IMAV2017-18
https://www.imavs.org/pdf/imav.2017.18

An Intelligent Unmanned Aircraft System for Wilderness Search and Rescue

Huai Yu*, Shijie Lin, Jinwang Wang, Kaimin Fu, Wen Yang
School of Electronic Information, Wuhan University, No.299 Bayi Road, Wuhan 430072, China
CETC key laboratory of aerospace information applications, Shijiazhuang 050081, China

ABSTRACT

In this paper, we presented a wilderness search and rescue (WiSAR) system based on DJI M100 Unmanned Aerial Vehicle (UAV) and a ground station to search and rescue the survivors in wild. We combined infrared and optical target detection to increase the detection speed and accuracy and used multiple sensors to make this system can autonomous avoiding obstructions and landing on mobile platform. For further increase the Average Precision of SSD, we build a field people dataset UAV-PP and use ResNet-101 as the base net. The actual flying test have been conducted in multiple situations to verify the feasibility of our WiSAR system. Our WiSAR system laying a solid foundation for building a more intelligent search and rescue system based on UAV.

1 INTRODUCTION

Wilderness search and rescue (WiSAR) is very necessary and difficult due to its vast territory and frequent field disasters. Generally, WiSAR is racing with time, every second counts. Search and rescue operations usually need a lot of manpower and resources. Traditional rescue methods, like human-base search, are inefficient and can easily miss the best rescue time. In recent years, the rapid development of Unmanned Ariel Vehicles (UAV) provides another better way for rapid search and rescue. UAV equipped with image acquisition cameras and a variety of sensors, transport the obtained video to the ground station. In addition, the UAV is agile, flexible, and can perform actions that are difficult to perform by humans. These features make UAV more suitable for WiSAR. However, currently UAV in WiSAR mainly uses the image acquisition module, cares little about the automation, the application scenarios are relatively limited. Therefore, the key technologies in the design of UAV for WiSAR are studied, i.e., the autonomous obstacle avoidance, path planning and automatic landing. Meanwhile, we utilize the target detection and recognition technologies to efficiently detect and locate survivors.

*Email address: yuhuai@whu.edu.cn

Figure 1: The hardware about WiSAR system.

2 WISAR SYSTEM DESCRIPTION

The hardware of our WiSAR system is shown in Figure 1.The whole system can be divided into three components, i.e., drone system, ground station and remote control. The details about the three parts are listed as follows,

- Drone system consists of a DJI Matrice 100 drone with an on-bard PC Maniford, five stereo-vision sensors to provides around and downside depth cloud maps, two ultrasonic sensors to keep a safety height, a DJI Zenmuse X3 camera to provide optical images, a FLIR VUE Pro to provide infrared images.
- Ground work station is a DELL laptop with a very powerful GPU NVIDIA Quadro M5000, which is mainly used to process images.
- The remote control is a DJI standard remote control with an Android phone.

Our WiSAR system works as: after the whole system is powered on, the on-bard computer Manifold starts to take control and guide the drone autonomously take off. Then the drone flies follow the global setting path. When there are obstacles in the front or downside (point clouds from the stereo-vision sensors), the Robot Operating System (ROS) [1] (Robot Operating System) navigation package is used to set the local path. Meanwhile, a searching command is sent from the APP in remote control module to the drone by 2.4GHz/5.8GHz

IMAV2017-18
https://www.imavs.org/pdf/imav.2017.18

wireless communication module. The on-bard computer begins screening the infrared video and sending ROI (Region of Interest) to remote control. When the ground station receives the searching command and ROI, the improved-SSD algorithm is used to detect target using the optical video stream transported from the remote control. The data link is shown in Figure 2.

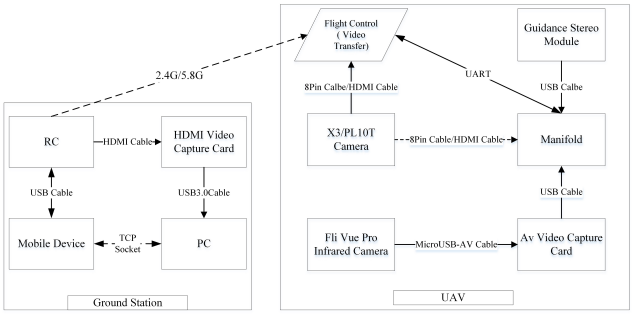


Figure 2: The data link diagram of Our WiSAR system.

3 People Search and Rescue

Finding lost people is the main job of our WiSAR system. In this section, we provide a method to reduce the false alarm of detection by combining infrared detection with optical detection. We use the infrared images for assistance. We use morphological methods to obtain the salient region, then converted it into the target candidate regions. After that, we align the infrared image and the optical image. We use the improved SSD [2] model to real-time detect people, based on the candidate region.

3.1 Infrared image target detection

Usually, living target, like humans and other warm blooded animals, can be distinctly displayed on infrared images because these targets can radiate more energy than the background. In our WiSRA system, we using FLIR VUE Pro, a thermal imager with 640×512 pixels resolution, to effectively detect the thermal information within hundred meters away. When our WiSAR system fly in the air, the resolution of living targets is usually 30×30 pixels. This relative low makes the information easily lost and other disturbances make the analysis of infrared image difficult (Figure 3a). So we apply several methods described below to handle the infrared images first, making it easy to process.

First, it is necessary to carry out a equalization of the image because strong light will cause the temperature of the water in the air rising, radiating more infrared rays that can affect thermal imager. While the distribution of water vapor is usually evenly distributed in a smaller area, the infrared energy radiated outward is also evenly distributed. Let the length of the image I be W, the width is H, $P(x, y)$ is the gray level of the position (x, y), $P'(x, y)$ is the gray value of (x, y) after the mean translation, then we have:

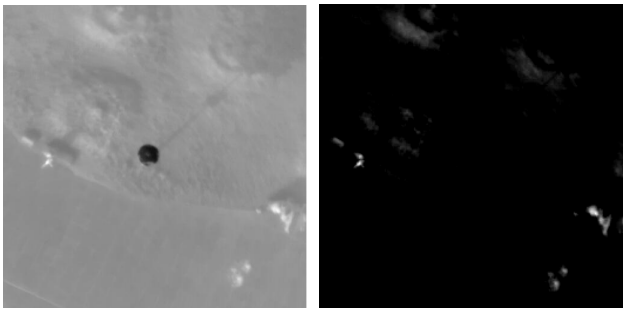
$$P'(x, y) = P(x, y) - \frac{1}{WH} \sum_{i,j \epsilon W,H} P(i, j) \tag{1}$$

After equalization, we use gray-scale transformation to let the gray scale range from 0 to 255 value instead of negative value. The linear gray-scale transformation function is define as,

$$D_b = f(D_a) = kD_a + b \tag{2}$$

where k is the slope, b is the intercept. D_a indicates the grayscale of the input infrared image. D_b indicates the grayscale of the output image.

When the gradient k is greater than 1, the contrast of the processed image will increase, and if the gradient k is less than 1, the contrast will decrease. Here, we highlight the target by increasing the contrast. By limiting the grayscale, the gray values of the infrared images can be distributed equally in different lighting conditions, allowing for further processing. By doing this, the significant region (temperature anomaly region) is effectively extracted (Figure 3b).



(a) (b)

Figure 3: Infrared image gray scale processing. (a) Blurry infrared image contained mutiple targets. (b) Image after equalization and gray-scale transformation.

Further, we use the Top Hat transformation to reduce the impact of light. Considering that the target size under the UAV perspective is small and the brightness in the image is high, Using the top hat transformation is reasonable.

From the previous steps, we have been able to get less noise results. However, the background of the wild environment is very complex, just using infrared search is unable to achieve the reliable results. Therefore, if the number of candidate regions, obtained by the above image analysis process, is greater than 0 (Figure 4), the UAV will transmit report instructions and the coordinates of candidate regions to the ground station. The ground station will automatically use optical detection to search people based on these informations.

3.2 Optical image taget detection

Recently, the state-of-the-art deep conventional neural network (DCNN) frameworks like YOLO9000 [3] and SSD

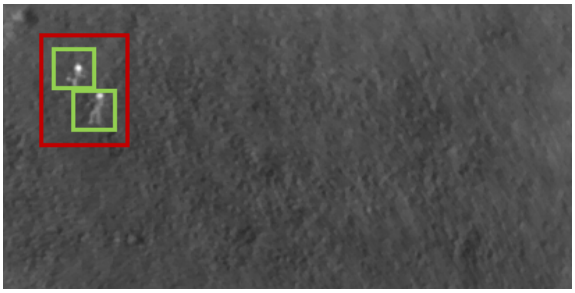


Figure 4: Candidate region detected by thermal camera.

[2] have been presented. Considering the past target detection methods based on statistical learning, such as HOG [4] and DPM [5], are used artificial designed features, not enough for field complex scenes. And field scenarios, people's contours, edges, textures may not be very clear, and even there will be a certain degree of distortion (Figure 5a and 5b). Thus, we decide to use the state-of-the-art methods to overcome the problems above, that is DCNN. Through the use of our own field people dataset UAV-PP, our WiSAR system can effectively locate the field people. Based on the requirements for real-time and high accuracy, we compared multiple DCNN frameworks and finally decided to adopt the improved-SSD framework.

(a) (b)

Figure 5: People under UAV perspective. (a) People in different postures, (b) People with partial occlusions and without partial occlusions.

The network structure of SSD [2] is very clear and easy to understand. First it uses a forward-propagating CNN as a base network. This forward convolution network can produce a series of fixed-size brackets and these enclosing boxes contain scoring of various categories of objects. And then it uses Non-maximum Suppression (NMS) to filter out the final prediction results. The feature maps, from the same network while at different levels, have different size of receptive field. However, the SSD model does not have to let the default bounding box corresponding to the receptive field in every layer, but let the feature map to be responsible for the prediction of a particular scale. Suppose you want to use m feature graphs to predict. Then the scale of the default bounding box corresponding to each feature map s_k can be calculated according to the following formula:

$$s_k = s_{min} + \frac{s_{max} - s_{min}}{m-1}(k-1), k\epsilon[1, m] \quad (3)$$

We changed the scale partition, making m equal to 4, s_{min} equal to 0.4 and s_{max} equal to 0.8, because the scale of the target in UAV perspective does not change much. In addition, we increased the batch size from 128 to 512, because the actual size of the target is relatively small.

The original base network of SSD uses the prediction layer in front of VGG16 [6]. VGG16 model has been proved to be a good classification prediction model, but its structure is too complex and with many layers, a 32×32 target after VGG becomes 2×2 size, makes the extra layer is easy to lose semantic information. So the original SSD for small size targets (such as UAV view of the people) detection effect has been affected. Therefore, for the small target detection, it is necessary to increase the semantic information of the context[7]. Residual Network(ResNet)[8] proposed by He et al can preserve as much target semantic information as possible. ResNet introduces a shortcut between the output and the input (shortcut), that is identity function, rather than a simple stack network. This can solve the problem that the semantic loss occurs due to the network being too deep. Allowing people to further increase the depth of the network. So we use the depth of the ResNet-101 structure as the SSD base network to improve. It is worth noting that even using ResNet-101 will not significantly increase the time required for target detection. Through the above improvements, SSD-ResNet model for small target detection has been improved.

3.3 Combination of infrared and optical detection

When trying to combine the optical and infrared target detection, we need some coordinate transformation, because the optical camera and the infrared camera have a certain physical distance during installation . In order to get the correct candidate region coordinate mapping, we need to get the conversion relationship between the two camera coordinate systems. In Figure 6,

θ_1 is the viewing angle of the infrared camera,

θ_2 is the viewing angle of the optical camera (taking the x-axis as an example),

The point $p_f(u_f, u_f)$ in the infrared camera image coordinate system will be mapped to the image coordinates $p_p(u_p, u_p)$ of the optical camera by the following transformations.

$$u_p = u_{po} + WIDTH_P * (u_f - u_{fo'})/WIDTH_F + \frac{d}{f_x} \quad (4)$$

$$v_p = v_{po} + HEIGTH_P * (v_f - v_{fo'})/HEIGTH_F + \frac{d}{f_x} \quad (5)$$

Where,

$WIDTH_P$ is the width of the optical image,

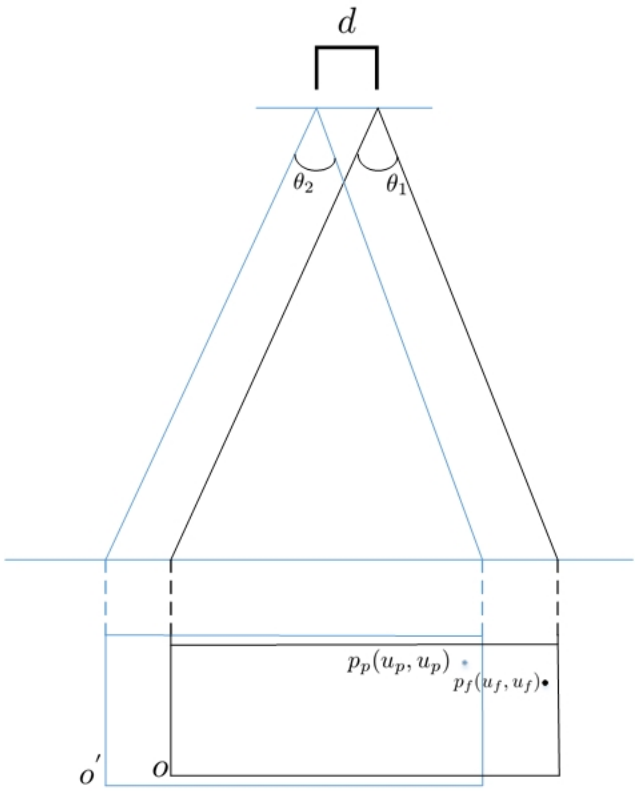


Figure 6: Relationship between infrared and optical camera coordinate system.

$WIDTH_F$ is the width of the optical image,
$HEIGTH_P$ is the height of the infrared image,
$HEIGTH_F$ is the height of the infrared image,
d is the installation position deviation, in our WiSAR is 6 cm,
f_x is actual physical length of each pixel in an optical camera.

By using two of the above image coordinate transformation equations, it is possible to obtain the corresponding position of the target candidate region on the optical image. Then set it as the region of interest (ROI) in optical images and perform optical target detection based on this ROI.

4 Autonomous Obstacle Avoidance and Landing

As for autonomous obstacle avoidance, we adopt the obstacle avoidance scheme based on binocular vision. It mainly uses the parallax principle to carry out motion estimation, so as to obtain the depth information of the obstacle in front of the UAV. And then update the cost map based on the point clouds, and further update the obstacle avoidance route according to the cost map, so as to achieve the purpose of avoiding obstacles [9]. If the UAV encounters an uneven terrain like Figure 8, the ultrasonic detector mounted on the bottom

Figure 7: Human detection results using our WiSAR system.

of the UAV will initiative send signal to keep a safety flight altitude. The results of the obstacle avoidance experiments for static obstacles (wall, tree, etc.) and moving obstacles (pedestrian) show that the vision avoidance scheme based on binocular vision is more reliable than Light Detection And Ranging (LIDAR).

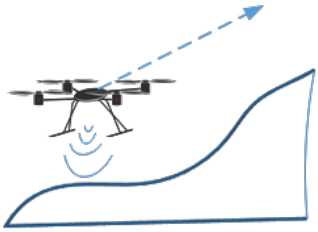


Figure 8: UAV use ultrasonic to avoid obstruction .

Regarding the autonomous landing, we introduce the visual fiducial system AprilTag [10] from the Augmentation Reality filed. By encoding an AprilTag, the detection algorithm can greatly reduce the false alarm, and because of the introduction of fault tolerance mechanism, the miss rate is also maintained at a very low level. After detecting the mark, the UAV can dynamically adjust its pose to achieve accurate landing on the mobile platform by solving the PnP problem. Experimental results show that the position deviation of landing is less than 15 cm, which meets the requirement of real landing scenario where a UAV is likely to land on a platform with limited area, such as roof and truck rear.

AprilTag information is only expressed in black and white blocks, without any symmetry(Figure 9a). In our WiSAR system, we use the 25h9 set. First, the useful contour information is extracted by image graying(Figure 9b) and used the adaptive binarization algorithm to binarize gray scale images (Figure 9c), and then processed by Gaussian filtering (because Gaussian filtering can better preserve the edge information [11]), and finally through the basic and further screening (preventing false contours detection in internal AprilTag), we can get the correct results (Figure 9d). Further, we introduced the

Hamming code [12] to improve error correction and screen out false alarms, making our AprilTag looks like Figure 9e.

(a) (b) (c) (d) (e)

Figure 9: Detection process of AprilTag. (a) Original image with 6 AprilTags, (b) Image after gray processing, (c) Image after binarization, (d) AprilTag edge detection, (e)AprilTag after Hamming coding.

5 Experiments and Results

5.1 *Field people dataset UAV-PP*

We build and put forward a field people dataset UAV-PP and used the VOC [13] dataset format to increase the versatility of UAP-PP dataset. Because the people under the UAV perspective are very different with the people under the ground camera perspective, and the deep learning is highly rely on large volume of data. So lacking UAV perspective people database will heavily influence the average precision of detection.

UAV-PP including 3180 original images, resolution of these images are 1000×1000 pixels, each image contains 10 to 20 targets (people), each positive sample has a corresponding ground truth bounding box. We provide a labeling tool (Voc-Annotation-Tool) to facilitate the placement of Ground Truth. It has the ability to batch rename pictures, import pictures, and mark the image in VOC format and generate the corresponding structure files. The software runs as Figure 10.

Figure 10: Voc-Annotation-Tool.

To improve the robustness of the DCNN model, we mainly use side view and bottom view samples. The ratio of the human training images in the various postures are about 1: 1: 1 (bottom view $85°$, side view $45°$, and other gestures respectively). Figure 11 shows the example targets in UAV-PP dateset.

Figure 11: Human sample images.

5.2 *Comparison of target detection algorithm*

In the same test platform (as described in Table 1), we used Average Precision (AP) to compare different target detection frameworks and found that:

Table 1: Algorithm test platform

CPU	Intel E3-1505M
GPU	NVIDIA Quadro M5000, 8GB
RAM	64GB

DPM has a high demand for image capture quality and perspective. In the different perspectives of people, the DPM algorithm can not extract the characteristics from numerous samples. DPM received 63.82% AP on the test set in the UAV-PP dataset, with an average speed: 3.0 Seconds per image.

As a successful deep learning framework, the detection effect of Fast-RCNN is quite good. But due to the various viewing angles, there are still some missed targets can't be detected. Using Imagenet dataset, learning rate α equal to 0.0003, batch size equal to 128, after 60000 times iterations, we have 72.47% AP of Fast-RCNN, the average speed: 0.16 Seconds per image.

SSD is faster and more accurate than Fast-RCNN when using the same dataset (ImageNet [14] dataset) and test set. Average speed of SSD: 0.16 Seconds per image. AP of SSD: 80.85%.

After that, we changed the original SSD base network to the residual network, and optimized the parameters of SSD, and got 88.92% AP. Meanwhile, the average speed did not increase, which still was 0.08 seconds per image. The comparisons of AP and speed are shown in Figure 12 and Figure 13.

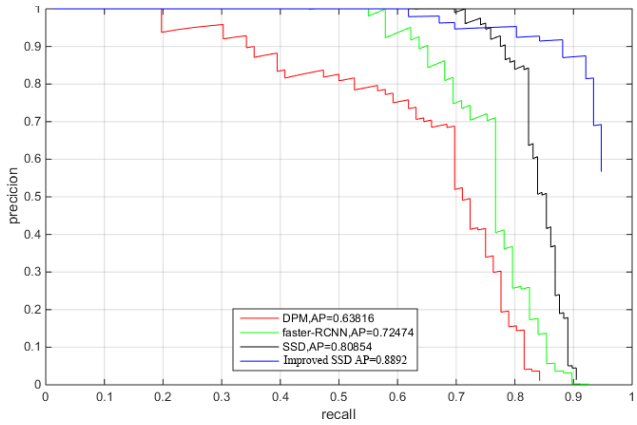


Figure 12: Average Precision between different target detection algorithms.

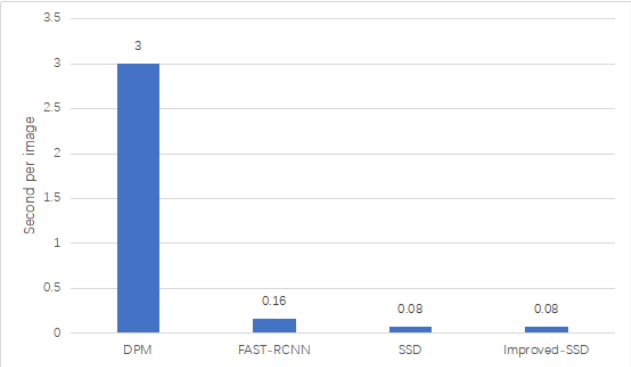


Figure 13: Speed between different target detection algorithms.

5.3 *WiSAR system test*

The whole WiSAR system is a multi-module integrated system. For the UAV side, data processing and instruction transmission is mainly in ROS. For the ground station side, the system includes a supporting demonstration system to satisfy the practical need. Which includes an Android application (Figure14) and a PC application (Figure15).

First of all, we conducted an independent obstacle test, the results show that the UAV can effectively avoid moving pedestrians and other obstacles. Then we carried out 20 tests to test the autonomous landing (Figure 17a and 17b). Let the center of the platform as the origin point,we have the results that: the average deviation of X axis is 9.2 cm, the average deviation of Y axis is 9.0 cm. The results are enough to meet the requirements of landing on a moving truck. Finally, we use 15 tests to test the time and distance flying ability of our WiSAR system when under full load. The average farthest flight distance is 4.2 km at full load and the average maximum flight time is 15 minutes and 10 seconds.

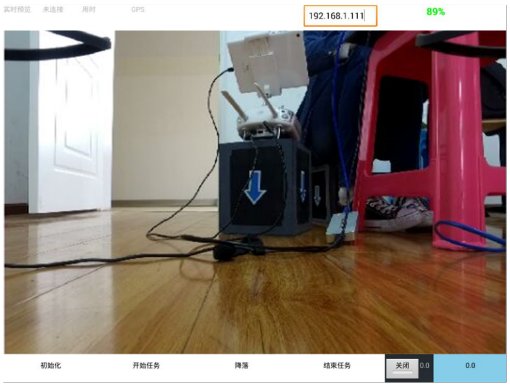


Figure 14: Android application interface of our WiSAR system.

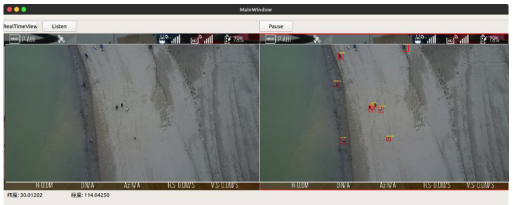


Figure 15: PC application interface of our WiSAR system.

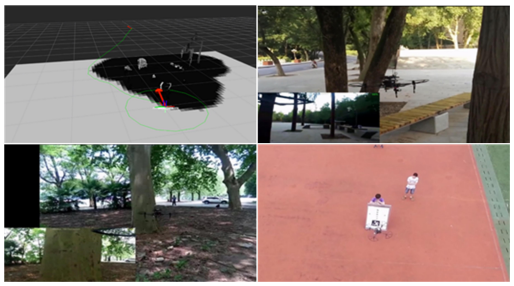


Figure 16: Some results of obstacle avoidance.

(a) (b)

Figure 17: Autonomous landing test. (a) Autonomous landing on truck rear, (b) Autonomous landing on simulated truck rear.

IMAV2017-18
https://www.imavs.org/pdf/imav.2017.18

6 Conculsion

In conclusion, we have completed the design, prototyping and construction of our WiSAR system. The system adopts modular development strategy, leading to low coupling degree and high portability. The accompanying software applications include an Android application and a multi-threaded graphical PC application. Both applications respond quickly and interactively. The experimental results show that the performance of our WiSAR system is fully functional. The simulated search and rescue tasks can be successfully accomplished, which lay a solid foundation for building a more intelligent search and rescue system based on UAV.

Acknowledgment

The research was partially supported by the CETC key laboratory of aerospace information applications under Grant KX162600018.

References

[1] Morgan Quigley, Ken Conley, Brian Gerkey, Josh Faust, Tully Foote, Jeremy Leibs, Rob Wheeler, and Andrew Y Ng. Ros: an open-source robot operating system. In *ICRA workshop on open source software*, 2009.

[2] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng Yang Fu, and Alexander C. Berg. Ssd: Single shot multibox detector. In *ECCV*. Springer, 2016.

[3] Joseph Redmon and Ali Farhadi. Yolo9000: better, faster, stronger. In *CVPR*. IEEE, 2017.

[4] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *CVPR*. IEEE, 2005.

[5] Pedro Felzenszwalb, David McAllester, and Deva Ramanan. A discriminatively trained, multiscale, deformable part model. In *CVPR*. IEEE, 2008.

[6] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICRL*, 2015.

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. In *ECCV*. Springer, 2014.

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*. IEEE, 2016.

[9] Lionel Heng, Dominik Honegger, Gim Hee Lee, and et al. Autonomous visual mapping and exploration with a micro aerial vehicle. *Journal of Field Robotics*, 31(4):654–675, 2014.

[10] Edwin Olson. Apriltag: A robust and flexible visual fiducial system. In *ICRA*. IEEE, 2011.

[11] Guang Deng and Laurence Cahill. An adaptive gaussian filter for noise reduction and edge detection. In *NSS/MIC*. IEEE, 1993.

[12] Chi-Shiang Chan and Chin-Chen Chang. An efficient image authentication method based on hamming code. *Pattern Recognition*, 40(2):681–690, 2007.

[13] Mark Everingham, Luc Van Gool, Christopher K. I. Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88(2):303–338, jun 2010.

[14] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*. IEEE, 2009.