# Meta-token Learning for Thermal Object Detection with Transformers

1st Marco Parola
*University of Pisa*, *Pisa*, Italy
marco.parola@ing.unipi.it
0000-0003-4871-4902

2nd Anders S. Johansen
*Aalborg University*, Aalborg, Denmark
asjo@create.aau.dk
0000-0002-9330-522X

3rd Neelu Madan
*Aalborg University*, Aalborg, Denmark
nema@create.aau.dk
0000-0001-5778-3470

4th Mario G. C. A. Cimino
*University of Pisa*, *Pisa*, Italy
mario.cimino@unipi.it
0000-0002-1031-1959

5th Kamal Nasrollahi
*Aalborg University*, Aalborg, Denmark
*Milestone Systems A/S*, Brøndby, Denmark
kn@create.aau.dk
0000-0002-1953-0429

6th Thomas B. Moeslund
*Aalborg University*, Aalborg, Denmark
tbm@create.aau.dk
0000-0001-7584-5209

***Abstract*—Transformers have emerged as a dominant architecture in computer vision, demonstrating strong performance across a range of tasks. In this work, we investigate their effectiveness for object detection in thermal imagery, a setting often challenged by variable environmental conditions. We focus on a long-term thermal surveillance dataset captured using a stationary thermal camera subjected to diverse weather scenarios. To address the influence of such external factors, we propose an analysis that integrates weather information as meta-tokens alongside thermal images, enabling the model to account for environmental context during detection. Among the fusion strategies explored, we find that a token-level concatenation allows transformers to partially exploit the auxiliary weather data, leading to improved detection performance. Our study highlights the potential of meta-token transformer-based architectures for robust detection in challenging thermal environments.**

***Index Terms*—Object detection, Thermal imaging, Meta-token learning, DETR, YOLOS, Weather information.**

## I. Introduction

Transformer architectures, originally designed for natural language processing [1], have progressively gained significant attention in computer vision by exploiting self-attention mechanisms and posing as major competitors to traditional convolutional neural networks (CNNs) for several downstream tasks [2] and computer vision domains, including healthcare [3], [4], text classification [5], and thermal imaging [6], [7]. The use of transformer and deep learning (DL) models for thermal image recognition in long-term analysis presents several challenges due to numerous factors that can influence the visual appearance and, consequently, the performance of such methods. Thermal images are based on the capture of infrared radiation through thermal cameras, and their visual identity not only varies with the seasons but also dramatically changes during the day and night or with different weather conditions. Indeed, thermal imaging cameras detect infrared radiation to visualize heat emitted by objects and are generally divided into two types: qualitative and quantitative. Qualitative cameras highlight temperature variations within a scene through a recalibration process carried out periodically to maximize the temperature gap detected. Quantitative or absolute cameras, instead, assign precise temperature values to each pixel, allowing a more accurate and coherent heat analysis over time [8].

This study explores the challenges of using transformers for thermal image recognition under varying weather conditions, with a focus on object detection as a downstream task. The analysis is conducted on an extended version of the Harborfront image dataset [9], collected using a qualitative thermal camera, to understand how external weather variations affect the performance of DL models in detecting and accurately identifying objects in thermal images. Further details on the dataset will be provided in Section II-B.

The dataset was collected using a relative thermal camera to record images emphasizing thermal differences, thanks to a recalibration process that maintains this relative difference high over time. Additionally, some environmental data, such as temperature and humidity, have been coupled with the images. The initial insight that motivated this project is recognizing that, since the camera operates on a relative basis without an absolute thermal reference point, the appearance of the images could vary over time due to external factors.

This insight is confirmed by regression analysis between the visual appearance of images and the environmental factors to understand any dependencies. Figure 1 presents some examples of the analysis where image features, extracted using a self-supervised method with a convolutional autoencoder, were employed to perform regression on metadata related to the external conditions corresponding to each image. The results clearly indicate the presence of a relationship between the extracted image features and the associated external conditions. This relation can potentially be exploited to deploy DL models in computer vision systems that are more robust to external weather noise.

In addition to this insight supported by a preliminary exploratory analysis, previous research on the same dataset

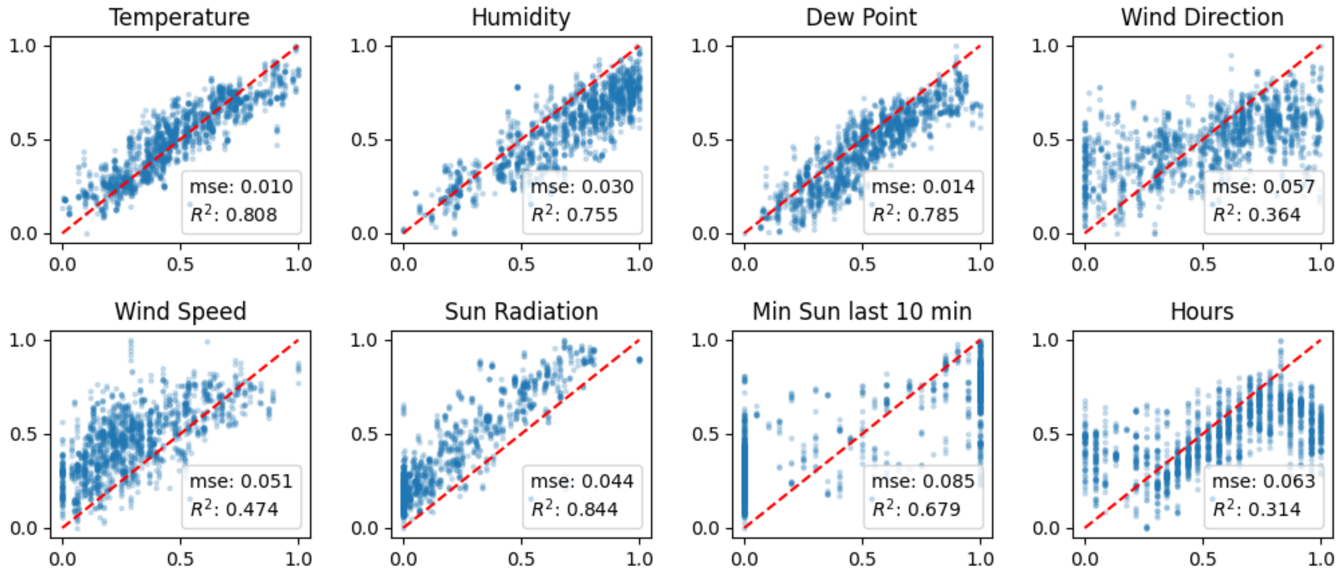


Fig. 1: Regression analysis showing weather condition variables fitted to self-supervised visual image features

[10] shows how a causal connection and significant correlation between model performance demonstrated variations in atmospheric conditions, particularly humidity and temperature. While less pronounced, but still present, is the association between scene activity and the day/night cycle. The authors attempted to resolve the phenomenon of conceptual drift by identifying its occurrence and incorporating more data into the training process.

Based on these premises, this paper investigates the following research question: How much can the integration of seasonality and weather information into DL processing help mitigate performance degradation and improve model robustness under varying external conditions? Furthermore, we explore methodologies to integrate this information into DL models effectively.

## II. Background and Related Work

This section is divided into two parts. First, we examine SoTA DL architectures for object detection. Then, we review the incorporation of contextual information techniques on metadata to improve the performance of DL-based computer vision methods.

### A. *Deep learning for object detection*

Object detection is a problem in image processing involving the identification and localization of objects within an image or a video sequence. It can be defined as follows: given an input image $I$, the goal of object detection is to produce a set of bounding boxes $B = \{b_1, b_2, \ldots, b_n\}$ and corresponding class labels $C = \{c_1, c_2, \ldots, c_n\}$, where each bounding box $b_i$ specifies the location of an object in the image; while each class label $c_i$ indicates the category of the object from a predefined set. Each bounding box $b_i$ is defined by a tuple $(x_i, y_i, w_i, h_i)$, where $x_i$ and $y_i$ represent the coordinates of the top-left corner of the bounding box. $w_i$ and $h_i$ denote the width and height of the bounding box, respectively.

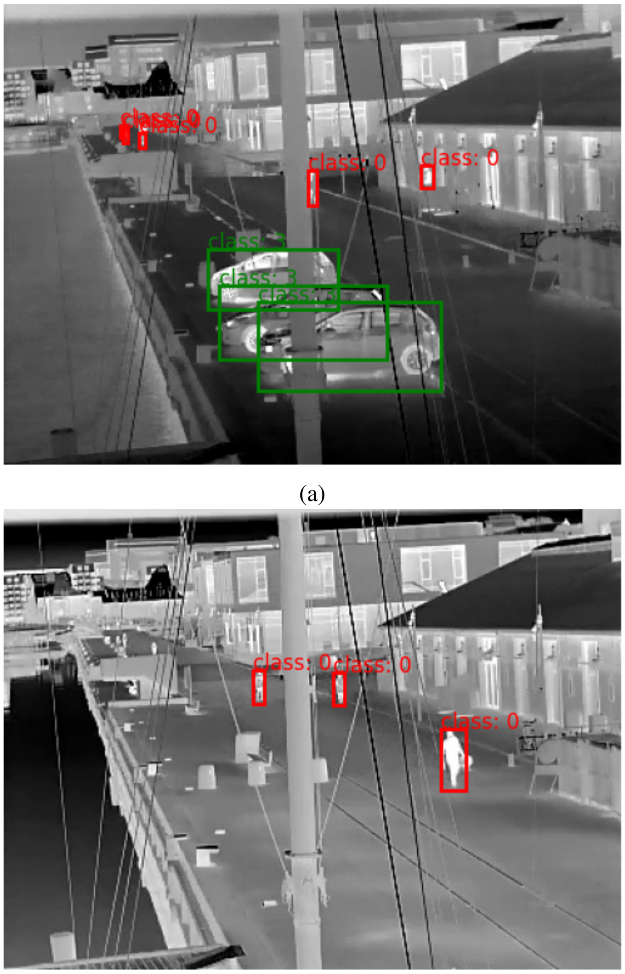
Two main families of architectures addressing the object detection task were examined: convolutional models and transformer-based models. Among the convolutional models, one of the most popular architectures for object detection is YOLO (You Only Look Once) [11]. YOLO processes the entire image in a single step, simultaneously identifying objects and classifying them into predefined categories. A key feature of YOLO is non-maximum suppression (NMS). After providing multiple bounding boxes for detected objects, NMS is used to eliminate redundant boxes. The NMS process begins by ranking the provided bounding boxes according to their confidence scores. The bounding box with the highest score is chosen as the benchmark. Each subsequent box is then compared to this benchmark using the Intersection over Union (IoU) metric. When the IoU value between a box and the benchmark exceeds a specific threshold, the box is considered redundant and removed. Conversely, if the IoU value is below the threshold, the frame is retained as a separate detection.

Transformers are DL architectures originally developed for sequential data in natural language processing and have since demonstrated significant advancements in computer vision. These models process images by generating tokens through convolutional backbones that extract visual features or by dividing images into patches, which are then reduced in dimensionality via linear projection. Positional embeddings are incorporated into each token to retain spatial information, allowing the model to maintain the relative positioning information of each token. The attention mechanism is a key component of Transformers, allowing the model to measure the significance level of each token based on its relationship to others. This capability allows Transformers to effectively capture complex spatial and contextual dependencies within visual data, enhancing their performance across a variety of vision tasks.

Among the transformer-based architectures, we examine (i) DEtection TRansformer (DETR), belonging to the encoder-decoder subcategory and (ii) You Only Look at One Sequence (YOLOS) as an encoder-only model. DETR [12] consists of a convolutional backbone followed by an encoder-decoder transformer that can be trained end-to-end for object detection. It greatly simplifies the complexity of models such as Faster-R-CNN and Mask-R-CNN, which use elements such as region proposal, non-maximum suppression procedure, and anchor generation. Our implementation relies on DenseNet as the backbone [13], instead of ResNet as the original implementation. YOLOS, proposed in 2021 [14] is a modified version of Vision Transformer (ViT) [15] trained using the DETR loss. Despite its simplicity, a base-sized YOLOS model is able to achieve 42 AP on COCO validation 2017 (similar to DETR and more complex frameworks such as Faster R-CNN). This experiment allows us to propose a general framework for transformers since we are also considering the encoder-only architecture.

### B. Harborfront Dataset

The dataset comprises 1,069,051 thermal images annotated with bounding boxes and class labels for four object categories: person, bicycle, motorcycle, and vehicle. Each image is paired with weather data. Captured from surveillance footage in Aalborg, Denmark, the dataset follows a periodic recording schedule of two minutes every 30 minutes, enabling the construction of a time series of weather conditions. Due to anomalies in environmental data acquisition, some values are missing; these were imputed using a forward-fill method, replacing gaps with the most recent available data, a valid approach given the gradual variability of environmental factors [16].



(a)

(b)

Fig. 2: Two thermal images examples of the dataset: (a) taken in summer at 8 a.m. and (b) taken in winter at 8 a.m.

### C. Integrating metadata for recognition

Several works in the context of adverse weather conditions focus primarily on image enhancement techniques, particularly for deraining or defogging images [17] [18] [19], or on improving model robustness through training on augmented datasets where various weather scenarios are simulated [20]. These approaches have shown effectiveness in RGB images, where cameras can capture detailed textures and granular patterns that are crucial for distinguishing between different objects and weather features. However, in the thermal imaging domain, these methods are less applicable as thermal cameras capture heat signatures, resulting in a reduced presence of granular texture and fine details.

An alternative approach that has proven to be effective is task-conditioned domain adaptation [21]. This method uses task-specific information to guide the model-fitting process. In the thermal domain, Kieu et al. [22] used the representation of weather as a constraining parameter, which forces the model to directly incorporate auxiliary information, forcing the network to adapt to be aware of and effectively utilize the induced contextual weather information. In this configuration, the network is explicitly guided to recognize and respond to specific contextual cues in the weather data. The auxiliary representation can be treated as an additional, potentially redundant source of information that guides the network indirectly. Auxiliary metadata does not directly influence the primary learning process of the model but provides additional cues that help refine the network's ability to interpret thermal images. Specifically, the task that conditions the main downstream task introduced in [22] is a binary classification problem in which the model must distinguish between day and night.

In our previous work [23], we implemented a fine-grained version for the same problem, in which the conditioning task does not solve a binary problem in a discrete domain, but a regression problem in a continuous domain to predict the exact metadata value (e.g., temperature). This fine-grained variant has been approached with two different implementations, referred to as direct- and indirect-conditioning from now on. The direct conditioning approach was implemented on the YOLOv5 architecture; an auxiliary branch is taken from one of the first stages of the feature extractor and added to the model.

The feature representation is then reduced from a sequence of fully connected layers to produce a single value to be regressed.

On the other hand, the indirect conditioning strategy was implemented in the transformer architecture. We included an additional token in each input sample, as was often done to solve a classification problem [15], and propagated it through each coding layer. Also, in this case, an auxiliary branch was appended to the model to regress the target weather value from the additional token of the last encoder layer. This disjoint approach is theoretically designed to allow the network to ignore data that does not contribute to weather prediction, unlike the direct conditioning method, in which the first convolutional blocks of the model are trained both to solve the main downstream task and to predict weather information.

In contrast to the task conditioning approach, an alternative strategy for integrating metadata into model training is to employ a multimodal analysis. This approach allows the model to utilize additional information without explicitly enforcing any specific constraints or conditioning factors. Multimodal analysis provides the metadata alongside the primary input, enabling the architecture to learn the relationships between different input modalities autonomously. In this regard, transformer-based architectures have proven to be an effective solution due to their ability to understand long-term relationships in sequences through attention mechanisms [24]. This strategy has been successfully applied in various fields, such as medical imaging [25] and speech recognition [26], effectively integrating different data sources to improve results. However, to the best of our knowledge, it has not been applied to the integration of meteorological information to solve downstream computer vision tasks on thermal data.

## III. Methodology

The task conditioning methods discussed in Section II-C were based on the hypothesis that improvement could result from a relationship between different loss functions. Specifically, metadata integrated through a dedicated loss function for task conditioning could influence model weights while training with the loss of the primary task, leading to an overall improvement in downstream task performance. However, this hypothesis did not prove effective, as no performance improvement was observed using this approach. Therefore, we design a more general transform-based multimodal analysis approach that does not impose strong assumptions, such as those between loss functions. Instead, it allows the model to autonomously learn any relationship in the data during training in a more adaptive manner if which can improve the performance of the downstream task.

Figure 3 presents the framework we follow to implement a multimodal analysis strategy where the multimodal inputs consist of images $X_i$ and multiple weather time series $X_m$ as metadata. Two main steps must be performed to handle multimodal inputs of different sizes from distinct sources: (i) tokenize the input and (ii) define an embedding common space to represent the data.

Images and time series are pre-processed to generate the feature vectors $\bar{X}_i$ and $\bar{X}_m$ using a CNN backbone and a linear projection, respectively. Then, the tokens are aggregated, and the attention mechanism captures the relationships between them. In the following sections, several aggregation modalities implementing different multimodal analysis strategies will be described, covering the processes of tokenization and embedding to produce the aggregated input $X$ to the transformer block.

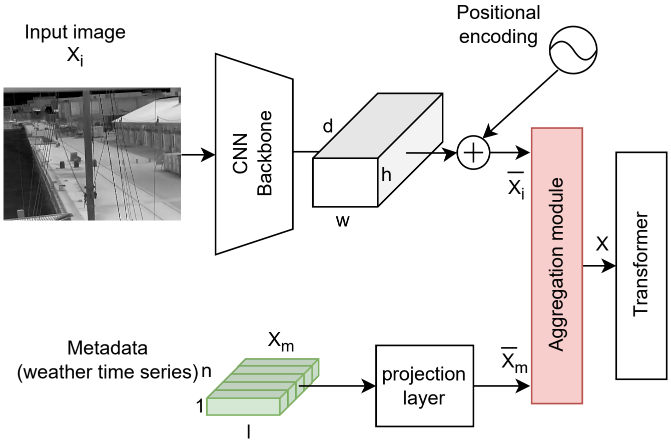


Fig. 3: General architecture scheme implementing the multimodal aggregation embeddings

For affine aggregation, metadata is utilized to modulate images through an affine transformation, as defined by Equation 1. The transformation adjusts the image representation by combining it with the modulating factors $\alpha$ and $\beta$ with respect to $\bar{X}_m$.

From this formulation, we realized specific implementations. We assume $\alpha = \bar{X}_m^T$ to remove the multiplicative factors, resulting in a pure additive relation between $\bar{X}_i$ and $\bar{X}_m$. Then, we set $\beta = 0$, resulting in a purely multiplicative relationship between them. Finally, we set $\alpha$ and $\beta$ as trainable parameters, letting the network learn the affine transformation. Regarding the tokenization aspect, we explore different approaches. We projected $\bar{X}_m$ into a single value, which was then used to scale and shift $\bar{X}_i$. In another implementation, we projected $\bar{X}_m$ into a dimensional space equivalent to that of $\bar{X}_i$. In all these cases, positional encoding is applied, and reshaping is performed to generate the sequence required for input into the transformer.

$$X = \bar{X}_i \cdot \alpha \bar{X}_m + \beta \bar{X}_m \tag{1}$$

In the concatenation strategy, after the positional encoding has been added, the output of the convolutional backbone $\bar{X}_i$ of size $[w \times h \times d]$ is reshaped in a sequence of $w \times h$ tokens having len of $d$ resulting in a tensor of size $[(w \cdot h) \cdot d \times 1 \times 1]$. Then, inspired by the original implementation of a classification token proposed for BERT [27] and ViT [15], we concatenate an extra token containing the metadata time series information. Hence, the weather time series is projected in a single token of size $[d \times 1 \times 1]$. Such token is forwarded across each transformer encoder block to prevent metadata information from being lost through processing between layers.

As transformer-based models, we use DETR and YOLOS, as introduced in Section II-A. To explore performance, in addition to the original base models, we explore some architectural variants. These variants differ in hyperparameters such as the number of attention layers, heads, embedding size, and fully connected layer size. Specifically, in addition to the base models, we also implement and evaluate reduced versions of DETR and YOLOS referred to as tiny. Further details about the implemented architecture hyperparameter values can be found on the project's GitHub repo [28]. Figure 3 illustrates the number of trainable parameters for all the combinations between the architecture size (base and tiny) and the metadata aggregation strategies (vanilla, concatenation, and affine).

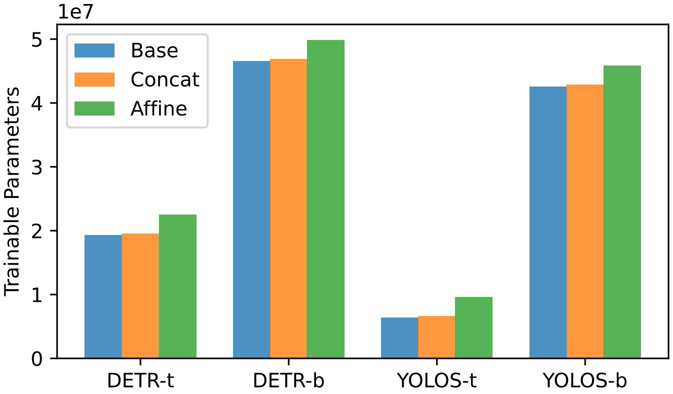


Fig. 4: Bar chart of the number of trainable parameters for different DETR and YOLOS implementations. The Base and Tiny versions, named -b and -t, respectively, are shown.

## IV. Experiments and Results

In this section, we present a comparison of results achieved by various models. For the task conditioning approaches, we reference our previous work as the state-of-the-art benchmark. Following this, we provide the results obtained using different multimodal analysis strategies. The training, validation, and test sets used in our experiments follow the original dataset split as described in [29].

### A. Evaluation metrics

Intersection over Union (IoU) is a popular metric in object detection. Given actual and predicted bounding boxes, IoU measures the ratio between the area of intersection and the area of their union, providing a measure of how closely the prediction overlaps the actual position of the object.

We can derive the average precision at a given IoU threshold (mAP@th) from this. It measures each object class's average precision (AP) at a given IoU threshold. Then it averages these AP values for all classes, providing a complete assessment of the model's detection performance. A more comprehensive evaluation metric is the average precision between th1 and th2 thresholds (mAP@th$_1$-th$_2$), calculating the average precision between multiple thresholds and averaging the results; it provides a more balanced evaluation and is less influenced by the performance of the single-threshold model. Equations 2 and 3 show the metrics and the actual threshold values used during the experiment phase, where $C$ is the number of classes.

$$mAP@50 = \frac{1}{C} \sum_{i=j}^{C} AP@50_j \tag{2}$$

$$mAP@50-95 = \frac{1}{C} \sum_{i=j}^{C} \sum_{i=0}^{9} AP@(50+5i)_j \tag{3}$$

### B. Experimental setup

The new experiments were conducted on a Linux machine equipped with an Nvidia GeForce RTX 4090 GPU with 24 GB of RAM. The experiments were implemented using PyTorch version 2.2 and CUDA version 12.3. Each model was trained for up to 150 epochs, with an early stopping condition based on monitoring the loss and a patience value set to 10 epochs. The batch size for all experiments was set to 16. A hyperparameter grid search was conducted to find the best learning rate in the range $[10^{-5}, 10^{-7}]$ by varying the step size logarithmically. The losses contributing to the final detection loss were weighted according to the guidelines provided in the original DETR paper. To ensure repeatability and transparency, the experiment code has been publicly released on GitHub [28].

As anticipated in the methodology section, we consider DETR and YOLOS, both base and tiny, as vanilla models on which to implement multimodal strategies. Among the multimodal methods outlined in the methodology, in addition to the concatenation strategy, we examined the affine strategy and two more specific cases of the multiplicative and additive approaches. Among the latter three, we present only the results of the affine approach, as it serves as a generalization of the multiplicative and additive methods and provides better performance. We conducted experiments by varying the metadata time series length. For the sake of brevity, in Table I, we also varied the length of the metadata time series, reporting results for three cases: length 1 (time-window preceding the image), 6 (thirty minutes), and 14 (seven hours).

### C. Ablation study

We also propose an ablation study to assess the effectiveness of incorporating metadata information into the model. The primary goal is to determine whether the model truly benefits from the integration of metadata or whether the observed performance improvements are merely due to an extra number of trainable parameters. Furthermore, this also allows us to analyze the performance of a model trained on both images and meteorological time series in a scenario where the latter are not available. We replaced the time series metadata with *dummy* data, keeping the metadata projection layer and the additional token for the concatenation approach. In particular,

TABLE I: mAP for direct/indirect task conditioning from [23], and multimodal strategies (Concat and Affine). Abbreviations: Temp-temperature, Hum-humidity, ToD-time of day, tsl-1/6/14-time series lengths. *previous paper.

| **Models** | | | **Metrics** | |
|---|---|---|---|---|
| Architecture | Strategy | param. | map@50 | map@95-50 |
| YOLOv5* | Baseline | - | **60.4** | **46.5** |
| | Dir-cond | Temp | 58.4 | 41.0 |
| | | Hum | 49.3 | 29.3 |
| | | ToD | 54.9 | 43.9 |
| Def. DETR* | Baseline | - | 33.2 | 20.2 |
| | Indir-cond | Temp | 29.7 | 18.4 |
| | | Hum | 21.3 | 11.4 |
| | | ToD | 28.9 | 17.8 |
| Tiny DETR | Baseline | - | 45.712 | 19.905 |
| | Concat. | tsl-1 | 47.981 | 21.985 |
| | | tsl-6 | 50.847 | 23.288 |
| | | tsl-14 | 51.628 | 24.022 |
| | Affine | tsl-1 | 46.613 | 19.731 |
| | | tsl-6 | 47.118 | 20.662 |
| | | tsl-14 | 48.157 | 21.300 |
| Base DETR | Baseline | - | 46.872 | 20.146 |
| | Concat. | tsl-1 | 48.809 | 22.407 |
| | | tsl-6 | 51.732 | 23.698 |
| | | tsl-14 | 52.539 | 24.363 |
| | Affine | tsl-1 | 47.056 | 20.005 |
| | | tsl-6 | 47.610 | 20.987 |
| | | tsl-14 | 48.751 | 21.622 |
| Tiny YOLOS | Baseline | - | 42.859 | 17.761 |
| | Concat. | tsl-1 | 44.733 | 19.530 |
| | | tsl-6 | 46.878 | 20.065 |
| | | tsl-14 | 47.591 | 20.641 |
| | Affine | tsl-1 | 42.643 | 19.109 |
| | | tsl-6 | 43.725 | 19.539 |
| | | tsl-14 | 43.849 | 19.751 |
| Base YOLOS | Baseline | - | 43.979 | 18.042 |
| | Concat. | tsl-1 | 45.582 | 19.817 |
| | | tsl-6 | 47.668 | 20.413 |
| | | tsl-14 | 48.326 | 20.942 |
| | Affine | tsl-1 | 43.853 | 19.459 |
| | | tsl-6 | 44.206 | 19.717 |
| | | tsl-14 | 44.795 | 20.024 |

we injected random values and sequences composed entirely of ones. The experiment was conducted both during inference and during training: injecting random values and ones during inference after training the model and training the architecture with these dummy sequences.

Figure 5 shows the results of the ablation study conducted on both DETR and YOLOS. For reasons of compactness, we present the results referring only to the injected sequence length of 1. Similar trends are also observed for other sequence values.

## V. Discussion

The success of transformer models is often correlated with their increasing size, number of trainable parameters, and computational complexity, with larger models typically achieving superior performance [30]. However, in our experiments, this trend was not as evident. Specifically, transitioning from the tiny models to the base models, despite the substantial increase in trainable parameters, resulted in only marginal improvements in performance. In contrast, the use of multimodal techniques, particularly the concatenation strategy, demonstrated a performance boost with a minimal increase in the number of trainable parameters. This finding highlights the importance of efficient architectural enhancements, such as metadata aggregation strategies, over simply parameter scaling to implement deeper models. However, these improvements are partially denied by the ablation study.

Indeed, from the ablation study plots, we can conclude that: (i) the performance improvements observed in Table 1 for both concatenation and affine strategies can be partially attributed to a larger number of trainable parameters, as evidenced by the general increase in performance when introducing metadata into the architecture for multimodal analysis, regardless of whether the input data is real, random or consists of sequences of one. (ii) The concatenation method for multimodal analysis proves to be more efficient and robust on this dataset. Incorporating meteorological metadata via concatenation provides better detection performance than the affine strategy. (iii) A further advantage of the concatenation approach is the capability to exploit metadata information during training to improve performance without completely relying on it. The model is able to maintain performance results comparable to the baseline even when dummy information is used instead of real weather data at the time of inference. This flexibility makes the concatenation approach particularly suitable for real-world applications, such as surveillance systems that may depend on external sensors or APIs to retrieve weather metadata. In cases where sensors are damaged or APIs are unreachable, the concatenation strategy ensures that the system can continue to operate with near-baseline performance, regardless of the availability of external information.

## VI. Conclusion

In this study, we investigated the application of Transformer-based architectures for object detection in thermal imagery, focusing on four categories: person, bicycle, motorcycle, and vehicle. While Transformers have demonstrated superior performance over CNNs in various downstream tasks and datasets, they seem to struggle with the low-resolution thermal imagery presented in this dataset [23] [31]. However, we introduced several methods for integrating meteorological data into transformer architectures alongside thermal images, including both direct and indirect conditioning methods, as well as various multimodal fusion techniques, such as concatenation and affine strategy. Among the tested approaches, the concatenation strategy applied to Transformer-based models yielded the highest improvement in mAP and robustness. Indeed, this method showed minimal performance degradation when meteorological data were unavailable during inference, making it highly suitable for real-time deployment in practical

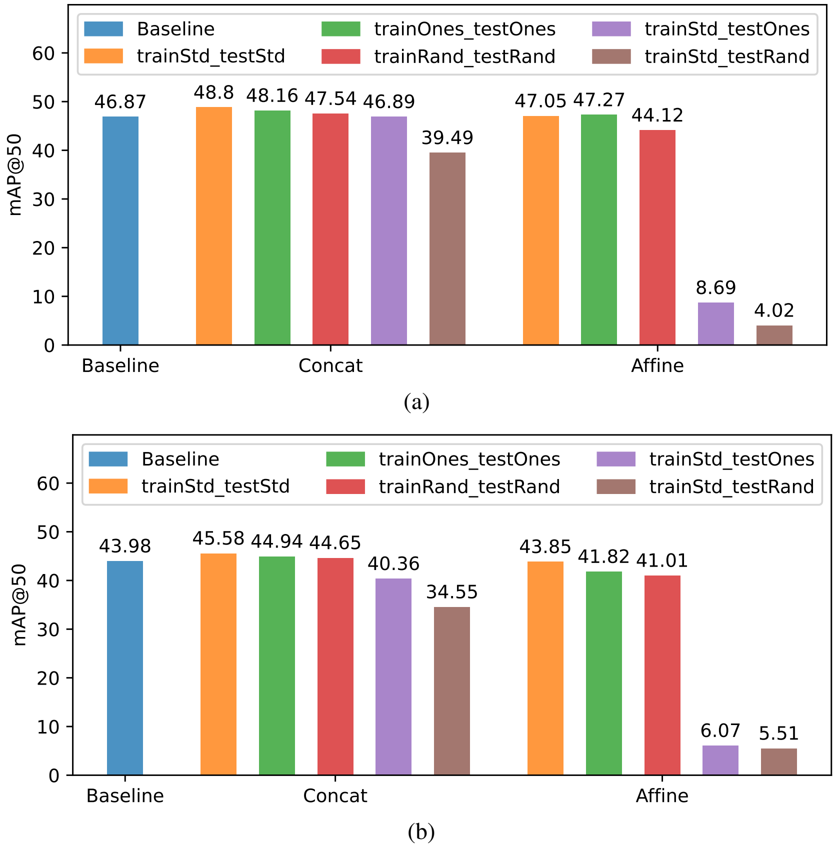


Fig. 5: Ablation study assessing concatenation and affine metadata integration strategies. (a) refers to base DETR, (b) refers to base YOLOS. Labels represent: 'Baseline', 'trainStd_testStd' (metadata inserted both in the training and test), 'trainOnes_testOnes' (ones replace metadata both for training and test), 'trainRand_testRand' (random values replace metadata both for training and test), 'trainStd_testOnes' (metadata for training, ones for testing), and 'trainStd_testRand' (metadata for training, random values for testing).

scenarios. This resilience is an important aspect in applications where meteorological data may be sourced from external sensors or APIs, which could be offline or malfunctioning. In such cases, the Transformer model would continue to perform effectively, maintaining robust detection capabilities despite the absence of weather information, thereby enhancing its reliability for real-world implementation.

## Acknowledgment

Work partially supported by: (i) the European Commission under the NextGenerationEU program, Extended Partnership PNRR PE1 - ”FAIR - Future Artificial Intelligence Research” - Spoke 1 ”Human-centered AI”, and PNRR - M4 C2, Investment 1.5 ”Creating and strengthening of ”innovation ecosystems”, building ”territorial R&D leaders”, project ”THE - Tuscany Health Ecosystem”, Spoke 6 ”Precision Medicine and Personalized Healthcare”; (ii) the Italian Ministry of Education and Research (MIUR) in the framework of the FoReLab project (Departments of Excellence), in the framework of the ”Reasoning” project, PRIN 2020 LS Programme, Project number 2493 04-11-2021; (iii) the Italian Ministry of Enterprises and Made in Italy , in the framework of the ”Agreements for Innovation” Projet ”4DDS - 4D Drone Swarms” Ref. no. F/310097/01-04/X56. (iv) the Tuscany Region in the framework of the WAU project, PR FESR 2021-2027, Project No. 27716.29122023.042000115 (v) the Digital Republic Fund supported by Google.org in the framework of the wAIne project, CrescerAI national call, Project No. 2023-CRE-00288

## References

[1] A. Vaswani, “Attention is all you need,” *Advances in Neural Information Processing Systems*, 2017.

[2] K. Han, Y. Wang, H. Chen, X. Chen, J. Guo, Z. Liu, Y. Tang, A. Xiao, C. Xu, Y. Xu, Z. Yang, Y. Zhang, and D. Tao, “A survey on vision transformer,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 1, pp. 87–110, 2023.

[3] M. Parola, F. A. Galatolo, G. La Mantia, M. G. Cimino, G. Campisi, and O. Di Fede, “Towards explainable oral cancer recognition: Screening on imperfect images via informed deep learning and case-based reasoning,” *Computerized Medical Imaging and Graphics*, vol. 117, p. 102433, 2024.

[4] M. G. Cimino, G. Campisi, F. A. Galatolo, P. Neri, P. Tozzo, M. Parola, G. La Mantia, and O. Di Fede, “Explainable screening of oral cancer via deep learning and case-based reasoning,” *Smart Health*, vol. 35, p. 100538, 2025.

[5] G. Soyalp, A. Alar, K. Ozkanli, and B. Yildiz, “Improving text classification with transformer,” in *2021 6th International Conference on Computer Science and Engineering (UBMK)*, pp. 707–712, 2021.

[6] T. Guo, C. P. Huynh, and M. Solh, “Domain-adaptive pedestrian detection in thermal images,” in *2019 IEEE International Conference on Image Processing (ICIP)*, pp. 1660–1664, 2019.

[7] G. Fu, Y. Zheng, G. Lei, C. Lu, X. Wang, and T. Wang, “Spindle thermal error prediction modeling using vision-based thermal measurement with vision transformer,” *Measurement*, vol. 219, p. 113272, 2023.

[8] B. Tejedor, E. Lucchi, and I. Nardi, *Application of Qualitative and Quantitative Infrared Thermography at Urban Level: Potential and Limitations*, pp. 3–19. Singapore: Springer Nature Singapore, 2022.

[9] I. Nikolov, M. P. Philipsen, J. Liu, J. V. Dueholm, A. S. Johansen, K. Nasrollahi, and T. B. Moeslund, “Long-term thermal drift dataset,” 2021.

[10] I. A. Nikolov, M. P. Philipsen, J. Liu, J. V. Dueholm, A. S. Johansen, K. Nasrollahi, and T. B. Moeslund, “Seasons in drift: A long-term thermal imaging dataset for studying concept drift,” in *Thirty-fifth Conference on Neural Information Processing Systems*, Neural Information Processing Systems Foundation, 2021.

[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*, pp. 779–788, 2016.

[12] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, “End-to-end object detection with transformers,” in *Computer Vision – ECCV 2020* (A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm, eds.), (Cham), pp. 213–229, Springer International Publishing, 2020.

[13] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, “Densely connected convolutional networks,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[14] Y. Fang, B. Liao, X. Wang, J. Fang, J. Qi, R. Wu, J. Niu, and W. Liu, “You only look at one sequence: Rethinking transformer in vision through object detection,” in *Advances in Neural Information Processing Systems* (M. Ranzato, A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan, eds.), vol. 34, pp. 26183–26197, Curran Associates, Inc., 2021.

[15] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, *et al.*, “An image is worth 16x16 words: Transformers for image recognition at scale,” *arXiv preprint arXiv:2010.11929*, 2020.

[16] M. Parola, H. Dirrhami, M. Cimino, and N. Squeglia, “Effects of environmental conditions on historic buildings: Interpretable versus accurate exploratory data analysis,” in *Proceedings of the 12th International Conference on Data Science, Technology and Applications - Volume 1: DATA*, pp. 429–435, INSTICC, SciTePress, 2023.

[17] C. H. Bahnsen and T. B. Moeslund, “Rain removal in traffic surveillance: Does it matter?,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 8, pp. 2802–2819, 2019.

[18] W. Wei, D. Meng, Q. Zhao, Z. Xu, and Y. Wu, “Semi-supervised transfer learning for image rain removal,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[19] J. Chen, C.-H. Tan, J. Hou, L.-P. Chau, and H. Li, “Robust video content alignment and compensation for rain removal in a cnn framework,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[20] S. S. Halder, J.-F. Lalonde, and R. d. Charette, “Physics-based rendering for improving robustness to rain,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

[21] E. Perez, F. Strub, H. De Vries, V. Dumoulin, and A. Courville, “Film: Visual reasoning with a general conditioning layer,” in *Proceedings of the AAAI conference on artificial intelligence*, vol. 32, 2018.

[22] M. Kieu, A. D. Bagdanov, M. Bertini, and A. del Bimbo, “Task-conditioned domain adaptation for pedestrian detection in thermal imagery,” in *Computer Vision – ECCV 2020* (A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm, eds.), (Cham), pp. 546–562, Springer International Publishing, 2020.

[23] A. S. Johansen, K. Nasrollahi, S. Escalera, and T. B. Moeslund, “Who cares about the weather? inferring weather conditions for weather-aware object detection in thermal images,” *Applied Sciences*, vol. 13, no. 18, 2023.

[24] P. Xu, X. Zhu, and D. A. Clifton, “Multimodal learning with transformers: A survey,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 10, pp. 12113–12132, 2023.

[25] G. Cai, Y. Zhu, Y. Wu, X. Jiang, J. Ye, and D. Yang, “A multimodal transformer to fuse images and metadata for skin disease classification,” *The Visual Computer*, vol. 39, no. 7, pp. 2781–2793, 2023.

[26] D. Min, D. B. Lee, E. Yang, and S. J. Hwang, “Meta-stylespeech: Multi-speaker adaptive text-to-speech generation,” in *International Conference on Machine Learning*, pp. 7748–7759, PMLR, 2021.

[27] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “Bert: Pre-training of deep bidirectional transformers for language understanding,” *arXiv preprint arXiv:1810.04805*, 2018.

[28] M. Parola, “Github weather conditioning transformer code repository, https://github.com/marcoparola/conditioning-transformer,” 2024.

[29] M. Parola, A. Aakerberg, A. S. Johansen, I. A. Nikolov, M. G. C. A. Cimino, K. Nasrollahi, and T. B. Moeslund, “Ltdv2: A large-scale long-term thermal drift dataset for robust multi-object detection in surveillance,” July 2025.

[30] L. Papa, P. Russo, I. Amerini, and L. Zhou, “A survey on efficient vision transformers: Algorithms, techniques, and performance benchmarking,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–20, 2024.

[31] A. S. Johansen, J. C. S. J. Junior, K. Nasrollahi, S. Escalera, and T. B. Moeslund, “Chalearn lap seasons in drift challenge: Dataset, design and results,” in *Computer Vision – ECCV 2022 Workshops* (L. Karlinsky, T. Michaeli, and K. Nishino, eds.), (Cham), pp. 755–769, Springer Nature Switzerland, 2023.