説明用レジュメ

**【システム構成】**

・ロボット側 OS：sciurus17 の制御に使用

・PC 側 OS：画像認識モデルおよび VLM（視覚言語モデル）の実行環境

両システム間の通信には ROS1 を採用し、リアルタイムでの情報交換を実現している

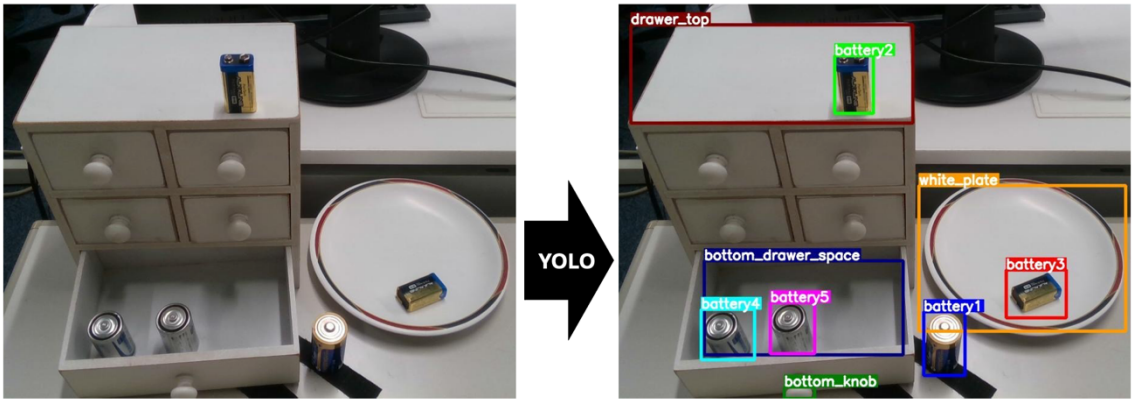
**【1. 画像認識プロセス】**

1.ロボット起動後、視角を調整し、撮影した画像を PC 側に送信する。

2.PC 側で YOLOv7 モデルを用いて画像認識を実行する。

事前に学習させた YOLOv7 モデルに入力する（YOLO モデルの学習および操作方法については、Github(https://github.com/laitathei/YOLOv7-Pytorch-Segmentation/tree/master）を参照。

3.認識結果としてラベルとボックスを含む画像を生成する。




4.検出された物体の 2 次元座標をロボット側に送信する。

5.ロボット側で 2 次元→3 次元座標変換およびカメラ座標→ロボット座標変換を実施し、各物体のロボット座標系における 3 次元位置を特定する。

**【2. 人間意図理解とロボット動作推論】**

1.VLM プログラムを起動し、ロボットの目の前の環境画像を記録する。

2.人間の動作後、環境の変化を記録する。（動作前後の 2 枚の画像を取得）

3.Claude モデルに人間の動作前後の画像を入力することで、人間の意図やロボットの

次にとるべき動作を推論する。(入力画像は元画像とラベル付き画像総計 4 枚)

**Claude API を利用するコード例：**

```python
import anthropic

client = anthropic.Anthropic(
    # defaults to os.environ.get("ANTHROPIC_API_KEY")
    api_key="my_api_key",
)

message = client.messages.create(
    model="claude-3-7-sonnet-20250219",
    max_tokens=20000,
    temperature=1,
    system="【System prompt】",
    messages=[
        {
            "role": "user",
            "content": [
                {
                    "type": "text",
                    "text": "【User input】"
                }
            ]
        }
    ]
)
print(message.content)
```

Claude モデルと対話するため、anthropic ライブラリをインポートする。

「client.messages.create」は、言語モデルと対話するためのメッセージを生成するメソッドである。 create メソッドに渡すパラメータには、「使用するモデルのバージョン」、「トークン数」、「システムプロンプト（VLM の生成ルールを規定する）」、「ユーザーの入力」などが含まれる。(ユーザー入力の部分には画像を入力することも可能)

「client.messages.create」を呼び出すと、人間の意図の推論やロボットが次に取るべき動作などを含むテキスト結果が返される。

**【3. ロボット動作シーケンスの生成と実機動作】**

本研究は、事前に一連のロボットの基本動作リストを準備しており、各動作のコマンドセットを定義する。これらの基本動作を組み合わせることで、複雑なタスクを達成することを目的としている。

1.ロボット動作シーケンスの生成用の VLM に、先ほど生成した「ロボットが次に取るべき動作」とロボット目の前の環境画像を入力する。(入力画像は元画像とラベル付き画像総計 2 枚)

2.VLM は適切な基本動作を選択・配列し、動作シーケンスを生成する。

3.シーケンスに従ってロボット実機動作を実行する。

**【4. 動作成否判断】**

1. シーケンスに沿って、ロボット動作を 1 step 実行した後、sciurus17 は PC に信号を送信し、動作の成否判断プロセスに入る。

2. 動作成否判断用 VLM に動作前後の画像と実行した動作のテキストを入力することで、動作が成功したかどうかを判断する。(入力画像は元画像とラベル付き画像総計 4 枚)

・成功の場合：PC は sciurus17 に信号を送信し、次のステップの動作を実行する。

・失敗の場合：新たな動作シーケンスを再生成する。

**【次の人間動作】**

シーケンスの動作が最後まで完了した後、再び VLM で人間の動作・意図を分析このサイクルを繰り返し、タスクを継続的に実行する。