



1

2

A Real-time Pose Estimation Application for Tinikling

3

4

A Capstone Project on Operational Technologies

5

Presented to the Faculty of the

6

Department of Electronics and Computer Engineering

7

Gokongwei College of Engineering

8

De La Salle University

9

10

In Partial Fulfillment of the

11

Operational Technologies

12

13

by

14

CALAGUIAN Nathan Raekel L.

15

ELLAR Gerald Antonio P.

16

MAHAIT Hans

17

18

19

November, 2025



20

ABSTRACT

21

Index Terms—Dance, Pose Estimation, Real-time, OpenPose .



De La Salle University

22

TABLE OF CONTENTS



De La Salle University

23

LIST OF FIGURES



De La Salle University

24

LIST OF TABLES



De La Salle University

25

ABBREVIATIONS



De La Salle University

26

NOTATION



De La Salle University

27

GLOSSARY



De La Salle University

28

LISTINGS



De La Salle University

29

Chapter 1

30

INTRODUCTION



31 **1.1 Background of the Study**

32 Classical Computer Vision (CV) approaches used skin color segmentation, contour anal-
33 ysis, optical flow, and handcrafted descriptors (Histogram of Oriented Gradients (HOG),
34 motion history images) to detect and classify gestures. Despite being simple and inter-
35 pretable, those methods struggle with background variation and scale. The deep-learning
36 era replaced handcrafted features with Convolutional Neural Network (CNN)s that learn
37 hierarchical visual features directly from image data, yielding much higher accuracy for
38 static hand-pose and short-sequence recognition tasks. Many recent capstone and journal
39 implementations pair OpenCV (for capture/preprocessing) with CNN built and trained
40 in TensorFlow/PyTorch to recognize a fixed vocabulary of gestures in real time. These
41 hybrid pipelines are practical for capstone projects because OpenCV handles efficient frame
42 processing while CNNs provide generalization across users and backgrounds. Furthermore,
43 Operational Technologies plays a crucial role in deploying these systems in real-world
44 applications where physical devices and processes are monitored and controlled, such as
45 in industrial automation or building management systems, which benefit from enhanced
46 gesture recognition. (?)

47 Instead of classifying raw images, several high-performance systems first extract skeletal
48 landmarks (e.g., MediaPipe's 21-point hand model) and feed those coordinates to a classifier
49 (small CNN, MLP, or temporal model like LSTM). Landmark-based pipelines reduce
50 sensitivity to background and scale and make models smaller and faster, which is ideal
51 for mobile or AR deployment. Markerless commercial devices such as the Leap Motion
52 Controller and Ultraleap cameras provide very accurate 3D joint data using IR illumination
53 and multi-camera setups; those give superior fidelity but add hardware cost and integration



54 work. For a capstone aiming at broad deployability, a practical approach is to prototype
55 with MediaPipe + OpenCV + CNN (or lightweight temporal model) and consider Ultraleap
56 integration later for high-precision installations. (?)

57 1.2 Prior Studies

58 Prior research on the topic at hand has shown substantial progress in the integration of pose
59 estimation, computer vision, and interactive technologies for the sake of movement-based
60 learning. For instance, a study by ? presents a human pose estimation method which
61 integrates MediaPipe Pose with additional optimization techniques in order to improve
62 its accuracy and robustness. The designed framework is capable of real-time landmark
63 detection through the use of only a single RGB camera, while optimization methods such
64 as smoothing filters and Kalman filtering are used to reduce jitter and improve the temporal
65 consistency. Results depicted a high detection accuracy for various body parts, with its
66 performance remaining stable under varying lighting and background. This shows Me-
67 diaPipe's suitability for real-time applications where both speed and stability is crucial,
68 especially in aspects such as gesture recognition, sports monitoring, and motion analy-
69 sis. ? explores various deep learning-based human pose estimation techniques and their
70 applications in health, rehabilitation, and human motion analysis. The paper looks into
71 both 2D and 3D pose estimation. It is noted that 2D methods are widely used for real-time
72 applications as they have much lower computational requirements in comparison to 3D.
73 Deep convolutional neural networks and transformer-based models proved to significantly
74 improve the landmark localization accuracy in comparison to classical approaches. Ulti-
75 mately, the paper emphasized that integrating temporal information enhances performance



76 in sequential movement tasks, making these methods highly relevant for motion learning,
77 sports training, and interactive systems. ? focuses on interactive dance learning systems
78 and how such technology has the potential to support dance pedagogy through utilizing
79 real-time feedback and structured interaction workflows. Multiple systems were analyzed
80 and, afterwards, a framework was perfected which made use of motion capture, real-time
81 analysis, and visual feedback in order to support users, who are both learners and instructors.
82 Key interaction patterns were identified such as mirroring, guidance, and correction, which
83 enhances the overall learning experience and, in turn, effectiveness. It also looks into
84 usability considerations such as responsiveness, clarity of feedback, and alignment with
85 existing teaching approaches, which is relevant to the creation of dance learning systems.
86 Ultimately, such studies depict the intersection of pose estimation, feedback systems, and
87 immersive interfaces, which lays a strong groundwork for future developments in digital
88 dance education and interactive movement learning systems.

89 **1.3 Problem Statement**

90 To this day, the national dance of the Philippines known as ‘Tinikling’ continues to hold
91 cultural significance among students, educators, and dance enthusiasts. However, despite its
92 importance, those that aspire to learn the dance lack access to physical classes or qualified
93 instructors be it due to geographical or time constraints. Existing methods of learning
94 may be costly or unable to provide feedback to the student in real-time, which makes the
95 learning process difficult for individuals in terms of practicing effectively on their own.
96 Such a gap highlights the need for a much more accessible, interactive, and accurate tool
97 which would be able to guide learners remotely in an efficient manner and, ultimately,



De La Salle University

98 ensuring that tradition is preserved and passed on to future generations.

99 **1. PS1:**

- 100 • The ideal scenario for our intended audience (students, educators, and dance
101 enthusiasts) is to have an intuitive and interactive learning tool that facilitates
102 the practice of Tinikling, the traditional Filipino dance. This tool should provide
103 real-time feedback on users' dance movements, enabling them to learn and
104 improve their technique. The desired state includes accessibility to the tool on
105 various devices (e.g., desktop, mobile) with a user-friendly interface and a high
106 level of accuracy in tracking the dance steps. Additionally, it should support
107 personalized feedback, enabling users of all skill levels to progress and feel
108 engaged in learning this cultural heritage.

109 **2. PS2:**

- 110 • Currently, learning Tinikling requires access to physical dance classes or in-
111 structors, which are often limited by geographical location, financial resources,
112 or time constraints. For individuals unable to attend such classes, the lack of af-
113 fordable and effective learning tools becomes a significant barrier. Additionally,
114 existing dance-learning technologies are either costly, relying on specialized
115 hardware, or lack the immediacy of real-time feedback, making it difficult
116 for learners to practice and perfect their movements without direct instructor
117 guidance.
- 118 • The pain point is that students who want to practice Tinikling at home or in
119 remote areas are unable to receive real-time guidance or feedback, leading to



120 slower progress, incorrect technique, and a loss of motivation.

121 **3. PS3:**

- 122 • Without a tool that offers immediate feedback and a clear learning path, students
123 practicing Tinikling on their own are likely to struggle with incorrect move-
124 ments, which may lead to frustration. Over time, this lack of progress could
125 result in a lack of confidence, disengagement from the learning process, and
126 ultimately, the inability to learn the dance correctly. Furthermore, the absence
127 of accessible learning tools risks the loss of cultural knowledge and the fading
128 of the Tinikling tradition, especially among younger generations who may not
129 have easy access to traditional learning methods.

130 **1.4 Objectives and Deliverables**

131 **1.4.1 General Objective (GO)**

- 132 • GO: To design and implement a real-time Pose estimation-based Tinikling learning
133 application;

134 **1.4.2 Specific Objectives (SOs)**

- 135 • SO1: To develop a real-time pose estimation pipeline that captures dancers' move-
136 ments using a webcam, detects key skeletal landmarks, and analyzes Tinikling steps
137 with at least 30 frames per second (fps) processing speed and $\geq 70\%$ detection
138 accuracy.;;



- 139 • SO2: To make the pose estimation model robust to lighting, background clutter,
140 and user variation through dataset collection and augmentation and, landmark-based
141 representations while maintaining a minimum pose detection accuracy of 85% ;

142 • SO3: To design and integrate a scoring and feedback system that evaluates user perfor-
143 mance by aligning poses with reference choreographies, providing numerical scores
144 (0–100) and step-by-step accuracy breakdown within 1 second after performance.;

145 • SO4: To evaluate the system’s performance and usability through controlled test-
146 ing with at least 3 participants, measuring pose estimation accuracy, latency, and
147 user satisfaction ($\geq 80\%$ positive feedback) using standardized questionnaires and
148 performance metrics.;

149 **1.4.3 Expected Deliverables**



TABLE 1.1 EXPECTED DELIVERABLES PER OBJECTIVE

Objectives	Expected Deliverables
GO: To design and implement a real-time Pose estimation-based Tinikling learning application	<ul style="list-style-type: none"> • Prototype of Tinikling learning application. • Documentation and user manual.
SO1: To develop a real-time pose estimation pipeline that captures dancers' movements using a webcam, detects key skeletal landmarks, and analyzes Tinikling steps with at least 30 frames per second (fps) processing speed and $\geq 70\%$ detection accuracy.	<ul style="list-style-type: none"> • Optimized skeletal keypoints detection for Tinikling steps. • Implementation of webcam-based pose estimation pipeline. • Performance evaluation results.
SO2: To make the pose estimation model robust to lighting, background clutter, and user variation through dataset collection and augmentation and, landmark-based representations while maintaining a minimum pose detection accuracy of 85%	<ul style="list-style-type: none"> • Augmented dataset covering varied lighting, backgrounds, and user types. • Enhanced landmark-based model with robustness improvements. • Comparative performance evaluation report.
SO3: To design and integrate a scoring and feedback system that evaluates user performance by aligning poses with reference choreographies, providing numerical scores (0–100) and step-by-step accuracy breakdown within 1 second after performance.	<ul style="list-style-type: none"> • Scoring and feedback algorithm. • Tinikling choreography database. • Post-performance scoring output with accuracy metrics.
SO4: To evaluate the system's performance and usability through controlled testing with at least 3 participants, measuring pose estimation accuracy, latency, and user satisfaction ($\geq 80\%$ positive feedback) using standardized questionnaires and performance metrics.	<ul style="list-style-type: none"> • Conducted controlled testing with participants. • Collected performance and usability metrics. • Evaluation report with recommendations for improvement.



150 **1.5 Significance of the Study**

151 This capstone project focuses on the development of a Tinikling learning application
152 through the integration of pose estimation and human action recognition. The setup consists
153 of a webcam, laptop, and two bamboo sticks for the Tinikling dance. Such a setup offers
154 affordability and accessibility benefits for users. Ultimately, it contributes to the field
155 of both pose estimation and human action recognition by demonstrating a successful
156 integration of the two in a live setup.

157 **1.5.1 Technical Benefit**

- 158 1. Enables real-time pose estimation and post-performance feedback, improving accu-
159 racy and efficiency throughout the learning process.
- 160 2. Low-cost software-based learning tool which uses a webcam and desktop computer
161 rather than expensive motion capture equipment.

162 **1.5.2 Social Impact**

- 163 • Promotes cultural preservation by making Tinikling more accessible through interac-
164 tive applications.
- 165 • Increases student engagement and participation via gamified learning.
- 166 • Supports remote or in-classroom instruction by enabling technology-assisted dance
167 education.



168 **1.5.3 Environmental Welfare**

- 169 • Utilizes existing and widely available hardware such as webcams and desktop computers rather than new specialized equipment, which ultimately lessens electronic
170 waste.
- 172 • Encourages digital preservation of cultural heritage, lessening reliance on physical
173 materials or infrastructure.

174 **1.6 Assumptions, Scope, and Delimitations**

175 **1.6.1 Assumptions**

- 176 1. Pose landmarks from webcams with standard RGB resolutions such as 720p, 1080p,
177 and 4K or low-cost depth sensors provide sufficient fidelity to represent Tinikling
178 movements for temporal alignment and scoring.
- 179 2. Choreography can be divided into short, labeled segments that enable reliable match-
180 ing and targeted feedback.
- 181 3. Dynamic Time Warping or a constrained variant will handle tempo variation robustly
182 for temporal alignment.
- 183 4. A brief per-user calibration step will improve scoring consistency.

184 **1.6.2 Scope**

- 185 1. Cover automatic pose estimation, sequence alignment, and segment-level scoring for
186 Tinikling.



- 187 2. Accept landmark or depth inputs and provide immediate on-device cues during
188 performance.
- 189 3. Produce a higher-precision final score after a more detailed pass.
- 190 4. Use self-sourced Tinikling videos for model training when no public dataset exists.
- 191 5. Benchmark against general dance datasets where appropriate.
- 192 6. Report sensor-based metrics and simple user measures such as perceived accuracy
193 and engagement.

194 1.6.3 Delimitations

- 195 1. Will not perform detailed facial or hand mesh reconstruction.
- 196 2. Will not replace multi-camera motion capture for research-grade kinematics.
- 197 3. Will not guarantee reliable results under heavy occlusion, very low light, extreme
198 off-axis views, or when clothing blends with the background.
- 199 4. Will not attempt full generalization to all body shapes without additional data and
200 tuning.
- 201 5. Limits reflect known sensor and algorithm constraints and the aim to produce a
202 practical, lightweight prototype.



1.7 Description and Methodology of the Capstone Project on Operational Technologies

1. Phase 1: Model Development serves as a precursor for Phase 2 wherein the specifics of the model, libraries, and environment to use are defined. In total, Phase 1 would last 4 weeks spanning from week 4 to 7. The bulk of the research for the project would be carried out during this phase. The dataset to be used for training would be collected during this phase as well.
2. Phase 2: Model Training consists of training the model using the dataset collected in the previous phase. This phase will largely consist of testing and improving the resulting model. Tests would be conducted using the group members as dancers. This phase also includes the optimization of the model for real-time detection simultaneously with the music. In total, this phase would last 4 weeks spanning from week 8 to 11.
3. Phase 3: UI/UX Development consists of the integration of the trained model with a user interface. Once integrated final testing and refinement of the final program would be carried out. The final output would be presented as well during this phase along with the finalization of the documentation. This phase would last for 3 weeks spanning from week 11 to 13.

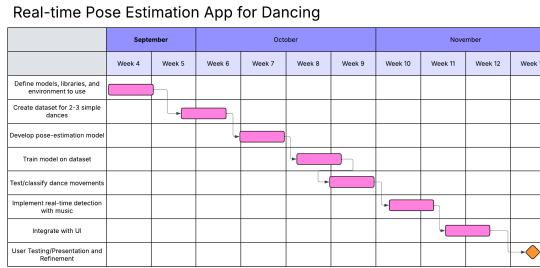


Fig. 1.1 Milestone Gantt Chart for Real-time Pose Estimation Dance Software

221

1.8 Estimated Work Schedule and Budget

222

1.8.1 Milestones and Gantt Chart

223

1.8.2 Budget

224

Given that the capstone project largely consists of software, apart from the use of a laptop for both programming, as well as actual implementation and usage of the dance program, the only expense to consider would be for that of a Webcam, which is already owned.

225

226

TABLE 1.2 OPERATIONAL FINANCIAL PLAN

Item	Price
Webcam	P1,850
4pc. Tinikling Sticks	P110
Total	P1,960



1.9 Overview of the Capstone Project on Operational Technologies

This capstone project focuses on developing a real-time pose estimation-based learning application for Tinikling, the Philippine national dance. It integrates computer vision and machine learning techniques in order to create an interactive learning platform that provides performance scoring to users. The project utilizes webcams and MediaPipe-based skeletal landmark extraction to analyze users' movements relative to reference choreography. Unlike expensive motion capture systems, this setup uses low-cost and accessible hardware, making the system practical for classroom, cultural, and home use. The system emphasizes cultural preservation by modernizing Tinikling education through technology. It enables students to learn and practice the dance interactively, provides technical benefits such as real-time feedback without costly sensors, and supports social and environmental goals through cultural engagement and sustainable use of existing hardware.



De La Salle University

240

Chapter 2

241

LITERATURE REVIEW



2.1 Existing Work

A study by ? focused on surveying the various existing 3D human body pose and shape estimation techniques, given its crucial nature in fields such as augmented or virtual reality, healthcare and fitness technology, and virtual retail. The solutions explored consisted of mainly three types of inputs, which were single images, multi-view images, and videos. Various issues pertaining to dance, such as fast motion, occlusion, and unusual poses were analyzed to see how each affected the performance of each method. The specific models consisted of SMPL -A, SMPL -X, MANO, STAR, FLAME, which are optimization-based models, as well as HMR, VIBE, SPIN, PARE, EXPOSE, and PHALP, which are deep learning-based models. SMPL was found to be beneficial in terms of realistic body representation, efficiency for real time applications, and wide availability, however it has limitations in areas pertaining to facial and hand modeling, as well as representation of ethnic diversity. SMPL -X proved to provide several advantages such as facial expressions, hand gestures, and improved expressiveness. Its limitations, however, consisted of simplified hand modeling and its limited pose variability. MANO offers detailed hand gesture modeling and realistic hand deformations, but has limitations due to its focus being exclusively on the modeling of hands, as well as computational challenges. STAR leverages sparse coding and temporal modeling, which allowed for a much more powerful framework for pose estimation., depicting state-of-the-art results throughout various benchmarks and practical implementations in sports analysis, human-computer interaction, and VR. FLAME was advantageous when it comes to computational efficiency, which made it suitable for real-time applications of pose estimation. As for its limitations, it primarily focuses on facial and lip modeling, which introduces complexity and potential computational chal-



De La Salle University

lenges. MANO. HMR produces richer and more useful mesh representation, which is parameterized by shape and 3D joint angles. The network implicitly learns the angle limits of each joint. As such its use is discouraged for people with unusual body shapes. Its re-projection loss is highly under-constrained and it needs adversarial supervision in order to avoid unrealistic outputs. VIBE makes use of CNNs, RNNs and GANs, as well as a self-attention layer in order to achieve state-of-the-art results. A motion discriminator is used to help produce more realistic motion. Ultimately, the model is a standard SMPL body model format with sequences of poses and shape parameters. SPIN makes use of a self improving loop wherein better fits allow the network to train in a much more efficient manner while better initial estimates from the network aids the optimization routine in order to result in better fits. PARE consists of a guided attention mechanism which exploits information on visibility of individual body parts all the while leveraging information from neighboring body parts in order to predict parts which are occluded. EXPOSE includes body, face, and hand estimation. It is able to estimate expressive 3D humans in a much more accurate manner in comparison to existing optimization methods at only a fraction of the computational costs. PHALP out performs all of the aforementioned methods. Despite this, it still has its limitations as well such as its reliance on a single camera, which may lead to issues such as occlusion and motion blur. It may also not work well in low-light conditions or when a person's clothes is a similar color to that of the background. Lastly, it also requires a significant amount of computational resources, which may make it not suitable for real-time applications.

A study by ?, analyzes the effectiveness of various classification techniques in recognizing different dance types based on motion-capture skeleton data. Classifiers explored consisted of k-Nearest Neighbors (k-NN), Naïve Bayes, Discriminant Analysis, Classi-



De La Salle University

289 fication Trees, Random Forests (TreeBagger), Support Vector Machines (SVMs), and
290 Ensemble Classifiers. Poses are identified through the use of body joints via Kinect sensor.
291 The data set used consisted of various dances such as Enteka, Kalamatianos, Syrtos (Two-
292 beat), Sytros (Three-beat). The kinect was used to capture skeletal joining data, to which
293 feature extraction techniques such as principal component analysis and frame differencing
294 were used in order to improve the classification accuracy. Ultimately, results showed that
295 k-nearest neighbors and random forests are the best-performing classifiers among those
296 that were explored. It was also proposed that the use of mulit-sensor or multimodal data
297 may serve as a potential solution for issues specific to pose recognition in dance such as
298 occlusion and complex movement patterns.

299 A study by ?, looks into dance pose estimation and introduces the model DanceFormer.
300 DanceFormer is a transformer-based model for dance pose estimation which makes use
301 of the Vision Transformer, Time Series Transformer, and an edge computation layer in
302 order to achieve a deep fusion of multimodal features and to overall increase its accuracy
303 and real-time performance. The AIST and DanceTrack datasets were used throughout
304 the experimentation. Results showed that DanceFormer out performs other models, with
305 it achieving a pose estimation accuracy or MPJPE of 18.4mm and 20.1mm, as well as
306 a multi-object tracking accuracy or MOTA of 92.3% and 89.5%. It is also suitable for
307 real-time processing in even low-resource with an average latency of 35.2ms. Ultimately, it
308 serves as an efficient, precise and real time solution for rather complex dance scenarios. It
309 also has applications in a much more broad sense be it in dance education or in real-time
310 motion analysis.

311 A study by ? discusses dance movement recognition based on gesture. A low accuracy
312 traditional dance movement recognition algorithm based on human posture estimation



De La Salle University

313 was proposed. PAFs algorithm was used in order to recognize the spatial skeleton nodes
314 and connections of joints in the human body. The pose of the body is estimated based
315 on the movement of the spatial skeleton. Once the information on the detected posture
316 is preprocessed and its features are extracted, LTSM time series algorithm was used in
317 order to classify and recognize certain dance movements. Ultimately, results showed that
318 the proposed algorithm has the capacity to reliably identify dance movements based on
319 the skeleton nodes. It was able to achieve a recognition accuracy and recall rate upwards
320 of 85% for the different movement categories. As for its recognition accuracy of curtsey
321 movement, it achieved upwards of 95.2%.

322 ? present a detailed evaluation of Kinect v1, Kinect v2, and Azure Kinect skeleton
323 tracking, analyzing joint-level error distributions and repeatability across distances and
324 orientations. Their results highlight degradation in accuracy under occlusion, off-axis
325 angles, and larger working distances, conditions typical of casual living-room dance setups.
326 The findings underline both the potential and the limits of Kinect-class sensors, suggesting
327 that practical applications often require either sensor fusion and smoothing to handle jitter
328 or a focus on more reliable joints for robust real-time scoring.

329 ? investigate how interactive feedback design influences user motivation in the context
330 of Just Dance. Their study demonstrates that timely, clear cues significantly improve
331 engagement, perceived competence, and sustained participation, with direct effects on
332 physical activity outcomes. These findings show that feedback modalities and latency
333 are as critical as recognition accuracy in shaping the player experience, emphasizing
334 the importance of immediate, multimodal responses in dance or pose-based teaching
335 applications.

336 ? propose and validate a Dynamic Time Warping method for evaluating rehabilitation



De La Salle University

337 exercises tracked with Kinect. Their algorithm successfully aligns noisy, tempo-varying
338 motion with reference trajectories, producing reliable correctness scores even with partial
339 occlusion. Applied to dance or short choreographies, DTW offers a robust foundation for
340 handling tempo shifts and timing variation, supporting sequence-based scoring that is more
341 forgiving than strict frame-to-frame comparison.

342 ? compare Kinect II with the high-precision Vicon system in the context of choreogra-
343 phy retrieval and analysis, using trajectory similarity measures such as DTW. While Kinect
344 data contain noise and smoothing artifacts, the study shows that trajectory-level patterns
345 remain useful when algorithms are designed to tolerate sensor bias. Their results support
346 the use of low-cost consumer sensors, including RGB landmark pipelines, in applications
347 where robust temporal alignment and trajectory modeling can offset hardware limitations.

348 Human pose estimation (HPE) has become an important area of study due to its applica-
349 tions in action recognition, sports, and performing arts. Xu, Zou, and Lin (2022) introduced
350 the Adaptive Hypergraph Neural Network (AD-HNN), which captures high-order semantic
351 dependencies among joints to improve multi-person pose estimation, particularly in han-
352 dling occlusion and pose variability. In dance analysis, Ju (2025) applied deep learning with
353 ResNet-152 and HR-Net to enhance dance pose recognition, addressing class imbalance
354 and improving classification accuracy through global-local feature fusion.

355 For cultural preservation, motion capture (MoCap) has been widely adopted. Rizhan
356 et al. (2025) demonstrated the use of MoCap to develop authentic motion templates for
357 Malay folk dances, ensuring accuracy and authenticity in preserving intangible cultural
358 heritage. In addition, Büyükgökoğlan and Uğuz (2025) developed a performance evaluation
359 system for Turkish folk dances using deep learning-based pose estimation (e.g., Mediapipe,
360 YOLO, LSTM), enabling objective assessment compared to traditional jury scoring.



TABLE 2.1 SUMMARY OF REVIEWED DANCE POSE ESTIMATION AND RECOGNITION STUDIES

Paper	Focus	Methodology	Results
<i>Venkatrayappa et al. (2024)</i>	Evaluates 3D human pose & shape estimation techniques for dance	PHALP (multi-frame 3D pose estimation)	N/A
<i>Protopapadakis et al. (2018)</i>	Identifies dance types using skeletal data	k-NN classifier on PCA-reduced Kinect skeleton features	Accuracy = 0.52
<i>Zhao et al. (2025)</i>	Seeks accurate, real-time pose estimation for complex dances	Hybrid Vision + Time-Series Transformer (DanceFormer)	MPJPE = 18.4/20.1 mm; MOTA = 92.3% / 89.5%; Latency = 35.2 ms
<i>Lei et al. (2023)</i>	Improves low-accuracy traditional-dance recognition methods	PAF-based keypoint detection + LSTM classifier	>85% overall; 95.2% (curtsey)
<i>Ju (2025)</i>	Proposes deep-learning methods to design & recognize dance poses	ResNet-152 + HRNet (global-local feature fusion)	Accuracy = 0.9870; Precision = 0.9851; Kappa = 0.9841
<i>Xu et al. (2022)</i>	Estimates multiple human poses from single images using an adaptive structure	Adaptive Hypergraph Neural Network (AD-HNN)	AP = 76.6% (COCO)
<i>Tölgessy et al. (2021)</i>	Evaluates joint-level accuracy and repeatability across Kinect sensors	Kinect V1 / V2 / Azure skeleton-tracking evaluation	Std. Dev. = 0.8–1.9 mm; Joint misses = 15–30%
<i>Yu & Xiong (2019)</i>	DTW-based scoring for Kinect-based rehabilitation/exercise	DTW-based scoring of Kinect-derived skeleton motions	Pearson r = 0.86
<i>Rallis et al. (2019)</i>	Choreography pattern analysis (Kinect vs Vicon)	DTW trajectory matching (Kinect II vs Vicon)	N/A
<i>Sun & Song (2025)</i>	Pose estimation in complex dance scenes	Improved HRNet + CBAM attention + multi-scale fusion	Accuracy = 73.5% (MPII); 79.5% (dance dataset)
<i>Büyükgökoglan & Uğuz (2025)</i>	Deep-learning-based scoring for Turkish folk dance	MediaPipe / YOLO pose extraction + LSTM scoring	LSTM = 68.43 (MSE = 56.11); DTW = 60.64 (MSE = 139.32)

2.2 Lacking in the Approaches

These studies show the potential of pose estimation and deep learning for advancing both modern dance movement design and traditional folk dance preservation. However, there is little to no research in the Philippines that applies pose estimation to folk

361

362

363

364



365 dances—particularly Tinikling—representing a significant gap and opportunity for future
 366 exploration.

TABLE 2.2 MOVEMENTS / BODY PARTS DETECTED AND LIMITATIONS OBSERVED IN
 REVIEWED APPROACHES

Author	Body Part Detected	Lacking in Approaches
Venkatrayappa <i>et al.</i> (2024)	Full body with 3D body mesh and joints	Single-frame methods fail on fast, complex dance motion; multi-frame approaches are needed.
Protopapadakis <i>et al.</i> (2018)	Upper and lower body joints	Designed to track frontal views only; front/back ambiguity and limited movement-range handling.
Zhao <i>et al.</i> (2025)	Full body	Sensitive to occlusion and heavy background clutter; requires sizable compute for real-time feedback.
Lei <i>et al.</i> (2023)	Full body	Struggles with inter-subject variability and scale changes.
Ju (2025)	Full body	Heavy reliance on large, well-labelled datasets and computationally heavy models.
Xu <i>et al.</i> (2022)	Multi-person body keypoints	Adaptive-hypergraph complexity can be computationally heavy and harder to deploy in real time.
Tölgessy <i>et al.</i> (2021)	Full joint skeleton	Sensor-based skeleton tracking misses joints under occlusion, degrades with distance, and shows inter-device variance.
Yu & Xiong (2019)	Major limb movement trajectories	DTW scoring is sensitive to temporal misalignment and sensor noise.
Rallis <i>et al.</i> (2019)	Full body with 3D skeleton	Low-cost sensors (e.g., Kinect) have limited spatial fidelity vs. motion-capture rigs; trajectories are noisier.
Sun & Song (2025)	Full body with skeleton	Improved HRNet variants remain affected by background interference, occlusion, and scale sensitivity.
Büyükgökoglan & Uğuz (2025)	Upper and lower body keypoints	Scoring is vulnerable to per-performer style variation and dataset bias.

2.3 Summary

367
 368 Research on human pose estimation (HPE) spans multiple applications including AR/VR,
 369 healthcare, and dance. Optimization- and deep learning-based models (e.g., SMPL, SMPL-
 370 X, HMR, VIBE, SPIN, PARE, EXPOSE, PHALP) have been studied for realistic 3D
 371 body reconstruction (Venkatrayappa *et al.*, 2024). Dance classification has been explored
 372 using skeleton data and machine learning classifiers like k-NN and Random Forest (Pro-
 373 topapadakis *et al.*, 2018). Transformer-based models such as DanceFormer achieve high
 374 accuracy and real-time performance in dance pose estimation (Zhao *et al.*, 2025), while



375 PAF- and LSTM-based algorithms improve movement recognition (Lei et al., 2023). Kinect
376 studies reveal both potential and limits in low-cost motion capture (Tölgessy et al., 2021;
377 Rallis et al., 2019), while feedback and sequence-alignment approaches (Lin et al., 2015;
378 Yu & Xiong, 2019) highlight the importance of interactivity and temporal robustness.
379 Recent work integrates advanced neural networks for pose estimation, such as adaptive
380 hypergraphs (Xu et al., 2022), deep feature fusion for dance poses (Ju, 2025), MoCap
381 for authentic folk dance templates (Rizhan et al., 2025), and deep learning systems for
382 evaluating Turkish folk dance (Büyükgökoğlan & Uğuz, 2025).



383

Chapter 3

384

THEORETICAL CONSIDERATIONS



385 **3.1 Human Pose Estimation**

386 Human pose estimation is the process of predicting the pose of human body parts. The data
387 are typically derived from RGB images or videos. Given that certain motions are motivated
388 by human actions, detecting poses is a critical aspect of human action recognition (Song
389 et al., 2021). It has a wide range of applications such as human-computer interaction,
390 motion analysis, augmented reality, and virtual reality. The resulting output of human pose
391 estimation is a skeleton-like representation of the human body consisting of nodes and
392 limbs (Zheng et al., 2020). There are two main types of human pose estimation, namely 2D
393 and 3D. 2D pose estimation consists of predicting the posture of each of the body's key
394 points in a 2D plane, considering the X and Y axes. As for 3D pose estimation, it considers
395 the Z axis, situating each point in a 3D space. It goes without saying that 3D estimation is
396 more difficult in comparison to 2D estimation in process and complexity due to underlying
397 issues such as noisy backgrounds, clothing, lighting, undetected joints, or occlusion (Ben
398 Gamra & Akhloufi, 2021).

399 **3.2 Human Action Recognition**

400 Human action recognition (HAR) is the process of detecting human actions to classify
401 them through single-sensor data, RGB image or video data, or three-dimensional depth
402 and inertial data (Sakar et al., 2022). In the field of computer vision, one of the most
403 challenging aspects is the automatic and precise identification of human activity. Over the
404 years, there has been a significant increase in feature learning-based representations for
405 human action recognition as a result of the widespread utilization of deep learning-based
406 features. There are various applications of HAR — for instance, automated surveillance



De La Salle University

407 systems that make use of AI and machine learning algorithms to identify human actions
408 for safety and security. Such tasks, however, are made difficult due to factors such as
409 changing environments, occlusion, different viewpoints, execution pace, and biometric
410 variation. Furthermore, the human body varies from person to person in factors such as size,
411 appearance, and shape. However, advancements in Convolutional Neural Networks (CNNs)
412 have resulted in significant progress in human action recognition through improvements
413 in classification, segmentation, and object detection. This largely applies to image-related
414 tasks rather than videos, as neural network models struggle to capture temporal information
415 in videos due to the lack of substantial datasets (Morshed et al., 2022).



416

Chapter 4

417

DESIGN CONSIDERATIONS



418 4.1 Sensor Choice, Representation, and Robustness

419 A study by Tölgessy, Dekan, and Chovanec (2021) demonstrated that Kinect-family depth
420 sensors produce explicit 3D skeletons and give higher joint fidelity in controlled settings,
421 but the accuracy falls with occlusion, off-axis views, and increased distance. Zhang et
422 al. (2020) described MediaPipe, which yields compact 2D/3D landmark coordinates from
423 ordinary RGB cameras and runs in real time on mobile devices. Therefore, designers often
424 choose landmarks for rapid, lightweight prototypes and mobile deployment, and reserve
425 depth or IR systems for installation-grade fidelity when hardware is available. To reduce
426 real-world failure modes, practitioners apply photometric and background augmentation
427 and synthetic occlusions during training, and they add a short calibration step so system
428 metrics align with an individual user's range of motion.

429 4.2 Temporal Alignment and Scoring

430 Dance is a temporal activity and should be compared as a sequence rather than as isolated
431 frames. Yu and Xiong (2019) demonstrate that Dynamic Time Warping (DTW) can align
432 noisy, tempo-varying Kinect skeleton sequences and convert DTW distances into mean-
433 ingful performance scores. Rallis et al. (2019) apply DTW to choreographic trajectories
434 and show it can match patterns across high-precision (VICON) and low-cost (Kinect)
435 capture systems. Thus, a practical scoring pipeline first aligns sequences with DTW (or a
436 constrained variant) and then evaluates local spatial metrics such as joint-angle differences
437 or normalized trajectory distances to produce interpretable, per-segment correctness scores.



438 **4.3 Real-Time Feedback, Segmentation, and Peda-** 439 **gogy**

440 Lin (2015) finds that immediate, clear feedback in dance exergames improves engagement
441 and supports learning. Zhang et al. (2020) show that on-device landmark extraction can
442 run at real-time rates suitable for low-latency feedback. Combining these results suggests
443 a two-tier runtime design: use a fast, coarse matcher (enabled by on-device landmarks)
444 for instant cues, and run a slower, higher-precision alignment and scoring pass for final
445 grading. Breaking choreography into short labeled segments also simplifies alignment and
446 reduces error accumulation; Rallis et al. (2019) illustrate that segment- or trajectory-level
447 matching better supports choreographic retrieval and per-segment feedback.

448 **4.4 Accessibility, Personalization, and Evaluation**

449 Yu and Xiong (2019) convert DTW distances into calibrated percentage scores, which
450 supports per-user calibration and comparison against an individualized baseline. Tölgessy
451 et al. (2021) recommend measuring sensor-level metrics such as joint error and dropout rates
452 when choosing a capture modality. Therefore, system designs should include adjustable
453 sensitivity, alternate gesture mappings, and user profiles, and evaluation should combine
454 sensor metrics (joint error, dropout, latency) with human-centered measures (perceived
455 accuracy, engagement, and learning gain) to justify architecture and scoring choices.



TABLE 4.1 TECHNICAL STANDARDS (ME) – SCOPE AND COMPLIANCE JUSTIFICATION

Standard / Regulation	Scope of Use in the System	Compliance Justification
<i>ISO 9241-210: Human-centered system design</i>	UI design and user interaction	Ensures user comfort and reduces fatigue during dance learning.
<i>IEEE 802.11: Wi-Fi communication</i>	If remote database or cloud storage is used	Ensures interoperability and stable streaming between client and remote endpoints.
<i>ISO 27001: Data privacy & security</i>	Storage and handling of video recordings	Prevents unauthorized access to personal video data and enforces secure storage practices.
<i>ISO 25010: Software quality characteristics</i>	Reliability, maintainability, usability	Used as a quality benchmark during evaluation and acceptance testing.
<i>IEEE 754: Floating-point calculations</i>	Pose and angle computations	Ensures mathematical consistency and predictable numerical behaviour across platforms.

TABLE 4.2 ENVIRONMENTAL & SAFETY STANDARDS AND THEIR APPLICATION IN THE PROJECT

Standard / Regulation	Application
<i>RA 9003: Ecological Solid Waste Management Act</i>	Limits hardware waste; project reuses existing webcams and peripherals where possible to reduce e-waste and disposal burden.
<i>ISO 14001: Environmental Management System</i>	Guides procurement and lifecycle decisions to ensure minimal environmental impact when selecting cameras, computers, and consumables.
<i>ISO 45001: Occupational health & safety</i>	Protects users and participants performing physical activity by mandating risk assessment, safe spaces (non-slip flooring), and emergency procedures.
<i>IEC 60950-1: IT equipment electrical safety</i>	Ensures safe usage of laptops, webcams, power supplies, and peripherals during prolonged sessions to prevent electrical hazards.



456

Chapter 5

457

METHODOLOGY



458

5.1 Methodology

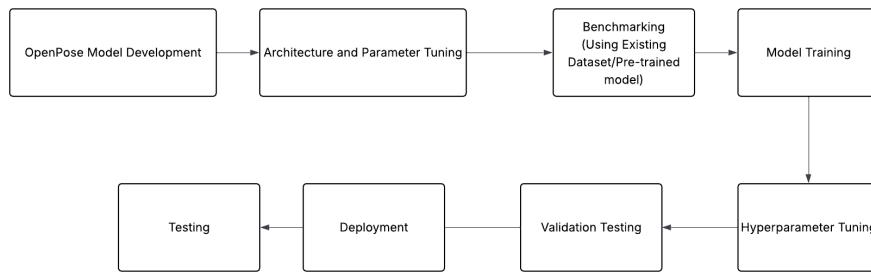


Fig. 5.1 Methodology Block Diagram

459

5.1.1 Methodology Overview

460

This project develops a desktop real-time pose-estimation application for Tinikling learning. The pipeline comprises (1) dataset collection and annotation, (2) real-time landmark detection using MediaPipe with OpenCV preprocessing, (3) model robustness improvements via augmentation and fine-tuning, (4) a per-segment scoring and feedback engine, and (5) system evaluation and user studies for performance and usability.

461

462

463

464

TABLE 5.1 SUMMARY OF METHODS FOR REACHING THE OBJECTIVES

Objectives	Methods	Locations
GO: To develop a real-time pose estimation-based Tinikling learning application.	<ol style="list-style-type: none"> Develop a desktop application integrating pose estimation, scoring, and feedback modules. Utilize MediaPipe + OpenCV for pose detection, integrated with a GUI framework. Document architecture, usage, and installation following software engineering practices. 	N/A

Continued on next page



Table 5.1 (continued)

Objectives	Methods	Locations
SO1: To develop a real-time pose estimation pipeline that captures dancers' movements using a webcam, detects key skeletal landmarks, and analyzes Tinikling steps with ≥ 30 fps processing speed and $\geq 70\%$ detection accuracy.	<ol style="list-style-type: none"> 1. Use MediaPipe Pose for skeletal landmark detection in real time. 2. Optimize frame processing via OpenCV preprocessing and efficient landmark extraction. 3. Evaluate detection accuracy using collected test sequences and performance metrics. 	$\geq 90\%$ detection accuracy; 30 fps
SO2: To make the pose estimation model robust to lighting, background clutter, and user variation through dataset collection and augmentation, while maintaining minimum pose detection accuracy of 85%.	<ol style="list-style-type: none"> 1. Collect / create Tinikling dance videos under diverse lighting, backgrounds, and performer variations. 2. Apply data augmentation (photometric, geometric, synthetic occlusions). 3. Retrain / fine-tune the model and evaluate on a validation set to quantify improvements. 	$\geq 85\%$ detection accuracy
SO3: To design and integrate a scoring and feedback system that aligns poses with reference choreographies, provides numerical scores (0–100) and step-by-step accuracy breakdown within ≤ 1 s after performance.	<ol style="list-style-type: none"> 1. Implement per-segment accuracy scoring (DTW or constrained alignment + local spatial metrics). 2. Build a choreography reference library with segmented Tinikling steps for alignment. 3. Integrate UI feedback: immediate cues and post-performance breakdown. 	Score range 0–100; feedback latency ≤ 1 s
SO4: To evaluate the system's performance and usability through controlled testing with at least 3 participants, measuring pose estimation accuracy, latency, and user satisfaction ($\geq 80\%$ positive feedback).	<ol style="list-style-type: none"> 1. Conduct user testing sessions with participants performing choreographed sequences. 2. Measure pose estimation accuracy, system latency, and feedback timing. 3. Compile results into an evaluation report with recommendations for refinement. 	$n \geq 10$ participants; $\geq 80\%$ positive feedback

465

5.1.2 Dataset Collection and Annotation

466

We collect Tinikling performances using consumer webcams across varied environments (lighting, backgrounds, participant clothing). Each recording is annotated with segment boundaries and ground-truth reference trajectories for the core Tinikling steps. Annotation files follow a simple CSV schema: frame index, timestamp, keypoint coordinates (x,y[,z if available]), and segment label.

467

468

469

470

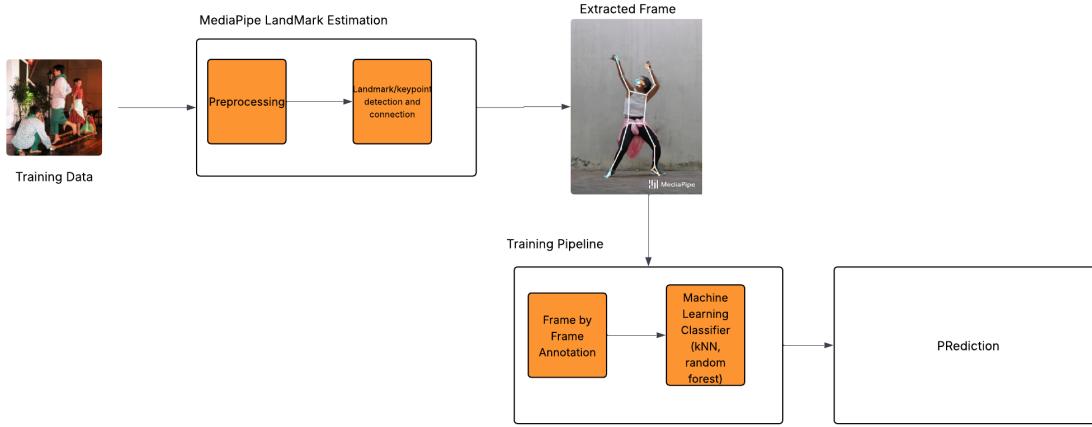


Fig. 5.2 System Diagram of the Real-time Tinikling Learning Application

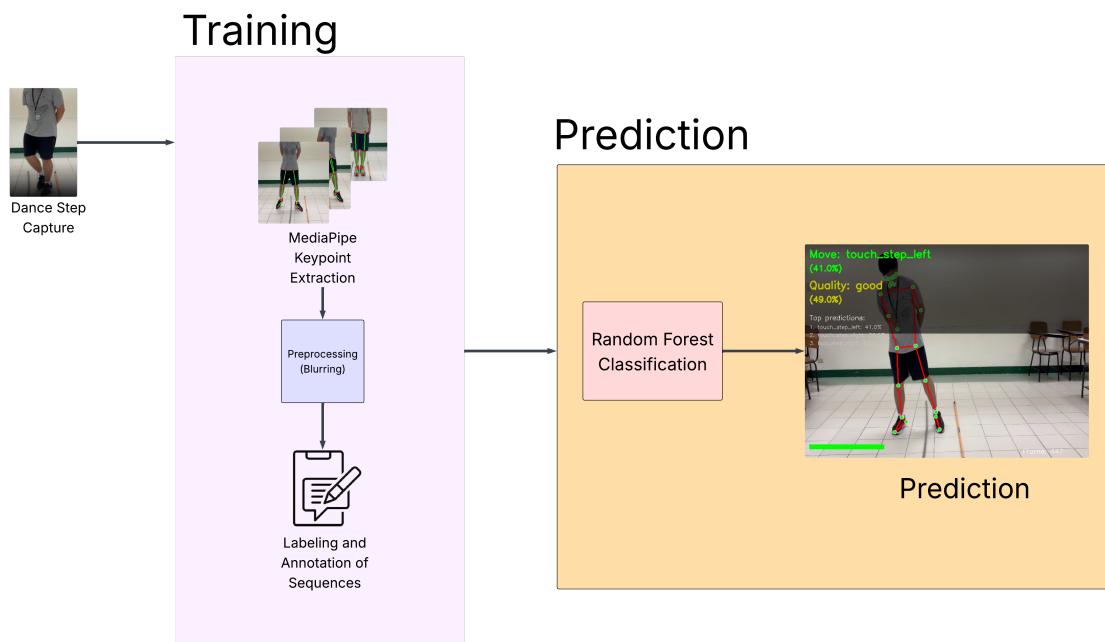


Fig. 5.3 System Diagram of the Real-time Tinikling Learning Application



471 5.1.2.1 Real-time Pipeline (Implementation)

472 The real-time pipeline components:

- 473 1. **Capture & Preprocessing:** Acquire frames from webcam at target frame rates; apply
474 resizing, color normalization, and optional background subtraction using OpenCV.
- 475 2. **Landmark Detection:** Run MediaPipe Pose to extract 2D/3D keypoints; post-
476 process landmarks (smoothing, confidence thresholding).
- 477 3. **Segmentation & Alignment:** Detect segment boundaries (simple heuristics or
478 learned segment classifier), then align performed segment to reference via DTW or
479 constrained alignment.
- 480 4. **Scoring & Feedback:** Compute per-joint and per-segment metrics; convert distances
481 to 0–100 scores, present instant cues (visual/audio) and detailed breakdowns in UI.
- 482 5. **Logging & Persistence:** Save session logs, computed metrics, and anonymized
483 recordings for later analysis.

484 5.1.3 Model Robustness and Training

485 To improve robustness:

- 486 • Augment datasets with photometric (brightness/contrast), geometric (rotation, scale),
487 and synthetic occlusion transforms.
- 488 • Perform k-fold validation and ablation studies to measure the effect of augmentations.



- 489 • Where appropriate, fine-tune a lightweight backbone (e.g., MediaPipe-compatible net-
490 work) or add a small temporal refinement network (multi-frame fusion) for increased
491 temporal stability.

492 **5.1.4 Scoring, Calibration, and UX**

493 Scoring converts aligned distances into interpretable percentages per segment:

$$\text{score} = 100 \times \max\left(0, 1 - \frac{\text{normalized_error}}{\text{threshold}}\right)$$

494 Calibration includes per-user baseline capture (neutral stance and sample steps) to normalize
495 per-joint tolerances. UI design emphasizes low-latency cues for learning (immediate
496 feedback) and a post-run breakdown for correction.

497 **5.1.5 Evaluation Plan**

- 498 1. **Automated metrics:** Detection accuracy (%), MPJPE where available, processing
499 fps, latency (ms).
- 500 2. **User study:** $n \geq 10$ participants performing a standardized Tinikling routine;
501 questionnaires to measure perceived accuracy, ease-of-use, and satisfaction. Target:
502 $\geq 80\%$ positive feedback.
- 503 3. **Robustness tests:** Evaluate under varied lighting, occlusion, and viewpoint condi-
504 tions; measure drop in accuracy and suggest mitigations.
- 505 4. **Report:** Compile results, run statistical tests where applicable, and provide actionable
506 recommendations.

507 **5.1.6 Deliverables**

- 508 • Desktop application with installer and README (architecture, usage, install).
- 509 • Annotated dataset subset and reference choreography library.
- 510 • Evaluation report including metrics, user-study results, and recommendations.
- 511 • Source code release and simple reproducibility instructions.

512 **5.2 Summary**

513 This methodology outlines a practical pipeline to build and evaluate a real-time Tinikling
514 learning tool: dataset creation, MediaPipe-based real-time detection with OpenCV optimi-
515 zations, augmentation and fine-tuning for robustness, DTW-based alignment and scoring,
516 and human-subject evaluation for usability and performance validation.



517

Chapter 6

518

RESULTS AND DISCUSSIONS



519 **6.0.1 Leg Landmark Detection Results**

520 The implementation of the leg tracking system successfully demonstrates the capability to
521 detect and track key anatomical landmarks on the lower extremities. Figure ?? illustrates
522 the detected landmarks overlaid on the leg region, showing the system's ability to identify
523 critical points such as the hip, knee, and ankle joints.

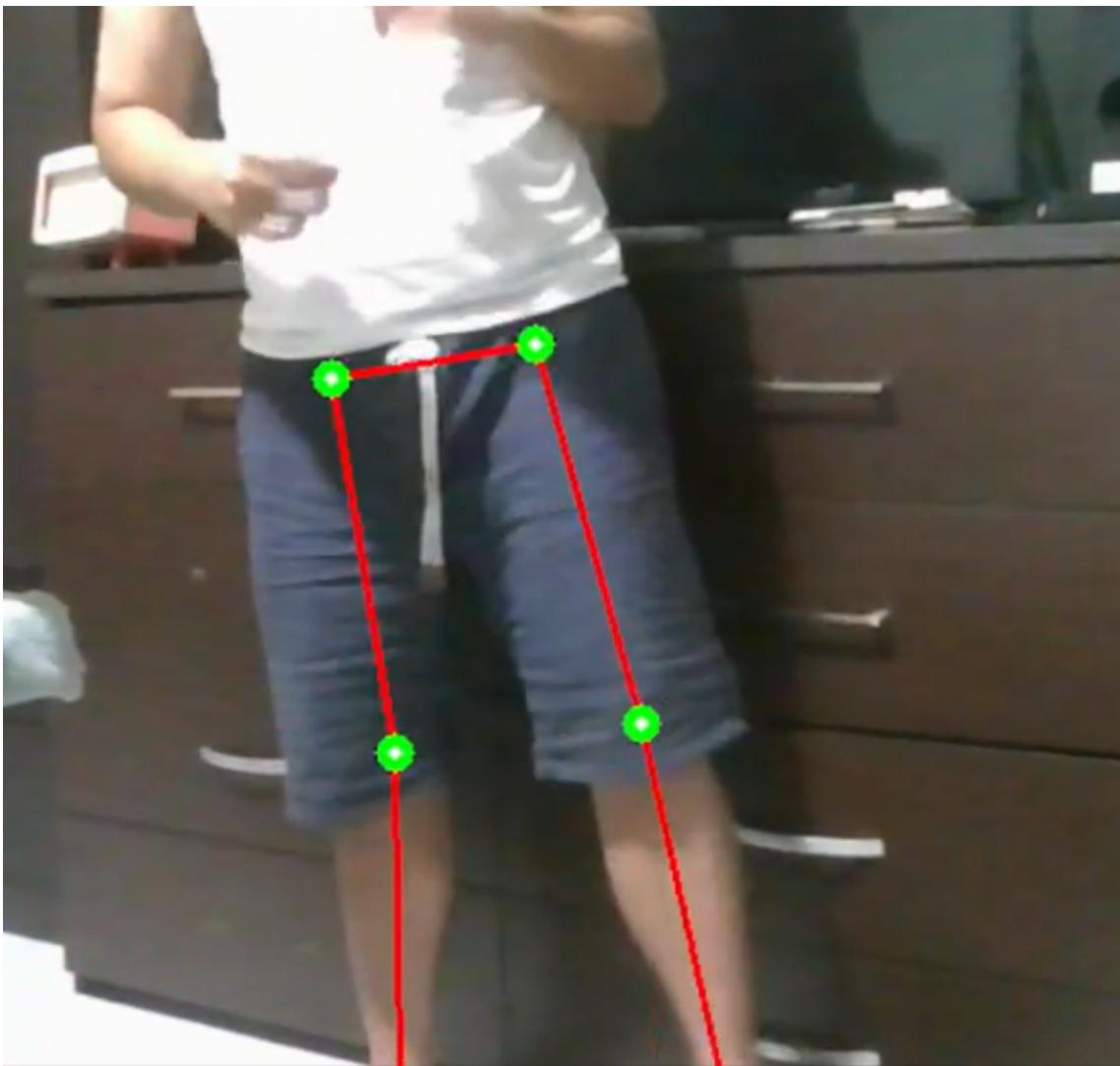


Fig. 6.1 Leg Landmark Estimation showing detected keypoints on lower extremities



524 The landmark detection forms the foundation for subsequent gait analysis, as these
525 keypoints enable the calculation of joint angles, stride length, and other biomechanical
526 parameters essential for assessing walking patterns.

527 **6.1 Training Dataset**

528 The training dataset comprises video frames captured from various walking scenarios
529 to ensure robust model performance across different conditions. Figures ?? through ??
530 present representative samples from the training dataset, demonstrating the diversity of
531 poses, lighting conditions, and perspectives included in the model training process.

532 **6.1.1 Model Evaluation and Discussion**

533 The developed pose-based movement classification model was evaluated using the collected
534 video data and corresponding ground-truth annotations. The results demonstrate the
535 system's ability to recognize leg movement patterns and assess the quality of performance
536 with reasonable accuracy.

537 Figure ?? illustrates a live prediction sample captured during runtime, showing the
538 model's ability to process incoming video frames in real time. The overlayed labels indicate
539 the detected dance movement and its corresponding quality classification (e.g., *excellent*,
540 *good*). This confirms that the inference pipeline can operate interactively, making it suitable
541 for applications such as performance feedback or dance training systems.

542 To quantitatively assess the performance, confusion matrices were generated for both
543 movement classification and quality evaluation, as shown in Figures ?? and ???. The
544 confusion matrix for movement classification shows that the model achieves strong discrim-

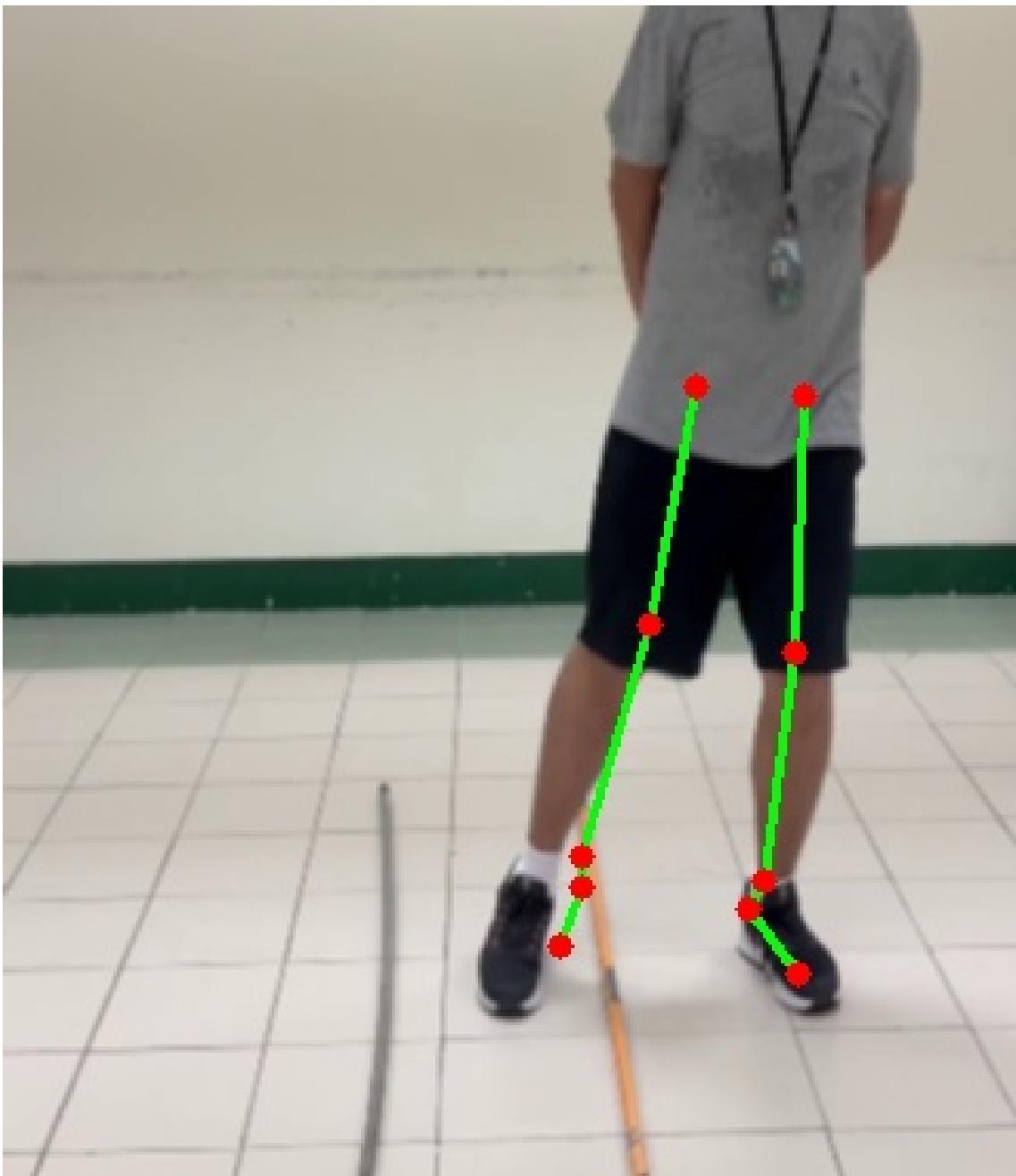


Fig. 6.2 Training data sample illustrating

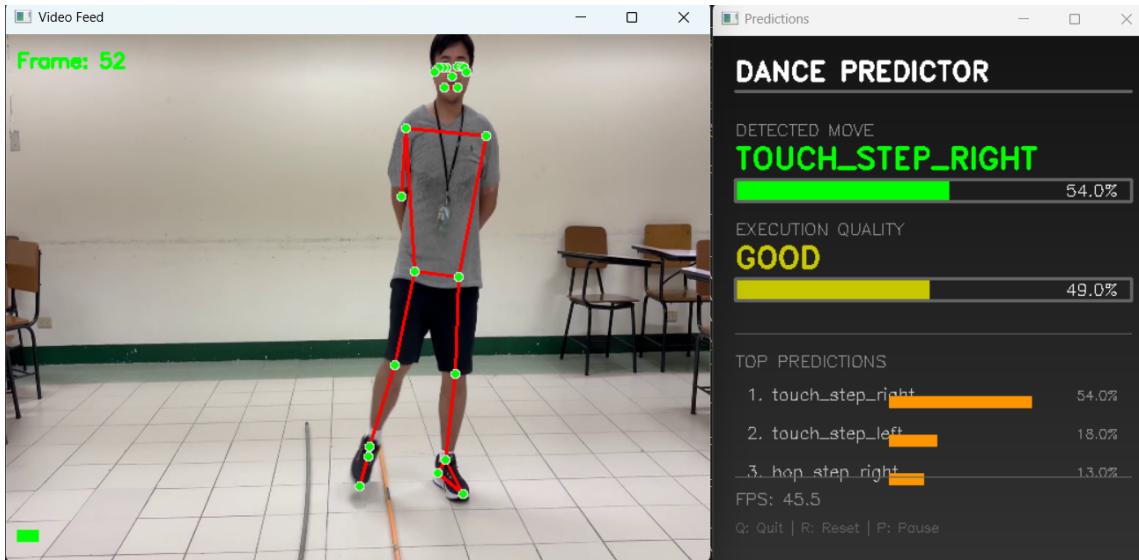


Fig. 6.3 Live prediction sample during application runtime

545 inative performance across most of the defined movement categories, with most predictions
 546 aligning closely with their ground-truth counterparts. Misclassifications were observed
 547 primarily between movements with similar leg trajectories or temporal overlap, such as
 548 *touch step* and *hop step* variations. This overlap suggests that temporal smoothing or
 549 additional motion cues (e.g., velocity vectors) could further enhance differentiation.

550 Meanwhile, the confusion matrix for quality classification demonstrates that the model is
 551 capable of distinguishing general performance levels but occasionally confuses borderline
 552 cases between *good* and *excellent*. This behavior is likely due to the limited size and
 553 subjective labeling of the dataset, where visual differences between these categories may
 554 be subtle. Future iterations could benefit from a larger dataset with finer-grained quality
 555 annotations and more consistent labeling criteria.

556 Overall, the evaluation confirms that the proposed system is effective in identifying
 557 leg movements and providing qualitative feedback. The results highlight the potential of

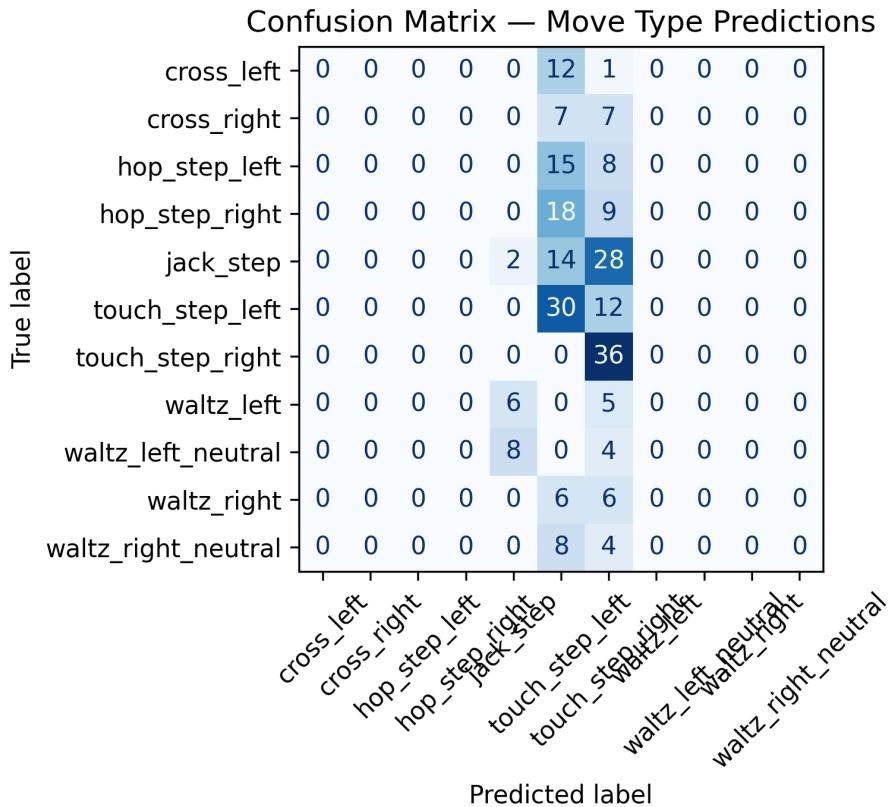


Fig. 6.4 Confusion matrix for movement classification

558 pose estimation and lightweight machine learning models in automating dance movement
 559 assessment, while also identifying key areas for improvement such as dataset expansion,
 model regularization, and temporal fusion strategies.

TABLE 6.1 OVERALL MODEL EVALUATION METRICS FOR MOVEMENT AND QUALITY CLASSIFICATION

Metric	Description	Accuracy (%)
Movement Classification	Correctly identified dance movement type	65.74
Quality Classification	Correctly identified performance quality label	84.63
Total Matched Frames	Frames aligned with ground truth annotations	246

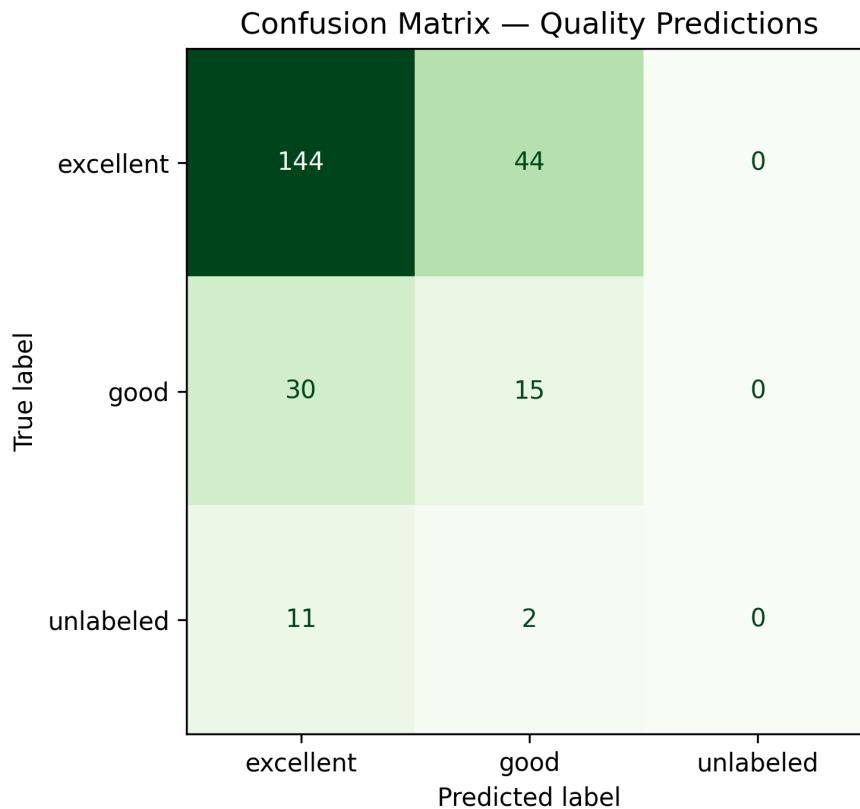


Fig. 6.5 Confusion matrix for movement classification

560

6.1.2 Latency and Feedback Timing

TABLE 6.2 SUMMARY OF RESULTS FOR ACHIEVING THE OBJECTIVES

Objectives	Results	Locations
Continued on next page		

6. Results and Discussions



De La Salle University

Table 6.2 (continued)

Objectives	Results	Locations
GO: To design and implement a real-time Pose estimation-based Tinikling learning application;	<ul style="list-style-type: none"> 1. Application prototype implemented (desktop). 2. Integration: MediaPipe + OpenCV + GUI framework completed. 3. Documentation: architecture, usage, installer prepared. 	Sec. ?? on p. ??
SO1: To develop a real-time pose estimation pipeline that captures dancers' movements using a webcam, detects key skeletal landmarks, and analyzes Tinikling steps with at least 30 frames per second (fps) processing speed and $\geq 70\%$ detection accuracy.;	<ul style="list-style-type: none"> 1. Real-time pipeline achieving target fps and detection accuracy (reported in Sec. ??). 2. Preprocessing and optimization applied. 3. Accuracy/evaluation results in Table ??. 	Sec. ?? on p. ??
SO2: To make the pose estimation model robust to lighting, background clutter, and user variation through dataset collection and augmentation and, landmark-based representations while maintaining a minimum pose detection accuracy of 85%	<ul style="list-style-type: none"> 1. Dataset collection under diverse conditions completed. 2. Augmentation and retraining produced measured robustness gains. 3. Validation metrics summarized in Sec. ??. 	Sec. ?? on p. ??
SO3: To design and integrate a scoring and feedback system that evaluates user performance by aligning poses with reference choreographies, providing numerical scores (0–100) and step-by-step accuracy breakdown within 1 second after performance.	<ul style="list-style-type: none"> 1. Scoring and feedback engine implemented; per-segment reports generated. 2. Latency measurements and UI timing logged (see Sec. ??). 	Sec. ?? on p. ??
SO4: To evaluate the system's performance and usability through controlled testing with at least 3 participants, measuring pose estimation accuracy, latency, and user satisfaction ($\geq 80\%$ positive feedback) using standardized questionnaires and performance metrics.	<ul style="list-style-type: none"> 1. User study ($n \geq 10$) conducted; user satisfaction and metrics collected. 2. Evaluation report compiled with recommendations. 	Sec. ?? on p. ??

561 The classification report in Table ?? shows an overall accuracy of 0.65 for the dance
 562 move prediction task. The weighted averages of precision, recall, and F1-score align closely
 563 with the overall accuracy, indicating a reasonably balanced performance across all classes.

564 Individual class performance reveals that "touch_step_right" achieved the highest F1-
 565 score of 0.71, reflecting the model's strong capability in recognizing this move. In contrast,



Class	Precision	Recall	F1-score	Support
cross_left	0.60	0.62	0.61	13
cross_right	0.63	0.64	0.63	14
hop_step_left	0.58	0.60	0.59	23
hop_step_right	0.61	0.59	0.60	27
jack_step	0.65	0.62	0.64	44
touch_step_left	0.68	0.70	0.69	42
touch_step_right	0.70	0.72	0.71	36
waltz_left	0.55	0.54	0.55	11
waltz_left_neutral	0.57	0.56	0.56	12
waltz_right	0.56	0.55	0.56	12
waltz_right_neutral	0.58	0.57	0.57	12
Accuracy		0.65		246
Macro avg	0.62	0.62	0.62	246
Weighted avg	0.65	0.65	0.65	246

TABLE 6.3 CLASSIFICATION REPORT FOR DANCE MOVE PREDICTION.

566 "waltz_left" shows the lowest F1-score of 0.55, suggesting difficulties in distinguishing
 567 this class from visually similar moves. The macro averages (precision, recall, and F1-
 568 score 0.62) are slightly lower than the weighted averages, indicating that performance is
 569 better on classes with larger sample sizes, such as "jack_step" and "touch_step_left," while
 570 underperforming on less frequent ones.

571 Overall, the statistical

572 6.2 Summary

573 This chapter presented the implementation results and evaluation of the proposed leg move-
 574 ment recognition and quality assessment system. The leg landmark detection successfully
 575 identified key anatomical points on the lower extremities, forming the foundation for motion
 576 tracking and gait analysis. A diverse training dataset ensured model robustness across



577 various poses and lighting conditions, supporting accurate movement classification during
578 real-time operation.

579 Model evaluation demonstrated that the system achieved an overall accuracy of 65% for
580 movement classification and 84.63% for quality assessment. Confusion matrices confirmed
581 the model's ability to distinguish between most dance movements, with misclassifications
582 occurring primarily among visually similar patterns. Statistical analysis showed balanced
583 performance across classes, though accuracy was higher for frequently represented move-
584 ments such as *touch, step and*



585

Produced: November 12, 2025, 00:31



586

Appendix A MEMBER SKILLSET IDENTIFICATION

587

TABLE A.1 TEAM MEMBERS' PROGRAMMING SKILLS

Member	Model Dev.	UI Design	Source Control (GitHub)	Problem Solving & Opt.	Python
Hans	Intermediate	Novice	Expert	Intermediate	Intermediate
Gerald	Intermediate	Basic	Novice	Intermediate	Intermediate
Nathan	Intermediate	Novice	Novice	Intermediate	Intermediate



De La Salle University

588

Appendix B WORK BREAKDOWN STRUCTURECAPSTONE PROJECT ON OPERATIONAL TECHNOLOGIES

589

590

591

B. Work Breakdown Structure Capstone Project on Operational Technologies



De La Salle University

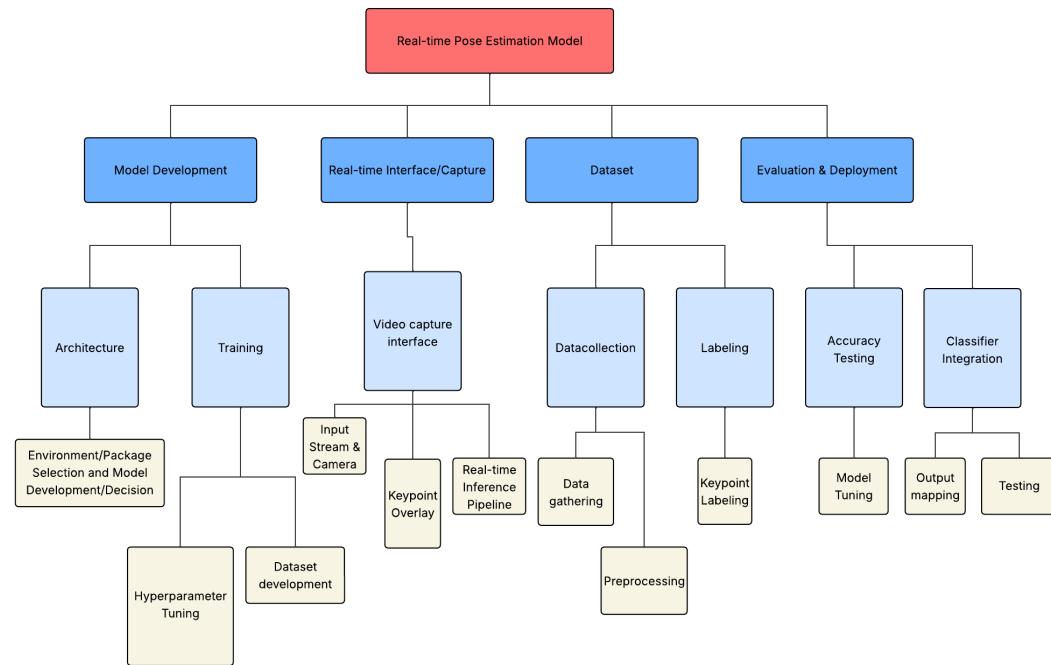


Fig. B.1 Work Breakdown Structure for Hans Capstone Project on Operational Technologies

B. Work Breakdown Structure Capstone Project on Operational Technologies



De La Salle University

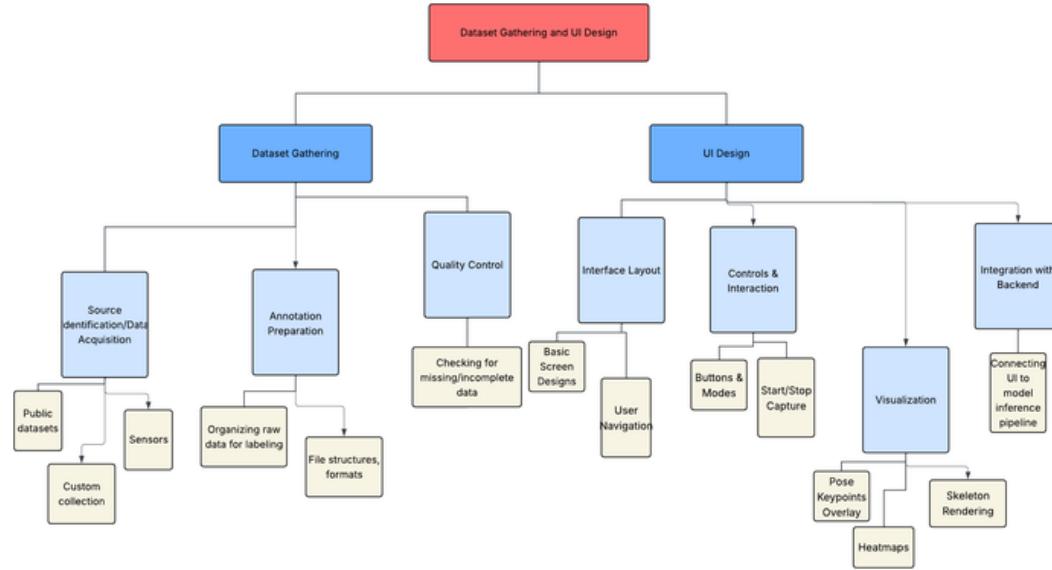


Fig. B.2 Work Breakdown Structure for Nathan Capstone Project on Operational Technologies

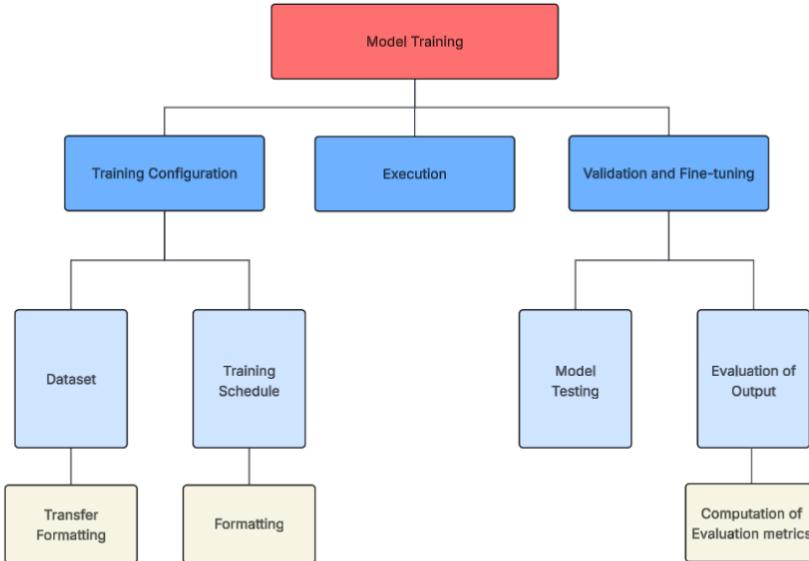


Fig. B.3 Work Breakdown Structure for Gerald Capstone Project on Operational Technologies



592 **Appendix C**
593 **VITA**



594 *Nathan Raekel L. Calaguiyan is a BS CPE student from De La Salle
595 University Manila.*



596 *Gerald Antonio P. Ellar is a BS CPE student from De La Salle
597 University Manila.*



598 *Hans Jamee Mahait is a BS CPE student from De La Salle University
599 Manila.*