

UTAF Physical AI カリキュラム



執筆 川嶋 凌児 (UTAF 代表)

〇〇 〇〇

△△ △△

監修 河原塚 健人

Contents

1. このカリキュラムについて	1
1.1 学習の流れ	1
2. SO-101 と LeRobot を体験しよう	2
2.1 SO-101 を組み立てよう	3
2.2 SO-101 をテレオペレーションしよう	5
2.3 SO-101 で模倣学習を体験しよう	6
3. 開発環境を整えよう	7
3.1 VS Code	8
3.2 Git と GitHub	9
3.3 Python Tutorial	10
4. ロボットを組み立てよう	11
4.1 1 個の Motor から関節角度を読み取ろう	12
4.2 1 個の Motor を動かそう	13
4.3 ゲインを変えて応答を比較しよう	14
4.4 2 個の Motor を一対一で同期しよう	15
4.5 Robot の Model をつくろう (URDF)	16
4.6 IK / FK を解いてみよう	17
4.7 Simulator を作ってみよう	18
4.8 Leader と Follower を自分でつないでみよう	19
4.9 Expert 最終課題：丸・三角・四角を描こう	20
5. 認識・ロボット学習・AI	21
5.1 カメラを取り付けて認識しよう	22
5.2 Data Collect して貯めてみよう	23
5.3 LSTM / ACT / Diffusion Policy を学習しよう	24
5.4 Deploy してみよう	25
5.5 VLA・ロボット基盤モデルを試そう	26
5.6 LLM とつないでみよう	27
6. QDD アクチュエータ	28
6.1 CAN 通信をセットアップしよう	29
6.2 QDD アクチュエータの状態を読み取ろう	30
6.3 トルクと PD ゲインを扱おう	31
6.4 2 個の QDD アクチュエータを同期しよう	32
7. 四脚ロボット	33
7.1 強化学習で Go2 の歩行方策を作って Deploy してみよう	34
7.2 四脚：強化学習で起き上がり方策を作ってみよう	35
7.3 四脚：LiDAR で SLAM してみよう	36
7.4 四脚：Nav2 で Navigation してみよう	37
8. 二脚ロボット	38
8.1 強化学習で K1 の歩行方策を作って Deploy してみよう	39
8.2 二脚：強化学習で起き上がり方策を作ってみよう	40
8.3 二脚：LiDAR で SLAM してみよう	41
8.4 二脚：Nav2 で Navigation してみよう	42

9. Mobile Manipulator	43
9.1 Mobile Manipulator : LiDAR で SLAM してみよう	44
9.2 Mobile Manipulator : Nav2 で Navigation してみよう	45
9.3 Mobile Manipulator を Teleoperation してみよう	46
9.4 台車も入れて模倣学習してみよう	47
10. Hardware	48
10.1 UMI を造ろう	49
10.2 2 指 Hand を自分で設計してみよう	50
10.3 新しい LiDAR を取り付けてみよう	51
11. 研究しよう	52
11.1 Master プロジェクトに取り組もう	52
11.2 論文を書いて国際会議へ投稿しよう	53
12. 付録	54
12.1 Linux	54
12.2 ROS 2	55
12.3 URDF	56
12.4 Fusion ビギナーレッスンと 3D プリンタ	57
12.5 MuJoCo チュートリアル	58
12.6 MJLab Tutorial	59
12.7 Robot Learning	60

1. このカリキュラムについて

1.1 学習の流れ

ハードとソフトの両輪から Physical AI を学び、約 1 年で第一線の研究に取り組めるように組み立てられたカリキュラムです。

1.1.1 称号と到達条件

このカリキュラムでは、学習と研究の進行を **Member**、**Expert**、**Master**、**GrandMaster** の 4 段階で表します。「SO-101 と LeRobot を体験しよう」は誰でも体験でき、この章を完了すると UTAF の正式なメンバーとなります。GrandMaster を目指して論文を書く段階になると、河原塚先生の指導を受けられます。

称号	到達条件
Member	SO-101 と LeRobot を体験しよう を完了する
Expert	ロボットを組み立てよう を完了する
Master	Master プロジェクトに取り組もう の完了レビューを通過する
GrandMaster	論文を書いて国際会議へ投稿しよう を完了する

1.1.2 進め方

「ロボットを組み立てよう」を終えた後は、Master プロジェクトのテーマに応じて必要な選択カリキュラムへ取り組みます。選択カリキュラムをすべて行う必要はありません。

1. [SO-101 と LeRobot を体験しよう](#)で、模倣学習までの一連の流れを体験する。
2. [開発環境を整えよう](#)で、自分で実装する準備をする。
3. [ロボットを組み立てよう](#)で、Motor からモデル、運動学、シミュレータ、ロボット全体へ進む。
4. チームを組み、「研究しよう」の [Master プロジェクト](#)で取り組むテーマを決める。
5. プロジェクトに応じて、[認識・ロボット学習・AI](#)、[QDD アクチュエータ](#)、[四脚](#)、[二脚](#)、[Mobile Manipulator](#)、[Hardware](#) から必要なカリキュラムに取り組む。
6. Master プロジェクトの完了レビューを通過し、成果を論文にまとめて[国際会議へ投稿](#)する。

1.1.3 演習の完了条件

- 演習ごとに、自分の GitHub リポジトリ内へコードや設計データを保存し、README.md に実行手順と結果をまとめる。
- 約 1 分の成果動画と、その README.md へのリンクを、UTAF Slack の # 進捗報告 チャンネルに提出する。

提出後、3 名のピアレビューが各演習ページの「チェックリスト」を使って確認します。チェックリストは受講者の自己確認とピアレビューで共通です。

2. SO-101 と LeRobot を体験しよう

SO-101 を使い、組み立てからテレオペレーション、データ収集、模倣学習までを一度通して体験します。この段階では用意された LeRobot のツールを使い、ロボット学習の雰囲気をつかみます。

1. [SO-101 を組み立てよう](#)
2. [SO-101 をテレオペレーションしよう](#)
3. [SO-101 で模倣学習を体験しよう](#)

3つの演習を完了したら Member です。

2.1 SO-101 を組み立てよう

2.1.1 演習内容

Windows の PowerShell を使い、モーター ID 設定、組み立て、キャリブレーションの順に進めます。以下のポート名とアーム名は例です。自分の環境で確認した値に置き換えてください。

1. 公式手順を確認する

[SO-101 の公式ドキュメント](#)を最後まで読み、Joint 1～6 の組み立て動画も確認します。モーター ID は、アームを組み立てる前に設定します。

2. 作業環境を有効化し、USB ポートを調べる

PowerShell を開くたびに、LeRobot をインストールした環境を有効化します。

```
conda activate lerobot
```

フォロワーとリーダーを 1 本ずつ接続し、それぞれの USB ポートを確認します。

```
lerobot-find-port
```

画面の指示に従って USB ケーブルを抜き、表示されたポート名を記録します。Windows では、COM3 や COM4 のような名前になります。

3. モーターの種類を確認する

フォロワーのモーターは 6 個ともギア比が 1/345 です。リーダーのモーターは、取り付ける関節によってギア比が異なります。モーターを混ぜる前に確認します。

リーダーの関節	ID	ギア比
Base / Shoulder Pan (土台の回転)	1	1/191
Shoulder Lift (肩の上下)	2	1/345
Elbow Flex (肘)	3	1/191
Wrist Flex (手首の曲げ)	4	1/147
Wrist Roll (手首の回転)	5	1/147
Gripper (グripper)	6	1/147

4. モーター ID を設定する

コントローラー基板には、指示されたモーターを 1 個だけ接続します。フォロワーとリーダーに対して、それぞれコマンドを実行します。

```
lerobot-setup-motors `
  --robot.type=so101_follower `
  --robot.port=COM4

lerobot-setup-motors `
  --teleop.type=so101_leader `
  --teleop.port=COM3
```

画面には、グripperから順番に接続するモーターが表示されます。'gripper' motor id set to 6 のように表示されたら、次のモーターへ進みます。ID 6 から 1 まで設定し、完了したモーターにはマスキングテープで ID 番号を貼ります。

5. アームを組み立てる

3D プリントパーツのサポート材をすべて除去し、公式動画に従って土台から Joint 1～6 の順に組み立てます。各モーターには、組み込む前に 3 ピンケーブルを 1 本挿

します。最後にモーターを数珠つなぎにし、ID 1 のモーターからコントローラー基板へ接続します。

3 ピンケーブルは、モーターにある 2 つのポートのどちらへ挿しても構いません。パーツが最後まではまらない場合は、凸側をやすりで少しずつ削り、削るたびにあまり具合を確認します。

6. キャリブレーションする

アームごとに名前を決めます。この名前でキャリブレーション結果が保存されるため、記録して以後も同じ名前を使います。

```
lerobot-calibrate `
  --robot.type=sol01_follower `
  --robot.port=COM4 `
  --robot.id=utaf_follower_01

lerobot-calibrate `
  --teleop.type=sol01_leader `
  --teleop.port=COM3 `
  --teleop.id=utaf_leader_01
```

全関節を可動域の中央にして Enter を押し、その後、各関節を端から端まで動かします。

エラーが出たとき

電源、PC と基板をつなぐ USB ケーブル、基板とモーターをつなぐ 3 ピンケーブルを確認します。Waveshare 基板を使う場合は、2 つのジャンパーが B チャンネル (USB) 側にあることも確認します。

conda を認識しない場合は、次のコマンドを一度実行して PowerShell を開き直します。

```
conda init powershell
```

日本語版 Windows で文字化けやクラッシュが起きる場合は、コマンドの前に次を実行します。

```
$env:PYTHONUTF8=1
```

2.1.2 提出物

約 1 分の成果動画と、README.md へのリンクを提出します。

2.1.3 チェックリスト

- ☐ 公式ドキュメントを最後まで読み、Joint 1~6 の動画を確認した
- ☐ フォロワーとリーダーの USB ポートを特定し、記録した
- ☐ すべてのモーターを 1 個ずつ基板につなぎ、ID を設定した
- ☐ ID 設定後のモーターに ID 番号を貼った
- ☐ リーダー用モーターのギア比と取り付け位置を確認した
- ☐ 3D プリントパーツのサポート材を除去した
- ☐ 各モーターに 3 ピンケーブルを挿してから、Joint 1~6 を組み立てた
- ☐ フォロワーとリーダーをキャリブレーションし、ポート名とアーム名を記録した

2.1.4 参考資料

[SO-101, LeRobot documentation](#)

2.2 SO-101 をテレオペレーションしよう

2.2.1 演習内容

組み立てとキャリブレーションを終えたリーダーとフォロワーを接続し、リーダーの動きにフォロワーを追従させます。

最初に、組み立て時に記録した次の 4 つの値を確認します。

- フォロワーの USB ポート
- フォロワーのアーム名
- リーダーの USB ポート
- リーダーのアーム名

PowerShell を開き、LeRobot をインストールした環境を有効化します。

```
conda activate lerobot
```

記録したポート名とアーム名を指定して、テレオペレーションを開始します。次の COM4、COM3、utaf_follower_01、utaf_leader_01 は例です。

```
lerobot-teleoperate \  
  --robot.type=sol01_follower \  
  --robot.port=COM4 \  
  --robot.id=utaf_follower_01 \  
  --teleop.type=sol01_leader \  
  --teleop.port=COM3 \  
  --teleop.id=utaf_leader_01
```

リーダーを手で動かし、フォロワーが同じように追従することを確認します。停止するときは Ctrl + C を押します。

エラーが出た場合は、電源、PC と基板をつなぐ USB ケーブル、基板とモーターをつなぐ 3 ピンケーブルを確認します。Waveshare 基板を使う場合は、2 つのジャンパーが B チャンネル (USB) 側にあることも確認します。

2.2.2 提出物

約 1 分の成果動画と、README.md へのリンクを提出します。

2.2.3 チェックリスト

- ☐ フォロワーとリーダーのポート名とアーム名を確認した
- ☐ リーダーを動かすと、フォロワーが同じように追従した
- ☐ Ctrl + C でテレオペレーションを停止できた

2.2.4 参考資料

[Getting Started with Real-World Robots, LeRobot documentation](#)

[SO-101, LeRobot documentation](#)

2.3 SO-101 で模倣学習を体験しよう

2.3.1 演習内容

準備中です。

2.3.2 提出物

約 1 分の成果動画と、README.md へのリンクを提出します。

2.3.3 チェックリスト

☐ 準備中です。

2.3.4 参考資料

[Getting Started with Real-World Robots, LeRobot documentation](#)

3. 開発環境を整えよう

SO-101 の体験後、自分でプログラムを書き、成果を共有するための開発環境を整えます。

すでに好みのエディタを使っている人や、Git と GitHub、Python に習熟している人は、該当する項目を飛ばして構いません。

1. [VS Code](#)
2. [Git と GitHub](#)
3. [Python Tutorial](#)

3.1 VS Code

3.1.1 チュートリアル

VS Code と Python 拡張をインストールします。

Getting Started with Python in VS Code を進め、練習用フォルダに Python ファイルを作成します。コードを実行し、ブレークポイントを設定して、デバッガで変数の値を確認します。

3.1.2 参考資料

[Getting Started with Python in VS Code](#)

3.2 Git と GitHub

3.2.1 チュートリアル

サル先生の Git 入門の「入門編」を進め、clone、status、add、commit、branch、pull、push を練習します。

GitHub の Hello World を進め、ブランチを作成し、Pull Request を開いてマージする流れを確認します。

GitHub へ SSH で接続し、リポジトリを clone、pull、push できることを確認します。

3.2.2 参考資料

[サル先生の Git 入門](#)

[Hello World, GitHub Docs](#)

[SSH を使用した GitHub への接続](#)

3.3 Python Tutorial

3.3.1 チュートリアル

Python 公式チュートリアル第 3 章から第 9 章までを進め、数値と文字列、リストと辞書、条件分岐、繰り返し、関数、モジュール、ファイル入出力、例外処理、クラスを学びます。

掲載されているサンプルコードを実際に実行します。

3.3.2 参考資料

[Python 公式チュートリアル（日本語）](#)

4. ロボットを組み立てよう

SO-101 を題材に、用意されたツールの中身を小さなプログラムとして自分で実装します。Motor を読み書きし、モデル、運動学、シミュレータ、ロボット全体へ段階的に進みます。

1. 1 個の Motor から関節角度を読み取ろう
2. 1 個の Motor を動かそう
3. ゲインを変えて応答を比較しよう
4. 2 個の Motor を一対一で同期しよう
5. Robot の Model をつくろう (URDF)
6. IK / FK を解いてみよう
7. Simulator を作ってみよう
8. Leader と Follower を自分でつないでみよう
9. Expert 最終課題：丸・三角・四角を描こう

すべての演習を完了し、最終課題のレビューを通過したら Expert です。

4.1 1 個の Motor から関節角度を読み取ろう

4.1.1 演習内容

準備中です。

4.1.2 提出物

約 1 分の成果動画と、README.md へのリンクを提出します。

4.1.3 チェックリスト

☐ 準備中です。

4.1.4 参考資料

[Linux](#)

4.2 1 個の Motor を動かそう

4.2.1 演習内容

準備中です。

4.2.2 提出物

約 1 分の成果動画と、README.md へのリンクを提出します。

4.2.3 チェックリスト

☐ 準備中です。

4.3 ゲインを変えて応答を比較しよう

4.3.1 演習内容

準備中です。

4.3.2 提出物

約 1 分の成果動画と、README.md へのリンクを提出します。

4.3.3 チェックリスト

☐ 準備中です。

4.4 2 個の Motor を一対一で同期しよう

4.4.1 演習内容

準備中です。

4.4.2 提出物

約 1 分の成果動画と、README.md へのリンクを提出します。

4.4.3 チェックリスト

☐ 準備中です。

4.5 Robot の Model をつくろう (URDF)

4.5.1 演習内容

準備中です。

4.5.2 提出物

約 1 分の成果動画と、README.md へのリンクを提出します。

4.5.3 チェックリスト

☐ 準備中です。

4.5.4 参考資料

[URDF](#)

4.6 IK / FK を解いてみよう

4.6.1 演習内容

準備中です。

4.6.2 提出物

約 1 分の成果動画と、README.md へのリンクを提出します。

4.6.3 チェックリスト

☐ 準備中です。

4.7 Simulator を作ってみよう

4.7.1 演習内容

準備中です。

4.7.2 提出物

約 1 分の成果動画と、README.md へのリンクを提出します。

4.7.3 チェックリスト

☐ 準備中です。

4.7.4 参考資料

[MuJoCo チュートリアル](#)

4.8 Leader と Follower を自分でつないでみよう

4.8.1 演習内容

準備中です。

4.8.2 提出物

約 1 分の成果動画と、README.md へのリンクを提出します。

4.8.3 チェックリスト

☐ 準備中です。

4.9 Expert 最終課題：丸・三角・四角を描こう

4.9.1 演習内容

準備中です。

4.9.2 提出物

約 1 分の成果動画と、README.md へのリンクを提出します。

4.9.3 チェックリスト

☐ 準備中です。

5. 認識・ロボット学習・AI

ロボットが環境を観測し、経験から行動を獲得して、人からの指示に応じるまでの流れを学ぶ、選択プログラムです。

1. カメラを取り付けて認識しよう
2. Data Collect して貯めてみよう
3. LSTM / ACT / Diffusion Policy を学習しよう
4. Deploy してみよう
5. VLA・ロボット基盤モデルを試そう
6. LLM とつないでみよう

5.1 カメラを取り付けて認識しよう

5.1.1 演習内容

準備中です。

5.1.2 提出物

約 1 分の成果動画と、README.md へのリンクを提出します。

5.1.3 チェックリスト

☐ 準備中です。

5.2 Data Collect して貯めてみよう

5.2.1 演習内容

準備中です。

5.2.2 提出物

約 1 分の成果動画と、README.md へのリンクを提出します。

5.2.3 チェックリスト

☐ 準備中です。

5.3 LSTM / ACT / Diffusion Policy を学習しよう

5.3.1 演習内容

準備中です。

5.3.2 提出物

約 1 分の成果動画と、README.md へのリンクを提出します。

5.3.3 チェックリスト

☐ 準備中です。

5.3.4 参考資料

[Robot Learning](#)

5.4 Deploy してみよう

5.4.1 演習内容

準備中です。

5.4.2 提出物

約 1 分の成果動画と、README.md へのリンクを提出します。

5.4.3 チェックリスト

☐ 準備中です。

5.4.4 参考資料

[Linux](#)

5.5 VLA・ロボット基盤モデルを試そう

5.5.1 演習内容

準備中です。

5.5.2 提出物

約 1 分の成果動画と、README.md へのリンクを提出します。

5.5.3 チェックリスト

☐ 準備中です。

5.5.4 参考資料

[Robot Learning](#)

[OpenVLA](#)

5.6 LLM とつないでみよう

5.6.1 演習内容

準備中です。

5.6.2 提出物

約 1 分の成果動画と、README.md へのリンクを提出します。

5.6.3 チェックリスト

☐ 準備中です。

5.6.4 参考資料

[Robot Learning](#)

6. QDD アクチュエータ

Mevius などに使われている高電圧・高トルクの QDD アクチュエータと CAN 通信を扱う選択プログラムです。

1. CAN 通信をセットアップしよう
2. QDD アクチュエータの状態を読み取ろう
3. トルクと PD ゲインを扱おう
4. 2 個の QDD アクチュエータを同期しよう

6.1 CAN 通信をセットアップしよう

6.1.1 演習内容

準備中です。

6.1.2 提出物

約 1 分の成果動画と、README.md へのリンクを提出します。

6.1.3 チェックリスト

☐ 準備中です。

6.2 QDD アクチュエータの状態を読み取ろう

6.2.1 演習内容

準備中です。

6.2.2 提出物

約 1 分の成果動画と、README.md へのリンクを提出します。

6.2.3 チェックリスト

☐ 準備中です。

6.3 トルクと PD ゲインを扱おう

6.3.1 演習内容

準備中です。

6.3.2 提出物

約 1 分の成果動画と、README.md へのリンクを提出します。

6.3.3 チェックリスト

☐ 準備中です。

6.4 2 個の QDD アクチュエータを同期しよう

6.4.1 演習内容

準備中です。

6.4.2 提出物

約 1 分の成果動画と、README.md へのリンクを提出します。

6.4.3 チェックリスト

☐ 準備中です。

7. 四脚ロボット

四脚ロボットを扱う選択プログラムです。

1. 強化学習で Go2 の歩行方策を作って Deploy してみよう
2. 強化学習で起き上がり方策を作ってみよう
3. LiDAR で SLAM してみよう
4. Nav2 で Navigation してみよう

7.1 強化学習で Go2 の歩行方策を作って Deploy してみよう

7.1.1 演習内容

準備中です。

7.1.2 提出物

約 1 分の成果動画と、README.md へのリンクを提出します。

7.1.3 チェックリスト

☐ 準備中です。

7.1.4 参考資料

[MJLab Tutorial](#)

[Robot Learning](#)

7.2 四脚：強化学習で起き上がり方策を作ってみよう

7.2.1 演習内容

準備中です。

7.2.2 提出物

約 1 分の成果動画と、README.md へのリンクを提出します。

7.2.3 チェックリスト

☐ 準備中です。

7.2.4 参考資料

[MJLab Tutorial](#)

[Robot Learning](#)

7.3 四脚：LiDAR で SLAM してみよう

7.3.1 演習内容

準備中です。

7.3.2 提出物

約 1 分の成果動画と、README.md へのリンクを提出します。

7.3.3 チェックリスト

☐ 準備中です。

7.3.4 参考資料

[ROS 2](#)

7.4 四脚：Nav2 で Navigation してみよう

7.4.1 演習内容

準備中です。

7.4.2 提出物

約 1 分の成果動画と、README.md へのリンクを提出します。

7.4.3 チェックリスト

☐ 準備中です。

7.4.4 参考資料

[ROS 2](#)

8. 二脚ロボット

二脚ロボットを扱う選択プログラムです。

1. 強化学習で K1 の歩行方策を作って Deploy してみよう
2. 強化学習で起き上がり方策を作ってみよう
3. LiDAR で SLAM してみよう
4. Nav2 で Navigation してみよう

8.1 強化学習で K1 の歩行方策を作って Deploy してみよう

8.1.1 演習内容

準備中です。

8.1.2 提出物

約 1 分の成果動画と、README.md へのリンクを提出します。

8.1.3 チェックリスト

☐ 準備中です。

8.1.4 参考資料

[MJLab Tutorial](#)

[Robot Learning](#)

8.2 二脚：強化学習で起き上がり方策を作ってみよう

8.2.1 演習内容

準備中です。

8.2.2 提出物

約 1 分の成果動画と、README.md へのリンクを提出します。

8.2.3 チェックリスト

☐ 準備中です。

8.2.4 参考資料

[MJLab Tutorial](#)

[Robot Learning](#)

8.3 二脚：LiDAR で SLAM してみよう

8.3.1 演習内容

準備中です。

8.3.2 提出物

約 1 分の成果動画と、README.md へのリンクを提出します。

8.3.3 チェックリスト

☐ 準備中です。

8.3.4 参考資料

[ROS 2](#)

8.4 二脚：Nav2 で Navigation してみよう

8.4.1 演習内容

準備中です。

8.4.2 提出物

約 1 分の成果動画と、README.md へのリンクを提出します。

8.4.3 チェックリスト

☐ 準備中です。

8.4.4 参考資料

[ROS 2](#)

9. Mobile Manipulator

Manipulator と台車を組み合わせたモバイル・マニピュレータを扱う選択プログラムです。

1. [LiDAR で SLAM してみよう](#)
2. [Nav2 で Navigation してみよう](#)
3. [Teleoperation してみよう](#)
4. [台車も入れて模倣学習してみよう](#)

9.1 Mobile Manipulator : LiDAR で SLAM してみよう

9.1.1 演習内容

準備中です。

9.1.2 提出物

約 1 分の成果動画と、README.md へのリンクを提出します。

9.1.3 チェックリスト

☐ 準備中です。

9.1.4 参考資料

[ROS 2](#)

9.2 Mobile Manipulator : Nav2 で Navigation してみよう

9.2.1 演習内容

準備中です。

9.2.2 提出物

約 1 分の成果動画と、README.md へのリンクを提出します。

9.2.3 チェックリスト

☐ 準備中です。

9.2.4 参考資料

[ROS 2](#)

9.3 Mobile Manipulator を Teleoperation してみよう

9.3.1 演習内容

準備中です。

9.3.2 提出物

約 1 分の成果動画と、README.md へのリンクを提出します。

9.3.3 チェックリスト

☐ 準備中です。

9.4 台車も入れて模倣学習してみよう

9.4.1 演習内容

準備中です。

9.4.2 提出物

約 1 分の成果動画と、README.md へのリンクを提出します。

9.4.3 チェックリスト

☐ 準備中です。

9.4.4 参考資料

[Robot Learning](#)

10. Hardware

ハードウェアを作る選択プログラムです。

1. [UMI を造ろう](#)
2. [2 指 Hand を自分で設計してみよう](#)
3. [新しい LiDAR を取り付けてみよう](#)

10.1 UMI を造ろう

10.1.1 演習内容

準備中です。

10.1.2 提出物

約 1 分の成果動画と、README.md へのリンクを提出します。

10.1.3 チェックリスト

☐ 準備中です。

10.1.4 参考資料

[Fusion ビギナーレッスンと 3D プリンタ](#)

10.2 2 指 Hand を自分で設計してみよう

10.2.1 演習内容

準備中です。

10.2.2 提出物

約 1 分の成果動画と、README.md へのリンクを提出します。

10.2.3 チェックリスト

☐ 準備中です。

10.2.4 参考資料

[Fusion ビギナーレッスンと 3D プリンタ](#)

10.3 新しい LiDAR を取り付けてみよう

10.3.1 演習内容

準備中です。

10.3.2 提出物

約 1 分の成果動画と、README.md へのリンクを提出します。

10.3.3 チェックリスト

☐ 準備中です。

10.3.4 参考資料

[Fusion ビギナーレッスンと 3D プリンタ](#)

[Linux](#)

11. 研究しよう

11.1 Master プロジェクトに取り組もう

11.1.1 演習内容

2～3 人のチームを組んで、簡単には解き切れない課題を一つ定め、3 か月（暫定）を目安にどこまで到達できるかを検証します。

11.1.1.1 課題の例

- Google Map 上で指定した地点までヒューマノイドで移動する
- 双腕ロボットでカレーを作る
- 提示された絵をロボットで再現する
- 筋骨格全身ヒューマノイドを作る
- 上記以外に取り組みたい課題がある場合は相談してください

11.1.1.2 提出物（検討中）

今後 UTAF 内で検討して決定します。

11.1.1.3 チェックリスト

☐ チェック項目は今後 UTAF 内で検討して決定します。

完了レビューの基準は、今後 UTAF 内で検討して決定します。

11.1.1.4 参考資料

[Robot Learning](#)

11.2 論文を書いて国際会議へ投稿しよう

11.2.1 演習内容

Master プロジェクトの成果を、投稿先の執筆要領・投稿規定を満たす論文としてまとめ、国際会議へ投稿します。UTAF Master が論文執筆へ進むときは、河原塚先生の指導を受けられます。

採択結果ではなく、論文を書き終えて国際会議へ投稿したことを GrandMaster の到達条件とします。

11.2.2 チェックリスト

☐ 投稿先の執筆要領・投稿規定を満たし、論文の投稿を完了している

12. 付録

12.1 Linux

12.1.1 最低限

[第 1 章「GNU/Linux チュートリアル」](#) を一通り学び、安全に試せる操作を実行します。

加えて、[6.3「リモートアクセスサーバーとユーティリティー \(SSH\)」](#) の 6.3.1～6.3.3 を学び、SSH でリモートホストへ接続できるようにします。

12.1.2 暇な人は全部やろう

[Debian リファレンス](#) 全体に目を通します。

12.2 ROS 2

ROS 2 は、センサー、認識、制御、ナビゲーションなどのプログラムを Nodeとして接続するための仕組みです。四脚・二脚・Mobile Manipulatorで SLAM や Nav2に取り組む前に学びます。

このカリキュラムでは、Ubuntu 24.04 と ROS 2 Jazzy Jalisco を基準にします。

12.2.1 最低限

[ROS 2 Jazzy の公式チュートリアル](#)を前から順に進め、次の内容を確認します。

1. ROS 2 環境の設定
2. turtlesim、ros2、rqt
3. Node、Topic、Service、Parameter、Action
4. rqt_console
5. Launch
6. rosbag2 によるデータの記録と再生

12.2.2 発展

ROS 2 で自分のプログラムを接続するときは、公式チュートリアルから次の項目へ進みます。

- Workspace、Package、colcon
- Python による Publisher と Subscriber
- tf2
- [URDF](#) と RViz

12.2.3 参考資料

[ROS 2 Jazzy を Ubuntu へインストールする](#)

[Beginner: CLI tools](#)

[Beginner: Client libraries](#)

[Nav2 Getting Started](#)

12.3 URDF

URDF (Unified Robot Description Format) は、ロボットを構成する Link と Joint、その形状、座標系、質量特性などを XML で記述する形式です。[Robot の Model をつくろう \(URDF\)](#) に取り組むときに参照します。

12.3.1 最低限

ROS 2 の公式 URDF チュートリアルを使い、次の内容を確認します。

1. link の形状と座標系を記述する
2. joint の親子関係、種類、原点、回転軸、可動範囲を記述する
3. visual、collision、inertial の役割を区別する
4. Link と Joint が一つの木構造になっていることを確認する
5. 関節を動かし、モデルの向きと可動範囲が実機に対応することを確認する

最初は box、cylinder、sphere などの基本形状で作ります。外形メッシュが必要になった場合に、Fusion などの CAD を使います。

12.3.2 発展

モデルの記述が長くなったら、Xacro で寸法、左右対称な部品、繰り返し構造を整理します。ROS 2 と RViz による表示や Joint State の確認は、[ROS 2](#) を参照します。

12.3.3 参考資料

[ROS 2 Jazzy : URDF](#)

[ROS 2 Jazzy : Building a movable robot model](#)

[ROS 2 Jazzy : Using URDF with robot_state_publisher](#)

[Fusion ビギナーレッスンと 3D プリンタ](#)

12.4 Fusion ビギナーレッスンと 3D プリンタ

12.4.1 チュートリアル

Autodesk Fusion ビギナー向けレッスンは、次の範囲で進めます。

- レッスン 1～3：基本（画面操作とモデリング）
- レッスン 4：必要に応じて（アセンブリ）
- レッスン 5：任意（図面）
- レッスン 6～10：CNC 加工を行うときに参照
- レッスン 11：強度解析が必要なときに参照

レッスン 1～3 を終えたら、作成したモデルを STL または 3MF で書き出し、スライサ設定と 3D プリンタでの試し刷りを行います。

12.4.2 参考資料

[Autodesk Fusion ビギナー向けレッスン](#)

12.5 MuJoCo チュートリアル

12.5.1 チュートリアル

物理演算シミュレーションエンジン、MuJoCo 公式の入門チュートリアルを Google Colab で実行します。重力、摩擦、時間刻みを変えたときのシミュレーション結果を確認します。

必要に応じて Getting Started に掲載されているほかのノートブックで学びます。

12.5.2 参考資料

[MuJoCo Introductory Tutorial \(Google Colab\)](#)

[MuJoCo Getting Started](#)

12.6 MJLab Tutorial

MuJoCo チュートリアルの後に取り組めます。

12.6.1 チュートリアル

MJLab の公式デモを実行し、環境、ロボット、タスクの構成を確認します。

12.6.2 参考資料

[mjlab](#)

12.7 Robot Learning

ETH Zürich の講義「Robot Learning: From Fundamentals to Foundation Models」は、模倣学習、強化学習、方策最適化から、VLA・ロボット基盤モデルまでを扱う講義です。講義スライド、録画、関連論文、演習コードが公開されています。

12.7.1 進め方

[SO-101 で模倣学習を体験しよう](#)では、講義に先立ってロボット学習の流れを一度体験します。Expert 取得後、ロボット学習を使う演習や [Master プロジェクト](#)に進むときに、講義を前から順に必要な範囲まで学びます。

1. Introduction to Robot Learning、PyTorch & NumPy Tutorial
2. Robot Control & MDPs
3. Imitation Learning
4. Reinforcement Learning I・II
5. Generative Models
6. Sequence Modeling and Transformers
7. World Models
8. Generalist Robot Policies
9. Embodied Reasoning and Test-time Scaling
10. Frontier & Open Problem

12.7.2 演習との対応

- Imitation Learning：[台車も入れて模倣学習してみよう](#)
- Robot Control & MDPs、Reinforcement Learning I・II：[強化学習で Go2 の歩行方策を作って Deploy してみよう](#)、[四脚：強化学習で起き上がり方策を作ってみよう](#)、[強化学習で K1 の歩行方策を作って Deploy してみよう](#)、[二脚：強化学習で起き上がり方策を作ってみよう](#)
- Imitation Learning、Generative Models、Sequence Modeling and Transformers：[LSTM / ACT / Diffusion Policy を学習しよう](#)
- Generalist Robot Policies：[VLA・ロボット基盤モデルを試そう](#)
- Embodied Reasoning and Test-time Scaling：[LLM とつないでみよう](#)
- その他の講義と関連論文：[Master プロジェクトに取り組もう](#)

12.7.3 参考資料

[Robot Learning, ETH Zürich](#)

[Course GitHub](#)