“FOLO”: A Vision-Based Human-Following Robot

Evan Chen[a]

Department of Information Art & Design, Tsinghua University, Beijing 100084, China;

[a]chenzhen15@mails.tsinghua.edu.cn

Keywords: Human-following, Learning Detection, Autonomous navigation.

Abstract. The main challenge in following robot is solving the real-time user localization problem. In this paper we introduce our solution for it together with our whole robot system called “FOLO”. Using extra-tool (such as Bluetooth) may cause inconvenience in interaction; 3D detector plus tracker ID approaches are at risks of computing consume and ROI flash; Approaches based on 2D appearance have challenges on appearance change, re-detecting and complex background. We present a 2D-appearance approach on “FOLO” which can work overcome above issues. Our approach utilizes consensus of corresponding method to track, and then, updates features by supervised learning into classifier cascade to against challenges of re-detecting and complex background. Moreover, we give pre-processing procedure on each frame to sharpen edges on image to enhance quality of tracking and use adaptive background to overcome challenge of complex background. This paper illustrates that our tracking approach can work against common challenges of tracking in our own designed experiments, has a rapid speed over 25 fps and can achieve state-of-the-art results. We use two-layer-PID method to control “FOLO” for a long-term task which allows “FOLO” succeeding to follow user in office environment.

Introduction

Applications of mobile robots become more popular in recent years, but there still exists many challenges on products. We focus on the problems of human-following robot and want to make it be robust in different environment. We believe that a reliable human-following robot can benefit our daily life to be more convenient.

Some applications of robot have human-following function by using extra sensors. [1] utilizes cameras network to help robot locate human’s position, [2] presents a following robot a following robot serve in Wal-Mart Walmart which utilizes sonar to follow user. [3] presents a following robot based on Bluetooth which can carry stuffs of user in golf course. However, extra tools may cause operating problems and make inconvenient for user if control tools lost. Some researchers notice such problems and try to accomplish this function with sensors only on robot.

Human tracking approach is the main issue of human-following robot without user-hold-extra sensors. [4][5] present people tracking method based on range data. However, mobile robot need robustness tracking result of people, thus, researchers try to use vision technology to accomplish human tracking function. RGB-D human detector is the most popular methods on mobile robot platform [7] [8]. Facing with the problem of low FPS, some researchers utilize GPU to enhance the performance of human tracking based on RGB-D detectors[9]. Stereo vision is also a choice for human-following robot [12][13], but calibration parameters, cost of computing disparity map and quality of image influence the performance of its robot application. [7] presents a following robot which uses Hog detector with online learning method to track people and use stereo vision to get distance from human. Researchers also pay efforts on multi-sensor fusion method which combine face detector by robot’s camera and leg detector by laser scan [11]. These works have similar idea is that detecting human and then give them a tracker ID. But the detecting method has an inevitable problem is that the ROI of object is not stable which is fateful for robot to locate user when it is following people.

Thus, we want to try a novel approach. We utilize 2D appearance tracking method instead of these detecting human plus tracker ID approaches in this work. We want to give our tracked object a

Copyright © 2018, the Authors. Published by Atlantis Press.
This is an open access article under the CC BY-NC license (http://creativecommons.org/licenses/by-nc/4.0/).

gradual change process whose purpose is to avoid sharply flash of object position which is common in above approaches. A survey of 2D appearance tracking methods is done in [15]. The main risks of tracking human on a mobile robot based on 2D appearance are large-appearance change, scale change, background change, re-detecting problem and robust bounding box got from tracking. To solve such problem, we utilized tracking, learning and detecting framework [16]. Our tracking module uses the idea clustering of corresponding method which is firstly introduce in [17]. We let our learning module update changed appearance of human into positive samples and changed background feature into negative samples frame by frame. The detecting module is used to check the quality of tracking module and accomplishing re-detecting function. With the learning module updating positive and negative samples, detecting module is also benefitted by it. Re-detecting function which uses the detector trained by learning module can re-detect human by recently changed appearance if occlusion happens. Furthermore, as our robot needing to run on a mobile platform, we add a pre-processing procedure which includes an adaptive background segment according user position change and sharpen edge procedure to enhance tracking quality.

And then, once tracking approach output a position (include distance and angle) to our controlling module. The method we used here is PID controller (proportion, integral, derivative framework). The movement layer PID controller is run to estimate the speed and rotation of robot needing to follow user by compare distance and angle from robot and user. And then, the inner layer is operated to let electricity motor of wheels to smoothly launch suitable rotate speeding according to the distance an angle estimated by movement layer PID controller. On “FOLO”, two-layer PID get an ideal quality which allows robot to keep ideal relative position between robot and user, and it best fit requirement of controlling in our system.

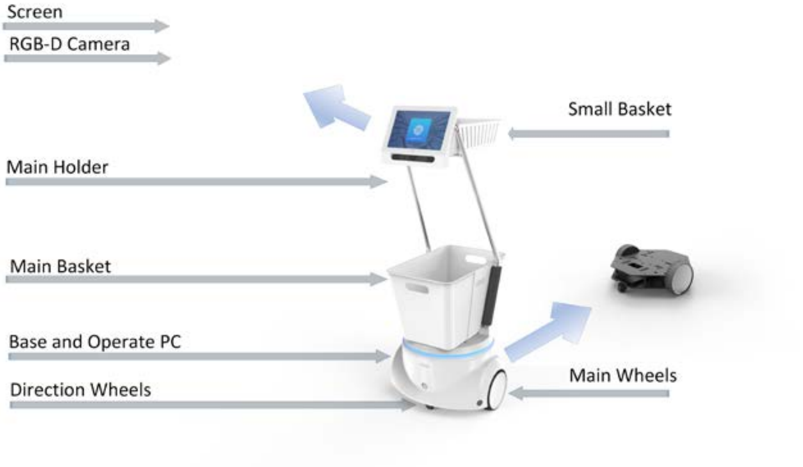

Fig. 1 Appearance of “FOLO”

Our robot “FOLO” uses Intel NUC (I3-5010U) as operating PC, Prime-sense as only input sensor. As all above steps are individual but needing inner transport among each step, we utilize ROS as our robot system which can transfer data via topic. Our approach runs on single core CPU and without GPU. Depth information of prime sense is just used to output distance from human to robot and do a background segment. As our approach is based on 2D appearance, duration of tracking human is greatly shorter than other 3D detector based approaches. Furthermore, the approach is greatly robust in human tracking even human are turning around. Controlling system works stably which allows our robot can follow user in a long-term task.

System Description

Overview. The appearance of "FOLO" is shown in Fig. 1. On the top our robot is a touch screen which is used to interact with user. RGB-D camera is installed under the scene in order to get a higher view and a basket is installed behind the screen to carry small stuffs. In this kind of top design, RGB-D camera can suitable height where can let it have a suitable view of user and screen can have interaction with user. Under the two holders, there is a large basket to carry large stuffs. On this height, robot can keep center of gravity on a low height which can avoid rollover when it is turning. Two driving wheels and two steering wheels are designed on batholith, and they allows it have suitable capacity of movement.

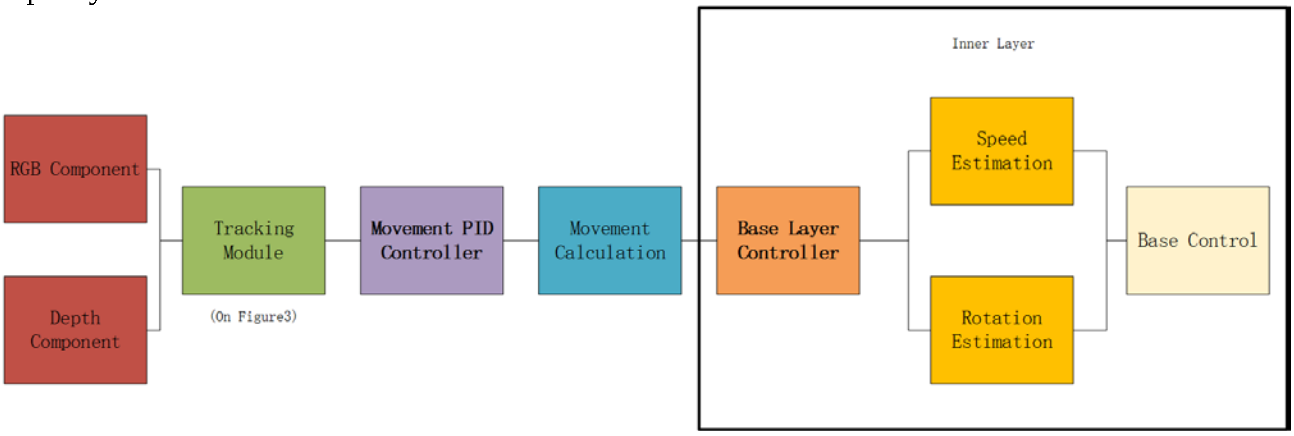

Fig. 2 Framework of “FOLO”

Framework on Fig. 2 shows the overview of our system. We utilize platform on WeChat[23] to accomplish interaction function between “FOLO” and user. User should stand in front of “FOLO” and send command to let “FOLO” initialize. Initialization procedure will keep continue until upperbody detecting is valid. And then, once upperbody detecting is valid, tracking module will initialize detector and feature samples according position of user detected by upperbody detecting. During tracking procedure, pre-processing run to each income frame in order to filter influence of challenges on tracking. And then, tracking approach reports a ROI and update changed information from environment frame by frame. On the other hand, re-detecting module is run when occlusion happen. It utilizes detector to slide among image and get valid position of user by detector sliding window. Controlling approach subscribes the ROI of user and get the distance and angle between user and “FOLO” via depth map. It counts motion of “FOLO” by Movement PID controller and computes rotation and speed on motor by Base layer PID. The final step is returning module. The returning is also based on vision technology. Firstly, it subscribes RGB-D frame from system, and then, inputs it into our trained random forests to get a pose and position of robot. However, this pose and position is not very precise, therefore, we use Monte Carlo Localization do to a local localization by turning around robot which can get an accurate pose and location of robot. Due to we are not satisfied with returning module (it is not robust in changeable environment), we propose to publish this approach after we improve it in the coming future.

Tracking Approach. First of all, tracking approach needs to be initialized and should be given an initial position of human. We utilize upperbody detector trained by SVM [22] to scan the image to get user's initial position on the scene at the first frame. Due to low cost and limit need for training samples, SVM can rapidly output the position of human. And then, we save feature points inside ROI as foreground points (FP) and outside ROI as background points (BP). Meanwhile, we also construct a detector based on PN learning by inputting texture information inside ROI as positive sample (PS) and outside it as negative sample (NS). The purpose of using FP and BP is to initialize the feature points which is used to vote, match and estimation during tracking procedure. Detector is used to check quality of tracking, and it will be updated (learning from renewed PS and NS) if check result is valid.

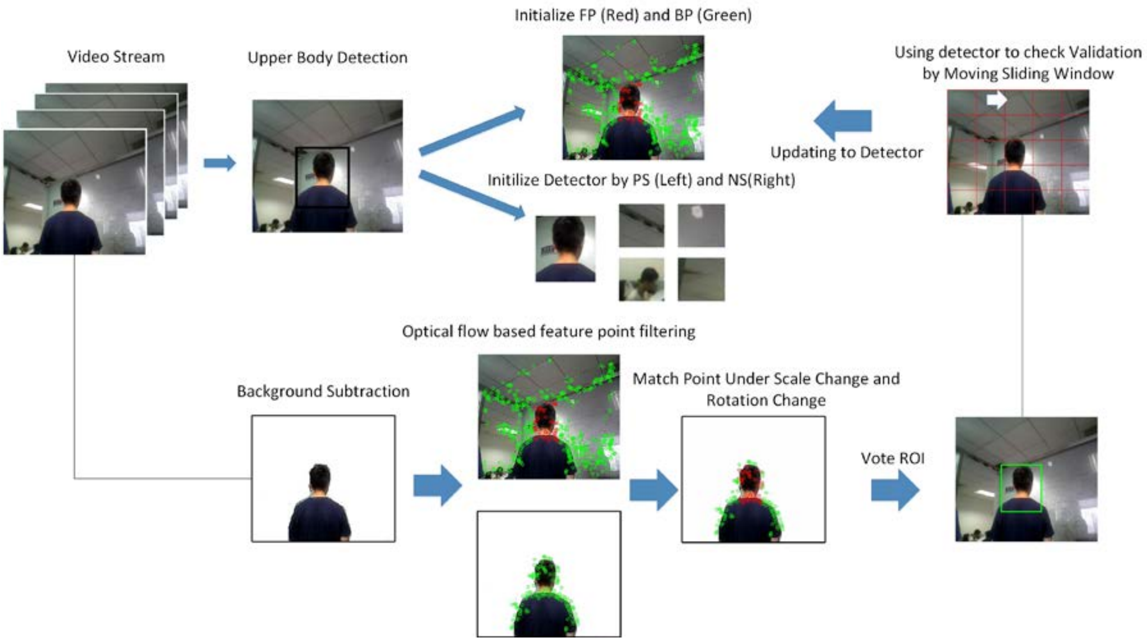
Video Stream
Upper Body Detection
Initialize FP (Red) and BP (Green)
Initilize Detector by PS (Left) and NS(Right)
Using detector to check Validation by Moving Sliding Window
Updating to Detector
Optical flow based feature point filtering
Background Subtraction
Match Point Under Scale Change and Rotation Change
Vote ROI

Fig. 3 Flow chart of tracking approach

After initialization, the tracking starts. Firstly, it utilizes optical-flow to track object's feature points from previous frame to current frame, and then, it filter out feature points by using optical flow to track feature points from current frame to previous frame. Secondly, by matching FP with feature points (both FP and BP) on image globally to get rough correspondences of feature points, with rotation and scale estimation, it votes for the most alike position of object center. And then it does a local matching to filter some background and disappeared points and then publishing it in tracking approach. To against the appearance change, we update the feature points to FP after successfully voting and filtered feature points to BP. While it is still not reliable if we do it just based on feature points. We utilize the texture feature of detector trained initialization step to check quality of tracking, and then, updating texture in bounding box as positive sample and that outside it as negative sample if it pass our check procedure. It will also have a re-detecting step if lost. Detector is used to check human frame by frame. If it passes the re-detecting procedure, FP will be updated by changed appearance feature and our robot turn into tracking mode again.

```
Tracking Algorithm
Input: $I_1 \ldots I_n$
Output: $O_1 \ldots O_n$
If (Upperbody_detect($I_n$) == valid) then
    $O_n$ ← Upperbody_detect($I_n$)
    $P$ ← detect($I_n$)
    $FP, BP, PS, NS$ ← Initilize ($I_n, O_n, P$)
    $Detector$ ← detector_initilize($PS, NS$)
End if
While (1)
    $P$ ← detect($I_n$)
    $M$ ← match ($P$, $O_{n-1}$)
    $T$ ← track ($I_{n-1}$, $I_n$)
    $K'$ ← $T \cup M$
    s ← estimate_scale (K',O)
    α ← estimate_rotation (K', O)
    V ← vote (K', O, s,α)
    $V^c$ ← consensus (V)
    $K_t$ ← $vote^{-1}(V^c)$
    if $|V^c| >= \theta * N^o$ then
        μ ← $\frac{1}{n}\sum_{i=t}^{n} V_i^c$
        $B_u$ ← bouding_box($b_1$, μ, s, α)
        If (check_valid ($B_u$, D)  ==  true) then
            $FP, BP, PS, NS$ ← update ($P, O_n$)
        Else
            $O_n$ ← ∅
        End if
    End if
```

Fig. 4 Pseudo-Code of tracking approach

Pre-processing and Details. Feature points is detected by GFTT and the descriptor of feature points is BRISK which allows our approach does not take too much time on detecting feature points. However, quality of GFTT is not reliable although its speed is rapid. Therefore, we try to utilize pre-processing method to enhance quality of feature detecting. In this work, we use sharpen approach Laplace operator to enhance the edge of human. In application, it is done by a 3X3 kernel (center is 8 and others are 1) and sharpen edge by moving sliding window. The processing time of sharpen is less than 10 ms, but the quality of tracking sharply increase. Although noise on image also increases due to Laplace operator making some new extreme points, the result of tracking is still greatly better than that not using it. The advantage overweight the disadvantage. Thus, pre-processing is run to each incoming frame to guarantee the quality of tracking approach. We compare the result of matching in tracking module using or not using this step and find that the successful matches and feature points increase 10 times in average in 50 sequence during user are turning around but noisy and error matches decrease 3% in average.

Table 1 Compare Sharpen Quality

	Feature Points	Matches	Error Matching
Without Sharpen	137	11	10%
With Sharpen	1314	130	7%

Furthermore, we add a background segment based before tracking approach on “FOLO”. Due to the design of our “FOLO” is to work in shopping mall, the background is greatly complex in such environment. Although our learning strategy can greatly improve the robustness of our tracking approach, background segment still can slightly improve the quality of tracking. Our background segment is based on depth information. Firstly, it aligns RGB image and depth image according to calibration data. Then, we find the pixels on depth image where distance is over 1.5m (we set this distance as MD) and lock such pixels during initialization step. On RGB image, we will filter pixels where is larger than 1.5m aligning to depth image. After initialization, we design our approach to estimate the MD adaptively according to results of tracking approach. We collect results of scale change (SC) and distance from user to “FOLO” each frame calculated by ROI center area aligning to depth map each frame, and then, estimate MD in current frame. Moreover, due to depth image generally having too many holes, this situation may influence quality on RGB image. Thus we add a filling-holes step on depth map before aligning to RGB image. It can be divided into two steps, firstly, it moves sliding window all over the depth map to find if invalid pixels are surrounding by valid pixels to reduce the areas of holes. After that, we dilate the depth map to cover these holes and erode depth map to clear cover dilated area.

Controlling Strategy. According to the output ROI of tracking approach, we calculate the angle and distance via information on depth image. As our quality of tracking is promising, we calculate the distance of object by using median value of all pixels inside the bounding box. The angle is counted by the relationship of center point of ROI and center point of image.

We publish the two values as a topic onto system of “FOLO”, and our controlling approach will subscribe this topic. The controlling approach has two layers of PID, outer layer collect the values of distance and angles from robot to user per frame. It estimate the need speed and angle of robot by comparing target distance and observed value. And then, the two values will be transported into inner layer as target value which will be compared them with the current rotate speed. The purpose of it is to control the electricity motor on our robot to smoothly reach the speed robot need.

Experiment

We evaluate our system and its tracking approach in four different scenarios with various settings for justifying their effectiveness. The first two scenarios covers various cases in an office environment, with the first one focusing on static scenes, for which robustness w.r.t. scale (distance) change, appearance change and occlusions are tested, and the second one covering dynamic scenes, where challenges such as changing background, motion blur, and illumination variations are concerned. The third scenario is real shopping environment, for which we let "FOLO" follow a customer and help carry items in a large shopping mall, and quantitatively evaluate the results by both the relative orientation angle and distance from “FOLO” and served customer. Finally, the last scenario is testing the tracking approach on public datasets in comparison with state-of-the-art competitors, for justifying the superiority and generalization ability of our tracking approach. Our experiment uses NUC I3 5010U 4G DDR31600MHz without individual GPU, sensor is prime sense and use own designed base to operate the robot.

The first item is occlusion test. As “FOLO” is designed to work in a real environment where maybe quite crowded, problems of overlapped and part-overlapped by other people or other obstacles may frequently happen. In this item, we let user to walk in front of “FOLO” and let “FOLO” to operate in following mode after initialization of tracking. And then, we let another developer to walk across the scene to simulate overlapping situation.

We can find that our tracking approach can stop tracking and moving as expected when large occlusion happens, the robot stop moving when other developer walk across the scene. Another situation is part-overlapped, we designed our “FOLO” to turn back to avoid crash if part-overlap happens. After occlusion, our tracking approach can re-detect user rapidly. In the end of this item, we ask them to stand together. Our tracking approach can re-detect right user rapidly which proves the success of re-detecting function of our approach.

The second item is scales change test. Although we propose to let “FOLO” keep 1.5m from user to robot, the real situation may be complex and may have some unexpected situation that distance from user to “FOLO” is too large, so we need the tracking to have some robustness to scale changes. In this item, we close the controlling module of "FOLO" and ask user to dolly move.

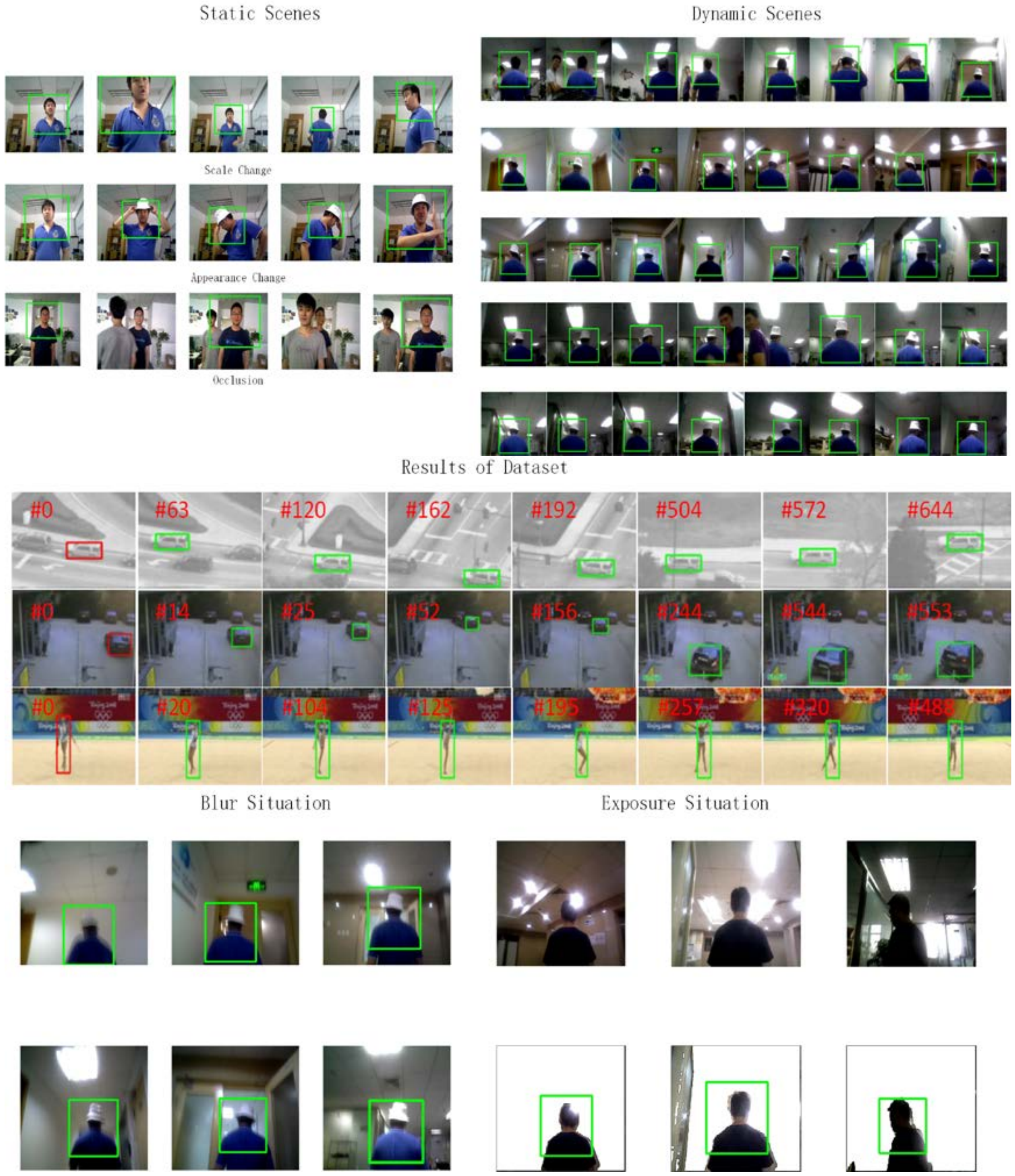

Fig. 5 Experiments of tracking

We can find that our tracking approach can robustly track human in different scales and report right position of user. We let user walk in different distances (from 0.8m to 4m) and play different poses which may happen in real use of "FOLO". The result shows that our tracking approach can keep tracking stably against scale change caused by user's movement. We also manually covered the camera to stop tracking in order to test quality of re-detecting in different scales, result shows that our tracking approach is not suspected by the distance and scale, which shows that our learning function successfully run to update appearance of user.

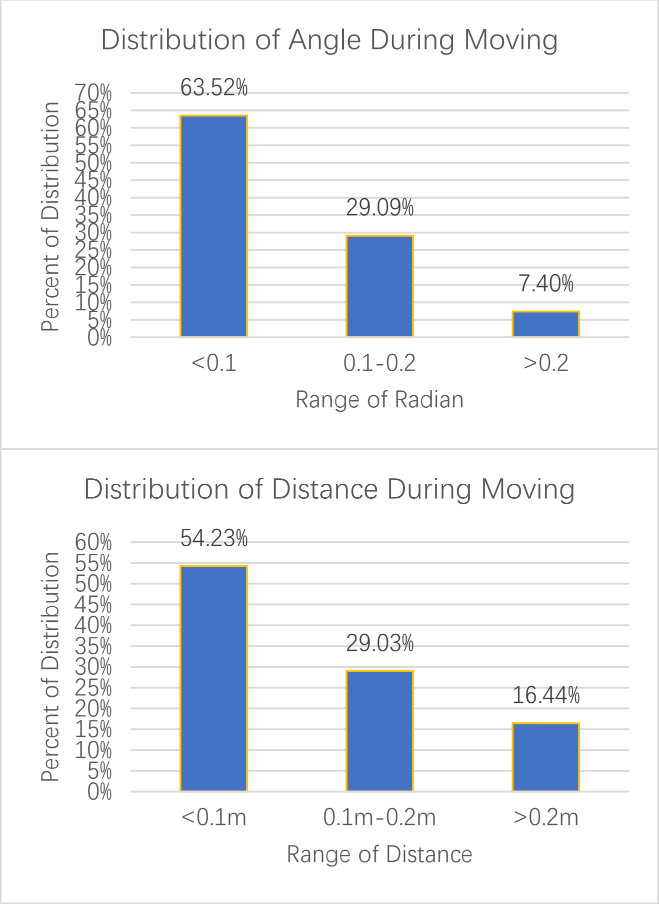

Fig. 6 Angle and Distance Distribution during one travel following

The third item is large appearance change. We let user to wear a white hat during the tracking, and then, ask him to play different poses with large appearance change. Result shows that our tracking approach can robustly run against appearance change even user play poses greatly different from image in initialization. It proves learning module of our approach can update features ideally.

The above tests are done when our robot doesn't operate. While real operating environment maybe more complex, so we test our robot in our office. The image sequence shows a travel of our robot goes from our hardware room to swimming pool and turning back. During the period of travel, "FOLO" works against influence of complex background and the illumination change in different rooms. We also let others test experiments in figure in dynamic situation. "FOLO" can accept appearance change of user when he wear a hat and learn such feature, consequently, it can recognize user in image after occlusion. We also let user speed up and brake during following, "FOLO" can keep tracking on user even scale greatly change.

In table 2, we also compare our tracking approach (without pre-processing) with state-of-the-art tracking approaches [16][17][24][25][26] in common datasets which includes challenges on heavy occlusion, appearance change, scale change and complex background which are common during tracking. The performance is evaluated by F-measure. Our method achieves the best performance on sequences of car, motocross, drunk and surfer. Our approach get best f-measure score 0.90 in average, and which is 0.18 higher than the second one STR. It demonstrates that our tracking approach can get a remarkable performance in common cases during tracking. The reasons for our approach not getting the highest score in gym and person are the out-of-plane rotation. These two sequences have very large out-of-plane rotating on object due to objects turning too fast, consequently, our tracking approach fails in matching among sequences and check quality by detector.

Table 2 Compare with State-of-Art Approaches

	STR	CT	STC	TLD	CMT	Ours
car	0.68	0.10	0.08	0.90	0.93	0.96
drunk2	0.66	0.66	0.29	0.41	0.65	0.96
gym	1.00	1.00	0.57	0.50	0.80	0.94
surfer	0.46	0.01	0.24	0.36	0.43	0.99
person	0.83	0.01	0.26	0.60	0.57	0.67
Mean f-measure	0.72	0.37	0.31	0.58	0.68	0.90
FPS	29	80	280	38	27	25

On Fig. 5, we also present the risks of our tracking approach. The exposure from light make our approach very difficult to accomplish tracking function. The result shows our pre-processing can overcome challenge of extreme situation from illumination.

Apart from experiment of tracking approach, we let “FOLO” to run in a real office environment and track a customer during this travel. “FOLO” is designed to keep 1.0 m and 0 degree angle from user. Data of the angle and distance from user to “FOLO” are recorded in this travel, angle is counted by the offset between image center and ROI center, the distance is counted by ROI position of depth map. The angle from user to “FOLO” generally keep lower than 0.1 radian with about to 65% in this sequence. Range 0.1 to 0.2 radian is 25% and Range over 0.2 radian is about 6% but most of such angle happen during user makes a turn. The distribution of distance from user to “FOLO” shows that it mostly keep distance offset lower than 0.1m (about 55%) but may be fluctuate when user make a turn. It shows that our control approach can accomplish following function according to result of tracking approach.

Conclusion

We have proved that our tracking approach achieves state-of-art performance compared with state-of-art tracking approaches and solves challenges on appearance change, re-detecting after lost and complex background which allows robot “FOLO” can accomplish long-term tracking task in office over 30 minutes and around 2 km without error. Our approach also avoids ROI flash happen in 3D detector’s approaches. As well, pre-processing also enhances the quality of following in extreme situation of indoor environment. Our approach has a rapid speed that over 25 FPS on mobile platform. The tracking approach allows our robot “FOLO” to successfully follow user in a long-term tracking in shopping mall.

References

[1] K. Morioka, J. H. Lee, and H. Hashimoto. "Human-following mobile robot in a distributed intelligent sensor network." IEEE Transactions on Industrial Electronics 51.1(2004):229-237.

[2] http://5elementsrobotics.com/

[3] http://www.caddytrek.com/

[4] L. Spinello, et al. "A Layered Approach to People Detection in 3D Range Data. " Twenty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2010, Atlanta, Georgia, Usa, July 2010.

[5] L. Spinello, M. Luber, and K. O. Arras. "Tracking people in 3D using a bottom-up top-down detector." Icra 47.10(2011):1304-1310.

[6] Arras, K. O., O. M. Mozos, and W. Burgard. "Using Boosted Features for the Detection of People in 2D Range Data." IEEE International Conference on Robotics & Automation 2007:3402-3407.

[7] L. Spinello, and O. A. Kai. "People detection in RGB-D data." 38.2(2011):3838-3843..

[8] Q. M. Do, and C. H. Lin. "Embedded human-following mobile-robot with an RGB-D camera." Iapr International Conference on Machine Vision Applications IEEE, 2015.

[9] W. G. Choi, C. Pantofaru, and S. Savarese. "Detecting and tracking people using an RGB-D camera via multiple detector fusion." IEEE International Conference on Computer Vision Workshops IEEE, 2011:1076-1083.

[10] O. H. Jafari., D. Mitzel, and B. Leibe. "Real-time RGB-D based people detection and tracking for mobile robots and head-worn cameras." IEEE International Conference on Robotics & Automation 2014:5636-5643..

[11] N. Bellotto and H. Hu. "Multisensor-based human detection and tracking for mobile service robots. " IEEE Transactions on Systems Man & Cybernetics Part B Cybernetics A Publication of the IEEE Systems Man & Cybernetics Society 39.1(2009):167-81.

[12] M. Jotsev, et al. "Robust Stereo-Based Person Detection and Tracking for a Person Following Robot." Traffic Management and Road Safety. Proceedings of Seminar K Held At Ptrc European Transport Forum, Brunel University, 1-5 September 1997. Volume P419 P 419(2009):PáGs. 58-60.

[13] J. Satake, and J. Miura. "Multiple-Person Tracking for a Mobile Robot using Stereo." MVA (2009).

[14] www.primesense.com

[15] X. Li, et al. "A survey of appearance models in visual object tracking." Acm Transactions on Intelligent Systems & Technology 4.4(2013):478-488

[16] Z. Kalal, K. Mikolajczyk, and J. Matas, "Tracking-Learning-Detection." IEEE Transactions on Pattern Analysis & Machine Intelligence 34.7(2012):1409-1422

[17] G. Nebehay, and R. Pflugfelder. "Clustering of static-adaptive correspondences for deformable object tracking." Computer Vision and Pattern Recognition 2015:2784-2791.J. B. Shi, and C. Tomasi. "Good features to track." volume 84.9(1994):593-600.

[18] J. B. Shi, and C. Tomasi. "Good features to track." volume 84.9(1994):593-600.

[19] Y. Tabe, et al. Person Following Robot with Vision-based and Sensor Fusion Tracking Algorithm. Computer Vision. 2008:75–93.

[20] J. Brookshire. "Person Following Using Histograms of Oriented Gradients." International Journal of Social Robotics 2.2(2010):137-146.

[21] http://www.ros.org/

[22] S. Netherlands. Support Vector Machine (SVM). Encyclopedia of Genetics, Genomics, Proteomics and Informatics. Springer Netherlands, 2008:1901-1901.

[23] http://weixin.qq.com/

[24] K. H. Zhang, L. Zhang, and M. H. Yang. "Real-Time Compressive Tracking." European Conference on Computer Vision Springer-Verlag, 2012:864-877.

[25] S. Hare, et al. "Struck: Structured Output Tracking with Kernels. " IEEE Trans Pattern Anal Mach Intell, 2011:263-270.

[26] Zhang, Kaihua, et al. Fast Visual Tracking via Dense Spatio-temporal Context Learning. Computer Vision – ECCV 2014. 2014:127-141.