Journal of Big Data

RESEARCH Open Access

Evolutionary computation-based self-supervised learning for image processing: a big data-driven approach to feature extraction and fusion for multispectral object detection

Xiaoyang Shen[1,2], Haibin Li[1,2*], Achyut Shankar[3,6,7,8,9,10], Wattana Viriyasitavat[4] and Vinay Chamola[5]

*Correspondence:
Haibin Li
hbli_ysu@163.com

Full list of author information is available at the end of the article

Abstract

The image object recognition and detection technology are widely used in many scenarios. In recent years, big data has become increasingly abundant, and big data-driven artificial intelligence models have attracted more and more attention. Evolutionary computation has also provided a powerful driving force for the optimization and improvement of deep learning models. In this paper, we propose an image object detection method based on self-supervised and data-driven learning. Differ from other methods, our approach stands out due to its innovative use of multispectral data fusion and evolutionary computation for model optimization. Specifically, our method uniquely combines visible light images and infrared images to detect and identify image targets. Firstly, we utilize a self-supervised learning method and the AutoEncoder model to perform high-dimensional feature extraction on the two types of images. Secondly, we fuse the extracted features from the visible light and infrared images to detect and identify objects. Thirdly, we introduce a model parameter optimization method using evolutionary learning algorithms to enhance model performance. Validation on public datasets shows that our method achieves comparable or superior performance to existing methods.

Keywords Evolutionary computation, Self-supervised learning, Image processing, Big data, Object detection

Introduction

As an important branch in of digital image analysis and pattern recognition, image target recognition detection technology is widely used in many scenarios such as autonomous driving, medical images, industrial inspection, intelligent robots, and intelligent video surveillance [1, 2]. Target detection is to locate and extract the target area of interest in the video or image by analyzing the target feature information, and accurately identifying the target category of each area and its corresponding bounding box. In recent

years, with the continuous breakthrough of deep learning in the field of image processing, object detection technology has also made significant progress [3].

Recently, deep learning has been widely used in the field of computer vision and has made great progress, it has strong learning ability for different levels of image visual features. Specifically, deep learning mainly trains convolutional neural networks for tasks such as object recognition or scene classification by using a large amount of artificially labeled data, so that the network can learn powerful visual representations suitable for image understanding tasks. For example, when the image features learned by the network in this supervised way are transferred to computer vision tasks such as object detection, semantic segmentation, pedestrian recognition, and image retrieval, good results can be achieved. The efficacy of deep convolutional neural networks is heavily reliant on their capacity to learn and the volume of training data available. Given that this data must be manually annotated, one significant drawback of supervised feature learning is the necessity for extensive manually labeled datasets. For instance, to train deep convolutional networks, expansive datasets like ImageNet [4] have been developed, enhancing their performance across numerous vision tasks through the utilization of complex architectures and large datasets. Nonetheless, amassing and labeling these vast datasets require substantial financial and temporal investment. Furthermore, video datasets present even greater challenges in terms of collection and annotation costs compared to image datasets. An example of this is the Kinetics [5] dataset, designed for training convolutional neural networks on video-based human action recognition, featuring 500,000 videos across 600 categories, with each video approximately 10 s in length. The endeavor of gathering and labeling such an expansive dataset demands considerable effort and time from workers.

In addition, there are three challenges to supervised training of neural networks. First, manual annotation of the tens of billions of visual data on the web, whether images or videos, is expensive and infeasible. Secondly, due to the particularity of the industry, the data in some research fields are confidential, and therefore cannot be obtained due to privacy issues. In addition, considering that manual annotation of some data requires professional knowledge, data in some research fields are very scarce. For example, medical data is not only difficult to obtain data information, but also difficult to obtain label information. Thirdly, with the continuous development of researches such as big data, artificial intelligence and deep learning, the types of learning tasks are also gradually increasing. If specific training data and labels must be generated for each new learning task, the research cost is huge and impossible. In the age of smart big data, circumventing the laborious and costly process of data labeling is crucial. Consequently, leveraging the vast quantities of unlabeled data effectively has emerged as a key area of interest among researchers. Here, self-supervised learning, which learns visual features from data without relying on human-provided annotations, presents an effective solution to this challenge.

Operating as an unsupervised technique, self-supervised learning autonomously creates pseudo-labels for unlabeled data, eliminating the need for manual labeling. Then supervised training of the convolutional neural network is carried out to obtain the method of image visual features with better performance. Given only images without any supervision information, it is obviously not known what the goal of the network is to learn. Therefore, how to find an effective information-supervised network learning

is the key to solving the problem of self-supervision. Currently, a widely adopted strategy involves the introduction of various pretext tasks, and automatically generate a kind of pseudo labels for images based on the properties of the proxy tasks, instead of manual labels to supervise network learning. The neural network is trained by learning the objective function of the agent task, and through this process, the features associated with high-level semantic information are learned.

The early self-supervised learning agent tasks, with the introduction of generative adversarial networks, were mainly based on generative models. To generate fake labels for images, the network generates fake samples like the original images during adversarial training. However, with the in-depth research of more scholars, it is found that there are still inevitable problems in this method. First, the model parameters oscillate frequently, making it difficult for the network to converge. Second, the discriminator is overfitting, which prevents the generation network from generating similar pseudo-labels, which in turn prevents learning from continuing. In order to prevent the convergence of the discriminator and the divergence of the generator, it is also necessary to achieve proper synchronization between the discriminator and the generator. Contrastive learning, in contrast to generative models, adopts a discriminative strategy designed to attract similar images closer while pushing dissimilar ones apart. Generation-based self-supervised methods are not effective in solving some downstream tasks. Some works start to divide the image into fixed-size blocks, and let the network predict the position information of the image blocks by shuffling the order, adding occlusion, etc. The purpose is to learn the structure of the context space through the relative position. Some approaches also leverage the color attributes of images by converting them to grayscale. This technique aims to enhance the network's ability to discern subtle color details within the image, thereby minimizing the likelihood of misclassification among categories with similar color profiles. Another popular method is to transform the image, by rotating, shearing, projecting, and other operations on the image, to ensure that the network learns more discriminative and semantically more advanced feature representations. However, most of the existing research focuses on designing various surrogate tasks, and little attention should be paid to what properties the learned image representations should have in order to facilitate the transfer of various downstream tasks.

Currently, most object detection algorithms are mainly based on visible light images. Although the visible light image contains rich texture and detail information, the environment of each target in the actual scene is usually complicated, resulting in occlusion, large scale variation, uneven illumination, and noise interference of the target, which makes the implement of target detection technology still face big challenge [6]. Infrared images mainly use thermal radiation energy for imaging, which is less affected by illumination, but the image contrast is low, and the target texture structure and other features are seriously lost, which greatly limits its application in the field of target detection. However, effectively leveraging this big data for robust object detection remains a challenging problem. Traditional deep learning models can be limited by their reliance on large, labeled datasets and their sensitivity to parameter settings. There is a need for methods that can utilize both labeled and unlabeled data efficiently and adapt to different types of input data, such as visible light and infrared images. Traditional approaches often rely on large, labeled datasets and focus primarily on either visible light images or infrared images [9, 10]. These methods, while effective, face limitations in challenging

environments, such as varying lighting conditions and occlusions. In our paper, the study of a target detection method based on the combination of visible light and infrared images can effectively achieve complementary performance and reduce interference, which will greatly promote the development of target detection technology and the application of practical scenes.

Based on the mentioned above, we study an image object detection method based on self-supervised and data-driven learning in this paper. Specifically, we combine visible light images and infrared images to detect and identify image targets. First, we use a self-supervised learning method to perform high-dimensional feature extraction on image data [7]. Then, we utilize and fuse the extracted features from two kinds of images, namely visible light image and infrared image, to detect and identify objects. Finally, to optimize the model performance, we investigate a model parameter optimization method based on evolutionary learning algorithms to improve the training performance of the model. Our contributions are listed as follows:

- We propose a self-supervised feature extraction and learning method based on autoencoders, which can effectively extract feature information in images.
- We design a model optimization method based on evolutionary computation, which can improve the efficiency of feature fusion from visible light image and infrared image.
- Compared with other methods, our method has achieved certain improvements in performance.

We organize the content of this paper as follows. Section 2 describes related research work on object detection and self-supervised learning. Section 3 presents the proposed object detection model in detail, including self-supervised learning for feature extraction, objection detection based on feature fusion, and model optimization based on evolutionary algorithm. Section 4 details the experimental outcomes, assessing the efficacy of the proposed model in image object detection tasks. Section 5 provides the conclusion of this paper.

Related work

The conventional workflow of target detection algorithms typically comprises steps such as image preprocessing, extraction of candidate frames, feature extraction, target classification, and post-processing. Although it has been widely used in various fields, there are problems such as numerous candidate frames, complex feature design, and poor algorithm migration. In order to alleviate the drawbacks of traditional algorithms, researchers apply deep learning methods to target detection. Through the end-to-end training method, the target detection accuracy is greatly improved.

In recent years, deep convolutional neural networks have achieved significant advancements in the field of computer vision [8]. Object recognition and detection in images have seen significant advancements in recent years, driven by the development of deep learning models and the increasing availability of big data [9, 10]. Nevertheless, the current deep learning models necessitate substantial volumes of labeled training data. In order to avoid manual annotation of large-scale data sets, how to learn semantically rich image feature representation in an unsupervised way has attracted many scholars' attention. Among them, self-supervised learning is a representative method. Self-supervised

learning is a kind of unsupervised learning. The core of it is to design a kind of agent task that only uses the attributes of the data itself and does not use the artificial labels, thus to give false labels to the data. This method is an important strategy to learn data feature representation using pseudo labels. Models trained on agent tasks are versatile and can be applied across various downstream computer vision tasks including classification, segmentation, detection, and retrieval. Furthermore, these applications are not limited to a specific type of data but can extend to images, videos, speech, signals, and text. Self-supervised learning methods generally include four categories, namely image-based generation, context-based prediction, image segmentation, and image transformation.

Generative models, particularly those utilizing generative adversarial networks (GANs), fall under the category of self-supervised learning. However, GANs are often characterized by instability during their training phase. Learning the hidden structural information in images is another important key point in the self-supervised learning surrogate task, and the image inpainting-based surrogate task is designed based on this idea. Pathak et al. [11] proposed an image inpainting technique to predict arbitrary missing regions based on information surrounding the image. Although this method can repair large image deletions and make the restored image conform to the semantics of the entire image, the restored image has the problem of local blurring. Lizuka et al. [12] proposed a new idea to use both global and local discriminators to ensure that the generated images not only adhere to the overarching semantics but also optimize the sharpness and contrast of localized areas. But the disadvantage is that it has not been migrated to other computer vision applications, and there is no comparison. Methods based on super-resolution can enhance low-resolution images to produce higher quality outcomes, leveraging convolutional neural networks. SRGAN [13], a versatile generative adversarial network designed for single-image super-resolution, stands out in this regard. Unlike conventional approaches that rely solely on Mean Squared Error (MSE) loss, SRGAN is capable of restoring the finer details of high-resolution images.

Employing image attributes as supervisory signals, instead of relying on manually annotated labels, enables convolutional neural networks to effectively grasp the semantic nuances of images during the resolution of agent tasks. This, in turn, facilitates the application of these learned features to a broader spectrum of computer vision tasks. Clustering algorithms has been widely used in the task of self-supervised learning of agents based on contextual similarity. In self-supervised scenarios, as a technique for image data clustering, a simple method is to cluster images based on hand-extracted features, such as HOG [14] or Fisher Vector [15]. Aiming at the problem of context similarity, researchers have proposed a series of surrogate task methods in clustering-based self-supervised learning. The K-means based method proposed by Coates et al. [16] firstly extracts some image blocks randomly from the unlabeled image, and then uses the K-means method to learn image features to obtain a data dictionary of image features, and then extracts the complete image features through the dictionary. The algorithm is simple, but the training is done layer by layer rather than end-to-end. Caron et al. [17] proposed a clustering method combined with deep learning. The whole process includes using convolutional neural network to extract image deep features and using K-means to group deep features. To minimize the parameter count, the principal component analysis (PCA) technique is utilized to condense feature vectors down to 256 dimensions through clustering. Following this dimensionality reduction, the clustering outcomes

serve as pseudo labels for updating the network's parameters, enabling the network to predict these pseudo labels. This dual process of parameter update and prediction based on pseudo labels is executed in an iterative manner. This algorithm looks simply, but it can learn some useful general features and achieve better performance than previous unsupervised methods. Different from DeepCluster, the training of convolutional neural network does not use clustering labels, but designs a loss function according to the characteristics of clustering. In the iterative process of network, the learned feature representation is conducive to image clustering. End-to-end optimization is achieved by integrating two processes into one process. Inspired by T-SNE [18], Xie et al. [19] proposed Deep Embedded Clustering (DEC) algorithm, which is not only linear in the number of data points, but also can be easily extended to large-scale datasets.

Addressing the challenge of vast amounts of unlabeled data, crafting an efficient agent task that leverages solely the image's intrinsic visual information stands as a crucial and complex issue within the realm of self-supervised learning. Feng et al. [20] proposed a new self-supervised learning algorithm to help the network learn a feature representation independent of rotation through a rotation prediction task and an instance discrimination task. Zhang et al. [21] proposed an unsupervised representation learning model based on Auto-Encoding Transformation (AET), which takes the prediction transformation as a self-supervised signal to train the model. After encoding and decoding, the reconstructed transformation is obtained. Guo et al. [22] proposed a new method based on Autoencoding Variational Transformations (AVT). Among them, the AET method is used to transform the image using affine transformation and projection transformation, and then a two-way twin network is used based on the original image and the transformed image. AVT attempts to train the network by maximizing the mutual information between image conversion operations and image feature representations. In recent years, self-supervised learning methods based on image transformation have been widely used in other tasks, for example, semi-supervised learning [23, 24]. These tasks usually include two classifiers that share image features, a master classifier, and a classifier for self-supervised tasks. However, some tasks force the primary classifier to be invariant to the image transformation representation of the self-supervised task, and enforcing such invariance may lead to increased complexity of the task. This kind of agent task based on image transformation can be regarded as a kind of unsupervised data enhancement without relying on the annotation of the enhanced samples, which on the one hand expands the range of samples that can be enhanced, and on the other hand increases the scope of application of the transformation.

The image object recognition task can be decomposed into two subtasks: object classification and object localization. Object classification is mainly used to determine whether there is a target in the image and to classify the detected object, while object localization is used to determine the exact position of the detected target in the image. The recognition and localization of image objects are mainly divided into object detection algorithms based on candidate regions and object detection algorithms based on image semantic segmentation. The object detection algorithm based on candidate regions refers to generating candidate boxes in an image and identifying the target by judging the image information within the candidate boxes. Commonly used image information generally includes color, brightness, edges, corners, and texture. Typical traditional classification methods include the scale-invariant feature transform method, the

histogram of oriented gradient feature matching method, and the speeded up robust features method [25, 26]. Taking the scale-invariant feature transform algorithm as an example, the algorithm is used to extract key points in an image, such as corner points and edge points. By detecting key points and extracting description vectors, local feature descriptors are constructed to achieve feature matching. Typical examples of deep learning-based object detection methods include Regional Convolution Neural Network (RCNN) [27], Fast Regional Convolution Neural Network (FAST-RCNN) [28], and YOLO (You Only Look Once, YOLO) [29] object detection networks. The core principle of target detection algorithms leveraging image semantic segmentation is to assign a unique label to each pixel within the image, subsequently classifying these pixels, and ultimately partitioning distinct, meaningful regions of the image into non-overlapping segments for precise target identification and localization. The realm of image semantic segmentation algorithms can be broadly categorized into four main groups: graph-based segmentation, clustering-based segmentation, classification-based segmentation, and hybrid approaches that integrate clustering and classification techniques. Graph-driven methods encompass the image minimum spanning tree algorithm [30], Graph Cuts algorithm [31], and unsupervised Superpixel Lattice segmentation [32]. Clustering-focused algorithms involve Geometric Flows superpixel generation, TurboPixels [33], and Simple Linear Iterative Clustering (SLIC) method [34]. Classification-based segmentation techniques, on the other hand, revolve around categorizing image feature information, exemplified by fully convolutional network (FCN) [35] and U-net deep learning approach [36].

Although most of the current research on object detection is based on visible images, some researchers have explored the fusion detection of visible and infrared images. Xiao et al. [37] used differential maximum loss function to guide the convolution network of infrared and visible light branches to extract target features, and designed feature enhancement and cascading semantic extension modules to improve the detection of targets of different scales. Banuls et al. [38] introduced a target detection algorithm employing decision-level fusion, utilizing an enhanced YOLOv3 network for the separate detection of visible and infrared images before executing a weighted fusion to enhance the detection outcomes. Moreover, evolutionary computation has been recognized for its potential in optimizing deep learning models, particularly in hyperparameter tuning and model architecture search [39, 40]. This approach leverages the principles of natural evolution to iteratively improve model performance. This deep learning-based method for merging infrared and visible light detection has been shown to significantly boost target detection performance. However, most methods extract features respectively and then fuse detection, which fails to make full use of the target features in the two types of images to complement each other.

Proposed method

This section elaborates on the object detection method outlined in this paper, which is anchored in self-learning and data-driven approaches. Our strategy merges visible light and infrared imagery to detect and identify targets within images. By integrating self-supervised learning with autoencoders, we introduce a framework for object detection that emphasizes feature extraction and fusion.

First, we utilize a self-supervised learning method and the autoencoders to perform high-dimensional feature extraction on the image data. Then, we utilize and fuse the extracted features from two kinds of images, namely the light image and infrared image, to detect and identify objects. Finally, to optimize the model performance, we investigate a model parameter optimization method based on evolutionary learning algorithms to improve the training performance of the model.

Self-supervised learning for feature extraction

The traditional method based on generative adversarial network is not only difficult to train, but also not outstanding. In order to fully explore the hidden information of the image itself, the widely used method is to make some appearance transformation of the image. After transformation, data enhancement is realized, and another label is provided, that is, transform set. All kinds of transformations are formed into random combinations so that the network can predict which kind of transformation combinations the image variants have done. However, the problem with self-supervision in this way is that the information learned is mostly global information. If the prediction is made only by appearance transformation, when the appearance of objects is similar and the transformation made is not obvious, the network will not be able to effectively identify them, because of the lack of local information of the target. We propose a self-supervised learning model that combines deep learning with traditional features to solve the problem of inadequate local information learning based on image geometric transformation. Specifically, in the design stage of the agent task, firstly, according to the study of method [41], the image is processed for Angle rotation and color channel ranking. Then, Angle rotation and channel sorting are combined to form the first pseudo-label. Finally, the traditional features of the image are extracted to form a second pseudo-label for self-supervised training. In the test phase, different convolutional layer features are extracted according to different networks and applied to different downstream tasks.

Data processing and translation

We combine traditional methods and deep learning to design a self-supervised learning model to make the model to learn a more semantically rich image feature based on local and global information. Figure 1 shows the feature extraction process based on the autoencoder. The feature extraction process mainly includes the following steps. Given an unlabeled image X, we first preprocess the image with simple cropping, translation, scaling, and flipping. Then, we introduce N different sets of rotation transformation operations S = {g (X, y_i)} (i=1, 2, …, N), and perform N different rotation transformation operations on the unlabeled image X. This operation obtains a total of K variants of the original image, denoted as X_i, and the pseudo-label y is assigned to the transformed image X_i, where i = {1, 2, …, N}, y = {1, 2, …, N}, and the rotation operation of the image can be expressed as follows.

$$g_1(X \mid y_i) = \mathrm{Rot}(X, (y_i - 1) * 90) \tag{1}$$

where $\mathrm{Rot}\ ()$ indicates the rotation operation. In the third step, to better learn the fine-grained color features in the image, we introduce M different channel transformation operation sets g_2, and further perform different color channel transformations on the rotated image variant X, and convert the pseudo label y_l is assigned to the transformed

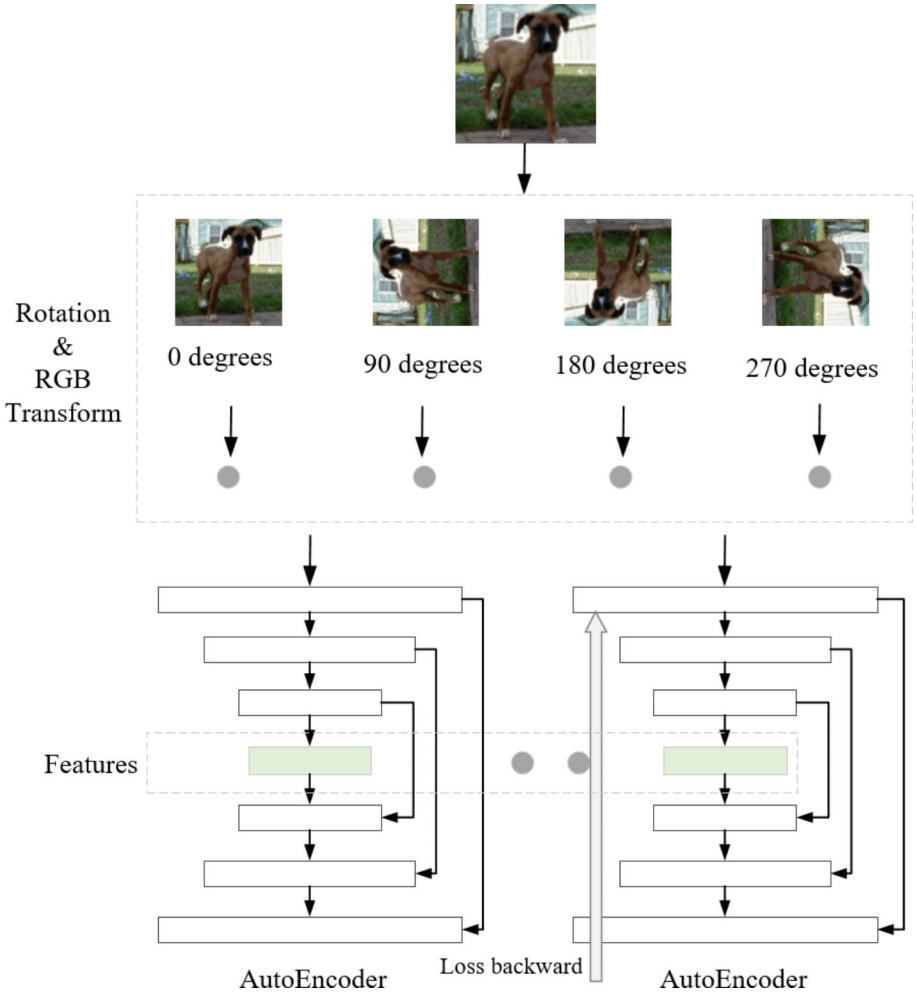

Fig. 1 The process of extracting features using an autoencoder

image, where y = {1, 2, …, M}, and the color channel arrangement operation of the image can be obtained as follows:

$$g_2(X \mid y_i) = \mathrm{Tan}(X, R, G, B) \tag{2}$$

where $\mathrm{Tan}()$ function is a color channel transformation operation, which means that the image is arranged in RGB color channels.

The advantage of image rotation is that it can effectively predict the orientation, structure, and other information of the target in the image. The advantage of image color channel transformation is to capture the color information of the image more accurately. However, after experimental verification, if only these two points are considered, the algorithm is limited to learning the global characteristics of the image target, which causes the problem that the local information is ignored. Specifically, when performing target recognition, the neural network can only predict the global appearance of the target in the image, while ignoring important information such as local edges and textures of the target. For objects of different classes with similar appearance and similar colors, the probability of predicting them as the same class is very high, but the global and color similarity are not necessarily the same, such as wolves and dogs. Therefore, the algorithm leads to certain errors. To more effectively capture the image's local details,

we utilize traditional image features as supervisory signals to aid in the training of the self-supervised network. The local appearance and shape of objects within the image are accurately represented through gradients or the directional density distribution of edges. Therefore, several different traditional feature vectors are extracted in the experiment and used as the second pseudo-label of the self-supervised training image. Among them, the dimension of the pseudo label is set differently according to different datasets.

Feature extraction

Our proposed method based on local and global feature information includes image transformation operations and traditional feature extraction operations. In the network training of self-supervised learning, the learning objective can be constrained by the following objective function until it converges to obtain a better pre-training model, as follows.

$$L = l_1 + \alpha\, l_2 \tag{3}$$

where l_1 is the loss function for image rotation and color channel transformation prediction, and l_2 is the loss function for traditional feature supervised network learning, the parameter α represents the proportion coefficient of deep features and traditional features in the training process. Their definitions are as follows.

$$l_1 = \frac{1}{N}\sum\nolimits_{i=1}^{N} loss(X_i, \theta\,) \tag{4}$$

$$loss\,(X_i, \theta\,) = \frac{1}{NM}\sum\nolimits_{x=1}^{NM} log(g_i\,(X \mid\ y_i)\,, \theta\,) \tag{5}$$

$$l_2 = \frac{1}{N}\sum\nolimits_{i=1}^{N} ||\, X_i - g_2\,(X \mid\ y_i)\,||^2 \tag{6}$$

The specific algorithm flow of self-supervised learning is as follows. First, the unlabeled images in the dataset are subjected to geometric transformation and color channel arrangement. Then, several traditional features of the image are extracted. Finally, the formed three pseudo-labels are combined into two for pre-training of the self-supervised network.

Objection detection based on feature fusion

The architecture of the infrared and visible light fusion target detection network proposed in this study is depicted in Fig. 2. This network is segmented into three main components: the feature extraction module, the feature fusion module, and the detection module. For processing both infrared and visible images, the feature extraction segment features two parallel branches, each with identical configurations. The deep autoencoder is mainly used as the basic unit of feature extraction. Combined with LeakyReLU activation layer, maximum pooling layer and upsampling operation, the feature information of infrared and visible images is extracted efficiently from shallow to deep. The feature fusion module models the features of the pooling layer of the two branches through the linear combination method, and uses the autonomous learning method to realize the information sharing of infrared and visible light, so that the features extracted from the two branches are complementary and the diversity of network features is improved. The

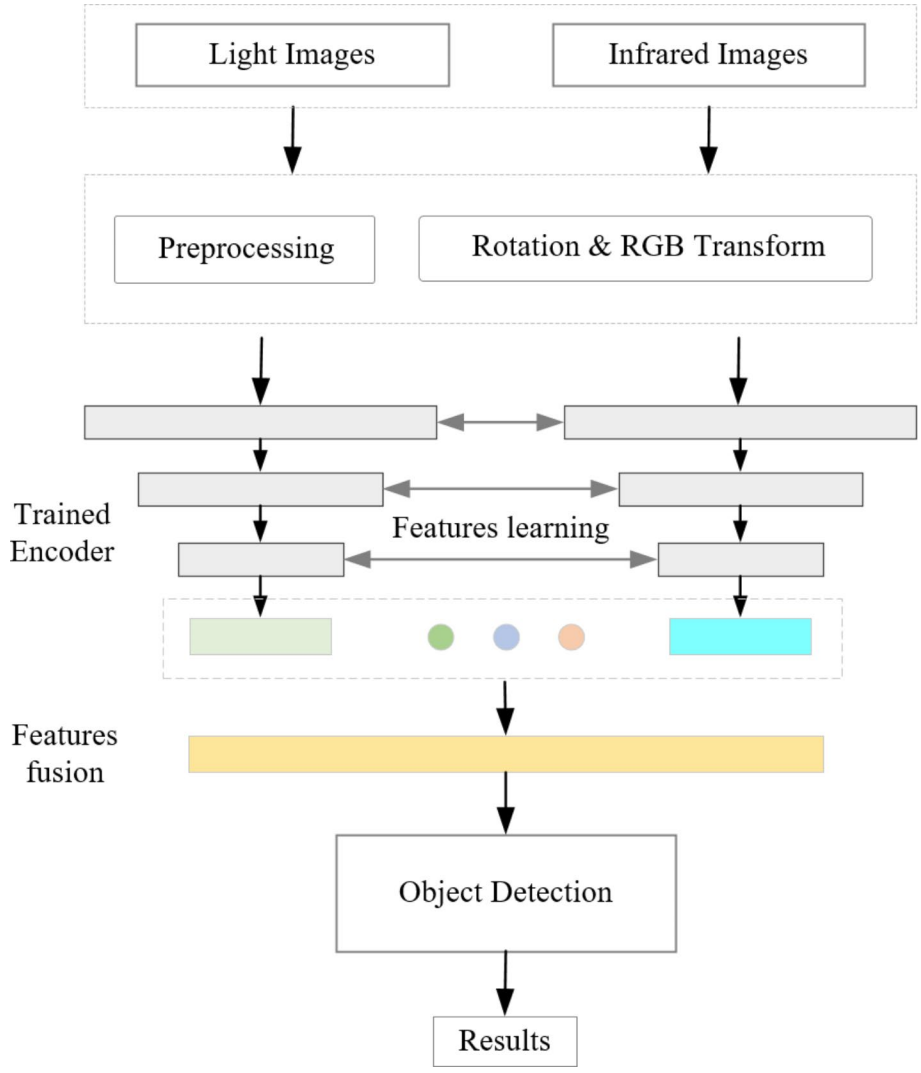

Fig. 2 The overall structure of infrared and visible light fusion object detection network

detection module uses multiple deep features of different scales to construct feature pyramid prediction structure by up-sampling and fusion operation, so that the network has strong semantic information at different scales, and ensures the network to accurately detect targets at different scales.

As the primary task of object detection, feature extraction directly determines the quality of object detection model. For traditional object detection, features are mainly designed manually [42]. The object detection is realized by capturing the features in the sliding window and using machine learning for classification. The deep learning-based object detection method broadens the scope of feature extraction and employs end-to-end training to autonomously learn object features, circumventing the constraints associated with manually crafted features. Therefore, detection algorithms based on deep learning can usually achieve better detection results than traditional methods. Based on this, this paper designs a parallel dual-branch feature extraction network suitable for infrared and visible images by referring to the current classical deep learning network. In order to effectively extract the shallow and deep features of each object in the image,

the feature extraction structure constructed in this paper is composed of multiple submodules with different feature scales in series and stacked.

The feature extraction structure consists of multiple CAT modules and LK modules, as shown in Fig. 3. The CAT module is shown in Fig. 3(a). This structure mainly preprocesses features of the original image and uses two branches of parallel convolution and pooling with step size of 2 to extract salient features of the target, which reduces the image dimension and filters part of the noise to ensure the in-depth feature extraction of the subsequent structure. The LK module is shown in Fig. 3(b). It is mainly constructed by residual structure using convolution layer and activation layer, and dimension reduction is carried out by 2×2 pooling operation with step size of 2 between different stages. Since features need to be extracted from infrared and visible images respectively, in order to avoid excessive network computation, the LK module adopts depth-separable convolution instead of traditional convolution to extract features, which effectively reduces network parameters and reduces the computation. Although the feature information extracted by autoencoder is lower than that of traditional convolution, the information fusion of double branches can better compensate for the lack of features. At the same time, the LK module incorporates a residual structure to mitigate the issues of vanishing and exploding gradients that can arise from excessively deep network layers during training. And LeakyReLU is taken as the activation function to make the network converge faster, which is defined as follows.

$$f(x) = \begin{cases} \alpha x, & x < 0 \\ x, & x \geq 0 \end{cases} \tag{7}$$

where α is the offset, which is a small value of the hyperparameter.

Visible images are rich in color, texture, and other details, offering more comprehensive information. However, their effectiveness is significantly influenced by variations in

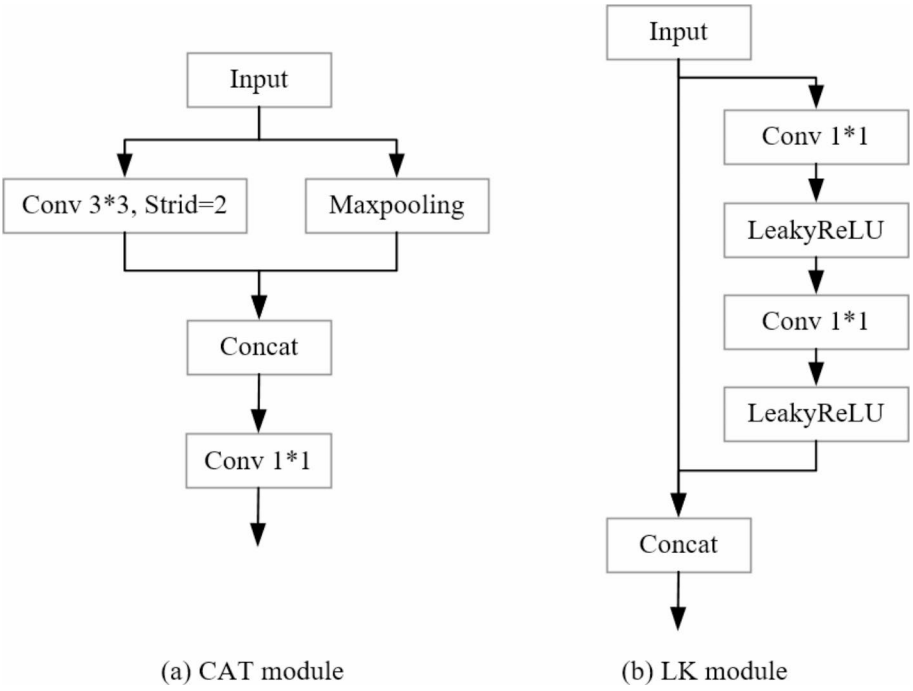

Fig. 3 The CAT module and LK module

light intensity and weather conditions. In contrast, infrared images rely on the thermal radiation energy emitted by targets to create imagery, which is not affected by illumination, but the image contrast is low, which will lose some of the texture, structure, and other appearance features of the target. Therefore, by fusing infrared and visible image information, targets can be better enhanced and discovered [43]. Building on this, we have developed a feature fusion module that works in tandem with the feature extraction architecture. This module is designed to ensure that the information extracted from infrared and visible images is complementary, enhancing the overall detection capability. Considering the network operation efficiency, the fusion structure mainly integrates the last layer of each scale in the feature extraction process.

The detection module uses the output of feature fusion structure as input. Since the output of feature fusion is two channels, i.e., the infrared and visible light, concatenation operation is used to concatenate the two channels of features as the detection input. At the same time, considering the obvious difference in the size of each target in the real scene, we adopt multiple fused features of different dimensions to construct a feature pyramid detection structure in a top-down manner. In this structure, firstly, the fused deep features are adjusted to be consistent with the shallow features through point convolution. Then, we up-sample it to the shallow feature scale size and concatenate it with the shallow feature scale. Next, we perform convolution operation on the spliced feature information to fully integrate the deep feature information. We splice and fuse features of different dimensions successively so that the detection module can fully obtain global and local feature information. Finally, the fusion features are used to predict the target category and location.

Model optimization based on evolutionary algorithms

The accuracy of neural network models is influenced by various factors, including the number of neurons, layers, weights, and the learning rate, each presenting distinct challenges during adjustment. Manually tweaking these parameters, such as the neuron count, network depth, and learning rate, demands extensive expertise, even from professionals. Weight optimization in neural networks predominantly uses gradient-based methods, which carry the risk of converging to local optima [44]. With the surge in artificial intelligence advancements and the increasing complexity of engineering tasks in recent years, researchers have introduced a plethora of neural network architectures and neuron models to address specific machine learning challenges. For instance, convolutional neural networks (CNNs) have demonstrated remarkable success in image processing, while recurrent neural networks (RNNs) have become a staple in natural language processing applications. As a key model of "brain-like" research, pulsed neural networks have attracted extensive attention. Some graph model-based neural network structures have also become research hotspots, and have achieved good performance in many industrial application scenarios. In the face of different learning tasks, how to optimize the hyperparameters of neural networks is the first task of applying neural networks to deal with complex problems.

Evolutionary computing is inspired by biological evolution in nature. By imitating the iterative process of biological evolution, such as environmental selection, gene crossover and mutation, genetic algorithms that can be used to solve optimization problems have been proposed [45]. The genetic algorithm employs population-based search techniques,

substituting a problem's individual solution with a population of solutions. Through the application of genetic operations like selection, crossover, and mutation on the current population, it generates a new generation. This process progressively drives the population towards a state that approximates the optimal solution. In the process of evolution, individuals with greater fitness have a greater probability of survival and obtain gene sequences that are more adapted to the environment. Therefore, it has the characteristics of strong robustness, self-organization, self-adaptation, and self-learning [46]. We use particle swarm optimization (PSO) algorithm to optimize the proposed network hyperparameters, including learning rate, epoch, and batch size. We encode the parameters in a form similar to network addresses and transform a fixed-length structure into a variable-length structure. PSO has few parameters during evolution, which can accelerate the process of finding the optimal structure.

Experiments and results

To validate the method proposed in this paper effectively, we employ the Pytorch deep learning framework for constructing the model in our experiments. The validation and comparison of the network are carried out using public datasets. This section will detail the datasets and evaluation metrics utilized in the experiments, alongside a comparison of the experimental outcomes with other object detection methods.

Data description and metrics

In the experiment, RGBT210 public dataset proposed in literature [47] was used for testing. The RGBT210 dataset consists of images captured by infrared and visible cameras with the same imaging parameters in 210 scenarios. The dataset contains about 210,000 images, covering infrared and visible image pairs of about 20 targets at different time periods and light intensities. Since the dataset is large and most of the images are similar, in order to quickly verify the proposed network, 10,000 images with low similarity are screened out for testing. The selected images include 10 categories such as people, animals, cars, and bicycles. In order to facilitate calculation and save computing time, the image size is uniformly processed as 448×448, and the training, verification and test sets are constructed at a ratio of 7:1:2.

For the accuracy and efficiency evaluation of the proposed network, we employ the Recall, Precision, the mean average precision (mAP), and the number of images per second (FPS) the network processes as performance metrics, respectively. Their definitions are:

$$\text{Recall} = \frac{TP}{TP + FN}$$

$$\text{Precision} = \frac{TP}{TP + FP}$$

$$\text{AP}_\text{i} = \frac{\sum \text{Precision}_\text{i}}{(\text{TP} + \text{TN} + \text{FP} + \text{FN})_\text{i}}$$

$$\text{mAP} = \frac{\sum \text{AP}_\text{i}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}$$

$$\mathrm{FPS} = N / \sum_{k}^{N} T_k$$

where TP denotes a correct identification of a positive instance, TN represents a correct identification of a negative instance, FP signifies an incorrect identification of a positive instance, and FN refers to an incorrect identification of a negative instance, i indicates the category type, N is the total number of the images, T_k is the time taken by the model to process the k-th image.

Model performance

We compare our method with other object detection models to evaluate the performance, including the classic CNN model [48], the AutoEncoder [49], Faster RCNN [50], YOLO [51], YOLO-based [52], MobileNet-based [53], and our proposed Evolutionary Computation-based Object Detection (ECOD). Table 1 shows the comparison results with other methods. The CNN model and AutoEncoder model in the table are used as reference baselines for comparison results, while Faster RCNN and YOLO are the models with outstanding performance in the field of target detection. As can be seen from the table, our proposed model can achieve comparable performance compared with the current mainstream high-precision and high-efficiency object detection networks. At the same time, we use autoencoders to replace traditional convolutions, and reference network construction strategies such as residuals and LeakyReLU activation functions. Compared with the YOLO network, the proposed network trades a small loss of accuracy in exchange for a large increase in network efficiency. However, compared with Faster RCNN, because the proposed model is a single-step detection, and the autoencoder loses some feature information compared with the traditional convolution, the accuracy is reduced. We also compared new algorithms in recent years, such as YOLO-based and MobileNet-based. From the results of these advanced algorithms, it can be seen that the results of our proposed ECOD model are similar to theirs. Although our model is not the optimal result, compared to the optimal model, our model is able to achieve the best result on some metrics. For example, our proposed model achieves a recall of 74.9%.

In order to further verify the feature complementarity of the infrared and visible light dual-branch structure and the effectiveness of the proposed feature fusion structure, we tested the visible light, infrared, and fused network performance in experiments. Table 2 showcases the outcomes of the ablation studies conducted. The comparison results in Table 2 clearly depict the comparison results in the three cases. The experimental results using only infrared image features performed the worst, with all results below 60%. The results of using visible light images all exceeded the results of infrared images. This phenomenon verifies from a certain angle that visible light images contain more features

Table 1 The comparison results with other methods

Method	Recall	Precision	mAP	FPS
CNN	72.1	71.6	63.2	31
AutoEncoder	70.3	69.5	61.4	20
Faster RCNN	74.6	76.6	68.7	27
YOLO	76.4	72.7	64.8	68
YOLO-based	76.8	73.1	65.7	63
MobileNet-based	75.9	75.6	66.9	42
ECOD (Ours)	74.9	73.4	66.3	55

Table 2 The results of the ablation experiments for each branch

Structure	Recall	Precision	mAP
Infrared only	58.1	56.3	53.7
Visible light only	63.7	63.4	56.8
Fusion	74.9	73.4	66.3

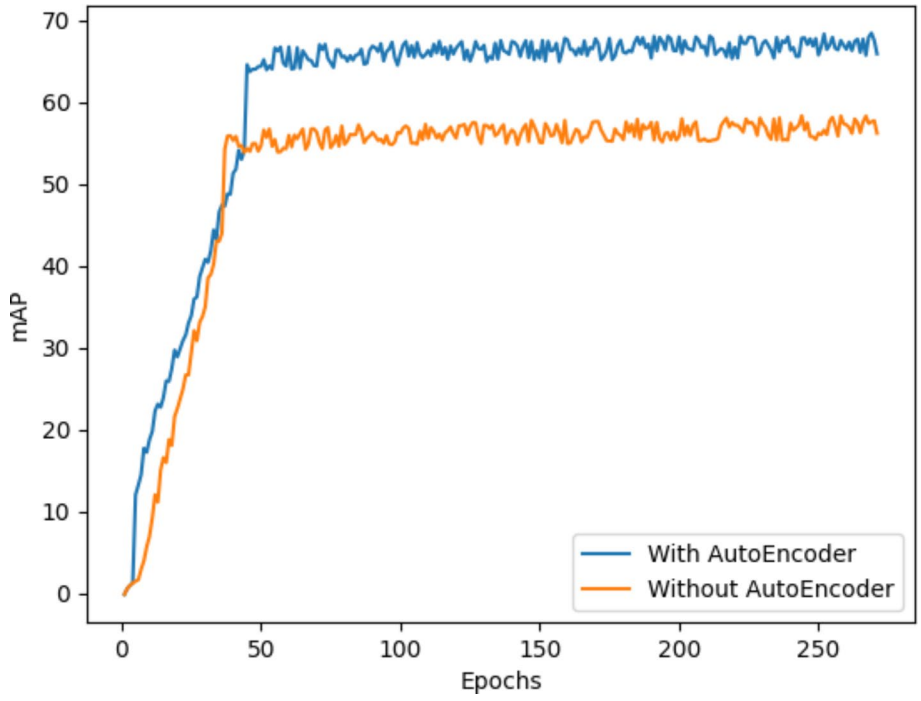

Fig. 4 The experimental results with or without AutoEncoder model

and are more in line with the visual features perceived by the human eye. Significantly, the model that integrates features from both types of images yields the most superior performance. This not only underscores the efficacy of the method we proposed but also demonstrates that infrared imagery indeed plays a supportive role in enhancing object detection tasks. Our findings demonstrate the effectiveness of combining visible light and infrared images for object detection. This approach can be applied to various fields such as surveillance, medical imaging, and autonomous driving, where different spectral bands provide complementary information. The success of multispectral data fusion in our study suggests that further exploration of other spectral combinations could yield additional benefits in different contexts.

Ablation study

In this paper, we design a strategy based on autoencoder feature extraction and feature fusion. To verify the role of these modules, we performed ablation experiments. First, we tested the effect of the autoencoder, that is, the results produced by the model in the case of using the autoencoder model are compared with the results produced by the ordinary convolutional network model. Figure 4 depicts the experimental results for both cases. In the figure, the green line represents the results obtained by the proposed method on the mAP metric when using the autoencoder model. The orange line is the result obtained without using the autoencoder. Looking at the model training process, the results with

the autoencoder are significantly better than those without. The mAP results with the autoencoder all exceed 60% in the later training, while the results without the autoencoder can only reach around 55%. These phenomena show that the autoencoder model can play a good role in feature extraction. The application of self-supervised learning for feature extraction shows that models can achieve high performance without relying heavily on large, labeled datasets. This has significant implications for areas with limited labeled data, promoting the use of self-supervised techniques in broader applications.

At the same time, we verified the evolutionary optimization algorithm to determine whether the optimization and performance of the model improved. We compare the experimental results with and without the optimization algorithm. Figure 5 shows a radar plot comparing the results in the two cases. Five metrics are compared in Fig. 5, including AP, mAP, Recall, Precision, and FPS. We can see that: (1) In terms of the FPS index, the results in the two cases are consistent; (2) In other indexes, such as mAP, Recall, AP, and Precision, the experimental results optimized by PSO algorithm are better than those without PSO algorithm. Furthermore, these outcomes solidly affirm that the Particle Swarm Optimization (PSO) algorithm introduced in this paper significantly contributes to enhancing the model's performance. The use of evolutionary computation for model optimization highlights its potential in fine-tuning complex models. This approach can be extended to optimize other machine learning models, particularly those with numerous hyperparameters. Evolutionary algorithms could be combined with other optimization techniques to develop hybrid methods that leverage the strengths of multiple approaches, potentially leading to more efficient and effective optimization strategies.

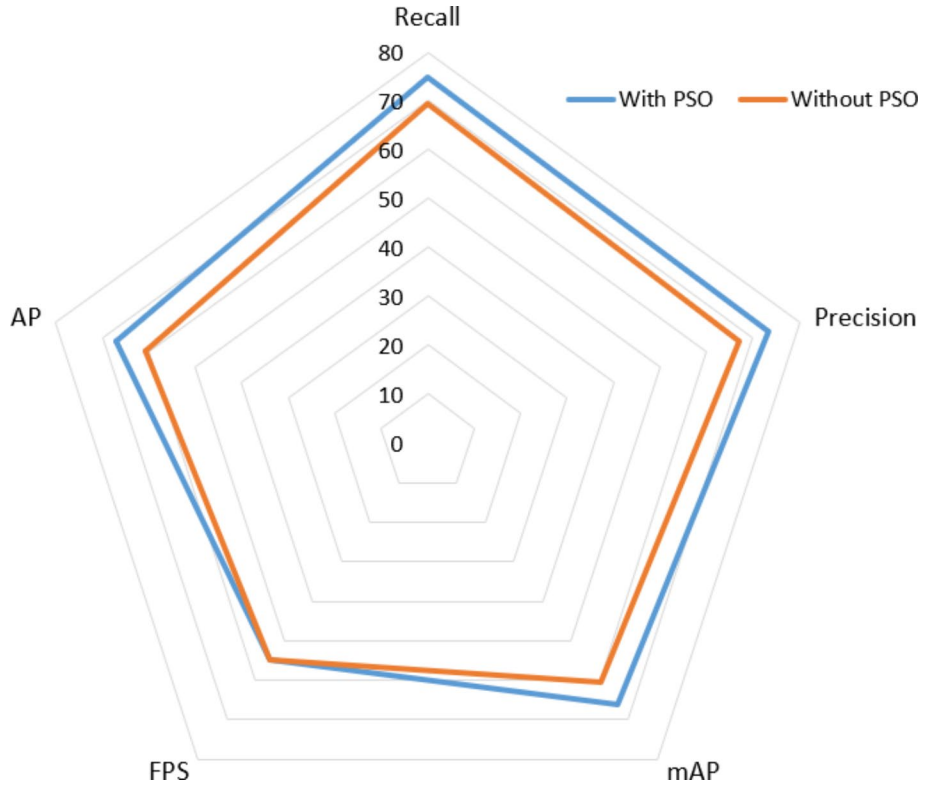

Fig. 5 The experimental results with or without PSO algorithm

Discussion

There are several internal and external threats to the validity of our results that we have identified and addressed to enhance the robustness of our findings. Internally, the quality of the data and preprocessing steps might introduce biases. We mitigated this threat by ensuring consistent preprocessing techniques such as normalization and resizing across all datasets. Data augmentation was also performed to improve model robustness. Another internal threat is related to model parameter tuning, where the choice of hyperparameters and model architectures can significantly influence performance. To address this, we employed evolutionary computation for systematic and unbiased parameter optimization, reducing the likelihood of overfitting to specific hyperparameters. Additionally, the way data is split into training and validation sets can impact evaluation. We used cross-validation techniques to ensure the model's performance is evaluated across multiple splits, providing a more reliable assessment.

Externally, the generalizability of our results to other datasets is a potential threat. The performance observed on the selected public datasets may not generalize to other datasets. To mitigate this, we validated our model on multiple publicly available datasets chosen for their diversity in content and imaging conditions. Future work includes testing on additional datasets to further assess generalizability. Another external threat is the application of our model in real-world scenarios. The controlled conditions of public datasets may not fully represent real-world scenarios. To address this, we included diverse and challenging scenarios within our selected datasets to mimic real-world conditions. Future research will involve testing the model in real-world applications to further validate its robustness. Additionally, variability in the quality and calibration of multispectral imaging devices may affect performance. We used standard calibration techniques for the infrared and visible light images in our datasets. Incorporating data from multiple devices in future studies will help address this threat.

Conclusion

This paper introduces an image object detection approach rooted in self-supervised and data-driven learning principles. We leverage a combination of visible light and infrared images to detect and pinpoint targets within images. First, we utilize a self-supervised learning method and the AutoEncoder to perform high-dimensional feature extraction on the two types of images. Second, we fuse the extracted features from two kinds of images, namely visible light image and infrared image, to detect and identify objects. Third, we investigate a model parameter optimization method to optimize the model performance based on evolutionary learning algorithms to improve the training performance of the model. However, all models have their limitations, and our model is no exception. The limitation of our proposed model is that its learning process is multi-faceted, as it requires feature learning and fusion. In future work, we will focus on improving and increasing the training and inference efficiency of the model.

Author contributions

X.S. contributed to the writing and conception; H.L. contributed to the method; A.S. contributed to the data analysis; W.V. contributed to the software; V.C. contributed to the writing polishing.

Funding

This study was supported by grants from the National Natural Science Foundation of China (No. 62106214), the National Natural Science Foundation of China (No. U23A2033), and the National Research Council of Thailand (NRCT) (No. N42A660902).

Data availability
The datasets generated and/or analyzed during the current study are not publicly available due to data privacy but are available from the corresponding author on reasonable request.

Declarations

Ethics approval and consent to participate
Not applicable.

Consent for publication
Not applicable.

Competing interests
The authors declare no competing interests.

Author details
[1]College of Electrical Engineering, Yanshan University, Qinhuangdao, Hebei 066000, China
[2]Key Laboratory of Industrial Computer Control Engineering of Hebei Province, Qinhuangdao, Hebei 066000, China
[3]WMG, University of Warwick, Coventry, UK
[4]Chulalongkorn Business School, Faculty of Commerce and Accountancy, Chulalongkorn University, Bangkok, Thailand
[5]BITS-Pilani, Pilani, India
[6]Department of Cyber Systems Engineering, WMG, Coventry CV74AL, United Kingdom
[7]University Centre for Research & Development, Chandigarh University, Mohali, Punjab 140413, India
[8]School of Computer Science Engineering, Lovely Professional University, Phagwara, Punjab 144411, India
[9]Department of Computer Science and Engineering, Graphic Era Deemed to be University, Dehradun 248002, India
[10]Center of Research Impact and Outcome, Chitkara University, Punjab, India

Received: 20 September 2023 / Accepted: 26 August 2024
Published online: 12 September 2024

References

1. Bekkerman I, Tabrikian J. Target detection and localization using MIMO radars and sonars[J]. IEEE Trans Signal Process. 2006;54(10):3873–83.
2. Lin C, Lu J, Wang G et al. Graininess-aware deep feature learning for pedestrian detection[C]//Proceedings of the European conference on computer vision (ECCV). 2018: 732–747.
3. Wu R, Feng M, Guan W et al. A mutual learning method for salient object detection with intertwined multi-supervision[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019: 8150–8159.
4. Deng J, Dong W, Socher R et al. Imagenet: A large-scale hierarchical image database[C]//2009 IEEE conference on computer vision and pattern recognition. Ieee, 2009: 248–255.
5. Kay W, Carreira J, Simonyan K et al. The kinetics human action video dataset[J]. arXiv preprint arXiv:1705.06950, 2017.
6. Nasrabadi NM. Hyperspectral target detection: an overview of current and future challenges[J]. IEEE Signal Process Mag. 2013;31(1):34–44.
7. Misra I, Maaten L. Self-supervised learning of pretext-invariant representations[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020: 6707–6717.
8. Rawat W, Wang Z. Deep convolutional neural networks for image classification: a comprehensive review[J]. Neural Comput. 2017;29(9):2352–449.
9. Zhao W, Xie S, Zhao F et al. Metafusion: Infrared and visible image fusion via meta-feature embedding from object detection[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2023: 13955–13965.
10. Zhang X, Demiris Y. Visible and infrared image fusion using deep learning[J]. IEEE Trans Pattern Anal Mach Intell. 2023;45(8):10535–54.
11. Pathak D, Krahenbuhl P, Donahue J et al. Context encoders: Feature learning by inpainting[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 2536–2544.
12. Iizuka S, Simo-Serra E, Ishikawa H. Globally and locally consistent image completion[J]. ACM Trans Graphics (ToG). 2017;36(4):1–14.
13. Ledig C, Theis L, Huszár F et al. Photo-realistic single image super-resolution using a generative adversarial network[C]// Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 4681–4690.
14. Dalal N, Triggs B. Histograms of oriented gradients for human detection[C]//2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05). Ieee, 2005, 1: 886–893.
15. Sánchez J, Perronnin F, Mensink T, et al. Image classification with the fisher vector: theory and practice[J]. Int J Comput Vision. 2013;105(3):222–45.
16. Coates A, Ng AY. Learning feature representations with k-means[M]//Neural networks: tricks of the trade. Berlin, Heidelberg: Springer; 2012. pp. 561–80.
17. Caron M, Bojanowski P, Joulin A et al. Deep clustering for unsupervised learning of visual features[C]//Proceedings of the European conference on computer vision (ECCV). 2018: 132–149.
18. Van Der Maaten L. Learning a parametric embedding by preserving local structure[C]//Artificial intelligence and statistics. PMLR, 2009: 384–91.
19. Xie J, Girshick R, Farhadi A. Unsupervised deep embedding for clustering analysis[C]//International conference on machine learning. PMLR, 2016: 478–487.

20. Noroozi M, Pirsiavash H, Favaro P. Representation learning by learning to count[C]//Proceedings of the IEEE international conference on computer vision. 2017: 5898–5906.
21. Kim D, Cho D, Yoo D et al. Learning image representations by completing damaged jigsaw puzzles[C]//2018 IEEE Winter Conference on Applications of Computer Vision (WACV). IEEE, 2018: 793–802.
22. Mundhenk TN, Ho D, Chen BY. Improvements to context based self-supervised learning[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 9339–9348.
23. Feng Z, Xu C, Tao D. Self-supervised representation learning by rotation feature decoupling[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019: 10364–10374.
24. Zhang L, Qi GJ, Wang L et al. Aet vs. aed: Unsupervised representation learning by auto-encoding transformations rather than data[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019: 2547–2555.
25. Putra F, A I A, Utaminingrum F, Mahmudy WF. HOG feature extraction and KNN classification for detecting vehicle in the highway[J]. IJCCS (Indonesian J Comput Cybernetics Systems). 2020;14(3):231–42.
26. Hassin A, Abbood D. Machine Learning System for human–ear Recognition using scale invariant feature Transform[J]. Artif Intell Rob Dev J, 2021: 1–12.
27. Li J, Wong HC, Lo SL, et al. Multiple object detection by a deformable part-based model and an R-CNN[J]. IEEE Signal Process Lett. 2018;25(2):288–92.
28. Li J, Liang X, Shen SM, et al. Scale-aware fast R-CNN for pedestrian detection[J]. IEEE Trans Multimedia. 2017;20(4):985–96.
29. Yang GC, Yang J, Su ZD, et al. Improved YOLO feature extraction algorithm and its application to privacy situation detection of social robots[J]. Acta Automatica Sinica. 2018;44(12):2238–49.
30. Felzenszwalb PF, Huttenlocher DP. Efficient graph-based image segmentation[J]. Int J Comput Vision. 2004;59:167–81.
31. Boykov Y, Funka-Lea G. Graph cuts and efficient ND image segmentation[J]. Int J Comput Vision. 2006;70(2):109–31.
32. Moore AP, Prince SJD, Warrell J et al. Superpixel lattices[C]//2008 IEEE conference on computer vision and pattern recognition. IEEE, 2008: 1–8.
33. Levinshtein A, Stere A, Kutulakos KN, et al. Turbopixels: fast superpixels using geometric flows[J]. IEEE Trans Pattern Anal Mach Intell. 2009;31(12):2290–7.
34. Achanta R, Shaji A, Smith K, et al. SLIC superpixels compared to state-of-the-art superpixel methods[J]. IEEE Trans Pattern Anal Mach Intell. 2012;34(11):2274–82.
35. Long J, Shelhamer E, Darrell T. Fully convolutional networks for semantic segmentation[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2015: 3431–3440.
36. Ronneberger O, Fischer P, Brox T. U-net: Convolutional networks for biomedical image segmentation[C]//Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5–9, 2015, proceedings, part III 18. Springer International Publishing, 2015: 234–241.
37. Qi GJ, Zhang L, Chen CW et al. Avt: Unsupervised learning of transformation equivariant representations by autoencoding variational transformations[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2019: 8130–8139.
38. Zhai X, Oliver A, Kolesnikov A et al. S4l: Self-supervised semi-supervised learning[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2019: 1476–1485.
39. Chen T, Zhai X, Ritter M et al. Self-supervised gans via auxiliary rotation loss[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019: 12154–12163.
40. Xiao X, Wang B, Miao L, et al. Infrared and visible image object detection via focused feature enhancement and cascaded semantic extension[J]. Remote Sens. 2021;13(13):2538.
41. Gidaris S, Singh P, Komodakis N. Unsupervised representation learning by predicting image rotations[J]. arXiv preprint arXiv:1803.07728, 2018.
42. Felzenszwalb P, McAllester D, Ramanan D. A discriminatively trained, multiscale, deformable part model[C]//2008 IEEE conference on computer vision and pattern recognition. Ieee, 2008: 1–8.
43. Xiang X, Lv N, Yu Z, et al. Cross-modality person re-identification based on dual-path multi-branch network[J]. IEEE Sens J. 2019;19(23):11706–13.
44. Sinha T, Verma B, Haidar A. Optimization of convolutional neural network parameters for image classification[C]//2017 IEEE Symposium Series on Computational Intelligence (SSCI). IEEE, 2017: 1–7.
45. Volkanovski A, Mavko B, Boševski T, et al. Genetic algorithm optimisation of the maintenance scheduling of generating units in a power system[J]. Volume 93. Reliability Engineering & System Safety; 2008. pp. 779–89. 6.
46. Tavakkoli-Moghaddam R, Safari J, Sassani F. Reliability optimization of series-parallel systems with a choice of redundancy strategies using a genetic algorithm[J]. Reliab Eng Syst Saf. 2008;93(4):550–6.
47. Li C, Zhao N, Lu Y et al. Weighted sparse representation regularized graph learning for RGB-T object tracking[C]//Proceedings of the 25th ACM international conference on Multimedia. 2017: 1856–1864.
48. Du J. Understanding of object detection based on CNN family and YOLO[C]//Journal of Physics: Conference Series. IOP Publishing, 2018, 1004(1): 012029.
49. Li B, Sun Z, Guo Y, Supervae. Superpixelwise variational autoencoder for salient object detection[C]//Proceedings of the AAAI Conference on Artificial Intelligence. 2019, 33(01): 8569–8576.
50. Ren S, He K, Girshick R et al. Faster r-cnn: towards real-time object detection with region proposal networks[J]. Adv Neural Inf Process Syst, 2015, 28.
51. Bochkovskiy A, Wang CY, Liao HYM. Yolov4: optimal speed and accuracy of object detection[J]. arXiv preprint arXiv:2004.10934, 2020.
52. Sirisha U, Praveen SP, Srinivasu PN, et al. Statistical analysis of design aspects of various YOLO-based deep learning models for object detection[J]. Int J Comput Intell Syst. 2023;16(1):126.
53. Krishnachaithanya N, Singh G, Sharma S et al. People Counting in Public Spaces using Deep Learning-based Object Detection and Tracking Techniques[C]//2023 International Conference on Computational Intelligence and Sustainable Engineering Solutions (CISES). IEEE, 2023: 784–788.

Publisher's note