Automatic Analysis of Pedestrian's Body Language in the Interaction with Autonomous Vehicles

Walter Morales-Álvarez[1], María José Gómez-Silva[2], Gerardo Fernández-López[1], Fernando García-Fernández[2] *Member, IEEE* and Cristina Olaverri-Monreal[3] *Member, IEEE*

***Abstract*—The work presented on this paper, aims to provide an automated tool to analyze the autonomous vehicle-pedestrian interaction. It detects pedestrians in the environment, by means of the most modern visual detection technology available in the literature, applied to an stereo camera system, and analyses if the driver looks at the vehicle. To achieve this task the algorithm does pose estimation, feature matching, and facial detection to acquire the position of the pedestrians and distinguish if they notice the vehicle. The algorithm calculates the 3D coordinates of a given pedestrian using the vehicle as a reference, tracking the movement of the pedestrian during all the process, thus providing meta-information of this interaction, which allows to process this information at higher levels: e.g. if the pedestrians feels conformable to cross, or if he performs any other maneuver moving away from the trajectory. The proposed algorithm was tested in campus scenarios where pedestrians and vehicle shared the environment, and the results proved the viability, providing a tool, useful for researchers which allows to process high amounts of information without the need of supervision.**

***Index Terms*—autonomous vehicles, human-computer interaction, pedestrian identification, behavior analysis**

I. Introduction

The advances in the driving paradigm derived from automated driving inevitably leads to changes in the interaction between pedestrians and vehicles. To do so, a high number of works focus in the analysis of the interaction of the vehicle and the pedestrians.

Autonomous vehicles are, essentially, robotic systems controlled by computers that can emulate the human driving and take relevant decisions in case of given circumstances. Humans drivers must be aware of the environment around them in order to avoid accidents and always be attentive to unexpected events and situations in which other road users are involved.

Autonomous driving technology has extremely improved in the last years. However, there is not a comprehensive study about the pedestrian's reaction and behavior regarding an approaching vehicle. This paper proposes an algorithm that can automatically categorize several factors that indicate the readiness of pedestrians to cross a marked crosswalk when an autonomous, driver-less vehicle is approaching. These factors will make possible to establish protocols that contain information to determine whether it is safe to cross the road for pedestrians. The information will base on the location of the individuals and the fact that some identified the vehicle as autonomous.

Many approaches to detect and track pedestrians use feature classifiers [1], [2], [3]. They calculate an approximate region of interest where it is possible that a person is located in an image. Afterwards, those detections are tracked using filters like Kalman-filter [4], [5] or cellular automata and backpropagation neuronal networks [6] that estimate the position of the pedestrian in the next frame.

Other works use representations of objects using generative forms models, a discriminative texture classifier and a Bayesian framework based on particle filtering to detect and track pedestrians [7].

In [8] a three-dimensional LADAR was used to detect and identify pedestrians, by classifying a subset of the points obtained from the laser using pattern recognition techniques based on geometric and motion features.

Although all these approaches can track a person among a video recorded, they don't secure that the detection estimated in the next frame, corresponds to the person in the previous one. This is crucial due to the importance of maintaining the analysis of a person over time.

In this paper we propose an algorithm to detect and track pedestrians in order to be able to categorize their movements for a latter analysis or their intention of crossing the road. To this end data was recorded by using a stereo-camera located in the vehicle. Further details of the methodology used are presented in the next section.

The algorithm designed and presented in this paper was tested on the iCab's driving recordings, a vehicle automated by the Intelligent System Lab's team (LSI) from the Carlos III University of Madrid (see Fig. 1), designed to act as in campus autonomous transport service, and presented in [9], [10].

II. Method

In order to explain in detail the algorithm designed, it is divided into the fundamental modules that compose it, as can be seen in Fig. 2.

A. Pedestrian Detection by Pose Estimation

The pedestrian detection was performed using the Open Pose library, published by CMU-Perceptual-Computing Lab

[1] Mechatronics Research Group, Simón Bolívar University, Caracas, Venezuela
{12-11153 gfernandez}@usb.ve
[2] Intelligent Systems Lab (LSI) Research Group, Carlos III University of Madrid, Spain.
{magomezs fegarcia}@ing.uc3m.es
[3] University of Applied Sciences Technikum Wien, Department of Information Engineering and Security, Vienna, Austria. olaverri@technikum-wien.at

 DOI: 10.1109/IVS.2018.8500425

Fig. 1: Picture of the iCab2.

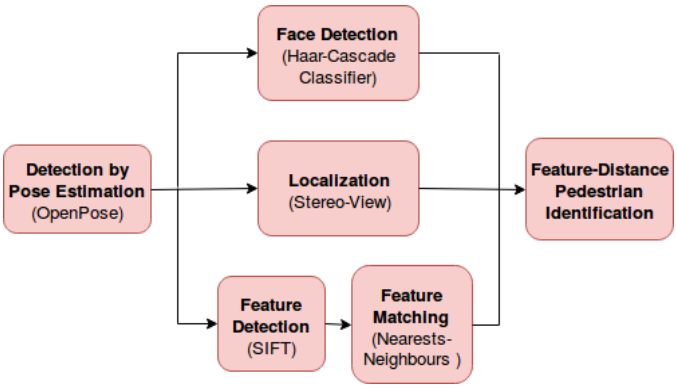


Fig. 2: Block diagram of the algorithm.

[11], [12], which designed a convolutional, feedback-based neural network that was responsible for locating the keypoints of an individual's pose. It calculates the heatmaps in which it considers the keypoints of the pose are located and the neural network uses a feature representation that preserves the location and orientation information of pedestrians limbs, called part affinity fields.

By using the library it is possible to obtain up to 18 keypoints of the pedestrians that made possible the detection of persons and their body parts (e.g. legs, arms, etc.) (see Fig. 3) which consequently allows the extraction of a ROI to analyze. This region of interest is defined as a square in the image with upper-left corner in the point (x_0, y_0), height h and width w.

$$x_0 = min(x_{k_i}) \quad i \in [0, 17] \tag{1}$$

$$y_0 = min(y_{k_i}) \quad i \in [0, 17] \tag{2}$$

$$w = max(x_{k_i}) - min(x_{k_i}) \quad i \in [0, 17] \tag{3}$$

$$h = max(y_{k_i}) - min(y_{k_i}) \quad i \in [0, 17] \tag{4}$$

These detections are a fundamental part of the algorithm because they are the main feature that the program will use to identify pedestrians throughout its execution.

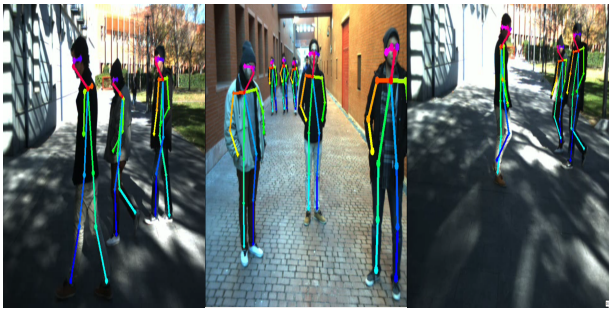


Fig. 3: Pedestrian detection by pose estimation using OpenPose.

Fig. 4: Disparity obtained of a given scenario.

B. 3D Coordinates and Distance using Stereo-Camera

The relative position in 3D coordinates was possible due to the stereo camera presented in the autonomous vehicle. This is based on two cameras aligned in such a way that the human vision is replicated, where from two different images of the same scenario, information regarding depth can be extracted. This is achieved because the depth of a given point in space is inversely proportional to the disparity in space, which is the difference between the views of both images. This disparity was calculated in a separate process by the algorithm in [13] from the viewpoint of the left camera.

In order to simplify the program, the algorithm on the stand-alone vehicle corrected the images so that both recorded planes were parallel. In turn, the camera was already calibrated so that the focal length parameters f, the distance between the lenses B, x and y fundamental pixels (c'_x, c'_y) were available. This

allowed the attainment of the right and left images of a given scenario.

Having exposed the above and being $d(x, y)$ the disparity at a point x, y in the image, then the 3D coordinates of that point are obtained as:

$$x_{3D} = \frac{(x - c'_x)B}{d(x, y)} \tag{5}$$

$$y_{3D} = \frac{(y - c'_y)B}{d(x, y)} \tag{6}$$

$$z_{3D} = \frac{Bf}{d(x, y)} \tag{7}$$

The orientation axes convention that the algorithm uses is presented in the Fig. 5.

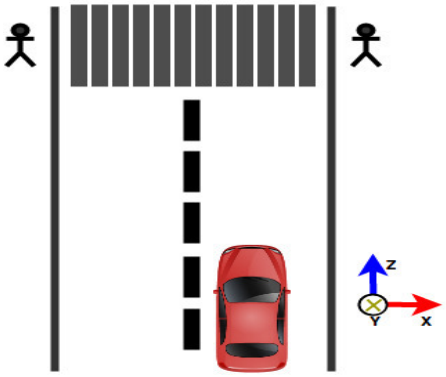


Fig. 5: Established axes of the given scenario.

C. Feature Detection and Matching

Fig. 6: Features detection and matching between two images of the same person in different frames.

In order to obtain a degree of "similarity" between two images, features, points of interest in the objects, are obtained and then it is calculated which of these coincide to make sure that the same person is tracked. A high number of matches indicates that a large part of one image is in the other. In order to obtain the features of the images, detection is used, which provides a description and keypoints of the features that it considers acceptable, to later save them in a database. These features and descriptors are then compared with those of the second image so that it is possible to determine which of these are in both.

To detect features, the scale-invariant feature detection (SIFT) algorithm introduced in [14], is used. It provides features that are invariant to changes in illumination, orientation, rotation and scale. This algorithm uses a representation of the image to determine features by applying the maximum and minimum at Difference of Gaussian, filtering the edges and low contrasts; and assigning the dominant orientations to them.

To make the match between features, the Brute Force matcher and the closest neighbors (knn) method is used. With the latter one, the Euclidean distance between the descriptors is compared in order to find out what the corresponding characteristics are.

By making this process between old and new detections, it is possible to determine which correspond to the same person so that pedestrian identification can be established.

D. Face Detection

In order to determine whether a pedestrian has visualized the stand-alone vehicle, a face detection module was implemented using Haar-likes features, as is done in [15] [16], which is based on the determination of the contrast change between rectangular groups of adjacent pixels. The Haar-like features are the composition of several of these groups with a variance of relative contrasts. These take advantage of several facts that occur in the face, such as the intensity of the nose and septum is usually higher than its laterals or that the eye area is usually darker than the cheekbones.

On the other hand, in order to reduce the computational cost of detection, there is Cascade based classifier, presented in [17], with which a level detection is carried out on the different regions of the image, in which a high value must be obtained with a determined Haar-like feature in each one to reach the next level. Fault regions where a Haar fault is eliminated. This classifier, then, makes it possible to identify the face of pedestrians.

In turn, OpenCV allows users to use HaarCascade classifier trained to detect frontal faces, a convenient event to determine if a pedestrian has visualized the vehicle.

E. Feature-Distance Pedestrian Identification

In order to do an optimal identification between the detections in previous and current frame, two arrays are created $(detections_{current}, detections_{old})$ to store the virtual representations of the identified detections on the previous frames and the detections in the current one. The members of the arrays are defined as a group of variables that the computer uses to interpret a pedestrian. They are an ID, used to identify the person during the process; the pose keypoints, given to estimate the position of the pedestrian's articulations on a given image; the ROI where the pedestrian is located in the image; the 3D real coordinates of the person; the direction and a flag that indicates if a person has seen the autonomous

vehicle. Each one of them conditions the comparison done in order to determine the correspondence between the pedestrians on the two arrays.

The "current" detections array is filled with persons that are considered suitable as pedestrians in order to reduce errors caused by bad resolution of distant detections. This made possible the extraction of their depth information, used to improve the response of the algorithm. To do so, a range of depth distance is set to establish which detections are acceptable. This is laid down to be $depth_i \in [3, 15]m$ due to be the range in which the detections have a suitable resolution to make an analysis of the pedestrian's behavior.

Initially, the first detected pedestrians are saved in the array of old detections due to the fact that this is empty at the beginning of the program so, for the frame where the firs t accepted detections occur $detections^i_{old} = detections^i_{current} \quad i \in [0, n]$. Each detection has an ID which increases each time an unidentified admissible detection is generated.

From the second frame with detections onwards, a pedestrian-to-pedestrian comparison is made between the members of $detections_{current}$ and $detections_{old}$. In order to avoid errors and decrease the comparisons, a criteria is established considering that a person can not move more than 1 m in x between two consecutive frames.

$$||detections^i_{current}.3D.x - detections^j_{old}.3D.x)|| \leq 1m \quad (8)$$

With regards to the parameter to be compared, that the detections of the current frames correspond to the previous detections that are closest and are as 'similar ' as possible. Within the scope of the first term mentioned above, the Euclidean distance between pedestrians in the current and previous frames is calculated.

As mentioned above, the degree of similarity will correspond to the number of matched features between a current and a previous detection calculated by the algorithm.

In order to obtain the most optimal identification of current detection, the Hungarian algorithm [18] is applied, which calculates the least expensive combination of column and row members of a cost matrix. For the purposes of this paper, pedestrians of the current and previous frame are established as columns and rows respectively. In this way, each cell of the cost matrix corresponds to the ratio between the distance referenced above and the number of matched features between the previous and current detections.

$$a_{ij} = -\frac{\#MF_{previous_i, current_j}}{||previous_i(x, y, z) - current_j(x, y, z)||} \quad (9)$$

In this way, a possible identification of the detections of the current frame is obtained. However, to avoid false identifications, it is established that $a_{ij} <= -5$. If the previous condition is met, the corresponding ID is assigned to the current detection in question and the variables that define the pedestrian in the arrangement of old detections are updated. On the other hand, new IDs are assigned to current detections that have not had a match in the processed cost matrix when $cols > rows$ because they correspond to new pedestrians. The case that $cols < rows$ implies that there are old detections that do not correspond to the current ones so they proceed to decrease a parameter called lives, which establishes the number of consecutive frames that an old pedestrian is allowed to not be detected.

The criteria established to determine if a pedestrian has observed the autonomous vehicle is the detection of its face since, as mentioned above, the Haar-cascade based classifier used for detection is trained with a dataset of the frontal view of several faces. In this way, a true value is assigned to the visualization flag if the pedestrian's face is detected.

Finally, a .csv file is filled with the variables of 3D position and flag of visualization of the vehicle to establish the proportion of pedestrians signalized to be aware of the proximity of an autonomous vehicle.

III. Algorithm Evaluation

In order to test the effectiveness of the proposed algorithm, five tests were designed in which pedestrians were exposed to a driverless vehicle in the proximity of a marked crosswalk. In the first and second tests, three pedestrians crossed simultaneously through the crosswalk from right to left and vice versa respectively.

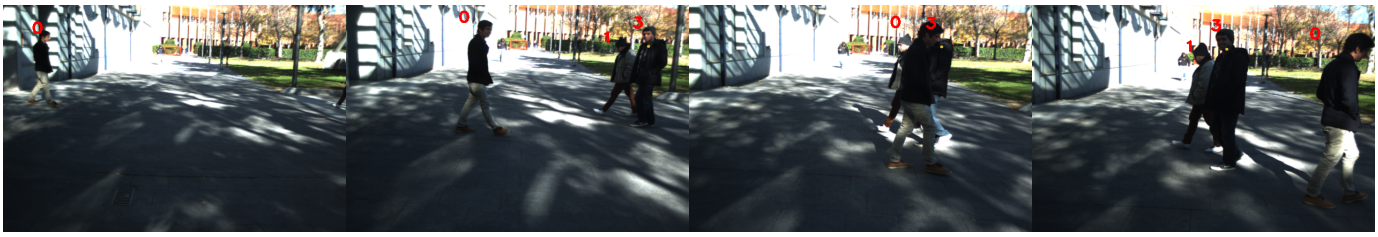
The third test was based on the scenario in which the pedestrians crossed from both sides of the street as an autonomous vehicle approached, a task that is complicated due to the fact that at a certain point people will cross each other generating occlusion problems.

In the first designed scenarios, the autonomous vehicle stops to allow pedestrians to move freely. In this way, two tests were carried out where the vehicle did not stop in order to see the algorithm's response in these circumstances. In the first of these, people approach the crosswalk but do not cross it because they wait for the automaton to pass through. The last designed scenario, people walked parallel to the vehicle and in the same direction.

IV. Results

The behavior of the above tests is shown in the Fig. 7 where it is observed that the ID of the pedestrians are maintained along the frames of the iCab driving video, corresponding to the red numbers that appear on the left of each pedestrian.

In the first two tests, the program identified pedestrians with the advantage that, in the course of the video, the distance between them increased, thus generating a 'cheaper' parameter on the cost matrix for the same person in different frames and a more expensive for comparisons between different individuals. On the other hand, in the first test, in 7% of the frames with acceptable detections, at least one pedestrian that has been identified previously, has not been identified for that image. However, in all cases, it recovers the correct ID in a maximum of five frames. In the second scenario, the value increases to 12%. These errors are not a problem for program performance because the computer re-identifies pedestrians,



(a) Pedestrians crossing from both sides.

(b) Pedestrians crossing from the left both side.

(c) Pedestrians crossing from the right both side.

Fig. 7: Results of the algorithm obtained from three scenarios.

this being an important issue because it also solves the problem of occlusions.

In the third test it can be seen from Fig 7a that the program loses people who are occluded by the closest pedestrian, however, thanks to the permission granted to the detections of the old frames to remain in the array $detention_{old}$ for 10 frames with valid pedestrians, the machine is able to re-identify the' lost' individuals. In this case, the percentage of loss of identifications, not counting the occlusions, is 8% value that is considered acceptable for the subsequent analysis that can be carried out with the data obtained with the algorithm.

Identification on the fourth scenario is easier because the problem is similar to that of the detection and identification of stopped pedestrians. For these scene, the percentage of ID loss does not exceed 5%.

The results described above are presented in Table 1.

test	*detection rate*	*ID loss*	*ID recovery*
1	96%	7%	yes
2	94%	12%	yes
3	97%	8%	yes
4	91%	3%	yes
5	90%	6%	yes

Table 1. Performance of the algorithm.

V. Conclusion and Future Work

This paper presents an approach that analyzes human behavior in a crossing scenario in which an autonomous vehicle approaches. The implemented algorithm allows the detection and tracking of multiple individuals that are found in the surroundings, by automatic means, in order to estimate their position and know if the individuals are aware of the vehicle, which is very helpful to researchers who are still investigating the impression of people in front of vehicles that do not have a human pilot, since these studies are performed manually, generating cost in time and money.

The use of position-based detection in conjunction with the stereo-camera allows the location of the persons in the three dimensions to obtain the distance between the autonomous vehicle and the individuals.

The process previously described could be tested in different scenarios in which pedestrians interact with autonomous vehicles, as it provides a significantly acceptable performance, and automatizes the process of classification of behavioral patterns.

Although the robust algorithm designed allowed pedestrians to be tracked through data recorded by autonomous vehicles, there is a possibility to improve its efficiency based on the absolute position of pedestrians. In the current algorithm, a relative position of pedestrians in relation to the autonomous

vehicle is obtained, which made it difficult to use distance to identify detections, which was already distorted due to the fact that the vehicle was in motion and generated a variation in the location, on the Z axis, of the individuals. However, it is possible to calculate the position of the pedestrians with respect to the starting point of the video, using the instantaneous speed of the vehicle. To this end, the distance traveled by the car, calculated from the aforementioned parameter, is added to the pedestrian Z coordinate.

On the other hand, the algorithm can be extended to automatically analyze pedestrian's movements, determine their behavioral patterns and make a later prediction of their actions (i. e. whether the pedestrian will cross the street or not) using the estimated poses and predictive models like Markov's decision process, Markov's models or Bayesian networks.

In addition a protocol of interaction between the computer and the human can be established, in which the first, can indicate to the pedestrian if it is safe to cross the street, based on distance between them and if the individual is visualizing the vehicle.

REFERENCES

[1] A. Mohan, C. Papageorgiou, and T. Poggio, "Example-based object detection in images by components," *IEEE transactions on pattern analysis and machine intelligence*, vol. 23, no. 4, pp. 349–361, 2001.

[2] C. Papageorgiou and T. Poggio, "A trainable system for object detection," *International Journal of Computer Vision*, vol. 38, no. 1, pp. 15–33, 2000.

[3] L. Zhao and C. E. Thorpe, "Stereo-and neural network-based pedestrian detection," *IEEE Transactions on Intelligent Transportation Systems*, vol. 1, no. 3, pp. 148–154, 2000.

[4] M. Meuter, U. Iurgel, S.-B. Park, and A. Kummert, "The unscented kalman filter for pedestrian tracking from a moving host," in *Intelligent Vehicles Symposium, 2008 IEEE*. IEEE, 2008, pp. 37–42.

[5] M. Bertozzi, A. Broggi, A. Fascioli, A. Tibaldi, R. Chapuis, and F. Chausse, "Pedestrian localization and tracking system with kalman filtering," in *Intelligent Vehicles Symposium, 2004 IEEE*. IEEE, 2004, pp. 584–589.

[6] C. Suppitaksakul, "Pedestrian detection and tracking," November 2006. [Online]. Available: http://nrl.northumbria.ac.uk/488/

[7] S. Munder, C. Schnorr, and D. M. Gavrila, *IEEE Transactions on Intelligent Transportation Systems*, no. 2, pp. 333–343, June.

[8] L. E. Navarro-Serment, C. Mertz, and M. Hebert, "Pedestrian detection and tracking using three-dimensional ladar data," *The International Journal of Robotics Research*, vol. 29, no. 12, pp. 1516–1528, 2010. [Online]. Available: https://doi.org/10.1177/0278364910370216

[9] A. Hussein, P. Marin-Plaza, D. Martin, A. de la Escalera, and J. M. Armingol, "Autonomous off-road navigation using stereo-vision and laser-rangefinder fusion for outdoor obstacles detection," *IEEE Intelligent Vehicles Symposium (IV)*, pp. 104–109, 2016.

[10] P. Marin-Plaza, J. Beltran, A. Hussein, B. Musleh, D. Martin, A. de la Escalera, and J. M. Armingol, "Stereo vision-based local occupancy grid map for autonomous navigation in ros," *Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP)*, vol. 3, pp. 703–708, 2016.

[11] Z. Cao, T. Simon, S. Wei, and Y. Sheikh, "Realtime multi-person 2d pose estimation using part affinity fields," *CoRR*, vol. abs/1611.08050, 2016. [Online]. Available: http://arxiv.org/abs/1611.08050

[12] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh, "Convolutional pose machines," in *CVPR*, 2016.

[13] J. Beltrán, C. Jaraquemada, B. Musleh, A. de la Escalera, and J. M. Armingol, "Dense semantic stereo labelling architecture for in-campus navigation." in *VISIGRAPP (5: VISAPP)*, 2017, pp. 266–273.

[14] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.

[15] A. Allamehzadeh and C. Olaverri-Monreal, "Automatic and manual driving paradigms: Cost-efficient mobile application for the assessment of driver inattentiveness and detection of road conditions," in *2016 IEEE Intelligent Vehicles Symposium (IV)*, June 2016, pp. 26–31.

[16] A. Allamehzadeh, J. U. de la Parra, A. Hussein, F. Garcia, and C. Olaverri-Monreal, "Cost-efficient driver state and road conditions monitoring system for conditional automation," in *2017 IEEE Intelligent Vehicles Symposium (IV)*, June 2017, pp. 1497–1502.

[17] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, vol. 1. IEEE, 2001, pp. I–I.

[18] H. W. Kuhn, "The hungarian method for the assignment problem," *Naval Research Logistics (NRL)*, vol. 2, no. 1-2, pp. 83–97, 1955.