CT6046 - Robotics and Computer Vision

MATEJ KONDROT
s1804317

Semester 2 2021/22

1 Introduction

This is an experimental research paper discussing the state-of-the-art in object detection and tracking algorithms in low-resolution images. Machine vision is a rapidly advancing field, with statistical and deep learning algorithms becoming more accurate and efficient year-on-year. This paper reviews current literature of the topic, critically evaluates two modern object detection algorithms, and comments on core issues and challenges within the field.

2 Literature Review

Object detection and tracking stand as hallmarks of machine vision. They are integral to solving practical and commercial problems in the field of automation and robotics, and have been applied to improve self-driving vehicles (Lapsiya, Jain, Shah, and Kachare, 2021), survelliance (Gawande, Hajari, and Golhar, 2020), sport (Brock, Ohgi, and Lee, 2017) and the retail sector (Fuchs, Grundmann, and Fleisch, 2019).

2.1 Object Detection

Object detection and recognition is a mid-to-high machine vision task (Figure 2.1.1). Objects can be detected by extracting and analysing information extracted from an image. Detected objects are then classified according to data learned specific to the domain.

Object recognition in low-resolution images may be aided by low-level machine vision (Gu et al., 2016), as it helps better identify footage with bad illumination, grain, or artefacts. However, research and development in low-level machine vision falls in the realm of digital image processing and won't be extensively explored in this literature review.

A variety of methods can be used for object detection (Yilmaz, Javed, and Shah, 2006), though modern methods mainly fall into two categories. Feature

Algorithm	Accuracy	Cost	Speed	Note
SIFT	Good	Moderate	Moderate	Illumination variant
SURF	Moderate	Moderate	Excellent	High rotation variance
ORB	Good	Low	Good	Most efficient FDD
AKAZE	Good	High	Moderate	Best FDD accuracy, Open license
YOLOv5	Excellent	High	Low (Excellent with CUDA)	Extensible library
RCNN	Excellent	Moderate	Moderate	Very expensive to train

Table 1: Comparative summary of modern object detection methods (Goel, Sharma, and Kapoor, 2020) (Tareen and Saleem, 2018) (Noman, Stanković, and Tawfik, 2019).

Detection and Description (FDD) algorithms focus on analysing the image through human-tailored, statistical and heuristic processes to determine key points of interest in each image. Convolutional Neural Networks focus on presenting data to a machine learning algorithm and training it. Key points and features are still detected by the algorithm, though connections are more complex.

FDD methods are accurate in situations where the nature of input data is well understood (Beryl Princess, Silas, and Rajsingh, 2020) while neural-nets currently excel at general object recognition.

Additionally, identified objects must be bounded. The simplest and fastest way to identify an object boundary is by drawing a box around its largest areas. The accuracy of the bounding region can be improved by rotating the box (Liu, Wang, Weng, and Yang, 2016) or through statistical methods such as minimum perimeter polygons (Gonzalez and Woods, 2001).

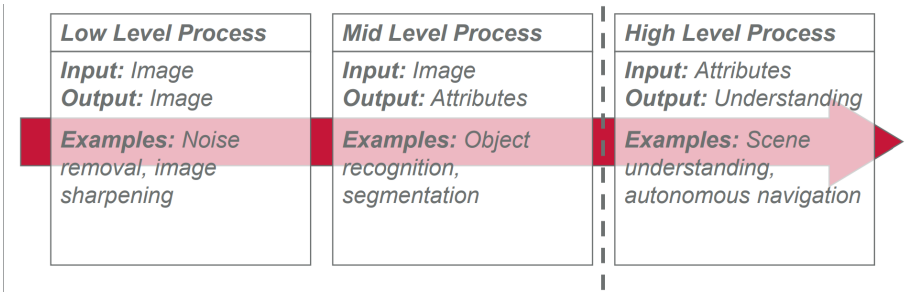

Figure 2.1.1: The use of and examples for each stage of machine learning. From the slides of Stathaki (2013), adapted from (Gonzalez and Woods, 2001)

2.1.1 Feature Detection and Description

As above, Feature Detection and Description (FDD) algorithms detect objects by identifying distinct key-points within images. FDD methods scale in complexity depending on their use-case.

Primitive shapes can be reliably identified by only using corner detection (Kitti, Jaruwan, and Chaiyapon, 2012). For applications which require more than a single feature to be detected, composite transformers must be used.

Combining statistics with strong heuristic methods, Lowe (2004) is able to precisely identify objects through a Scale Invariant Feature Transformer (SIFT).

Object recognition is possible by matching identified features to a database known objects' features. Though SIFT has a high precision for matches, it is computationally expensive compared to other FDD algorithms and its accuracy is low compared to the state-of-the-art (Figure 2). SIFT is seminal in the field of FDD, and often compared against by other algorithms.

2.1.2 Neural Nets

Recent advanced in computing technology have spurred development in deep learning solutions to machine vision problems, including object recognition. Convolutional Neural Networks (CNNs) rely on machine learning to correctly identify objects from large numbers of examples.

Since 2010, convolutional neural networks (CNNs) have outperformed all other methods in large-scale visual recognition benchmarks (Russakovsky et al., 2015). A seminal object recognition model developed by Krizhevsky, Sutskever, and Hinton (2012) achieved high object classification accuracy by training on a large dataset of

Krizhevsky, Sutskever, and Hinton (2012) developed ImageNet, a seminal object recognition model which achieved high object classification accuracy by training in a supervised manner via back-propagation. Girshick, Donahue, Darrell, and Malik (2014) states that object detection performance has plateaued in its past few years, their developed CNN model does not improve raw performance, but has a focus on domain-specific fine-tuning. Dosovitskiy, Springenberg, Riedmiller, and Brox (2014) explores unsupervised feature learning: although it does not currently achieve exceptional accuracy, the research has a high commercial value as there is a much larger amount of unlabelled data in real world applications.

State-of-the-art object recognition methods implement 'Very Deep' Convolutional Neural Networks (Simonyan and Zisserman, 2015). These networks are able to detect and classify objects with great precision, at the cost of real-world performance, slow training times and low learning rates. He2016DeepRL introduce deep residual learning, which allows for faster learning rates even with greatly increased depth.

Recent efforts explore consistent solutions which can be applied in a real-time environment. Instead of raw pixels, Gupta, Sunita, and Singh (2018) performs CNN object detection using a Histogram of Gradients descriptor; this results

in a high classification accuracy with low CNN complexity, since the size of the input layer is greatly reduced. Rashid et al. (2020) research sustainable object detection systems, which can maintain performance across the changing nature of an object.

2.2 Object Tracking

While objects may be tracked through continual detection, object tracking methods concern themselves with using temporal information to maintain an object's identity over several frames of footage (Yilmaz, Javed, and Shah, 2006). The tracking of objects can be initialised manually or be automated with the aid of object detection. Table **??** compares the merits and drawbacks of each method. Automated initialisation is more prominent in research as it allows for continuous tracking of new objects appearing on a screen (Luo et al., 2021).

A keystone for modern object tracking, Hare, Saffari, and Torr (2011) propose a structured tracking algorithm (Struck) which unifies object detection and tracking by learning a single object prediction function from a small number of samples. The predictive function allows for objects to be tracked at a high accuracy when they move through an occluded region. Though its flexibility and relatively high accuracy has made it a staple model in visual object tracking comparisons, modern architectures now outperform Struck in terms of both speed and accuracy (Bertinetto, Valmadre, Henriques, Vedaldi, and Torr, 2016) (Kristan et al., 2016). As with object recognition, CNN models have come to dominate general tracking methods.

> *"All top-performing trackers [in the Visual Object Tracking Challenge 2016] applied convolutional neural-net features [with a] different localization strategy"* Kristan et al. (2016)

Tracking systems implementing convolutional neural-nets (CNNs) have been able to achieve both very good accuracy in real-time frame-rates (Bertinetto et al., 2016) (Held, Thrun, and Savarese, 2016), and outstanding accuracy at lower frame-rates (Nam and Han, 2016). Zhang, Peng, Fu, Li, and Hu (2020) propose a state-of-the-art CNN which achieves a high accuracy and robustness while being inexpensive through the use of anchor-free object tracking.

3 Core Issues and Challenges

Although progress in machine vision is rapid, core issues exist in both object tracking and detection which prevent any algorithm in being perfect, or even best-in-class for every scenario.

Both traditional and CNN object detection and tracking methods face core issues in real-world applications.

'Traditional', handcrafted object detection methods employing feature extraction, such as SIFT and HOG, are able to provide a robust representation of an object; however, they are inherently shallow as they can't capture the variety

of object appearances, illumination conditions and backgrounds found in the real world (Zhao, Zheng, Xu, and Wu, 2019).

Object tracking systems face core challenges relative to their field such as frequent occlusions (Yang and Nevatia, 2012), the initialization and termination of object tracks, and interactions between multiple objects (Luo et al., 2021).

Convolutional Neural Networks are able to achieve state-of-the-art performance through training on large datasets of labelled, bounded boxes (Bertinetto et al., 2016). This may not always be viable due to dataset or computational limitations. Unsupervised learning approaches allow for much more data to be processed, but the issue of sourcing and processing data remains largely the same (Dosovitskiy et al., 2014). It takes time and resources to train your own CNN from scratch, which which is often only viable for industry-leading organisations. While efforts have been made so simplify CNN architecture (Gupta, Sunita, and Singh, 2018), better results are directly linked to the complexity of the model (He, Zhang, Ren, and Sun, 2016). A complex model results in task-specific 'fine-tuning' becoming more challenging (Girshick et al., 2014).

Although classification methods have surpassed the abilities of most humans (Krizhevsky, Sutskever, and Hinton, 2012), fine-grained object detection is prone to error. Fuchs, Grundmann, and Fleisch (2019) discusses challenging-edge cases in using machine vision systems for product identification, where products with similar packaging and subtle colour differences were difficult for the object detection system to differentiate.

As automotive systems become more prominent and mission-critical, machine vision systems must be sustainable: they must remain performant in a changing environment (Rashid et al., 2020). In other words, a machine vision system should remain equally performant years after its initial deployment. This is a challenge which may become more prevalent in the future of robotics and machine vision.

4 Experimental Comparison

This section will focus on the experimental aspect of the research paper, comparing the performance of two state-of-the-art machine vision algorithms in low-resolution images.

The experiment will compare a 'classical', keypoint-based algorithm to an object detection CNN. SIFT and YOLO will be used.

An object recognition CNN algorithm and python package, YoloV5 (You Only Look Once), was chosen as the second object recognition algorithm. It boasted state-of-the-art object detection as of 2016, and has since been reworked in pyTorch. It offers excellent object detection and an is wrapped in an intuitive library for this comparison.

In order to compare the two algorithms, accuracy, precision and performance are benchmarked against a public object tracking dataset. Accuracy depicts how consistently the object is detected across multiple frames, while precision dictates how many of the detections are within the ground-truth boundary for

the object.

Two low-resolution image sources will be compared in this paper, pedestrians and traffic. Pedestrians are historically a common thing for people to track because of the abundant number of data and papers already existing about it (find that literature review about tracking), while traffic is an equally valuable and growing field in object recognition due to the rising demand for self-driving vehicles.

4.1 Detection with SIFT

The Scale-Invarant Feature Transformer (SIFT) which combines 'hog' with a heuristic tuning approach accurately identify distinct patterns in texture. Object recognition can be achieved by mapping features across a database of pre-classified textures. A comparison of FDD algorithms can be found in Table **??**, though SIFT was chosen as it is a seminal algorithm for keypoint-based detection and is an exemplary algorithm for comparison with CNN.

SIFT analysis was performed using the OpenCV python library, with keypoints matched using the k-nearest-neighbour algorithm. Figure x shows the performance of SIFT when matching distinct and non-distinct objects. Sharp corners and logos are recognised well, while few keypoints are detected in ambiguous shapes.

Unlike pre-trained neural-networks, SIFT requires input examples for the object it is identifying. Object detection precision and accuracy were measured across an object tracking benchmark (Wu, Lim, and Yang, 2013), and three captures of the ground-truth bounding-box were used to create a 'database' of the object. In a real-time system existing footage may not be available, though the SIFT database would be larger and may likely contain two or three matches of objects which look similar.

Figure 4.1.1 includes the collage of three frames used for the pedestrian and motorbike benchmark respectively: one sample is from the beginning, middle, and end of the footage. The three frames were omitted from testing.

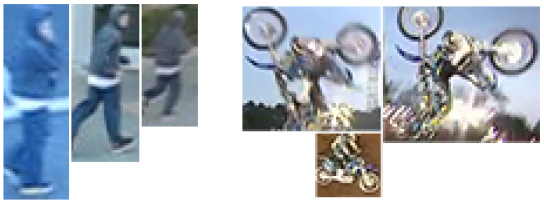

Figure 4.1.1: Object samples used to illustrate a database of objects.

Accuracy and precision for SIFT are measured by analysing the key points which fall within the ground truth bounding-box for each frame. The confidence level is set at 0.75 as it provides good results overall.

$$\text{Accuracy} = \frac{\text{Frames with confident match}}{\text{Total frames}}$$

$$\text{Precision} = \frac{\text{Number of keypoints within bounding box}}{\text{Total matched keypoints}}$$

Performance is measured through the python perf_counter function, with the data logged for each frame as it is iterated over. Data discussed in the results section is analysed using the numPy and Pandas python libraries.

4.2 Detection with YOLO

The You-Only-Look-Once v5 (YOLOv5) framework was chosen as the neural-net counterpart for the comparison experiment. YOLO, first introduced by [textcite] is a deep convolutional neural net framework which outperforms similar algorithms, such as Region-based CNN methods (Girshick, 2015), in real-time scenarios due to using a grid which identifies objects through a single regression problem.

4.2.1 Annotation

The YOLOv5 library works by annotating the bounding boxes of each object it is able to detect. For this paper, the principal code of the library was adjusted to only label the bounding boxes of the object class desired in addition, the co-ordinates of the bounding box were noted for further analysis. Below is a pseudo-code representation of the labelling process for pedestrian tracking.

```
function draw_boxes:
    for each box:
        if box.label is 'person':
            write box.coordinates to file 'yolo\_boxes.txt'
            if box.coordinates are closest to ground\_truth.coordinates:
                draw_box()
```

The annotations were drawn through two passes of the Yolo algorithm; The first pass calculated all the bounding boxes labelled as 'person' for each frame. The bounding box closest to the ground truth was calculated by comparing the euclidean distance of each box to the ground-truth co-ordinates. Figure 4.2.1 shows an isolated bounding box. This isolation process is used for evaluation only and has no impact on real-world use.

4.3 Results

Results were collected by comparing the performance of A metric for the consistency and performance of each algorithm was collected by comparing their

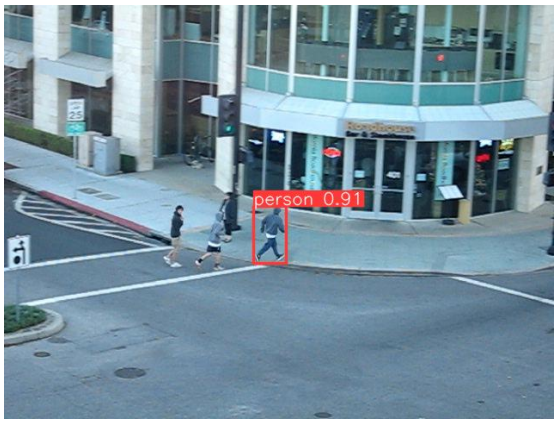

Figure 4.2.1: YOLO pedestrian detection.

performance on a set of public object tracking benchmarks. This allowed the experiment to be domain-specific as a continuous object is being tracked; complementary to the discussion as the images are of low-resolution; and reproducible as all data is publicly available. In addition, extra footage or detection techniques can be implemented without major codebase changes.

The two benchmarks datasets chosen are pedestrian tracking and a motorbike stunt. Performance results are shown in figure 4.3.1. A pedestrian dataset was chosen as it is a well-studied dataset with much commercial use. In this benchmark YOLO achieved a very high precision and accuracy, likely because pedestrian footage formed a large part of its training dataset. SIFT performed much worse in this task, likely due to the pedestrian's movement causing keypoints to change drastically over several frames.

For the motorbike dataset, the SIFT algorithm is able to outperform YOLO in accuracy score. Although YOLO was able to identify the motorbike on most frames, it struggled more than on the prior example as it has likely learned to better recognise motorbikes in traffic rather than ones performing stunts. SIFT performs well likely because the frame of the motorbike remains static, which allows identified keypoints to be detected throughout the length of the footage. In addition, the length of the motorbike footage is only 164 frames compared to the pedestrian's 500; the three sampled frames account for a greater proportion of the overall dataset.

Figure 4.3.2 compares the computational load of each algorithm. While SIFT is considered a slow FDD algorithm, it is able to outperform YOLO on an AMD Ryzen 9 3950X CPU. YOLO also carries a significant RAM overhead, although its speed would likely increase in a GPU-accelerated environment.

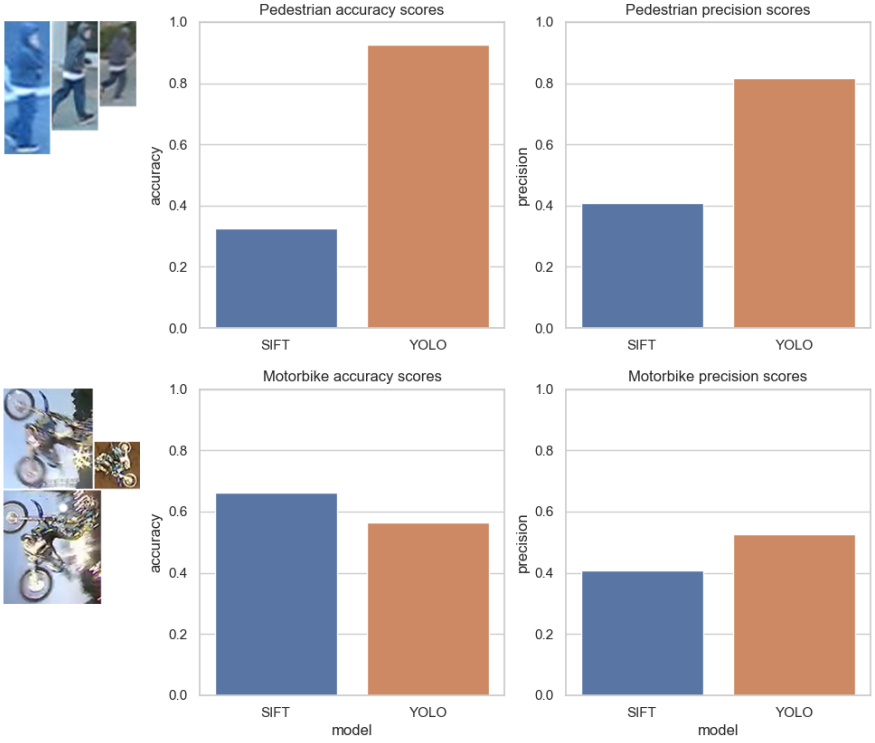

Figure 4.3.1: Accuracy and precision metrics for each dataset and algorithm.

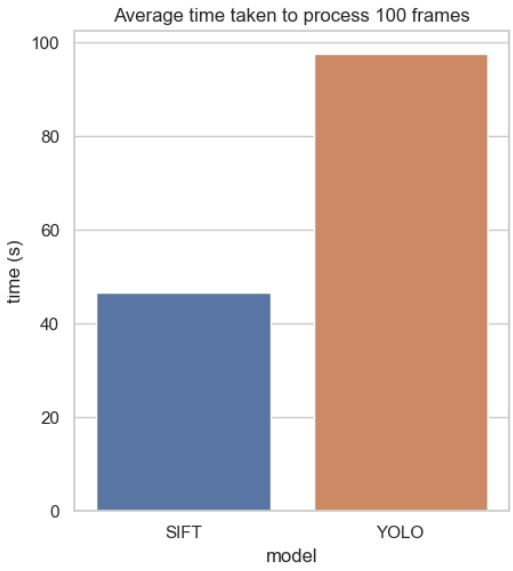

Figure 4.3.2: Time taken to process 100 frames of footage.

5 Conclusion

This paper discussed the development and limitations of modern object-detection and tracking methods. A comparison was made between a seminal Feature-Driven Detector and a Convolutional Neural Net detector, demonstrating that, when trained on appropriate data, CNNs currently achieve state-of-the-art performance in low-resolution image detection. FDDs continue to be powerful models for specific domain applications, and future works may want to explore a comparison between multiple FDD and CNN algorithms in a greater variety of public benchmarks.

References

Bertinetto, L. et al. (2016). 'Fully-Convolutional Siamese Networks for Object Tracking'. *ECCV Workshops*.

Beryl Princess, P. J., Silas, S., and Rajsingh, E. B. (2020). 'Performance Analysis of Feature Detection and Description (FDD) Methods on Accident Images'. *International Conference on Machine Learning, Image Processing, Network Security and Data Sciences*. Springer, pp. 384–396.

Brock, H., Ohgi, Y., and Lee, J. B. (2017). 'Learning to judge like a human: convolutional networks for classification of ski jumping errors'. *Proceedings of the 2017 ACM International Symposium on Wearable Computers*,

Dosovitskiy, A. et al. (2014). 'Discriminative Unsupervised Feature Learning with Convolutional Neural Networks'. *NIPS*.

Fuchs, K. L., Grundmann, T., and Fleisch, E. (2019). 'Towards Identification of Packaged Products via Computer Vision: Convolutional Neural Networks for Object Detection and Image Classification in Retail Environments'. *Proceedings of the 9th International Conference on the Internet of Things*,

Gawande, U., Hajari, K., and Golhar, Y. (2020). 'Pedestrian Detection and Tracking in Video Surveillance System: Issues, Comprehensive Review, and Challenges'. *EngRN: Dynamical System (Topic)*,

Girshick, R. B. (2015). 'Fast R-CNN'. *2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 1440–1448.

Girshick, R. B. et al. (2014). 'Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation'. *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 580–587.

Goel, R., Sharma, A., and Kapoor, R. (2020). 'State-of-the-Art Object Recognition Techniques: A Comparative Study'. *Advances in Intelligent Systems and Computing*. Springer Singapore, pp. 925–932. doi: 10.1007/978-981-15-0751-9_85. Available at: https://doi.org/10.1007/978-981-15-0751-9_85.

Gonzalez, R. C. and Woods, R. E. (2001). *Digital image processing: United States edition*. 2nd edn. Pearson. ISBN: 9780201180756.

Gu, S. et al. (2016). 'Weighted Nuclear Norm Minimization and Its Applications to Low Level Vision'. *International Journal of Computer Vision*, 121, pp. 183–208.

Gupta, V., Sunita, and Singh, J. P. (Aug. 2018). 'Study and Analysis of Back-Propagation Approach in Artificial Neural Network Using HOG Descriptor for Real-Time Object Classification'. *Advances in Intelligent Systems and Computing*. Springer Singapore, pp. 45–52. doi: 10.1007/978-981-13-0589-4_5. Available at: https://doi.org/10.1007/978-981-13-0589-4_5.

Hare, S., Saffari, A., and Torr, P. H. S. (2011). 'Struck: Structured output tracking with kernels'. *2011 International Conference on Computer Vision*, pp. 263–270.

He, K. et al. (2016). 'Deep Residual Learning for Image Recognition'. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778.

Held, D., Thrun, S., and Savarese, S. (2016). 'Learning to Track at 100 FPS with Deep Regression Networks'. *ECCV*.

Kitti, T., Jaruwan, T., and Chaiyapon, T. (2012). 'An Object Recognition and Identification System Using the Harris Corner Detection Method'. *International Journal of Machine Learning and Computing*, pp. 462–465.

Kristan, M. et al. (2016). 'A Novel Performance Evaluation Methodology for Single-Target Trackers'. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38.11, pp. 2137–2155. ISSN: 0162-8828. doi: 10.1109/tpami.2016.2516982.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). 'ImageNet classification with deep convolutional neural networks'. *Communications of the ACM*, 60, pp. 84 –90.

Lapsiya, Y. et al. (2021). 'Analysis Of Various Object Detection Techniques for Self-Driving Cars'. *2021 Asian Conference on Innovation in Technology (ASIANCON)*, pp. 1–6. doi: 10.1109/asiancon51346.2021.9545034.

Liu, Z. et al. (2016). 'Ship rotated bounding box space for ship extraction from high-resolution optical satellite images with complex backgrounds'. *IEEE Geoscience and Remote Sensing Letters*, 13.8, pp. 1074–1078.

Lowe, D. G. (2004). 'Distinctive Image Features from Scale-Invariant Keypoints'. *International Journal of Computer Vision*, 60, pp. 91–110.

Luo, W. et al. (2021). 'Multiple object tracking: A literature review'. *Artif. Intell.*, 293, p. 103448.

Nam, H. and Han, B. (2016). 'Learning Multi-domain Convolutional Neural Networks for Visual Tracking'. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4293–4302.

Noman, M., Stanković, V., and Tawfik, A. (2019). 'Object Detection Techniques: Overview and Performance Comparison'. *2019 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)*, pp. 1–5.

Rashid, M. et al. (2020). 'A Sustainable Deep Learning Framework for Object Recognition Using Multi-Layers Deep Features Fusion and Selection'. *Sustainability*, 12, p. 5037.

Russakovsky, O. et al. (2015). 'ImageNet Large Scale Visual Recognition Challenge'. *International Journal of Computer Vision (IJCV)*, 115.3, pp. 211–252. doi: 10.1007/s11263-015-0816-y.

Simonyan, K. and Zisserman, A. (2015). 'Very Deep Convolutional Networks for Large-Scale Image Recognition'. *CoRR*, abs/1409.1556.

Stathaki, T. (2013). *Digital Image Processing*. URL: https://www.commsp.ee.ic.ac.uk/~tania/teaching/DIP%202014/DIP-Introduction%20Lecture%2013-10-14.pdf. (Accessed: Apr. 14, 2022).

Tareen, S. A. K. and Saleem, Z. (Mar. 2018). 'A comparative analysis of SIFT, SURF, KAZE, AKAZE, ORB, and BRISK'. *2018 International Conference on Computing, Mathematics and Engineering Technologies (iCoMET)*. IEEE. doi: 10.1109/icomet.2018.8346440. Available at: https://doi.org/10.1109/icomet.2018.8346440.

Wu, Y., Lim, J., and Yang, M.-H. (2013). 'Online Object Tracking: A Benchmark'. *2013 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2411–2418.

Yang, B. and Nevatia, R. (2012). 'Online Learned Discriminative Part-Based Appearance Models for Multi-human Tracking'. *ECCV*.

Yilmaz, A., Javed, O., and Shah, M. (2006). 'Object tracking: A survey'. *ACM Comput. Surv.*, 38, p. 13.

Zhang, Z. et al. (2020). 'Ocean: Object-aware Anchor-free Tracking'. doi: 10.48550/arxiv.2006.10721. Available at: https://arxiv.org/abs/2006.10721.

Zhao, Z.-Q. et al. (2019). 'Object Detection With Deep Learning: A Review'. *IEEE Transactions on Neural Networks and Learning Systems*, 30, pp. 3212–3232.