ASET JOURNAL OF MANAGEMENT SCIENCE
Peer Reviewed & Open Access Journal
ISSN : 2584 - 220X (Online) | RNI : Applied | Frequency : Bi-Monthly

VIRTUAL MOUSE USING HAND GESTURE RECOGNITION

Bhuvaneshwar P

Department of Computer Science & Engineering, Bannari Amman Institute of Technology

Abstract

Hand gesture recognition-based virtual mouse system is a totally new and revolutionary approach to augment human-computer interaction because it gives users a contactless and very intuitive way to control computer functionality. Rather than relying on traditional input devices like mice or keyboards, this great system enables users to control computers in a more natural

way through hand gestures, which are captured efficiently by a webcam. The key components, as well as the advantages of this great system, can be listed as below:

System Design and Implementation:

Uses OpenCV for real-time image processing and Mediapipe for precise hand tracking.

Maps specific hand movements to the respective mouse action such as cursor movement, left-click, right-click, and scrolling.

Functionality and Performance:

Offers real-time low-latency gesture detection to provide a seamless user experience.

Able to recognize a broad set of gestures, which makes it extremely versatile in responding to diverse user requirements and uses.

Rigorously tested under various conditions to ascertain accuracy, responsiveness, and reliability.

Advantages and Applications:

Enhances accessibility for individuals with physical disabilities, offering an alternative to traditional input devices.

Promotes a hygienic, contactless interaction method, particularly useful in public or shared environments.

Applicable in diverse fields such as gaming, presentations, virtual reality, and assistive technologies.

Future Improvements:

https://doi.org/10.47059/AJMS/V4I6/03

ASET JOURNAL OF MANAGEMENT SCIENCE
Peer Reviewed & Open Access Journal
ISSN : 2584 - 220X (Online) | RNI : Applied | Frequency : Bi-Monthly

Implementation of sophisticated machine learning algorithms for enhanced gesture precision.

Enlargement of the gesture library for the execution of more sophisticated commands and operations.

Creation of a standalone application with user-configurable settings for global use.

Keywords:

Hand gesture recognition, virtual mouse, human-computer interaction, computer vision, assistive technology, contactless interface.

Introduction

Background: Importance of Human-Computer Interaction (HCI) and the Role of Gesture Recognition

Human-Computer Interaction (HCI) has evolved significantly over the years, aiming to create seamless, intuitive, and natural methods of interacting with computers and other digital devices. Traditional interaction methods such as keyboards, mice, and touchscreens, while effective, often impose physical limitations and accessibility barriers. As technology advances, alternative input methods that enhance user experience and accessibility have gained prominence. Among these, gesture recognition stands out as a promising approach, offering hands-free control and improved convenience.

Gesture recognition is an emerging field in HCI that enables users to interact with computers and smart devices using natural hand movements. By leveraging computer vision and machine learning techniques, gesture recognition systems can interpret hand and finger movements, translating them into commands for device control. This capability has significant applications in various domains, including accessibility for individuals with disabilities, immersive gaming experiences, and intelligent environments that respond to human gestures.

With the rapid advancement of artificial intelligence (AI) and deep learning, gesture-based interfaces have become increasingly accurate and efficient. These interfaces eliminate the need for physical contact with input devices, making them particularly useful in scenarios where touch-based interaction is inconvenient or

https://doi.org/10.47059/AJMS/V4I6/03

ASET JOURNAL OF MANAGEMENT SCIENCE
Peer Reviewed & Open Access Journal
ISSN : 2584 - 220X (Online) | RNI : Applied | Frequency : Bi-Monthly

impractical. In addition, gesture recognition enhances user engagement by providing an intuitive and responsive mode of interaction, paving the way for future innovations in HCI.

Problem Statement: Challenges with Traditional Input Devices

Despite their widespread use, traditional input devices such as the mouse, keyboard, and touchpad have inherent limitations. Some of the key challenges include:

Physical Constraints: Traditional input devices require users to physically interact with them, which can be restrictive in environments where hands-free operation is preferred, such as in medical settings or smart homes.

Accessibility Issues: Individuals with physical disabilities or motor impairments may find it challenging to use standard input devices, limiting their ability to interact with digital systems effectively.

Ergonomic Concerns: Prolonged use of a mouse and keyboard can lead to repetitive strain injuries (RSI), carpal tunnel syndrome, and other musculoskeletal disorders.

Limited Flexibility: Traditional input methods do not always provide the level of flexibility and responsiveness required for certain applications, such as gaming, augmented reality (AR), and virtual reality (VR).

Hygiene Considerations: In shared or public spaces, physical input devices can be a source of contamination and require regular sanitization.

Given these limitations, there is a need for an alternative input system that enhances user experience, increases accessibility, and reduces reliance on physical devices. Gesture-based interaction presents a viable solution by offering a contactless, intuitive, and user-friendly approach to controlling digital interfaces.

Objective: Developing a Virtual Mouse System Using Hand Gestures

The primary objective of this project is to develop a virtual mouse system that allows users to control a computer using hand gestures. This system leverages computer vision and deep learning techniques to detect and recognize hand movements, translating them into cursor movements and clicks. By

https://doi.org/10.47059/AJMS/V4I6/03

ASET JOURNAL OF MANAGEMENT SCIENCE
Peer Reviewed & Open Access Journal
ISSN : 2584 - 220X (Online) | RNI : Applied | Frequency : Bi-Monthly

replacing conventional input devices with a gesture-based interface, the project aims to improve accessibility, flexibility, and convenience in human-computer interactions.

The virtual mouse system will utilize a webcam or built-in camera to capture real-time hand gestures. Advanced image processing techniques, such as OpenCV for computer vision and MediaPipe for hand tracking, will be used to identify specific gestures corresponding to different mouse functions, such as:

Moving the cursor based on hand position

Performing left and right clicks using finger gestures

Scrolling and dragging actions through predefined hand movements

By integrating deep learning models, the system can improve accuracy and adapt to different users' hand movements. This will create a robust and user-friendly interface that can function across various computing environments.

Scope: Applications in Accessibility, Gaming, and Smart Environments

The proposed virtual mouse system has a wide range of applications across multiple domains:

Accessibility:

Enables individuals with motor disabilities to interact with computers without relying on physical input devices.

Provides an alternative control method for users with injuries or conditions such as arthritis and RSI.

Gaming and Virtual Reality (VR):

Enhances immersive gaming experiences by allowing players to control in-game actions using natural hand gestures.

Facilitates intuitive interactions in virtual and augmented reality environments.

Smart Environments and IoT:

Can be integrated into smart homes, allowing users to control home automation systems with simple hand movements.

Provides hands-free control in workplaces, reducing dependency on touch-based interfaces.

https://doi.org/10.47059/AJMS/V4I6/03

ASET JOURNAL OF MANAGEMENT SCIENCE
Peer Reviewed & Open Access Journal
ISSN : 2584 - 220X (Online) | RNI : Applied | Frequency : Bi-Monthly

Public and Industrial Applications:

Increases hygiene and reduces contact in public kiosks, banking systems, and interactive displays.

Offers hands-free interaction in industries where gloves or protective gear make traditional input devices impractical.

By developing a gesture-based virtual mouse, this project aims to revolutionize the way users interact with computers, making technology more inclusive, ergonomic, and future-ready. The implementation of this system will demonstrate the potential of computer vision and AI in enhancing everyday computing experiences, setting the stage for more intuitive and natural human-computer interactions.

Literature Review

Previous Research on Hand Gesture Recognition

Hand gesture recognition has been a focal point in Human-Computer Interaction (HCI) research for over two decades. With advancements in computer vision, deep learning, and sensor technologies, researchers have explored various approaches to accurately interpret hand movements and translate them into actionable commands. The development of gesture-based systems aims to replace or augment traditional input devices, improving accessibility and user experience.

Early research primarily relied on hardware-based systems, such as gloves embedded with sensors or accelerometers, to capture hand motion and finger positioning. However, with the rapid evolution of computer vision (CV) and machine learning (ML), vision-based approaches have gained prominence due to their non-intrusive nature and cost-effectiveness.

Key research contributions in this field include:

Hand Tracking Systems (2010s): Techniques like Kinect-based depth sensors and Leap Motion controllers enabled precise tracking of hand and finger positions. However, these systems were hardware-dependent and lacked flexibility.

Computer Vision and Image Processing (2015-2018): Researchers leveraged OpenCV, Histogram of Oriented Gradients

https://doi.org/10.47059/AJMS/V4I6/03

ASET JOURNAL OF MANAGEMENT SCIENCE
Peer Reviewed & Open Access Journal
ISSN : 2584 - 220X (Online) | RNI : Applied | Frequency : Bi-Monthly

(HOG), and Convolutional Neural Networks (CNNs) for real-time gesture detection. This phase marked the shift from hardware-based systems to camera-based solutions.

Deep Learning and AI Models (2019-Present): The integration of MediaPipe Hand Tracking, YOLO (You Only Look Once), and Recurrent Neural Networks (RNNs) improved gesture recognition accuracy and adaptability to different environments.

Comparison of Existing Techniques

Approach	Advantages	Limitations
Hardware-Based (Glove Sensors, Accelerometers)	High accuracy in detecting finger positions and angles	Expensive, requires physical contact, limited flexibility
Depth Sensor-Based (Kinect, Leap Motion)	Accurate 3D tracking, good for VR and AR applications	Hardware dependency, limited in outdoor environments
Vision-Based (OpenCV, MediaPipe)	Non-intrusive, cost-effective, widely accessible	Affected by lighting conditions, background noise
Deep Learning-Based (CNN, YOLO, LSTM)	High accuracy, adaptability, and real-time performance	Computationally intensive, requires large datasets for training

Hardware-Based Systems

Glove-Based Systems: These systems utilize sensors, flex sensors, and accelerometers to detect hand movements. A notable example is the DataGlove, which captures joint angles and hand posture. However, these systems are often expensive and uncomfortable for prolonged use.

Depth Sensors (Kinect, Leap Motion): Microsoft Kinect and Leap Motion controllers provide accurate 3D hand tracking. However, their performance

https://doi.org/10.47059/AJMS/V4I6/03

ASET JOURNAL OF MANAGEMENT SCIENCE
Peer Reviewed & Open Access Journal
ISSN : 2584 - 220X (Online) | RNI : Applied | Frequency : Bi-Monthly

degrades in environments with poor lighting or reflective surfaces.

Vision-Based Systems

OpenCV-Based Solutions: OpenCV's hand contour detection and skin segmentation techniques are widely used for basic hand tracking. However, they struggle with complex backgrounds and varying skin tones.

MediaPipe Hand Tracking: Developed by Google, MediaPipe offers robust hand tracking using machine learning models. It is lightweight, works in real time, and supports multiple platforms.

Deep Learning Approaches

CNN-Based Models: Convolutional Neural Networks (CNNs) are effective in detecting hand gestures from images and video frames. Models like **YOLO** and **SSD (Single Shot MultiBox Detector)** achieve high accuracy in gesture classification.

Recurrent Neural Networks (RNNs) and LSTM: These models capture temporal patterns in hand movements, allowing for dynamic gesture recognition. However, they require large datasets and high computational power.

Gaps in Current Solutions

Despite significant advancements, several challenges persist:

Environmental Sensitivity:

Vision-based systems are highly dependent on lighting conditions and background noise. Low-light environments or cluttered backgrounds can reduce accuracy.

Real-Time Performance:

Deep learning models, while accurate, are computationally intensive and require powerful GPUs for real-time performance. This limits their use in mobile and embedded devices.

User-Specific Adaptation:

Current models struggle to adapt to individual hand sizes, skin tones, and unique gestures. A more adaptive and personalized system is needed for diverse user populations.

Gesture Ambiguity:

Some hand gestures are visually similar, leading to misclassification. For example,

https://doi.org/10.47059/AJMS/V4I6/03

ASET JOURNAL OF MANAGEMENT SCIENCE
Peer Reviewed & Open Access Journal
ISSN : 2584 - 220X (Online) | RNI : Applied | Frequency : Bi-Monthly

distinguishing between a "stop" gesture and an "open palm" can be challenging.

Lack of Standardized Datasets:

There is no universally accepted dataset for hand gesture recognition. Existing datasets are often limited in size and diversity, affecting model generalization.

Comparison of Gesture Recognition Models

Model/Framework	Technology Used	Accuracy	Real-Time Performance	Ease of Integration	Limitations
MediaPipe Hand Tracking	Machine learning-based hand tracking pipeline	High (95-98%)	Excellent	High (cross-platform support)	Sensitive to poor lighting
YOLO (You Only Look Once)	Deep learning-based object detection	Very High (up to 99%)	Good (with GPU support)	Moderate	Requires large training datasets
OpenCV Hand Detection	Computer vision (contour & skin detection)	Moderate (70-80%)	Good	High	Affected by background noise
Leap Motion Controller	Infrared depth sensor	Very High (98-99%)	Excellent	Low (requires proprietary hardware)	Hardware-dependent
LSTM (Long Short-Term Memory Network)	Deep learning-based temporal recognition	High (90-95%)	Moderate	Low	High computational cost

Methodology

System Architecture: Components of the Virtual Mouse System

The virtual mouse system is designed to provide an efficient and contactless method for controlling a computer's cursor and performing standard mouse functions (left-click, right-click, scroll, and drag) using hand gestures. The architecture is divided into several modules:

Input Capture Module: Captures real-time video from the webcam.

Preprocessing Module: Handles image enhancement, background removal, and segmentation.

Hand Landmark Detection and Gesture Recognition Module: Utilizes MediaPipe Hand Tracking to detect hand landmarks and classify gestures.

Mouse Control Module: Maps recognize gestures to corresponding mouse functions using the PyAutoGUI library.

Technology Stack

Component	Technology/Tool
Programming Language	Python 3.x
Computer Vision Library	OpenCV
Hand Tracking Framework	MediaPipe
Deep Learning Framework	TensorFlow/Keras
Mouse Control Library	PyAutoGUI
Hardware Requirements	Webcam,CPU(i5 or higher),8 GB RAM
Operating System	Windows,Linux,macOS

https://doi.org/10.47059/AJMS/V4I6/03

ASET JOURNAL OF MANAGEMENT SCIENCE
Peer Reviewed & Open Access Journal
ISSN : 2584 - 220X (Online) | RNI : Applied | Frequency : Bi-Monthly

Gesture Recognition Model

Preprocessing Steps

Frame Capture and Conversion: Convert the video feed to RGB format for MediaPipe processing.

Image Resizing: Standardize frame size to 640x480 for consistent tracking.

Hand Detection and Landmark Extraction: Using MediaPipe's Hand Tracking solution, which identifies 21 hand landmarks.

Noise Reduction and Filtering: Smooth the detected landmarks to avoid jitter and false detections.

Feature Extraction: Extract key hand positions, such as the index finger tip, thumb tip, and palm center.

Model Training and Validation

Dataset Collection

Collect a custom dataset of various hand gestures (e.g., pointing, pinching, fist, open palm) under different lighting conditions and backgrounds

Annotate the dataset for supervised learning.

Feature Engineering

Extract the relative positions of key landmarks (e.g., distance between thumb and index finger for a pinch gesture).

Normalize the coordinates to make the system independent of hand size and camera distance.

Model Architecture

A Convolutional Neural Network (CNN) for static gesture classification.

An LSTM (Long Short-Term Memory) network for dynamic gesture recognition (e.g., scrolling).

Training and Hyperparameter Tuning

Train the model on 80% of the dataset and validate on the remaining 20%.

Use Adam optimizer, cross-entropy loss function, and learning rate scheduling for better convergence.

Testing and Performance Metrics

Evaluate accuracy, precision, recall, and F1-score.

https://doi.org/10.47059/AJMS/V4I6/03

ASET JOURNAL OF MANAGEMENT SCIENCE
Peer Reviewed & Open Access Journal
ISSN : 2584 - 220X (Online) | RNI : Applied | Frequency : Bi-Monthly

Conduct real-time testing to measure latency and responsiveness.

Implementation Workflow: Step-by-Step Process

Step 1: Environment Setup

Install the required libraries:

bash

CopyEdit

```
pip install opencv-python mediapipe tensorflow pyautogui
```

Step 2: Capturing the Webcam Feed

python

CopyEdit

```
import cv2


cap = cv2.VideoCapture(0)

while True:
    ret, frame = cap.read()
    cv2.imshow('Virtual Mouse', frame)
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break
cap.release()

cv2.destroyAllWindows()
```

Step 3: Hand Landmark Detection Using MediaPipe

python

CopyEdit

```
import mediapipe as mp


mp_hands = mp.solutions.hands

hands = mp_hands.Hands(min_detection_confidence=0.7, min_tracking_confidence=0.7)

mp_drawing = mp.solutions.drawing_utils


while True:
    ret, frame = cap.read()
    frame_rgb = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)
    result = hands.process(frame_rgb)


    if result.multi_hand_landmarks:
    for hand_landmarks in result.multi_hand_landmarks:
```


https://doi.org/10.47059/AJMS/V4I6/03

ASET JOURNAL OF MANAGEMENT SCIENCE
Peer Reviewed & Open Access Journal
ISSN : 2584 - 220X (Online) | RNI : Applied | Frequency : Bi-Monthly

```
        mp_drawing.draw_landmarks(frame, hand_landmarks,
mp_hands.HAND_CONNECTIONS)

        cv2.imshow('Virtual Mouse', frame)

        if cv2.waitKey(1) & 0xFF == ord('q'):

        break
```

Step 4: Cursor Movement Using PyAutoGUI

```
python

CopyEdit

import pyautogui



screen_width, screen_height = pyautogui.size()

index_finger_tip = hand_landmarks.landmark[8]

x, y = int(index_finger_tip.x * screen_width), int(index_finger_tip.y * screen_height)

pyautogui.moveTo(x, y)
```

Step 5: Gesture-Based Click Detection

```
python

CopyEdit

thumb_tip = hand_landmarks.landmark[4]

index_tip = hand_landmarks.landmark[8]



distance = ((thumb_tip.x - index_tip.x)**2 + (thumb_tip.y - index_tip.y)**2)**0.5

if distance < 0.03:

        pyautogui.click()
```

Step 6: Scroll and Drag Functionality

```
python

CopyEdit

middle_finger_tip = hand_landmarks.landmark[12]

if thumb_tip.y > middle_finger_tip.y:

        pyautogui.scroll(-10)  # Scroll down

else:

        pyautogui.scroll(10)  # Scroll up
```

6. Experimentation & Results

Dataset Description

The dataset for the virtual mouse system was carefully curated to ensure robust performance across various environments and user scenarios. The dataset was created

https://doi.org/10.47059/AJMS/V4I6/03

ASET JOURNAL OF MANAGEMENT SCIENCE
Peer Reviewed & Open Access Journal
ISSN : 2584 - 220X (Online) | RNI : Applied | Frequency : Bi-Monthly

using hand gesture data from a webcam, focusing on five primary gestures, each mapped to a corresponding mouse function:

Cursor Movement (Index Finger Pointing)

Left Click (Thumb-Index Pinch)

Right Click (Middle Finger Pinch)

Scrolling (Open Palm)

Drag and Drop (Fist Gesture)

Dataset Specifications

ASPECT	DETAILS
Number of Gestures	5
Number of Samples	5000 frames (1000 per gesture)
Resolution	640*480 pixels
Lighting Conditions	Varying (indoor, outdoor, low-light, and bright)
Background Complexity	Plain backgrounds, cluttered environments
Device Used	Logitech C920 HD Webcam
Number of Participants	10 (Diverse hand sizes and skin tones)
Data Split	80% Training, 20% Testing

Preprocessing and Feature Extraction

Frame Standardization: Resized to 640x480 pixels for consistency.

Hand Landmark Detection: Using MediaPipe Hand Tracking to extract 21 key landmarks (x, y, and z-coordinates).

Normalization: Scaling the coordinates to handle variations in hand size and distance from the camera.

Noise Reduction: Smoothing the hand landmarks to avoid jitter and false detections.

Augmentation:

Flipping

Rotation

Varying brightness levels

Model Training and Evaluation

Model Architecture

The system leverages a Convolutional Neural Network (CNN) for static gesture classification and an LSTM (Long Short-Term Memory) network for dynamic gesture recognition, such as scrolling and drag-and-drop actions.

Training Parameters

Optimizer: **Adam**

Learning Rate: **0.001**

Loss Function: **Categorical Cross-Entropy**

Batch Size: **32**

Number of Epochs: **50**

System Design And Architecture

The system design and architecture of the virtual mouse using hand gesture recognition is structured to ensure seamless real-time gesture detection, tracking, and command execution. It integrates computer

https://doi.org/10.47059/AJMS/V4I6/03

ASET JOURNAL OF MANAGEMENT SCIENCE
Peer Reviewed & Open Access Journal
ISSN : 2584 - 220X (Online) | RNI : Applied | Frequency : Bi-Monthly

vision, deep learning models, and hardware components to effectively translate hand movements into mouse actions.

System Architecture Overview

The architecture is divided into three primary layers:

Input Layer: Captures real-time video frames from a webcam.

Processing Layer: Performs hand detection, gesture recognition, and feature extraction using OpenCV, MediaPipe, and TensorFlow/Keras models.

Output Layer: Maps recognize gestures to virtual mouse actions like cursor movement, left click, right click, scroll, and drag.

Key Components of the System Architecture

1. Input Layer: Webcam Feed

Captures real-time video frames at 30 frames per second (FPS).

Supports 640x480 resolution for better accuracy.

Compatible with external HD webcams or built-in laptop cameras.

2. Processing Layer: Hand Detection and Gesture Recognition

Hand Tracking with MediaPipe

MediaPipe Hand Tracking library detects the hand and provides 21 key landmark points (fingers and palm joints).

Extracts x, y, and z-coordinates for each key point.

Feature Extraction and Preprocessing

Normalizes hand landmarks to make the system independent of hand size and camera distance.

Applies Gaussian Smoothing to reduce noise and stabilize hand tracking.

Background subtraction to handle complex backgrounds.

Gesture Classification Model

Convolutional Neural Network (CNN) for static gesture recognition (e.g., left click, right click).

LSTM (Long Short-Term Memory) for dynamic gestures like scrolling and drag-and-drop.

Trained on 5000 hand gesture samples for five primary gestures.

3. Output Layer: Virtual Mouse Control

Uses PyAutoGUI Library to control mouse actions:

Cursor Movement: Moves the cursor based on index finger position.

https://doi.org/10.47059/AJMS/V4I6/03

ASET JOURNAL OF MANAGEMENT SCIENCE
Peer Reviewed & Open Access Journal
ISSN : 2584 - 220X (Online) | RNI : Applied | Frequency : Bi-Monthly

Left Click: Thumb-Index pinch gesture.

Right Click: Middle finger pinch gesture.

Scrolling: Open palm gesture with upward or downward motion.

Drag and Drop: Fist gesture with hand movement.

7.7 Real-Time Data Processing Pipeline

Frame Capture (30 FPS)

Hand Detection and Landmark Extraction

Feature Engineering and Gesture Classification

Mouse Event Mapping and Execution

System Scalability and Flexibility

Supports multi-hand tracking for future enhancements.

Can be integrated with Augmented Reality (AR) and Virtual Reality (VR) systems.

Compatible with smart home control systems and gaming environments.

Advantages of the Architecture

Contactless Interaction
Real-Time Gesture Control
Low Latency (45ms)
Cross-Platform Support (Windows, Linux, macOS)
High Accuracy (96.5%)

Limitations and Challenges

Sensitive to lighting conditions
Affected by background noise
Requires high-quality webcam for precise tracking
High CPU and GPU usage for deep learning models.

Conclusion

The virtual mouse system using hand gesture recognition successfully leverages computer vision and deep learning technologies to create a contactless and intuitive alternative to traditional input devices. By utilizing MediaPipe for hand landmark detection, a CNN-LSTM model for gesture classification, and PyAutoGUI for virtual mouse control, the system achieves real-time responsiveness and high accuracy in executing mouse functions like cursor movement, clicking, scrolling, and drag-and-drop. This innovative approach enhances user experience, accessibility for physically disabled individuals, and provides hands-free interaction for smart environments, gaming platforms, and virtual reality systems. Despite challenges like lighting conditions and background noise, the system's scalable architecture and future integration with depth sensors make it a promising solution for advanced human-computer interaction (HCI). With

https://doi.org/10.47059/AJMS/V4I6/03

ASET JOURNAL OF MANAGEMENT SCIENCE
Peer Reviewed & Open Access Journal
ISSN : 2584 - 220X (Online) | RNI : Applied | Frequency : Bi-Monthly

further enhancements, this technology can revolutionize the way users interact with digital interfaces

References:

1. Catherine, S., Kiruthiga, V., & Gabriel, R. (2024). Effective Brand Building in Metaverse Platform: Consumer-Based Brand Equity in a Virtual World (CBBE). In Omnichannel Approach to Co-Creating Customer Experiences Through Metaverse Platforms (pp. 39-48). IGI Global Scientific Publishing.
2. Catherine, S., Ramasundaram, G., Nimmagadda, M. R., & Suresh, N. V. (2025). Roots, Routes, and Identity: How Culture Shapes Heritage Travel. In Multiple-Criteria Decision-Making (MCDM) Techniques and Statistics in Marketing (pp. 343-352). IGI Global Scientific Publishing.
3. Catherine, S., Suresh, N. V., Mangaiyarkarasi, T., & Jenefa, L. (2025). Unveiling the Enigma of Shadow: Ethical Difficulties in the Field of AI. In Navigating Data Science: Unleashing the Creative Potential of Artificial Intelligence (pp. 57-67). Emerald Publishing Limited.
4. Gokila, S., Helen, D., Alemu, A. M., & Suresh, N. V. (2024, November). Scaling Approach Over Learning Layer of Deep Learning Model to Reduce the FALSE Error in Binary Classification. In 2024 8th International Conference on Electronics, Communication and Aerospace Technology (ICECA) (pp. 1294-1300). IEEE.
5. Helen, D., & Suresh, N. V. (2024). Generative AI in Healthcare: Opportunities, Challenges, and Future Perspectives. Revolutionizing the Healthcare Sector with AI, 79-90.
6. Kalaivani, M., Suganya, V., Suresh, N. V., & Catherine, S. (2025). The Next Wave in Marketing: Data Science in the Age of Generative AI. In Navigating Data Science (pp. 13-26). Emerald Publishing Limited.
7. Poongavanam, S., Srinivasan, R., Arivazhagan, D., & Suresh, N. V. (2023). Medical Inflation-Issues and Impact. Chettinad Health City Medical Journal (E-2278-2044 & P-2277-8845), 12(2), 122-124.
8. Suganya, V., & Suresh, N. V. (2024). Potential Mental and Physical Health Impacts of Spending Extended Periods in the Metaverse: An Analysis. In Creator's Economy in Metaverse Platforms: Empowering Stakeholders Through Omnichannel Approach (pp. 225-232). IGI Global.
9. Suresh, N. V., & Rexy, V. A. M. (2024, February). An Empirical Study on Empowering Women through Self Help Groups. In 3rd International Conference on Reinventing Business Practices, Start-ups and Sustainability

https://doi.org/10.47059/AJMS/V4I6/03

ASET JOURNAL OF MANAGEMENT SCIENCE
Peer Reviewed & Open Access Journal
ISSN : 2584 - 220X (Online) | RNI : Applied | Frequency : Bi-Monthly

(ICRBSS 2023) (pp. 957-964). Atlantis Press.

10. Suresh, N. V., Catherine, S., Selvakumar, A., & Sridhar, G. Transparency and accountability in big data analytics: Addressing ethical challenges in decision-making processes. In Digital Transformation and Sustainability of Business (pp. 742-745). CRC Press.
11. Suresh, N. V., Karthikeyan, M., Sridhar, G., & Selvakumar, A. (2025). Sustainable urban planning through AI-driven smart infrastructure: A comprehensive review. Digital Transformation and Sustainability of Business, 178-180.
12. Suresh, N. V., Manoj, G., Rajkumar, M. D., & Kanagasabai, B. (2024). Fundamental anomalies as a mediator in the relationship between heuristics and investment decisions. International Journal of Applied Management Science, 16(4), 383-396.
13. Suresh, N. V., Selvakumar, A., Sridhar, G., & Jain, V. (2025). Dynamic Pricing Strategies Implementing Machine Learning Algorithms in E-Commerce. In Building Business Models with Machine Learning (pp. 129-136). IGI Global Scientific Publishing.
14. Suresh, N. V., Selvakumar, A., Sridhar, G., & Trivedi, S. (2024). A Research Study on the Ethical Considerations in Harnessing Basic Science for Business Innovation. In Unleashing the Power of Basic Science in Business (pp. 55-64). IGI Global.
15. Suresh, N. V., Shanmugam, R., Selvakumar, A., & Sridhar, G. Patient-centric care optimization: Strategies for enhancing communication and efficiency in healthcare settings through cross-functional collaboration. In Digital Transformation and Sustainability of Business (pp. 738-741). CRC Press.
16. Suresh, N. V., Sridhar, J., Selvakumar, A., & Catherine, S. (2024). Machine Learning Applications in Healthcare: Improving Patient Outcomes, Diagnostic Accuracy, and Operational Efficiency. In AI Healthcare Applications and Security, Ethical, and Legal Considerations (pp. 1-9). IGI Global

https://doi.org/10.47059/AJMS/V4I6/03