CLIPort with Safety Constraints

Authors: Mohit Shridhar, Lucas Manuelli, Dieter Fox

Team 9
Deepak,Dheeraj, Manikanta, Raghavendra

Topics

- Introduction
- Architecture of CLIPort
- Baseline
- Tasks Generation and metrics
- Proposed solution
- Results
- Future Scope

Introduction

We planned to work on:

- End-to-end system
- Understand linguistic goals and perform real-world tasks.
- Model that understands spatial distribution
- Links to linguistic targets.
- Leverage CLEVR dataset
- Transporter Networks
- End effector poses.

CLIPort

We present CLIPort, a language conditioned imitation-learning agent that integrates the semantic understanding (what) of CLIP(by OpenAI) with the spatial precision (where) of Transporter(by Google).

CLIPort uses the following:-

- Word embeddings.
- Transporter networks.
- Spatial symmetry features.
- Attention map.
- Key and query for convolution.
- Identifying the end pose.
- Region of interest.

Architecture of CLIPort

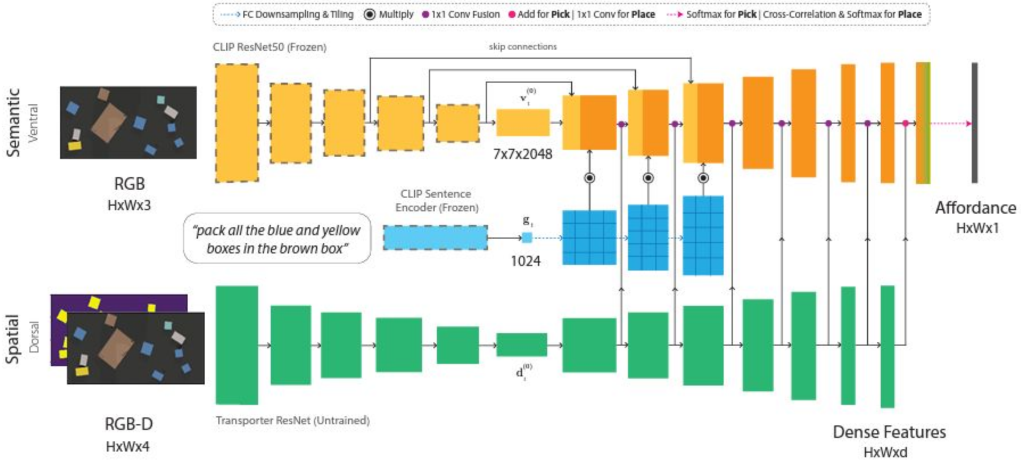
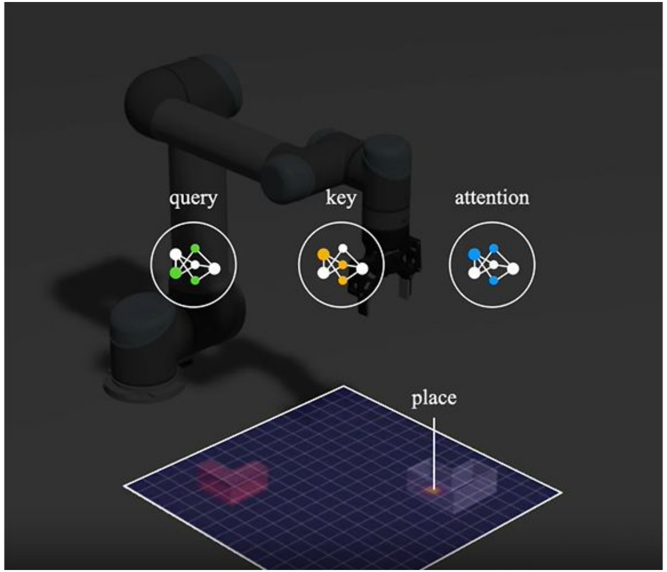

FC Downsampling & Tiling
Multiply
1x1 Conv Fusion
Add for **Pick** | 1x1 Conv for **Place**
Softmax for **Pick** | Cross-Correlation & Softmax for **Place**
CLIP ResNet50 (Frozen)
skip connections
Semantic
Ventral
RGB
HxWx3
$v_t^{(0)}$
7x7x2048
CLIP Sentence Encoder (Frozen)
"pack all the blue and yellow boxes in the brown box"
g_t
1024
Affordance
HxWx1
Spatial
Dorsal
RGB-D
HxWx4
$d_t^{(0)}$
Transporter ResNet (Untrained)
Dense Features
HxWxd

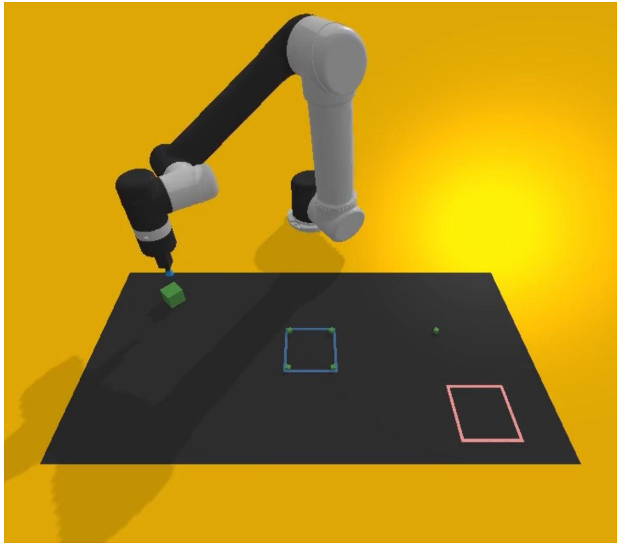
Figure 2. **CLIPort Two-Stream Architecture.** An overview of the **semantic** and **spatial** streams. The **semantic** stream uses a frozen CLIP ResNet50 [1] to encode RGB input, and its decoder layers are conditioned with tiled language features from the CLIP sentence encoder. The **spatial** stream encodes RGB-D input, and its decoder layers are laterally fused with the **semantic** stream. The final output is a map of dense pixelwise features that is used for pick or place affordance predictions. This same two-stream architecture is used in all 3 Fully-Convolutional-Networks f_{pick}, Φ_{query}, and Φ_{key} with f_{pick} is used to predict pick actions, and Φ_{query} and Φ_{key} are used to predict place actions. See Appendix C for the exact architecture.



Reference: CLIPORT- What and Where Pathways for Robotic Manipulation

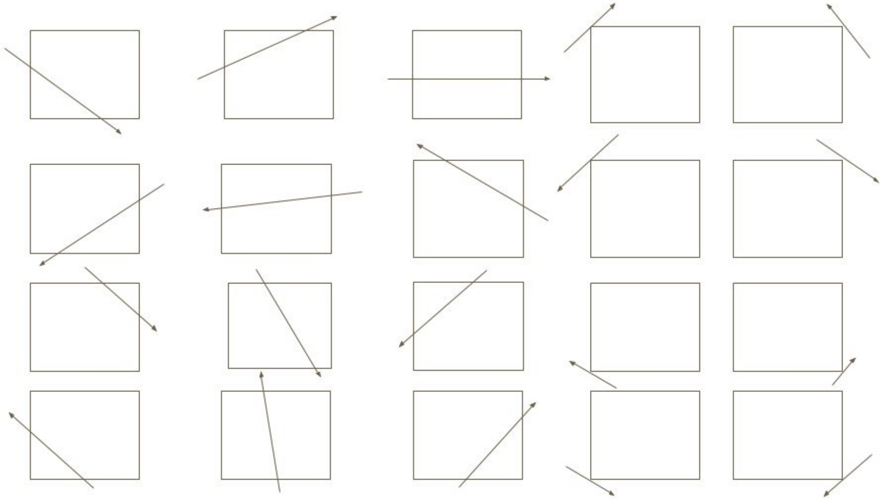
Task Generation and Metrics

- Created new task “ avoid-region”.
- Provided with language-goal “pick the green block and place in the pink square region by avoiding the blue region”.
- Robot failed to follow language-goal because Transporter Network provides only two point (Tpick,Tplace).
    - Tpick = argmax (u,v) Qpick((u, v)|yt)
    - Tplace = argmaxΔτ Qplace(Δτ |yt, Tpick)
        - Qplace(Δτ |yt, Tpick) = Φquery(yt[Tpick]) ∗ Φkey(yt) [Δτ]
- So to evaluate the task we created evaluation matrix
    - It will calculator number of robot trajectory point that are passing through the forbidden square region.



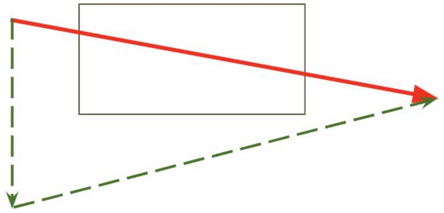
Proposed Solution

- Looks like a mathematical problem
- Distance between 2 points is a straight line
- Intersections points: 2 points between polygon and line
    - Eg: Square
- Possible Trajectories 25



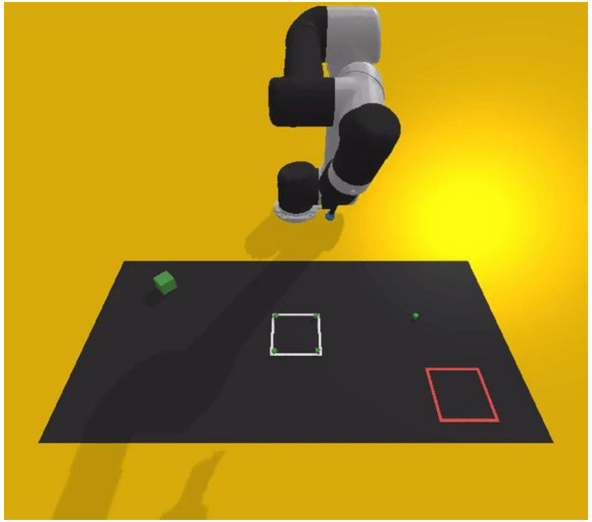
Proposed Solution

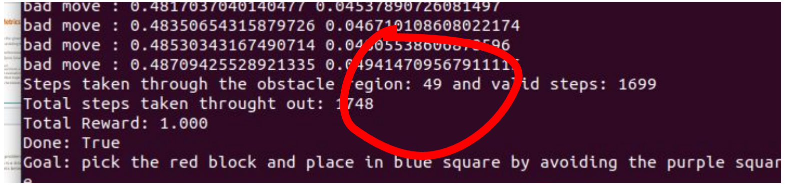
- Find the intersection points using clipping algorithm **Liang-Barsky Algorithm.**
- Get the points and check how far is the final point from the intersection point and add offset either in x or y axis to get intermediate point.



Results

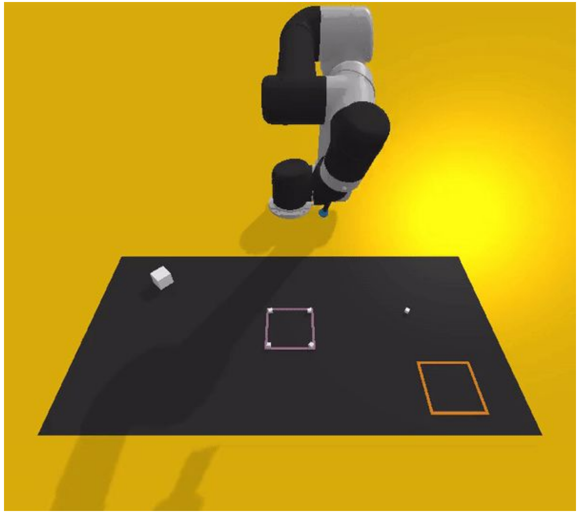
Without safety constraint

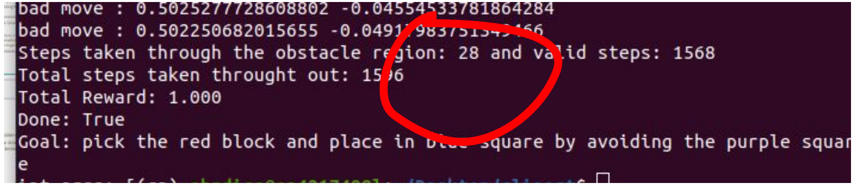




```
bad move : 0.4817037040140477 0.04537890726081497
bad move : 0.48350654315879726 0.046710108608022174
bad move : 0.48530343167490714 0.04[illegible]05538600[illegible]596
bad move : 0.48709425528921335 0.0494147095679111[illegible]
Steps taken through the obstacle region: 49 and valid steps: 1699
Total steps taken throught out: 1748
Total Reward: 1.000
Done: True
Goal: pick the red block and place in blue square by avoiding the purple squar
e
```

With safety constraint





```
bad move : 0.5025277728608802 -0.04554533781864284
bad move : 0.502250682015655 -0.0491[illegible]983751[illegible]466
Steps taken through the obstacle region: 28 and valid steps: 1568
Total steps taken throught out: 1596
Total Reward: 1.000
Done: True
Goal: pick the red block and place in blue square by avoiding the purple squar
e
```

Future Scope

- Generate intermediate point that is equidistant from goal and initial point and away from the region.

Future Scope

- Generate intermediate point that is equidistant from goal and initial point and away from the region.
- Add an additional pathway

Future Scope

- Generate intermediate point that is equidistant from goal and initial point and away from the region.
- Add an additional pathway
- Change the reward formulation to award reward at each instance of the episode as opposed to end of the episode

Future Scope

- Generate intermediate point that is equidistant from goal and initial point and away from the region.
- Add an additional pathway
- Change the reward formulation to award reward at each instance of the episode as opposed to end of the episode
- The proposed system consists of two language goals one for the task and one for thee safety constraint

Thank you!