Journal of Systems Engineering and Electronics
Vol. 34, No. 5, October 2023, pp.1309 – 1318

Scene image recognition with knowledge transfer for drone navigation

DU Hao[1,2], WANG Wei[2,3], WANG Xuerao[1], ZUO Jingqiu[2], and WANG Yuanda[1,*]

1. School of Automation, Southeast University, Nanjing 210096, China; 2. Autonomous Control Robot Laboratory, Jiangsu Zhongke Institute of Applied Research on Intelligent Science and Technology, Changzhou 213164, China; 3. Jiangsu Collaborative Innovation Center of Atmospheric Environment and Equipment Technology, Nanjing University of Information Science and Technology, Nanjing 210044, China

Abstract: In this paper, we study scene image recognition with knowledge transfer for drone navigation. We divide navigation scenes into three macro-classes, namely outdoor special scenes (OSSs), the space from indoors to outdoors or from outdoors to indoors transitional scenes (TSs), and others. However, there are difficulties in how to recognize the TSs, to this end, we employ deep convolutional neural network (CNN) based on knowledge transfer, techniques for image augmentation, and fine tuning to solve the issue. Moreover, there is still a novelty detection problem in the classifier, and we use global navigation satellite systems (GNSS) to solve it in the prediction stage. Experiment results show our method, with a pre-trained model and fine tuning, can achieve 91.3196% top-1 accuracy on Scenes21 dataset, paving the way for drones to learn to understand the scenes around them autonomously.

Keywords: scene recognition, convolutional neural network, knowledge transfer, global navigation satellite systems (GNSS)-aided.

DOI: 10.23919/JSEE.2023.000096

1. Introduction

Humans can perceive and understand the scenes around them because of their vision, it is one of the most natural and magical things for adults to recognize the scene at a glance. In the past, drones only paid attention to their coordinates but not the "scenery" around them. With the development and application of convolutional neural network (CNN) and graphics processing unit (GPU), it has become possible for drones to recognize the environment around them like humans. The corresponding goal of computer vision is to achieve scene recognition, object detection, and image segmentation through digital methods that can reach to a similar effect of human vision. In the past, extensive research has been conducted by scholars and significant advancements have been achieved. Li and Hu [1] proposed a distributed CNN architecture and pre-training approach for remote sensing image target classification. Lu et al. [2] proposed a novel channel called the l-channel based on conventional red green blue (RGB) images to enhance the performance of object recognition. In addition, the object detection frameworks based on deep learning that have a profound impact include Fast R-CNN [3], Faster R-CNN [4], Yolov3 [5], Yolo9000 [6], Sdcnet [7], etc. It has made huge progress in object detection tasks stems from the rise of CNN and the public datasets recently [8–10]. In the domain of scene recognition, a divide and conquer clustering strategy based on scene perception is proposed to cluster the motion crowd [11]. Yang et al. [12] proposed a novel latent topic model to learn and recognize scenes and places . Eslami et al. [13] proposed the generative query network (GQN), a framework without human labels or domain knowledge. Furthermore, the impact of convolutional network depth on its accuracy in large-scale image recognition settings has also been investigated. The studies have shown that when the depth of weight layer reaches 16−19, significant enhancements can be achieved compared to existing techniques [14]. However, these studies have revealed a paucity of research studies on the application of scene recognition for the navigation of drones [15−17]. For object detection, we only need to recognize the objects in the images or videos, but for scene recognition, we also need to analyze which category it belongs to according to the ambient content, layout and context reasoning, e.g., if an animal appears in

Manuscript received February 16, 2022.
*Corresponding author.
This work was supported by the National Natural Science Foundation of China (62103104), the Natural Science Foundation of Jiangsu Province (BK20210215), and the China Postdoctoral Science Foundation (2021M690615).

the woods, the scene may belong to "mountain forest", "zoo" or "forest path", a bed inside a room may belong to a family or a hotel. In other words, the object detection focuses on the foreground (positive samples, objects) of the image, while scene recognition focuses more on the background (negative samples), this is one of the thorny problems of scene recognition. Although there is much literature on scene recognition in the past, there is still no research on scene recognition for drone navigation. Consequently, the research motivation of this paper is to explore a scene recognition that is suitable for drone navigation. Generally, we can apply vision-based simultaneous localization and mapping (SLAM) or light detection and ranging (LiDAR)-based SLAM for drone navigation in indoor scenes, and global navigation satellite systems (GNSS) for drone autonomous navigation in outdoor scenes. The signal of GNSS performs well in outdoor scenes, but there is no signal in indoor scenes, the worst case is that the signal is discontinuous in the transitional scenes (TSs), which is fatal to the drone. If the drone can perceive the flight scene in advance, and then switch different navigation modes in time, e.g., visual-based SLAM navigation, LiDAR-based SLAM navigation or multi-sensor fusion navigation mode, it will greatly improve the flight safety of the drone.

Firstly, we train the model based on CNN backbone with transfer learning on our dataset Scenes21 which is based on [18]. Secondly, we judge whether the result of the test stage is correct through the novelty detection module. The certain category will be output directly if its probability is high, otherwise, we will make further judgment through the signal strength of the GNSS. The key contributions of this paper are as follows: First, we employ a knowledge transfer method [19] based on ResNet [20] and ResNeXt [21], use the pre-trained weights to initialize the network, then modify the classifier and retrain the model with fine-tuning. Second, dataset Scenes21 is created based on Places365 dataset [18], we reclass the dataset into three macro-classes according to the drone navigation task, namely outdoor special scenes (OSSs), TSs and others, especially redefine the TSs category. In addition, we expand some classes of the dataset with data augmentation. Finally, we use GNSS to assist in solving the problem of scene classification novelty detection and other scenes recognition in the evaluation stage.

Next, we present the method details of this paper in Section 2. In Section 3, we describe the experimental platform and model training details. The experimental results are analyzed in Section 4. In Section 5, we give the conclusions and look forward to future work.

2. Method

The method of scene image recognition is shown in Fig. 1.

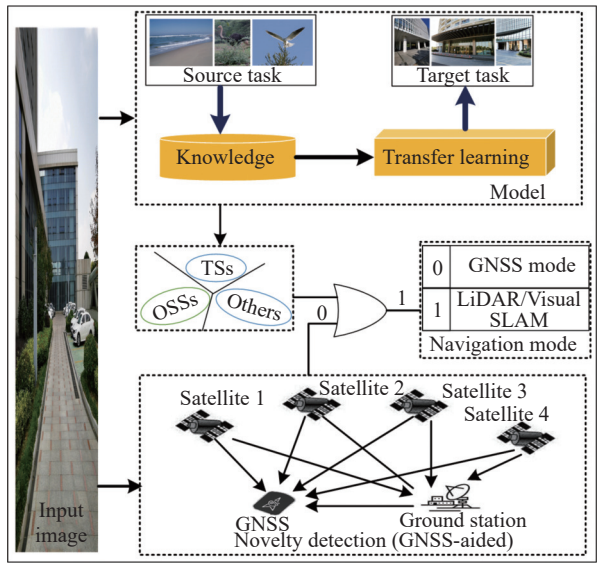


Fig. 1 Method of scene image recognition

2.1 Novelty detection

The dataset is a set of image samples x_i with corresponding class labels ω_i. Bayes decision rule is an important method of pattern recognition [22], which can be written as

$$x \in \omega_i \leftrightarrow P(x|\omega_i)P(\omega_i) > P(x|\omega_j)P(\omega_j), \quad \forall j \neq i \tag{1}$$

where $P(x|\omega_i)$ is referred to as the likelihood function of ω_i with respect to x, $P(\omega_i)$ is the a priori probability, $P(x|\omega_j)$ and $P(\omega_j)$ are the same as above [23]. The Bayes decision rule in (1) assigns every sample x_i to one class ω_i. The scene classifications in the dataset are only part of the scenes we may encounter, and the reality scenes are more complex, the drone may encounter scenes that do not belong to any training dataset when it is flying, how to divide this scene is what we call novelty detection, the next problem is how to estimate $P(x|\omega_i)$. Theoretically, we can solve this problem by calculating the histogram of sample feature vectors of the training dataset, however, due to the so-called curse of dimensionality, this method cannot be used in practice. The single-layer perception classifiers are simple, but their classification capabilities are very limited. As described in [22], the multilayer perceptron (MLP) and CNNs are inherently incapable of novelty detection, it is usually necessary to collect samples to form an explicit rejection class to equip MLP and CNNs with the capability of novelty

detection. To solve this problem, a dataset and CNN-based model are required, which will be discussed in Subsection 2.2 and Subsection 2.3.

2.2 Datasets with image enhancement

All classifiers need a method to obtain probabilities or segmented hypersurfaces, to achieve this purpose, a training dataset is needed [22]. In the paper, we propose a trimmed dataset named Scenes21 based on the Places365 dataset [18] according to the task of drone navigation. The dataset Scenes21 is divided into three macro-classes: OSSs, TSs, and others, there are about 100 000 images in the Scenes21 dataset after reclassification. The "TSs" include doorway, exterior, interior, building facade and porch. The key features of "TSs" is the sudden disappearance or sudden appearance of GNSS signals, because the drone is flying from indoor to outdoor, or from outdoor to indoor, so the signals in these areas are unstable. The "OSSs" include 16 categories, e.g., airfield, alley, forest path and street, indoor and other scenes are classified as others. As mentioned earlier, the problem of multi-path signals of GNSS in urban canyon areas, such as alleys, skyscrapers, viaducts [24]. Besides that, the signal of GNSS is weak or unstable in the OSSs, e.g., forest path, broadleaf, and forest_road. We discover in the Places365 dataset that a scene image can depict multiple independent categories such as "building facade" and "exterior". We reclassified the two categories into one category named "transition exterior". In addition, we assign OOSs, indoor and other unpredictable scenes to the others uniformly. The key features of the "OOSs" are wide space, good vision, and good GNSS signal, and there is no GNSS signal for the drone in indoor scenes. The indoor scenes and the "OOSs" can be judged according to the signal quality of GNSS in the prediction stage. The detailed classification of Scenes21 is shown in Fig. 2.

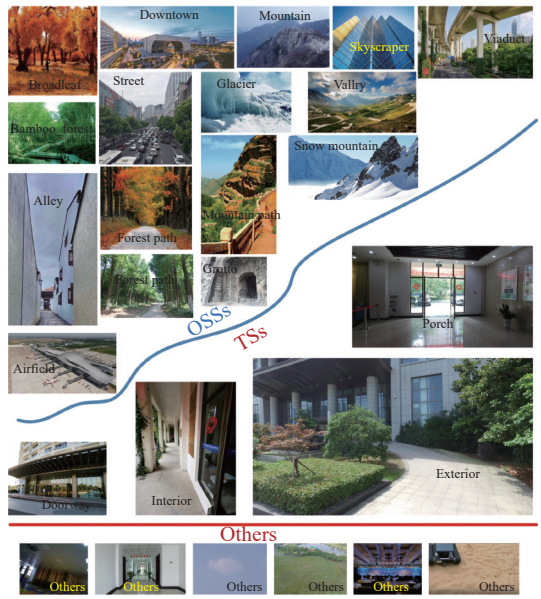


Fig. 2 Scenes21 dataset

The datasets we use often have noise due to the influence of random error and the error in the measuring camera. We can expand the scale of training datasets and reduce the influence of noise through the techniques for image enhancement that can reduce the dependence of the model on some attributes of images, to improve the generalization ability of the model. An image is composed of a limited number of pixels, which reflect the brightness at a specific position of the image, so we can reduce the sensitivity of the model to contrast by adjusting the brightness. In addition, the common techniques include flipping images horizontally or vertically, clipping, color transformation, expansion and rotation. The left figures show the data enhancement results of the middle-upper figure, and the right figures show the data enhancement results of the middle-lower image in Fig. 3.

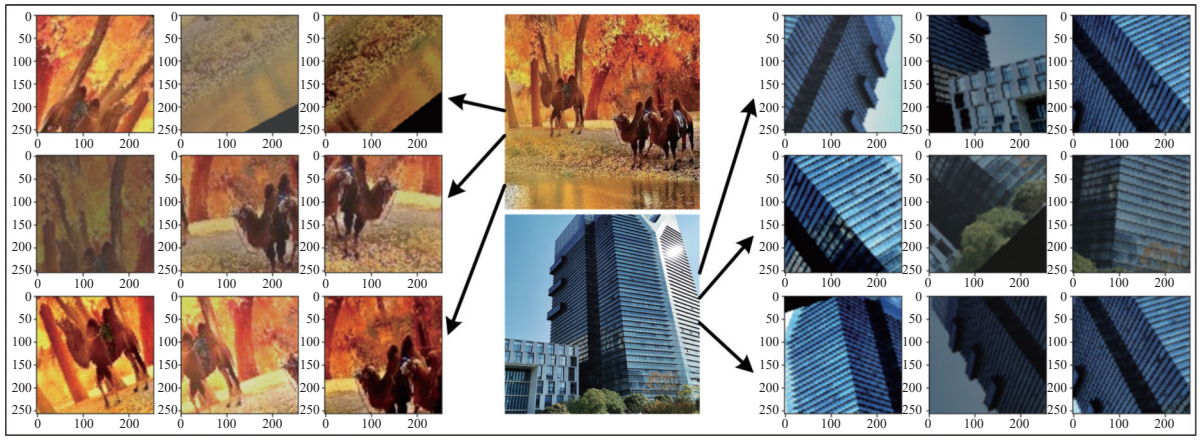


Fig. 3 Raw image and the images are processed by combining multiple techniques for image augmentation

2.3 Knowledge transfer

The scene image classification for drone navigation is the focus of this paper. Currently, the public largescale scene training model Places-CNN and ImageNet-CNNs (e.g., visual geometry group (VGG), ResNet, Inception-v3, ResNeXt, Wide ResNet) cannot fully meet our requirements. Moreover, due to the limitation of computing capability for our GPU, it takes a lot of time to retrain the model on the public large-scale datasets. This paper tries two transfer learning methods on our newly classified Scenes21 dataset, one is to train only the full connection layer after freezing all weights for all other networks, the other is to train all parameters after loading the weights of the pre-trained model. We compare the error and recognition accuracy of the two methods respectively. Then, we retrain two types of backbones based on the transfer learning method with good performance, one based on ResNet, and the other based on ResNeXt, and analyze the performance of the Scenes21-ResNet by comparing its error and accuracy with the Scenes21-ResNeXt backbone in this paper.

The detailed architecture of Scenes21-ResNet with ResNet-101 is shown in Fig. 4, b1 represents the bottleneck without shortcut convolution and stripe=1, b2 stands for the bottleneck with shortcut convolution and stripe=1, b3 stands for the bottleneck with shortcut convolution and stripe=2, bottleneck block was originally proposed in [20]. The bottleneck has three convolution layers, and the convolution kernel size is 1×1, 3×3, and 1×1. The number of input and output channels of the 3×3 convolution layer is less than that of 1×1, to realize efficient calculation. The full connection lay is changed from 1 000 to 21 categories based on our dataset Scenes21.

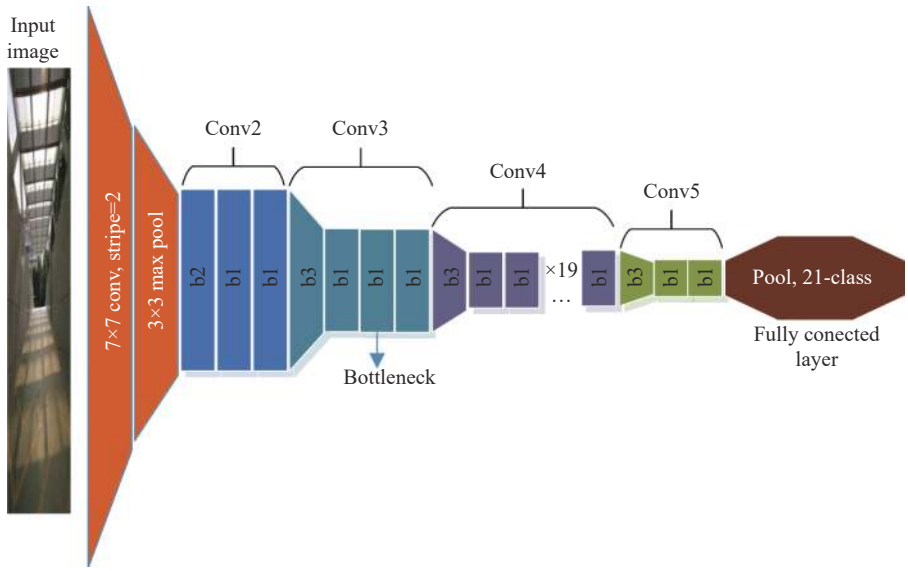


Fig. 4 Detailed architecture of Scenes21-ResNet with ResNet-101

2.4 GNSS-aided

We discussed that the classifier model based on CNN itself does not have the capacity for novelty detection [22] in Subsection 2.1. In addition to the method of adding an anomaly class others introduced above, this paper also relies on the GNSS to assist novelty detection. In the prediction stage, firstly, we obtain the confidence of the scene classification based on the trained model. If the classification belongs to the OSSs or TSs, we set M_1=1, in other cases, we set M_1=0.

$$M_1 = \begin{cases} 1, & \omega_i \in \text{OSSs or } \omega_i \in \text{TSs} \\ 0, & \text{others} \end{cases} \tag{2}$$

Secondly, we simulate the GNSS signal strength of the actual scenes to assist the decision-making. When the number of satellites received by the GNSS receiver is less than λ_T, we set $M_2 = 1$, and when the number of satellites is higher than λ_T, we set $M_2 = 0$, the threshold of λ_T is usually set to 13.

$$M_2 = \begin{cases} 1, & N_{\text{star}} < \lambda_T \\ 0, & N_{\text{star}} \geqslant \lambda_T \end{cases} \tag{3}$$

Then, we perform OR logic operates for M_1 and M_2, the output is either 0 or 1. If the output M = 1, it means the navigation is based on LiDAR and SLAM or Visual-SLAM. If it is 0, it means navigation based on GNSS.

The corresponding truth table for the OR operation is given in Table 1. By doing so, when the scenes belong to the OSSs or belong to the TSs, even if the GNSS signal is good, the navigation mode will immediately switch to LiDAR-SLAM or Vision-SLAM, which can avoid drone crashes or safety accidents.

Table 1 Truth table for OR operates

M_1	M_2	M(OR)	Mode
0	0	0	GNSS (BDS/GPS/GLONASS)
0	1	1	LiDAR-SLAM or Vision-SLAM
1	0	1	LiDAR-SLAM or Vision-SLAM
1	1	1	LiDAR-SLAM or Vision-SLAM

2.5 Loss function and optimization algorithm

The loss function is also called the cost function, which is applied to measure the error of the model. The smaller the value of the loss function is, the better the model and parameters conform to the training dataset. Therefore, the process of training the model is the process of optimizing the loss function. The research goal of this paper is to classify scene images, we assume that the number of samples in the training dataset is N, so we use the cross-entropy loss function

$$J = -\frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{M} y_k(i)\log \hat{y}_k(i) \tag{4}$$

where M is the number of label classifications, $y_k(i)$ represents the one-hot encoding of the k label of sample i. If the label given by the training data is $k(k=0,1,\cdots,M-1)$, then $y_k(i)=1$, and the others are 0. $\hat{y}_k(i)$ represents the prediction probability that sample i belongs to category k. In order to transform $\hat{y}_k(i)$ into a probability distribution that lie in the range [0, 1] and sum to 1, we use SoftMax function [25], given by

$$\hat{y}_k(i) = \frac{\exp(z_k(i))}{\sum_{k=0}^{M}\exp(z_k(i))} \tag{5}$$

where $z_k(i)$ is the output of classification k of sample i. We can use the optimizer to optimize the model based on the loss function. This article uses momentum based stochastic gradient descent (SGD). The momentum calculation method [26,27] is expressed as

$$\begin{cases} m_{t+1} = \gamma m_t - \xi \nabla f(J_t) \\ J_{t+1} = J_t + m_{t+1} \end{cases} \tag{6}$$

where ξ is the learning rate, $\gamma \in [0,1]$ is coefficient of the momentum, and $\nabla f(J_t)$ is the gradient at J_t. In addition to the SGD optimization algorithm, we also use the adaptive moment (ADAM) [28] estimation algorithm. In the early training stage, ADAM algorithm is used in the early stage of training, then we switch to the SGD optimization method after getting better parameters.

3. Implementation details

Our implementation is based on [20,21] and the publicly available code [18] in the deep learning framework PyTorch. On the trimmed Scenes21 dataset, the input image is 224×224 randomly cropped and flipped horizontally from a resized image using the scale and center. Firstly, we change the fully connected layer to 21 categories. One method is to freeze the parameters of all layers except the fully connected layer, initialize with the parameters of the ResNet network, and train only the parameters of the fully connected layer. Another method is to initialize with the parameters of the ResNet model, retrain all layers. Then we compare the errors of the two methods, select the method with the smaller error for retraining. We use SGD optimization method [26,27] with a batch size of 64 on one GPU (RTX 2 080 Ti) for ResNet-18 and ResNet-34, a batch size of 32 for ResNet-50 and ResNeXt-50, and compare which optimization method is better. The weight decay is 0.000 1 and the momentum is 0.9. We start from a learning rate of 0.1, and then decayed it by 10 every 30 epochs in the training [18]. Then we retrain by adjusting the initial learning rate to 0.01 and 0.001. Based on the ADAM optimization method, we train ResNet-34 with initial learning rates of 0.01 and 0.001, respectively, and ResNet-50 with initial learning rates of 0.001 and 0.000 1, respectively. We also consider the fusion use of ADAM and SGD optimization algorithms, for example, we first use ADAM with an initial learning rate of 0.000 1 and then use SGD optimization algorithm with an initial learning rate of 0.001 for ResNet-34 and ResNet-50. We first use ADAM optimization algorithm for training, obtain better parameters, and then switch to SGD with momentum optimization method to achieve the best performance. Finally, we retrain the model based on the ResNeXt [21] with the ADAM optimization algorithm and transfer learning method, and get which method and model has smaller error and a higher accuracy.

4. Experimental results

4.1 Scene image classification on Scenes21 dataset

We conduct comparative experiments on the 21-class Scenes21 classification task based on [18]. We follow two methods to compare the error base on 18-layer, 34-layer and 50-layer residual networks. One method is to freeze the parameters of all layers except the fully con-

nected layer, and train only the parameters of the fully connected layer, the other method is to train all layers. Both methods are initialized with the parameters of pre-trained model ResNet. We can get the experiment results shown in Fig. 5 after training.

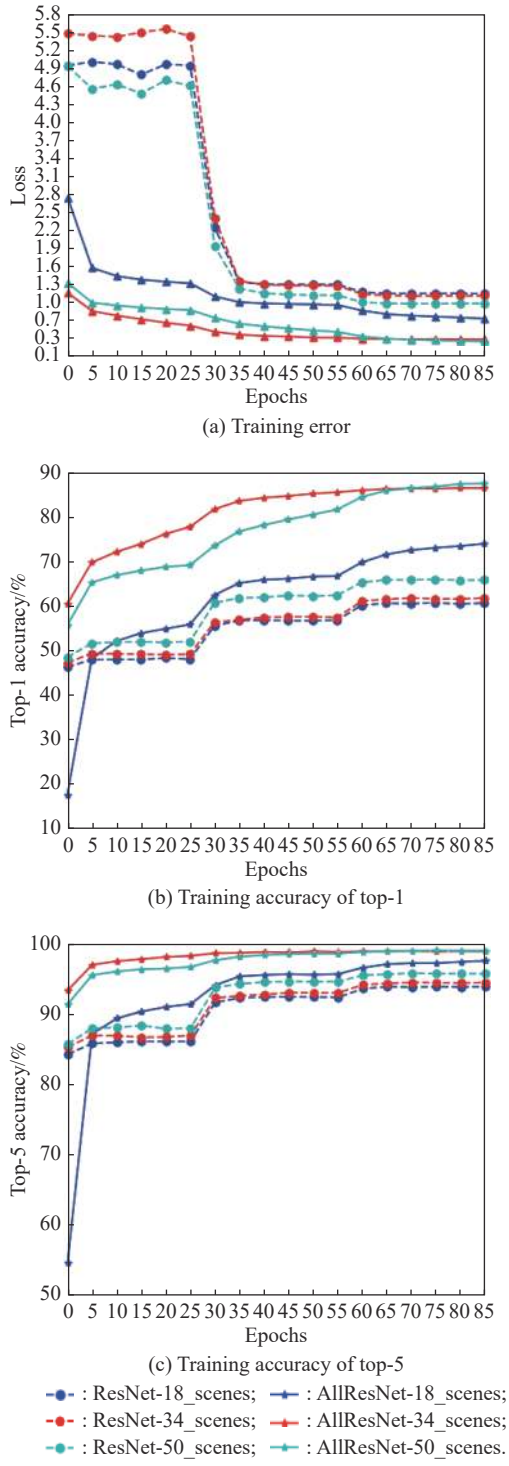
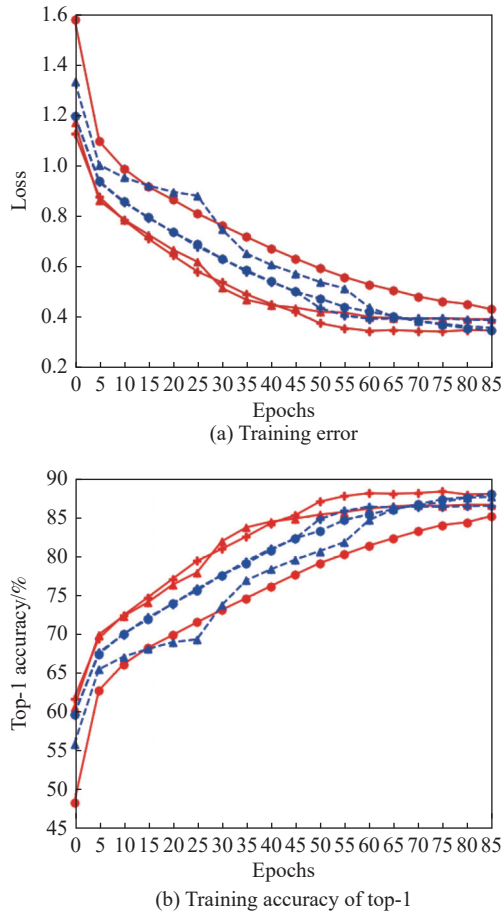


Fig. 5 Training on Scenes21 dataset with different transfer learning method

In Fig. 5, we show the training results based on the Scenes21 dataset, and all models are trained using only the SGD optimization algorithm. The weight decay is 0.000 1 and the momentum is 0.9. We start from a learning rate of 0.1, and then decay it by 10 every 30 epochs in the training stage. The blue dashed line, the red dashed line and the cyan dashed line represent the errors and the accuracy of top-1 and top-5 based on ResNet-18, ResNet-34 and ResNet-50 with only the fully connected layer is trained, respectively, while the blue solid line with triangles, the red solid line and the cyan solid line represent the errors based on ResNet-18, ResNet-34 and ResNet-50, respectively, and all layers are trained. We have found that training all parameters of the model has smaller error and higher accuracy from the experimental results. In the following experiments, we train all parameters of the model and compare the error and accuracy of different optimization methods. The experimental results are shown in Fig. 6.



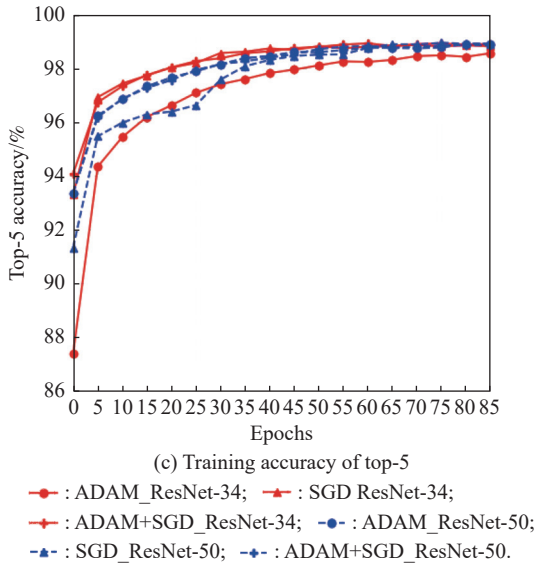


Fig. 6 Training on Scenes21 dataset with different optimization method

We apply the optimization algorithms of SGD, ADAM and the fusion of ADAM and SGD based on ResNet-18, ResNet-34 and ResNet-50, and compare which method is better. In Fig. 6, the solid red lines with circles, triangles and crosses represent the errors of using ADAM, SGD and a fusion of ADAM and SGD optimization algorithms and training based on ResNet-34 network, respectively. The blue dashed lines with circles, triangles and crosses represent the errors and the accuracy of top-1 and top-5 of using ADAM, SGD and a fusion of ADAM and SGD optimization algorithms and training based on ResNet-50 network, respectively.

In Fig. 7, we show the training results based on the Scenes21 dataset with ResNet and ResNeXt, and the backbones are trained using only the ADAM optimization algorithm. We start from a learning rate of 1e−4, betas = (0.9, 0.999), eps = 1e−8. The blue solid line with circles and the red dashed line with circles represent the errors and the accuracy of top-1 and top-5 based on ResNet-50 with ADAM optimization, respectively. The error of the ResNet-50 and ResNeXt-50 with ADAM is 0.3488 and 0.2567. The experimental results show that the ResNeXt backbone has smaller error and higher accuracy.

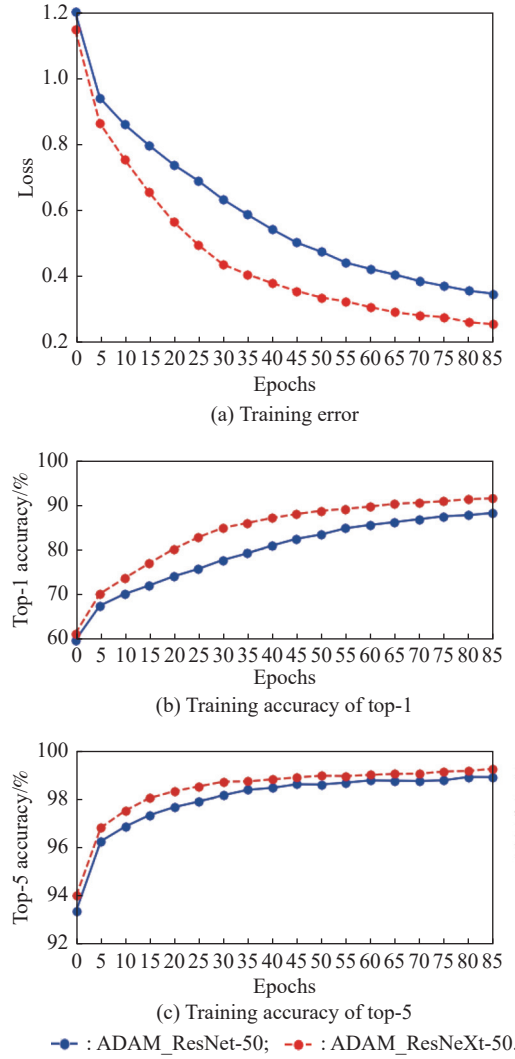


Fig. 7 Training on Scenes21 dataset with the backbone of ResNet and ResNeXt

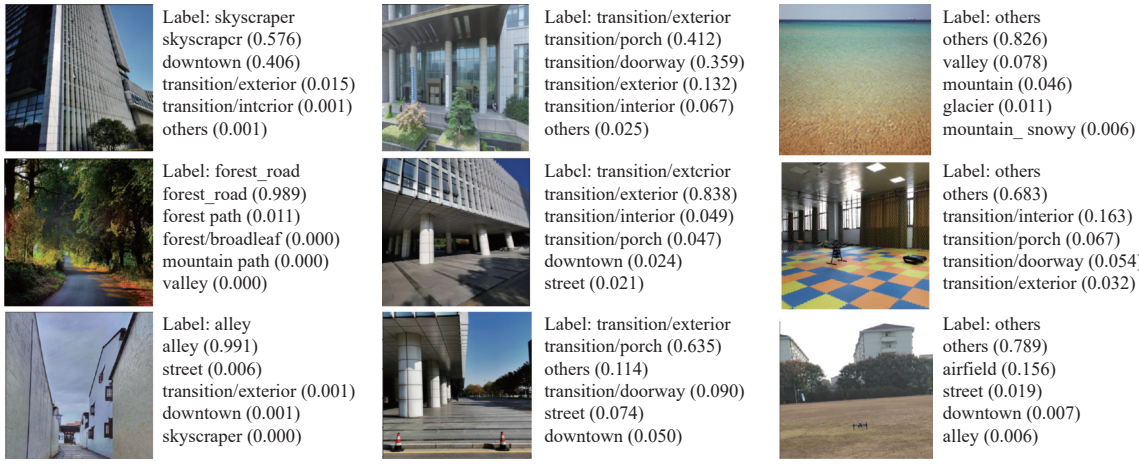
In order to analyze the experimental results of Fig. 6 and Fig. 7 more clearly, we show the results and training parameters in Table 2.

Table 2 Error and accuracy of the trained model based on Scenes 21 dataset

Method	Batch-size	Learning rate	Error	Top-1 acc/%	Top-5 acc/%
ResNet-34 with ADAM	64	0.001	0.4333	85.1582	98.5694
ResNet-34 with SGD	64	0.001	0.3939	86.6048	98.8251
ResNet-34 with ADAM + SGD	64	ADAM:1e−4, SGD:1e−3	0.3498	88.0166	98.9125
ResNet-50 with ADAM	32	0.0001	0.3488	88.0570	98.9162
ResNeXt-50 with ADAM	32	0.0001	0.2567	91.3196	99.2593
ResNet-50 with SGD	32	0.01	0.3599	87.6350	99.9191
ResNet-50 with ADAM + SGD	32	ADAM:1e−4, SGD:1e−3	0.3913	86.4732	98.8608

In Table 2, the model trained based on ResNet-50 backbone with ADAM optimization algorithm has less error and higher accuracy (acc) than that trained based on ResNet-34 backbone with ADAM, SGD and a fusion of ADAM and SGD optimization algorithm. The model trained based on ResNet-50 with ADAM has a smaller error and higher accuracy of top-1 than that trained by other optimization methods. The model trained based on ResNet-50 with SGD has higher accuracy of top-5 than that trained by other optimization methods, so we only apply the ADAM as the optimization algorithm based on the ResNeXt-50 backbone. The model trained based on ResNeXt-50 with ADAM optimization algorithm has smaller error and higher accuracy of top-1 and top-5 than the model trained based on ResNet-50 with ADAM optimization algorithm. In short, the model trained based on ResNeXt-50 backbone with ADAM optimization algorithm has the smallest error and the highest accuracy.

The results of Scene images which are predicted by the model trained on the Scenes21 dataset with ResNeXt-50 backbone is shown in Fig. 8. The three images on the left all belong to the category "OSSs", and their ground-truth labels are skyscraper, forest road, and alley, respectively. The three images in the middle belong to the "TSs" category, and their ground-truth labels are all transition/exterior. The three images on the right belong to the "others" category, and their ground-truth labels are all others, which contain indoor scenes and outdoor open scenes. Labels and the top-5 predictions are shown, and the number in each bracket represents the prediction confidence. We can see that most of the top five responses are very relevant to the scene description.



(a) Classification: OSSs (b) Classification: TSs (c) Classification: others

Fig. 8 Images from the validation set are predicted based on the Scenes21-ResNeXt

4.2 Novelty detection based on GNSS

The class "others" includes outdoor open scenes and indoor scenes, which can be distinguished according to GNSS in the evaluation stage, and some errors or ambiguities can also be corrected based on the GNSS, the specific method is introduced in Subsection 2.4. The number of satellites in different environments is shown in Fig. 9. The real-time signal of GNSS in different scenes is shown in Fig. 9. The abscissa represents time, and the ordinate represents the number of satellites. In the range of 0−153 s, GNSS is in the initialization stage (INIT), and in the range of 154−311 s, the number of satellites is stable above 14, which means that the signal is very good, it is in an outdoor open environment. In the range of 312−529 s, the number of signals fluctuates between 7−14, which is very unstable and is in the transition state from indoors to outdoors or outdoors to indoors. In the range of 530−570 s, the number of satellites is close to 0, which means it is in an indoor environment.

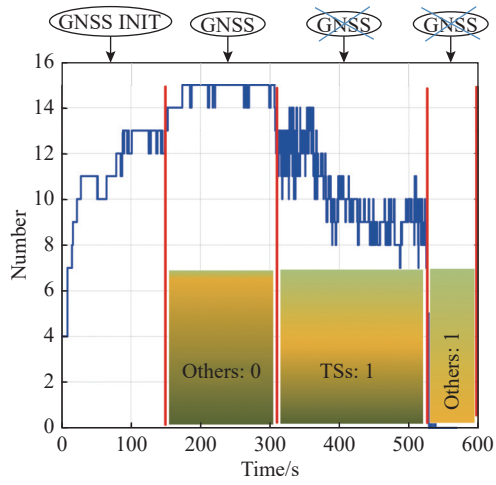


Fig. 9 Number of GNSS satellites in indoor scenes, outdoor scenes, and TSs

5. Conclusions and future work

This article does not propose a new backbone for scene recognition, but trains the model based on the existing ResNet and ResNeXt backbone and a transfer learning method. The Scenes21 dataset proposed in this paper is based on Places365 dataset, but has been cut into 21 categories according to the flight scenes of the drone, and the scene images collected by our drone are also added. Although the dataset contains a novelty detection class, the drone may still encounter scene problems that do not belong to the Scenes21 dataset in real-time flight, that is, novelty detection problems. In this paper, we combine the scene recognition results with the simulated GNSS signal in real-time scene to solve the novelty problem, and pave the way for real-time scene recognition in the future. Due to the influence of computing power, we train the model based on ResNet-18, ResNet-34 and ResNet-50 backbones with ADAM, SGD and fusion of ADAM and SGD optimization algorithm for respectively, and only train the model based on ResNeXt-50 backbone with ADAM. It is found that based on the Scenes21 dataset, the model trained with ResNeXt-50 backbone and ADAM optimization algorithm has the smallest error and the highest accuracy. In the future work, we will continue to improve the dataset according to the common scenes of the drone. In addition, we will perform real-time scene recognition in flight by loading the model on the computing platform of the drone and be able to realize autonomous navigation in various scenes.

References

[1] LI B Q, HU X H. Effective distributed convolutional neural network architecture for remote sensing images target classification with a pre-training approach. Journal of Systems Engineering and Electronics, 2019, 30(2): 238–244.

[2] LU C W, TSOUGENIS E, TANG C K. Improving object recognition with the l-channel. Pattern Recognition, 2016, 49: 187–197.

[3] GIRSHICK R. Fast R-CNN. Proc. of the IEEE International Conference on Computer Vision, 2015: 1440−1448.

[4] REN S Q, HE K M, GIRSHICK R, et al. Faster R-CNN: towards real-time object detection with region proposal networks. IEEE Trans. on Pattern Analysis and Machine Intelligence, 2017, 39(6): 1137–1149.

[5] REDMON J, FARHADI A. Yolov3: an incremental improvement. https://doi.org/10.48550/arXiv.1804.02767.

[6] REDMON J, FARHADI A. Yolo9000: better, faster, stronger. Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, 2017: 6517−6525.

[7] MA Y L, WANG C Y. Sdcnet for object recognition. Computer Vision and Image Understanding, 2022, 215: 103332.

[8] WU X W, SAHOO D, HOI S. Recent advances in deep learning for object detection. Neurocomputing, 2020, 396: 39–64.

[9] ZOU Z X, CHEN K, SHI Z W, et al. Object detection in 20 years: a survey. Proceedings of the IEEE, 2023, 111(3): 257−276.

[10] LI Y L, WANG S J, TIAN Q, et al. Feature representation for statistical-learning-based object detection: a review. Pattern Recognition, 2015, 48(11): 3542–3559.

[11] ZHANG X G, MA D X, YU H, et al. Scene perception guided crowd anomaly detection. Neurocomputing, 2020, 414: 291–302.

[12] YANG J F, ZHANG S S, WANG G H, et al. Scene and place recognition using a hierarchical latent topic model. Neurocomputing, 2015, 148: 578–586.

[13] ESLAMI S M A, REZENDE D, BESSE F, et al. Neural scene representation and rendering. Science, 2018, 360(6394): 1204–1210.

[14] SIMONYAN K, ZISSERMAN A. Very deep convolutional networks for large-scale image recognition. https://doi.org/10.48550/arXiv.1409.1556.

[15] XIE L, LEE F F, LIU L, et al. Scene recognition: a comprehensive survey. Pattern Recognition, 2020, 102: 107205.

[16] ZENG D L, LIAO M Y, TAVAKOLIAN M, et al. Deep learning for scene classification: a survey. https://arXiv.org/abs/2101.10531.

[17] ZHANG X W, WANG L, SU Y. Visual place recognition: a survey from deep learning perspective. Pattern Recognition, 2021, 113: 107760.

[18] ZHOU B L, LAPEDRIZA A, KHOSLA, et, al. Places: a 10 million image database for scene recognition. IEEE Trans. on Pattern Analysis and Machine Intelligence, 2018, 40(6): 1452–1464.

[19] NEYSHABUR B, SEDGHI H, ZHANG C. What is being transferred in transfer learning? Proc. of the 34th Conference on Neural Information Processing Systems, 2020: 512−523.

[20] HE K M, ZHANG X Y, REN S Q, et, al. Deep residual learning for image recognition. Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, 2016: 770−778.

[21] XIE S N, GIRSHICK R, DOLLAR P, et al. Aggregated residual transformations for deep neural networks. Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, 2017: 5987−5995.

[22] STEGER C, ULRICH M, VIEDEMANN C. Machine vision algorithms and applications. New Jersey: Wiley-VCH GmbH, 2018.

[23] THEODORIDIS S, KOUTROUMBAS K. Pattern recognition. 4th ed. Boston: Academic Press, 2009.

[24] XIE P, PETOVELLO M G. Measuring GNSS multipath distributions in urban canyon environments. IEEE Trans. on Instrumentation and Measurement, 2015, 64(2): 366–377.

[25] BRIDLE J S. Training stochastic model recognition algorithms as networks can lead to maximum mutual information estimation of parameters. Proc. of the Advances in Neural Information Processing Systems, 1990: 211−217.

[26] QIAN N. On the momentum term in gradient descent learning algorithms. Neural Networks, 1999, 12(1): 145–151.

[27] SUTSKEVER I, MARTENS J, DAHL G, et al. On the importance of initialization and momentum in deep learning. Proc. of the 30th International Conference on Machine Learning, 2013: 1139−1147.

[28] KINGMA D, BA J. Adam: a method for stochastic optimization. Proc. of the 3th International Conference on Learning Representations, 2015. https://doi.org/10.48550/arXiv.1412.6980.

Biographies

DU Hao was born in 1987. He received his B.E. and M.S. degrees in electronic information engineering and system analysis and integration from Nanjing University of Information Science and Technology, Nanjing, China, in 2009 and 2012, respectively. He is currently pursuing his Ph.D. degree in the School of Automation, Southeast University, Nanjing, China. His current research interests include multi-sensor fusion navigation for the drone, computer vision and visual simultaneous localization and mapping.
E-mail: du-hao@seu.edu.cn

WANG Wei was born in 1972. He received his B.E., M.E., and Ph.D. degrees from Chiba University, Chiba, Japan, in 2004, 2006, and 2009, respectively. He is currently a professor at the School of Automation, Nanjing University of Information Science and Technology, Nanjing, China. His current research interests include nonlinear system control, intelligent control of the drone.
E-mail: wwcb@nuist.edu.cn

WANG Xuerao was born in 1996. She received her B.S. degree in engineering from Qingdao University of Technology, Qingdao, China, in 2016, and M.S. degree in engineering from University of Science and Technology Beijing, Beijing, China, in 2019. She is currently pursuing her Ph.D. degree in control science and engineering with the School of Automation, Southeast University, Nanjing, China. Her research interests include intelligent control, nonlinear system control, and reinforcement learning.
E-mail: wangxuerao@seu.edu.cn

ZUO Jingqiu was born in 1995. She received her B.S. degree in engineering from Jilin University, Changchun, China, in 2017, and M.S. degree in engineering from Osaka University, Suita, Japan, in 2020. She is currently an engineer in the Jiangsu Zhongke Institute of Applied Research on Intelligent Science & Technology, China. Her research interests include mobile robot and autonomous navigation.
E-mail: zuojingqiu@arist.ac.cn

WANG Yuanda was born in 1993. He received his B.S. degree in automation from Nanjing University of Information Science and Technology, Nanjing, China in 2014, and Ph.D. degree in control science and engineering from Southeast University, Nanjing, China in 2020. Currently, he is working as a postdoctoral researcher with the School of Automation, Southeast University, Nanjing, China. He has been a visiting Ph.D. student with the Department of Electrical, Computer, and Biomedical Engineering, University of Rhode Island, Kingston, USA from 2016 to 2018. His current research interests include deep reinforcement learning, neural networks, and multi-agent systems.
E-mail: wangyd@seu.edu.cn