STUDENT LAB SAFETY, HIRARC, DATASHEET VERIFICATION CHECKLIST, AND BRIEF REPORT SUBMISSION

EXPERIMENT KW14: Line Following with ArmPi Pro

STUDENT NAME : _____
MATRIC NO. : _____
GROUP : _____
DATE : _____

A. Equipment and System Verification Checklist

Item	Verification Requirement	✓
ArmPi Pro Robotic Kit	The robot, Raspberry Pi expansion board, mecanum-wheel chassis, camera, motors, robotic arm, battery, and associated components are present and appear serviceable.	□
Battery Condition	The ArmPi Pro battery is sufficiently charged and securely connected before powering the system.	□
Robot Operating Area	A clear, level, dry, and unobstructed operating area is prepared for line-following trials.	□
Coloured Tape Paths	Three coloured tape paths are securely attached to the floor or test surface without raised edges, folds, or loose sections.	□
Line Length	Each test line is measured using a measuring tape or ruler, and its total length, L, is recorded in metres.	□
Computer / Laptop	The computer or laptop is operational and has a functioning Wi-Fi connection, supported web browser, and NoMachine software.	□
ArmPi Pro Power-Up	The Raspberry Pi expansion board starts correctly, with LED indications observed according to the operating procedure.	□
Wi-Fi Connection	The computer is connected to the ArmPi Pro Wi-Fi hotspot beginning with “HW”.	□
NoMachine Connection	The Raspberry Pi remote desktop can be accessed successfully through NoMachine.	□
Camera Position	The camera is securely mounted, unobstructed, and positioned to capture the coloured line within the required field of view.	□
Camera Image Transmission	The live camera or processed image can be viewed through the ROS image-topic interface.	□
Source-Code File	The required visual_patrol_node.py file has been copied to the correct source-code location before testing.	□
ROS Services	The visual-patrol enter, start, target-selection, stop, and exit services are accessible through the terminal.	□
Colour Recognition	The target colour can be selected, and the coloured line is distinguishable in the processed camera image.	□
LAB Threshold Configuration	LAB threshold values are checked for each target colour before collecting performance data.	□
Motor-Velocity Parameters	Linear velocity, direction angle, and yaw-rate parameters are identified before editing the program.	□
Measurement Instruments	A measuring tape or ruler and a stopwatch or timer are available and functioning correctly.	□

Data Recording	A results sheet is prepared to record line length, programmed velocity, trial time, calculated speed, deviation, tracking success, and observations.	□
Emergency Stop Readiness	Students know how to stop the visual-patrol operation and switch off the robot safely if unexpected movement occurs.	□

B. HIRARC Checklist (OSH-Based Risk Assessment)

Hazard / Unsafe Condition	Potential Consequence	Control Measure	✓
Robot powered on during physical adjustment	Unexpected wheel or robotic-arm movement may cause minor contact injury or equipment damage	Stop the running program and switch off the robot before adjusting the camera, wheels, battery, robotic arm, or wiring.	□
Hands placed near moving mecanum wheels	Fingers, clothing, or loose items may be caught by the wheel assembly	Keep hands, cables, clothing, and personal items clear while the robot is operating.	□
Hands placed within the robotic-arm workspace	Contact with a moving servo or robotic-arm link	Keep the robotic-arm workspace clear and do not manually restrain a powered servo.	□
Robot operated near a table	Robot may fall and cause injury or equipment damage	Conduct all trials on a level surface with adequate clearance from edges, stairs, doors, and pedestrian routes.	□
Obstructed operating area	Collision with equipment, furniture, bags, cables, or people	Establish and maintain a clearly defined robot testing zone before starting each trial.	□
Loose	Mecanum wheels may catch the tape, resulting in tracking failure or sudden change of direction	Secure the full length of each tape path and inspect the path before every trial.	□
Slippery, wet, or uneven test surface	Robot may skid, deviate from the path, or overturn	Use a clean, dry, flat, and consistent test surface.	□
Damaged or unsecured battery	Equipment malfunction, power interruption, overheating, or damage	Inspect and secure the battery before use. Stop using the system if the battery or connector appears damaged.	□
Incorrect power-up or shutdown sequence	Corrupted files, incomplete software initialization, or hardware malfunction	Follow the prescribed startup sequence and shut down the Raspberry Pi through the operating-system menu before switching off the expansion board.	□
Incorrect Wi-Fi network connection	Failure to establish communication with the robot or unintended connection to another device	Confirm the ArmPi Pro hotspot name before connecting and verify remote-desktop access before operating the robot.	□

Incorrect login or remote-desktop session	Robot software cannot be accessed or controlled correctly	Confirm that the intended ArmPi Pro remote desktop is selected before opening the program.	□
Incorrect source-code replacement	Program failure, wrong control behaviour, or inability to run the experiment	Back up the original file and confirm the filename and source-code location before replacement.	□
Editing code while the patrol program is active	Updated parameters may not be executed correctly, or the robot may move unexpectedly	Stop and exit the visual-patrol program before modifying the source code. Restart the robot safely after saving the updated code.	□
Incorrect ROS service command	Program may fail to start, fail to stop, or use the wrong tracking configuration	Check the command carefully before execution and verify the terminal response after each ROS service call.	□
Incorrect target-colour selection	Robot follows the wrong object or fails to detect the intended line	Confirm the selected target colour and inspect the processed camera image before allowing the robot to move.	□
Incorrect LAB thresholds	Poor segmentation, false detection, intermittent tracking, or complete loss of the line	Calibrate the L, a, and b limits so that the target line appears white and unrelated regions remain black in the binary image.	□
Lighting variation or reflection	Inconsistent colour detection and unreliable performance comparison	Maintain consistent lighting and camera orientation throughout all comparative trials. Recalibrate thresholds when required.	□
Camera obstruction or displacement	Loss of line detection and uncontrolled deviation	Ensure the camera lens is clean, unobstructed, and securely positioned before every trial.	□
Excessive linear velocity	Overshoot, delayed steering correction, line loss, collision, or abrupt motion	Increase velocity only according to the experiment procedure and monitor robot behaviour from a safe position.	□
Incorrect direction-angle parameter	Robot travels in an unintended direction	Verify the direction-angle value before starting the line-following operation.	□
Excessive yaw rate	Sudden rotation, oscillation, poor tracking, or loss of path	Adjust the yaw rate cautiously and confirm the robot response within the controlled test zone.	□
Observer stands directly in front of the robot	Contact may occur if the robot leaves the test line	Stand beside or behind the operating path and maintain a clear route for stopping the robot.	□
Attempting to catch a moving or falling robot	Hand or body injury	Use the software stop command when possible. Do not place hands near moving wheels or joints.	□

Inconsistent stopwatch operation	Inaccurate trial time and calculated speed	Use a consistent start and stop reference point and, where practical, assign the same student to time all comparative trials.	□
Incorrect line-length measurement	Systematic error in calculated robot speed	Measure the effective travel distance from the defined starting point to the finishing point before testing.	□
Insufficient repeated trials	Results may not represent repeatable robot performance	Complete at least three trials for every colour and programmed-velocity setting as required by KW14.	□
Unsaved threshold or source-code changes	Modified settings are lost or inconsistent between trials	Save the LAB configuration and program file before restarting the system or collecting data.	□
Extended computer use	Eye strain or physical discomfort	Maintain an ergonomically appropriate posture and take short visual breaks during prolonged configuration or analysis.	□
Uncontrolled shutdown after code modification	Software conflict or failure to execute the latest source code	Stop and exit the visual-patrol service, shut down the Raspberry Pi correctly, and only then switch off the expansion board.	□

C. Engineering Practice Verification

Requirement	✓
The ArmPi Pro hardware, camera, battery, mecanum wheels, Raspberry Pi, and operating area are inspected before testing.	□
The ArmPi Pro is powered on using the correct startup sequence.	□
The computer is connected to the correct ArmPi Pro Wi-Fi hotspot.	□
The Raspberry Pi remote desktop is accessed successfully through NoMachine.	□
The required visual_patrol_node.py file is placed in the correct location.	□
The camera image and processed visual output are inspected before robot movement.	□
The target line colour is selected using the appropriate ROS service.	□
The initial red-coloured test line is at least 1 metre long, as required by the procedure.	□
The effective line length, L, is measured and recorded in metres.	□
The time, t, required to complete the line-following task is measured consistently.	□
Robot speed is calculated using $v = L/t$.	□
At least three trials are completed for every tested combination of colour and velocity setting.	□
Two additional tape colours are configured and evaluated in addition to red.	□
LAB threshold values are determined and recorded for each target colour.	□
The target-colour region appears white while non-target regions appear black in the processed binary image.	□

The programmed linear velocity is changed to the required values, including 50 and 150.	□
The robot is safely restarted after source-code modifications.	□
The same path length and comparable operating conditions are maintained across comparative trials.	□
The average time and average measured speed are calculated for each colour and velocity setting.	□
The difference between programmed velocity and measured speed is calculated and analysed.	□
Line-following success, deviation, oscillation, overshoot, correction behaviour, and line loss are recorded using objective observations.	□
The effect of velocity on line-following performance is evaluated for each colour.	□
The effect of colour recognition and threshold selection on tracking performance is discussed.	□
The visual-patrol operation is stopped and exited correctly after completing the experiment.	□
The Raspberry Pi is shut down through the operating system before the expansion board is switched off.	□
All settings, measurements, screenshots, observations, calculations, and engineering conclusions are documented.	□

D. Student Declaration

I confirm that:

☐ I have completed the equipment inspection, safety verification, HIRARC assessment, and engineering-practice checks for Experiment KW14.

☐ I have verified that the ArmPi Pro operating area is clear and that the camera, battery, mecanum wheels, software connection, source-code file, ROS services, line-colour configuration, and measurement instruments are ready for use.

☐ I will keep hands and personal items clear of all moving components, operate the robot only within the designated test area, and report any hardware, software, battery, camera, motor, or safety issue immediately to the laboratory instructor, demonstrator, or technical staff.

Student Signature : _____

Date : _____

D. Experiment Data Results and Discussion

In Lab Task (During Lab)

Task 1: ArmPi Pro Setup and System Verification

Table E1.1: Hardware and Communication Verification

Verification Item	Required Condition	Status	Observation
Raspberry Pi expansion board	LED indications observed during startup	Pass / Fail	
ArmPi Pro Wi-Fi hotspot	Hotspot beginning with “HW” is detected	Pass / Fail	
Wi-Fi connection	Computer connects successfully	Pass / Fail	
NoMachine connection	Raspberry Pi remote desktop is accessible	Pass / Fail	
Camera image	Live image is visible	Pass / Fail	
Processed image	Visual-processing result is visible	Pass / Fail	
Source-code file	Required file is placed in the intended location	Pass / Fail	
ROS patrol service	Enter command executes successfully	Pass / Fail	
ROS running service	Line-following operation can be started	Pass / Fail	
ROS target service	Selected target colour is accepted	Pass / Fail	
ROS stop service	Robot operation can be stopped safely	Pass / Fail	
ROS exit service	Visual-patrol program exits correctly	Pass / Fail	

Table E1.2: Initial System Information

Parameter	Recorded Information
Group number	
Test date	
Test surface	
Lighting condition	
Camera position or height	
Effective line length, L	
Tape colour 1	Red
Tape colour 2	
Tape colour 3	
Initial linear velocity setting	
Initial direction-angle setting	
Initial yaw-rate setting	
Source-code filename	visual_patrol_node.py
Remarks	

Task 2: Line-Colour Calibration Data Sheet

Table E2.1: LAB Threshold Calibration

Colour	L Minimum	L Maximum	a Minimum	a Maximum	b Minimum	b Maximum	Target Segmentation Quality
Red							Good / Acceptable / Poor
Colour 2: _____							Good / Acceptable / Poor
Colour 3: _____							Good / Acceptable / Poor

Table E2.2: Colour-Recognition Verification

Colour	Target Appears White	Background Appears Black	Noise Present	Line Centre Detected	Calibration Accepted
Red	□	□	Yes / No	Yes / No	Yes / No
Colour 2: _____	□	□	Yes / No	Yes / No	Yes / No
Colour 3: _____	□	□	Yes / No	Yes / No	Yes / No

Task 3: Line-Following Speed Measurement

Calculate the measured speed for every trial using:

$$v = \frac{L}{t}$$

where:

- v= measured robot speed
- L= effective line length in metres
- t= measured travel time in seconds

Table E3.1: Velocity Setting 1

Programmed linear velocity : _____
Direction angle : _____
Yaw rate : _____

Tape Colour	Line Length, L(m)	Trial	Time, t(s)	Measured Speed, $v = L/t$(m/s)	Completed Line?	Tracking Observation
Red		1			Yes / No	
		2			Yes / No	
		3			Yes / No	
Colour 2: _____		1			Yes / No	
		2			Yes / No	
		3			Yes / No	

Colour 3: _____		1			Yes / No	
		2			Yes / No	
		3			Yes / No	

Table E3.2: Velocity Setting 2
Programmed linear velocity: 50
Direction angle : _____
Yaw rate : _____

Tape Colour	Line Length, L(m)	Trial	Time, t(s)	Measured Speed, $v = L/t$(m/s)	Completed Line?	Tracking Observation
Red		1			Yes / No	
		2			Yes / No	
		3			Yes / No	
Colour 2: _____		1			Yes / No	
		2			Yes / No	
		3			Yes / No	
Colour 3: _____		1			Yes / No	
		2			Yes / No	
		3			Yes / No	

Table E3.3: Velocity Setting 3
Programmed linear velocity: 150
Direction angle : _____
Yaw rate : _____

Tape Colour	Line Length, L(m)	Trial	Time, t(s)	Measured Speed, $v = L/t$(m/s)	Completed Line?	Tracking Observation
Red		1			Yes / No	
		2			Yes / No	
		3			Yes / No	
Colour 2: _____		1			Yes / No	
		2			Yes / No	
		3			Yes / No	
Colour 3: _____		1			Yes / No	
		2			Yes / No	
		3			Yes / No	

Task 4: Summary of Average Results

Table E4.1: Average Time and Speed

Programmed Velocity	Tape Colour	Average Time (s)	Average Measured Speed (m/s)	Successful Trials	Overall Tracking Quality
	Red				Good / Acceptable / Poor
	Colour 2: _____				Good / Acceptable / Poor
	Colour 3: _____				Good / Acceptable / Poor
50	Red				Good / Acceptable / Poor
50	Colour 2: _____				Good / Acceptable / Poor
50	Colour 3: _____				Good / Acceptable / Poor
150	Red				Good / Acceptable / Poor
150	Colour 2: _____				Good / Acceptable / Poor
150	Colour 3: _____				Good / Acceptable / Poor

Table E4.2: Programmed and Measured Velocity Comparison

Tape Colour	Programmed Velocity	Average Measured Speed (m/s)	Difference / Error	Percentage Error, if units are comparable	Remarks
Red					
Colour 2: _____					
Colour 3: _____					
Red	50				
Colour 2: _____	50				
Colour 3: _____	50				

Red	150				
Colour 2: _____	150				
Colour 3: _____	150				

Where the programmed and measured values use directly comparable units:

$$\text{Percentage Error} = \left|\frac{v_{\text{measured}} - v_{\text{programmed}}}{v_{\text{programmed}}}\right| \times 100\%$$

Table E4.3: Tracking-Performance Assessment

Programmed Velocity	Colour	Initial Detection	Straight-Line Tracking	Oscillation / Lateral Deviation	Line Loss	Completion Result
	Red					
	Colour 2					
	Colour 3					
50	Red					
50	Colour 2					
50	Colour 3					
150	Red					
150	Colour 2					
150	Colour 3					

Task 5: Image-Processing Verification

Table E5.1: Image-Processing Stages

Processing Stage	Purpose	Student Observation / Result	Screenshot Reference
RGB-to-LAB conversion	Separate luminance information from colour components for thresholding		
Binarisation	Separate the target line from the background		
Erosion	Reduce small noise regions in the binary image		
Dilation	Improve line-region continuity after erosion		
Contour detection	Locate candidate target regions		

Maximum valid contour selection	Select the primary detected line while rejecting small interference		
Minimum enclosing rectangle	Determine the spatial boundary of the selected contour		
Centre-point calculation	Determine the line position used for steering correction		
Motor command generation	Adjust robot movement according to the detected line position		

Task 6: In-Lab Analytical and Open-Ended Questions

Q1. LAB Colour Recognition

Question / Task	Key Concept to Discuss	Analysis / Explanation Requirement
Explain the colour-recognition process used by ArmPi Pro.	**LAB colour space and thresholding**	Explain why the camera image is converted from RGB to LAB and describe how the minimum and maximum L, a, and b values isolate the target colour.

Answer:

Q2. Morphological Processing

Question / Task	Key Concept to Discuss	Analysis / Explanation Requirement
Explain the purpose of erosion and dilation.	**Noise removal and target refinement**	Describe how erosion and dilation modify the binary image and explain how these operations improve contour detection.

Answer:

Q3. Contour and Centre-Point Detection

Question / Task	Key Concept to Discuss	Analysis / Explanation Requirement
Explain how the robot finds the line position.	**Contour extraction and geometric position**	Explain how the valid contour, enclosing rectangle, vertices, and centre point are obtained. Relate the centre-point position to movement correction.

Answer:

Q4. Velocity and Tracking Performance

Question / Task	Key Concept to Discuss	Analysis / Explanation Requirement
Compare the robot's performance at different velocity settings.	**Dynamic tracking response**	Use the measured results to discuss travel time, average speed, lateral deviation, oscillation, correction behaviour, line loss, and successful completion.

Answer:

Q5. Programmed Velocity and Measured Speed

Question / Task	Key Concept to Discuss	Analysis / Explanation Requirement
Compare the programmed velocity with the measured speed.	**Commanded and actual motion**	Explain whether the robot's measured speed changed consistently with the programmed setting. Discuss relevant causes of differences, including acceleration, wheel slip, steering correction, timing uncertainty, battery condition, surface condition, and software-command scaling.

Answer:

Q6. Effect of Tape Colour

Question / Task	Key Concept to Discuss	Analysis / Explanation Requirement
Compare line-following performance for three tape colours.	**Colour segmentation quality**	Compare detection consistency, binary-image quality, line loss, tracking stability, and completion success. Support the comparison using the recorded thresholds and experimental observations.

Answer:

Q7. Effect of Lighting

Question / Task	Key Concept to Discuss	Analysis / Explanation Requirement
Evaluate the influence of lighting on colour recognition.	**Luminance variation and threshold sensitivity**	Explain how brightness, reflection, shadows, and camera exposure may influence LAB values and binary segmentation. State how the experiment controlled these variables.

Answer:

Q8. Effect of Increased Velocity on Vision-Based Control

Question / Task	Key Concept to Discuss	Analysis / Explanation Requirement
Determine whether increased velocity affects colour detection and line following.	**Image-processing and motion-control interaction**	Distinguish between failure to detect the line and failure to respond quickly enough to the detected position. Use camera evidence and robot-motion observations to justify the conclusion.

Answer:

Q9. Repeatability and Experimental Uncertainty

Question / Task	Key Concept to Discuss	Analysis / Explanation Requirement
Evaluate the repeatability of the results.	**Variation between repeated trials**	Compare the three trial times for every test condition. Identify possible sources of experimental variation and explain how the procedure could be improved.

Answer:

Q10. Engineering Improvement

Question / Task	Key Concept to Discuss	Analysis / Explanation Requirement
Propose improvements to the line-following system.	**Control, vision, and experimental design**	Recommend improvements to threshold calibration, camera placement, path design, motor control, velocity selection, timing, or data acquisition. Relate each recommendation to an observed limitation.

Answer:

Task 7: Overall Results and Engineering Discussion

Analysis Area	Required Discussion	Student Findings
ArmPi Pro setup	Discuss whether the robot, Raspberry Pi, camera, Wi-Fi, NoMachine, and ROS services operated correctly.	
Colour calibration	Compare the LAB thresholds and segmentation quality for the three colours.	
Initial line following	Describe the robot's performance at the initial velocity setting.	
Velocity setting 50	Discuss average speed, tracking stability, line loss, and completion performance.	
Velocity setting 150	Discuss average speed, tracking stability, line loss, and completion performance.	
Velocity comparison	Determine how increased programmed velocity affected actual travel time and measured speed.	
Colour comparison	Determine which colour produced the most consistent tracking and support the conclusion with evidence.	
Image-processing performance	Discuss the effectiveness of thresholding, morphology, contour extraction, and centre-point calculation.	
Commanded versus measured motion	Discuss differences between programmed velocity and measured speed.	
Repeatability	Discuss variation among repeated trials.	
Experimental limitations	Identify relevant limitations involving lighting, surface, tape quality, camera position, timing, battery, or software configuration.	
Engineering recommendation	Recommend a suitable velocity and colour-calibration approach based on the experimental results.	

DISCUSSION:

CONCLUSION:

LAB LECTURER **INSTRUCTOR**

SIGNATURE : _____ **SIGNATURE : _____**

DATE : _____ **DATE : _____**