RGB-D Human Detection and Tracking for Industrial Environments

Matteo Munaro, Christopher Lewis, David Chambers, Paul Hvass and Emanuele Menegatti

Abstract Reliably detecting and tracking movements of nearby workers on the factory floor are crucial to the safety of advanced manufacturing automation in which humans and robots share the same workspace. In this work, we address the problem of multiple people detection and tracking in industrial environments by proposing algorithms which exploit both color and depth data to robustly track people in real time. For people detection, a cascade organization of these algorithms is proposed, while tracking is performed based on a particle filter which can interpolate sparse detection results by exploiting color histograms of people. Tracking results of different combinations of the proposed methods are evaluated on a novel dataset collected with a consumer RGB-D sensor in an industrial-like environment. Our techniques obtain good tracking performances even in an industrial setting and reach more than 30 Hz update rate. All these algorithms have been released as open source as part of the ROS-Industrial project.

Keywords Human detection and tracking · ROS-Industrial · RGB-D · Open source

M. Munaro (✉) · E. Menegatti
Department of Information Engineering, University of Padova, Via Gradenigo 6B,
35131 Padova, Italy
e-mail: munaro@dei.unipd.it

E. Menegatti
e-mail: emg@dei.unipd.it

C. Lewis · D. Chambers · P. Hvass
Southwest Research Institute, San Antonio, TX 78238, USA
e-mail: christopher.lewis@swri.org

D. Chambers
e-mail: david.chambers@swri.org

P. Hvass
e-mail: paul.hvass@swri.org

E. Menegatti et al. (eds.), *Intelligent Autonomous Systems 13*,
Advances in Intelligent Systems and Computing 302,
DOI 10.1007/978-3-319-08338-4_119

1 Introduction

The next generation of robots, both for service or cooperative work, is expected to interact with people more directly than today. Industry is more and more interested in exploiting both the dexterity and versatility of people and the precision and repeatability of robots by enabling collaboration in dynamic and reconfigurable manufacturing environments. Such collaborations, however, are not yet possible because robots are still not capable of safely interacting cooperatively with their human coworkers in highly variable task scenarios. In most cases, safety of the users interacting with industrial robot manipulators has been addressed by using safety guards or other barriers to cordon robots off from people. This assumption cannot be considered if humans and robots have to share the physical environment or collaborate. In that case, people have to be detected in order to prevent collisions with the robot. Moreover, if people are not only detected, but their movements are tracked over time, their intentions and future positions can be estimated. Based on this high-level information, the robot can choose the strategy to avoid collision which could minimize the time needed to resume operations, thus reducing productivity slowdowns. In this work, we propose methods for detecting and tracking people by means of vision. This approach allows to avoid the costs and hazards involved with perimeter fencing.

People detection and tracking have been deeply studied by the computer vision and robotics communities. However, unlike video surveillance or service robotics scenarios, people tracking for human–robot interaction in industrial workspaces requires to meet a certain number of constraints which should guarantee the safety of workers. Among these, all people within the workspace should be detected, even if partially occluded, and their position should be sent to the robot control system as soon as they enter the operating area. To meet the latter constraint, the system update rate should be high (from 15 to 30 Hz) and the latency should be minimum (< 0.2 s). It is worth noting that system reactivity is not necessarily related to overall tracking accuracy. Finally, it would be desirable if the system could also work when the camera is placed onboard of a moving vehicle or horizontally translated on a track to follow a person while moving in a wide area. This assumption requires to avoid people detection techniques based on background subtraction in the color or depth image.

Algorithms which rely on 2D images [3, 8] are usually too slow and sensitive to clutter and occlusion. Thus, depth-based approaches exploiting passive or active sensors are usually preferred. Passive sensors, such as stereo cameras [9, 10], have the need for finding correspondences between left and right image points, which is a computationally expensive operation and it can fail for scenes where texture is poor. For these reasons, an active sensor is usually preferable. In the past, approaches based on active depth sensors, such as Laser Range Finders [4, 14, 17, 22, 23], have been limited by the fact that 3D sensors had low resolution and high prices. With the advent of reliable and affordable RGB-D sensors, such as Microsoft Kinect,[1] cameras providing aligned color and depth measurements of the scene became available, thus allowing for algorithms which could exploit this combined information. Since a

[1] http://www.microsoft.com/en-us/kinectforwindows.

single sensor can provide a dense 3D representation of the scene at 30 Hz, robust algorithms can be applied and the system installation becomes straightforward. In this work, the proposed algorithms are targeted to be used on RGB-D data from structured light, stereo, or time-of-flight sensors.

The remainder of the paper is organized as follows: in Sect. 2, the literature on RGB-D people detection and tracking is reviewed, while the ROS-Industrial project is presented in Sect. 3. Section 4 describes the dataset we use in this paper, while our people detection and tracking algorithms are reported in Sects. 5 and 6 and the full pipelines are described in Sect. 7. In Sect. 8, we present the experiments we performed and conclusions are drawn in Sect. 9.

2 Related Work

Kinect SDK[2] performs people detection based on the distance of the subject from the background, while NiTE middleware[3] relies on motion detection. Both these approaches work in real time with CPU computation, but they are thought to be used with a static camera and for entertainment applications. Thus, they are not suitable to work in cluttered environments and if other moving objects (robots) are present because of the assumption that the background is static. Moreover, these algorithms only detect people up to 4 m of distance from the camera.

In [21], a people detection algorithm for RGB-D data is proposed, which exploits a combination of *Histogram of Oriented Gradients* (HOG) and *Histogram of Oriented Depth* (HOD) descriptors and is not limited to static sensors or to a restricted distance range. However, each RGB-D frame is densely scanned to search for people, thus requiring a GPU implementation for being executed in real time. Also [6] and [7] rely on a dense GPU-based object detection, while [13] investigates how the usage of the people detector can be reduced using a depth-based tracking of some *Regions of Interest* (ROIs). However, the obtained ROIs are again densely scanned by a GPU-based people detector.

In [12], a tracking algorithm on RGB-D data is proposed, which exploits the multi-cue people detection approach described in [21]. It adopts an online detector that learns individual target models and a multi-hypothesis decisional framework. No information is given about the computational time needed by the algorithm and results are reported for some sequences acquired from a static platform equipped with three RGB-D sensors.

Algorithms for tracking people in real time from a mobile platform have been presented in [1, 15] and [16]. These people detection techniques estimate the ground plane equation and exploit a depth-based clustering followed by color-based classification. They obtain high accuracy and framerate while only relying on CPU

[2] http://www.microsoft.com/en-us/kinectforwindows/develop.

[3] http://www.primesense.com/solutions/nite-middleware.

computation. A similar approach has been implemented in [25], where the concept of *depth of interest* is introduced to identify candidates for detection.

In this work, we propose a people detection approach which combines the techniques proposed in [1, 15] and [16] with the algorithms validated in [5] for detecting people from stereo data. Moreover, we exploit a people tracking algorithm which can recover from missed detections and identity switches by means of a particle filter guided by color histograms of the tracked people.

3 ROS-Industrial

ROS-Industrial is an open-source project that extends the advanced capabilities of the Robot Operating System (ROS) [19] software to new industrial applications. Among the goals of ROS-Industrial, there is the aim to develop robust and reliable software that meets the needs of industrial applications. This is achieved by combining the relative strengths of ROS with existing industrial technologies (i.e., combining ROS high-level functionality with the low-level reliability and safety of industrial robot controllers). ROS-Industrial already provides libraries, tools, and drivers for interfacing with a number of industrial manipulators and it is creating standard interfaces to stimulate *hardware-agnostic* software development. All these aspects concur to create a generation of robots which could be versatile in the tasks they execute and collaborative with humans. Thus, ROS-I is also concerned with the problem of detecting people in shared robot–human workspaces and of defining standard requirements which should be satisfied by such algorithms in order to ensure people safety in industrial environments.

4 ROS-Industrial People Dataset

In order to evaluate the performance of the proposed tracking pipelines in a real industrial setting, the *ROS-Industrial People Dataset* has been collected in a warehouse very similar to an industrial scenario. The dataset contains about 11,000 RGB images and their corresponding disparity images acquired with an *Asus Xtion Pro Live* at VGA resolution. One to five people were moving in front of the camera. Other than people, also robots, objects and industrial tools feature the dataset. About one-fifth of the images do not contain people, in order to test robustness to false positives, and some images were collected while the camera was moving parallel to the ground plane. Some sample RGB, disparity images, and annotations from this dataset are reported in Fig. 1. About one-tenth of the dataset frames are annotated, that is, the corresponding ground truth is provided in files which contain the bounding boxes of the people present in the images.

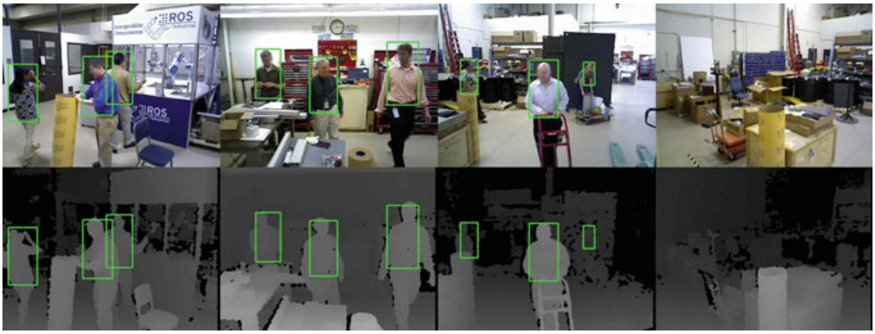

Fig. 1 Sample RGB, corresponding disparity images, and ground truth annotations from the *ROS-Industrial People Dataset*

5 People Detection

In this section, we present the people detection approach we used in this work. Since we cannot rely on the assumption that the background is static, we developed a detection technique which could work from a single frame without knowledge inherited from the previous ones.

5.1 Cascade Classifier Modularity

For limiting the computational onerosity while preserving accuracy, we organized the detection process as a cascade of different algorithms. In this cascade, detection methods are ordered in decreasing order of speed and increasing order of classification accuracy. In this way, we applied a very simple and fast verification process when many detection windows have to be processed, while we keep more accurate but slower detection methods for the last stages of the cascade, when only a small number of detection windows have to be analyzed.

In the next paragraphs, we will describe the algorithms we developed for composing the single stages of the cascade. They can be divided in consistency, depth-based and color-based algorithms.

In Fig. 2, the detections produced at every stage of a detection cascade that we will see in Sect. 7 are shown.

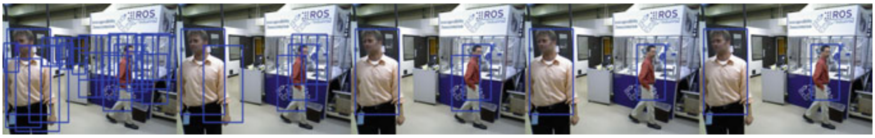

Fig. 2 Visualization of the output detections at every stage of the detection cascade used in the pipeline of Fig. 4a for a frame of the *ROS-Industrial People Dataset*

5.2 Consistency Constraints

The first module we present implements two constraints which allow us to check if a detection window is compatible with people being on the ground plane. The first constraint is a height in image versus disparity constraint. The idea behind this constraint is that a pedestrian of a certain height in an image, measured in pixels, should have a disparity that places the pedestrian at a reasonable distance.

The second constraint forces objects to be located on the ground plane. For doing this, we estimate the ground plane equation with the Hough-based method proposed in [5] and then limit search windows by constraining object height and y location.

The thresholds for these contraints have been set empirically, after an evaluation performed on some training videos with the camera placed at 180 cm from the ground.

In addition to these constraints, we remove high overlapping detection windows resulting from the consistency module. In particular, we remove those windows which overlap more than 80 % with another one. This operation allows to considerably reduce the number of detection windows which have to be analyzed by the following stages of the detection cascade, thus saving computational time and reducing the false positive rate. Later on in the paper, we will use the *Cons* abbreviation for the consistency module, while its version which removes highly overlapping detection windows will be called *CROW (Consistency with Removal of Overlapping Windows)*. In Fig. 3, the output of the *Cons* and *CROW* modules can be compared for an image of the *ROS-Industrial People Dataset*.

5.3 Color-Based Algorithms

Haar-like Features Extraction on Color Images We utilize Haar-like features in the same manner as [18] but with the key differences proposed in [5].

These features are used with two classifiers. The first of these is a preliminary AdaBoost classifier which utilizes the Haar features and simple decision trees as the weak classifiers. The classifier is trained with preliminary weights which favor correct detections over limiting the false positive rate, the purpose being to eliminate windows which are obviously not persons. The preliminary AdaBoost classifier used in our final implementation has a detection rate of about 97 % and a false positive rate of about 5 % on the *INRIA Person Dataset* [8].

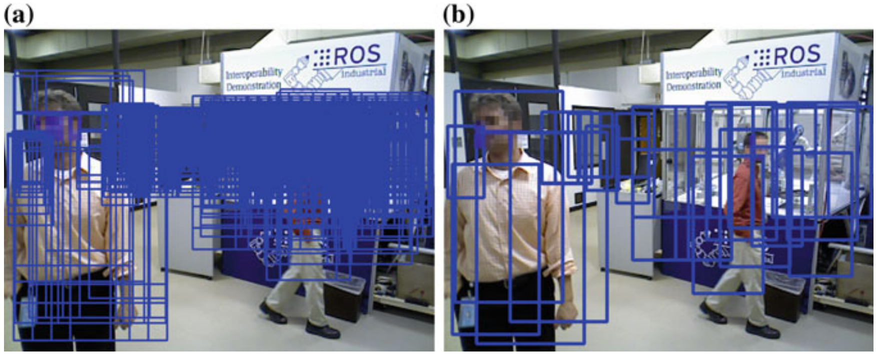

Fig. 3 Effect of removing highly overlapping windows in the consistency module. **a** Cons, **b** CROW

We also tried these Haar-like features with a Support Vector Machine (SVM) classifier. Using SVM is advantageous because it allows to obtain the same performance obtained by Viola and Jones [24] with Adaboost, but the training process is faster.

We will refer to the approach using Haar features extracted on the color image and the Adaboost classifier as *HaarAda*, while we will call *HaarSvm* the method which uses the Support Vector Machine.

HOG-like Descriptors We implemented two methods exploiting two different versions of the HOG descriptor which is then used together with a Support Vector Machine classifier. The first method (called *HogSvm*), computes HOG features only in chosen parts of the detection window, where the outside edges of the person should be. This method is motivated by the attempt to extract only relevant information and discard gradients on the background.

In this work, we also use a HOG implementation which extracts features from the whole detection window, thus being similar to the original version of Dalal and Triggs [8]. This second version is also optimized to use SSE2 instructions, which allow to speed up descriptor computation up to four times. Since this algorithm has been contributed to the *Point Cloud Library* [20], we will refer to this method as *HogSvmPCL*. We trained the Support Vector Machine classifier in two different ways: on whole person bodies (*HogSvmPCLWholebody*) and on half person bodies containing only persons' head and torso (*HogSvmPCLHalfbody*). The latter mode is computationally faster because the descriptor size is a half and it should behave better when only the upper part of a person is visible.

The HOG-based classifiers, though expensive in terms of both feature computation and classification, serve as a final verifier and are usually placed at the end of the detection cascade, when only a small number of detection windows remain.

5.4 Depth-Based Algorithms

Depth-Based Clustering Similarly to the consistency method proposed in Sect. 5.2, this module is intended to act as the first stage of a detection cascade in order to find a small number of detection windows which are then better analyzed by the following stages of the cascade. However, unlike the consistency method, this algorithm requires a point cloud of the scene as input, instead of a disparity image. Given the point cloud, this method removes the ground plane points and performs a clustering of the remaining points based on their Euclidean distance. The obtained clusters are then subclustered in order to center them on peaks of a height map which contains the distance of the points from the ground plane. These peaks are intended to represent the position of people's head. The abbreviated name for this method will be *DC (Depth Clustering)*.

Haar-like Features Extraction on Disparity Images With some slight modifications, the same preliminary Haar-like features classifier presented in Sect. 5.3 can be used to locate people in disparity images. The key difference in implementation is that the scaling for computing the features must take pixels with unknown disparity into account. In our system, pixels for which disparity cannot be determined are assigned a value of zero. We apply a special scaling function in computing the Haar-like feature maps which does not include zero pixels in its area averages. The abbreviated name for this method will be *HaarDispAda*.

It is worth noting that the features from the disparity map are quite independent from the features of the color image. Thus, the combination of the disparity map classifier and preliminary Haar-like features classifier leads to very high detection rates and low false positive rates.

6 Tracking Algorithm

The tracking algorithm we propose exploits the output of the people detection cascade in order to perform data association between existing tracks and new detections. This matching is based on computing color histograms of persons from the color image. If a detection is not associated with any existing track, a new track is initialized in a new thread. This thread is dedicated to a particle filter in the x-y plane which exploits detections and the corresponding color histograms to search for the target person within the image. Such an approach allows to detect people even when the detection cascade returns no valid output. Moreover, the use of a particle filter allows to recover from errors of the tracker, such as identity switches. In this data association scheme, multiple detections can be associated to the same track, in order to avoid multiple tracks generated from the same person. It should be noted that this algorithm requires a separated thread for every track, thus the number of tracks which can be handled is limited by the multi-threading capabilities of the computer.

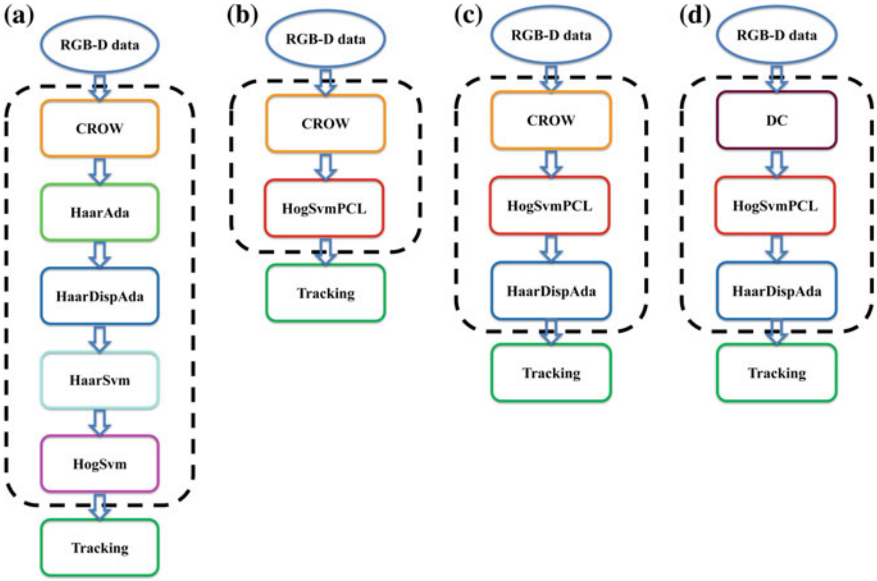

Fig. 4 People detection and tracking pipelines

7 Detection and Tracking Pipelines

In this work, we compare different pipelines for people detection and tracking. These pipelines differ in the detection cascade, which is composed by a combination of the detection modules described in Sect. 5. In Fig. 4, we report the main pipelines we tested. It should be noted that the more are the stages of the detection cascade and higher is the system latency, thus a shorter cascade is preferable. However, a too short cascade could result to be inefficient if the single modules are not fast enough. As we mentioned in Sect. 5, we tried both the *CROW* and the *DC* modules at the first stage of the cascade. As we will see in Sect. 8.3, the *HogSvmPCL* module resulted to be the best option for the second stage in terms of both framerate and accuracy.

8 Experiments

In this section, the pipelines proposed in Sect. 7 are evaluated in terms of accuracy and framerate. These pipelines differ in the detection cascade, while the tracking node is the same for all, thus the name chosen to represent them is the description of the detection cascade.

8.1 Tracking Evaluation

We evaluate tracking results in terms of false positives and missed detections. In order to compute these quantitative indices, we compare ground truth and tracking bounding boxes in the image by means of the PASCAL rule usually exploited for evaluating object detectors [11]. Given that tracking results can be considered good even if the computed bounding box is a bit off with respect to the person center, a threshold of 0.3 has been used in the PASCAL rule, instead of the standard 0.5 threshold. Associations between results ROIs and ground truth ROIs are computed with a Global Nearest Neighbor approach, by means of the Hungarian algorithm.[4] Then, a number of quantitative indices are computed. In this work, these two indices have been used:

- *False Rejection Rate* (%) (FRR): 100*miss/(TP + miss) = % miss
- *False Positives Per Frames* (FPPF): FP/frames,

where FP is the number of false positives, TP is the number of true positives and miss is the number of false negatives.

8.2 Results on ROS-Industrial People Dataset

In Fig. 5, the pipelines described in Sect. 7 are compared by means of *Detection Error Trade-Off* (DET) curves reporting FRR in the x-axis and FPPF in the y-axis. These curves have been obtained while performing detection and tracking on the whole *ROS-Industrial People Dataset* and by varying the threshold on the SVM score associated to the HOG descriptor. The ideal working point for these curves is located at the bottom-left corner (with FRR = 0 % and FPPF = 0). For visualization purposes, the curves are reported in logarithmic scale.

It can be noticed that the removal of overlapping windows in the *CROW* module leads to a very small increase in the percentage of missed people. This means that some good detection windows are sometimes removed because highly overlapping with other detection windows. The algorithm classifying Haar features from the disparity image (*HaarDispAda*) proved to be very effective in reducing the number of false positives which are produced by the methods operating on the color image. Moreover, the *HogSvmPCLHalfbody* technique, which is trained on half person bodies, seems to be the essential module for obtaining the minimum number of false positives, while tracking people in about 80 % of the frames. This proved to be true both in combination with the *CROW* and with the *DC* methods at the first stage of the detection cascade. In fact, for the pipeline *CROW* + *HogSvmPCLHalfbody* + *HaarDispAda*, with a threshold of –0.925 on the HOG threshold for people detection, we obtain FRR = 21.95 % and FPPF = 0.17. Some qualitative tracking results obtained with this approach are shown in Fig. 6.

[4] http://en.wikipedia.org/wiki/Hungarian_algorithm.

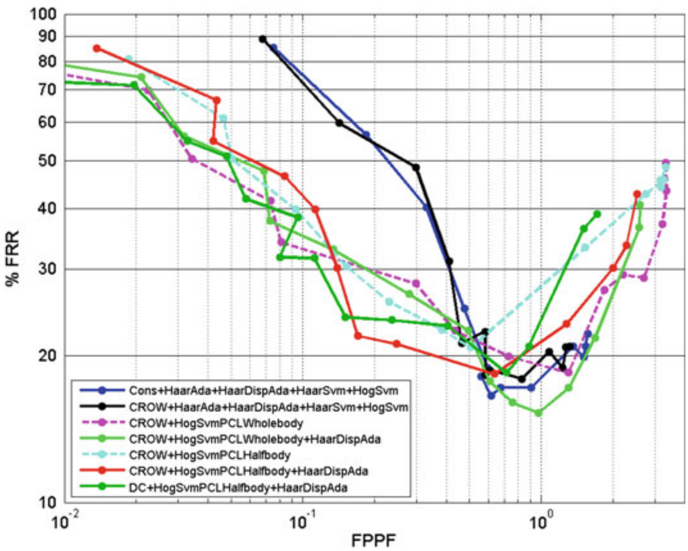

Fig. 5 DET curves comparing the main approaches proposed in this work

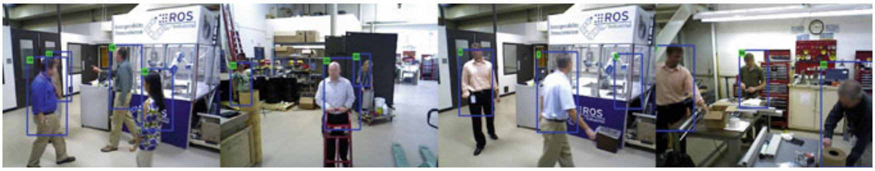

Fig. 6 Tracking results with the *CROW* + *HogSvmPCLHalfbody* + *HaarDispAda* pipeline on some sample images of the *ROS-Industrial People Dataset*

From a qualitative point of view, the detections produced by the cascades which exploit the *DC* module at the first stage are more persistent and well centered on people, thus the people detector obtains superior performance. However, it seems not to make the difference at the tracking level with respect to pipelines based on the *CROW* module because the particle filter is able to interpolate when some detections are missing.

Since our detection methods independently process every frame, we obtained good tracking results also for the part of the dataset where the camera is moving.

8.3 Framerate Analysis

In Fig. 7, the framerates measured for all the tested pipelines are reported. The measurements have been done every second, while performing tracking on the *ROS-*

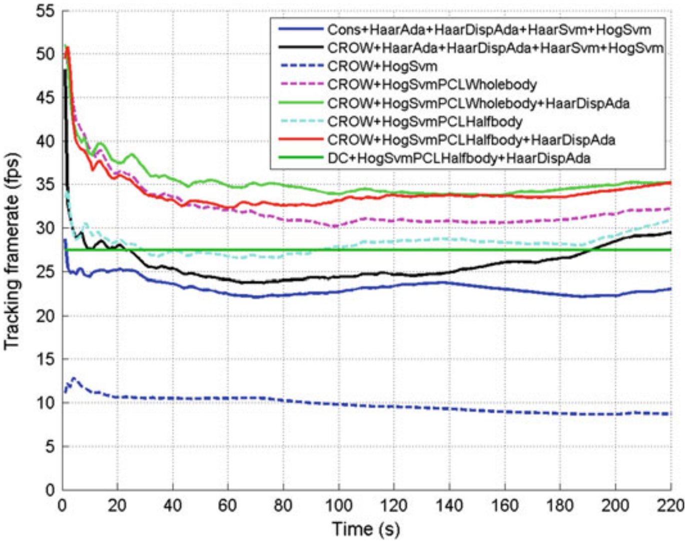

Fig. 7 Framerate measurements for the main approaches which are proposed in this work

Industrial People Dataset. In order to detect the maximum framerate achievable by each pipeline, tracking has been performed when streaming the dataset images at 50 Hz, while the sensor framerate was of 30 Hz. Only for the pipeline using the *DC* approach, we could not use this measurement method because it was not possible to stream point clouds at 50 Hz. Thus, the framerate we report has been measured while processing a live Kinect stream (at 30 Hz).

For comparing our two implementations of the HOG descriptor, we also report the framerate of the *CROW* + *HogSvm* pipeline, which resulted three times slower than what was obtained when using the *HogSvmPCL* module.

All our people detection and tracking algorithms are implemented in C++ and exploit the Robot Operating System (ROS) [19] and open-source libraries for 2D [2] and 3D [20] computer vision.

The detection cascades are composed of ROS nodelets which share the same process, thus avoiding data copying between them and gaining in terms of framerate.

The best approches in terms of framerate are the *CROW* + *HogSvmPCLWhole body* + *HaarDispAda* and *CROW* + *HogSvmPCLHalfbody* + *HaarDispAda*, which track people at framerates between 33 and 35 fps. It is worth noting that these pipelines resulted to be the best also in terms of accuracy.

9 Conclusions

In this work, we proposed algorithms for detecting and tracking people from aligned color and depth data in industrial environments. Several pipelines have been considered and evaluated on a dataset featured by levels of clutter and occlusion typical of an industrial setting. Our detection approach combines depth-based and color-based techniques in a cascade and our tracking algorithm allows to track people even if detections are missing for many frames and to recover from identity switches. The best pipelines we proposed allow to track people for about 80 % of the frames with a very low number of false tracks and at a framerate higher than 30 fps.

More than one pipeline showed very good performance and all the codes have been publically released in ROS-Industrial Human Tracker repository,[5] so that a wide choice of algorithms is available to users and developers.

Even if the obtained results are good, safety measures impose that no person should be missed in industrial applications. Thus, as a future work, we envision to further improve the detection of people while retaining the same false positive rate. For this purpose, we will add a validation phase within the tracking algorithm, so that false tracks could be immediately deleted if the detection confidence does not pass a proper multi-frame check. Moreover, we will work on making the tracking algorithm more scalable in the number of tracked people.

Acknowledgments We would like to thank Open Perception Inc., the Southwest Research Institute (SwRI) and the National Institute of Standards and Technology (NIST) for funding this research and Radu Bogdan Rusu for the fruitful discussions on this topic.

References

1. F. Basso, M. Munaro, S. Michieletto, E. Pagello, and E. Menegatti. Fast and robust multi-people tracking from rgb-d data for a mobile robot. In *12th Intelligent Autonomous Systems Conference (IAS-12)*, pages 265–276, Jeju Island, Korea, June 2012.
2. G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000.
3. Michael D. Breitenstein, Fabian Reichlin, Bastian Leibe, Esther Koller-Meier, and Luc Van Gool. Robust tracking-by-detection using a detector confidence particle filter. In *International Conference on Computer Vision (ICCV) 2009*, volume 1, pages 1515–1522, October 2009.
4. Alexander Carballo, Akihisa Ohya, and Shin'ichi Yuta. Reliable people detection using range and intensity data from multiple layers of laser range finders on a mobile robot. *International Journal of Social Robotics*, 3(2):167–186, 2011.
5. David R. Chambers, Clay Flannigan, and Benjamin Wheeler. High-accuracy real-time pedestrian detection system using 2d and 3d features. *Proc. SPIE*, 8384:83840G–83840G-11, 2012.
6. W. Choi, C. Pantofaru, and S. Savarese. Detecting and tracking people using an rgb-d camera via multiple detector fusion. In *International Conference on Computer Vision (ICCV) Workshops 2011*, pages 1076–1083, 2011.
7. W. Choi, C. Pantofaru, and S. Savarese. A general framework for tracking multiple people from a moving camera. *Pattern Analysis and Machine Intelligence (PAMI)*, 35(7):1577–1591, 2012.
8. N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Computer Vision and Pattern Recognition (CVPR) 2005*, volume 1, pages 886–893, June 2005.
9. A. Ess, B. Leibe, K. Schindler, and L. Van Gool. A mobile vision system for robust multi-person tracking. In *Computer Vision and Pattern Recognition (CVPR) 2008*, pages 1–8, 2008.

[5] At the moment of writing, the source code is in the `develop` branch of the `human_tracker` GitHub repository: https://github.com/ros-industrial/human_tracker/tree/develop.

10. A. Ess, B. Leibe, K. Schindler, and L. Van Gool. Moving obstacle detection in highly dynamic scenes. In *International Conference on Robotics and Automation (ICRA) 2009*, pages 4451–4458, 2009.
11. Mark Everingham, Luc Gool, Christopher K. Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88:303–338, June 2010.
12. Matthias Luber, Luciano Spinello, and Kai O. Arras. People tracking in rgb-d data with on-line boosted target models. In *International Conference On Intelligent Robots and Systems (IROS) 2011*, pages 3844–3849, 2011.
13. D. Mitzel and B. Leibe. Real-time multi-person tracking with detector assisted structure propagation. In *International Conference on Computer Vision (ICCV) Workshops 2011*, pages 974–981. IEEE, 2011.
14. Oscar Mozos, Ryo Kurazume, and Tsutomu Hasegawa. Multi-part people detection using 2d range data. *International Journal of Social Robotics*, 2:31–40, 2010.
15. M. Munaro, F. Basso, and E. Menegatti. Tracking people within groups with rgb-d data. In *Proc. of the International Conference on Intelligent Robots and Systems (IROS)*, pages 2101–2107, Algarve, Portugal, October 2012.
16. M. Munaro and E. Menegatti. Fast rgb-d people tracking for service robots. *Autonomous Robots Journal*, 2014.
17. Luis E. Navarro-Serment, Christoph Mertz, and Martial Hebert. Pedestrian detection and tracking using three-dimensional ladar data. In *The International Journal of Robotics Research, Special Issue on the Seventh International Conference on Field and Service Robots*, pages 103–112, 2009.
18. C. Papageorgiou, T. Evgeniou, and T. Poggio. A trainable pedestrian detection system. In *In Proceedings of IEEE Intelligent Vehicles Symposium '98*, pages 241–246, 1998.
19. Morgan Quigley, Brian Gerkey, Ken Conley, Josh Faust, Tully Foote, Jeremy Leibs, Eric Berger, Rob Wheeler, and Andrew Ng. Ros: an open-source robot operating system. In *International Conference on Robotics and Automation (ICRA)*, 2009.
20. Radu Bogdan Rusu and Steve Cousins. 3D is here: Point Cloud Library (PCL). In *International Conference on Robotics and Automation (ICRA) 2011*, pages 1–4, Shanghai, China, May 9–13 2011.
21. Luciano Spinello and Kai O. Arras. People detection in rgb-d data. In *International Conference On Intelligent Robots and Systems (IROS) 2011*, pages 3838–3843, 2011.
22. Luciano Spinello, Kai O. Arras, Rudolph Triebel, and Roland Siegwart. A layered approach to people detection in 3d range data. In *Conference on Artificial Intelligence AAAI'10*, PGAI Track, Atlanta, USA, 2010.
23. Luciano Spinello, Matthias Luber, and Kai O. Arras. Tracking people in 3d using a bottom-up top-down people detector. In *International Conference on Robotics and Automation (ICRA) 2011*, pages 1304–1310, Shanghai, 2011.
24. Paul Viola and Michael Jones. Rapid object detection using a boosted cascade of simple features. In *Computer Vision and Pattern Recognition (CVPR) 2001*, volume 1, pages 511–518, 2001.
25. Hao Zhang, C. Reardon, and L.E. Parker. Real-time multiple human perception with color-depth cameras on a mobile robot. *IEEE Transactions on Cybernetics - Part B*, 43(5):1429–1441, Oct 2013.