A performance analysis of invariant feature descriptors in eye tracking based human robot collaboration

A Performance Analysis of Invariant Feature Descriptors in Eye Tracking based Human Robot Collaboration

Lei Shi[1], Cosmin Copot[1,2], Stijn Derammelaere[2], Steve Vanlanduit[1]
Op3Mech[1], University of Antwerp
CoSys-Lab[2], University of Antwerp
Antwerp, Belgium
e-mail: lei.shi@uantwerpen.be

***Abstract*—For eye tracking applications in Human Robot Collaboration (HRC), it is essential for the robot to be aware of where the human gaze is located in the scene. Using feature detectors and feature descriptors, the human gaze can be projected to the image from which robot could know where a human is looking at. The motion that occurs during the collaboration may affect the performance of the descriptor. In this paper, we analyse the performance of SIFT, SURF, AKAZE, BRISK and ORB feature descriptor in a real scene for eye tracking in HRC where different variances co-exist. We use a robotic arm and two cameras to test the descriptors instead of directly testing on eye tracking glasses in order that different accelerations can be tested quantitatively. Results show that BRISK, AKAZE and SURF are more favourable considering accuracy, stability and computation time.**

***Keywords*—feature descriptor, eye tracking, HRC**

I. Introduction

In computer vision, feature detectors and descriptors are used to find interesting points in two images and match the detected features to establish correspondence. Numerous feature descriptors have been proposed in the past. SIFT (Scale Invariant Feature Transform) [1] [2] and SURF (Speed Up Robust Feature) [3] are among the most well known robust feature descriptors. More recent work includes ORB (Oriented FAST and Rotated BRIEF) [4], AKAZE (Accelerated-KAZE) [5], BRISK (Binary Robust Invariant Scalable Keypoints) [6] and FREAK (Fast Retina KeyPoint) [7]. Various applications of detectors and descriptors include tracking [8] [9], SLAM (Simultaneous localisation and mapping) [10] and eye tracking for Human-Robot Collaboration (HRC). In [11], FREAK is used to project gaze into a stationary coordinate system in robotic eye-hand coordination. In [12], the authors use FAST (Features from Accelerated Segment Test) [13] and HOG (Histograms of Oriented Gradients) [14] to compensate the effects of head movement with respect to gaze in a Human-Robot shared manipulation.

Eye tracking glasses provides the human gaze information and scene image, where human is looking. In lots of cases robots also acquire visual information from a camera to perform tasks like pick and place. In an eye tracking based HRC scenario, it is also important for the robot to know where human is looking at in the world. Using feature descriptor, the gaze point can be projected from the image acquired from eye tracking scene camera to the image acquired from the robot camera. Therefore robot could know the human gaze in the world. To use feature descriptor for eye tracking based HRC, invariance is an important factor. During the collaboration, a human may move head or body while wearing eye tracking glasses which means the image from eye tracking glasses will have scale, rotation and viewpoint changes. The variances also exist between the eye tracking glasses and the robot camera. Furthermore, also due to the human movement, a fairly high accuracy of feature descriptor under motion becomes essential. And the computation speed of the feature descriptor is certainly another factor to consider, especially for real time requirement.

Several evaluation works on descriptors have been carried out. In [15], authors compared feature descriptors on a dataset contains images with artificially introduced variances and motion blur. They concluded that GLOH (Gradient Location and Orientation Histogram) and SIFT outperforms other tested descriptors in their experiments. Some more recent descriptors, BRISK, BRIEF (Binary Robust Independent Elementary Features) [16] and ORB are analysed in [17] with same dataset. BRISK has better performance when rotation and scale changes both exist. BRIEF has slightly more desirable performance. Also in [18], the authors compared eight descriptors with different scale, rotation, viewpoint and illumination conditions. BRISK and FREAK perform almost equally well for scale invariance. SIFT and ORB have the best results for rotation invariance test. SIFT, SURF, FREAK and BRISK have similar results for viewpoint changing test. All of them perform better when the viewpoint change is small. The performance decreases with the increasement of viewpoint difference.

In real world cases, the complexity of the scene is seriously increased. All the variances mentioned before may happen simultaneously as indicated in Fig.1. Although a lot of evaluation work has been done previously, it is still tricky to simply select one ”best” descriptor for real world application based on the evaluation work on datasets. The comparison work of descriptors specifically for eye tracking based HRC is also less explored. Thus in this paper, we evaluate and analyse the performance of SIFT, SURF, AKAZE, BRISK and ORB during motion in a real scenario where different

types of variances exist at the same time. We use a RGB camera to replace the eye tracking glasses for convenience and we use a robotic arm to carry out experiment with different acceleration. The paper is organised as follows: In section II, a brief description of tested descriptors and how they are applied for eye tracking based HRC is given. Section III describes the setup of experiment and evaluation criteria. In section IV and V, experimental results are discussed.

II. Feature Descriptor

Feature descriptors can be categorised into two categories namely floating point descriptor and binary descriptor. SIFT and SURF are floating point descriptors, AKAZE, ORB and BRISK are binary descriptors. These descriptors and application in eye tracking based HRC are explained briefly in this section.

A. Floating Point Descriptor

SIFT creates a 16×16 kernel region around a keypoint and this region is further divided into 4×4 sub kernels. For each sub kernal, its orientation histogram is calculated. All orientation histograms form a vector to describe the keypoint feature.

SURF is designed to increase the speed of SIFT. A kernel whose size is 20 times larger than the keypoint scale is considered. It is divided into 4×4 sub kernels. In each sub kernel, Haar wavelet response in x direction (dx) and y direction(dy) is calculated and a vector $\mathbf{v} = (\Sigma dx, \Sigma dy, \Sigma|dx|, \Sigma|dy|)$ represents sub kernel. All vectors of sub kernels form the keypoint descriptor.

B. Binary Descriptor

AKAZE is the accelerated version of KAZE [19] feature descriptor. It uses a Modified-Local Difference Binary (M-LDB) descriptor. LDB [20] calculates the average intensity, gradient in x and y direction of same sized sub regions within an image patch and compares them between subregion pairs. For a rotated image patch, the sub regions are also rotated by using the estimated orientation of keypoint in KAZE and sub-sample the area of sub regions for binary test.

ORB makes improvement based on BRIEF descriptor. BRIEF performs intensity binary test on pixels in an image patch. ORB includes the orientation of keypoints to the binary test to make it invariant to rotation.

BRISK uses its unique sampling pattern around a keypoint to calculate the intensity of point pair and the local gradient. Local gradient is used to get rotation information and all the point pairs whose distance is smaller than a threshold form the descriptor.

C. Application in Eye Tracking

Eye tracking glasses are usually equipped with eye camera(s) and a scene camera. Eye camera(s) are used to capture eye images to estimate gaze point and other eye activities(e.g. fixation). Scene camera captures the image of the world. Gaze point is represented as point $g_s = (x, y)$ in the image captured by scene camera I_s. Robot uses another camera to get the location information of objects in a scene. If the gaze point g_s could be projected to the image of robot camera I_r, then the robot will know where a human being is looking at in the world. For the keypoints obtained by feature detector in I_s and I_r, their correspondence keypoints $\mathbf{f}_s$ and $\mathbf{f}_r$ are used to calculate the homography matrix H between I_s and I_r. From a sequence of images $I_s^1 ... I_s^n$ and $I_r^1 ... I_r^n$, the gaze point is projected by

$$g_r^i = H_i g_s^i, 0 < i < n \tag{1}$$

where g_r^i is the projected gaze at ith robot image, g_s^i is gaze point on I_s^i and H_i is ith homography matrix.

III. Experiment Setup and Evaluation

A. Experiment Setup

In order to evaluate the effects of the motion to eye tracking glasses quantitatively, a robotic arm is used instead of directly testing with eye tracking glasses. The motion change can be set via robot controller. For the purpose of convenience, we use a RGB camera to instead of the scene camera of the eye tracking glasses. The scene camera is mounted on the end effector of robot. A second camera (robot camera) observes the same scenario without movement. The setup of the robot and camera is shown in Fig. 2a. The robotic arm makes angular movement around the z-axis for $15°$. The velocity is set to $100°/s$. 3 different accelerations, $50°/s^2$, $100°/s^2$ and $150°/s^2$, are tested with the same angular movement. The angular motion is repeated ten times for each acceleration.

All the descriptors are implemented with OpenCV (Open Source Computer Vision Library). The default feature detectors for descriptors in OpenCV are used. Tuning parameters of detectors and descriptors will lead to different performance, we keep default parameter settings in our experiment. The image sizes are 480×360 and 720×405.

In order to verify the projected gaze point intuitively, we use an object in the scene as reference. As indicated in Fig. 2b and 2c, the green blocks detected are drawn with white bounding box. This could be interpreted as gaze point is located on the block all the time. The projected gaze point g_r thus is replaced by the projected bounding box. In a sequence of the images of the moving scene camera $I_s^1 ... I_s^n$, the bounding boxes are projected to the images of the robot camera $I_r^1 ... I_r^n$ using

$$B_r^i = H_i B_s^i, 0 < i < n \tag{2}$$

where H_i is homography matrix at ith image, B_s^i is the bounding box from ith scene image and B_r^i is the projected bounding box on robot image. The detected bounding boxes B_g^i in robot images are considered as ground truth.

Fig. 1: An example of a sequence of images with viewpoint change during a camera movement. The viewpoint change from left image to right image is $15°$ and acceleration is $100°/s^2$.

(a) Experimental setup. (b) An image taken from the scene camera. (c) An image taken from the robot camera.

Fig. 2: (a): The RGB camera attached on the robot end effector is the replacement of the scene camera of eye tracking glasses. The camera on the right is the robot camera which observes the world. (b) and (c) are the example images taken from the scene camera and robot camera respectively. The detected bounding boxes are drawn in white and the projected one is drawn in red.

B. Evaluation Criteria

To evaluate the accuracy of the projected data, we first calculate the ratio of the overlapping area of the bounding box $\mathbf{B_r}$ and $\mathbf{B_g}$,

$$\mathbf{O} = \frac{\alpha(\mathbf{B}_r) \cap \alpha(\mathbf{B}_g)}{\alpha(\mathbf{B}_r) \cup \alpha(\mathbf{B}_g)} \tag{3}$$

where $\alpha(\mathbf{B})$ is the area of bounding boxes. For a sequence of calculated overlapping ratio $\mathbf{O} = \{O_1 ... O_n\}$,

$$C(O_i, B_s^i, B_r^i) = \begin{cases} \textit{True Positive} & \text{if } O_i > T \\ \textit{False Postive} & \text{if } O_i \leq T \\ \textit{False Negative} & B_s^i \neq 0 \cap B_r^i = 0 \end{cases} \tag{4}$$

where C is classifier function, T is threshold value and $0 < i < n$. T can be set to different values [21], we use 0.6 for out experiment. The F1 score is used to evaluate the projection,

$$\textit{Precision} = \frac{\textit{True Positive}}{\textit{True Positive} + \textit{False Positive}} \tag{5}$$

$$\textit{Recall} = \frac{\textit{True Positive}}{\textit{True Positive} + \textit{False Negative}} \tag{6}$$

$$\textit{F1} = 2 \times \frac{\textit{Precision} \times \textit{Recall}}{\textit{Precision} + \textit{Recall}} \tag{7}$$

To better evaluate the performance during motion, the overlapping areas at the motion start position and end position are taken into account. Since different detectors and descriptors have different performances even at the same viewpoint, a better understanding of the effect of motion could be obtained by also considering the performance during standstill. The standstill overlapping area ratio at start position $\mathbf{O}_{start}$ is obtained by (3). The standstill overlapping area ratio at the end position $\mathbf{O}_{end}$ is calculated in the same way. They are considered as the baseline with respect to $\mathbf{O}_{motion}$, which is the overlapping area ratio during motion.

To evaluate the speed of the descriptor, average process time of feature detector and descriptor and the frame loss rate l is used,

$$l = 1 - \frac{N_p}{N_c} \tag{8}$$

where N_p is the number of images processed during the motion and N_c is the number of images captured during the motion.

IV. Result

Fig. 3 shows the F1 score of all descriptor used in the experiment and Table I gives the recall. When the camera is standing still at the start and end position, all descriptors have high F1 values. It is desirable that the bounding box is accurately projected. With motion introduced, the F1 scores drop significantly which suggests that motion has a huge impact on the distinctiveness. Fig. 4 shows the overlapping ratio

of descriptors in both standstill and motion. All descriptors have a comparable overlapping ratio when the scene camera is standstill at start and end position, which is consistent with the observation from Fig. 3. The median of overlapping ratios is around 0.7. AKAZE, BRISK and SIFT are more stable than others. All the descriptors are affected by the motion. With the increasing acceleration, the overlapping ratio drops in general and becomes more unstable. One exception is AKAZE at accelaretion= $50°/s^2$, it still has comparable result to the standstill case but slightly less stable. ORB is severely affected by the motion. AKAZE is affected least among all descriptors. SURF and BRISK is less sensitive towards the change in acceleration.

Table II indicates the loss rate of descriptors and average process time of feature detection and description. ORB and BRISK can potentially meet the real-time requirement. Reduce image resolution could decrease the processing time, but a more promising solution is to use GPU accelerated detector and descriptor to achieve real-time performance [9] [22].

In a HRC scenario, when there is no head or body movement from human, all the descriptors will correctly project the gaze. When motion occurs, the F1 score drops due to the increasing number of False Negative and False Positive. False Negative represents the gaze point in the scene camera of eye tracking glasses which could not be projected to the robot camera. As indicated in Table I, only a small amount of False Negative presents for BRISK, ORB and SIFT during motion. The minor loss of the projected gaze point is tolerable. False Positive means an incorrectly projected gaze point onto the image of robot camera. An extremely small overlapping ratio means the projected gaze is severely distorted which is not desired. Fig. 5 shows some examples of the distorted bounding box in robot image. In addition, if the values of the overlapping ratio locate in a wide range, the projected gaze becomes very unstable, such as in Fig. 4c. This generates high frequency noise over time. Considering both accuracy and processing speed, BRISK, AKAZE and SURF appears to meet the requirement better.

TABLE I: Recall

Recall	AKAZE	BRISK	ORB	SIFT	SURF
start position	1.0	1.0	1.0	1.0	1.0
$50°/s^2$	1.0	0.99	0.975	1.0	1.0
$100°/s^2$	1.0	0.95	0.967	1.0	1.0
$150°/s^2$	1.0	0.847	0.852	0.962	1.0
end position	1.0	1.0	1.0	1.0	1.0

TABLE II: Descriptor Loss Rate

	AKAZE	BRISK	ORB	SIFT	SURF
loss rate	0.179	0.03	0.02	0.481	0.399
average process time (detector+descriptor in ms)	63.06	35.69	20.79	106.39	81.85

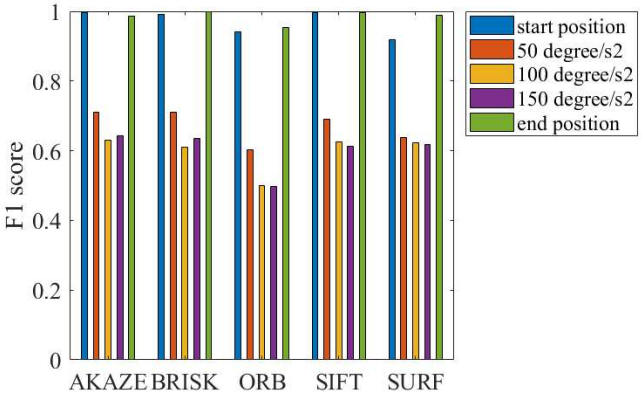
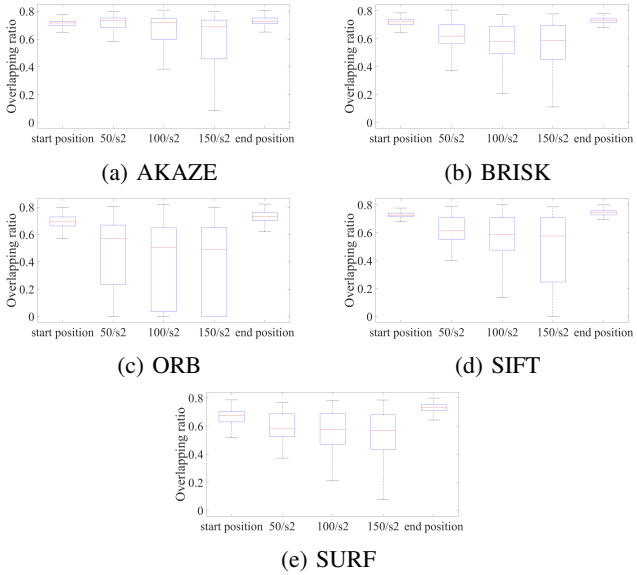


Fig. 3: The F1 score of descriptors. For each descriptor, the F1 score at the start and end position, with acceleration $50°/s^2$, $100°/s^2$ and $150°/s^2$ are displayed.



(a) AKAZE (b) BRISK

(c) ORB (d) SIFT

(e) SURF

Fig. 4: The overlapping ratio of descriptors. For each descriptor, the overlapping ratio at the start and end position, with acceleration $50°/s^2$, $100°/s^2$ and $150°/s^2$ are displayed.

V. Conclusion

In this paper, we analyse the AKAZE, BRISK, ORB, SIFT and SURF feature descriptor for application in eye tracking based HRC. We perform the experiment in a real world scenario where variances in scale, rotation and viewpoint co-exist. We focus on how accurate and how fast can gaze be projected from image of eye tracking glasses to an image of the robot camera. BRISK, AKAZE and SURF are seemingly more desirable than other descriptors by considering accuracy and processing time. However, both AKAZE and SURF cannot meet real-time requirement. Using GPU accelerated detectors and descriptor could possibly solve the problem of computation time.

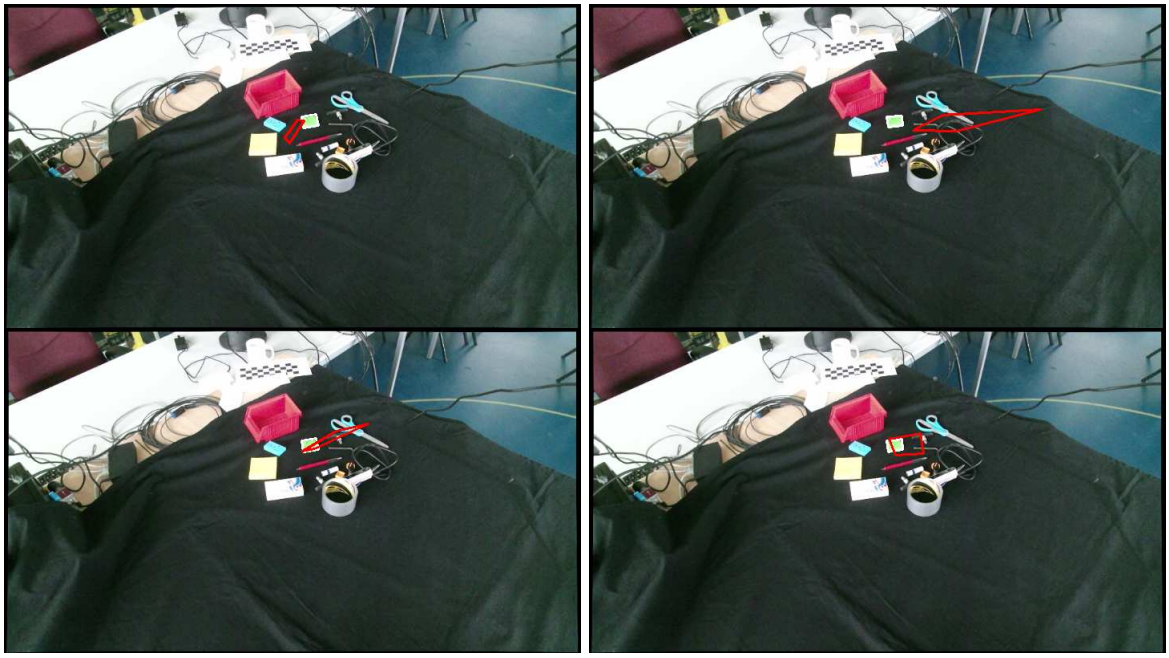


Fig. 5: Examples of incorrectly projected bounding box on robot image.

REFERENCES

[1] D. G. Lowe, "Object recognition from local scale-invariant features," in *Computer vision, 1999. The proceedings of the seventh IEEE international conference on*, vol. 2. Ieee, 1999, pp. 1150–1157.

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

[3] H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: Speeded up robust features," in *European conference on computer vision*. Springer, 2006, pp. 404–417.

[4] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "Orb: An efficient alternative to sift or surf," in *Computer Vision (ICCV), 2011 IEEE international conference on*. IEEE, 2011, pp. 2564–2571.

[5] P. F. Alcantarilla and T. Solutions, "Fast explicit diffusion for accelerated features in nonlinear scale spaces," *IEEE Trans. Patt. Anal. Mach. Intell*, vol. 34, no. 7, pp. 1281–1298, 2011.

[6] S. Leutenegger, M. Chli, and R. Y. Siegwart, "Brisk: Binary robust invariant scalable keypoints," in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 2548–2555.

[7] A. Alahi, R. Ortiz, and P. Vandergheynst, "Freak: Fast retina keypoint," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*. Ieee, 2012, pp. 510–517.

[8] S. Gauglitz, T. Höllerer, and M. Turk, "Evaluation of interest point detectors and feature descriptors for visual tracking," *International journal of computer vision*, vol. 94, no. 3, p. 335, 2011.

[9] A. Pieropan, M. Björkman, N. Bergström, and D. Kragic, "Feature descriptors for tracking by detection: A benchmark," *arXiv preprint arXiv:1607.06178*, 2016.

[10] A. Gil, O. M. Mozos, M. Ballesta, and O. Reinoso, "A comparative evaluation of interest point detectors and local descriptors for visual slam," *Machine Vision and Applications*, vol. 21, no. 6, pp. 905–920, 2010.

[11] Y. Razin and K. M. Feigh, "Learning to predict intent from gaze during robotic hand-eye coordination." in *AAAI*, 2017, pp. 4596–4602.

[12] R. M. Aronson, T. Santini, T. C. Kübler, E. Kasneci, S. Srinivasa, and H. Admoni, "Eye-hand behavior in human-robot shared manipulation," in *Proceedings of the 2018 ACM/IEEE International Conference on Human-Robot Interaction*. ACM, 2018, pp. 4–13.

[13] E. Rosten and T. Drummond, "Fusing points and lines for high performance tracking," in *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, vol. 2. IEEE, 2005, pp. 1508–1515.

[14] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1. IEEE, 2005, pp. 886–893.

[15] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *IEEE transactions on pattern analysis and machine intelligence*, vol. 27, no. 10, pp. 1615–1630, 2005.

[16] M. Calonder, V. Lepetit, C. Strecha, and P. Fua, "Brief: Binary robust independent elementary features," in *European conference on computer vision*. Springer, 2010, pp. 778–792.

[17] J. Heinly, E. Dunn, and J.-M. Frahm, "Comparative evaluation of binary features," in *Computer Vision–ECCV 2012*. Springer, 2012, pp. 759–773.

[18] H. Chatoux, F. Lecellier, and C. Fernandez-Maloigne, "Comparative study of descriptors with dense key points," in *Pattern Recognition (ICPR), 2016 23rd International Conference on*. IEEE, 2016, pp. 1988–1993.

[19] P. F. Alcantarilla, A. Bartoli, and A. J. Davison, "Kaze features," in *European Conference on Computer Vision*. Springer, 2012, pp. 214–227.

[20] X. Yang and K.-T. Cheng, "Ldb: An ultra-fast feature for scalable augmented reality on mobile devices," in *2012 IEEE international symposium on mixed and augmented reality (ISMAR)*. IEEE, 2012, pp. 49–57.

[21] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. Van Gool, "A comparison of affine region detectors," *International journal of computer vision*, vol. 65, no. 1-2, pp. 43–72, 2005.

[22] M. Björkman, N. Bergström, and D. Kragic, "Detecting, segmenting and tracking unknown objects using multi-label mrf inference," *Computer Vision and Image Understanding*, vol. 118, pp. 111–127, 2014.