HyperPalm: DNN-based hand gesture recognition interface for intelligent communication with quadruped robot in 3D space

Elena Nazarova  
*Intelligent Space Robotics Laboratory*  
*Skoltech*  
Moscow, Russian Federation  
elena.nazarova@skoltech.ru

Ildar Babataev  
*Intelligent Space Robotics Laboratory*  
*Skoltech*  
Moscow, Russian Federation  
ildar.babataev@skoltech.ru

Nipun Weerakkodi  
*Intelligent Space Robotics Laboratory*  
*Skoltech*  
Moscow, Russian Federation  
nipun.weerakkodi@skoltech.ru

Aleksey Fedoseev  
*Intelligent Space Robotics Laboratory*  
*Skoltech*  
Moscow, Russian Federation  
aleksey.fedoseev@skoltech.ru

Dzmitry Tsetserukou  
*Intelligent Space Robotics Laboratory*  
*Skoltech*  
Moscow, Russian Federation  
d.tsetserukou@skoltech.ru

Abstract—Nowadays, autonomous mobile robots support people in many areas where human presence either redundant or too dangerous. They have successfully proven themselves in expeditions, gas industry, mines, warehouses, etc. However, even legged robots may stick in rough terrain conditions requiring human cognitive abilities to navigate the system. While gamepads and keyboards are convenient for wheeled robot control, the quadruped robot in 3D space can move along all linear coordinates and Euler angles, requiring at least 12 buttons for independent control of their DoF. Therefore, more convenient interfaces of control are required.

In this paper we present HyperPalm: a novel gesture interface for intuitive human-robot interaction with quadruped robots. Without additional devices, the operator has full position and orientation control of the quadruped robot in 3D space through hand gesture recognition with only 5 gestures and 6 DoF hand motion.

The experimental results revealed to classify 5 static gestures with high accuracy (96.5%), accurately predict the position of the 6D position of the hand in three-dimensional space. The absolute linear deviation Root mean square deviation (RMSD) of the proposed approach is 11.7 mm, which is almost 50% lower than for the second tested approach, the absolute angular deviation RMSD of the proposed approach is 2.6 degrees, which is almost 27% lower than for the second tested approach. Moreover, the user study was conducted to explore user’s subjective experience from human-robot interaction through the proposed gesture interface. The participants evaluated their interaction with HyperPalm as intuitive (2.0), not causing frustration (2.63), and requiring low physical demand (2.0).

I. INTRODUCTION

Legged robots are increasingly being applied in the indoor and outdoor scenarios of exploration, delivery, and interaction with humans [1]. While achieving lower velocities on planar surfaces compared with wheeled mobile robots, they have the potential to locomote on irregular terrains and navigate in cluttered environments. For example, Hiller et al. [2] presented a quadruped robot design and control for an unstructured environment. Barasuol et al. [3] introduced a reactive controller for quadrupedal locomotion on challenging terrain. Self-organized locomotion with neural control was explored by Sun et al. [4], where researchers demonstrated a successful simulation of a legged robot on flat terrain and in the presence of low-height obstacles.

Static gait for quadruped robots walking on uneven ter-

Fig. 1. (a) CAD design of the HyperPalm Robot. (b) DNN-based gesture recognition of the operator’s hand. (c) HyperPalm Robot position and orientation control.
rains, such as stairs, was investigated by Li et al. [5] and Ye et al. [6]. The researchers achieved high results in the simulation of passability for the developed quadruped robots. The stair-climbing robot dog was also proposed by Campos et al. [7], where the robot utilized CNN for both object detection and hand gesture recognition as part of its HRI strategy. Saputra et al. [8] proposed an adaptive quadruped robot inspired by domestic felines, that was supporting advanced terrain climbing. Thus, the versatility of legged robots and quadruped robots, in particular, could potentially allow them to support several crucial tasks, in addition to the exploration and industrial e.g., navigation of people with sight disadvantages, explored with the guide robot dogs developed by Chuang et al. [9] and Xiao et al. [10], or promote social distancing in crowded urban environments in the COVID-19 pandemic situation, as suggested by Chen et al. [11].

However, the live motion control of a four-legged robot is still a significant challenge in the robotic control field. Hence, many successful attempts of their automotive motion are by this day only demonstrated in simulations, while navigation of robots in the dynamic environment is supported by human operators.

II. RELATED WORKS

With the emergence of the CNN-based and DNN-based approaches, the visual recognition of hand gestures has been significantly improved, finding their application in various scenarios of remote control. For example, a multi-sensor system for recognition of the driver’s hand motion was suggested by Molchanov et al. [12], where the RGBD camera, and near-field radar data were combed for higher stability. Stancic et al. [13] proposed an inertial-based wearable system, that can apply hand-gesture dynamics for robotic control over high distances. Wearable gesture interfaces based on flex sensors were suggested by Afzal et al. [14] for the control of robotic end-effector with four fingers and by Fedoseev et al. [15] for the 6 DoF robotic arm control in drone catching task.

Moreover, many works are currently devoted to the study of the interaction between a human and a swarm of unmanned aerial vehicles (UAV) and mobile robots through an interface based on gesture recognition. For example, a gesture interface developed Tsykunov et al. [16] based on impedance swarm control and haptic feedback for HRI. Chen et al. [17] proposed the HRI multichannel robotic system in augmented reality (AR). A control approach with human hand arm and motions, which are recorded by a wearable armband, controlling a swarm’s shape and formation was suggested by Suresh et al. [18]. Alonso-Mora et al. [19] and Kim et al. [20] suggested real-time input interfaces with swarm formation control. However, their approach was developed only for mobile robot operation in 2D space.

III. GENERAL SYSTEM OVERVIEW

The developed HyperPalm system includes two Intel Realsense d435 RGB-D cameras, NVIDIA Jetson Nano onboard personal computer (PC) and operator PC located remotely, and HyperPalm robot itself with low-level communications (Fig. 2).

The first camera is located on the robot and captures the RGB image of its front view. This image is used for the operator to have understanding of the robot localization in the environment. The system is aimed at solving the problem of teleoperation when the robot performs the task remotely, and it cannot achieve the help from operator. Therefore, the operator needs visual feedback from the robot. The second camera captures the RGB-D image of the operator’s hand. Both cameras send their output to the operator’s PC. The robot’s camera uses a wireless connection via open-source robotics middleware suite Robot Operating System (ROS). The operator camera uses a wired connection via a Universal Serial Bus (USB) cable. The operator’s PC processes the input data through the hand gesture module, which will be described in the Hand Gesture Module Overview section. This module sends control data to the onboard computer NVIDIA Jetson Nano via ROS. Next, the onboard computer processes the input data from gesture module in the control system module and sends movement commands to the robot itself.

A. HyperPalm Robot Design

HyperPalm was designed and assembled using light 3D-printed and carbon fiber parts. Moreover, instead of the high-cost brushless direct current (DC) motors it utilizes DC servo motors, which allow HyperPalm to accomplish a high-precision movement. The leg motion is supported by the dampers, which can be seen in Fig. 3 to achieve a smooth transition of its feet on slippery and uneven surface.

HyperPalm is a 12-DoF legged robot with (WxHxD: 300x175x240mm) external dimension. Each leg of the robot consists of 3 joints for hip, upper and lower legs. This helps to achieve a wide range of capabilities for robot movements. The robot itself has 5 kg weight and 2 kg pay-load capability. The robot is powered by an 8.4V and 8.8Ah Li-Ion battery pack, which allows the robot to run for about 30 minutes on a charge.
IV. HAND GESTURE MODULE OVERVIEW

A. Gesture Recognition

To switch the operating modes of the robot, proposed to use 5 static gestures: "One", "Two", "Three", "Open", and "Close". This is necessary so that when the system is turned on and the operator’s hands or others fall into the camera field of view, the robot does not react to the given changes in the position and orientation of the hand without specific run command.

We use DNN-based gesture recognition to achieve high precision in human gesture classification (Fig. 4). DNN is a fully connected network with four layers: input layer, two hidden layers, output layer. Moreover, between layers the network has Rectified Linear Unit (ReLU) non-linear function, and batch normalization. The input layer takes 62 neurons, the first hidden layer has 256 neurons, the second hidden layer has 128, the last third output layer has 5 neurons with probabilities for each gesture class.

![DNN model for gestures classification.](image)

A gesture dataset for the model training consists of five gestures of 2500 arrays per each gesture. In total 12500 arrays with normalized X, Y coordinates of 21 hand landmarks and 20 normalized vectors length between 21 hand landmarks. We divided dataset on train and test sets with a ratio of 75% (9375 arrays) and 25% (3125 arrays), respectively. It resulted in accuracy of 96.5% when performing validation on a test set, which is shown in Fig. 5.

The model is invariant of the position and orientation of the human hand. If the probability of a gesture is less than 85%, the algorithm takes this predicted gesture as the "None".

B. Hand 6D Pose Estimation

Determining the position of three linear coordinates of the hand, implemented by calculating relative translation of a hand in 3D space. Foremost, the algorithm defines a point in a space relative to which the deviation is calculated. The 21 x and y coordinates of landmarks clearly receives from MediaPipe framework in pixels, which are converted to meters. The information obtains about the z coordinate for each landmark of the hand from a pre-calibrated depth image in meters. Further, the algorithm calculates the 3D center of mass and analyzes its linear movement in space. It is necessary to obtain the entire hand and not 21 landmarks isolated.

The algorithm for calculating hand Euler angles consists of three parts. Firstly, we extract x, y, z coordinates of 21 hand landmarks as for determining the position of three linear coordinates. In addition, we implemented an algorithm to generate a point cloud of the hand via k-nearest neighbor approach. This is justified by the fact that the accuracy of expanded point cloud approximation to a plain is higher. On average, we increase the number of points in point cloud from 21 to 2000.

The next step is an approximation of the point cloud to a plane by least squares approach. The equation for a plane is shown in:

\[ ax + by + c = z \]  

where a, b, c are the coefficients of the scalar equation of the plane. Thus, we set up matrices with x, y, z coordinates for each point in point cloud as follows:

\[
\begin{bmatrix}
 x_0 & y_0 & 1 \\
 x_1 & y_1 & 1 \\
 \vdots & \vdots & \vdots \\
 x_n & y_n & 1 \\
\end{bmatrix}
\begin{bmatrix}
 a \\
 b \\
 c \\
\end{bmatrix}
= 
\begin{bmatrix}
 z_0 \\
 z_1 \\
 \vdots \\
 z_n \\
\end{bmatrix}
\]  

where n is the number of points in point cloud and is equal 2000 in our system. Since there are always more than three points in a point cloud, the system is over-determined.
Therefore, we used the left pseudo inverse matrix as given in:
\[
\begin{bmatrix}
a \\
b \\
c \\
\end{bmatrix} = (A^T A)^{-1} A^T B \tag{3}
\]
where \( A \) and \( B \) are the matrices described by:
\[
A = 
\begin{bmatrix}
x_0 & y_0 & 1 \\
x_1 & y_1 & 1 \\
\vdots & \vdots & \vdots \\
x_n & y_n & 1 \\
\end{bmatrix}, \tag{4}
\]
\[
B = 
\begin{bmatrix}
z_0 \\
z_1 \\
\vdots \\
z_n \\
\end{bmatrix} \tag{5}
\]

Then we calculated the Euler angles of the plane in \( x, y, z \) axes. The angle between two planes is equal to the angle determined by the normal vectors of the planes, as defined in:
\[
\text{Ang} = \cos^{-1}\left(\frac{a_1 a_2 + b_1 b_2 + c_1 c_2}{\sqrt{a_1^2 + b_1^2 + c_1^2}\sqrt{a_2^2 + b_2^2 + c_2^2}}\right) \tag{6}
\]
where \( a_1, b_1, c_1, \) and \( a_2, b_2, c_2 \) are the direction ratios of normal to the first and second planes respectively.

C. Control system to navigate HyperPalm

In the proposed system, the operator can control the linear coordinates and Euler angles separately or together by switching the control mode. The operator shows a gesture called "One" to activate linear control mode. Next, the operator should show the "Open" gesture to initiate the calculation of the relative hand movements in 3D space. Finally, the operator demonstrates the "Close" gesture to disable 3D hand control immediately. The Euler angles control of the robot is activated by performing the "Two" gesture, combined linear and angular by "Three" gesture. The further control procedure is the same as for the linear coordinates control, Fig. 6.

V. EXPERIMENT: HUMAN HAND 6D POSE ESTIMATION APPROACH SELECTION

To choose a more accurate approach for determining the six-dimensional position of the hand in space, two approaches were tested and validated. The first approach is based on a convolutional neural network (CNN), the second is the algorithm for calculating the 3D center of mass and approximating the hand point cloud to the plane, which was presented in the previous paragraph.

The standard Root-Mean-Square Deviation (RMSD) metric was used to validate these approaches. It was chosen because RMSD can be interpreted as an absolute error. The RMSD represents the square root of the second sample moment of the differences between predicted values and observed values or the quadratic mean of these differences:
\[
\text{RMSD} = \sqrt{\frac{\sum_{t=1}^{T}(\hat{y}_t - y_t)^2}{T}} \tag{7}
\]
where \( T \) is a number of non-missing data points, \( \hat{y}_t \) is estimated time series, and \( y_t \) is actual observations time series.

A. Yolo-based 6D hand pose estimation approach

First approach based on work Tekin et al. [21] for simultaneously detecting an object in an RGB image and predicting its 6D pose without requiring multiple stages or having to examine multiple hypotheses. The key component of this method is CNN architecture inspired by works Redmon et al. [22] and [23] that directly predicts the 2D image locations of the projected vertices of the object’s 3D bounding box. The object’s 6D pose is then estimated using a Perspective-n-Point (PnP) algorithm.

To evaluate the hand posture, we collected our own dataset based on LineMOD benchmark for 6D object pose estimation, which consists of 2500 sequence capture RGB-D images, a processed binary masks for each image and a 3D model of a hand. We used an open-source project to create masks, bounding box labels, and 3D reconstructed object mesh for object sequences captured with an RGB-D camera. Moreover, we implemented a raw 3D model acquisition through aruco markers and ICP registration pipeline and processed it manually. In addition, the dataset was increased by adding the background augmentation using 17000 random images from the Internet.

B. Proposed 6D hand pose estimation approach

Second approach is our method to estimate 6D hand pose based on the MediaPipe, depth maps and building point
cloud for further processing.

Since for this approach the data were not pre-marked in advance, there is no ground for calculating RMSD. We invited 6 participants to determine the approximate ground truth. Each participant disposed in front of the UR3 from Universal Robots robot with a statically directed hand towards the camera, which is located on the robot (Fig. 7). The robot moved along a pre-programmed trajectory along linear and angular axes, taken independently of each other. We discretely saved the linear and angular position of the robot and hand in millimeters (mm) and degrees. The repeatability of the UR3 is 0.1 mm. This is small enough to allow the robot’s trajectory to be used as an approximate ground truth.

Thus, we obtained the trajectory of the robot, which, in the calculation for RMSD, is considered as ground truth and the trajectory of the hand, which was calculated by the algorithm for determining the 6D position of the hand, Fig. 8.

C. Experimental Results

We used the standard RMSD to compare 6D pose estimation error of two approaches in mm for linear and degrees for angular coordinates.

The linear RMSD result for Yolo-based approach for all validation images in the best model the deviation is 23.6 mm. For our proposed approach for all participants who took part in the experiment the deviation is 11.7 mm, which is almost 50% less than for the Yolo-based approach.

The angular RMSD result for Yolo-based approach for all validation images in the best model the deviation is 3.6 degrees. For our proposed approach for all participants who took part in the experiment the deviation is 2.6 degrees, which is almost 27% less than for the Yolo-based approach.

According to RMSD result, our proposed approach has better performance to 6D hand pose estimation than Yolo-based approach. This is due to the fact that networks such as the Yolo-based depend on the shape changes of the object over time. We trained the algorithm for the human palm without specifically shape changing, but the human palm is exposed to external factors and its shape changes with time.

VI. USER EXPERIENCE EVALUATION

The goal of this paper is to evaluate the intuitive interaction between human and quadruped robot with the gesture recognition interface based on DNN. We provided an experiment to evaluate user experience with gesture-based control of HyperPalm system.

Participants: We invited 8 participants (3 females) aged 19 to 27 years (mean = 23.0) to test HyperPalm system. Three of them have never interacted with gesture interfaces before, others were familiar with CV-based gesture interfaces. We covered student from different professional tracks.

Procedure: Participants were asked to execute several commands for gesture control. Before testing, they had a short briefing about test procedure and commands description. After the experiment, each participant completed a questionnaire based on The NASA Task Load Index (NASA-TLX) and three specific extra questions which give information such as age, gender of the participant, and how intuitive it was to control the robot with DNN-based gestures. The participants performed a training session, where each participant familiarized themselves with live interaction between themselves and HyperPalm with Gesture-based Interface. They were required to test several random commands and after recover initial position of robot.

An extra “intuitiveness” parameter was introduced in the survey, being an essential criterion in the teleoperation tasks to provide adjustable control over the swarm behavior in real-time. Therefore, the participants provided feedback on seven questions.

Experimental Results

The results of the NASA-TLX based survey are shown in Fig. 9.

We conducted a chi-square analysis based on the frequency of answers in each category. The results showed that the parameters are all independent \( \min p = 0.12 > 0.05 \).

In summary, six of the participants found the experiment with HyperPalm system exciting, whereas five did not feel...
any discomfort during control robot. The results revealed that participants were fully engaged in the gesture-based control and seven of them were not tired of robot operation procedure. All participants did not feel any additional physical effort during the gesture control performance (mean 2.0 out of 5.0).

VIII. CONCLUSIONS AND FUTURE WORK

This paper introduced a novel DNN-based hand gesture recognition interface for intelligent communication with quadruped robot in 3D space. Where gesture recognition interface allows the operator to guide the quadruped robot using 5 gestures and changing the position of the palm in space.

The HyperPalm interface provides immersive and intuitive control to the user. Using HyperPalm the participants’ feedback show that the interaction was mostly intuitive with a low frustration level (2.63 out of 5.0) and low physical demand (2.0 out of 5.0). The proposed human hand 6D pose estimation algorithm achieved the best results compare with Yolo-based 6D hand pose estimation approach. The linear RMSD is almost 50% less, the angular RMSD is almost 27% less.

In the future, we will provide the additional experiment with a scenario close to the real industrial environment. In this experiment, participants will be asked to navigate robot along a particular trajectory with several existing control interfaces and the developed the HyperPalm interface. Gesture recognition in various lighting levels and visual noise will be also experimentally evaluated.

REFERENCES

[1] P. Biswal and P. Mohanty, “Development of quadruped walking robots: A review,” Ain Shams Engineering Journal, vol. 12, 12 2020.
[2] M. Hiller, D. Germann, and J. Morgado de Goes, “Design and control of a quadruped robot walking in unstructured terrain,” in Proceedings of the 2004 IEEE International Conference on Control Applications, 2004., vol. 2, 2004, pp. 916–921 Vol.2.
[3] V. Barasuol, J. Buchli, C. Semini, M. Frigerio, E. R. De Pieri, and D. G. Caldwell, “A reactive controller framework for quadrupedal locomotion on challenging terrain,” in 2013 IEEE International Conference on Robotics and Automation, 2013, pp. 2554–2561.
[4] T. Sun, D. Shao, Z. Dai, and P. Manoonpong, “Adaptive neutral control for self-organized locomotion and obstacle negotiation of quadruped robots,” in 2018 27th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN), 2018, pp. 1081–1086.
[5] X. Li, H. Gao, J. Li, Y. Wang, and Y. Guo, “Hierarchically planning static gait for quadruped robot walking on rough terrain,” Journal of Robotics, vol. 2019, pp. 1–12, 05 2019.
[6] L. Ye, Y. Wang, X. Wang, H. Liu, and B. Liang, “Optimized static gait for quadruped robots walking on stairs,” in 2021 IEEE 17th International Conference on Automation Science and Engineering (CASE), 2021, pp. 921–927.
[7] G. Campos, D. Pozza, M. Reyes, A. Zacate, H. Ponce, J. Brieva, and E. Moya-Albor, “Stair climbing robot based on convolutional neural networks for visual impaired,” in 2019 International Conference on Mechatronics, Electronics and Automotive Engineering (ICMEAE), 2019, pp. 108–113.
[8] A. A. Saputra, N. Takesue, K. Wada, A. J. Ijspeert, and N. Kabota, “Aquro: A cat-like adaptive quadruped robot with novel bio-inspired capabilities,” Frontiers in Robotics and AI, vol. 8, p. 35, 2021. [Online]. Available: https://www.frontiersin.org/article/10.3389/frobi.2021.562524
[9] T.-K. Chiung, N.-C. Lin, J.-S. Chen, C.-H. Hung, Y.-W. Huang, C. Teng, H. Huang, L.-F. Yu, L. Giarré, and H.-C. Wang, “Deeptrail-following robotic guide dog in pedestrian environments for people who are blind and visually impaired - learning from virtual and real worlds,” in 2018 IEEE International Conference on Robotics and Automation (ICRA), 2018, pp. 5840–5855.
[10] A. Xiao, W. Tong, L. Yang, J. Zeng, Z. Li, and K. Sreenath, “Robotic guide dog: Leading a human with leapsh-guided hybrid physical interaction,” in 2021 IEEE International Conference on Robotics and Automation (ICRA), 2021, pp. 11 470–11 476.
[11] Z. Chen, T. Pan, X. Zhao, J. Liang, C. Shen, H. Chen, D. Manocha, J. Pan, and W. Zhang, “Autonomous social distancing in urban environments using a quadruped robot,” IEEE Access, vol. 9, pp. 8392–8403, 2021.
[12] M. Molchanov, S. Gupta, K. Kim, and K. Pulli, “Multi-sensor system for driver’s hand-gesture recognition,” in 2015 11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG), vol. 1, 2015, pp. 1–8.
[13] I. Stančič, J. Musić, and T. Grujić, “Gesture recognition system for robotic mobile robot control based on inertial sensors and motion strings,” Engineering Applications of Artificial Intelligence, vol. 66, pp. 33–48, 2017. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0952179717301975
[14] W. Afzal, “Gesture control robotic arm using flex sensor,” Applied and Computational Mathematics, vol. 6, p. 171, 01 2017.
[15] A. Fedoseev, V. Serpiva, E. Karmanova, M. A. Cabrera, V. Shirokun, I. Vasiliev, S. Savushkin, and D. Tssetserukou, “Dronetraper: Drone catching in midair by soft robotic hand with color-based feature detection and hand gesture recognition;” in 2021 IEEE 4th International Conference on Soft Robotics (RoboSoft), 2021, pp. 261–266.
[16] E. Tsykunov, R. Agishev, R. Ibrahimov, L. Labazanova, A. Tleugazy, and D. Tssetserukou, “Swarmtouch: Guiding a swarm of micro-quadrotors with impedance control using a wearable tactile interface,” IEEE Transactions on Haptics, vol. 12, no. 3, pp. 363–374, 2019.
[17] M. Chen, P. Zhang, Z. Wu, and X. Chen, “A multichannel human-swarm robot interaction system in augmented reality,” Virtual Reality & Intelligent Hardware, vol. 2, no. 6, pp. 518–533, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0952179719300905
[18] A. Suresh and S. Martínez, “Gesture based human-swarm interactions for formation control using interpreters,” IFAC-PapersOnLine, vol. 51, no. 34, pp. 83–88, 2019, 2nd IFAC Conference on Cyber-Physical and Human Systems CPHS 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2405896319300357
[19] J. Alonso-Mora, S. Haegele Lohaus, P. Leemann, R. Siegwart, and P. Beardsley, “Gesture based human - multi-robot swarm interaction and its application to an interactive display,” in 2015 IEEE International Conference on Robotics and Automation (ICRA), 2015, pp. 5948–5953.
[20] L. H. Kim, D. S. Drew, V. Domova, and S. Pollner, “User-defined swarm robot control,” in Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems, ser. CHI ’20. New York, NY, USA: Association for Computing Machinery, 2020, p. 1–13. [Online]. Available: https://doi.org/10.1145/3313831.3376814
[21] B. Tekin, S. N. Sinha, and P. Fu, “Real-time seamless single shot 6d object pose prediction,” in 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018, pp. 292–301.
[22] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: Unified, real-time object detection,” 06 2016, pp. 779–788.
[23] J. Redmon and A. Farhadi, “Yolo9000: Better, faster, stronger,” in 2018 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 6517–6525.