

12 Flowcharts

2025-10-19

Contents

1	Flowcharts - Vision-Based Pick and Place System	1
1.1	Document Control	1
1.2	1. Introduction	2
1.3	2. Main System Flowchart	2
1.3.1	2.1 End-to-End Pick-Place Workflow	2
1.4	3. Vision Pipeline Flowchart	5
1.4.1	3.1 Object Detection & Pose Estimation	5
1.5	4. Grasp Planning Flowchart	7
1.5.1	4.1 Grasp Synthesis & Selection	7
1.6	5. Motion Planning Flowchart (MoveIt2)	9
1.6.1	5.1 Trajectory Planning & Execution	9
1.7	6. State Machine Flowchart	13
1.7.1	6.1 Task Orchestrator FSM	13
1.8	7. Error Handling Flowchart	16
1.8.1	7.1 Error Recovery Logic	16
1.9	8. Calibration Flowchart	18
1.9.1	8.1 Hand-Eye Calibration Procedure	18
1.10	9. Deployment Flowchart	21
1.10.1	9.1 System Deployment & Commissioning	21
1.11	10. Maintenance Flowchart	24
1.11.1	10.1 Preventive Maintenance Procedure	24
1.12	Summary	26

1 Flowcharts - Vision-Based Pick and Place System

1.1 Document Control

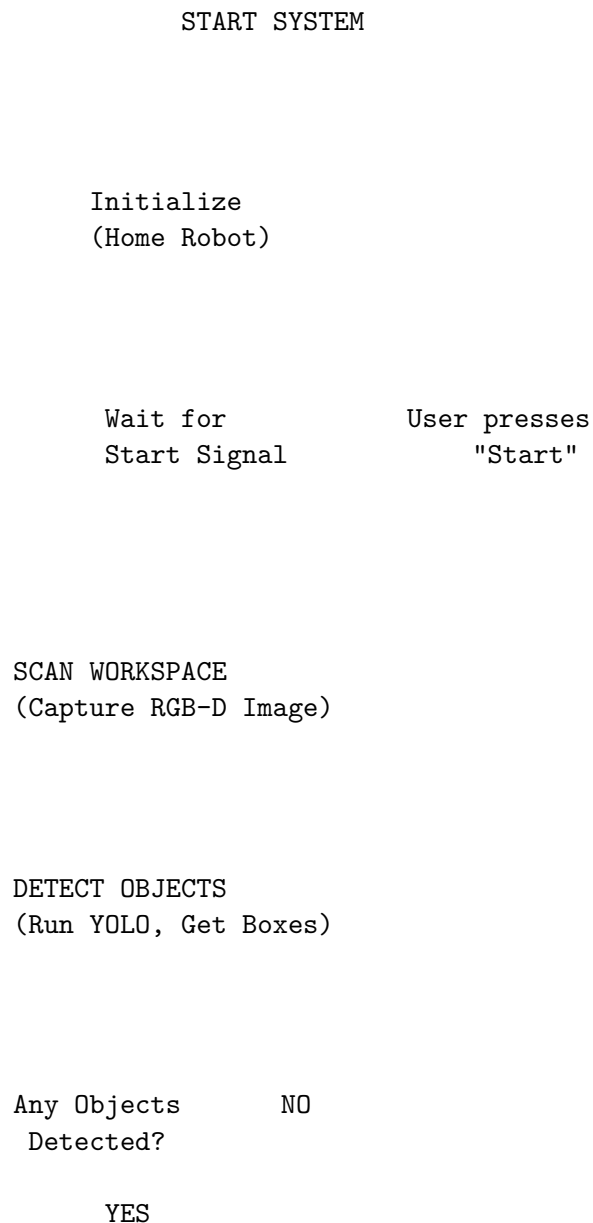
Item	Details
Document Title	System Flowcharts
Version	1.0
Date	2025-10-18
Status	Draft
Author(s)	System Architect, Technical Lead

1.2 1. Introduction

This document provides flowcharts for all major processes in the vision-based pick-and-place robotic system. Flowcharts are presented using ASCII art and can be rendered with tools like Mermaid or PlantUML.

1.3 2. Main System Flowchart

1.3.1 2.1 End-to-End Pick-Place Workflow



ESTIMATE POSE (6DoF for each obj)	Log: No Objects Return to SCAN
--------------------------------------	-----------------------------------

SELECT OBJECT
(Highest confidence)

PLAN GRASP
(Compute gripper pose
+ approach vector)

Grasp Valid?	NO
--------------	----

(Quality >0.5)

YES

PLAN PICK MOTION (MoveIt: Home→Grasp)	Skip Object Select Next
--	----------------------------

Path Found?	NO
-------------	----

(Collision-free)

YES

EXECUTE PICK - Move to pre-grasp - Move to grasp - Close gripper	Replan with Relaxed Constraints
---	---------------------------------------

Grasp Successful? NO

(F/T sensor check)

YES

RETRY GRASP

PLAN PLACE MOTION (Increase
(Grasp→Target) force)

EXECUTE PLACE

- Move to target
- Open gripper
- Retract

VERIFY PLACEMENT FAIL

(Optional camera
check)

PASS

Log Error

More Objects? Continue or
Alert User

YES NO

RETURN HOME

END (IDLE)

(Loop back to SCAN)

1.4 3. Vision Pipeline Flowchart

1.4.1 3.1 Object Detection & Pose Estimation

START VISION PIPELINE

Camera Trigger
(Request RGB-D frame)

Receive RGB-D Frame
- RGB: 1920x1080
- Depth: 1280x720

PREPROCESSING
- Resize to 640x640
- Normalize pixels
- Convert to tensor

OBJECT DETECTION
(YOLOv8 Inference)
Input: 640x640 RGB
Output: [x,y,w,h,c]

Detections NO
> 0 ?

YES

Non-Max Suppression Return
(Remove duplicates) Empty List

Filter Low Confidence
(threshold = 0.7)

For Each Detection:
POSE ESTIMATION

Extract ROI
(Crop RGB-D to bbox)

Generate Point Cloud
(Deproject depth)

Estimate 6DoF Pose
- Method: PnP / ICP
- Output: (x,y,z,
 qx,qy,qz,qw)

Transform to Robot
Frame (TF2)
Camera → Base

Publish Object Poses

(/vision/object_poses

END

1.5 4. Grasp Planning Flowchart

1.5.1 4.1 Grasp Synthesis & Selection

START GRASP PLANNING

Input:

- Object Pose
- Point Cloud
- Gripper Type

SAMPLE GRASPS

Method:

- Parallel Jaw:
 - Antipodal points
- Suction:
 - Top-down normals

Generate N=50

Candidate Grasps

For Each Grasp:

COLLISION CHECK

Gripper- Object Collision?	Gripper- Table Collision?
----------------------------------	---------------------------------

YES	YES
-----	-----

Mark Grasp INVALID

NO (both checks)

COMPUTE QUALITY

Metrics:

- Force closure
- Reachability
- Stability

Quality > 0.5?	NO
----------------	----

YES

Add to Valid List	Discard
-------------------	---------

All Grasps Checked?	NO
---------------------	----

YES

Valid Grasps > 0?	NO
-------------------	----

YES

RANK GRASPS
(Sort by quality)

Return
Failure

SELECT TOP GRASP

Compute Approach
Vector (pre-grasp)

Return:
- Grasp Pose
- Approach Vector
- Quality Score

END

1.6 5. Motion Planning Flowchart (MoveIt2)

1.6.1 5.1 Trajectory Planning & Execution

START MOTION PLANNING

Input:
- Target Pose
- Current State
- Planning Scene

UPDATE PLANNING
SCENE
- Add obstacles
 (from point cloud)

INVERSE KINEMATICS
Target Pose \rightarrow []

IK Solution Found?	IK Failed?
YES	YES

Try Alternate
IK Solver
(TRAC-IK)

(if found)

(if still fails)

Return Error
(Unreachable)

VALIDATE GOAL
- Joint limits OK?
- Self-collision?

Valid	Invalid
-------	---------

YES NO

Return Error

PATH PLANNING

Algorithm: RRT*

- Start: current
- Goal: IK solution

Planning Loop

(max 5 sec timeout)

Path Timeout?
Found?

YES YES

Relax
Constraints
(Retry)

(if found)

(if still fails)

Return Error
(No solution)

TRAJECTORY
GENERATION
- Time-param path
- Apply vel/acc
 limits

TRAJECTORY
SMOOTHING
- Shortcut
- Jerk limiting

SEND TO CONTROLLER
(FollowJoint
 Trajectory Action)

MONITOR EXECUTION
- Track feedback
- Check errors

Success Error?

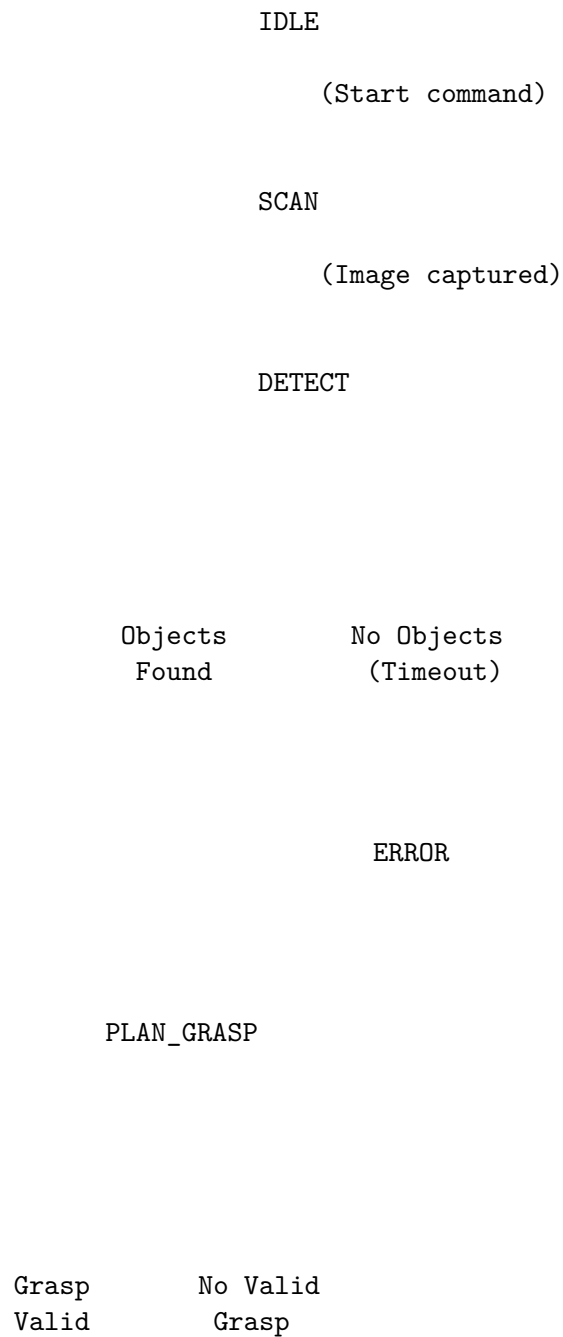
E-stop or
Following
Error Large

Return Error

END

1.7 6. State Machine Flowchart

1.7.1 6.1 Task Orchestrator FSM



PLAN_PICK

Path	Planning
Found	Failed

EXECUTE_PICK

Success	Grasp
	Failed

PLAN_PLACE

Path	Planning
Found	Failed

EXECUTE_PLACE

Success Failed

VERIFY

Pass Fail

ERROR

Retry or
Abort?

(Retry: back to SCAN)

(Abort)

IDLE

RETURN_HOME

IDLE

1.8 7. Error Handling Flowchart

1.8.1 7.1 Error Recovery Logic

ERROR DETECTED
(Vision fail, grasp fail, etc.)

Classify Error
- Vision timeout
- Grasp failure
- Planning failure
- Execution error
- Safety violation

Log Error
(timestamp, type,
context)

Safety
Violation?
(E-stop,
collision)

Recoverable
Error?

YES

Retry Count Non-
< Max (3)? Recoverable

YES NO
(Try Again) (Give Up)

RECOVERY ACTION

- Rescan
- Adjust params
- Retry with
fallback

Recovery
Successful?

YES NO

RESUME TASK

CRITICAL ERROR HANDLING

1. Stop all motion (E-stop if safety)
 2. Move to safe state (home position)
 3. Alert operator (dashboard, alarm)
 4. Await manual intervention
- Option A: Operator fixes issue, resume
 - Option B: Operator aborts task

User Action
Required

Resume	Abort
Continue Task	IDLE

1.9 8. Calibration Flowchart

1.9.1 8.1 Hand-Eye Calibration Procedure

START CALIBRATION WIZARD

Display Welcome
Instructions
- Place checkerboard
- Ensure good light

Initialize:
- Calibration data
- Position counter
(i = 1)

Move Robot to
Position i
(Pre-defined joint
angles)

Prompt User:
"Press OK when

robot stopped"

Capture Image
(RGB from camera)

Detect Checkerboard
Corners

Corners Detected?	Detection Failed?
YES	YES

Display Error
"Retry
Position i"

Record:
- Robot pose (FK)
- Image corners

i = i + 1

i <= N (e.g., 5)? NO

YES

COMPUTE CALIBRATION

- Solve $AX=XB$
- Hand-eye matrix

VALIDATION

- Place known
object
- Detect & measure
position error

Error <5mm?	Error ≥5mm?
YES	NO

Warn User
"Recalibrate"

SAVE CALIBRATION

- Write to YAML
- /config/camera_
robot_tf.yaml

Display Success
"Calibration
Complete!"

END

1.10 9. Deployment Flowchart

1.10.1 9.1 System Deployment & Commissioning

START DEPLOYMENT

SITE PREPARATION

- Clear workspace
- Install power
- Network setup

HARDWARE INSTALL

- Mount robot
- Install camera
- Connect cables

POWER-ON CHECKS

- Verify voltages
- E-stop test
- Network ping

All Checks

Any Checks

Pass?

Fail?

YES

YES

Troubleshoot
- Recheck
connections

(if fixed)

SOFTWARE INSTALL

- Docker pull
- Load ROS2 pkgs
- Config files

CALIBRATION

- Hand-eye calib
- Workspace zones
- Gripper tuning

SMOKE TEST

- Single pick-place
- Verify all
subsystems work

Success

Failure

Debug & Fix

(if fixed)

TRAINING

- Operator (2 days)
- Maintenance (1 d)

ACCEPTANCE TEST

- Run 100 picks
- Measure KPIs

All	Any KPI
Pass?	Fail?
YES	YES

- Remediate
- Tune params
 - Retest

(if fixed)

CUSTOMER SIGN-OFF

- UAT approval
- Handover docs

PRODUCTION READINESS

- Transition to ops
- Support handoff

END

1.11 10. Maintenance Flowchart

1.11.1 10.1 Preventive Maintenance Procedure

MAINTENANCE DUE
(Calendar-based or condition-based)

Schedule Downtime
(Notify operators)

RUN DIAGNOSTIC
HEALTH CHECK
- Camera test
- Motor test
- Sensor test

All Tests
Pass?

YES

Any Test
Fail?

YES

REPAIR
- Replace
component

LUBRICATION

- Joint bearings
- Gripper mechanics

CLEAN & INSPECT

- Camera lens
- Cables, connectors

UPDATE LOGS

- Maintenance date
- Parts replaced
- Next due date

POST-MAINT TEST

- Run smoke test
- Verify performance

Success Issues

Troubleshoot

RETURN TO SERVICE

(Notify operators)

END

1.12 Summary

This document provides **10 comprehensive flowcharts** covering:

1. **Main System** - End-to-end pick-place workflow
2. **Vision Pipeline** - Object detection and pose estimation
3. **Grasp Planning** - Grasp synthesis and selection
4. **Motion Planning** - MoveIt2 trajectory planning
5. **State Machine** - Task orchestrator FSM
6. **Error Handling** - Recovery logic
7. **Calibration** - Hand-eye calibration wizard
8. **Deployment** - System commissioning
9. **Maintenance** - Preventive maintenance procedure

Usage: - Convert to Mermaid diagrams for rendering - Use in design reviews, training materials - Reference during development and debugging

Document Status: Complete **Last Updated:** 2025-10-18 **Format:** ASCII art (convertible to Mermaid/PlantUML) **Review Status:** Pending Technical Review