Content-Aware Hierarchical Representation Selection for Cross-View Geo-Localization

Zeng Lu[1,3], Tao Pu[2], Tianshui Chen[1] ⋆, and Liang Lin[2]

[1] Guangdong University of Technology, China
[2] Sun Yat-Sen University, China
[3] Guangzhou Quwan Network Technology Co., Ltd

Abstract. Cross-view geo-localization (CVGL) aims to retrieve the images that contain the same geographic target content and are from different views. However, the target content usually scatters over the whole image, and they are indiscernible from the background. Thus, it is difficult to learn feature representation that focuses on these contents, rendering CVGL a challenging and unsolved task. In this work, we design a Content-Aware Hierarchical Representation Selection (CA-HRS) module, which can be seamlessly integrated into current deep networks to facilitate CVGL. This module can help focus more on the target content while ignoring the background region, thus as to learn more discriminative feature representation. Specifically, this module learns hierarchical important factors to each location of the feature maps according to their importance and enhances the feature representation based on the learned factors. We conduct experiments on several large-scale datasets (i.e., University-1652, CVUSA and CVACT), and the experiment results show the proposed module can obtain obvious performance improvement over current competing algorithms. Codes are available at https://github.com/Allen-lz/CA-HRS.

Keywords: Geo Localization · Feature Selection · Image Retrieval.

1 Introduction

As a practical and challenging sub-task of image retrieval [27, 14, 30], cross-view geo-localization (CVGL) aims to find the target images in one view among large-scale candidates (gallery) that have the same contents with the input query image in another view. Formally, there are three views of images, i.e., satellite-view, drone-view, and ground-view images. It contains three types of tasks according to different views of the input and target images: **Drone** → **Satellite** with the input image of drone-view and the target images of satellite-view; **Satellite** →

⋆ Corresponding author is Tianshui Chen. This work is supported in part by National Natural Science Foundation of China (NSFC) under Grant No. 62206060 and in part by Science and Technology Project of Guangdong Province under Grant No. 2021A1515011341.

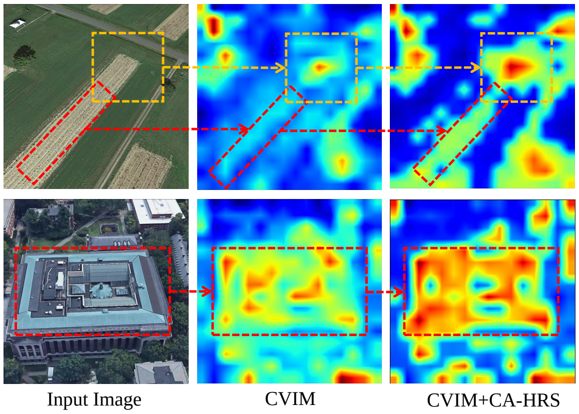

Fig. 1. Two examples of the input image (left), the learned feature maps by the baseline CVIM (middle), and the learned feature maps by integrating the CA-HRS module (right).

Drone with the input image of satellite-view and the target images of drone-view; **Ground** → **Satellite** with the input image of ground-view and the target images of satellite-view.

Recently, CVGL receives increasing attention as it benefits variant applications such as agriculture, aerial photography, event detection, and accurate delivery [27, 10, 32, 18]. Current works for this task combine metric learning [15, 2, 14] or domain adaptation [12, 23] with deep neural networks to learn view-invariant feature representation. More recent works further introduce manually annotated orientation information to regularize training to improve CVGL performance. However, this works either increase the model complexity and inference time or incur additional annotation overhead, making them impractical and unscalable. On the other hand, the target contents usually scatter over the whole image. Current algorithms roughly find the content regions but can not well highlight these regions to learn more discriminative feature representation. As shown in Figure 1, the learned feature representation is slightly obvious but can not be distinguished from the surrounding background regions.

To address these issues, we design a novel yet effective Content-Aware Hierarchical Representation Selection (CA-HRS) module that helps to better focus on the target content meanwhile suppress the background regions. We experimentally find that it has higher activation values on the content regions and slightly lower activation values on the background regions. Thus, it is expected to set the regions with higher activation values the higher value and set the regions with lower activation values with lower values, and thus make the contents distinguished from the background regions. To achieve this end, the CA-HRS module computes an average representation as a threshold. Then, it sets the locations with the activation values higher than this threshold as positive while those values lower than this threshold as positive to obtain an enhancement coefficient map. Moreover, the content regions usually have different scales for different images, and we propose to compute multi-scale enhancement coefficient maps, and combine them to obtain the hierarchical enhancement coefficient maps. Finally, we design an adaptive residual fusion mechanism to seamlessly and flexibly integrate the CA-HRS into current CVGL algorithms for feature enhancement to facilitate the performance of CVGL. As shown in Figure 1, by integrating the CA-HRS module into the current cross-view image matching (CVIM) algorithm [30], it can learn feature maps that obviously focus on the content regions while ignoring most of the background regions. Moreover, the CA-HRS incur no additional parameters and very limited computational overhead (i.e., about 1.0%), and thus it can be integrated into variant CVGL algorithms to boost their performance.

The contributions of this work can be summarized in the following. First, we design a novel yet effective content-aware hierarchical feature selection (CA-HRS) module that can help feature enhancement by focusing more on content regions while ignoring the background regions. Second, we introduce an adaptive residual fusion mechanism that can integrate the CA-HRS into current algorithms flexibly and seamlessly. Finally, we conduct extensive experiments on the large-scale University-1652, CVUSA, and CVACT datasets, and the experiment results show that the proposed module can obviously improve the performance of current state-of-the-art algorithms.

2 Relate Work

With the advancement of deep learning in images [17, 16, 4], cross-view geo-localization based on deep learning has achieved significant development. Siamese network [6] and metric learning are often used in image retrieval. The contrastive loss can reduce the distance between two matched positive samples and increase the distance between two unmatched negative samples [15]. The triple loss can simultaneously reduce and increase the distance between positive and negative samples [14, 2]. There is still a lot of works that use metric learning to train deep neural networks to learn discriminative features [8, 7, 11, 22].

In order to reduce the distance between two different domains, the most direct method is to transform the image features in one domain to another domain, namely cross-domain adaptation task [28, 5]. Lin et al. introduce a cross-view feature translation approach to greatly extend the reach of image geo-localization methods [12]. Shi et al. applied a regular polar transform to warp a satellite image such that its domain is closer to that of a ground-view panorama [21]. Shi et al. proposed a novel Cross-View Feature Transport (CVFT) technique to explicitly establish cross-view domain transfer that facilitates feature alignment between ground and satellite images [23].

The orientation information is also integrated into the neural network for learning. Liu et al. integrated the orientation information of each pixel into the convolution neural network for cross-view geo-localization, which improved the geo-localization accuracy [14]. Vo et al. proposed a new loss function, which combined rotation invariance and orientation regression in the training process, so that the network learned orientation and got a better feature representation [24]. Rodrigues et al. proposed a semantic-driven data enhancement technology that enables Siamese Network to discover objects that are difficult to capture [20]. Then, the enhanced samples are input to a multi-scale attention embedding network to perform the matching task. Zhu et al. [33] propose to estimate the orientation and align a pair of cross-view images with unknown alignment information.

In order to enable the network to focus on the feature extraction of images in different domains. CBMA [26] simply combines convolutional layers with sigmoid to extract key features. Zheng et al. applied Instance loss [31] to cross-view geo-localization [30]. Satellite images, ground images, and drone images were extracted by using corresponding backbone network to extract features. These features share the same classifier. They continued to use this network structure in their subsequent work LPN [25], LPN used a fixed division method to extract local features using context, so that the features were discriminative. Arandjelovic et al. proposed NetVLAD [1], which is a scene recognition method. It can extract local features and aggregate them to enhance the expressive ability of features. the method can also be applied to CVGL. Hu et al. introduced a CVM-Net [11], in which NetVLAD is embedded in Siamese network [6]. CVM-Net extracts the local features and then integrate them for image retrieval and geo-localization. Experiments show that the network with local features is better than that with only global features.

3 Hierarchical Enhancement Coefficient Map

In this section, we present the computing process of the hierarchical enhancement coefficient map (HECM) which helps to pay more attention to the important content regions while ignoring the unimportant background regions. In the context of the CVGL task, we observe the activation values of the content regions are usually slightly higher than those of the background regions. Thus, it is expected to increase the higher activation values even higher to emphasize the content regions and meanwhile to decrease the smaller activation values to even smaller to suppress the background. On the other hand, the different content regions usually share different scales. To achieve the above end, we propose to compute HECM that has higher important factors for regions with higher activation values and has smaller important factors otherwise.

Specifically, given the input feature maps of layer l, denoted as $\mathbf{f}^l \in \mathcal{R}^{W^l \times H^l \times C^l}$ in which W^l, H^l and C^l are the width, height, and channel number, we first compute an mean activation value for each location, formulated as

$$a^l = \frac{1}{C^l} \sum_{c=1}^{C^l} \mathbf{f}^{l,c}, \tag{1}$$

where $\mathbf{f}^{l,c}$ is the c-th feature map of $\mathbf{f}^l$. Then, average the activation values over all locations to obtain the mean representation, formulated as

$$thr_0^l = \frac{1}{W^l H^l} \sum_{w=1}^{W^l} \sum_{h=1}^{H^l} a_{wh}^l. \tag{2}$$

As discussed above, we consider the regions with activation values higher than the mean representation as important content region while those with activation values smaller than the mean representation as unimportant background regions. Intuitively, we can compute the ECM $\mathbf{m}_0^l \in \mathcal{R}^{W^l \times H^l}$, in which the value $\mathbf{m}_{0,wh}^l$ denote the important of location (w, h) and it can be computed by

$$\mathbf{m}_{0,wh}^l = \mathbf{1}(a_{wh}^l \geq thr_0^l). \tag{3}$$

In this way, we can obtain an ECM $\mathbf{m}_0^l$ to indicate the importance of each location. Considering different scales of content regions, we further introduce the average pooling with different kernel sizes that operates on the mean activation map a^l to obtain the thresholds and ECMs for different scales. For scale i, we first perform an average pooling with a kernel size of $k_i^l \times k_i^l$ on a^l to obtain an new mean activation map $a_i^l \in \mathcal{R}^{W_i^l \times H_i^l}$. Then, the threshold can be computed by

$$thr_i^l = \frac{1}{W_i^l H_i^l} \sum_{w=1}^{W_i^l} \sum_{h=1}^{H_i^l} a_{i,wh}^l. \tag{4}$$

Similarly, we compare the activation value of each location of a with the threshold to obtain the corresponding ECM $\mathbf{m}_i^l$, in which $\mathbf{m}_{i,wh}^l$ can be computed by

$$\mathbf{m}_{i,wh}^l = \mathbf{1}(a_{wh}^l \geq thr_i^l). \tag{5}$$

Finally, we combine all the ECMs to obtain the HECM $\mathbf{m}^l$. For each location (w, h), the value can be compute by

$$\mathbf{m}_{wh}^l = 1 + \log_{10}(1 + \sum_{i=0}^{K} \mathbf{m}_{i,wh}^l), \tag{6}$$

where K is the number of scales, and the log functions are used to normalize the important values for more stable training.

Selection of the kernel sizes. To ensure seamless and flexible integration with current CVGL algorithms, the kernel sizes of the pooling operations should be automatically adjusted according to the size of the input feature maps. Concretely, it is expected that the kernel size of the largest kernel can not large than $min(W^l, H^l)/2$ and the kernel sizes of different pooling have great variance. Suppose there are K scales of pooling operation, we can first obtain the maximal kernel size and base kernel variation stride:

$$k_m^l = min(h, w)/s, \tag{7}$$

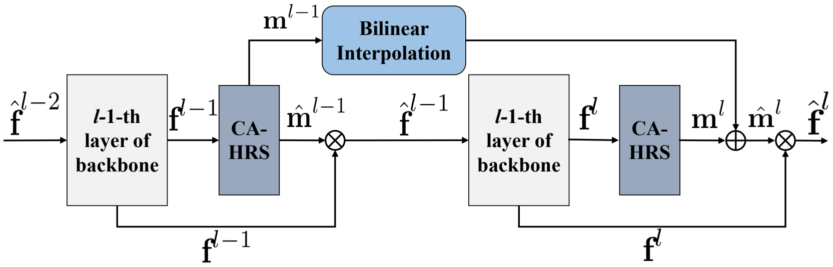

Fig. 2. A illustration of adaptive residual fusion (ARF). In particular, $\hat{\mathbf{f}}_l$ is the final enhanced feature map of l-th layer, $\hat{\mathbf{m}}_l$ is the final hierarchical enhancement coefficient map of CA-HRS of l-th layer.

$$s_b^l = max(1, min(W_l, H^l)/K - 1). \tag{8}$$

Then, we can compute the kernel size for the i-th by:

$$k_i^l = k_m^l - i \times s_b^l. \tag{9}$$

4 Adaptive residual fusion mechanism

As suggested in previous works, local information may be lost if the network goes deeper. This may lead to fuzzy boundaries of the target content, and thus resulting in degration of the CVGL performance. Inspired by previous work [9], we design an adaptive residual fusion (ARF) mechanism that takes the HECM for enhancement to avoid losing the local information. Figure 2 presents an overall computing process of the ARF mechanism. It first uses the bilinear interpolation to re-sample the previous HECM $\mathbf{m}^{l-1}$ to the same size with $\mathbf{m}^l$, and adds them to obtain the final HECM for layer l, formulated as

$$\hat{\mathbf{m}}^l = \mathbf{m}^l + \phi_{bi}(\mathbf{m}^{l-1}, W^l, H^l), \tag{10}$$

where ϕ_{bi} is the bilinear interpolation operation that re-samples the $\mathbf{m}^{l-1}$ from the size of $W^{l-1} \times H^{l-1}$ to the size of $W^l \times H^l$. Once we obtain $\hat{\mathbf{m}}^l$, we perform dot product of the final HECM $\hat{\mathbf{m}}^l$ and each channel of the feature maps:

$$\hat{\mathbf{f}}_c^l = \mathbf{f}_c^l \cdot \hat{\mathbf{m}}^l. \tag{11}$$

We perform the operation for all channels and obtain the final enhanced feature representation $\hat{\mathbf{f}}^l$.

5 Experiments

In this section, we present in-depth ablative studies to analyze the effect of each component of the proposed CA-HRS module. We also combine the CA-HRS module with current leading algorithms and compare it with state-of-the-art algorithms to show its superiority.

5.1 Experimental Settings

Datasets For a fair comparison, we follow previous works [25] to conduct experiments on the CVUSA [27], CVACT [14], and University-1652 [30] datasets. CVUSA and CVACT are two most-used datasets that cover the ground-view and satellite-view images and are used to evaluate the subtask of Ground → Satellite. Therein, CVUSA contains a training set of 35,532 ground-and-satellite image pairs and a validation set of 8884 image pairs. CVACT contains 35,532 ground-and-satellite image pairs for training, 8884 image pairs for validation, and 92,802 image pairs for testing. There exists merely one true-matched image for each query image on the CVUSA test set and exist several true-matched images for each query image on the CVACT test set. Different from the above two datasets, University-1652 covers the satellite-view and drone-view images, which are used to evaluate both two subtasks of Satellite → drone and Drone → Satellite. Specifically, in the Satellite → Drone task, it provides 37,855 drone-view images in the query set and 701 true-matched satellite-view images and 250 satellite-view distractors in the gallery. There is only one true-matched satellite-view image under this setting. In the Drone → Satellite task, it provides 701 satellite-view query images, and 37,855 true-matched drone-view images and 13,500 drone-view distractors in the gallery. There are multiple true-matched drone-view images under this setting.

Implementation Details We only perform simple data enhancement with random cropping at a certain size and a 0.5 probability of horizontal flipping for all images that are input to the network. Since there are few aerial images, we also perform a 90°random rotation operation on images. None of the above data enhancements are used in the testing stage. The height and width of the input image are set to 256. n and s are set to 3 and 2 respectively, the corresponding scale number is 3. For the first 5 epochs of training, a warmup strategy is utilized to slowly increase the learning rate to its initial value. And, after every 80 epochs, the learning rate change to 1/10 of its original value. The Stochastic Gradient Descent (SGD) is used as the optimizer. We adopt the structure of D2 for backbone ResNet-50, and adopt CA-HRS module after the last convolutional layer for backbone VGG-16 since VGG-16 does not divide the layers like ResNet-16.

Evaluation Protocol In the evaluation phase, the feature map output by backbone is transformed into a vector through the shape change of the tensor. Then the vectors belonging to the query image and gallery image will be normalized. Finally, the cosine similarity between them is calculated to measure the similarity between images, and the retrieval result is generated according to the similarity. The images ranked in the top-10 of similarity will be used as the retrieved results.

Table 1. Comparison of R@1 and AP of different integration strategies. The best results are highlighted in bold.

Structure	Drone → Satellite		Satellite → Drone	
	R@1	AP	R@1	AP
S1	62.85	66.54	75.46	62.28
S2	63.01	67.32	77.03	62.89
S3	63.85	68.36	78.07	64.82
S4/D4	63.16	67.74	77.19	63.05
D3	63.54	68.42	77.51	63.62
D2	**64.87**	**69.28**	**80.03**	**64.73**
D1	64.02	68.76	78.73	64.17

Table 2. Comparison of R@1 and AP of different scale numbers. The best results are highlighted in bold.

Scale number	Drone → Satellite		Satellite → Drone	
	R@1	AP	R@1	AP
1	63.13	67.52	76.89	63.01
2	63.93	68.01	77.85	63.28
3	**64.87**	**69.28**	**80.03**	**64.73**
4	64.16	68.53	79.19	63.92
5	64.09	68.34	78.75	63.64
6	63.82	68.17	78.46	63.33

Table 3. Comparison of R@1 and AP of the CVIM+CA-HRS with and without the ARF mechanism. The best results are highlighted in bold.

Method	Drone → Satellite		Satellite → Drone	
	R@1	AP	R@1	AP
CVIM+CA-HRS w/o ARF	63.12	67.48	78.32	64.11
CVIM+CA-HRS w/ ARF	**64.87**	**69.28**	**80.03**	**64.73**

5.2 Analyses the CA-HRS module

To analyze the effect of the CA-HRS module, and integrate it into two baselines, namely cross-view image matching (CVIM) [30] and local pattern network (LPN) [25]. Both two algorithms use the ResNet-50 and VGG-16 that have four layer blocks as the backbone.

Analysis of integration strategy As a plug-and-play module, CA-HRS can be integrated into any layer of the deep neural network. However, it may lead to different effects if integrating this module into different layers. In this part, we analyze the effect of this choice. Here, we conduct experiments using the CVIM baseline with ResNet-50 backbone on the University-1652 dataset. We design two categories of integration strategies: shallow layer integration that mainly integrates the CA-HRS in shallow layers and deep layer integration that

Table 4. Comparison of R@1, R@Top1% and AP of the LPN and CVIM with and without the CA-HRS module foar the Satellite → Drone, Drone → Satellite and Ground → Satellite subtasks on the University-1652, CVUSA and CVACT datasets. The best results are highlighted in bold.

Dataset	Task	Methods	Backbone	R@1	R@Top1%	AP
University-1652	Satellite → Drone	CVIM [30]	ResNet-50	74.47	97.15	59.45
		CVIM [30] + CA-HRS	ResNet-50	**80.03**	**98.29**	**64.27**
		LPN [25]	ResNet-50	86.45	-	74.79
		LPN [25] + CA-HRS	ResNet-50	**86.88**	**98.72**	**74.83**
	Drone → Satellite	CVIM [30]	ResNet-50	58.23	86.00	62.91
		CVIM [30] + CA-HRS	ResNet-50	**64.87**	**90.45**	**69.28**
		LPN [25]	ResNet-50	75.93	-	79.14
		LPN [25] + CA-HRS	ResNet-50	**76.67**	**93.76**	**79.77**
CVUSA	Ground → Satellite	CVIM [30]	VGG-16	43.91	91.78	-
		CVIM [30] + CA-HRS	VGG-16	**48.83**	**93.96**	**53.82**
		LPN [25]	VGG-16	79.69	98.50	-
		LPN [25] + CA-HRS	VGG-16	**84.89**	**99.39**	**87.18**
		LPN [25]	ResNet-50	85.79	99.41	-
		LPN [25] + CA-HRS	ResNet-50	**87.16**	**99.49**	**89.15**
CVACT	Ground → Satellite	LPN [25]	VGG-16	73.83	95.87	-
		LPN [25] + CA-HRS	VGG-16	**77.15**	**96.96**	**80.11**
		LPN [25]	ResNet-50	79.99	97.03	-
		LPN [25] + CA-HRS	ResNet-50	**80.91**	**97.07**	**83.20**

mainly integrates the CA-HRS in deep layers. As shown in Table 1, the backbone of ResNet-50 contains 4 layers, Sx indicates that CA-HRS is preferentially embedded in the shallow layers of ResNet-50, Dx indicates that CA-HRS is preferentially embedded in the deep layers of ResNet-50, and x represents the number of CA-HRS. We find the performance increasingly becomes better from strategy S1 to S3, as stacking more CA-HRS may better enhance feature representation. However, the performance degrades when adding more CA-HRS, i.e., S4. One possible reason for this phenomenon is that may over-emphasize the content regions and lose some less-obvious but equally-important regions. Thus, the performance inversely increases from setting D4 to D2. As shown D2 achieves the best performance for both the Drone → Satellite and Satellite → Drone subtasks. Thus, we select the D2 strategy.

Analysis of scale number The number of scales in the CA-HRS module controls the richness of the scale information and it also plays key roles in the CVGL tasks. To analyze its effect, we further conduct experiments that vary the scale number from 1 to 6, and present the performance comparisons on the University-1652 dataset. As shown in Table 2, the R@1 and AP both the Drone → Satellite and Satellite → Drone subtasks increases obviously when increasing the scale number from 1 to 3, as it enhances feature representation from more scale and thus focus more and better on the content regions. However, the R@1 and AP become saturate or even worse when further increasing it from 3 to 6. Obviously, the scale information is saturated, and thus adding more scale can

Table 5. The comparison results of LPN with CA-HRS module and current state-of-the-art competitors for the Drone → Satellite and Satellite → Drone subtasks on the University-1652 dataset. The best results are highlighted in bold.

Methods	Dataset	Backbone	Drone → Satellite		Satellite → Drone	
			R@1	AP	R@1	AP
CVIM [30]	University-1652	ResNet-50	58.49	63.31	71.18	58.74
Contrastive Loss [13]	University-1652	ResNet-50	52.39	57.44	63.91	52.24
Triplet Loss (M = 0.3) [3]	University-1652	ResNet-50	55.18	59.97	63.62	53.85
Triplet Loss (M = 0.5) [3]	University-1652	ResNet-50	55.58	58.60	64.48	53.15
Soft Margin Triplet Loss [11]	University-1652	ResNet-50	53.21	58.03	65.62	54.47
LPN [25]	University-1652	ResNet-50	75.93	79.14	86.45	74.79
LPN+CA-HRS	University-1652	ResNet-50	**76.67**	**79.77**	**86.88**	**74.83**

not help capture more information and have the risk to be over-fitting. Based on these analyses, we set the scale number as 3 in the experiments.

Analysis of the ARF mechanism In this work, we introduce the ARF mechanism to better update the HECM. Here, we further conduct an experiments to analyze its contribution by comparing the results that removes this mechanism. As shown 3, we find the R@1 and AP suffer from evident drop on both Drone → Satellite and Satellite → Drone subtasks.

Analysis of complexity and efficiency. As we introduce an additional CA-HRS module, we also analyze the model complexity and efficiency. As discussed above, the CA-HRS module does not contain any learnable parameters, and thus the model size is the same as the baselines without integrating the CA-HRS module. Here, we main analyze the number of multiply-accumulate operations (MAC) and inference time with and without the CA-HRS module. We find the number of MAC are nearly the same for both the CVIM and LPN baselines with and without the CA-HRS modules. In addition, the inference time increases from 6.80 ms to 6.87 ms and from 6.95 ms to 7.01 ms, with the relative increases of 1.03% and 0.86%, respectively. These comparisons suggest the CA-HRS does not incur additional computation overhead and is practical for real-world applications.

Contribution of CA-HRS module. As the above-mentioned description, we use the CVIM and LPR algorithms as baselines. Here, we emphasize the comparison with these two baselines to show the contribution of the CA-HRS module.

(i) Comparisons with the CVIM baseline. To ensure fair comparisons, we conduct experiments to compare the results in paper [30]. Here, we perform the comparison with the ResNet-50 as the baseline on the University-1652 dataset and with the VGG-16 as the baseline on the CVUSA dataset. The results are presented in Table 4. On the University-1652 dataset, integrating the CA-HRS module obviously improves all metrics on both Satellite → Drone, Drone → Satellite subtasks. For example, it outperforming the baseline CVIM by 5.56%, 1.14%, 4.82% in R@1, R@Top1 and AP for the Satellite → Drone task, and

Table 6. The comparison results of LPN with CA-HRS module and current state-of-the-art competitors for Ground → Satellite subtask on the CVUSA and CVACT datasets. The best results are highlighted in bold. - indicates the corresponding results are not provided.

Methods	Backbone	CVUSA				CVACT			
		R@1	R@5	R@10	R@Top1%	R@1	R@5	R@10	R@Top1%
MCVPlaces [27]	AlexNet	-	-	-	34.40	-	-	-	-
Regmi [19]	X-Fork	48.75	-	81.27	95.98	-	-	-	-
Siam-FCANet [2]	ResNet-34	-	-	-	98.30	-	-	-	-
CVM-Net [11]	VGG-16	18.80	44.42	57.47	91.54	20.15	45.00	56.87	87.57
Zhai [29]	VGG-16	-	-	-	43.20	-	-	-	-
Orientation [14]	VGG-16	27.15	54.66	67.54	93.91	46.96	68.28	75.48	92.04
CVIM [30]	VGG-16	43.91	66.38	74.58	91.78	31.20	53.64	63.00	85.27
CVFT [23]	VGG-16	61.43	84.69	90.94	99.02	61.05	81.33	86.52	95.93
LPN [25]	VGG-16	79.69	91.70	94.55	98.50	73.85	87.54	90.66	95.87
LPN+CA-HRS	VGG-16	**84.89**	**95.18**	**97.04**	**99.39**	**77.15**	**90.11**	**92.50**	**96.96**
LPN [25]	ResNet-50	85.79	95.38	96.98	99.41	79.99	90.63	92.56	97.03
LPN+CA-HRS	ResNet-50	**87.16**	**95.98**	**97.55**	**99.49**	**80.91**	**90.95**	**92.93**	**97.07**

6.64%, 4.45%, and 6.37% for the Drone → Satellite subtask, respectively. On the CVUSA dataset, it also obtains evident improvement by integrating the CA-HRS module. Specifically, the R@1 and R@Top1 improvements are 4.92% and 2.18%. These comparisons well demonstrate the effectiveness of the CA-HRS module.

(ii) Comparison with the LPN baseline. LPR is a more recent-proposed algorithm and it achieves better overall performance. Here, we also compare with the results that are reported in the original paper [25] for fair comparisons. Here, we conduct experiments with ResNet-50 as the backbone on the University-1652 dataset and with both ResNet-50 and VGG-16 on the CVUSA and CVACT datasets. As shown in Table 4, integrating the CA-HRS with the LPN algorithm also leads to performance improvement over all settings. On the University-1652 dataset, integrating the CA-HRS module improves the R@1 and AP from 86.45% and 74.79% to 86.88% and 74.83% for the Satellite → Drone and from 75.93% and 79.14% to 76.67% to 79.77%, respectively. On the CVUSA and CVACT datasets with VGG-16 as the backbone, the R@1 and R@Top1 improvements are 5.20% and 0.89% on the CVUSA dataset and 3.32% and 1.09% on the CVACT dataset. These comparisons further suggest that the CA-HRS can generalize to different baseline algorithms to facilitate the CVGL task.

5.3 Comparison with State of the Arts

In this part, we present the comparisons with current state-of-the-art algorithms to show the superiority of the proposed CA-HRS. Here, we present the results of LPN+CA-HRS as it achieves the overall best performance.

Performance on University-1652 As all of the current algorithms that have reported their results on University-1652 use the ResNet-50 as the backbone, we also present our results with the same backbone for fair comparisons. As shown in Table 5, LPN is the previous best-performing algorithm for both the Drone → Satellite and Satellite → Drone tasks, which obtains very obvious improvement compared with early works. By integrating the CA-HRS module, it can further improve the performance. Specifically, it leads to 0.74% and 0.43% R@1 improvement on both two subtasks, respectively.

Performance on CVUSA and CVACT On the CVUSA and CVACT datasets, current algorithms use ResNet-50, VGG-16, and some other networks as backbones. For fair comparisons, we divide them into three groups according to the used backbone networks for fair comparisons, i.e., ResNet-50-based, VGG-16-based, and other-net-based. Besides, current algorithms [25][23] mainly present the R@K (K=1,5,10) and R@Top1% and do not report the AP, and thus we also present these metrics for comparisons. The comparison results are presented in Table 6. When using the VGG-16 as the backbone, the current best-performing algorithm is also LPN that achieves the R@1, R@5, R@10, R@Top1% of 79.69%, 91.70%, 94.55%, 98.50% on the CVUSA dataset and 73.85%, 87.54%, 90.66%, 95.87% on the CVACT dataset. By integrating the CA-HRS module into the LPN, it boosts these metrics by 5.20%, 3.48%, 2.49%, 0.89% on the CVUSA dataset and 3.30%, 2.57%, 1.84%, 1.09% on the CVACT dataset. It is noteworthy that the improvement is more obvious for the more strict metric. When using the ResNet-50 as the backbone, the LPN can achieve even better performance compared with those using VGG-16. Expectedly, it can still improve the performance when integrating the CA-HRS module.

6 Conclusion

In this work, we design a novel yet effective content-aware hierarchical representation selection module that can be seamlessly integrated into current CVGL algorithms to facilitate the performance of CVGL. The proposed module helps to locate the content regions while ignoring the background regions to learn discriminative feature representation. We conduct extensive experiments on multiple CVGL datasets (e.g., University-1652, CVUSA and CVACT) to demonstrate the superiority of our proposed module.

References

1. Arandjelovic, R., Gronat, P., Torii, A., Pajdla, T., Sivic, J.: Netvlad: Cnn architecture for weakly supervised place recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 5297–5307 (2016)
2. Cai, S., Guo, Y., Khan, S., Hu, J., Wen, G.: Ground-to-aerial image geo-localization with a hard exemplar reweighting triplet loss. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 8391–8400 (2019)

3. Chechik, G., Sharma, V., Shalit, U., Bengio, S.: Large scale online learning of image similarity through ranking (2010)
4. Chen, T., Pu, T., Wu, H., Xie, Y., Lin, L.: Structured semantic transfer for multi-label recognition with partial labels. In: Proceedings of the AAAI conference on artificial intelligence. vol. 36, pp. 339–346 (2022)
5. Chen, T., Pu, T., Wu, H., Xie, Y., Liu, L., Lin, L.: Cross-domain facial expression recognition: A unified evaluation benchmark and adversarial graph learning. IEEE transactions on pattern analysis and machine intelligence (2021)
6. Chopra, S., Hadsell, R., LeCun, Y.: Learning a similarity metric discriminatively, with application to face verification. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05). vol. 1, pp. 539–546. IEEE (2005)
7. Deng, W., Zheng, L., Ye, Q., Kang, G., Yang, Y., Jiao, J.: Image-image domain adaptation with preserved self-similarity and domain-dissimilarity for person reidentification. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 994–1003 (2018)
8. Hadsell, R., Chopra, S., LeCun, Y.: Dimensionality reduction by learning an invariant mapping. In: 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06). vol. 2, pp. 1735–1742. IEEE (2006)
9. He, K., Zhang, X., Ren, S., Sun, J.: Identity mappings in deep residual networks. In: European conference on computer vision. pp. 630–645. Springer (2016)
10. Hsieh, M.R., Lin, Y.L., Hsu, H.W.: Drone-based object counting by spatially regularized regional proposal network. ICCV pp. 4165–4173 (2017)
11. Hu, S., Feng, M., Nguyen, R.M., Lee, G.H.: 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 (2018)
12. Lin, T.Y., Belongie, S., Hays, J.: Cross-view image geolocalization. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 891–898 (2013)
13. Lin, T.Y., Cui, Y., Belongie, S., Hays, J.: Learning deep representations for ground-to-aerial geolocalization. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 5007–5015 (2015)
14. Liu, L., Li, H.: 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 (2019)
15. Melekhov, I., Kannala, J., Rahtu, E.: Siamese network features for image matching. In: 2016 23rd International Conference on Pattern Recognition (ICPR). pp. 378–383. IEEE (2016)
16. Pu, T., Chen, T., Wu, H., Lin, L.: Semantic-aware representation blending for multi-label image recognition with partial labels. arXiv preprint arXiv:2203.02172 (2022)
17. Pu, T., Chen, T., Xie, Y., Wu, H., Lin, L.: Au-expression knowledge constrained representation learning for facial expression recognition. In: 2021 IEEE International Conference on Robotics and Automation (ICRA). pp. 11154–11161. IEEE (2021)
18. Qian, Y., Chaofeng, W., Barbaros, C., X., S.Y., Frank, M., Ertugrul, T., H., K.L.: Building information modeling and classification by visual learning at a city scale (2019)
19. Regmi, K., Shah, M.: Bridging the domain gap for ground-to-aerial image matching. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 470–479 (2019)

20. Rodrigues, R., Tani, M.: Are these from the same place? seeing the unseen in cross-view image geo-localization. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 3753–3761 (2021)
21. Shi, Y., Liu, L., Yu, X., Li, H.: Spatial-aware feature aggregation for image based cross-view geo-localization. Advances in Neural Information Processing Systems **32**, 10090–10100 (2019)
22. Shi, Y., Yu, X., Campbell, D., Li, H.: Where am i looking at? joint location and orientation estimation by cross-view matching. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4064–4072 (2020)
23. Shi, Y., Yu, X., Liu, L., Zhang, T., Li, H.: Optimal feature transport for cross-view image geo-localization. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 34, pp. 11990–11997 (2020)
24. Vo, N.N., Hays, J.: Localizing and orienting street views using overhead imagery. In: European conference on computer vision. pp. 494–509. Springer (2016)
25. Wang, T., Zheng, Z., Yan, C., Zhang, J., Sun, Y., Zhenga, B., Yang, Y.: Each part matters: Local patterns facilitate cross-view geo-localization. IEEE Transactions on Circuits and Systems for Video Technology (2021)
26. Woo, S., Park, J., Lee, J.Y., Kweon, I.S.: Cbam: Convolutional block attention module. In: Proceedings of the European conference on computer vision (ECCV). pp. 3–19 (2018)
27. Workman, S., Souvenir, R., Jacobs, N.: Wide-area image geolocalization with aerial reference imagery. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 3961–3969 (2015)
28. Xie, Y., Chen, T., Pu, T., Wu, H., Lin, L.: Adversarial graph representation adaptation for cross-domain facial expression recognition. In: Proceedings of the 28th ACM international conference on Multimedia. pp. 1255–1264 (2020)
29. Zhai, M., Bessinger, Z., Workman, S., Jacobs, N.: Predicting ground-level scene layout from aerial imagery. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 867–875 (2017)
30. Zheng, Z., Wei, Y., Yang, Y.: University-1652: A multi-view multi-source benchmark for drone-based geo-localization. In: Proceedings of the 28th ACM international conference on Multimedia. pp. 1395–1403 (2020)
31. Zheng, Z., Zheng, L., Garrett, M., Yang, Y., Xu, M., Shen, Y.D.: Dual-path convolutional image-text embeddings with instance loss. ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM) **16**(2), 1–23 (2020)
32. Zhu, P., Wen, L., Bian, X., Ling, H., Hu, Q.: Vision meets drones: A challenge. arXiv: Computer Vision and Pattern Recognition (2018)
33. Zhu, S., Yang, T., Chen, C.: Revisiting street-to-aerial view image geo-localization and orientation estimation. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 756–765 (2021)