Lost in Compression: the Impact of Lossy Image Compression on Variable Size Object Detection within Infrared Imagery

Neelanjan Bhowmik[1], Jack W. Barker[1], Yona Falinie A. Gaus[1], Toby P. Breckon[1,2]

Department of {Computer Science[1], Engineering[2]}, Durham University, Durham, UK

Abstract

Lossy image compression strategies allow for more efficient storage and transmission of data by encoding data to a reduced form. This is essential enable training with larger datasets on less storage-equipped environments. However, such compression can cause severe decline in performance of deep Convolution Neural Network (CNN) architectures even when mild compression is applied and the resulting compressed imagery is visually identical. In this work, we apply the lossy JPEG compression method with six discrete levels of increasing compression {95, 75, 50, 15, 10, 5} to infrared band (thermal) imagery. Our study quantitatively evaluates the affect that increasing levels of lossy compression has upon the performance of characteristically diverse object detection architectures (Cascade-RCNN, FSAF and Deformable DETR) with respect to varying sizes of objects present in the dataset. When training and evaluating on uncompressed data as a baseline, we achieve maximal mean Average Precision (mAP) of 0.823 with Cascade R-CNN across the FLIR dataset, outperforming prior work. The impact of the lossy compression is more extreme at higher compression levels (15, 10, 5) across all three CNN architectures. However, re-training models on lossy compressed imagery notably ameliorated performances for all three CNN models with an average increment of ∼ 76% (at higher compression level 5). Additionally, we demonstrate the relative sensitivity of differing object areas {tiny, small, medium, large} with respect to the compression level. We show that tiny and small objects are more sensitive to compression than medium and large objects. Overall, Cascade R-CNN attains the maximal mAP across most of the object area categories.

1. Introduction

The use of infrared-band (thermal) camera imagery (Figure 1) within the task of visual surveillance is well established with applications within target detection, visual tracking, behaviour analytics, home monitoring, and automotive environment perception [3, 24–26, 29, 44, 46]. The synergistic nature of combining both visible-band and thermal-
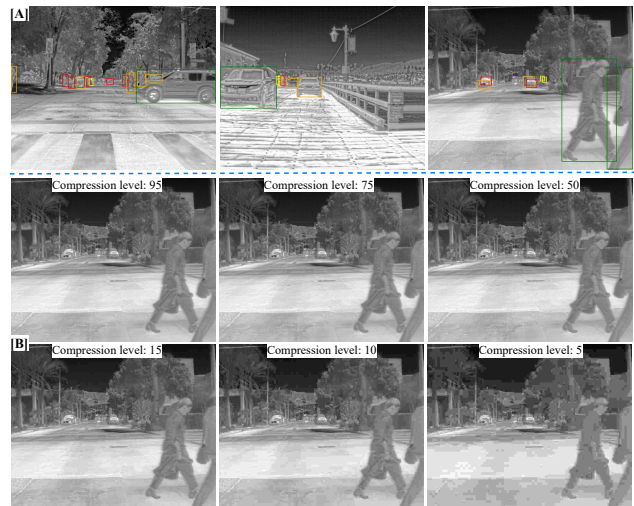

Figure 1. Exemplar infrared imagery from FLIR [16] containing different classes with varying object areas in bounding boxes: *tiny*, *small*, *medium*, *large* [A]. Example imagery at different lossy compression levels [B].

band imagery is advantageous when applied to a range of computer vision applications [30, 34]. Valuable visual cues can be provided in the infrared domain by thermal sensors which visible-band sensors will fail to capture [22, 35]. A major advantage of thermal imagery is that it is rarely influenced by surrounding lighting changes and shadows, meaning that objects-of-interest can be readily distinguished in the dark, fog and other complex environments as opposed to the scene illumination requirements of visible-band imagery.

The recent rise of Convolutional Neural Networks (CNN) [23] have revolutionised visual tasks significantly advancing the state-of-the-art in many applications. Within object detection specifically, most efforts have focused on detecting objects-of-interest in standard colour imagery by using multi-stage [4, 20, 42], singe-stage [31, 41, 53] and transformer-based [7, 55] detectors. These aforementioned object detection based CNN methods rely heavily on architectures that have been trained on large-scale colour imagery datasets such as ImageNet [11] PASCAL Visual Object Classes (PASCAL-VOC) [15] or Microsoft Common

Objects in Context (MS-COCO) [32]. Introducing CNN to object detection within thermal imagery is significantly hindered by the absence of such annotated datasets of the same scale and variety. In order to combat this challenge, methods such as transfer learning [17], generation of pseudo-RGB equivalents [12], and domain adaptation [36] are used to benefit an existing CNN model and raise an equivalent level of CNN success for thermal band imagery. The work of [17] focuses on thermal object detection by employing a transfer learning approach. In this approach, the knowledge obtained from the visible spectrum is transferred to the thermal domain for object detection in the thermal domain. The work of [12] implements image-to-image translation frameworks to generate pseudo-RGB equivalents of a given thermal image and then use a CNN architecture for object detection in the thermal image. Recent work of [36] proposes a self-supervised domain adaptation via an encoder-decoder transformer network to develop a robust thermal image object detector in autonomous driving.

Most of the solutions stated above have made a step towards real-time processing which is vital to the application of autonomous driving. Autonomous vehicles and other such agents can utilise thermal-band sensors to provide a necessary solution to accurately perceive the environment even in differing light conditions, or night time. However, the real-time transmission of high-resolution visible-band and thermal-band data collected by the agent during inference can require substantial transmission overhead to process in real-time, and large storage requirements for storing such data. Every fraction of a second of reaction time counts for autonomous vehicles dealing with unexpected obstructions in the path of motion, so reducing the transmission overhead is vital for improved safety. The use of commonplace lossy compression techniques, such as JPEG [49] and MPEG [27] tackles such transmission and storage overheads by compressing the data to a reduced form; a downside of which is potential image quality degradation (Figure 1, B) which can affect the performance of computer vision models. Work by [14] evaluates the affect of five types of quality distortions and compression (blur, noise, contrast, JPEG and JPEG2000) using CNN architectures when classifying a subset of the ImageNet dataset [11]. More recent work by [38] thoroughly evaluates the impact of JPEG and H.264 lossy compression on CNN architectures. They evaluate Faster R-CNN [42] upon the Pascal VOC dataset [15] and find similarly to [14] that performance degrades rapidly at high lossy compression levels.

The prior work on image compression stated above largely focuses on the use of compression applied to visible-domain imagery. Relatively few studies investigate the impact upon CNN task performance with respect to differing levels of compression applied to infrared-band (thermal) at inference (deployment) time. In this work we make the following contributions:

- we investigate three diverse end-to-end CNN object detection architectures, which differ in operation, i.e., multi-stage (Cascade R-CNN [4]), single-stage (FSAF [53]) and transformer-based (Deformable DETR [55]) over thermal imagery.
- as thermal imagery exhibits inherently different properties from standard visible-band imagery, we examine the impact that lossy image compression at differing levels has upon CNN-based thermal object detection.
- furthermore, we thoroughly examine the impact of lossy image compression with respect to in-image object size definition, (*tiny, small, medium, large*) and determine within which domains compression is most impactful upon performance and hence where image quality is most pertinent to deployed object detection model performance.

2. Related Work

Object detection and classification in thermal imagery is an active area of research [28, 37, 43]. A range of trial works in the literature address the task of detecting people and objects in thermal imagery [12, 17, 22]. There has been a significant amount of work on classifying and detecting people and objects in thermal imagery using standard computer vision and machine learning models. An early method such as a template-based approach, where Bertozzi et al, [2] implements probabilistic human shape templates whilst Davis and Keck [10] uses generalized person templates derived from contour saliency maps for pedestrian detection under thermal imagery. In [29], features decomposed by the wavelet transform from the high brightness property of the pedestrian pixels under thermal imagery were used as an input to a support vector machine (SVM) classifier.

With the increasing popularity of deep CNN architectures, several methods have been proposed for applying deep learning methods to thermal imagery [12, 17, 28, 37, 43]. In most cases, this research was carried out in the field of autonomous driving, where accurate detection of pedestrians and vehicles is vital. A transfer learning approach with the YOLO [40] architecture has been carried out by Abbot et al. [1], in which high-resolution thermal imagery is used for training and low-resolution thermal is used for evaluation purpose (to classify pedestrians and vehicles). Devaguptapu et al. [12] address the data scarcity problem in thermal imagery by utilising image-to-image translation frameworks [54] to generate pseudo-RGB equivalents of given thermal imagery, then employing Faster-RCNN [42] for detecting object-of-interest. Faster-RCNN [42] has also been trained with thermal imagery under a super-resolution method to deal with the issue of a small number of pixels which targets at the long-range have [51]. Chao et al. [5] take another approach by proposing the one-stage detector ThermalDet which utilises all the features in different levels

of the feature pyramid extracted by the backbone network, resulting in higher detection accuracy than baseline (Faster R-CNN) [42].

Some works propose the use of thermal imagery as a complement of colour imagery by fusing both domains, achieving superior results. The work of Liu et al. [33] integrates features from both colour and thermal at different stages, resulting in better detection accuracy when compared to the baseline (Faster R-CNN). A fusion method was also explored via multi-layer fusion RPN. This was used for integrating features in different branches, significantly reducing the detection miss rate in several challenging datasets [9]. Later work such as [45] fuses both visible and thermal imagery via a fusion-based network for the semantic segmentation of urban scenes. Namely the results from RGB-Thermal Fusion Network (RTFNet) demonstrate the superiority of the such an approach, even in challenging lighting conditions. Similar to RTFNet, Multi-spectral Fusion Networks (MFNet) architecture [19] which fuses visible and thermal imagery resulting in similar or higher accuracy than state-of-the-art segmentation methods such as SegNet.

While these efforts have shown good detection accuracy performance under thermal imagery, implementing deep CNN architectures requires heavy memory usage, substantial storage and transmission infrastructure. There has been substantial investigation of efficient storage and reducing memory utilisation [14, 39, 50] but very few works have examined the performance of these techniques applied to thermal imagery. These limited studies open the door only slightly on the question - *what is the generalised impact of lossy compression across a diverse set of deep neural network object detection architectures with respect to varying object size within infrared-band (thermal) imagery?*

3. Proposed Approach

We outline the approach of this paper in the following sections. Our method utilises the JPEG compression method (Section 3.2) across infrared-band (thermal) image data. We iterate through six discrete levels of compression and compare the effect that each level has on the performance of object detection architectures (Section 3.1).

3.1. Object Detection Architectures

In this study we utilise three state-of-the-art, well-established and characteristically diverse object detection architectures as outlined in Table 1. These are namely: Cascade R-CNN [4], Feature Selective Anchor-Free [53], and Deformable End-to-End Detection with Transformers [55]. All architectures in this study utilise a ResNet-50 [21] backbone and are initialised with weights trained on COCO [32].
Cascade R-CNN (CR-CNN) [4]: Cascade Region-based Convolutional Neural Network (Cascade R-CNN) is an extension of R-CNN [18] which offers a solution to the

Table 1. Summary of object detection architectures.

Architecture	Key Features
Cascade R-CNN [4]	two-stage, anchor-based
FSAF [53]	single-stage, anchor-free
Deformable DETR [55]	transformer-based single-stage, anchor-free

trade-off between low Intersection over Union (IoU) thresholds inducing noisy detections and performance degradation with high IoU thresholds. It accomplishes this by training a sequence of detectors stage-by-stage with increasing IoU thresholds to be more selective against false positives.
FSAF [53]: Feature Selective Anchor-Free (FSAF) is a module for single-shot object detection can be added to detectors with a feature pyramid structure and performs online feature selection upon multi-level anchor-free branches. This aims to address the limitations induced by heuristic-guided feature selection and overlap-based anchor sampling, leading to increased performance while introducing negligible inference overhead.
Deformable DETR (DDETR) [55]: Detection Transformer (DETR) combines convolutional features with a transformer architecture [47] which powerfully model sequential relations using multi-head attention. This means that the DETR architecture does not rely on hand-crafted components and as such can be trained fully end-to-end. Deformable DETR is an extension to this architecture which speeds up the convergence by having attention modules only attend to a small set of neighbouring points as well as tackling the problem of representing objects at varying scale.

3.2. Lossy Image Compression

For this work we use the lossy JPEG [48] compression algorithm which is based on the discrete cosine transform (DCT). The first step of JPEG is colour conversion where the RGB data of the image is converted using the respective components of luminance (Y), blue projection (U) and red projection (V). The resulting YUV image is then split into 8×8 pixel blocks to be processed into the frequency domain by the Discrete Cosine Transform (DCT). The next step of quantisation is where the reduction of information required to store the image takes place in which the resultant DCT coefficient matrix is divided and rounded by the quantisation matrix leading to a reduced form that provides resolution amount with respect to how perceivable a given image part is.

The quantisation is where the image quality of the resulting compressed representation can be controlled by the compression level parameter. If the value of this parameter is 1, maximum compression will produce the lowest quality of the data but will yield the smallest file size required to store the data. Conversely, a value of 100 offers the least effective compression which will negligibly affect the quality

of the resultant image, producing a visually lossless image and as such the file size will remain close to the original image prior to compression.

3.3. Object Area Definition

The object detection dataset comprises of several object classes with varying bounding box area size. In this work, we further analyse each object class based on the object area size. We adhere to the COCO [32] benchmark definitions of three discrete categories of the object areas: *small, medium, large*. Additionally, we introduce a new category, *tiny*, where the object area is less than or equal to 20^2 pixels. The object areas (Figure 1, A) used in this work are outlined in Table 2. In this work, we incorporate the thorough study of the impact of the lossy image compression on the four defined object areas (Table 2), and outline the detection performance of CNN architectures (Section 3.1).

Table 2. Object area definition.

Object	Area
tiny	area $\leq 20^2$
small	area $\leq 32^2$
medium	$32^2 <$ area $\leq 96^2$
large	$96^2 <$ area

4. Experimental Setup

This section presents the dataset used, experimental strategy and the implementation details of our experiments.

4.1. Dataset

The experimental setup comprises of following dataset. **FLIR [16]**: The *FLIR* dataset provides annotated single channel grayscale infrared imagery (Figure 1) of multiple object classes, which are captured under clear-sky conditions during both day (60%) and night (40%). Thermal images are acquired with a FLIR Tau2 camera (Long Wave Infrared Cameras - LWIR) with image resolution of (640×512). In this work, we use the default training and testing split provided in the dataset. We consider primarily three classes, {*Person, Bicycle, Car*}, for our object detection task. The training and testing set consist of $7,859$ and $1,360$ images respectively. The details of the FLIR dataset statistics are presented in Table 3. The object area-wise (Section 3.3) statistics of each class is illustrated in Table 3.

Compressed datasets. To determine how much lossy JPEG compression [48] is achievable within CNN object detection architectures on infrared imagery, the original uncompressed *FLIR* [16] dataset is compressed at six different levels {*95, 75, 50, 15, 10, 5*} to create the compressed versions of the dataset (*FLIR_c*), as depicted in (Figure 1, B).

4.2. Experimental Protocol

Our experiments consist of the following settings:

Table 3. FLIR [16] dataset statistics: class-wise number of instances, % of {Train, Test} set.

	Person	Bicycle	Car
Train-set	22,372	3,986	41,260
Obj_{tiny}	8923, 13.15%	993, 1.46%	10832, 15.97%
Obj_{small}	16304, 24.03%	2419, 3.57%	20665, 30.46%
Obj_{med}	5590, 8.24%	1505, 2.22%	17056, 25.14%
Obj_{large}	478, 0.7%	62, 0.09%	3539, 5.22%
Test-set	5,779	471	5,432
Obj_{tiny}	765, 6.54%	55, 0.47%	807, 6.9%
Obj_{small}	2885, 24.67%	219, 1.87%	2248, 19.22%
Obj_{med}	2575, 22.02%	230, 1.97%	2465, 21.08%
Obj_{large}	319, 2.73%	22, 0.19%	719, 6.15%
	Train		Test
Images	7,859		1,360

- Firstly, the three CNN architectures (Section 3.1) are trained and evaluated on uncompressed imagery, (*FLIR ⇒ FLIR*), for benchmark purpose.
- Secondly, the CNN architectures are trained on uncompressed imagery and evaluated on the compressed imagery, (*FLIR ⇒ FLIR_c*).
- Finally, each CNN architecture is re-trained with compressed imagery at each of the six lossy compression levels to determine whether resilience to compression could be improved, and how much compression can be achieved before a significant impact on object detection performance is observed (*FLIR_c ⇒ FLIR_c*).

4.3. Implementation Details

The CNN architectures (Section 3.1) are implemented using the MMDetection framework [8]. All experiments are initialised with weights pretrained on the COCO dataset [32]. The CNN architectures (Section 3.1) are trained using a ResNet$_{50}$ [21] backbone with the following training configuration:

- Cascade R-CNN [4] and FSAF [53]: backpropagation optimisation is performed via Stochastic Gradient Descent (SGD), with initial learning rates of $1e - 2$, trained for 20 epochs.
- Deformable DETR [55]: backpropagation optimisation is performed via the Adam optimiser, with initial learning rates of $1e - 4$, trained for 50 epochs.

Standard data augmentation techniques, such as Random Crop, Random Flip, have applied during model training with an application probability of 0.5.

5. Results

The model performance is evaluated through MS-COCO metrics [32], with IoU greater than 0.5, using Average Precision (AP) for class-wise, and mAP for the overall performance measurement. Additionally, we compare model performance via: Complexity (number of parameters in mil-

lions, C), the ratio between mAP and the number of parameters in the architecture (mAP:C), and the inference time in milliseconds (ms) taken by the respective model to process an individual frame. Model inference is carried out on the NVIDIA 1080Ti GPU. The highlighted values in each table denote the maximal performance achieved.

5.1. Benchmarking on Infrared dataset

Table 4. FLIR: Performance (class name indicates AP, mAP indicates mean average precision of all classes) of CNN models.

| Model | Average Precision (AP) | | | mAP |
	Person	Bicycle	Car	
MMTOD-UNIT [13]	0.644	0.494	0.707	0.615
ThermalDet [6]	0.782	0.600	0.855	0.746
Pseudo-two-stage [52]	0.787	0.624	0.855	0.755
CR-CNN	**0.877**	**0.681**	0.911	**0.823**
FSAF	0.867	0.675	0.904	0.815
DDETR	0.863	0.659	**0.913**	0.812

Table 4 presents object detection performance in infrared imagery for the first set of experiments using the CNN architectures set out in Section 3.1. These models are applied to the *FLIR* [16] dataset to provide benchmark performance. The best performance on *FLIR* (mAP: 0.823, Table 4, lower) is obtained by Cascade R-CNN [4] achieving the highest AP on two classes (*Person, Bicycle*) out of three. FSAS [53] and DDETR [55], also produce comparable mAP (0.815 and 0.812), where DDETR achieves the maximal AP on the *Car* class. All three models significantly outperform the prior works of [6, 13, 52] (mAP: 0.615, 0.746, 0.755, Table 4, upper). Overall, CR-CNN marginally outperforms the one-stage and transformer-based models in this work. This is possibly due to the architectural design of CR-CNN, where images are sampled with increasing IoU thresholds to tackle different training distributions.

Table 5. FLIR: Model complexity and efficiency.

Model	C	mAP	mAP:C	ms
CR-CNN	69.16	0.823	0.011	93.20
FSAF	36.24	0.815	**0.022**	**64.69**
DDETR	40.09	0.812	0.020	104.98
Inference on: NVIDIA 1080Ti GPU				

Additionally, we present the computational efficiency, and speed, which are crucial criteria for operational and real-world deployment perspectives in Table 5. Being the smallest model, FSAF [53], which has $\sim 2\times$ fewer parameters than Cascade R-CNN [4], obtains the maximal computational efficiency (mAP:C) of 0.022 (Table 5). All three models achieve real-time throughput with FSAF obtaining the fastest individual frame processing speed of 64.69ms, $\sim 1.5\times$ faster compared to Cascade R-CNN [4] and Deformable DETR [55] which obtain 93.2ms and 104.98ms respectively across the same metric.

5.2. Impact of Lossy Compression

Table 6 presents the results from evaluating CNN models (Section 3.1) using *FLIR* dataset after undergoing JPEG lossy compression at six different quality levels: $\{95, 75, 50, 15, 10, 5\}$. As outlined in Section 4.2, the models are first trained on the uncompressed/original dataset (*FLIR*) and evaluated on the compressed variants (*FLIR_c*). Subsequently, the models are re-trained with the compressed imagery (*FLIR_c*) at the respective level and evaluated on the compressed variants (*FLIR_c*).

Table 6. FLIR: Impact of lossy compression on object detection.

| Compression Level | $FLIR \Rightarrow FLIR\_c$ | | | $FLIR\_c \Rightarrow FLIR\_c$ | | |
| | mAP | | | mAP | | |
	CR-CNN	FSAF	DDETR	CR-CNN	FSAF	DDETR
95	0.822	0.816	0.811	0.826	0.821	0.808
75	0.813	0.802	0.803	0.816	0.805	0.800
50	0.789	0.789	0.782	0.806	0.796	0.796
15	0.623	0.606	0.663	0.749	0.743	0.750
10	0.426	0.427	0.489	0.718	0.691	0.705
5	0.109	0.149	0.152	0.613	0.592	0.606

When the models are trained across the original dataset, and evaluated on compressed variants (Table 6, left), we observe that the models retain similar performance at the $\{95\}$ compression level, only decreasing by $\sim 1\%$ on average at the $\{75\}$ compression level. At the compression level of $\{50\}$, the mAP is decreased only by $\sim 3.5\%$ on average across all three models. We observe that a compression level of $\{15\}$ and below has a greater impact on object detection performance. The model performances suffer significantly (mAP decreased by $\sim 80\%$), when the images are compressed heavily, such as at $\{5\}$ compression level across all three models.

Re-training the models on lossy compressed imagery (*FLIR_c*) at different compression levels and subsequently evaluating on the compressed variants (*FLIR_c* $\Rightarrow$ *FLIR_c*) significantly ameliorated the performance (Table 6, right) for all three CNN models at higher compression levels. Re-training at the compression levels of $\{95, 75, 50\}$ does not fully recover the performance compared to original training and evaluation protocol (*FLIR* $\Rightarrow$ *FLIR_c*). However, the impact of lossy compression at higher compression level ($\{5\}$), is mitigated through retraining with all models, achieving mAPs: $\{0.613, 0.592, 0.606\}$ (Cascade R-CNN, FSAF, Deformable DETR correspondingly, Table 6, right), compared to uncompressed settings (mAP: $\{0.109, 0.149, 0.152\}$), an increment of 82%, 74%, and 74% respectively. Amongst three models, two-stage Cascade R-CNN [4] offers superior performance across most of the compression levels, followed by transformer-based Deformable DETR [55], whilst one-stage FSAF [53] are more robust to higher compression levels. Similar performance enhancement is discerned for compression levels of $\{15, 10\}$. The benefit of re-training on compressed images
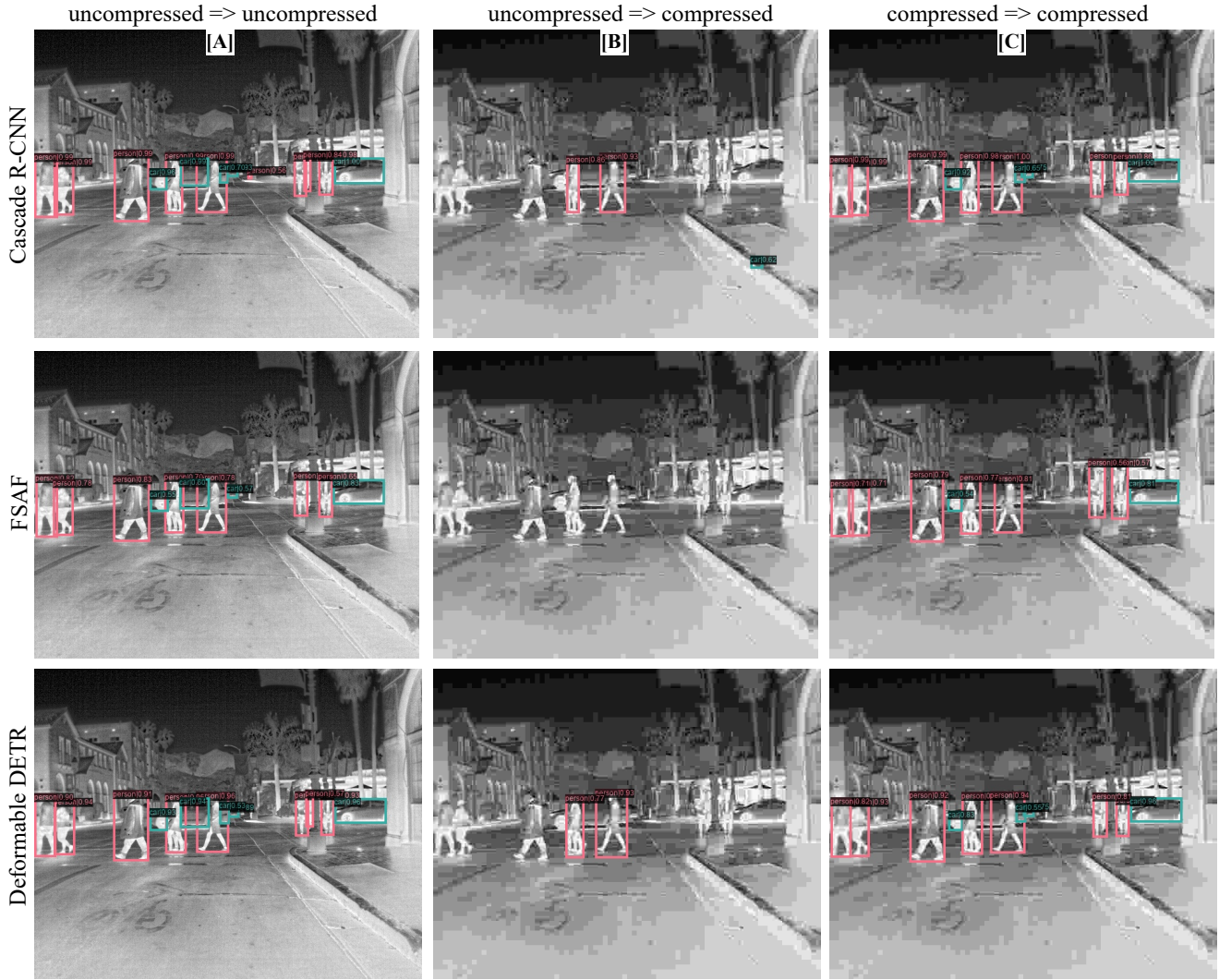
Figure 2. Detection using varying CNN models: [A] uncompressed settings (*FLIR ⇒ FLIR*), [B] *FLIR ⇒ FLIR_c* at compression level 5, [C] re-training with compressed images at compression level 5 (*FLIR_c ⇒ FLIR_c*).
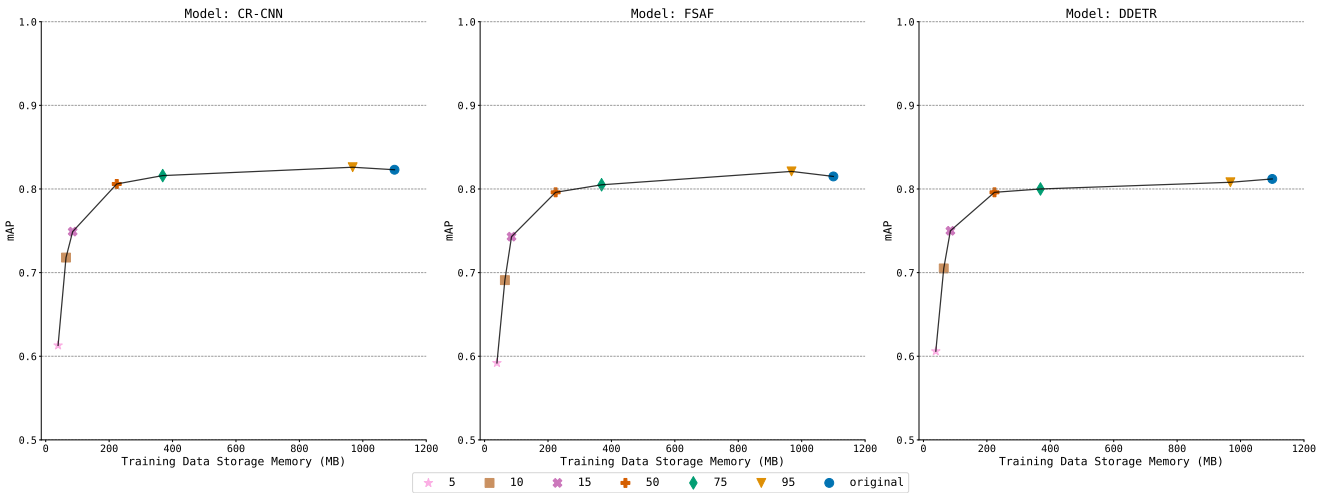

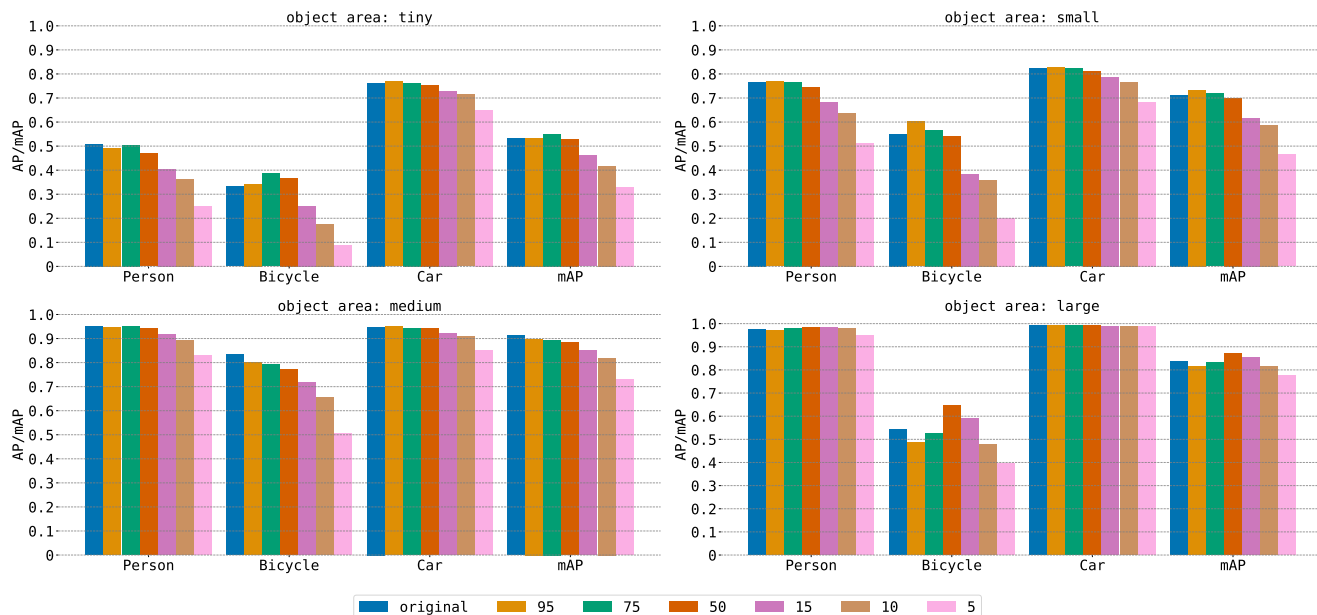Figure 3. FLIR: CNN models performance (mAP) against training data memory storage (Megabyte) at different compression levels.

Figure 4. FLIR: Object area-wise AP/mAP analysis at different compression levels using Cascade R-CNN [4].

is depicted in Figure 2, C, where compressed image trained models successfully detect the objects, contrary to the missing detection in Figure 2, B.

Overall, the re-training improves the detection performance while affording a lossy JPEG compression rate much higher in terms of reduced image storage requirements, as illustrated in Figure 3. We observe that the detection performance of the CNN models is resilient up to the compression level of $\{50\}$, while requiring only $1/5^{th}$ of training data storage overhead memory compared to uncompressed data storage (224 MB vs 1100 MB, Figure 3).

5.3. Object Area-wise Performance Analysis

This section reports the in-depth analysis of the detection performance based on the discrete object area definitions (Section 3.3) for each class in the dataset. First, we calculate AP for object area-wise using the uncompressed dataset setting ($FLIR \Rightarrow FLIR$). Subsequently, we follow the compressed dataset setting ($FLIR\_c \Rightarrow FLIR\_c$) to obtain the object area-wise statistics as presented in Figures 4, 5, and 6.

The *FLIR* test-set comprises of $\sim 45\%$ *small* and $\sim 45\%$ *medium* area category objects (Table 3) across three different classes. We observe that the impact of the heavy compression is more prevalent on *small* object areas than *medium*. The *small* object area mAP is reduced by $\sim 35\%$ on compressed imagery (at compression of 5) compared to the uncompressed training settings across three CNN models (Figures 4, 5, and 6), with FSAF [53] suffering the most. On the other hand, the *medium* area performance is decreased by average 19%. The impact of lossy compression (at compression of 5) is also noticeable on the *tiny* cate-

gory with mAP reduced by 36% on average. Nonetheless, the *large* category can withstand the maximal compression without severely affecting the detection performance.

Amongst the three object classes of *FLIR* dataset, the *car* class is less impacted by the compression than *person, bicycle* across all object area categories. This is possibly due to *car* class having a higher percentage of *medium* and *large* area than other two classes. Bicycle was the worst performing class and the one most sensitive to compression. This is likely due to the reasons of: 1) The thin frames of bicycles become harder to see under higher degrees compression and 2) The rotational bias of bicycles (i.e. When a bike is viewed from the front rather than from the side, it appears thin and close to one-dimensional) and thus are harder to detect. Furthermore, the performances of CNN models are less impacted at the lower compression levels ($\{95, 75, 50\}$) across all object area categories. Overall, Cascade R-CNN [4] achieves the maximal mAP on *tiny, small, medium* object area category, whilst transformer-based Deformable DETR [55] suited to *large* category for both uncompressed and compressed train and evaluation settings.

6. Conclusion

In this work we conduct an extensive study into the impact that lossy JPEG compression applied at differing discrete levels to infrared-band imagery has upon a common set of diverse object detection architectures. We evaluate the performance of three operationally diverse object detection methods, each with vastly differing detection approaches with respect to six discrete classes, $\{95, 75, 50, 15, 10, 5\}$,
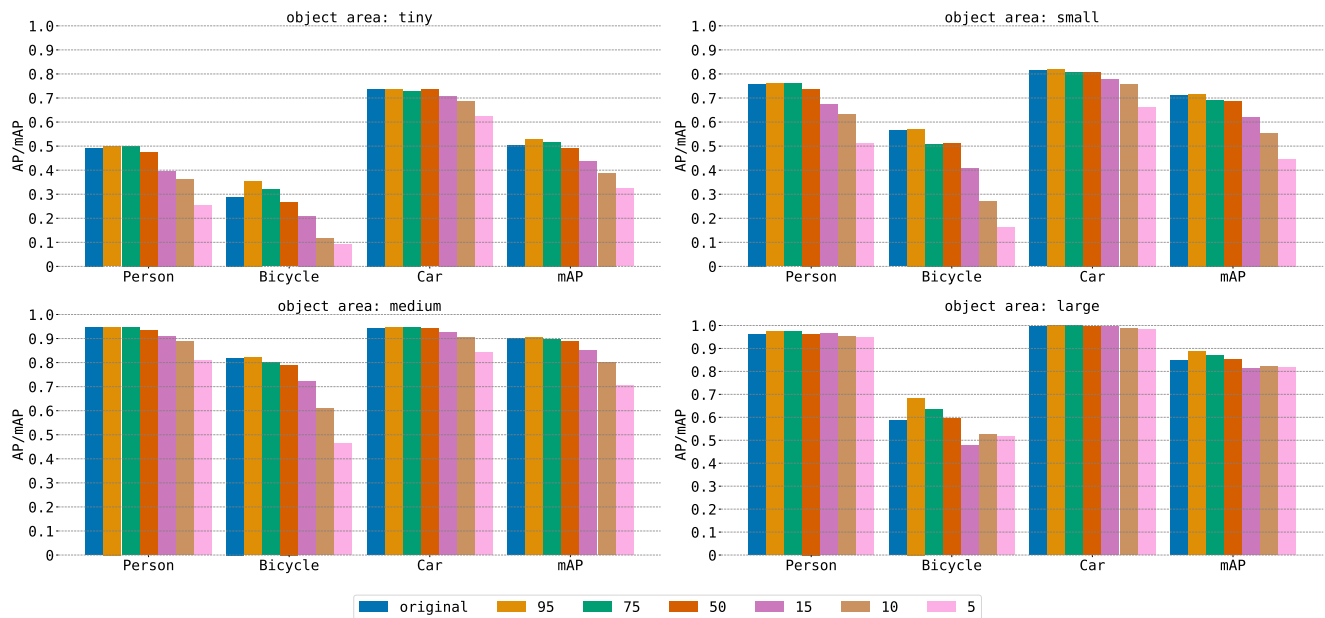
Figure 5. FLIR: Object area-wise AP/mAP analysis at different compression levels using FSAF [53].
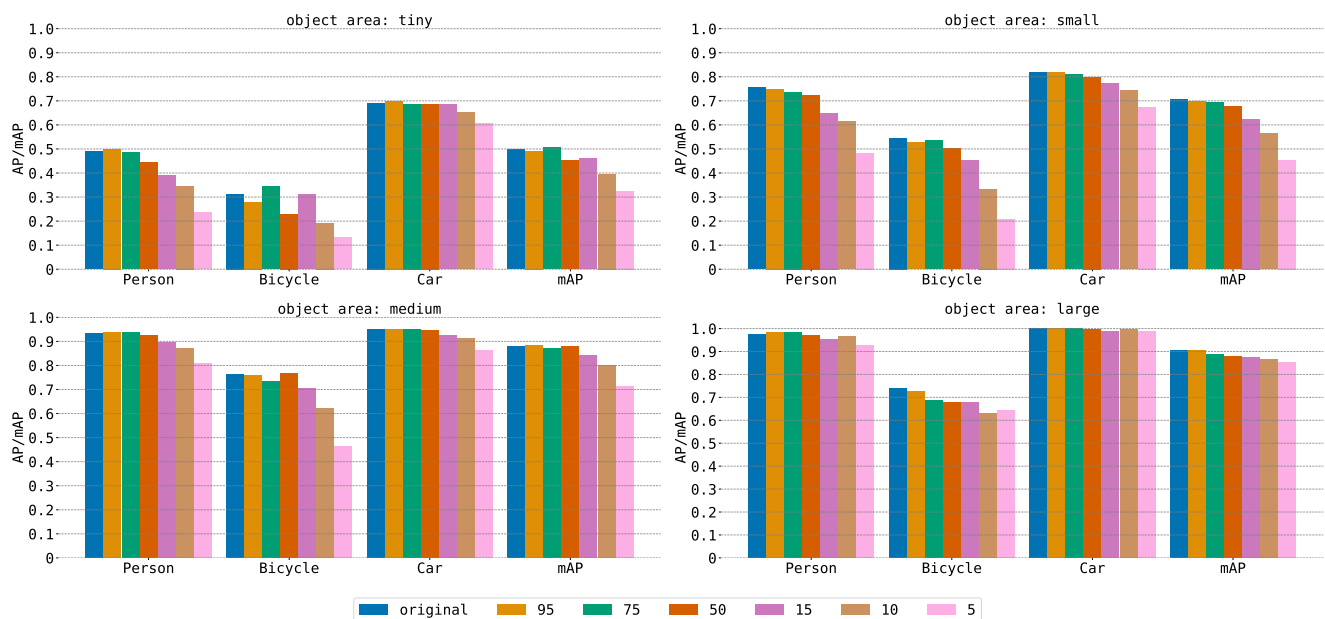

Figure 6. FLIR: Object area-wise AP/mAP analysis at different compression levels using Deformable DETR [55].

of varying degrees of compression applied to the input data. Multi-stage Cascade R-CNN performs slightly better than single-stage and transformer-based architectures. However, single-stage FSAF, which has the smallest number of parameters, achieves the fastest inference time. Within this study, we report that significant compression between 75 and 50 will negligibly affect performance, but will reduce the storage capacity of the data by over 50%. We show that inferring models trained on uncompressed imagery will fail to detect objects across compressed images, but re-training

the same models across compressed images will allow them to detect objects with high accuracy. We also quantitatively report on how objects of differing size are affected by compression by assigning each object to a discrete class according to the bounding box area. We hope that the results presented in this work will help future real-world applications of object detection models applied to thermal imagery to reduce storage and transmission overhead while obtaining minimal performance impact.

References

[1] R Abbott, J Del Rincon, B Connor, and N Robertson. Deep object classification in low resolution lwir imagery via transfer learning. In *Proc. of IMA Conf. on Mathematics in Defence*, volume 2, 2017. 2

[2] M Bertozzi, A Broggi, C Hilario Gomez, RI Fedriga, G Vezzoni, and M DelRose. Pedestrian detection in far infrared images based on the use of probabilistic templates. In *Intelligent Vehicles Symposium*, pages 327–332. IEEE, 2007. 2

[3] Raluca Brehar and Sergiu Nedevschi. Pedestrian detection in infrared images using hog, lbp, gradient magnitude and intensity feature channels. In *Proc. Conf. on Intelligent Transportation Systems*, pages 1669–1674. IEEE, 2014. 1

[4] Z. Cai and N. Vasconcelos. Cascade R-CNN: High quality object detection and instance segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 2019. 1, 2, 3, 4, 5, 7

[5] Yu Cao, Tong Zhou, Xinhua Zhu, and Yan Su. Every feature counts: An improved one-stage detector in thermal imagery. In *Proc. Int. Conf. on Computer and Communications*, pages 1965–1969. IEEE, 2019. 2

[6] Yu Cao, Tong Zhou, Xinhua Zhu, and Yan Su. Every feature counts: An improved one-stage detector in thermal imagery. In *Proc. Int Conf. on Computer and Communications*, pages 1965–1969, 2019. 5

[7] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *Proc. European Conf. on Computer Vision*, pages 213–229, 2020. 1

[8] Kai Chen, Jiaqi Wang, Jiangmiao Pang, Yuhang Cao, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jiarui Xu, Zheng Zhang, Dazhi Cheng, Chenchen Zhu, Tianheng Cheng, Qijie Zhao, Buyu Li, Xin Lu, Rui Zhu, Yue Wu, Jifeng Dai, Jingdong Wang, Jianping Shi, Wanli Ouyang, Chen Change Loy, and Dahua Lin. MMDetection: Open mmlab detection toolbox and benchmark. *arXiv preprint arXiv:1906.07155*, 2019. 4

[9] Yunfan Chen, Han Xie, and Hyunchul Shin. Multi-layer fusion techniques using a cnn for multispectral pedestrian detection. *IET Computer Vision*, 12(8):1179–1187, 2018. 3

[10] James W Davis and Mark A Keck. A two-stage template approach to person detection in thermal imagery. In *Proc. Workshops on Applications of Computer Vision*, volume 1, pages 364–369. IEEE, 2005. 2

[11] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *Proc. Conf. Computer Vision and Pattern Recognition*, pages 248–255, 2009. 1, 2

[12] Chaitanya Devaguptapu, Ninad Akolekar, Manuj M Sharma, and Vineeth N Balasubramanian. Borrow from anywhere: Pseudo multi-modal object detection in thermal imagery. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019. 2

[13] Chaitanya Devaguptapu, Ninad Akolekar, Manuj M Sharma, and Vineeth N Balasubramanian. Borrow from anywhere: Pseudo multi-modal object detection in thermal imagery. In *Proc. of the Conf. on Computer Vision and Pattern Recognition Workshops*, June 2019. 5

[14] Samuel Dodge and Lina Karam. Understanding how image quality affects deep neural networks. In *Proc. Int. Conf. on quality of multimedia experience*, pages 1–6. IEEE, 2016. 2, 3

[15] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *Int. journal of computer vision*, 88(2):303–338, 2010. 1, 2

[16] FLIRSystems. FLIR Thermal Datasets for Algorithm Training. https://www.flir.co.uk/oem/adas/dataset/. 1, 4, 5

[17] Yona Falinie A Gaus, Neelanjan Bhowmik, Brian KS Isaac-Medina, and Toby P Breckon. Visible to infrared transfer learning as a paradigm for accessible real-time object detection and classification in infrared imagery. In *Counterterrorism, Crime Fighting, Forensics, and Surveillance Technologies IV*, volume 11542, page 1154205, 2020. 2

[18] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proc. of the Conf. on Computer Vision and Pattern Recognition*, page 580–587. IEEE Computer Society, 2014. 3

[19] Qishen Ha, Kohei Watanabe, Takumi Karasawa, Yoshitaka Ushiku, and Tatsuya Harada. MFNet: Towards real-time semantic segmentation for autonomous vehicles with multi-spectral scenes. In *Proc. Int. Conf. on Intelligent Robots and Systems*, pages 5108–5115. IEEE, 2017. 3

[20] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proc. Int. Conf. on Computer Vision*, pages 2980–2988, 2017. 1

[21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proc. on Computer Vision and Pattern Recognition*, pages 770–778, 2016. 3, 4

[22] Soonmin Hwang, Jaesik Park, Namil Kim, Yukyung Choi, and In So Kweon. Multispectral pedestrian detection: Benchmark dataset and baseline. In *Proc. of the Conf. on computer vision and pattern recognition*, pages 1037–1045, 2015. 1, 2

[23] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25, 2012. 1

[24] Mikolaj E Kundegorski, Samet Akçay, Grégoire Payen de La Garanderie, and Toby P Breckon. Real-time classification of vehicles by type within infrared imagery. In *Optics and Photonics for Counterterrorism, Crime Fighting, and Defence XII*, volume 9995, pages 266–281. SPIE, 2016. 1

[25] Mikolaj E Kundegorski and Toby P Breckon. A photogrammetric approach for real-time 3d localization and tracking of pedestrians in monocular infrared imagery. In *Optics and Photonics for Counterterrorism, Crime Fighting, and Defence X; and Optical Materials and Biomaterials in Security and Defence Systems Technology XI*, volume 9253, page 92530I, 2014. 1

[26] Mikolaj E Kundegorski and Toby P Breckon. Posture estimation for improved photogrammetric localization of pedestrians in monocular infrared imagery. In *Optics and Photonics for Counterterrorism, Crime Fighting, and Defence XI;*

and Optical Materials and Biomaterials in Security and Defence Systems Technology XII, volume 9652, pages 114–125. SPIE, 2015. 1

[27] Didier Le Gall. MPEG: A video compression standard for multimedia applications. *Communications of the ACM*, 34(4):46–58, 1991. 2

[28] Eun Ju Lee, Byoung Chul Ko, and Jae-Yeal Nam. Recognizing pedestrian's unsafe behaviors in far-infrared imagery at night. *Infrared Physics & Technology*, 76:261–270, 2016. 2

[29] Jianfu Li, Weiguo Gong, Weihong Li, and Xiaoying Liu. Robust pedestrian detection in thermal infrared imagery using the wavelet transform. *Infrared Physics & Technology*, 53(4):267–273, 2010. 1, 2

[30] Shih-Schon Lin. Extending visible band computer vision techniques to infrared band images. 2001. 1

[31] T. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár. Focal loss for dense object detection. In *Proc. Int. Conf. on Computer Vision*, pages 2999–3007, 2017. 1

[32] 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*, pages 740–755, 2014. 2, 3, 4

[33] Jingjing Liu, Shaoting Zhang, Shu Wang, and Dimitris N Metaxas. Multispectral deep neural networks for pedestrian detection. *arXiv preprint arXiv:1611.02644*, 2016. 3

[34] Michael Loveday and Toby P Breckon. On the impact of parallax free colour and infrared image co-registration to fused illumination invariant adaptive background modelling. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 1186–1195, 2018. 1

[35] Amir Mukhtar, Likun Xia, and Tong Boon Tang. Vehicle detection techniques for collision avoidance systems: A review. *IEEE transactions on intelligent transportation systems*, 16(5):2318–2338, 2015. 1

[36] Farzeen Munir, Shoaib Azam, and Moongu Jeon. Sstn: Self-supervised domain adaptation thermal object detection for autonomous driving. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 206–213. IEEE, 2021. 2

[37] Min Peng, Chongyang Wang, Tong Chen, and Guangyuan Liu. Nirfacenet: A convolutional neural network for near-infrared face identification. *Information*, 7(4):61, 2016. 2

[38] M. Poyser, A. Atapour-Abarghouei, and T.P. Breckon. On the impact of lossy image and video compression on the performance of deep convolutional neural network architectures. In *Proc. Int. Conf. Pattern Recognition*, 2020. 2

[39] Matt Poyser, Amir Atapour-Abarghouei, and Toby P Breckon. On the impact of lossy image and video compression on the performance of deep convolutional neural network architectures. In *Proc. Int. Conf. on Pattern Recognition*, pages 2830–2837. IEEE, 2021. 3

[40] J. Redmon and A. Farhadi. Yolo9000: Better, faster, stronger. In *Proc. on Computer Vision and Pattern Recognition*, pages 6517–6525, 2017. 2

[41] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018. 1

[42] S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6):1137–1149, 2017. 1, 2, 3

[43] Iain Rodger, Barry Connor, and Neil M Robertson. Classifying objects in lwir imagery via cnns. In *Electro-Optical and Infrared Systems: Technology and Applications XIII*, volume 9987, page 99870H, 2016. 2

[44] Arnold WM Smeulders, Dung M Chu, Rita Cucchiara, Simone Calderara, Afshin Dehghan, and Mubarak Shah. Visual tracking: An experimental survey. *IEEE transactions on pattern analysis and machine intelligence*, 36(7):1442–1468, 2013. 1

[45] Yuxiang Sun, Weixun Zuo, and Ming Liu. Rtfnet: Rgb-thermal fusion network for semantic segmentation of urban scenes. *IEEE Robotics and Automation Letters*, 4(3):2576–2583, 2019. 3

[46] Michael Teutsch, Thomas Muller, Marco Huber, and Jurgen Beyerer. Low resolution person detection with a moving thermal infrared camera by hot spot classification. In *Proc. of the Conf. on Computer Vision and Pattern Recognition Workshops*, pages 209–216, 2014. 1

[47] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, 2017. 3

[48] Gregory K. Wallace. The jpeg still picture compression standard. *Commun. ACM*, 34(4):30–44, Apr. 1991. 3, 4

[49] Gregory K Wallace. The jpeg still picture compression standard. *IEEE transactions on consumer electronics*, 38(1):xviii–xxxiv, 1992. 2

[50] Thomas W Webb, Neelanjan Bhowmik, Yona Falinie A Gaus, and Toby P Breckon. Operationalizing convolutional neural network architectures for prohibited object detection in x-ray imagery. In *Proc. Int. Conf. on Machine Learning and Applications (ICMLA)*, pages 610–615. IEEE, 2021. 3

[51] Huaizhong Zhang, Chunbo Luo, Qi Wang, Matthew Kitchin, Andrew Parmley, Jesus Monge-Alvarez, and Pablo Casaseca-De-La-Higuera. A novel infrared video surveillance system using deep learning based techniques. *Multimedia Tools and Applications*, 77(20):26657–26676, 2018. 2

[52] Tong Zhou, Zhentao Yu, Yu Cao, Hongyang Bai, and Yan Su. Study on an infrared multi-target detection method based on the pseudo-two-stage model. *Infrared Physics Technology*, 118:103883, 2021. 5

[53] Chenchen Zhu, Yihui He, and Marios Savvides. Feature selective anchor-free module for single-shot object detection. In *Proc. of the Conf. on computer vision and pattern recognition*, pages 840–849, 2019. 1, 2, 3, 4, 5, 7, 8

[54] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017. 2

[55] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable DETR: Deformable transformers for end-to-end object detection. In *Proc. Int. Conf. on Learning Representations*, 2021. 1, 2, 3, 4, 5, 7, 8