Streetseek - Understanding Public Space Engagement Using Deep Learning & Thermal Imaging

Ciarán O'Mara*, Eoghan Mulcahy*, Pepijn Van de Ven, and John Nelson

University of Limerick, Limerick, V94 T9PX, Ireland

Abstract. In this paper, a platform for analysing public space engagement is described. This research focused on efforts to better understand the various ways people interact with the city environment, for example; the number of persons on a street, the average time spent, and topically - due to Covid-19, the physical distance maintained between people. A novel data collection method was used to capture imagery from several streets in a low-cost, scalable, and privacy ensuring fashion. Insights were captured in real-time over several months on a five-minute interval, for nine hours a day and seven days a week, across multiple cameras. These insights were generated through a novel CNN trained on thermal camera imagery - which maintained the individual's right to privacy by ensuring that no person was identifiable in the captured data-set. Finally, a SORT based tracking algorithm was used to measure interactions over time.

Keywords: Smart Cities · Computer Vision · Data Engineering · Machine Learning · Object Detection · Object Tracking · Thermal Cameras

1 Introduction

The *Streetseek* project has been undertaken in response to an open call as part of the +CityxChange smart city program [1], funded by the European Union's Horizon 2020 research and innovation program. The goal of this program is closely aligned to the UN Sustainable Development Goals (SDG), specifically UN SDG 11 – "Making cities and human settlements inclusive, safe, resilient and sustainable". The development of Streetseek took place over the course of three months from May to August 2020. The idea of data-driven decision making has been around for centuries. However, in recent times advancements in information and communication technology have changed the way in which we use data for policy-making and urban growth [2]. It's essential to understand how we use our cities in order to consistently innovate in urban areas while also ensuring they have the correct facilities and systems to cater for the growing urban populations. In 2019, the United Nations estimated that more than half the world's population (4.2 billion people) now live in urban areas and by 2041, this figure will increase to 6 billion people [3]. The need to capture large scale actionable data has been highlighted further in recent times due to the Covid-19 pandemic. Policy-makers adapt their guidelines and restrictions based on data surrounding positive tests, hospital admissions, and deaths. Although the effectiveness of these guidelines can be inferred by examining these pieces of data there is no data to suggest in real-time how people are adhering to the measures that have been put in place.

A thermal and deep learning technology based platform has been built, capable of gathering real time, actionable insights directly from streets and lane-ways. Although local government are the immediate stakeholders for this type of system, the public and academic researchers will also have interest in this data. Therefore, the insights capture platform is built upon a collaborative, open data platform. This type of design ensures data is easily accessible and therefore can be easily communicated.

2 System Overview

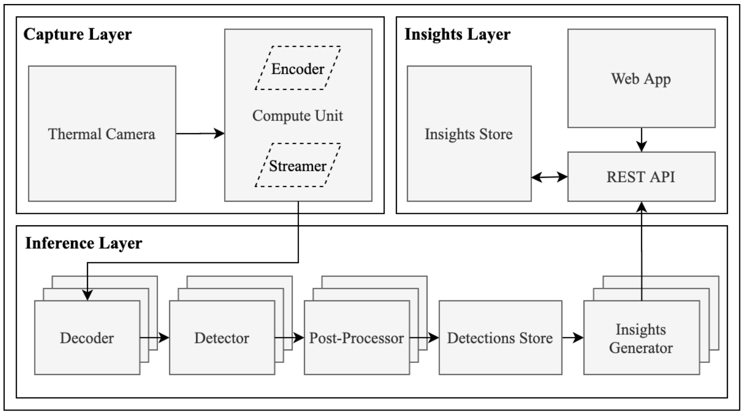


Fig. 1: System Level Diagram

The capabilities of computer vision applications have grown exponentially in recent years. This is driven by advancement in deep learning algorithms allowing for more accurate object detection within imagery across a wider range of environments. Such development has allowed for camera systems to become less passive and evolve into real world sensing tools. One of the major benefits of using a camera system as a sensor is the high resolution of the data collected. Many different insights can be generated through various algorithms depending on the intended use case. Processing of data can take place either at the edge (on the camera compute unit) or in the cloud. Cloud processing was chosen due to hardware limitations in the capture layer . Therefore, the capture layer (see *Fig. 1*) was vastly simplified, requiring only a video encoder and streaming software, resulting in minimal computational specifications.

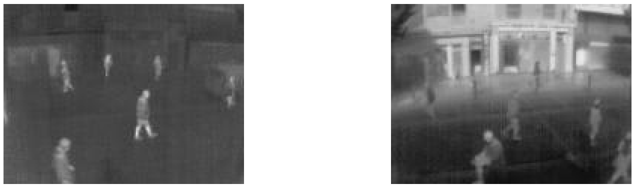
This reduced complexity in the capture layer translates to increased complexity in the inference layer (see *Fig. 1*). If the detection algorithms were running at the edge the upstream packets would be much smaller, and would simply consist of the detection data processed by the edge compute unit. Instead, video data was streamed from multiple cameras up to the cloud at a resolution of 160x120 pixels at 7 frames per second. Therefore, the decoding engine at the front of the inference layer was required to be scale-able on demand to the number of incoming streams. A cluster based approach was used to handle this requirement.

The YOLOv3 [4] single shot detector algorithm (discussed in section 3.2) ran within the detector block. An API endpoint was exposed where decoded images from the decoder engine were forwarded in order to generate the detection data. The detections store (see *Fig. 1*) used a NoSQL database which included tables and items. Primary keys were used to uniquely identify each item in a table and a secondary index was used to provide more querying flexibility. Having generated relevant bounding boxes, this data was used as the input to the insights generator function. The SORT [5] algorithm was used in this block to derive the various insights required of the application. The Insights generator ran on a 5 minute interval querying a batch of data between two timestamps from the detections store and sent a request to an API in order to write the insights to the Insights Store. A REST API was developed to interact with an Insights Store. This API exposed GET and POST requests to access the data contained within the Insights Store.

3 Thermal Person Detection

3.1 Background

Person detection is a well researched problem. However, there are significant privacy concerns surrounding cameras and public spaces. Thermal cameras detect temperature by recognizing and capturing different levels of infrared light, invisible to the naked eye.



(a) No direct sunlight

(b) Direct sunlight

Fig. 2: Street 2, Limerick - direct sunlight affecting natural segmentation.

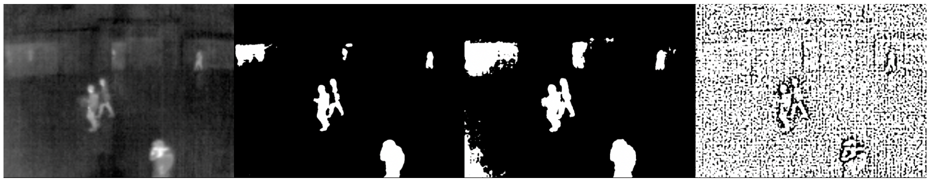
As a result they do not capture details which could be used to identify an individual. This poses a problem, as ‘off the shelf’ human detection models are trained on feature rich RGB images. The algorithm developed to detect humans must rely on foreground/background segmentation, which can vary in different conditions as shown in *Fig. 2*. Furthermore, the cost of thermal sensors is significantly higher than an RGB sensor. In order for this system to be financially feasible a low resolution (160x120) 9fps (frames per second) thermal camera was used. The camera feed was streamed at 4.5fps to reduce computational cost in the cloud.

3.2 Method

Initially what would be referred to as ‘classical approaches’ were used in an attempt to detect pedestrians. A series of thresholding techniques [6] were applied. The first approach involved applying a set thresholding value (as shown in *Fig. 3(b)*) which was calculated through a trial and error process. The subsequent two methods tested were adaptive based thresholding techniques. The first, the Otsu algorithm exhaustively searches for the threshold that minimizes the intra-class variance, defined as a weighted sum of variances of the two classes:

$$\sigma_w^2(t) = \omega_0(t)\sigma_0^2(t) + \omega_1(t)\sigma_1^2(t).$$

The mean adaptive thresholding method involved examining the mean pixel intensity values of the local neighbourhoods of each pixel. Initial results showed that thresholding alone would not suffice as shown in *Fig 3.* This became even more apparent as the scene began to get more complex, with multiple pedestrians and varying environment temperatures. Furthermore, thresholding is incapable of detecting individual pedestrians when they are grouped together.



(a) Original segmentation (b) Global Thresholding (t=80) (c) Otsu Thresholding (d) Adaptive Mean Thresholding

Fig. 3: Thresholding Techniques Results

After it became apparent that thresholding would only work in simple scenarios, efforts pivoted to a background subtraction algorithm. This includes the training of a *background model* which can be subtracted from each video frame resulting in the foreground objects. The following algorithms [7] were tested to access the their suitability for this use case; Gaussian Mixture-based Background/Foreground Segmentation Algorithm (MOG & MOG2), K-Nearest Neighbours background subtraction algorithm, statistical background image estimation and per-pixel Bayesian segmentation algorithm (GMG) and the CouNT high speed background subtraction algorithm (CNT). The results presented in *Fig. 4* were marginally better across frames than the results of the thresholding techniques. However, the top and bottom of people were often split in two.

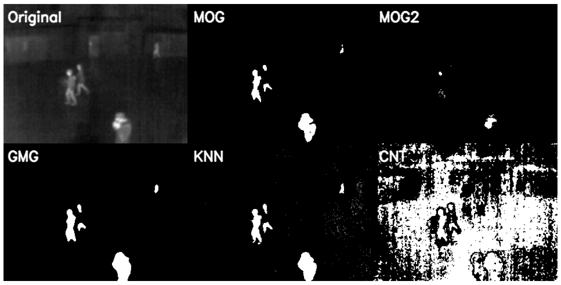
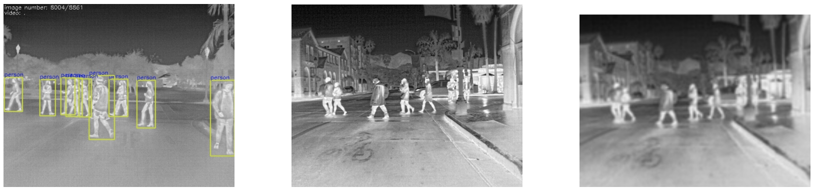


Fig. 4: Background Subtractor Algorithms Results

In an attempt to produce a more robust and reliable thermal pedestrian detector, a deep learning approach was adopted. A relatively lightweight CNN architecture was needed to achieve near real-time inference while also keeping cloud computing costs low. The YOLOv3 architecture [4], presented in 2018 was selected.



(a) Annotated (b) 640x512 (c) 160x120

Fig. 5: FLIR Thermal Dataset

The out of the box YOLOv3 model has been trained on the COCO dataset [8]. In order to train a model capable of classifying pedestrians in a thermal image, a transfer learning approach [9] was used. The final fully connected layers in the model were stripped and output was reduced to 3 classes (person, car, bicycle). The FLIR thermal imagery driving dataset [10] which was used for the transfer learning. The original 640x512 images were initially letter-boxed to convert them to a 4:3 aspect ratio (640x480). A 4x4 kernel was then run across the image, averaging pixels to reduce the image resolution to 160x120 as shown in *Fig 5.* This was necessary for transfer learning as the training images needed to be of the same resolution as the street camera. Finally, by inspection it appeared that any detection who's bounding box was <20px^2 was discarded since it appeared as noise.

Dataset	Images	# of Person	# of Car	# of Bicycle	Model Version
COCO	328,000	900,000	100,000	20,000	baseline
FLIR	10,228	28,151	46,692	4,457	v1
Street Cameras	922	1,030	0	0	v2

Table 1: Descriptions of datasets used to train each model

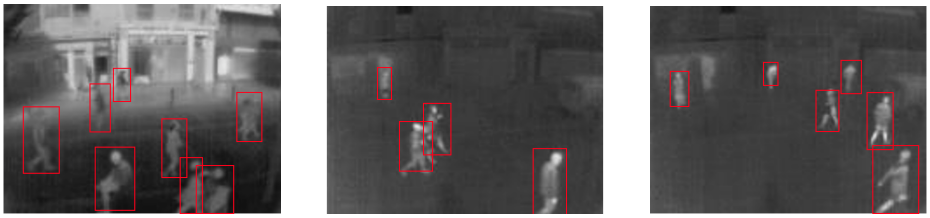
Test and training scripts were used to evaluate and generate new iterations of the model. The original version of YOLO trained on RGB images was used as a baseline. As expected the performance of this model on the captured 160x120 thermal dataset was poor. A transfer learning technique was used to generate a new model based on the FLIR [10] thermal dataset which comprises bicycles, cars and people. This dataset came pre-annotated and following down-sampling to 160x120 was trained for 50 epochs. Performance increased slightly on this iteration as the model (v1) became more familiar with thermal data. However, the model was still not performing acceptably as it had not seen data from the 160x120 street cameras. The model was then fine-tuned (initialised with v1 weights) with an annotated dataset consisting of 922 thermal images from the Street 1 and 2 cameras, for a further 50 epochs - after which it achieved performance metrics shown in *Table 2.* This model (v2) was chosen for deployment.

Model	F1	mAP@50	Precision	Recall
baseline	0.3360	0.1990	0.9530	0.2040
v1	0.3630	0.2200	0.9500	0.2240
v2	0.8897	0.9471	0.8904	0.8890

Table 2: Model metrics using the Street 1 validation set

4 Insights Generation

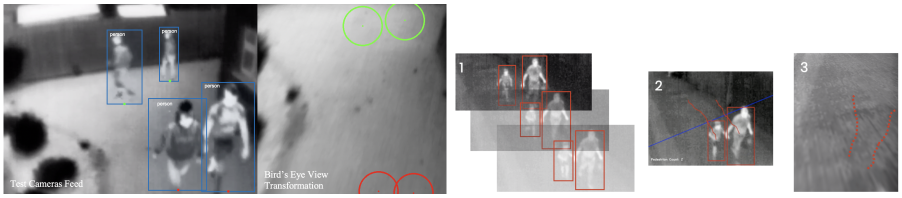
After detecting a pedestrian in an image, in order to understand their behaviour and interaction with both the public space and others they must be tracked



(a) Miscount by 3 (b) Miscount by 1 (c) Correct Count

Fig. 6: Street 2, Limerick - YOLOv3 thermal person classifier results.

across video frames. In order to track pedestrian centroids in a 2D space the SORT (Simple Online Real-Time Tracking)[5] algorithm was implemented. A newer version of this algorithm built upon a deep learning architecture called DeepSORT was also implemented. This architecture adds a pre-trained neural net to generate features for objects. However, the computational cost of running DeepSORT was not required for this use-case and so its less computational predecessor SORT was chosen. The SORT algorithm uses a Kalman filter for object tracking. The Kalman filter is also known as linear quadratic estimation (LQE) algorithm that uses a series of detection centroids over time to produce an estimate as to where the next centroid will be. The SORT algorithm then uses IOU (intersection over union) criteria to accept the estimate. The critical aspect of this algorithm is the association of objects between frames. IOU is not a good approach for small objects as there is naturally less of an overlap of their bounding boxes. DeepSORT [11] addresses this issue by adding a pre-trained neural network to generate features for objects. Using this method the association can be made based on feature similarity instead of overlap. Although DeepSORT offers improvement on overall accuracy when compared with SORT, it comes at computational cost. In this case SORT was used to speed up cloud processing time which contributes to keeping system costs down.



(a) Social Distancing Compliance (b) Tracking

Fig. 7: Bird's Eye Transformation used in image processing

The calculation of each metric is an extension of the SORT implementation. The measurement distance or speed from a camera feed can be difficult due to perspective, perceived closeness and pixel to distance calibration. The concept of perspective is the idea that humans project the real (3D) world onto a 2D image in order to understand distance and depth. A camera sees the world in the same way and thus the Euclidean distance in the 2D plane is not a good approximation of the 3D or real world Euclidean distance. To solve this a *bird's eye view* virtual camera transform presented in [12] was implemented. The perspective transformation was developed for *camera-on-vehicle* discussing the serious perspective effect on the image caused by the camera angle and height. The same issue is evident in this pedestrian camera feed and therefore the method can be transferred for use in this application.

The scene is transformed as shown in *Fig. 7* and some basic pixel to distance calibration is performed. This allows for the distance between pedestrians to be calculated, as well as the avg speed they move at, the estimated time that they speed in the frame based on the SORT Id assigned to each person and finally a generalised heat-map that can be used to understand how pedestrians use the public space. Furthermore a count line can be positioned in the frame to count pedestrians and the direction that they are walking. The whole tracking process is described in *Fig. 7(b)*.

A total of seven pieces of data were calculated and stored. After a video frame is processed by the YOLOv3 detection model, the pedestrian bounding boxes are stored in a detections database with a unix timestamp linking the data to the frame. Every five minutes an insights generator program fetches the last five minutes of detection data and calculates the metrics. This process is described in *Algorithm 1.*

```
Algorithm 1: Insights Generation Algorithm
 Input: detection_data
 Output: insights_object
insights_object
 ← {"personCountLeft" : 0, "personCountRight" : 0, "avgSpeed" :
 0, "estTimeSpent" : 0, "socialDistCompliance" : 100, "heatmap" : [$H_{25\times19}$]}
for frame_detections in detetection_data do
    tracked_objects ← sort_tracker.update(detection_data)
    for x1, y1, x2, y2, obj_id in tracked_objects do
        feet ← $(x1 + (bbox_w/2)), (y2)$
        feet_transformed ← $perspective\_transformation(feet)$
        object_paths[obj_id].append(feet_transform)
        insights_object ← $update\_insights\_metrics(object\_paths)$
    end
end
return insights_object
```

5 Results & Discussion

5.1 Thermal Pedestrian Model and Counter

The three models presented in *Table 3* were tested using some of the captured imagery from the Street 1 camera. The test dataset included 804 frames (half in direct sunlight half in shade) with a total of 72 people counted across the frame sequence and 6480 pedestrian instances. The recall metric $(TP/TP + FN)$ was used to evaluate how well the models performed as for this particular use case there is only one class and for tracking false positives are not a concern.

Model Name	Recall (instances)	Recall (crossing line)
baseline	0.19	0.08
v1	0.35	0.28
v2	0.82	0.89

Table 3: Deployment dataset model results

It was clear that for this test data *Model v2* out performed the other two models as shown in *Table 3.* The training data used for *Model v2* was very similar to that of the test set, including imagery from both streets. All three models struggled with the direct sunlight frames as well as instances where pedestrians overlap and there was very little contrast to segment them.

5.2 Insights Generated

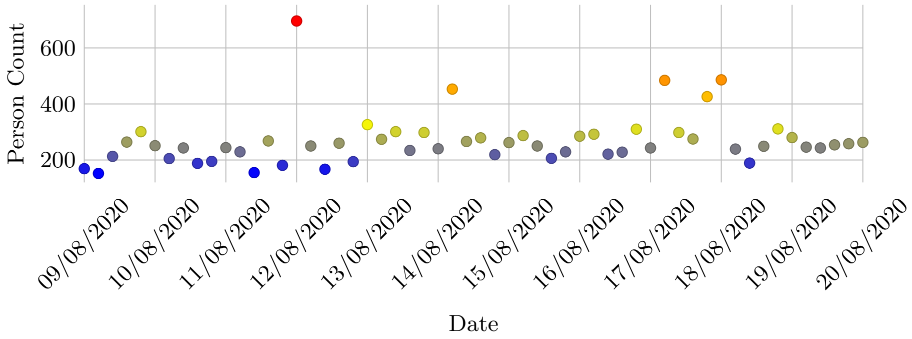


Fig. 8: Daily Person Count on Street 1

The main objective of the captured insights was to communicate the data in the hope of starting discussions which can in some cases lead to positive change in the city. The insights data was captured at a 5 minute level of granularity and can be queried through a REST API. The person count (over a 2 month period) for Street 1 is shown in *Fig. 8*.

The first spike of people on the 2^{nd} of September and subsequent spikes thereafter were as a result of renovation work being carried out and workers walking up and down the laneway throughout the day. This camera was installed to measure the impact that these installations had on pedestrian footfall. It could be argued that the marginal increase in pedestrian traffic seen after the initial spike was as a direct result of the renovation work on the laneway.

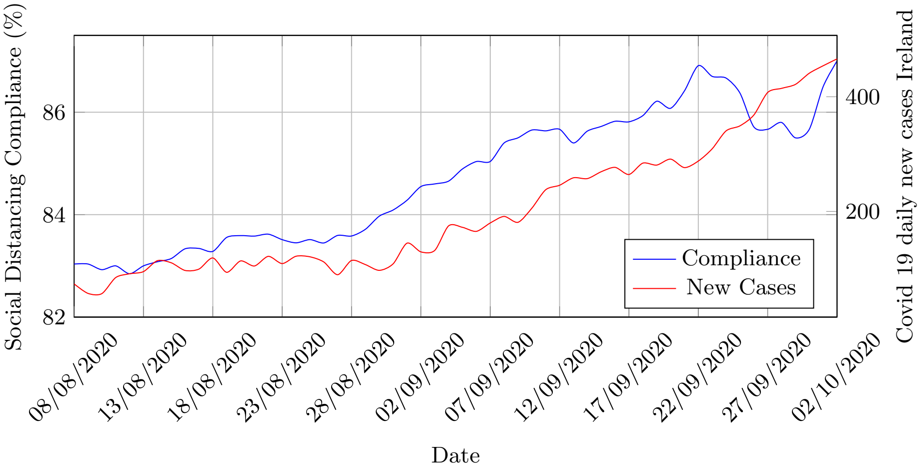


Fig. 9: 5-day Rolling Average - Social Distancing Compliance on Street 1 and New Covid-19 Cases in Ireland

The social distancing compliance on Street 1 is presented in *Fig. 9* The question which could be posed here is whether the media surrounding the increase in Covid-19 case numbers in Ireland resulted in better social distancing compliance in this laneway. Furthermore, measures could be introduced in this laneway to keep pedestrians distanced and monitored in real-time using this system.

The heatmap overlay is presented in *Fig 10.* It would seem that the left-hand side of the laneway is more popular, as is the top of the frame where there is a seating area and an entrance to a café. This would suggest that the lane is predominantly used to access the café and could be used to start a conversation surrounding the pedestrianization of this laneway.

Fig. 10: Street 1 Heatmap Overlay

6 Future Work

Future research should further improve the accuracy of the thermal person detection models, and also could examine how accurate the model is in detecting cars and bicycles. The detection of cars and bicycles will offer more insight into how urban spaces are used. Furthermore, there is scope to examine how cloud computing costs could be minimised by potentially using an intermediate background subtraction layer to identify movement, before passing a frame to the inference layer. Finally, over 4.5 million thermal images from several streets have been captured and stored as part of this research. Future work will also include the annotation of a large 160x120 thermal imagery dataset.

7 Conclusion

To conclude, a thermal and deep learning based platform that uses AI algorithms to collect information on how pedestrians use public spaces has been developed and deployed in Limerick city. The system is capable of counting pedestrians (and their direction of movement), their average walking pace, the estimated time they spend in the frame, their compliance with the social distancing guidelines and a generalised heat-map. These new understandings can be leveraged at city planning level to introduce measures and invest in infrastructure that make urban spaces inclusive, safe, resilient and sustainable. The insights and thermal imagery dataset discussed will be released alongside this paper.

8 Acknowledgements

With thanks to Darryl Connell and Liam Mulcahy. This publication has emanated from research supported in part by a Grant from Science Foundation Ireland under Grant number 18/CRT/6049.

References

1. Home - +CityxChange, https://cityxchange.eu/.
2. World Urbanization Prospects - Population Division - United Nations, https://population.un.org/wup/Publications/.
3. Goi, C.: The impact of technological innovation on building a sustainable city. International Journal of Quality Innovation. 3, (2017).
4. Redmon, J., Farhadi, A.: YOLOv3: An Incremental Improvement, http://arxiv.org/abs/1804.02767, (2018).
5. Bewley, A., Ge, Z., Ott, L., Ramos, F., Upcroft, B.: Simple online and realtime tracking. 2016 IEEE International Conference on Image Processing (ICIP). (2016).
6. Gade, R., Moeslund, T., Nielsen, S., Petersen, H., Andersen, H., Basselbjerg, K., Dam, H., Jensen, O., Jørgensen, A., Lahrmann, H., Madsen, T., Bala, E., Povey, B.: Thermal imaging systems for real-time applications in smart cities. International Journal of Computer Applications in Technology. 53, 291 (2016).
7. Trnovszký, T., Sýkora, P., Hudec, R.: Comparison of Background Subtraction Methods on Near Infra-Red Spectrum Video Sequences. Procedia Engineering. 192, 887-892 (2017).
8. Lin, T., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.: Microsoft COCO: Common Objects in Context. Computer Vision – ECCV 2014. 740-755 (2014).
9. Li, G., Song, Z., Fu, Q.: A New Method of Image Detection for Small Datasets under the Framework of YOLO Network. 2018 IEEE 3rd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC). (2018).
10. FREE - FLIR Thermal Dataset for Algorithm Training — FLIR Systems, https://www.flir.com/oem/adas/adas-dataset-form/.
11. Wojke, N., Bewley, A., Paulus, D.: Simple online and realtime tracking with a deep association metric. 2017 IEEE International Conference on Image Processing (ICIP). (2017).
12. Luo, Lin-Bo Koh, In-Sung Min, Kyeong yuk Wang, Jun Chong, Jongwha. (2010). Low-cost implementation of bird's-eye view system for camera-on-vehicle. ICCE 2010 - 2010 Digest of Technical Papers International Conference on Consumer Electronics. 311 - 312. 10.1109/ICCE.2010.5418845.
13. 1.O'Mahony, N., Campbell, S., Carvalho, A., Harapanahalli, S., Hernandez, G., Krpalkova, L., Riordan, D., Walsh, J.: Deep Learning vs. Traditional Computer Vision. Advances in Intelligent Systems and Computing. 128-144 (2019).