Application of visual servoing and eye tracking glass in human robot interaction : a case study

Application of Visual Servoing and Eye Tracking Glass in Human Robot Interaction: A case study

Lei Shi
Op3Mech
University of Antwerp
Antwerp, Belgium
lei.shi@uantwerpen.be

Cosmin Copot
Op3Mech
University of Antwerp
Antwerp, Belgium
cosmin.copot@uantwerpen.be

Steve Vanlanduit
Op3Mech
University of Antwerp
Antwerp, Belgium
steve.vanlanduit@uantwerpen.be

***Abstract*—Human gazes reveal a lot of information about their intention, which could be used for intuitive Human Robot Interaction(HRI). Using gaze as a modality to interact with robots has huge value in increasing freedom for human. It has a strong application potential in different domains. In this paper, we propose a system for eye tracking based HRI with which a human can select an object by his/her gaze and a robot manipulator receives the intended object and waits for the confirmation from the human before it starts the grasping task. We use Image-based Visual Servoing (IBVS) to guide the robot manipulator to perform the grasping automatically which can also bring more robustness to the system. The first hand results indicate that the use of both visual servoing features and eye tracking technique can provide successful results in term of human robot interaction.**

***Index Terms*—Human Robot Interaction, eye tracking glasses, gaze, visual servoing**

I. Introduction

In the last years, Human Robot Interaction (HRI) has drawn a lot of interests. Different kinds of modalities have been proposed for HRI by researchers. Speech based HRI allows human to control robot by voice commands [1] [2]. Hand and body gestures also allow human to interact with robots. Different sensors are used to recognize gestures such as camera [3], IMU unit [4] and wearable device [5]. Eye tracking technology provides another modality for interacting with robots in a wide range of applications. For instance, to control UAV with gaze [6] [7], to control a robotics arm [8] and in shared autonomy [9]. One of the advantages to include eye tracking technology is the possibility to increase freedom in the sense that, for instance, when a human is doing a task and he or she needs to communicate with a robot for another task, the human doesn't have to stop the current task and interacting with robots for the other task. In [8], a camera is attached to a robotic manipulator to provide image in a Laparoscopic surgery. The surgeon could control the manipulator via eye gestures to move the camera while he or she is still performing the surgical operation. Using eye tracking device will also benefit the people with disability. In [10], upper-limb rehabilitation robot is developed which allows stroke patient to guide the movement of the exoskeleton robot during a rehabilitation session.

As part of the HRI scenario, actions need to be taken so that the robot can move. One of them is that robot performs movements according to predefined actions. For instance, a manipulator moves forward or backward when it receives a command in the form of gaze gestures [8]. A robot can also calculate and plan the path according to the destination appointed by human gaze and then move to the destination autonomously [7]. In human robot shared autonomy, the robot will consider the user input during autonomous movement [9].

Visual servoing could also be used as a mean of action. By using this technique, the robot will be kept guiding toward the desired pose for further action by using visual feedback. Based on the visual features used to define the control law the servoing architectures can be classified in two main categories named Position-based Visual Servoing (PBVS) and Image-based Visual Servoing (IBVS). In the case of PBVS architecture 3D information with respect to the working environment and the visual sensor are used to design the controller, while for the IBVS the control law is designing using visual features (2D information) extracted from the image. For further development in this paper, an IBVS control structure is being considered. The main goal of image-based visual servo is to find a control algorithm capable to drive the robot arm so that a set of image features, (composed of image coordinates of several Cartesian points belonging to the tracked target) will reach the desired value. The behaviour of any visual servoing scheme is influenced by two main aspects: the type of image features used to derive the control law and the form of the control law. The most common approach to generate the control signal for the robots is to use a classical proportional controller [11], [12].

In this paper, we propose an eye tracking based HRI system which enables the human be able to select an object by looking at the object and let the robot pick up the selected object automatically. We use a mobile eye tracking device mounted on the head of a human and a robotic manipulator using IBVS to robustly locate the grasp position. The paper is organized as follows: in section II, the related work is reviewed followed by the eye tracking based HRI method presented in section III. The experimental setup and results are given in section IV and conclusions are presented in section V.

II. Image based visual servoing control: general aspects

A. Visual Control Law

Image-based visual servoing (IBVS) is a methodology based on the use directly the image measurements as feedback to control the motion of a robot. The image based error

function (expressed as the robot's positioning task) needs to be minimized by employing an optimal control law. Given that the IBVS is not solving the Cartesian pose explicitly, the accuracy of the models does not influence the performance of the IBVS. Anuhow, as the control law is in image space, a direct control over the Cartesian or joint space trajectory of the robot end-effector cannot be defined.

In Fig. 1 a general schematic of an image-based control architecture of manipulator robots with 6 degrees of freedom is illustrated.

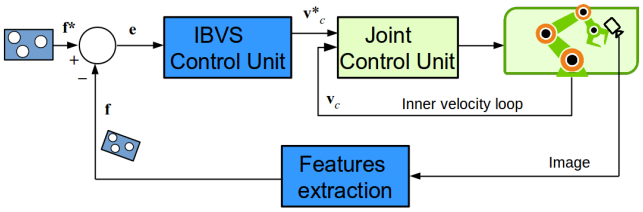


Fig. 1. Visual servoing architecture to control manipulator robots

The core of the architecture is represented by the image-based controller which requires apriori information related to the system behavior in order to minimize the error between the current configuration of the visual features $\mathbf{f}$ and the desired configuration of the features $\mathbf{f}^*$:

$$\mathbf{e} = \mathbf{f}\left(\mathbf{r}(t)\right) - \mathbf{f}^* \tag{1}$$

where $\mathbf{r}$ represents the relative position between the visual sensor and the object. If the visual features are defined with respect to the camera coordinate system, then the variation of visual features induced by the movement of the camera or movement of the objects can be correlated with the camera velocity:

$$\dot{\mathbf{f}} = \frac{\partial \mathbf{f}}{\partial \mathbf{r}}\frac{d\mathbf{r}}{dt} + \frac{\partial \mathbf{f}}{\partial t} = \mathbf{L_f}\mathbf{v}_c + \frac{\partial \mathbf{f}}{\partial t} \tag{2}$$

where $\frac{\partial \mathbf{f}}{\partial t}$ represents the time variation of the visual features with respect to the target motion (for static object $\frac{\partial \mathbf{f}}{\partial t} = 0$), $\mathbf{L_f}$ is the interaction matrix attach to $\mathbf{f}$ and $\mathbf{v}_c$ is the camera velocity with respect to the working space.

In a case of a static object ($\frac{\partial \mathbf{f}}{\partial t} = 0$), the relationship between the time variation of the error and the camera velocity is given by:

$$\dot{\mathbf{e}} = \mathbf{L_f}\mathbf{v}_c \tag{3}$$

The classical servoing control laws employs an exponential decrease of the error ($\dot{\mathbf{e}} = -\lambda\mathbf{e}$), thus the visual control law is defined as:

$$\mathbf{v}_c = -\lambda \mathbf{L_f^+}\mathbf{e} \tag{4}$$

where $\mathbf{L_f^+}$ is the Moore-Penrose pseudoinverse of $\mathbf{L_f}$.

The proportional controller is very easy to implement in IBVS applications, but the unsatisfactory behavior and the difficulty of constraint handling are the main drawbacks. The classical proportional control law generates a robot motion without considering the robot workspace, the joint limitations, or the visibility constraints. The critical issue can be solved by using advanced control algorithms. Among these algorithms, model predictive control had a real success in IBVS applications.

In order to overcome the problems of proportional controller, while keeping its simplicity implementation, an adaptive gain controller was proposed in [13]. The adaptive gain $\lambda(x)$ of the visual servoing control law is calculated using:

$$\lambda(x) = (\lambda_0 - \lambda_\infty)\, e^{-\frac{\lambda_s}{\lambda_0 - \lambda_\infty} x} + \lambda_\infty \tag{5}$$

where $x = \|\mathbf{e}\|$ is the norm of feature errors; λ_0 is the expected gain when $x = 0$, which represents the gain when $\|\mathbf{e}\|$ it is very close to the desired features; λ_∞ is the expected gain when $\|\mathbf{e}\|$ is towards infinity, which represents the gain at the beginning of the visual servoing. λ_s is the slope of λ when $\|\mathbf{e}\| = 0$. Fig. 2 illustrates the calculated values of gain λ versus the feature error with different adaptive gain parameters. λ is low while the norm of feature error is large and it is high while the norm of feature error is small which means close to convergence. The advantage of adaptive gain is that, it can avoid high output velocity at beginning of servoing and increasing the velocity while the robot is close to the desired pose.

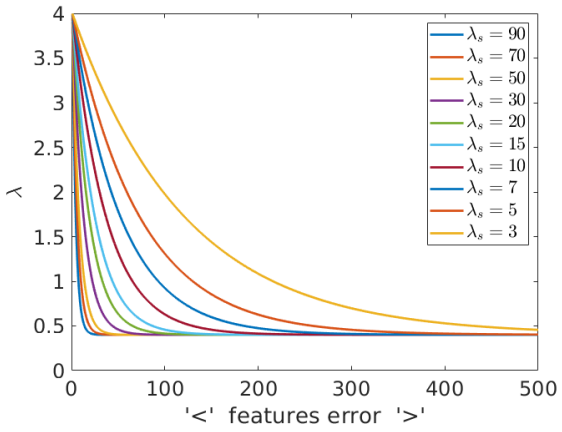


Fig. 2. Lambda profiles

Starting from the idea of adaptive controller presented in [13], an automatic tuning methodology for visual servoing system was presented in [14]. This methodology is based on predictive model approach to enable an automatic selection of control parameter (the adaptive gain) for the proportional visual control law.

B. Visual Features

The input for the visual control architecture is represented by the visual features extracted from images acquired using visual sensors. These visual features are of importance in terms of performance and accuracy of the servoing system and therefore their selection is a key aspect in visual servoing systems. The minimum number of visual features used in the control architecture in order to drive the movement of a robot manipulator depends on the number of degrees of freedom of the robot. As such is required to have a correlation between the visual features and the motion of the visual sensor. The most common visual features which can be employed as input to the control architecture are geometric features.

Geometric features are: i) 2D - used to described the geometric contend of an working area and ii) 3D - used to perform the correlation between the coordinate axes attach to

the robot and the coordinate axes attach to the object. Both, 2D features and 3D features, can be used simultaneously in designing the control architecture and thus a hybrid control architecture can be obtained. The 2D visual features are extracted from the image plane and represents the coordinates of the point features, parameters that defined lines or ellipse, regoin of interest, contours [15]. In order to extract the point features, algorithms such as Harris operator and SIFT descriptor can be employed.

If we consider a point feature $\mathbf{x} = (x, y)^T$, with:

$$\begin{cases} \dot{x} = -v_x/Z + xv_z/Z + xy\omega_x - (1+x^2)\omega_y + y\omega_z \\ \dot{y} = -v_y/Z + yv_z/Z + (1+y^2)\omega_x - xy\omega_y - x\omega_z \end{cases} \tag{6}$$

the interaction matrix for a 2D point $\mathbf{x}$ of coordinates (x, y) can be calculated by:

$$\mathbf{L_x} = \begin{bmatrix} -1/Z & 0 & x/Z & xy & -(1+x^2) & y \\ 0 & -1/Z & y/Z & 1+y^2 & -xy & -x \end{bmatrix} \tag{7}$$

where Z is the depth of the point relative to the camera frame.

Another type of visual features that can be used in visual servoing applications are image moments [11], [12]. Using image moments as visual features, significant improvements of the performance of servoing systems occur due to the fact that these visual features allows a general representation of the image and also allows the description of complex object. Another advantage of these features is that it can be used to design decoupled servoing system and in the same time to minimize the non-linearities introduced by the interaction matrix. Considering the image function $I(x, y)$ as a probability density of a 2D random variable and assuming that the non-zero pixel values represent the object (image regions), the image moments m_{ij} of order $(i+j)$ are defined as:

$$m_{ij} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} x^i y^j I(x, y)dxdy \tag{8}$$

Assuming that an object in the image I is described by a set of n point features of coordinates (x, y) with $I(x, y) = 1$, then image moments m_{ij} of order $(i + j)$ are defined as:

$$m_{ij} = \sum_{k=1}^{n} x_k^i y_k^j \tag{9}$$

whereas the centered moments μ_{ij} of order $(i + j)$ are calculated as:

$$\mu_{ij} = \sum_{k=1}^{n} (x_k - x_g)^i (y_k - y_g)^j \tag{10}$$

where $x_g = \frac{m_{10}}{m_{00}}$ and $y_g = \frac{m_{01}}{m_{00}}$ represent the coordinate of the gravity center attached to the object.

Considering a visual servoing application, a set of image moments $\mathbf{f}_m = [x_n, y_n, a_n, \gamma, \delta, \alpha]$ can be used to design the image-based control law. The first three components of $\mathbf{f}_m$ are used to control the linear components of the camera velocity $\mathbf{v}_c$ and are defines as:

$$a_n = Z * \sqrt{\frac{a*}{a}}, \quad x_n = a_n x_g, \quad y_n = a_n y_g \tag{11}$$

where Z^* is the desired depth between the visual sensor and the desired configuration, $a = \mu_{20} + \mu_{02}$ is the object area [12] and a^* is the desired object area.

The angular camera velocities are controlled using the last three components of $\mathbf{f}_m$ which are calculated as [12]:

$$\gamma = \frac{I_{n1}}{I_{n3}}, \quad \delta = \frac{I_{n2}}{I_{n3}}, \quad \alpha = \frac{1}{2}\arctan\left(\frac{2\mu_{11}}{\mu_{20} - \mu_{02}}\right) \tag{12}$$

where

$$\begin{aligned} I_{n1} &= (\mu_{50} + 2\mu_{32} + \mu_{14})^2 + (\mu_{05} + 2\mu_{23} + \mu_{41})^2 \\ I_{n2} &= (\mu_{50} - 2\mu_{32} - 3\mu_{14})^2 + (\mu_{05} - 2\mu_{23} - 3\mu_{41})^2 \\ I_{n3} &= (\mu_{50} - 10\mu_{32} + 5\mu_{14})^2 + (\mu_{05} - 10\mu_{23} + 5\mu_{41})^2 \end{aligned} \tag{13}$$

The analytical form of the interaction matrix for the general case is given in [6]. For the discrete case the time derivative of image moments μ_{ij} is obtained from differentiation of (2):

$$\begin{aligned} \dot{\mu}_{ij} = \sum_{k=1}^{n} & i(x_k - x_g)^{i-1}(y_k - y_g)^j(\dot{x}_k - \dot{x}_g) \\ & + j(x_k - x_g)^i(y_k - y_g)^{j-1}(\dot{y}_k - \dot{y}_g) \end{aligned} \tag{14}$$

Using (14), the interaction matrix of the centred moments μ_{ij} is obtained

$$\mathbf{L}_{\mu_{ij}} = [\mu_{vx} \quad \mu_{vy} \quad \mu_{vz} \quad \mu_{\omega x} \quad \mu_{\omega y} \quad \mu_{\omega z}] \tag{15}$$

The detailed structure of the parameters from previous equation can be found in [12]. The interaction matrix $\mathbf{L}_{\mathbf{f}m}$ for a set of image moments $\mathbf{f}_m$ can be calculated as:

$$\mathbf{L}_{\mathbf{f}m} = \begin{bmatrix} -1 & 0 & 0 & a_n e_{11} & -a_n(1+e_{12}) & y_n \\ 0 & -1 & 0 & a_n(1+e_{21}) & -a_n e_{11} & -x_n \\ 0 & 0 & -1 & -e_{31} & e_{32} & 0 \\ 0 & 0 & 0 & \gamma_{\omega x} & \gamma_{\omega y} & 0 \\ 0 & 0 & 0 & \delta_{\omega x} & \delta_{\omega y} & 0 \\ 0 & 0 & 0 & \alpha_{\omega x} & \alpha_{\omega y} & -1 \end{bmatrix} \tag{16}$$

The components of the interaction matrix are calculated as described in [12].

III. Eye Tracking Based HRI

Fig. 3 indicates the scene of the HRI. The human is wearing a pair of eye tracking glasses which has a camera to observe the world. A manipulator robot is equipped with a camera which is attached to the end effector of the manipulator. Both the human and the robot are observing the same scene. A human will assign an object for the robot by looking at it and the robot will pick up the object.

Eye gaze contains a lot of information about the human intention. One approach to model the human intention is classifying raw gaze data into different gaze events. The simplest classification will recognize Fixation and Saccade. The most common algorithms to identify fixation include Velocity-Threshold Identification(I-VT) and Dispersion-Threshold Identification(I-DT) [16]. I-DT classifies the fixation based on the location of the gaze point. For a set of consecutive gaze points, the dispersion, which is the sum of the maximum differences of the coordinates of the gaze points in x and y direction, is calculated. A fixation event f is considered recognized if the total dispersion is

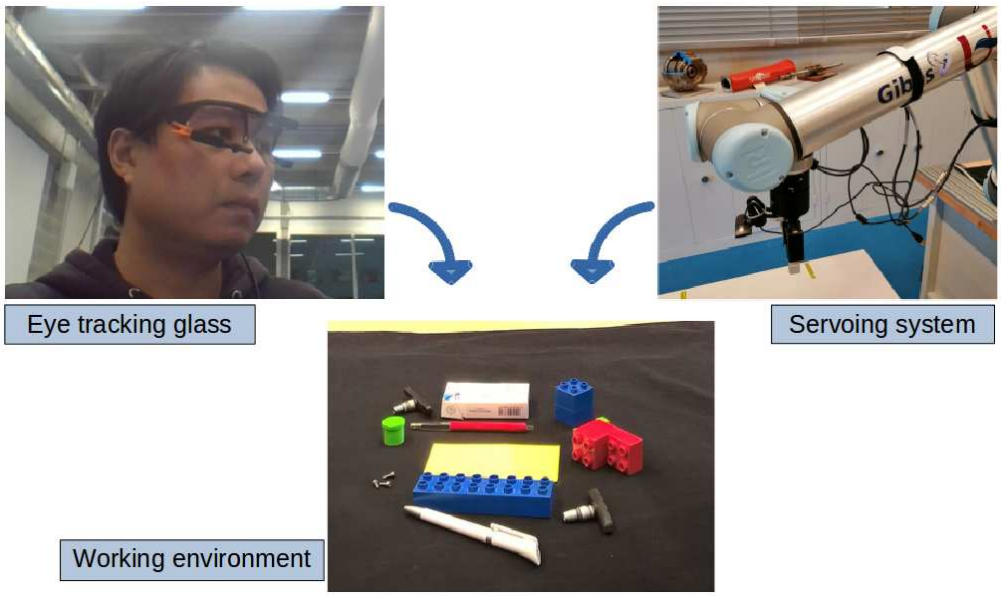


Fig. 3. Overview of the an eye tracking based HRI scenario.

below the dispersion threshold, otherwise it is not considered as a fixation. I-VT simply detect fixation events according to the velocity of the gaze. It is based on the characteristic that the gaze velocity during a saccade is much higher than the velocity during a fixation. A velocity threshold is used to classify fixation and saccade.

More recently, other approaches are proposed to detect eye movement events. In [17], Bayesian Decision theory is used to detect fixation, sacadde and smooth pursuit on the basis that the three events have different gaze velocity patterns. Machine learning is another track of research for identifying eye movement events. In [18], random forest algorithm is used by the authors to identify different gaze events including fixation and saccade with different gaze data acquired by different device fps and with different noise levels.

In [19], the authors propose a solution to infer the car driver's intention during driving. They assume that the driver's intention always locates on one of the detected interesting points in the scene and they use Markov Random Field as inference model which considering both current and past gaze points and interest points. An energy function is built so that the intention is referred by minimizing it.

IV. Experiment And Result

This section presents the experimental results that were conducted in order to illustrate the synergy between the eye tracking glass and visual servoing in the context of HRI . In the conducted experiments, a mobile eye tracking device for eye tracking system was used. I-DT is used for detecting fixations, all other eye movements are considered as saccades. An object in the scene is selected if the gaze is fixating within the bounding box of the object. After the object is selected, the robot manipulator uses IBVS to go to the pose for grasping.

Pupil Labs eye tracking glasses [20] is used for eye tracking, while for I-DT fixation detection the algorithms implemented by Pupil Labs were considered. The object manipulation is done using a Universal Robot UR10 [21] and the visual servoing task is implemented using the adaptive control law presented in [22]. In order to communicate with the UR10 robot, the Robotics System Toolbox of Matlab® and the ROS platform were used.

Fig. 4 shows the results of selecting the object by fixation. On the left is the image acquired by the scene camera of eye tracking glasses. The yellow circle represents the gaze point. On the right is the image captured from the camera mounted on the end effector of the manipulator. The green solid circle represents the projected gaze from the image on the left and the green bounding box represents the detected object.

Fig. 5 shows the feature errors of the IBVS. The IBVS starts when an object is selected and it takes 290 iterations for the errors to converge. Fig. 6 shows the camera output velocity, the translational camera velocity and the rotational camera velocity.

V. Conclusion

In this paper the hypothesis of using eye tracking based HRI to send a command to the robot to execute a certain task was presented. The idea is that the human can select an object by his/her gaze then the information is received by the manipulator robot and once the human confirms the robot can take action and grasp the selected object. To achieve the goal of our investigation the image based visual servoing technique has been employed. The first hand results indicate that the use of both visual servoing features and eye tracking technique can provide successful results in terms of human robot interaction.

References

[1] R. Paul, J. Arkin, N. Roy, and T. M Howard, "Efficient grounding of abstract spatial concepts for natural language interaction with robot manipulators," 2016.

[2] J. Thomason, S. Zhang, R. J. Mooney, and P. Stone, "Learning to interpret natural language commands through human-robot dialog," in *Twenty-Fourth International Joint Conference on Artificial Intelligence*, 2015.

Fig. 4. Gaze position of a fixation event.

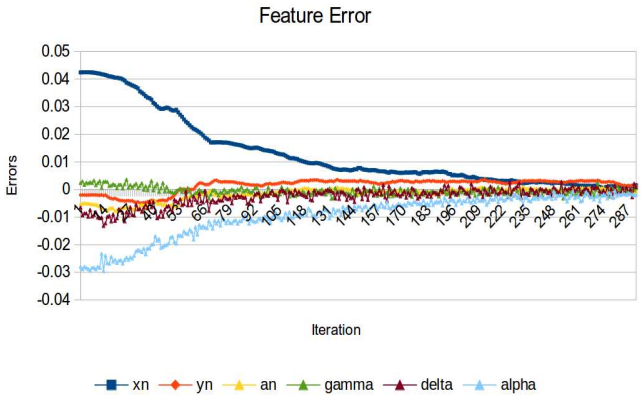


Fig. 5. Time evolution of image feature errors.

[3] O. Mazhar, S. Ramdani, B. Navarro, R. Passama, and A. Cherubini, "Towards real-time physical human-robot interaction using skeleton information and hand gestures," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 1–6.

[4] P. Neto, M. Simão, N. Mendes, and M. Safeea, "Gesture-based human-robot interaction for human assistance in manufacturing," *The International Journal of Advanced Manufacturing Technology*, vol. 101, no. 1-4, pp. 119–135, 2019.

[5] B. Gromov, L. M. Gambardella, and G. A. Di Caro, "Wearable multi-modal interface for human multi-robot interaction," in *2016 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*. IEEE, 2016, pp. 240–245.

[6] M. Yu, Y. Lin, D. Schmidt, X. Wang, and Y. Wang, "Human-robot interaction based on gaze gestures for the drone teleoperation," *Journal of Eye Movement Research*, vol. 7, no. 4, pp. 1–14, 2014.

[7] L. Yuan, C. Reardon, G. Warnell, and G. Loianno, "Human gaze-driven spatial tasking of an autonomous mav," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1343–1350, 2019.

[8] K. Fujii, G. Gras, A. Salerno, and G.-Z. Yang, "Gaze gesture based human robot interaction for laparoscopic surgery," *Medical image analysis*, vol. 44, pp. 196–214, 2018.

[9] 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.

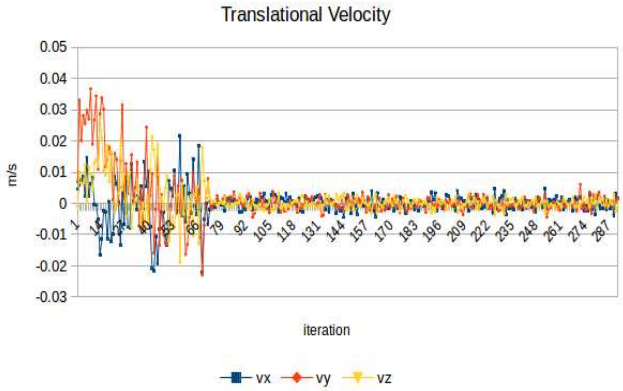
[10] Q. Li, C. Xiong, and K. Liu, "Eye gaze tracking based interaction method of an upper-limb exoskeletal rehabilitation robot," in *International Conference on Intelligent Robotics and Applications*. Springer, 2017, pp. 340–349.

[11] F. Chaumette and S. Hutchinson, "Visual servo control, part I: Basic approaches," *IEEE Robotics and Automation Magazine*, vol. 13, no. 4, pp. 82–90, 2006.

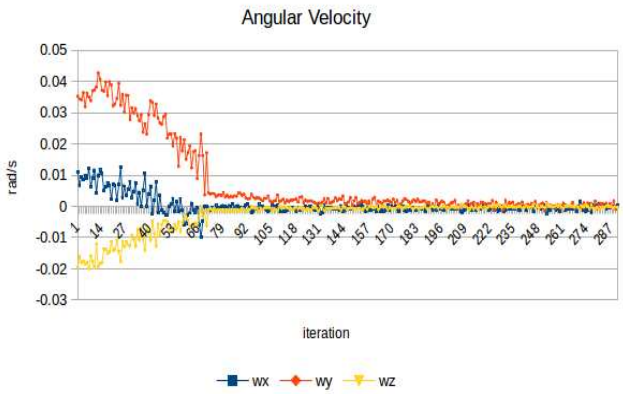
[12] O. Tahri and F. Chaumette, "Point-based and region-based image moments for visual servoing of planar objects," *IEEE Transaction on Robotics*, vol. 21, no. 6, pp. 1116–1127, 2005.

[13] E. Marchand, F. Spindler, and F. Chaumette, "Visp for visual servoing a generic software platform with a wide class of robot control skills," *IEEE Robotics and Automation Magazine*, pp. 40–52, 2005.

[14] C. Copot, L. Shi, and S. Vanlanduit, "Automatic tuning methodology of visual servoing system using predictive approach," *IEEE International Conference on Control and Automation*, 2019.



(a)



(b)

Fig. 6. Evolution of translational and rotational camera velocity.

[15] N. Gans, S. Hutchinson, and P. Corke, "Performance tests for visual servo control systems, with application to partitioned approaches to visual servo control," *International Journal of Robotics Research*, vol. 22, no. 10, pp. 955–981, 2003.

[16] D. D. Salvucci and J. H. Goldberg, "Identifying fixations and saccades in eye-tracking protocols," in *Proceedings of the 2000 symposium on Eye tracking research & applications*. ACM, 2000, pp. 71–78.

[17] T. Santini, W. Fuhl, T. Kübler, and E. Kasneci, "Bayesian identification of fixations, saccades, and smooth pursuits," in *Proceedings of the Ninth Biennial ACM Symposium on Eye Tracking Research & Applications*. ACM, 2016, pp. 163–170.

[18] R. Zemblys, D. C. Niehorster, O. Komogortsev, and K. Holmqvist, "Using machine learning to detect events in eye-tracking data," *Behavior research methods*, vol. 50, no. 1, pp. 160–181, 2018.

[19] Y.-S. Jiang, G. Warnell, and P. Stone, "Inferring user intention using gaze in vehicles," in *Proceedings of the 2018 on International Conference on Multimodal Interaction*. ACM, 2018, pp. 298–306.

[20] M. Kassner, W. Patera, and A. Bulling, "Pupil: An open source platform for pervasive eye tracking and mobile gaze-based interaction," in *Adjunct Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, ser. UbiComp '14 Adjunct. New York, NY, USA: ACM, 2014, pp. 1151–1160. [Online]. Available: http://doi.acm.org/10.1145/2638728.2641695

[21] T. T. Andersen, "Optimizing the universal robots ros driver." Technical University of Denmark, Department of Electrical Engineering, Tech. Rep., 2015. [Online]. Available: http://orbit.dtu.dk/en/publications/optimizing-the-universal-robots-ros-driver(20dde139-7e87-4552-8658-dbf2cdaab24b).html

[22] E. Marchand, F. Spindler, and F. Chaumette, "Visp for visual servoing: a generic software platform with a wide class of robot control skills," *IEEE Robotics and Automation Magazine*, vol. 12, no. 4, pp. 40–52, December 2005.