A Robust Real-Time Vision Pipeline for Planar-Pushing Shape Matching under Fisheye Distortion

Project: Cloud Robotics Manipulation Challenge (RGMC), IEEE ICRA 2026 — Cloud Robotics Track **Result:** Team 2nd Place Worldwide — independently authored vision module **Component:** Perception / shape-matching module (`src/otsu_pipeline.py`)

---

1. Problem

In the RGMC Cloud Robotics *Planar Pushing* task, a remote agent pushes a 3D-printed object so that its **bottom face** overlaps a target pose. The run is scored by the **time-integrated Intersection-over-Union (IoU)** between the object's bottom-face contour and the target contour over the first 120 s of execution.

This places three demands on perception:

1. **Geometric correctness** — the only camera is a wide-angle **fisheye** base camera, so raw pixels are heavily distorted and cannot be measured directly.
2. **Selective segmentation** — the objects carry a hexagonal extrude used for resetting; only the **bottom face** is scored, so the perception layer must isolate the bottom face rather than the whole object silhouette.
3. **Real-time, closed-loop operation** — the IoU is integrated continuously, so pose must be estimated every frame fast enough to drive the controller online.

The module described here estimates, per frame, the object's bottom-face contour and its in-plane orientation, then computes the IoU against the target.

2. Pipeline Overview

Each frame is processed through a fixed, deterministic sequence — no learned components and no per-run tuning:

Step	Operation	Purpose
1	Fisheye **undistortion** (per-robot calibrated `K`, `D`)	Restore metric geometry
2	**Red object detection** (HSV mask)	Localize the object
3	**Otsu thresholding** on the HSV **V-channel** inside the object mask	Isolate the darker **bottom face**
4	**Contour undistortion**	Map the detected contour into the undistorted frame
5	**Template projection** of the known 3D object model	Generate a metrically correct shape prior
6	**Brute-force 360° rotation search**	Align the template to the observation
7	**IoU** of aligned contour vs. target	Produce the score signal
8	Render (green = estimate, red = target)	Visualization / debugging

3. Method Detail

Red-object detection. The object is segmented in HSV using two hue bands `[0–7]` and `[170–179]` to capture red across the hue wrap-around, followed by an elliptical morphological *open* (k = 7) to suppress speckle. The **largest external contour** is taken as the object region.

Bottom-face isolation via Otsu. Within the object mask, the brightness (V) channel is thresholded with **Otsu's method**. Because the scored bottom face is consistently darker than the rest of the body, Otsu's data-driven threshold cleanly separates it. The pipeline also degrades gracefully: if the dark region is too small to be reliable, it falls back to the full object contour.

Template projection. Rather than fitting a generic polygon, the known 3D corner model of each object (square / circle / T-shape, defined in millimetres) is projected into image coordinates, giving a shape prior that is correct by construction under the calibrated camera.

Pose alignment by brute-force search. The template is aligned to the observed contour by a centroid-anchored rigid transform, sweeping orientation in **5° steps over the full 360°** (72 candidates) and selecting the angle that **maximizes IoU** with the observation.

4. Design Rationale & Advantages

The pipeline is engineered around the competition's emphasis on *generalization and robustness*. Each stage is chosen for a concrete strength:

- **Parameter-free, deterministic segmentation.** Otsu thresholding on the V-channel adapts its cutoff to the image content automatically, with no hand-set value to retune. It is single-pass and reproducible: the same input always yields the same bottom-face mask, and the threshold tracks lighting changes across robots and sessions without manual intervention.
- **A shape prior that is correct by construction.** Because alignment uses the projected 3D object model rather than a fitted polygon, the matched contour always has the exact geometry of the real object under the calibrated fisheye camera. This removes shape-fitting error as a source of IoU loss.
- **Globally optimal, initialization-free alignment.** The exhaustive 360° rotation sweep directly maximizes the **same IoU metric used for scoring**. It has no starting-guess dependence and cannot be trapped in a local optimum, so orientation is recovered reliably even for near-symmetric square and circle objects where local refinement is fragile.
- **Predictable, bounded latency.** The search evaluates a fixed 72 candidates per frame, so cost is constant and identical frame-to-frame — there is no iterative optimizer that may stall or diverge. This makes the module safe to run inside a real-time control loop.
- **No learned weights, no per-scene tuning.** The entire pipeline is classical and self-contained, which makes its behavior transparent, easy to debug under competition conditions, and directly portable across the platform's different robots.

5. Real-Time Performance

The full per-frame algorithm (undistort, detect, align, IoU) runs in **≈ 6 ms per frame** — roughly 160 FPS, far above the 30 FPS camera rate. Because the rotation search evaluates a fixed number of candidates, this latency is constant and predictable frame-to-frame, leaving ample headroom to drive the pushing controller in closed loop throughout the 120 s scoring window.

6. Outcome

This vision module — developed independently — provided a stable, distortion-correct, real-time IoU estimate that served as the feedback signal for the planar-pushing controller, contributing to the team's **2nd-place finish worldwide** in the RGMC Cloud Robotics track at IEEE ICRA 2026.

7. Limitations & Future Work

- Rotation resolution is capped by the 5° sweep; a coarse-to-fine refinement around the best angle would tighten alignment at negligible extra cost.
- Segmentation assumes the bottom face is the darker region; strong specular highlights could weaken the split, where a short temporal prior would help.
- The approach is purely geometric; fusing it with a lightweight learned detector could extend it to cluttered or multi-object scenes.