# Multispectral Human Presence Detection using Adapted YOLO Network

Frencis Balla and Raju Shrestha
Department of Computer Science, Oslo Metropolitan University (OsloMet)
Oslo, Norway

***Abstract*—Search and Rescue (SAR) operations increasingly use drones with thermal cameras to locate individuals in distress. However, human operators are required for navigation and detection. Given modern drones' self-navigation capabilities, automating human detection can enhance efficiency. While previous studies leveraged multi-spectral imagery, specifically RGB and thermal, for object detection, they often introduced computational overhead. Building on recent YOLO advancements, we adapt YOLOv8 to support multi-spectral input using early feature fusion, modified convolution kernels, and improved up-scaling blocks for better small-object detection. Experiments on RGB-thermal multispectral data show a 22% mAP50-95 improvement over the baseline and a 10% gain over prior work, while maintaining real-time performance.**

***Index Terms*—human detection, object detection, multi-spectral, thermal, real-time, yolo**

## I. Introduction

Human presence detection locates individuals in an environment using techniques ranging from surveillance cameras to autonomous sensor-equipped systems [1]. This task is crucial for Search And Rescue (SAR) operations, which assist individuals in distress due to disasters, accidents, or emergencies. Given the narrow time window for saving lives [2], rapid detection is essential.

Traditional SAR methods, such as foot searches, rely on volunteers and can be unreliable in inaccessible terrains, requiring costly specialized equipment. Some SAR institutions, like Idaho Mountain SAR[1], have adopted unmanned aerial vehicles (UAVs) equipped with thermal cameras, enabling rapid area coverage and accurate victim localization. UAVs are cost-effective, operate in hazardous environments without endangering human lives, and facilitate efficient large-scale searches [3]. However, each UAV requires a human operator, limiting scalability.

Recent advancements in autonomous UAV navigation suggest that integrating AI-driven object detection can enhance SAR efficiency [4]. AI enables multiple UAVs to operate simultaneously without direct human supervision, making operations scalable by investing in additional drones rather than personnel.

SAR environments present challenges such as dense vegetation, rugged terrain, and adverse weather. Thermal imaging mitigates visibility issues by detecting infrared emissions from human bodies [5] as illustrated in Fig. 1. However, thermal cameras lack the fine details of visible-spectrum cameras, which are useful in distinguishing humans from heat-emitting background elements. A fusion of visible and thermal data offers a promising solution.

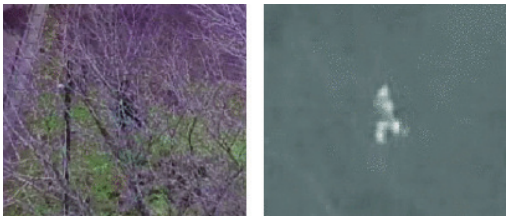


Fig. 1: Example of a human presence not visible in the RGB image (left) but visible in the thermal image (right) [6].

Several studies on multispectral human detection [7–9] have increased model complexity, raising computational demands. Ozyurt *et al.* [6], use thermal images exclusively, reducing accuracy. Balancing complexity and detection accuracy is crucial, particularly given UAV constraints on power and computation [10]. The YOLO model family [11], known for real-time efficiency, presents a viable approach.

Additionally, recognizing tiny human figures in aerial images remains challenging due to typical UAV flight altitudes (50–200m) [7, 9, 12]. To address these issues, we extend the YOLOv8 to support $n$-channel multispectral imagery while enhancing small-object detection. Our modifications improved `mAP50-95` by 22% over the baseline and 10% over a recent study. Despite a minor inference speed reduction (7 FPS), our model maintains real-time performance at 52 frames per second (FPS).

These improvements, combined with UAV self-navigation, can help SAR institutions scale operations efficiently while retaining compatibility with future YOLO updates.

## II. Related Work

Recent advancements in object detection have focused on improving accuracy, efficiency, and feature representation. Li *et al.* [13] introduced the Large Selective Kernel Network (LSKNet) for remote sensing, dynamically adjusting receptive fields to model contextual information. Its spatial selective mechanism across large depth-wise kernels achieved state-of-the-art results on HRSC2016[2] and DOTA-v1.0[3] datasets.

[1]Idaho Mountain Search & Rescue: https://imsaru.org/

[2]HRSC2016: https://www.kaggle.com/datasets/guofeng/hrsc2016

[3]DOTA: https://captain-whu.github.io/DOTA/

YOLO-based architectures have evolved significantly. A recent widely used version, YOLOv8, is adapted for UAV image recognition using Bi-PAN-FPN, GhostblockV2, and WiseIoU loss, improving small target detection on VisDrone2019[4] [14]. Wang *et al.* [15] proposed YOLOv9, integrating Programmable Gradient Information (PGI) and Generalized Efficient Layer Aggregation Network (GELAN) to enhance gradient flow and feature reuse, outperforming previous versions on MS-COCO [16]. Several variants have specifically targeted small object detection. YOLO-S [17] uses a lightweight architecture with compact feature extractors and skip connections for aerial imagery. YOLO-C [18] employs attention mechanisms and deformable convolutions to enhance robustness in agricultural and traffic domains. Scaled-YOLOv4 [19] improves performance via cross-stage partial network scaling. However, these models primarily rely on RGB inputs and lack native support for multispectral fusion, limiting their adaptability to complex environments.

To address the limitations of thermal-only imaging, Ozyurt *et al.* [6] applied CLAHE filtering to convert single-channel thermal images into a 3-channel format, enhancing background detail while maintaining real-time efficiency.

Multispectral object detection has been explored in autonomous driving and surveillance. Roszyk *et al.* [7] extended YOLOv4 [20] for pedestrian detection using early, middle, and late fusion strategies. Middle fusion, which added a dedicated backbone for thermal features, achieved the best performance but increased model complexity. Similarly, Xie *et al.* [8] improved YOLOv5 with a Feature Interaction and Self-Attention Fusion Network (FISAFN) to handle lighting variation, but relied on a dual-backbone architecture limited to 4-channel inputs.

Zou *et al.* [9] enhanced YOLOv5 with a multidimensional attention mechanism and background suppression loss, improving multispectral feature fusion at the cost of increased complexity. For small object detection, Tang *et al.* [21] proposed HIC-YOLOv5, incorporating an extra prediction head and involution blocks to boost mAP on VisDrone-2019-DET dataset[5].

While these works demonstrate promising directions, many rely on RGB-pretrained weights or fixed input modalities, which may limit generalization. In contrast, our approach adapts YOLOv8 for flexible multispectral fusion using a streamlined architecture. Its balance between accuracy and efficiency makes it well-suited for real-time search and rescue scenarios, and our modifications can be extended to future YOLO versions.

## III. Modified YOLO Model

This section presents our modifications to YOLOv8 for multispectral detection while preserving its core architecture. Key changes include adapting the backbone to support general $n$-channel inputs, adjusting kernel sizes, and enhancing up-sampling with bicubic interpolation.

### A. Backbone Modification

The default YOLOv8 backbone is designed for three-channel RGB images, limiting its application to multispectral data. To enable support for general $n$-channel inputs, we modified the first convolutional layer, which processes input images before feature extraction begins (Fig. 2). This modification allows the backbone to accommodate various multispectral inputs without altering its fundamental structure.

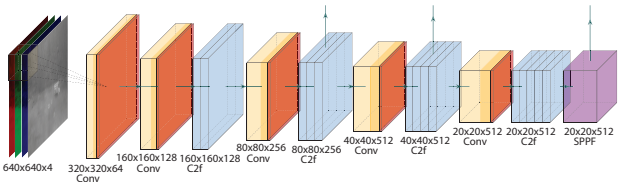



Fig. 2: Modified YOLOv8 backbone network.

In this research, we fuse RGB and thermal images into a 4-channel input before feeding them into the network. The backbone consists of multiple convolutional layers and C2f blocks [11], which extract features at different levels. By modifying the input layer to accept an arbitrary number of channels, we ensure compatibility with multispectral datasets while preserving the hierarchical feature extraction process.

### B. Kernel Size Adaptation

Since the original backbone uses 3×3×3 kernels in its first convolutional layer, we adapted the kernel size to 3×3×$n$ for multispectral images. For our 4-channel RGBT input, the kernel is set to 3×3×4, keeping stride and padding unchanged. This lets the network extract low-level features from all spectral channels without affecting later layers.

By modifying only the kernel size, the backbone continues to generate feature maps with the same depth as for 3-channel inputs, ensuring compatibility with the downstream C2f blocks and the PAN (Path Aggregation Network) used in YOLOv8's neck. This maintains the model's ability to refine and aggregate features without requiring additional structural changes.

### C. Enhanced Up-Sampling with Bicubic Interpolation

In YOLOv8, up-sampling layers in the neck use nearest neighbor interpolation [22] to align feature maps from the backbone with those from the Spatial Pyramid Pooling-Fast (SPPF) block [23]. Although efficient, nearest neighbor interpolation can cause information loss, especially for small object detection.

To improve feature retention, we replaced nearest neighbor interpolation with bicubic interpolation [24]. Unlike nearest neighbor, which considers only four pixels, bicubic interpolation utilizes a 16-pixel neighborhood, preserving finer details during up-scaling. This modification enhances the model's ability to capture small object features, potentially improving detection accuracy.

[4]VisDrone-Dataset: https://github.com/VisDrone/VisDrone-Dataset

[5]VisDrone 2019 Object Detection Challenge Dataset: https://www.kaggle.com/datasets/shisuiotsutsuki/visdrone2019-det

## IV. Experiments and Results

We implemented the model using the PyTorch-based Ultralytics YOLOv8 framework[6], trained on an NVIDIA A100 GPU with an Adam optimizer, an initial learning rate of 0.001, a batch size of 16, and an input image resolution of $640 \times 640$. Data augmentation, including mosaic augmentation, random perspective transformations, and random HSV adjustments, was used to improve generalization. Evaluation was based on precision, recall, and mean Average Precision (mAP) at various IoU thresholds.

### A. Ablation Study

To assess the baseline and modified models, we conducted an ablation study using the NII-CU dataset [25], which contains numerous tiny objects. The baseline model, trained on the RGB portion of the dataset using transfer learning from MS-COCO [16], ran for 234 epochs.

In the first modification, we fused visible and thermal images, upgrading the backbone for 4-channel inputs, and trained it for 319 epochs. In the second modification, we added the upgraded up-sampling method, and trained for 404 epochs.

TABLE I: Results of the ablation study on the NII-CU dataset.

| Model | Precision | Recall | mAP50 | mAP50-95 |
|---|---|---|---|---|
| Baseline | 0.867 | 0.759 | 0.851 | 0.448 |
| + 4CH | 0.958 | 0.950 | 0.979 | 0.667 |
| + 4CH + Bicubic | **0.980** | **0.973** | **0.989** | **0.675** |

Table I shows significant improvements with each modification. The 4-channel input boosted precision, recall, and mAP by 19-21%. The upgraded up-sampling method provided an additional 1% improvement in mAP and 2% in precision/recall.

These results suggest that 4-channel inputs enhance detection accuracy by leveraging thermal images, which are less influenced by environmental factors. This is confirmed visually in Fig. 3, where the modified model detects occluded objects more reliably than the baseline. The improved up-sampling method further refines feature map resolution, aiding in the localization of small objects.

### B. Visual Inspection

To visualize the improvements, we present the prediction results of the baseline and best-performing model alongside the ground truth labels one one random sample image, in Fig. 3. As shown in Fig. 3b, the base model struggles to detect human presence in areas with dense vegetation, resulting in false negatives. In contrast, our model reliably detects occluded objects with higher confidence (see Fig. 3c).

### C. Accuracy Over Scale

We tested the accuracy of our proposed solution across different model scales to assess performance trade-offs for use in production environments. We trained scale variants of our best-performing model (nano, small, medium, large, and extra large) on the NII-CU dataset, with parameters ranging from 3 to 68 million. The mAP50-95 metric was used for comparison.

[6]Ultralytics YOLOv8: https://docs.ultralytics.com/models/yolov8/


(a) Ground Truth (b) Base Model (c) Ours

Fig. 3: Sample images comparing predictions from the two models with ground truth on the NII-CU dataset.

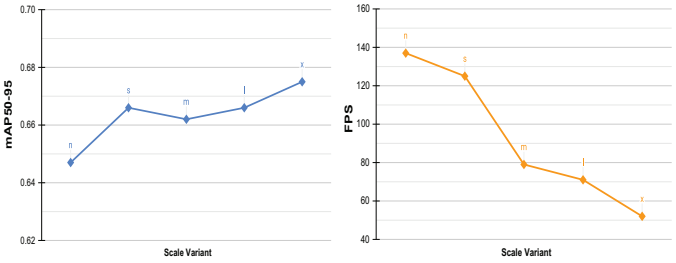
As shown in Fig. 4a, performance decreases with smaller models but remains above 64% accuracy for the smallest (nano) variant. The three middle variants exhibit similar accuracy, with the largest model achieving 67.5%.

The accuracy over scale experiment highlights the trade-off between model size and accuracy. While smaller models offer faster inference, they come with a performance penalty. The fact that even the smallest variant maintains over 64% accuracy suggests that it could be suitable for resource-constrained environments where real-time performance is critical.

### D. Speed Over Scale

We also evaluated the speed of inference for different model scales to assess trade-offs between accuracy and speed. The inference speed was measured on the validation set from the NII-CU dataset.

As expected, the inference speed improves with smaller models, while larger models yield higher accuracy but lower inference speed. The smallest (nano) model achieved an average inference speed of 95 FPS, while the largest model (extra-large) reached 25 FPS. These results are plotted in Fig. 4b.




(a) Accuracy (b) Inference speed

Fig. 4: Accuracy and inference speed of our proposed models across different scale variants of YOLOv8.

This speed/accuracy trade-off allows practitioners to choose a model size based on their specific application requirements, optimizing for either speed or accuracy as necessary. The speed over scale experiment further demonstrates the trade-off between model size and speed, allowing for the selection of an appropriate model variant based on the specific application requirements.

### E. Comparative Study

We extended the YOLOv8 network to support 4-channel imagery for enhanced detection and compared our results with YOLO-MS [8], a multi-spectral detection model with a double backbone. Xie *et al.* [8] evaluated YOLO-MS on the FLIR-aligned[7] and M3FD [26] datasets. Due to accessibility issues, we used only the M3FD dataset, which includes pedestrian detection labels for multiple classes like people, vehicles, and street lamps.

Our multi-spectral baseline model was trained on a merged LLVIP [27] and NII-CU dataset, improving generalization across diverse conditions. The merged dataset includes 21,368 image pairs, mainly containing small and medium-sized objects, aligning with our detection goals. We trained our baseline model on this dataset for 228 epochs. Table II shows its performance. After training on the M3FD dataset for 865 epochs, we compared results with YOLO-MS in Table III. The results showed that our model outperformed YOLO-MS by 4% in mAP50 and 10% in mAP50-95, showing significant improvements despite much lower complexity.

TABLE II: Results of our best performing model pretrained on the merged dataset.

| Model | Precision ↑ | Recall ↑ | mAP50 ↑ | mAP50-95 ↑ |
|---|---|---|---|---|
| MS Baseline | 0.955 | 0.927 | 0.969 | 0.656 |

TABLE III: Comparative results on M3FD dataset.

| Model | Precision ↑ | Recall ↑ | mAP50 ↑ | mAP50-95 ↑ |
|---|---|---|---|---|
| YOLO-MS [8] | - | - | 0.857 | 0.552 |
| **Ours** | 0.892 | 0.837 | **0.897** | **0.656** |

In Fig. 5, we show object detection on three random sample M3FD images, demonstrating our model's ability to detect various objects effectively.


(a) Ground Truth


(b) Predictions

Fig. 5: Sample images showing the multi-class object detection capabilities of our model on M3FD dataset.

[7]https://www.flir.com/oem/adas/adas-dataset-form

### F. Real-Time Performance

A key goal of this research is to maintain inference speed while improving model performance. To evaluate this, we measured the average inference speed on the NII-CU dataset [25], which includes 485 images, comparing the baseline model with modified versions.

TABLE IV: Results of the real-time study of the models on NII-CU dataset. ↑ and ↓ indicate higher or lower metric values for better performance.

| Model Configuration | Pre ↓ | Inference ↓ | Post ↓ | Total ↓ | FPS ↑ |
|---|---|---|---|---|---|
| Baseline Model | 0.6ms | 14ms | 2.3ms | 16.9ms | 59 |
| + 4CH | 0.5ms | 14.2ms | 2.2ms | 16.9ms | 59 |
| + 4CH + Bicubic | 0.6ms | 16.1ms | 2.6ms | 19.3ms | 52 |

Table IV shows that the proposed 4-channel input model maintains similar inference times to the baseline, with no significant impact on FPS (59). However, the bicubic up-sampling method adds 2.4ms to the total inference time, reducing FPS to 52, indicating the added complexity.

In summary, the 4-channel input modification does not affect real-time performance, but the upgraded up-sampling method introduces a slight trade-off between accuracy and speed, which may be a factor for real-time applications.

## V. Conclusion

In this research, we adapted the YOLOv8 architecture for real-time multispectral object detection, enabling $n$-channel inputs and an enhanced up-sampling strategy while preserving real-time performance and compatibility with newer YOLO versions.

Our experiments on RGB-thermal multispectral datasets showed a 22% improvement in `mAP500-95` over the baseline and a 10% improvement over YOLO-MS [8], without needing multiple backbones. Despite these improvements, the model maintained real-time inference at 52 FPS.

We also evaluated all YOLOv8 variants, finding smaller models sacrificed some accuracy (down to 64.7%) but significantly increased speed, with the small variant offering the best trade-off at 66.6% `mAP500-95` and 125 FPS.

## References


[1] Y. Benezeth, H. Laurent, B. Emile, and C. Rosenberger, "Towards a sensor for detecting human presence and characterizing activity," *Energy and Buildings*, vol. 43, no. 2, pp. 305–314, Mar. 2011, ISSN: 0378-7788. DOI: https://doi.org/10.1016/j.enbuild.2010.09.014.

[2] A. L. Adams *et al.*, "Search is a time-critical event: When search and rescue missions may become futile," *Wilderness & Environmental Medicine*, vol. 18, no. 2, pp. 95–101, Feb. 2007. DOI: 10.1580/06-WEME-OR-035R1.1.

[3] B. Mishra, D. Garg, P. Narang, and V. Mishra, "Drone-surveillance for search and rescue in natural disaster," *Computer Communications*, vol. 156, pp. 1–10, Mar. 2020. DOI: 10.1016/j.comcom.2020.03.012.

[4] S. Evans, *Autonomous drone navigation advances with brain-inspired system*, IoT World Today, Apr. 2023.

[5] I. Ignatov, O. Mosin, H. Niggli, C. Drossinakis, and G. Tyminski, "Methods for registering non-ionizing radiation emitted from the human body," *European Reviews of Chemical Research*, vol. 3, no. 1, pp. 4–24, 2015.

[6] U. Ozyurt, B. Cicekdag, Z. D. Budak, and S. Ertekin, "Enhanced thermal human detection with fast filtering for UAV images," in *2023 4th International Informatics and Software Engineering Conference (IISEC)*, 2023, pp. 1–7. DOI: 10.1109/IISEC59749.2023.10391031.

[7] K. Roszyk, M. Nowicki, and P. Skrzypczyński, "Adopting the yolov4 architecture for low-latency multispectral pedestrian detection in autonomous driving," *Sensors*, vol. 22, no. 3, p. 1082, Jan. 2022, ISSN: 1424-8220. DOI: 10.3390/s22031082.

[8] Y. Xie, L. Zhang, X. Yu, and W. Xie, "YOLO-MS: Multispectral object detection via feature interaction and self-attention guided fusion," *IEEE Transactions on Cognitive and Developmental Systems*, vol. 15, no. 4, pp. 2132–2143, 2023. DOI: 10.1109/TCDS.2023.3238181.

[9] X. Zou, T. Peng, and Y. Zhou, "UAV-based human detection with visible-thermal fused YOLOv5 network," *IEEE Transactions on Industrial Informatics*, vol. PP, pp. 1–10, Jan. 2023. DOI: 10.1109/TII.2023.3310792.

[10] C. Wankmüller, M. Kunovjanek, and S. Mayrgündter, "Drones in emergency response – evidence from cross-border, multi-disciplinary usability tests," *International Journal of Disaster Risk Reduction*, vol. 65, Nov. 2021, ISSN: 2212-4209. DOI: 10.1016/j.ijdrr.2021.102567.

[11] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016, pp. 779–788.

[12] H. Yu, Y. Tian, Q. Ye, and Y. Liu, "Spatial transform decoupling for oriented object detection," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 7, pp. 6782–6790, Mar. 2024. DOI: 10.1609/aaai.v38i7.28502.

[13] Y. Li, Q. Hou, Z. Zheng, M.-M. Cheng, J. Yang, and X. Li, "Large selective kernel network for remote sensing object detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, Oct. 2023, pp. 16 794–16 805.

[14] Y. Li, Q. Fan, H. Huang, Z. Han, and Q. Gu, "A modified YOLOv8 detection network for UAV aerial image recognition," *Drones*, vol. 7, no. 5, Apr. 2023, ISSN: 2504-446X. DOI: 10.3390/drones7050304.

[15] C.-Y. Wang, I.-H. Yeh, and H.-Y. M. Liao, *YOLOv9: Learning what you want to learn using programmable gradient information*, Feb. 2024. arXiv: 2402.13616 [cs.CV].

[16] T.-Y. Lin *et al.*, "Microsoft COCO: Common objects in context," in *Computer Vision – ECCV 2014*, D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds., Cham: Springer International Publishing, 2014, pp. 740–755, ISBN: 978-3-319-10602-1.

[17] A. Betti and M. Tucci, "YOLO-S: A lightweight and accurate yolo-like network for small target detection in aerial imagery," *Sensors*, vol. 23, no. 4, 2023, ISSN: 1424-8220. DOI: 10.3390/s23041865.

[18] Z. Liang, G. Cui, M. Xiong, X. Li, X. Jin, and T. Lin, "YOLO-C: An efficient and robust detection algorithm for mature long staple cotton targets with high-resolution rgb images," *Agronomy*, vol. 13, no. 8, 2023, ISSN: 2073-4395. DOI: 10.3390/agronomy13081988.

[19] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, *Scaled-YOLOv4: scaling cross stage partial network*, 2021. arXiv: 2011.08036 [cs.CV].

[20] A. Bochkovskiy, C. Wang, and H. M. Liao, "YOLOv4: Optimal speed and accuracy of object detection," *CoRR*, vol. abs/2004.10934, Apr. 2020. arXiv: 2004.10934. [Online]. Available: https://arxiv.org/abs/2004.10934.

[21] S. Tang, Y. Fang, and S. Zhang, *HIC-YOLOv5: Improved YOLOv5 for small object detection*, Sep. 2023. arXiv: 2309.16393 [cs.CV].

[22] R. Olivier and C. Hanqiang, "Nearest neighbor value interpolation," *International Journal of Advanced Computer Science and Applications*, vol. 3, no. 4, 2012. DOI: 10.14569/ijacsa.2012.030405.

[23] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 9, pp. 1904–1916, Jan. 2015. DOI: 10.1109/TPAMI.2015.2389824.

[24] R. Keys, "Cubic convolution interpolation for digital image processing," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 29, no. 6, pp. 1153–1160, Dec. 1981, ISSN: 0096-3518. DOI: 10.1109/TASSP.1981.1163711.

[25] S. Speth *et al.*, "Deep learning with RGB and thermal images onboard a drone for monitoring operations," *Journal of Field Robotics*, vol. 39, no. 6, pp. 840–868, May 2022. DOI: 10.1002/rob.22082.

[26] J. Liu *et al.*, "Target-aware dual adversarial learning and a multi-scenario multi-modality benchmark to fuse infrared and visible for object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2022, pp. 5802–5811.

[27] X. Jia, C. Zhu, M. Li, W. Tang, and W. Zhou, "LLVIP: A visible-infrared paired dataset for low-light vision," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, Oct. 2021, pp. 3496–3504.