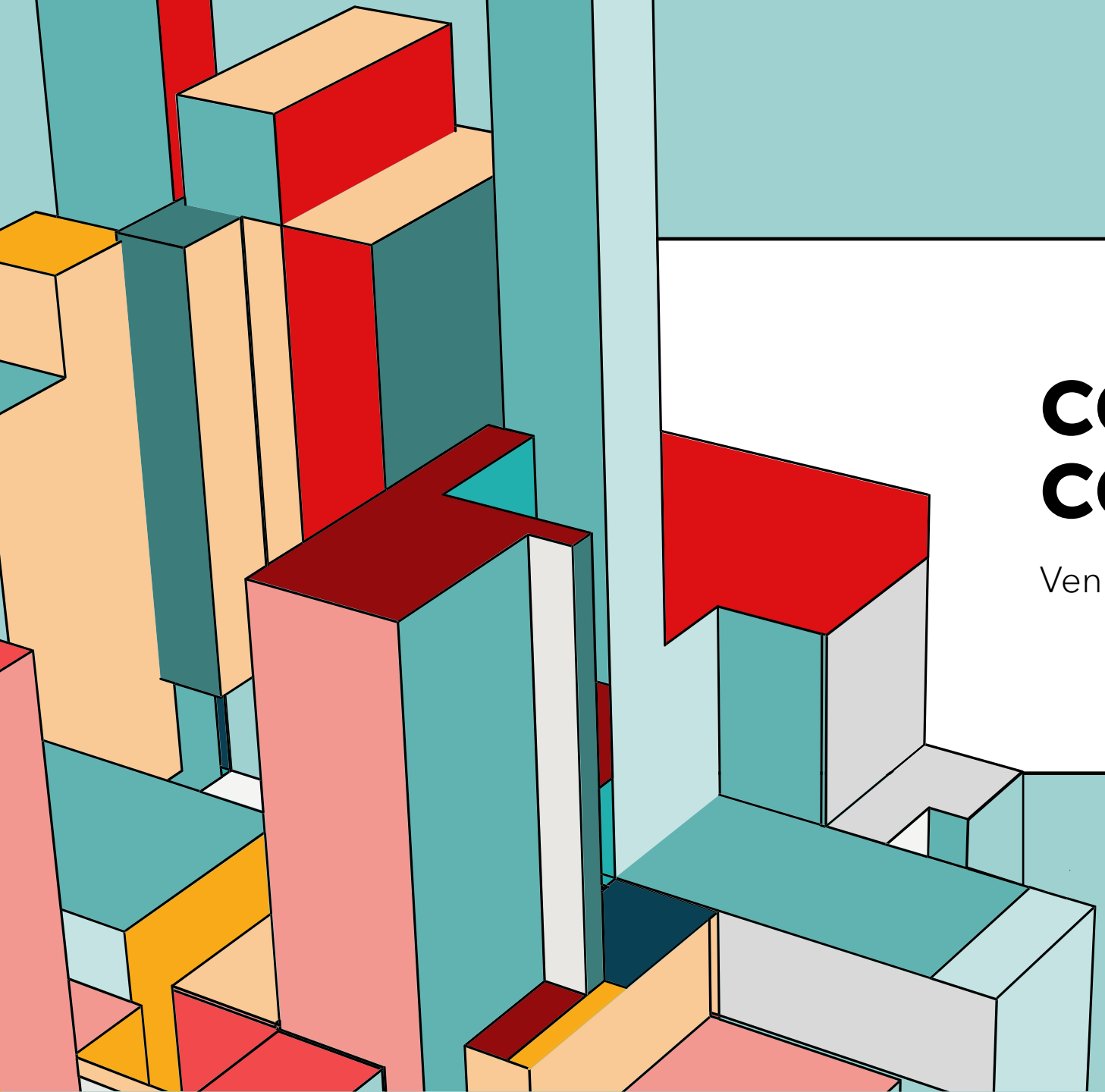


COZMO QUBE CONSTRUCTION

Venkat Gella

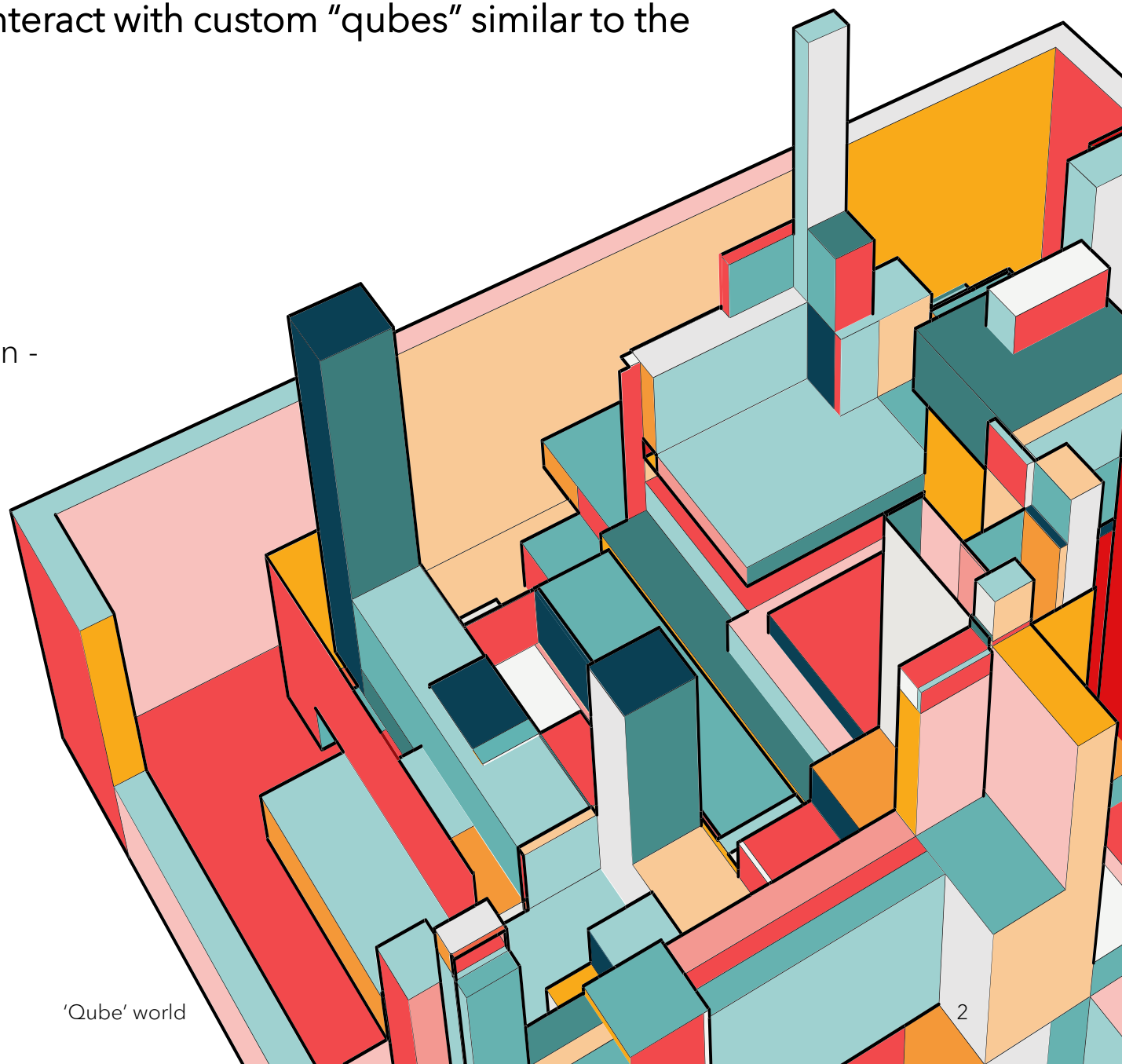
PROJECT IDEA:

Can cozmo interact with custom “qubes” similar to the light cubes?

Project: Build any 2d structures from magnetic Qubes that Cozmo can interact with similar to the lightcube

State Machine workflow: Qube Detection -> Dock Cube -> Placement of qube, repeat (safety checks in between)

Demo: Build this shape



‘Qube’ world

SUBPROBLEMS

Subproblem: Qube design

-> Solution: Qubes have magnets in the center at, cozmo has magnets on each lift post; pearson correlation between image after drive to cube and corner of marker to determine success of dock

Subproblem: Qube macrostructure assembly

->Solution: Qubes alternate magnetic poles; FSM alternates poles during retrieval; represent structure as a “stack” of sequential moves

Subproblem: Accuracy of Cozmo’s SDK object coordinates for the ‘Qubes’ is lacking

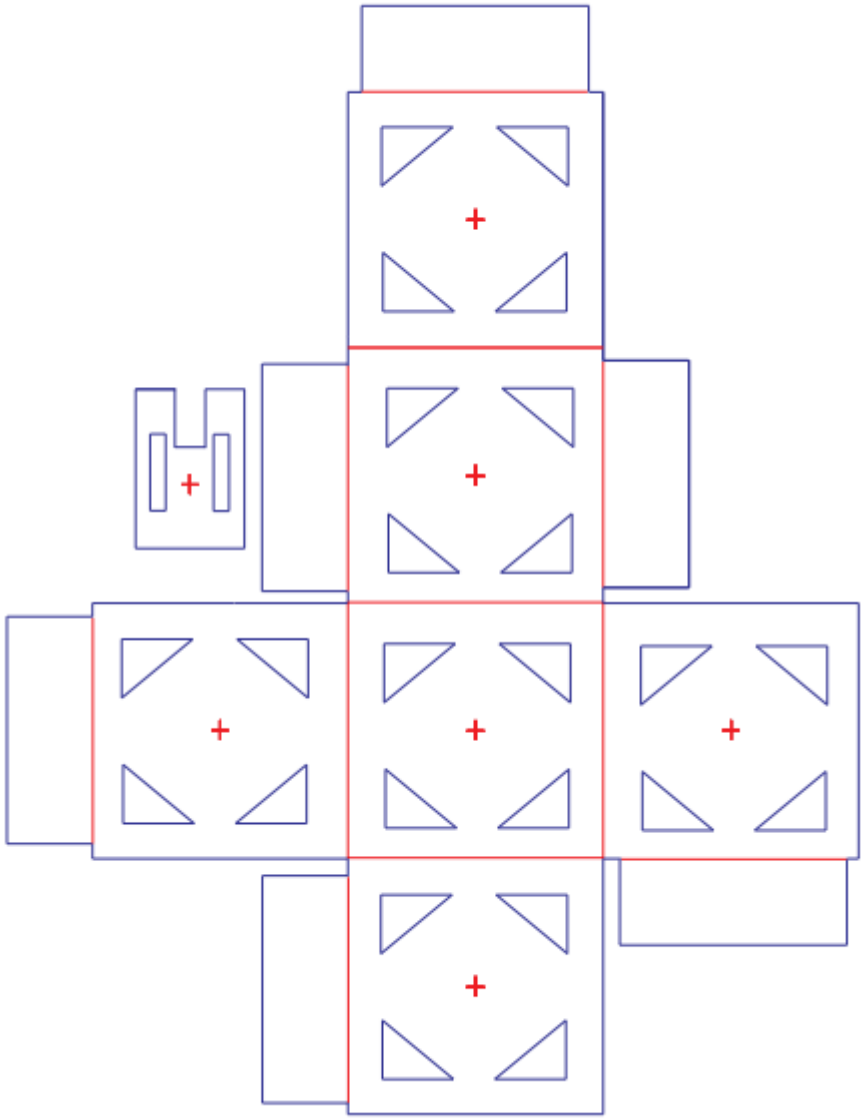
-> Solution: ArUco ‘Qubes’!

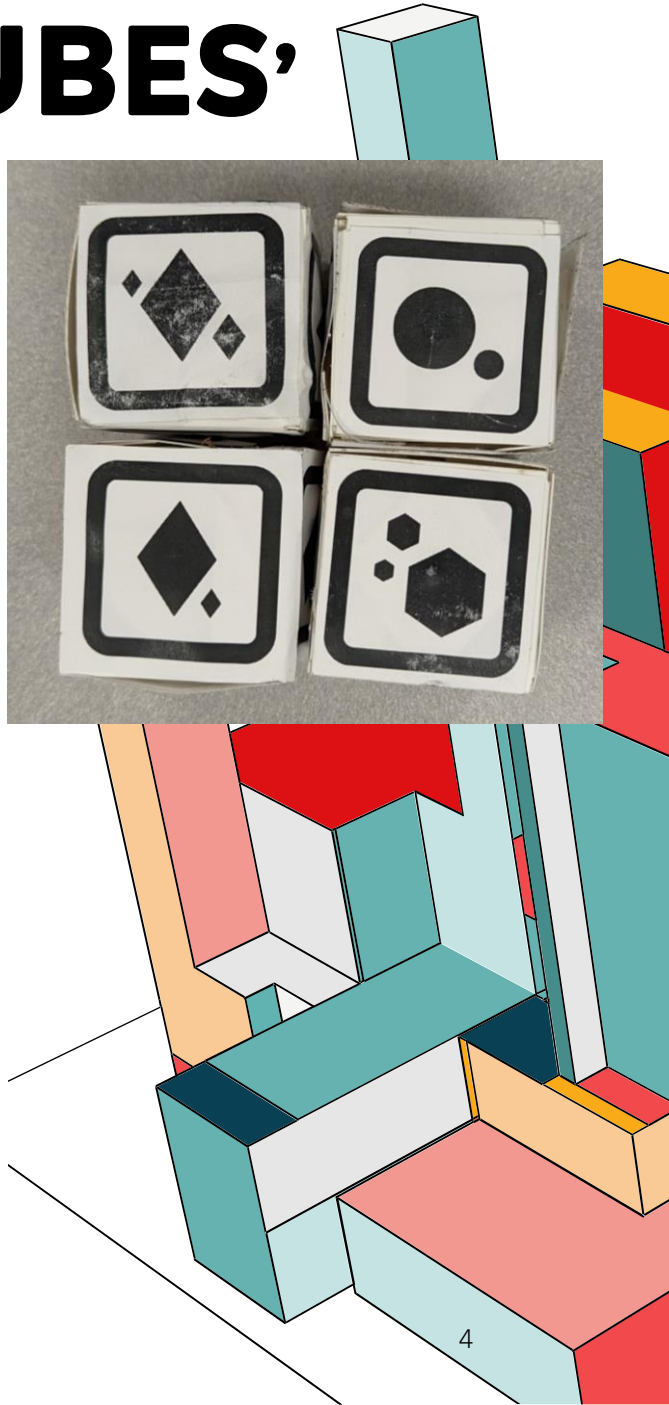
‘Qube’ world

‘QUBES’

Plastic Cubes with Magnets in Center

- Qubes alternate poles and cozmo interacts via lift magnetic interaction attachments
- Can be assembled without hot-glue using sticker
- Lightweight design allows easy interaction with cozmo lift magnets and structural integrity of ‘Qube’ interactions (can actually build structures that stay together!)





Cozmo SDK gives coordinates

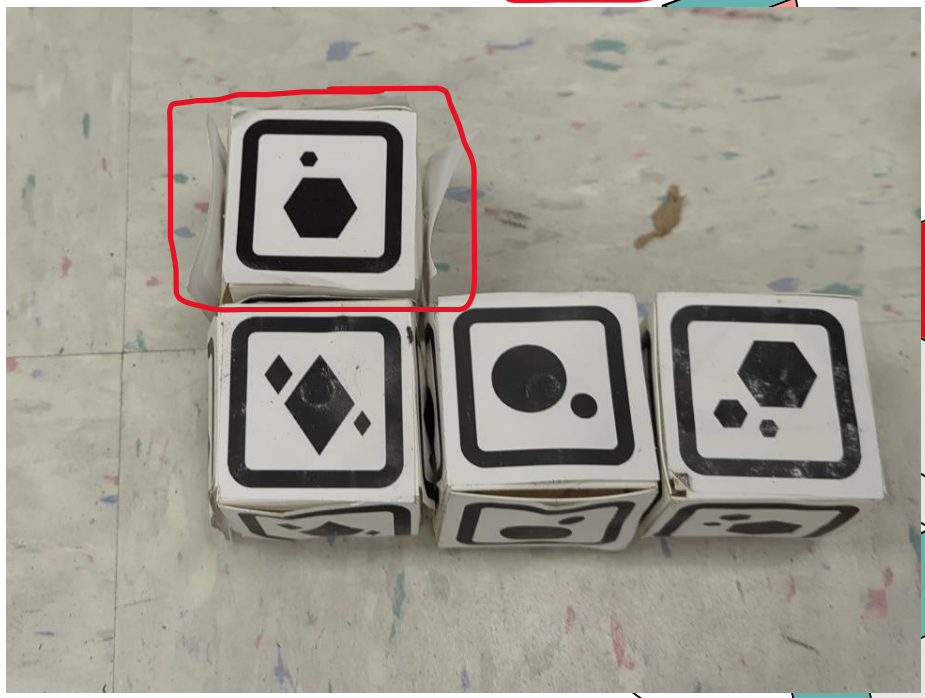
- custom marker object interface is good but inaccurate
- need aruco markers to calibrate location during construction

‘Qube’ world

REPRESENTING THE STRUCTURE

The structure is represented as a “stack”, implemented as a list, of two-tuples (number of qubes, compass direction) with the last element representing the starter ‘qube’.

[(2, ‘east’), (1, ‘south’), (1, ‘starter’)]



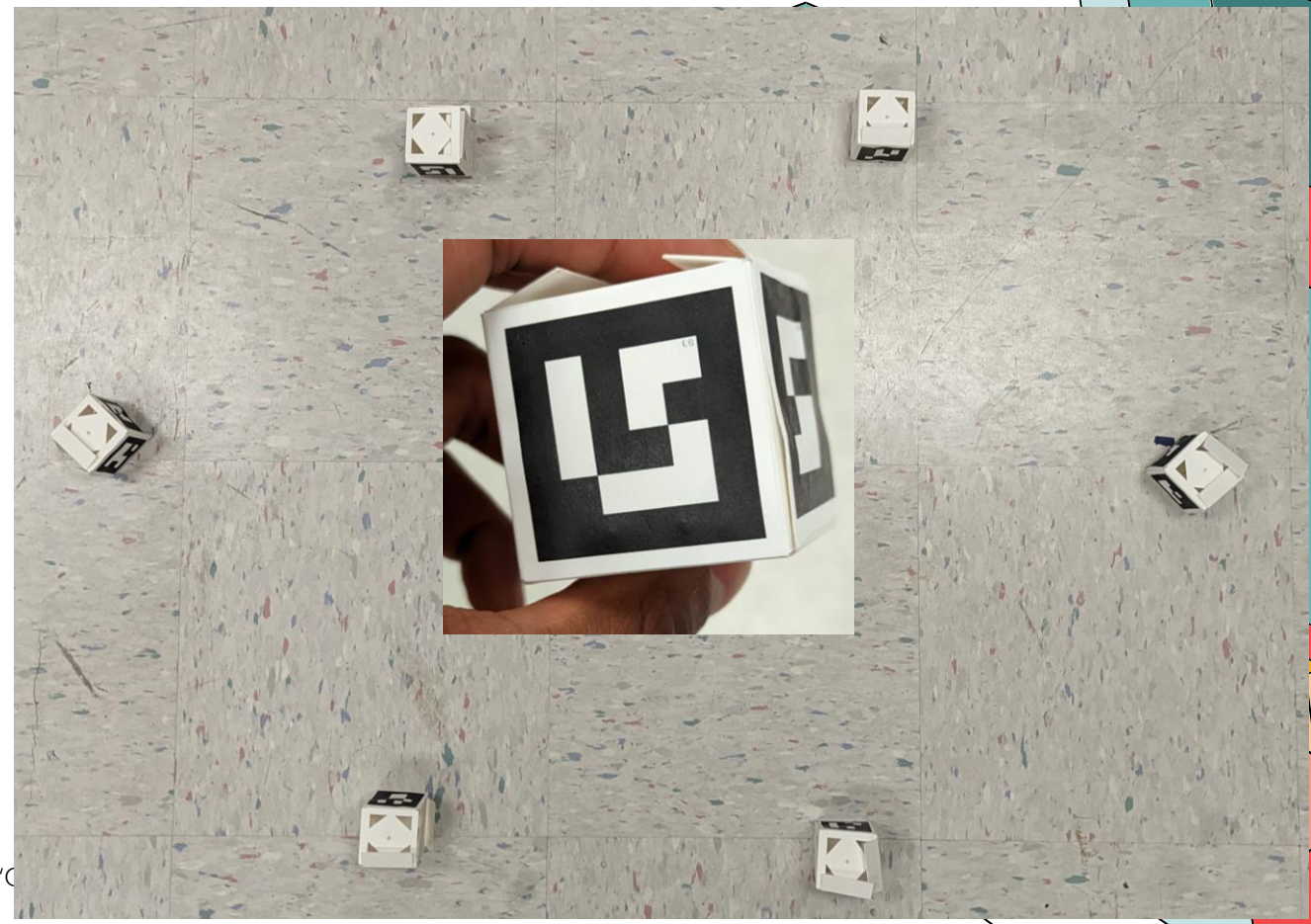
Implementation currently supports any structure where only one ‘qube’ is connected to one other ‘qube’. Could be expanded to represent multiple ‘qubes’ attached to one ‘qube’, however there are lots of collisions to consider

ARUCO ‘QUBES’

The ‘board’ for Qube Construction requires an arUco qube at each corner (500mm from cozmo) and an arUco qube between front side and back side of ‘board’ (400mm from cozmo).

Can be placed anywhere in the environment to add accuracy to cozmo’s sense of location

ArUco ‘qubes’ in the environment helped drive ‘qubes’ to the precise locations of other qubes

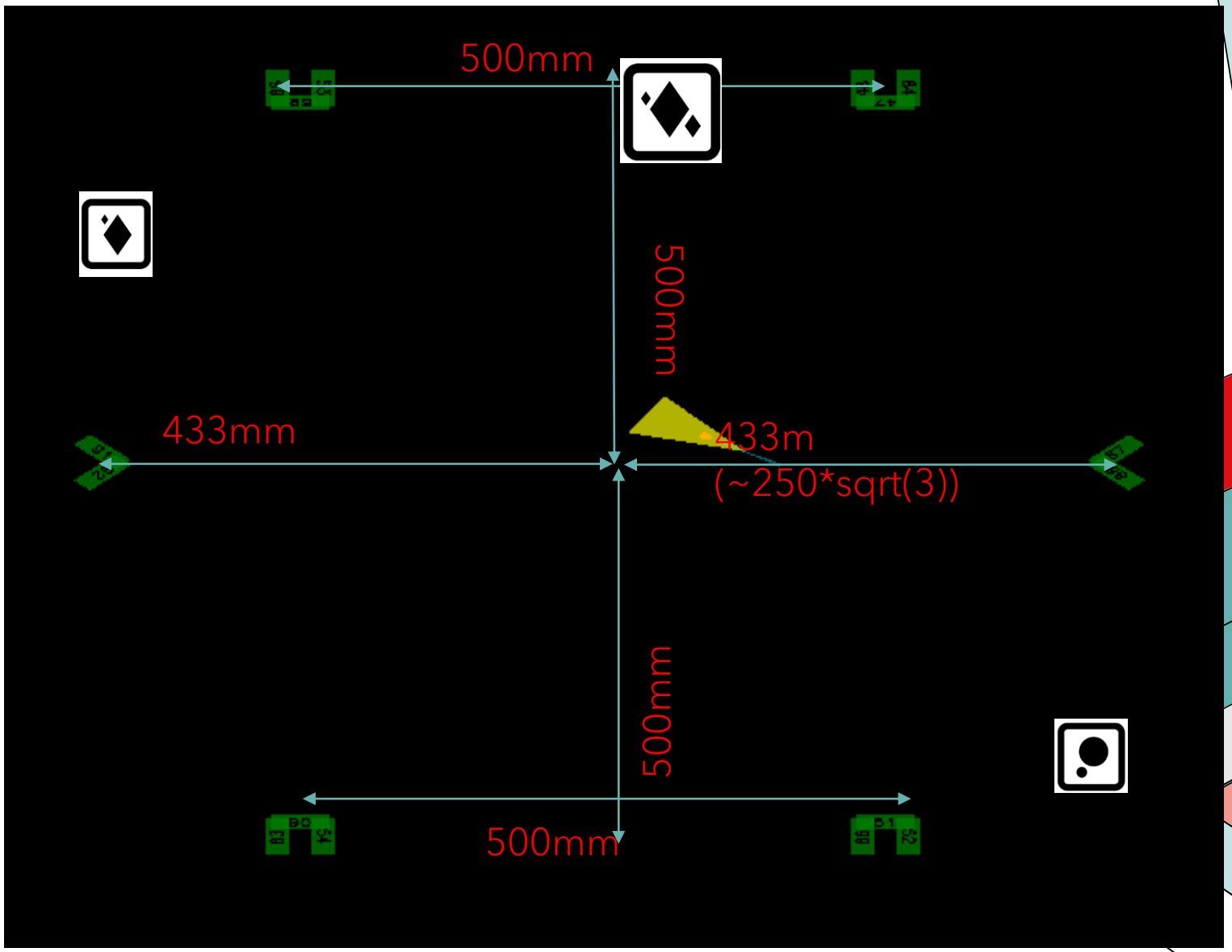


Can be used in any cozmo task to provide better precision in navigating worldmap

Since these arUco ‘Qubes’ don’t trigger ‘collides’ events, they can be pushed around so be careful to navigate around them explicitly in the code (run a check for each move that navigates around the aruco qube

ARUCO ‘QUBES’ BOARD AND CONSTRUCTION QUBE PLACEMENT EXAMPLE

If you use this placement and the y coordinates of the 2 shape cubes don’t correlate well with 433mm then the battery needs charge



CHATGPT UI

Can be skipped by typing “tm” and pressing enter

This can be further developed for more intuitive user inputs, perhaps even shapes or letters.

This demonstrates chatgpt can create python objects somewhat reliably. I use an except handler for correct code.

Organize queries in the following format:

“What is the list representation of one cube”… with “one cube” explicitly written out for the starter cube, and each cube in the structure written out in the format “one cube” + direction. for a southern cube “one cube down”, for an eastern cube “one cube to the right” for a western cube “one cube to the left” and for a northern cube “one cube up”. If I change the interface to “qube” in place of “cube” it does not work, GPT knows the word cube. Sometimes order is mixed up between two elements in the stack.

```
runtm what is the list representation of one cube one cube down one cube to the right?
C> made move stack: [(1, 'east'), (1, 'south'), (1, 'starter')]
building this: [(1, 'east'), (1, 'south'), (1, 'starter')]

True
[]
currently looking for['CustomMarkerObj-12', 'CustomMarkerObj-10', 'CustomMarkerObj-11'
[(1, 'east'), (1, 'south'), (1, 'starter')]
True
[]
currently looking for['CustomMarkerObj-12', 'CustomMarkerObj-10', 'CustomMarkerObj-11'
[(1, 'east'), (1, 'south'), (1, 'starter')]
True
[]
currently looking for['CustomMarkerObj-12', 'CustomMarkerObj-10', 'CustomMarkerObj-11'
[(1, 'east'), (1, 'south'), (1, 'starter')]
```

```
<PlaceStartQube placestartqube>

C> clear_landmarks: Landmarks are fixed in this particle filter.
launching opengl event loop
0
1
2
3
4
5
6
7
wt_inc 150.0 applied for max_weight -798.7028609078725
wt_inc 150.0 applied for max_weight -1164.3814294524705
wt_inc 150.0 applied for max_weight -1522.6045019555754
wt_inc 150.0 applied for max_weight -1372.6045019555754
wt_inc 150.0 applied for max_weight -1222.6045019555754
8
wt_inc 150.0 applied for max_weight -1999.8540886439343
wt_inc 150.0 applied for max_weight -1849.8540886439343
wt_inc 150.0 applied for max_weight -1699.8540886439343
9
wt_inc 150.0 applied for max_weight -1609.1583076092754
wt_inc 150.0 applied for max_weight -1510.0804768858025
wt_inc 150.0 applied for max_weight -1406.6893199526507
wt_inc 150.0 applied for max_weight -1289.2947594249042
show particle_viewer
Type 'h' in the particle viewer window for help.
C> tm what is the list of one cube one cube up one cube to the lef
C>
C> made move stack: [(1, 'north'), (1, 'west'), (1, 'starter')]
building this: [(1, 'north'), (1, 'west'), (1, 'starter')]

True
[]
currently looking for['CustomMarkerObj-12', 'CustomMarkerObj-10',
[(1, 'north'), (1, 'west'), (1, 'starter')]
```

```
<PlaceStartQube placestartqube>

C> clear_landmarks: Landmarks are fixed
0
1
2
3
4
5
6
7
wt_inc 150.0 applied for max_weight -385.4082812068824
wt_inc 150.0 applied for max_weight -397.9249437968256
8
wt_inc 150.0 applied for max_weight -549.0628806563607
wt_inc 150.0 applied for max_weight -399.0628806563607
9
tm what is the list of three cubes in a line
C> made move stack: [(2, 'south'), (1, 'starter')]
building this: [(2, 'south'), (1, 'starter')]

True
[]
currently looking for['CustomMarkerObj-12', 'CustomMarkerObj-10',
[(2, 'south'), (1, 'starter')]
<CustomCubeObj-10 5: (-293.7,-325.3, 15.9) @ -163 deg. visible>
```

The current program does not check if the user input structure is a valid structure, cozmo will build if it fits the format

Try and except handler ensures cozmo will always build a structure, the default is defined in code as variable “move_stack_example = [(1,’west’), (1,’south’), (1, ‘starter’)]
The user input that creates this move stack is “what is the list representation of one cube one cube down one cube to the right.

WHAT THIS MEAN FOR FUTURE COZMO

More objects to interact with

Quality qubes that can be recreated and offer more diversity to current light cube objects

Cozmo games: can be used for a physical communication between cozmos

One cozmo can interact with another or instruct action to another cozmo by building some structure

Build many unique structures

The handling of structures as a list of directional moves can be expanded to build any possible 2d structure of magnetic qubes

CURRENT LIMITATIONS/IMPLICATIONS OF PROJECT

Limitations:

Lots of qubes = Lots of GoalCollides

Navigating the 'board' is computationally expensive; many (>5) qubes means many goalcollides pathtopose events; user can physically rearrange qubes to solve this

Takes a long time to assemble

5-10 minutes >=3 qube structures

Can endlessly just tour qubes

Collision events signal Cozmo to slightly move and try again; this can loop, leaving cozmo touring the cubes

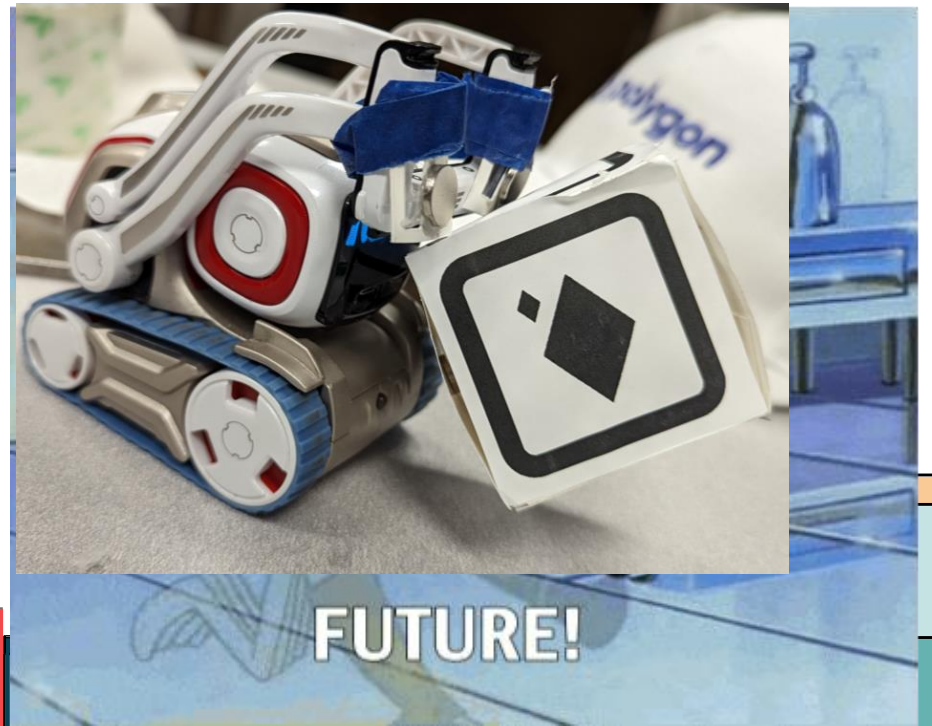
Battery affects performance

If cozmo is continuously turning, charge and try again the degrees measurement precision decreases with battery as well as accuracy of the particle filter

Implications:

In general: construction robotics

Stack representations of construction structures and reference aruco "qubes" is a meaningful solution to construction problems in robotics; since the moves are sequential this could be applied elsewhere as a methodology for brick-and-mortar construction robots



DEMO LINK: