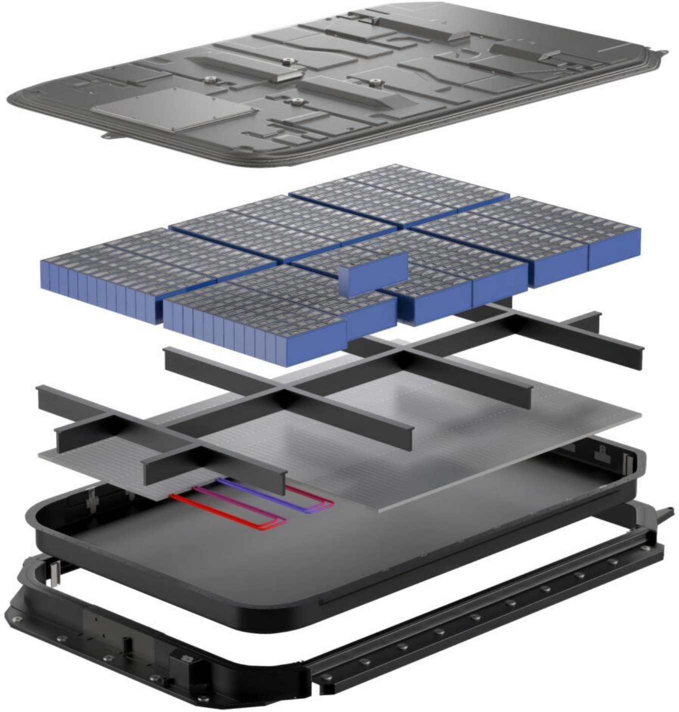
Automated Battery Pack Disassembly

Computer Vision Lead Case Study
Circu Li-ion

Emre Girgin

Task

- Automated disassembly of EV battery packs
- Particularly with computer vision techniques
- Analysis of requirements
- Designing a sophisticated pipeline
- Planning of disassembly instructions
- Hazard detection capability
- Possible improvements
- Example implementations



Tasks and Workflow

1. Localization of battery pack
2. Removal of casing
    a. Detection and 3D localization of screws and grabbing pose of the upper case
3. Task planning
    a. 2D detection and 3D localization of components: screw heads, screw holes, fasteners, bolts, nuts, metal and plastic plates, wires, battery modules, cable connectors, BMS unit, thermo sensors, busbar, cable guides, gas vent, ...
    b. In which order to dismantle components
4. Visual Precautions
    a. Detection of potential sparks and overheats

Vision Problems

- Object detection
  - 2D localization of assembly components on image plane
- Pose estimation
  - 2D/3D orientation of assembly components
- 3D Localization
  - 3D position of assembly components in the camera coordinate frame
  - Depth map extraction
- Visual Task Planning
  - Determining the order of disassembly
- Visual Precautions
  - Detection and 2D localization of hazardous signals

Proposed Hardware Overview

Stereo Cameras (~500$):

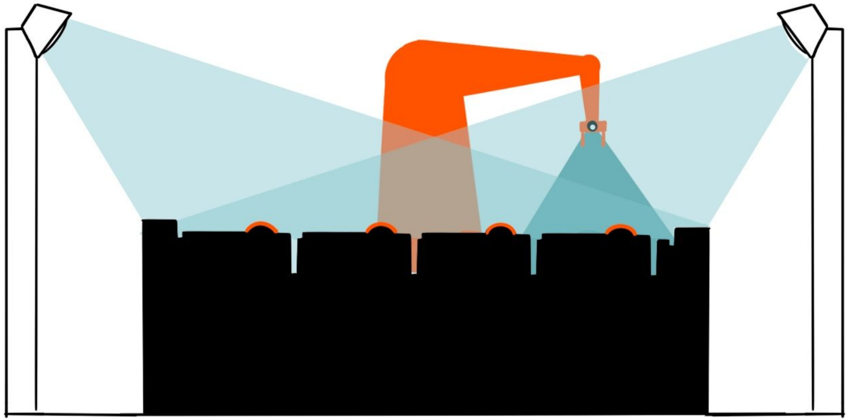
- Object Detection
- Pose Estimation
- Position Estimation
- Visual Task Planning

Thermal Camera (~ 350$):

- Hazard Detection

Wrist Camera (6.5k $):

- Object Detection
- Position Estimation

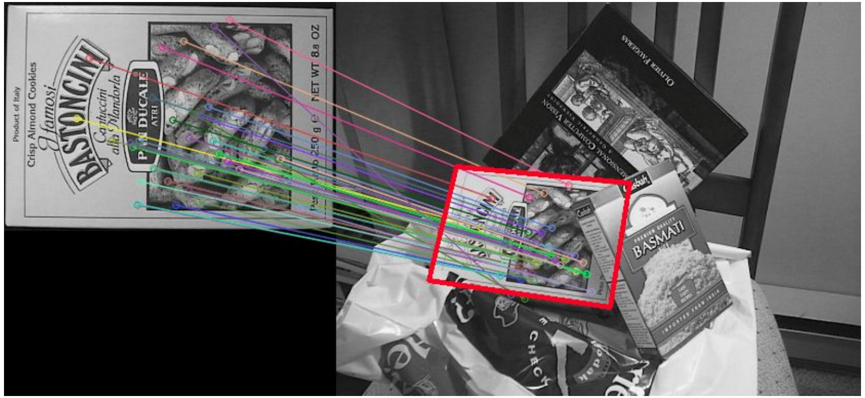


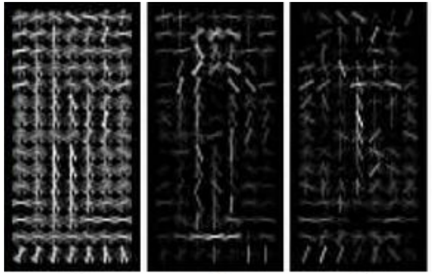
Processing Unit:

- Jetson Nano (~500 $)
- Internet connection
  - Diagnostics and software update

Vision Capabilities: Traditional Approaches

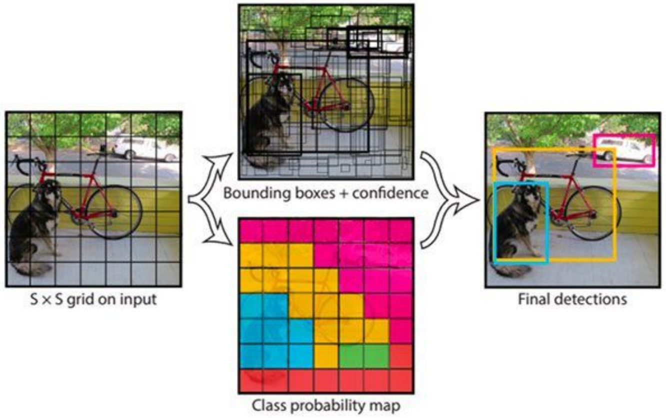
- Feature matching, sliding windows, morphological operations, ...
- Pros:
  - Does not require annotated data
  - Easier to interpret
  - Lightweight
- Cons:
  - Limited performance
  - Requires manual feature engineering
  - Very hard to generalize





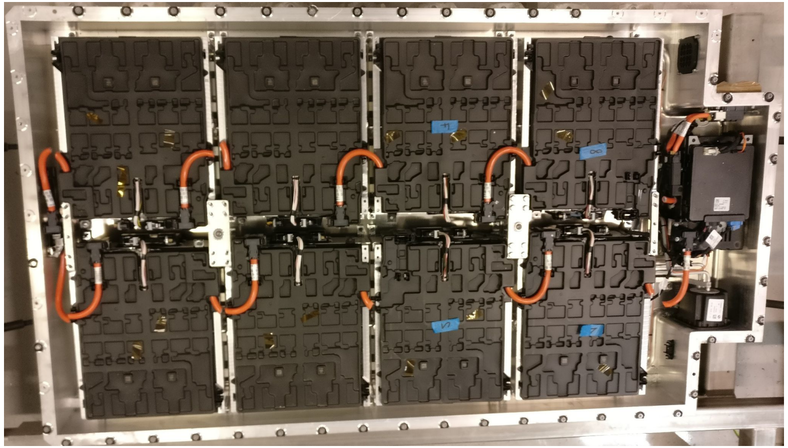
Vision Capabilities: Deep Learning

- YOLO, SSD, ...
- Pros:
  - Better performance
  - End-to-end, no need to manually extract features
  - Mostly run in real-time
- Cons:
  - Requires lots of labeled data: Inadequate data leads overfitting
  - Requires fairly powerful processing unit
  - Black-box: hard to explain why it has failed

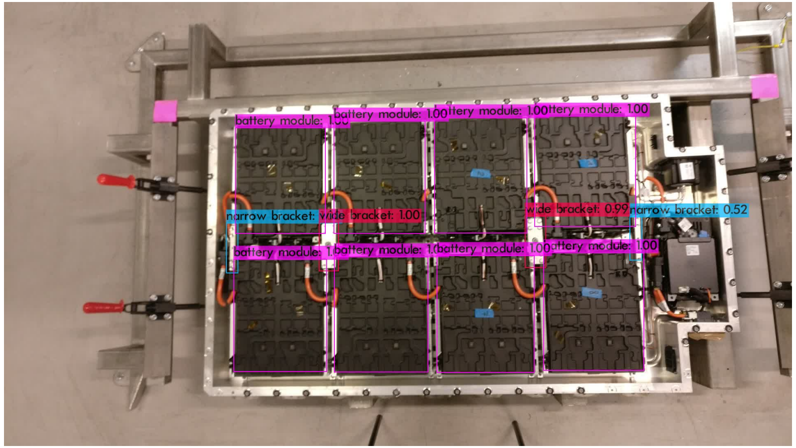


Vision Capabilities: Object Detection

- 2D Detection and localization of components to be unmounted

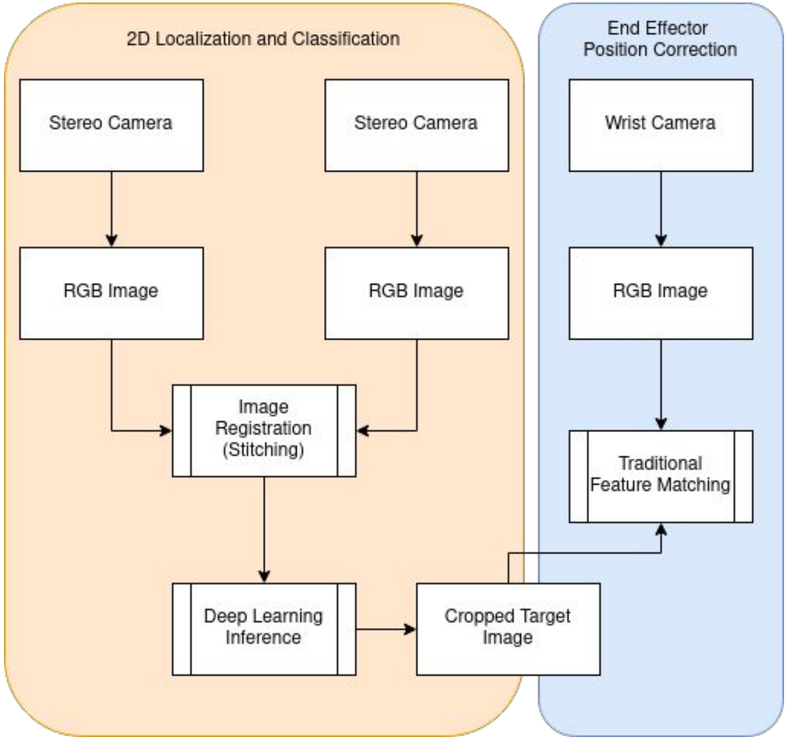


Input



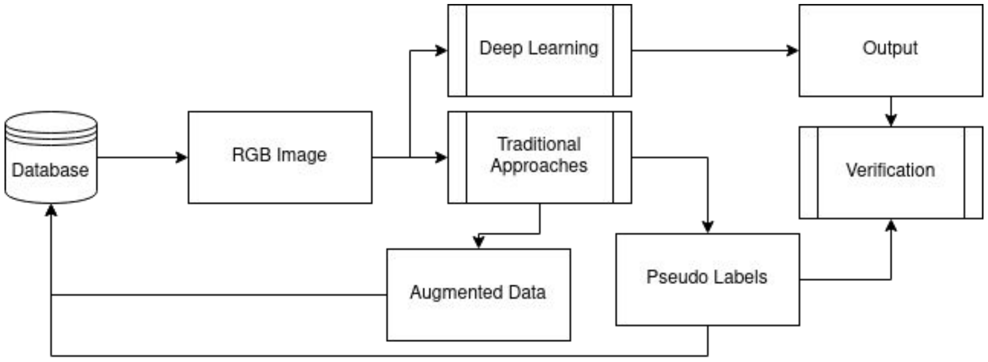
Output

Proposed Object Detection Pipeline



Object Detection Workflow

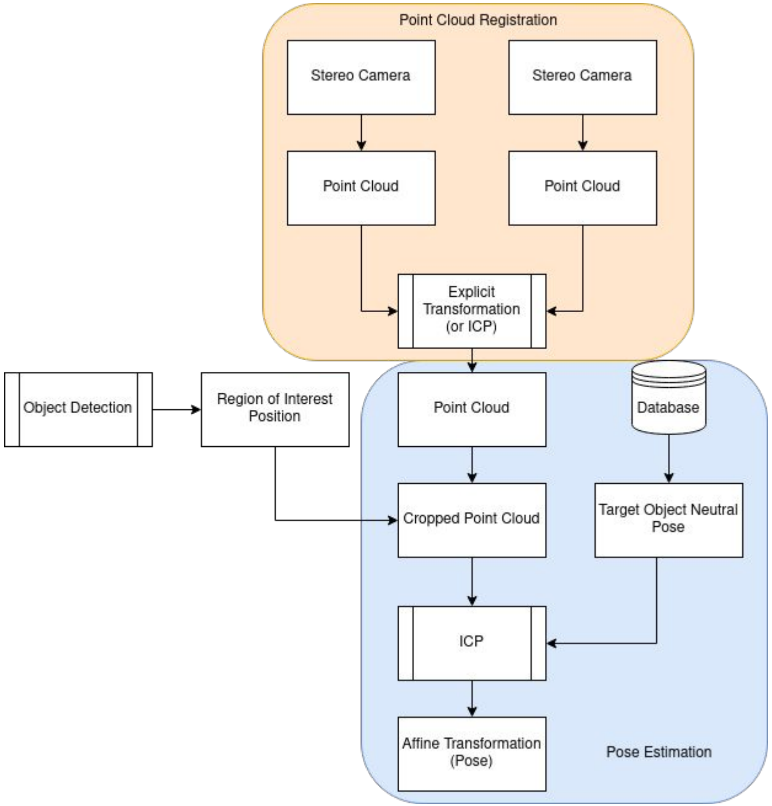
- Deep learning based object detection is an advanced field.
- Traditional methods are also very useful.



Cooperation with traditional methods

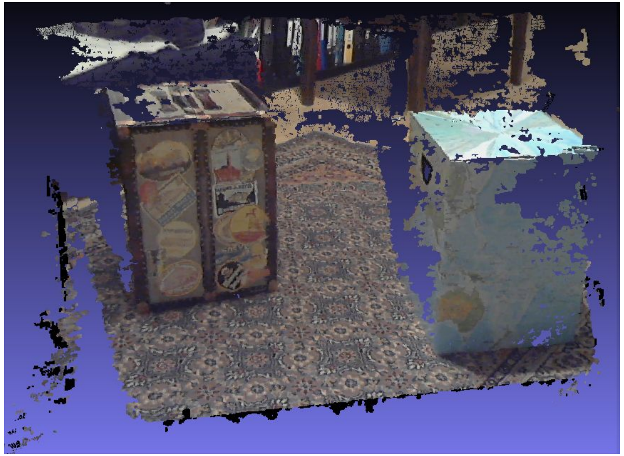
Proposed Pose Estimation Pipeline

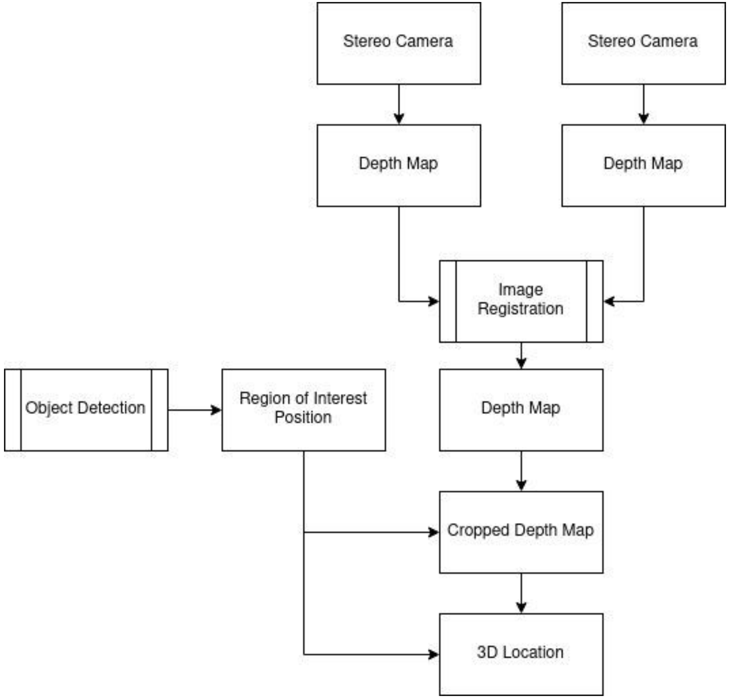
- Object surface is a clue
- Drawbacks:
  - Reliance on object localization
  - Requires 3D model of component
- Relative position of components is another clue
  - In the example implementation
- Deep learning
  - Existing domains are quite different than battery packs
  - Surface normal estimation might be useful for modules



Proposed Position Estimation Pipeline

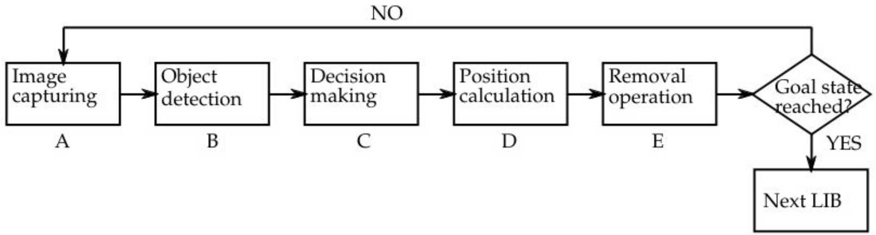
- Input of inverse kinematics solution
- Straightforward with stereo cameras



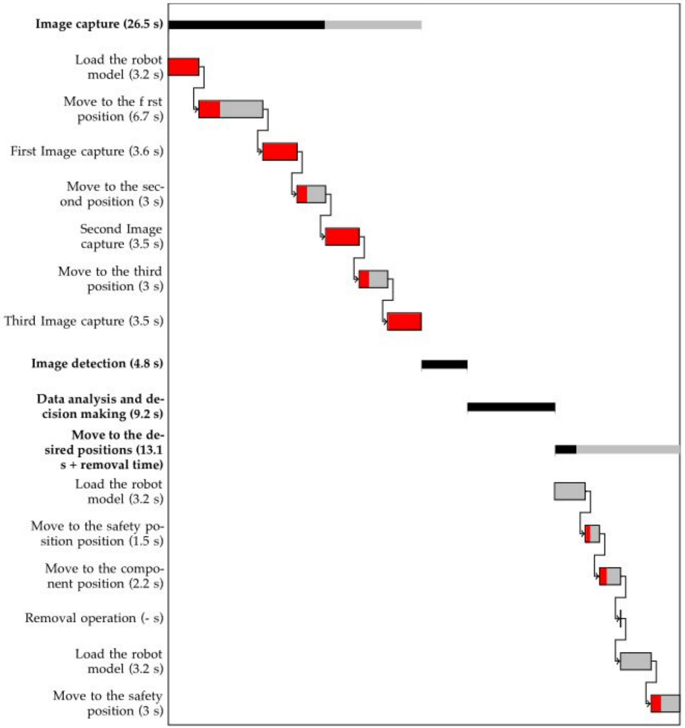


Example Visual Task Planning

- The order of disassembly may vary
- Inter-relations between components are crucial
  - More on that later on
- Main loop requires trial & error
  - May lead to wear or tears
- PDDL integration may be useful



Task Planner Main Loop



Example task order and timings

Performance Metrics

Object Detection

- mAP
- IoU
- Speed
- Per-class F1-score

Pose Estimation

- Rotation Error
- Speed

Position Estimation

- L2 position distance
- IoU
- Chamfer distance (3D)
- Speed

Task Planner

- Number of completed plans
- Number of failed loops
- Difference between predicted and actual time spent
- Speed

Deployment

Edge Computing		Cloud Computing	
Pros	**Cons**	**Pros**	**Cons**
• Low latency • Privacy and security • Offline operation • Bandwidth efficiency (network) • Real-time processing	• Limited resources (computing & space) • Challenging maintenance • Challenging update • Dependency to manufacturer • Scalability	• High and flexible performance (better DL models) • Scalability • Easy to maintain and update • Access anywhere	• Low latency • Security • Internet connection • Monthly cost • Data storage regulations

Point cloud and images require high bandwidth. Object detection modules are lightweight to run.

Hybrid approach: Process on edge, update over cloud.

Safety Precautions

- Thermal cameras can detect sparks and overheats
- Setting a threshold to the image works fine
- Thermal camera FPS is crucial

Continuous Improvement

- Generation of component through deep learning contributes generalization
  - Generated 3D objects can be combined in a battery pack in simulation
  - Each component is auto labeled
  - Diffusion models, novel-view synthesis (NeRFs, 3DGS, ...)
- Simulations are also useful in that sense
  - The variation in data is also possible
  - Illumination, rotation, occlusion, etc.

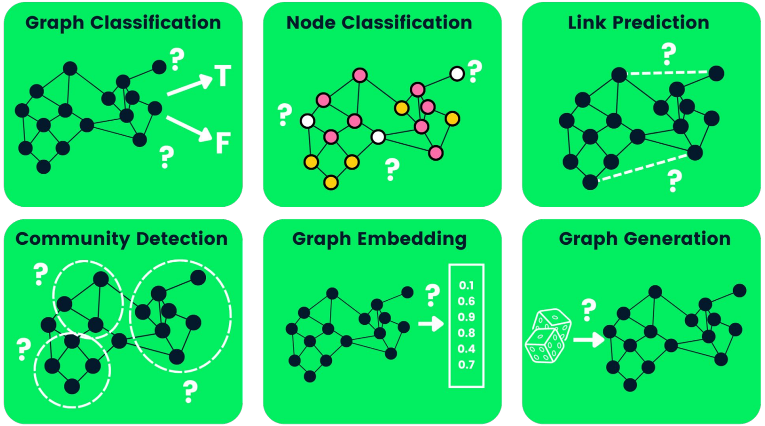
Risks / Uncertainties

- Deep learning has a curse
  - Data distribution shift
  - Hard to estimate possible variations on the product
- Unexpected component positions
  - Under the case or too deep and dark to be detected
- End effector is suitable for unmounting
  - Even screws may vary, metric or imperial

Possible Improvements

Graph Neural Networks

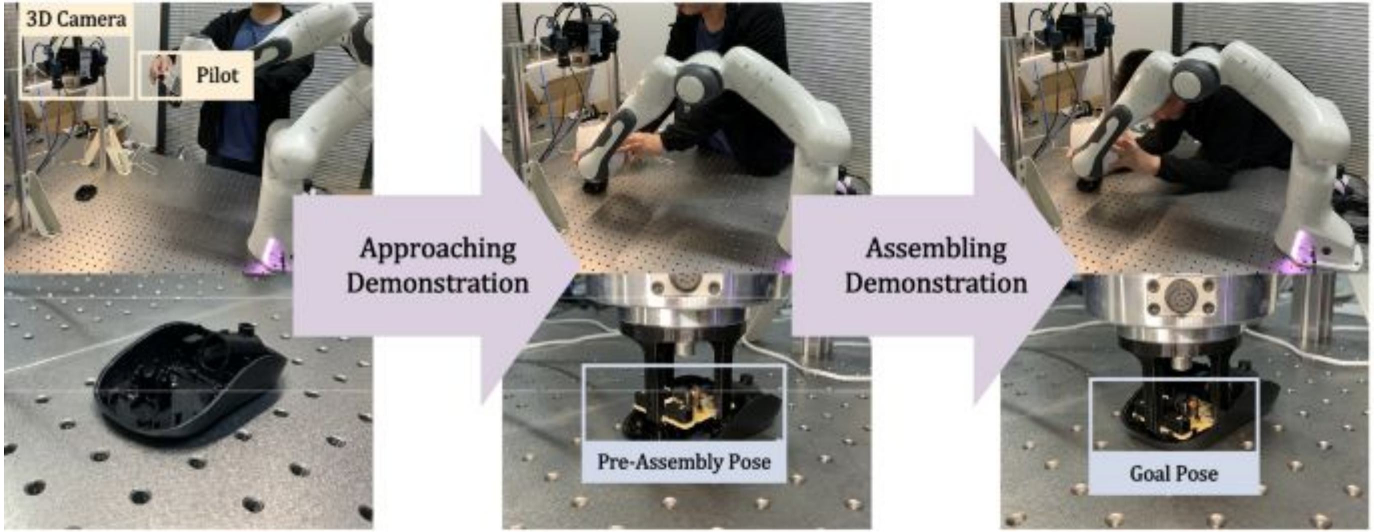
- Inter-relation of components can be represented as graphs
- Graph neural networks can provide a deep learning based task planning
- Requires conversion of object detection and pose estimation outputs to graphs



Possible Improvements

Imitation Learning / Learning from Demonstration

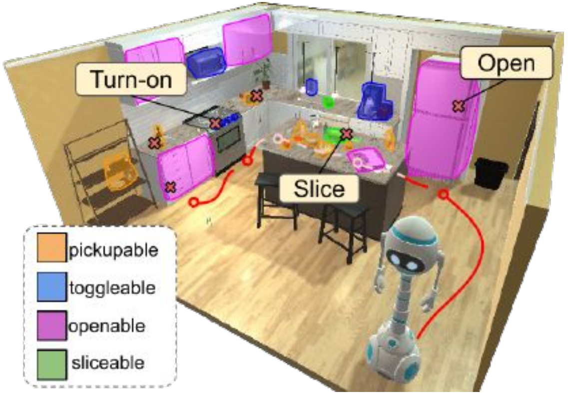
- An expert shows the solution to the robot
- Allows robot directly correlate image and action



Possible Improvements

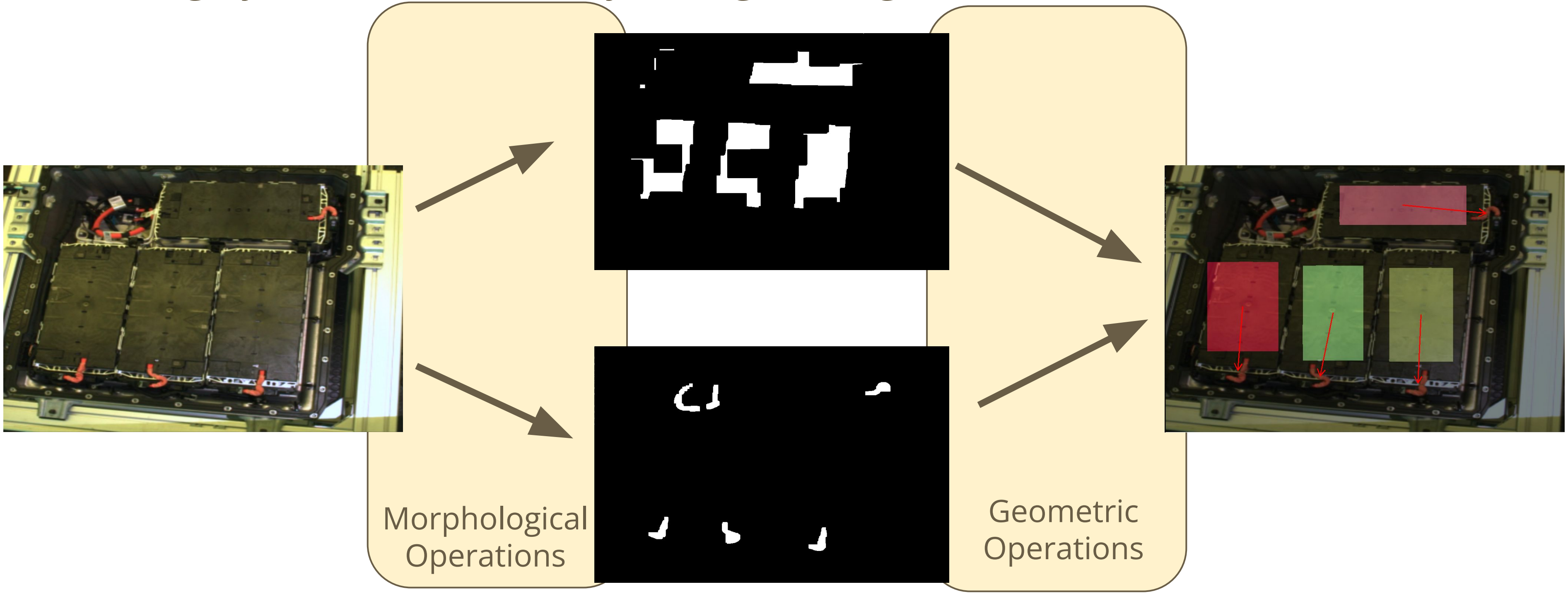
Object Affordances

- Affordance learning is an approach in robot learning
- Objects to be interacted with similar actions are learned
    - Screws, bolts, nuts, ... they all requires a similar action
- Provides a more compact and general solution

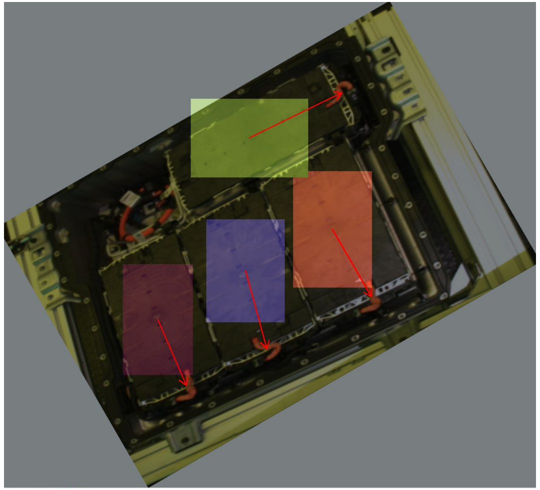


Example Implementation: Morphological Solution

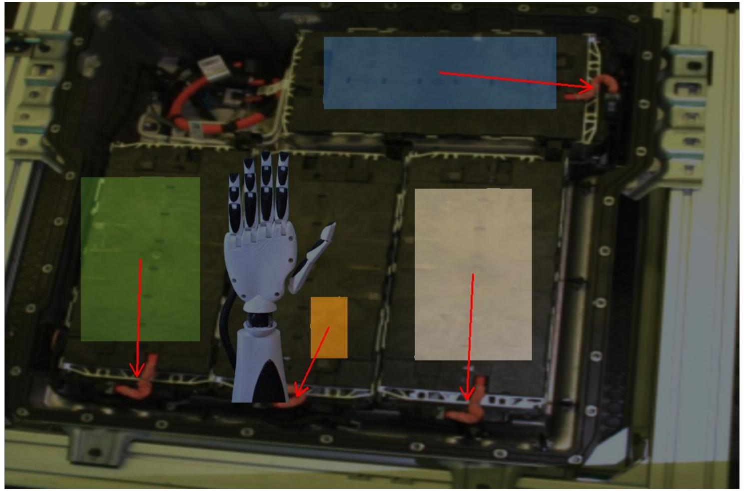
Masking by color and manually tuning the segments



Example Implementation: Morphological Solution



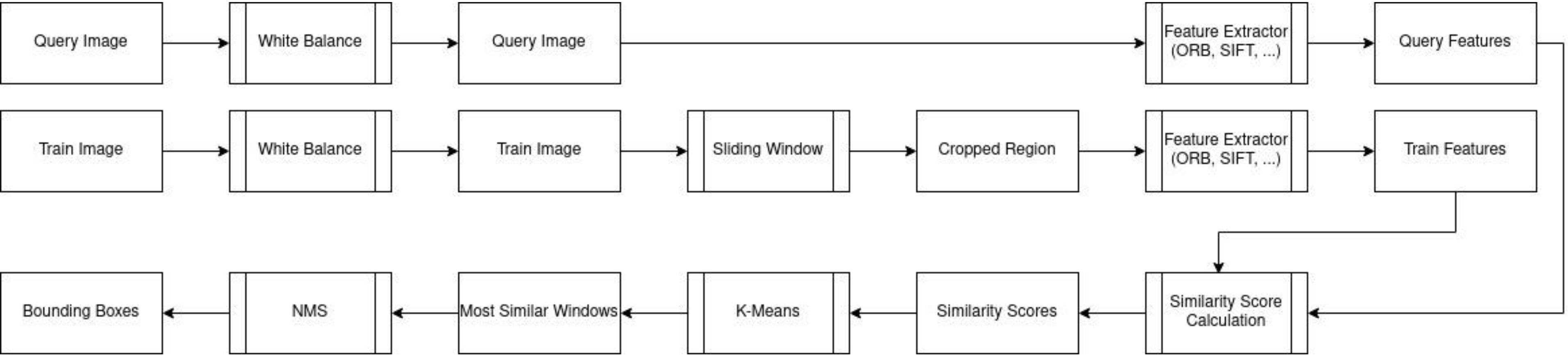
Rotation



Occlusion

Example Implementation: Feature Matching

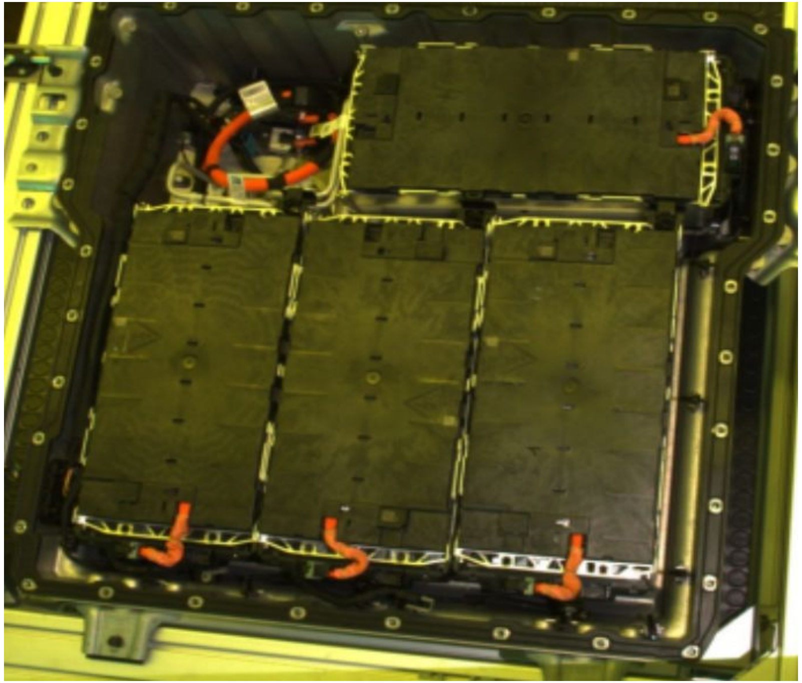
- Features extracted from query image is matched with the sliding window on train image
- The most similar candidates are selected by K-means and NMS fix repetitive detections



Object Detection by feature matching pipeline.

Example Implementation: Feature Matching

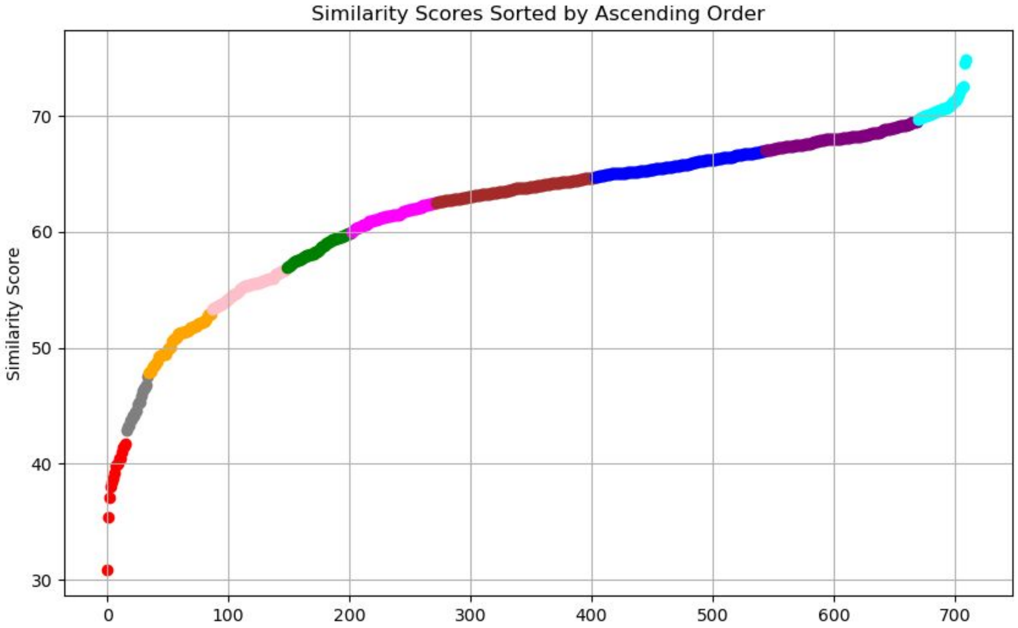
Target battery module.



Traversed battery pack.

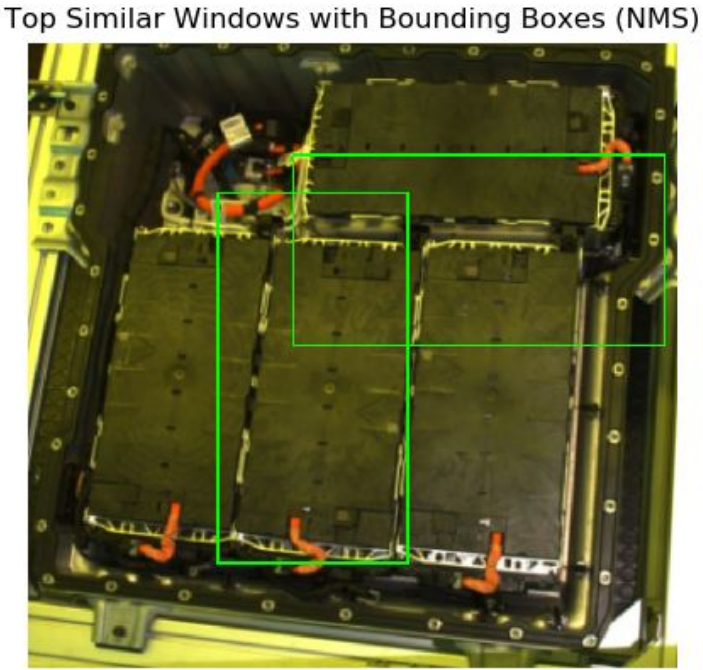
Example Implementation: Feature Matching

- K-means selects the most similar windows.



Example Implementation: Feature Matching

- Not very successful but the pipeline is obvious.



References

- Choux, Martin, Eduard Marti Bigorra, and Ilya Tyapin. "Task planner for robotic disassembly of electric vehicle battery pack." Metals 11.3 (2021): 387.
- Kay, Ian, et al. "Robotic disassembly of electric vehicles' battery modules for recycling." Energies 15.13 (2022): 4856.
- Rehnholm, Jonas. "Battery pack part detection and disassembly verification using computer vision." (2021).
- Tan, Wei Jie, et al. "A hybrid disassembly framework for disassembly of electric vehicle batteries." International Journal of Energy Research 45.5 (2021): 8073-8082.
- Zorn, Merle, et al. "An approach for automated disassembly of lithium-ion battery packs and high-quality recycling using computer vision, labeling, and material characterization." Recycling 7.4 (2022): 48.