Target person identification and following based on omnidirectional camera and LRF data fusion

Mehrez Kristou, Akihisa Ohya and Shin'ichi Yuta
Intelligent Robot Laboratory, University of Tsukuba, Japan.
{anjin,ohya,yuta}@roboken.esys.tsukuba.ac.jp

Abstract— **In this paper, we present the current progress of our approach to identify and follow a target person for a service robot application. The robot is equipped with LRF and Omni directional camera. Our approach is based on multi-sensor fusion in which a person is identified using the panoramic image and tracked using the LRF. The selection of the target person is implemented to improve the identification when multiple candidates are detected. Our approach is successfully implemented on a mobile robot. A simplified target person following behavior is implemented to focus on the proposed method's efficiency. Several experiments are conducted and showed the effectiveness of our approach to identify and follow human in indoor environments.**

I. Introduction

A luggage cart is the most used tool in airports. It is handy and useful to carry luggage for long distances. However, in many cases this tool can be problematic to handle and difficult to use. A robot having as function to carry luggage and to follow the customer can be of a great help. This kind of robot rises many challenges. Such a robot, called service robot, should be able to interact with people and coexist with them in the same space. The robot also needs to fulfill its functions in a crowded environment designed basically for humans.

The tour-guide robot of Burgard et al. [1] adopts only laser sensor to implement people tracking both for interacting with users and for mapping the environment, discarding human occlusions. Other research consider the usage of camera. In the work of Jianpeng Zhou et al. [2], they present a real time robust human detection and tracking system for video surveillance which can be used in varying environments. To ensure more accuracy, it is common to use sensor fusion and especially a combination of camera and LRF. For instance, in the work of Luo et al. [3] and Bellotto et al. [4], the method uses a laser to extract body features, which are fused then with the face detected by a camera. The solution is useful for pursuing a person in front of the robot. However, the usage of face detection to identify target person limits the expected natural behavior. In the work of Wilhelm et al. [5], a skin color based identification with LRF data fusion is used to detect and follow a human in a path-way. The usage of skin color is useful to detect a human, however it is not enough discriminative to identify a human among others.

In our work presented in [6], we implemented a tentative method for human identification and tracking from a mobile robot in a fixed situation. We used a registered set of person clothes patterns to identify the target person in the image and locate its positions in the laser scans clusters. The resulting identified clusters are the potential target person positions. Our early proposed method relied on the target person clothes pattern registration step which required a full spin of a person in front of the robot. In the work presented in this paper, the patch registration was redesigned to get patches while the target person cluster, initially identified, is not yet lost. Using the calibration of the patch size and location in the image and the cluster size and distance, we improved the extraction of patches from the chest level of the target person while he is moving. In this work, we try to keep the individual modules' complexity low and focus on modules' data fusion to reach better performance. Also, We use sensor fusion approach to solve the target person identification and tracking.

Also, to improve the patch detection processing speed, we are limiting the search area to the clusters view angles. We improved the cluster tracking stage by including, in addition to the position and size, the speed and direction's information to each cluster data. We introduce in this paper a multi hypothesis target cluster selection to choose the target among the potential target person positions. Once the target person position is detected, the target person following step will allow the robot to keep a constant distance and a facing direction to the target.

II. Sensors configuration

Our system describes a luggage cart robot which is able to identify its temporary target person and follow him in a relatively crowded environment. To implement these tasks, we propose the following configuration of sensors.

The robot in Fig.1 is a mobile wheeled robot equipped of two sensors: Omni directional camera and LRF. The camera is mounted on the same height of the human head to allow the face recognition. The LRF is mounted under the camera and shares the same vertical axis with it. The scan field of the LRF is limited to 270deg centered to the front of the robot. We use the left blind field to pass the cabling and the fixing frame to the robot. The two sensors are rigidly attached to the robot. Using the mirror transformation derived in the work of Baker *et al.* [7], we transform the circular image into a panoramic image.

The resulting panoramic image is a 360deg image. The origin of the image coordinate system is the top left corner. The horizontal axis represents the angular dimension. Its

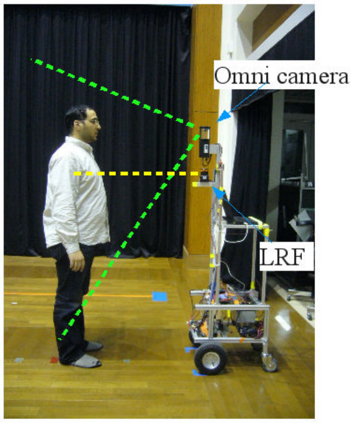

Fig. 1. Sensors configuration. The camera and the LRF are sharing the same vertical axis and their measured angular position is calibrated. The green dotted line represents the camera field of view and the yellow line represent the LRF detection level.

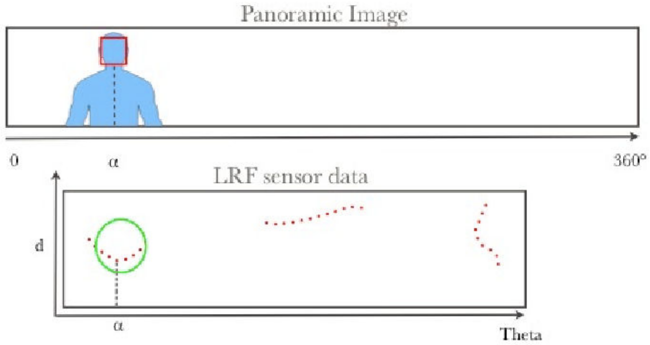

Fig. 2. Plotting the face detection result and the LRF clustering result in the same polar coordinate allows a direct transformation of the calculated angle α from one sensor to another.

angular resolution is 0.309 $degree/pixel$. The vertical axis represents the field of view. The image width is 1161 $pixels$ and its height is 186 $pixels$. The camera and LRF are sharing the same Z axis. As shown in Fig.2, using the position of the detected face in the image and the degree/pixel ratio, we convert its position into an angle. After calibration, we convert the face angle from the image coordinate into LRF coordinate system. The cluster located at the same angle as the face is associated to it.

III. Human tracking stage

Our human tracking algorithm adopts multi-sensor data fusion techniques to integrate the following two different sources of information: the panoramic image adapted from an omni directional camera and laser scans of the LRF.

A. *People tracking*

Using the LRF data, we propose to keep track of all clusters having a shape and a size likely to belong to a person. For each cluster, we need to know its position, speed and direction. First, we consider the points received from the LRF as chain of data. We compute the Euclidean distance between two successive points, if the distance is more than a pre-defined constant, we break the chain into two and we continue to the last point. The result will be a set of clusters of data corresponding to the visible unconnected objects.

Then, we apply a series of rejectors on the set of clusters to have in the end clusters which may belong to the target person. These rejectors include a linearity filter to eliminate clusters having straight line shape (like walls and flat surfaces) and a maximum human width to keep only clusters having likely a human shape and width.

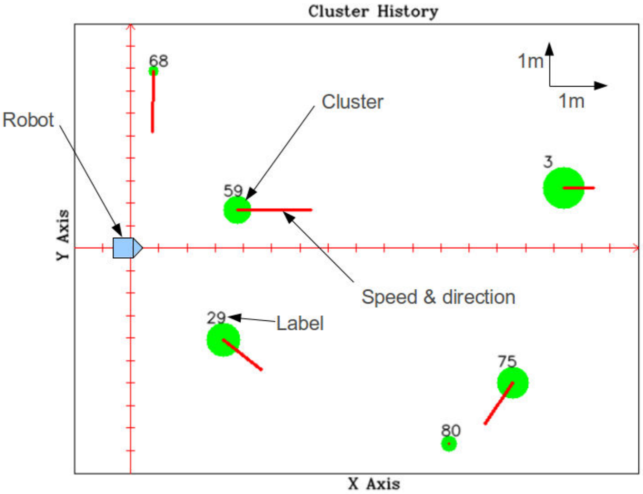

Fig. 3. The cluster history output screen-shot.

For each LRF scan, we need to label each cluster so that a cluster belonging to the same object should have the same label. Two clusters of successive LRF scans and corresponding to same object represent two instances of the object's observation. The ordered list of these instances is the track of that object. The top of the list is the latest instance and the bottom is the oldest. Each instance's speed and direction are calculated from the average of the last instances' time and space differences to have stable readings. When a new cluster is detected, we forecast the position of each top instance of each track. Then we associate that cluster with its corresponding track if it satisfies the minimum Euclidean distance to its forecasted instance's position. Finally its speed and direction are updated accordingly. If track does not get a new instance for a fixed duration, its state is affected to "lost" and removed. The structure holding all tracks is called the positions history. The positions history structure, shown in fig.4, can be defined as 2D table where each row corresponds to a track and each column correspond to a time instance when the corresponding cluster is detected. So rows are candidates and columns are instances.

Fig. 3 shows an example of this step's output where currently detected clusters are represented as circles and the speed and direction are shown as a segment. The circle's size presents the cluster width and the length of the segment is proportional to the cluster's speed.

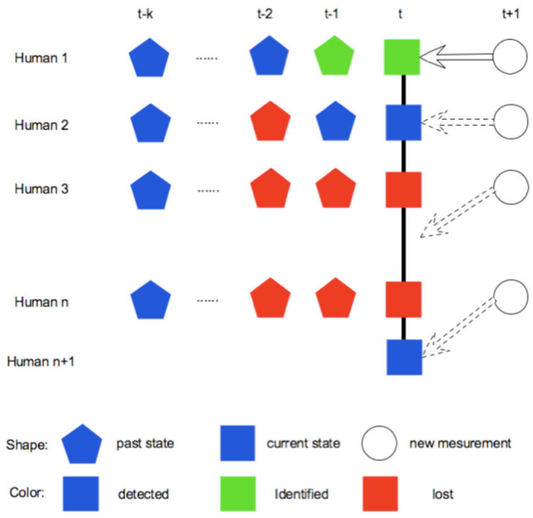

Fig. 4. Positions history structure where rows are candidates, columns are instances and cells are candidates' position and state. Measurements are affected to the corresponding candidate's row depending on the shortest Euclidean distance between the last position of the candidate and the measurement.

B. *Target person identification*

The identification of the target person relies on the color pattern matching of patches extracted from all sides of the target person on the level of the target chest in the current frame of the omni-directional camera. In the previously proposed method [6], we order the target to have a full spin in front of the robot. This scenario was criticized for its non applicability in real applications. In this paper to propose to extract the needed color patches without imposing a special behavior to the target person.

Initially, the robot is waiting, in a parking lot, for a human interaction. When a person is present in front of the robot, a face detection's algorithm[8] gives his face position and size in the image. If the person confirms the operation, the identification of the target person starts. This step collects samples of the clothes of the target person, we call them patches. The frame origin is the upper left corner and its y-axis is the vertical axis. We want to have a patch from the chest level so we use the position and dimension of the face detection to calculate the region of the patch. The dimension of the patch and its horizontal position are equal to the face's dimension and position. The vertical position of the patch y_{Patch} , expressed in pixels, is relative to the vertical position of the face y_{Face} and its height $height_{Face}$ where $y_{Patch} = y_{Face} + 1.25 height_{Face}$. y_{Face} is the top edge of the face detection region. That region corresponds to the chest level of the detected human. Using the relationship deducted from the calibration of a cluster distance with the match's size and vertical position shown in Fig. 5, we collect as much different patches as possible while the human is in a range distance of 2 meters starting from the start of the system. When exceeding this distance, the patch size gets too small to be used. Fig. 6 shows a sample of patches and there location in the image. Fig. 7 shows a typical system start with patch extraction. After detecting the face and calibrating the patch extraction region, the system starts extracting patches while the target person is in the 2 meters distance range from the robot. The green square depicts the patch extraction region and the red one shows the patch detection results.

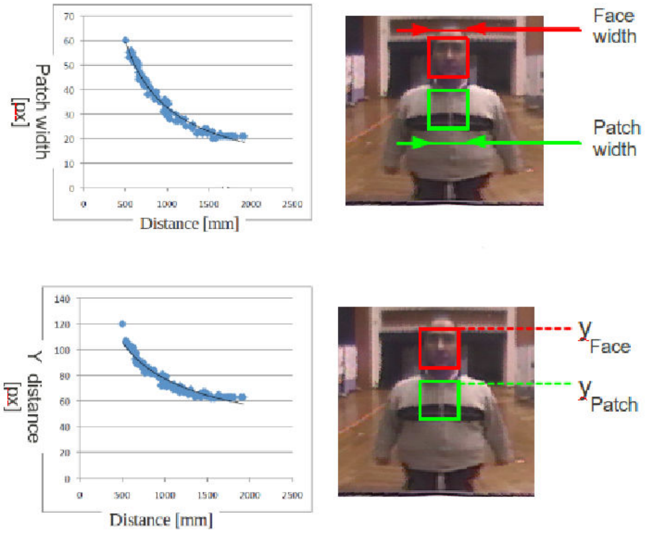

Fig. 5. The calibration of a cluster distance and its relationship with the match's size and vertical position

In current method, identifying the temporary target person in the panoramic image relies on the color histogram matching algorithm coupled with a down-sampling algorithm implemented in OpenCV[9] to accelerate the searching time. The output of this algorithm is a correlation map calculated using the panoramic image and each collected patch. The maximum intensity value corresponds to the best match. The searching area is limited to the area where currently detected clusters of the position history structure are seen. We convert each matched region's position from the image coordinate to an angle expressed in the LRF coordinate system. The result is a list of the highest matched regions with their respective angular positions. We use a threshold on minimum coefficient value to limit the number of selected matches. The upper row of Fig. 9 shows a screen-shot of the output of this stage where red squares are the matched area and the number beside them are their respective matching coefficients.

C. *Target person's selection*

Each of the previously presented steps gives an information about the target person seen from different sensors. The the target person's selection step has to use all this information to select the cluster which belongs to the target person and gives his position to the target person's following step.

Using the outputs of the last two steps: the angular position of the matches in the panoramic image and the positions history from the LRF. This step matches cluster tracking results with vision to identify visually matched tracks.

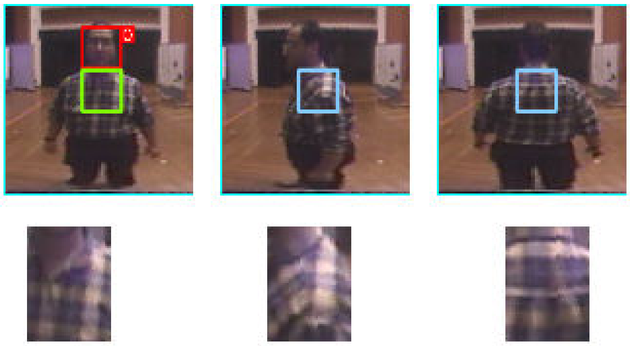

Fig. 6. Upper row: target person from three different angles. The red square is the face detection output. The green square is the extraction reference position and the first patch. The blue squares are the next extracted patches positions. Bottom row: the extracted patches.

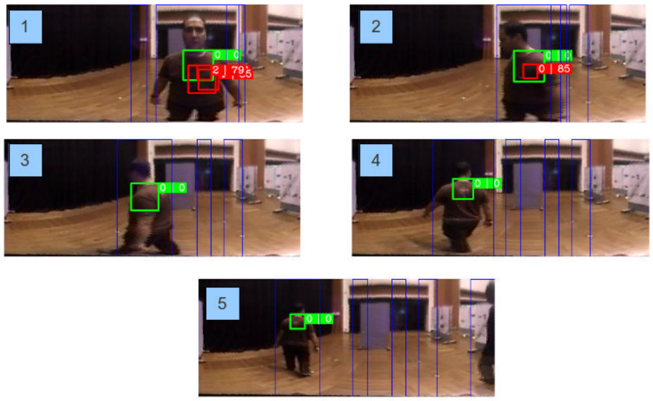

Fig. 7. A sequence of images in a typical system stating with patch extraction. Green squares depict the patch extraction region. The red squares depict the patch detection results.

Using the calibration results in Fig. 5, we determined the relationship between the match's width of the person chest in the image and the real distance of the person from the robot. We associate bigger matches with nearer clusters and vice versa. Thus we associate to each match a validity range (represented in bottom of Fig. 9 as vertical red segment). A track having instances located at the same angle as the detected match and in its validity range is labeled as "visually matched track". The latest instance of the visually matched track is the target to follow and its position will be sent to following step.

Knowing that the image processing is relatively time consuming, we needed to treat the synchronization problem between the two steps. We apply the matching between the received matches and its corresponding instance whenever they are near in the time space. If one instance of a track is matched, the track is considered as visually matched. Fig. 8 shows a screen-shot where an instance initially unidentified (Blue color) gets identified when it overlaps with matches (vertical red segments). The data fusion is done in the past, so the overlapping is not to be seen in the current instance but the track gets identified (violet color).

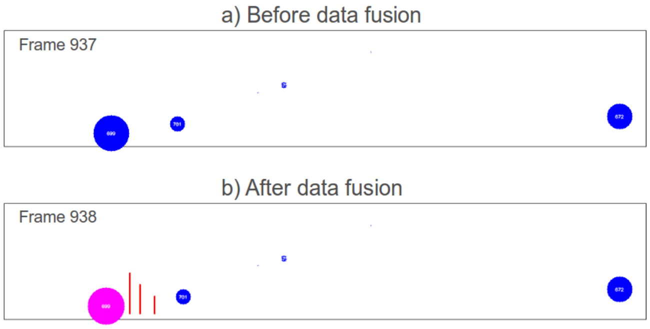

Fig. 8. Two successive screen-shots of the target person selection step.

The matches and tracks fusion can result in many visually identified tracks, so we need to select the target of the current time among them.

Having as a base the enriched positions history (the position history with all its tracks states) and the last selected target, the robot has to decide the current selected target to follow. To cover all cases, we use a truth table of all possible situations. We consider:

- NO: No visually matched track in the current time.
- OO: Only one visually matched track in the current time.
- MO: Many visually matched tracks in the current time.
- EA: Last identified target is found in all track of the current time.
- EO: Last identified target is found in the visually matched tracks.

TABLE I

THE TRUTH TABLE OF THE TARGET SELECTION STEP.

	$\neg EA$	$EA \wedge \neg EO$	EO
NO	lost	update position	Not applicable
OO	Select	wait next	update position
MO	no decision	wait next	update position

Considering all cases in Table I, the perfect case is expressed when the last identified target appears in the current scan $((EO \wedge OO) \vee (EO \wedge MO) \vee (EA \wedge \neg EO \wedge NO))$. In that case, his position is updated and the following stage will take care of approaching him. When the tracked target is lost $(\neg EA \wedge NO)$, the robot stops the following and waits for other possible case. When no decision can be made $(\neg EA \wedge MO)$, the robot waits for the next scan to confirm the situation. The difficult situation is faced when the last identified target exists in the track list but not in visually matched and a new visually matched track appears $((EA \wedge \neg EO) \wedge (OO \vee MO))$. In this case, a probabilistic method can be applied. For our current system, we select the new visually matched track to be the following stage target.

IV. Target person following stage

The result of the human tracking stage is the identified track of target person. This track has the spacial position, the average speed and the average direction. The main purpose of this stage is to allow the robot to keep a fixed distance to the target person. This fix distance is called the target's proximity distance. To evaluate the proposed fusion technique while the robot is moving, we keep the following stage's implementation simple.

We designed this step to follow the target person by driving the robot directly towards the person's location. Basically, we try to keep a constant distance separating the robot from the target person. We control the robot velocity and angular speed by keeping them proportional to the person distance and angle.

V. Experimental results

To test the performance of the proposed approach, the system has been implemented on a mobile robot, shown in Fig. 1, provided with a LRF sensor (UTM-30LX, Hokuyo Automatic Co., Ltd. [10]) as a LRF and an omni-directional camera (a parabolic mirror mounted on a Sony's CCD camera). The two sensors are mounted on a rigid frame at approximately 1.65m from the floor to allow the face detection. The resolution of the laser device is ±1% of the distance, with a scan every 0.36deg at 40 Hz, whereas the camera provide images with a resolution of 640 x 480 pixels at 30fps. The on-board PC is a Core 2 Duo 2.5 GHz with 4 GB of RAM. The whole software has been written in C++.

To understand the system internal state better, we implemented a set of screen outputs. Fig. 9 shows the sensors configuration and their respective scan fields. We use the polar coordinate system because it makes easy to identify objects appearing in the image and their corresponding tracks.

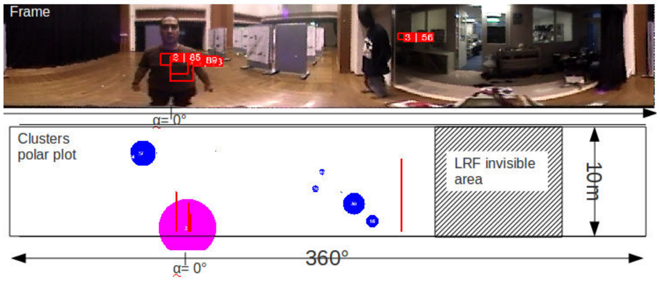

Fig. 9. Screen-shot of our system. The upper image is the 360deg image. The lower figure is the position history building result where circles are latest instances and vertical lines are the validity range of the detected patch. The circle diameter is relative to the cluster width.

To check the system behavior in case of target person's occlusion, we conducted the experiment in a hall in presence of two persons. The target person stands in front of the robot and starts the process. Then, the target walks around while the other person tries to cross between him and the robot. During the experiment, the maximum robot's speed is set to 1 m/s and the target's proximity distance to 0.5 meter for the safety reasons.

Fig. 10 shows a segment of a typical run during which the robot covers an area of 6x6 meters developing maximum speed of 1 m/s. This particular run was chosen because it demonstrates several interesting situations that could happen during the robot service time. In a relatively crowded environment, it is common that robot cannot keep distance which does not allow people to cross between it and the target person. Fig. 10 shows such a case. When the robot is tracking correctly the target person's cluster, a temporary loss of the target person does not influence the overall following behavior because the robot continues to approach the last location where he was seen. If the locked target is lost, the first visually identified track is selected to be a locked target for following. Fig. 11 shows three successive target identification result's frames where in frame 24 a target is successfully identified. In frame 26 the target person is occluded by another person and his position reading is lost. When no target is found, the robot stops waiting for the target to reappear. After an occlusion, the target person is identified again and the robot continues its following. Fig. 12 shows the described scenario from the target selection screen output. The instances associated to the target person (violet) gets occluded by another instance (blue). After the occlusion, a new track appears and gets visually identified.

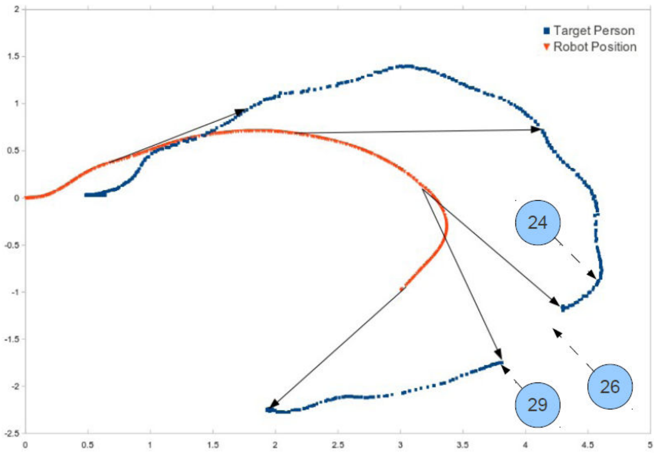

Fig. 10. A segment of a typical run: The red points represent the robot positions, the blue points represent the estimated positions of the target person. The errors show the correspondence of some critical robot positions to the target positions. Circles represent the associated frame number.

VI. Conclusion and Future Works

In this paper, we have presented the current progress of a multi-sensor based human identification and following system for autonomous luggage cart robot. Many improvements and missing step were added to the proposed method presented in earlier work [6]. The initialization step was replaced integrated in the target person's identification. Samples of the target clothes were extracted while he moves away from the robot. The target person's selection was added to choose the target among possible candidates which results from the fusion. The target following stage was added to check the performance of the proposed method while the robot is moving. The experimental results show the validity of

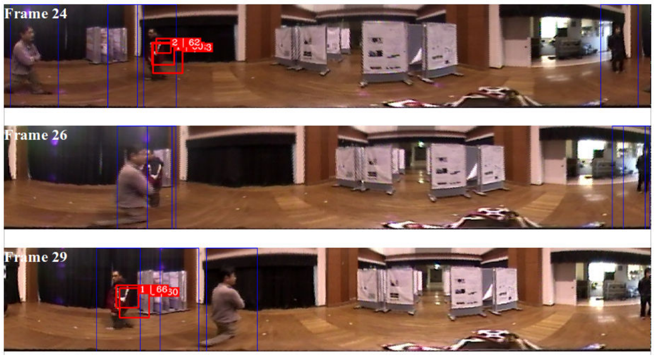

Fig. 11. Three successive target person identification frames corresponding to the typical run 10. The red squares represent the patch detection result and the number shows the matching coefficient.

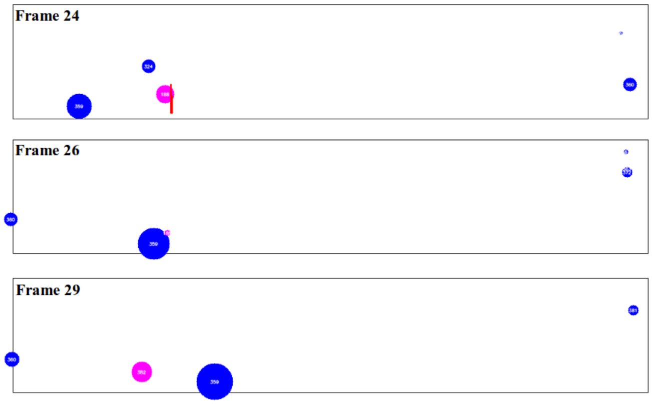

Fig. 12. Three successive target person selection frames corresponding to the typical run 10. The pink circle represent the visually identified track corresponding to the estimated position of the target person. Blue circles are unidentified tracks.

the method. A future work is to increase the environment complexity to check the method limits. Also, it is necessary to implement a moving obstacles' avoidance method to allow the robot to navigate smoothly in a crowded environment.

References

[1] W. Burgard, P. Trahanias, D. Hähnel, M. Moors, D. Schulz, H. Baltzakis, and A. Argyros, "Tourbot and webfair: Web-operated mobile robots for tele-presence in populated exhibitions," in *Proc. IROS Workshop Robots Exhib.*, 2002, pp. 1–10.

[2] R. C. Luo, Y. J. Chen, C. T. Liao, and A. C. Tsai, "Mobile robot based human detection and tracking using range and intensity data fusion," in *Proc. IEEE Workshop Adv. Robot. Social Impacts*, 2007, pp. 1–6.

[3] Q. Zhu, S. Avidan, M. Yeh, and K. T. Cheng, "Fast human detection using a cascade of histograms of oriented gradients," in *TR2006-068*, Jun 2006.

[4] N. Bellotto and H. Hu, "Vision and laser data fusion for tracking people with a mobile robot," in *Robotics and Biomimetics, 2006. ROBIO '06. IEEE International Conference on*, dec. 2006, pp. 7 –12.

[5] T. Wilhelm, H.-J. Bohme, and H.-M. Gross, "Sensor fusion for vision and sonar based people tracking on a mobile service robot," *In Proc. International workshop on Dynamic Perception*, pp. 315–320, 2002.

[6] M. Kristou, A. Ohya, and S. Yuta, "Panoramic vision and lrf sensor fusion based human identification and tracking for autonomous luggage cart," in *The 18th IEEE International Symposium on Robot and Human Interactive Communication, 2009. RO-MAN 2009.*, 27 2009-Oct. 2 2009, pp. 711 –716.

[7] S. Baker and S. K. Nayar, "Single viewpoint catadioptric cameras," in *Panoramic Vision: Sensors, Theory, Applications*, R. Benosman and S. B. Kang, Eds. Springer-Verlag, 2001.

[8] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, December 2001.

[9] Willow Garage, "Open source computer vision library (opencv)," Retrieved August 1, 2010, from http://opencv.willowgarage.com, 2010.

[10] H. Kawata, A. Ohya, S. Yuta, W. Santosh, and T. Mori, "Development of ultra-small lightweight optical range sensor system," in *IEEE International Conference on Intelligent Robots and Systems (IROS),*, 2005, pp. 3277–3282.