

Keywords: infrared sensors; linearization algorithm; collision detection; blind data algorithms; tracking objects; Intel R200

Piotr SZABLATA*, Paweł ŁAKOWSKI, Janusz POCHMARA

Poznan University of Technology, Institute of Automatic Control and Robotic
Piotrowo 3A, Poznań 60-965, Poland

*Corresponding author. E-mail: Piotr.Szablata@gmail.com

MOVING OBJECT DETECTION IN CAR TRAFFIC WITH IMPLEMENTATION OF OPTICAL SENSORS

Summary. In this investigation, the problem of moving object detection - without any knowledge - is classified. It describes a technique that will allow real-time localization with usage of IR sensors. The proposed algorithm is simplistic, and in the future, it might be implemented into any vehicle, premium or entry level. It is guided by AI that must calculate its next moves in the blink of an eye without user noticing any delays. The main problem of moving object recognition was extraction of proper features, description of the events, and choice of only the crucial ones. The presented novel approach does not follow any standard algorithms. It is a practical hardware implementation of custom solution, based on processing system, which can be well situated in the safety modules of future cars.

1. INTRODUCTION

Road safety might take advantage of additional expenditures on route planning process to avoid sudden and unexpected traffic collisions. In this paper, the authors dwell on automotive accidents; hence, it is worth to quote a few relevant statistics. In Poland, in recent years, the number of road accidents has decreased slightly, but in the end, the number of collisions has actually increased, with both reaching high values. In 2015, there were 362 265 collisions and 32 967 incidences, resulting in 2938 deaths [1].

In European Union, a similar trend was observed. In 2013, there were 1054744 accidents that led to 25938 deaths [2]. When it comes to worldwide general values, it is almost impossible to gather statistics as precise as the ones available for EU or Poland. Still, the WHO estimates that approximately 1.25 million people died on the roads during the past year. These numbers picture how important it is to ensure high level of safety on the roads, how irresponsible and insufficiently capable to avoid road accidents are humans, and how serious is the need of driving assistance implementation or even self-driving AI usage.

To this end, a lot of studies and experiments were carried out. Their purpose was to solve problems of collisions, through protection of vehicle movements and by reacting appropriately for uncertain environment and interference in the movement. To react adequately to a situation, it was crucial to acquire necessary information about vehicle's surroundings in real time. In most cases, the researchers focused on gathering information from a camera mounted on the moving vehicle, but those presented insufficient data for image processing. Numerous methods were used, such as methods based on image matching and frame coupling [3], methods based on background subtraction approach, like spatial-temporal nonparametric background subtraction [4], spatiotemporal Gaussian mixture model [5], or estimation affine transformation parameters using least median of squares [6]. Different approaches were presented in many papers [7 - 10], where authors used an optical flow or additional units like direction and speed-tuned operators [7], and artificial optical flow block estimated with means of a

homography or Gaussian mixture model [10]. In general, it can be stated that in all of those papers, the authors focused on object detection. Only two of them [7, 8] made one step beyond and estimated the velocities of detected objects, which is crucial for making responsible decisions by driver assistance system. What is maybe even more important, they did not prove that this method can provide the system necessary data in real time yet.

In this paper, the authors would like to present a solution that could be practically applied in hardware where the reaction time is crucial to make the difference for human life. In the presented investigation, problem of moving object detection without any knowledge is classified. Therefore, proposed is the new technique that will allow moving object detection in real time. Detection of moving objects acquired by moving camera requires standard pre-processing steps such as motion compensation, moving object detection, and tracking. In the previous papers [11, 12], a quadratic motion model was used to compute car motion on flat road surfaces. Outliers were detected using robust multi-resolution techniques. Such solution was elegant but required expensive computing power and did not account for vehicle vibration on uneven surfaces.

The fundamental problem in moving object recognition is the extraction of good features to describe the actions. In this work, focus is on motion features (trajectories). Motion trajectories are informative, compact, and spatiotemporally continuous, which makes them useful for action recognition [13, 14]. A novel approach that does not follow the standard steps and accordingly avoids the aforementioned difficulties is presented in this paper. It is possible to reconstruct the shape of an object hidden from the spectator. Real-time algorithm will detect multiple moving objects in urban traffic. Our technique does not require any knowledge about traffic patterns. In the future investigations, possibility of estimating motion in determining object movement will be explored. This could be used for locating moving objects and describing probable velocities of those objects. Hardware implementation of proposed solution based on the processing system which can be well situated in the car modules is also presented in this paper.

2. ESTABLISHING RESEARCH AREA

In March 2016, National Highway Traffic Safety Administration (NHTSA) and the Insurance Institute for Highway Safety (IIHS) announced that the manufacturers of 99% of U.S. automobiles had to agree for inclusion of automatic emergency braking systems as a standard feature on virtually all new cars sold in the U.S. by 2022. NHTSA project assumes that the accelerated rollout of automatic emergency braking would prevent an estimated 28,000 collisions and 12,000 injuries [15].

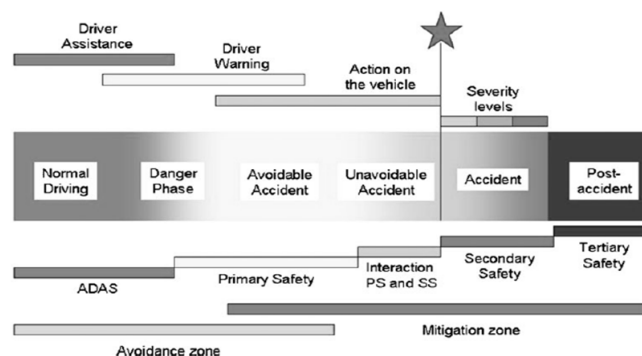


Fig. 1. Safety model actions, model represents idea of 6 phases of accidents

From the driver's point of view, there is never enough time for reaction. General safety model proposes integrated safety model presented in Fig. 1 [16]. Some of the actions are relying on automatic steering system, and some actions are still dependent on the driver. The evaluation of the effectiveness is possible for calculation of pre-collisions systems. For example, problem of collisions can be divided

into two areas: first is dedicated to normal configuration of collision, and second is dedicated to special collision; both are illustrated in Fig. 2 [16].



Fig. 2. Normal configuration of collision and special configuration of collision

3. SYSTEM ARCHITECTURE

The presented investigation puts focus on early collision detection. Pre-collision system (PCS) is based on two forms: active safety and passive safety systems. Active safety system operates on signals and information gathered. Typically, either of those alerts the driver of a dangerous situation or assist in important maneuvers like steering while braking. However, passive safety system offers support only when necessary.

Those solutions are proven but not sufficient in era of intelligent vehicles. Therefore, methods and systems which improve and contribute ascendable the degree of safety driving are needed. With this information, PCS unit can determine another car's position, distance, speed, and relative velocity almost immediately. If any sudden changes in those factors could potentially cause a collision, system can provide information or assist the driver in avoiding potential accident. In this investigation, system based on such architecture is proposed, which is also presented in the Fig. 3.

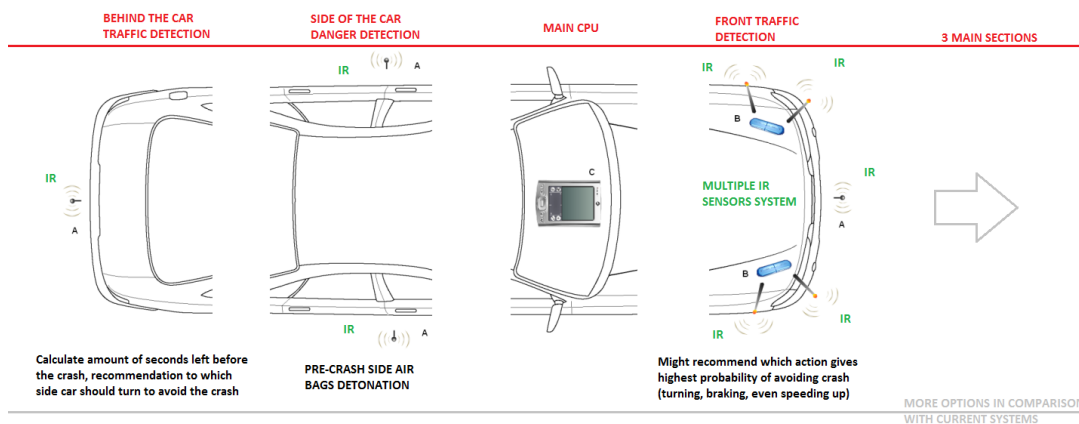


Fig. 3. Tracking system of move objects in car traffic where A is a long-distance radar, B is RealSense detector and C is a controller

4. VEHICLES' EVENT SIMLUATOR

Main aim is to create a consistent model of road danger level, especially with the participation of pedestrians, taking into account numerous statistics allowing to conduct realistic simulations. This simulation can be crucial for future research on accidents with pedestrians, for testing solutions designed to avoid them. All statistics used to develop appropriate model come from studies [17 - 21]. In model design, it was decided to make few simplifications; hence, analysis relates only to the road situations when pedestrians cross the street and does not include traffic light activity. Still that is enough to cover 92.1% of accidents with pedestrians. What is more, assumption was made, based on statistics shown in *2013 Report of vehicle speeds in Poland* [19] and day-to-day observations, that

average car speed in the city, when excluded traffic congestions, is about 45 km/h and an average daily distance covered by car in built-up area, with this average speed, is estimated at 10 kilometers.

There are approximately 23 million vehicles in use in Poland, and there are 8000 of accidents with pedestrians per year. This lets us state that there is one accident every $8.4 \cdot 10^{-8}$ seconds; hence, in simulation, each iteration represents one second, and probability of accident is equal to $1.2 \cdot 10^{-9}$. All those parameters were assumed on basis of parameters that are well described in police statistics, but there is one more important parameter to be determined: number of road crossings made by pedestrians. Owing to lack of elaborations covering this topic and impossibility of gathering statistics, it was decided to collect data by the authors themselves. Day-to-day observations showed average number of pedestrians crossing the road in front of a car moving with the average speed assumed above, estimated at 10 per day. Pedestrians crossing with traffic lights or during traffic congestions were excluded. With this assumption, road crossing will happen on average every 80 seconds.

Based on those statistics, two types of road crossing events were implemented: first, when pedestrian cross the road in a safe distance to a car, and second, when crossing takes place closer than braking distance [21]. Those probabilities were achieved with basic C++ pseudo-random number generator. Moreover, following assumptions were made: road average width equals to 5.0m and car average length and width equals to 4.5 m and 1.85 m, respectively. Pedestrian population was divided, based on studies [18] into four groups: children, youth, adults, and elders. Adequate speeds of road crossing had been assigned to all of the groups (Tab. 1).

Table 1

Average speed during the road crossing related to age group

	Children	Youth	Adults	Elders
Average speed [m/s]	1.315	1.427	1.389	1.168
Population share [%]	16	12	45	27

For best simulation of reality, detailed statistics were used; hence, simulation takes into account the different number of pedestrians during 24-hour period of time (greater number of pedestrians between 6 am and 10 pm); prevalence of accidents with pedestrians in industrial areas in relation to accidents in undeveloped areas, which affects the number of cars parked at the edge of the road, which decrease visibility; and also different weather conditions, which increase braking distance significantly [21].

To facilitate development and unit testing, it was decided to make simple 2D visualization with usage of OpenGL. In the pictures presented within this paper, it can be seen consecutively, safe road crossing, dangerous road crossing, and collisions with pedestrian. In the first situation, a pedestrian crosses the road in front of a vehicle, but at considerable distance from it. In the second one, the system will classify situation as dangerous owing to the small distance between pedestrian and car and their likely collision trajectories. The third picture (Fig. 4) presents moment of impact, which is observed as a pedestrian is in the car's outline. System based on pseudo-code presented in the Listing 1 is proposed.

Fig. 4 presents visualization for simulation of collision probability. The bottom rectangle represents car moving on the road, rectangles on the right represent cars parked along the road, and small white squares stand for pedestrians.

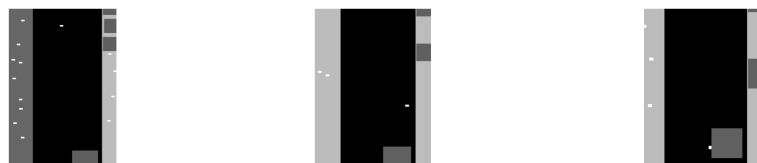


Fig. 4. Sim. visualization: (a) safe situation, (b) dangerous situation, (c) collision

```

initialize objects
randomize positions of objects
while time < STOP_TIME:
    calculate new positions and velocities
    if positions > SIMULATION_RANGE:
        randomize positions and velocities of objects
        asses safeness
    if situation not safe:
        collect data
        increase time

```

Listing 1. Random generator for collision algorithm

5. CALCULATE COLLISIONS

Procedure for collision in time is based on the calculation proposed in many video games. It is simple, and calculation of it in an embedded system will be immediate. There is not enough time (in vehicles) for special complex algorithms with similar purpose.

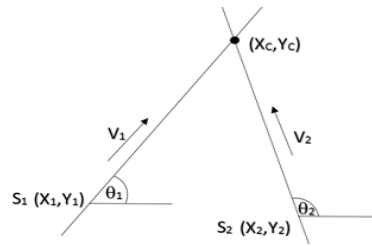


Fig. 5. Diagram for calculation of the intersection point

The main concept of the algorithm and final expression needed for calculation is that for each vehicle, the following is known: initial x position, which represents speed in x direction, and initial y position, which represents speed in y direction. For two points $S1$ and $S2$, there are following variables defined: $X1$ - initial x position, Xat - speed in x direction, $Y1$ - initial y position, Yat - speed in y direction, $X2$ - initial x position, Xbt - speed in x direction, $Y2$ - initial y position, and Ybt - speed in y direction.

The distance between two points is given by Pythagoras theorem. By expressing this distance as a function of time (based on the given variables presented above), it is possible to solve the minimum possible distance. Outcome will be given in the form of a formula, which will indicate the time of minimum distance. Detailed calculations are presented as follows.

$$M_1 = X_1 \cdot X_{at} - X_2 \cdot X_{at} - (X_1 - X_2) \cdot X_{bt} + Y_1 \cdot Y_{at} - Y_2 \cdot Y_{at} - (Y_1 - Y_2) \cdot Y_{bt} \quad , \quad (1)$$

$$M_2 = X_{at}^2 - 2 \cdot X_{at} \cdot X_{bt} + X_{bt}^2 + Y_{at}^2 - 2 \cdot Y_{at} \cdot Y_{bt} + Y_{bt}^2 \quad . \quad (2)$$

Optimal time calculated from the formula:

$$timemi = -\frac{M_1}{M_2} \quad . \quad (3)$$

This is a straightforward expression using only the given variables. The resulting value $timemi$ is the time in seconds (units that speeds are expressed in) when the two points will be the closest. This time is relative to the time of the starting positions; if this time is negative, then the potential collision would have been in the past. In other words, there is no pending collision in the future.

If the calculated $timemi$ is positive, then the time of minimum distance is in the future. Next step then is to determine just how close that minimum distance will be occurring. The distance at any time t is given by the following equation (4).

$$dis = \sqrt{(t \cdot X_{at} - t \cdot X_{bt} + X_1 - X_2)^2 + (t \cdot Y_{at} - t \cdot Y_{bt} + Y_1 - Y_2)^2} \quad . \quad (4)$$

Once again, a fairly simple expression is presented. It depends on the given variables in addition to t . To calculate minimum distance, it is required to evaluate it by using the minimum time calculated previously. Then calculated t when $t=timemi$.

$$dismi = \sqrt{(timemi \cdot X_{at} - timemi \cdot X_{bt} + X_1 - X_2)^2 + (timemi \cdot Y_{at} - timemi \cdot Y_{bt} + Y_1 - Y_2)^2} \quad . \quad (5)$$

Final formula results in value of minimum distance for evaluation of minimum time left before the collision ($dismi$).

6. CAMERA AND SENSOR CAPABILITIES

IR-based depth sensors are commonly used in robotics for short-range object detection. Those components are useful for avoiding bumps and damages for small-scale solutions but are lacking possibility to be used with ranges above 500mm [22]. With a much bigger size of problem, where in place of robot, 1500kg car is placed, such maximum limit is unacceptable. With a large mass and volume, as is the case with all vehicles, amount of injuries recorded by bumping into pedestrians is massive. By DEKRA report, only in Germany, there were 206,696 incidents over the period of one year [23]. This is why, a solution to avoid those incidents is as important as pursuit for inventing sensors that will be able to detect objects in front of car speeding through the built-up areas.

First challenge for this goal appeared while searching for the best depth sensor on the market. It meant finding a device that will offer most promising functions and set of features. Large selections of hardware parts were tested, and even popular solutions, like both generations of Microsoft Kinect sensors. In the end, the best sensor was supplied by Intel as part of SDK training for RealSense project. Selected developers currently have access to R200 sensor with high-resolution IR technology and refresh rate of 90 samples per second. A 90-Hz mode was implemented in the last firmware update from 2016. Intel RealSense sensors are constantly improving with newer firmware and software functions; this ensures project would be able to evolve within time.

Experiment was conducted to confirm hardware measurement precision. Special construction was made for this purpose, with vertical shelves placed in 14 dedicated places kept in one line. Camera could see only one of those at each moment. Sensor was placed at one end to keep all of the shelves just in front of it. Distance between those positions was kept at 90 mm. Such distance was used to simulate road driven by the car for each sample (1/90 s) at 30 km/h speed limit, which was recommended by European Transport Safety Council in its report from 2014 [24]. Additionally, proven precision increasing algorithm was used to produce the best results [25].

Middle point of the sensor data matrix was exactly the middle spot of the first shelf, and for each round of tests, different shelf was placed into this dedicated slot. Results were taken 1 second after each shelf changed to stabilize result. Measurements were made 15 minutes after turning on the R200 device; optic parts were warmed and in optimal temperature. This experiment was to simulate objects appearing before the car's front bumper at different ranges.

It was proven that camera works very precisely at distance between 900 and 2000 mm from the shelf, with some good results for bigger objects (>20 cm wide) with maximum detection range up to 3000 mm and person tracking up to 5500 mm. Results were gathered in Tab. 2.

Results were smoothed by eliminating all readings that were showing value of zero. It was about 10% of all the data sent by the sensor. Even after such operation, still some stability issues were observed within the results. By comparing the size of error per segment for different samples, it was clear that range of fluctuation was between 0.01% and 0.54%. There was still room for improvement, but even with such unfiltered results, solution achieved average error at level of 0.19%, which translated into distance of 3.952 mm, if calculated for last Sample N. Such distance was well within minimum limit of distance that must be detected over the period of 1/90 s, which was estimated at

level of 90 mm. It is exactly 4.39% of this value, which means that presented solution achieves massive margin of error, which is a must when designing hardware concerned with safety. Proven 3.952-mm distance of error also meant that system will work in full effectiveness (real results for all samples within a second) at speed as low as 2km/h. This experiment proves that R200 sensor will be sufficient for traffic analysis.

Table 2

Test in range between 900mm and 2000mm, highest precision expected							
	Sample A	Sample B	Sample C	Sample D	Sample E	Sample F	Sample G
Distance measured mech.	910mm	1000mm	1090mm	1180mm	1270mm	1360mm	1450mm
Distance measured by IR sensor	909.8mm	1001.6mm	1091.6mm	1180.8mm	1263.2mm	1360.2mm	1448.6mm
Error (percentage)	0.02%	0.16%	0.15%	0.07%	0.54%	0.01%	0.10%
	Sample H	Sample I	Sample J	Sample K	Sample L	Sample M	Sample N
Distance measured mech.	1540mm	1630mm	1720mm	1810mm	1900mm	1990mm	2080mm
Distance measured by IR sensor	1538.8mm	1636.6mm	1725.0mm	1807.6mm	1903.3mm	1997.0mm	2076.4mm
Error (percentage)	0.08%	0.40%	0.29%	0.13%	0.17%	0.35%	0.17%

The experiment shows capability of solution for one single point. It was also important to prove that all 270 thousand of samples refreshing at frequency of 90 Hz (3 times more frequent in comparison with Microsoft Kinect [26]) are as precise as the middle point. Another test was conducted, where exactly the same construction was used as before, but this time, horizontal and vertical positions were changed to verify if sensor's field of view was wide enough to observe objects reaching as far as 2m to the sides. This meant that there had to be 4m of operating range for this sensor. Such width would simulate automobile size with additional meter for each side of the car to observe what was happening around the vehicle. With FOV at 70 degrees on x and y axis [27], it was possible to achieve such result at 2m distance from the sensor. The sensor kept precision of distance measurement even on edges of this area, and there was no notable difference in error margin percentage for this test.

Intel RealSense R200 integrates one depth sensor and two high-resolution cameras, which are working at maximum of 60 frames per second and can be used to supply additional data from the hardware. Both streams inform about 2 million of pixel values in full RGB range, which makes it possible to implement an additional algorithm that changes frequency of depth sensor sampling depending on color contrast readings from the photographs. For darker periods of time (night), depth sensor would be working like night vision, with just barely noticeable help from camera data. For bright days, depth sensor would be able to share workload with RGB streams and collect massive amounts of data from every source. It also offers 33 ms exposure time, which translates into bright and high ISO image just after 1 frame spent for setting up. Second frame is already adjusted for proper focus and light conditions. Algorithm needed to use contrast data from RGB matrix was already available and ready for implementation [25].

For multiple sensor devices, it was very important to have constant live access to all the data, from each source, to make sure detection was possible as soon as possible, not delayed by data transfer or frame drops. In case of this solution, all three data matrixes were constantly updated, and USB 3.0 transfer standard was there to make sure no data was lost and no additional lag was forced. The only disadvantage of such solution is a must to keep constant 900 mA of power current at the end of data cable, which means active USB extenders are used after each meter of cable.

One noticeable limitation of R200 sensor is its minimum range of operation. It starts detection from 900mm of distance between device and the object. There is no possibility to force it for data capture at area closer than almost one meter. To expand viewing range and cover closer objects, a second camera/sensor must be used – Intel F300, which is a dedicated variant of RealSense project for short-range detection. A strong feature of Intel’s software is simultaneous access up to 3 different devices at the same time. Such multi-connection would not slow down any of the data streams.

7. SAFETY SYSTEM – PROPOSED SOLUTIONS

Sunila Gollapudi presents in her work main areas of investigation in form of tree chart explaining connections in machine learning semantics [28]. The presented project also focuses on technology branch and classification, searching for mechanism and building intelligence into a self-learning machine, implying that it will be able to do better in the future from its own experience.

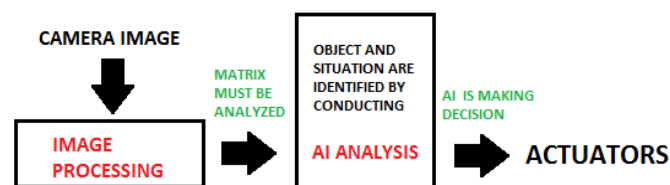


Fig. 6. Safety model configuration

Fig. 6 presents proposal of a safety model in the practical form which can be implemented into vehicle. This part of investigation includes the study of how machines perceive the environment, learn to discriminate behavior of interest from the rest, and are able to take reasonable decisions about categorizing each behavior. Next set focuses on implementation of a general-purpose algorithm that solves practical problem. It calculates data sets manually, where the input data and the expected output data are available. The second dataset is where it has all the input data and is interested in predicting the expected output.

Experiments with object detection were conducted. Results were reached with Microsoft Kinect sensor. Precision of algorithm was under investigation while conducting this research. Figs. 7-13 present real-time situations with urban traffic car.



Fig. 7. View on multiple objects (pedestrians) without calibration: (a) view from center of the car mask, and (b) change the angle of projection



Fig. 8. View on multiple pedestrians with calibration: (a) view from centre of the car mask and (b) change the angle of projection

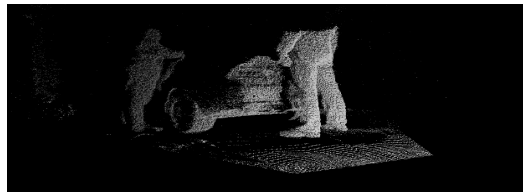


Fig. 9. View on multiple objects and changing position of the camera



Fig. 10. View on multiple cars: (a) the depth of shades and (b) infrared detector



Fig. 11. Camera mounted: (a) view from car mask and (b) view from car bumper

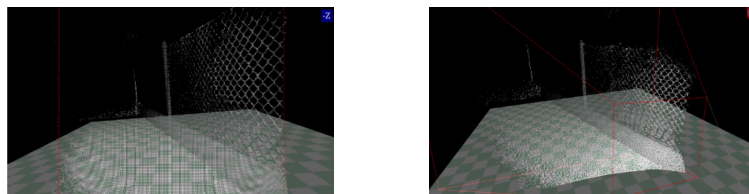


Fig. 12. View of the side of the road with depth detection, camera mounted: (a) on car mask and (b) on car right fender

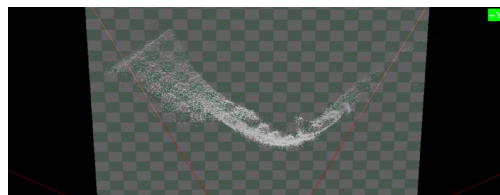


Fig. 13. The car mask reconstruction in 3D cloud points

It is clear that algorithm is well situated for detection of objects (human and vehicle). The next step of research was directed to the “needs” of traffic. Tests were made to prove occurrence detection for different speeds and different braking points used for test object. Coordinates of exact point of detection were recorded for almost 4000 generated objects. There was also full information about starting, braking, and occurrence speed saved for the whole period of experiment. Figs. 14a and 14b present histograms and observed occurrences (variety of distances).

X and y coordinates were compared to detect any instabilities or errors influencing experiment results. No errors were found, and coordinates were stable (changes on x-axis were visible without delay on y-axis). Additionally, there was also trend and probability test made to check if results for start and braking speeds were recorded at values expected in this test. Test from Fig. 15a gave much

more detailed information about change in occurrence of the recorded speed while changing (lowering) starting speed. Angle of downtrend was much sharper for occurrence speed at the beginning of experiment. This meant that making starting speed slower was giving better results at low values, in comparison with what it was showing for greater values. Gain was smaller with high speeds.

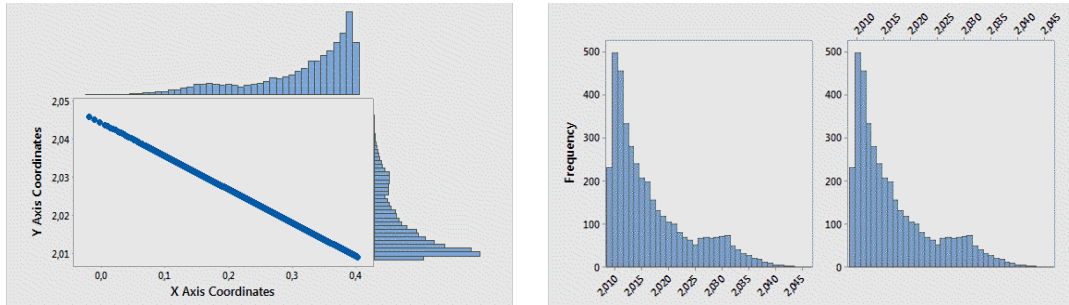


Fig. 14. (a) Histograms of X and Y coordinates, (b) X and Y coordinates of observed occurrence

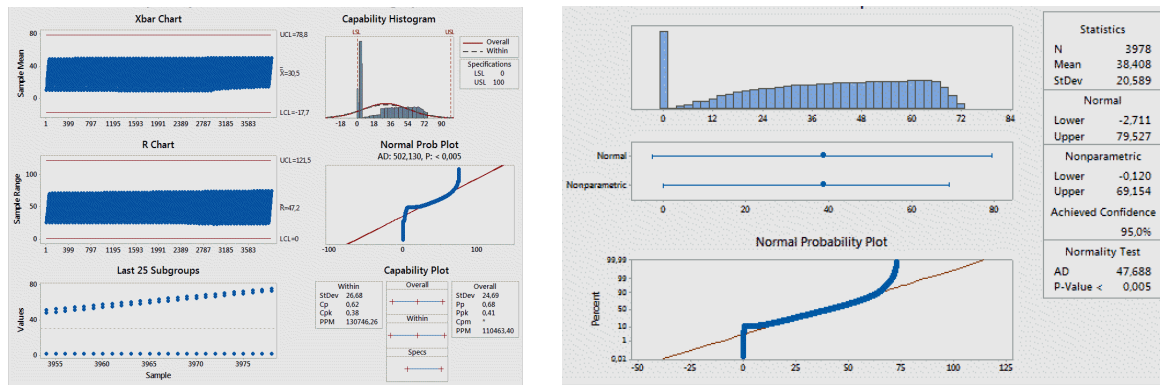


Fig. 15. (a) Probability and Trend Analysis for comparison of three streams, three speeds recorded, and (b) Tolerances observed for start, occurrence and braking speeds

Charts from Fig. 15b show process of the observed tolerance. It clearly presents how test was conducted and how gradually speed was rising. Those results were mostly placed along the 100% probability line, which means process was controlled in the whole range of experiment, and no signal errors occurred.

8. CONCLUSION

In this article, a real-time algorithm to detect multiple moving objects in urban traffic was presented. Technique was proven to require no knowledge about traffic. In the future investigations, there might be more need of exploring the possibility of estimating motion object to determining their direction of motion. This could work better for locating moving objects and describing probable velocities of objects. Hardware implementation of proposed solution based on the processing system was also explained, which could be well situated in the car system. The idea to implement simple artificial intelligence to approach support of human actions and improve the recognition performance of any actions should also be expanded in the future; it is promising and well worth it.

To further expand the knowledge about IR sensors, the next experiments will focus on improvement of vehicle safety, through an interaction phase between primary and secondary safety systems, which has been defined in Real Sense system. It needs to be able to use information provided by both, as it is essential to precisely discriminate whether a collision is avoidable or not. There is a

need of development for general purpose algorithm implementation solving problem for safety systems coexistence.

Acknowledgment

The authors acknowledge the support received from Poznan University of Technology while conducting studies required for this paper. We would like to especially express gratitude to Piotr Kicki and Krzysztof Kolanowski from the Institute of Automatic Control and Robotic.

References

1. *Polish statistics about traffic collisions 2016. Chapter 2, Data about cars and collisions, Chapter 3, Time and place of collisions.* ITS. Warsaw, 2016. P. 5-21. Available at: <http://statystyka.policja.pl/download/20/236480/Wypadki2016.pdf>.
2. *Automotive emergency braking initiative, NHTSA-2015-0101.* U.S. Department of Transportation. National Highway Traffic Safety Administration. March 2016.
3. Chen, Y. & Zhang, Rh. & Shang, L. A Novel Method of Object Detection from a Moving Camera Based on Image Matching and Frame Coupling. In: *PLoS ONE*. 2014. Vol.9. No.10. P. 3-5.
4. Shengping, Z. & Hongxun, Y & Shaohui, L. Spatial-temporal nonparametric background subtraction in dynamic scenes. In: *2009 IEEE International Conference on Multimedia and Expo, ICME 2009*. New York City. 2009. P. 2-3.
5. Wei, Z. & Xiangzhong, F. & Xiaokang, Y, & Wu, J. Spatiotemporal Gaussian mixture model to detect moving objects in dynamic scenes. *Journal of Electronic Imaging*. May 2007. Vol. 16. No. 2. P. 2-4.
6. Royden, C & Holloway, M. Detecting moving objects in an optic flow field using direction- and speed-tuned operators. *Vision Research*. May 2014. Vol. 98. P. 14-25.
7. Royden, C. & Sannicandro, S. & Webber, L. Detection of moving objects using motion- and stereo-tuned operators. *Journal of Vision*. June 2015. Vol. 15. No. 8. P. 3-4.
8. Rodríguez-Canosa, G. & Thomas, S & Cerro, J. & Barrientos, A. & MacDonald, B. Real-Time Method to Detect and Track Moving Objects (DATMO) from Unmanned Aerial Vehicles (UAVs) Using a Single Camera. *Remote Sensing*. April 2012. Vol. 4. No. 4. P. 9-12.
9. Araki, S & Matsuoka, T. & Takemura, H. & Yokoya, N. Real-time tracking of multiple moving objects in moving camera image sequences using robust statistics. In: *Proceedings. Fourteenth International Conference on Pattern Recognition*. 1998. P. 3-4.
10. Kim, D. & Kwon, J. Moving Object Detection on a Vehicle Mounted Back-Up Camera. *Sensors*. December 2015. Vol. 16. No. 1. Switzerland. P. 4-8.
11. Talukder, A & Goldberg, S. Matthies, L. Ansar, A. Real-time detection of moving objects in a dynamic scene from moving robotic vehicles. In: *Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2003*. Las Vegas. October 2003. P. 3-5.
12. Lefaix, G. & Marchand, E. & Bouthemy, P. Motion-based obstacle detection and tracking for car driving assistance. In: *IAPR Int. Conf. on Pattern Recognition. ICPR'02*. Quebec, 2002. P. 74-77.
13. Chen, Y. & Zhang, Rh. & Shang, L. & Hu, E. Object detection and tracking with active camera on motion vectors of feature points and particle filter, *Review of Scientific Instruments*. June 2013. Vol.84. No. 6. P. 3-4.
14. Junghye, M. & Kasturi, R. Activity recognition based on multiple motion trajectories. In: *Proceedings of the 17th International Conference on Pattern Recognition, ICPR 2004*. September 2004. Vol. 4. Cambridge. P. 3.
15. *U.S. DOT and IIHS announce historic commitment of 20 automakers to make automatic emergency braking standard on new vehicles*, U.S. Department of Transportation. National Highway Traffic Safety Administration. Washington, DC. March 17, 2016. Available at: <http://www.nhtsa.gov/About+NHTSA/Press+Releases/nhtsa-iihs-commitment-on-aeb-03172016>.

16. Jiménez, F. & Naranjo, J. & García, F. An Improved Method to Calculate the Time-to-Collision of Two Vehicles, *International Journal of Intelligent Transportation Systems Research*. January 2013. Vol. 11. No. 1. P. 34-42.
17. *Polish statistics about traffic collisions 2015, Chapter 2, Data about cars and collisions, Chapter 3, Time and place of collisions*, ITS. Warsaw. 2015. P. 5-20. Available at: <http://statystyka.policja.pl/download/20192140/Wypadki2015.pdf>.
18. Mroziak, M. Tests for speed evaluation of pedestrian traffic users move. *Advances in Science and Technology Research Journal*. 2010. Vol. 5. P. 138-144. Lublin.
19. *Prędkości pojazdów w Polsce Raport 2013*, Krajowa Rada Bezpieczeństwa Ruchu Drogowego, Gdańsk, Kraków, Warszawa, Poland. 2014. P. 13-18. Available at: <https://www.obserwatoriumbrd.pl/resource/38187803-1a05-48c9-ac57-0b9b3e9b6801:JCR> [In Polish: *2013 Report of vehicle speeds in Poland*, National Road Safety Council].
20. *2013 Production Statistics*, Int. Organization of Motor Vehicle Manufacturers. OICA. Paris, France. 2013. Available at: <http://www.oica.net/category/production-statistics/2013-statistics/>.
21. *Opis i rekonstrukcja wypadków drogowych*, Faculty of automotive and construction machinery engineering, Warsaw University of Technology. Warszawa. 2016, P. 117-133. Available at: http://www.mechatronika.info.pl/dump/Opis_i_rek_wyp_dr.pdf [In Polish: *Reconstruction of traffic collisions*].
22. Blais F. Review of 20 years of range sensor development. *Journal of Electronic Imaging*. January 2004. Vol. 13. No. 1. P. 3-4.
23. *Integrating Safety into the EU's Urban Transport Policy*. European Transport Safety Council. ETSC. Brussels, Belgium. June 2013. P. 1-10. Available at: http://etsc.eu/wp-content/uploads/2014/03/ETSC_Position_on_Urban_Transport_June_2013.pdf.
24. *European Road Safety Report 2014 Urban Mobility, Strategies to prevent accidents on Europe's roads*. DEKRA Automobil. GmbH. 2014. P.6-31. Available at: <https://www.dekra-roadsafety.com/media/dekra-roadsafety-report-2014-engl.pdf>.
25. Szablata, P. n& Łakowski, P. & Pochmara, J. Processing depth distance data to increase precision of multiple infrared sensors in the automatic visual inspection system. *Optical Engineerin*. August 2017. Vol. 56. No. 8. P. 3-6.
26. Mankoff, K & Russo, T. The Kinect: a low-cost, high-resolution, short-range 3D camera. *Earth Surface Processes and Landforms*. November 2012. Vol. 38. No. 9. P. 926-936. John Wiley & Sons Ltd.
27. *Intel® RealSense™ Camera R200, Embedded Infrared Assisted Stereovision 3D Imaging System with Color Camera*, Rev. 001. Intel Corporation. June 2016. P. 15-19. Available at: http://www.mouser.com/pdfdocs/intel_realsense_camera_r200.pdf.
28. Gollapudi, S. *Machine learning vs Deep learning vs Artificial Intelligence*. 24 July 2017. Available at: <https://www.linkedin.com/pulse/machine-learning-vs-deep-artificial-intelligence-sunila-gollapudi/>.

Received 12.06.2019; accepted in revised form 27.12.2020