ISSN 2616-5775 Vol. 7, Issue 10: 1-7, DOI: 10.25236/AJCIS.2024.071001

Multi-environment adaptive positioning method based on UAV and satellite images

Ling Wei[1,a,*], Hao Liang[1,b], Juncai Wang[1,c]

[1]*School of Information Science and Engineering, Chongqing Jiaotong University, Chongqing, 400074, China*
[a]*2689874779@qq.com,* [b]*1422268226@qq.com,* [c]*1226430832@qq.com*
[*]*Corresponding author*

Abstract: *Drone image geolocation aims to estimate the geographic location of drone-captured images. Given a query image with an unknown location, the task involves retrieving the most similar reference image from a database and using its GPS information to estimate the location of the query image. This is fundamentally an image retrieval problem, where deep neural networks are employed to learn effective image descriptors. However, current research primarily focuses on closing the gap between drone and satellite views, often leading to performance drops under real-world conditions such as rain and fog. This issue primarily arises because the dataset used for training the model does not fully capture the complex environments encountered in real-world applications, leading to a domain gap between training and testing. To address this challenge, we propose a dual-branch multi-environment adaptation network (MuSe-Net) designed to dynamically adjust and adapt to environmental changes. The network consists of two branches: the multi-environment style extraction network, which captures weather-related style information, and the adaptive feature extraction network, which uses an adaptive modulation module to minimize the style differences caused by environmental conditions. Extensive experiments on the University-1652 benchmark show that MuSe-Net delivers strong performance in geolocation across various environmental conditions.*

Keywords: *Deep Learning, Image Retrieval, Multisource Domain Generalization, Geo-Localization*

1. Introduction

Drone image geolocation determines the location of the drone by retrieving images between drone view images and satellite view images. This technology is widely used in drone navigation, drone positioning, event detection, aerial photography and other fields [1,2]. In specific operations, the system searches for the most relevant images for a given drone image from a library of satellite images with geotags (GPS), thereby helping the drone determine its geographic location. Although drone images have good visibility, the offset across view domains makes this task challenging due to differences in perspective and environment.

In existing UAV image geolocation, CNNs are mainly used to learn the geographic features of images. Dual-branch networks based on convolutional neural networks are widely used in related research [3], in which metric learning [4] and classification loss [5] are the two main choices for optimizing models. Of course, the image geolocation task can also be completed by adjusting the spatial layout of image semantics or aligning local information [6,7]. All of the above existing methods focus on alleviating the cross-view domain gap introduced by viewpoint changes. How should the model cope with the complex and changeable real-world weather environment? Usually a well-trained network often performs poorly when facing unfamiliar inputs, and drones are also likely to encounter bad weather environments that they did not encounter before takeoff after takeoff. Therefore, UAV image geolocation based on multi-environment adaptive networks is a meaningful and practical study. In the study of domain generalization, one common approach is to train the model on a large, diverse dataset, allowing it to learn the location distribution across various environments. However, previous studies [8,9] have demonstrated that it is quite difficult to make a model generalize effectively to all possible domains. In our approach, we aim to enable the model to selectively adjust to domain shifts caused by environmental changes. Specifically, we first use the MobileNetV3 network branch to extract environment-related features from the image. These features are then input into the IBN-Net network, which contains an adaptive modulation module. This module dynamically adjusts according to the style changes introduced by different environmental

ISSN 2616-5775 Vol. 7, Issue 10: 1-7, DOI: 10.25236/AJCIS.2024.071001

conditions, thereby effectively reducing the impact of these environmental changes. For the simulation of realistic weather environments, we selected a ready-made image-based style transfer library to preprocess the image. Nine synthetic environment images of a geographic location are obtained, namely fog, rain, snow, fog and rain, fog and snow, rain and snow, darkness, overexposure, and wind. Multi-environment style extraction branch We use a supervised learning method to use a lightweight MobileNetV3 network to extract the weather style features of the image. The adaptive feature extraction branch processes the image through the IBN-Net network to obtain visual features [10]. The IBN-Net network integrates batch normalization (BN) and instance normalization (IN) into the residual block. BN is used to retain the differences between different categories [11], and IN is used to eliminate domain-specific statistical features to make the features of the same category in different domains more consistent. However, IN uses the same processing method for different domain features when eliminating domain-specific statistical features. In order to meet dynamic adaptability, we further introduced an adaptive feature modulation block (AFM) and integrated the AFM module into the IBN-Net network. The AFM module is inserted after the instance normalization module, which dynamically modulates the instance normalization by learning the scale and deviation from the outside, so that the IBN-Net network can adaptively extract the geographical features of the image. Extensive experiments on the widely used University-1652 benchmark show that the proposed MuSe-Net achieves competitive results in geolocation in multiple environments.

2. Basic Theory

2.1. MobileNetV3 Model Principle

MobileNetV3, introduced by Google in 2019, is a lightweight attention-based model. It integrates several key elements: the depthwise separable convolutions from MobileNetV1, the inverted residuals with linear bottlenecks from MobileNetV2, and the squeeze-and-excitation mechanism from MnasNet [12]. It further optimizes the network structure. This design enables MobileNetV3 to provide efficient computing performance and excellent accuracy in resource-constrained environments, and is particularly suitable for deep learning applications in mobile devices and embedded systems.

Depthwise separable convolution divides the convolution kernel into single-channel filters and applies convolution operations to each channel independently. This process preserves the depth of the input feature layer while producing an output feature layer with the same number of channels as the input feature map, as illustrated in Figure 1. For example, a 20 × 20 × 3 feature layer is input and convolved with a 7 × 7 × 1 × 3 convolution kernel to obtain a 14 × 14 × 3 feature layer. The depth of both the input and output is 3.

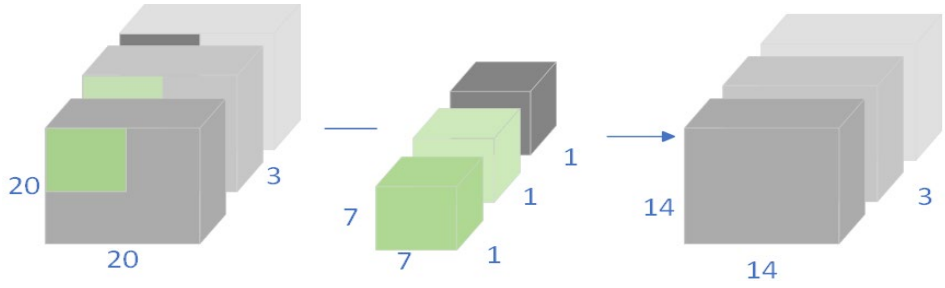

Figure 1: Depthwise convolution process in depthwise separable.

Squeeze-and-Excitation Networks is an architectural module. It mainly enhances the network's expressiveness by adaptively reweighting the channel features of each convolutional layer. As shown in Figure 2, U is a feature layer with dimensions C × H × W, and is also the input of the attention module. The implementation details of the attention module include two parts: Squeeze and Excitation. In the Squeeze part, all the eigenvalues of each channel of U are averaged, see formula (1), and a eigenvector with dimension 1 × 1 × C is obtained, which corresponds to the module $F_{sq}(\cdot)$ in the figure.

$$z_C = F_{sq}(u_c) = \frac{1}{W \times H} \sum_{i=1}^{W} \sum_{j=1}^{H} u_c(i, j) \tag{1}$$

Where: z_c refers to the eigenvalue of the cth channel in the feature vector; $u_c(i, j)$ represents the eigenvalue of the i-th row and j-th column of the feature layer; W and H are the width and height of the

ISSN 2616-5775 Vol. 7, Issue 10: 1-7, DOI: 10.25236/AJCIS.2024.071001

feature layer respectively.

In the Excitation part, a feature vector of dimension 1 × 1 × C is connected to two fully connected layers (F_c Layers), aiming to use the correlation between channels to train a weight vector that conforms to the channel enhancement and attenuation rules. The Sigmoid function then maps the value of each dimension to the range of (0, 1) and multiplies it with the original input feature layer U to implement the attention mechanism of each channel of the feature layer.

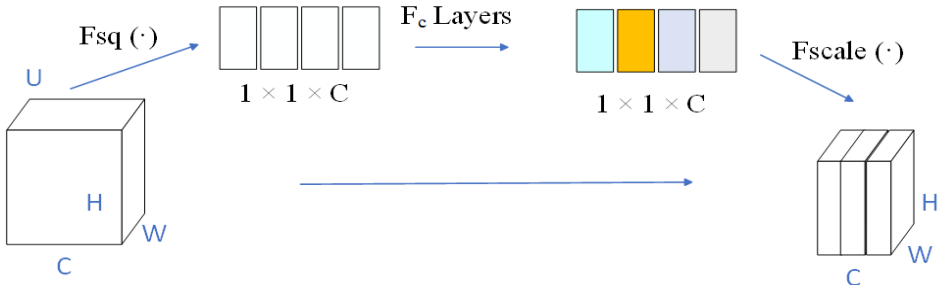

Figure 2: Squeeze-and-Excitation Networks.

2.2. IBN-Net Network

IBN-Net (Instance-Batch Normalization Network) is an improved convolutional neural network architecture that aims to improve the generalization ability of deep learning models in visual tasks, especially in cross-domain (domain generalization) tasks. Its core idea is to use IN and BN simultaneously in the convolutional layer to achieve dual normalization of local texture and global features. This method enhances the adaptability of the network in different fields by introducing two types of normalization processing.

IN is mainly used to eliminate the style information of the image and retain the content information. It normalizes each channel of each sample independently. Its formula (2) is as follows:

$$IN(\mathrm{x}) = \frac{x - \mu_{ins\,\tan ce}}{\sigma_{ins\,\tan ce}} \tag{2}$$

Where μinstance and σinstance are the mean and standard deviation of the channel of the sample respectively.

BN normalizes a batch of samples during training to reduce the impact of internal covariate shift, making model training more stable and accelerating convergence. Its formula (3) is as follows:

$$BN(\mathrm{x}) = \frac{x - \mu_{batch}}{\sigma_{batch}} \tag{3}$$

Where μ_{batch} and σ_{batch} are the mean and standard deviation of the channel in the current batch of samples.

3. MuSe-Net Model

Figure 3 shows the Multi-Environment Adaptive Network (MuSe-Net). It consists of two branches, Estyle and Econtent, with the same input. Estyle aims to capture style information related to different weather conditions, while upsampling and concatenating the captured features, and then feeding the concatenated features into the Adaptive Feature Modulation Module (AFM) for convolution. Finally, Econtent uses the affine parameters obtained from the Adaptive Feature Modulation Module (AFM) block to dynamically adjust the instance normalization layer in Econtent.

ISSN 2616-5775 Vol. 7, Issue 10: 1-7, DOI: 10.25236/AJCIS.2024.071001

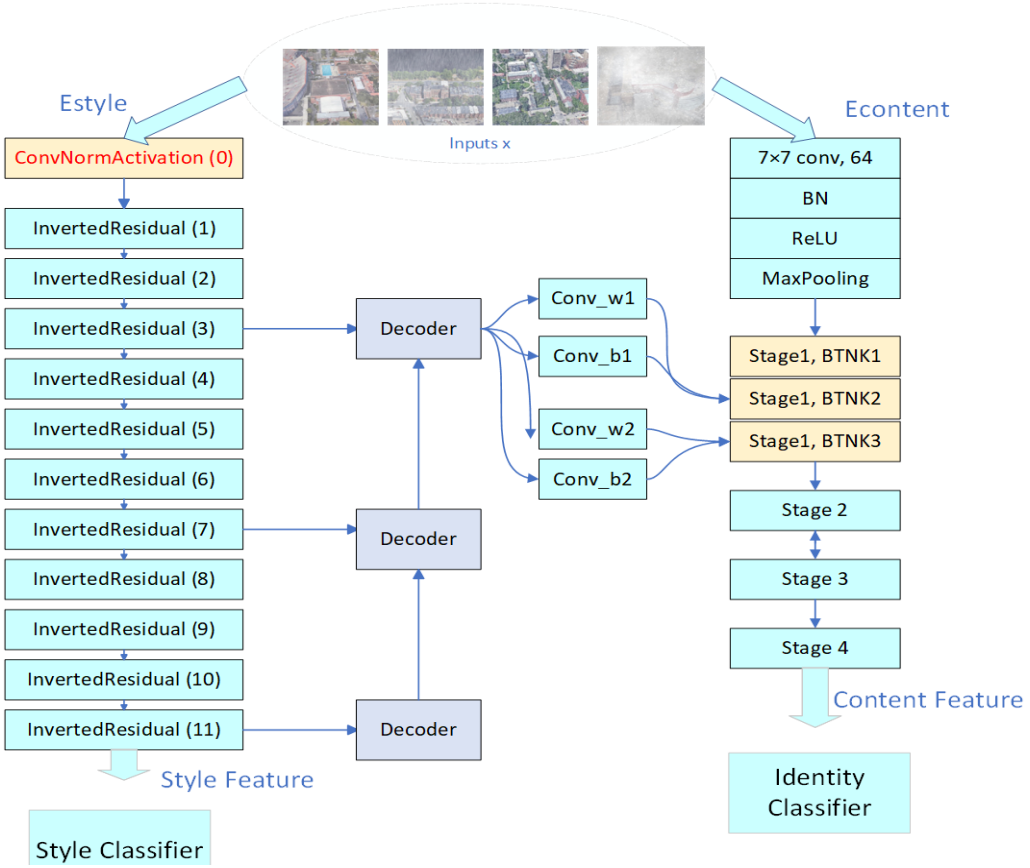

Figure 3: The Multi-Environment Adaptive Network.

3.1. Weather Environment Style Extraction Branch Based on MobileNetV3

This branch comprises two components: (1) the encoder, which utilizes the ConvNormActivation module from MobileNetV3 along with 11 inverted residual blocks to form a weather feature extraction network. The ConvNormActivation module, consisting of convolutional layers, batch normalization (BN) layers, and Hardswish activation, increases the number of channels while reducing image size to preserve key features. The inverted residual blocks incorporate depthwise separable convolutions and squeeze-and-excitation modules to reduce computational costs and enhance feature extraction. As shown in Figure 4, to capture weather-related features mainly found in shallow layers, extraction begins from the third, seventh, and eleventh layers, with upsampled features concatenated. (2) The style classifier uses BN layers, dropout layers, and fully connected layers (FC) to classify the weather environment of the image.

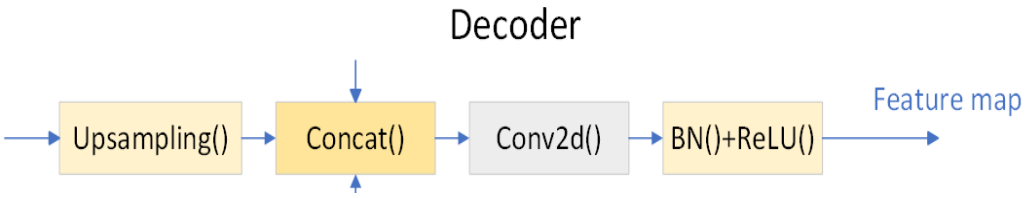

Figure 4: Upsampling operation.

3.2. Adaptive Feature Extraction Branch Embedded in AFM

This branch consists of an identity classifier and a content encoder embedded with an adaptive feature modulation module. The content encoder is built on the IBN-Net backbone, which has a structure similar to ResNet-50. Both IBN-Net and ResNet-50 consist of four stages, each containing a different number of bottleneck layers. In IBN-Net, the first stage contains 3 bottleneck layers. We integrate the adaptive feature modulation module into the second and third bottlenecks of stage 1 (see Figure 5). The identity classifier shares the same components as the style classifier.

ISSN 2616-5775 Vol. 7, Issue 10: 1-7, DOI: 10.25236/AJCIS.2024.071001

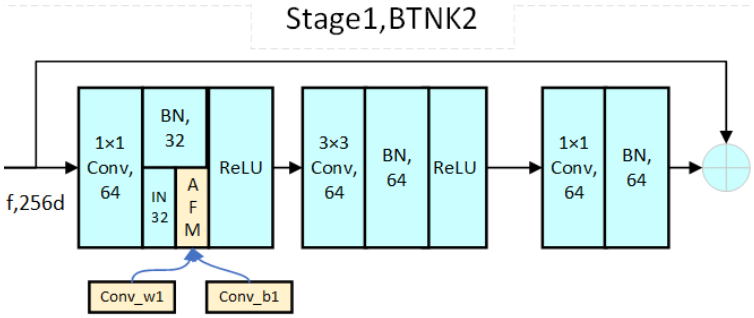

Figure 5: Bottleneck layer of content encoding (embedded AFM).

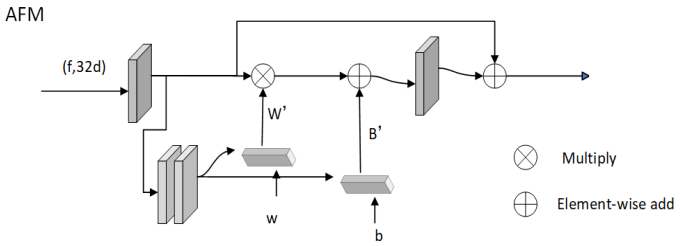

Figure 6: Adaptive feature modulation block.

3.3. Adaptive Feature Modulation Block

The adaptive feature modulation block (AFM) first passes through two convolutional layers, namely Conv_w1 and Conv_b1, for learning the scale and bias (see Figure 5). Then we input the global features of the image into the two convolutional layers to predict a set of adaptive 3×3 convolution kernels to dynamically adjust and optimize the scale and bias through the global information of the image (see Figure 6). Through this adjustment, the scale and bias after convolution will better reflect or adapt to the global weather conditions. Finally, feature modulation is performed through the scale and bias after convolution, as shown in formula (4).

$$AFM(u,v) = \sigma(v)\cdot IN(u) + \mu(v) + IN(u) = IN(u)\cdot(1+\sigma(v)) + \mu(v) \quad (4)$$

Let u denote the input feature and v represent the corresponding style feature. μ(u) and σ(u) are the mean and variance of the input features. σ(v) and μ(v) are the learning scale and bias for adjusting the normalized feature u. IN(·) is the instance normalization operation.

4. Experimental Analysis

4.1. Data Processing

We use the University-1652 dataset to train and evaluate the proposed method. University-1652 is a multi-scene, cross-view image retrieval dataset designed for studying image retrieval based on multi-view and multi-scene data. Developed by the University of Science and Technology of China, it is widely used for tasks like cross-view image matching and retrieval, particularly between drone and ground views. The dataset includes images from drones, satellites, and mobile phones, covering 1,652 buildings across 72 universities globally, with non-overlapping training and test sets. It supports applications like drone-based geolocation and guidance.

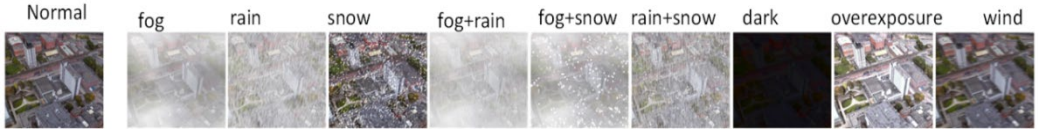

Figure 7: Example of a synthetic environment on university-1652.

When performing drone view target positioning and drone guidance in multiple weather environments, we need drone view images to reproduce multiple weather environments in real life. Here we choose a ready-made image-based style transfer library to preprocess the images. After preprocessing,

ISSN 2616-5775 Vol. 7, Issue 10: 1-7, DOI: 10.25236/AJCIS.2024.071001

nine synthetic weather environment images of a geographical location are obtained, namely fog, rain, snow, fog and rain, fog and snow, rain and snow, darkness, overexposure, and wind, as shown in Figure 7.

4.2. Evaluation Indicators and Environment Cnfiguration

In the experiment, we use recall (Recall@K) and average precision (AP) to evaluate model performance. Recall@K measures the proportion of correctly matched images in the top K. The higher the value, the better the network performance. AP refers to the area under the precision-recall curve. These two indicators are used for performance evaluation in drone positioning and navigation tasks, respectively.

During training, we use stochastic gradient descent (SGD) with a momentum of 0.9 and a weight decay of 0.0005 to optimize the model, with a batch size of 16. Each branch uses cross-entropy as the loss function, while style loss and identity loss are combined to train MuSe-Net. The initial learning rate is set to 0.0005 and decays by 0.1 and 0.01 at 120 and 180 epochs, respectively, over a total of 210 epochs. During testing, the Euclidean distance is used to measure similarity between the query image and the gallery candidates.

We implemented our code based on Pytorch and used the NVIDIA GeForce RTX 3080 Ti graphics card for experiments.

4.3. Experimental Results and Analysis

We conducted two tasks on the university-1652 dataset: drone-view target localization (drone→satellite) and drone navigation (satellite→drone). At the same time, we re-implemented four methods as comparative experiments. The content encoders in the four compared methods are VGG16, DenseNet121, Swin-T and IBN-Net50-a.

In our experiment, we maintained a consistent weather environment style for the satellite images in the University-1652 dataset, while converting drone images into 10 different weather conditions. As detailed in Table 1, among the four re-implemented methods for the drone → satellite task, IBN-Net shows a notable improvement over the others. However, our method outperforms IBN-Net across all weather conditions. Specifically, our approach enhances the R@1 accuracy from 62.30% to 64.83% (+2.54%) and the AP accuracy from 66.46% to 68.89% (+2.43%).

Table 1: Comparison with other advanced methods.

Method	**initial**	**fog**	**rain**	**snow**	**fog+ rain**	**fog+ snow**	**rain+ snow**	**dark**	**over- exposure**	**wind**	**Ave**
	R@1 AP	R@1 AP	R@1 AP	R@1 AP	R@1 AP	R@1 AP	R@1 AP	R@1 AP	R@1 AP	R@1 AP	R@1 AP
					Drone	→	Satellite				
VGG16	59.98 64.69	56.21 61.11	53.97 58.90	50.07 55.08	50.43 55.63	42.77 48.01	51.08 56.10	39.10 44.30	45.16 50.47	50.84 56.05	49.96 55.03
DenseNe-t121	69.48 73.26	64.25 68.47	63.47 67.64	59.29 63.70	59.68 64.13	50.41 55.20	60.21 64.57	48.57 53.41	54.04 58.88	60.74 65.14	59.01 63.44
Swin-T	69.27 73.18	66.46 70.52	65.44 69.60	61.79 66.23	63.96 68.21	**56.44** **61.07**	62.68 67.02	50.27 55.18	55.46 60.29	63.81 68.17	61.56 65.95
IBN-Net	72.35 75.85	66.68 70.64	67.95 71.73	62.77 66.85	62.64 66.84	51.09 55.79	64.07 68.13	50.72 55.53	57.97 62.52	66.73 70.68	62.30 66.46
Ours	**74.68** **78.02**	**69.47** **73.24**	**70.55** **74.14**	**64.78** **68.93**	**65.59** **69.64**	53.55 58.24	**66.19** **70.21**	**54.05** **58.60**	**61.05** **65.51**	**68.46** **72.37**	**64.83** **68.89**
					Satellite	→	Drone				
VGG16	75.89 58.50	75.18 55.42	71.61 53.03	68.19 48.29	71.18 49.34	65.48 40.87	69.47 50.03	64.34 35.74	64.91 44.20	68.90 49.53	69.52 48.50
DenseNe-t121	83.74 70.34	82.31 66.32	81.17 65.23	78.60 60.33	79.46 61.66	74.61 51.14	78.46 61.68	74.47 47.88	74.32 55.26	78.32 61.63	78.55 60.15
Swin-T	80.74 68.94	81.03 67.46	81.17 66.39	78.46 61.33	79.17 64.65	74.89 56.57	78.89 63.49	75.61 48.43	76.60 56.57	78.74 64.45	78.53 61.83
IBN-Net	86.31 73.54	84.59 **67.61**	84.74 **69.03**	80.88 64.44	83.31 63.71	77.89 52.14	83.02 65.74	78.46 50.77	79.46 58.64	84.02 67.94	82.27 63.36
Ours	**86.88** **74.47**	**84.74** 67.47	**85.02** 67.78	**84.45** **67.14**	**84.02** **65.65**	**81.17** **54.09**	**84.88** **67.75**	**80.74** **53.01**	**81.60** **62.09**	**84.17** **69.25**	**83.77** **64.87**

The satellite → drone task, in contrast, is less complex than drone → satellite. Table 1 illustrates that the recall rate and average precision for satellite → drone are generally higher than for drone → satellite. Despite this, our method still maintains a superior performance across 10 different weather environments.

ISSN 2616-5775 Vol. 7, Issue 10: 1-7, DOI: 10.25236/AJCIS.2024.071001

We achieved an increase in R@1 accuracy from 82.27% to 83.77% (+1.50%) and an improvement in AP from 63.36% to 64.87% (+1.51%).

Through the above two experiments, we can draw two conclusions. First, compared with other methods, IBN-Net can obtain better results when facing multi-domain data. It can filter out domain shifts caused by different weather environment styles through instance normalization (IN). Secondly, the MuSe-Net network proposed by us based on IBN-Net can further improve performance by dynamically adjusting instance normalization through an adaptive modulation module (AFM).

5. Conclusions

In this paper, we address the challenges of UAV-view image localization and UAV guidance, focusing on the cross-view gap between different viewpoints and the domain gap caused by varying weather and illumination. While previous studies primarily tackled the cross-view gap, we propose MuSe-Net, an end-to-end learning network designed to address the domain gap across different environmental conditions. Additionally, we introduce an adaptive feature modulation module, which dynamically balances environmental domain shifts within the adaptive feature extraction network. To validate MuSe-Net's performance, we conducted experiments using the University-1652 dataset and achieved competitive results. Looking forward, we plan to enhance geolocation performance in two areas: first, by improving the multi-environment extraction network to capture more expressive weather-related features, and second, by increasing the image geolocation speed to enhance its practical value.

References

[1] Ding, L., Zhou, J., Meng, L., & Long, Z. (2020). A practical cross-view image matching method between UAV and satellite for UAV-based geo-localization. Remote Sensing, 13(1), 47.

[2] Zeng, Z., Wang, Z., Yang, F., & Satoh, S. I. (2022). Geo-localization via ground-to-satellite cross-view image retrieval. IEEE Transactions on Multimedia, 25, 2176-2188.

[3] Liu, L., & Li, H. (2019). Lending orientation to neural networks for cross-view geo-localization. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 5624-5633).

[4] Hu, S., Feng, M., Nguyen, R. M., & Lee, G. H. (2018). Cvm-net: Cross-view matching network for image-based ground-to-aerial geo-localization. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 7258-7267).

[5] Wang, T., Zheng, Z., Yan, C., Zhang, J., Sun, Y., Zheng, B., & Yang, Y. (2021). Each part matters: Local patterns facilitate cross-view geo-localization. IEEE Transactions on Circuits and Systems for Video Technology, 32(2), 867-879.

[6] Fu, Y., Wang, X., Wei, Y., & Huang, T. (2019, July). Sta: Spatial-temporal attention for large-scale video-based person re-identification. In Proceedings of the AAAI conference on artificial intelligence (Vol. 33, No. 01, pp. 8287-8294).

[7] Sun, Y., Zheng, L., Yang, Y., Tian, Q., & Wang, S. (2018). Beyond part models: Person retrieval with refined part pooling (and a strong convolutional baseline). In Proceedings of the European conference on computer vision (ECCV) (pp. 480-496).

[8] Chattopadhyay, P., Balaji, Y., & Hoffman, J. (2020). Learning to balance specificity and invariance for in and out of domain generalization. In Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part IX 16 (pp. 301-318). Springer International Publishing.

[9] Ilse, M., Tomczak, J. M., Louizos, C., & Welling, M. (2020, September). Diva: Domain invariant variational autoencoders. In Medical Imaging with Deep Learning (pp. 322-348). PMLR.

[10] Pan, X., Luo, P., Shi, J., & Tang, X. (2018). Two at once: Enhancing learning and generalization capacities via ibn-net. In Proceedings of the European conference on computer vision (ECCV) (pp. 464-479).

[11] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 770-778).

[12] Hu, J., Shen, L., & Sun, G. (2018). Squeeze-and-excitation networks. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 7132-7141).