Computer Modeling in Engineering & Sciences

Tech Science Press

DOI: 10.32604/cmes.2023.028018

REVIEW

Exploring Deep Learning Methods for Computer Vision Applications across Multiple Sectors: Challenges and Future Trends

Narayanan Ganesh[1], Rajendran Shankar[2], Miroslav Mahdal[3], Janakiraman Senthil Murugan[4], Jasgurpreet Singh Chohan[5] and Kanak Kalita[6,*]

[1]School of Computer Science and Engineering, Vellore Institute of Technology, Chennai, 600 127, India

[2]Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Vaddeswaram, 522502, India

[3]Department of Control Systems and Instrumentation, Faculty of Mechanical Engineering, VSB-Technical University of Ostrava, Ostrava, 708 00, Czech Republic

[4]Department of Computer Science and Engineering, Vel Tech High Tech Dr. Rangarajan Dr. Sakunthala Engineering College, Chennai, 600 062, India

[5]Department of Mechanical Engineering and University Centre for Research & Development, Chandigarh University, Mohali, 140413, India

[6]Department of Mechanical Engineering, Vel Tech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology, Avadi, 600 062, India

*Corresponding Author: Kanak Kalita. Email: drkanakkalita@veltech.edu.in

Received: 26 November 2022 Accepted: 05 September 2023 Published: 30 December 2023

ABSTRACT

Computer vision (CV) was developed for computers and other systems to act or make recommendations based on visual inputs, such as digital photos, movies, and other media. Deep learning (DL) methods are more successful than other traditional machine learning (ML) methods in CV. DL techniques can produce state-of-the-art results for difficult CV problems like picture categorization, object detection, and face recognition. In this review, a structured discussion on the history, methods, and applications of DL methods to CV problems is presented. The sector-wise presentation of applications in this paper may be particularly useful for researchers in niche fields who have limited or introductory knowledge of DL methods and CV. This review will provide readers with context and examples of how these techniques can be applied to specific areas. A curated list of popular datasets and a brief description of them are also included for the benefit of readers.

KEYWORDS

Neural network; machine vision; classification; object detection; deep learning

1 Introduction

Deep learning (DL) refers to a group of techniques that make use of deep architectures to acquire expert-level feature representation learning abilities. In simple terms, DL can be defined as a subset of machine learning (ML) [1] that utilizes neural networks (NN) having three or more layers. These NN "learn" using extensive datasets to mimic human brain activity. DL is a relatively new method that has

already found widespread use in numerous AI-related fields like natural language processing (NLP) [2], semantic parsing [3], computer vision (CV) [4], transfer learning [5], pixel restoration [6], etc. The substantial growth in chip processing powers (such as GPU units), the significantly reduced cost of computing gear, and the considerable breakthroughs in ML techniques are the primary reasons for the current blossoming of DL [7].

The field of CV has made great strides in recent years thanks to the application of DL methods. Due to memory, CPU, and GPU constraints, DL methods had a tough time in the beginning stages of CV development. Approaches like Bayes classifier, AdaBoost, Decision Tree, Expectation-Maximization (EM), Haar Classifier, K-means, K-Nearest Neighbour (KNN), Naive Random Forest, Support Vector Machine (SVM), etc. Viola and Jones [8] built a face recognition system using the Adaboost algorithm. OpenCV is based on the Haar classifier, which was developed by Lienhart and Maydt [9].

In Chai et al.'s [10] review, they classified the DL methods into ten different categories: Convolutional Neural Networks (CNNs), Long Short-Term Memory Networks (LSTMs), Recurrent Neural Networks (RNNs), Generative Adversarial Networks (GANs), Radial Basis Function Networks (RBFNs), Multilayer Perceptrons (MLPs), Self-Organizing Maps (SOMs), Deep Belief Networks (DBNs), Restricted Boltzmann Machines (RBMs), and Autoencoders. Based on a comparison of CNN, RBM, Autoencoder and Sparse Coding, Guo et al. [7] commented that CNN is most desirable for CV applications.

In this review article, the application of DL frameworks to CV applications is focused on. Though a few well-written reviews on this area are available, this review aims to further augment them by exploring the various applications of DL-based CV. This paper provides a comprehensive review of the various applications of DL-based CV across multiple sectors, including Agriculture, Healthcare, Manufacturing, Sports, and Transportation. Additionally, the current challenges and future trends in this field are discussed which would provide insights for researchers and practitioners in this field. After the brief introduction section, Section 2 delves into a summarized history of NNs. Section 3 presents the DL methods in a structured way. CNNs, RNNs, DBNs, DBMs, DEMs and Autoencoders are covered in this section. Section 4 presents the applications of the DL-based CV in a sector-wise manner. Agriculture, healthcare, manufacturing, sports and transportation sector applications are covered in good detail in this section. This will help researchers from these niche fields have a good picture of the capabilities of DL-based CV in their domain. Datasets are an important component of any methodology development framework and thus various popular datasets are included in Section 5. Conclusions based on the literature search are presented in concise form in Section 6.

2 History of Neural Networks

The idea of deep learning (DL) originated from research on artificial neural networks (ANNs). In 1943, McCulloch and Pitts presented the first model of an artificial neuron, which laid the foundation for the study of ANNs [11]. The Hebb learning rule, introduced by Hebb in 1949 [12], is considered a technique for learning in biological neurons and is often summarized as "Neurons that fire together wire together" [13]. In 1958, Rosenblatt developed the Mark I Perceptron, a model of a McCulloch-Pitts neuron that could learn weights from inputs passed in sequence. In the late 1950s, Rosenblatt coined the term ANN when he developed the perceptron network and its associated learning rule [14]. Widrow and Hoff also developed adaptable linear ANNs, taught using a new learning approach called the Widrow-Hoff learning rule [15].

However, in the book by Minsky and Papert [16], they pointed out the deficiencies of both the Rosenblatt perceptron and the Widrow-Hoff learning rule, leading many to believe that research on ANNs was pointless. Despite this, research on ANNs continued, with notable contributions from Kohonen [17] and Anderson [18]. During the same time, Grossberg's [19] study on self-organizing networks was also gaining impetus. The decade of 1980s saw a renewed interest in research on ANNs due to the elimination of obstacles that existed in the 1960s and the development of essential new ideas. In 1982, physicist Hopfield utilized statistical mechanics to describe the operation of a certain sort of recurrent network that may act as an associative memory [20]. The backpropagation algorithm, first discovered in the 1960s by Seppo Linnainmaa, gained acceptance in the 1970s [21] and was popularized by Rumelhart et al. [22] and LeCun et al. [23].

The usage of CNNs in commercial settings became less common as more straightforward approaches such as support vector machines and linear classifiers gained traction. A considerable increase in picture classification accuracy was demonstrated by Krizhevsky et al. [24] in 2012, which sparked interest in CNNs. These findings illustrate the potential of recent developments in the field of ML, such as the approach developed by Hinton et al. [25] for facilitating learning in deep NNs. It is currently normal practice, thanks to these methodologies, to train networks with a large number of hidden layers, which indicates that the networks themselves can be a great deal more sophisticated. In addition to this, it has been discovered that these perform far better than shallow NNs do on several different challenges [26].

3 Deep Learning Methods

In this section, the various types of DL methods are discussed in brief. Fig. 1 shows a flowchart depicting the classification of DL methods.

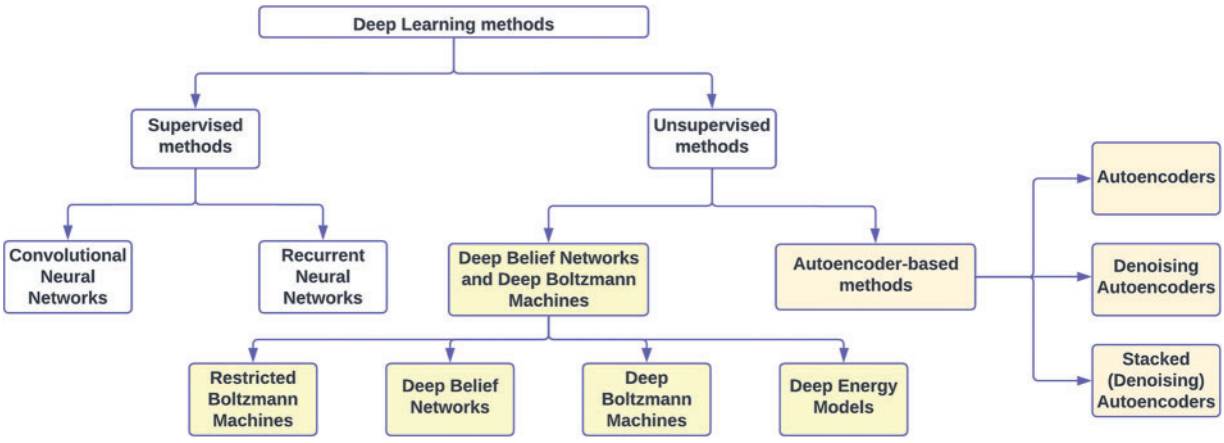

Figure 1: Flowchart showing the categorization of different deep learning methods

3.1 Supervised Methods

3.1.1 Convolutional Neural Networks

In the field of DL, CNNs are among the most well-known methods. In CNN, robust training of several layers is possible [27]. It is the most popular choice for many CV tasks and has proven to be quite effective. Following the work of Rumelhart et al. [22], LeCun et al. [23] demonstrated the efficacy of stochastic gradient descent via backpropagation for training CNNs, a class of models that

augment the noncognition. With 64,660 connections across three hidden layers—two of which were convolutional—LeCun et al. [23] used it to decipher handwritten ZIP codes.

A typical CNN (Fig. 2) has three distinct neural layers: a convolutional layer, a pooling layer, and a fully connected layer [28]. Each type of layer provides distinct functions. NN training consists of two phases: forward and backward. The forward stage aims to accurately depict the input image using the current layer settings (weights and bias). The loss cost is then calculated using the ground-truth labels and the prediction output. In the reverse stage, the gradients of each parameter are calculated using chain rules, all following the loss cost [7]. After incorporating the gradients into the parameters, a new forward calculation can begin. The network learning process can be terminated after a sufficient number of forward and backward rounds.

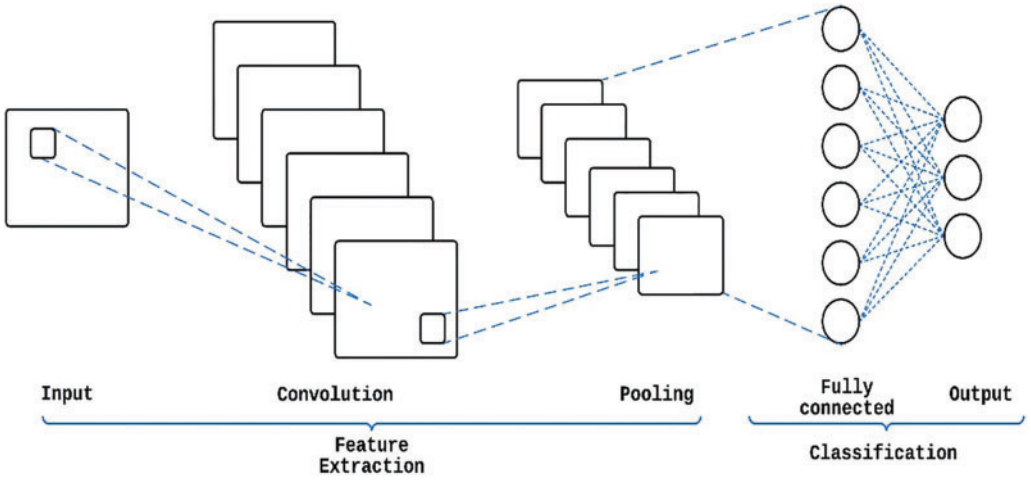

Figure 2: CNN architecture

In the convolutional layer, the input image is convolved with a set of learnable filters or kernels, which are used to extract local features from the input image. Mathematically, the convolution operation can be represented as follows:

$$S(i,j) = (I * K)(i,j) = \sum_{m} \sum_{n} I(m,n) K(i-m, i-n) \tag{1}$$

where $S(i,j)$ is the value of the output feature map at position (i,j), $I(i,j)$ is the input image, and $K(m,n)$ is the filter or kernel.

3.1.2 Recurrent Neural Networks

In many applications like the translation of natural languages, music, time-series data, handwriting recognition, video processing, etc., RNNs are used to handle sequential data to reflect time dependencies. RNNs have a higher number of sophisticated training parameters than other types of NNs. To train the sequential phases, the RNN model requires complicated architecture, optimization, and training. Since the Hidden Markov Model requires previous data to function, it seems infeasible to model data with a large temporal dependence. RNN eliminates this issue because it focuses solely on the currently available data and processes it very precisely. To date, in general, RNN has proven to be superior to other NNs in terms of its ability to accurately caption and analyze images. A typical RNN architecture is shown in Fig. 3.

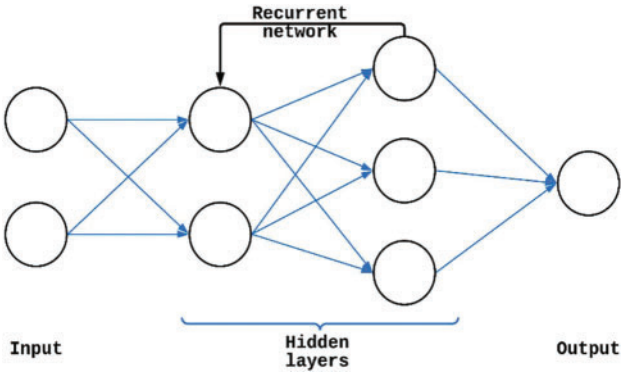

Figure 3: RNN architecture

3.2 Unsupervised Methods

3.2.1 Deep Belief Networks and Deep Boltzmann Machines

Two DL models that use the RBM as their learning module are DBM and DBN. RBM is a generative stochastic NN. In a DBN, the top two layers form an RBM and are connected undirectedly, while the lower layers are connected directedly [29]. DBMs have undirected connections between all layers of the network [29].

Restricted Boltzmann Machines

Hinton et al. [30] described a generative random NN called RBM. A typical RBM architecture is shown in Fig. 4. RBM requires a bipartite graph consisting of units from both the hidden and the visible layers. Due to the RBM architecture being a double-barreled graph, the units in both the hidden layer (H) and the visible layer (V_1) are conditionally independent of one another. Thus, it can be stated that

$$P(HV_1) = P(H_1V_1)\,P(H_2V_1)\ldots P(H_nV_1) \tag{2}$$

where given V_1 as input, H can be derived using $P(HV_1)$. Similarly, V_1 is obtained through $P(HV_1)$. By altering the parameters, the difference between V_1 and V_2 can be reduced, and the resulting H will give a fine lineament of V_1.

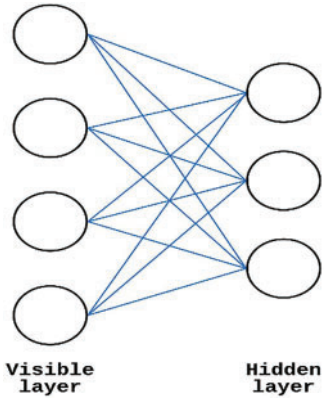

Figure 4: RBM architecture

Deep Belief Networks

DBN was first proposed by Hinton et al. [25] in 2006. This model is a probabilistic productive model that allows for the distribution of group expectations over distinct labels and data. Benefits of this layer-by-layer training method include (1) Poor initialization of the network is a result of parameter selection, which leads to a suboptimal solution up to a particular threshold. (2) Because it is an unsupervised learning technique, it draws an inference from the clustered data without the use of labelled data during training. However, the computational cost of this method is extremely high.

Deep Boltzmann Machines

DBM is built from several layers of masked elements, with the odd-numbered layers being conditionally independent of the even-numbered layers. All supervised models are co-trained during DBM training. Significant gains in both likelihood and classification accuracy can be achieved using this method of training [31]. The joint optimization of the parameters of DBMs is impractical when dealing with large datasets because of the large amount of processing time required for approximate inference, which is far more than that required by a DBN.

Deep Energy Models

A strategy for training deep architectures called DEM was introduced by Ngiam et al. [32]. With only one layer of random masked units, DEM is superior to DBMs and DBNs. This is because it avoids the problem of sharing the feature of having several random masked layers. As a result, DEM allows for more systematic training and better data-based conclusions. Many CV-related tasks favour CNNs over RBMs.

3.2.2 Autoencoder-Based Methods

Autoencoders

An autoencoder is trained to encode the input x into a representation $r(x)$ in a way that input can be reconstructed from $r(x)$ [33,34]. As a result, the input to the autoencoder is the desired output. Therefore, the dimensions of the output vectors match those of the input vectors. The corresponding code is the feature being learned as the reconstruction error is reduced. If the network is trained with a single linear hidden layer with mean squared error as the criterion, the k hidden units will learn to project the input within the range of the first k principal components of the data [35]. In contrast to PCA, a nonlinear hidden layer in an autoencoder allows it to capture multimodal features of an input distribution [36]. By adjusting the model's parameters, the average reconstruction error can be minimized. One of the most common reconstruction error measurement metrics is squared error,

$$L = \| x - f(r(x)) \|^2 \tag{3}$$

where function f is the decoder and $f(r(x))$ is the reconstruction produced by the model. A typical autoencoder architecture is shown in Fig. 5.

It is possible to represent the loss function of the reconstruction as cross-entropy if the input is viewed as bit vectors or vectors of bit probabilities.

$$L = -\sum_i \mathrm{x}_i \log \mathrm{f}_i(\mathrm{r}(\mathrm{x})) + (1 - \mathrm{x}_i) \log (1 - \mathrm{f}_i(\mathrm{r}(\mathrm{x}))) \tag{4}$$

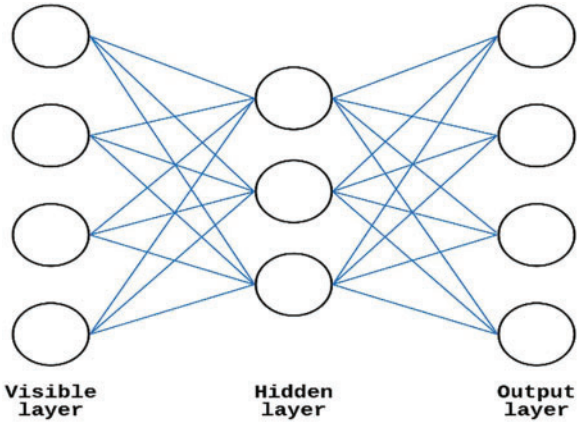

Figure 5: Autoencoder architecture

Denoising Autoencoders

Patrick et al. [37] initially introduced denoising autoencoders but its popularity was achieved only after the seminal work of Vincent et al. [38]. Denoising autoencoders are a stochastic variant of autoencoders in which the input is stochastically corrupted but the original, uncorrupted input is still used as a target for reconstruction. Denoising autoencoders have two basic purposes: encoding the input (i.e., preserving information about the input) and rectifying the effects of a stochastically applied corruption process to the autoencoder's input. Only by recording the statistical interdependencies of the inputs can the latter be achieved. The denoising autoencoder is proven to optimize a lower bound on the log-likelihood of a generative model. Several inputs are arbitrarily set to zero by the stochastic corruption process in Pascal et al. [38]. For subsets of missing patterns chosen at random, the denoising autoencoder attempts to predict corrupted values from uncorrupted ones. In essence, a sufficient requirement for fully capturing the joint distribution between a collection of variables is the capacity to predict any subset of variables from the remaining ones [29].

Stacked (Denoising) Autoencoders

By using the output code from a lower-layer denoising autoencoder as input to a higher-layer autoencoder, a deep network may be constructed. Such an architecture requires unsupervised pretraining at each layer. Denoising autoencoder training works by having each layer learn to recreate its input with as little error as possible. When the first k layers are learned, the $(k+1)^{\text{th}}$ layer can be trained because now its latent representation can be computed from the layer below it. When all of the network layers have been pre-trained, the next stage of training, known as fine-tuning, begins. When trying to minimize the amount by which a model deviates from the true answer while performing a supervised task, supervised fine-tuning should be among the methods you explore. To achieve this goal, the network's output layer's code is augmented with a logistic regression layer. After this, the resulting network is trained in a manner analogous to that of a multilayer perceptron, with only the encoders of the individual autoencoders being taken into account. This is a supervised learning stage because the target class is considered during the training process.

Training stacked autoencoders follow the same approach as training DBNs but with autoencoders rather than RBMs. Larochelle et al. [39] showed that DBNs tend to outperform stacked autoencoders.

Bengio et al. [40] also reported similar results. However, Vincent et al. [38] showed that Stacked Denoising Autoencoders can outperform DBNs.

4 Applications

Machine learning and deep learning models are widely used in various fields [41] and have been a great success. In this section, several prominent areas of application of DL-based CVs are discussed. However, considering the paucity of space, this section is not exhaustive. Beyond the application discussed here, considerable applications exist in various fields ranging from instance segmentation to Image analysis [42] to automated classification [43,44]. Some common applications of DL-based CVs are shown in Fig. 6.

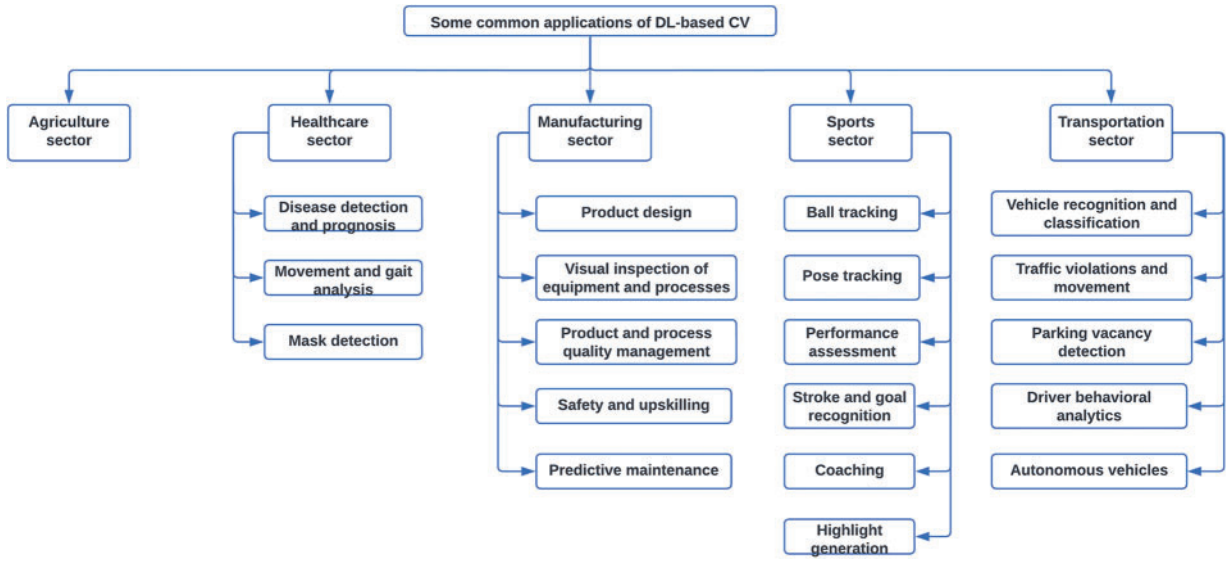

Figure 6: Some common applications of DL-based CVs

4.1 Agriculture Sector

In agriculture, DL-based CVs find wide application in areas like crop monitoring, plant disease detection, insect detection, weed detection, flowering detection, harvesting status, yield assessment, quality assessment and animal monitoring. An illustration of the application of DL-based CV in crop disease detection or yield estimation is shown in Fig. 7. Well-written specialized reviews on applications of DL-based CVs in agriculture by Tian et al. [45], Zhang et al. [46], Paul et al. [47], etc., document this domain aptly.

Whether or not there will always be enough to eat depends on how well staple crops like rice and wheat produce. Monitoring crop development has always been haphazard and inaccurate due to its reliance on human judgement. To track how plants are developing in response to their nutrient needs, CV technologies allow for continuous, non-destructive monitoring. Tian and coworkers [48] have used imaging technology to record and monitor crop growth. Monitoring crop development in real time using CV technology has the potential to detect small changes in crops owing to malnutrition far sooner than manual operations and gives a reliable and accurate basis for prompt management. Moreover, CV programs can be utilized to assess development markers or growth stages in plants. Wang et al. [49] used ResNet and ResNeXt, to detect internal mechanical damage in blueberries using

hyperspectral transmittance data. In that work, ResNeXt was shown to outperform ResNet as well as several conventional ML algorithms.

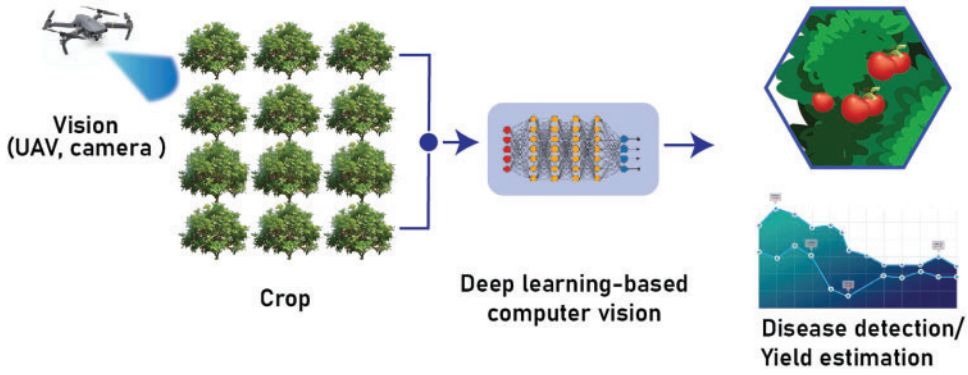

Figure 7: Application of DL-based CV in crop disease detection or yield estimation

Predicting yield loss, managing diseases, and ensuring food security all depend on automatic assessments of disease severity. In contrast to traditional methods, DL does not require time-consuming feature engineering or threshold-based image segmentation. Wang et al. [50], comparing the performance of the VGG16, VGG19, Inception-v3, and ResNet50 networks, showed that fine-tuning pre-trained deep models can greatly enhance the performance on little data. Among the tested models, an accuracy of 90.4% on the test set for the fine-tuned VGG16 model was achieved.

Recognizing and counting flying insects quickly and accurately is crucial, especially for pest management. However, the time-consuming and wasteful process of manually identifying and counting flying insects is not sustainable. Flying insects can be counted and identified easily by visual systems like YOLO.

In agronomy, weeds are regarded as undesirable plants due to their competition with crops for soil water, minerals, and other nutrients. The potential for pesticides to contaminate crops, humans, animals, and water sources is considerably reduced when they are sprayed solely in the precise locations of weeds. One of the most important factors in the growth of agriculture is the intelligent detection and elimination of weeds. Potato plants and three types of weeds can be identified using a CV system based on a NN, allowing for targeted spraying in real-time.

Wheat's heading date is a crucial metric for determining the success of a wheat harvest. The time of wheat heading can be estimated with the help of an automated CV surveillance system. The benefits of CV technology include analysis that is both dynamic and continuous, cheap cost, low error, great efficiency, and good robustness.

One of the major determinants of market prices and consumer satisfaction is the quality of agricultural products. The exterior quality checks that may be performed with the help of a CV are much more thorough than any human inspector could ever achieve. Artificial intelligence vision systems can accomplish great degrees of adaptability and repetition with minimal cost and high accuracy. Iraji et al. [51] built a SUB-adaptive neuro-fuzzy inference system using a tomato picture data set with seven input features, which involved merging multiple input features, NNs, regression, and extreme learning machines. A deep stacked sparse auto-encoder method for quality assessment was created as an alternative to analyzing features extracted from the tomato images themselves.

4.2 Healthcare Sector

DL-based CVs find wide application in the healthcare sector. An illustration of the application of DL-based CV in the healthcare sector is shown in Fig. 8.

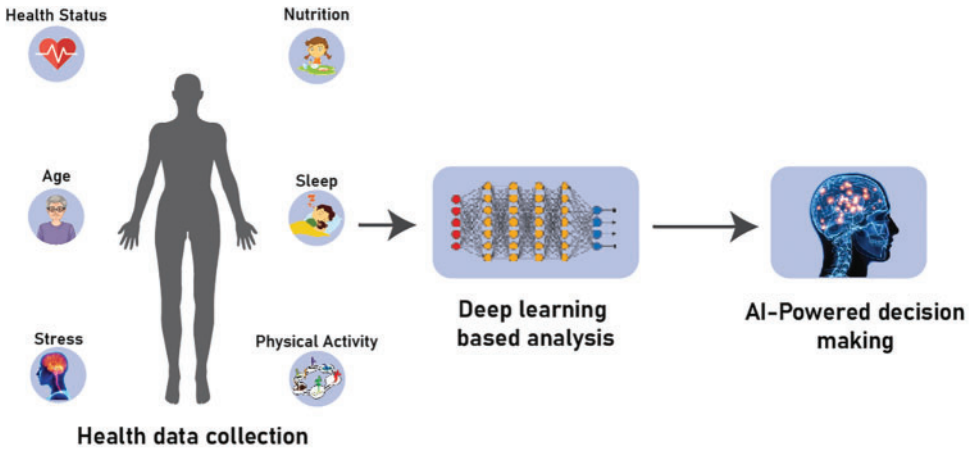

Figure 8: Application of DL-based CV in healthcare sector

4.2.1 Disease Detection and Prognosis

Many applications, such as health care and medical imagery incorporate artificial intelligence to provide efficient solutions [52,53]. Skin and breast cancer are two examples of how ML is used in the medical field. By using image recognition, researchers have been able to distinguish between cancerous and benign tumours in MRI scans and photographs. High-resolution imagery like Computed tomography and endoscopy have become an indispensable part of modern-day medical diagnosis for various diseases [54,55]. In the field of disease detection by using DL-based CV, cancer is widely researched, with ample reviews available. Domingues et al. [56] carried out an extensive literature review on esophageal cancer whereas Zahoor et al. [57] carried out a similar study on breast cancer.

Demir et al. [58] employed an object graph-based technique for the segmentation of colon WSI, whereas Nayak et al. [59] relied on a variant of the RBM to learn the important elements of the image signature. When it comes to colon cancer screening, Korbar et al. [60] trained a CNN to characterize colorectal polyps. As a means of detecting the existence of adenoma, Song et al. [61] presented a CNN model that was trained and validated on a cohort from a Chinese Hospital and used a binary classification. The system's 90.4% accuracy was on par with that of human pathologists.

MRI images can be used to diagnose brain cancers and Deep Neural Networks (DNNs) are frequently used for this purpose. Software that uses DL to detect cancers is vital to the medical field since it greatly improves the accuracy with which malignancies may be diagnosed. More precise methods of making these diagnoses are continually being developed [62]. For the challenge of segmenting gliomas using multiple modalities in an MRI scan, Urban et al. [63] presented a 3D CNN architecture. In contrast, Zikic et al. [64] created an interpretation approach to modify the 4D data, allowing the brain tumour segmentation challenge to be solved using conventional 2D-CNN architectures. Recent research on segmenting MRI images of brain tumours with DL approaches was documented in a review paper by Isin et al. [65].

CV has the potential to be utilized to manage coronavirus. The diagnosis of COVID-19 using X-rays can be done using a variety of DL CV algorithms. COVID-Net, created by researchers at Darwin AI in Canada [66], is currently the gold standard for identifying COVID-19 using digital chest X-ray radiography (CXR) pictures. Ulhaq et al. [67] recently presented a literature review on CV applications. In their paper, Chen et al. [68] proposed a CT image collection consisting of 46,096 scans of both healthy and diseased patients, all of which have been meticulously categorized by board-certified radiologists. It was gathered from 106 hospitalized patients, 51 of whom had confirmed cases of COVID-19 pneumonia and 55 healthy controls. Segmentation was performed solely with DL models to distinguish between healthy and diseased regions of CT scans. Using transfer learning on RESNET50, Li et al. [69] suggested extracting visual features from volumetric chest CT. Shorten et al. [70] highlighted the role of DL in combating COVID-19 by exploring its applications in CV. It discussed the availability of big data and how learning tasks are constructed in each application.

4.2.2 Movement and Gait Analysis

DL-based CV has the potential to automatically identify diseases and abnormalities related to movement and gait like impending strokes, blood pressure spikes, and gait difficulties. By using posture estimation in CV applications, doctors can analyze patients' motion more accurately which leads to improved diagnoses.

In this section, we discuss some recent advances in movement and gait analysis techniques, focusing mainly on the design process and optimization methods employed in the literature.

Multimodal Biometrics Systems

A method for identifying individuals was proposed by Sultana et al. [71], which takes into account both the individual's social behaviour and visual characteristics, such as their face and ears. Li et al. [72] developed a finger-based multimodal biometrics system by extracting features from various finger pattern modalities. The authors employed a fusion strategy to combine the extracted features, improving the overall recognition performance. Tiong et al. [73] proposed a multi-feature fusion CNN for multimodal face biometrics, which achieved competitive performance on benchmark datasets. The model leverages both local and global features and utilizes a fusion strategy to combine these features for improved biometric recognition.

Gait Representation and Analysis

Li et al. [74] presented DeepGait, a model for video sensor-based gait representation that blends deep convolutional features and Joint Bayesian. The model extracts features from the input video frames using a CNN and then combines these features with Joint Bayesian, which models the within-person and between-person variations for improved gait recognition performance. Wang and Yan [75] employed LSTM for cross-view gait identification, demonstrating its ability to model temporal dependencies and learn discriminative features from gait sequences.

Gesture Recognition

Babaee et al. [76] proposed a multi-stage model for gesture recognition. The model is trained using cyclic examples from the dataset and reconstructs missing images in the sequence. The network is composed of nine fully convolutional networks, each responsible for one of the network's nine recursive

sub-components. This design allows the model to learn rich spatial-temporal features from the input data, improving gesture recognition performance [77].

Wearable Sensor-Based Analysis

Potluri et al. [78] demonstrated the use of LSTM for identifying gait irregularities through a wearable sensor system. The model captures temporal dependencies in the sensor data and learns to recognize gait patterns associated with specific conditions, facilitating the detection of gait irregularities.

Although these studies have made significant progress in movement and gait analysis, challenges remain in terms of data overlap and individual identification based on behaviour, as highlighted by Sokolova and Konushin [79]. Further research is needed to develop more robust and accurate models for movement and gait analysis in various real-world applications.

4.2.3 Mask Detection

CNNs have been extensively used for object detection tasks, which typically involve identifying and localizing objects within an image [80]. Two main approaches for object detection using CNNs are one-stage and two-stage algorithms. One-stage algorithms, such as YOLO [81] and SSD [82], directly predict the class and bounding box coordinates in a single forward pass through the network. In contrast, two-stage algorithms, such as R-CNN [83] and Faster R-CNN [84], first generate a set of region proposals and then classify and refine these proposals in a second stage.

In response to the COVID-19 pandemic, CV systems have been employed to support mask-wearing enforcement by detecting and recognizing individuals wearing protective face masks. Companies like Uber have integrated facial recognition technologies into their smartphone apps to promote mask-wearing and reduce the risk of virus transmission during rides. Recent advancements in object detection and recognition techniques have enabled the development of highly accurate mask detection systems [85,86].

Two main approaches have been used in the development of mask detection systems: region-based methods and single-stage methods. Region-based methods, such as RetinaFace [87], employ a multi-scale detection architecture similar to SSD but incorporate a feature pyramid network (FPN) to fuse high- and low-level semantic information. RetinaFace has been further adapted for mask detection by Fan and Jiang [88] in their RetinaFaceMask algorithm.

On the other hand, single-stage methods, such as YOLOv3 [69], have also been used for mask detection. Li et al. [69] improved the accuracy of mask recognition by employing a mix-up and multi-scale approach based on YOLOv3 and optimizing the post-processing with a distance intersection over a union non-maximum suppression strategy. Vinh and Anh [89] developed a real-time mask detector using the YOLOv3 algorithm and the Haar cascade classifier. Nagrath et al. [90] presented SSDMNV2, a single-shot multi-box detector and MobileNetV2-based deep neural network for real-time mask detection. Gupta and Gill [91] also developed a similar mask detection system.

Recently, the YOLOX [92] algorithm has been introduced as an advanced one-stage object detection method, which has demonstrated improved performance compared to its predecessors. YOLOX could potentially be adapted for mask detection tasks, providing better performance and faster processing times compared to existing methods.

Thus, from the literature, it is seen that various approaches, including region-based and single-stage methods, have been employed to develop mask detection systems with different levels of

performance and efficiency. The choice of method depends on factors such as real-time processing requirements, hardware constraints, and the desired level of accuracy [93–95].

4.3 Manufacturing Sector

4.3.1 Product Design

Reconstructing 3D models from 2D photos of existing products is one of the most popular uses of CV in product design [96]. Milanova et al. [97] showed that range data may be used to create geometric surfaces whereas Ye et al. [98] used scanned data to create solid models. 2D photos and the spatial information in CAD models were used by Ulrich et al. [99] to estimate 3D poses. Jiang et al. [100] used a feature-based technique for fast detection of CAD model symmetry.

In product design, simulation and validation are two more crucial CV applications. When digital models are thought of as digital twins, modelling and simulation become a serious task. Alexopoulos et al. [101] proposed a framework for using Digital Twins to accelerate the training phase of ML models for smart manufacturing. They hypothesized that by creating synthetic training datasets and automatically labelling them using simulation tools, the user's involvement during training could be reduced. Testing and validation of vehicle instrument cluster design via a hybrid of hardware-in-the-loop simulation and CV is just one example of the widespread usage of modelling and simulation in vehicle instrument testing [102]. da Silva Lopes et al. [103] employed Images to make sense of the dashboard instruments and indicators. Pattern matching, edge detection, and Optical Character Recognition (OCR) methods were applied to these images using CV. Huang et al. [104] showed that to automate the validation of a vehicle's instrument cluster's design, the system in question integrates model-based testing with CV technology. The system's instrument cluster is put through its paces by simulating real-world vehicle operations and receiving all necessary signals from a hardware-in-the-loop (HIL) tester.

4.3.2 Visual Inspection of Equipment and Processes

CV has also demonstrated its capacity for doing three-dimensional position measurements of components, products, and instruments. The recognition of certain parts or features, such as the positions of screw holes, was the primary objective of the early approaches. Some later methods include contour-based approaches as well as expectation–maximization algorithms. In more recent times, laser dynamic triangulation has been implemented in manufacturing robots to detect the three-dimensional coordinates of objects [105].

CV has been used in the control of a variety of production processes. Trdic et al. [106] used CV for trajectory control of molten rock used in the creation of mineral wool. Carfagni et al. [107] measured the height and density of the fibres using CV. In the iron industry, many characteristics of bubbles were identified and studied based on froth images captured in flotation cells by Liu et al. [108]. These characteristics included bubble sizes, numbers, velocities, and stability. As pointed out by Zhao et al. [109], CV tracking techniques can also help overcome the challenge of marking and monitoring steel materials due to the high temperatures. Tsai and Lin [110] used a wavelet-based histogram matching approach in the spatial domain to extract pattern features of a multi-crystalline solar wafer. Mehrabi et al. [111] showed that features can then be used to control the flotation process. Alexopoulos et al. [112] used a DL and CNN-based CV system for automated estimation of the fill level in industrial metal scrap waste containers. They validated it in the case of the copper tube production industry with an accuracy of 77.5% to 95%.

4.3.3 Product and Process Quality Management

The use of CV systems allows for the scalable implementation of automated quality control in even the most complex production and assembly lines. In this way, DL leverages real-time object detection to produce better outcomes such as detection accuracy, speed, objectivity and reliability. AI vision inspection uses ML approaches that are more resilient than those used in classic CV systems. There has been a lot of interest in applying DL techniques to the problem of product defect detection and quality improvement. For example, automatic flaw detection in sewer pipes was the focus of Cheng and Wang's [113] DL-based solution. The detection model was taught with 3,000 images taken from sewer pipe CCTV inspection films.

Since the interior textures of timber affect sawing quality, CV is useful in assisting the creation of lumber sawing plans. Lumber CT images allow for the localization of internal flaws, which can then be utilized to inform improved sawing tactics by Bhandarkar et al. [114]. CV system created by Tao et al. [115] used sophisticated image processing techniques for checking out car screens. The devised approach substantially simplified the process of laborious validation testing and allowed for the possibility of laborious repeated tests. Similarly, Li et al. [116] carried out a turned surface inspection whereas Zhu et al. [117] worked on defect identification of spring clamps. Remote quality inspection of the production process carried out by Chiou et al. [118] is another mechanical machining application of CV-based techniques. Inspection techniques used often in automated assembly machines include blob analysis, optical flow, and running average [119]. CV methods have found extensive use in the automobile sector, such as in wheel alignment [120]. Images of cars can be analyzed using multiscale matrix fusion techniques to spot any flaws [121].

Sitthi-Amorn et al. [122] showed that CV-based systems can perform self-calibration of printheads. Printing error detection by superimposing virtual 3D models to real objects was demonstrated by Ceruti et al. [123]. Defects in 3D printing can be spotted using multi-view approaches, which shift the perspective throughout the printing process [124].

4.3.4 Safety and Upskilling

One of the core techniques of smart manufacturing is the use of CV for visual inspection. The use of vision-based inspection technologies, such as mask detection and helmet detection, for automated inspection of PPE is also on the rise. CV is useful for keeping tabs on how well workers in smart factories and construction sites are following safety regulations.

Fang et al. [125] used Faster-R-CNN to predict whether or not construction workers needed to wear safety goggles on the job. Fang et al. [126] found that the safety harness and person were recognized when operating at heights, which should help reduce the number of accidents caused by falls.

You can prevent tool and part collisions by making sure your machining setup matches the CAD model [127]. By applying the same labels to the same detected object throughout time, vision-based tracking can produce a report of the trajectory over time [128]. To avoid any accidents involving personnel and heavy machinery [129], visual data were collected using the readily available cameras in the machinery and 3D location coordinates using a monocular camera.

4.3.5 Predictive Maintenance

Predictive maintenance has emerged as a critical application of DL techniques in CV across multiple sectors like manufacturing, transportation and energy [130]. By employing advanced algorithms, these systems can analyze large volumes of sensor data which may be in the form of thermal images, vibration patterns, acoustic emissions, etc. [131,132]. The use of CNNs and RNNs has been particularly promising in extracting meaningful features and identifying potential anomalies [133]. These insights can significantly reduce downtime, extend equipment life and optimize maintenance activities [134]. As DL continues to advance, it is expected that the accuracy and efficiency of predictive maintenance will improve, with an increasing number of sectors adopting these methods to enhance their operational performance and reduce costs [135]. A typical workflow of the application of DL-based CV in predictive maintenance is shown in Fig. 9.

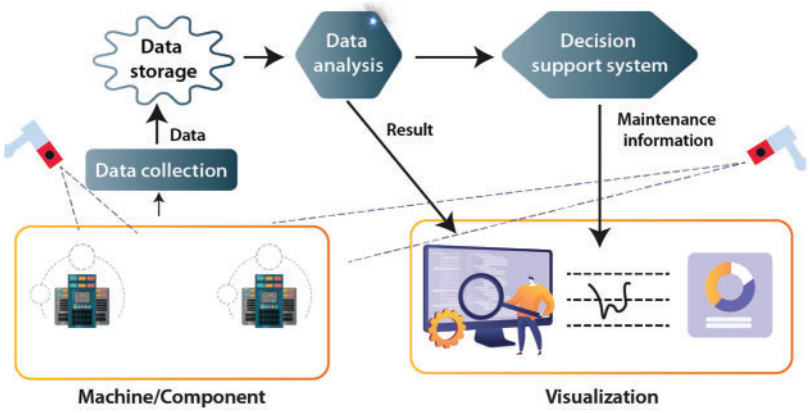

Figure 9: Typical flowchart of DL-based CV in predictive maintenance

4.4 Sports Applications

4.4.1 Ball Tracking

Object motion patterns can be detected and recorded in real-time with the help of object-tracking technology [136,137]. One of the most essential and useful pieces of data for assessing player performance and dissecting game strategies is ball trajectory records. Therefore, using DL to recognize and then track the ball in video frames is an application of ball tracking. Sports with expansive playing fields (like football, cricket, hockey, etc.) can benefit greatly from ball tracking so that commentators and analysts can better understand the action and make more informed decisions. Huang et al. [138] developed a DL-based tool called TrackNet to track the tennis ball from broadcast videos. Kamble et al. [139] developed a DL ball-tracking system for soccer. An illustrative example of a DL-based CV in sports applications is shown in Fig. 10.

4.4.2 Pose Tracking

AI vision can analyse video streams for consistent patterns in human body movement and attitude across several frames. Human posture estimation has been used in the real world, for instance in underwater and above-water movies of swimmers shot by a single stationary camera. Without having to manually annotate the body components in each video frame, the records can be used to statistically measure the athletes' performance. To automatically deduce the necessary position information and recognize a swimmer's swimming style, CNN is generally employed.

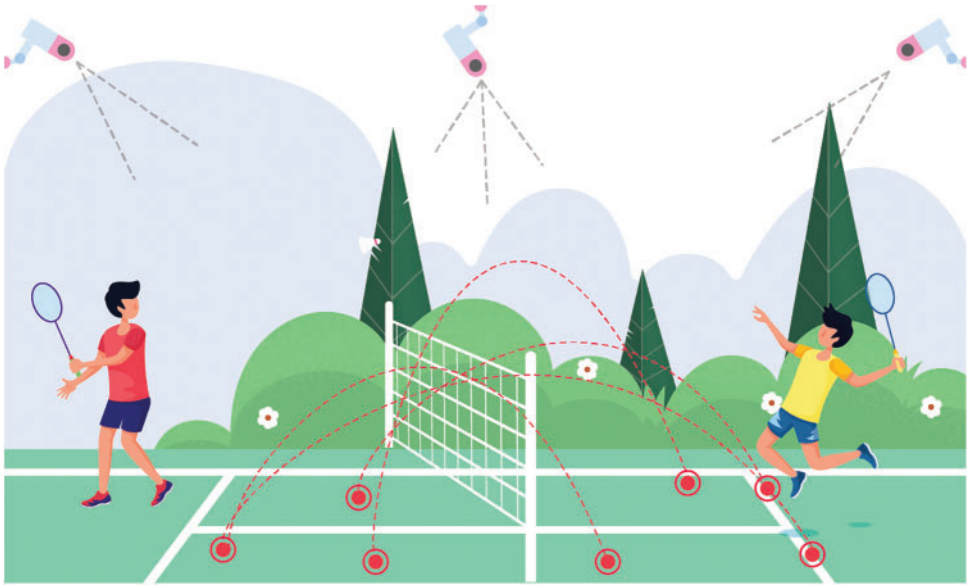

Figure 10: Example of DL-based CV in sports applications

CV algorithms can analyse sports videos shot with a single camera or from numerous angles to determine each player's position and how they move as a unit. Multiple players' 2D or 3D Pose estimates could be used for a variety of purposes in sports, such as team effectiveness, analysis of performance, capturing motion, etc. [140].

4.4.3 Performance Assessment

The limitations of traditional techniques of performance analysis are eliminated by automated detection and recognition of sport-specific motions (subjectivity, quantification, reproducibility). Data from CV systems can be combined with information from wearables and other sensors for a more complete picture of a person's condition. Common applications include tracking and analyzing bowling motions in cricket, as well as in other sports including swimming, golf, and alpine skiing. Professional sports analysts frequently engage in analysis to glean strategic and tactical insights regarding player and team behaviour. Manual video analysis, in which analysts learn to recognize and label scenes, takes a lot of time, though. For regional, team formation, event, and player analysis in team sports, CV algorithms can extract trajectory data from video material and apply movement analysis techniques (for example, in soccer team sports analysis). Song et al. [141] developed an optimized CNN based on the DL model to ensure successful detection and risk assessments of sport-medicine diseases. It uses a cloud-based loop model to create an advanced medical data network for sports medicine.

4.4.4 Stroke and Goal Recognition

Applications that use CV can recognize and categorise strokes (for example, classifying strokes in table tennis). Additional interpretations and labelled predictions of the identified instance are required for movement recognition or categorization (for example, differentiating tennis strokes as forehand or backhand). The goal of stroke recognition is to help teachers, coaches, and players better analyse table tennis games and develop their skills.

To help them make the right call, referees can use camera-based technology to verify whether or not a goal has been scored. The AI vision-based technology does not alter the conventional football equipment as sensors do. These Goal-Line Technology setups rely on high-speed cameras to triangulate the location of the ball based on its image. An algorithm for detecting balls that uses analysis of candidate ball regions to identify the ball geometry.

4.4.5 Coaching

Improved resource efficiency and shorter feedback periods for time-constrained jobs are two benefits of CV-based sports video analytics. Before the next race on the event schedule, coaches and athletes who are engaged in time-intensive notational duties, such as analyzing swim races afterwards, might benefit from quick, objective feedback. A related new area of study in CV is the design of self-training systems for physical activity in the realm of sports. Self-training is crucial in sports exercise, but there is only so far, a trainee can go without a coach's guidance. A yoga self-training app, for instance, would provide guidance on how to practice certain yoga poses correctly, to reduce the risk of injury due to improper form. Posture correction guidance can also be provided by vision-based self-training systems.

4.4.6 Highlight Generation

Making sports highlight reels is a labour-intensive endeavour that calls for some level of specialization, especially in sports with a complicated set of rules played over an extended time (e.g., Cricket). Automatic Cricket highlight generation is an example of an application that uses event-driven and excitement-based features to identify and clip crucial moments from a cricket match. As another use, a CV can be used to automatically select the most exciting moments from golf videos. Midhu and Padmanabhan [142] developed a DL Algorithm to generate highlights of a cricket video by detecting important events such as replay, pitch view, boundary view, bowler, batsman, umpire, spectator, player's gathering, etc. Concept mining is done in the second part by using the apriori algorithm and labelled frame events are input. Khan and Shao [143] presented a DL-based network SPNet that recognizes exciting sports activities by exploiting high-level visual feature sequences and automatically generates highlights. It achieved the highest performance for views, action, and situation activities with an average accuracy of 76% on the SP-2 dataset and 82% on the C-sports dataset.

4.5 Transportation Sector

4.5.1 Vehicle Recognition and Classification

There is a long history of CV applications for automatic vehicle classification. Over the years, technology for automatic vehicle classification and vehicle counting has evolved. DL technologies enable the implementation of large-scale traffic analysis systems employing conventional, low-cost security cameras [144]. Vehicles can be recognized, tracked, and categorized in many lanes at the same time using quickly growing inexpensive sensors such as closed-circuit television cameras, light detection and ranging (LiDAR), and even thermal imaging devices. Combining various sensors, such as thermal imaging, LiDAR imaging, and RGB cameras, can increase vehicle categorization accuracy. Furthermore, there are several specializations; for example, a DL-based CV solution for construction vehicle recognition has been used for safety monitoring, productivity assessment, and managerial decision-making. In another instance, Rozantsev et al. [145] proposed an approach to synthesizing training images for object detectors using a small set of real images. They estimated the rendering parameters to generate similar images with coarse 3D models [146]. They showed that significantly

better performance was achieved for drone, plane, and car detection. Sapkota et al. [147] proposed a framework for detecting small drones and estimating their positions and velocities in a 3D environment using a single moving camera. Zheng et al. [148] used YOLOv5s algorithm for ship target detection. Zheng et al. [149] and Qian et al. [150] carried out similar studies. Several well-written reviews on the applications of DL-based CV for vehicle recognition and classification are available in the literature [151,152].

4.5.2 Traffic Violations and Movement

To lower risky driving behaviour, law enforcement organizations and municipalities are boosting the implementation of camera-based traffic monitoring systems [153]. The identification of stopped automobiles in unsafe places is probably the most significant application [154]. In addition, CV techniques are increasingly being used in smart cities to automate the detection of offences like speeding, running red lights or stop signs, driving the wrong way, and making illegal turns [155]. Franklin and Mohana [156] used a YOLOV3 object detection for traffic violation detections such as signal jump, vehicle speed, and the number of vehicles.

Traffic flow analysis has been intensively researched for intelligent transportation systems utilizing both intrusive (tags, under-pavement coils, etc.) and non-invasive (cameras) technologies [157]. With the advancement of CV and AI, video analytics can now be used for omnipresent traffic cameras, which can have a significant impact on ITS and smart cities [158]. CV can be used to observe traffic movement and measure some of the factors required by traffic engineers. Rahim and Hassan [159] developed a DL-based approach with a customized f1-loss function to predict the severity of traffic crashes. Naseer et al. [160] used DL techniques to build prediction and classification models from the road accident data. Xu et al. [161] compared the driver performance using DL-based CV. Ding et al. [162] detected frauds in taxi trips using DL.

4.5.3 Parking Vacancy Detection

Visual parking spot monitoring is used to identify parking lot occupancy. CV applications, particularly in smart cities, power decentralized and efficient systems for visual parking lot occupancy monitoring based on a deep CNN. Furthermore, video-based parking management systems incorporating stereoscopic imagery (3D) or thermal cameras have been implemented. The benefits of camera-based parking lot detection include scalability for large-scale use, low maintenance and installation costs, and the possibility of reusing security cameras. Khan et al. [163] used faster R-CNN on the PKLot dataset and reported an 8% improvement over the baseline model. Valipour et al. [164] used deep CNNs to develop a parking-stall vacancy indicator system.

4.5.4 Driver Behavioral Analytics

Distracted driving detection, which includes daydreaming, cell phone use, and gazing outside the car, contributes to a significant share of road traffic fatalities globally. AI is being utilized to better understand driving patterns and discover ways to reduce road traffic accidents. Road surveillance technology, including DL-based seat belt recognition, is utilized to observe passenger compartment violations. Visual sensing, analysis, and feedback are the primary focuses of in-vehicle driver monitoring devices [165]. Directly from inside driver-facing cameras and indirectly from outward scene-facing cameras or sensors, driver behaviour can be deduced. Face and eye detection techniques based on driver-facing video analytics use algorithms for gaze direction, head pose estimate, and facial expression monitoring. Face detection algorithms can distinguish between attentive and

inattentive faces [166]. DL algorithms can distinguish between focused and unfocused eyes, as well as indicators of intoxication while driving. In real-time distraction detection, numerous vision-based applications for real-time distracted driver posture classification using multiple DL approaches are used. Streiffer et al. [167] developed a DL solution for distracted driving detection called Darnet. Guo et al. [168] combined Autoencoder and Self-organized Maps (AESOM) to extract latent features and classify driving behaviour.

4.5.5 Autonomous Vehicles

Autonomous vehicles have become a prominent application of DL techniques in CV, with transformative potential across various sectors [169,170]. The development of advanced driver assistance systems and fully autonomous driving capabilities relies heavily on the ability to accurately perceive and interpret the surrounding environment in real time [171]. CNN has been widely adopted for tasks such as object detection, semantic segmentation, and depth estimation [172]. RNNs and reinforcement learning techniques have shown promise in predicting future states and making intelligent driving decisions [173]. Despite the significant progress in recent years, challenges remain in ensuring the safety and reliability of autonomous vehicles, especially under complex and adverse conditions [174]. Future trends in deep learning research for autonomous vehicles are expected to focus on enhancing robustness, generalization, and interpretability while integrating with other emerging technologies, such as edge computing and 5G connectivity [175,176]. Janai et al. [177] provided a comprehensive survey of CV for autonomous vehicles which covered various datasets, techniques and open problems. Güzel [178] in his review covered the intricacies involved in autonomous vehicle navigation using CV. A typical workflow of the application of DL-based CV in predictive maintenance is shown in Fig. 11.

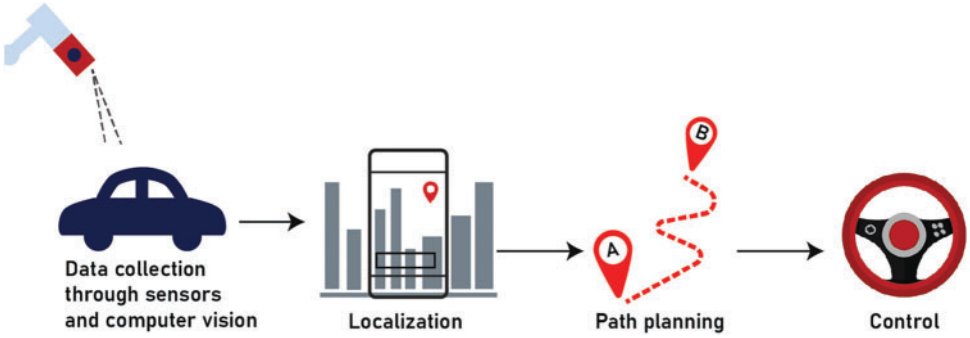

Figure 11: Typical flowchart of DL-based CV in autonomous vehicles

5 Datasets

The inputs to DL-based CV systems are generally in the form of images and video clips. Several open-source datasets are available for implementing and testing DL frameworks for CV applications. A few of them are compiled below in Table 1.

Table 1: Popular datasets for DL-based CV applications

Dataset	Data type	Source link	Remarks
MNIST [28]	Grayscale images	http://yann.lecun.com/exdb/mnist/	60,000 training and 10,000 testing 28 × 28 images in 10 classes
ImageNet [179]	Colour images	http://www.image-net.org/	millions of cleanly sorted images
IMDB-wiki [180]	Colour images	https://data.vision.ee.ethz.ch/cvl/rrothe/imdb-wiki/	523,051 face images with gender and age labels
MS-COCO [181]	Colour images	http://cocodataset.org/#home	2.5 million labelled instances in 328 k images of 91 objects types
MPII Human pose [182]	Colour images	http://human-pose.mpi-inf.mpg.de/#	25K images containing over 40K people with annotated body joints of 410 human activities
Open images [183]	Colour images	https://storage.googleapis.com/openimages/web/factsfigures.html	16 million bounding boxes for 600 object classes on 1.9 million images
CIFAR-10 [184]	Colour images	https://www.cs.toronto.edu/~kriz/cifar.html	60000 32 × 32 color images with 10 classes
CIFAR-100 [185]	Colour images	https://www.cs.toronto.edu/~kriz/cifar.html	60000 32 × 32 color images with 100 classes
COVID-19 X-Ray (V7)	Images	https://github.com/v7labs/covid-19-xray-dataset	6500 images of AP/PA chest X-rays of 517 cases
LSUN [186]	Images	https://paperswithcode.com/dataset/lsun	Approx. 1 million labelled images for each of 10 scene categories and 20 object categories
LabelMe-12-50k [187]	JPEG images	https://www.kaggle.com/datasets/dschettler8845/labelme-12-50k	40,000 training, 10,000 testing images with 12 classes

(Continued)

Table 1 (continued)

Dataset	Data type	Source link	Remarks
Cityscapes [188]	Stereo video	https://www.cityscapes-dataset.com/	5,000 frames of high-quality pixel-level annotations and 20,000 weakly annotated frames
Kinetics-700 [189]	Video	https://paperswithcode.com/dataset/kinetics-700	650,000 video clips and covers 700 human action classes
20BN-Something-something [190]	Video	https://paperswithcode.com/dataset/something-something-v2	168,913 training, 24,777 validations, 27,157 testing videos
VisualQA [190]	Images	https://visualqa.org/	265,016 images, an average of 5.4 questions per image, 10 ground truth answers per question
SVHN [191]	Images	http://ufldl.stanford.edu/housenumbers/	6,30,420 images of 10 classes
Fashion-MNIST [192]	Images	https://github.com/zalandoresearch/fashion-mnist	70,000 images in 10 classes
YouTube-boundingboxes [193]	Video	https://research.google.com/youtube-bb/	380,000 videos, approx. 19 s long, 10.5 million human annotations, 5.6 million bounding boxes
OAK [194]	Video	https://oakdata.github.io/	80 video snippets (∼17.5 hours) for 105 object categories in outdoor scenes
Prophesee GEN1 automotive detection [195]	Video	https://www.prophesee.ai/2020/01/24/prophesee-gen1-automotive-detection-dataset/	39 hours of automotive recordings, 255,000 labels
SYNTHIA-AL [196]	Video	http://synthia-dataset.net/downloads/	9400 multi-viewpoint photo-realistic frames, 13 classes

(Continued)

Table 1 (continued)

Dataset	Data type	Source link	Remarks
THGP [197]	Video	https://drive.google.com/file/d/1hF7Vr6g0fG56Oy3Jdnm2t9Y3TK9W9bn4	5960 video frames
USC-GRAD-STDdb [198]	Video	https://gitlab.citius.usc.es/brais.bosquet/USC-GRAD-STDdb	115 video segments, 25,000 annotated frames
FONTS invalid source specified	Images	https://github.com/davidstutz/cvpr2019-adversarial-robustness	randomly transformed characters "A" to "J"
Synthetic digits invalid source specified	Images	https://www.kaggle.com/datasets/prasunroy/synthetic-digits	12,000 synthetically generated English digit images

6 Current Challenges

In this section, several prominent challenges of DL-based CVs are discussed. Fig. 12 shows some of the current challenges of DL-based CVs.

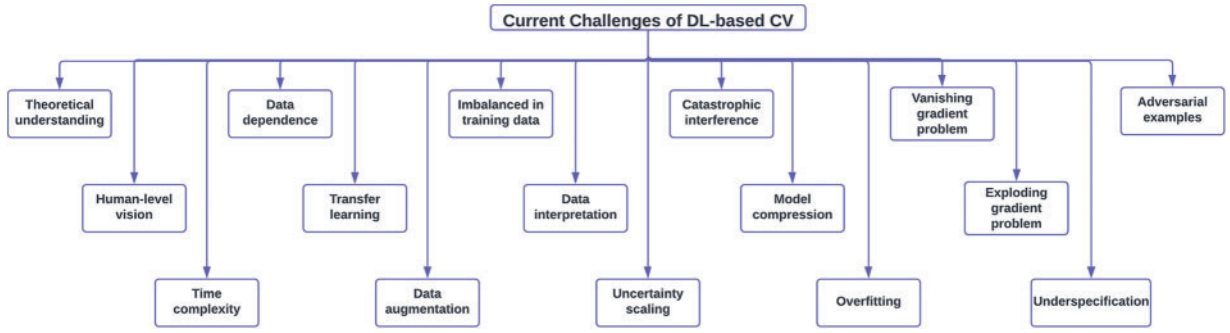

Figure 12: Current challenges of DL-based CVs

Theoretical understanding: Although deep learning methods have produced encouraging results in solving computer vision challenges, the underlying theory is poorly understood, and it is unclear which architectures should perform best [199]. It is difficult to decide which structure, number of layers, or number of nodes in each layer is appropriate for a given task, and it requires specific knowledge to choose sensible values such as the learning rate, regularizer strength, etc. Historically, the design of the architecture has been determined ad hoc [200].

Human-level vision: Even with simple visual representations or changes to geometric transformations, background variation, and occlusion, the human visual system excels at computer vision tasks to a remarkable degree. Human-level vision can refer to bridging the semantic gap in terms of precision or integrating new insights from studies of the human brain into machine-learning architectures. CNN

mimics the structure of the human brain and generates multi-layer activations for mid-level or high-level features, as opposed to the traditional low-level features.

Time complexity: Early CNNs were deemed inapplicable for real-time applications because they required significant computational resources [201]. One of the trends is the development of new architectures that enable the real-time operation of CNNs.

Data dependence: For DL to produce a well-behaved performance model, an enormous quantity of data is required; as the amount of data increases, a more well-behaved performance model can be produced. Various approaches are available for effectively addressing this concern [199].

- Utilization of the transfer-learning following the collection of data from similar jobs. Though the transferred data will not directly enhance the actual data, it will aid in enhancing both the original input data representation and its mapping function [202].
- Employing a well-trained model from a comparable assignment and fine-tuning the conclusion of two layers or even one layer depending on the restricted initial data [203].
- Data augmentation is extremely useful for enhancing image data, as image translation, mirroring, and rotation typically do not affect the image label [204].
- There is potential for using simulated data to increase the size of the training set.

Transfer learning: To combat the issue of insufficient training data, transfer learning is often proposed as a solution due to its relaxation of the presumption that the training data must be independent and identically distributed with the test data. Since the model in the target domain does not need to be trained from the start and the distribution of training and test data is not required to be the same, the amount of data and time required for training can be significantly reduced through transfer learning. Deep learning networks learn both their bias and their weights as they are trained on massive amounts of data. These weights are then used to retrain or test a new model in a variety of networks. This means the novel model can make use of pre-trained weights rather than starting with a blank slate during training.

Data augmentation: Data augmentation strategies can be used to expand the amount of available data while minimizing the risk of overfitting. If you have an issue but only a small amount of data, these methods are the way to go. The term data augmentation describes a group of techniques used to expand and enhance data sets used for training purposes. As a result, using these methods allows DL networks to function more effectively. Label-preserving modifications are used as a method of data augmentation for deep neural network audio modelling to handle data Sparsity. For both deep neural networks (DNNs) and convolutional neural networks (CNN), the vocal folds length perturbations (VTLP) and stochastic features mappings (SFM) data augmentation techniques are examined (CNNs). In addition, it is suggested to integrate VTLP and SFM as complementing techniques using a two-stage data augmentation strategy built on a stacked architecture. Assamese and Haitian Creole, two IARPA Babel programme management languages, are the subjects of research, and it is noted that these languages perform better in terms of automated voice recognition (ASR) and keyword search (KWS).

Imbalanced in training data: If say negative samples are more common than positive ones, it can cause problems when training a DL model. This can be solved by using appropriate loss evaluation criteria and checking that the model works effectively with both small and large classes.

Data interpretation: Despite the efficacy of DL methods, it is difficult to foresee when they will fail due to the lack of transparency. This makes their application to fields like medicine, bioinformatics,

etc. somewhat unsuitable. Even though DL methods are sometimes treated as a black box, they can be understood by rigorous mathematical analysis.

Uncertainty scaling: The confidence score measures the reliability of the model's forecast. The confidence score is an important feature in any setting since it protects users from placing faith in inaccurate forecasts. Uncertainty scaling is often critically important in fields like medicine, bioinformatics, etc. where it is used to assess the quality of machine learning-based illness diagnosis and the accuracy of automated clinical decisions [205].

Catastrophic interference: Neural networks tend to suddenly and completely forget everything they've ever learnt to make room for new knowledge. It appears that representation overlap in the buried layer of distributed neural networks is the main cause of catastrophic interference [206]. When several weights where knowledge is stored are modified, it perhaps messes with the previous knowledge, leading to catastrophic forgetfulness. To solve this problem, historical and current data can be used to train a brand-new model. But it is time-consuming and computationally demanding.

Model compression: The goal of model compression is to reduce the complexity of a model without compromising its predictive ability. This field of study, known as model compression, has seen a lot of activity over the past few years as researchers seek to find ways to deploy cutting-edge deep networks in low-power and resource-limited devices without sacrificing accuracy. Several strategies have been proposed to reduce the overall size of deep networks, including parameter pruning, low-rank factorization, and weight quantization.

Overfitting: Overfitting is a problem in machine learning that happens when a model performs well on training data but not on novel data. According to Everitt and Skronda [207], "an overfitted model is a mathematical model that contains more parameters than can be justified by the data".

Vanishing gradient problem: As the number of network layers increases, the partial derivative of the loss function lowers until it eventually becomes near zero and disappears. Changing the network's activation function is the quickest and easiest way to fix the issue. Choose an activation function other than sigmoid, for as ReLU. Another option is residual networks, which enable direct connections from later layers to the underlying ones. To sum up, batch normalization layers can fix the problem. The derivatives vanish when the input space, |x|, is tiny, giving rise to the vanishing gradient problem. Batch normalization alleviates this issue by adjusting the input so that |x| is kept from exceeding the sigmoid function's bounds.

Exploding gradient problem: The derivatives or slope of a gradient can explode if they grow exponentially greater with each successive backward layer in backpropagation. This is the exact opposite of the vanishing gradients problem. Not the activation function itself is to blame; rather, it is the weights that produce this issue. Common methods for preventing explosive gradients include modifying the error derivative before re-feeding it into the network to update the weights. The risk of an overflow or underflow is greatly reduced by rescaling the error derivative, and hence the updates to the weights.

Underspecification: When put to the test in real-world applications like computer vision, medical imaging, natural language processing, and medical genomics, ML models, deep learning models included, often exhibit surprisingly bad behaviour [208]. Underspecification is to blame for the poor performance. It has been demonstrated that even slight adjustments can drive a model to a whole new solution and result in different predictions in deployment areas. To solve the underspecification problem, various methods have been developed. One of these is developing "stress tests" to determine how well a model performs on real-world data and identify any problems it may have.

Adversarial examples: Adversarial examples are a well-known challenge in DL and CV applications [209]. These examples are specifically crafted to mislead ML models by adding imperceptible perturbations to the input data. This causes even the high-accuracy models to make incorrect predictions with a high degree of confidence [210]. Adversarial examples can be created for a wide range of tasks, such as image classification, object detection, semantic segmentation, etc. [211]. One of the main challenges of adversarial examples is that they pose security risks in critical applications such as autonomous driving, facial recognition, and medical diagnosis [212]. Attackers can easily generate adversarial examples to deceive even the best models, which can lead to catastrophic consequences. Thus, understanding and addressing the vulnerabilities of DL models to adversarial attacks is critical [213]. In the last few years, several researchers have proposed numerous interesting approaches to improve the robustness of DL models against adversarial examples. One commonly used strategy is adversarial training, which includes training the model on both clean and adversarial examples, thereby making the models more resistant to such attacks [214]. Other techniques like defensive distillation, feature squeezing, gradient regularization, etc. Reference [215] are also viable alternatives against adversarial examples. Recent advancements in adversarial examples have also paved the way for the development of inverse adversaries. These are generated by inverting the direction of the edge detection process used to create adversarial examples. Inverse adversaries have thus emerged as an important area of research for improving the robustness of DL models [216].

7 Future Trends

In this section, the future trends of DL-based CVs are discussed. Fig. 13 shows some of the future trends of DL-based CVs.

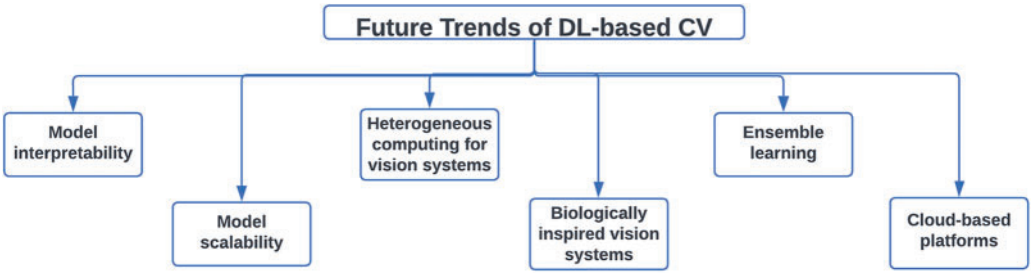

Figure 13: Future trends of DL-based CVs

Model interpretability: For applications that need explanations of the features involved in modelling, the interpretability of DNN models has always been a limiting constraint. High accuracy in Deep Learning models comes at the cost of a high level of abstraction. One path forward is to make DL models more easily understood and visualized.

Model scalability: Due to the complexity and time-consuming nature of DL models, it is important to look at model scalability as a key criterion in a model's evaluation. It has been well established in the literature that often by increasing model complexity, better accuracy can be obtained. Scalable model is an idea worth exploring.

Heterogeneous computing for vision systems: The state-of-the-art CV algorithms consist of many interdependent functional blocks that can be implemented on multiple hardware platforms. As a result, building computer vision systems for a single target hardware platform is wasteful and often fails to

fulfil performance and power budgets, especially for embedded and remote operations. Thus, switching to a heterogeneous architecture. However, there has been very limited research in this area.

Biologically inspired vision systems: Biological vision systems can quickly and precisely analyze vast quantities of visual input to make survival decisions, such as where to find food, how to avoid danger, and how to return to a safe location. A model of how the brain processes visual information is an effective method. Future applications for these biologically inspired vision systems may include surveillance systems and intelligent sensors capable of alerting vehicles to pedestrians and other obstacles.

Ensemble learning: Ensemble learning integrates many models to reduce the variance of neural network models by training numerous models instead of a single model and by combining their predictions.

Cloud-based platforms: A significant role for cloud-based platforms is expected to grow in the creation of computational DL applications. With cloud computing, it is possible to organize all of this data. Also, productivity rises as expenses fall. The flexibility it offers in training DL architectures is also a major plus.

8 Conclusions

This paper presents a comprehensive review of deep learning methods applied to computer vision applications. Convolutional Neural Networks, Recurrent Neural Networks, Autoencoders, Deep Belief Networks and Deep Boltzmann Machines are discussed in detail. The application of deep learning methods to computer vision is discussed as per application sectors ranging from agriculture to health to manufacturing to sports to transport. It is shown through the diverse literature review that deep learning has allowed computer vision systems to surpass human performance on several recognition tasks. These superior performances are due to the synergistic effects of advanced hardware, better/larger datasets, larger models, innovative algorithms and enhanced network architectures.

Despite the advancements, it is challenging to forecast the future of deep NNs. The partial reason for this is the still limited understanding of the functioning of the human brain, the inferotemporal pathway of the human visual system, etc. which are inspirations to the NNs. Another challenge is to conclude under what conditions DL models will perform well or outperform other approaches, and how to establish the ideal structure for a specific task.

Acknowledgement: Authors acknowledge the support of their respective institutes.

Funding Statement: This article was supported by the Project SP2023/074 Application of Machine and Process Control Advanced Methods supported by the Ministry of Education, Youth and Sports, Czech Republic.

Author Contributions: Conceptualization, Narayanan Ganesh, Rajendran Shankar, Miroslav Mahdal, Janakiraman Senthil Murugan, Kanak Kalita; Formal analysis, Narayanan Ganesh, Rajendran Shankar, Miroslav Mahdal, Janakiraman Senthil Murugan, Kanak Kalita; Methodology, Narayanan Ganesh, Rajendran Shankar, Miroslav Mahdal, Janakiraman Senthil Murugan, Jasgurpreet Singh Chohan; Visualization, Jasgurpreet Singh Chohan, Kanak Kalita; Writing—original draft, Narayanan Ganesh, Rajendran Shankar, Miroslav Mahdal, Janakiraman Senthil Murugan; Writing—review & editing, Narayanan Ganesh, Rajendran Shankar, Miroslav Mahdal, Janakiraman

Senthil Murugan, Jasgurpreet Singh Chohan, Kanak Kalita. All authors have read and agreed to the published version of the manuscript.

Availability of Data and Materials: This paper is a review that summarizes existing methods and literature findings. This investigation utilized only data obtained from publicly accessible sources. These datasets are accessible via the sources listed in the References section of this paper. As the data originates from publicly accessible repositories, its accessibility is unrestricted.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

1. Misra, S., Li, H., He, J. (2019). *Machine learning for subsurface characterization*. Cambridge, MA, USA: Gulf Professional Publishing.
2. Montejo-Ráez, A., Jiménez-Zafra, S. M. (2022). Current approaches and applications in natural language processing. *Applied Sciences, 12(10),* 4859. https://doi.org/10.3390/app12104859
3. Hao, T., Li, X., He, Y., Wang, F. L., Qu, Y. (2022). Recent progress in leveraging deep learning methods for question answering. *Neural Computing and Applications, 34(4),* 2765–2783. https://doi.org/10.1007/s00521-021-06748-3
4. Gumbs, A. A., Grasso, V., Bourdel, N., Croner, R., Spolverato, G. et al. (2022). The advances in computer vision that are enabling more autonomous actions in surgery: A systematic review of the literature. *Sensors, 22(13),* 4918. https://doi.org/10.3390/s22134918
5. Pinto, G., Wang, Z., Roy, A., Hong, T., Capozzoli, A. (2022). Transfer learning for smart buildings: A critical review of algorithms, applications, and future perspectives. *Advances in Applied Energy, 5,* 100084. https://doi.org/10.1016/j.adapen.2022.100084
6. Karavarsamis, S., Gkika, I., Gkitsas, V., Konstantoudakis, K., Zarpalas, D. (2022). A survey of deep learning-based image restoration methods for enhancing situational awareness at disaster sites: The cases of rain, snow and haze. *Sensors, 22(13),* 4707. https://doi.org/10.3390/s22134707
7. Guo, Y., Liu, Y., Oerlemans, A., Lao, S., Wu, S. et al. (2016). Deep learning for visual understanding: A review. *Neurocomputing, 187,* 27–48. https://doi.org/10.1016/j.neucom.2015.09.116
8. Viola, P., Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 1–10. Kauai, USA. https://doi.org/10.1109/CVPR.2001.990517
9. Lienhart, R., Maydt, J. (2002). An extended set of haar-like features for rapid object detection. *International Conference on Image Processing*, pp. 1–10. Rochester, USA. https://doi.org/10.1109/ICIP.2002.1038171
10. Chai, J., Zeng, H., Li, A., Ngai, E. W. T. (2021). Deep learning in computer vision: A critical review of emerging techniques and application scenarios. *Machine Learning with Applications, 6,* 100134. https://doi.org/10.1016/j.mlwa.2021.100134
11. McCulloch, W. S., Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics, 5(4),* 115–133. https://doi.org/10.1007/BF02478259
12. Hebb, D. O. (1949). *The organization of behavior: A neuropsychological theory*. New York, NY, USA: John Wiley and Sons, Inc.
13. Yoo, H. J. (2015). Deep convolution neural networks in computer vision: A review. *IEIE Transactions on Smart Processing and Computing, 4(1),* 35–43. https://doi.org/10.5573/IEIESPC.2015.4.1.035
14. Rosenblatt, F. (1958). The Perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review, 65(6),* 386–408. https://doi.org/10.1037/h0042519

15. Widrow, B., Hoff, M. E. (1960). Adaptive switching circuits. *IRE WESCON Convention Record, 4,* 96–104. https://doi.org/10.21236/AD0241531
16. Minsky, M., Papert, S. (1969). *Perceptron expanded edition*. Cambridge, MA, USA: MIT Press.
17. Kohonen, T. (1972). Correlation matrix memories. *IEEE Transactions on Computers, C-21(4),* 353–359. https://doi.org/10.1109/TC.1972.5008975
18. Anderson, J. A. (1972). A simple neural network generating an interactive memory. *Mathematical Biosciences, 14(3–4),* 197–220. https://doi.org/10.1016/0025-5564(72)90075-2
19. Grossberg, S. (1976). Adaptive pattern classification and universal recoding: I. Parallel development and coding of neural feature detectors. *Biological Cybernetics, 23(3),* 121–134. https://doi.org/10.1007/BF00344744
20. Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences of the United States of America, 79(8),* 2554–2558. https://doi.org/10.1073/pnas.79.8.2554
21. Werbos, P. (1974). *Beyond regression: "New tools for prediction and analysis in the Behavioral Science." (Ph.D. Thesis)*. Harvard University.
22. Rumelhart, D. E., Hinton, G. E., Williams, R. J. (1986). Learning internal representations by error propagation. In: Rumelhart, D. E., Mcclelland, J. L. (Eds.), *Parallel distributed processing: Explorations in the microstructure of cognition: Foundations*, pp. 318–362. Cambridge, MA, USA: MIT Press.
23. LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E. et al. (1989). Backpropagation applied to handwritten ZIP code recognition. *Neural Computation, 1(4),* 541–551. https://doi.org/10.1162/neco.1989.1.4.541
24. Krizhevsky, A., Sutskever, I., Hinton, G. E. (2017). ImageNet classification with deep convolutional neural networks. *Communications of the ACM, 60(6),* 84–90. https://doi.org/10.1145/3065386
25. Hinton, G. E., Osindero, S., Teh, Y. W. (2006). A fast learning algorithm for deep belief nets. *Neural Computation, 18(7),* 1527–1554. https://doi.org/10.1162/neco.2006.18.7.1527
26. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S. et al. (2015). Going deeper with convolutions. *Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–9. Boston, MA, USA. https://doi.org/10.1109/CVPR.2015.7298594
27. LeCun, Y., Bengio, Y., Hinton, G. (2015). Deep learning. *Nature, 521(7553),* 436–444. https://doi.org/10.1038/nature14539
28. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE, 86(11),* 2278–2324. https://doi.org/10.1109/5.726791
29. Voulodimos, A., Doulamis, N., Doulamis, A., Protopapadakis, E. (2018). Deep learning for computer vision: A brief review. *Computational Intelligence and Neuroscience, 2018,* 7068349. https://doi.org/10.1155/2018/7068349
30. Hinton, G. E., Sejnowski, T. J. (1986). Learning and relearning in Boltzmann machines. *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, 1,* 282–317.
31. Younes, L. (1999). On the convergence of Markovian stochastic algorithms with rapidly decreasing ergodicity rates. *Stochastics and Stochastic Reports, 65(3–4),* 177–228. https://doi.org/10.1080/17442509908834179
32. Ngiam, J., Chen, Z., Koh, P. W., Ng, A. Y. (2011). Learning deep energy models. *Proceedings of the 28th International Conference on Machine Learning (p. ICML-11)*, pp. 1105–1112. Bellevue, WA, USA.
33. Bengio, Y. (2009). Learning deep architectures for AI. *Foundations and Trends® in Machine Learning, 2,* 1–127. https://doi.org/10.1561/2200000006
34. Li, J., Xu, K., Chaudhuri, S., Yumer, E., Zhang, H. et al. (2017). GRASS: Generative recursive autoencoders for shape structures. *ACM Transactions on Graphics, 36(4),* 1–14. https://doi.org/10.1145/3072959.3073637

35. Bourlard, H., Kamp, Y. (1988). Auto-association by multilayer perceptrons and singular value decomposition. *Biological Cybernetics, 59(4–5),* 291–294. https://doi.org/10.1007/BF00332918
36. Japkowicz, N., Jos, S. J., Gluck, M. A. (2000). Nonlinear autoassociation is not equivalent to PCA. *Neural Computation, 12(3),* 531–545. https://doi.org/10.1162/089976600300015691
37. Gallinari, P., Lecun, Y., Thiria, S., Soulie, F. F. (1987). Mémoires associatives distribuées: Une comparaison (distributed associative memories: A comparison). *Proceedings of the COGNITIVA 87*, Paris, La Villette.
38. Vincent, P., Larochelle, H., Bengio, Y., Manzagol, P. A. (2008). Extracting and composing robust features with denoising autoencoders. *Proceedings of the 25th International Conference on Machine Learning*, pp. 1096–1103. Helsinki, Finland. https://doi.org/10.1145/1390156.1390294
39. Larochelle, H., Erhan, D., Courville, A., Bergstra, J., Bengio, Y. (2007). An empirical evaluation of deep architectures on problems with many factors of variation. *Proceedings of the 24th International Conference on Machine Learning*, pp. 473–480. New York, NY, USA. https://doi.org/10.1145/1273496.1273556
40. Benigo, Y., Lamblin, P., Popovici, D., Larochelle, H. (2007). Greedy layer-wise training of deep networks. In: *Advances in neural information processing systems*, pp. 153–160. Cambridge, MA, USA: MIT Press.
41. Zhang, J., Zhu, C., Zheng, L., Xu, K. (2021). ROSEFusion: Random optimization for online dense reconstruction under fast camera motion. *ACM Transactions on Graphics, 40(4),* 1–17.
42. Liu, A. -A., Zhai, Y., Xu, N., Nie, W., Li, W. et al. (2022). Region-aware image captioning via interaction learning. *IEEE Transactions on Circuits and Systems for Video Technology. Transactions of the on Circuits and Systems for Video Technology, 32(6),* 3685–3696. https://doi.org/10.1109/TCSVT.2021.3107035
43. Zheng, W., Tian, X., Yang, B., Liu, S., Ding, Y. et al. (2022). A few shot classification methods based on multiscale relational networks. *Applied Sciences, 12(8),* 4059. https://doi.org/10.3390/app12084059
44. Cao, B., Fan, S., Zhao, J., Tian, S., Zheng, Z. et al. (2021). Large-scale many-objective deployment optimization of edge servers. *IEEE Transactions on Intelligent Transportation Systems, 22(6),* 3841–3849. https://doi.org/10.1109/TITS.2021.3059455
45. Tian, H., Wang, T., Liu, Y., Qiao, X., Li, Y. (2020). Computer vision technology in agricultural automation—A review. *Information Processing in Agriculture, 7(1),* 1–19. https://doi.org/10.1016/j.inpa.2019.09.006
46. Zhang, Q., Liu, Y., Gong, C., Chen, Y., Yu, H. (2020). Applications of deep learning for dense scenes analysis in agriculture: A review. *Sensors, 20(5),* 1520. https://doi.org/10.3390/s20051520
47. Paul, A., Ghosh, S., Das, A. K., Goswami, S., Das Choudhury, S. et al. (2020). A review on agricultural advancement based on computer vision and machine learning. In: Mandal, J. K., Bhattacharya, D. (Eds.), *Emerging technology in modelling and graphics*, pp. 567–581. Singapore, Springer. https://doi.org/10.1007/978-981-13-7403-6_50
48. Tian, H., Huang, N., Niu, Z., Qin, Y., Pei, J. et al. (2019). Mapping winter crops in China with multi-source satellite imagery and phenology-based algorithm. *Remote Sensing, 11(7),* 820. https://doi.org/10.3390/rs11070820
49. Wang, Z., Hu, M., Zhai, G. (2018). Application of deep learning architectures for accurate and rapid detection of internal mechanical damage of blueberry using hyperspectral transmittance data. *Sensors, 18(4),* 1126. https://doi.org/10.3390/s18041126
50. Wang, G., Sun, Y., Wang, J. (2017). Automatic image-based plant disease severity estimation using deep learning. *Computational Intelligence and Neuroscience, 2017,* 2917536–2917538. https://doi.org/10.1155/2017/2917536
51. Iraji, M. S. (2019). Comparison between soft computing methods for tomato quality grading using machine vision. *Journal of Food Measurement and Characterization, 13(1),* 1–15. https://doi.org/10.1007/s11694-018-9913-2

52. Zhou, L., Ye, Y., Tang, T., Nan, K., Qin, Y. (2022). Robust matching for SAR and optical images using multiscale convolutional gradient features. *IEEE Geoscience and Remote Sensing Letters, 19,* 1–5. https://doi.org/10.1109/LGRS.2021.3105567
53. Zhou, G., Yang, F., Xiao, J. (2022). Study on pixel entanglement theory for imagery classification. *IEEE Transactions on Geoscience and Remote Sensing, 60,* 1–18. https://doi.org/10.1109/TGRS.2022.3167569
54. Zhuang, Y., Jiang, N., Xu, Y. (2022). Progressive distributed and parallel similarity retrieval of large CT image sequences in mobile telemedicine networks. *Wireless Communications and Mobile Computing, 2022,* 1–13. https://doi.org/10.1155/2022/6458350
55. Zhuang, Y., Chen, S., Jiang, N., Hu, H. (2022). An effective WSSENet-based similarity retrieval method of large Lung CT Image databases. *KSII Transactions on Internet and Information Systems, 16(7)*. https://doi.org/10.3837/tiis.2022.07.013
56. Domingues, I., Sampaio, I. L., Duarte, H., Santos, J. A. M., Abreu, P. H. (2019). Computer vision in esophageal cancer: A literature review. *IEEE Access, 7,* 103080–103094. https://doi.org/10.1109/ACCESS.2019.2930891
57. Zahoor, S., Lali, I. U., Khan, M. A., Javed, K., Mehmood, W. (2020). Breast cancer detection and classification using traditional computer vision techniques: A comprehensive review. *Current Medical Imaging, 16(10),* 1187–1200. https://doi.org/10.2174/1573405616666200406110547
58. Gunduz-Demir, C., Kandemir, M., Tosun, A. B., Sokmensuer, C. (2010). Automatic segmentation of colon glands using object-graphs. *Medical Image Analysis, 14(1),* 1–12. https://doi.org/10.1016/j.media.2009.09.001
59. Nayak, N., Chang, H., Borowsky, A., Spellman, P., Parvin, B. (2013). Classification of tumor histopathology via sparse feature learning. *Proceedings IEEE International Symposium on Biomedical Imaging 10th International Symposium on Biomedical Imaging*, pp. 410–413. San Francisco, CA, USA. https://doi.org/10.1109/ISBI.2013.6556499
60. Korbar, B., Olofson, A. M., Miraflor, A. P., Nicka, C. M., Suriawinata, M. A. et al. (2017). Deep learning for classification of colorectal polyps on whole-slide images. *Journal of Pathology Informatics, 8,* 30. https://doi.org/10.4103/jpi.jpi_34_17
61. Song, Z., Yu, C., Zou, S., Wang, W., Huang, Y. et al. (2020). Automatic deep learning-based colorectal adenoma detection system and its similarities with pathologists. *BMJ Open, 10(9),* e036423. https://doi.org/10.1136/bmjopen-2019-036423
62. Chen, L. C. O., Qin, L., Xu, Z., Yin, Y., Wang, X. et al. (2021). Corynoxine protects dopaminergic neurons through inducing autophagy and diminishing neuroinflammation in rotenone-induced animal models of Parkinson's disease. *Frontiers in Pharmacology, 13,* 642900.
63. Urban, G., Bendszus, M., Hamprecht, F. A., Kleesiek, J. (2014). Multi-modal brain tumor segmentation using deep convolutional neural networks. *Proceedings of the BRATS-MICCAI*, pp. 31–35. Boston, MA, USA.
64. Zikic, D., Ioannou, Y., Brown, M., Criminisi, A. (2014). Segmentation of brain tumor tissues with convolutional neural networks. *MICCAI Workshop on Multimodal Brain Tumor Segmentation Challenge (BRATS)*, pp. 36–39. Boston, Massachusetts.
65. Işın, A., Direkoğlu, C., Şah, M. (2016). Review of MRI-based brain tumor image segmentation using deep learning methods. *Procedia Computer Science, 102,* 317–324. https://doi.org/10.1016/j.procs.2016.09.407
66. Wang, L., Lin, Z. Q., Wong, A. (2020). Covid-net: A tailored deep convolutional neural network design for detection of COVID-19 cases from chest X-ray images. *Scientific Reports, 10(1),* 19549. https://doi.org/10.1038/s41598-020-76550-z
67. Ulhaq, A., Born, J., Khan, A., Gomes, D. P. S., Chakraborty, S. et al. (2020). COVID-19 control by computer vision approaches: A survey. *IEEE Access, 8,* 179437–179456. https://doi.org/10.1109/ACCESS.2020.3027685

68. Chen, J., Wu, L., Zhang, J., Zhang, L., Gong, D. et al. (2020). Deep learning-based model for detecting 2019 novel coronavirus pneumonia on high-resolution computed tomography. *Scientific Reports, 10(1),* 19196. https://doi.org/10.1038/s41598-020-76282-0
69. Li, L., Qin, L., Xu, Z., Yin, Y., Wang, X. et al. (2020). Artificial intelligence distinguishes COVID-19 from community acquired pneumonia on chest CT. *Radiology,* 200905.
70. Shorten, C., Khoshgoftaar, T. M., Furht, B. (2021). Deep Learning applications for COVID-19. *Journal of Big Data, 8(1),* 18. https://doi.org/10.1186/s40537-020-00392-9
71. Sultana, M., Paul, P. P., Gavrilova, M. L. (2018). Social behavioral information fusion in multimodal biometrics. *IEEE Transactions on Systems, Man, and Cybernetics: Systems, 48(12),* 2176–2187. https://doi.org/10.1109/TSMC.2017.2690321
72. Li, S., Zhang, B., Fei, L., Zhao, S. (2021). Joint discriminative feature learning for multimodal finger recognition. *Pattern Recognition, 111,* 107704. https://doi.org/10.1016/j.patcog.2020.107704
73. Tiong, L. C. O., Kim, S. T., Ro, Y. M. (2020). Multimodal facial biometrics recognition: Dual-stream convolutional neural networks with multi-feature fusion layers. *Image and Vision Computing, 102,* 103977. https://doi.org/10.1016/j.imavis.2020.103977
74. Li, C., Min, X., Sun, S., Lin, W., Tang, Z. (2017). DeepGait: A learning deep convolutional representation for view-invariant gait recognition using joint Bayesian. *Applied Sciences, 7(3),* 210. https://doi.org/10.3390/app7030210
75. Wang, X., Yan, W. Q. (2020). Human gait recognition based on frame-by-frame gait energy images and convolutional long short-term memory. *International Journal of Neural Systems, 30(1),* 1950027. https://doi.org/10.1142/S0129065719500278
76. Babaee, M., Li, L., Rigoll, G. (2019). Person identification from partial gait cycle using fully convolutional neural networks. *Neurocomputing, 338,* 116–125. https://doi.org/10.1016/j.neucom.2019.01.091
77. Wang, S., Sheng, H., Yang, D., Zhang, Y., Wu, Y. et al. (2022). Extendable multiple nodes recurrent tracking framework with RTU++. *IEEE Transactions on Image Processing, 31,* 5257–5271. https://doi.org/10.1109/TIP.2022.3192706
78. Potluri, S., Ravuri, S., Diedrich, C., Schega, L. (2019). Deep Learning based gait abnormality detection using wearable sensor system. *Proceedings of the 2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 3613–3619. Berlin, Germany. https://doi.org/10.1109/EMBC.2019.8856454
79. Sokolova, A., Konushin, A. (2017). Gait recognition based on convolutional neural networks. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, XLII–2/W4,* 207–212. https://doi.org/10.5194/isprs-archives-XLII-2-W4-207-2017
80. Yan, L., Shi, Y., Wei, M., Wu, Y. (2023). Multi-feature fusing local directional ternary pattern for facial expressions signal recognition based on video communication system. *Alexandria Engineering Journal, 63,* 307–320. https://doi.org/10.1016/j.aej.2022.08.003
81. Redmon, J., Divvala, S., Girshick, R., Farhadi, A. (2016). You only look once: Unified, real-time object detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 779–788. Las Vegas, NV, USA. https://doi.org/10.1109/CVPR.2016.91
82. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S. et al. (2016). SSD: Single shot multibox detector. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (Eds.), *Computer vision, Lecture notes in computer science,* pp. 21–37, Cham, Switzerland: Springer.
83. Girshick, R., Donahue, J., Darrell, T., Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 580–587. Columbus, OH, USA. https://doi.org/10.1109/CVPR.2014.81
84. Ren, S., He, K., Girshick, R., Sun, J. (2015). Faster R-CNN: Towards real-time object detection with region proposal networks. *Proceedings of the 28th International Conference on Neural Information Processing Systems*, pp. 91–99. Sanur, Bali, Indonesia. https://doi.org/10.1109/TPAMI.2016.2577031

85. Wu, Y., Sheng, H., Zhang, Y., Wang, S., Xiong, Z. et al. (2023). Hybrid motion model for multiple object tracking in mobile devices. *IEEE Internet of Things Journal, 10(6),* 4735–4748. https://doi.org/10.1109/JIOT.2022.3219627
86. Cong, R., Sheng, H., Yang, D., Cui, Z., Chen, R. (2023). Exploiting spatial and angular correlations with deep efficient transformers for light field image super-resolution. *IEEE Transactions on Multimedia,* 1–14. https://doi.org/10.1109/TMM.2023.3282465
87. Deng, J., Guo, J., Ververas, E., Kotsia, I., Zafeiriou, S. (2020). Retinaface: Single-shot multi-level face localisation in the wild. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5203–5212. Seattle, WA, USA. https://doi.org/10.1109/CVPR42600.2020.00525
88. Jiang, M., Fan, X., Yan, H. (2020). Retinamask: A face mask detector. arXiv preprint arXiv:2005.
89. Vinh, T. Q., Anh, N. T. N. (2020). Real-time face mask detector using YOLOv3 algorithm and Haar cascade classifier. *International Conference on Advanced Computing and Applications (ACOMP)*, pp. 146–149. Quy Nhon, Vietnam. https://doi.org/10.1109/ACOMP50827.2020.00029
90. Nagrath, P., Jain, R., Madan, A., Arora, R., Kataria, P. et al. (2021). SSDMNV2: A real time DNN-based face mask detection system using single shot multibox detector and MobileNetV2. *Sustainable Cities and Society, 66,* 102692. https://doi.org/10.1016/j.scs.2020.102692
91. Gupta, C., Gill, N. S. (2020). Coronamask: A face mask detector for real-time data. *International Journal of Advanced Trends in Computer Science and Engineering, 9(4),* 5624–5630. https://doi.org/10.30534/ijatcse/2020/212942020
92. Ge, Z., Liu, S., Wang, F., Li, Z., Sun, J. (2021). YOLOX: Exceeding YOLO series in 2021. arXiv preprint arXiv:2107.08430.
93. Lin, Z., Wang, H., Li, S. (2022). Pavement anomaly detection based on transformer and self-supervised learning. *Automation in Construction, 143,* 104544. https://doi.org/10.1016/j.autcon.2022.104544
94. Wang, Y., Xu, N., Liu, A. A., Li, W., Zhang, Y. (2022). High-order interaction learning for image captioning. *IEEE Transactions on Circuits and Systems for Video Technology, 32(7),* 4417–4430. https://doi.org/10.1109/TCSVT.2021.3121062
95. Zhang, Y., Chen, J., Ma, X., Wang, G., Bhatti, U. A. et al. (2023). Interactive medical image annotation using improved attention U-net with compound geodesic distance. *Expert Systems with Applications, 237,* 121282. https://doi.org/10.1016/j.eswa.2023.121282
96. Zhou, X., Sun, K., Wang, J., Zhao, J., Feng, C. et al. (2023). Computer vision enabled building digital twin using building information model. *IEEE Transactions on Industrial Informatics, 19(3),* 2684–2692. https://doi.org/10.1109/TII.2022.3190366
97. Milanova, M., Nikolov, L., Fotev, S. (1996). Three dimensional computer vision for computer aided design and manufacturing applications. *International Workshop on Structural and Syntactic Pattern Recognition*, pp. 279–288.
98. Ye, X., Liu, H., Chen, L., Chen, Z., Pan, X. et al. (2008). Reverse innovative design—An integrated product design methodology. *Computer-Aided Design, 40(7),* 812–827. https://doi.org/10.1016/j.cad.2007.07.006
99. Ulrich, M., Wiedemann, C., Steger, C. (2012). Combining scale-space and similarity-based aspect graphs for fast 3D object recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 34(10),* 1902–1914. https://doi.org/10.1109/TPAMI.2011.266
100. Jiang, J., Chen, Z., He, K. (2013). A feature-based method of rapidly detecting global exact symmetries in CAD models. *Computer-Aided Design, 45(8–9),* 1081–1094. https://doi.org/10.1016/j.cad.2013.04.005
101. Alexopoulos, K., Nikolakis, N., Chryssolouris, G. (2020). Digital twin-driven supervised machine learning for the development of artificial intelligence applications in manufacturing. *International Journal of Computer Integrated Manufacturing, 33(5),* 429–439. https://doi.org/10.1080/0951192X.2020.1747642

102. Raj, M. D., Kumar, V. S. (2017). Vision based feature diagnosis for automobile instrument cluster using machine learning. *Fourth International Conference on Signal Processing, Communication and Networking (ICSCN)*, pp. 1–5. Chennai, India. https://doi.org/10.1109/ICSCN.2017.8085671
103. da Silva Lopes, J. L., Moreira Marques, C. A., de Moura Vasconcelos, G., Barreto Vieira, R., Ventura de Melo Ferreira, F. F. et al. (2016). Automated tests for automotive instrument panel cluster based on machine vision. *25th SAE BRASIL International Congress and Display*, USA, SAE Technical Paper Series. https://doi.org/10.4271/2016-36-0235
104. Huang, Y., McMurran, R., Dhadyalla, G., Jones, R. P., Mouzakitis, A. (2009). Model-based testing of a vehicle instrument cluster for design validation using machine vision. *Measurement Science and Technology, 20(6),* 65502. https://doi.org/10.1088/0957-0233/20/6/065502
105. Lindner, L., Sergiyenko, O., Rodríguez-Quiñonez, J. C., Rivas-Lopez, M., Hernandez-Balbuena, D. et al. (2016). Mobile robot vision system using continuous laser scanning for industrial application. *Industrial Robot, 43(4),* 360–369. https://doi.org/10.1108/IR-01-2016-0048
106. Trdič, F., Širok, B., Bullen, P. R., Philpott, D. R. (1999). Monitoring mineral wool production using real-time machine vision. *Real-Time Imaging, 5(2),* 125–140. https://doi.org/10.1016/S1077-2014(99)80010-2
107. Carfagni, M., Furferi, R., Governi, L. (2005). A real-time machine-vision system for monitoring the textile raising process. *Computers in Industry, 56(8–9),* 831–842. https://doi.org/10.1016/j.compind.2005.05.010
108. Liu, J., Gui, W., Tang, Z., Hu, H., Zhu, J. (2013). Machine vision based production condition classification and recognition for mineral flotation process monitoring. *International Journal of Computational Intelligence Systems, 6(5),* 969–986. https://doi.org/10.1080/18756891.2013.809938
109. Zhao, Q. J., Cao, P., Tu, D. W. (2014). Toward intelligent manufacturing: Label characters marking and recognition method for steel products with machine vision. *Advances in Manufacturing, 2(1),* 3–12. https://doi.org/10.1007/s40436-014-0057-2
110. Tsai, D. M., Lin, M. C. (2013). Machine-vision-based identification for wafer tracking in solar cell manufacturing. *Robotics and Computer-Integrated Manufacturing, 29(5),* 312–321. https://doi.org/10.1016/j.rcim.2013.01.009
111. Mehrabi, A., Mehrshad, N., Massinaei, M. (2014). Machine vision based monitoring of an industrial flotation cell in an iron flotation plant. *International Journal of Mineral Processing, 133,* 60–66. https://doi.org/10.1016/j.minpro.2014.09.018
112. Alexopoulos, K., Catti, P., Kanellopoulos, G., Nikolakis, N., Blatsiotis, A. et al. (2023). Deep learning for estimating the fill-level of industrial waste containers of metal scrap: A case study of a copper tube plant. *Applied Sciences, 13(4),* 2575. https://doi.org/10.3390/app13042575
113. Cheng, J. C. P., Wang, M. (2018). Automated detection of sewer pipe defects in closed-circuit television images using deep learning techniques. *Automation in Construction, 95,* 155–171. https://doi.org/10.1016/j.autcon.2018.08.006
114. Bhandarkar, S. M., Faust, T. D., Tang, M. (2002). Design and prototype development of a computer vision-based lumber production planning system. *Image and Vision Computing, 20(3),* 167–189. https://doi.org/10.1016/S0262-8856(01)00087-7
115. Tao, X., Wang, Z., Zhang, Z., Zhang, D., Xu, D. et al. (2018). Wire defect recognition of spring-wire socket using multitask convolutional neural networks. *IEEE Transactions on Components, Packaging and Manufacturing Technology, 8(4),* 689–698. https://doi.org/10.1109/TCPMT.2018.2794540
116. Li, X., Wang, L., Cai, N. (2004). Machine-vision-based surface finish inspection for cutting tool replacement in production. *International Journal of Production Research, 42(11),* 2279–2287. https://doi.org/10.1080/00207540420001977O2
117. Zhu, X., Chen, R., Zhang, Y. (2014). Automatic defect detection in spring clamp production via machine vision. *Abstract and Applied Analysis, 2014,* 1–9. https://doi.org/10.1155/2014/164726

118. Chiou, R., Mookiah, P., Kwon, Y. (2009). Manufacturing e-quality through integrated web-enabled computer vision and robotics. *International Journal of Advanced Manufacturing Technology, 43(7–8),* 720–730. https://doi.org/10.1007/s00170-008-1747-3
119. Chauhan, V., Surgenor, B. (2017). Fault detection and classification in automated assembly machines using machine vision. *International Journal of Advanced Manufacturing Technology, 90(9–12),* 2491–2512. https://doi.org/10.1007/s00170-016-9581-5
120. Furferi, R., Governi, L., Volpe, Y., Carfagni, M. (2013). Design and assessment of a machine vision system for automatic vehicle wheel alignment. *International Journal of Advanced Robotic Systems, 10(5),* 242. https://doi.org/10.5772/55928
121. Zhou, Q., Chen, R., Huang, B., Liu, C., Yu, J. et al. (2019). An automatic surface defect inspection system for automobiles using machine vision methods. *Sensors, 19(3),* 644. https://doi.org/10.3390/s19030644
122. Sitthi-Amorn, P., Ramos, J. E., Wangy, Y., Kwan, J., Lan, J. et al. (2015). MultiFab. *ACM Transactions on Graphics, 34(4),* 1–11. https://doi.org/10.1145/2766962
123. Ceruti, A., Liverani, A., Bombardi, T. (2017). Augmented vision and interactive monitoring in 3D printing process. *International Journal on Interactive Design and Manufacturing, 11(2),* 385–395. https://doi.org/10.1007/s12008-016-0347-y
124. Shen, H., Sun, W., Fu, J. (2019). Multi-view online vision detection based on robot fused deposit modeling 3D printing technology. *Rapid Prototyping Journal, 25(2),* 343–355. https://doi.org/10.1108/RPJ-03-2018-0052
125. Fang, Q., Li, H., Luo, X., Ding, L., Luo, H. et al. (2018). Detecting non-hardhat-use by a deep learning method from far-field surveillance videos. *Automation in Construction, 85,* 1–9. https://doi.org/10.1016/j.autcon.2017.09.018
126. Fang, W., Ding, L., Luo, H., Love, P. E. D. (2018). Falls from heights: A computer vision-based approach for safety harness detection. *Automation in Construction, 91,* 53–61. https://doi.org/10.1016/j.autcon.2018.02.018
127. Karabagli, B., Simon, T., Orteu, J. J. (2016). A new chain-processing-based computer vision system for automatic checking of machining set-up application for machine tools safety. *International Journal of Advanced Manufacturing Technology, 82(9–12),* 1547–1568. https://doi.org/10.1007/s00170-015-7438-y
128. Yilmaz, A., Javed, O., Shah, M. (2006). Object tracking. *ACM Computing Surveys, 38(4),* 13. https://doi.org/10.1145/1177352.1177355
129. Son, H., Seong, H., Choi, H., Kim, C. (2019). Real-time vision-based warning system for prevention of collisions between workers and heavy equipment. *Journal of Computing in Civil Engineering, 33(5),* 4019029. https://doi.org/10.1061/(ASCE)CP.1943-5487.0000845
130. Pech, M., Vrchota, J., Bednář, J. (2021). Predictive maintenance and intelligent sensors in smart factory: Review. *Sensors, 21(4),* 1470. https://doi.org/10.3390/s21041470
131. Zonta, T., da Costa, C. A., da Rosa Righi, R., de Lima, M. J., da Trindade, E. S. et al. (2020). Predictive maintenance in the Industry 4.0: A systematic literature review. *Computers and Industrial Engineering, 150,* 106889. https://doi.org/10.1016/j.cie.2020.106889
132. Yang, S., Li, Q., Li, W., Li, X., Liu, A. A. (2022). Dual-level representation enhancement on characteristic and context for image-text retrieval. *IEEE Transactions on Circuits and Systems for Video Technology, 32(11),* 8037–8050. https://doi.org/10.1109/TCSVT.2022.3182426
133. Mateos García, N. M. (2019). Multi-agent system for anomaly detection in industry 4.0 using machine learning techniques. *ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal, 8(4),* 33–40. https://doi.org/10.14201/ADCAIJ2019843340
134. Çınar, Z. M., Abdussalam Nuhu, A., Zeeshan, Q., Korhan, O., Asmael, M. et al. (2020). Machine learning in predictive maintenance towards sustainable smart manufacturing in Industry 4.0. *Sustainability, 12(19),* 8211. https://doi.org/10.3390/su12198211

135. Nayak, R., Pati, U. C., Das, S. K. (2021). A comprehensive review on deep learning-based methods for video anomaly detection. *Image and Vision Computing, 106,* 104078. https://doi.org/10.1016/j.imavis.2020.104078
136. Wang, Y., Su, Y., Li, W., Xiao, J., Li, X. et al. (2023). Dual-path rare content enhancement network for image and text matching. *IEEE Transactions on Circuits and Systems for Video Technology, 33(10),* 6144–6158. https://doi.org/10.1109/TCSVT.2023.3254530
137. Nie, W., Bao, Y., Zhao, Y., Liu, A. (2023). Long dialogue emotion detection based on commonsense knowledge graph guidance. *IEEE Transactions on Multimedia,* 1–15. https://doi.org/10.1109/TMM.2023.3267295
138. Huang, Y. C., Liao, I. N., Chen, C. H., İk., T. U., Peng, W. C. (2019). TrackNet: A deep learning network for tracking high-speed and tiny objects in sports applications. *16th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pp. 1–8. Taipei, Taiwan. https://doi.org/10.1109/AVSS.2019.8909871
139. Kamble, P. R., Keskar, A. G., Bhurchandi, K. M. (2019). A deep learning ball tracking system in soccer videos. *Opto-Electronics Review, 27(1),* 58–69. https://doi.org/10.1016/j.opelre.2019.02.003
140. Cui, Z., Sheng, H., Yang, D., Wang, S., Chen, R. et al. (2023). Light field depth estimation for non-lambertian objects via adaptive cross operator. *IEEE Transactions on Circuits and Systems for Video Technology*. https://doi.org/10.1109/TCSVT.2023.3292884
141. Song, H., Han, X. Y., Montenegro-Marin, C. E., krishnamoorthy, S. (2021). Secure prediction and assessment of sports injuries using deep learning based convolutional neural network. *Journal of Ambient Intelligence and Humanized Computing, 12(3),* 3399–3410. https://doi.org/10.1007/s12652-020-02560-4
142. Midhu, K., Anantha Padmanabhan, N. K. A. (2018). Highlight generation of cricket match using deep learning. In: *Computational vision and bio inspired computing*, pp. 925–936. Springer, Cham. https://doi.org/10.1007/978-3-319-71767-8_79
143. Khan, A. A., Shao, J. (2022). SPNet: A deep network for broadcast sports video highlight generation. *Computers and Electrical Engineering, 99,* 107779. https://doi.org/10.1016/j.compeleceng.2022.107779
144. Chen, J., Wang, Q., Peng, W., Xu, H., Li, X. et al. (2022). Disparity-based multiscale fusion network for transportation detection. *IEEE Transactions on Intelligent Transportation Systems, 23(10),* 18855–18863. https://doi.org/10.1109/TITS.2022.3161977
145. Rozantsev, A., Lepetit, V., Fua, P. (2015). On rendering synthetic images for training an object detector. *Computer Vision and Image Understanding, 137,* 24–37. https://doi.org/10.1016/j.cviu.2014.12.006
146. Cao, B., Li, M., Liu, X., Zhao, J., Cao, W. et al. (2021). Many-objective deployment optimization for a drone-assisted camera network. *IEEE Transactions on Network Science and Engineering, 8(4),* 2756–2764. https://doi.org/10.1109/TNSE.2021.3057915
147. Sapkota, K. R., Roelofsen, S., Rozantsev, A., Lepetit, V., Gillet, D. et al. (2016). Vision-based unmanned aerial vehicle detection and tracking for sense and avoid systems. *16th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pp. 1–8. Daejeon, Korea (South). https://doi.org/10.1109/IROS.2016.7759252
148. Zheng, Y., Zhang, Y., Qian, L., Zhang, X., Diao, S. et al. (2023). A lightweight ship target detection model based on improved YOLOv5s algorithm. *PLoS One, 18(4),* e0283932. https://doi.org/10.1371/journal.pone.0283932
149. Zheng, Y., Liu, P., Qian, L., Qin, S., Liu, X. et al. (2022). Recognition and depth estimation of ships based on binocular stereo vision. *Journal of Marine Science and Engineering, 10(8),* 1153. https://doi.org/10.3390/jmse10081153
150. Qian, L., Zheng, Y., Li, L., Ma, Y., Zhou, C. et al. (2022). A new method of inland water ship trajectory prediction based on long short-term memory network optimized by genetic algorithm. *Applied Sciences, 12(8),* 4073. https://doi.org/10.3390/app12084073

151. Sanjana, S., Sanjana, S., Shriya, V. R., Vaishnavi, G., Ashwini, K. (2021). A review on various methodologies used for vehicle classification, helmet detection and number plate recognition. *Evolutionary Intelligence, 14(2),* 979–987. https://doi.org/10.1007/s12065-020-00493-7
152. Shobha, B. S., Deepu, R. (2018). A review on video based vehicle detection, recognition and tracking. *3rd International Conference on Computational Systems and Information Technology for Sustainable Solutions (CSITSS)*, pp. 183–186. Bengaluru, India. https://doi.org/10.1109/CSITSS.2018.8768743
153. Long, W., Xiao, Z., Wang, D., Jiang, H., Chen, J. et al. (2022). Unified spatial-temporal neighbor attention network for dynamic traffic prediction. *IEEE Transactions on Vehicular Technology, 72(2),* 1515–1529. https://doi.org/10.1109/TVT.2022.3209242
154. Chen, J., Xu, M., Xu, W., Li, D., Peng, W. et al. (2023). A flow feedback traffic prediction based on visual quantified features. *IEEE Transactions on Intelligent Transportation Systems, 24(9),* 10067–10075. https://doi.org/10.1109/TITS.2023.3269794
155. Zhang, X., Wen, S., Yan, L., Feng, J., Xia, Y. (2022). A hybrid-convolution spatial-temporal recurrent network for traffic flow prediction. *Computer Journal, 593,* 522. https://doi.org/10.1093/comjnl/bxac171
156. Franklin, R. J. (2020). Traffic signal violation detection using artificial intelligence and deep learning. *5th International Conference on Communication and Electronics Systems (ICCES)*, pp. 839–844. Coimbatore, India. https://doi.org/10.1109/ICCES48766.2020.9137873
157. Chen, J., Wang, Q., Cheng, H. H., Peng, W., Xu, W. (2022). A review of vision-based traffic semantic understanding in ITSs. *IEEE Transactions on Intelligent Transportation Systems, 23(11),* 19954–19979. https://doi.org/10.1109/TITS.2022.3182410
158. Xu, J., Park, S. H., Zhang, X., Hu, J. (2022). The improvement of road driving safety guided by visual inattentional blindness. *IEEE Transactions on Intelligent Transportation Systems, 23(6),* 4972–4981. https://doi.org/10.1109/TITS.2020.3044927
159. Rahim, M. A., Hassan, H. M. (2021). A deep learning based traffic crash severity prediction framework. *Accident; Analysis and Prevention, 154,* 106090. https://doi.org/10.1016/j.aap.2021.106090
160. Naseer, A., Nour, M. K., Alkazemi, B. Y. (2020). Towards deep learning based traffic accident analysis. *10th Annual Computing and Communication Workshop and Conference (CCWC)*, pp. 817–820. Las Vegas, NV, USA. https://doi.org/10.1109/CCWC47524.2020.9031235
161. Xu, J., Guo, K., Sun, P. Z. H. (2022). Driving performance under violations of traffic rules: Novice vs. experienced drivers. *IEEE Transactions on Intelligent Vehicles, 7(4),* 908–917. https://doi.org/10.1109/TIV.2022.3200592
162. Ding, Y., Zhang, W., Zhou, X., Liao, Q., Luo, Q. et al. (2021). FraudTrip: Taxi fraudulent trip detection from corresponding trajectories. *IEEE Internet of Things Journal, 8(16),* 12505–12517. https://doi.org/10.1109/JIOT.2020.3019398
163. Khan, G., Farooq, M. A., Tariq, Z., Khan, M. U. G. (2019). Deep-learning based vehicle count and free parking slot detection system. *22nd International Multitopic Conference (INMIC)*, pp. 1105–1112. Islamabad, Pakistan. https://doi.org/10.1109/INMIC48123.2019.9022687
164. Valipour, S., Siam, M., Stroulia, E., Jagersand, M. (2016). Parking-stall vacancy indicator system, based on deep convolutional neural networks. *3rd World Forum on Internet of Things (WF-IoT)*, pp. 655–660. Reston, VA, USA. https://doi.org/10.1109/WF-IoT.2016.7845408
165. Xu, J., Zhang, X., Park, S. H., Guo, K. (2022). The alleviation of perceptual blindness during driving in urban areas guided by saccades recommendation. *IEEE Transactions on Intelligent Transportation Systems, 23(9),* 16386–16396. https://doi.org/10.1109/TITS.2022.3149994
166. Xu, J., Pan, S., Sun, P. Z. H., Hyeong Park, S. H., Guo, K. (2023). Human-factors-in-driving-loop: Driver identification and verification via a deep learning approach using psychological behavioral data. *IEEE Transactions on Intelligent Transportation Systems, 24(3),* 3383–3394. https://doi.org/10.1109/TITS.2022.3225782

167. Streiffer, C., Raghavendra, R., Benson, T., Srivatsa, M. (2017). Darnet: A deep learning solution for distracted driving detection. *Proceedings of the 18th ACM/IFIP/Usenix Middleware Conference: Industrial Track*, pp. 22–28. New York, USA. https://doi.org/10.1145/3154448.3154452
168. Guo, J., Liu, Y., Zhang, L., Wang, Y. (2018). Driving behaviour style study with a hybrid deep learning framework based on GPS data. *Sustainability, 10(7),* 2351. https://doi.org/10.3390/su10072351
169. Han, Y., Wang, B., Guan, T., Tian, D., Yang, G. et al. (2023). Research on road environmental sense method of intelligent vehicle based on tracking check. *IEEE Transactions on Intelligent Transportation Systems, 24(1),* 1261–1275. https://doi.org/10.1109/TITS.2022.3183893
170. Wang, S., Sheng, H., Zhang, Y., Yang, D., Shen, J. et al. (2023). Blockchain-empowered distributed multi-camera multi-target tracking in edge computing. *IEEE Transactions on Industrial Informatics,* 1–10. https://doi.org/10.1109/TII.2023.3261890
171. Sivaraman, S., Trivedi, M. M. (2013). Looking at vehicles on the road: A survey of vision-based vehicle detection, tracking, and behavior analysis. *IEEE Transactions on Intelligent Transportation Systems, 14(4),* 1773–1795. https://doi.org/10.1109/TITS.2013.2266661
172. Jayaraman, D., Grauman, K. (2016). Slow and steady feature analysis: Higher order temporal coherence in video. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3852–3861. Las Vegas, NV, USA. https://doi.org/10.1109/CVPR.2016.418
173. Sadigh, D., Sastry, S., Seshia, S. A., Dragan, A. D. (2016). Planning for autonomous cars that leverage effects on human actions. *Robotics: Science and Systems, 2,* 1–9. https://doi.org/10.15607/RSS.2016.XII.029
174. Chen, F., Wu, F., Xu, J., Gao, G., Ge, Q. et al. (2021). Adaptive deformable convolutional network. *Neurocomputing, 453,* 853–864. https://doi.org/10.1016/j.neucom.2020.06.128
175. Sheng, H., Wang, S., Yang, D., Cong, R., Cui, Z. et al. (2023). Cross-view recurrence-based self-supervised super-resolution of light field. *IEEE Transactions on Circuits and Systems for Video Technology*. https://doi.org/10.1109/TCSVT.2023.3278462
176. Jiang, H., Xiao, Z., Li, Z., Xu, J., Zeng, F. et al. (2022). An energy-efficient framework for internet of things underlaying heterogeneous small cell networks. *IEEE Transactions on Mobile Computing, 21(1),* 31–43. https://doi.org/10.1109/TMC.2020.3005908
177. Janai, J., Güney, F., Behl, A., Geiger, A. (2020). Computer vision for autonomous vehicles: Problems, datasets and state of the art. *Foundations and Trends in Computer Graphics and Vision, 12(1–3),* 1–308. https://doi.org/10.1561/0600000079
178. Güzel, M. S. (2013). Autonomous vehicle navigation using vision and mapless strategies: A survey. *Advances in Mechanical Engineering, 5,* 234747. https://doi.org/10.1155/2013/234747
179. Deng, J., Dong, W., Socher, R., Li, L. J., Li, K. et al. (2009). Imagenet: A large-scale hierarchical image database. *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255. Miami, FL, USA. https://doi.org/10.1109/CVPR.2009.5206848
180. Rothe, R., Timofte, R., van Gool, L. V. (2015). Dex: Deep expectation of apparent age from a single image. *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pp. 252–257. Santiago, Chile. https://doi.org/10.1109/ICCVW.2015.41
181. Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P. et al. (2014). Microsoft coco: Common objects in context. *Computer Vision-ECCV 2014: 13th European Conference*, pp. 740–755. Zurich, Switzerland. https://doi.org/10.1007/978-3-319-10602-1_48
182. Andriluka, M., Pishchulin, L., Gehler, P., Schiele, B. (2014). 2d human pose estimation: New benchmark and state of the art analysis. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3686–3693. Columbus, OH, USA. https://doi.org/10.1109/CVPR.2014.471
183. Kuznetsova, A., Rom, H., Alldrin, N., Uijlings, J., Krasin, I. et al. (2020). The open images dataset v4. *International Journal of Computer Vision, 128(7),* 1956–1981. https://doi.org/10.1007/s11263-020-01316-z

184. Krizhevsky, A., Nair, V., Hinton, G. (2020). The CIFAR-10 dataset (2014). https://www.cs.toronto.edu/~kriz/cifar.html (accessed on 01/08/2023)
185. Krizhevsky, A., Nair, V., Hinton, G. (2009). CIFAR-10 and CIFAR-100 datasets. https://www.cs.toronto.edu/~kriz/cifar-100-python.tar.gz (accessed on 01/08/2023)
186. Yu, F., Seff, A., Zhang, Y., Song, S., Funkhouser, T. et al. (2015). LSUN: Construction of a large-scale image dataset using deep learning with humans in the loop. arXiv preprint arXiv:1506.03365.
187. Russell, B. C., Torralba, A., Murphy, K. P., Freeman, W. T. (2008). LabelMe: A database and web-based tool for image annotation. *International Journal of Computer Vision, 77(1–3),* 157–173. https://doi.org/10.1007/s11263-007-0090-8
188. Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M. et al. (2016). The cityscapes dataset for semantic urban scene understanding. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3213–3223. Las Vegas, NV, USA. https://doi.org/10.1109/CVPR.2016.350
189. Carreira, J., Noland, E., Hillier, C., Zisserman, A. (2019). A short note on the kinetics-700 human action dataset. arXiv preprint arXiv:1907.06987.
190. Goyal, Y., Khot, T., Summers-Stay, D., Batra, D., Parikh, D. (2017). Making the v in vqa matter: Elevating the role of image understanding in visual question answering. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6904–6913. Honolulu, HI, USA.
191. Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B. et al. (2011). Reading digits in natural images with unsupervised feature learning. *NIPS Workshop on Deep Learning and Unsupervised Feature Learning*. Granada, Spain.
192. Xiao, H., Rasul, K., Vollgraf, R. (2017). Fashion-MNIST: A novel image dataset for benchmarking machine learning algorithms. arXiv preprint arXiv:1708.07747.
193. Real, E., Shlens, J., Mazzocchi, S., Pan, X., Vanhoucke, V. (2017). YouTube-boundingboxes: A large high-precision human-annotated data set for object detection in video. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5296–5305. Honolulu, HI, USA. https://doi.org/10.1109/CVPR.2017.789
194. Wang, J., Wang, X., Shang guan, Y., Gupta, A. (2021). Wanderlust: Online continual object detection in the real world. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 10829–10838. Montreal, QC, Canada. https://doi.org/10.1109/ICCV48922.2021.01065
195. de Tournemire, P., Nitti, D., Perot, E., Migliore, D., Sironi, A. (2020). A large scale event-based detection dataset for automotive. arXiv preprint arXiv:2001.08499. https://doi.org/10.48550/arXiv.2001.08499
196. Bengar, J. Z., Gonzalez-Garcia, A., Villalonga, G., Raducanu, B., Aghdam, H. H. et al. (2019). Temporal coherence for active learning in videos. *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pp. 1105–1112. Seoul, Korea (South). https://doi.org/10.1109/ICCVW.2019.00120
197. Duran-Vega, M. A., Gonzalez-Mendoza, M., Chang-Fernandez, L., Suarez-Ramirez, C. D. (2021). TYolov5: A temporal Yolov5 detector based on quasi-recurrent neural networks for real-time handgun detection in video. arXiv preprint arXiv: 2111.08867.
198. Bosquet, B., Mucientes, M., Brea, V. M. (2020). STDnet: Exploiting high resolution feature maps for small object detection. *Engineering Applications of Artificial Intelligence, 91,* 103615. https://doi.org/10.1016/j.engappai.2020.103615
199. Anilkumar, P., Venugopal, P. (2022). Research contribution and comprehensive review towards the semantic segmentation of aerial images using deep learning techniques. *Security and Communication Networks, 2022,* 1–31. https://doi.org/10.1155/2022/6010912
200. Alzubaidi, L., Zhang, J., Humaidi, A. J., Al-Dujaili, A., Duan, Y. et al. (2021). Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions. *Journal of Big Data, 8(1),* 53. https://doi.org/10.1186/s40537-021-00444-8

201. Guan, Z., Jing, J., Deng, X., Xu, M., Jiang, L. et al. (2023). DeepMIH: Deep invertible network for multiple image hiding. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 45(1),* 372–390. https://doi.org/10.1109/TPAMI.2022.3141725
202. Yosinski, J., Clune, J., Bengio, Y., Lipson, H. (2014). How transferable are features in deep neural networks? *Advances in Neural Information Processing Systems, 27,* 3320–3328.
203. Tan, C., Sun, F., Kong, T., Zhang, W., Yang, C. et al. (2018). A survey on deep transfer learning. *International Conference on Artificial Neural Networks*, pp. 270–277. Rhodes, Greece. https://doi.org/10.1007/978-3-030-01424-7_27
204. Liu, X., He, J., Liu, M., Yin, Z., Yin, L. et al. (2023). A scenario-generic neural machine translation data augmentation method. *Electronics, 12(10),* 2320. https://doi.org/10.3390/electronics12102320
205. Nair, T., Precup, D., Arnold, D. L., Arbel, T. (2020). Exploring uncertainty measures in deep networks for multiple sclerosis lesion detection and segmentation. *Medical Image Analysis, 59,* 101557. https://doi.org/10.1016/j.media.2019.101557
206. French, R. M. (1991). Using semi-distributed representations to overcome catastrophic forgetting in connectionist networks. *Proceedings of the 13th Annual Cognitive Science Society Conference*, pp. 173–178. Chicago USA. https://doi.org/10.1007/978-3-030-01424-7_27
207. Everitt, B. S., Skrondal, A. (2010). *The Cambridge dictionary of statistics*, 4th edition. Cambridge, UK: Cambridge University Press.
208. D'Amour, A., Heller, K., Moldovan, D., Adlam, B., Alipanahi, B. et al. (2022). Underspecification presents challenges for credibility in modern machine learning. *Journal of Machine Learning Research, 23(1),* 10237–10297.
209. Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D. et al. (2013). Intriguing properties of neural networks. arXiv preprint arXiv:1312.6199.
210. Li, J., Deng, Y., Sun, W., Li, W., Li, R. et al. (2022). Resource orchestration of cloud-edge-based smart grid fault detection. *ACM Transactions on Sensor Networks, 18(3),* 1–26. https://doi.org/10.1145/3586058
211. Goodfellow, I. J., Shlens, J., Szegedy, C. (2014). Explaining and harnessing adversarial examples. arXiv preprint arXiv:1412.6572.
212. Liu, H., Yuan, H., Hou, J., Hamzaoui, R., Gao, W. (2022). PUFA-GAN: A frequency-aware generative adversarial network for 3D point cloud upsampling. *IEEE Transactions on Image Processing, 31,* 7389–7402. https://doi.org/10.1109/TIP.2022.3222918
213. Dong, J., Wang, Y., Lai, J. H., Xie, X. (2022). Improving adversarially robust few-shot image classification with generalizable representations. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9015–9024. New Orleans, LA, USA. https://doi.org/10.1109/CVPR52688.2022.00882
214. Madry, A., Makelov, A., Schmidt, L., Tsipras, D., Vladu, A. (2017). Towards deep learning models resistant to adversarial attacks. arXiv preprint arXiv:1706.06083.
215. Dang, K., Lai, J., Dong, J., Xie, X. (2022). Adversarial training inspired self-attention flow for universal image style transfer. *Pattern Recognition: 6th Asian Conference, ACPR 2021*, pp. 476–489. Jeju Island, South Korea. https://doi.org/10.1007/978-3-031-02444-3_36
216. Dong, J., Moosavi-Dezfooli, S. M., Lai, J., Xie, X. (2023). The enemy of my enemy is my friend: Exploring inverse adversaries for improving adversarial training. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 24678–24687. Vancouver, BC, Canada. https://doi.org/10.1109/CVPR52729.2023.02364