# 07 Demo Scenarios

2025-10-19

## Contents

# 1 Demo Scenarios - Vision-Based Pick and Place System

## 1.1 Overview

This document outlines demonstration scenarios organized by priority using the **MoSCoW method**: - **Must Have:** Essential scenarios for MVP validation - **Should Have:** Important scenarios for production-readiness - **May Have:** Advanced scenarios showcasing full capabilities

Each scenario includes: setup, execution steps, success criteria, and robotics concepts demonstrated.

---

## 1.2 1. Must Have Demo Scenarios

### 1.2.1 Scenario M1: Basic Pick and Place (Single Object)

**Objective:** Demonstrate end-to-end workflow with a single known object

**Setup:** - Robot: UR5e with Robotiq 2F-85 gripper - Object: Red cube (50mm × 50mm × 50mm) on white table - Camera: RealSense D435i mounted eye-to-hand (above workspace) - Lighting: Uniform LED lighting (5000K, 2000 lumen) - Target: Marked drop zone (300mm from pick zone)

**Execution Steps:** 1. Start system: Press "Start" button on HMI 2. **Scan:** Camera captures RGB-D image, displays in RViz2 3. **Detect:** YOLO detects cube, bounding box overlays on image 4. **Localize:** Pose estimation outputs (x,y,z) = (0.4m, 0.2m, 0.05m) 5. **Plan Grasp:** Compute top-down grasp, gripper opens to 80mm 6. **Plan Pick:** MoveIt plans trajectory (home → pre-grasp → grasp) 7. **Execute Pick:** Robot moves, gripper closes, F/T sensor confirms grasp (20N) 8. **Plan Place:** Plan trajectory (pick → pre-place → place) 9. **Execute Place:** Robot moves to target, gripper opens, object released 10. **Return:** Robot returns to home position

**Success Criteria:** - Cycle time: <10 seconds (total time step 1-10) - Grasp success: Object lifted without slipping - Placement accuracy: <10mm from target center - No collisions detected

**Concepts Demonstrated:** - Computer vision (detection, pose estimation) - Inverse kinematics - Motion planning (collision-free trajectory) - Grasp planning (force closure) - State machine (task sequencing) - Coordinate transforms (camera → robot frame)

**Demo Video Deliverable:** 60-second video with screen capture (RViz) + real robot

---

### 1.2.2 Scenario M2: Multiple Objects (Sequential Picking)

**Objective:** Pick 5 objects sequentially from cluttered workspace

**Setup:** - Objects: 5 colored cubes (red, blue, green, yellow, black) randomly placed - Workspace: 600mm × 400mm area - Objects may partially occlude each other

**Execution Steps:** 1. Start system 2. For each object (repeat 5 times): - Scan workspace - Detect all visible objects - Select highest-confidence detection - Pick object - Place in designated zone (indexed by color) 3. Report total time and success rate

**Success Criteria:** - All 5 objects picked and placed - Total cycle time: <60 seconds - No "object not found" errors - Objects placed in correct color-coded zones

**Concepts Demonstrated:** - Multi-object detection - Scene understanding (occlusion handling) - Task planning (object prioritization) - Real-time replanning (workspace changes after each pick)

**Demo Video Deliverable:** 90-second time-lapse with analytics overlay (objects remaining, cycle time)

---

#### 1.2.3 Scenario M3: Error Recovery (Grasp Failure)

**Objective:** Demonstrate graceful error recovery when grasp fails

**Setup:** - Object: Slippery cylinder (low friction, challenging grasp) - Intentionally weak grasp force (50% of optimal)

**Execution Steps:** 1. Start pick sequence 2. Gripper grasps cylinder with insufficient force 3. During lift, F/T sensor detects drop (force spike → 0N) 4. System detects grasp failure 5. Robot returns to pre-grasp position 6. System displays error: "Grasp failed - Retrying with increased force" 7. Retry grasp with 100% force 8. Successfully lift and place object 9. Log failure event

**Success Criteria:** - Grasp failure detected within 500ms - Retry succeeds on 2nd attempt - No objects damaged - Error logged with timestamp and cause

**Concepts Demonstrated:** - Force/torque sensing (grasp verification) - Error detection (sensor-based) - Adaptive control (adjust grasp force) - State machine (error state → recovery state)

**Demo Video Deliverable:** Split-screen (RViz + real robot) showing failure and recovery

---

#### 1.2.4 Scenario M4: Calibration Wizard

**Objective:** Demonstrate ease of camera-robot calibration

**Setup:** - Checkerboard pattern (8×6, 25mm squares) on table - Camera uncalibrated (no prior hand-eye transform)

**Execution Steps:** 1. Launch calibration wizard 2. Wizard prompts: "Move robot to Position 1" (pre-defined joint angles) 3. Operator confirms, wizard captures image 4. Repeat for Positions 2-5 (different robot poses) 5. Wizard computes hand-eye transformation matrix 6. Validation: Place known object, system predicts position 7. Wizard displays error: "Calibration error: 2.3mm (PASS)" 8. Save calibration to `/config/camera_robot_tf.yaml`

**Success Criteria:** - Calibration completes in <5 minutes - Reprojection error <5mm - Validation test passes (object detected at correct position) - Calibration persists across restarts

**Concepts Demonstrated:** - Hand-eye calibration (eye-to-hand configuration) - Coordinate frame transformations - Usability (guided wizard for non-experts)

**Demo Video Deliverable:** Screen capture of wizard UI, narrated walkthrough

---

#### 1.2.5 Scenario M5: Safety E-Stop

**Objective:** Demonstrate emergency stop functionality

**Setup:** - Robot executing pick sequence (mid-motion) - E-stop button accessible

**Execution Steps:** 1. Start pick sequence 2. Robot moving toward object (50% into trajectory) 3. Operator presses E-stop button 4. Robot halts immediately, motors de-energized 5. System displays: "EMERGENCY STOP - Press Reset to Continue" 6. Operator releases E-stop, presses "Reset" 7. System prompts: "Return to Home? (Y/N)" 8. Operator selects "Y", robot returns to home position 9. System ready for next pick

**Success Criteria:** - Robot stops <100ms after E-stop pressed - No drift after stop (brakes engaged) - Cannot restart without deliberate reset action - Event logged with timestamp

**Concepts Demonstrated:** - Safety-rated E-stop (SIL 2) - Real-time control loop (fast response) - State machine (emergency state)

**Demo Video Deliverable:** Real-time video showing E-stop activation and recovery

---

### 1.3 2. Should Have Demo Scenarios

#### 1.3.1 Scenario S1: Pose Variation Handling

**Objective:** Pick objects in arbitrary orientations

**Setup:** - Objects: 3 rectangular boxes (100mm × 50mm × 30mm) placed at different angles - Orientations: 0°, 45°, 90° around vertical axis

**Execution Steps:** 1. For each object: - Detect object, estimate 6DoF pose (x,y,z,roll,pitch,yaw) - Compute aligned grasp (gripper oriented to object's longest axis) - Pick and place 2. Display pose estimates in RViz (TF frames)

**Success Criteria:** - All 3 objects picked regardless of orientation - Pose estimation error: <5° rotation, <5mm position - Grasp aligned to object geometry

**Concepts Demonstrated:** - 6DoF pose estimation (not just centroid) - Grasp planning (orientation-aware) - TF visualization

**Demo Video Deliverable:** RViz visualization showing estimated object frames overlaid on point cloud

---

#### 1.3.2 Scenario S2: Dynamic Conveyor Picking

**Objective:** Pick objects from a moving conveyor belt

**Setup:** - Conveyor belt moving at 0.1 m/s (constant speed) - Objects: 4 cubes placed at 200mm intervals - Camera: Mounted above belt, tracking motion

**Execution Steps:** 1. Vision system tracks objects on belt (optical flow / multi-frame tracking) 2. Predict object position at time of grasp (t__grasp = t__detect + t__plan + t__move) 3. For each object: - Estimate arrival time at pick zone - Pre-position robot (anticipatory motion) - Pick object in motion (dynamic grasping) - Place in static zone 4. Repeat until all objects picked

**Success Criteria:** - All 4 objects picked without stopping conveyor - Grasp success rate >90% - No collisions with conveyor

**Concepts Demonstrated:** - Motion prediction (object tracking) - Real-time planning (replanning during execution) - Trajectory execution (moving target)

**Demo Video Deliverable:** Side view + top view (camera) showing synchronized pick

---

#### 1.3.3 Scenario S3: Workspace Customization

**Objective:** Demonstrate GUI for defining pick/place zones

**Setup:** - Blank workspace (table only) - RViz2 with interactive markers

**Execution Steps:** 1. Operator opens zone definition tool in RViz 2. Draws pick zone (polygon tool, defines 2D boundary + height range) - Pick zone: 400mm × 400mm, height: 0-200mm 3. Draws place zone (300mm × 300mm, height: 50mm) 4. Draws exclusion zone (obstacle, 100mm × 100mm) 5. Save configuration to `zones.yaml` 6. Run pick-place with new zones 7. System only picks from pick zone, places in place zone, avoids exclusion

**Success Criteria:** - Zones defined in <2 minutes (intuitive UI) - Configuration saved and reloaded correctly - Robot respects zone boundaries (no picks outside pick zone)

**Concepts Demonstrated:** - Planning scene management - Collision objects (exclusion zones) - User-friendly configuration

**Demo Video Deliverable:** Screen capture of zone definition + robot respecting zones

---

#### 1.3.4 Scenario S4: Multi-Gripper Support

**Objective:** Swap gripper types and adapt grasp strategy

**Setup:** - Test with 2 gripper types: - Parallel jaw (for cubes, boxes) - Suction (for flat, smooth objects like PCBs)

**Execution Steps:** 1. **Test 1: Parallel Jaw** - Object: Cube - Grasp: Pinch grasp from sides - Success: Lifted with 20N force 2. Swap gripper (manual or auto-tool-changer) 3. System detects gripper change, loads suction gripper config 4. **Test 2: Suction** - Object: Flat PCB (100mm × 100mm) - Grasp: Top-down suction - Success: Vacuum pressure confirms seal (>0.5 bar)

**Success Criteria:** - Grasp planner adapts strategy per gripper type - Both gripper types successfully pick objects - Gripper swap detected automatically (if using tool changer)

**Concepts Demonstrated:** - End-effector modularity - Grasp planning (type-specific algorithms) - Hardware abstraction

**Demo Video Deliverable:** Side-by-side comparison of parallel jaw vs suction grasps

---

#### 1.3.5 Scenario S5: Performance Dashboard

**Objective:** Display real-time KPIs during operation

**Setup:** - Grafana dashboard open on separate monitor - System running continuous pick-place loop (10 objects)

**Execution Steps:** 1. Start pick-place loop 2. Dashboard displays (real-time updates): - Current state (SCAN, PICK, PLACE) - Objects processed (counter) - Cycle time (current, average, p95) - Success rate (%) - Error log (scrolling list) - CPU/GPU utilization graphs 3. Operator observes dashboard while robot works

**Success Criteria:** - Dashboard updates with <1 second latency - Metrics accurate (verified against ground truth) - Graphs show historical trends (last 10 minutes)

**Concepts Demonstrated:** - Monitoring & observability - Prometheus + Grafana integration - Real-time data visualization

**Demo Video Deliverable:** Split-screen (robot + dashboard) for 60 seconds

---

#### 1.3.6 Scenario S6: Simulation Validation

**Objective:** Run same workflow in simulation and real hardware

**Setup:** - Gazebo simulation with UR5e model, virtual camera, physics engine - Identical object (cube) spawned in sim workspace

**Execution Steps:** 1. **In Simulation:** - Launch: `ros2 launch vision_pickplace gazebo.launch.py` - Run pick-place workflow - Record: cycle time, trajectory, grasp success 2. **On Real Hardware:** - Launch: `ros2 launch vision_pickplace real_robot.launch.py` - Run identical workflow - Record same metrics 3. Compare results (sim vs real)

**Success Criteria:** - Simulation runs without errors - Cycle time difference <20% (sim vs real) - Trajectory similar (verified via joint plots) - Grasp success in both environments

**Concepts Demonstrated:** - Simulation fidelity (Gazebo) - Sim-to-real transfer - Testing without hardware risk

**Demo Video Deliverable:** Side-by-side video (Gazebo + real robot) synchronized

---

### 1.4 3. May Have Demo Scenarios (Advanced)

#### 1.4.1 Scenario A1: Bin Picking with Pile Segmentation

**Objective:** Pick objects from a cluttered bin (random pile)

**Setup:** - Bin: 400mm × 400mm × 200mm deep - Objects: 20 cubes randomly dumped (overlapping, various orientations)

**Execution Steps:** 1. Capture point cloud of bin 2. Segment individual objects (clustering, region growing) 3. Identify graspable objects (top layer, unoccluded) 4. Pick top object 5. Repeat until bin empty (re-scan after each pick)

**Success Criteria:** - All 20 objects picked (may take multiple scans) - No collisions with bin walls - Success rate >85% (some failures expected with occlusions)

**Concepts Demonstrated:** - 3D point cloud processing (PCL) - Segmentation (clustering) - Iterative scene understanding

**Demo Video Deliverable:** Time-lapse (accelerated 5x) showing bin emptying

---

### 1.4.2 Scenario A2: Collaborative Operation (Human-in-Loop)

**Objective:** Safely operate with human present in workspace

**Setup:** - Human (volunteer) standing near workspace - Vision-based human detection (YOLO person class) - Safety zones defined (inner: stop zone, outer: slow zone)

**Execution Steps:** 1. Robot executing pick-place at normal speed (100%) 2. Human approaches workspace (enters outer zone) 3. System detects human, robot slows to 50% speed 4. Human enters inner zone 5. Robot stops immediately (<100ms) 6. System displays: "Human detected - Waiting" 7. Human exits zone 8. After 2-second timeout, robot resumes

**Success Criteria:** - Human detected within 500ms - Robot stops before human contact - Speed reduction smooth (no jerks) - System resumes automatically when safe

**Concepts Demonstrated:** - Human-robot collaboration (ISO/TS 15066) - Vision-based safety (redundant to laser scanners) - Adaptive speed control

**Demo Video Deliverable:** Wide-angle video showing human and robot interaction

---

### 1.4.3 Scenario A3: AI Model Retraining Loop

**Objective:** Demonstrate model improvement from production data

**Setup:** - System collects 1000 pick images over 1 week (auto-logged) - Data scientist uses collected data to retrain YOLO

**Execution Steps:** 1. **Data Collection:** - System logs all RGB-D images + labels (bounding boxes) - Store in `/data/production_logs/` 2. **Retraining:** - Load data into Label Studio (review annotations) - Train YOLOv8 with fine-tuning (10 epochs) - Export to ONNX 3. **Deployment:** - Upload new model to robot - A/B test: 50% traffic to old model, 50% to new - Compare accuracy (new model: 96% mAP, old: 92%) 4. **Rollout:** - New model promoted to 100% traffic

**Success Criteria:** - Data collection pipeline works autonomously - Retraining improves accuracy (>2% mAP gain) - A/B test infrastructure functional - Deployment seamless (no downtime)

**Concepts Demonstrated:** - ML Ops (training pipeline, model registry) - Continuous improvement - A/B testing

**Demo Video Deliverable:** Screencast of MLflow experiments + before/after accuracy comparison

---

### 1.4.4 Scenario A4: Multi-Robot Coordination

**Objective:** Two robots working collaboratively in shared workspace

**Setup:** - 2× UR5e robots with shared workspace (overlapping reach) - 10 objects to be sorted (5 per robot)

**Execution Steps:** 1. Task allocator assigns objects to robots based on proximity 2. Both robots execute pick-place concurrently 3. Collision avoidance ensures no robot-robot collision 4. If paths conflict, lower-priority robot yields (waits)

**Success Criteria:** - All 10 objects sorted in <30 seconds (faster than single robot) - No collisions between robots - Load balanced (5 objects per robot)

**Concepts Demonstrated:** - Multi-robot planning - Conflict resolution - Distributed task allocation

**Demo Video Deliverable:** Overhead view showing both robots working

---

### 1.4.5 Scenario A5: Predictive Maintenance

**Objective:** Predict motor failure before it happens

**Setup:** - Logged data: motor temperatures, vibration, cycle counts (simulated 6 months) - Trained ML model (LSTM) predicts remaining useful life (RUL)

**Execution Steps:** 1. System monitors motor health in real-time 2. Model predicts: "Joint 3 RUL: 14 days" (based on temperature trend) 3. Alert triggered: "Maintenance recommended for Joint 3" 4. Maintenance scheduled (proactive, before failure) 5. Post-maintenance: RUL resets to nominal

**Success Criteria:** - Prediction accuracy >80% (validated on historical data) - Alert triggers 2 weeks before predicted failure - No unexpected downtime

**Concepts Demonstrated:** - Predictive analytics (ML for maintenance) - Time-series forecasting (LSTM) - Proactive maintenance

**Demo Video Deliverable:** Grafana dashboard showing RUL trends + alert

---

## 1.5 4. Demo Scenario Summary Table

| Scenario | Category | Duration | Complexity | Key Concepts |
|---|---|---|---|---|
| M1 | Must Have | 60 sec | Low | Vision, IK, motion planning, grasp planning |
| M2 | Must Have | 90 sec | Medium | Multi-object detection, task planning |
| M3 | Must Have | 90 sec | Medium | Error recovery, adaptive control, F/T sensing |
| M4 | Must Have | 5 min | Medium | Hand-eye calibration, transforms |
| M5 | Must Have | 30 sec | Low | Safety, E-stop, state machine |

| Scenario | Category | Duration | Complexity | Key Concepts |
|---|---|---|---|---|
| S1 | Should Have | 60 sec | Medium | 6DoF pose estimation, oriented grasping |
| S2 | Should Have | 120 sec | High | Dynamic picking, motion prediction |
| S3 | Should Have | 5 min | Low | Workspace customization, planning scene |
| S4 | Should Have | 90 sec | Medium | Multi-gripper support, hardware abstraction |
| S5 | Should Have | 60 sec | Low | Monitoring, Grafana, observability |
| S6 | Should Have | 90 sec | Medium | Simulation, Gazebo, sim-to-real |
| A1 | May Have | 5 min | High | Bin picking, point cloud segmentation |
| A2 | May Have | 90 sec | High | Human-robot collaboration, safety zones |
| A3 | May Have | 10 min | High | ML Ops, model retraining, A/B testing |
| A4 | May Have | 60 sec | Very High | Multi-robot coordination, conflict resolution |
| A5 | May Have | 5 min | High | Predictive maintenance, time-series forecasting |

**Total Demo Time:** ~40 minutes (all scenarios)

---

## 1.6 5. Demo Event Planning

### 1.6.1 5.1 Suggested Demo Flow (30-minute presentation)

**Segment 1: Introduction (5 min)** - System overview (slide deck) - Problem statement and value proposition - Live system walkthrough (components: robot, camera, control PC)

**Segment 2: Core Functionality (15 min)** - **M1:** Basic pick-place (2 min live + narration) - **M2:** Multiple objects (2 min) - **M3:** Error recovery (2 min) - **M4:** Calibration wizard (5 min, interactive) - **M5:** E-stop (1 min)

**Segment 3: Advanced Features (8 min)** - **S1:** Pose variation (video, 1 min) - **S2:** Conveyor picking (video, 2 min) - **S5:** Dashboard (live, 2 min) - **A2:** Collaborative operation (video, 3 min)

**Segment 4: Q&A (2 min)**

---

### 1.6.2 5.2 Demo Checklist

**Pre-Demo (1 hour before):** - [ ] Power on robot, camera, control PC - [ ] Verify network connectivity (ROS2 topics visible) - [ ] Run health check (all sensors green) - [ ] Load demo objects in workspace - [ ] Open dashboards (RViz, Grafana) on presentation display - [ ] Test E-stop button

**During Demo:** - [ ] Narrate each step clearly (explain what system is doing) - [ ] Pause for questions between scenarios - [ ] If failure occurs: explain error, show recovery (don't hide issues) - [ ] Point out key visualizations (bounding boxes, trajectories, TF frames)

**Post-Demo:** - [ ] Collect feedback (what impressed? what needs improvement?) - [ ] Record demo metrics (cycle times, accuracy, uptime) - [ ] Update demo scenarios based on feedback

---

## 1.7 6. Demo Risk Mitigation

| Risk | Mitigation |
|---|---|
| Network failure (ROS2 comms) | Pre-check network, have backup recordings |
| Camera not detecting object | Backup objects (high-contrast, known-good) |
| Grasp failure during demo | Tune gripper force beforehand, test 10× pre-demo |
| Robot E-stop during demo | Test E-stop recovery procedure beforehand |
| Laptop/display issues | Backup laptop with pre-loaded software |
| Power outage | UPS for critical systems |
| Software crash | Restart procedure documented, <2 min recovery |

---

## 1.8 7. Demo Metrics to Collect

| Metric | Target | Measurement Method |
|---|---|---|
| Cycle time (M1) | <10 sec | Timestamp start to finish |
| Multi-object throughput (M2) | >5 picks/min | Total time / objects picked |
| Grasp success rate | >95% | Successful picks / total attempts |
| Calibration time (M4) | <5 min | Stopwatch |
| E-stop response time (M5) | <100 ms | Oscilloscope (button press → stop) |
| Detection accuracy | >95% mAP | Test on labeled dataset |
| Pose estimation error | <5mm, <5° | Ground truth from CMM |
| Dashboard update latency | <1 sec | Wall clock vs dashboard timestamp |

---

## 1.9 8. Audience-Specific Demo Variants

### 1.9.1 For Technical Audience (Engineers, Researchers)

**Emphasize:** - Architecture (show ROS2 node graph) - Algorithms (explain YOLO, IK solver, RRT*) - Code walkthrough (brief, show key modules) - Performance benchmarks (latency, throughput)

**Recommended Scenarios:** M1, M2, M3, S1, S6, A3

---

### 1.9.2 For Business Audience (Managers, Executives)

**Emphasize:** - ROI (cycle time improvement, labor savings) - Ease of use (M4 calibration wizard) - Reliability (M3 error recovery, uptime metrics) - Dashboard (S5 KPI visualization)

**Recommended Scenarios:** M1, M2, M3, M4, S5

---

### 1.9.3 For Safety Officers / Regulators

**Emphasize:** - Safety compliance (ISO 10218, ISO/TS 15066) - E-stop functionality (M5) - Human detection (A2) - Audit logs (immutable logs, retention)

**Recommended Scenarios:** M5, A2, plus walk through safety documentation

---

### 1.9.4 For Customers / End Users

**Emphasize:** - Ease of deployment (M4 calibration) - Reliability (M3 error recovery) - Performance (M2 throughput) - Support (mention 24/7 support SLA)

**Recommended Scenarios:** M1, M2, M3, M4, S3, S5

---

## 1.10 9. Demo Video Production Guidelines

**For Each Scenario:** 1. **Introduction Slide (5 sec):** Scenario name, objective 2. **Setup Overview (5 sec):** Wide shot of workspace, label objects 3. **Execution (variable):** Multiple camera angles: - Robot close-up (gripper action) - Workspace overhead (full scene) - Screen capture (RViz, dashboard) 4. **Results (5 sec):** Success/fail indicators, metrics overlay 5. **Conclusion Slide (3 sec):** Key takeaway

**Technical Specs:** - Resolution: 1080p (1920×1080) - Frame rate: 30 fps - Format: MP4 (H.264 codec) - Audio: Narration (clear voice, background music optional) - Graphics: Lower-third text overlay with scenario name

---

## 1.11 10. Conclusion

This demo scenario collection provides: - **5 Must Have scenarios:** Core functionality validation - **6 Should Have scenarios:** Production-readiness features - **5 May Have scenarios:** Advanced capabilities showcase - **Total demo time:** 40 minutes (all scenarios) - **Audience-specific variants:** Tailored for technical, business, safety, customer audiences - **Risk mitigation:** Backup plans for common failures - **Metrics to collect:** Quantitative validation data

**Next Steps:** 1. Implement Must Have scenarios first (MVP) 2. Record high-quality demo videos for remote presentations 3. Create interactive demo for trade shows (allow audience participation) 4. Gather feedback and refine scenarios iteratively

---

**Document Status:** Complete **Last Updated:** 2025-10-18 **Author:** Product & Demo Team
**Review Status:** Pending Review