2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
Macau, China, November 4-8, 2019

Position-based monocular visual servoing of an unknown target using online self-supervised learning

Chungkeun Lee, Hoseong Seo, and H. Jin Kim*

Abstract— **Visual servoing, i.e. control with visual information, is a valuable capability in many robotic applications. In particular, position based visual servoing (PBVS) estimates position information from the observed image to generate visual servo control. However, the estimation of the position of an unknown target using monocular images is still difficult due to the complexity of the image information. For the target estimation problem, we propose to integrate three complementary techniques for monocular visual servoing. First, to estimate the probability of a target's existence, the learning model with spatial features from convolution neural network is proposed. Second, the extended Kalman filter based on epipolar geometry estimates the 3D position of the target; moreover, from this 3D position, the perception model is trained online by self-generated virtual ground-truth. Finally, visual servo control is generated, and the resulting movement helps to construct epipolar geometry. Finally. the experimental validation is performed in a challenging setting involving occlusion and target's shape change.**

I. Introduction

Visual servoing generates velocity control commands for a robot from visual information [1]–[15]. In particular, position-based visual servoing (PBVS) uses the estimated relative position between current and desired coordinates, whereas image-based visual servoing (IBVS) generates control from the difference of current and desired visual.

This paper investigates PBVS with monocular vision. Although a significant body of works have been reported PBVS approaches [16]–[18] that estimate relative pose of the target from image, and then generate control command from the estimated pose, relative pose estimation is still not trivial for monocular cameras that only observe 2D projection of a real world. In this reason, most PBVS researches have focused on generating control when the relative pose was estimated well by RGB-D camera or from known markers such as QR, ArUco.

Recent approaches to utilize learning for visual servoing [12]–[15] generate a control law from a trained model. This model needs to be trained via dataset that was collected in an off-line stage. Training dataset contains expensive information, such as a pair of images and their relative pose. In addition, for robustness with respect to environments or target, a large amount of data need to be collected under various conditions.

To overcome those difficulties, we propose to utilize spatial features from convolution neural network (CNN) in visual servoing control. Since CNN features involve characteristics of the image, they could provide useful information for generic target perception. However, their meaning is not known. To solve this problem, we construct a learning model, simple logistic regression, that can interpret the CNN features into existence probability map. To guarantee robustness with target deformation or viewpoint change, the proposed model is learned online in a self-supervised manner, i.e. by supervised learning with the self-generated virtual ground-truth, that will be generated from the estimated position. For accurate ground-truth generation using the estimated relative pose information, we propose extended Kalman filter based on epipolar geometry.

Additionally, a proper control strategy is necessary for good pose estimation. To estimate 3D position with monocular camera, the robot needs to construct epipolar geometry with viewpoint parallax. Generating parallax by robot movement, however, may cause a loss of the target out of view due to viewpoint change. To balance those sub-missions to achieve visual servo, we utilize iterative linear quadratic regulator(iLQR) [19], one of the optimal control methods, with a proper cost function reflecting them.

Contributions of the proposed method can be summarized as follows:

- CNN spatial features for perception of a generic unknown target from a single frame without prior information or markers.
- Self-supervised online learning for robust target perception in challenging environments.
- Virtual ground-truth generation using the extended Kalman filter for online learning.
- iLQR controller with a suitable cost to help estimation and keep the field-of-view constraint.
- Experimental validation of the proposed method.

In this paper, related works are described in Section II. Then, the proposed method is described in two sections; relative position estimation with visual perception in Section III, and visual servo control in Section IV. Experimental results are in Section V, and the paper concludes in Section VI.

II. Related Works

Visual servoing [20] has a rich history. In this section, some related works in visual servoing are categorized into

Chungkeun Lee, Hoseong Seo, and H. Jin Kim are with the Department of Mechanical and Aerospace Engineering, Seoul National University, Gwanak-gu, Seoul, 08826, Korea e-mail: {elkein, hosung37, hjinkim} @ snu.ac.kr.

This work was supported by Institute of Information Communications Technology Planning & Evaluation(IITP) grant funded by the Korea government(MSIT) (No. 2019-0-00399, Development of A.I. based recognition, judgement and control solution for autonomous vehicle corresponding to atypical driving environment)

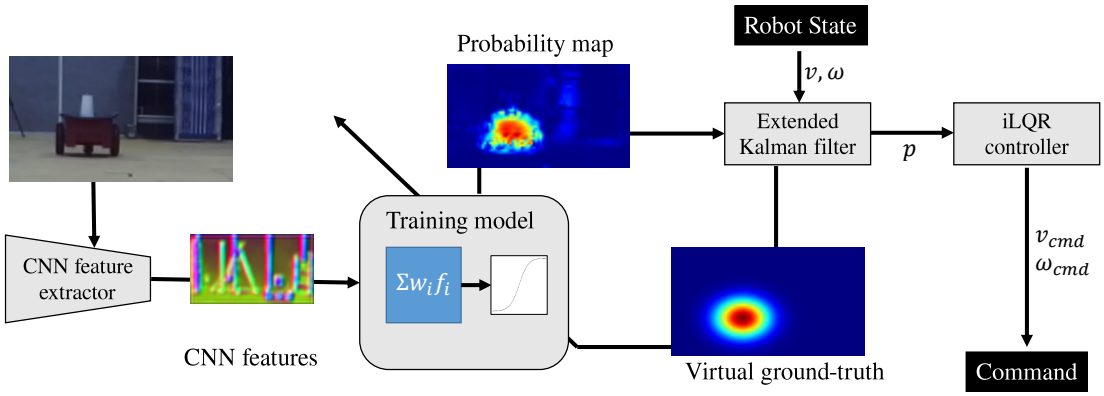


Fig. 1. The flowchart of target position estimation. From a given image, the target existence probability map is estimated from the CNN model. Then, the extended Kalman filter is updated by probability map for target's pose estimation. Finally, from the estimated pose, the iterative linear quadratic regulator(iLQR) controller generates control input and CNN model is trained online.

four parts: feature-based IBVS, visual tracking with visual servo control, position-based visual servoing, and learning-based visual servoing.

A. Feature-based visual servoing

Feature-based visual servoing utilizes specified features [1], such as four single-colored points or circles in an other colored background in [2]–[5], four black-white two-by-two checkerboards in [3], one circle with known size in [6], and edges of a single-colored rectangle in [7]. In most cases, single-colored features were set as a target and users were given this information.

Unlike most feature-based visual servoing methods that require easily recognizable features whose information has to be known to the user, the proposed method does not require any target information except the initial position. For a given initial position of the target, feature information is automatically generated as an existence probability map.

B. Visual tracking with visual servo control

In classical IBVS, monocular images do not provide robust features. In such cases, a visual tracker can be utilized as a robust feature extractor. Nowadays, state-of-the-art learning-based trackers can function even in challenging environments [21]–[24]. However, to achieve those performance, they need lots of data, lots of training time, and lots of test time. Most of those methods spends approximately 100-1000ms on one step tracking even in high-end desktop computer. Mobile robot application cannot afford this huge computation.

To overcome this problem, some researches have proposed real-time visual tracking [25]–[27]. Many tracking methods provide rectangle-shape perception, which may cause inaccurate perception of the target's size. When additional information in inertial coordinate is available, such as the pose of the robot itself, it may help track the target. but most tracking algorithms do not consider this. Unlike most algorithms that do not consider such information, the proposed algorithm utilizes the target's 3D pose information during model update for robustness to target change.

C. Position-based visual servoing

Most PBVS consist of relative pose estimation and then control input generation with the known relative pose. Some researches [16]–[18] utilize easy-to-find targets, such as a single-colored rigid object, known markers or known points.

The relative pose estimation from unknown target, however, is challenging. The proposed visual servoing utilizes robot state to increase accuracy of the position estimation with respect to an unknown target.

D. Learning-based visual servoing

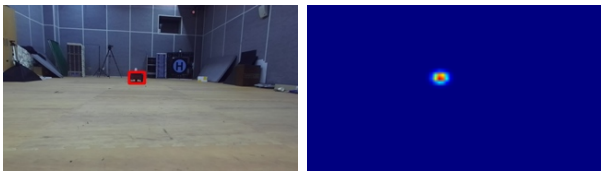
Learning-based approaches such as convolution neural networks are making advances for visual systems. In [12], [13], CNN was trained to estimate the relative pose of two given images for PBVS. In [14], [15], CNN estimated the next image from the current image and robot action. Then, the control law was generated based on the predictive model of trained CNN.

Most learning-based approaches require a large amount of expensive training data with the relative pose or robot action. The proposed method only needs the initial image with the target position. Additionally, the proposed CNN network has a lighter computation load, thanks to the pre-trained CNN with a logistic training model.

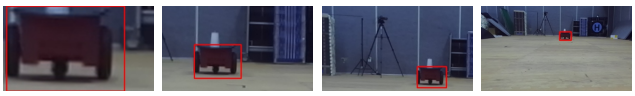
III. Position Estimation of Generic Target

To estimate the position of a target with a monocular camera, we need to overcome two main problems: where a target exists in the image, and how to find out the depth of the target. To perceive the target from an observed image in real time, we propose the learning model with features from pre-trained convolution neural network. To estimate the depth of the target, we construct an extended Kalman filter with the perceived target in image coordinate.

The overall flowchart of overall estimation is shown in Fig. 1. First, the convolution neural network extracts spatial features from a color image. Second, the learning model estimates the probability map of the target existence in image coordinate. Then, to find out the position of the target in



(a) The initial image (left) and ground-truth probability (right)



(b) Augmented images for train

(c) Validation result of training images

Fig. 2. An example of initial training. From an initially given image and region pair, the ground-truth is generated as in (a). This procedure is iterated for several generated images in (b). After training, the network output looks like (c). Ten cropped images are generated in this example.

$\mathbb{R}^3$, the extended Kalman filter is updated from the target's existence. Finally, the estimation result is used for generating the visual servo command and online training of the learning model.

A. Target perception with spatial features from CNN

CNNs are widely utilized as a generic feature extractor in [14], [28]. In this work, to maintain 2D spatial information of features, we utilize spatial features from imagenet pre-trained CNN. It can extract hundreds of down-sampled spatial features for a given color image, but meaning of those features cannot be interpreted by the human.

To convert those features into meaningful information, we construct a simple logistic model that allows to estimate the probability map of the target's existence. This model is derived as (1). Thanks to the simplicity of a logistic model, it can be trained online in real-time, almost 100 times faster than training a full CNN.

The probability map p is obtained by

$$p = \sigma(\sum_i w_i f_i + b) \tag{1}$$

where $\sigma(x)$ is a sigmoid function, w_i and b are trained weights, and f_i's are pre-trained CNN feature.

To train this model, we minimize cross-entropy loss function $\mathfrak{L}$ in (2), which is widely used for regression of probability function,

$$\mathfrak{L} = -\frac{1}{N}\sum_{n=1}^{N}[p_n \log \hat{p}_n + (1-p_n)\log(1-\hat{p}_n)] \tag{2}$$

where p is the ground-truth probability, $\hat{p}$ is the estimated probability.

From an initially given image and region pair, we initialize the network by supervised optimization. From a given rectangle region, we generate ground-truth by Gaussian-like probability map for network training. Additionally, to increase robustness with respect to the scale, we generate some virtually cropped images containing the target region and ground-truth as training image. Fig. 2 shows an example of the training data and validation result.

B. Pose estimation with extended Kalman filter

The extended Kalman filter has been widely utilized to estimate the target's pose information from observation in image coordinate [29]–[31]. To construct an observation model from the image coordinate, we use a pinhole camera model with its viewing direction x.

The main problem of the monocular camera is that the x directional distance (depth) cannot be estimated from only one point. Thus, for x direction estimation, the consistently observed multiple points or size information are necessary. From unknown arbitrary target, however, it is difficult to extract those information consistently, because the target's color, shape, or appearance may be deformed or changed.

One solution is to utilize epipolar geometry by viewpoint parallax. The view angle of the target varies as the robot and target move, and if the robot moves with generating viewpoint parallax, the x direction information can be estimated.

Additionally, to generate the virtual ground-truth for perception, the probability map needs to be generated from current state information. Since the target's shape is unknown, the generic observation model, i.e. the Gaussian-like ellipse shape probability map in (3), is assumed.

$$p(X) = \exp\left(-(X-\bar{X})^T S^{-1}(X-\bar{X})\right) \tag{3}$$

where $X = (x, y)$ is a point of observation, $\bar{X}$ is a position of the target and $S \in \mathbb{R}^{3\times3} > 0$ is a parameter for scale representation.

To express the probability model as in (3), the target's position and scale parameter are selected as the states, as in (4). each of six scale parameters in (6) represents standard deviation or correlation. To imply epipolar geometry in linearized measurement model, all states are represented in inertial coordinate.

$$\zeta = (x, y, z, \sigma_x, \sigma_y, \sigma_z, \rho_{yz}, \rho_{xz}, \rho_{zx}) \tag{4}$$

$$X = \begin{pmatrix} x & y & z \end{pmatrix} \tag{5}$$

$$S = \begin{pmatrix} \sigma_x^2 & \sigma_x\sigma_y\rho_{xy} & \sigma_x\sigma_z\rho_{xz} \\ \sigma_x\sigma_y\rho_{xy} & \sigma_y^2 & \sigma_y\sigma_z\rho_{yz} \\ \sigma_x\sigma_z\rho_{xz} & \sigma_y\sigma_z\rho_{yz} & \sigma_z^2 \end{pmatrix} \tag{6}$$

where ζ is the state of the Kalman filter, and X, S are position and scale parameter of the target in the inertial frame respectively.

To simplify the probability model in the image coordinate, we assume that all observed points have the same depth with the target's nominal position as in (7). Then, the probability model in image coordinate as in (8)-(10).

$$\mathfrak{d} = (Re_1)^T(X - X_r) \tag{7}$$

$$X_n = R^T(X - X_r)/\mathfrak{d} \tag{8}$$

$$S_n = R^T S R/\mathfrak{d}^2 \tag{9}$$

$$p(X_n) = \exp\left(-(X_n - \bar{X}_n)^T S_n^{-1}(X_n - \bar{X}_n)\right) \tag{10}$$

where (X_r, R) is a position and rotation pair of the robot, and $\bullet_n$ is an image coordinate representation of $\bullet$.

As a measurement, instead of pixel-wise probability, the parameterize measurement h in (11) is utilized for fast filter update without loss of information.

$$h = (\bar{x}_n, \bar{y}_n, \sigma_{nx}, \sigma_{ny}, \rho_{nxy}) \quad (11)$$
$$\bar{X}_n = \mathbb{E}_p[X_n] \quad (12)$$
$$S_n = \mathbb{E}_p\left[(X_n - \bar{X}_n)(X_n - \bar{X}_n)^T\right] \quad (13)$$

Here $\mathbb{E}_p$ is an expectation function with respect to the observed probability map p.

The mean value, however, is susceptible to the false positive noise. Thus, the noise reduction is necessary, and we perform it with two models. The first model is the suppression of low-probability pixel keeping the total area as in (14). The second model is the suppression of faraway pixel from the major region as in (15).

$$p_b(x) = \min(kp^n(x), 1) \quad (14)$$
$$p_n(x) = \min(p(x), \exp\left(-m(x-\mu)^T\Sigma(x-\mu)\right)) \quad (15)$$

where $k > 0$ is a positive gain satisfying $\int p_b(x) = \int p(x)$, and μ, Σ are mean and covariance of p, and $n > 1$, $m > 1$ are parameters for reduction ratio.

IV. Visual Servo Control

In this paper, we control a non-holomonic 2-dof ground robot using the iterative linear quadratic regulator(iLQR) controller [19] as a base controller. iLQR control is an optimal control method, which generates optimal state and input from time discretization and system linearization by differential dynamic programming. For iLQR, discretized dynamics and cost function are necessary, and the mission can be represented as the cost function.

The discretized dynamics is simple Euler integration formula of the simple car model with velocity limit. The velocity limit is directly applied by the tanh function.

$$x_{n+1} = x_n + v_M \tanh u_v \sin\theta_n \, \Delta t \quad (16)$$
$$y_{n+1} = y_n + v_M \tanh u_v \cos\theta_n \, \Delta t \quad (17)$$
$$\theta_{n+1} = \theta_n + \omega_M \tanh u_\omega \, \Delta t \quad (18)$$

where x_n, y_n, θ_n are states at time step n(2D position and heading), u_v, u_ω are inputs, v_M, ω_M is the linear, angular velocity limit, Δt is the discretized time interval.

To construct the cost function for the visual servo problem, four criteria are considered. First, to achieve the main visual servoing mission, the distance between the robot and the target is kept to the desired distance. Second, the robot needs to generate a path to help the pose estimation of Kalman filter not to fail the depth estimation. Third, the robot should capture the target in camera. Lastly, the robot should not approch too close not to collide with the target.

To help estimation, covariance of the Kalman filter can be used as a criterion of accuracy estimation. For example, the covariance update at each measurement update step could be utilized, but its direct usage is not straightforward for controller optimization.

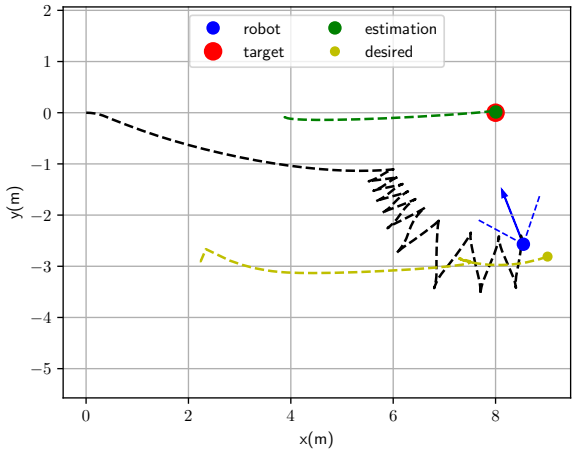


Fig. 3. The simulation of iLQR control with EKF estimation. The estimation of the target (green) converges to the ground-truth (red). Also, the robot (blue) moves toward the virtual desired state (yellow).

Instead, we generate the virtual desired state to minimize covariance. The variance of heading direction decreases very little due to the pinhole camera model in a monocular camera, and the robot needs to avoid heading to the direction of large variance. By this concept, the virtual desired states are formulated in (19)-(22). By this virtual desired state, first two criteria of the previously mentioned are held.

$$\alpha_d = \frac{\lambda_1\theta_2 + \lambda_2\theta_1 + k\theta}{\lambda_1 + \lambda_2 + k} \quad (19)$$
$$x_d = x_t + \mathfrak{d}_d \cos(\alpha_d) \quad (20)$$
$$y_d = y_t + \mathfrak{d}_d \sin(\alpha_d) \quad (21)$$
$$\theta_d = -\alpha_d \quad (22)$$

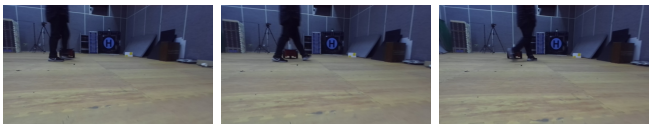
where $\mathfrak{d}_d$ is the desired distance, (λ_i, v_i) is an eigenvalue, eigenvector pair of the covariance matrix of target's 2D position, $\theta_i = \angle(v_i)$ is an eigenvector in angle space, and k is a coefficient to adjust how much the robot's heading will be changed.

The cost function is the sum of three parts: quadratic cost of the state error and input to chase the target as in (23), soft-constraint about the field of view limit of the camera as in (24), and penalty cost to avoid getting too close to the target as in (25). At the final stage, a different quadratic cost as in (26) is considered to emphasis the error of final state.

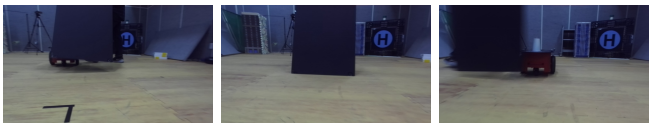
$$J = (X - X_d)^T Q (X - X_d) + u^T R u \quad (23)$$
$$+ k_f\left[1 + \tanh\left(\kappa_f(\theta_d + \theta_f)\right)\tanh\left(\kappa_f(\theta_d - \theta_f)\right)\right] \quad (24)$$
$$+ k_d\left(1 - \tanh\kappa_d\left(\|X - X_r\|^2 - d\right)\right) \quad (25)$$
$$J_f = (X_f - X_d)^T Q_f (X_f - X_d) \quad (26)$$

where $X = (x, y, \theta)$ is the state of the robot, X_f is the final state of the robot, $\cdot_d$ is a desired state of the robot, u is a input, θ_f is an field of view limit, and $Q, R, k_f, k_d, \kappa_f, \kappa_d, Q_f$ are coefficients.

Fig. 3. shows the simulation result with the combination of the extended Kalman filter estimator and iLQR controller.



(a) Partial occlusion for 1 second



(b) Full occlusion for 3 seconds

(c) Shape change by rotation of the target

Fig. 4. The snapshot of the scenario: the target is partially occluded in (a), fully occluded in (b), and changes its shape in (c)

Since the robot orients toward the x-direction at start, the x-direction estimation is poor; thus, the virtual desired state is generated to prevent heading to the x-direction like the yellow-line. Because of the field-of-view constraint with non-holonomic system, the shape of path in the final stage is like a petal, and such movement helps to avoid losing the depth estimation quality.

V. Experiment

We test the proposed method in a challenging condition. In this experiment, a ground robot will follow the target whose shape may be changable by rotating itself and occlusion. To evaluate the estimation quality, the ground-truth pose is observed from VICON system. The VICON data is used only for quantifying the estimation error, and the robot does not know any information from VICON.

A. Scenario

The robot follows the target, a static pioneer. The experiment involves partial occlusion at approx. 1 second, total occlusion at approx. 3 seconds, and the shape change due to the pioneer rotating itself.

B. Setup

As the hardware, a pioneer 3-DX robot is constituted with a laptop (i7-7700HQ@2.8Hz and GTX 1050ti) and a monocular camera that gives 1080×720 images, and the horizontal field of view is 90 deg. To estimate the pose of the robot itself, we utilize the odometry given from pioneer.

In this experiment, the follow parameters are used.

- pre-trained CNN: Alexnet [32]
- initial guess of the distance: 8 m
- field of view: $\theta \in [-0.7, 0.7]$ rad
- velocity: $v \in [-0.25, 0.25]$ m/s, $\omega \in [-0.25, 0.25]$ rad/s
- EKF covariance
  - prediction: $[0.01, 0.1, 0.001]$ for $[x, \sigma, \rho]$
  - observation: $[0.01, 0.1]$ for $[x_n, \sigma_n]$

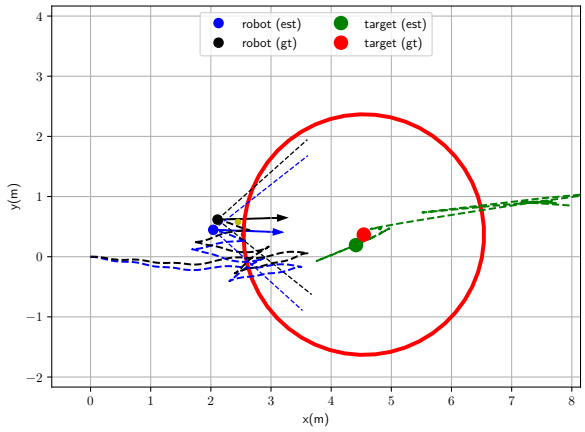


Fig. 5. The visual servoing experiment result (top-down view). The ground-truth(gt) is observed from VICON, and estimation(est) is estimated by proposed algorithm.

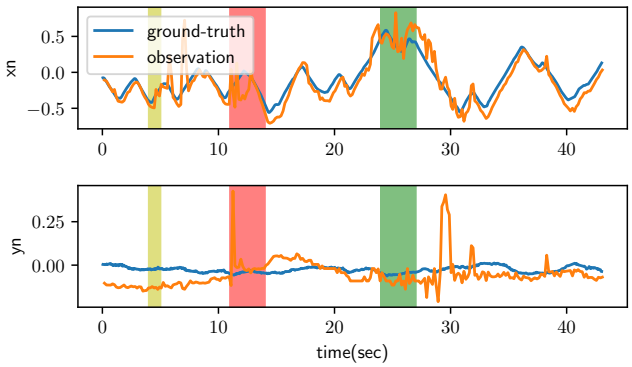


Fig. 6. The perception result of the nominal point of the target in normalized image coordinate. The target is partially/fully occluded in the yellow/red areas respectively, and rotating in green area as in Fig. 4 The ground-truth is from VICON.

- iLQR steps: 0.05 sec, 50 steps
- iLQR cost
  - $Q = [0.1, 0.1, 0]$ for $[x, y, \theta]$
  - $Q_f = [10, 10, 0.001]$ for $[x, y, \theta]$
  - $R = [0.0001, 0.0001]$
  - $k_f, \kappa_f = 10, 10$
  - $k_d, \kappa_d = 10, 10$

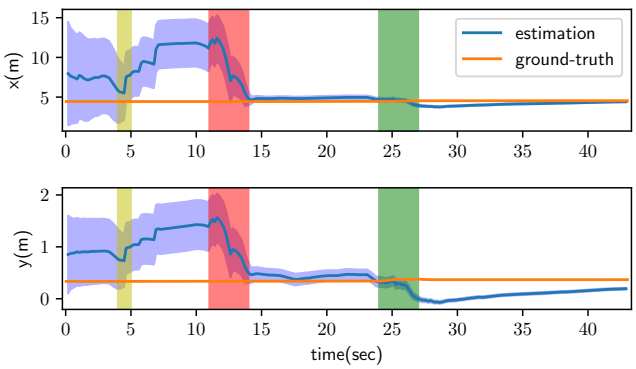
C. Result

Figs 5-7 show the experimental result. Fig. 5 shows the overall path and estimation result. The position estimation in inertial frame seems inaccurate in Fig. 7 (a) due to the pioneer odometry, but the proposed the relative pose estimation gives accurate results as shown in Fig. 7 (b).

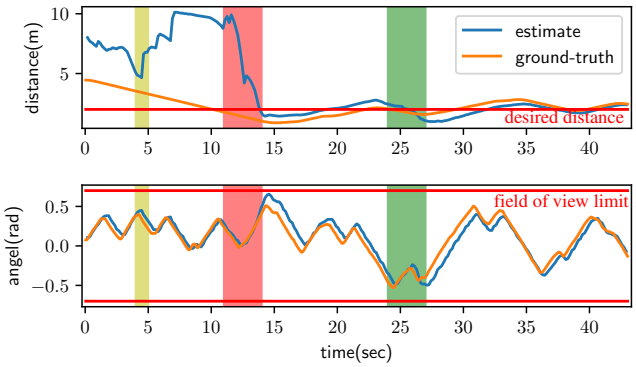
Fig. 8 shows the quality of the proposed perception method. Most perception areas, shown in red, are focused on the target. This can be seen in fig. 6 which shows the perception result in normalized image plane.

D. Comparison

In this section, we track the target using state-of-the-art



(a) Target position estimation in inertial coordinate



(b) Target position estimation in body coordinate

Fig. 7. Visual servo experiment result. Although the estimation in the inertial frame has error as in Fig. (a), it is due to the error of the robot odometry estimation given from pioneer. The relative pose estimation in the body frame according to the proposed method is accurate as shown in Fig. (b). The target is partially/fully occluded in the yellow/red areas respectively, and rotating in green area as in Fig. 4

trackers [25]–[27] with logged image from experiment. Fig. 9 shows the tracking result against occlusion. The proposed one can track even with occlusion, because it can estimate the target's position in image coordinate thanks to 3D pose estimation.

VI. Conclusions

In this paper, we propose position based visual servoing based on perception from spatial CNN features, estimation using extended Kalman filter, and control using iterative linear quadratic regulator. By spatial CNN features, the proposed approach can perceive an unknown target only even when the initial information changes. From extended Kalman filter with epipolar geometry, we can estimate the target position, which is utilized for online training of perception. Also, iLQR controller generates the path to estimate the target's depth easily. The proposed method is validated from experiments, which include occlusion or shape change.

References

[1] F. Chaumette and S. Hutchinson, "Visual servo control. i. basic approaches," *IEEE Robotics & Automation Magazine*, vol. 13, no. 4, pp. 82–90, 2006.

[2] E. Marchand and F. Chaumette, "Visual servoing through mirror reflection," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, May 2017, pp. 3798–3804.

[3] X. Zhang, Y. Fang, B. Li, and J. Wang, "Visual servoing of nonholonomic mobile robots with uncalibrated camera-to-robot parameters," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 1, pp. 390–400, 2017.

[4] D. Zheng, H. Wang, J. Wang, S. Chen, W. Chen, and X. Liang, "Image-based visual servoing of a quadrotor using virtual camera approach," *IEEE/ASME Transactions on Mechatronics*, vol. 22, no. 2, pp. 972–982, April 2017.

[5] A. McFadyen, M. Jabeur, and P. Corke, "Image-based visual servoing with unknown point feature correspondence," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 601–607, April 2017.

[6] H. Wang, B. Yang, Y. Liu, W. Chen, X. Liang, and R. Pfeifer, "Visual servoing of soft robot manipulator in constrained environments with an adaptive controller," *IEEE/ASME Transactions on Mechatronics*, vol. 22, no. 1, pp. 41–50, 2017.

[7] D. Xu, J. Lu, P. Wang, Z. Zhang, and Z. Liang, "Partially decoupled image-based visual servoing using different sensitive features," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, no. 8, pp. 2233–2243, Aug 2017.

[8] G. Silveira and E. Malis, "Direct visual servoing: Vision-based estimation and control using only nonmetric information," *IEEE Transactions on Robotics*, vol. 28, no. 4, pp. 974–980, Aug 2012.

[9] A. Dame and E. Marchand, "Mutual information-based visual servoing," *IEEE Transactions on Robotics*, vol. 27, no. 5, pp. 958–969, Oct 2011.

[10] C. Collewet and E. Marchand, "Photometric visual servoing," *IEEE Transactions on Robotics*, vol. 27, no. 4, pp. 828–834, Aug 2011.

[11] Q. Bateux and E. Marchand, "Histograms-based visual servoing," *IEEE Robotics and Automation Letters*, vol. 2, no. 1, pp. 80–87, Jan 2017.

[12] A. Saxena, H. Pandya, G. Kumar, A. Gaud, and K. M. Krishna, "Exploring convolutional networks for end-to-end visual servoing," in *Robotics and Automation (ICRA), 2017 IEEE International Conference on*. IEEE, 2017, pp. 3817–3823.

[13] Q. Bateux, E. Marchand, J. Leitner, F. Chaumette, and P. Corke, "Training deep neural networks for visual servoing," in *Robotics and Automation (ICRA), 2018 IEEE International Conference on*, 2018, pp. 3307–3314.

[14] A. X. Lee, S. Levine, and P. Abbeel, "Learning visual servoing with deep features and fitted q-iteration," *arXiv preprint arXiv:1703.11000*, 2017.

[15] C. Finn and S. Levine, "Deep visual foresight for planning robot motion," in *Robotics and Automation (ICRA), 2017 IEEE International Conference on*. IEEE, 2017, pp. 2786–2793.

[16] W. J. Wilson, C. W. Hulls, and G. S. Bell, "Relative end-effector control using cartesian position based visual servoing," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 5, pp. 684–696, 1996.

[17] G. Dong and Z. Zhu, "Position-based visual servo control of autonomous robotic manipulators," *Acta Astronautica*, vol. 115, pp. 291 – 302, 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0094576515002313

[18] A. Al-Shanoon, A. Hao, H. Lang, and Y. Wang, "Mobile robot regulation with position based visual servoing," in *2018 IEEE International Conference on Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA)*, June 2018, pp. 1–6.

[19] Y. Tassa, N. Mansard, and E. Todorov, "Control-limited differential dynamic programming," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2014, pp. 1168–1175.

[20] B. Espiau, F. Chaumette, and P. Rives, "A new approach to visual servoing in robotics," *IEEE Transactions on Robotics and Automation*, vol. 8, no. 3, pp. 313–326, Jun 1992.

[21] H. Nam and B. Han, "Learning multi-domain convolutional neural networks for visual tracking," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4293–4302.

[22] L. Wang, W. Ouyang, X. Wang, and H. Lu, "Visual tracking with fully convolutional networks," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 3119–3127.

[23] M. Danelljan, G. Bhat, F. S. Khan, M. Felsberg, *et al.*, "Eco: Efficient convolution operators for tracking." in *CVPR*, vol. 1, no. 2, 2017, p. 3.

[24] F. Li, C. Tian, W. Zuo, L. Zhang, and M.-H. Yang, "Learning spatial-

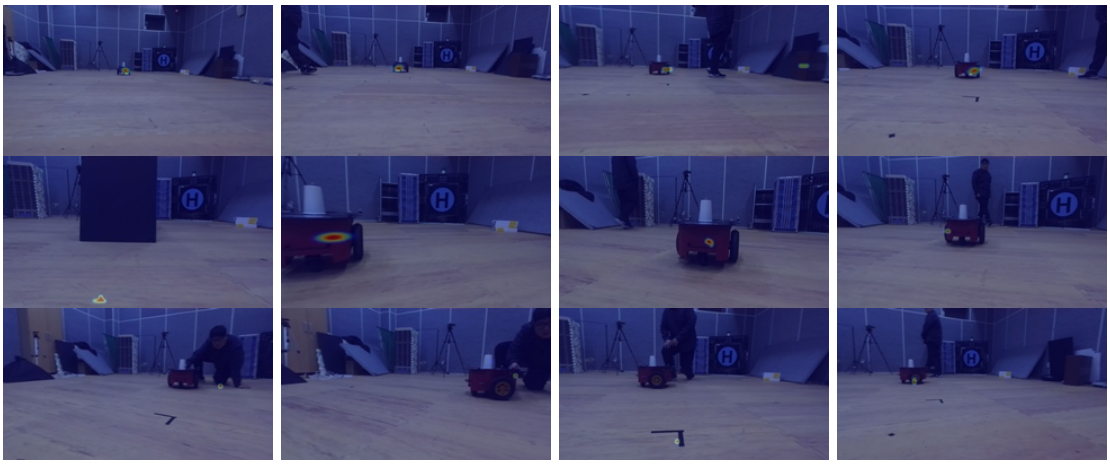
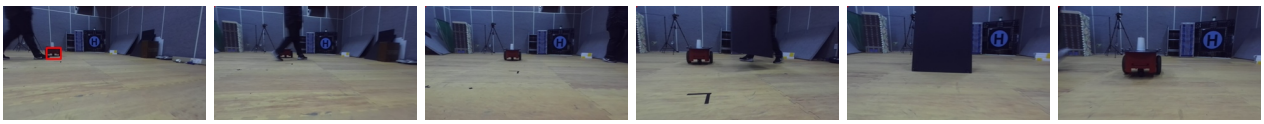


Fig. 8. The snapshot of perception result with captured image. Blue area is low existence probability and red area is high existence probability. (per 2 second)

(a) The proposed perception



(b) KCF tracker [26]

(c) Re^3 tracker [25]

(d) Staple tracker [27]

Fig. 9. Target tracking result with various trackers. KCF tracker fails after a short occlusion, and Re^3, staple tracker fail after a long occlusion.

temporal regularized correlation filters for visual tracking," *arXiv preprint arXiv:1803.08679*, 2018.

[25] D. Gordon, A. Farhadi, and D. Fox, "Re 3: Real-time recurrent regression networks for visual tracking of generic objects," *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 788–795, 2018.

[26] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "High-speed tracking with kernelized correlation filters," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 3, pp. 583–596, 2015.

[27] L. Bertinetto, J. Valmadre, S. Golodetz, O. Miksik, and P. H. Torr, "Staple: Complementary learners for real-time tracking," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 1401–1409.

[28] X. Wang and A. Gupta, "Unsupervised learning of visual representations using videos," *arXiv preprint arXiv:1505.00687*, 2015.

[29] J. Wang and W. J. Wilson, "3d relative position and orientation estimation using kalman filter for robot control," in *Proceedings 1992 IEEE International Conference on Robotics and Automation*, May 1992, pp. 2638–2645 vol.3.

[30] M. Ficocelli and F. Janabi-Sharifi, "Adaptive filtering for pose estimation in visual servoing," in *Proceedings 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems. Expanding the Societal Role of Robotics in the the Next Millennium (Cat. No.01CH37180)*, vol. 1, Oct 2001, pp. 19–24 vol.1.

[31] F. Janabi-Sharifi and M. Marey, "A kalman-filter-based method for pose estimation in visual servoing," *IEEE Transactions on Robotics*, vol. 26, no. 5, pp. 939–947, Oct 2010.

[32] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.