RETRIEVE THE VISIBLE FEATURE TO IMPROVE THERMAL PEDESTRIAN DETECTION USING DISCREPANCY PRESERVING MEMORY NETWORK

*Yuxuan Hu[1,2], Ning Zhang [3] and Lubin Weng[4]**

[1]State Key Laboratory of Multimodal Artificial Intelligence Systems, CASIA, Beijing, China
[2]School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing, China
[3]Shanghai Aerospace Electronic Technology Institute, Shanghai, China
[4]Research Center of Aerospace Information, Institute of Automation, CAS, Beijing, China

2023 IEEE International Conference on Image Processing (ICIP) | 978-1-7281-9835-4/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICIP49359.2023.10222560

ABSTRACT

We propose an approach for enhancing pedestrian detection in thermal infrared images using paired visible-thermal images in training. Recently, approaches that retrieve the corresponding visible features from thermal features using a key-value memory network have been proven effective for improving detection results. However, for memory networks storing thermal-visible features, random initialization and end-to-end training may not be ideal, as this can reduce the diversity of memory slots. Also, the retrieved visible features have different reliability as the overall similarities between key slots in the memory network and thermal features differ. These motivate us to propose a DIscrepancy Preserving (DIP) Memory that is updated manually to prevent convergence of key-value memory slots. We also evaluate the reliability of each retrieved visible feature and adjust the training protocol of the detection head. Experiment results on two visible-infrared pedestrian detection datasets demonstrate the superiority of our framework.

Index Terms— Thermal infrared pedestrian detection, DIscrepancy Preserving (DIP) memory

1. INTRODUCTION

Pedestrian detection is a crucial research area of object detection and has numerous applications in autonomous driving and video surveillance [1]. Visible images are vulnerable to changes in illumination, and multispectral pedestrian detection is gaining popularity [2, 3] as it achieves around-the-clock detection. However, it requires sensors with beam splitter configuration to acquire registered image pairs [4]. Such sensor is expensive to use in real-world applications. What's more, deep-leaning based multispectral pedestrian detection networks require much computation as they usually adopt two-stream architecture. Furthermore, the use of visible or multispectral systems can raise privacy concerns [5]. In this context, thermal imagery-based pedestrian detection is gaining popularity in recent years.

Using thermal infrared images can perform well at night but deteriorates at daytime due to massive heat radiation. There's a compromise solution of using visible images in aiding the thermal images in training only. Kieu *et al.* [5] introduced an auxiliary classification task and proposed the conditioning layer to utilize the internal representation. They also proposed the bottom-up and layerwise adaptation strategies to train the detector incrementally [6]. Some researchers have used GANs [7, 8, 9] to generate pseudo visible images for use in training. Marnissi *et al.* [10] considered this problem as an unsupervised domain adaptation problem.

Recently, Park *et al.* [11] proposed the VPA Memory for storing paired thermal-visible RoI features in a key-value memory network. In the testing phase, the thermal features serve as queries to recall the visible features from memory. Similar architecture has been applied to enhance the detection results of small-scale pedestrians [12] and build detectors that can handle both visible and thermal inputs [13]. Memory networks were first used in QA tasks [14], and since then they have been utilized in other tasks such as anomaly detection [15], video object detection [16], and multimodal learning [17]. Generally, the parameters of the memory network can be updated automatically by back-propagation or manually with specially designed algorithms.

In this paper, we build upon the work of Park *et al.* and present the DIscrepancy Preserving (DIP) Memory. The diversity of memory slot pairs is crucial for achieving better generalization, but end-to-end training cannot ensure this diversity, as there may exist many similar RoI features in training. We have designed a specific algorithm to update the parameters of the DIP memory and explicitly delete adjacent slot pairs. Furthermore, we evaluate the reliability of each retrieved feature and incorporate this information into the training protocol. Experiments on the KAIST [18] and FLIR [19] datasets demonstrate the superiority of our framework.

2. PROPOSED METHOD

Fig. 1 illustrates the overall architecture of our framework. We choose Faster RCNN [20] as the base detector and two

*Corresponding author(lubin.weng@ia.ac.cn)

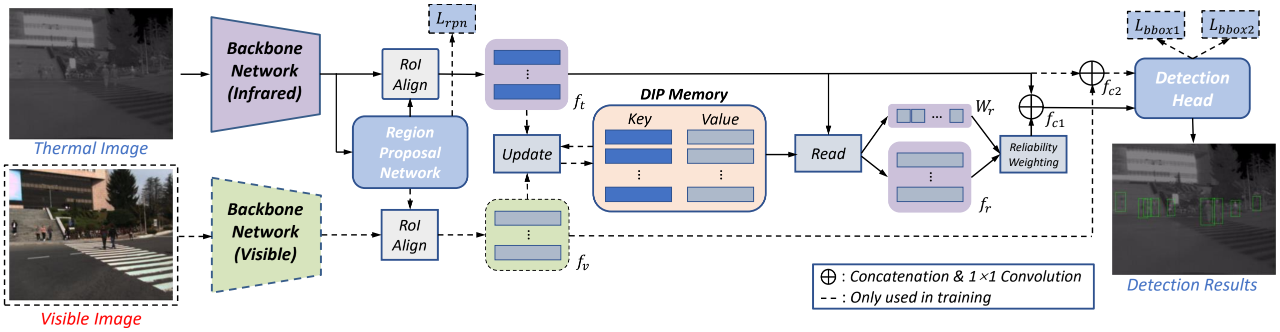

Fig. 1: Overall architecture of our framework. The read operation retrieves f_r and reliability score W_r using thermal RoI features f_t from the DIP memory. The paired RoI features of thermal and visible images f_t and f_v are used to update the key-value slot pairs of the DIP memory. The data flows and structures represented by dashed lines are only used in training.

backbone networks are employed to extract features of thermal and visible images in training. The region proposals are generated using thermal feature map and paired RoI features f_t and f_v are utilized to update the key-value slot pairs in the DIP memory. For the read operation, we get the retrieved RoI features f_r and the corresponding reliability scores W_r. After concatenation and convolution, the final RoI features f_{c1} and f_{c2} are feed into the detection head to compute L_{bbox1} and L_{bbox2}. For each training iteration, the initialize or update operation is conducted first, followed by the read operation.

2.1. Initialize and Update Operation

We initialize or update the parameters of the DIP memory manually and delete the adjacent slot pairs explicitly to preserve discrepancy. In each training iteration, N pairs of RoI features f_t^i and f_v^i $(i = 1 \dots N)$ with shape $c{\times}h{\times}w$ are used. Let K_j and V_j $(j = 1 \dots L)$ denote a key-value slot pair in the DIP memory, they have the same shape as f_t^i and f_v^i. We randomly choose two pairs of RoI features and randomly generate two weights that sum to 1 to initialize every key-value slot pair at the first iteration.

As we use N pairs of RoI features in an iteration, we can get a similarity matrix $S_{KF} \in \mathbb{R}^{N \times L}$ by calculating cosine similarity between each pair of f_t^i and K_j. The softmax function is applied to S_{KF} along the vertical direction to get W_V.

Now we assign each RoI feature pair to the memory slot pair with highest similarity. The set of RoI feature pair indices assigned to the ith memory slot pair is U_F^i:

$$U_F^i = \{i | i = argmax(S_{KF}^{j,0}, \dots S_{KF}^{j,L}), \forall j\}. \tag{1}$$

We update the key-value memory pairs using EMA:

$$K_i = \gamma K_i + (1-\gamma) \sum_{k \in U_F^i} w_i^k f_t^k, \tag{2}$$

$$V_i = \gamma V_i + (1-\gamma) \sum_{k \in U_F^i} w_i^k f_v^k, \tag{3}$$

$$w_i^k = \frac{W_V^{i,k}}{\sum_{k \in U_F^i} W_V^{i,k}}, \tag{4}$$

where $W_V^{i,k}$ means the element in the ith row and kth column of W_V. γ is a hyperparameter and we set it to 0.9.

To preserve discrepancy, we should delete the adjacent memory slot pairs each iteration before updating. We can get the cosine similarity matrix $S_K \in \mathbb{R}^{L \times L}$ between every K_i pairs and select the indices of key-value pairs having a similarity score higher than the threshold τ to get the set U_D:

$$U_D = \{j | S_K^{i,j} > \tau, j < i, \forall i\}, \tag{5}$$

and memory slot pairs whose indices are in U_D will not be assigned to any f_t^i-f_v^i pairs as in (1).

We substitute the key-value pairs using RoI feature pairs that have low similarity with the key slots. Set U_Q contains the indices of f_t^i whose maximum cosine similarity score is lower than a threshold δ:

$$U_Q = \{i | max(S_{KF}^{i,0}, \dots S_{KF}^{i,L}) < \delta, \forall i\}. \tag{6}$$

Usually $|U_Q|$ is larger than $|U_D|$, so we sort the elements of U_Q in ascending order according to the maximum similarity score and substitute each key-value pairs in U_D with f_t^i-f_v^i pairs in U_Q in order.

2.2. Read Operation

Read operation is performed after update operation in training. We recompute S_{KF} and apply softmax to it along the horizontal direction to get W_H. The retrieved feature f_r^i and the reliability score W_r^i of f_r^i are computed as follows:

$$f_r^i = \sum_{j=1}^{L} W_H^{i,j} V_j, \tag{7}$$

$$W_r^i = max(S_{KF}^{i,0}, \dots S_{KF}^{i,L}). \tag{8}$$

2.3. Reliability Weighting

The reliability score measures the domain discrepancy between query feature f_t and the key slots of DIP memory. Since the number of memory slots is limited and the distribution of queries is much wider than the space of memory

Table 1: Comparison with other thermal pedestrian detectors trained using thermal and visible images on KAIST dataset.

Method	MR_all	MR_day	MR_night
Domain Adaptor† [8]	42.65	49.59	26.70
Kieu *et al.*† [9]	25.62	31.86	12.92
Bottom-up [6]	22.54	29.04	9.65
TC Det [5]	22.17	28.64	9.21
Park *et al.* [11]	20.83	26.68	9.81
Ours	**19.12**	**25.16**	**7.49**

Table 2: Comparison with other thermal detectors trained using thermal and visible images on FLIR dataset.

Method	car	person	bicycle	mAP
Bottom-up [6]	82.09	70.17	50.55	67.60
TC Det [5]	85.49	74.42	26.93	72.28
Park *et al.* [11]	**87.31**	80.42	61.88	76.54
Ours	86.95	**80.66**	**62.93**	**77.28**

keys, it is natural that some queries have low similarity with all key slots. These queries result in unreliable retrieved features, so we focus more on the queries themselves during the detection process. The reliability weighting in generating final RoI features can be formulated as follows:

$$f_{c1}^i = \mathrm{Conv}([f_t^i, v_i f_r^i]), \tag{9}$$

$$v_i = \frac{1}{1 + e^{-\alpha(W_r^i - \beta)}}, \tag{10}$$

where Conv represents 1×1 convolution, $[\cdot]$ denotes concatenation operation and α, β are hyperparameters. The same convolution layer is used in generating f_{c2}^i in training.

3. EXPERIMENTS

3.1. Data Description

The KAIST dataset is a popular benchmark for visible-thermal pedestrian detection. We use the sanitized training annotations [21] and Liu's testing annotations [21]. The experiments are conducted on the “reasonable” setting [18] and log average miss rate (MR) is the metric. We use the evaluation code provided by Kim *et al.* [4]. FLIR is a road scene dataset and we use the “aligned” version [22] with the three most frequent classes. VOC2007 style AP50 is the metric.

3.2. Implementation Details

We use pretrained VGG16 [23] as the backbone to make a fair comparison with other works for KAIST and pretrained ResNet50 [24] with FPN [25] for FLIR. The RoI features have 7×7 feature maps, and we select 256 RoIs to train the detection head per image. Our experiments are implemented on MMDetection [26] toolbox using a TITAN Xp GPU. We train the network using SGD optimizer with batchsize 4 and initial learning rate of 0.004. For KAIST, we train for 4 epochs and “1x” scheduler is used to train FLIR. The default number of memory slots L is 100. For other hyperparameters, $\tau = 0.9$, $\delta = 0.5$, $\alpha = 20$ and $\beta = 0.75$. When calculating the softmax function, we use a temperature parameter of 0.0625.

Table 3: Ablation study on the KAIST dataset.

Memory	DIP	RW	MR_all	MR_day	MR_night
✗	✗	✗	24.44	31.27	11.41
✓	✗	✗	20.26	26.36	9.05
✓	✓	✗	19.54	**24.86**	8.79
✓	✓	✓	**19.12**	25.16	**7.49**

Table 4: Effects of memory slot number L on KAIST dataset

L	25	50	100	150	200
MR_all	20.4	20.24	**19.12**	19.81	19.35
MR_day	26.76	26.34	**25.16**	25.83	25.44
MR_night	8.29	9.32	**7.49**	8.19	8.05

3.3. Comparison to State-of-the-art

To validate the superiority of our framework, we compare the performance with other thermal detectors trained using both thermal and visible images. The results on KAIST and FLIR datasets are shown in Tab. 1-2. We re-implement the methods for fair comparison except for those marked with a dagger in the tables. Our framework outperforms Park *et al.* [11] whose VPA memory is trained end-to-end in all three metrics on the KAIST dataset. On the more challenging FLIR dataset with diverse scenes and multiple object categories, our framework also achieves consistent improvements.

3.4. Detection Results Visualization

In Fig. 2, we visualize some detection results on the two datasets in comparison to Park *et al.* [11]. The first and second columns are visible and thermal images with ground truth annotations. The rest two columns display detection results. Remember that we only use thermal images for testing, the thermal features are somewhat not distinguishable and false positives occur. Better visible features can be retrieved in our framework to achieve more robust classification.

3.5. Ablation Study

We conduct the ablation study to investigate the effects of our DIP memory and the reliability weighting. In Table 3, “Memory” signifies the DIP memory without the deletion procedure in updating, and “RW” refers to the reliability weighting of retrieved RoI features. The first row is the simple Faster RCNN baseline. The effectiveness of deleting adjacent slot pairs is evident from the table. Reliability weighting can further improve performance. In Table 4, we change the number of memory slots from 25 to 200. Decreasing memory slot numbers significantly affects the miss rate and increasing the number of slots doesn't guarantee better performance.

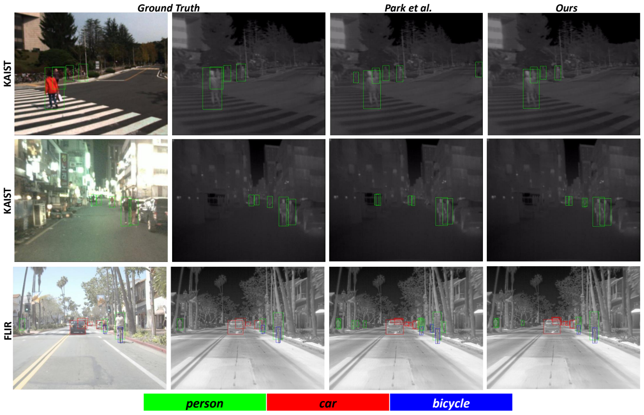

Fig. 2: Qualitative comparison of detection results on the two datasets. We visualize the results of our framework, Park *et al.* [11] and the ground truth annotations.

4. DISCUSSION

4.1. Discrepancy Preserving and Deletion Procedure

The domain shift between the training set and testing set exists and under certain circumstances, the difference of distribution between key slots and RoI features can be large. We find a test image to exemplify this phenomenon using t-SNE in Fig. 3(a). The red dots represent key slots and blue dots query features. Almost all keys are far away from the queries for VPA memory [11], while some keys of our DIP memory are close to the queries, indicating greater discrepancy and better generalization. In Fig. 3(b) we visualize the cosine similarity matrices between key (lower left part) and value slots (upper right part) with or without the proposed deletion procedure. The locations marked with yellow denotes similarity scores higher than 0.9. It is clear that without deletion, high-similarity key slot pairs exist and value slots have high concentration, reducing the diversity of retrieved RoI features.

4.2. Reliability of RoI features

In our framework, thermal RoI features with low similarity with the key slots will have low reliability scores on the retrieved features. This approach is empirical but in general, these retrieved features may provide incorrect information. In Fig. 3(c) we select a pair of images from the KAIST dataset and visualize the thermal and retrieved features of two RoIs marked with red and green with different reliability scores. For the green RoI, the f_r concentrates differently with the reference f_v while f_r and f_v of the red RoI have more similar patterns. So it is reasonable to down-weight the retrieved features with low reliability scores to reduce interference.

4.3. Distribution of memory slots

As RoI features can be divided into positive and negative samples in training, we investigate the distribution of key slots to see the proportion of slot pairs near positive and negative samples. We select part of all KAIST training samples and assign each RoI feature to the nearest key slot, as in Sec. 2.1. In Fig. 3(d), the proportion of RoI features assigned to each key slot as a percentage of total feature numbers is shown. It can be concluded from the figure that the concentration of positive and negative features are different, a key slot either attracts positive samples or negative samples. So it is not necessary to explicitly divide the memory slots into positive and negative parts for datasets with only one object category, but it deserves further study for multi-category datasets.

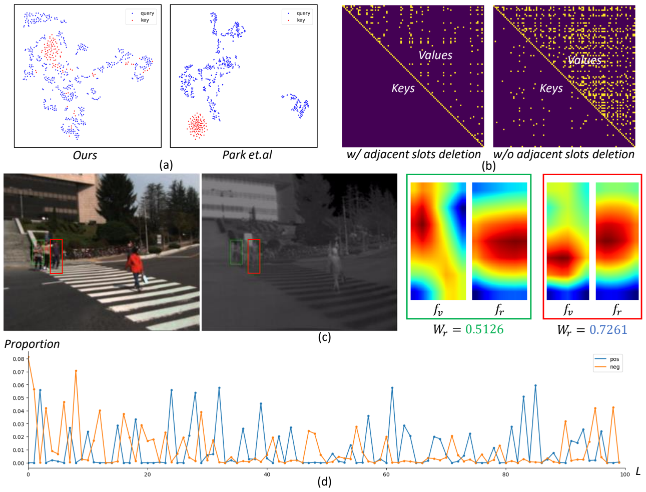

Fig. 3: Some visualized analysis. (a) exemplifies the key slots of our DIP memory has greater discrepancy and better generalization. (b) illustrates the effects of deletion procedure. (c) provides examples of RoI features with different reliability. (d) shows the different concentration distributions of positive and negative samples on DIP memory slots.

5. CONCLUSION

In this paper, we propose a novel DIscrepancy Preserving (DIP) Memory for retrieving visible RoI features to improve the performance of two-stage thermal pedestrian detectors. Unlike previous key-value memory network which is trained end-to-end, our DIP memory is manually updated and adjacent slot pairs are deleted, resulting in better generalization. Additionally, we introduce a reliability weighting mechanism to mitigate the interference brought by the unreliable retrieved features. Comprehensive experiments and visualizations demonstrate the effectiveness of our DIP memory and the improvement it brings to thermal pedestrian detection. Code is available at `https://github.com/a21401624/DIP_memory`.

6. ACKNOWLEDGEMENTS

This research was supported by the National Key Research and Development Program of China under Grant No.2018AAA0100400, and the National Natural Science Foundation of China under Grant 62076242.

7. REFERENCES

[1] Di Feng, Christian Haase-Schütz, Lars Rosenbaum, et al., "Deep multi-modal object detection and semantic segmentation for autonomous driving: Datasets, methods, and challenges," *IEEE TITS*, vol. 22, no. 3, pp. 1341–1360, 2021.

[2] Kailai Zhou, Linsen Chen, and Xun Cao, "Improving multispectral pedestrian detection by addressing modality imbalance problems," in *ECCV*, 2020, pp. 787–803.

[3] Tianshan Liu, Kin-Man Lam, Rui Zhao, et al., "Deep cross-modal representation learning and distillation for illumination-invariant pedestrian detection," *IEEE TCSVT*, vol. 32, no. 1, pp. 315–329, Jan. 2022.

[4] Jiwon Kim, Hyeongjun Kim, Taejoo Kim, et al., "MLPD: Multi-label pedestrian detector in multispectral domain," *IEEE RA-L*, vol. 6, no. 4, pp. 7846–7853, 2021.

[5] My Kieu, Andrew D. Bagdanov, Marco Bertini, et al., "Task-conditioned domain adaptation for pedestrian detection in thermal imagery," in *ICCV*, 2020, pp. 546–562.

[6] My Kieu, Andrew D. Bagdanov, and Marco Bertini, "Bottom-up and layerwise domain adaptation for pedestrian detection in thermal images," *ACM TOMM*, vol. 17(1), pp. 1029–1038, apr 2021.

[7] Chaitanya Devaguptapu, Ninad Akolekar, Manuj M Sharma, et al., "Borrow from anywhere: Pseudo multimodal object detection in thermal imagery," in *CVPRW*, 2019, pp. 1029–1038.

[8] Tiantong Guo, Cong Phuoc Huynh, and Mashhour Solh, "Domain-adaptive pedestrian detection in thermal images," in *ICIP*, 2019, pp. 1660–1664.

[9] My Kieu, Lorenzo Berlincioni, Leonardo Galteri, et al., "Robust pedestrian detection in thermal imagery using synthesized images," in *ICPR*, 2021, pp. 8804–8811.

[10] Mohamed Amine Marnissi, Hajer Fradi, Anis Sahbani, et al., "Unsupervised thermal-to-visible domain adaptation method for pedestrian detection," *PRL*, vol. 153, pp. 222–231, 2022.

[11] Sungjune Park, Dae Hwi Choi, Jung Uk Kim, et al., "Robust thermal infrared pedestrian detection by associating visible pedestrian knowledge," in *ICASSP*, 2022, pp. 4468–4472.

[12] Jung Uk Kim, Sungjune Park, and Yong Man Ro, "Robust small-scale pedestrian detection with cued recall via memory learning," in *ICCV*, 2021, pp. 3030–3039.

[13] Jung Uk Kim, Sungjune Park, and Yong Man Ro, "Towards versatile pedestrian detector with multisensory-matching and multispectral recalling memory," in *AAAI*, 2022, vol. 36(1), pp. 1157–1165.

[14] Alexander Miller, Adam Fisch, Jesse Dodge, et al., "Key-value memory networks for directly reading documents," in *EMNLP*, Nov. 2016, pp. 1400–1409.

[15] Dong Gong, Lingqiao Liu, Vuong Le, et al., "Memorizing normality to detect anomaly: Memory-augmented deep autoencoder for unsupervised anomaly detection," in *ICCV*, 2019, pp. 1705–1714.

[16] Hanming Deng, Yang Hua, Tao Song, et al., "Object guided external memory network for video object detection," in *ICCV*, 2019, pp. 6677–6686.

[17] Minsu Kim, Joanna Hong, Se Jin Park, et al., "Multimodality associative bridging through memory: Speech sound recollected from face video," in *ICCV*, 2021, pp. 296–306.

[18] Soonmin Hwang, Jaesik Park, Namil Kim, et al., "Multispectral pedestrian detection: Benchmark dataset and baseline," in *CVPR*, 2015, pp. 1037–1045.

[19] F. Team et al., "Free FLIR thermal dataset for algorithm training," Online, 2019, `https://www.flir.com/oem/adas/adas-dataset-form/`.

[20] Shaoqing Ren, Kaiming He, Ross Girshick, et al., "Faster R-CNN: Towards real-time object detection with region proposal networks," in *NIPS*, Dec. 2015, vol. 28, pp. 91–99.

[21] Chenyang Li, Dan Song, Ruofeng Tong, et al., "Multispectral pedestrian detection via simultaneous detection and segmentation," in *BMVC*, Sept. 2018, pp. 225.1–225.12.

[22] Heng Zhang, Elisa Fromont, et al., "Multispectral fusion for object detection with cyclic fuse-and-refine blocks," in *ICIP*, Oct. 2020, pp. 276–280.

[23] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *ICLR*, May 2015, pp. 1–14.

[24] Kaiming He, Xiangyu Zhang, Shaoqing Ren, et al., "Deep residual learning for image recognition," in *CVPR*, June 2016, pp. 770–778.

[25] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, et al., "Feature pyramid networks for object detection," in *CVPR*, July 2017, pp. 936–944.

[26] Kai Chen, Jiaqi Wang, Jiangmiao Pang, et al., "MMDetection: Open mmlab detection toolbox and benchmark," *arXiv preprint arXiv:1906.07155*, 2019.