ascending speed is close to zero, i.e. when the peak of the jump has been reached. The procedure for the detection of climbing over turnstiles is shown in Figure 2. The introduced approach has been tested using footage taken from news reports and datasets such as CAUCAFall and KFall, containing activities of daily living, to test its robustness.

4. Trajectory estimation and analysis

4.1. Trajectory tracking

The introduced fall detection framework has been adapted also to track the trajectory followed by a person during a jump. To do so, it is necessary to define criteria to determine when the jump starts and when it ends. One possible criterion to determine when a jump starts is to check whether the ascending speed is greater than a given threshold value. Similarly, checking that the speed of the center of gravity of a person is lower than a given threshold value allows one to determine when a jump ends. This criterion is justified by the fact that, when a jump ends, the shock caused by hitting the ground causes a strong deceleration. The temporal succession of the 3d positions of the center of gravity of the person, during the two previously identified instants, represents the trajectory.

4.2. Trajectory estimation

Since to estimate a parabola in 2D, the vertex point is required, criteria to identify it among a set of temporally ordered COG points should be defined. One robust criterion to do so is to look for the first point such that its y coordinate value is lower than the value of the y coordinate of the previous point belonging to the trajectory. Using all the points that belong to the trajectory from the moment the vertex was detected, it is possible to estimate the trajectory. Many methodologies can be adopted, the one used is the least squares polynomial fit for a second-order polynomial.

4.3. Detection deviation from the reference trajectory criteria

Using the above-mentioned trajectory tracking and estimation procedures, it is possible to define criteria to detect if a person is following the expected (parabolic) trajectory during a jump or not. To detect if a person is following or not the estimated parabolic trajectory it is possible to compute the estimated trajectory using all the COG points that belong to the trajectory, minus the current one. After that, the distance between the y coordinates of the detected COG point and the expected COG point, i.e. the one lying on the estimated trajectory, is computed. If this distance is less than a reference distance, the person is considered to follow a parabolic trajectory without opposing any resistance to the inertial motion, thus showing a, potentially dangerous, passive behaviour. In case a person does follow the estimated trajectory, recomputed for all the following points, for a time that is greater than a given threshold, an alarm is raised. The above-described procedure is shown in Figure 3.

5. Detection of dangerous situations in railways

The proposed detection framework for dangerous situations in railways is composed of the following stages: preprocessing, projective transformation, dangerous areas definition, background estimation, background subtraction, connected components detections and alarm triggering criteria. During the projective transformation stage, a projective mapping is computed. Such mapping, that is applied to the camera frames, allows the usage of a different perspective, i.e. a point of view from above, to make the task of monitoring dangerous areas easier. During the next stage, the areas that are considered dangerous are defined.
The identified areas are three:

- the area in the platform beyond the yellow line
- the area above the track that is closest to the platform
- the area above the other tracks.

After that, the background is estimated using a Gaussian Mixture-based Background/Foreground Segmentation Algorithm [3] and subtracted from the current frame, to highlight the changes in the scene. The connected components are extracted from the previously computed difference frame. The area of connected components can be used as a discrimination criterion to differentiate objects, such as litter, from people, birds and trains.
Using this consideration, the following algorithms have been developed:

- train detection algorithm (Algorithm 1)
- people crossing the yellow line detection algorithm
- people crossing the track detection algorithm (Algorithm 2).

Using the previously defined algorithms, the proximity warning algorithm is defined. This algorithm detects when a person is crossing the yellow line and there is no train close to the platform. Furthermore, an algorithm for the detection of people on the track is also defined (Algorithm 3). The proposed approach has been tested using some footage taken from news reports. The types of detections that were analyzed are the following: train detection, proximity warning and alarm for the presence of people on the track.

6. Equations, Figures, Tables and Algorithms

6.1. Figures

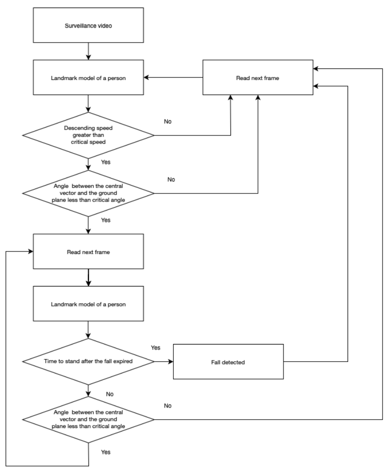

Figure 1: Fall detection workflow

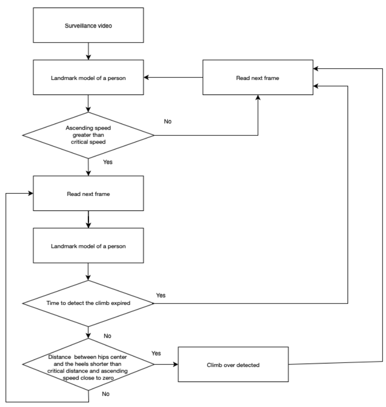

Figure 2: Climb over turnstiles detection workflow

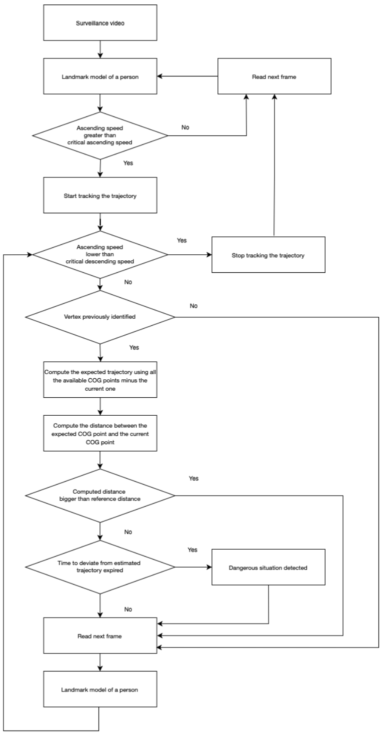

Figure 3: Detection passive behaviour during a jump workflow

6.2. Tables

Table 1: Type of falls detected

Type of fall
Forward fall when trying to sit down
Backward fall when trying to sit down
Lateral fall when trying to sit down
Forward fall when trying to get up
Lateral fall when trying to get up
Lateral fall while sitting, caused by fainting
Vertical(forward) fall while walking caused by fainting
Fall while walking, use of hands to dampen fall, caused by fainting
Forward fall while walking caused by a trip
Forward fall while jogging caused by a trip
Forward fall while walking caused by a slip
Lateral fall while walking caused by a slip
Backward fall while walking caused by a slip

6.3. Algorithms

Algorithm 1
Train detection algorithm

Input
I: current frame,
B: estimated background,
P: projective mapping,
D: train detection threshold,
A: area above the track

$DiffF \leftarrow$ I - B
$TranfDiffFrame \leftarrow ProjTr$(DiffF, P)
$croppedF \leftarrow cropF$(TranfDiffFrame,A)
[Areas,$Idexes$] $\leftarrow CompConnC$(croppedF)
$MaxArea \leftarrow$ 0
for CurrA, $CurrentInd \in [Areas, Idexes]$ **do**
  if $CurrA \geq$ MaxArea **then**
    $MaxArea \leftarrow$ CurrA
  end if
end for
if $MaxArea \geq$ D **then**
  True
else
  False
end if

Algorithm 2
People crossing the track detection algorithm

Input
I: current frame,
B: estimated background,
P: projective mapping,
D: people detection threshold,
A: area above the track

$DiffF \leftarrow$ I - B
$TranfDiffFrame \leftarrow ProjTr$(DiffF, P)
$croppedF \leftarrow cropF$(TranfDiffFrame,A)
[Areas,$Idexes$] $\leftarrow CompConnC$(croppedF)
for CurrA, $CurrentInd \in [Areas, Idexes]$ **do**
  if $CurrA \geq$ D **then**
    True
  end if
end for
False

Algorithm 3
Alarm people on the track algorithm

Input
I: current frame,
B: estimated background,
P: projective mapping,
Tt: train detection threshold,
Tp: people detection threshold,
At: area above the track

1: $PDT \leftarrow PeopCrossTrackAlg$(I,B,P,Tp,At)
2: $TD \leftarrow TrainDetectionAlg$(I,B,P,Tt,At)
3: **if** PDT **and** *not* TD **then**
4: True
5: **else**
6: False
7: **end if**

7. Conclusions

In this work, a heuristic vision-based approach for fall detection has been presented. One of the advantages of the proposed approach for fall detection, w.r.t. to other approaches in the academic literature is the fact that, since it recognizes the essence of the action, it doesn't need to learn to recognize a fall using long training sets. This results in improved reliability and reduces sensibly the risk of not recognizing a type of fall not present in the training dataset. The framework used for fall detection has been adapted to recognize the action of climbing over the turnstile. Also, in this case, the developed heuristic has been tested against footage of real-world situations. A methodology for trajectory tracking and estimation has been developed and tested. Furthermore, a heuristic to identify passive behaviours during a jump has been introduced. The results of this study can be used to improve the accuracy of the developed fall detection system and discriminate a (dangerous) passive fall from a reactive one. The presented framework can be easily extended to recognize different actions w.r.t. to the presented ones by developing a suitable heuristic. Finally, a methodology for the detection of dangerous situations in railways has been presented. The presented systems can be easily extended to work in real-time situations due to their low computational cost. The fact that all the presented systems rely on a single RGB camera has the disadvantage of potentially missing capturing some action dynamics due to occlusion. Future works could extend the presented approaches using multiple cameras, with the disadvantage of the additional overheads and costs.

8. Acknowledgements

First, I would like to thank you Professor Caglioti for giving me the opportunity to write this thesis. I would also like to thank my parents and my brother for their support.

References

[1] Djamila Romaissa Beddiar, Mourad Oussalah, and Brahim Nini. Fall detection using body geometry and human pose estimation in video sequences. *J. Vis. Comun. Image Represent.*, 82(C), jan 2022.

[2] Weiming Chen, Zijie Jiang, Hailin Guo, and Xiaoyang Ni. Fall detection based on key points of human-skeleton using openpose. *Symmetry*, 12, 2020.

[3] Zoran Zivkovic. Improved adaptive gaussian mixture model for background subtraction. *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.*, 2:28–31 Vol.2, 2004.