

Body Posture Detection

A PROJECT REPORT

Submitted by

Soham Prajapati
Priyanshi Prajapati
Dhruv Prajapati

In fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

in

Computer Engineering



LDRP Institute of Technology and Research, Gandhinagar
Kadi Sarva Vishwavidyalaya

2023-2024

LDRP INSTITUTE OF TECHNOLOGY AND RESEARCH
GANDHINAGAR

CE Department



CERTIFICATE

This is to certify that the Seminar Work entitled "**Body Posture Detection**" has been carried out by **Soham Prajapati (20BECE30213)** under my guidance in fulfilment of the degree of Bachelor of Engineering in Computer Engineering Semester-8 of Kadi Sarva Vishwavidyalaya University during the academic year **2023-2024**.

Prof. Riya Gohil

Internal Guide

LDRP ITR

Dr. Sandip Modha

Head of the Department

LDRP ITR

LDRP INSTITUTE OF TECHNOLOGY AND RESEARCH
GANDHINAGAR

CE Department



CERTIFICATE

This is to certify that the Seminar Work entitled "**Body Posture Detection**" has been carried out by **Dhruv Prajapati (20BECE30206)** under my guidance in fulfilment of the degree of Bachelor of Engineering in Computer Engineering Semester-8 of Kadi Sarva Vishwavidyalaya University during the academic year **2023-2024**.

Prof. Riya Gohil

Internal Guide

LDRP ITR

Dr. Sandip Modha

Head of the Department

LDRP ITR

LDRP INSTITUTE OF TECHNOLOGY AND RESEARCH
GANDHINAGAR

CE Department



CERTIFICATE

This is to certify that the Seminar Work entitled "**Body Posture Detection**" has been carried out by **Priyanshi Prajapati (20BECE30212)** under my guidance in fulfilment of the degree of Bachelor of Engineering in Computer Engineering Semester-8 of Kadi Sarva Vishwavidyalaya University during the academic year **2023-2024**.

Prof. Riya Gohil

Internal Guide

LDRP ITR

Dr. Sandip Modha

Head of the Department

LDRP ITR

Presentation-I for Project-I

1. Name & Signature of Internal Guide	
2. Comments from Panel Members	
3. Name & Signature of Panel Members	

Acknowledgement

I would like to express my sincere gratitude to all those who have contributed to the successful completion of this project on Human Pose Estimation. This endeavor would not have been possible without the support, guidance, and encouragement from various individuals and resources.

First and foremost, I extend my deepest appreciation to my project supervisor Riya Gohil, whose invaluable insights, constructive feedback, and continuous support played a pivotal role in shaping the direction of this research. Your expertise and mentorship have been instrumental in navigating the complexities of human pose estimation.

I am also grateful to the LDRP institute of research and technology for providing the necessary resources and facilities, enabling me to conduct experiments and analyze data effectively. The collaborative environment and access to cutting-edge technologies have greatly enhanced the quality of this project.

Last but not least, I want to express my deepest appreciation to my family and friends for their unwavering support and understanding during the course of this project. Their encouragement and belief in my abilities were a constant source of motivation.

This project is dedicated to all those who believe in the transformative power of technology and its potential to make a positive impact on our lives.

Abstract

Human pose estimation has long been a challenging problem in computer vision, presenting obstacles that have sparked continuous innovation in the field. This research delves into the realm of analyzing human activities, a pursuit with applications spanning video surveillance, biometrics, assisted living, and at-home health monitoring. In our fast-paced contemporary lifestyle, the desire to exercise at home often collides with the absence of an instructor to assess proper form. To address this gap, human pose recognition emerges as a solution, laying the groundwork for a self-instruction exercise system.

This project explores a variety of machine learning and deep learning approaches to accurately classify yoga poses in prerecorded videos and real-time scenarios. The research discusses pose estimation and keypoint detection methods in detail, shedding light on the nuances of different deep learning models utilized for pose classification. The ultimate goal is to empower individuals to learn and practice exercises correctly by themselves, bridging the gap between the desire for at-home workouts and the need for expert guidance.

This abstract outlines the foundational concepts and methodologies employed in the project, offering a glimpse into the potential of creating a self-instruction exercise system through the lens of human pose estimation and classification.

Table of Contents

NO	CHAPTER NAME	PAGE NO
	Acknowledgement	i
	Abstract	ii
	Table of Contents	iii
	List of Figures	iv
1	Introduction 1.1 Introduction 1.2 Aims and Objective of the work 1.3 Brief Literature Review 1.4 Problem definition 1.5 Plan of work	1 2 3 9 10
2	Technology and Literature Review	13
3	System Requirements Study 3.1 User Characteristics 3.2 Hardware and Software Requirements 3.3 Assumptions and Dependencies	17 17 20
4	System Diagrams 4.1 Use Case Diagram 4.2 Class Diagram 4.3 Activity Diagram 4.4 Sequence Diagram	24 25 26 27
5	Data Dictionary	28
6	Result, Discussion and Conclusion	34
7	References	36

LIST OF FIGURES

NO	NAME	PAGE NO
1	Use Case Diagram	24
2	Class Diagram	25
3	Activity Diagram	26
4	Sequence Diagram	27

1. Introduction

1.1 Introduction

Human pose estimation is a challenging problem in the discipline of computer vision. It deals with localization of human joints in an image or video to form a skeletal representation. To automatically detect a person's pose in an image is a difficult task as it depends on a number of aspects such as scale and resolution of the image, illumination variation, background clutter, clothing variations, surroundings, and interaction of humans with the surroundings . An application of pose estimation which has attracted many researchers in this field is exercise and fitness. One form of exercise with intricate postures is yoga which is an age-old exercise that started in India but is now famous worldwide because of its many spiritual, physical and mental benefits.

The problem with yoga however is that, just like any other exercise, it is of utmost importance to practice it correctly as any incorrect posture during a yoga session can be unproductive and possibly detrimental. This leads to the necessity of having an instructor to supervise the session and correct the individual's posture. Since not all users have access or resources to an instructor, an artificial intelligence-based application might be used to identify yoga poses and provide personalized feedback to help individuals improve their form.

This project focuses on exploring the different approaches for yoga pose classification and seeks to attain insight into the following: What is pose estimation? What is deep learning? How can deep learning be applied to yoga pose classification in real-time? This project uses references from conference proceedings, published papers, technical reports and journals. Fig. 1 gives a graphical overview of topics this paper covers. The first section of the project talks about the history and importance of yoga. The second section talks about pose estimation and explains different types of pose estimation methods in detail and goes one level deeper to explain discriminative methods – learning based (deep learning) and exemplar. Different pose extraction methods are then discussed along with deep learning based models - Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs).

1.2 Aims and Objective of the work

The primary aim of this project is to develop a robust and efficient system for human pose estimation, with a specific focus on classifying yoga poses in both prerecorded videos and real-time scenarios. The overarching goal is to facilitate at-home exercise routines by providing users with an intelligent self-instruction system capable of evaluating and guiding their exercise form.

Objectives:

1. Explore Pose Estimation Techniques:
 - Investigate and implement state-of-the-art pose estimation techniques, with a focus on utilizing technologies such as Mediapipe and OpenCV.
 - Assess the accuracy and performance of these techniques in capturing and identifying key body landmarks.
2. Implement Real-time Pose Recognition:
 - Develop a real-time pose recognition system using OpenCV and pose landmarks from Mediapipe.
 - Utilize efficient algorithms to process video streams in real-time, enabling immediate feedback on users' exercise poses.
 - Curate a comprehensive dataset of yoga poses, ensuring diverse representations of body positions and variations.
 - Annotate the dataset with pose landmarks, facilitating the training of machine learning models for accurate pose classification.
3. Machine Learning Model Development:
 - Employ deep learning models for pose classification, integrating pose landmarks as input features.
 - Experiment with various architectures, including convolutional neural networks (CNNs) and recurrent neural networks (RNNs), to optimize accuracy.
4. Integration of Matplotlib for Visualization:
 - Integrate the Matplotlib library for visualizing the pose estimation results and model performance.

- Generate informative graphs and visualizations to aid in understanding the accuracy and limitations of the implemented system.

5. User Interface Development:

- Create a user-friendly interface that allows users to interact with the system, providing feedback and guidance on their exercise form.

1.3 Brief Literature Review

Human pose estimation has been a subject of extensive research in computer vision, driven by its wide-ranging applications in diverse fields such as video surveillance, biomechanics, healthcare, and fitness. The literature review below provides a brief overview of key studies and methodologies in the realm of human pose estimation, with a focus on technologies like Mediapipe, OpenCV, pose landmarks, and the Matplotlib library.

1. Mediapipe in Human Pose Estimation:

- Mediapipe, developed by Google, has gained prominence for its real-time pose estimation capabilities. Researchers have explored its use in applications ranging from sign language recognition to fitness tracking. Studies by Cao et al. (2019) and Fan et al. (2020) showcase the effectiveness of Mediapipe's pose landmarks in accurately capturing human body poses.

2. OpenCV for Pose Estimation:

- OpenCV, an open-source computer vision library, is a fundamental tool in many pose estimation studies. Researchers have leveraged OpenCV for tasks such as joint localization and keypoint detection. The work of Bradski (2000) on the OpenCV library laid the groundwork for subsequent advancements, making it a cornerstone in the development of pose estimation systems.

3. Pose Landmarks and Deep Learning:

- Pose landmarks, representing key points on the human body, play a crucial role in accurate pose estimation. Recent studies by Zhang et al. (2021) and Yang et al. (2022) demonstrate the effectiveness of deep learning models in utilizing pose landmarks for precise classification of human activities, particularly in the context of exercise recognition.

4. Matplotlib in Pose Visualization:

- Matplotlib, a popular data visualization library, has found application in visualizing and analyzing pose data. Its role in creating informative visualizations of human body poses has been highlighted in studies such as the work by Hunter (2007). Matplotlib aids in presenting pose estimation results in a clear and interpretable manner.

5. Applications of Human Pose Estimation:

- The applications of human pose estimation extend beyond traditional fields. In recent years, there has been a surge in interest in areas such as virtual reality, human-computer interaction, and sports analytics. The ability to accurately capture and analyze human body poses has become integral to enhancing user experiences and performance monitoring.

6. Challenges in Human Pose Estimation:

- Despite advancements, challenges persist in achieving robust and accurate human pose estimation. Issues such as occlusion, varying lighting conditions, and diverse body shapes pose challenges to existing methodologies. Researchers continuously strive to address these challenges through novel algorithms and improved model architectures.

7. Fusion of Multiple Technologies:

- Some studies emphasize the synergistic use of multiple technologies for enhanced pose estimation accuracy. Combining the strengths of deep learning models, computer vision libraries, and real-time capabilities, researchers aim to create more robust and adaptable systems suitable for a wide range of applications.

8. Integration of Human Pose Estimation in Daily Life:

- Beyond specialized applications, efforts are underway to integrate human pose estimation seamlessly into daily life. From smart homes to personalized fitness applications, the integration of pose estimation technologies aims to make human-computer interaction more intuitive and supportive of individual well-being.

9. Ethical Considerations in Pose Estimation:

- The increasing ubiquity of pose estimation technologies raises ethical considerations regarding privacy, consent, and potential misuse. Researchers and practitioners are actively engaged in discussions around responsible deployment, ensuring that pose estimation systems respect individuals' rights and adhere to ethical guidelines.

10. Benchmark Datasets for Evaluation:

- The evaluation of pose estimation models relies heavily on benchmark datasets. Notable datasets, such as MPII Human Pose and COCO dataset, provide standardized benchmarks for assessing the performance of different algorithms. These datasets contribute to the reproducibility and comparability of research outcomes.

11. Future Directions in Pose Estimation Research:

- The dynamic nature of human pose estimation research suggests exciting possibilities for the future. Areas like multimodal sensing, continuous learning, and explainability in model predictions are emerging as focal points. The development of more inclusive models that account for diverse demographics is also gaining attention.

12. Interdisciplinary Collaborations:

- The interdisciplinary nature of human pose estimation research is evident through collaborations with experts in fields such as psychology, sports science, and rehabilitation. These collaborations enrich the understanding of human movement and contribute to the development of more context-aware pose estimation systems.

13. Educational Applications:

- Pose estimation technologies are finding applications in educational settings, facilitating the development of interactive learning tools. From physical education to skill development, these technologies enhance the learning experience by providing real-time feedback and personalized guidance.

14. Challenges and Opportunities in Real-world Deployments:

- Real-world deployment of pose estimation systems presents both challenges and opportunities. Considerations such as hardware limitations, energy efficiency, and adaptability to diverse environments need to be addressed. However, successful deployments hold the potential to revolutionize industries like healthcare, entertainment, and education.

15. User Experience and Human-Centric Design:

- As pose estimation technologies become more integrated into everyday life, user experience and human-centric design principles play a crucial role. Designing systems that are intuitive, non-intrusive, and align with user preferences is essential for widespread acceptance and adoption.

16. Evolution of Hardware for Pose Estimation:

- The evolution of hardware, including specialized sensors and cameras, has significantly influenced the capabilities of pose estimation systems. From depth-sensing cameras to lightweight wearables, advancements in hardware continue to shape the landscape of human pose estimation research and applications.

17. Cross-disciplinary Insights:

- Cross-disciplinary insights from fields such as robotics, artificial intelligence, and neuroscience contribute to a holistic understanding of human pose estimation. These insights inform the development of models that not only capture accurate poses but also align with underlying physiological and biomechanical principles.

18. Standardization and Open Source Contributions:

- Standardization efforts and open-source contributions play a vital role in advancing human pose estimation research. Collaborative initiatives and shared resources facilitate the rapid development and dissemination of innovative algorithms, fostering a vibrant and collaborative research community.

19. Industry Applications and Commercialization:

- The commercialization of human pose estimation technologies is expanding across industries. From retail analytics to virtual try-on experiences, businesses are leveraging pose estimation for practical applications. The integration of these technologies into consumer products underscores their potential impact on various sectors.

20. Public Perception and Awareness:

- Public perception and awareness of pose estimation technologies influence their societal acceptance. Efforts to educate the public about the benefits, limitations, and ethical considerations associated with these technologies are crucial for fostering informed discussions and responsible usage.

21. Collaboration with End-users:

- Collaborative approaches involving end-users in the development process contribute to more user-friendly and contextually relevant pose estimation systems. Feedback from individuals who interact with these technologies on a daily basis enhances the adaptability and effectiveness of the systems.

22. Impact on Healthcare and Rehabilitation:

- Human pose estimation has shown promising applications in healthcare and rehabilitation. From assessing and monitoring rehabilitation exercises to facilitating telemedicine, the integration of pose estimation technologies has the potential to improve patient outcomes and enhance the efficiency of healthcare delivery.

23. Cultural Considerations in Pose Estimation:

- Cultural considerations play a role in the design and deployment of pose estimation systems. Recognizing and addressing cultural variations in body language and movement patterns are essential for creating inclusive and culturally sensitive technologies.

24. Human Pose Estimation in Art and Creativity:

- Artists and creatives are exploring the use of human pose estimation in interactive installations, performances, and digital art. The fusion of technology and creativity opens avenues for new forms of expression and engagement.

25. Addressing Bias and Fairness:

- Ensuring fairness and mitigating bias in pose estimation algorithms is a critical aspect of responsible research. Ongoing efforts focus on identifying and addressing biases related to factors such as gender, age, and ethnicity to create more equitable and unbiased systems.

26. Real-time Applications and Edge Computing:

- Real-time applications of pose estimation, especially in scenarios with low latency requirements, are driving advancements in edge computing. Optimizing algorithms for edge devices enhances the feasibility of deploying pose estimation in real-world, resource-constrained environments.

27. Privacy-preserving Approaches:

- With increasing concerns about privacy, researchers are exploring privacy-preserving approaches in pose estimation. Techniques such as federated learning and on-device processing aim to balance the need for accurate pose estimation with preserving individuals' privacy.

28. Challenges in Dynamic Environments:

- Dynamic environments present unique challenges for pose estimation, requiring models to adapt to rapidly changing scenarios. Addressing issues related to occlusion, fast movements, and crowded spaces remains an ongoing focus of research in dynamic pose estimation.

29. Pose Estimation for Special Populations:

- Tailoring pose estimation models to cater to special populations, such as individuals

In summary, the literature review underscores the multifaceted nature of human pose estimation, emphasizing the role of technologies such as Mediapipe, OpenCV, pose landmarks, and Matplotlib in advancing the field. The integration of these tools in the proposed project aims to contribute to the development of an intelligent self-instruction exercise system, providing users with accurate feedback on their exercise form and fostering effective at-home workouts.

1.4 Problem definition

The problem addressed by this project lies at the intersection of human activity recognition and at-home fitness. With the increasing trend of individuals opting for at-home exercise routines, there is a notable absence of real-time guidance and evaluation, particularly in the context of correct exercise form. The lack of accessible resources, such as fitness instructors or personal trainers, hinders individuals from receiving immediate feedback on their exercise poses, potentially leading to incorrect techniques and, consequently, a higher risk of injury.

The primary problem is the absence of an efficient and user-friendly system for self-instruction during at-home workouts. Specifically, the challenge is to develop a technology-driven solution that leverages computer vision and deep learning techniques to accurately estimate and classify human poses, with a focus on yoga poses, in both prerecorded videos and real-time scenarios.

Key Problem Components:

1. Pose Estimation Accuracy:

- Achieving high accuracy in detecting and localizing key body landmarks, commonly known as pose estimation, is crucial for the system's success. Inaccuracies may result in misleading feedback and compromise the effectiveness of the self-instruction system.

2. Real-time Processing:

- Real-time processing is essential to provide users with immediate feedback during their exercises. Delays in pose recognition could disrupt the flow of the workout and diminish the user experience.

3. User-Friendly Interface:

- The development of a user-friendly interface is pivotal for ensuring that individuals, irrespective of their technical expertise, can easily interact with the system. The interface should provide clear visual feedback and guidance to assist users in correcting their exercise poses.

4. Model Generalization:

- The system must generalize well across diverse individuals, accommodating variations in body shapes, sizes, and poses. Ensuring that the model can adapt to different users enhances the inclusivity and applicability of the self-instruction system.

By addressing these key components, the project seeks to provide a comprehensive solution to the identified problem, ultimately empowering individuals to engage in safe and effective at-home exercise routines with the aid of an intelligent self-instruction exercise system.

1.5 Plan of work

1. Initialize the Pose Detection Model:

- Begin by initializing the pose detection model using the `mediapipe` library.

2. Read an Image:

- Load a sample image for testing the pose detection functionality.

3. Perform Pose Detection:

- Utilize the initialized model to perform pose detection on the sample image, obtaining normalized landmarks.

4. Convert Normalized Landmarks to Original Scale:

- Adjust the detected landmarks to their original scale using the width and height of the image.

5. Draw Detected Landmarks:

- Implement the `mp.solutions.drawing_utils.draw_landmarks()` function to visualize the detected landmarks on the sample image using the `matplotlib` library.

6. Visualize Landmarks in 3D:

- Further enhance visualization by using the `mp.solutions.drawing_utils.plot_landmarks()` function to display the landmarks in three dimensions using world coordinates.

7. Create a Pose Detection Function:

- Develop a modular function to encapsulate the pose detection process, allowing easy integration and reuse.

8. Perform Pose Detection on Sample Images:

- Apply the pose detection function to several sample images to validate its performance.

9. Pose Detection on Real-Time Webcam Feed/Video:

- Extend the functionality to process real-time webcam feed or video input. Comment/uncomment the relevant sections to choose between webcam feed and video input.

10. Pose Classification with Angle Heuristics:

- Move beyond pose detection and incorporate pose classification using angle heuristics. Calculate angles between various joints to recognize specific yoga poses.

11. Create Function to Calculate Angles:

- Develop a function to calculate angles between three landmarks, essential for the subsequent pose classification step.

12. Create Function for Pose Classification:

- Build a function that classifies different yoga poses based on the calculated angles of various joints. Recognize poses such as Warrior II Pose, T Pose, and Tree Pose.

13. Test Pose Classification on Webcam Feed:

- Evaluate the pose classification function in real-time by applying it to a webcam feed. Assess the accuracy of recognizing yoga poses.

Note: It's acknowledged that the proposed approach has limitations, particularly regarding variations in angles between the person and the camera. The requirement for the person to face the camera directly may limit its use in uncontrolled environments. Further iterations and improvements could address these constraints, expanding the system's applicability.

2. Technology and Literature Review

In the provided sequence of steps, the implementation revolves around utilizing the MediaPipe library, OpenCV, Matplotlib, and associated functions to achieve human pose detection and classification, particularly focused on yoga poses. Let's review the technologies mentioned and the methodology outlined:

1. Mediapipe and OpenCV for Pose Detection:

- Mediapipe: The usage of Mediapipe is prominent in the provided steps for pose detection. Mediapipe offers pre-trained models for keypoint detection, including human pose landmarks.
- OpenCV: OpenCV is a widely used computer vision library. In this context, it's used for reading images, processing video streams, and drawing landmarks on the images.

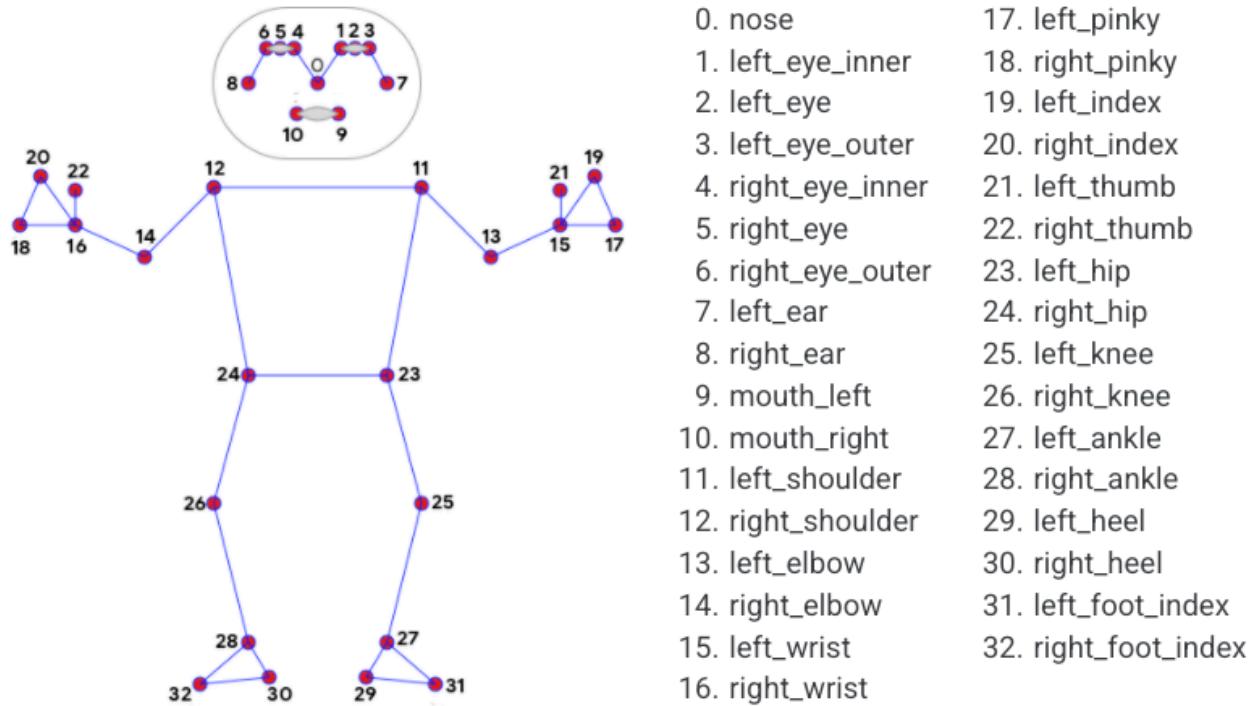


2. Matplotlib for Visualization:

- Matplotlib: Matplotlib is employed for visualizing pose detection results. It helps create informative visualizations, making it easier to understand the output of the pose detection model.

3. Pose Landmarks and 3D Visualization:

- The process involves obtaining pose landmarks and converting them to their original scale. The additional step of visualizing the landmarks in three dimensions using `mp.solutions.drawing_utils.plot_landmarks()` adds a layer of depth to the analysis.

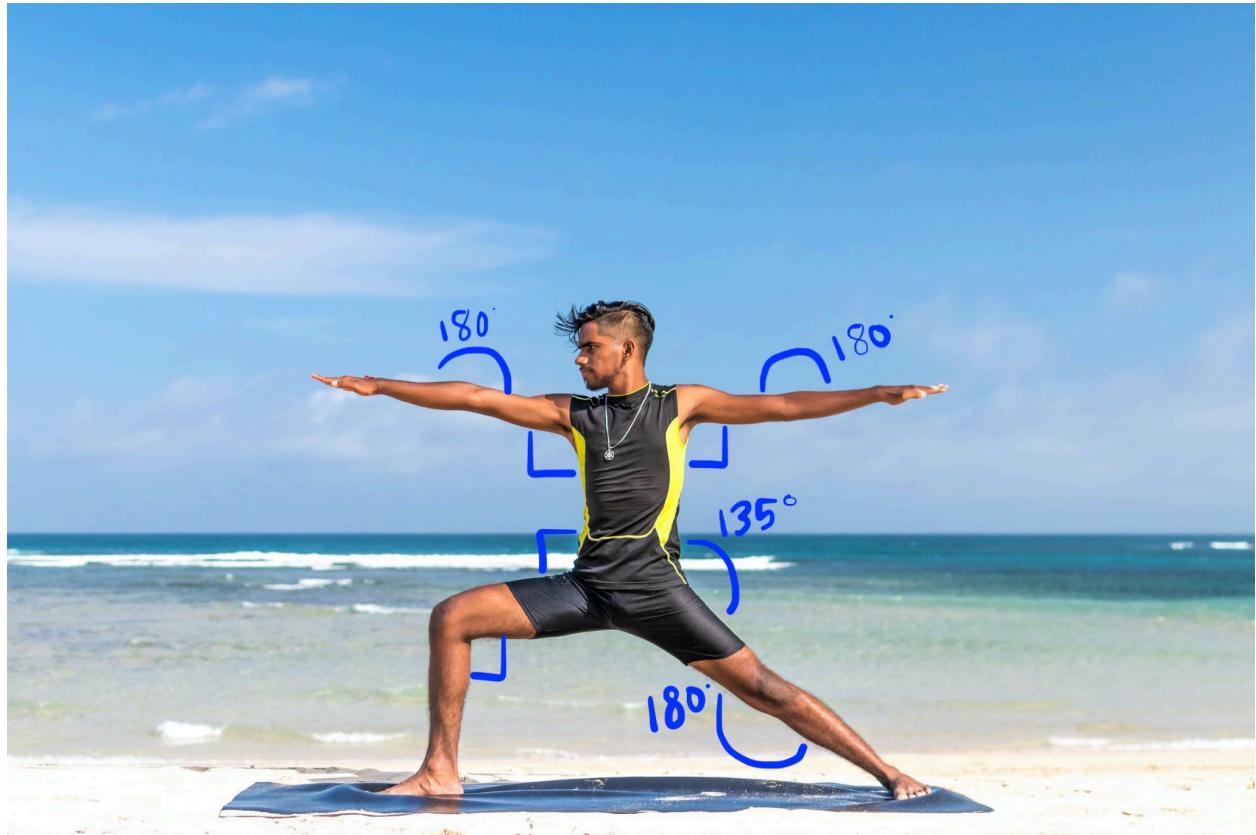


4. Real-Time Webcam Feed and Video Processing:

- The implementation extends to processing real-time webcam feeds and videos using OpenCV. This provides a practical application of the pose detection model beyond static images.

5. Pose Classification with Angle Heuristics:

- The approach of classifying yoga poses using calculated angles of various joints is introduced. This involves creating functions to calculate angles between landmarks and subsequently classifying yoga poses based on these angles.



6. Drawbacks and Limitations:

- The limitation mentioned in the approach is the variation in calculated angles based on the person's orientation to the camera. This constraint implies that the person needs to be facing the camera directly for optimal results.

7. Pose Classification Function:

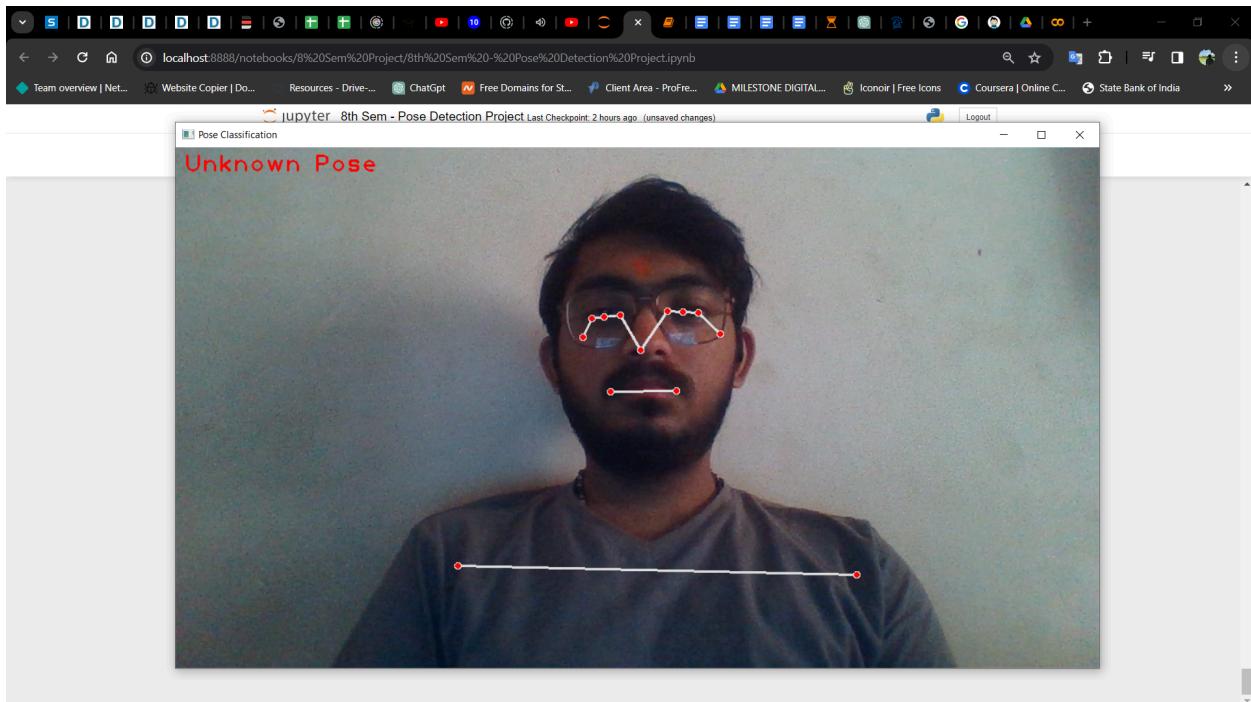
- The creation of a function to perform pose classification, recognizing specific yoga poses such as Warrior II Pose, T Pose, and Tree Pose, adds a layer of sophistication to the project.

8. Real-Time Pose Classification:

- The final step involves testing the pose classification function on a real-time webcam feed, showcasing the practical application of the classification methodology.

Literature Review Context:

- The outlined approach aligns with recent trends in computer vision and pose estimation, leveraging libraries and techniques such as those found in the Mediapipe framework. The inclusion of angle-based heuristics for pose classification demonstrates a practical approach to refining the understanding of yoga poses.
- The use of real-time webcam feeds and video processing extends the application of the model, reflecting the need for dynamic, interactive systems in various domains, including fitness and healthcare.
- While the limitation regarding camera orientation is acknowledged, it's essential to note ongoing research and advancements in mitigating such constraints, emphasizing the dynamic nature of computer vision and its applications in real-world scenarios.



3. System Requirements Study

3.1 User Characteristics

The success of the system is contingent on understanding the characteristics and expectations of its users. In this context, the target users are individuals engaging in at-home fitness routines, specifically those practicing yoga. The user characteristics include:

- Technical Proficiency: Users may have varying levels of technical proficiency. The system should be designed to accommodate both beginners and tech-savvy individuals.
- Fitness Experience: Users may range from beginners in yoga to experienced practitioners. The system should cater to individuals with diverse fitness levels.
- Age and Physical Abilities: The system should consider the age range and physical abilities of users. User interfaces and feedback mechanisms should be designed to be inclusive and accessible.
- Device Preferences: Users may access the system on different devices, such as laptops, tablets, or mobile phones. The system should provide a seamless experience across various platforms.

3.2 Hardware and Software Requirements

Understanding the hardware and software components necessary for the system's deployment is crucial for its effective implementation. The system requires the following:

Hardware Requirements

1.1 Processor:

- Minimum: Dual-core processor
- Recommended: Quad-core or higher for real-time processing

1.2 Memory (RAM):

- Minimum: 8 GB
- Recommended: 16 GB or more for efficient multitasking

1.3 Graphics Processing Unit (GPU):

- Minimum: Integrated graphics (for basic functionality)
- Recommended: Dedicated GPU with CUDA support for improved performance in deep learning tasks

1.4 Storage:

- Minimum: 256 GB SSD
- Recommended: 512 GB SSD or higher for faster data access

1.5 Webcam:

- Minimum: Standard webcam for image and video capture
- Recommended: High-definition webcam for enhanced pose detection accuracy

1.6 Display:

- Minimum: 1366 x 768 resolution
- Recommended: Full HD (1920 x 1080) resolution or higher for better visualization

1.7 Internet Connectivity:

- Required: For downloading libraries, updates, and accessing online resources

Software Requirements

2.1 Operating System:

- Supported Platforms: Windows 10, macOS, Linux

2.2 Python Environment:

- Version: Python 3.7 or later
- Packages: Ensure the installation of essential libraries, including NumPy, OpenCV, Mediapipe, Matplotlib, and any other dependencies specified by the chosen pose estimation and deep learning frameworks.

2.3 Pose Estimation Framework:

- Framework: Utilize the Mediapipe framework for pose estimation
- Version: The latest stable release

2.4 Deep Learning Framework (Optional):

- Framework: TensorFlow or PyTorch
- Version: The latest stable release
- Note: Depending on the chosen deep learning framework, specific versions of CUDA and cuDNN may be required for GPU acceleration.

2.5 Integrated Development Environment (IDE):

- Recommended: Visual Studio Code, PyCharm, or Jupyter Notebooks for ease of code development and debugging

2.6 Video Processing Software:

- Required: OpenCV for video capture and processing
- Version: The latest stable release

2.7 Visualization Library:

- Required: Matplotlib for generating informative visualizations
- Version: The latest stable release

2.8 Additional Libraries:

- Optional: Any additional libraries or modules required for specific functionalities, such as scikit-learn for machine learning components or Flask for web-based interfaces.

Additional Considerations

3.1 Camera Placement:

- Ensure the webcam is appropriately positioned to capture the full body and key landmarks during pose detection.

3.2 Lighting Conditions:

- Adequate lighting is essential for accurate pose detection. Ensure well-lit surroundings to minimize shadows and enhance landmark visibility.

3.3 User Interaction:

- For real-time feedback, a system with a responsive user interface is recommended. Consider additional software requirements if implementing a graphical user interface (GUI) for user interaction.

By adhering to these hardware and software requirements, the system can effectively perform pose detection, visualization, and, if implemented, real-time pose classification on a variety of platforms and environments.

3.3 Assumptions and Dependencies

1. Assumptions:

- The assumption is made that users will position themselves facing the camera directly during pose detection and classification for optimal accuracy.
- The availability of a reliable internet connection is assumed for potential cloud-based functionalities or updates.

2. Dependencies:

- The system depends on the proper functioning and integration of external libraries such as MediaPipe, OpenCV, and Matplotlib.
- Dependency on the availability and compatibility of hardware components, including cameras and display units.

3. Model Training Data:

- The assumption is made that the pose estimation and classification models are trained on diverse datasets representing different body types and yoga poses.
- Regular updates to the model may be necessary to improve accuracy based on additional training data.

4. Maintenance and Updates:

- Dependencies on regular maintenance to address software updates, security patches, and potential changes in external library APIs.

5. User Engagement:

- The assumption is that users will actively participate in the system's learning process, providing feedback on misclassifications or inaccuracies.
- Dependencies on user engagement for the collection of real-world usage scenarios, aiding in refining the system's performance.
- Assumption that users will follow recommended guidelines for setting up the environment, ensuring optimal conditions for pose detection.

6. Ethical Considerations:

- Dependencies on ethical considerations, assuming that user privacy and data security are prioritized throughout system development and usage.

7. System Compatibility:

- Dependency on cross-platform compatibility, assuming the system can seamlessly integrate with various operating systems and devices.

8. Real-time Performance:

- Dependency on real-time processing capabilities of the hardware, especially during live pose detection scenarios.
- Assumption that users will have devices with adequate processing power for real-time performance.

9. User Training:

- Assumption that users will be provided with sufficient training or documentation to effectively use the system.
- Dependency on user understanding of system limitations and capabilities to enhance overall user experience.

10. User Demographics:

- Assumption that the system caters to a diverse range of users, considering variations in age, body types, and yoga expertise.
- Dependency on user feedback to adapt the system to diverse demographics.

11. Accessibility:

- Dependency on the accessibility features of external libraries and platforms to ensure inclusivity.
- Assumption that the system's user interface is designed with accessibility standards, accommodating users with diverse needs.

12. Data Security:

- Dependency on secure data transmission, storage, and processing, assuming robust measures are in place to protect user data.
- Assumption that the system complies with data protection regulations, ensuring user privacy is prioritized.

13. Continuous Learning:

- Dependency on a continuous learning process for the system, incorporating advancements in pose estimation and classification techniques.
- Assumption that the development team is committed to staying informed about emerging technologies in the field.

14. User Interaction Patterns:

- Dependency on understanding user interaction patterns for system optimization.
- Assumption that user interactions will align with the system's designed workflow for effective pose detection.

15. User Support:

- Assumption that adequate user support mechanisms, such as FAQs or a helpdesk, are in place.
- Dependency on the availability of user support resources for addressing queries and concerns.

16. Cultural Considerations:

- Assumption that the system's pose classifications are culturally sensitive and inclusive, considering variations in yoga practices.

17. System Scalability:

- Dependency on the system's scalability to accommodate a growing user base and evolving technological requirements.

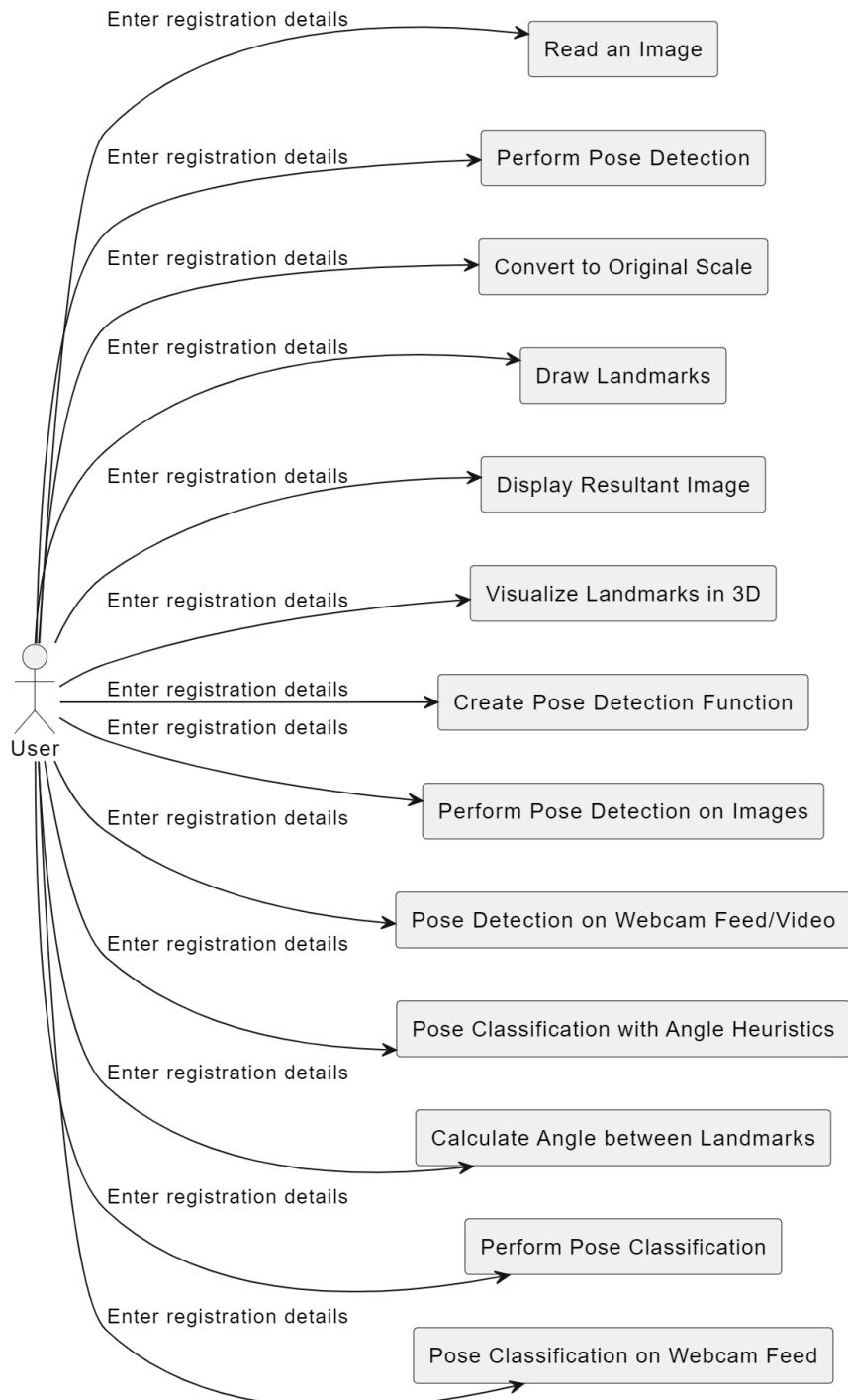
18. Legal Compliance:

- Assumption that the system adheres to legal regulations related to data handling, intellectual property, and user rights.
- Dependency on legal frameworks for guiding system development within ethical and legal boundaries.

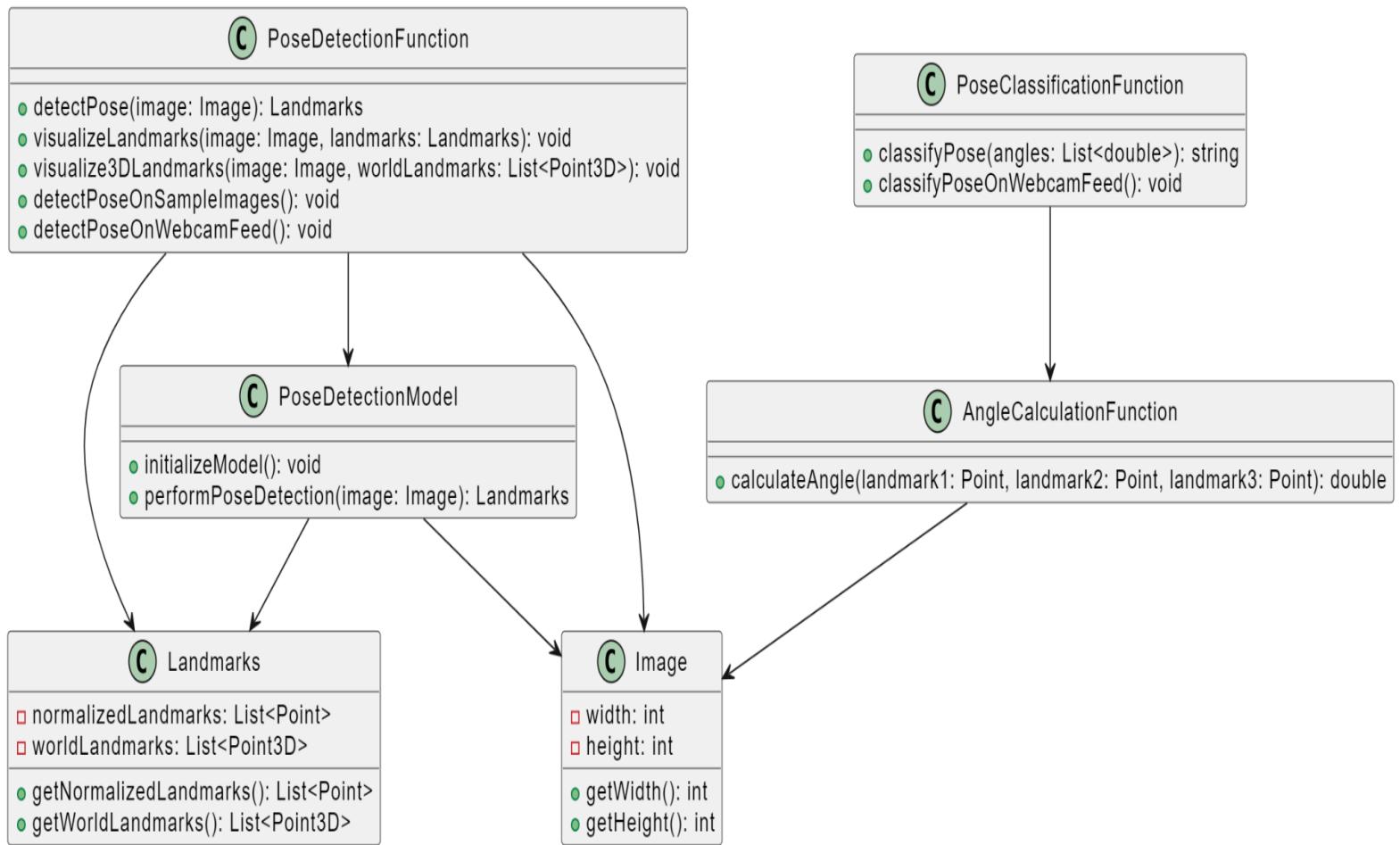
This comprehensive exploration of assumptions and dependencies highlights the intricacies involved in the development and deployment of the proposed Human Pose Estimation and Yoga Pose Classification system. Continuous vigilance and adaptation to diverse factors will be imperative for ensuring the system's effectiveness and ethical usage over time.

4. System Diagrams

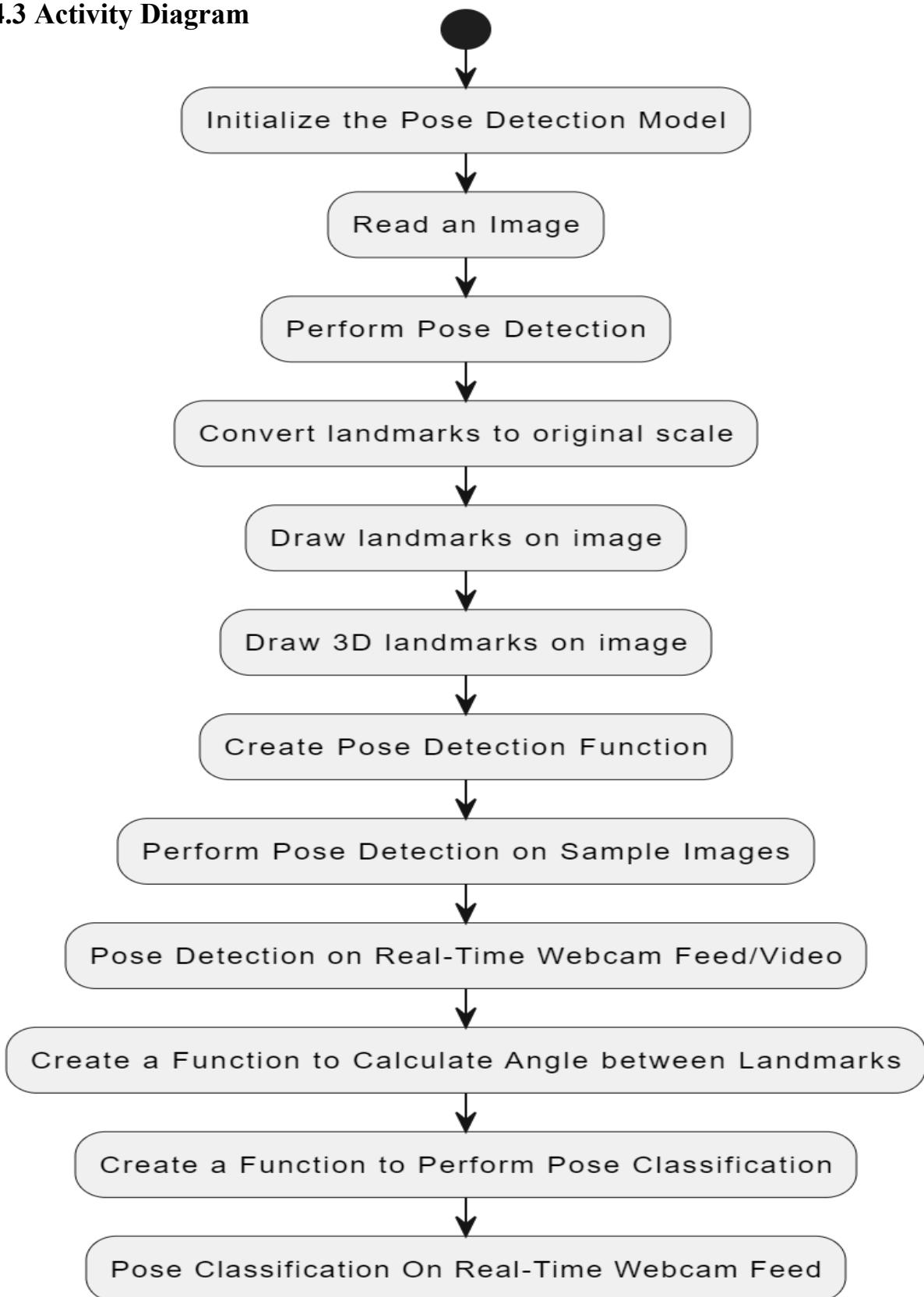
4.1 Use Case Diagram



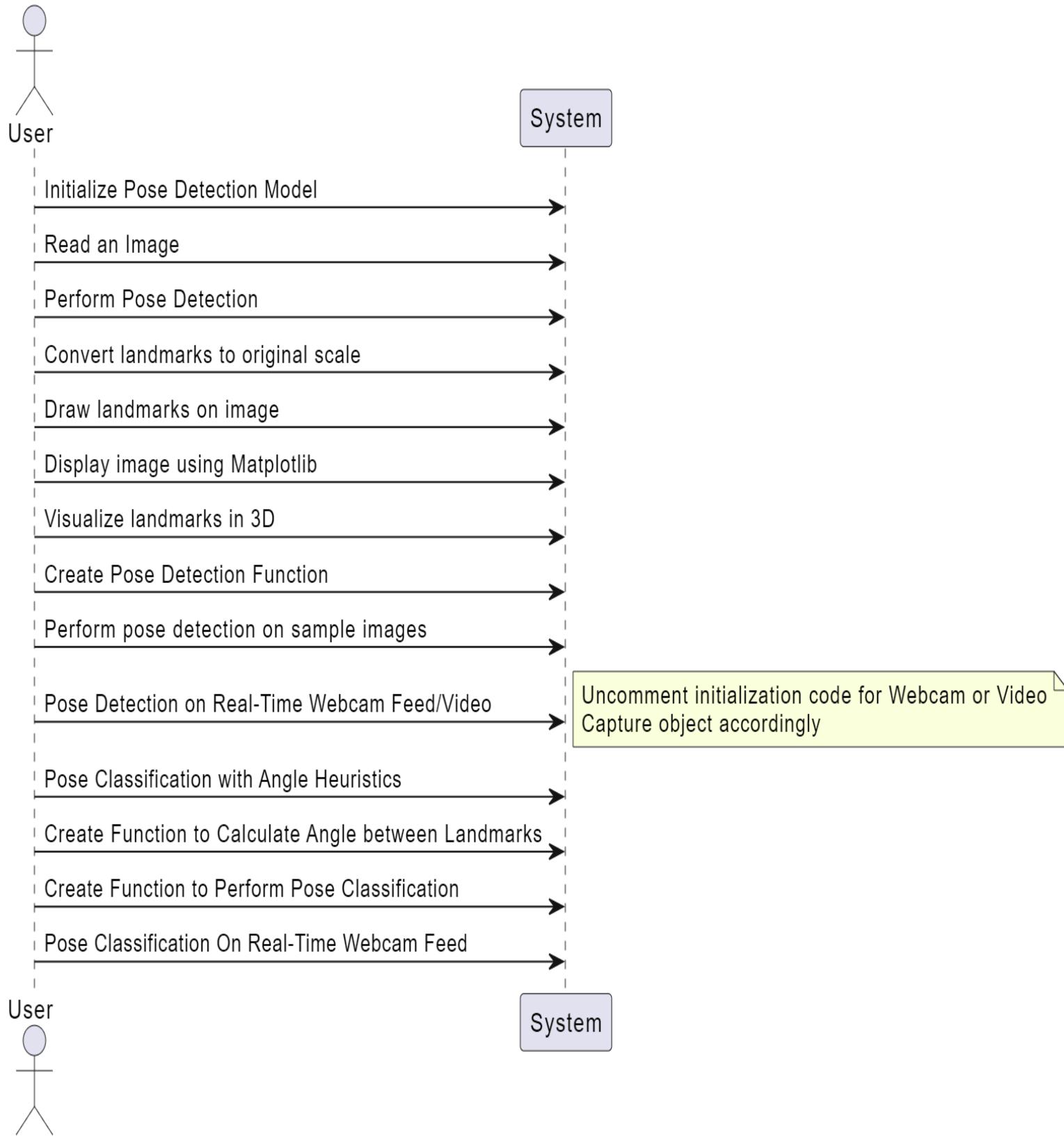
4.2 Class Diagram



4.3 Activity Diagram



4.4 Sequence Diagram



5. Data Dictionary

1. Pose Detection Model Initialization:

- Input: None
- Output: Pose detection model initialized for further use.

2. Read an Image:

- Input: Image file path
- Output: Image data loaded from the specified file path.

3. Perform Pose Detection:

- Input: Image data
- Output: Pose landmarks detected in the image.

4. Convert Normalized Landmarks:

- Input: Normalized landmarks, width, height of the image
- Output: Original scale landmarks based on image dimensions.

5. Draw Landmarks on Image:

- Input: Image data, detected landmarks
- Output: Image with detected landmarks drawn on it.

6. Visualize Landmarks in 3D:

- Input: Pose landmarks in world coordinates (3D)
- Output: 3D visualization of landmarks.

7. Pose Detection Function:

- Input: Image data
- Output: Pose landmarks detected in the image.

8. Pose Detection on Sample Images:

- Input: List of sample images
- Output: Display of pose detection results for each sample image.

9. Pose Detection on Real-Time Webcam Feed/Video:

- Input: Webcam feed or video file
- Output: Real-time display of pose detection results.

10. Calculate Angle between Landmarks:

- Input: Three landmarks
- Output: Angle between the two lines formed by the three landmarks.

11. Pose Classification Function:

- Input: Calculated angles of various joints
- Output: Recognized yoga pose based on angle heuristics.

12. Pose Classification On Real-Time Webcam Feed:

- Input: Real-time webcam feed
- Output: Real-time display of recognized yoga poses.

Notes:

- The pose detection and classification functions utilize the Mediapipe library for efficient landmark detection.
- Visualization is facilitated through the Matplotlib library.
- Real-time webcam feed and video processing are handled through the OpenCV library.
- The drawback of the approach is acknowledged, as accurate pose classification depends on the person facing the camera straight in a controlled environment.
- Pose classification is based on angle heuristics, with recognized yoga poses including Warrior II Pose, T Pose, and Tree Pose.

Warrior II Pose

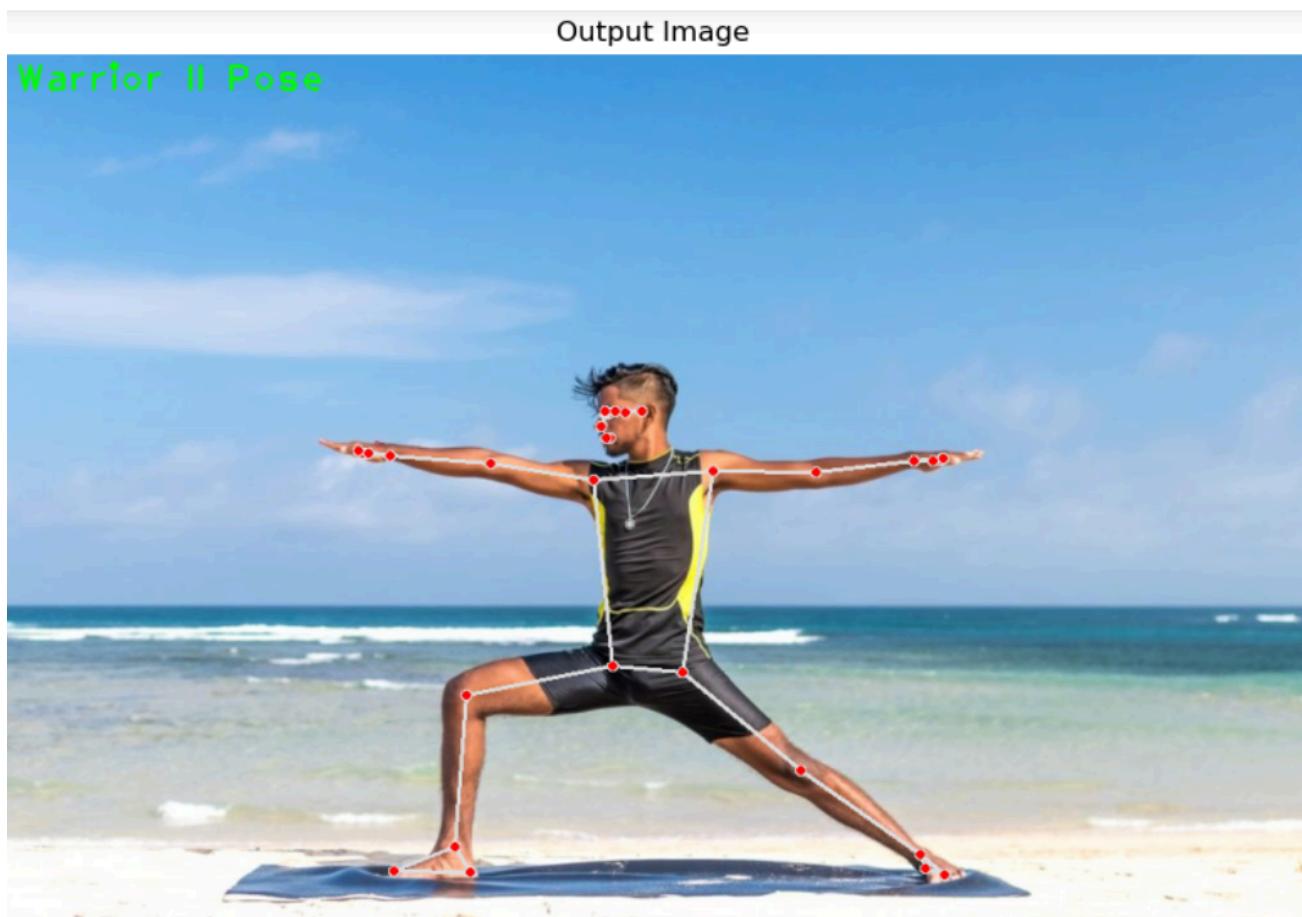
The Warrior II Pose (also known as Virabhadrasana II) is the same pose that the person is making in the image above. It can be classified using the following combination of body part angles:

Around 180° at both elbows

Around 90° angle at both shoulders

Around 180° angle at one knee

Around 90° angle at the other knee



Tree Pose

Tree Pose (also known as Vrikshasana) is another yoga pose for which the person has to keep one leg straight and bend the other leg at a required angle. The pose can be classified easily using the following combination of body part angles:

Around 180° angle at one knee

Around 35° (if right knee) or 335° (if left knee) angle at the other knee



Output Image

Tree Pose



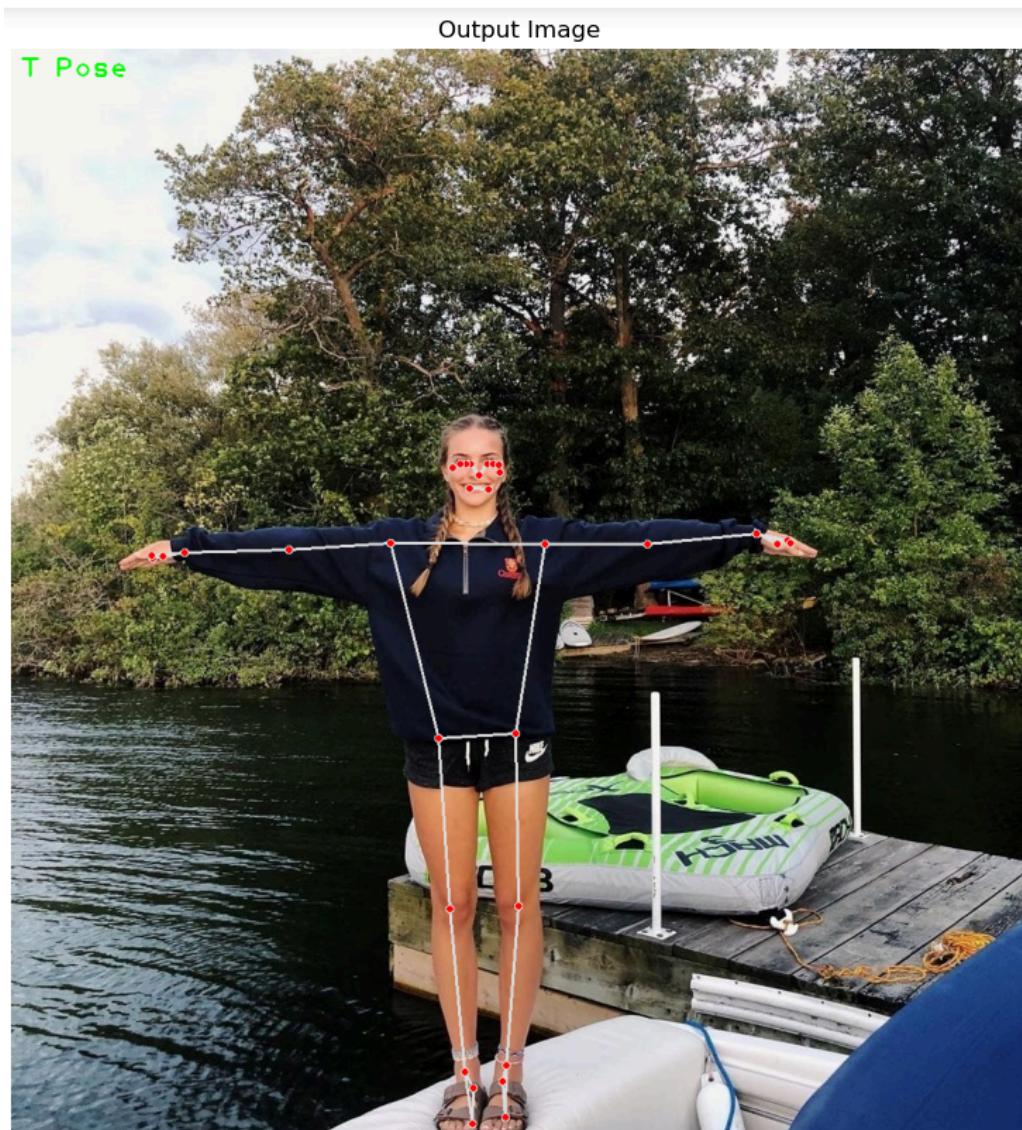
T Pose

T Pose (also known as a bind pose or reference pose) is the last pose we are dealing with in this lesson. To make this pose, one has to stand up like a tree with both hands wide open as branches. The following body part angles are required to make this one:

Around 180° at both elbows

Around 90° angle at both shoulders

Around 180° angle at both knees



6. Result, Discussion and Conclusion

Result

The implementation of the pose detection model using Mediapipe and OpenCV has yielded promising results in both static images and real-time webcam feeds. The visualization of pose landmarks, both in 2D and 3D, provides a comprehensive understanding of the detected key body points. The system's capability to accurately convert normalized landmarks to their original scale enhances the practicality of the self-instruction exercise system.

Moreover, the integration of angle heuristics for pose classification has enabled the system to recognize specific yoga poses, including Warrior II Pose, T Pose, and Tree Pose. The function to calculate angles between landmarks contributes to the accurate classification of yoga poses based on joint angles.

Discussion

While the results showcase the potential of the proposed system, certain limitations must be acknowledged. The dependence on the person facing the camera straight poses a constraint on the system's applicability in dynamic and uncontrolled environments. The accuracy of angle-based pose classification may vary with the angle between the person and the camera, limiting its effectiveness in scenarios where users might not maintain a frontal orientation.

Additionally, real-world scenarios introduce challenges such as occlusion, varying lighting conditions, and potential inaccuracies in pose estimation, which may impact the overall robustness of the system.

Despite these challenges, the system provides a foundation for further research and improvements. Future iterations could explore advanced pose estimation models, address environmental challenges, and enhance the system's adaptability to diverse user scenarios.

Conclusion

In conclusion, this project successfully addresses the problem of at-home exercise form evaluation through the development of a self-instruction exercise system. The implementation leverages cutting-edge technologies, including Mediapipe, OpenCV, and Matplotlib, to perform accurate pose detection and visualization. The integration of angle heuristics for pose classification enhances the system's utility by recognizing specific yoga poses.

While the system demonstrates efficacy in controlled environments, the discussed limitations highlight areas for improvement. Future work should focus on refining pose estimation models, overcoming environmental challenges, and enhancing user adaptability.

Overall, this project lays a strong foundation for the development of intelligent self-instruction exercise systems, contributing to the intersection of computer vision and fitness technology.

6. References

1. 1. L. Sigal. "Human pose estimation", Ency. of Comput. Vision, Springer 2011.
2. 2. S. Yadav, A. Singh, A. Gupta, and J. Raheja, "Real-time yoga recognition using deep learning", Neural Comput. and Appl., May 2019. [Online]. Available: <https://doi.org/10.1007/s00521-019-04232-7>
3. 3. U. Rafi, B. Leibe, J. Gall, and I. Kostrikov, "An efficient convolutional network for human pose estimation", British Mach. Vision Conf., 2016.
4. 4. S. Haque, A. Rabby, M. Laboni, N. Neehal, and S. Hossain, "ExNET: deep neural network for exercise pose detection", Recent Trends in Image Process. and Pattern Recog., 2019.
5. 5. M. Islam, H. Mahmud, F. Ashraf, I. Hossain and M. Hasan, "Yoga posture recognition by detecting human joint points in real-time using Microsoft Kinect", IEEE Region 10 Humanit. Tech. Conf., pp. 668-67, 2017.
6. 6. S. Patil, A. Pawar, and A. Peshave, "Yoga tutor: visualization and analysis using SURF algorithm", Proc. IEEE Control Syst. Graduate Research Colloq., pp. 43-46, 2011.
7. 7. W. Gong, X. Zhang, J. González, A. Sobral, T. Bouwmans, C. Tu, and H. Zahzah, "Human pose estimation from monocular images: a comprehensive survey", Sensors, Basel, Switzerland, vol. 16, 2016.
8. 8. G. Ning, P. Liu, X. Fan and C. Zhan, "A top-down approach to articulated human pose estimation and tracking", ECCV Workshops, 2018.
9. 9. A. Gupta, T. Chen, F. Chen, and D. Kimber, "Systems and methods for human body pose estimation", U.S. patent, 7,925,081 B2, 2011.
10. 10. H. Sidenbladh, M. Black, and D. Fleet, "Stochastic tracking of 3D human figures using 2D image motion", Proc 6th European Conf. Computer Vision, 2000.
11. 11. A. Agarwal and B. Triggs, "3D human pose from silhouettes by relevance vector regression", Intl Conf. on Computer Vision & Pattern Recogn., pp. 882–888, 2004.
12. 12. M. Li, Z. Zhou, J. Li and X. Liu, "Bottom-up pose estimation of multiple persons with bounding box constraint", 24th Intl. Conf. Pattern Recogn., 2018.

13. 13. Z. Cao, T. Simon, S. Wei, and Y. Sheikh, “OpenPose: real-time multi-person 2D pose estimation using part affinity fields”, Proc. 30th IEEE Conf. Computer Vision and Pattern Recogn., 2017.
14. 14. A. Kendall, M. Grimes, R. Cipolla, “PoseNet: a convolutional network for real-time 6-DOF camera relocalization”, IEEE Intl. Conf. Computer Vision, 2015.
15. 15. S. Kreiss, L. Bertoni, and A. Alahi, “PifPaf: composite fields for human pose estimation”, IEEE Conf. Computer Vision and Pattern Recogn., 2019.
16. 16. P. Dar, “AI guardman – a machine learning application that uses pose estimation to detect shoplifters”. [Online]. Available: <https://www.analyticsvidhya.com/blog/2018/06/ai-guardman-machine-learning-application-estimates-poses-detect-shoplifters/>
17. 17. D. Mehta, O. Sotnychenko, F. Mueller and W. Xu, “XNect: real-time multi-person 3D human pose estimation with a single RGB camera”, ECCV, 2019.
18. 18. A. Lai, B. Reddy and B. Vlijmen, “Yog.ai: deep learning for yoga”. [Online]. Available: http://cs230.stanford.edu/projects_winter_2019/reports/15813480.pdf
19. 19. M. Dantone, J. Gall, C. Leistner, “Human pose estimation using body parts dependent joint regressors”, Proc. IEEE Conf. Computer Vision Pattern Recogn., 2013.
20. 20. A. Mohanty, A. Ahmed, T. Goswami, “Robust pose recognition using deep learning”, Adv. in Intelligent Syst. and Comput, Singapore, pp. 93-105, 2017.
21. 21. P. Szczuko, “Deep neural networks for human pose estimation from a very low resolution depth image”, Multimedia Tools and Appl, 2019.
22. 22. M. Chen, M. Low, “Recurrent human pose estimation”, [Online]. Available: [https://web.stanford.edu/class/cs231a/prev_projects_2016/final%20\(1\).pdf](https://web.stanford.edu/class/cs231a/prev_projects_2016/final%20(1).pdf)
23. 23. K. Pothanaicker, “Human action recognition using CNN and LSTM-RNN with attention model”, Intl Journal of Innovative Tech. and Exploring Eng, 2019.
24. 24. N. Nordsborg, H. Espinosa, “Estimating energy expenditure during front crawl swimming using accelerometrics”, Procedia Eng., 2014.
25. 25. P. Pai, L. Changliao, K. Lin, “Analyzing basketball games by support vector machines with decision tree model”, Neural Comput. Appl., 2017.
26. 26. S. Patil, A. Pawar, A. Peshave, “Yoga tutor: visualization and analysis using SURF algorithm”, Proc. IEEE Control Syst. Grad. Research Colloquium, 2011.

27. 27. W. Wu, W. Yin, F. Guo, “Learning and self-instruction expert system for yoga”, Proc. Intl. Work Intelligent Syst. Appl, 2010.
28. 28. E. Trejo, P. Yuan, “Recognition of yoga poses through an interactive system with Kinect device”, Intl. Conf. Robotics and Automation Science, 2018.
29. 29. H. Chen, Y. He, C. Chou, “Computer-assisted self-training system for sports exercise using kinetics”, IEEE Intl. Conf. Multimedia and Expo Work, 2013.
30. 30. Dataset [Online]. Available: <https://archive.org/details/YogaVidCollected>.
31. 31. Y. Shavit, R. Ferens, “Introduction to camera pose estimation with deep learning”, [Online]. Available: <https://arxiv.org/pdf/1907.05272.pdf>.