applied sciences

MDPI

*Article*

# UAV Detection with Transfer Learning from Simulated Data of Laser Active Imaging

**Shao Zhang [1,2], Guoqing Yang [3], Tao Sun [1], Kunyang Du [1,2] and Jin Guo [1,*]**

[1] State Key Laboratory of Laser Interaction with Matter, Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences, Changchun 130033, China; ciompzs@163.com (S.Z.); suntao@ciomp.ac.cn (T.S.); dukunyang1995@163.com (K.D.)
[2] University of Chinese Academy of Sciences, Beijing 100049, China
[3] Visual Computing Research Center, Shenzhen University, Shenzhen 518061, China; yanggq@szu.edu.cn
* Correspondence: ciompgj@sina.com

**Abstract:** With the development of our society, unmanned aerial vehicles (UAVs) appear more frequently in people's daily lives, which could become a threat to public security and privacy, especially at night. At the same time, laser active imaging is an important detection method for night vision. In this paper, we implement a UAV detection model for our laser active imaging system based on deep learning and a simulated dataset that we constructed. Firstly, the model is pre-trained on the largest available dataset. Then, it is transferred to a simulated dataset to learn about the UAV features. Finally, the trained model is tested on real laser active imaging data. The experimental results show that the performance of the proposed method is greatly improved compared to the model not trained on the simulated dataset, which verifies the transferability of features learned from the simulated data, the effectiveness of the proposed simulation method, and the feasibility of our solution for UAV detection in the laser active imaging domain. Furthermore, a comparative experiment with the previous method is carried out. The results show that our model can achieve high-precision, real-time detection at 104.1 frames per second (FPS).

**Keywords:** laser active imaging; object detection; transfer learning



**Citation:** Zhang, S.; Yang, G.; Sun, T.; Du, K.; Guo, J. UAV Detection with Transfer Learning from Simulated Data of Laser Active Imaging. *Appl. Sci.* **2021**, *11*, 5182. https://doi.org/10.3390/app11115182

Academic Editor: João Carlos de Oliveira Matias

Received: 30 April 2021
Accepted: 27 May 2021
Published: 2 June 2021



## 1. Introduction

With the increasing demand for high-precision data collection, commercial quadrotor UAVs have emerged. As ideal platforms of data acquisition, UAVs have high maneuverability even in complex environments. UAVs play a significant role in a variety of fields such as photography, disaster monitoring, and traffic guidance nowadays [1,2]. Moreover, UAVs can also be deployed in a lot of military applications [3]. Due to the lack of effective regulation of UAVs, their abuse poses a threat to the privacy of citizens and the flight safety in specific places, such as airports. What's more, the lack of regulation gives rise to smuggling, terrorist attacks, and other illegal activities. To build a system of regulation, it is crucial to carry out research on the detection of UAVs. However, this is not an easy task due to the small size of the UAVs and the limited field of view (FOV) of the detection system, especially since it would become more difficult when there is no proper illumination.

The laser is a promising source to compensate for the situation of low illumination, due to its high intensity and high collimation. Therefore, many laser active imaging systems have been proposed for long-range target identification at night [4]. Gating technology is usually adopted in these systems to mitigate the scattering effects of obscurants and reduce the influence of background clutter. Researchers have proposed a common implementation of laser active imaging systems using a long-wave infrared camera with a large FOV as the detection device to search for the target. The distance of the target is determined by the time of flight (ToF) of the laser for range gating. At last, a laser range-gated short wave infrared camera is used for target identification [5]. In 2015 [6], this author proposed a target

recognition algorithm that used laser active imaging based on fast contour torque features. However, the algorithm only considers a single background and cannot be extended to complex scenes. All of the studies mentioned above recognize the target in a narrow FOV, but cannot extend to a complex scenario. Our work here focuses on using laser active imaging to perform target detection, including target location and identification, even in a complex scene.

There are many traditional hand-crafted, feature-based models, such as histograms of oriented gradients (HOG), scale-invariant feature transform (SIFT), Haar-like and deformable part-based model (DPM) [7], which have achieved good object detection performance in natural images. However, compared with the conventional passive imaging system, laser active imaging produces gray images with higher noise owing to the laser's coherence, which hinders the performance of traditional object detection algorithms. Since the great success of Alexnet [8] in 2012, convolutional neural networks (CNNs) have been widely used in a variety of fields of computer vision, including image classification [9], object detection [10], and image segmentation [11]. CNNs can learn more robust features from data automatically compared with traditional algorithms, in which sophisticated features need to be designed manually. Furthermore, object detection algorithms based on CNN rely on a large amount of annotated data for its training. However, collecting enough data to satisfy the requirements is time-consuming, laborious, and even impossible in many situations. Laser active imaging is a typical situation where few samples can be gathered. Transfer learning is an effective and promising method to alleviate the situation of insufficient data in many domains [12]. Deep transfer learning aims to transfer the prior features extracted from the source domain $D_{src}$ of a large dataset to the target domain $D_{tar}$ where data is scarce. Yang et al. [13] achieved a significant improvement in military object recognition by equipping the model with the prior knowledge learned from ImageNet [14]. Another study [15] applied transfer learning in the medical field, achieving glioma grading on conventional magnetic resonance images. These works mentioned above still need some labeled data in the target domain, while there is no dataset available for our application. Besides, it is also difficult to collect and construct such datasets under various scenarios. Therefore, we propose a method to simulate the imaging process of laser active illumination to generate synthesis images to construct the simulated dataset $D_{sim}$ for the training of the proposed deep neural network. Much research has been conducted about UAV detection. Zhu et al. [16] used deep transfer learning to recognize targets in a UAV-to-ground situation, which is the opposite of our ground-to-UAV situation. Sommer et al. [17] trained a CNN to detect and recognize UAVs in a natural light scene, while we mainly want to detect UAVs at night. Zhao et al. [18] used a Doppler radar signal to detect small UAVs, while we want to use laser active illumination to detect UAVs. To the best of our knowledge, our work is the first attempt to detect small UAVs through deep transfer learning from simulated data in a laser active imaging field.

Figure 1 shows the schematic illustration of the proposed method. Firstly, the model learns the general features in a large dataset of common objects in context (COCO) [19]. This step gives proper initial values to the network parameters. Then, the model learns the UAV features in the constructed simulated dataset. Finally, the trained model is applied to the real laser active imaging images to realize the high-precision detection of the UAVs. Our main contributions can be summarized as follows:

(1) A real-time UAV detection framework is established based on a CNN cooperating with transfer learning. To the best of our knowledge, this is the first study to analyze the problem of zero-shot object detection in the laser active imaging domain.
(2) A dataset is constructed by simulating the process of laser active imaging. The knowledge learned from the simulated dataset is beneficial to UAV detection in real data.
(3) We experimentally show that our algorithm can realize a high-precision UAV detection for our laser active imaging system, which proves the authenticity of the simulated data and the success of our solution.

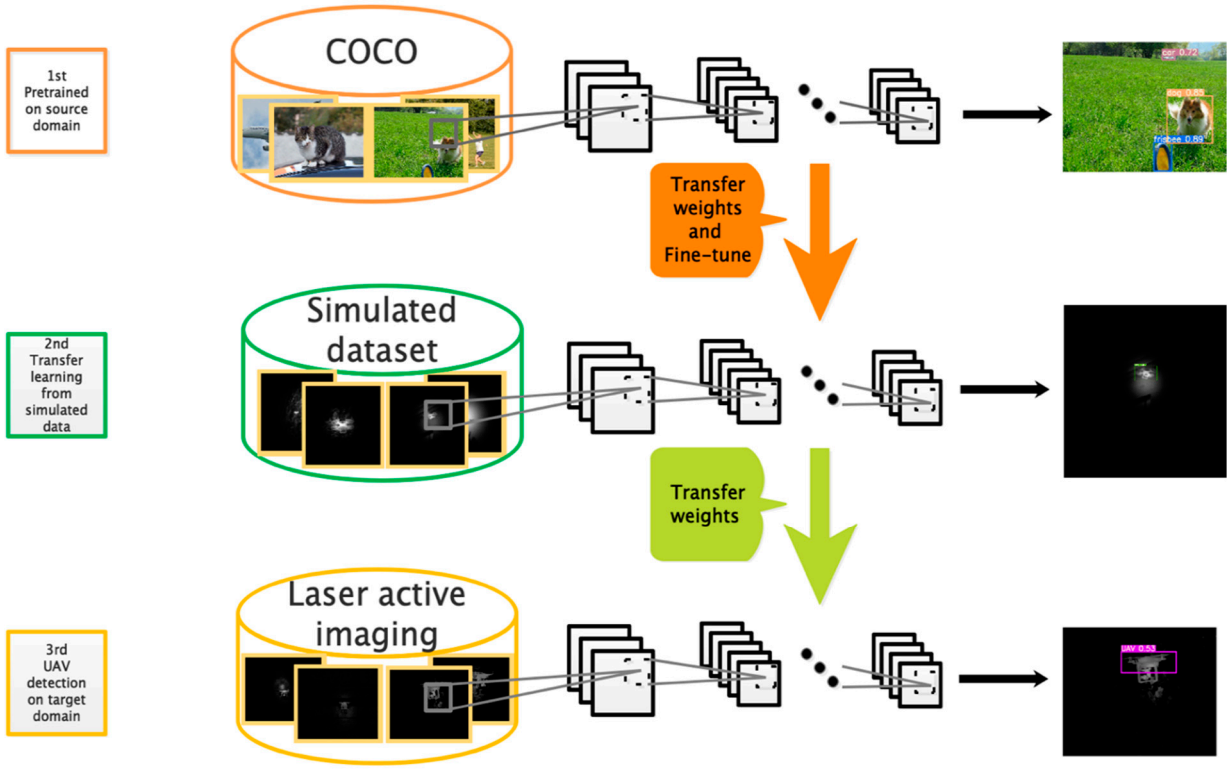



**Figure 1.** The flowchart of the proposed method. First, the model was pre-trained on the source domain. Second, the model was fine-tuned on the simulation dataset. Third, the model detected UAV in the target domain.

The remainder of this paper is as follows. In Section 2, we briefly introduce the related work. The simulation process is analyzed in Section 3. Then, the adopted algorithm and its improvement, as well as the experimental process, are described in Section 4. The analysis of the experimental results is in Section 5, and Section 6 summarizes and prospects our work.

## 2. Related Work

### *2.1. Laser Active Imaging*

Active imaging systems are widely used at night or in low light conditions. Many 2D laser-illuminated imaging systems are proposed due to their advantages over passive imaging systems. Renold et al. [20] quantitatively analyzed the effect of laser speckle on the target identification performance of a 2-D laser active imaging system. The modeling of the system is presented, which emphasizes mainly the effect of speckle and atmospheric scintillation [5]. In [21], the author designs and implements a range-gated underwater laser imaging system and realizes the underwater target detection at a distance of 40 m. In this paper, our system uses a continuous laser to illuminate the target and doesn't need the target distance information, so we can quickly obtain and process 2D images of targets.

### *2.2. Object Detection*

Traditional object detection methods use handcrafted features. The performance of the algorithm depends on the robustness of the features to a large extent. With the appearance of the huge amount of annotated data and high-performance hardware, the deep learning-based method has been widely adopted in object detection, which can learn semantic, high-level features with good robustness automatically. Deep learning-based object detection methods can be roughly divided into two types. The former is a two-step method, generating region proposals firstly, and then classifying and identifying each proposal by CNN. The typical frameworks of this type include region-based CNN (R-CNN) series, Mask R-CNN, etc. The latter regards object detection as a regression problem, which

directly gives the object category and location information simultaneously, for example you only look once (Yolo) series or single shot multibox detectors (SSD) [7]. Compared with the two-step method, the latter can achieve faster detection speed while maintaining comparable detection accuracy. YOLOv5 [22] is the latest version of the YOLO series, which has a significant increment in performance compared to older versions. In this paper, we adopted the smallest YOLOv5s model as the backbone model, which can commendably meet the requirements of our application scenarios.

### *2.3. Transfer Learning*

Transfer learning is a promising technology to ease the problem of limited labeled data in the majority of domains of interest. The definition of transfer learning can be summarized as follows: given a source domain $D_{src}$ with large labeled data and a target domain $D_{tar}$ with a few labeled data, transfer learning aims to learn knowledge in $D_{tar}$ based on the prior knowledge learned from $D_{src}$. When the knowledge of data is learned by a deep convolutional neural network, transfer learning becomes deep transfer learning. As the simplest and the most effective measure of deep transfer learning, fine-tuning is widely used in the early stage. Jason et al. conducted a survey of the transferability of deep neural networks and declared that fine-tuning is a desirable mean to overcome the domain gap of different datasets [23]. There is no doubt that transfer learning can be used for UAV detection. In [24], the author achieved UAV-Bird image classification using deep transfer learning with a synthetic dataset; Sommer et al. [17] detected UAVs in visible imagery with a two-step approach: flying object detection and subsequent object classification. In the latter stage, fine-tuning pre-existing weights is important for stable training with the small amount of UAV data. In this paper, our goal is to explore the transferability of knowledge learned from the simulated dataset we constructed, so we choose the simple fine-tuning method to complete deep transfer learning.

## 3. Data Simulation

To simulate the imaging process, we need to have complete knowledge of the laser active imaging system. The active imaging system uses its light source to irradiate the target area and receives the reflected signal of the target. Hence, it is not limited by the illumination of the scene. At the same time, the laser has the characteristics of high brightness and high collimation, so it is an ideal light source. The commonly used laser active illumination imaging system is shown in Figure 2. The laser is emitted to irradiate the object, and then the camera images at the image plane by receiving the reflected signal of the object.

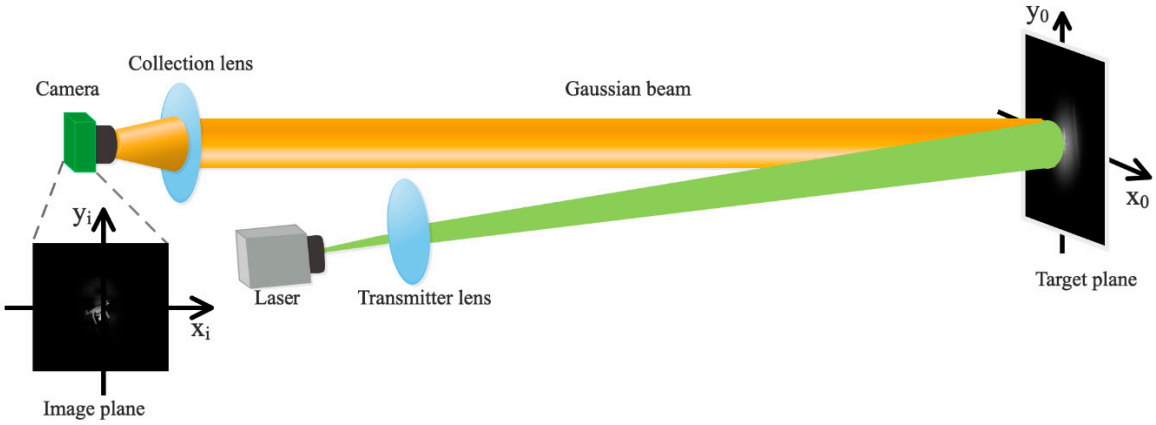



**Figure 2.** Schematic diagram of our laser active imaging system.

In the process of laser illumination, the laser can be modeled as a Gaussian beam [25], so the intensity of light can be written as:

$$I(r,z) = I_0\left(\frac{w_0}{w(z)}\right)^2 \exp\left(\frac{-2r^2}{w^2(z)}\right) \tag{1}$$

where

$$w(z) = w_0\sqrt{1+\left(\frac{z}{z_R}\right)^2} \tag{2}$$

is the spot size in the forward propagation direction $z$. $r$ is the radial coordinate, taking the optical axis center as a reference. $I_0 = I(0,0)$ is the irradiance at the center of beam waist $w_0$. $z_R = \pi {w^2}_0/\lambda$ is the Rayleigh length, with $\lambda$ being the wavelength.

Reflection occurs when the laser reaches the target surface. It is a non-trivial problem to determine the reflectivity of each point on the target plane. For simplicity, the reflectivity was approximated by the gray level of the panchromatic visible image. Ignoring the influence of atmospheric turbulence, the irradiance at the target plane can be approximated as:

$$I_0(x_0,y_0) \approx I_c \exp\left(\frac{-2\times\left((x_0-x_c)^2+(y_0-y_c)^2\right)}{w^2(z)}\right)\cdot I_{gray} \tag{3}$$

where $I_{gray}$ is the gray level of the image in natural light; $I_c$ denotes the irradiance of the center point $(x_c, y_c)$ of the Gaussian beam at the target plane. Then the process from the object surface to the imaging surface can be represented based on standard statistical optics:

$$\begin{aligned} I_{im}(x_i,y_i) &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} J_0'(x_0,y_0;x_0',y_0') \\ &\times h(x_i-x_0,y_i-y_0)\times h^*(x_i-x_0',y_i-y_0')dx_0dy_0dx_0'dy_0' \end{aligned} \tag{4}$$

where $(x_i, y_i)$ denotes the position in the image plane; $J_0'(x_0,y_0;x_0',y_0')$ is the mutual intensity of $(x_0, y_0)$ and $(x_0', y_0')$, two points in the target plane; $h(x_i, y_i)$ is the amplitude spread function of the imaging system. This representation of the imaging process is general, but it is difficult to determine the mutual intensity. For simplicity, the mutual intensity is usually modeled in two extreme cases: fully coherent and completely incoherent. The real laser works between these two extremes.

### *3.1. Coherent Imaging*

When the light is coherent, the mutual intensity is given by

$$J_0'(x_0,y_0;x_0',y_0') = U_0(x_0,y_0)U_0^*(x_0',y_0') \tag{5}$$

where $U_0\ (x_0, y_0)$ and $U_0^*(x_0', y_0')$ denote time-averaged field quantities at the target plane. Under this limited condition, the image intensity can be simplified as:

$$I_{im}(x_i,y_i) = \left|\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} U_0(x_0,y_0)h(x_i-x_0,y_i-y_0)dx_0dy_0\right|^2 \tag{6}$$

According to [20], the amplitude of $U_0$ could be approximated by taking the square root of $I_0$. The phase of $U_0$ was chosen from a uniform distributed random variable over the interval $[0, 2\pi)$.

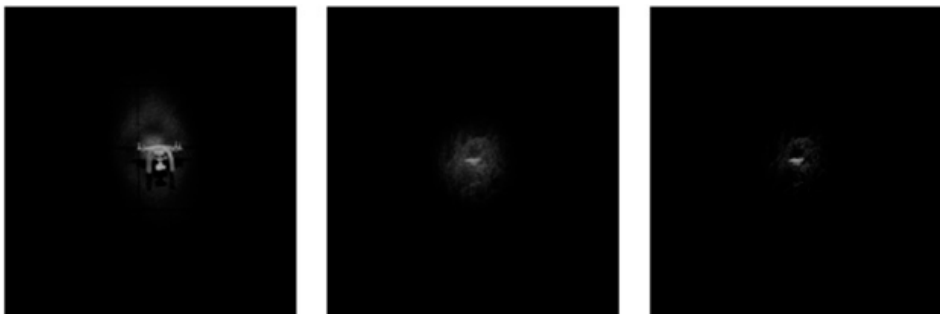
### *3.2. Incoherent Imaging*

When the light is spatially incoherent, the imaging system is a linear transfer for the irradiance:

$$I_{im}(x_i, y_i) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} I_0(x_0, y_0) |h(x_i - x_0, y_i - y_0)|^2 dx_0 dy_0 \quad (7)$$

The image irradiance is the convolution of the object irradiance with the squared magnitude of the amplitude spread function.

We use both coherent imaging and incoherent imaging to generate the simulated image of laser active imaging. Figure 3 shows a real laser active illumination image of the UAV and two simulated images of coherent imaging and incoherent imaging, respectively. In addition to the different imaging mechanisms, there are also the factors of background and target distance, which together cause the visual differences between the three images.



(**a**) (**b**) (**c**)

**Figure 3.** Real vs. simulated images. (**a**) Real image. (**b**) Coherent imaging image. (**c**) Incoherent imaging image.

## 4. Methodology

In this section, we will give a brief introduction of the principle of YOLO, the network architecture of YOLOv5s, and the bounding box regression loss used in our solution first, then describe the datasets used for training and evaluation, and finally introduce the training protocol of our method.

### *4.1. Principle of YOLO*

YOLO series algorithm is a typical representative of the deep convolution neural network (DCNN) in the field of image object detection. The reason why it is called DCNN is that it has a multilayer structure and can extract very deep features. The key structures to make a neural network deeper are the convolution layer and the pooling layer. The function of a convolution layer is to extract local features; the function of a pooling layer is to select features and prevent overfitting. With the development of DCNN, many mature design skills have been proposed and also adopted by the YOLO series algorithm. For example, small ($3 \times 3$) convolution filters are adopted throughout the whole net since a stack of $3 \times 3$ convolution layers can cover the same size receptive field of $5 \times 5$ or $7 \times 7$ convolution filters while retaining a small number of parameters [9]; Batch Normalization [26] is usually used to accelerate the training process of DCNN; leaky rectified linear unit (LReLU) [27] is chosen as the activation function to accelerate optimization and improve performance. Furthermore, inspired by feature pyramid networks (FPN) [10], YOLOv3 [28] and later versions predict bounding boxes at 3 different scales to improve the detection accuracy.

### *4.2. Network Structure*

In YOLOv5, the author adopts a variety of tricks to improve network performance. The network structure of YOLOv5s used in this paper is shown in Figure 4. Compared with the previous version, there are mainly two modules adopted. One is the BottleneckCSP module, which draws lessons from the cross stage partial network (CSPNet) [29]; the other one is the path augmentation structure, which is inspired by the path aggregation network (PANet) [30]. CSP module divides the shallow feature map into two parts, then merges them after going through different paths. The network can extract abundant features while reducing the amount of computation necessary by utilizing this strategy. In order to better integrate high-level and low-level features, the path aggregation structure is used to add a bottom-up path on the basis of FPN. Although FPN can mix the high-level features and low-level features, the shallow features have to pass through dozens or even hundreds of convolution layers in the bottom-up process of the original path, which will cause serious loss of shallow information. By adding the path of fewer than ten layers, the shallow features can be better preserved to integrate with deep features for improving detection ability. The specific structure of YOLOv5s is listed in Table 1. It is composed of two parts, a backbone which is used to extract features and a head which is designed to detect targets. The backbone part consists of one Focus unit, four convolutional layers, four BottleneckCSP units, and one SPP layer. The Focus unit slices the $640 \times 640 \times 3$ inputs into $320 \times 320 \times 12$, and then turns it into $320 \times 320 \times 32$ feature maps via 32 separate $3 \times 3$ filters. The function of the Focus unit is to realize down-sampling and feature extraction without losing information. Each convolution layer is followed by a pooling layer and each BottleneckCSP unit contains two convolutional layers of $1 \times 1$ and $3 \times 3$ filters. The purpose of incorporating the SPP layer is to increase the robustness of the model against object deformations. The head part consists of four convolutional layers, two Upsample units, four Concatenation units, and one Detection unit. The role of Upsample and Concatenation is to fuse features from different levels. The Detection unit is carried out on three different scales: $20 \times 20$, $40 \times 40$, and $80 \times 80$ outputs (i.e., P5, P4, and P3 in Figure 4). YOLOv5 has released four models of different sizes ranging from YOLOv5s to YOLOv5x, and there is good integration between them. In model depth, the number of BottleneckCSP modules in YOLOv5m/l/x is 2/3/4 times that of YOLOv5s; in model width, the number of layer filters in YOLOv5m/l/x is 1.5/2/2.5 times that of YOLOv5s, respectively.

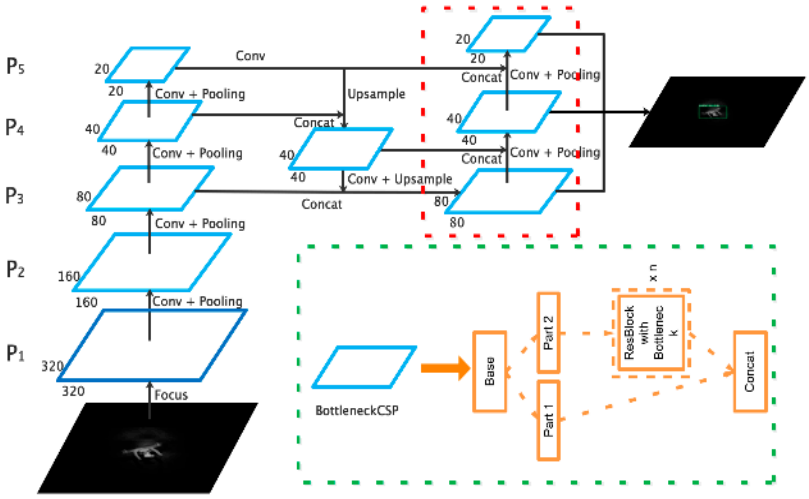



**Figure 4.** Illustrations of the YOLOv5s model [22]. There is a path augmentation structure in the red dotted box and a structural schematic diagram of the CSP module in the green dotted box.

**Table 1.** Architecture of the YOLOv5s model [22].

| Type | | Filters | Size | Output |
|---|---|---|---|---|
| | | Backbone | | |
| Focus | | 32 | 3 × 3 | 320 × 320 |
| Convolutional | | 64 | 3 × 3 | 160 × 160 |
| BottleneckCSP | | 64 | 1 × 1 + 3 × 3 | 160 × 160 |
| Convolutional | | 128 | 3 × 3 | 80 × 80 |
| BottleneckCSP | 3 × | 128 | 1 × 1 + 3 × 3 | 80 × 80 |
| Convolutional | | 256 | 3 × 3 | 40 × 40 |
| BottleneckCSP | 3 × | 256 | 1 × 1 + 3 × 3 | 40 × 40 |
| Convolutional | | 512 | 3 × 3 | 20 × 20 |
| SPP | | | | 20 × 20 |
| BottleneckCSP | | 512 | 1 × 1 + 3 × 3 | 20 × 20 |
| | | Head | | |
| Convolutional | | 512 | 1 × 1 | 20 × 20 |
| Upsample | | | 2 × 2 | 40 × 40 |
| Concatenation | | | | 40 × 40 |
| BottleneckCSP | | 256 | 1 × 1 + 3 × 3 | 40 × 40 |
| Convolutional | | 256 | 1 × 1 | 40 × 40 |
| Upsample | | | 2 × 2 | 80 × 80 |
| Concatenation | | | | 80 × 80 |
| BottleneckCSP | | 128 | 1 × 1 + 3 × 3 | 80 × 80 |
| Convolutional | | 128 | 3 × 3 | 40 × 40 |
| Concatenation | | | | 40 × 40 |
| BottleneckCSP | | 256 | 1 × 1 + 3 × 3 | 40 × 40 |
| Convolutional | | 256 | 3 × 3 | 20 × 20 |
| Concatenation | | | | 20 × 20 |
| BottleneckCSP | | 512 | 1 × 1 + 3 × 3 | 20 × 20 |
| Detection | | | | |

### 4.3. GIoU Loss

The conventional deep learning algorithms usually use MSE loss to directly regress the center coordinates as well as the length and width of the bounding box. Directly estimating the coordinates of these points is the way that treats these points as independent variables, and ignores the integrity of the object. To deal with this problem, the researchers adopted intersection over union (*IoU*) loss as an optimization objective, which is scale-invariant. However, *IoU* loss also has a serious problem: if there is no intersection area between the predicted box and the ground truth box, the gradient of the loss would be zero and cannot be optimized during the iteration. In order to solve this problem, we adopt the generalized *IOU* (*GIoU*) proposed in [31] as the bounding box regression loss. The formula of *GIoU* is as follows:

$$GIoU = IoU - \frac{|A_C - U|}{A_C} \tag{8}$$

where *IoU* denotes the ratio between the intersection and union of two boxes; $U$ is the union of the two boxes and $A_C$ represents the smaller circumscribed rectangle of the two boxes. Similar to *IoU*, *GIoU* is also insensitive to scale. $L_{GIoU} = 1 - GIoU$ can be the loss. Compared to *IoU* loss, *GIoU* loss always has a non-zero gradient whether there is an overlapping area between two bounding boxes or not. Therefore, it is the proper policy to choose *GIoU* loss as the optimization objective in our work.

### 4.4. Dataset

There are two datasets used in the experiments, which are described below.

(1) Simulated dataset: The dataset consists of simulated laser active illumination images according to the method described in Section 3. Firstly, 744 natural images of UAVs in different scenes are collected by a camera. Then ten simulated images with different

illumination centers $(x_c, y_c)$ and spot sizes $w(z)$ are generated from each image under coherent imaging and incoherent imaging, respectively. The selection of illumination center and spot size is random following the constraint that the illumination area covers the UAV target and does not exceed the image edge.

(2) Real dataset: We first construct our laser active imaging system according to Figure 2. We choose a continuous laser as the illumination source. The laser beam is collimated and expanded by the transmitting lens and then illuminates the target. The callback signal is acquired in an intensified CCD camera after passing through the collection lens. The detail parameters of the camera and laser are listed in Table 2. The setup and experimental scene are shown in Figure 5 left and right respectively. The transmitting and receiving equipment are placed on a turntable to facilitate scene scanning and subsequent tracking and monitoring. We collect three laser active imaging videos of UAVs using this system in different scenes including city, forest, and sky. The distance between UAV and imaging system is 100–500 m. Then we extracted 861 images from the videos to make up the real dataset.

**Table 2.** Experimental parameters.

| Camera | | Laser | |
|---|---|---|---|
| No. of camera pixels | 1280 × 1024 | Wavelength | 532 nm |
| Pixel size | 4.8 μm | Power | 0–10 w |
| Frame rate | 210 | Divergence angle | 10 mrad |

(**a**) (**b**)

**Figure 5.** Photograph of experimental devices (**a**) and experimental scene (**b**).

### *4.5. Training Protocol*

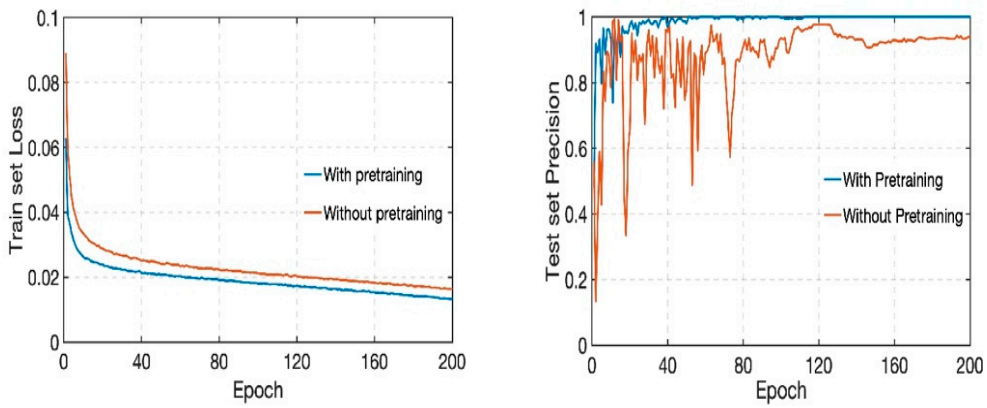
The hardware used for training was a single NVIDIA GTX1080 graphics processing unit (GPU) with 8 GB of memory. The operating system was Ubuntu 16.04, and the models were implemented on the PyTorch framework. The training process was organized in two stages: initial training and deep transfer learning. During the initial training stage, the model was pre-trained on the available dataset COCO, which consists of over 200,000 images with over 500,000 annotated object instances from 80 categories. For convenience, the trained model weights derived from YOLOv5′s model export were downloaded. Then we employed transfer learning by fine-tuning the entire model on the simulated dataset. The fine-tuning process used the following settings: SGD optimizer with initial learning rate = 0.01, weight decay = 0.0005, number of epochs = 200, batch size = 16, single-class training mode = True, and the real dataset for evaluation.

## 5. Experimental Results

In this section, we test the model and carry out experiments to explore the transferability of the knowledge learned from simulated data. The experiment results show that the *GIoU* loss has appealing properties compared with *IoU* loss. Furthermore, the model is compared with the previous algorithm. At last, the UAV detection results of our algorithm on real data are shown and discussed.

### *5.1. Model Initialization*

We firstly design an experiment to verify the importance of the first step of the proposed method, that is, pre-trained on the COCO dataset. We do this experiment based on incoherent imaging simulated data using weights random initialized and pre-trained on the COCO dataset, respectively. In Figure 6, the loss on the training set and the precision on the test set are shown for each epoch, with epoch denoting the number of batch iterations corresponding to all images in the training set. It is clear that pre-training on the COCO dataset makes the network converge faster on the simulated dataset and achieve higher precision on the real data. This result is consistent with our understanding that pre-training on a large dataset can help the model grasp general features which are beneficial for a specific task, and shows the feasibility of the laser active imaging task.




(**a**) (**b**)

**Figure 6.** Training loss values (**a**) and test precision values (**b**) for each epoch. The different color of the curve represents whether pre-training is used or not.

### *5.2. Transferability of Simulated Data*

We do another comparative study to illustrate the significance of knowledge learned from simulated data under four different training sets, including natural images, coherent imaging images, incoherent imaging images, and a mixture of coherent imaging images and incoherent imaging images. All the models are tested on the same real dataset. The evaluation metrics we used are precision, recall, F1_Score, AP, AP50, and AP75, as described in [19].

Table 3 shows the quantitative results of the adopted YOLOv5s backbone on four different training sets. From the table, the performance of the models training on any of the simulated data, no matter whether dealing with coherent imaging or incoherent imaging, is significantly better than the one trained on gray images. This is because the addition of the simulation process reduces the difference between the training set and the test set. These results show the effectiveness of knowledge learned from simulated data, which in turn proves the rationality of our simulation of a laser active imaging system. Furthermore, the performance of incoherent imaging is better than that of the other two situations. The reason for its promising performance is that the noise introduced by coherent imaging simulation destroys the contour and texture information of the UAVs, which is of critical importance for a deep learning object detection algorithm. In order to verify this, we use

the acknowledged peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) [32] metrics to measure the similarity between them and real data, as shown in Table 4. The SSIM assesses the visual impact of three characteristics of an image: luminance, contrast, and structure. The better the image similarity, the higher the PSNR and SSIM scores. We calculate the metrics between 10% of the training set and the whole real dataset, then take the mean value. It can be seen that the metrics scores agree with the model performance across different training sets, which confirms our explanation. Therefore, we will eventually use the incoherent imaging simulation method to generate the training set.

**Table 3.** The performance of the model on the test set with different training sets.

| Data Composition | Precision | Recall | F1-Score | AP | AP50 | AP75 |
|---|---|---|---|---|---|---|
| Gray images | 0.7486 | 0.6592 | 0.7011 | 0.2442 | 0.6802 | 0.0893 |
| Coherent imaging | 0.9755 | 0.6667 | 0.7921 | 0.3815 | 0.8094 | 0.2560 |
| Incoherent imaging | 1.0000 | 0.8847 | 0.9388 | 0.5344 | 0.9870 | 0.5350 |
| 50% coherent and 50% incoherent | 0.9903 | 0.8516 | 0.9157 | 0.5200 | 0.9610 | 0.5240 |

**Table 4.** The metrics between different training sets and the real dataset.

| Metric\Training Set | Gray Images | Coherent Imaging | Incoherent Imaging | 50% Coherent and 50% Incoherent |
|---|---|---|---|---|
| PSNR | 20.03 | 21.70 | 22.54 | 22.22 |
| SSIM | 0.39 | 0.61 | 0.83 | 0.75 |

### *5.3. GIoU Loss vs. IoU Loss*

In this subsection, we explore the advantages of *GIoU* loss over *IoU* loss based on an incoherent imaging training set. The results of the test set have been reported in Table 5. Moreover, Figure 7 shows the mAP values of the model trained with *IoU* and *GIoU* losses against different *IoU* thresholds, i.e., $IoU = \{0.5, 0.55, \ldots, 0.95\}$.

**Table 5.** Comparison of model performance between IoU and GIoU losses.

| Loss\Evaluation | AP | AP50 | AP75 |
|---|---|---|---|
| *IoU* | 0.489 | 0.886 | 0.448 |
| *GIoU* | 0.534 | 0.987 | 0.535 |
| Relative improv.% | 9.20% | 11.4% | 19.4% |

The result in Table 5 shows that we can improve the performance of the model significantly by using *GIoU* loss as the bounding box regression loss. As shown in Figure 7, consistent improvement can be obtained across different values of *IoU* thresholds. However, this improvement is not stable with the change of *IoU* threshold. Nevertheless, by incorporating *GIoU* loss into our algorithm, we can slightly improve UAV detection performance in the laser active imaging domain.

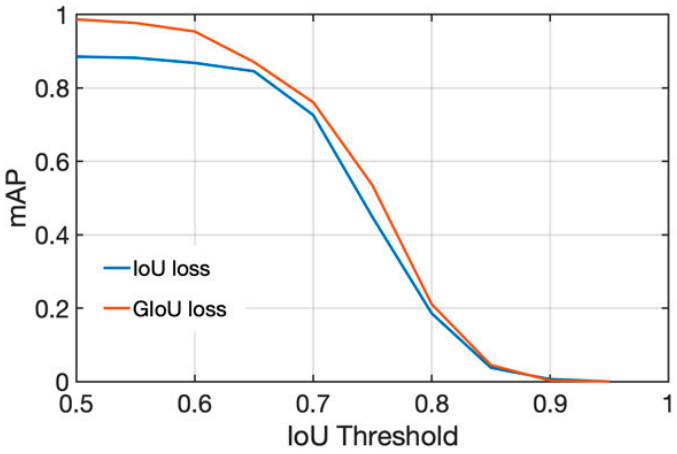



**Figure 7.** mAP values against different *IoU* thresholds for *IoU* loss and *GIoU* loss.

### *5.4. Comparison with the Previous Method*

In the experiments, we compared the adopted YOLOv5s model with several representative methods including HOG [33], DPM [34], and YOLOv3 [28]. The performance of the methods is evaluated in terms of both detection accuracy and detection speed. The experimental results are shown in Table 6. The manually designed feature-based HOG and DPM are implemented using CPU; the CNN-based YOLOv3 and YOLOv5s are implemented using GPU. From the comparison of these methods, we can observe that the algorithm based on CNN is higher than the traditional algorithm by a large margin. For the aspect of detection speed, the HOG and DPM methods are inefficient during the redundancy of a sliding window strategy while the two YOLO methods can achieve real-time detection performance. Furthermore, our adopted YOLOv5s model is more than three times faster than YOLOv3 while maintaining comparable detection accuracy. The fast detection benefits from small model size. The model size of YOLOv5s is 14.3 MB, which is much smaller than the 243.7 MB size of YOLOv3. Therefore, the YOLOv5s was selected as the basic framework, which is sufficient to meet the requirements of our laser active imaging system.

**Table 6.** The method comparison in terms of the detection accuracy and detection speed.

| Method | Precision | Recall | F1-Score | Device | FPS |
|---|---|---|---|---|---|
| HOG | 0.247 | 0.436 | 0.315 | Intel Core i7-7700K | 1.484 |
| DPM | 0.3404 | 0.598 | 0.434 | Intel Core i7-7700K | 0.816 |
| YOLOv3 | 1.000 | 0.888 | 0.941 | GeForce GTX 1080 | 29.412 |
| YOLOv5s | 1.000 | 0.884 | 0.938 | GeForce GTX 1080 | 104.167 |

### *5.5. Experimental Results on Laser Active Imaging System*

Figure 8 shows the detection results of the model on the data accumulated by our laser active imaging system. The results show that our algorithm can accurately detect the UAV in different backgrounds and states. Compared with buildings and trees, there is no background reflection signal in the sky. The camera will only present the illumination image of the UAV without an illumination area, which is different from our simulated data. Nevertheless, our method can obtain satisfactory detection results in this situation. Furthermore, the model can also detect UAVs in flight, even when its score is lower than that of the UAV in hover. In the last one, our method is still effective, although the imagery

suffers from motion blur due to the flight of the UAV. In summary, our algorithm is a useful UAV detection method for our laser active imaging system.

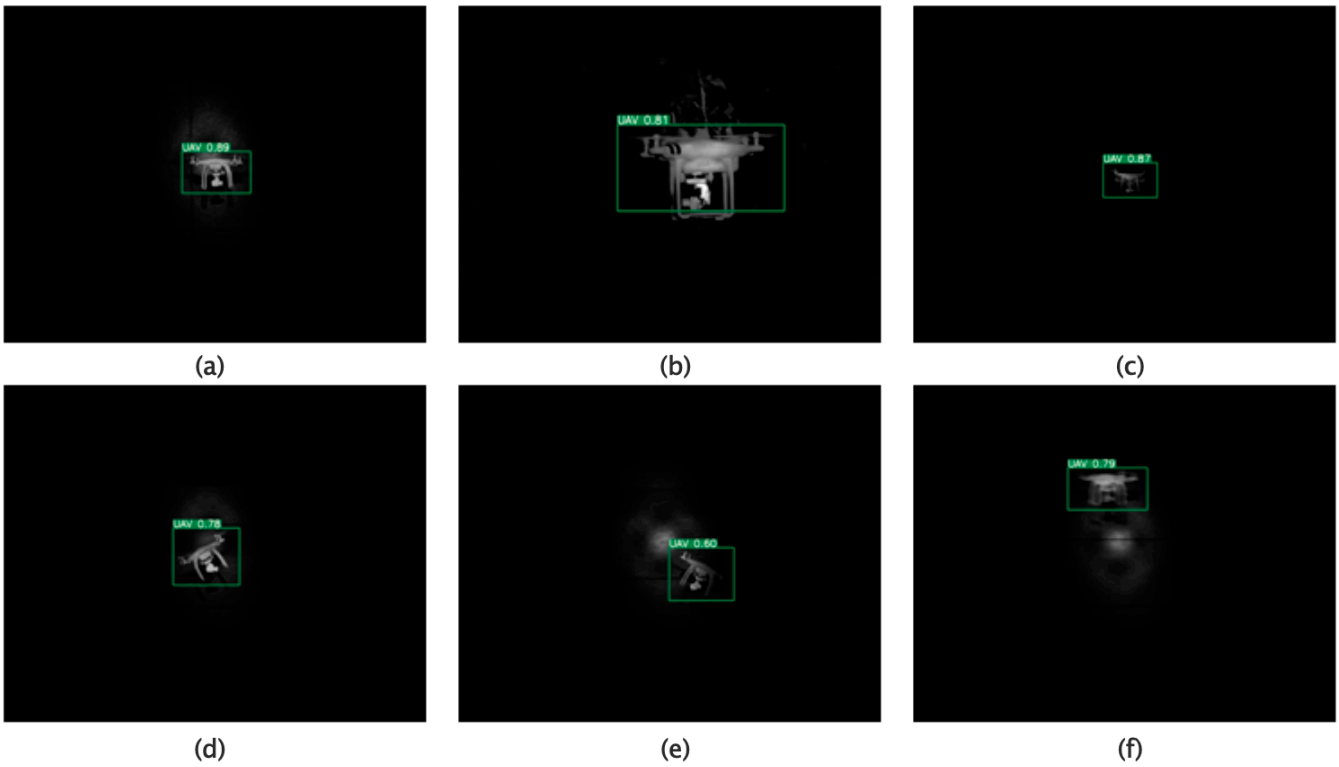



**Figure 8.** Detection results of our approach on real laser active imaging data. The top row presents the detection results of UAVs hovering in different backgrounds, from left to right are buildings (**a**), trees (**b**), and the sky (**c**). The bottom row (including **d**–**f**) shows the detection results when the UAV is at different flight attitudes with buildings in the background.

## 6. Conclusions

In this paper, we propose a high-precision, real-time UAV detection method to be used in the field of laser active imaging by combining a deep CNN-based object detection algorithm with transfer learning, which provides the basis for the following tracking and monitoring. For specific tasks like small-scaled UAV detection of laser active imaging, in which few samples can be used for training, the performance of the model degrades sharply. To solve this problem, we firstly embedded prior knowledge learned from a large available dataset into the model. Then, we provided sufficient data for the specific task by simulating the imaging process of laser active illumination. The experiment carried out on our real laser active imaging system demonstrates the transferability of simulated data and the effectiveness of our solution.

However, limited by the laser power, we can only collect targets within 500 m. In future work, bigger and more realistic datasets will be gathered, including longer distances, more complex scenes such as occlusions by trees, and confusing objects such as birds. Further work that can be carried out is to add the depth information of the target plane in the simulation process, which will contribute to producing more realistic images. This measure has great potential to improve the performance of small-scale UAV detection in the laser active imaging domain.

**Author Contributions:** Conceptualization, S.Z. and G.Y.; methodology, S.Z.; software, S.Z.; validation, S.Z. and T.S.; formal analysis, S.Z.; investigation, S.Z.; resources, T.S.; data curation, S.Z.; writing—original draft preparation, S.Z.; writing—review and editing, G.Y. and K.D.; visualization, S.Z.; supervision, J.G.; project administration, K.D.; funding acquisition, J.G. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported by the National Natural Science Foundation of China (#61675200, #61904178).

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Gaszczak, A.; Breckon, T.P.; Han, J. Real-time people and vehicle detection from UAV imagery. *Proc. SPIE Int. Soc. Opt. Eng.* **2011**, *7878*, 78780B.
2. Wang, J.; Jiang, C.; Han, Z.; Ren, Y.; Maunder, R.G.; Hanzo, L. Taking Drones to the Next Level: Cooperative Distributed Unmanned-Aerial-Vehicular Networks for Small and Mini Drones. *IEEE Veh. Technol. Mag.* **2017**, *12*, 73–82. [CrossRef]
3. Li, B.; Yang, Z.P.; Chen, D.Q.; Liang, S.Y.; Ma, H. Maneuvering target tracking of UAV based on MN-DDPG and transfer learning. *Def. Technol.* **2020**, *17*, 457–466. [CrossRef]
4. Busck, J. Underwater 3-D optical imaging with a gated viewing laser radar. *Opt. Eng.* **2005**, *44*, 6001. [CrossRef]
5. Espinola, R.L.; Jacobs, E.L.; Halford, C.E.; Vollmerhausen, R.; Tofsted, D.H. Modeling the target acquisition performance of active imaging systems. *Opt. Express* **2007**, *15*, 3816–3832. [CrossRef]
6. Wang, C.; Sun, T.; Wang, T.; Chen, J. Fast contour torque features based recognition in laser active imaging system. *J. Light Electronoptic* **2015**, *126*, 3276–3282. [CrossRef]
7. Zhao, Z.Q.; Zheng, P.; Xu, S.T.; Wu, X. Object Detection With Deep Learning: A Review. *IEEE Trans. Neural Netw. Learn. Syst.* **2019**, *30*, 3212–3232. [CrossRef] [PubMed]
8. Krizhevsky, A.; Sutskever, I.; Hinton, G. ImageNet Classification with Deep Convolutional Neural Networks. *Adv. Neural Inf. Process. Syst.* **2012**, *25*, 1097–1105. [CrossRef]
9. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv* **2014**, arXiv:1409.1556.
10. Lin, T.Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature Pyramid Networks for Object Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017.
11. Ronneberger, O.; Fischer, P.; Brox, T. U-Net: Convolutional Networks for Biomedical Image Segmentation. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Munich, Germany, 5–9 October 2015.
12. Pan, S.J.; Yang, Q. A Survey on Transfer Learning. *IEEE Trans. Knowl. Data Eng.* **2010**, *16*, 1345–1359. [CrossRef]
13. Yang, Z.; Yu, W.; Liang, P.; Guo, H.; Xia, L.; Zhang, F.; Ma, Y.; Ma, J. Deep transfer learning for military object recognition under small training set condition. *Neural Comput. Appl.* **2019**, *31*, 6469–6478. [CrossRef]
14. Deng, J.; Dong, W.; Socher, R.; Li, L.J.; Li, F.F. ImageNet: A large-scale hierarchical image database. In Proceedings of the IEEE Conference on Computer Vision & Pattern Recognition, Miami, FL, USA, 20–25 June 2009.
15. Yang, Y.; Yan, L.F.; Zhang, X.; Han, Y.; Nan, H.Y.; Hu, Y.C.; Hu, B.; Yan, S.L.; Zhang, J.; Cheng, D.L.; et al. Glioma Grading on Conventional MR Images: A Deep Learning Study With Transfer Learning. *Front. Neurosci.* **2018**, *12*, 804. [CrossRef]
16. Zhu, L.; Zhang, S. Multilevel Recognition of UAV-to-Ground Targets Based on Micro-Doppler Signatures and Transfer Learning of Deep Convolutional Neural Networks. *IEEE Trans. Instrum. Meas.* **2020**, *70*, 1–11. [CrossRef]
17. Sommer, L.; Schumann, A.; Muller, T.; Schuchert, T.; Beyerer, J. Flying object detection for automatic UAV recognition. In Proceedings of the IEEE International Conference on Advanced Video & Signal Based Surveillance, Lecce, Italy, 29 August–1 September 2017.
18. Zhao, Y.; Yi, S. Cyclostationary Phase Analysis on Micro-Doppler Parameters for Radar-Based Small UAVs Detection. *IEEE Trans. Instrum. Meas.* **2018**, *67*, 2048–2057. [CrossRef]
19. Lin, T.Y.; Maire, M.; Belongie, S.; Hays, J.; Zitnick, C.L. *Microsoft COCO: Common Objects in Context*; Springer: Cham, Switzerland, 2014.
20. Driggers, R.G. Impact of speckle on laser range-gated shortwave infrared imaging system target identification performance. *Opt. Eng.* **2003**, *42*, 738–746. [CrossRef]
21. Ge, W.L.; Zhang, X.H. Design and implementation of range-gated underwater laser imaging system. *Int. Soc. Opt. Photonics* **2014**, *9142*, 914216.
22. Glenn, J.; Liu, C.; Adam, H.; Yu, L.; changyu98; Rai, P.; Sullian, T. Ultralytics/yolov5: Initial Release (Version v1.0). Zenodo. Available online: http://doi.org/10.5281/zenodo.3908560 (accessed on 13 July 2020).
23. Yosinski, J.; Clune, J.; Bengio, Y.; Lipson, H. How transferable are features in deep neural networks. In *International Conference on Neural Information Processing Systems*; MIT Press: Palais des Congrès de Montréal, MO, Canada, 2014.
24. Öztürk, A.E.; Erçelebi, E. Real UAV-Bird Image Classification Using CNN with a Synthetic Dataset. *Appl. Sci.* **2021**, *11*, 3863. Available online: https://www.mdpi.com/2076-3417/11/9/3863 (accessed on 21 May 2021). [CrossRef]
25. Liu, J.M. Simple technique for measurements of pulsed Gaussian-beam spot sizes. *Opt. Lett.* **1982**, *7*, 196. [CrossRef]

26. Ioffe, S.; Szegedy, C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In Proceedings of the 32nd International Conference on International Conference on Machine Learning, Lille, France, 6–11 July 2015; Volume PMLR 37, pp. 448–456.
27. Maas, A.L.; Hannun, A.Y.; Ng, A.Y. Rectifier nonlinearities improve neural network acoustic models. *Proc. Icml* **2013**, *30*, 3.
28. Redmon, J.; Farhadi, A. YOLOv3: An Incremental Improvement. *arXiv* **2018**, arXiv:1804.02767.
29. Wang, C.Y.; Liao, H.Y.M.; Wu, Y.H.; Chen, P.Y.; Yeh, I.H. CSPNet: A New Backbone That Can Enhance Learning Capability of CNN. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Seattle, WA, USA, 14–19 June 2020.
30. Liu, S.; Qi, L.; Qin, H.; Shi, J.; Jia, J. Path Aggregation Network for Instance Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18-23 June 2018.
31. Rezatofighi, H.; Tsoi, N.; Gwak, J.Y.; Sadeghian, A.; Savarese, S. Generalized Intersection Over Union: A Metric and a Loss for Bounding Box Regression. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019.
32. Wang, Z.; Bovik, A.C.; Sheikh, H.R.; Simoncelli, E.P. Image Quality Assessment: From Error Visibility to Structural Similarity. *IEEE Trans. Image Process.* **2004**, *13*, 600–612. [CrossRef] [PubMed]
33. Dalal, N. Histograms of Oriented Gradients for Human Detection. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Diego, CA, USA, 20–25 June 2005.
34. Felzenszwalb, P.F.; Girshick, R.B.; McAllester, D.; Ramanan, D. Object Detection with Discriminatively Trained Part-Based Models. *IEEE Trans. Pattern Anal. Mach. Intell.* **2010**, *32*, 1627–1645. [CrossRef] [PubMed]