Meta Learning based Object Tracking Technology: A Survey

Ji-Won Baek[1], and Kyungyong Chung[2*]

[1]Department of Computer Science, Kyonggi University
154-42, Gwanggyosan-ro, Yeongtong-gu, Suwon-si, Gyeonggi-do, 16227, South Korea
[e-mail: jwbaek@kyonggi.ac.kr]
[2]Division of AI Computer Science and Engineering, Kyonggi University
154-42, Gwanggyosan-ro, Yeongtong-gu, Suwon-si, Gyeonggi-do, 16227, South Korea
[e-mail: dragonhci@gmail.com]
*Corresponding author: Kyungyong Chung

Received December 7, 2023 ; revised March 16, 2024; revised May 28, 2024; accepted July 19, 2024; published August 31, 2024

Abstract

Recently, image analysis research has been actively conducted due to the accumulation of big image data and the development of deep learning. Image analytics research has different characteristics from other data such as data size, real-time, image quality diversity, structural complexity, and security issues. In addition, a large amount of data is required to effectively analyze images with deep-learning models. However, in many fields, the data that can be collected is limited, so there is a need for meta learning based image analysis technology that can effectively train models with a small amount of data. This paper presents a comprehensive survey of meta-learning-based object-tracking techniques. This approach comprehensively explores object tracking methods and research that can achieve high performance in data-limited situations, including key challenges and future directions. It provides useful information for researchers in the field and can provide insights into future research directions.

Keywords: Convolution Neural Network, Object Tracking, Meta Learning, Deep Learning, Object Detection

This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT) (No. 2022R1F1A1068828). Additionally, this research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education(2020R1A6A1A03040583).

http://doi.org/10.3837/tiis.2024.08.001 **ISSN : 1976-7277**

1. Introduction

In recent years, image and video analysis technologies have been developing rapidly. As a result, research in detection and object-tracking techniques has become increasingly important. Object detection is the identification of one or more different objects in an image or video, and the location of the object is represented by a bounding box. As object detection technology improves, it is increasingly used in various fields such as autonomous driving, smart farms, robotics, and healthcare. For example, in the field of autonomous driving, object detection technology based on deep learning and machine learning enables cars to recognize their surroundings and accurately identify other vehicles, pedestrians, road signs, etc. In addition, smart farms can monitor crop growth or detect pests at an early stage [1, 2].

Object tracking is used to continuously identify and track specific objects in video or live streaming. It is utilized in various fields such as autonomous driving, security and surveillance, and sports analytics [3, 4, 5]. However, it is difficult to identify each object when the shape of the object changes, the similarity between objects is high, or the object is occluded. In addition, object tracking in real-time applications is computationally expensive.

Meta-learning helps deep learning models optimize their learning process for new tasks based on what they've learned before, and allows models to learn quickly with less data, so they can adapt quickly without the need for large amounts of data and iterative training. As a result, you can expect improved performance with smaller amounts of data collected across industries. In this paper, we analyze a set of methodologies for object detection and tracking techniques based on meta-learning. **Table 1** shows the categorization of the object detection and tracking methods investigated in this paper.

Table 1. Categorization of research methods

Categorize	Methods
Convolution Network based Object Tracking Technology	CNN-based Object Tracking Technology
	Object Tracking using Kernelized Correlation Filters
	SORT-based Object Tracking Technology
	Transformer-based Object Tracking Technology
Meta Learning based Object Tracking Technology	Object Tracking using Model-based Meta-learning
	Object Tracking using Optimization based Meta Learning
	Object Tracking using Distance-based Meta-learning

This paper is organized as follows Chapter 2 describes convolutional network-based object tracking. Chapter 3 describes object tracking based on meta-learning. Chapter 4 concludes.

2. Convolutional Network-based Object Detection and Tracking Technology

2.1 CNN-based Object Tracking Technology

CNN-based object tracking techniques use convolutional neural networks (CNNs) to detect objects in an input image or video. Object detection predicts the location and bounding box of the object. It uses the middle layer of the CNN to extract the features of the detected object. Feature extraction quantifies the object's features and extracts information for tracking. Use the extracted features to track the object in subsequent frames of the video. Typically, an

algorithm is used to compare the features of the object being tracked with the existing image and find the matching features. **Fig. 1** shows the structure of a CNN network.

In general, CNN-based object-tracking technology suffers from the following disadvantages. First, it requires high computational resources and computational costs due to the large neural network structure. Second, if an object temporarily disappears and reappears, the CNN-based tracking algorithm may lose the object. Third, it is difficult to maintain accuracy in the object bounding box when objects are occluded or overlapped. Fourth, CNN models require large amounts of labeled data, which makes data collection and labeling tasks time-consuming and resource-intensive. Finally, it is difficult to track effectively when the size and shape of objects in the data change. Therefore, existing CNN-based object tracking techniques cannot fully address the limitations of correlation filter-based trackers.

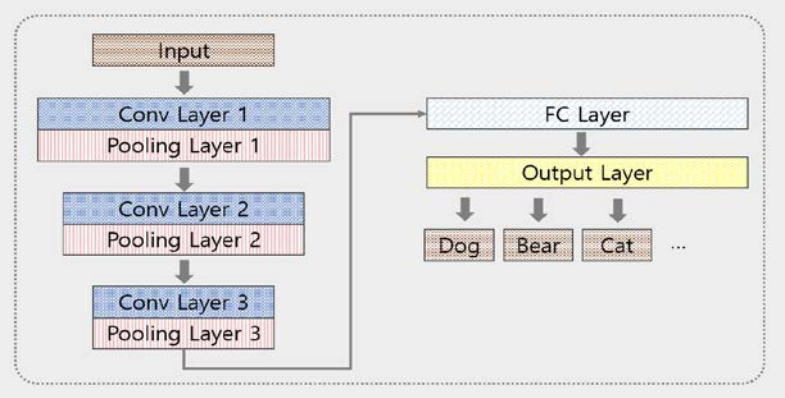

Fig. 1. CNN network architecture

D. Guo et al. [6] proposed a visual tracking method using Siamese full convolutional classification and regression (SiamCAR). SiamCAR is considered to be an improvement of the existing Siamese structure and consists of three steps: feature extraction, bounding box product regression, and object classification. This method improves the performance of object recognition and tracking by enriching the feature plane with deep learning-based representation learning. However, it needs to be improved in terms of speed and has the limitation that it does not fully utilize the information related to movement in consecutive frames.

2.2 Object Tracking using Kernelized Correlation Filters

Kernelized Correlation Filters (KCFs) are an important tool for object tracking in the field of computer vision. KCFs utilize kernel tricks to transform a nonlinear classification problem into a linear classification problem in a high-dimensional space and use the Fast Fourier Transform (FFT) to increase computational efficiency. These features allow KCF to provide fast and accurate tracking while maintaining high frame rates [7]. **Fig. 2** shows an algorithmic flow chart of KCF.

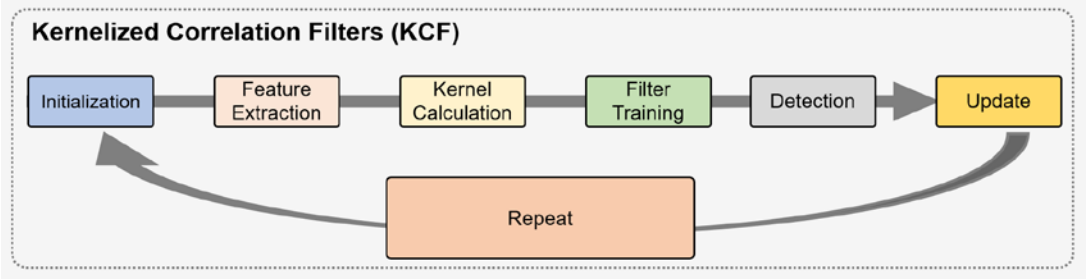

Fig. 2. KCF's algorithmic flow chart of KCF [7]

However, there are still some issues with the KCF technique. The first is that KCF is sensitive to pixel-level changes, making it relatively inaccurate for tracking large or fast-moving objects. The second problem is that KCF is inherently sensitive to rotation and size changes, and tracking can fail if these changes occur in a fast time frame. The third problem is that KCF is inherently optimized for single-object tracking and requires additional algorithms to track multiple objects simultaneously.

Y. Zhou et al. [7] proposed Scale-Adaptive KCF Mixed with Deep Features for Pedestrian Tracking. The proposed method is particularly effective for pedestrian tracking and shows higher accuracy compared to existing methods. This represents an important advance in reducing the sensitivity of KCFs to rotation and size changes while providing scalability to make KCFs applicable to different scenarios and environments. However, it is computationally resource intensive. B. Pu et al. [8] proposed a high-speed tracking with multi-templates correlation filters to track multiple objects simultaneously. The proposed method contributed to improving the tracking performance in various objects and scenarios by proposing a multi-template correlation filter-based tracker. The technique is inspired by various data augmentation methods in deep learning to extend the CF tracker with multiple handcrafted training samples. The method reformulates the original optimization problem and provides a closed-form solution to maintain high-speed computation. Through comprehensive experiments, this research represents an important advance in improving tracking performance and further enhancing tracking performance on different objects and scenarios, further highlighting the versatility and flexibility of KCF. **Table 2** shows a summary of studies using Kernelized Correlation Filters.

Table 2. Summary of studies using Kernelized Correlation Filters.

Method	Characteristics	Advantage	Weakness
Y. Zhou et al. [7]	· Effective for pedestrian tracking. · Reduced sensitivity of KCF to rotation and size changes while providing scalability to adapt KCF to different scenarios and environments.	· High accuracy for pedestrian tracking. · Adaptable to different scenarios and environments.	· Computationally resource intensive, making it difficult to apply in real-time.
B. Pu et al. [8]	· Improved tracking performance across different objects and scenarios	· Track multiple objects simultaneously with improved speed.	· Increased algorithmic complexity due to the use of multiple templates

		· Utilizes data augmentation techniques to improve tracking performance. · Highly flexible for different scenarios and objects.	

2.3 SORT-based Object Tracking Technology

An object tracking algorithm, SORT, is proposed to track multiple objects in a video. SORT (Simple Online and Real-time Tracking) is an algorithm for object tracking that can be used in various fields such as video analysis and autonomous vehicles. SORT works in five steps. First, detect objects in each frame. In order for SORT to work, it must first detect an object. For this purpose, object detection algorithms such as YOLO and Mask R-CNN are used to detect objects. Second, extract the features of the extracted objects. When extracting the features of objects, SORT usually uses CNN to extract the features of objects. Use the extracted features to identify the objects. Third, predict the location and speed of the object based on the location information of the tracked objects. For prediction, the Kalman filter is used. The Kalman filter is a recursive algorithm that uses observations and predictions to estimate the current state. This allows us to predict which objects will be present in the next frame and continue tracking. Fourth, to increase the reliability of object tracking, we calculate the intersection over union (IoU) of objects. IoU measures the detected area of the detected object and the tracked object and matches them as the same object if it is above a certain threshold. This allows us to measure the similarity between the existing tracked object and the newly detected object. In addition, DeepSORT is based on SORT and improves the accuracy and reliability of object tracking through feature extraction using deep learning. **Fig. 3** shows the structural difference between SORT and DeepSORT.

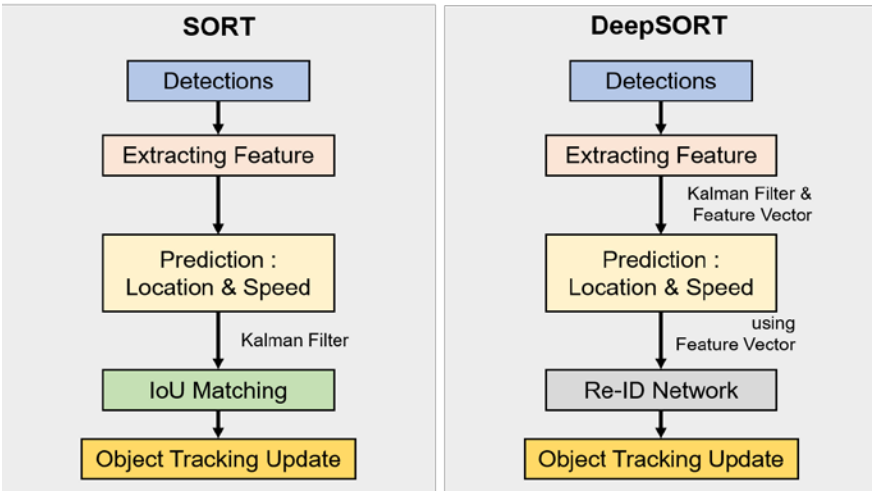

Fig. 3. Comparison of SORT and Deep SORT Structures

M. L. Mokeddem et al. [9] proposed face tracking using DeepSORT. The proposed method uses YOLOv4 to store face images. The stored face data is passed to DeepSORT to track the same person, which can cut and store the face only once per person in the sequence. However, it has the disadvantage of low accuracy. M. I. H. Azhar et al. [10] use DeepSORT to track people. Unlike object detection frameworks such as CNN, the proposed method does not simply detect people in real-time but uses the learned information to track their trajectory until they leave the camera frame. It uses YOLO to detect people and then uses Deep SORT to process the detected people frame by frame to predict their path of travel. Thus, DeepSORT can be used to track people in situations where they are partially or completely occluded for a period of time. **Table 3** shows a comparative summary of SORT-based object tracking studies.

Table 3. Summary of SORT-based object tracking studies.

Method	Characteristics	Advantage	Weakness
M. L. Mokeddem et al. [9]	・Detecting and storing face images using YOLOv4, then tracking the same person with DeepSORT. ・Cropping and storing faces only once per person in a sequence	・Increased efficiency by cropping and saving faces only once in a sequence.	・Tracking is not accurate. ・Tracking is difficult if a face is detected incorrectly.
M. I. H. Azhar et al. [10]	・Track people in real time, even when they leave the camera frame, with learned information about their trajectory. ・Track people in partial or occlusion situations.	・Can detect occluded people. ・Detect and track people in real-time.	・Requires high computational resources for real-time processing and response to occlusion situations.

2.4 Transformer-based Object Tracking Technology

Transformer technology, which emerged for natural language processing, has been applied to the field of computer vision has shown excellent performance, and has been extended to the field of object tracking. transformer-based object tracking technology improves inference speed by inferring detection and object re-identification, i.e., the tracking process, from a single model.

To solve the problem of accuracy reduction caused by performing detection and tracking in one model, Y. Zhang et al. [11] proposed Fair Multi-Object Tracking (FairMOT). The proposed model improves tracking robustness by balancing the detection branch and re-ID branch without using anchors. The detection branch consists of three heads that predict the heatmap, object center offset, and bounding box size. Re-ID branch extracts identity embedding features for each object's location by taking advantage of the similarity of the same object's location in different frames. The features extracted by the two branches are used to perform box linking using standard online tracking algorithms. Experimental results on several benchmark datasets show that FairMOT outperforms other models in terms of accuracy and speed. F. Zeng et al. [12] proposed a Multi-Object Tracking Transformer (MOTR) model. Like FairMOT, the proposed model uses a single-stage method that integrates object detection and object tracking. The proposed model is a framework in which the entire structure is end-to-

end, which reduces the complexity of the existing object-tracking model and improves learning efficiency. Based on the encoder output and the information from the past frame, the concept of Track Query is introduced, which allows the object position of the current frame to be determined and the object ID to be maintained. This enables object tracking to be performed simultaneously with object detection, thereby improving multi-object tracking performance. The model solved the ID Switch problem in which the ID values of two or more objects overlap due to the positions of two or more objects, by discriminating the interaction between the frames. In addition, the evaluation of the MOT dataset shows higher accuracy than other transformer-based models. In addition, object tracking models such as TrackFormer [13,14], which uses DETR (End -to- End to-object detection with Transformers) to detect objects and track queries for tracking, and TransTrack [15], which uses multi-model feature neural networks, have been actively studied. **Table 4** shows a comparative summary of Transformer-based Object Tracking Technology studies.

Table 4. Transformer-based Object Tracking Technology studies

Method	Characteristics	Advantage	Weakness
Y. Zhang et al. [11]	· Constructing three heads for heat maps, object-centered offsets, and bounding box size estimation. · Identity embedding feature extraction for tracking same object location.	· Performs well on a variety of benchmark datasets. · Simple and efficient in structure.	· Highly scenario dependent
F. Zeng et al. [12]	· Determines current frame object position and maintains identity based on coder output and past frame information	· Reduces structural complexity. · Increased efficiency of learning. · Solves the problem of changing ID values between objects	· Requires a lot of computational resources. · Requires optimization for real-time applications

3. Meta Learning based Object Tracking Technology

Meta-learning is a machine learning technique where artificial intelligence models are used to learn new things, and it aims to improve the learning algorithm through different tasks. This allows it to learn quickly on tasks it hasn't learned before. While deep learning typically uses a lot of data to train a model, meta-learning can be done with less data. Also, the performance of a deep learning model depends on many factors. One of the factors is hyperparameters. The proper setting of hyperparameters is a very important contribution to improving the performance of the model. However, finding the optimal hyperparameters requires iterative experimentation and evaluation. This is due to the high time and human cost of data collection, differences in the environment, etc. Meta-learning can solve the problem of insufficient data and insufficient computing resources. In addition, meta-learning is used to improve data sparsity, model complexity, and generalization problems [16].

The representative approaches of meta-learning are distance-based, model-based, and optimization learning methods. To this end, various types of task data are input to the meta-learning algorithm. The results of meta-learning algorithms can be applied to learning algorithms such as supervised learning, unsupervised learning, and semi-supervised learning. As a result, the learning algorithm can predict new task data and test data. This makes it possible to handle various new tasks based on existing experience with less data. Therefore, meta-learning can be applied to model selection, hyperparameter optimization, transfer learning, etc. This allows deep learning models to quickly acquire and apply new knowledge.

Furthermore, meta-learning is categorized into meta-data collection, meta-learner development, and application. The metadata collection stage collects data including features, performance metrics, and hyperparameter settings for each task. It is also used to capture and store knowledge extracted from learning tasks. The meta-learner develops a meta-learning algorithm based on the meta-data. It analyzes the data and explores the relationship between the features and performance metrics of each task to determine the optimal model selection, hyperparameter settings, and learning strategies for other tasks. This improves the performance of the learning algorithm and makes learning for new tasks more efficient.

3.1 Object Tracking using Model-based Meta-learning

Model-based meta-learning can quickly learn new tasks through pre-trained weights and structures. This has the advantage of being able to adapt quickly from previous learning experiences, even when there is little training data available. A representative model-based meta-learning algorithm is Memory-Augmented Neural Networks (MANN) [16]. MANNs learn to solve problems efficiently by storing past data in external memory. This allows it to quickly encode new information and apply it to a new task with only a small number of samples. The components of a MANN are as follows. The first is the controller. The controller is the neural network that performs the main tasks and can be configured as an LSTM or feed-forward network. The controller also processes input data and manages memory. The second element is the external memory. This is a space where data can be stored and retrieved through unique addresses, and data can be accessed and modified efficiently. The last is the memory address mechanism. It enables the interaction between the controller and the external memory, generating and managing the addresses of the memory when the controller writes or reads data. **Fig. 4** shows the structure of MANN.

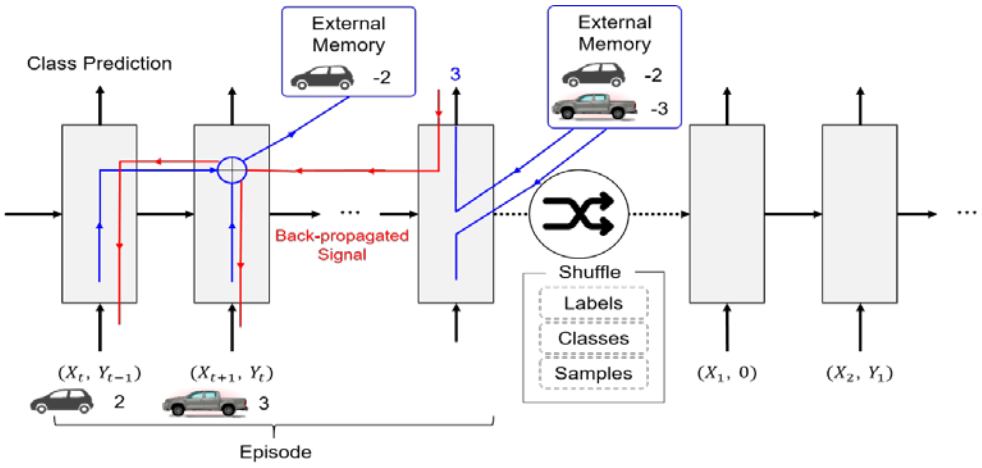

Fig. 4. The structure of MANN [16]

MANN's learning process includes a way for memory to store information continuously so that the appropriate label can be output at a later time. Thus, depending on the training episode, the true label y_t appears with one step offset X_{t+1}, as in (X_{t+1}, y_t). At the previous time step t, the true label is used as input, but at $t+1$ it appears as part of the input. Thus, the MANN behaves in such a way that it can remember information about the new dataset. This means that the memory must remember the input until the label for the current input is output later. This allows it to pull back on previously stored information when making predictions. MANNs can also use memory to store and track important information, solve long-term dependency problems, and can be used for many tasks, such as predicting time series data, translating sentences, and answering questions.

However, there are limitations to model-based meta-learning. The first is the low generalization ability of the model. Since meta-model learning is built on meta-data, it is difficult to generalize to all tasks and situations. The second is data bias. Since meta-models are trained based on data from previous training tasks, the bias of the initial data will affect the performance of the model. In addition, there is the problem of problem complexity. Model-based meta-learning has computational requirements to train the meta-model and make predictions for new tasks. Typically, meta-learning is used to solve complex problems, but its performance is limited for intractable problems. As a result, if the meta-model has a complex structure or uses large amounts of data, it requires the highest computational resources for training and inference and has the disadvantage of providing limited results for very complex problems because there is a limit to what can be inferred.

F. Marchetti et al. [17] solved the multimodal trajectory prediction problem using memory-augmented neural networks. It utilizes a recurrent neural network to learn store and retrieve trajectory embeddings from the past and the future. Next trajectory prediction is performed by decoding in-memory future encodings conditioned on the past. Accordingly, the CNN is learned on semiotic scene maps to incorporate scene knowledge into the decoding state. Therefore, memory growth may be limited.

3.2 Object Tracking using Optimization Learning-based Meta-learning

The model's optimization algorithm and initial weights are learned at the meta-level. This is a method to find the optimal optimization strategy and initial weights that can learn new tasks faster. A representative meta-learning algorithm based on optimization learning is Model-Agnostic Meta-Learning (MAML). MAML adjusts initial parameters to improve the generalization ability of the model to quickly adapt to new tasks. This can improve the generalization ability of most models on small datasets [18, 19]. **Fig. 5** shows the structure and parameter update process of MAML.

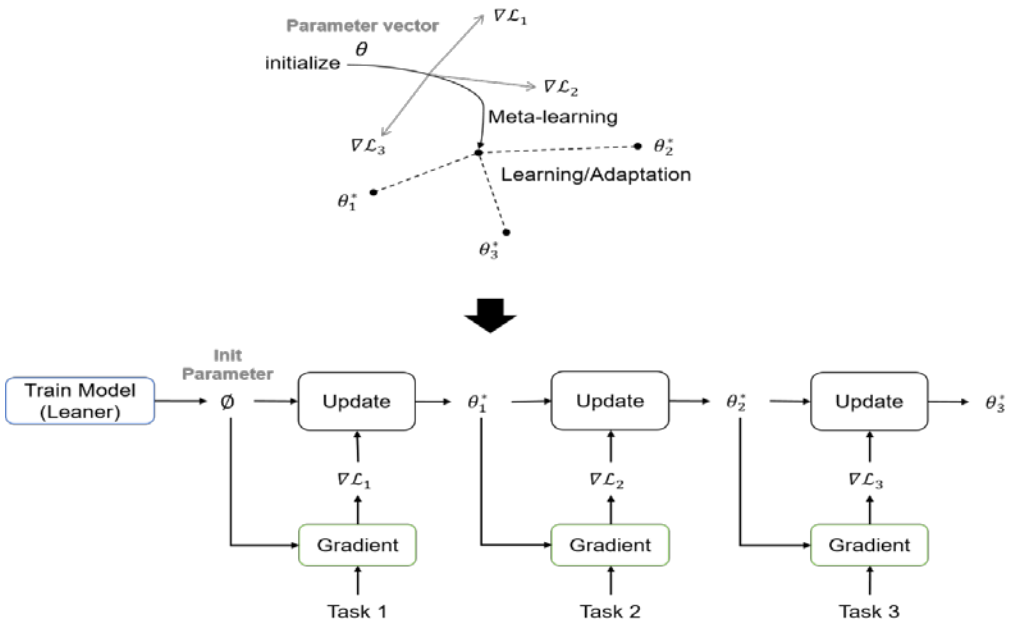

Fig. 5. The structure and parameter update process of MAML

In **Fig. 5**, MAML performs gradient descent to find the parameter Θ of the generalized model. The point of Θ is not optimal for task 1, task 2, and task 3. However, this is a point where you can quickly adapt to task 1, task 2, and task 3. Therefore, the meta-parameter Θ moves to the point (changes shape) indicated by the arrow. Accordingly, an update is performed with the optimal model parameter Θ^* suitable for the new task T_i. This is updated to a value that can minimize the loss of each task through the gradient descent method. Accordingly, it can be generalized through adaptation to new tasks. Therefore, MAML can further improve model parameters and improve generalization ability.

The disadvantages and limitations of MAML are as follows. First, the process of adjusting the initial parameters using gradient descent-based optimization requires a lot of computational cost for training and inference. Second, MAML can quickly adapt to small datasets but depends on the quality and diversity of the data. Accordingly, there is a disadvantage in that it is difficult to learn and adapt appropriate initial parameters when the quality of the data is low or lacks diversity. Third, the influence of initial parameters is strong. Accordingly, inappropriate settings of initial parameters can reduce MAML model performance. Lastly, MAML can generalize to small datasets, but overfitting problems may occur if initial parameters are overly adapted to small datasets.

Z. Li et al. [20] proposed Fast and Robust Visual Tracking with Few-Iteration Meta-Learning. In the proposed method, the base learners are mainly object and background classifiers, and an object bounding box prediction regression network is used here. Also, the main goal of a transformer-based meta-learner is to learn the representations used by the classifier. This can solve the problem of deteriorating real-time performance when there are too many iterations in the model optimization process during offline training or the model update process during online tracking.

3.3 Object Tracking using Distance-based Meta-learning

Distance-based measures the similarity between the support set and the query set. This method is mainly used in tasks that use a small number of training data. The support set is a data set used in the meta-learning process. It usually consists of a few samples of each class, each sample containing information about the class. The support set is used to learn the initial model or base model. Query sets are used to evaluate and measure the performance of meta-trained models. In addition, generalization performance is evaluated. Distance-based meta-learning determines whether two data points belong to the same class. Accordingly, the structure and relationships between data can be considered. Therefore, generalization performance can be improved by grouping similar data and assigning weights according to the distance. Distance-based meta-learning can be effectively used in fields such as image classification, object detection, and natural language processing. Representative models of distance-based meta-learning include Siamese neural network, matching network, and relationship network.

Siamese neural network is a method of learning similarity functions through deep learning methods. It has the following characteristics. First, the Siamese neural network consists of a parallel structure that uses two image data as input. Accordingly, each of the two images is input to one network, and the probability value for the class is used as a weight, and whether or not the two images are the same is determined through the probability value. Therefore, relationships between input data can be learned and similarity measured. Similarity calculates the distance and similarity between input data through methods such as Euclidean distance, Manhattan distance, and cosine similarity distance. It can perform various tasks that measure or compare similarity in data comparison, classification, object detection, etc. Second, the Siamese neural network has symmetry. For example, given two images a and b, $a \circ b$ means that the two image data belong to the same class. The Siamese neural network is symmetric because it uses a distance-based metric. Therefore, $a \circ b$ and $b \circ a$ have the characteristic that they must always be the same. Lastly, the Siamese neural network is useful when training data is insufficient and has the feature of improving generalization ability by learning the relative relationships between data [21]. **Fig. 6** shows the structure of the Siamese neural network.

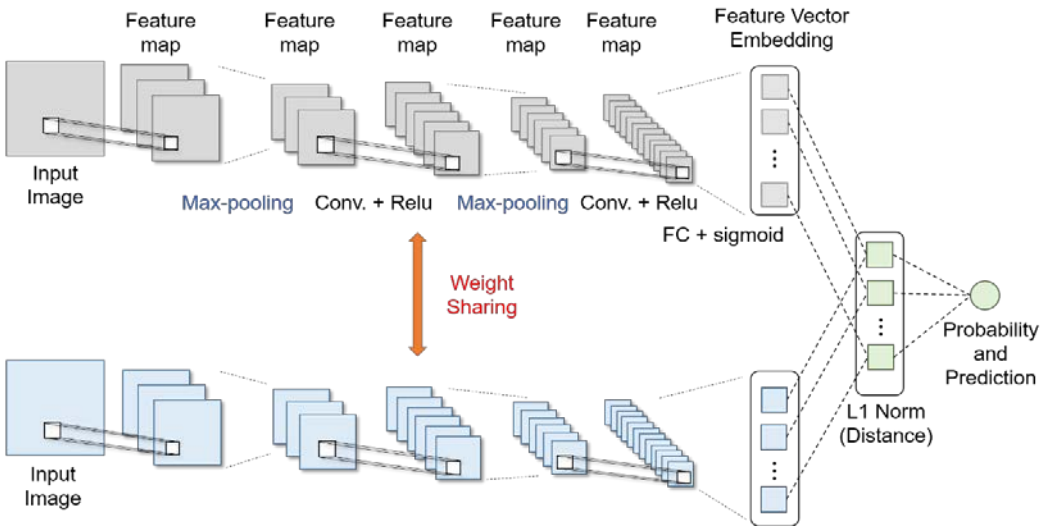

Fig. 6. The structure of the Siamese neural network [21]

In the structure of the Siamese neural network in **Fig. 6**, the L1 Norm is obtained using the features derived from each network. L1 Norm refers to the distance between each feature. Accordingly, single values 0 and 1 are expressed through a fully connected network and a sigmoid function. 1 means the same class, and 0 means the class is not the same. In addition, the loss value is obtained through the binary cross entropy for the predicted value and target appearing in the fully connected network. For example, in the case of a one-shot task, input the test image into one network and input the images included in the support set into the other network one by one. Accordingly, the images with the highest probability value are classified into the same class.

A relationship network is a model that represents the similarity between each piece of data. This is similar to a Siamese network, but there are several differences. The first difference is that in Siamese networks, relationships are extracted through L1 distance. On the other hand, in relational networks, predictions are made through a convolutional classifier. Accordingly, the relationship score is extracted by concatenating the features extracted between the sample dataset used as input and the test data. The second difference is that Siamese networks use cross-entropy as the object function. On the other hand, relationship networks use the MSE Loss function to predict relationship scores that are more suitable for regression than binary classification [22]. **Fig. 7** shows the structure of the relationship network.

In the relationship network structure shown in **Fig. 7**, each data is first embedded. Accordingly, the sample data set and the test data are paired, and the features extracted from the two image data are concatenated and used as input to the relationship module. Therefore, a relationship score is extracted, and similarity is extracted based on the relationship score. Accordingly, similar classes are determined by comparing test data and sample data. This concatenates the embedding value and uses it as an additional input to the relationship module, using the output value as a similarity.

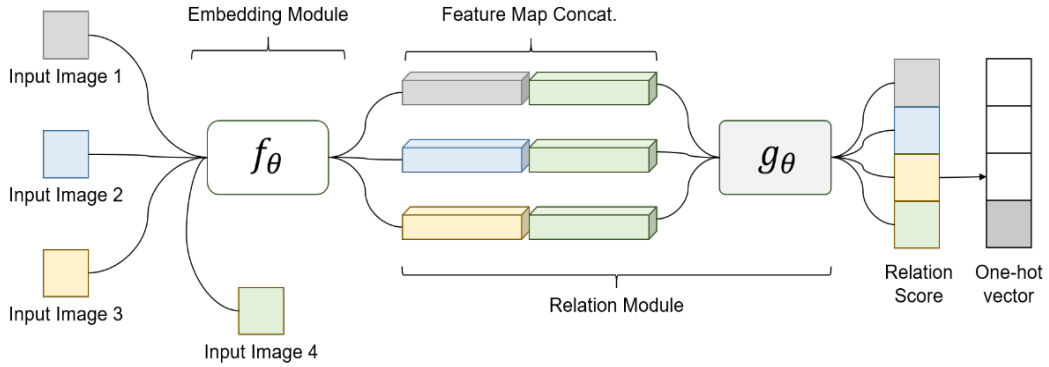

Fig. 7. The structure of the relationship network [22]

A matching network is a method of learning a classifier using a small amount of support sets. The classifier defines a probability distribution for the output labels for a given test sample image [23]. Accordingly, the output of the classifier is defined as the label sum of the support samples with weights applied in the attention kernel, and the value of the attention kernel has the characteristic of being proportional to the degree of similarity between the support set image and the test set image. Additionally, because a non-parametric approach is used, the number of parameters varies depending on the size of the learning data. This has the advantage of being more flexible because it does not assume that the data follows a specific distribution. The goal of the matching network is to find the similarity between the support set

and the batch set through the attention mechanism [24, 25, 26]. Accordingly, argumax (the *x* value that maximizes the function *f(x)*) is applied to set the class with the highest similarity as the label of the batch image. **Fig. 8** shows the structure of the matching network.

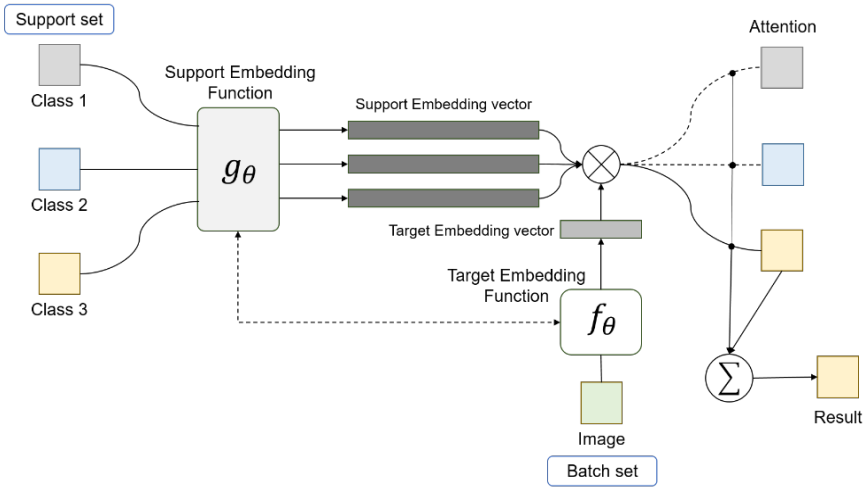

Fig. 8. The structure of the matching network [23]

In the structure of **Fig. 8**, there are images with three classes in the support set. There is also a target image for testing. Through a matching network, the weight is determined through the label and attention value of the batch set, and classification is performed through the probability value for the highest weight.

A. Deng et al. [24] proposed a Slight Aware Enhancement Transformer and Multiple Matching Network for Real-Time UAV Tracking. The proposed SiamSTM can leverage a lightweight transformer to encode strong target appearance features and use a multi-matching network to fully recognize the response map information and improve the tracker's ability to distinguish between targets and backgrounds. Thus, Siamese neural network trackers have solved complex factors such as occlusion, viewpoint change, and interference of similar objects that occur when tracking unmanned aerial vehicles.

4. Conclusion

Recent video-based object detection and tracking using deep learning is not only difficult to identify due to object shape changes, similarity, and occlusion, but also difficult to respond to various situations such as movement or speed differences between objects. Deep learning internally, CNN-based object tracking faces problems such as computational cost, continuous tracking, out-of-plane object tracking, data volume, class imbalance, object size and shape changes, occlusion, and overlap, and existing methods such as SORT Object tracking research does not consider interactions between objects, so there are limitations in accurately detecting slow-moving objects or responding to differences in speed between objects. To solve this, algorithms such as DeepSORT improve accuracy by applying another deep learning model, but problems still exist in aspects such as computational cost, response speed, and continuous tracking limitations due to structural problems. Externally, deep learning-based anomaly detection algorithms have difficulties in securing abnormal pattern data, limitations in adaptability and generalization ability to new abnormal behavior, and limitations due to

computational complexity. Transformer technology is used for object detection in a different approach from existing CNN-based object detection methods and uses a self-attention mechanism to model interactions between features. The self-attention mechanism can identify interactions between objects, enabling more accurate detection that considers dependencies or relationships within the image. However, Transformer-based object detection has higher computational costs and may require a large model size compared to existing CNN-based methods, which increases the computational resources required for learning and inference. Various deep-learning technologies are used to solve these problems. Therefore, in this paper, we investigated meta-learning-based object-tracking technology. This introduced object detection and tracking technology and meta-learning-based object tracking technology.

References

[1] G. I. Kim, K. Chung, "ViT-Based Multi-Scale Classification Using Digital Signal Processing and Image Transformation," *IEEE Access*, vol.12, pp.58625-58638, Apr. 2024. Article (CrossRef Link)
[2] H. Yoo, S. E. Lee, K. Chung, "Deep Learning-Based Action Classification Using One-Shot Object Detection," *CMC-Computers, Materials & Continua*, vol.76, no.2, pp.1343-1359, Aug. 2023. Article (CrossRef Link)
[3] J. W. Baek, K. Chung, "Captioning Model based on Meta-Learning using Prior-Convergence Knowledge for Explainable Images," *Personal and Ubiquitous Computing*, vol.27, no.3, pp.1191-1199, Jun. 2023. Article (CrossRef Link)
[4] J. W. Baek, K. Chung, "Multi-Context Mining-Based Graph Neural Network for Predicting Emerging Health Risks," *IEEE Access*, vol.11, pp.15153-15163, Feb. 2023. Article (CrossRef Link)
[5] B. U. Jeon, K. Chung, "CutPaste-Based Anomaly Detection Model using Multi scale Feature Extraction in Time Series Streaming Data," *KSII Transactions on Internet and Information Systems (TIIS)*, vol.16, no.8, pp.2787-2800, Aug. 2022. Article (CrossRef Link)
[6] D. Guo, J. Wang, Y. Cui, Z. Wang, S. Chen, "SiamCAR: Siamese Fully Convolutional Classification and Regression for Visual Tracking," in *Proc. of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.6269-6277, 2020. Article (CrossRef Link)
[7] Y. Zhou, W. Yang, Y. Shen, "Scale-Adaptive KCF Mixed with Deep Feature for Pedestrian Tracking," *Electronics*, vol.10, no.5, Feb. 2021. Article (CrossRef Link)
[8] B. Pu, K. Xiang, J. Ji, X. Wang, "High-speed tracking with multi-templates correlation filters," *Journal of Electronic Imaging*, vol.30, no.6, Dec. 2021. Article (CrossRef Link)
[9] M. L. Mokeddem, M. Belahcene, S. Bourennane, "COVID-19 risk reduce based YOLOv4-P6-FaceMask detector and DeepSORT tracker," *Multimedia Tools and Applications*, pp.23569-23593, Nov. 2022. Article (CrossRef Link)
[10] M. I. H. Azhar, F. H. K. Zaman, N. M. Tahir, and H. Hashim, "People Tracking System using DeepSORT," in *Proc. of 10th IEEE International Conference on Control System, Computing and Engineering (ICCSCE)*, pp.137-141, 2020. Article (CrossRef Link)
[11] Y. Zhang, C. Wang, X. Wang, W. Zeng, and W. Liu, "FairMOT: On the Fairness of Detection and Re-identification in Multiple Object Tracking," *International Journal of Computer Vision*, vol.129, pp.3069-3087, 2021. Article (CrossRef Link)
[12] F. Zeng, B. Dong, Y. Zhang, T. Wang, X. Zhang, and Y. Wei, "MOTR: End-to-End Multiple-Object Tracking with Transformer," in *Proc. of Computer Vision – ECCV 2022, 17th European Conference, Tel Aviv, Israel*, pp.659-675, 2022. Article (CrossRef Link)
[13] F. Marchetti, F. Becattini, L. Seidenari, A. D. Bimbo, "MANTRA: Memory Augmented Networks for Multiple Trajectory Prediction," in *Proc. of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.7143-7152, 2020. Article (CrossRef Link)
[14] T. Meinhardt, A. Kirillov, L. Leal-Taixé, and C. Feichtenhofer, "TrackFormer: Multi-Object Tracking with Transformers," in *Proc. of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.8844-8854, 2022. Article (CrossRef Link)

[15] P. Sun, J. Cao, Y. Jiang, R. Zhang, E. Xie, Z. Yuan, C. Wang, and P. Luo, "TransTrack: Multiple Object Tracking with Transformer," *arXiv:2012.15460*, 2020. Article (CrossRef Link)
[16] A. Santoro, S. Bartunov, M. Botvinick, D. Wierstra, T. Lillicrap, "Meta-Learning with Memory-Augmented Neural Networks," in *Proc. of the 33rd International Conference on Machine Learning, PMLR*, pp.1842-1850, 2016. Article (CrossRef Link)
[17] F. Marchetti, F. Becattini, L. Seidenari, A. Del Bimbo, "MANTRA: Memory Augmented Networks for Multiple Trajectory Prediction," in *Proc. of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.7141-7150, 2020. Article (CrossRef Link)
[18] T. Hospedales, A. Antoniou, P. Micaelli, A. Storkey, "Meta-Learning in Neural Networks: A Survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.44, no.9, pp.5149-5169. Sep. 2022. Article (CrossRef Link)
[19] C. Finn, P. Abbeel, S. Levine, "Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks," in *Proc. of the 34th International Conference on Machine Learning, PMLR*, pp.1126-1135, 2017. Article (CrossRef Link)
[20] Z. Li, X. Zhang, L. Xu, W. Zhang, "Fast and Robust Visual Tracking with Few-Iteration Meta-Learning," *Sensors*, vol.22, no.15, Aug. 2022. Article (CrossRef Link)
[21] D. Chicco, "Siamese Neural Networks: An Overview," *Artificial Neural Networks*, vol.2190, pp.73-94, 2021. Article (CrossRef Link)
[22] Z. Li, Z. Hu, W. Luo, X. Hu, "SaberNet: Self-attention based effective relation network for few-shot learning," *Pattern Recognition*, vol.133, Jan. 2023. Article (CrossRef Link)
[23] W. Fu, L. Zhou, J. Chen, "Bidirectional Matching Prototypical Network for Few-Shot Image Classification," *IEEE Signal Processing Letters*, vol.29, pp.982-986, Feb. 2022. Article (CrossRef Link)
[24] H. J. Ye, X. R. Sheng, D. C. Zhan, "Few-shot learning with adaptively initialized task optimizer: a practical meta-learning approach," *Machine Learning*, vol.109, pp.643-664. Mar. 2020. Article (CrossRef Link)
[25] S. E. Lee, H. Yoo, K. Chung, "Pose pattern mining using transformer for motion classification," *Applied Intelligence*, vol.54, no.5, pp.3841-3858, Mar. 2024. Article (CrossRef Link)
[26] G. I. Kim, K. Chung, "Augmented and End-to-End Models for Defect Classification of Structures," in *Proc. of DEBS '24: Proceedings of the 18th ACM International Conference on Distributed and Event-based Systems*, pp.183-184, 2024. Article (CrossRef Link)

Ji-Won Baek has received B.S. degrees from the School of Computer Information Engineering, Sangji University, South Korea in 2017. She has worked for Data Management Department, Infiniq Co., Ltd. She has received an M.S. and Ph.D. degrees in 2020 and 2024 from the School of Department of Computer Science, Kyonggi University, South Korea. She has been a researcher at Data Mining Lab., Kyonggi University. Her research interests include data mining, data management, knowledge system, automotive testing, deep learning, medical data mining, healthcare, and recommendation.

Kyungyong Chung has received his B.S., M.S., and Ph.D. degrees in 2000, 2002, and 2005, respectively, from the Department of Computer Information Engineering, Inha University, South Korea. He has worked for the Software Technology Leading Department, South Korea IT Industry Promotion Agency (KIPA). From 2006 to 2016, he was a professor at the School of Computer Information Engineering, Sangji University, South Korea. Since 2017, he is currently a professor in the Division of Computer Science and Engineering, Kyonggi University, South Korea. He was named a 2017 Highly Cited Researcher by Clarivate Analytics. His research interests include data mining, artificial intelligence, healthcare, knowledge systems, HCI, and recommendation systems. Since 2021, he has served as Editor-in-Chief of the Journal of Artificial Intelligence Convergence Technology and as Vice President of the Korea Artificial-Intelligence Convergence Technology Society.