

Review Article

Deep CNN and Deep GAN in Computational Visual Perception-Driven Image Analysis

R. Nandhini Abirami ¹, **P. M. Durai Raj Vincent** ¹, **Kathiravan Srinivasan** ¹,
Usman Tariq ² and **Chuan-Yu Chang** ³

¹*School of Information Technology and Engineering, Vellore Institute of Technology (VIT), Vellore 632014, India*

²*College of Computer Engineering and Sciences, Prince Sattam Bin Abdulaziz University, Al-Kharj 11942, Saudi Arabia*

³*Department of Computer Science and Information Engineering, National Yunlin University of Science and Technology, Yunlin 64002, Taiwan*

Correspondence should be addressed to P. M. Durai Raj Vincent; pmvincent@vit.ac.in and Chuan-Yu Chang; chuanyu@yuntech.edu.tw

Received 7 January 2021; Revised 11 February 2021; Accepted 18 March 2021; Published 20 April 2021

Academic Editor: Dr Shahzad Sarfraz

Copyright © 2021 R. Nandhini Abirami et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Computational visual perception, also known as computer vision, is a field of artificial intelligence that enables computers to process digital images and videos in a similar way as biological vision does. It involves methods to be developed to replicate the capabilities of biological vision. The computer vision's goal is to surpass the capabilities of biological vision in extracting useful information from visual data. The massive data generated today is one of the driving factors for the tremendous growth of computer vision. This survey incorporates an overview of existing applications of deep learning in computational visual perception. The survey explores various deep learning techniques adapted to solve computer vision problems using deep convolutional neural networks and deep generative adversarial networks. The pitfalls of deep learning and their solutions are briefly discussed. The solutions discussed were dropout and augmentation. The results show that there is a significant improvement in the accuracy using dropout and data augmentation. Deep convolutional neural networks' applications, namely, image classification, localization and detection, document analysis, and speech recognition, are discussed in detail. In-depth analysis of deep generative adversarial network applications, namely, image-to-image translation, image denoising, face aging, and facial attribute editing, is done. The deep generative adversarial network is unsupervised learning, but adding a certain number of labels in practical applications can improve its generating ability. However, it is challenging to acquire many data labels, but a small number of data labels can be acquired. Therefore, combining semisupervised learning and generative adversarial networks is one of the future directions. This article surveys the recent developments in this direction and provides a critical review of the related significant aspects, investigates the current opportunities and future challenges in all the emerging domains, and discusses the current opportunities in many emerging fields such as handwriting recognition, semantic mapping, webcam-based eye trackers, lumen center detection, query-by-string word, intermittently closed and open lakes and lagoons, and landslides.

1. Introduction

Computer vision (CV), the core component of machine intelligence, is an interdisciplinary field enabling computers to achieve a visual understanding of digital images. For a machine to view as animals or people do, it relies on computer vision. Table 1 contains the list of abbreviations and their expansion used in the manuscript. CV is a

booming field and is applied to many of our everyday activities; some of them are face detection, object detection, biometrics, a medical diagnosis from faces, self-checkout kiosk, autonomous vehicles, image recognition, image enhancement, image deblurring, motion tracking, video surveillance, control of robots, analysis of mammography, and X-rays [22–26]. The fundamental goal of all these applications is to create a human observer replica in interpreting the

TABLE 1: List of abbreviations used in this manuscript along with their expansion.

Abbreviation	Full form	Authors
CV	Computer vision	Roberts, 1963 [1]
D-CNN	Deep convolutional neural network	Lecun et al., 1998 [2]
RNN	Recurrent neural network	Graves, 2006 [3]
DNN	Deep neural network	Ivakhnenko, 1971 [4]
AI	Artificial intelligence	John McCarthy, 1956
MNIST	Mixed National Institute of Standards and Technology	Lecun et al. (http://yann.lecun.com/exdb/mnist/)
ReLU	Rectified linear unit	Hahnloser et al., 2000 [5]
COCO	Common objects in context	Lin et al., 2014 [6]
D-GAN	Deep generative adversarial network	Goodfellow et al., 2014 [7]
DCGAN	Deep convolutional GAN	Radford et al., 2015 [8]
SRGAN	Super-resolution generative adversarial networks	Ledig et al., 2017 [9]
APGAN	Laplacian pyramid GAN	Denton et al., 2015 [10]
SAPGAN	Self-attention generative adversarial networks	Zhang, et al., 2019 [11]
GRAN	Generating images with recurrent adversarial networks	Im, et al., 2016 [12]
GPF-CNN	Gated peripheral-foveal convolutional neural network	Hahnloser et al., 2000 [5]
PSGAN	Pose and expression robust spatial-aware GAN	Jiang et al., 2019 [13]
ResNet	Residual neural network	He Zhang et al., 2016 [28]
CRGAN	Conditional recycle GAN	Li et al., 2018 [14]
ACGAN	Auxiliary classifier GAN	Odena et al., 2017 [15]
CGAN	Conditional GAN	Gauthier et al., 2014 [16]
InfoGAN	Information maximizing GAN	Chen et al., 2016 [17]
LAPGAN	Laplacian pyramid of adversarial networks	Denton et al., 2015 [10]
SAGAN	Self-attention GAN	Zhang et al., 2018 [11]
VAEGAN	Variational autoencoder GAN	Larsen et al., 2016 [12]
BIGAN	Bidirectional GAN	Rui et al., 2020 [18]
AAE	Adversarial autoencoders	Makhzani et al., 2016 [19]
MCGAN	Mean and covariance feature matching GAN	Mroueh et al., 2017 [20]
GRAN	Generative recurrent adversarial networks	Daniel et al., 2016 [12]
LSGAN	Least squares generative adversarial networks	Mao et al., 2016 [29]
WGAN	Wasserstein GAN	Martin et al., 2017 [21]

scene in a broad sense and perform decision-making for the task at hand [27]. CV and image processing are confusing terms and are often used interchangeably. Image processing receives images as the input, processes them, and outputs images, while CV receives images as the input, processes them, and interprets the images. The output generated by CV is an abstract representation of the image's constituent or the entire image. D-GAN is proposed as unsupervised learning, but adding a certain number of labels in practical applications can improve its generating ability. However, it is challenging to acquire many data labels, but a small number of data labels can be obtained. Therefore, combining semisupervised learning and GAN is one of the future directions.

Figure 1(a) represents the general architecture of the deep convolutional neural network (D-CNN). D-CNN is similar to a neural network where D-CNN is built with neurons having learning weights and biases [30]. However, in recent times, D-CNN is widely used over a standard neural network as it is faster and computationally inexpensive compared to neural networks. An image, which is a matrix of pixels, is flattened and fed into the neural network. Furthermore, to flatten an image of size $40 \times 30 \times 1$, 1200 neurons are required at the input layer. Here, the complexity is manageable using a neural network. Colored images have layers corresponding to RGB. A total of 3 layers for each color make the number of neurons required at the input

layer very high. When an image of size $1024 \times 1024 \times 3$ has to be fed into the neural network, 3,145,728 neurons are required at the input layer, which is computationally expensive. The number of neurons needed at the input layer increases exponentially as the size of the image increases.

Figure 1(b) represents the architecture of deep generative adversarial networks (D-GAN), where G captures the data distribution and generates fake data $G(z)$ whose distribution is $p_z(z)$. The generative model improvises to generate distributions similar to p data(x), the real data distribution. The discriminator D is fed either with an actual data sample or generated data sample $G(z)$. The discriminator outputs a probability $f(x)$ belonging to $(0, 1)$, indicating the data source. The generative model is trained to capture the data distribution from the original data. The discriminative model predicts the probability that data have originated from the generator G rather than the original training data. The generator's goal is to generate data close to the distribution of real data and deceive the discriminator. The purpose of the discriminator is to identify the fake data generated by the generator. Model distributions are generated by feeding random noise as the input through a multilayer perceptron. The discriminator has a multilayer perceptron with a classifier at the end [31].

The generator and discriminator compete until the counterfeits generated by the generator are indistinguishable from the data distribution. Through adversarial training, the

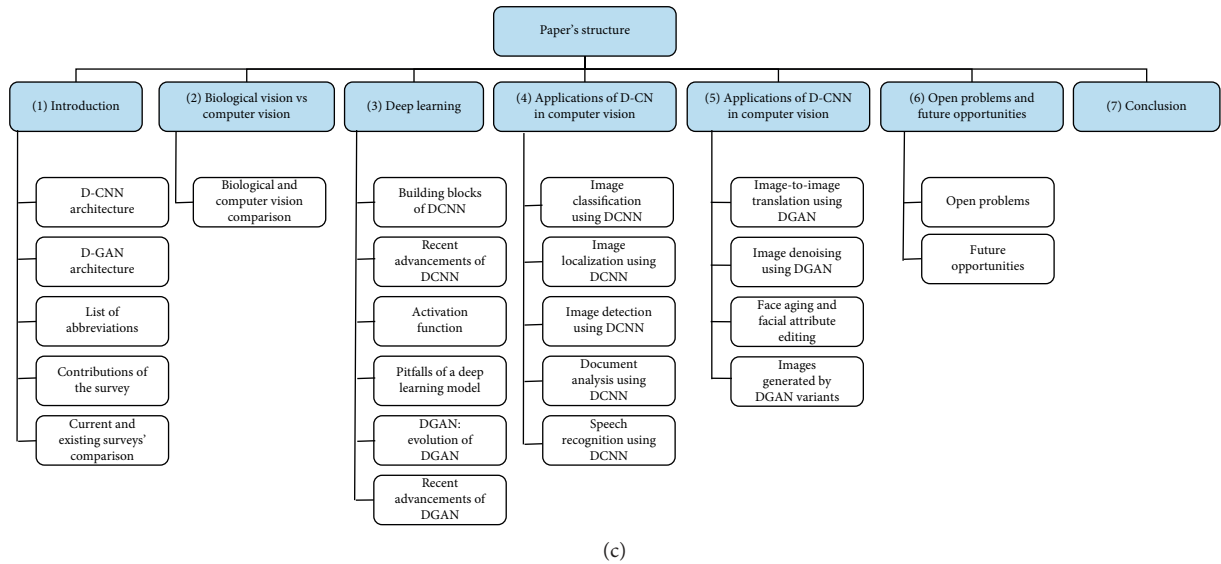
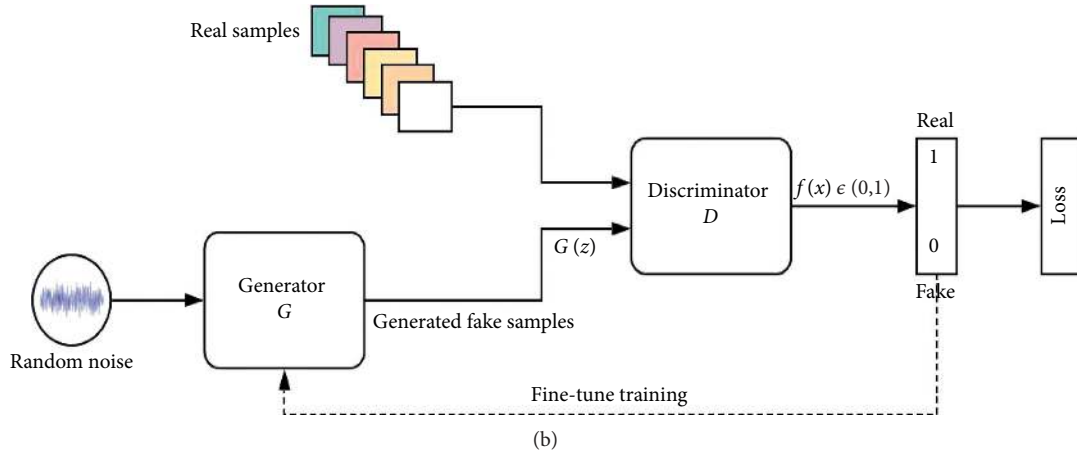
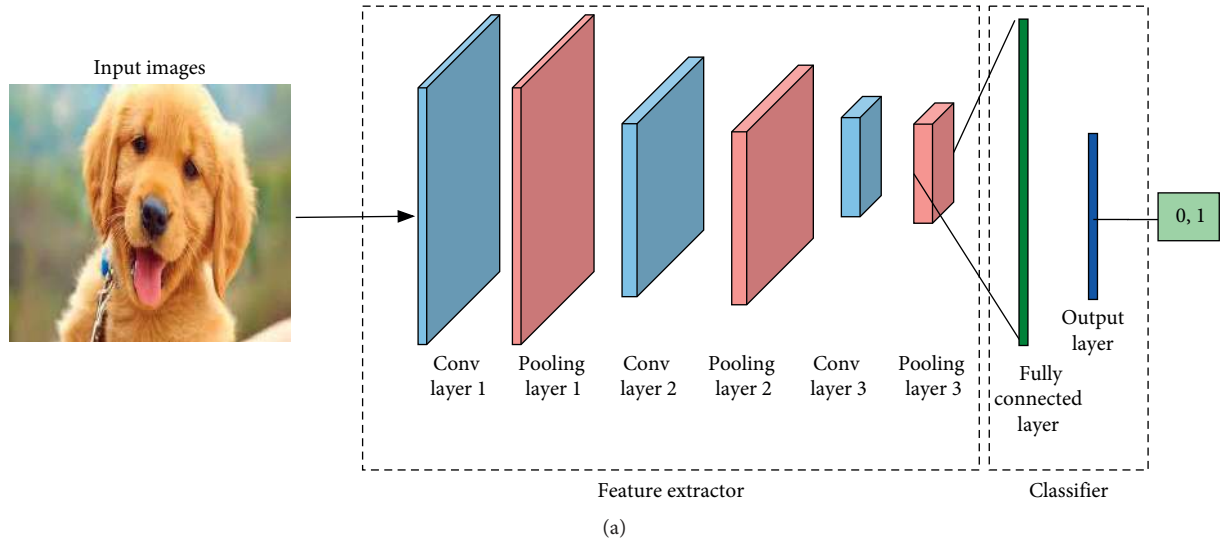


FIGURE 1: (a) The general architecture of deep CNN. (b) The architecture of deep generative adversarial networks. (c) Structure of this survey.

quality of data generated by the generator gradually improves [32]. The quality of data samples generated by the generator and the discriminator's identifying capability improves each iteration interactively. The generator can be any neural network such as artificial neural network, convolutional neural network, recurrent neural network, or long short-term memory, whose task is to learn the data distribution. Simultaneously, a discriminator is essentially a binary classifier capable of classifying the input to be real or fake. The entire network is trained using backpropagation to fine-tune the training. The error is estimated using the sample label, and the discriminator output and the parameters of G are updated using an error backpropagation algorithm. D-GAN is inspired by two-player minimax game theory, which has two players, one benefitting at the loss of the other, and is represented by the following equation [7]:

$$\min G \max D V(D, G) = E_{x \sim p \text{ data}(x)} [\log D(x)] + E_{z \sim p_z(z)} [\log(1 - D(G(z)))], \quad (1)$$

where $p \text{ data}(x)$ is the model data distribution and is the $G(z)$ generated data distribution.

The output of a discriminator is a probability indicating the origin of the data sample. A probability of 1 or a number very close to 1 represents that the data sample is real data. A probability of 0 or a number close to 0 represents the fake data. When the probability is close to 0.5, it indicates that the discriminator finds it hard to identify counterfeit samples. G is trained repeatedly to make D 's output approach 1 for the data samples generated by G . The model is trained until Nash equilibrium is achieved where a change in strategy does not change the game anymore. The Nash equilibrium is achieved when the generator has gained the capability to generate data close to the real data. The discriminator does not distinguish the real data and generator data. The generator is now considered to have to learn the real-data distribution.

1.1. Contributions of This Survey. This paper presents the general architecture of D-CNN, its application, various methodologies adopted, and its application-based performance. An overview of D-GANs is also discussed with their existing variants and their application in different domains. Furthermore, this paper identifies GANs' advantages, disadvantages, and recent advancements in the field of computer vision. Also, it aims to investigate and present a comprehensive survey of the essential applications of GANs, covering crucial areas of research with their architectures. Figure 1(c) shows the structure of the survey. This survey presents a detailed description of D-CNN and D-GAN with their architectures. Recent advancements of D-CNN and D-GAN are discussed with their applications. Activation functions used for the CNN are discussed, and various pitfalls of deep learning with their possible solutions are discussed in detail. Applications of D-CNN and D-GAN are analyzed in Sections 4 and 5. Table 2 shows a comparison between the current survey and existing surveys on D-CNN and D-GAN.

2. Biological Vision vs. Computer Vision

Biological vision has tremendous capabilities in retrieving vital information from visual data and analyzing them for functional needs. The perceptual mechanism used by people and animals to interpret the visual world is diverse. Research on biological vision is an excellent source of inspiration for CV and focuses on computationally understanding brain functions' mechanism for visual interpretation. Understanding the perceptual mechanism of biological vision is the initial step towards interpreting the visual data. Computational understanding of biological vision in the current research studies is based on the framework defined by David Marr [40]. Biological vision can perform tasks with high reliability, even if the visual data are noisy, cluttered, and ambiguous. It can efficiently solve computationally complex problems and that are still challenging for CV. The fundamental goal of CV, the science of image analysis, is to automate computational methods to extract visual information and understand the image's content for decision-making [41, 42]. From CV's perspective, an image is a sequence of square pixels that may be aligned as an array or matrix. At a higher level, the structure of both biological and computer vision is the same. Nevertheless, both systems' objective is the same: to extract and represent the visual data into useful information for making actions.

3. Deep Learning

Deep learning or hierarchical learning has emerged as a subfield of machine learning, which, in turn, is a subfield of artificial intelligence [43]. Artificial intelligence is an effort to make machines think and automatically perform intellectual tasks otherwise performed by humans. AI is a classical programming paradigm where humans craft rules for the data, and the machine outputs the answers. Questions arose as if a machine could automatically learn data processing rules by looking at the data. Machine learning, a new programming paradigm, came into existence as an answer to this question. With machine learning, data and the solutions were fed for the machines to craft the rules. A machine learning model is trained rather than being explicitly programmed. Machine learning and deep learning came into existence when a need arose to solve fuzzy and more complex problems such as language translation, speech recognition, and image classification [44, 45]. At its core, deep learning and machine learning are about learning the representation of the data at hand to get the expected output. Deep learning models are capable of learning complex relationships existing between the inputs and the outputs. Deep learning uses multiple processing layers to discover the data's intricate structure with multiple abstraction levels [46]. The deep in deep learning is a reference to successive layers of representation. Weights parameterize the multiple nonlinear hidden layers in a deep learning model. For a network to correctly map the inputs to its targets in a deep learning model, proper values are to be set for the weights of all layers present in a network.

TABLE 2: Comparison between the current and the existing surveys in the literature.

S. no	Paper title	Survey objective (existing)	Survey objective (current)
1	A survey of generative adversarial networks [33]	State-of-the-art GAN architectures are surveyed, and their application domains on natural language processing and computer vision are discussed. The loss functions of the GAN variants are discussed.	State-of-the-art GAN is discussed along with its performance on the MNIST dataset. Generator and discriminator losses are visually represented for the GAN variants.
2	Recent progress on generative adversarial networks (GANs): A survey [34]	Basic theory and different GAN models are summarized. The models derived from the GAN are classified, and evaluation metrics are discussed.	Variants of the GAN, their application, architecture, methodology, advantage, and disadvantages are analyzed and summarized. Evolution of the GAN with conditions, encoders, loss functions, and process discrete data are separately discussed.
3	A survey of the recent architectures of deep convolutional neural networks [35]	An overview of different layers of D-CNN, namely, the convolutional layer and pooling layer, is discussed. An outline of the pitfalls of deep learning is briefed.	Different layers of D-CNN, namely, the convolution layer, pooling layer, and the operations performed in the convolution and pooling layers, are discussed in detail. A detailed review of deep learning pitfalls, namely, overfitting, underfitting, and data insufficiency, is discussed along with their possible solutions.
4	Deep learning for generic object detection: A survey [36]	Recent achievements in the field of object detection have been discussed.	Recent advancements of the D-CNN in computer vision have been tabulated and discussed with their methodology and performance. Activation functions that are used for computer vision problems are tabulated.
5	A survey on image data augmentation for deep learning [37]	This survey presents the existing methods for data augmentation.	Advantages of data augmentation and comparing results showing the model's performance with and without data augmentation are accomplished.
6	Adversarial-learning-based image-to-image transformation: A survey [38]	This survey presents an overview of adversarial learning-based methods by focusing on the image-to-image transformation scenario.	The existing survey mainly focused on image-to-image translation. This survey discusses several applications based on adversarial learning.
7	Survey of convolutional neural networks for image captioning [39]	This survey presents a shallow overview of image captioning performed using D-CNN.	This survey elaborately discusses various applications using the D-CNN.

Any deep learning problem has an actual target value and the predicted value. The difference between the actual and the predicted value is called the loss function. A distance score, which represents the network performance, is computed based on real and predicted values. Initially, the input's weights are randomly assigned, and the expected output is far from the actual output, and accordingly, the distance score is very high. The weights are then adjusted, the training loop is repeated, and the distance score decreases. Tens and hundreds of iterations are performed over thousands of examples, and a minimal loss value represents that the outputs are close to the target. Deep learning has exponentially improved state-of-the-art object detection, speech recognition, and many other domains [47]. Figure 2 shows the deep learning models. D-CNN has excelled in processing images, speech, audio, and video, while RNN has brought a breakthrough in processing sequential data.

3.1. Deep Convolutional Neural Network. The deep convolutional neural networks, popularly known as D-CNN or

D-ConvNets, are a robust ANN class and are the most established deep learning algorithm that have become dominant in computer vision and tons of other applications [48]. Convolutional layers, pooling layers, and fully connected layers are the D-CNN [49] building blocks [50]. D-CNN is designed to process the data that arrive in multiple arrays or grids through its numerous building blocks. The convolutional and pooling layers' role is to extract the features, while fully connected layers map the extracted features to the output. Deep CNN has multiple convolutions and pooling layers, followed by single or multiple fully connected layers. The input passes through these layers and gets transformed into output through forwarding propagation. Convolution operation and activation function are the backbones of the D-CNN [51].

Tensor, kernel, and feature map are the three essential terminologies to perform convolution operation. Tensor is the input, which is a multidimensional array, the kernel is a small array of numbers, and the feature map is the output tensor, shown in Figure 3. The convolution operation is a linear process where a dot product is performed between the

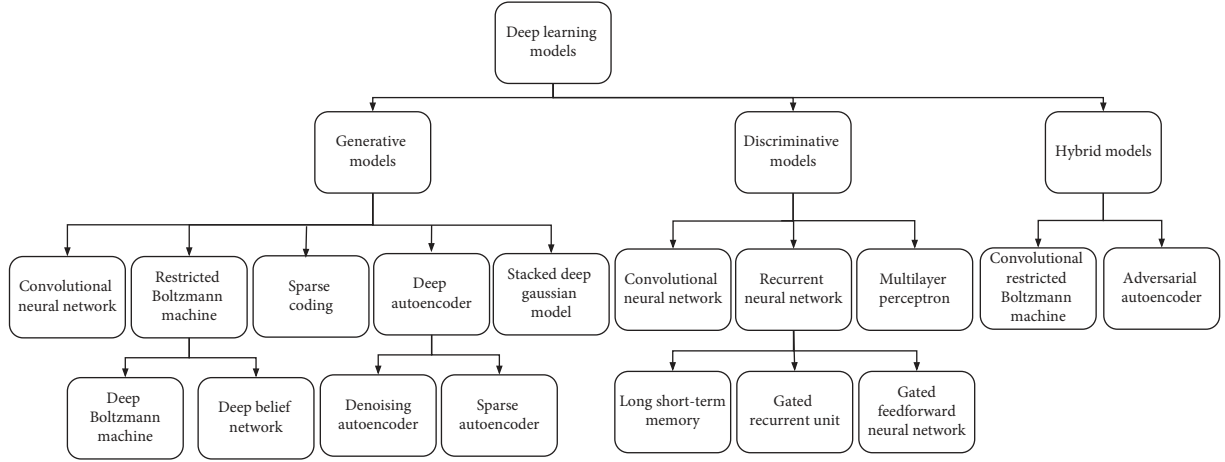


FIGURE 2: Deep learning models—taxonomy.

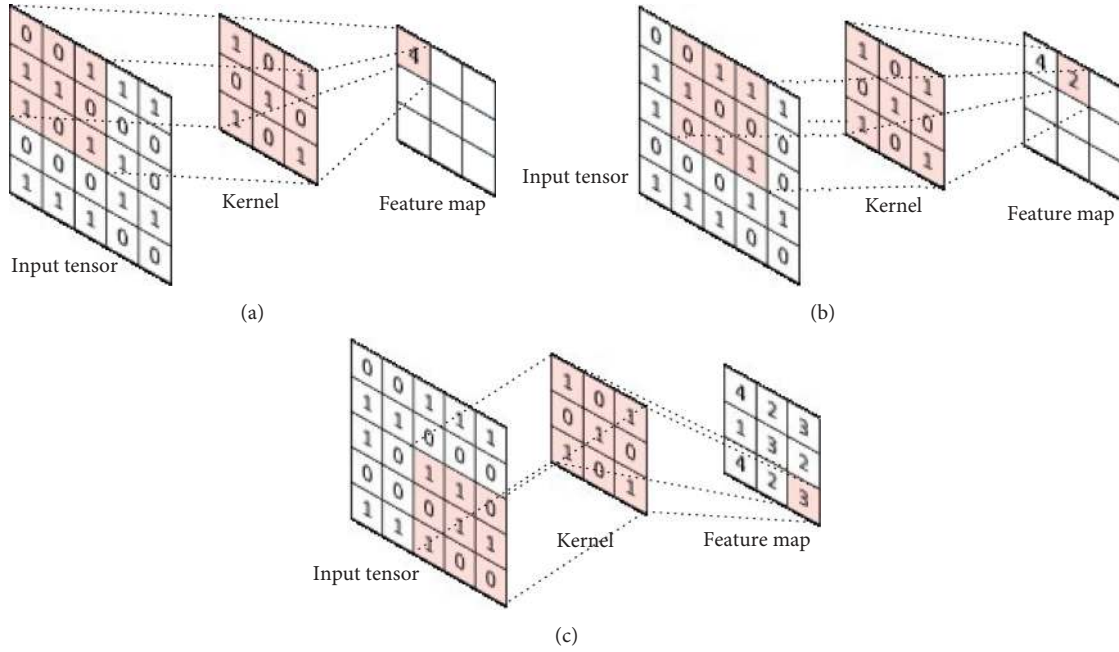


FIGURE 3: Convolution operation.

kernel and the input tensor. Each element of the kernel is multiplied with the corresponding tensor element and summed up to arrive at the output value placed in the feature map's corresponding position [52]. The convolution operation is defined by kernel size and the number of kernels. The kernel size may be 3×3 , 5×5 , or 7×7 based on the size of the input tensor. The number of kernels is arbitrary, each kernel representing various characteristics of the input. The convolution operation is repeated for each kernel. Figures 3(a)–3(c) show an example of a convolution operation. Here, the kernel size is 3×3 , which is applied across each input tensor element to perform a dot product and return the corresponding value for the output tensor. A drawback of the convolution operation is that the feature map shrinks in size compared to the input tensor. Moreover,

this is because the kernel center is not applied across the bordering elements at the input tensor's right. With a 5×5 input tensor, the feature map size shrunk to 3×3 . The size of the feature map f_s for a $t \times t$ tensor and a $k \times k$ kernel is determined using the following formula:

$$f_s = (t - k + 1) * (t - k + 1). \quad (2)$$

Applying this formula, a 49-pixel input will shrink into a 25-pixel feature map, which will further shrink when the process is repeated, resulting in the loss of essential features of the input. Furthermore, to address this issue in deep CNN models with more layers, padding is used, where columns and rows of zeroes are added on all the input tensor sides. Moreover, this is performed to fit the center of the kernel to

the input tensor's rightmost bordering elements for maintaining the size of the feature map the same as that of the input tensor. Figure 4 shows zero padding where rows and columns are added to all the sides of the input tensor. As a result of zero padding, the feature map's size is 5×5 , the same as the input tensor.

The number of pixel shifts performed by the kernel over the input tensor is called stride. When the value of stride is 1, the kernel shifts 1 pixel at a time. When the value of stride is 2, the kernel shifts 2 pixels at a time, and so on. Activation functions that are frequently used are logistic sigmoid, hyperbolic tangent, and ReLU. Table 3 presents the recent advancements of the D-CNN in computer vision.

3.2. Activation Functions. The deep learning mechanism is the input is fed into the network, and to the product of input and the weights, a bias is added. An activation function is then applied to the result, and the same process is repeated until the last layer is reached. Activation functions play a significant role in a neural network to define a neuron's output for a given set of inputs. The activation function takes up the weighted sum of inputs and performs a transformation operation to compress the output between a lower and upper limit. Activation functions are of two types: linear and nonlinear. Deep learning uses nonlinear activation functions for all its classification problems as the output lies between 0 and 1. Without nonlinearity, each layer of the network would execute linear transformations, in which case an equivalent single layer can replace the hidden layers. For a back-propagation to be executed, it is required that the activation function be differentiable. For deep learning, an activation function has to be both nonlinear and differentiable. Some of the standard activation functions in deep learning are sigmoid, tanh, ReLU, and leaky ReLU.

Table 4 shows the most frequently used activation functions. Sigmoid transforms the output between 0 and 1. In recent times, sigmoid has become one of the least used activation functions because of its drawbacks. First, it causes gradients to vanish when the neuron's activation saturates close to 0 or 1; the gradients in this region are close to zero. The second drawback is that the output is not zero-centered. tanh is another activation function that performs better than the sigmoid activation function. The output lies in the range of -1 and 1 [69]. ReLU is the most popular and frequently used activation function in deep learning. The two problems overcome with ReLU are slow training time of the S-type activation function and vanishing gradient [70, 71]. The mathematics behind ReLU is when the output is 0, conversely if, the output is a linear function [72]. The range of the output is between 0 and infinity.

Since ReLU has zero output for the input's negative values, the gradient will be zero at this point because the network will not respond to any variations in the input or the error. This problem can make part of the network passive because of dead neurons. This problem called dying leaky ReLU can overcome ReLU. Leaky ReLU is similar to ReLU, except that leaky ReLU does not make the negative input to

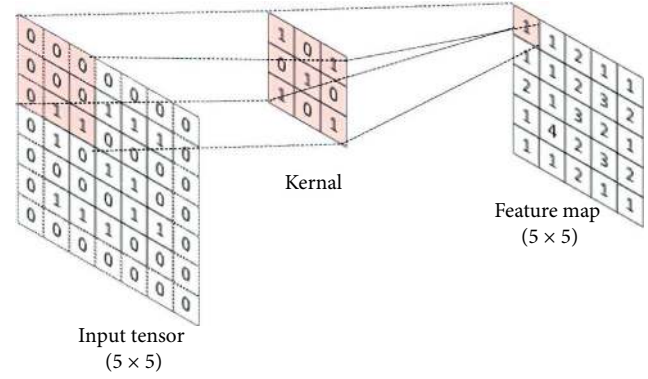


FIGURE 4: Convolution operation with zero padding.

zero. Instead, it gives a small nonzero value of 0.01 in case of a negative regime of the input. The range is between $-\infty$ and ∞ . The purpose of leaky ReLU is to minimize the dying neuron input problem. In multiclass classification problems, the output layer employs softmax as the classification function [73]. Softmax produces output that sums up to a numerical value of 1. So, the output of softmax specifies the probability distribution for n different classes of the target. Moreover, this is why softmax is used explicitly in multiclass classification problems. Due to the gradient disappearance problem, tanh and sigmoid activation functions are not employed on the D-CNN.

3.3. Pitfalls of a Deep Learning Model. The model's performance indicates how well the model is trained and how well it generalizes new or unseen data. Evaluating the performance of a model is a crucial step in data science. The common barriers in creating high-performance models are overfitting, underfitting, and significantly few training data. Figure 5 shows overfitting, and it is said to occur when a model performs so well on a training set. The performance of the model depreciates on the validation set—loss during the training phase decreases, but the loss during the validation phase increases. Furthermore, this is because the model learns even the unnecessary information from the training set; hence, the model's performance is too good on the training set.

Nevertheless, the model fails to perform on the validation set. Overfitting can be addressed by improving the model and obtaining more training data. The model can be enhanced by randomly omitting feature detectors from the model's architecture. This technique is called dropout, developed by Geoff Hinton [74]. A vast number of different networks can be trained in a reasonable time using random dropouts. Thus, different networks are presented for each training case. In a nutshell, the dropout technique assumes that a randomly selected portion of the network is muted for each training case [75]. Furthermore, this is a useful technique as it prevents any single neuron within the network from becoming excessively influential. Thus, the model does not rely too much on any specific feature of the data. Dropout is used when there is an overfitting. Dropout can improve the validation accuracy in later epochs, even if there is no overfitting. Dropout is added based on experimentation, and the usual dropout range is

TABLE 3: Recent advancements of the D-CNN in computer vision.

S. no	Application	The objective of the study	Methodology/ network architecture	Performance	Dataset	Depth/layer sizes
1	Olive fruit variety classification [53]	To provide computer vision methodology for automatic classification of seven different olive fruit varieties using an image processing technique	Six different D-CNN architectures, namely, AlexNet, residual neural network-50, residual neural network-101, inception-residual neural network V2, Inception V1, and Inception V3, were employed.	AlexNet: 89.90%, residual neural network-50: 94%, residual neural network-101: 95.91%, inception-residual neural network V2: 91.81%, Inception V1: 94.86%, and Inception V3: 95.33%.	The dataset was generated from 400 photographs of olive fruits of each variety.	Five convolutional layers
2	Fabric defect detection [54]	On-loom fabric defect detection combines image preprocessing, candidate defect map generation, fabric motif determination, and D-CNN	Seven-layer D-CNN with pairwise potential activation layer as a third hidden layer.	Accuracy for predicting the presence of a defect in an image: 95%. Accuracy for counting the number of defects in an image: 98%.	Fabric defect dataset created using an on-loom fabric imaging system.	Seven-layer CNN, which includes pairwise potential activation layer
3	Polarimetric synthetic aperture radar image classification [55]	Polarimetric synthetic aperture radar image classification classifies image pixels of terrain types, namely, forest, water, grass, and sand	Complex-valued D-CNN and real-valued D-CNN.	Complex-valued D-CNN: 93.4%; real-valued D-CNN: 89.9%.	The performance is analyzed with the airborne steered array radar dataset and the electronically steered array radar dataset.	Six-layer CNN
4	Visual aesthetic quality assessment [56]	To present a biological model for three tasks: aesthetic score regression, aesthetic quality classification, and aesthetic score distribution prediction	A double-subnet gated peripheral-foveal convolutional neural network: a foveal and a peripheral subnet. The peripheral subnet mimics peripheral vision, while foveal extracts fine-grained features.	Gated peripheral-foveal convolutional neural networks (VGG16): 80.70%.	Standard aesthetic visual assessment datasets and photo. Net datasets are used for unified aesthetic prediction tasks.	Nine-layer CNN
5	3D object recognition [57]	To take multiview images captured from partial angles as the input and perform 3D object detection using the 3D CNN	3D object information is encoded from the 3D spatial dimension. 3D kernel, the view images are applied to perform 3D convolution.	ModelNet10 dataset: 94.5%. ModelNet40 dataset: 93.9%.	Pearl image dataset with 10,500 images split into seven classes.	Eight layers
6	Medical image classification [58]	To develop a feature extractor using a fine-tuned D-CNN and to classify medical images	Fine-tuned AlexNet and GoogLeNet D-CNN architectures were used in two ways: (i) as an image feature extractor and to train multiclass support vector machines; (ii) as a classifier to generate softmax probabilities.	GoogLeNet: 81.03%. AlexNet: 77.55%.	Image CLEF 2016 medical image public dataset with 6776 training images and 4166 testing images.	Eight layers with five convolutional layers followed by three fully connected layers and max pooling layers

TABLE 3: Continued.

S. no	Application	The objective of the study	Methodology/ network architecture	Performance	Dataset	Depth/layer sizes
7	Detection and recognition of dumpsters [59]	Visual detection of dumpsters using a twofold methodology with minimal labeling of the dataset to have census of their type and numbers	Google Inception v3 is used, and a D-CNN is pretrained with 1,500,000 images corresponding to 1000 different classes. ReLU is used as an activation function.	94%.	Dumpsters dataset with 27,624 labeled images provided by Ecoembes.	27-layer deep CNN
8	Classification of rice grain images [60]	Localization and classification of rice grain images using contrast-limited adaptive histogram equalization technique	Region-based D-CNN is used to localize and classify rice grains. Dropout regularization and transfer learning were used to avoid overfitting.	81% accuracy is achieved as against 50–76% accuracy of human experts.	MIMR1 to MIMR8 datasets to classify rice into sticky and paddy rice.	Residual neural network-50 is used as the prominent architecture, which is 50 layers deep
9	Object recognition [61]	To evaluate the performance of the inception, recurrent residual convolutional neural network model on benchmark datasets, namely, Tiny ImageNet-200, Canadian Institute for Advanced Research-100, CU3D-100, and Canadian Institute for Advanced Research-10	An inception recurrent residual convolutional neural network, a deep CNN model, was introduced. It utilizes the power of the residual network, inception network, and recurrent convolutional neural network.	72.78% accuracy was achieved, which is 4.53% better than a recurrent CNN.	The model's performance is evaluated on different benchmark datasets, namely, Canadian Institute for Advanced Research, Canadian Institute for Advanced Research-100, Tiny ImageNet-200, and CU3D-100.	Five-layer CNN
10	3D object classification [62]	To successfully classify 3D objects for mobile robots irrespective of starting positions of object modeling	A novel 3D object representation using the 3D CNN with a row-wise max pooling layer and cylinder occupancy grid was introduced.	The results showed that the cylindrical occupancy grid performed better than the existing rectangular algorithms with accuracy 91%.	The performance assessment is done on the ModelNet10 dataset and dataset with six classes collected using the mobile robot.	Three-layer CNN
11	Product quality control [63]	To automatically detect defects in products using fast and robust deep CNN	A simplified D-CNN architecture consisting of nested convolutional and pooling layers with the ReLU activation function is used. It has two parts: a classification frame and a detection frame.	An accuracy of 99.8% is achieved on a benchmark dataset provided by the German Association of Pattern Recognition.	Deutsche Arbeitsgemeinschaft fur Mustererkennung e.V German dataset with six classes. Each class consists of 1000 defect-free images and 150 defective images.	11-layer CNN
12	Recognition of Chinese food [64]	To propose an efficient D-CNN architecture for Chinese food recognition	A 5-layer deep CNN architecture performs a pipeline of processing to optimize the entire network through backpropagation.	97.12% accuracy was achieved, which is better than other bag feature methods.	Chinese food image dataset composed of 8734 images under 25 food categories.	Five-layer deep CNN

TABLE 3: Continued.

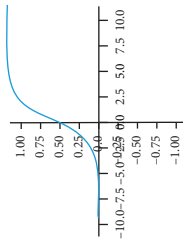
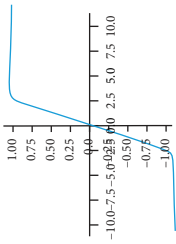
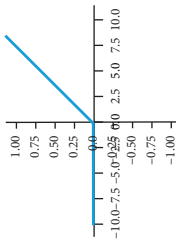
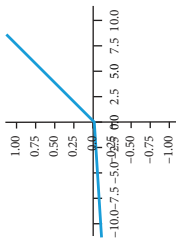
S. no	Application	The objective of the study	Methodology/ network architecture	Performance	Dataset	Depth/layer sizes
13	Age recognition from facial images [65]	Recognition of age from the facial image using pretrained models	MobileNetV2, residual neural network, and pretrained models such as soft stagewise regression networks were used for age recognition. Multiclass classification is performed, which is followed by regression to calculate the age.	Residual neural networks performed better than the other two network models.	The dataset contains 460,723 photographs from the internet movie database cinema website and another dataset with 62,328 pictures from Wikipedia.	Three-layer CNN
14	Person recognition [66]	To develop an effective and efficient multiple-person recognition system for face recognition in random video sequences using the D-CNN	Multiple video faces are detected, and the VGG-19 D-CNN classifier is trained to identify the facial images. The model is tested using standard labeled faces in the wild database.	96.83% accuracy is achieved using the VGG-19 D-CNN classifier.	In the training phase, images from the Chinese Academy of Sciences WebFace database with 9000 classes were utilized. In the validation phase, labeled faces in the wild were used.	Three-layer CNN
15	Multiformat digit recognition [67]	To recognize unconstrained natural images of digits using DIGI-Net	DIGI-Net D-CNN architecture is trained and tested on the MNIST, CVL single digit dataset, and the Chars74K dataset digits.	MNIST: 99.11%, Computer Vision Lab single digit dataset: 93.29%, and digits of the Chars74K dataset: 97.60%.	The performance evaluation is done on MNIST, CVL single digit dataset, and the Chars74K dataset digits.	Seven-layer deep CNN
16	Liver image classification [68]	To develop a perpetual hash-based D-CNN to classify liver images to reduce the time taken to classify liver computed tomography images	A fused perceptual hash-based D-CNN, a hybrid model, was designed to identify malignant and benign masses using computer tomography images.	98.2% accuracy was achieved using the fused perceptual hash-based D-CNN.	The dataset is obtained from Elazig Education and Research Hospital Radiology Laboratory.	Seven-layer deep CNN
17	Object detection [49]	Hardware-oriented ultrahigh-speed object detection using the D-CNN	Two-stage high-speed object detection bounding boxes in the first stage and D-CNN-based classification in the second stage.	MNIST dataset: 98.01%. Self-built dataset: 96.5%.	MNIST and self-built dataset.	Five-layer deep CNN

between 20 and 50 percent of the neurons. The graph is shown in Figures 5(a) and 5(b) which show the accuracy and loss curves after two dropouts with a dropout rate of 0.25 and 0.5 have been added. It can be seen that accuracy has improved, and loss decreased after adding dropout.

In many cases, the data available may not be sufficient to train a model. With significantly few training data, the model may not learn patterns from the training data and inhibit the model's capability to generalize unseen data [76]. It is challenging, expensive, and time-consuming to collect the required new data to train the model [77]. Under such circumstances, data augmentation is warranted, which is a powerful technique for mitigating overfitting. It is a powerful and computationally

inexpensive technique of artificially inflating training data size with the data in hand without collecting new data [71, 78]. One or more deformations are applied to the data, while the labels' semantic meaning is preserved during the transformation [79]. With more data provided to the model, it will generalize well on the validation data. Some popularly used data augmentation techniques are rotation, flip, skew, crop, contrast, brightness adjustment, and zoom in/out [80, 81]. Figure 6 shows different augmentations applied to a single image. Here, flip, rescale, zoom, height, and width shift augmentations are applied to a cat image. Training the model with the additional deformed data makes the model generalize better with unseen data. Figure 7 shows an improvement in accuracy after

TABLE 4: Activation functions used in deep learning.

Name	Plot	Equation	Derivative	Characteristics	Advantages	Disadvantages
Sigmoid		$f(x) = \sigma(x) = 1/(1 + e^{(-x)})$	$f'(x) = f(x)(1 - f(x))$	(i) It takes real-value inputs and outputs a value in the range (0, 1)	(i) Smooth gradient (ii) Have activations within a specific range	(i) Vanishing gradient problem (ii) Not zero-centered
tanh		$f(x) = \tanh(x) = (\sinh(x)/\cosh(x)) = ((e^x - e^{-x})/(e^x + e^{-x}))$	$f'(x) = 1 - f(x)^2$	(i) It ranges from -1 to 1 (ii) Resembles a sigmoidal shape	(i) Derivatives are steeper when compared to sigmoid	(i) Vanishing gradient problem
ReLU		$f(x) = \begin{cases} 0, & \text{for } x < 0, \\ x, & \text{for } x \geq 0. \end{cases}$	$f'(x) = \begin{cases} 0, & \text{for } x < 0, \\ x, & \text{for } x \geq 0. \end{cases}$	(i) ReLU is half rectified, ranging from 0 to infinity	(i) Biological plausibility (ii) Better gradient propagation	(i) Negative values become zero
Leaky ReLU		$f(x) = \begin{cases} 0.01x, & \text{for } x < 0, \\ x, & \text{for } x \geq 0, \end{cases}$	$f'(x) = \begin{cases} 0.01x, & \text{for } x < 0, \\ x, & \text{for } x \geq 0, \end{cases}$	(i) Allows a small nonzero value for negative x	(i) Solves dying ReLU problems	(i) Cannot be used for complex classification problems

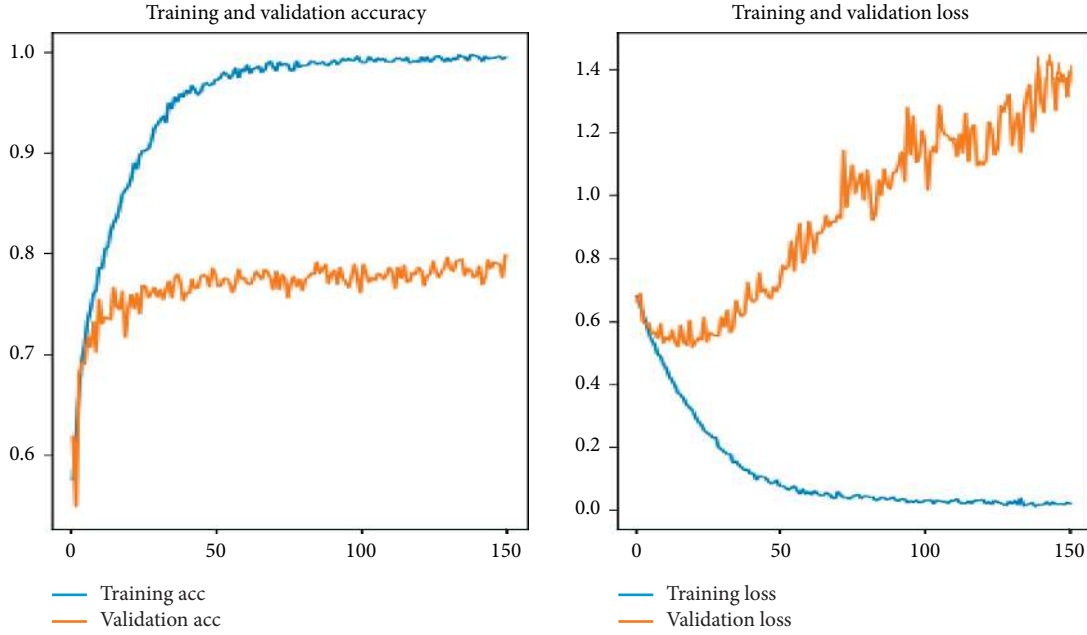


FIGURE 5: Overfitting. (a) Accuracy curve after dropout. (b) Loss curve after dropout.

adding dropout and applying different augmentations to the data.

The next problem usually faced by AI practitioners is underfitting. The model is said to underfit when it cannot learn the patterns even from the training set and exhibits poor performance on the training set. Moreover, this can be addressed by increasing the training data, improving the model complexity, and increasing the training epochs. Deeper models with more neurons per layer can avoid underfitting [82]. Imbalanced datasets are another crucial problem faced as they widely exist in real-world situations and have proven to be the greatest challenge in classification problems. Data access has become comfortable with the advancement of technology; however, data imbalance has become ubiquitous in most of the collected datasets. For example, in medical data, most people are healthy, and unhealthy people are less in proportion than healthy people, significantly affecting classification accuracy. Here, the classes with adequate samples are called the majority class, and classes with inadequate samples are called the minority class. Prediction of minority class becomes problematic as it has a fewer number of samples or insufficient samples.

Several techniques are adopted to handle data imbalance in the dataset. Some of them are weight balancing, over- and undersampling (resampling), and penalizing algorithms [83]. Weight balancing is performed by modifying the weights carried by the training samples when computing the loss. Resampling is one of the frequently adopted techniques where undersampling is done to remove samples from the majority class or oversampling is done to add more samples to the minority class. Oversampling is done by duplicating random records from the minority class. Penalizing algorithm is another technique where the cost of classification mistakes is increased on the minority

class. Label noise is another problem in deep learning, and some of its sources are nonexpert labeling, automatic labeling, and data poisoning, adversaries. Dan Hendrycks et al. [84] recommended a loss correction technique to utilize trusted data with clean labels. The authors effectively used trusted data to overcome the effects of label noise on classification.

3.4. Deep Generative Adversarial Network. Generative adversarial networks are a framework proposed by Goodfellow et al. in the year 2014 [7]. Significant improvements were achieved in computer vision applications such as image super-resolution [85], image classification [86], image steganography [87], image transformation [88], video generation, image synthesis [89], video super-resolution [90], and image style transformation. Variants of the D-GAN model were also proposed in recent times.

Figure 8 shows the D-GAN architecture where the generator generates fake images of human faces, and the discriminator's job is to distinguish the real faces from the fake faces. In general, plausible data are generated by the generator. These data generated by the generator become negative examples of training the discriminator. The discriminator, a binary classification neural network, takes in the real samples and the samples from the generator and learns to distinguish the real samples from the generator's fake samples. Two loss functions, generator loss and discriminator loss, are backpropagated to the generator and discriminator. The discriminator ignores the generator loss. The generator and the discriminator update the weights based on the loss, where the noise samples i ranging from 1 to m are represented by



FIGURE 6: Data augmentation of a single image. (a) Original image. (b) Rescale. (c) Height shift augmentation. (e) Width shift augmentation.

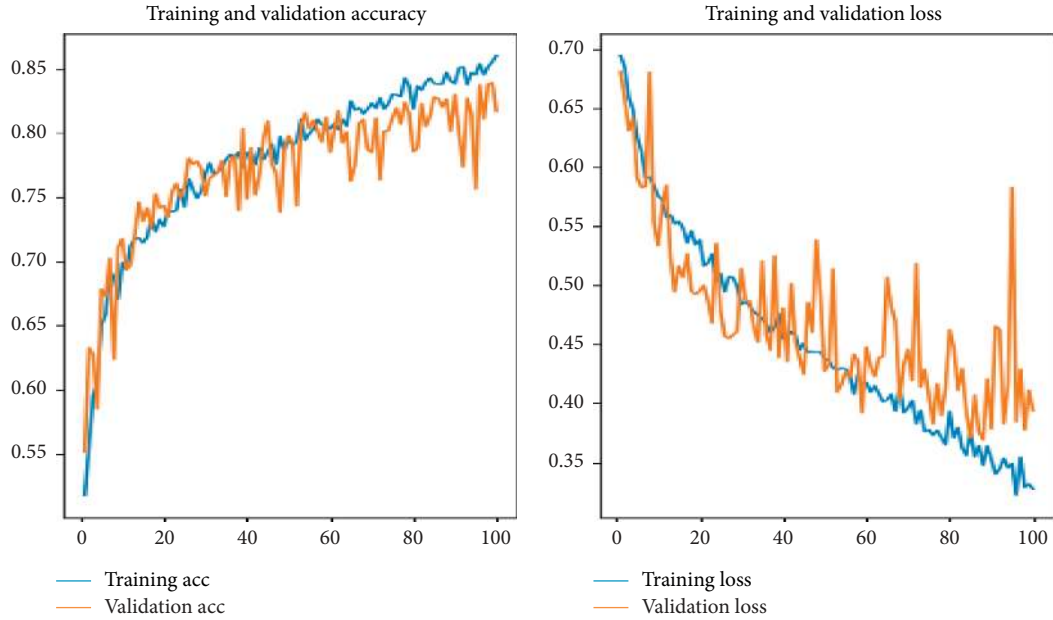


FIGURE 7: Accuracy and loss after adding dropout and applying augmentations.

$$D_{\text{loss}} = \frac{1}{m} \sum_{i=1}^m [\log(D(x^i)) + \log(1 - D(G(z^i)))], \quad (3)$$

$$G_{\text{loss}} = \frac{1}{m} \sum_{i=1}^m -\log(D(G(z^i))).$$

3.5. Evolution of the Deep GAN. With deep GAN's advent by Goodfellow, several variants of the deep GAN were proposed for various CV applications. These deep GAN variants have their own architecture, methodology, advantages, and disadvantages but with the same two-player minimax game theory as the base. Figure 9 shows D-GAN's evolution with conditions, encoders, loss functions, and process discrete data. Table 5 shows D-GAN's evolution with its application, architecture, methodology, advantage, and disadvantage.

D-GAN is successfully used in many computer vision applications, and image generation is at the forefront of all these. D-GAN generates images, gradually enhancing the resolution and the quality of images generated. Variants of D-GAN are used for various applications such as image transformation, image deraining [88], increasing image resolution, facial attribute transformation, and fusion of the image. Table 6 shows some of the progressively increasing applications of the D-GAN in computer vision.

4. Applications of the D-CNN in Computer Vision

Most of the D-CNN applications are related to images, while applications of the D-GAN are related to data generation. This section will progress through the essential applications of the D-CNN.

4.1. Image Classification Using the D-CNN. There are several image classification tasks performed using D-CNN [11, 104–109]. One of the vital image classification tasks is handwritten digit recognition which recognizes numbers between 0 and 9, where the data from the MNIST database are obtained to predict the correct label for the handwritten digits. MNIST is a database of handwritten numbers widely used as a testbed for various deep learning applications. It has 70,000 images, of which 60,000 are training images and 10,000 are testing images [110]. Figure 10 shows sample images from the MNIST dataset. The images are greyscaled with 28×28 pixels, as represented in Figure 10. The 28×28 pixels are flattened into a 1D vector of size 784 pixels, and each of these pixels has values between 0 and 255. The black pixel takes the value 255, while the white pixel takes the value one, and various other shades of grey take values between 0 and

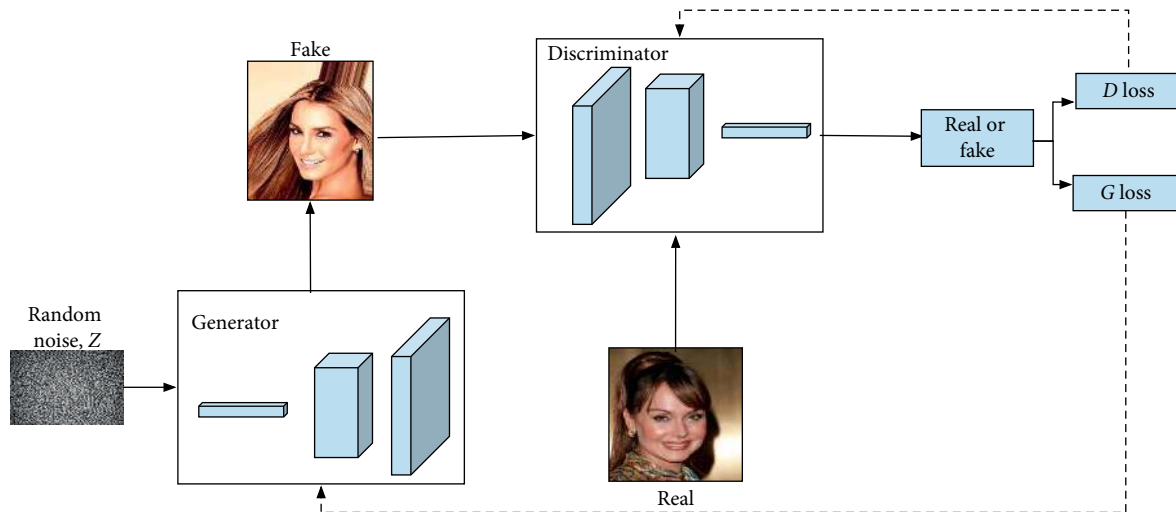


FIGURE 8: Deep generative adversarial network—human face generation.

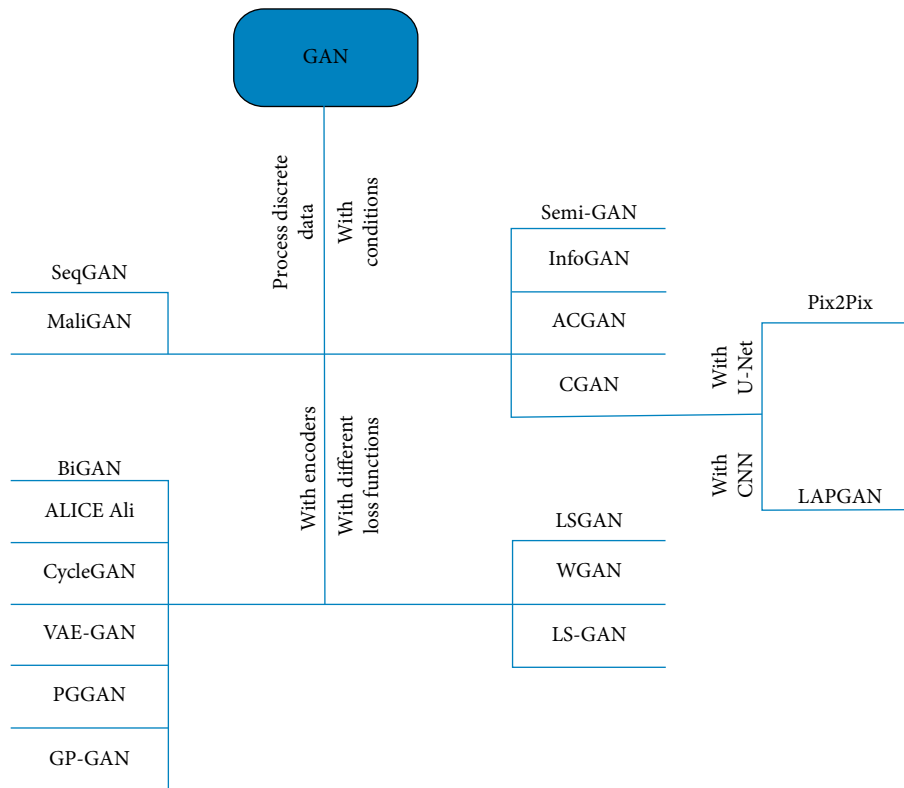


FIGURE 9: Evolution of the deep GAN with conditions, encoders, loss functions, and process discrete data separately.

255. Handwritten digits are recognized using the D-CNN, which is considered the most suitable model for performing this task.

Data are downloaded from the MNIST database, and it takes some time to download the data during the first run, and the subsequent runs fetch the cached data. The data obtained from the MNIST database have features and labels. The features range from 0 to 225, corresponding to pixels of 28×28 images representing digits 0 through 9. The labels represent digits 0–9 of the respective image. It is normalized

to scale the data to be between 0 and 1. The actual image from the MNIST dataset and the normalized image are represented in Figures 11(a) and 11(b).

Handwritten digit recognition is implemented using eight hidden layers, where the first layer is the convolutional layer used for feature extraction. ReLU is used as the activation function with 32 filters and a kernel size of 3×3 pixels. Another convolutional layer is used with ReLU as the activation function, 64 filters, and kernel size of 3×3 pixels. The next hidden layer is a pooling layer where max pooling

TABLE 5: Evolution of the deep GAN.

S. no	Model	Application	Network architecture	Methodology	Key advantage	Major limitation
1	SRGAN [9]	Image super-resolution	Generator: two convolutional layers with small 3×3 kernels and 64 feature maps, batch normalization layers, and parametric ReLU. Discriminator: leaky ReLU activation function.	A perceptual loss function is proposed, which is composed of content loss and adversarial loss	Low-resolution images are converted into a high-resolution image for 4x upscaling factors	Texture information is not real enough, accompanied by some noise
2	ACGAN [15]	Image synthesis	Generator: has a series of deconvolutional layers, also known as transposed convolutional layers. Discriminator: has a set of 2D convolutional layers with leaky ReLU followed by linear layers and softmax and sigmoid functions for each of its outputs.	Two variants of the model were trained to generate images of resolution 64×64 and 128×128 spatial resolutions	Accuracy can be assessed for individual classes	Ignores the loss component arising from class labels when a label is unavailable for a given training image
3	CGAN [16]	Image-image translation, image tagging, and face generation (Gauthier, J. (2014))	Generator and discriminator are conditioned on some arbitrary external information with the ReLU activation function and sigmoid for the output layer.	Minimize the value function for G and maximize the value function for D	CGAN could easily accept a multimodal embedding as the conditional input	CGAN is not strictly unsupervised; some kind of labeling is required for them to work
4	InfoGAN [1, 2]	Facial image generation	Generator: upconvolutional architecture and normal ReLU activation function. Discriminator: leaky ReLU with leaky rate 0.1 applied to hidden layers as nonlinearity.	Learn disentangled representations by maximizing mutual information	InfoGAN is capable of learning disentangled and interpretable representation	Sometimes, it requires adding noise to the data to stabilize the network
5	DCGAN [91]	General image representations	Generator: the ReLU activation except for the output layer, which uses the tanh function and batch norm. Discriminator: leaky ReLU for high-resolution modeling and batch norm.	The hierarchy of representations is learned from object parts	Stable, good representations of images, easy convergence	Gradients disappear or explode
6	LAPGAN [10]	Generation of images	Laplacian pyramid of convolutional networks.	Image generation in a coarse-to-fine fashion	Independent training of each pyramid level	Nonconvergence and mode collapse
7	SAGAN [11]	Generation of images	Spectral normalization is applied to the GAN generator.	Realistic images are generated	Inception score is boosted, Frechet inception distance is reduced, and images are generated sequentially	Attention is not extended

TABLE 5: Continued.

S. no	Model	Application	Network architecture	Methodology	Key advantage	Major limitation
8	GRAN [12]	Generating realistic images	Recurrent CNN with constraints.	Images are generated by incremental updates to the canvas using a recurrent network	Sequential generation of images	Samples collapse on training for a long duration
9	VAE-GAN [92]	Facial image generation	GAN discriminator is used in place of a VAE's decoder to learn the loss function.	Combines GAN and VAE to produce an encoder of data into the latent space	Its advantage, GAN, and VAE are combined in a single model	The major drawback of the VAE is the blurry output it generates
10	BIGAN [18]	Image generation	Generator: in addition to G, it has an encoder to map the data to latent representations. Discriminator: discriminates both data and latent spaces.	Learn features for related semantic tasks and use them in unsupervised settings	Minimization of the reconstruction loss	Generator and discriminator are highly dependent on each other in the training phase
11	AAE [19]	Dimensionality reduction, data visualization, disentangling the style of the image, and unsupervised clustering	Generative autoencoder.	Variational inference is performed by matching the arbitrary prior distribution with the autoencoder's aggregated posterior of the hidden code vector	Balanced approximation; this method can be extended to semisupervised learning and is better than variational autoencoders	Samples generated are blurry and smoothened (Zhang, J. et al., 2018)
12	Pix2Pix [93]	Image-image translation	Generator: UNET architecture. Discriminator: PatchGAN classifier. ReLU activation function. Batch normalization.	The network learns the loss associated with the data and the task, making it applicable for a wide variety of tasks	Parameter reduction, and realistic images are generated	The required images are to be one-one paired

with 2×2 pixels as pool size is used. Next to the pooling layer is the dropout, a technique adapted to prevent overfitting in the neural network [111]. The dropout technique's key idea is to randomly drop a few units from the network and its connections during the training to reduce overfitting significantly. A dropout rate of 0.25 randomly drops out 1 in 4 units from the network. Between the convolutional layer and the fully connected output layer is the flatten layer. The flatten layer aims to transform the 2D matrix into a 1D vector fed into the fully connected layer. The flattened 1D vector is then passed on to the fully connected layer with ReLU as the activation function. Another dropout with a dropout rate of 0.50 is used. Finally, the output layer with the softmax activation function is used. The softmax activation function is used as its role is to specify the probability distribution for ten different classes. Since the task in hand is a multiclass classification, the output layer has ten nodes or perceptron corresponding to each of the ten categories to predict each class's probability distribution. The perceptron with the highest probability is picked, and the label associated with it is returned as the output. The model is fit over 12 epochs. The test accuracy achieved is 98.56%, and the test loss is 0.0513, as represented in Figures 12(a) and 12(b). It can be seen that the accuracy increases with the increase in epoch, and loss decreases with the increase in accuracy.

Image classification is a classical problem of computer vision and deep learning. It is challenging because of the image's variations due to light effects and misalignments [112]. Image classification in a computer sense is a course of action for grouping and categorizing images and labeling them based on their features and attributes [80]. It trains computers to use a well-defined dataset to interpret and classify images to narrow the gap between human vision and computer vision [113]. Some of the existing use cases of image classification are gender classification, social media applications such as Facebook and Snapchat which use image classification to enhance the user experience, and self-driving cars where various objects on their path, namely, vehicle, people, and other moving objects, are recognized [114].

Amerini et al. [115] proposed a novel framework called FusionNet by combining two D-CNN architectures to identify the source social network based on the images. 1D-CNN learns discriminative features, while 2D-CNN architecture infers unique attributes from the image. The learned features are fused using FusionNet, and then the classification is performed. Distinctive traces of social networks embedded in the images are exploited to identify the source. The full-frame images are broken into fixed dimension patches, and the patches are then classified independently. Each of the image patches is processed with D-CNN, and the predictions

TABLE 6: Recent advancements of the deep GAN in computer vision.

S. no	Application	The objective of the study	Methodology/network architecture	Performance	Dataset
1	Facial attribute transformation [14]	To develop a novel conditional recycle D-GAN that can transform high-level face attributes retaining the face's identity	The developed conditional recycle D-GAN model has two phases. In the first phase, conditional D-GAN attempts to generate fake facial images with a condition. In the second phase, recycling D-GAN is used to generate facial images to modify the attributes without changing the identity.	The results were compared with existing D-GAN architectures to prove the efficiency of CRGAN. CRGAN performed better than existing D-GAN architectures.	CelebA dataset
2	Fusion of images [94]	To propose a method to fuse images belonging to different spectra using D-GAN	FusionNet architecture was developed using the Pix2Pix architecture to generate fused images from different spectra fragments of images.	The proposed FusionNet model was compared with existing fusion methods such as the cross bilateral filter, the weighted least squares, and the sparse joint representation. The FusionNet technique performed equally well with the existing methods.	Dataset provided by experts
3	Synthesis of high-quality faces [95]	To propose a D-GAN-based method to synthesize high-quality images from polarimetric images	The proposed model has a generator subnetwork built based on an encoder-decoder network and a discriminator subnetwork. The generator is trained by optimizing identity loss, perceptual loss, and identity preserving loss.	The qualitative and quantitative performance of the developed model is compared with state-of-the-art methods. The use of perceptual loss generated visually pleasing results.	A dataset with polarimetric and visible facial signatures from 111 subjects
4	Vehicle detection in aerial images [96]	To develop a lightweight deep CNN model to detect vehicles in aerial images using D-GAN effectively	The architecture has two parts: lightweight deep CNN was developed to accurately detect vehicles and a multicondition-constrained GAN to generate samples to cope with data insufficiency.	The model tested on the Munich dataset achieved a mean average precision of 86.9%.	Performance evaluation is done on Munich public dataset and the collected dataset.
5	Image deraining [97]	To develop a deep learning model to remove rain streak from images	A feature supervised D-GAN was developed to remove rain from a single image. Feature supervised D-GAN has two subnets to generate derained images that are very close to the real image.	The developed model was tested on synthetic and real-world images. It showed better performance than the existing state-of-the-art deraining methods.	Performance evaluated on real-world images and two synthetic datasets.
6	Scene generation [12]	To develop a model to generate scenes based on the conditional D-GAN	A model named PSGAN was developed to generate a multidomain particular scene. The quality of the images is improved by spectral normalization.	The developed model is compared with Pix2Pix and StarGAN. 97% accuracy is achieved using PSGAN as against 95% accuracy achieved using StarGAN.	The performance of the model is evaluated on MNIST, CIFAR-10, and LSUN.

TABLE 6: Continued.

S. no	Application	The objective of the study	Methodology/network architecture	Performance	Dataset
7	Human pose estimation [98]	To develop self-attention D-GAN to perform human pose estimation	The D-GAN model used hourglass networks as its backbone. Hourglass architecture has Conv-Deconv architecture and residual blocks. The generator predicts the pose, while the discriminator enforces structural constraints to refine the postures.	The model outperformed the state-of-the-art methods on benchmark datasets.	The performance of the model is evaluated on Leeds Sports Pose and MPII human pose dataset.
8	Automatic pearl classification [57]	To develop deep learning models to perform automatic classification of pearls	Multiview GAN is used to expand the pearl images dataset. Multistream CNN is trained using the expanded dataset.	The image generated using the multiview GAN is used to reduce the existing multistream CNN significantly.	The dataset includes 10,500 pearls, with seven categories and each category containing 1,500 pearls.
9	Image dehazing [99].	To develop a deep learning model to recover the image's texture information and enhance hazy scenes' visual performance	Attention-to-attention generative network model is developed to map hazy images to haze-free images. All the instance normalization layers are removed to generate high-quality images.	The developed model performed better state-of-the-art methods for both real-world and synthetic images	NYU2 synthetic dataset with 1300 images and SUN3D synthetic dataset with 150 images.
10	Gesture recognition [100]	To propose a new gesture recognition algorithm based on D-CNN and DCGAN	For a particular gesture, the model recognizes the meaning of the gesture. DCGAN is used to solve overfitting in case of data insufficiency. Preprocessing is done to improve illumination conditions.	An accuracy of 90.45% is achieved.	Data collected using a computer containing 1200 images for each gesture.
11	Face depth estimation [101]	To develop a D-GAN-based method to estimate the depth map for a given facial image	D-GAN architecture is used to estimate the depth of a 2D image for 3D reconstruction. Data augmentation is done to improve the robustness of the models. Transformations such as slight rotation clockwise, Gaussian blur, and histogram equalization were applied to the image.	Several variants of the D-GAN were evaluated for depth estimation. Wasserstein GAN was found to be the most robust model for depth estimation.	The Texas 3D face recognition database and Bosphorus database for 3D face analysis.
12	Image enhancement [102]	To propose an image enhancement model using the conditional D-GAN based on the nonsaturating game	The super-resolution method is combined with the D-GAN to generate a clearer image. The architecture has 23 layers composed of convolution layers with the ReLU activation function.	The model is compared with existing methods, which showed an improvement in peak signal-to-noise ratio by 2.38 dB.	Images from Flickr and ImageNet datasets were used without augmentation.
13	Retinal image synthesis [103]	To propose multiple-channels-multiple-landmarks, a preprocessing pipeline to synthesize retinal images from optic cup images	Residual neural network and U-Net were integrated to form residual U-Net architecture. Residual U-Net is capable of capturing finer-scale details. Multiple-landmark maps comprise of batch normal layer, convolution layer, and ReLU activation. The final layer has a sigmoid activation function.	The proposed multiple-channels-multiple-landmarks model outperformed the existing single vessel-based methods. Pix2Pix, using the proposed method, generated realistic images.	Public fundus image datasets DRIVE and DRISHTI-GS were used.

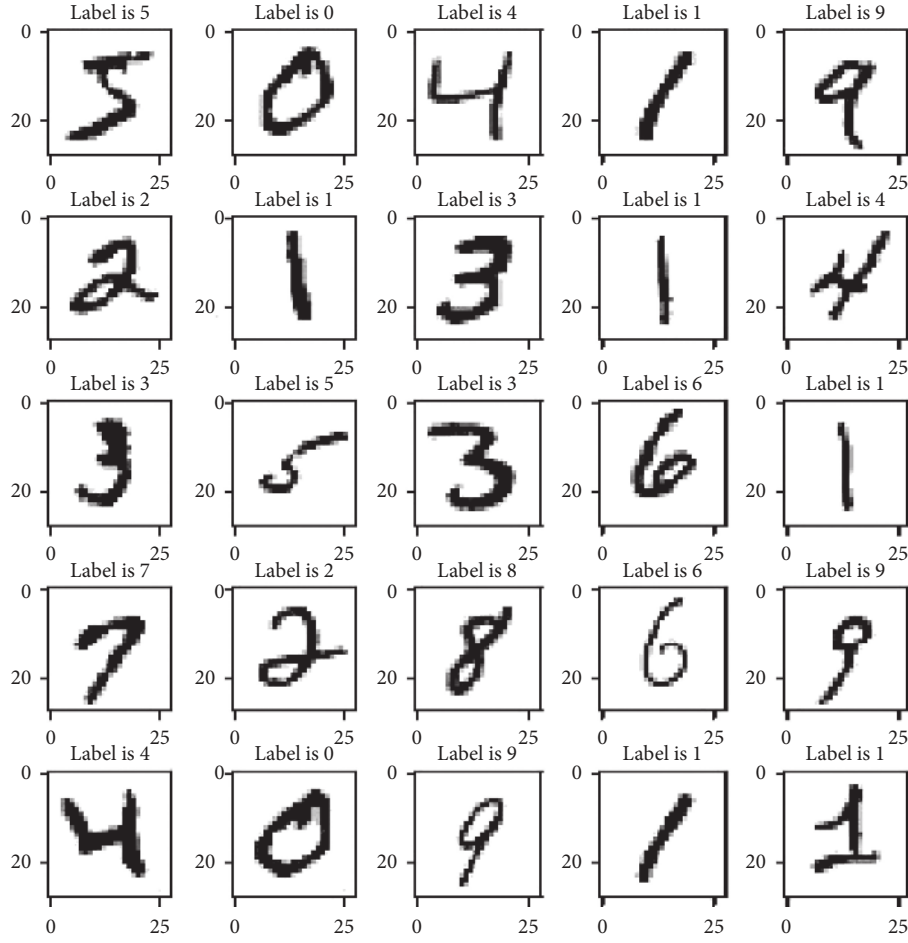


FIGURE 10: Sample images from the MNIST dataset.

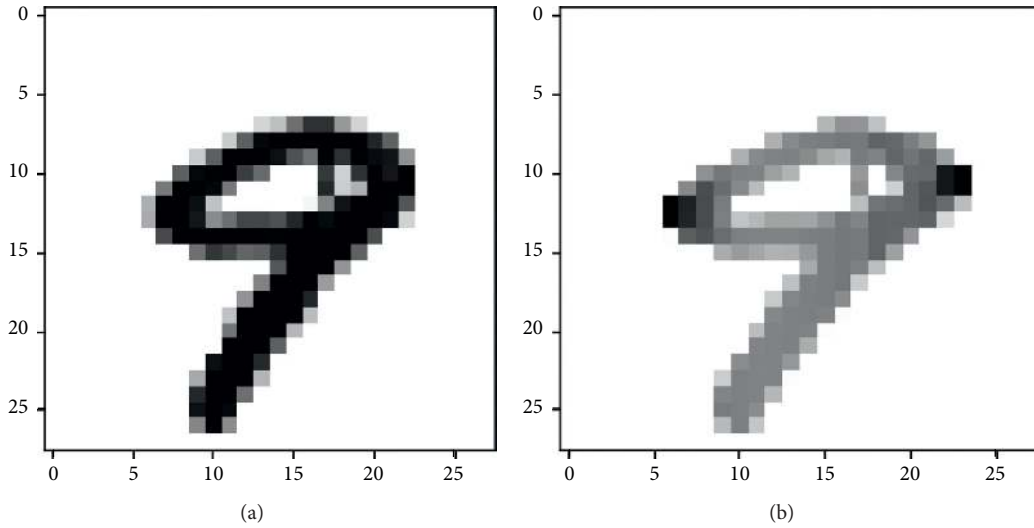


FIGURE 11: (a) Image from the MNIST dataset. (b) Normalized image.

are obtained. Furthermore, to get the prediction at the image level, a voting strategy is applied at each patch. The label with the majority vote is assigned as the final prediction label. The average accuracy of 94.77% is achieved at the patch level.

4.2. Image Localization and Detection Using the D-CNN. On a glance over the object, human vision is capable of detecting the object, its size, location, and various other features. Object detection using deep learning allows

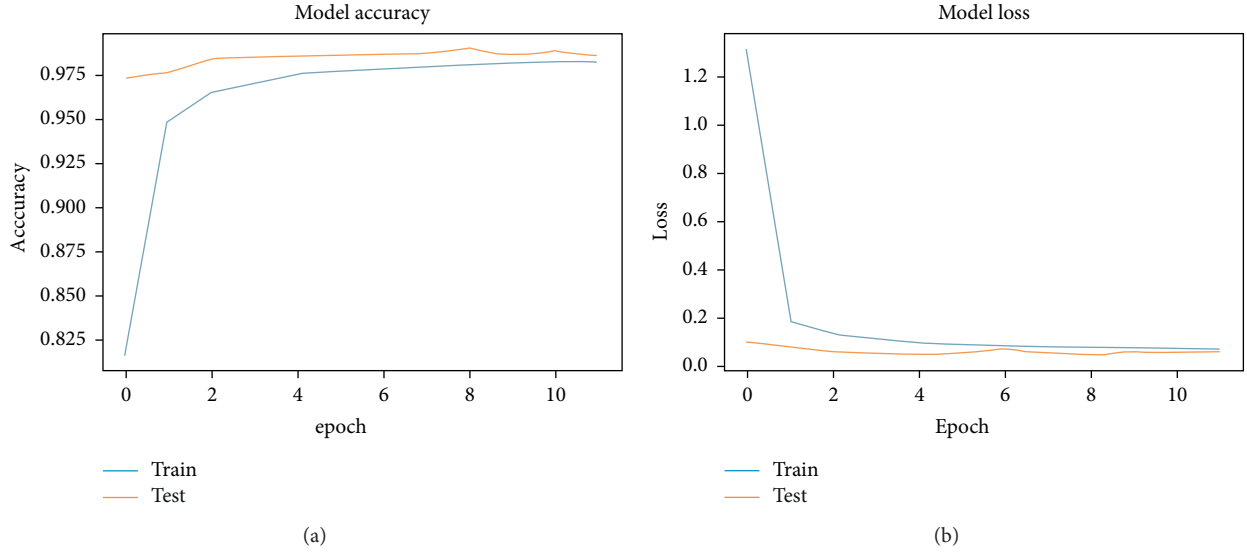


FIGURE 12: Accuracy and loss curves. (a) Accuracy vs. epoch. (b) Loss vs. epoch.

computers to play a crucial role in many real-world applications such as robots, smart vehicles, and self-driving cars [116]. Object detection is one of the most challenging problems and the most important goal of computer vision. Object detection involves identifying different objects in an image using a bounded box. The identified objects can be further analyzed at a granular level to digging deeper into the image. Earlier object identification was performed by splitting the image into multiple pieces and then passing them into a classifier for object detection. Splitting the images into multiple pieces is performed using a sliding window algorithm. In this approach, the detection window is slid through the actual image at multiple positions, and each grid is a smaller piece of the image. Robust visual descriptors needed for object detection are extracted from the image using image processing. The convolutional neural network used visual descriptors to make object or nonobject decisions [117]. Since the process had to be repeated multiple times, it was computationally expensive.

Moreover, to overcome the sliding window algorithm's shortcomings, object detection was performed using image segmentation. Segmentation is categorized into boundary-based, thresholding, region-based, and boundary-based. When a digital image is passed, the neurons are synchronized based on pixels with similar intensities to form a connected region [118]. It can be contextual if spatial relationships in an image are considered or noncontextual if spatial relationships are not considered. The goal of image segmentation is to alter the representation of an image into a form that is meaningful and easier for analysis. The accuracy of object detection is based on the quality of image segmentation. Similar to image classification problems, networks that are deeper exhibited better performance in object detection. Object localization is the next level of object classification where the objects' position was also determined with a bounding box and labeling the objects. The difference between localization and detection is that

classification with localization handles only one object, whereas detection finds multiple objects in an image and labels.

Tu et al. [119] proposed a method to detect passion fruits based on multiple-scale faster region-based CNN. The detection phase involves multiple-scale feature extractors that extract low- and high-level features. Data augmentation is done to enlarge the training data size. Pretrained residual neural network-101 architecture is used for object detection.

4.3. Document Analysis Using the D-CNN. Documents are the source of information for several cognitive processes, namely, graphic understanding, document retrieval, and OCR. Document analysis plays a crucial role in cognitive computing to extract information from document images. Document analysis is performed by identifying and categorizing images based on regions of interest. There are several existing methods for document analysis, such as pixel-based classification methods, region-based classification method, and connected component classification method. The region-based classification method segments document images into zones and classifies them into semantic classes. Pixel-based classification methods perform document analysis by taking each pixel and generate labeled images using the classifier. The connected component method creates the object hypothesis using local information, further inspected, refined, and classified [120].

D-CNN is widely adopted for document analysis to reduce computational complexity, cost, and data without compromising accuracy. With D-CNN, it is possible to classify images directly from segmented objects without extracting handcrafted features. Maryem Rhanoui et al. [121] performed document-level sentiment analysis using a combination of D-CNN and bidirectional long short-term memory and achieved an accuracy of 90.66%. The features are extracted by the D-CNN, and the extracted features are

passed as the input to long short-term memory. Vectors' built-in word embedding is passed as the input to the CNN. Four filters are applied, and the layer of max pooling is applied after each filter. The results of max pooling were concatenated and passed as the input to binary long short-term memory. The output of binary long short-term memory is passed as the input to a fully connected layer. The fully connected layer connects each piece of information from the input with output information. Finally, the softmax function is applied as an activation function to produce the desired output by assigning classes to articles.

4.4. *Speech Recognition Using the D-CNN.*

Human-machine interaction for intelligent devices, namely, domestic robots, smartphones, and autonomous cars, is becoming increasingly common in daily life. Hence, noise robust automatic speech recognition has become very crucial for the human-machine interface. The basic idea behind speech recognition is to utilize the speaker's lip movement's visual information to complement the corrupted audio speech inputs. Automatic speech recognition models the relationship between phones and acoustic speech signals by extracting features and classifying speech signals. Furthermore, this is usually performed in two steps, where in the first step, the raw speech signal is transformed into features using dimensionality reduction and information selection. The second step estimates phonemes using generative or discriminative models. Phoneme class conditional probabilities can be estimated using the D-CNN through the raw speech signal as the input. The features are learned from the raw speech signal in both continuous speech recognition and phoneme recognition tasks.

Kuniaki Noda et al. [122] proposed a CNN-based approach for audiovisual speech recognition. Here, the authors first used a denoising autoencoder to acquire noise features. Then, the authors used the CNN to extract features from mouth area images. The training data for the CNN were raw images and their corresponding phoneme outputs. Lastly, the authors applied the multistream hidden Markov model to integrate audio and visual hidden Markov models trained with corresponding features. The model achieved a 65% word recognition rate with denoised mel-frequency cepstral coefficients with the signal-to-noise ratio under 10 dB for the audio signal input.

5. Applications of the D-GAN in Computer Vision

5.1. Image-to-Image Translation Using the Deep GAN. Remarkable progress has been achieved in image-image translation with the advent of the deep GAN. The image-image translation aims to learn the mapping to translate the image within two different domains, from the source to a target domain, without losing the original image's identity and reducing the reconstruction loss. Some of the essential image-image translations are converting the real-world images into cartoon images, coloring the greyscale images, and changing a nighttime picture to a daylight picture.

D-GAN's role is to confuse the discriminator by generating images that are close to the real images. D-GAN is incredibly successful in super-resolution, representation learning [123], image generation [124, 125], and image-image translation.

Kim et al. proposed a novel method for image-image translation by incorporating a learnable normalization function and a new attention module. Existing attention-based models lacked behind in handling the geometric changes. This model is incredibly successful in translating images with massive shape changes. The auxiliary classifier is used to obtain an attention map to distinguish between source and target domains. Furthermore, this is done to focus on the region of interest, ignoring other minor regions. Attention maps are inserted both in the generator and discriminator to focus on the region of interest. The attention map embedded in the generator focuses on the essential areas that distinguish the two domains. In contrast, the attention map embedded in the discriminator focuses on distinguishing the target domain's real and fake image. The choice of the normalization mechanism dramatically improves the quality of the transformed images. Adaptive Layer-Instance Normalization is used to select a ratio between layer normalization and instance normalization adaptively, and the parameters are learned during the training process. The class activation map gives discriminative image regions to determine the class. The model's performance is superior to the existing state-of-the-art methods on both style transfer and object transfiguration.

5.2. Image Denoising. Image denoising removes noise from images retaining the detailing of the images. Image denoising is significantly improved with the advancement in the D-CNN [126]. However, D-CNN models focus mainly on reducing the mean squared error resulting in images lacking high-frequency details. Furthermore, to overcome this issue, D-GAN is applied to remove noise from images [127, 128]. Zhong et al. proposed a method to remove noise from images using the D-GAN. The architecture of the generator in the D-GAN has a convolutional block and eight dense blocks. Each block comprises a convolutional layer, batch normalization, and ReLU activation. Each layer, except the last layer in the network, is fed with each of the previous layers using skip connections. This method effectively reduces the vanishing gradient problem. The convolutional layer extracts low-level features, while the dense blocks extract the high-level features. The generator network is capable of learning the residual difference between the ground truth and the noisy image. The final 3×3 convolutional layer generates the output images. The discriminator network differentiates the fake and the ground truth image, making the final denoised visually appealing image. The model can handle different types of noise, but it cannot handle unknown real noises.

5.3. Face Aging and Facial Attribute Editing. Deep GAN-based methods have been proposed to alter facial attributes to anticipate a person's future look. Conditional GAN has been widely adopted to perform face aging [129]. D-GANs

are also incorporated in facial attribute editing, manipulating facial images' attributes to generate face with the desired attribute, retaining other facial images' details. The latent representation of the facial images is decoded to edit the facial attributes. GAN-based methods have been proposed for facial attribute editing, which changes only the desired attributes and preserves the other identities of the facial images, retaining the facial image's identity [130]. The work uses reconstruction learning to preserve the attribute details and *"only change what you want."* The authors applied attribute classification constraints to the generated image rather than imposing constraints on the latent representation to warrant the desired attributes' correct change. The facial attributes are manipulated to change the facial image with and without a beard, black hair and brown hair, mouth open and mouth close, brown hair and blond hair, and young and old. Yujun Shen et al. [131] interpreted the latent codes of trained models such as StyleGAN and progressive GAN and encoded various semantics in the interpreted latent space. Given a synthesized face, different face attributes such as pose, age, and expression are edited without having to retrain the D-GAN model. Table 7 shows the comparison of handwritten digits generated by D-GAN variants.

6. Open Problems and Future Opportunities for Computational Visual Perception-Driven Image Analysis

This paper discussed the development of computational visual perception with D-GAN and D-CNN. The advantages and disadvantages of various architectures of D-GAN models are discussed. Future research with D-GAN can be performed on model collapse, nonconvergence, and training difficulties. Also, various other shortcomings of the CNN and their solution are reviewed. Table 8 lists the challenges and open problems for computational visual perception-driven image analysis.

- (i) Handwriting recognition mainly relies on the language model that we furnish to the system and the character modeling quality. This research can be performed to obtain a better handwriting recognition system that is faster and more accurate.
- (ii) Semantic mapping is promising in autonomous vehicles, but state-of-the-art methods still need improvement to produce reliable tools. To do this, better 3D geometry must be included in mapping to achieve more accurate results in the semantic segmentation process. Moreover, map updating has to be done to ensure that maps are always coherent with reality.
- (iii) The biggest problem with calibration in webcam-based eye trackers is a variation in the head pose. Furthermore, this has to be handled without modifying the base components of the system. The future works can be directed towards a 3D model-based head tracking, gaze estimation calculated

geometrically, and accurate iris segmentation method.

- (iv) Lumen center detection is based on the geometry and appearance of the lumen. Future works can be directed towards lumen segmentation using the center point of the lumen computed previously as the seed and filling tracheal ring discontinuities to improve segmentation accuracy.
- (v) In query-by-string word, spotting the latent semantic model's performance improves when more samples are used to build the model. However, acquiring the transcription of handwritten documents can be tricky so that synthetic information can train the whole framework. Another problem that requires a solution is the vast possible parameter combinations of the bag-of-visual-words model. An adaptive framework can automatically generate spatial representation and codebook based on the model training phase's indexation errors.
- (vi) Coastal lagoons alternate between being open and closed to the ocean, resulting in intermittently closed and open lakes and lagoons. These are features by a berm, originated from sediments and sand deposited by tides, winds, and waves from the ocean. Moreover, this helps prevent ocean water from flow further into the lagoon, but rain can cause the lagoon to overflow. Hence, future research can explore the adoption of existing computer vision technology and techniques to monitor ICOLLs, including obtaining water level measurement and berm height and improving the decisions for when to open/close a lagoon entrance.
- (vii) Presently, scientists are working towards the development of a vaccine to control coronavirus disease. The clinicians and scientists are making all the attempts to prevent, treat, and control the spread of coronavirus. However, there is an urgent need to perform research into the use of SARS-Cov-2 in suitable animal models to examine the replication, transmission, and pathogenesis. The task remains in identifying the source of coronavirus, the immunological basis of the virus, the immune responses, and whether the mutation-prone positive-sense ssRNA virus of coronavirus will become endemic or alters into forms that are more lethal shortly.
- (viii) The medical imaging community utilizes transfer learning where large-scale annotated data are required. Here, the transfer learning model's performance can be improved by carefully selecting the source and target domain. Residual learning can make significant advancements if effectively utilized in medical image analysis tasks.
- (ix) With the potential use of image processing techniques in computer vision, disaster management can be improved. Satellite images of areas that are prone to landslide are taken during a specific

TABLE 7: Comparisons of images generated by D-GAN variants.

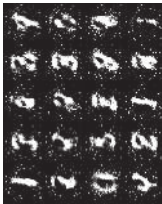
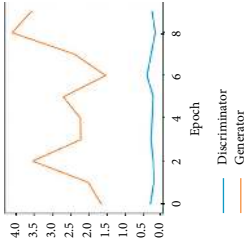

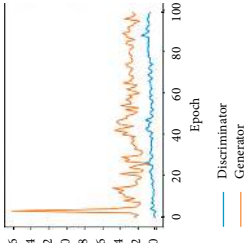

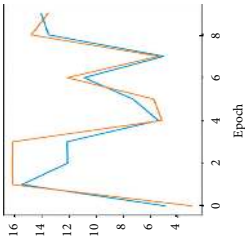

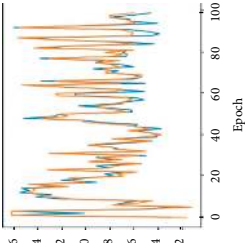
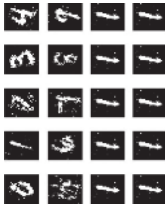
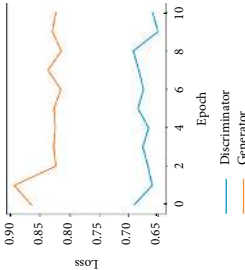
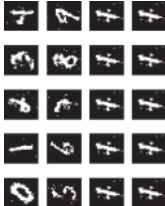
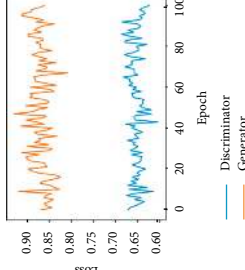
Type of D-GAN	Epoch 10	Training losses	Epoch 100	Training losses	Characteristics	Shortcomings
D-GAN					Generator-discriminator framework via a minimax game where samples are directly generated.	The model parameters oscillate, destabilize, and never converge.
DCGAN					It uses convolutional stride and transposed convolution for the downsampling and the upsampling.	Gradients disappear or explode.
CGAN					Conditional generation of images.	CGAN is not strictly unsupervised. Some labeling is required for it to work strictly.

TABLE 7: Continued.


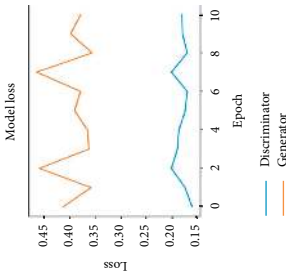

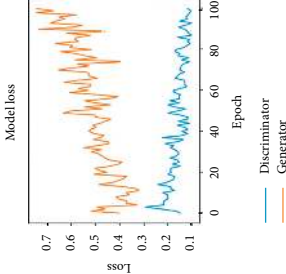

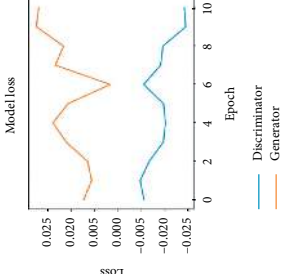

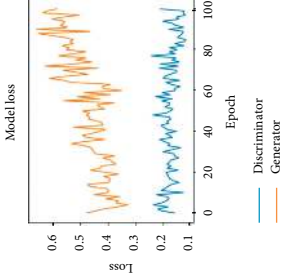
Type of D-GAN	Epoch 10	Training losses	Epoch 100	Training losses	Characteristics	Shortcomings
LSGAN					It creates high-quality images compared to the GAN. It is more stable during training.	Additional computational cost.
					Stability of learning and overcomes the mode collapse problem.	The difficulty in the WGAN is to enforce the Lipschitz constraint.

TABLE 8: Challenges and open problems for computational visual perception-driven image analysis.

Research direction	Key references	Dataset	Challenges and open problems	Future research
Water level measurement	[132, 133]	GNSS dataset	Image-based in situ water level measurement faces several challenges: image distortions, low visibility, and ambient noises.	The image can be changed from 2D to 3D, and the water level measurement can be done by utilizing advanced computer vision techniques.
Image processing	[134, 135]	Sequential image captured with the camera	Classical image processing techniques are based on a controlled environment. Environmental changes require image processing techniques such as custom filters, thresholding, and limited site coverage.	The model is generalized using deep learning techniques in a dynamic environment.
Flood detection	[136]	Manually collected dataset	Lack of open-source data to train computer vision algorithms.	The data can be collected and opened to train the proposed model.
UAV image processing	[137, 138]	Datasets of AOGCM simulations	Proposed solutions have limited generalizability.	Advanced convolutional neural networks can be used. Instead of using image processing techniques, the model's generalizability can be assessed using real-world data for the testing phase.

interval. Image registration techniques can be used to register these to each other, and the movement can be recorded, which can help predict future landslides. This technique will be beneficial when satellite images of areas that are prone to landslides are available. Deviations in the positions of hills are estimated to predict the number of future landslides.

- (x) The research related to underwater imaging is evolving to expose undiscovered species underwater. It is not possible to identify all the underwater species by continuously visualizing the recorded videos underwater. Therefore, an automated system to classify or detect the species underwater is required.

7. Conclusions

This paper summarized a comprehensive survey of deep CNN and deep GAN, their basic principles, GAN variants, and their computational visual perception applications. A comparison between biological and computer vision is made to better understand the background of computer vision and understand the architecture of neural networks. This survey presents an extensive comparison of current and existing surveys. This survey extensively surveys the applications of D-GAN and D-CNN. The building blocks of the CNN, recent advancements of the CNN, activation functions, D-GAN evolution, and its recent advancement are discussed in detail. The pitfalls of deep learning and its solutions are discussed briefly. Both state-of-the-art and classical applications of deep GAN and deep CNN are evaluated. Developments in computational visual perception or computer vision with D-GAN and D-CNN in recent years are reviewed. D-GAN is proved to solve the problem of insufficient data and improve the quality of image generation. Experimental results are discussed to explore the ability of

variants of D-GAN models. The advantages, disadvantages, and network architectures of different D-GAN models are discussed. Besides, D-CNN and D-GAN applications that have achieved remarkable achievements in various computer vision applications are discussed. Furthermore, D-GAN has a wide variety of applications combined with other deep learning algorithms.

This article extensively surveyed the current opportunities and future challenges in all the emerging domains. This article discussed the current opportunities in many emerging domains such as handwriting recognition, semantic mapping, webcam-based eye trackers, lumen center detection, query-by-string word, intermittently closed and open lakes and lagoons, and landslides. Future research with the D-GAN has to be directed towards model collapse, nonconvergence, and training difficulties. Though there are vast improvements such as weight regularization, weight pruning, and Nash equilibrium, future research in this area is still mandatory. D-GAN in the security domain has more research scope as adversarial attacks on neural networks have become very common. Slight perturbation in samples may lead to the wrong classification by neural networks. Furthermore, to outdo adversarial attacks, it is necessary to make D-GANs more robust to adversarial attacks. Though D-GAN is put forward as unsupervised learning, adding labels to the data will significantly improve the D-GAN's data quality. Modifying the D-GAN in this way is one of the future research directions.

Conflicts of Interest

The authors declare no conflicts of interest.

Authors' Contributions

N. A. R, P. D. R. V, and C-Y. C. performed the conceptualization and supervised the data. C-Y. C. carried out the funding acquisition. N. A. R, P. D. R. V, K. S, and C-Y.

C. investigated the data and performed the methodology. C.-Y. C. and K. S. carried out the project administration and validated the data. N. A. R, P. D. R. V, K. S., and U. T wrote, reviewed, and edited the manuscript. All authors read and agreed to the published version of the manuscript.

Acknowledgments

This work was supported by the Ministry of Science and Technology, Taiwan (Grant no. MOST 103-2221-E-224-016-MY3). This research was partially funded by the “Intelligent Recognition Industry Service Research Center” from the Featured Areas Research Center Program within the framework of the Higher Education Sprout Project by the Ministry of Education (MOE) in Taiwan, and the APC was funded by the aforementioned project.

References

- [1] L. Roberts, *Machine Perception of Three-Dimensional Solids*, IEEE, New York, NY, USA, 1963.
- [2] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.
- [3] A. Graves, “Connectionist temporal classification,” in *Studies in Computational Intelligence*, pp. 61–93, Springer, Berlin, Germany, 2012.
- [4] A. G. Ivakhnenko, “Polynomial theory of complex systems,” *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 1, no. 4, pp. 364–378, 1971.
- [5] R. H. R. Hahnloser, R. Sarpeshkar, M. A. Mahowald, R. J. Douglas, and H. S. Seung, “Digital selection and analogue amplification coexist in a cortex-inspired silicon circuit,” *Nature*, vol. 405, no. 6789, pp. 947–951, 2000.
- [6] T.-Y. Lin, M. Maire, S. Belongie et al., “Microsoft COCO: common objects in context,” in *Proceedings of the Computer Vision—ECCV 2014*, D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds., pp. 740–755, Springer International Publishing, Cham, Switzerland, 2014.
- [7] I. Goodfellow, J. Pouget-Abadie, M. Mirza et al., “Generative adversarial nets,” in *Advances in Neural Information Processing Systems 27*, Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, Eds., pp. 2672–2680, Curran Associates, Inc., Red Hook, NY, USA, 2014.
- [8] A. Radford, L. Metz, and S. Chintala, “Unsupervised representation learning with deep convolutional generative adversarial networks,” 2016.
- [9] C. Ledig, L. Theis, F. Huszar et al., “Photo-realistic single image super-resolution using a generative adversarial network,” 2017, <https://arxiv.org/abs/1609.04802>.
- [10] E. L. Denton, S. Chintala, A. Szlam, and R. Fergus, “Deep generative image models using a Laplacian pyramid of adversarial networks,” in *Advances in Neural Information Processing Systems*, C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, Eds., vol. 28, pp. 1486–1494, Curran Associates, Inc., Red Hook, NY, USA, 2015.
- [11] H. Zhang, I. Goodfellow, D. Metaxas, and A. Odena, “Self-attention generative adversarial networks,” in *Proceedings of the International Conference on Machine Learning*, PMLR, pp. 7354–7363, Long Beach, CA, USA, June 2019.
- [12] D. J. Im, C. D. Kim, H. Jiang, and R. Memisevic, “Generating images with recurrent adversarial networks,” 2016, <https://arxiv.org/abs/1602.05110>.
- [13] M. Xudong, L. Qing, H. Xie, and Y. K. Raymond, “Least squares generative adversarial networks,” in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 2242–2250, Venice, Italy, October 2017.
- [14] H.-Y. Li, W.-M. Dong, and B.-G. Hu, “Facial image attributes transformation via conditional recycle generative adversarial networks,” *Journal of Computer Science and Technology*, vol. 33, no. 3, pp. 511–521, 2018.
- [15] A. Odena, C. Olah, and J. Shlens, “Conditional image synthesis with auxiliary classifier GANs,” 2016, <https://arxiv.org/abs/1610.09585>.
- [16] M. Mirza and S. Osindero, “Conditional generative adversarial nets,” 2014, <https://arxiv.org/abs/1411.1784>.
- [17] X. Chen, Y. Duan, R. Houthoofd, J. Schulman, I. Sutskever, and P. Abbeel, “Infogan: interpretable representation learning by information maximizing generative adversarial nets,” in *Proceedings of the Advances in Neural Information Processing Systems*, pp. 2172–2180, Barcelona, Spain, December 2016.
- [18] D. Rui, G. Guo, X. Yan, B. Chen, Z. Liu, and X. He, “BiGAN: collaborative filtering with bidirectional generative adversarial networks,” in *Proceedings of the 2020 SIAM International Conference on Data Mining*, pp. 82–90, Cincinnati, OH, USA, May 2020.
- [19] A. Makhzani, J. Shlens, N. Jaitly, I. Goodfellow, and B. Frey, “Adversarial autoencoders,” 2016, <https://arxiv.org/abs/1511.05644>.
- [20] Y. Mroueh, T. Sercu, and V. Goel, “Mcgan: mean and covariance feature matching gan,” in *Proceedings of the International Conference on Machine Learning*, pp. 2527–2535, Sydney, Australia, August 2017.
- [21] A. Martin, C. Soumith, and B. Léon, “Wasserstein generative adversarial networks,” *Proceedings of the 34th International Conference on Machine Learning*, PMLR, vol. 70, pp. 214–223, 2017.
- [22] D. Gerónimo, J. Serrat, A. M. López, and R. Baldrich, “Traffic sign recognition for computer vision project-based learning,” *IEEE Transactions on Education*, vol. 56, no. 3, pp. 364–371, 2013.
- [23] J. Thevenot, M. B. López, and A. Hadid, “A survey on computer vision for assistive medical diagnosis from faces,” *IEEE Journal of Biomedical and Health Informatics*, vol. 22, no. 5, pp. 1497–1511, 2018.
- [24] H. Hassan, A. K. Bashir, M. Ahmad et al., “Real-time image dehazing by superpixels segmentation and guidance filter,” *Journal of Real-Time Image Processing*, 2020.
- [25] M. Ahmad, A. K. Bashir, and A. M. Khan, “Metric similarity regularizer to enhance pixel similarity performance for hyperspectral unmixing,” *Optik*, vol. 140, pp. 86–95, 2017.
- [26] H. Hassan, A. K. Bashir, R. Abbasi, W. Ahmad, and B. Luo, “Single image autofocus estimation by modified Gaussian function,” *Transactions on Emerging Telecommunications Technologies*, Wiley, Hoboken, NJ, USA, 2019.
- [27] G. Danuser, “Computer vision in cell biology,” *Cell*, vol. 147, no. 5, pp. 973–978, 2011.
- [28] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, Las Vegas, NV, USA, June 2016.
- [29] X. Mao, Q. Li, H. Xie, R. Y. K. Lau, Z. Wang, and S. Paul Smolley, “Least squares generative adversarial networks,” in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2794–2802, Venice, Italy, October 2017.

- [30] Z. Li, W. Yang, S. Peng, and F. Liu, "A survey of convolutional neural networks: analysis, applications, and prospects," 2020, <https://arxiv.org/abs/2004.02806>.
- [31] X. Wu, K. Xu, and P. Hall, "A survey of image synthesis and editing with generative adversarial networks," *Tsinghua Science and Technology*, vol. 22, no. 6, pp. 660–674, 2017.
- [32] X. Yi, E. Walia, and P. Babyn, "Generative adversarial network in medical imaging: a review," *Medical Image Analysis*, vol. 58, p. 101552, 2019.
- [33] K. Zhu, X. Liu, and H. Yang, "A survey of generative adversarial networks," in *Proceedings of the 2018 chinese automation congress (CAC)*, pp. 2768–2773, Xi'an, China, November–December 2018.
- [34] Z. Pan, W. Yu, X. Yi, A. Khan, F. Yuan, and Y. Zheng, "Recent progress on generative adversarial networks (GANs): a survey," *IEEE Access*, vol. 7, pp. 36322–36333, 2019.
- [35] A. Khan, A. Sohail, U. Zahoor, and A. S. Qureshi, "A survey of the recent architectures of deep convolutional neural networks," *Artificial Intelligence Review*, vol. 53, no. 8, pp. 5455–5516, 2020.
- [36] L. Liu, W. Ouyang, X. Wang et al., "Deep learning for generic object detection: a survey," 2019, <https://arxiv.org/abs/1809.02165>.
- [37] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *Journal of Big Data*, vol. 6, p. 60, 2019.
- [38] Y. Chen, Y. Zhao, W. Jia, L. Cao, and X. Liu, "Adversarial-learning-based image-to-image transformation: a survey," *Neurocomputing*, vol. 411, pp. 468–486, 2020.
- [39] S. Kalra and A. Leekha, "Survey of convolutional neural networks for image captioning," *Journal of Information and Optimization Sciences*, vol. 41, no. 1, pp. 239–260, 2020.
- [40] N. V. K. Medathati, H. Neumann, G. S. Masson, and P. Kornprobst, "Bio-inspired computer vision: towards a synergistic approach of artificial and biological vision," *Computer Vision and Image Understanding*, vol. 150, pp. 1–30, 2016.
- [41] J. Sanchez-Riera, K. Srinivasan, K.-L. Hua, W.-H. Cheng, M. A. Hossain, and M. F. Alhamid, "Robust RGB-D hand tracking using deep learning priors," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 28, no. 9, pp. 2289–2301, 2018.
- [42] A. Kumar, K. Srinivasan, W.-H. Cheng, and A. Y. Zomaya, "Hybrid context enriched deep learning model for fine-grained sentiment analysis in textual and visual semiotic modality social data," *Information Processing & Management*, vol. 57, no. 1, Article ID 102141, 2020.
- [43] L. Deng and D. Yu, "Deep learning: methods and applications," *Foundations and Trends in Signal Processing*, vol. 7, no. 3–4, pp. 197–387, 2014.
- [44] W. Zhou, Y. Niu, and G. Zhang, "Sensitivity-oriented layer-wise acceleration and compression for convolutional neural network," *IEEE Access*, vol. 7, pp. 38264–38272, 2019.
- [45] S. Lin and G. C. Runger, "GCRNN: group-constrained convolutional recurrent neural network," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 10, pp. 4709–4718, 2018.
- [46] D. Yu and L. Deng, "Deep learning and its applications to signal and information processing [exploratory DSP]," *IEEE Signal Processing Magazine*, vol. 28, no. 1, pp. 145–154, 2011.
- [47] "Deep Learning | Nature," October 2020, <https://www.nature.com/articles/nature14539>.
- [48] W. Hu, Y. Huang, L. Wei, F. Zhang, and H. Li, "Deep convolutional neural networks for hyperspectral image classification," *Journal of Sensors*, vol. 2015, Article ID 258619, 12 pages, 2015.
- [49] J. Li, X. Long, S. Hu, Y. Hu, Q. Gu, and D. Xu, "A novel hardware-oriented ultra-high-speed object detection algorithm based on convolutional neural network," *Journal of Real-Time Image Processing*, vol. 17, pp. 1703–1714, 2020.
- [50] S. Hijazi, R. Kumar, and C. Rowen, *Using Convolutional Neural Networks for Image Recognition*, pp. 1–12, Cadence Design Systems, Inc., San Jose, CA, USA, 2015.
- [51] F. H. C. Tivive and A. Bouzerdoum, "Efficient training algorithms for a class of shunting inhibitory convolutional neural networks," *IEEE Transactions on Neural Networks*, vol. 16, no. 3, pp. 541–556, 2005.
- [52] Z.-Q. Zhao, P. Zheng, S.-T. Xu, and X. Wu, "Object detection with deep learning: a review," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 11, pp. 3212–3232, 2019.
- [53] J. M. Ponce, A. Aquino, and J. M. Andújar, "Olive-fruit variety classification by means of image processing and convolutional neural networks," *IEEE Access*, vol. 7, pp. 147629–147641, 2019.
- [54] W. Ouyang, B. Xu, J. Hou, and X. Yuan, "Fabric defect detection using activation layer embedded convolutional neural network," *IEEE Access*, vol. 7, pp. 70130–70140, 2019.
- [55] Z. Zhang, H. Wang, F. Xu, and Y.-Q. Jin, "Complex-valued convolutional neural network and its application in polarimetric SAR image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 55, no. 12, pp. 7177–7188, 2017.
- [56] X. Zhang, X. Gao, W. Lu, and L. He, "A gated peripheral-foveal convolutional neural network for unified image aesthetic prediction," *IEEE Transactions on Multimedia*, vol. 21, no. 11, pp. 2815–2826, 2019.
- [57] Q. Xuan, Z. Chen, Y. Liu, H. Huang, G. Bao, and D. Zhang, "Multiview generative adversarial network and its application in pearl classification," *IEEE Transactions on Industrial Electronics*, vol. 66, pp. 8244–8252, 2018.
- [58] A. Kumar, J. Kim, D. Lyndon, M. Fulham, and D. Feng, "An ensemble of fine-tuned convolutional neural networks for medical image classification," *IEEE Journal of Biomedical and Health Informatics*, vol. 21, pp. 31–40, 2016.
- [59] I. Ramírez, A. Cuesta-Infante, J. J. Pantrigo et al., "Convolutional neural networks for computer vision-based detection and recognition of dumpsters," *Neural Computing and Applications*, vol. 32, pp. 13203–13211, 2018.
- [60] K. Aukkapinyo, S. Sawangwong, P. Pooyoi, and W. Kusakunniran, "Localization and classification of rice-grain images using region proposals-based convolutional neural network," *International Journal of Automation and Computing*, vol. 17, pp. 233–246, 2019.
- [61] M. Z. Alom, M. Hasan, C. Yakopcic, T. M. Taha, and V. K. Asari, "Improved inception-residual convolutional neural network for object recognition," *Neural Computing and Applications*, vol. 32, no. 1, pp. 279–293, 2020.
- [62] J. Moon, H. Kim, and B. Lee, "View-point invariant 3d classification for mobile robots using a convolutional neural network," *International Journal of Control, Automation and Systems*, vol. 16, no. 6, pp. 2888–2895, 2018.
- [63] T. Wang, Y. Chen, M. Qiao, and H. Snoussi, "A fast and robust convolutional neural network-based defect detection model in product quality control," *The International Journal*

- of *Advanced Manufacturing Technology*, vol. 94, no. 9–12, pp. 3465–3471, 2018.
- [64] J. Teng, D. Zhang, D.-J. Lee, and Y. Chou, “Recognition of Chinese food using convolutional neural network,” *Multimedia Tools and Applications*, vol. 78, no. 9, pp. 11155–11172, 2019.
 - [65] D. V. Pakulich, S. A. Yakimov, and S. A. Alyamkin, “Age recognition from facial images using convolutional neural networks,” *Optoelectronics, Instrumentation and Data Processing*, vol. 55, no. 3, pp. 255–262, 2019.
 - [66] N. Puhalanthi and D.-T. Lin, “Effective multiple person recognition in random video sequences using a convolutional neural network,” *Multimedia Tools and Applications*, vol. 79, no. 15–16, pp. 11125–11141, 2020.
 - [67] A. Madakannu and A. Selvaraj DIGI-Net, “DIGI-Net: a deep convolutional neural network for multi-format digit recognition,” *Neural Computing and Applications*, vol. 32, no. 15, pp. 11373–11383, 2020.
 - [68] F. Özyurt, T. Tuncer, E. Avci, M. Koç, and İ. Serhatlioğlu, “A novel liver image classification method using perceptual hash-based convolutional neural network,” *Arabian Journal for Science and Engineering*, vol. 44, no. 4, pp. 3173–3182, 2019.
 - [69] K. Chauhan and S. Ram, “Image classification with deep learning and comparison between different convolutional neural network structures using tensorflow and keras,” *International Journal of Advanced Research and Development*, vol. 5, pp. 533–538, 2018.
 - [70] Z. Qiumei, T. Dan, and W. Fenghua, “Improved convolutional neural network based on fast exponentially linear unit activation function,” *IEEE Access*, vol. 7, pp. 151359–151367, 2019.
 - [71] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in *Proceedings of the Advances in Neural Information Processing Systems*, pp. 1097–1105, Lake Tahoe, NV, USA, December 2012.
 - [72] A. F. Agarap, “Deep learning using rectified linear units (relu),” 2018, <https://arxiv.org/abs/1803.08375>.
 - [73] W. Zhao and S. Du, “Spectral-spatial feature extraction for hyperspectral image classification: a dimension reduction and deep learning approach,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 8, pp. 4544–4554, 2016.
 - [74] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, “Improving neural networks by preventing co-adaptation of feature detectors,” 2012, <https://arxiv.org/abs/1207.0580>.
 - [75] J. Ba and B. Frey, “Adaptive dropout for training deep neural networks,” in *Advances in Neural Information Processing Systems* 26, C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, Eds., pp. 3084–3092, Curran Associates, Inc., Red Hook, NY, USA, 2013.
 - [76] X. Yu, X. Wu, C. Luo, and P. Ren, “Deep learning in remote sensing scene classification: a data augmentation enhanced convolutional neural network framework,” *GIScience & Remote Sensing*, vol. 54, no. 5, pp. 741–758, 2017.
 - [77] L. Taylor and G. Nitschke, “Improving deep learning with generic data augmentation,” in *Proceedings of the 2018 IEEE Symposium Series on Computational Intelligence (SSCI)*, pp. 1542–1547, Bangalore, India, November 2018.
 - [78] T. Tran, T. Pham, G. Carneiro, L. Palmer, and I. Reid, “A bayesian data augmentation approach for learning deep models,” 2017, <https://arxiv.org/abs/1710.10564>.
 - [79] J. Salamon and J. P. Bello, “Deep convolutional neural networks and data augmentation for environmental sound classification,” *IEEE Signal Processing Letters*, vol. 24, no. 3, pp. 279–283, 2017.
 - [80] L. Perez and J. Wang, “The effectiveness of data augmentation in image classification using deep learning,” 2017, <https://arxiv.org/abs/1712.04621>.
 - [81] E. D. Cubuk, B. Zoph, D. Mane, V. Vasudevan, and Q. V. Le, “AutoAugment: learning augmentation policies from data,” 2019, <https://arxiv.org/abs/1805.09501>.
 - [82] M. A. Alsheikh, D. Niyato, S. Lin, H.-p. Tan, and Z. Han, “Mobile big data analytics using deep learning and Apache spark,” *IEEE Network*, vol. 30, no. 3, pp. 22–29, 2016.
 - [83] R. A. Bauder, T. M. Khoshgoftar, and T. Hasanin, “Data sampling approaches with severely imbalanced big data for medicare fraud detection,” in *Proceedings of the 2018 IEEE 30th International Conference on Tools with Artificial Intelligence (ICTAI)*, pp. 137–142, Volos, Greece, November 2018.
 - [84] D. Hendrycks, M. Mazeika, D. Wilson, and K. Gimpel, “Using trusted data to train deep networks on labels corrupted by severe noise,” in *Proceedings of the Advances in Neural Information Processing Systems*, pp. 10456–10465, Montréal, Canada, December 2018.
 - [85] C. You, W. Cong, M. W. Vannier et al., “CT super-resolution GAN constrained by the identical, residual, and cycle learning ensemble (GAN-CIRCLE),” *IEEE Transactions on Medical Imaging*, vol. 39, no. 1, pp. 188–203, 2020.
 - [86] Y. Zhan, D. Hu, Y. Wang, and X. Yu, “Semisupervised hyperspectral image classification based on generative adversarial networks,” *IEEE Geoscience and Remote Sensing Letters*, vol. 15, no. 2, pp. 212–216, 2018.
 - [87] D. Hu, L. Wang, W. Jiang, S. Zheng, and B. Li, “A novel image steganography method via deep convolutional generative adversarial networks,” *IEEE Access*, vol. 6, pp. 38303–38314, 2018.
 - [88] C. Wang, C. Xu, C. Wang, and D. Tao, “Perceptual adversarial networks for image-to-image transformation,” *IEEE Transactions on Image Processing*, vol. 27, no. 8, pp. 4066–4079, 2018.
 - [89] N. Li, Z. Zheng, S. Zhang, Z. Yu, H. Zheng, and B. Zheng, “The synthesis of unpaired underwater images using a multistyle generative adversarial network,” *IEEE Access*, vol. 6, pp. 54241–54257, 2018.
 - [90] A. Lucas, A. K. Katsaggelos, S. Lopez-Tapia, and R. Molina, “Generative adversarial networks and perceptual losses for video super-resolution,” in *Proceedings of the 2018 25th IEEE International Conference on Image Processing (ICIP)*, pp. 51–55, Athens, Greece, October 2018.
 - [91] A. Radford, L. Metz, and S. Chintala, “Unsupervised representation learning with deep convolutional generative adversarial networks,” 2016, <https://arxiv.org/abs/1511.06434>.
 - [92] A. B. L. Larsen, S. K. Sønderby, H. Larochelle, and O. Winther, “Autoencoding beyond pixels using a learned similarity metric,” in *Proceedings of the International Conference on Machine Learning; PMLR*, pp. 1558–1566, Boston, MA, USA, October 2016.
 - [93] P. Isola, J. Y. Zhu, T. Zhou, and A. A. Efros, “Image-to-image translation with conditional adversarial networks,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1125–1134, Honolulu, HI, USA, July 2017.

- [94] Yu.V. Vizil'ter, O. V. Vygolov, D. V. Komarov, and M. A. Lebedev, "Fusion of images of different spectra based on generative adversarial networks," *Journal of Computer and Systems Sciences International*, vol. 58, pp. 441–453, 2019.
- [95] H. Zhang, B. S. Riggan, S. Hu, N. J. Short, and V. M. Patel, "Synthesis of high-quality visible faces from polarimetric thermal faces using generative adversarial networks," *International Journal of Computer Vision*, vol. 127, no. 6–7, pp. 845–862, 2019.
- [96] J. Shen, N. Liu, H. Sun, and H. Zhou, "Vehicle detection in aerial images based on lightweight deep convolutional network and generative adversarial network," *IEEE Access*, vol. 7, pp. 148119–148130, 2019.
- [97] P. Xiang, L. Wang, F. Wu, J. Cheng, and M. Zhou, "Single-image de-raining with feature-supervised generative adversarial network," *IEEE Signal Processing Letters*, vol. 26, no. 5, pp. 650–654, 2019.
- [98] X. Wang, Z. Cao, R. Wang, Z. Liu, and X. Zhu, "Improving human pose estimation with self-attention generative adversarial networks," *IEEE Access*, vol. 7, pp. 119668–119680, 2019.
- [99] W. Wang, A. Wang, Q. Ai, C. Liu, and J. Liu, "AAGAN: enhanced single image dehazing with attention-to-attention generative adversarial network," *IEEE Access*, vol. 7, pp. 173485–173498, 2019.
- [100] W. Fang, Y. Ding, F. Zhang, and J. Sheng, "Gesture recognition based on CNN and DCGAN for calculation and text output," *IEEE Access*, vol. 7, pp. 28230–28237, 2019.
- [101] A. T. Arslan and E. Seke, "Face depth estimation with conditional generative adversarial networks," *IEEE Access*, vol. 7, pp. 23222–23231, 2019.
- [102] C. Xu, Y. Cui, Y. Zhang, P. Gao, and J. Xu, "Image enhancement algorithm based on generative adversarial network in combination of improved game adversarial loss mechanism," *Multimedia Tools and Applications*, vol. 79, no. 13–14, pp. 9435–9450, 2020.
- [103] Z. Yu, Q. Xiang, J. Meng, C. Kou, Q. Ren, and Y. Lu, "Retinal image synthesis from multiple-landmarks input with generative adversarial networks," *BioMedical Engineering OnLine*, vol. 18, no. 1, p. 62, 2019.
- [104] M. Ahmad, M. Mazzara, R. A. Raza et al., "Multiclass non-randomized spectral-spatial active learning for hyperspectral image classification," *Applied Sciences*, vol. 10, no. 14, p. 4739, 2020.
- [105] M. Ahmad, A. M. Khan, M. Mazzara, S. Distefano, M. Ali, and M. S. Sarfraz, "A fast and compact 3-D CNN for hyperspectral image classification," 2020, <https://arxiv.org/pdf/2004.14152.pdf>.
- [106] M. Ahmad, S. Shabbir, D. Oliva, M. Mazzara, and S. Distefano, "Spatial-prior generalized fuzziness extreme learning machine autoencoder-based active learning for hyperspectral image classification," *Optik*, vol. 206, Article ID 163712, 2020.
- [107] M. Ahmad, "Ground truth labeling and samples selection for hyperspectral image classification," *Optik*, vol. 230, Article ID 166267, 2021.
- [108] M. Ahmad, M. A. Alqarni, A. M. Khan, R. Hussain, M. Mazzara, and S. Distefano, "Segmented and non-segmented stacked denoising autoencoder for hyperspectral band reduction," *Optik*, vol. 180, pp. 370–378, 2019.
- [109] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, Las Vegas, Nevada, USA, June–July 2016.
- [110] F. Siddique, S. Sakib, and M. A. B. Siddique, "Recognition of handwritten digit using convolutional neural network in Python with tensorflow and comparison of performance for various hidden layers," in *Proceedings of the 2019 5th International Conference on Advances in Electrical Engineering (ICAEE)*, pp. 541–546, Dhaka, Bangladesh, September 2019.
- [111] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. D. Salakhutdinov, "A simple way to prevent neural networks from overfitting," *Journal of Machine Learning Research*, vol. 15, pp. 1929–1958, 2014.
- [112] T.-H. Chan, K. Jia, S. Gao, J. Lu, Z. Zeng, and Y. Ma, "PCANet: a simple deep learning baseline for image classification," *IEEE Transactions on Image Processing*, vol. 24, no. 12, pp. 5017–5032, 2015.
- [113] A. Saxena, "Image classification using convolutional neural networks," *Journal of the Gujarat Research Society*, vol. 21, pp. 72–80, 2019.
- [114] D. V. A. Bharadi, A. I. Mukadam, M. N. Panchbhai, and N. N. Rode, "Image classification using deep learning," *International Journal of Engineering Research and Technology*, vol. 6, 2017.
- [115] I. Amerini, C.-T. Li, and R. Caldelli, "Social network identification through image classification with CNN," *IEEE Access*, vol. 7, pp. 35264–35273, 2019.
- [116] R. Socher, B. Huval, B. Bhat, C. D. Manning, and A. Y. Ng, "Convolutional-recursive deep learning for 3D object classification," in *Proceedings of the 25th International Conference on Neural Information Processing Systems—Volume 1*, pp. 656–664, Curran Associates Inc., Red Hook, NY, USA, December 2012.
- [117] S. ul Hussain and B. Triggs, "Feature sets and dimensionality reduction for visual object detection," in *Proceedings of the British Machine Vision Conference 2010*, pp. 112.1–112.10, British Machine Vision Association, Aberystwyth, UK, August–September 2010.
- [118] H. Zhuang, K.-S. Low, and W.-Y. Yau, "Multichannel pulse-coupled-neural-network-based color image segmentation for object detection," *IEEE Transactions on Industrial Electronics*, vol. 59, no. 8, pp. 3299–3308, 2012.
- [119] S. Tu, J. Pang, H. Liu et al., "Passion fruit detection and counting based on multiple scale faster R-CNN using RGB-D images," *Precision Agriculture*, vol. 21, no. 5, pp. 1072–1091, 2020.
- [120] D. Augusto Borges Oliveira and M. Palhares Viana, "Fast CNN-based document layout analysis," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pp. 1173–1180, Venice, Italy, October 2017.
- [121] M. Rhanoui, M. Mikram, S. Yousfi, and S. Barzali, "A CNN-BiLSTM model for document-level sentiment analysis," *Machine Learning and Knowledge Extraction*, vol. 1, no. 3, pp. 832–847, 2019.
- [122] K. Noda, Y. Yamaguchi, K. Nakadai, H. G. Okuno, and T. Ogata, "Audio-visual speech recognition using deep learning," *Applied Intelligence*, vol. 42, no. 4, pp. 722–737, 2015.
- [123] F. Ghorban, N. Milani, D. Schugk et al., "Conditional multichannel generative adversarial networks with an application to traffic signs representation learning," *Progress in Artificial Intelligence*, vol. 8, no. 1, pp. 73–82, 2019.
- [124] T. Zhang, P. Jiang, and M. Zhang, "Inter-frame video image generation based on spatial continuity generative adversarial networks," *Signal, Image and Video Processing*, vol. 13, no. 8, pp. 1487–1494, 2019.

- [125] Z. Wang, G. Healy, A. F. Smeaton, and T. E. Ward, "Use of neural signals to evaluate the quality of generative adversarial network performance in facial image generation," *Cognitive Computation*, vol. 12, no. 1, pp. 13–24, 2020.
- [126] C. Tian, Y. Xu, and W. Zuo, "Image denoising using deep CNN with batch renormalization," *Neural Networks*, vol. 121, pp. 461–473, 2020.
- [127] G. Zhao, J. Liu, J. Jiang, and W. Wang, "A deep cascade of neural networks for image inpainting, deblurring and denoising," *Multimedia Tools and Applications*, vol. 77, no. 22, pp. 29589–29604, 2018.
- [128] J. Zhang, H. Dang, H. K. Lee, and E.-C. Chang, "Flipped-adversarial auto encoders," 2018, <https://arxiv.org/pdf/1802.04504.pdf>.
- [129] H. T. Shen, J. Song, J. Zhang, L. Gao, and X. Liu, "Dual conditional GANs for face aging and rejuvenation," in *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence IJCAI-18*, pp. 899–905, Stockholm, Sweden, July 2018.
- [130] Z. He, W. Zuo, M. Kan, S. Shan, and X. Chen, "AttGAN: facial attribute editing by only changing what you want," *IEEE Transactions on Image Processing*, vol. 28, no. 11, pp. 5464–5478, 2019.
- [131] Y. Shen, J. Gu, X. Tang, and B. Zhou, "Interpreting the latent space of GANs for semantic face editing," 2020, <https://arxiv.org/abs/1907.10786>.
- [132] Z. Zhang, Y. Zhou, H. Liu, and H. Gao, "In-situ water level measurement using NIR-imaging video camera," *Flow Measurement and Instrumentation*, vol. 67, pp. 95–106, 2019.
- [133] E. Ridolfi and P. Manciola, "Water level measurements from drones: a pilot case study at a dam site," *Water*, vol. 10, p. 297, 2018.
- [134] S. Udomsiri and M. Iwahashi, "Design of FIR filter for water level detection," *World Academy of Science, Engineering and Technology*, vol. 48, pp. 47–52, 2008.
- [135] S. Park, N. Lee, Y. Han, and H. Hahn, "The water level detection algorithm using the accumulated histogram with band pass filter," *World Academy of Science, Engineering and Technology*, vol. 56, pp. 193–197, 2009.
- [136] W. Kang, Y. Xiang, F. Wang, L. Wan, and H. You, "Flood detection in gaofen-3 SAR images via fully convolutional networks," *Sensors*, vol. 18, no. 9, p. 2915, 2018.
- [137] Y. Hirabayashi, R. Mahendran, S. Koiralet al., "Global flood risk under climate change," *Nature Climate Change*, vol. 3, no. 9, pp. 816–821, 2013.
- [138] A. L. Sumalan, D. A. N. Popescu, and L. Ichim, "Flooded areas detection based on LBP from UAV images," in *Proceedings of the 2017 21st International Conference on System Theory, Control and Computing (ICSTCC)*, pp. 186–191, Sinaia, Romania, October 2015.