# 1

# Role of Graph Convolutional Neural Networks (GCNN) in Computer Vision Applications

**A. Malini[1*], Vandana Sharma[2], J. Felicia Lilian[3], Rajesh Kumar Dhanaraj[4], Sharangapriyan S.[3] and Shrinivas S.[3]**

*[1]School of Computer Science and Engineering, Vellore Institute of Technology, Chennai, Tamil Nadu, India*
*[2]Christ University, Bengaluru, India*
*[3]Department of Computer Science and Business Systems, Thiagarajar College of Engineering, Tamil Nadu, India*
*[4]Symbiosis Institute of Computer Studies and Research (SICSR), Symbiosis International (Deemed University), Pune, India*



## *Abstract*

Graph Convolution Neural Networks (GCNNs) are an important concept in advancing computer vision by transforming the understanding and modeling of graph-structured data. They have a unique capability to capture intricate relations along with the visual content that goes beyond the traditional and usual convolutional neural networks, it also empowers computers to observe and interpret the complex interconnection between the elements in images, which enhances the depth and nuance of visual dentata analysis. As a revolutionary study in computer vision, GCNNs are poised to transform various industries by unleashing new frontiers in the visual information domain's analysis and interpretation. Their multifaceted applications promise to reshape the landscape of computer vision.

***Keywords***: Graph convolutional neural networks (GCNNs), computer vision, graph-structured data, visual data analysis, visual information analysis

**Corresponding author*: a.malinicse@gmail.com; a.malini@vit.ac.in

Malini Alagarsamy, Rajesh Kumar Dhanaraj, J. Felicia Lilian, Vandana Sharma and Gheorghita Ghinea (eds.) Graph Convolutional Neural Networks for Computer Vision, (1–20) © 2026 Scrivener Publishing LLC

## 1.1 Introduction

In our rapidly changing universe, computer vision's sphere is undergoing an amazingly unusual growth transforming several areas and enhancing human life's standards significantly. This paradigm change is supported by the huge expansion of Profound Learning and Nervous Networks in the computer vision realm.

Unlike humans, who reflexively gather footage and images, computers are carefully designed and trained to understand data enabling sophisticated analysis and rapid decision making. This innovative path has extensive implications for several fields, including the progress of autonomous vehicles, advanced medical testing, and social media recommendation system enhancement.

In today's world, GCNN has many applications. In social media analysis, GCNNs can analyze using nodes and edges as social networks by modeling users and their interactions, improving suggestion systems, class detection, and studies on influence propagation and predicting atomic behavior, molecular properties, and interactions by molecule models as graphical structures are crucial for genomics research in bioinformatics. For autonomous vehicles, GCNNs can model the environment as a graph, where nodes represent objects, and edges represent spatial relationships allowing for better scene understanding and decision making.

Additionally, GCNNs are employed in 3D computer vision tasks, such as shape analysis and 3D reconstruction, where 3D objects are represented as graphs of connected vertices. In conclusion, Graph Convolutional Neural Networks are revolutionizing the way complex data structures are analyzed and processed expanding the horizons of what is achievable in computer vision and beyond. Their ability to leverage the power of graphs opens up new possibilities for innovation and application across various domains further enhancing the impact of computer vision technologies on our everyday lives and enhancing human life's standards significantly.

## 1.2 Understanding Convolutional Neural Network in Computer Vision

Space Invariant Artificial Neural Networks have been the spearhead of reform in computer vision, which is the foundation of many refinements in visual analysis and interpretations. It has a profound effect on object recognition and scene understanding. CNN is often referred to as ConvNets representing a confined type of artificial neural network originated to operate mesh-like data such as images and videos. They have a very long history heading back to the 1980s, with seminal efforts by researchers like Yann LeCun, Geoffrey Hinton, and Yoshua Bengio. However, CNN came to action in the 2010s largely due to advances in computational power, the availability of large labeled datasets, and improved training algorithms.

## 1.3 Core Components of CNN

1. **Convolutional Layers:** These layers apply filters (also termed as kernels) to the feed-in data to pull features. These filters are accustomed to extract the computing dot in every position, and it captures the local patterns available in the given input data.
2. **Layers for Pooling:** Pooling layers lessen the geometrical extents of the characteristic map produced by convolutional layers. General pooling functions consist of average pooling and max pooling, which retain the most important data while reducing the computational complexity.
3. **Connected Layers (Fully Connected):** The layers that are fully connected are considered standard neural networks that connect all axons from one layer to another typically leading to the final output. They serve as classifiers or regressors in the environs of computer vision assignments.

### 1.3.1 Hierarchy Feature Learning

One of CNNs' key advantages is its capacity for hierarchical learning from unfiltered pixel data. Higher layers combine these properties into more complicated representations as the data traverses through the network allowing the network to eventually recognize objects, patterns, and abstract concepts. Lower layers collect basic elements like edges and corners.

## 1.4 Extending CNNs to Handle Graph-Structured Data

For their remarkable performance in grid-like data, CNN has been utilized for a long time, especially in computer vision applications involving photos and videos. Graph-structured data, which has its own special set of difficulties and potential, is included in the broad landscape of data kinds, which goes far beyond grids.

### 1.4.1 Graphs—A Universal Data Structure

For data with complicated linkages, graphs provide a flexible representation. Each node can carry important information, and they are made up of vertices and the edges that connect them. Graphs are used in many different sectors, such as social media networks, biology, transportation, recommendation engines, and more. To derive useful insights from graphs, it is essential to interpret and analyze the patterns inside graph-structured data.

### 1.4.2 Challenges in Processing Graphs with CNN

1. **Irregular Data Structures:** Graphs do not have homogeneous grids like ordinary grids do, which makes it challenging to extract the crucial data from the input.
2. **Variable Neighborhoods:** Traditional CNNs struggle to handle the variable-sized neighborhoods that nodes in a graph can have.

### 1.4.3 Graph Convolutional Neural Networks (GCNNs): Bridging the Gap

To effectively extract features and patterns from structured data, GCNNs expand the concepts of CNNs to the world of graphs. By using graph convolutions that operate on nodes and their neighboring nodes, they provide an advantage by capturing the contextual information required for various applications.

### 1.4.4 Architectural Components of Graph Convolutional Layers

At the heart of GCNN is the representation of the input graph. This typically involves encoding both the graph structure and node attributes. Common representations include the following as given in Figure 1.1.

1. **Adjacency Matrix:** The adjacency matrix, often referred to as A, is a key data structure that encodes the connectivity of nodes in a graph. In a plot with N nodes, the adjacency matrix is an $N \times N$ matrix, where each element (i,j) indicates whether there is an interrelation between node i, j. Typically, it is a binary matrix where the value of 1 signifies the presence of an edge, while a value of 0 indicates no connection.

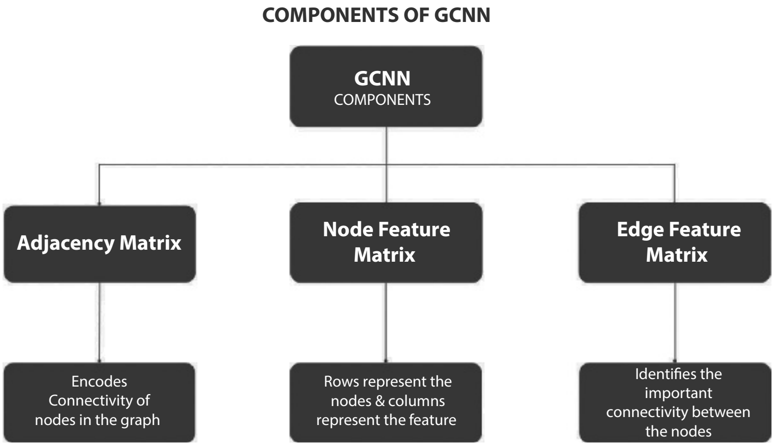


**Figure 1.1** Components of GCNN.

2. **Node Feature Matrix:** This is a matrix in which each row indicates a node, and each column indicates an attribute. The node feature matrix is a structured data representation where each row conforms to a node in a graph, and each node represents the specific feature or attribute associated with that node. In essence, it acts as a table where nodes are entities, and the attributes are the properties of characteristics of those nodes.
3. **Edge Feature Matrix:** If the edge features are available, they can be encoded in a separate matrix. Edge feature matrices are responsible for enhanced connectivity between the nodes, which will give important information that is present in between the nodes. These added facts can reproduce various particulars of the relationships between nodes.

### 1.4.5 Graph Convolutional Layers: Adaptation of Convolutional Layers

The convolutional neural networks (CNNs) are famous for their effectiveness on structured data like time signals and images. They rely on the core operation of convolution with learned filters to extract meaningful features from input data. However, intrinsic limitations come with CNNs specifically as suitable only for regular structured data sources. This drawback restricts their use in cases where data have irregular or non-uniform patterns. In response to this limitation, Graph Signal Processing has emerged as a powerful replacement paradigm. GSP provides an entirely new way of dealing with signals that are inherently represented by graphs or networks. Such signals inherently encapsulate irregular and complex structures, which are not easily amenable to traditional convolutional operations. To bridge this gap, GSP introduces the pioneering concept of graph convolutional filters. These filters lay the foundation for developing Convolutional Graph Neural Networks (GNNs), which is explicitly designed to work on graph-structured data. The adaptability of these GNNs comes from two different interpretations they have about graph convolutions: diffusion- and aggregation-type conversions. As such, it presents a flexible framework within which GNNs can be designed for specific tasks [1].

## 1.5 Application of GCNN in Computer Vision

GCNN, which operates on graph-structured data, plays a pivotal part in the field of computer vision. Tasks that involve complex relations can be easily solved using it, especially for analyzing and understanding structures and pointing layers within the images. Some of its application are explained below.

### 1.5.1 Semantic Segmentation

Semantic segmentation is a process of dividing an image into classes and labeling them by training a model that has a main goal to locate and make a boundary around the objects. It is based on initializing every pixel in an image using a specific patch. Semantic segmentation involves multiple steps starting from image processing to visualization. It has lots of applications such as medical image analysis, understanding the scene, robotic perception, video security, augmented reality, and image condensing. Semantic segmentation uses the range of evolving efforts in instance and semantic segmentation, including pixel labeling using convolutional networks, encoder–decoder architectures, pyramid-based approaches and multiscale, recurrent networks, and non-human primate visual attention models [1].

### 1.5.2 Object Detection and Localization with GCNN

Object detection and localization using GCNNs involves representing an image as a graph, with nodes and edges denoting relationships. In computer vision, the relationship between object detection and graphs lies in how we represent and analyze images for the purpose of identifying objects within them. In deep learning, YOLO (You Only Look Once) processes an image by dividing it into a grille and predicts media boxes and class probabilities directly within each grille cell. YOLO's architecture allows for efficient real-time object detection by simultaneously processing the entire image in a single forward pass. It balances speed and accuracy making it suitable for various applications like autonomous driving and object tracking [2].

### 1.5.3 Graph-Based Image Classification

Image classification is an elementary assignment in computer vision, where the end result is to classify an image based on its visual content. For example, we can train an image classification algorithm to answer if a car is present in an image or not. While detecting an object is minor for humans, tough image classification is still a challenge in computer vision applications. It is a proposal to construct images as graphs and to try machine learning algorithms that operate on graphs to the emerging representations. Notably, the present approaches for representing images as graphs are dissimilar, and we endow on deep-rooted neural network architectures developed for graph-structured data to deal with image classification tasks. The proposed models leverage properties innate in images such as stationarity of statistics and locality of pixel dependencies. The proposed models in image classification tasks are compared with standard GCNN architectures.

### 1.5.4 Point Cloud Analysis and 3D Object Recognition

Deep learning methods are becoming more advanced for 3D object identification and categorization in self-driving applications for autonomous vehicles. A self-driving car system is reliant on the task. The availability of 3D scanners, such as LiDAR, in sensor technology produces more precise data. Newer versions of how LiDAR data works have recently undergone some significant development. High-definition LiDAR laser scanners capture the majority of 3D point clouds [3]. Additionally, by adopting a non-supervised ground and segmentation process to reduce scene complexity, it decreases the need for manually labeled training data [4].

### 1.5.5 Video Understanding and Action Recognition through Graphs

Videos address the issue of detecting human behavior. It is a graph-based structure to learn the way interactions occur between peoples and objects. It analyzes issues with spatiotemporal action perception and mundane action detection. Given a clip, the approach is to localize each actor and classify his actions. Both the actor and object has a space and time relationship between them. A representation of an actor and object exists as nodes and edges in a graph. Since there are significant correlations between actors and objects, the scene is represented as a fully connected graph, where every node is related to every other node and serves as the input for our neural network. The graph-based description is further subdivided into an attention layer and an adjacency matrix for segmentation to handle the consecutive clips acquired from the action detection of human behavior [5].

## 1.6 Enhancing Performance and Interpretability with GCNN

**Interpretability:** It is a long standing limitation of deep neural networks for classification results. It is the most straightforward approach for generating a sensitivity map over input data to discover the gradient map within the substructures. The gradient maps are noisy, and smoothing these maps by enhancing the performance of the interpretability of GCNN is necessary.

Enhancing the performance and interpretability of GCNNs is crucial for their broader adoption in applications that require both accuracy and transparency, such as social media network analysis, recommendation engines, and biology. Balancing performance and interpretability is an ongoing challenge, as more complex models might achieve better performance at the cost of interpretability. It is essential to choose the right techniques and strategies that align with your specific goals and constraints in any given application.

### 1.6.1 Handling Graph Irregularities and Noisy Data

Thresholding is fundamental for improving the performance of higher-level computer vision tasks including segmentation, classification, and object recognition. When the data are visualized as graph vertices, the convolution operation is converted into a generalized graph convolution. Denoising has always been a significant study area for developing deep learning systems when the graph is irregular. When the information state is structured as a graph and the data are represented as nodes in the graph, prolonging the GCNNs signals with a systematic form, such as images and videos, can produce a graph-structured signal. The graphs are actively computed in feature space utilizing GCDN architecture and graph convolutional layers [6].

### 1.6.2 Transfer Learning and Pre-Trained GCNN Models

Transfer learning often starts with pre-trained models, which are typically huge neural networks learned on massive datasets. These models have been trained to recognize and extract useful data features. The idea is to apply what you learned from the source task to improve your performance on the target task.

#### *1.6.2.1 Transfer Learning Model for Waste Classification*

This is a new combination classification model that processes the ImageNet database and achieves high classification accuracy using pretrained CNN models. The transfer learning model, based on each pretrained model, is built as a candidate classifier, and the best output from the CNN pretrained model's candidate classifiers is chosen as the final classification outcome [7].

#### *1.6.2.2 Pre-Trained GCNN Models to Identify Finger Vein*

CNN is the technology that is redefining the field of computer vision research, and it continues to advance accurate image classification in addition to being essential for performing tasks involving feature extraction and recognition, including image retrieval, object detection, and semantic segmentation. Because of the present era's technological advancements, which include improved computer power and GPU acceleration, CNN may be used robustly and effectively [8]. AlexNet, a model of two fully connected layers, five convolutional layers, and a softmax output layer, is based on deep CNN. This network is coherent in contrast to other, more rudimentary networks that use Rectified Linear Units, dropout, and overlap pooling.

### 1.6.3 Addressing Overfitting and Generalization Challenges

The amount of good solutions appears to be incompatible with the observed overfitting solutions when training deep neural networks. After training a deep neural network model on labeled data, it is typically tested on unlabeled data for evaluation. The model's capacity to generalize implies that it is capable of performing well on test data. Overfitting occurs when a model performs well on training data but fails on test data. More particular, the model learns the noise patterns contained in the training data resulting in a substantial difference between training and test error due to overfitting. Underfitting occurs when the model fails to catch patterns in both training and test data [9].

## 1.7 Future Directions and Emerging Trends

Graph Convolutional Neural Networks (GCNNs) have made enormous advances in the last decade and remain an important area of research. Among the future directions and developing developments for GCNNs are the following:

**Improving Efficiency:** Deep learning has always been a problem, especially when it comes to GCNNs. This is because scientists are now in search of ways to make GCNNs more efficient in terms of computation and memory. These techniques can be categorized as sparsity, quantization, and model compression.
**AutoML for GCNNs:** With the invention of GCNNs, automated machine learning tools (AutoML) have become more popular. AutoML is helping, automatically configuring, and optimizing GCNN models for specific applications.
**Graph Neural Architecture Search (GNAS):** GNAS involves finding the best architectures for specific tasks in AutoML. It saves time spent on architecture design and resources used thereof.
**Graph Adversarial Attacks and Defenses:** Since there are security applications of GCNNs in fraud detection, current investigations concentrate on graph aspects related to adversarial attacks and defenses. Among them is developing robust models against adversarial attacks targeted at defined systems using Graph Convolutional Networks (GCNs).

### 1.7.1 Advances in Graph Neural Network Architectures

#### *1.7.1.1 Graph Attention Network*

The Graph Attention Network introduced a mechanism for learning node-specific attention weights enabling GNNs to focus on different neighbors when aggregating information.

A GAT is a type of neural network design that works with graph-structured data. It generates methods based on graph convolutions using self-attention layers. Deep networks can be built by stacking GAT layers on top of each other [10]. The model can learn more abstract representations of nodes by allowing them to attend over their neighborhood's features multiple times by stacking layers. Through this stacking, the model is able to capture complicated linkages and hierarchies in the network.

#### *1.7.1.2 GNNExplainer: Generating Explanations for Graph Neural Networks*

GNN is a significant technique for machine learning on graphs. It integrates node feature information with graph structure by recursively propagating neural messages along the edges of the input graphs. GNN explainers are comprehensible explanations for GNN predictions. The use of this explainer increases trust in the GCNN model, transparency, and allows us to understand the network in it [11]. Given a trained GNN model, GNNEXPLAINER attempts to uncover the underlying elements driving these predictions. We have distinct kinds of predictions as follows: single-instance explanation and multi-instance explanation prediction.

##### 1.7.1.2.1 Single-Instance Explanation

For single-instance explanations, GNNEXPLAINER identifies a subgraph within the computation graph and selects a subset of node features that are most influential in determining a specific prediction. This process allows us to gain insight into the critical components affecting individual predictions.

##### 1.7.1.2.2 Multi-Instance Explanation

In scenarios where we need to explain a set of predictions, GNNEXPLAINER aggregates explanations from each prediction within the set. It then automatically summarizes these individual explanations into a prototype explanation providing a holistic view of the features and subgraphs that collectively influence the entire set of predictions [12].

### 1.7.2 Integration of Graph Structures with Traditional CNNs

The integration of graph structures with traditional CNNs has took to the development of Graph Convolutional Networks (GCNs) and other graph-based deep learning architectures. These models enable the processing of graph data while respecting the data's intrinsic structure making them applicable in various domains where graphs are a common representation.

#### *1.7.2.1 Locally Connected and Spectral Networks on Graphs*

- **Graph-Structured Data**

There are many real-world datasets that are typically represented as graphs, where the nodes represent entities and the edges that denote relationships or connections between these entities.

- **Spectral Graph Theory**

Spectral Graph Conjecture is one thing of arithmetic that queries the rudimentary peculiarities of diagrams utilizing linear arithmetic as an aperture. It exerts the links within the form of a diagram and its phantasmal attributes, especially the eigenvalues and eigen-drones of the diagram's proximity or Laplacian matrices [13]. The spectral measurements give valuable details about the connections between the nodes of the graph. It has clustering tendencies and community structures. Spectral Graph Theories have applications in analyzing networks, segmenting images, clustering data, and creating recommendation systems.

- **Deep Locally Connected Networks on Graphs**

DLCN on Graphs are a new-fashioned way to employ deep learning methods, especially Convolutional Neural Networks (CNNs), to graph-structured data. By contemplating graphs as squares, these networks authorize the modeling of non-grid facts giving a frame for applying convolutional processes to graph-structured datasets. Deep local connect networks, not like average CNNs, take a less stiff approach where every node may have an odd set of neighbors. This allows for learning of localized and non-Euclidean traits on graphs, which can be handy in various fields like social network scrutiny, molecular chemistry, and recommendation systems. These networks have unlocked fresh possibilities for tackling complex problems using non-grid, interconnected data by broadening deep learning activities to graph details.

### 1.7.3 Explainable AI in GCNNs for Computer Vision

Gradient maps have been used with GCNN to describe the plainability procedures in Explainable AI. Every graph is indicated as a node, where each node is represented as an attributed graph for classification purposes. In this graph, the node functions as a matrix, and the adjacency matrix encodes the node's connectedness. We acquire scalars over nodes when the explainable AI algorithm is applied to all samples. Using explainable AI techniques, the following metrics are computed from the provided samples: fidelity, contrastivity, sparsity, and heatmap [14].

## 1.8 Challenges and Open Research Questions

There are a number of urgent problems and unanswered research topics with graph convolutional neural networks (GCNNs). First, a basic problem remains: how to make GCNNs scalable on massive graphs with millions or billions of edges and nodes. The problem of adapting GCNNs to dynamic graphs, especially ones that change over time, and diverse data is complex. For practical applications, it is important to improve the explainability and generalization capacity of GCNNs, particularly for graphs with little labeled input.

Robust regularization strategies are necessary to prevent overfitting when there is small or noisy graph data. Enabling GCNNs for inductive learning, where they can generalize to unseen nodes or graphs, and exploring unsupervised learning approaches are open areas of investigation. Furthermore, addressing adversarial attacks and developing defenses is critical, especially in security-critical applications. Scalable architectural search, graph generative models, and the potential of quantum computing to accelerate graph-based computations are emerging research directions with significant challenges.

### 1.8.1 Scalability of GCNNs to Large Graphs

Graph Convolutional Neural Networks (GCNNs) face fundamental challenges in the field of graph-based deep learning: their scalability to huge networks. Countless numbers of nodes and edges can be found in large graphs, which presents severe memory and processing demands.

#### 1.8.1.1 Scalable Graph Convolutional Networks

For handling the plot-structured data, GCNN with a swift sectional spectral filter for directed graphs is used. Two graphing methods currently in use are the spatial method, which focuses on each node's neighborhood, and the spectral method for graphs based on Laplacian. The majority of current methods operate on undirected graphs, which results in information loss [15]. To avoid this, one can use graph Laplacian, commonly known as a swift directed acyclic graph convolutional network, which operates on directed graphs. They use spatial linear aggregation and operate entirely on directed graphs.

For semi-supervised node classification, a fast directed graph convolutional network model, known as FDGCN, is used. The operations for a directed graph are precise, and node categorization is developed into a convolutional model based on the operations [16].

### 1.8.2 Robustness of GCNNs Against Adversarial Attacks

In applications requiring graph-structured data, the robustness of Graph Convolutional Neural Networks (GCNNs) against adversarial attacks is essential. Adversarial attacks entail changing the graph's structure or node attributes in such a way that the GCNN's predictions are misled potentially resulting in security breaches, misinformation, or incorrect recommendations. The creation of robust training algorithms and protection mechanisms, as well as recognizing the vulnerability of different GCNN architectures to various attack tactics, are open research problems in this domain. The defense techniques used, such as adversarial training, resilient pooling algorithms, or outlier detection, are determined by the application and threat model.

#### 1.8.2.1 Understanding Adversarial Robustness of Symmetric Network

Furthermore, adversarial robustness—which is known to be vulnerable—is achieved by neural network models. Adversarial attacks are deployed on group-equivalent convolutional neural networks that are rotation equivariant.

### 1.8.3 Combining Graph-Based and Spatial Feature Representations

In deep learning, combining spatial and graph-based feature representations is a potent strategy, especially for jobs involving grid-based data, like pictures, and structured data, like graphs. By combining the best features of each representation, the model performs better overall.

#### 1.8.3.1 Analysis of Graph Convolutional Networks and Recent Datasets for Visual Question Answering

GCNN is used in non-structural contexts because of its interpretability and significant performance. Graph reasoning models are utilized in visual question answering to develop a response system for an image provided as input, while simultaneously identifying the image's semantic meaning. A GCNN interpretability model is used for this question-answering system.

### 1.8.4 Ethical Considerations and Fairness in GCNN Applications

Ethical considerations and fairness in Graph Convolutional Neural Network (GCNN) applications are crucial aspects of deploying these models responsibly [17]. While GCNNs offer powerful tools for data analysis and decision making, they can also raise ethical concerns and introduce biases. Here are key ethical considerations and fairness issues in GCNN applications:

- Data Bias
- Transparency
- Explainability
- Fairness
- Bias Mitigation

#### 1.8.4.1 Neutrality and Abstraction in Sociotechnical Systems

The GCNN program uses neutrality-aware learning algorithms to step in at various points in the decision-making process and ensure a just conclusion. This algorithm uses decision-making mechanisms and surrounds itself with traps to achieve fairness in sociotechnical phenomena [18].

Artificial Intelligence and ML algorithms are in charge of decision-making processes in the transportation, healthcare, and many other sectors. These days, decision algorithms are prone to unfairness-based applications [19]. This can be avoided by employing machine learning algorithms to detect and measure fairness in a way that is fully and precisely processed. Fairness bias boosting mechanisms are examined and applied to the decision-making scenarios.

## 1.9 Case Studies: Real-World Applications

In computer vision, GCNN is used in numerous real-world applications and case studies are tabulated and shown in Table 1.1. It has a significant impact on every situation that arises mainly in computer vision is shown in Figure 1.2. A few instances from the real world where it has had a significant influence is as follows [20]:

I. **Image Segmentation:** Images can be segmented using GCNN, which can recognize and label various objects in an image. Applications like robotics, self-driving cars, and medical image analysis can benefit from that.

II. **Object Detection:** Object detection in photos and videos can be accomplished with GCNNs—applications like traffic monitoring, security and surveillance, and retail analytics can benefit from this [21].

III. **Pose Estimation:** GCNNs are useful for estimating object poses in both videos and images. Applications like virtual reality, augmented reality, and human–computer vision can benefit from that.

**Table 1.1** Applications of GCNN in computer vision.

| Case study | Description | Potential benefits from GCNN |
|---|---|---|
| **Medical image analysis** | • In medical image analysis context, GCNNs are used to learn features from medical images that are relevant to disease diagnosis.<br>• A GCNN-based system developed by researchers at Stanford University has been shown to be more accurate than human pathologists in detecting cancer cells in breast cancer images | • Improvised accuracy<br>• Reduced subjectivity<br>• Increased efficiency<br>• Reduced costs |

(*Continued*)

**Table 1.1** Applications of GCNN in computer vision. (*Continued*)

| Case study | Description | Potential benefits from GCNN |
|---|---|---|
| **Self-driving cars** | • Self-driving cars are utilizing an array of sensors, various cameras, radar, and LiDAR gathering data surrounding them<br>• Within the self-driving car scenario, graph data may visualize the connections amid diverse objects on the highway | • Robustness<br>• Reduced latency<br>• Improvised detection and tracking system |
| **Robotics** | • One of the crucial challenges in robot grasp is that it is hard to model the complex interactions between a robot grip and an object<br>• Graph data can signify the relationships between the various parts of the robot's grip and an object to be grasped<br>• This detail can be useful to predict the result of a trail and to plan grasping tactics that are likely successful | • Reduced planning time<br>• Improvised grasping accuracy and success rate<br>• Increased robustness |
| **Environmental monitoring** | • Spotting and tracking environmental hazards such as wildfires, oil spills, and illegal logging<br>• Researchers at the University of California, Berkeley, have developed a GCNN-based system that can detect wildfires with high definiteness<br>• Scientists at the Massachusetts Institute of Technology have used GCNNs to develop a system that can monitor water caliber instances | • Enhanced surveillance<br>• High accurate prediction |

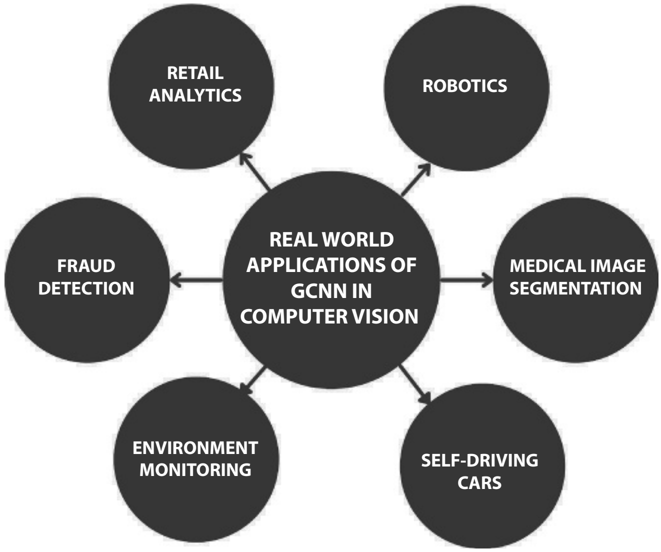


**Figure 1.2** Real-world applications of GCNN in CV.

## 1.10 Conclusion

Graph Convolutional Neural Networks (GCNNs) are potent tools for computer vision tasks. GCNNs are capable of acquiring knowledge about complex features from graph-structured data, which makes them requisite for tasks such as object detection and image segmentation. GCNNs are still under development, but they have the chance to remodel the way that we perceive and interact with the world around us by improving the accuracy, efficiency, and robustness of computer vision, such as developing more intelligent and capable solutions and enhancing the reliability of anything that is related to graphical structure. There are a number of exciting areas of future research in GCNN-based computer vision such as developing GCNN-based systems that are more interpretable. This will help us to understand how GCNNs make predictions and identify potential biases in GCNN models. Finally, it is important to develop CNN-based systems that are robust to noise and uncertainty. This is important for real-world applications, where the data may be noisy or incomplete.

## References

1. Minaee, S., Boykov, Y., Porikli, F., Plaza, A., Kehtarnavaz, N., Terzopoulos, D., Image segmentation using deep learning: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.*, 44, 7, 3523–3542, 2021.
2. Dhillon, A. and Verma, G.K., Convolutional neural network: a review of models, methodologies and applications to object detection. *Prog. Artif. Intell.*, 9, 2, 85–112, 2020.
3. Basheer, S., Singh, K.U., Sharma, V., Bhatia, S., Pande, N., Kumar, A., A robust NIfTI image authentication framework to ensure reliable and safe diagnosis. *PeerJ Comput. Sci.*, 9, e1323, 2023.
4. Nikolentzos, G., Thomas, M., Rivera, A.R., Vazirgiannis, M., Image Classification Using Graph-Based Representations and Graph Neural Networks, in: *Complex Networks & Their Applications IX: Volume 2, Proceedings of the Ninth International Conference on Complex Networks and Their Applications Complex Networks 2020*, Springer International Publishing, pp. 142–153, 2021.
5. Babahajiani, P., Fan, L., Gabbouj, M., Object recognition in 3D point cloud of urban street scene, in: *Computer Vision-ACCV 2014 Workshops: Singapore*, Singapore, November 1–2, 2014, Springer International Publishing, pp. Revised Selected Papers, Part I 12 pp. 177–190, 2015.
6. Tomei, M., Baraldi, L., Calderara, S., Bronzin, S., Cucchiara, R., Video action detection by learning graph-based spatio-temporal interactions. *Comput. Vision Image Understanding*, 206, 103187, 2021.
7. Chen, Y., Huang, W., Zhang, D., Chen, W., An open-source Matlab code package for improved rank-reduction 3D seismic data denoising and reconstruction. *Comput. Geosci.*, 95, 59–66, 2016.
8. Zheng, J., Xu, M., Cai, M., Wang, Z., Yang, M., Modeling group behavior to study innovation diffusion based on cognition and network: an analysis for garbage classification system in Shanghai, China. *Int. J. Environ. Res. Public Health*, 16, 18, 3349, 2019.
9. Fairuz, S., Habaebi, M.H., Elsheikh, E.M.A., Pre-trained based CNN model to identify finger vein. *Bull. Electr. Eng. Inf.*, 8, 3, 855–862, 2019.
10. Salman, S. and Liu, X., *Overfitting mechanism and avoidance in deep neural networks*, 2019, *arXiv preprint arXiv:1901.06566.*
11. Veličković, P., *et al.*, *Graph attention networks*, 2017, *arXiv preprint arXiv:1710.10903.*
12. Ying, Z., *et al.*, Gnnexplainer: Generating explanations for graph neural networks. *Adv. Neural Inf. Process. Syst.*, 32, 2019.
13. Lucic, A., *et al.*, Cf-gnnexplainer: Counterfactual explanations for graph neural networks. *International Conference on Artificial Intelligence and Statistics. PMLR*, 2022.
14. Bruna, J., *et al.*, *Spectral networks and locally connected networks on graphs*, 2013, *arXiv preprint arXiv:1312.6203.*

15. Pope, P.E., *et al.*, Explainability methods for graph convolutional neural networks. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019.
16. Li, C., Qin, X., Xu, X., Yang, D., Wei, G., Scalable Graph Convolutional Networks With Fast Localized Spectral Filter for Directed Graphs. *IEEE Access*, 8, 105634–105644, 2020.
17. Kamath, S. and Deshpande, A., *Understanding Adversarial Robustness of Symmetric Networks*, 2018.
18. Yusuf, A.A., Chong, F., Xianling, M., An analysis of graph convolutional networks and recent datasets for visual question answering. *Artif. Intell. Rev.*, 55, 6277–6300, 2022.
19. Angwin, J., Larson, J., Mattu, S., & Kirchner, L. Machine bias, in: *Ethics of data and analytics*, pp. 254-264, 2022. Auerbach Publications.
20. Smith, B., Khojandi, A., Vasudevan, R., Bias in Reinforcement Learning: A Review in Healthcare Applications. *ACM Comput. Surv.*, 56, 2, 1–17, 2023.
21. Borkar, K.K., Aljrees, T., Pandey, S.K., Kumar, A., Singh, M.K., Sinha, A., Singh, K.U., Sharma, V., Stability Analysis and Navigational Techniques of Wheeled Mobile Robot: A Review. *Processes*, 11, 3302, 2023. https://doi.org/10.3390/pr11123302.