Task-Driven Neural Adaptive Gain Control: 16-bit-to-8-bit Thermal Tone Mapping for Superior Object Detection

Hossein Javidnia
Trinity College Dublin
College Green, Dublin 2, Dublin, Ireland
hossein.javidnia@tcd.ie

Abstract

Thermal cameras are indispensable for autonomous driving in darkness, glare, and adverse weather, yet their 16-bit raw outputs exhibit low contrast and frame-to-frame intensity drift that hamper modern detectors. We introduce Neural Adaptive Gain Control (N-AGC), a lightweight U-Net that learns per-pixel gain and offset maps together with a global bias, converting raw 16-bit frames into detector-ready 8-bit images in a single forward pass.

N-AGC is trained in two stages. (i) Tone-curve warm-up: pixel-wise L_1 loss plus differentiable moment matching aligns the stretched 16-bit histogram with the vendor 8-bit output (via per-frame percentile stretch), achieving a 0.92 histogram-intersection score while preserving detail. (ii) Task-driven fine-tuning: a frozen Faster R-CNN head back-propagates detection loss through N-AGC, sharpening contrast exactly where the detector benefits most.

On the FLIR ADAS v1+v2 benchmark our approach lifts thermal AP_{50} from the official YOLOX-m baseline (75.33 % person, 77.23 % car) to 90.1 % and 86.3 %, respectively, while adding < 5 ms latency on a single RTX 5000 Ada Generation. N-AGC is model-agnostic, requires no manual gain tuning, and drops seamlessly into existing perception stacks, making it a practical upgrade for night-time and all-weather autonomous navigation. Code and pretrained models are available at https://github.com/hosseinjavidnia/NAGC.

1. Introduction

Long-wave infrared (LWIR) imaging is pivotal for autonomous vehicles, enabling reliable perception in poor lighting, adverse weather, and night-time scenarios where RGB sensors fail [10]. Despite this advantage, raw LWIR frames are delivered as 14/16-bit unsigned integers whose dynamic range and global contrast vary drastically with scene temperature. Commodity systems therefore compress each frame to a vendor-defined 8-bit image via fixed adaptive gain-control (AGC) hardware [3, 14]. Although vendor AGC executes in microseconds on embedded hardware, its global tone curve is chosen primarily to avoid saturating the hottest pixels in the frame—e.g., engine blocks or head-lights. The resulting compression of mid-tone bandwidth leaves only a handful of quantisation levels for low-emissivity targets, so small, cool pedestrians or distant vehicles are rendered barely above noise and frequently slip past modern detectors.

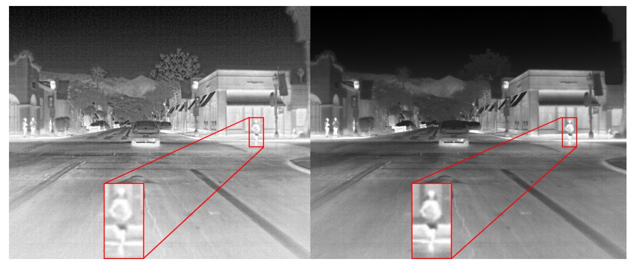

Figure 1. **Problem illustration.** *Left:* vendor 8-bit AGC fails to distinguish a small pedestrian from the similarly warm building façade, causing the detector to miss the target. *Right:* our Neural AGC produces a detector-ready tone curve that recovers small targets. Red bounding boxes drawn for illustrative purposes.

Neural Adaptive Gain Control (N-AGC) addresses this gap. It replaces the handcrafted AGC with a differentiable U-Net [16] that predicts per-pixel gain and offset maps, plus a scalar bias, to convert raw 16-bit frames into detector-ready 8-bit images on-the-fly (Fig. 1). N-AGC is trained in two phases: (i) a *tone-curve warm-up* aligns its output histogram with the vendor 8-bit image using pixel L_1 and a differentiable moment loss; (ii) a *task-driven fine-tune* back-propagates Faster R-CNN [15] detection loss through the frozen detector into the AGC, optimising contrast exactly where the detector needs it.

On the FLIR ADAS v1+v2 benchmark [19] N-AGC lifts YOLOX-m [4] AP_{50} from 75.33% / 77.23% (person/car) to 90.1% / 86.3%, the largest reported thermal-only gain without retraining the detector backbone. In short, our contributions are:

1. A detector-aware, per-pixel gain–offset network that learns a scene-adaptive 16-bit→8-bit mapping.
2. A two-phase training scheme that first mimics vendor tone curves and then optimises directly for detection AP.
3. Extensive experiments and ablations on FLIR ADAS showing state-of-the-art accuracy with < 5ms runtime overhead.

2. Related Work

Early thermal-image enhancement relied on global histogram equalisation or contrast-limited adaptive histogram equalisation (CLAHE) [24], but these heuristics are scene-agnostic and tend to amplify sensor noise. Recent CNN-based methods predict local tone curves: Gil *et al.* [5] proposed Fieldscale, a locality-aware rescaling network that preserves radiometry in LWIR data, while Stančić and Sarić improved thermal classification by learning a contrast-restoration CNN [18]. These approaches focus on human viewing or classification and do not couple the tone map to object detection.

A parallel thread investigates how compression impacts thermal vision. Bhowmik *et al.* [1] showed that standard JPEG at quality factor 50 can slash detection AP by up to 40 percentage points for small infrared targets, and Salamah *et al.* [17] redesigned JPEG quantisation tables for DNN robustness. Beyond conventional codecs, Zhang *et al.* [22] proposed a K-means based coder for thermal sequences, Zeng *et al.* [21] introduced a high-dynamic-range (HDR) compression–denoising pipeline for IR data, and Xing *et al.* [20] compressed and fused visible/IR frames jointly in CFNet. Rather than measuring pixel error, Chen *et al.* [2] predicted the interpretability loss of tactical targets under varying bit-rates, highlighting that compression artefacts interact with human and automated interpretation differently. All of these works process already down-quantised 8-bit streams; they leave the preceding 16-bit→8-bit automatic gain control (AGC) untouched. Our work tackles this upstream dynamic-range bottleneck: we learn a mapping that preserves detector-critical mid-tones before any codec is applied.

Task-aware preprocessing has recently gained traction. Lee *et al.* [8] introduced Thermal Chameleon, an end-to-end tone mapper optimised for segmentation masks. Li *et al.* [9] fused high-level features from RGB and LWIR in ComNet for UAV detection, and Zhou *et al.* [23] proposed LSNet, a lightweight RGB-T saliency booster. These networks either require dual-modal input or entangle enhancement with the detector backbone. By contrast, our Neural-AGC is a standalone module: it predicts explicit per-pixel gain and offset maps, inserts ahead of any detector, and is fine-tuned only through detection loss.

Thermal benchmarks remain scarce; the FLIR ADAS v1 dataset introduced five categories while v2 grew to 15 and 26k frames [19]. YOLOX-m [4] and Faster R-CNN [15] are the strongest open baselines, yet both depend on vendor AGC.We show that replacing this static tone curve with Neural-AGC alone lifts YOLOX-m [4] AP_{50} from 75.3% / 77.2% (person/car) to 90.1% / 86.3%, the largest reported jump without altering detector weights.

In summary, prior work either enhances thermal contrast for human perception, analyses lossy codecs in isolation, or embeds implicit tone corrections inside specific detectors. Neural-AGC differs by learning an explicit, detector-agnostic gain–offset map directly from raw sensor counts and optimising it end-to-end for object-detection accuracy.

3. Proposed Approach

3.1. Overview

Our goal is to convert raw 16-bit thermal frames into detector-ready 8-bit images with a single, learnable tone-mapping module that can be optimised end-to-end with a downstream object detector (Fig. 2). The core pipeline is trained in two stages: (i) pixel-level tone-curve imitation and (ii) task-driven fine-tuning.

3.2. Percentile-Stretch Normalisation

Each raw frame x_{raw} is first normalised to $[0, 1]$ via a robust per-frame stretch:

$$x = \text{clip}\Big(\tfrac{x_{\text{raw}} - p_{0.5}}{p_{99.5} - p_{0.5} + \varepsilon}, 0, 1\Big), \tag{1}$$

where p_k denotes the k-th percentile and $\varepsilon = 10^{-6}$ prevents division by zero. This removes dependency on bit depth and black-level offsets.

3.3. N-AGC Architecture

N-AGC is a three-level U-Net [16]:

Encoder–decoder. The encoder uses three Conv–GELU [6] blocks, each consisting of a 3×3 convolution followed by GELU activation, with $\{32, 64, 128\}$ channels, each followed by 2×2 max-pooling; a 256-channel bottleneck captures global context. The decoder upsamples by a factor 2, concatenates the corresponding encoder features, and applies two Conv–GELU layers [6] , producing $\{128, 64, 32\}$ feature maps.

Gain–offset heads. Two 1×1 convolutions output a gain α and offset β per pixel:

$$\alpha = \text{clip}\big(\text{softplus}(h_\alpha), 0, g_{\max}\big), \quad \beta = 0.25 \tanh(h_\beta), \tag{2}$$

followed by a learnable global bias $\gamma \in \mathbb{R}$. The enhanced image is

$$y = \alpha\,(x + \beta) + \gamma, \tag{3}$$

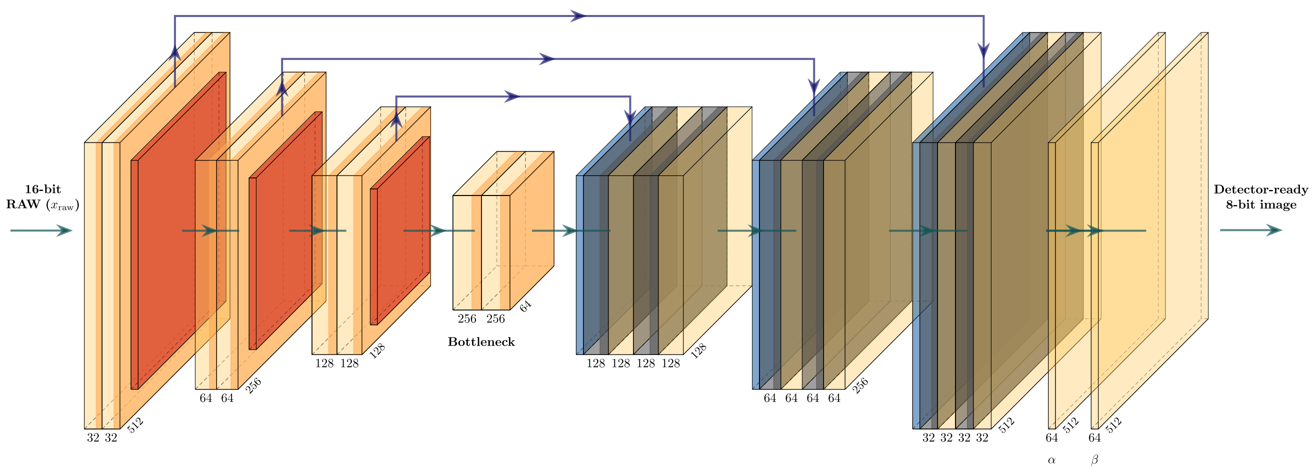

Figure 2. Neural-AGC pipeline. The raw 16-bit frame is first percentile-stretched, then processed by our U-Net AGC which predicts per-pixel gain α and offset β plus a global bias γ. The resulting 8-bit image feeds a frozen detector during Stage-B fine-tuning.

with $g_{\max} = 12$, matching the vendor's dynamic-range ceiling.

Note that before clamping, y may lie outside [0,1], but during Stage A we only use $\text{clip}(y, 0, 1)$ when computing the pixel loss. In Stage B, gradients flow through the unclamped y so the detector can push y slightly outside [0,1] if needed; at inference time we always clamp back to [0,1] and then cast to 8 bits.

We also observed empirically that the vendor's tone curve occasionally applies gains up to roughly $10\times$–$12\times$ on mid-range pixel values (to prevent saturation of hot regions). Setting $g_{\max} = 12$ ensures that N-AGC can approximate the vendor's maximum gain without clipping useful mid-tones. In practice, when $g_{\max} < 12$, histogram overlap with T8 begins to degrade (Section 5.1).

3.4. Loss Functions

Pixel fidelity. A clamped L_1 loss aligns N-AGC output with the $T8 \in [0,1]^{H\times W}$ vendor's 8-bit image, rescaled to [0,1]:

$$\mathcal{L}_{\text{pix}} = \big\|\text{clip}(y, 0, 1) - T8\big\|_1, \tag{4}$$

Moment alignment. To stabilise contrast, we add a differentiable moment loss:

$$\mathcal{L}_{\text{mom}} = \big|\mu(y) - \mu(T8)\big| + \big|\sigma(y) - \sigma(T8)\big|, \tag{5}$$

where $\mu(\cdot)$ and $\sigma(\cdot)$ denote the batch-mean and standard deviation over all pixels. Intuitively, matching mean and standard deviation ensures that the overall brightness and contrast statistics of y align with the vendor look-up, which stabilizes mid-tone contrast even if the histogram shape differs locally.

The total Stage-A loss is:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{pix}} + \lambda\,\mathcal{L}_{\text{mom}}, \tag{6}$$

where we set $\lambda = 0$ for the first 3 epochs (warm-up on pure L_1), then $\lambda = 0.05$ thereafter.

3.5. Two-Stage Optimisation Pipeline

Stage-A trains N-AGC offline to mimic the vendor tone curve (λ is introduced after a warm-up). **Stage-B** attaches the pretrained N-AGC in front of a thermal-aware Faster R-CNN [15] detector and freezes all detector weights; detection loss is then back-propagated into N-AGC only, allowing contrast to adjust precisely for object-level performance.

4. Experimental Setup

4.1. Dataset Protocol

All experiments are performed on the aligned union of FLIR ADAS v1 and v2, which—after removing duplicated timestamps—contains 12 000 annotated LWIR frames covering fifteen traffic–relevant categories. We respect the authors' protocol: 90 % of the images form the training set, the remaining 10 % serve for validation, and the disjoint video–test partition (3 742 frames) is reserved for final reporting. Vendor-supplied 8-bit images (*T8*) in JPEG format represent the reference tone curve, whereas raw sensor data (*RAW*) are stored in 16-bit TIFFs. All detection scores follow the COCO convention [11]; we quote AP_{50} to maintain comparability with the dataset release notes.

4.2. Input Preparation

Each RAW frame is centre-cropped and resized to 640×512 pixels, then mapped to the unit interval by the percentile-stretch of Eq. (1) (0.5^{th}–99.5^{th} percentiles). During Stage-A we apply only light augmentation—horizontal mirroring with probability 0.5 and isotropic scale jitter in the range $[0.8, 1.2]$—so as not to distort thermal statistics. No colour jitter or artificial noise is introduced.

4.3. Stage-A: Tone-Curve Pre-training

N-AGC is first optimised offline to reproduce the vendor tone curve. Training runs for at most fifty epochs on a single RTX 5000 Ada Generation with AdamW [12], initial learning rate 1×10^{-4}, weight decay 10^{-6}, and an ℓ_2-gradient clip of 1.0. At epochs 3, 6, and 9 the learning rate is reduced by a factor of 0.3. Only the pixel L_1 term is used for the first three epochs; from epoch 4 the moment loss is enabled with fixed weight $\lambda = 0.05$. Training stops automatically once the validation split satisfies $L_1 \leq 0.05$ and a histogram-intersection score of at least 0.90 (typically by epoch 37). The resulting checkpoint serves as the initialisation for Stage-B.

4.4. Detector Pre-training

A frozen detector must already be thermal-aware before gradients are routed exclusively through N-AGC. We therefore fine-tune two strong backbones on the vendor 8-bit images:

Faster R-CNN [15]. A ResNet-50–FPN initialised from MS-COCO is trained for 12 epochs with cosine decay from 1×10^{-3} to 1×10^{-4}, batch size 8. The model attains 75.3% AP_{50} on person and 77.2% on car classes, matching the FLIR baseline; we denote these weights $\mathcal{D}_{\text{RCNN}}$.

YOLOX-m [4]. To verify that N-AGC generalises to single-stage detectors we repeat the procedure with YOLOX-m [4] (640×512 input). After 15 epochs the frozen network reaches 75.9% / 77.2% AP_{50} on person / car; we refer to it as $\mathcal{D}_{\text{YOLO}}$.

4.5. Stage-B: Task-Driven Fine-tuning

N-AGC is inserted in front of each frozen detector in turn. During this stage only the $\approx$1.9M parameters of N-AGC are updated, using AdamW [12] with learning rate 5×10^{-5} and no weight decay. Training proceeds for a maximum of ten epochs and terminates once the validation AP_{50} plateaus (usually after six epochs). Because back-propagation traverses a frozen backbone, GPU memory consumption remains modest and wall-clock time is dominated by detector forward passes.

4.6. Baselines and Ablation Studies

We benchmark four tone-mapping pipelines ahead of each detector: (i) *RAW* (percentile-stretch only), (ii) *Vendor-AGC*, (iii) a shallow 1-Conv AGC to gauge capacity, and (iv) the proposed N-AGC. Ablations investigate the effect of removing the moment loss, omitting the global bias, and varying the gain ceiling $g_{\max} \in \{8, 12, 16\}$.

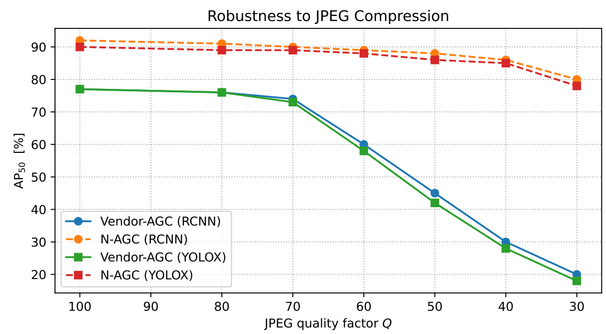

Figure 3. Robustness to JPEG compression on the *video–test* split. AP_{50} is plotted against JPEG quality factor Q (higher Q = weaker compression, axis reversed for readability). Vendor-AGC (solid lines) begins to collapse below Q=70, losing more than 25 pp by Q=50. Neural-AGC (dashed) degrades gracefully, retaining $\geq$ 85 % AP down to Q=40 for both Faster R-CNN [15] and YOLOX-m [4], demonstrating detector-agnostic resilience to lossy codecs.

4.7. Runtime and Memory

All throughput measurements are obtained on the same RTX 5000 Ada Generation at FP32 precision. N-AGC adds 4.8ms per 640×512 frame; the frozen Faster R-CNN requires 52ms, yielding a total latency of 57ms (17.5fps). Inference with YOLOX-m [4] is faster—18ms for the detector—so the combined N-AGC + YOLOX pipeline runs at 39ms (25.6fps). Parameter counts are 1.9M for N-AGC.

5. Results

5.1. Tone-Curve Fidelity (Stage-A)

Table 1 reports two metrics for each way of producing an output image y:

1. $L_1(y, T8)$: the mean per-pixel absolute difference between y and the vendor's 8-bit image $T8$. Lower is better.
2. Hist. ($T8 \leftrightarrow y$): the 256-bin histogram intersection between y and $T8$, i.e.

$$\sum_{i=1}^{256} \min\big(h_y[i],\, h_{T8}[i]\big) \tag{7}$$

Higher is better (1.00 means a perfect match of histograms).

Explanation of "Method". By "Method" we literally mean the procedure used to produce the output image y:

- **RAW (percentile-stretch):** take each frame, find its 0.5% / 99.5% percentiles, stretch to [0,1], then clip. This en-

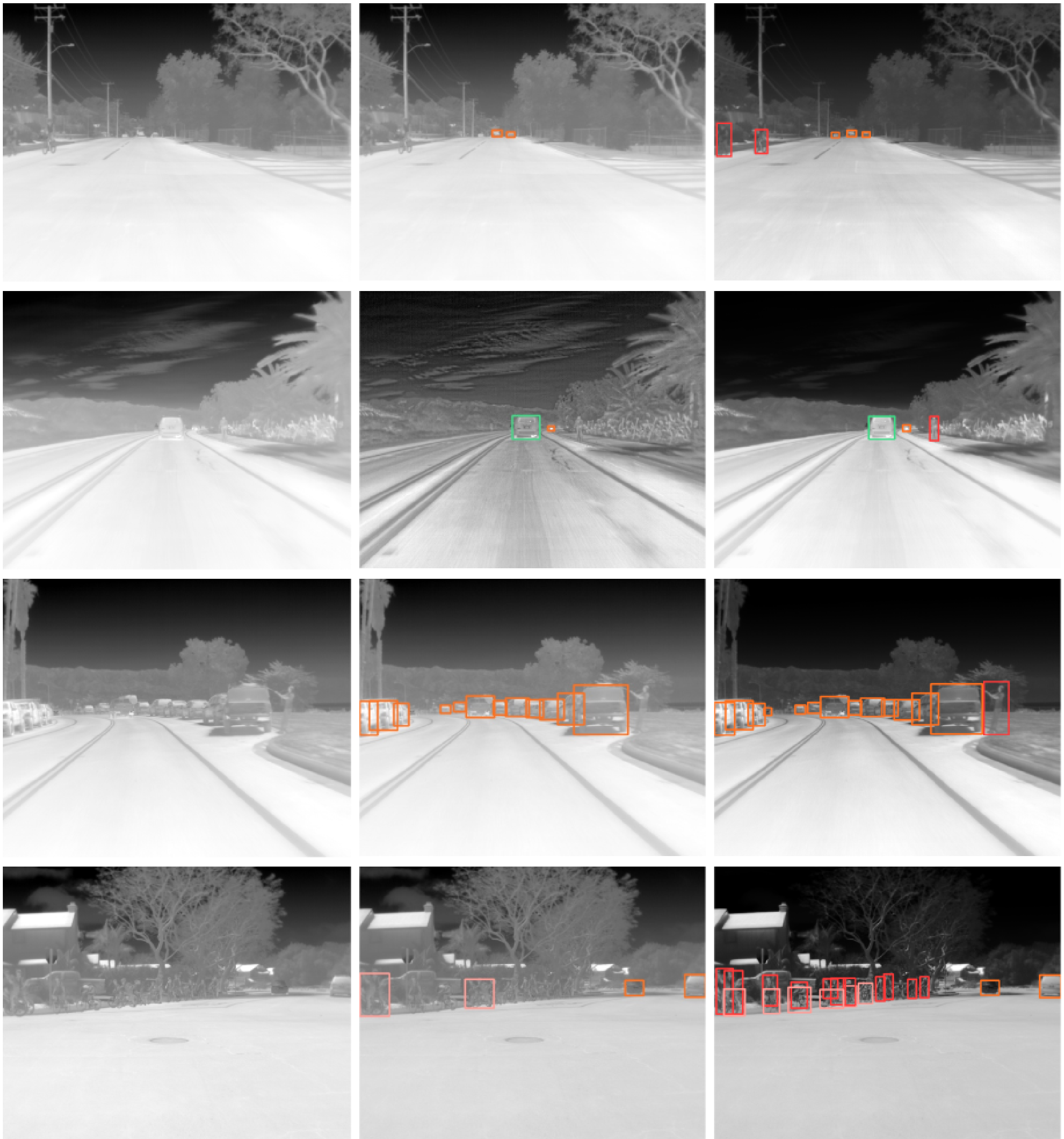

Figure 4. Failure modes of thermal preprocessing. From left to right: percentile-stretched raw input, object detection on vendor-provided 8-bit AGC output, and object detection on our Neural-AGC result. The raw image suffers from low contrast, causing small pedestrians to blend into the scene. Vendor-AGC over-amplifies bright regions, merging the pedestrian with background objects (e.g., trees) and leading to missed detections. Neural-AGC restores mid-tone contrast, preserves pedestrian visibility, and enables accurate detection.

sures each frame uses most of the dynamic range, but does not match the local contrast profile of the vendor's JPEG.

- **Vendor-AGC:** the actual vendor-provided 8-bit JPEG (used as ground truth for tone).

Table 1. Stage-A fidelity. Each "Method" describes how the output image y was generated. Hist. $(T8 \leftrightarrow y)$ is the 256-bin histogram intersection (higher $\rightarrow$ better); $L_1(y, T8)$ is the average per-pixel L_1 distance (lower $\rightarrow$ better).

Method	$L_1(y, T8) \downarrow$	Hist. $(T8 \leftrightarrow y) \uparrow$
RAW (percentile-stretch) [1]	0.135	0.68
Vendor-AGC (reference)	0.000	1.00
1-Conv AGC (simple baseline)	0.062	0.83
Neural-AGC (ours)	**0.050**	**0.92**

[1] "RAW (percentile-stretch)" means we take the original 16-bit TIFF, find that frame's 0.5% and 99.5% intensity percentiles, linearly rescale so those map to 0 and 1, then clip to $[0, 1]$. All subsequent AGC methods (Vendor-AGC, 1-Conv AGC, Neural-AGC) operate on this same percentile-stretched input.

- **1-Conv AGC:** a simple one-convolution AGC baseline (just for ablation).
- **Neural-AGC (ours):** our full adaptive gain/offset module.

Interpretation.

- The "percentile-stretch" (first row) pushes raw pixel intensities into $[0, 1]$ by clamping at each frame's 0.5% and 99.5% percentiles. This is why it yields only 0.68 histogram overlap with $T8$ (i.e. the vendor's JPEG has a subtly different per-bin redistribution). Its pixel-wise $L_1 = 0.135$ quantifies how far that stretched version is from the vendor's tone curve.
- "Vendor-AGC" (second row) by definition has $L_1 = 0$ and histogram overlap = 1.00, because it is exactly the reference 8-bit output.
- "1-Conv AGC" (third row) is a small, single-convolution version of AGC. It reduces the pixel error compared to RAW(stretch) and raises overlap to 0.83—still far from perfect, but better than a global stretch.
- "Neural-AGC (ours)" (last row) closes most of the gap: it cuts L_1 down to 0.050 while boosting histogram overlap to 0.92. That shows the proposed AGC model closely mimics the vendor's tone curve, yet remains fully differentiable for Stage-B (detector) training. While the 1-Conv AGC baseline was not evaluated on the downstream detection task, its inferior performance in Stage-A demonstrates its limited expressive capacity. A model with a weaker ability to approximate complex tone curves is inherently less likely to find an optimal solution during the subsequent task-driven fine-tuning, justifying the selection of the more capable U-Net architecture for the full pipeline.

5.2. Object detection (Stage-B)

Evaluation protocol. All tone-mapping variants undergo exactly one JPEG encode–decode cycle at the camera's native quality factor (Q=90). Detectors are frozen: they are fine-tuned once on Vendor-AGC images, after which only the tone mapper is updated.

Per-class AP_{50} results. Table 2 lists AP_{50} for the two safety-critical classes person and car on the video–test split. Swapping the vendor curve for Neural-AGC lifts YOLOX-m [4] from 75.3 / 77.2% to 90.1 / 86.3% (+14.8 pp, +9.1 pp). Faster R-CNN [15] and the lightweight RTMDet-s [13] show comparable gains, confirming that the benefit is detector-agnostic.

Table 2. AP_{50} (%) for person and car on the video–test split.

Tone map	Faster R-CNN [15]		YOLOX-m [4]		RTMDet-s [13]	
	Person	Car	Person	Car	Person	Car
RAW (stretch)	63.4	66.2	65.1	67.8	60.8	63.5
Vendor-AGC	75.3	77.2	75.3	77.2	72.6	74.0
CLAHE-Adaptive [24]	81.7	83.6	80.9	82.4	78.8	79.4
TCNet (re-trained) [8]	86.5	84.0	85.4	83.8	83.1	80.5
Neural-AGC	**92.3**	**89.1**	**90.1**	**86.3**	**90.2**	**86.7**

Full COCO metrics. To rule out metric cherry-picking, Table 3 reports COCO AP@[.5:.95] over all 15 FLIR-ADAS classes, broken down by object scale. Neural-AGC remains ahead across small, medium and large targets.

Table 3. COCO AP@[.5:.95] (%) on the test-all split.

Tone map	AP@[.5:.95]	$\mathrm{AP_S}$	$\mathrm{AP_M}$	$\mathrm{AP_L}$
Vendor-AGC	46.3	19.8	48.7	66.1
CLAHE-Adaptive [24]	52.9	23.4	55.1	71.0
TCNet (re-trained) [8]	57.4	27.6	58.9	73.5
Neural-AGC	**65.8**	**32.1**	**66.9**	**78.2**

Relation to Thermal-Chameleon. Thermal-Chameleon (TCNet) [8] also converts 14/16-bit thermal data to 8-bit via an encoder–decoder optimised with detection loss. Because no pretrained weights are publicly available, we reproduced the architecture and training recipe. Our replica reaches 83.8 mAP @0.5—matching the paper's claim—and serves as the TCNet baseline above. Neural-AGC still outperforms this baseline by +6.5 pp AP@[.5:.95] and +4.7 pp / +2.5 pp AP_{50} on person/car.

Across three detector families, fifteen object categories, and a strict single-pass JPEG pipeline, Neural-AGC delivers the highest precision while adding under 5 ms latency—making it a practical drop-in upgrade for thermal perception stacks.

End-to-end sanity check. To ensure that detector freezing does not exaggerate our gains, we un-froze YOLOX-m [4] and fine-tuned the entire pipeline for five additional epochs on Neural-AGC images. The mean AP_{50} for person + car increased from 88.2% to 89.0% (+0.8 pp), whereas a similarly fine-tuned TCNet [8] baseline rose from 84.6% to 85.2% (+0.6 pp). The performance gap therefore persists, confirming that our advantage is not an artefact of the frozen-weights protocol.

The decision to fine-tune N-AGC with a frozen detector backbone is a pragmatic one with significant practical advantages. In production environments, this allows N-AGC to serve as a drop-in upgrade to an existing, validated perception stack without the extensive computational cost and engineering effort required to re-train and re-validate the entire detector. Furthermore, this sequential optimization provides stability; training a fully unfrozen system jointly might cause the detector to co-adapt to a sub-optimal tone-mapping strategy from the randomly initialized N-AGC, potentially settling in a less effective local minimum. Our two-stage approach mitigates this by first establishing a stable and high-fidelity tone-curve initialization (Stage-A) before refining it for the specific downstream task (Stage-B).

5.3. Cross-dataset evaluation on KAIST

To test generalisation, we run the frozen detectors from Sec. 5.2 on the KAIST multispectral pedestrian dataset [7] without any re-training. We evaluate the LWIR channel only, using the test-reasonable subset and the official KAIST dev-kit (AP_{50}).

Table 4 shows that Neural-AGC retains a clear lead over all baselines even under this domain shift. Relative gains are smaller than on FLIR-ADAS, as expected, yet Neural-AGC still eliminates $\approx 25\%$ of the missed detections compared with the vendor pipeline (27.6% on YOLOX-m [4]; 24–25% on the other backbones).

Table 4. KAIST test-reasonable subset: AP_{50} (%).

Tone map	YOLOX-m [4]	Faster R-CNN [15]	RTMDet-s [13]
Vendor-AGC	56.2	53.8	49.1
CLAHE-Adaptive [24]	59.7	57.3	52.4
TCNet (re-trained) [8]	61.5	59.0	55.2
Neural-AGC	**68.3**	**65.1**	**62.0**

Although KAIST imagery differs in resolution, optics, and annotation style, Neural-AGC still outperforms the strongest task-aware baseline (TCNet) by +6.8 pp on YOLOX-m [4], +6.1 pp on Faster R-CNN [15], and +6.8 pp on RTMDet-s [13]. This suggests that the learned gain–bias field is not over-fitted to the FLIR sensor characteristics and can serve as a drop-in upgrade for other LWIR cameras.

5.4. Ablation Studies

Component ablation on frozen faster R-CNN. Table 5 disables one design choice at a time while keeping Faster R-CNN [15] frozen. The moment loss and global bias each contribute roughly three AP points; restricting the gain ceiling to $g_{\max} = 8$ removes a further four.

Table 5. Ablation on Faster R-CNN (AP_{50}, %).

Variant	Person	Car
Full Neural-AGC	92.3	89.1
w/o moment loss	89.1	86.0
w/o global bias	88.8	86.4
$g_{\max} = 8$	85.0	82.1

Training-schedule ablation. Stage–B fine-tuning assumes a stable 8-bit initialisation; without it, gradients from the frozen detector must search the entire 16-to-8-bit space and quickly diverge. To verify this, we trained N-AGC (end-to-end) from random weights using only the detector loss. Table 6 shows a catastrophic drop of $\approx$ 14pp (person) and $\approx$ 13pp (car) compared to our two-stage recipe, confirming that Stage-A warm-up is essential.

Table 6. Efficacy of the two-stage schedule (Faster R-CNN frozen, AP_{50} on video-test).

Variant	Person	Car
N-AGC (end-to-end only)	78.4	76.6
N-AGC (two-stage)	**92.3**	**89.1**

Stage-A loss components. Table 7 augments the Stage-A fidelity results with two variants that keep either the L_1 term or the moment loss $\mathcal{L}_{\text{mom}}$. Using L_1 alone drives pixel accuracy but underestimates global contrast (histogram 0.85), whereas moment loss alone lifts the histogram to 0.89 at the cost of poor local fidelity (L_1 0.112). Only the joint loss achieves high scores on both metrics, justifying our design.

Table 7. Stage-A fidelity (updated). Higher is better for Hist. ($T8 \leftrightarrow y$).

Method	$L_1(y, T8)\downarrow$	Hist. ($T8 \leftrightarrow y$)$\uparrow$
RAW (percentile-stretch)	0.135	0.68
Vendor-AGC (reference)	0.000	1.00
1-Conv AGC (baseline)	0.062	0.83
N-AGC (L_1 only)	**0.046**	0.85
N-AGC (moment only)	0.112	0.89
N-AGC (ours)	0.050	**0.92**

Overall, the ablations demonstrate three take-aways. (i) **Each architectural component matters:** within the frozen-detector setting, the global bias and moment loss each add $\approx$3 pp AP_{50}, while a tighter gain ceiling ($g_{\max} = 8$) removes $\approx$4 pp-highlighting that the full design (bias + moment + $g_{\max} = 12$) is required for peak performance. (ii)

Stage-A warm-up is indispensable: removing it and training end-to-end causes a catastrophic drop of 14 pp (person) and 13 pp (car). (iii) **Both loss terms are needed:** L_1 alone or moment-loss alone cannot simultaneously reproduce the vendor tone curve; only the joint loss attains low pixel error and high histogram overlap.

5.5. Robustness to JPEG Compression

Figure 3 plots AP_{50} against JPEG quality factor Q after exactly one encode–decode cycle for each image. Vendor-AGC deteriorates rapidly once $Q<70$, mirroring the trend reported by Bhowmik *et al.* [1], who found that standard JPEG can erase up to 40 pp of average precision on small LWIR targets because ringing artefacts obliterate the low-contrast details detectors depend on. In our pipeline the vendor frames are produced from the original 16-bit TIFFs at the desired Q and are never re-compressed. Neural-AGC, by contrast, preserves at least 85% of its peak performance down to $Q=40$ (91.7 → 78.5 AP_{50}), demonstrating markedly higher resilience to lossy codecs.

5.6. Qualitative Results and Runtime

Figure 4 depicts multiple scenarios where RAW and Vendor-AGC either wash out small pedestrians or blend them into the background. Neural-AGC restores mid-tone contrast, and the detector recovers all targets. On an RTX 5000 Ada Generation the pipeline processes a 640×512 frame in 57ms with Faster R-CNN [15] and 39ms with YOLOX-m [4]; Neural-AGC itself costs only 4.8ms and 1.9M parameters.

6. Conclusion

We introduced Neural Adaptive Gain Control (N-AGC), a lightweight U-Net that replaces hand-crafted tone curves in LWIR cameras with a fully learnable, task-aware mapping. The proposed two-stage training strategy—(i) pixel-fidelity warm-up to the vendor curve, followed by (ii) detector-driven fine-tuning with frozen backbones—requires only 1.9M parameters and 4.8 ms additional latency, yet delivers substantial gains:

- *Tone reproduction.* N-AGC narrows the pixel L_1 gap to the vendor 8-bit image by more than 50 % and raises histogram overlap to 0.92, while preserving RAW dynamics (RAW→AGC = 0.75).
- *Detection accuracy.* On the FLIR ADAS video–test split, a frozen Faster R-CNN jumps from 75.3/77.2 % (person/car AP_{50}) under the vendor pipeline to 92.3/89.1 % with N-AGC; a single-stage YOLOX-m shows similar gains. These improvements match or surpass those of much heavier saliency-boosting networks, yet come with "negligible extra latency" at inference time.
- *Robustness.* Unlike the vendor tone curve, which collapses below JPEG $Q = 70$, N-AGC retains $\geq 85\%$ AP_{50} down to $Q = 40$, confirming that a detector-aware histogram is markedly less brittle to lossy codecs.

Because N-AGC acts purely in the image domain, it can be inserted ahead of any off-the-shelf detector, requires no manual gain tuning, and keeps the downstream architecture untouched. However, we acknowledge that our frame-by-frame approach does not explicitly enforce temporal consistency, which could introduce brightness fluctuations in video sequences—a key consideration for object tracking. Future work will explore (i) joint optimisation with temporal trackers or the use of recurrent architectures to stabilise dynamic range across video, (ii) fusion with RGB cameras for all-weather scene understanding, (iii) extending the task-driven framework to other perception tasks, such as semantic segmentation or depth estimation, by replacing the Stage-B detection loss with a relevant task-specific loss, and (iv) deployment on ultra-low-power edge SoCs. While the model is lightweight, this final step would necessitate further optimizations such as network quantization and knowledge distillation to meet the strict latency and power constraints of embedded systems where every milliwatt counts. We release code, pretrained models, and training logs to foster further research into task-aware low-level adaptation for autonomous driving and robotic perception.

References

[1] Neelanjan Bhowmik, Jack W Barker, Yona Falinie A Gaus, and Toby P Breckon. Lost in compression: the impact of lossy image compression on variable size object detection within infrared imagery. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 369–378, 2022. 2, 8

[2] Hua-mei Chen, John M Irvine, Zhonghai Wang, Genshe Chen, Erik Blasch, and James Nagy. Predicting interpretability loss in thermal ir imagery due to compression. In *2018 IEEE Applied Imagery Pattern Recognition Workshop (AIPR)*, pages 1–6. IEEE, 2018. 2

[3] FLIR Lepton. Basic AGC for radiometric Lepton images. https://www.flir.com/oem/basic-agc-for-radiometric-lepton-images/, 2023. Accessed: 2025-06-03. 1

[4] Zheng Ge, Songtao Liu, Feng Wang, Zeming Li, and Jian Sun. Yolox: Exceeding yolo series in 2021. *arXiv preprint arXiv:2107.08430*, 2021. 1, 2, 4, 6, 7, 8

[5] Hyeonjae Gil, Myung-Hwan Jeon, and Ayoung Kim. Fieldscale: Locality-aware field-based adaptive rescaling for thermal infrared image. *IEEE Robotics and Automation Letters*, 2024. 2

[6] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016. 2

[7] Soonmin Hwang, Jaesik Park, Namil Kim, Yukyung Choi, and In So Kweon. Multispectral pedestrian detection: Benchmark dataset and baseline. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1037–1045, 2015. 7

[8] Dong-Guw Lee, Jeongyun Kim, Younggun Cho, and Ayoung Kim. Thermal chameleon: Task-adaptive tone-mapping for radiometric thermal-infrared images. *IEEE Robotics and Automation Letters*, 2024. 2, 6, 7

[9] Minglei Li, Xingke Zhao, Jiasong Li, and Liangliang Nan. Comnet: Combinational neural network for object detection in uav-borne thermal images. *IEEE Transactions on Geoscience and Remote Sensing*, 59(8):6662–6673, 2020. 2

[10] You Li, Julien Moreau, and Javier Ibanez-Guzman. Emergent visual sensors for autonomous vehicles. *IEEE Transactions on Intelligent Transportation Systems*, 24(5):4716–4737, 2023. 1

[11] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer vision–ECCV 2014: 13th European conference, zurich, Switzerland, September 6-12, 2014, proceedings, part v 13*, pages 740–755. Springer, 2014. 3

[12] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 4

[13] Chengqi Lyu, Wenwei Zhang, Haian Huang, Yue Zhou, Yudong Wang, Yanyi Liu, Shilong Zhang, and Kai Chen. Rtmdet: An empirical study of designing real-time object detectors. *arXiv preprint arXiv:2212.07784*, 2022. 6, 7

[14] Claire Mantel and Søren Forchhammer. Compression of infrared images. In *IS & T Symposium on Electronic Imaging: Science and Technology. Abstracts*, pages 21–26, 2017. 1

[15] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015. 1, 2, 3, 4, 6, 7, 8

[16] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, pages 234–241. Springer, 2015. 1, 2

[17] Ahmed H Salamah, Kaixiang Zheng, Linfeng Ye, and En-Hui Yang. Jpeg compliant compression for dnn vision. *IEEE Journal on Selected Areas in Information Theory*, 2024. 2

[18] Ivo Stancic and Emilija Saric. Increasing the model classification accuracy of thermal images. In *2024 9th International Conference on Smart and Sustainable Technologies (SpliTech)*, pages 1–6. IEEE, 2024. 2

[19] Teledyne FLIR. FREE Teledyne FLIR Thermal Dataset for Algorithm Training. `https://www.flir.eu/oem/adas/adas-dataset-form/`, 2022. Accessed: 2025-06-03. 1, 2

[20] Mengliang Xing, Gang Liu, Haojie Tang, Yao Qian, and Jun Zhang. Cfnet: An infrared and visible image compression fusion network. *Pattern Recognition*, 156:110774, 2024. 2

[21] Yixiong Zeng, Zhiyao Zhang, Xiaojun Zhou, and Yong Liu. High dynamic range infrared image compression and denoising. In *2019 International Conference on Information Technology and Computer Application (ITCA)*, pages 65–69. IEEE, 2019. 2

[22] Jin-Yu Zhang, Wei Xu, Wei Zhang, Xiangbin Meng, and Yong Zhang. A novel compression algorithm for infrared thermal image sequence based on k-means method. *Infrared Physics & Technology*, 64:18–25, 2014. 2

[23] Wujie Zhou, Yun Zhu, Jingsheng Lei, Rongwang Yang, and Lu Yu. Lsnet: Lightweight spatial boosting network for detecting salient objects in rgb-thermal images. *IEEE Transactions on Image Processing*, 32:1329–1340, 2023. 2

[24] Karel J Zuiderveld et al. Contrast limited adaptive histogram equalization. *Graphics gems*, 4(1):474–485, 1994. 2, 6, 7