THE LEGO SORTER

Team 52: Rada Atanasova, Jadon Tsai, Nancy Fang, Memo Ozdincer

OPPORTUNITY

Toronto LEGO User Group (ToroLUG)has large piles of unsorted LEGOs which are timely to sort. Therefore, a large-scale, automated LEGO sorting device can help them save time on LEGO organization.

Objectives

Accuracy:

LEGO are accurately sorted

An open-sourced AI program is used to identify type and color

A light is shined on the LEGOs to minimize the variations in color due to shadows

Adaptability:

Adjustable to accommodate a wide variety of LEGO

The user manually sets a LEGO type to each compartment to adjust how the LEGO is sorted

The shutter opens and closes to accommodate at least an 8x8 plate

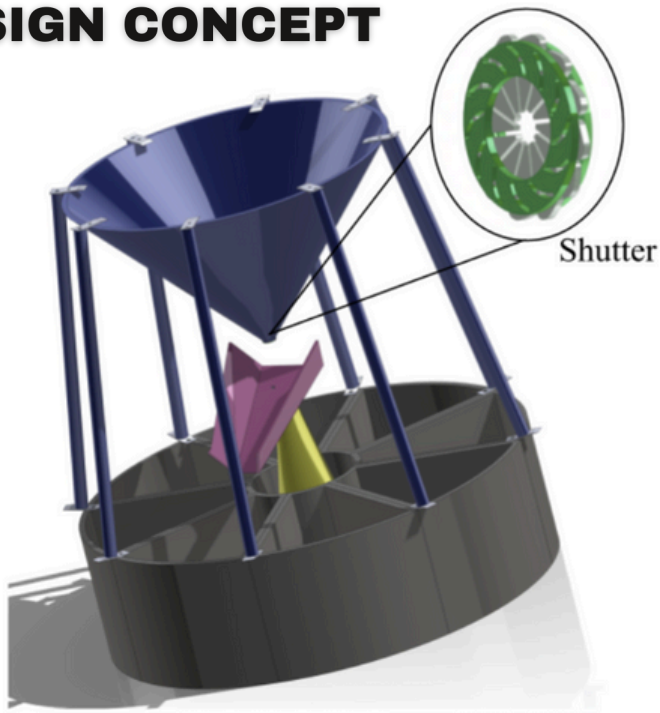
User-Centric design:

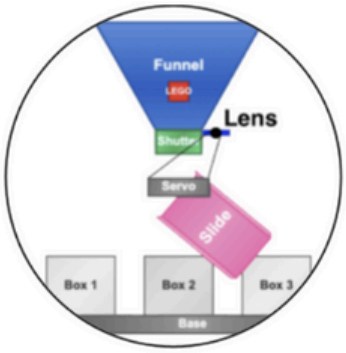
Seamless integration into existing systems and user needs

Operation is fully automatic apart from setup and offloading

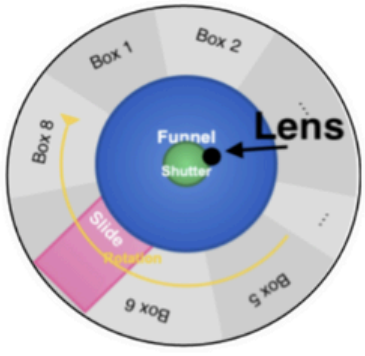
The design can fit on standard tabletop dimensions as requested by ToroLUG

FINAL DESIGN CONCEPT



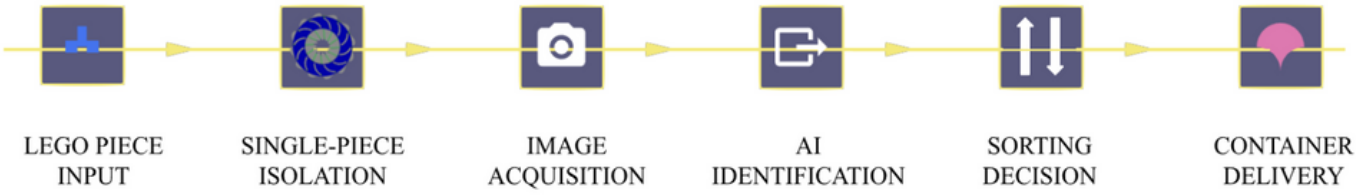


Material Overview

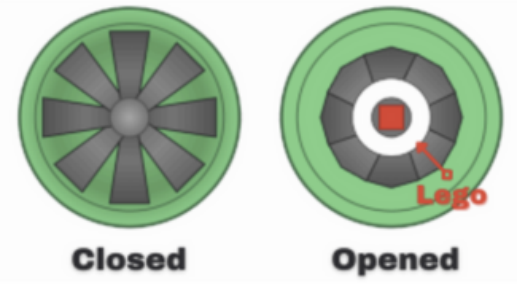


Top View

Sorting Process and Operation



1. User assigns a type of LEGO to each compartment.
2. User loads LEGO into the funnel.
3. A shutter at the bottom of the funnel accommodates various LEGO sizes while ensuring only one falls through at a time. The shutter opens until a LEGO piece is sensed on the camera then closes.



Verification timings of steps #4, #5:
(Ensuring accurate analysis in motion)
- *LEGO takes 0.3s to leave frame, 0.7s to exit slide.*
- AI analyzes in 0.1s, servo rotates in 0.4s.

Tested using 10° decline; camera 12cm from 2x2 & 3x3 bricks.

4. The lens captures an image of the LEGO and an open-sourced AI identifies the LEGO type in reference to the LEGO types set by the user. LEGO not matching the descriptions set by the user are stored in an "other" compartment.
5. The servo spins the slide before the LEGO slides off so the LEGO lands in the correct compartment.

Verification

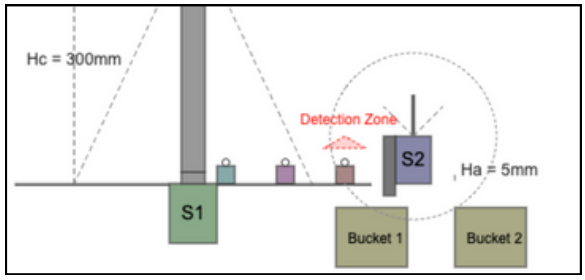


Figure 2. Side view of Physical Prototype Diagram

Verification Tests with Poor Performance ⚠ + Next Steps

1. (Req. 9) Shutter/funnel feeders couldn't place single LEGOs consistently. **I.** Plate cannot support falling LEGO weight**.**

Fix: I: Replaced plate with a slide. **II**. LEGOs now always fall in same position.

2. (Req. 4) The sorter required too many rounds of refilling as it had only two bins.

Fix: Redesigned feeder (slide instead of a plate) allowing an increase from a max of 2 bins.

Well-Performing Verification Tests ✅

(R1) Ease of Use: Intuitive sorter design

(R7) Customizability: Adapts to specific sorting habits.

Tested using **proxy tests: User Surveys, (III), Accuracy measurement (IV), Customizability (V).**

Next Steps

1. Higher-fidelity prototypes should be made and tested to ensure they adhere to product requirements
2. Further validation should be conducted with a larger variety of ToroLUG members to ensure continued approval of the primary stakeholder.
3. The requirements framework should be continuously revised to suit stakeholder needs.