

---

# Mech-Mind Application

Mech-Mind

2023年08月02日

## 目次:

<b>1</b>	<b>ビジョンシステムの構築</b>	<b>2</b>
1.1	ハードウェアの取り付けと接続	2
1.2	ソフトウェアのインストール	14
1.3	産業用 PC の LAN ケーブルの接続と IP アドレス設定	15
1.4	画像品質の確認	20
<b>2</b>	<b>応用事例</b>	<b>21</b>
2.1	整列して並べられたクランクシャフト供給	21
<b>3</b>	<b>アプリケーションの最適化</b>	<b>52</b>
3.1	カメラ選択	52
3.2	照明制御	53
3.3	治具の設計	54
3.4	電源制御	54
3.5	ロボットの選択	55
3.6	誤差制御	55
3.7	タクトと経路の最適化	60
<b>4</b>	<b>注意事項</b>	<b>72</b>
4.1	注意事項	72
4.2	他の注意事項	74

Mech-Mind ビジョンシステムは、Mech-Eye 産業用 3D カメラ、Mech-Mind ソフトウェアシステム、ロボットおよび外部設備で構成されています。本章では、Mech-Mind ビジョンシステムの構築方法や推奨事項について説明します。

## ビジョンシステムの構築

Mech-Mind ビジョンシステムは、Mech-Mind が提供する完全な 3D ビジョンソリューションです。

本節では、Mech-Mind ビジョンシステムの構成と構築方法について説明します。

Mech-Mind ビジョンシステムを構築するには、次のことを完了させる必要があります。

- ハードウェアの取り付けと接続
- ソフトウェアのインストール
- IP アドレスの設定
- 画像品質の確認

### 1.1 ハードウェアの取り付けと接続

Mech-Mind ビジョンシステムのハードウェアは、主にカメラ、産業用 PC、ロボットとそのコントローラ、周辺機器（PLC、上位システムなど）で構成されています。

ハードウェアの取り付けと接続を行うには、以下のことを完了させてください。

1. 梱包内容の確認
2. 取り付け前の確認
3. ハードウェアの取り付け
4. ハードウェアの接続
5. 取り付け後の確認

#### 1.1.1 梱包内容の確認

1. お受け取りの際は、梱包に問題がないことを確認してください。
2. 梱包中にある同梱包一覧を確認し、商品や付属品に欠品や損傷がないことを確認してください。



同梱品一覧には以下のことが含まれています。

番号	製品	機能
1	キャリブレーションボード	カメラのキャリブレーションに使用されます
2	カメラの付属品ボックス	カメラの取り付けに使用されます
3	産業用 PC	Mech-Mind ソフトウェアシステムが組み立てられます
4	dongle	ソフトウェアのライセンス供与に使用されます
5	カメラ使用説明書	Mech-Eye 産業用 3D カメラの使用説明書です
6	Mech-Eye 産業用 3D カメラ	画像の取得に使用されます
7	産業用 PC の電源ケーブル	産業用 PC の電源を入れます
8	産業用 PC アダプター	
9	フランジ	キャリブレーションボードを接続するために使用されます
10	DC 電源コード (カメラ)	カメラをレール電源に接続します
11	カメラの LAN ケーブル	カメラを産業用コンピュータに接続します
12	レール電源	Mech-Eye 産業用 3D カメラの電源を入れます
13	同梱品一覧	梱包に含まれるすべての商品と付属品を一覧表示します

ヒント: 上記の同梱品一覧は参照のみを目的としています。実際の商品とは異なる場合があります。万一、紛失や破損が発生してしまった場合は、Mech-Mind にお問い合わせください。

## 1.1.2 取り付け前の確認

取り付け前に以下のカメラ、産業用 PC、ロボット関連の確認が必要です。

### カメラを確認

カメラを取り付ける前に、カメラの取り付け方を決定する必要があります。取り付け方式によって、確認しておく必要のあることが異なります。

### カメラの取り付け方を決定

カメラの取り付け方式（Eye in Hand または Eye to Hand）は、ロボットに対するカメラの相対位置に応じて決定してください。

	Eye in Hand (EIH)	Eye to Hand (ETH)
説明	カメラはロボットエンドに取り付けられ、ロボットと一緒に移動します。	カメラはスタンドに固定されています。
図示		

### カメラスタンドをチェック

Eye to Hand 方式でカメラが取り付けられている場合、次のことを確認する必要があります。

- カメラの取り付け高さが適切であること：Mech-Eye Viewer の [視野計算機](#) を使用してカメラの取り付け高さを計算します。
- カメラスタンドの材質が適切であること：機械加工されたスチールを推奨します。アルミの強度が弱く、長期間の使用によりスタンドが変化する可能性があるため、推奨しません。
- カメラスタンドの構造が適切であること：カメラスタンドはしっかりと取り付けられ、支持力は安定しています。より強力な支持力のために、ガントリーを使用することをお勧めします。
  - ロボットが移動するなどの振動が発生しても、カメラの取り付け位置は変わりません。
  - カメラスタンドにスライド機構がある場合、カメラが移動しても揺れないことや、位置決め精度が間違っていないことを確認します。

## ケーブルを確認

カメラがロボットまたはほかのモバイル型の機器に取り付けられている場合は、カメラの配線および次のことを確認する必要があります。

- DC 電源ケーブルと LAN ケーブルをカメラにしっかりと接続します。
- カメラを移動する際に、ケーブルやプラグが引っ張られたり、破損したりすることはありません。

## 産業用 PC を確認

産業用 PC を取り付ける前に、取り付け環境が以下の要件を満たしていることを確認します。

- 産業用 PC は、ほこり、鉄粉や水しぶきの少ない場所に設置します。
- 動作中の周囲温度は 60°C を超えないことを確認します。
- 産業用 PC の放熱構造を覆わないことを確認します。

## ロボットを確認

ロボットを確認する際には、ロボットの設置環境、架台、ゼロ位置、可動範囲を確認する必要があります。

- ロボットの設置環境を確認：
  - ロボットに付随する電源ケーブルの断線はありません。
  - ロボットコントローラの内部は乾燥しています。
- ロボット架台を確認：

ロボット架台をケミカルボルトで固定し、しっかりと取り付けられていることを確認します。ロボットが移動しても架台の揺れはありません。
- ロボットのゼロ位置を確認：

ロボットの **ゼロ位置** が正しいことを確認します。そうでなければ、ゼロ位置を再キャリブレーションする必要があります。
- ロボットの可動範囲を確認：
  - ロボットの動作範囲：ロボットは動作範囲、特にエッジ、最高層、最低層、コーナーにある部品を正常に把持・配置することができます。
  - ロボットの特異点：ロボットを手動で移動し、ロボットの動作範囲に特異点があるかどうかを確認します。ロボットの動作範囲に特異点がある場合、Mech-Mind 株式会社にお問い合わせください。
- ロボットの絶対精度を確認：

ロボットの絶対精度の不正確さによる TCP の不正確さを回避するために、ロボットの絶対精度を確認します。

### 1.1.3 ハードウェアの取り付け

取り付け前の確認が完了したら、カメラ、産業用 PC と付属品の取り付けを行います。

#### カメラの取り付け

カメラは、以下の2つの取り付け方法に対応しています。

- L型アダプターによる取り付け（カメラ背面にL型アダプターを取り付けた状態で出荷されます）
- カメラのねじ穴による取り付け

カメラの取り付けは、付属品ボックス内のネジを使用するか、お客様ご自身でスパナやネジをご用意いただき、カメラを取り付けることができます。必要なネジとスパナのサイズは、以下のように取得することができます。

- CAD モデルをダウンロード します。
- 技術仕様の寸法情報を確認 します。

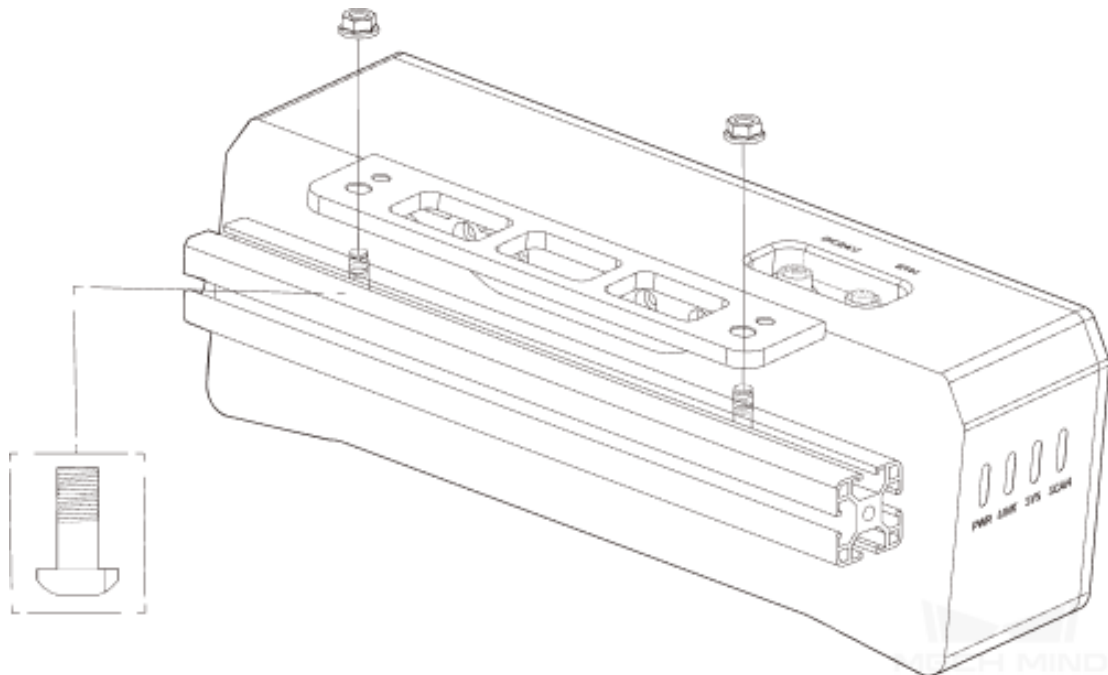
---

**ヒント:** カメラを取り付けたら、レンズ保護フィルムを剥がしてご使用ください。

---

#### L 字型アダプターによる取り付け

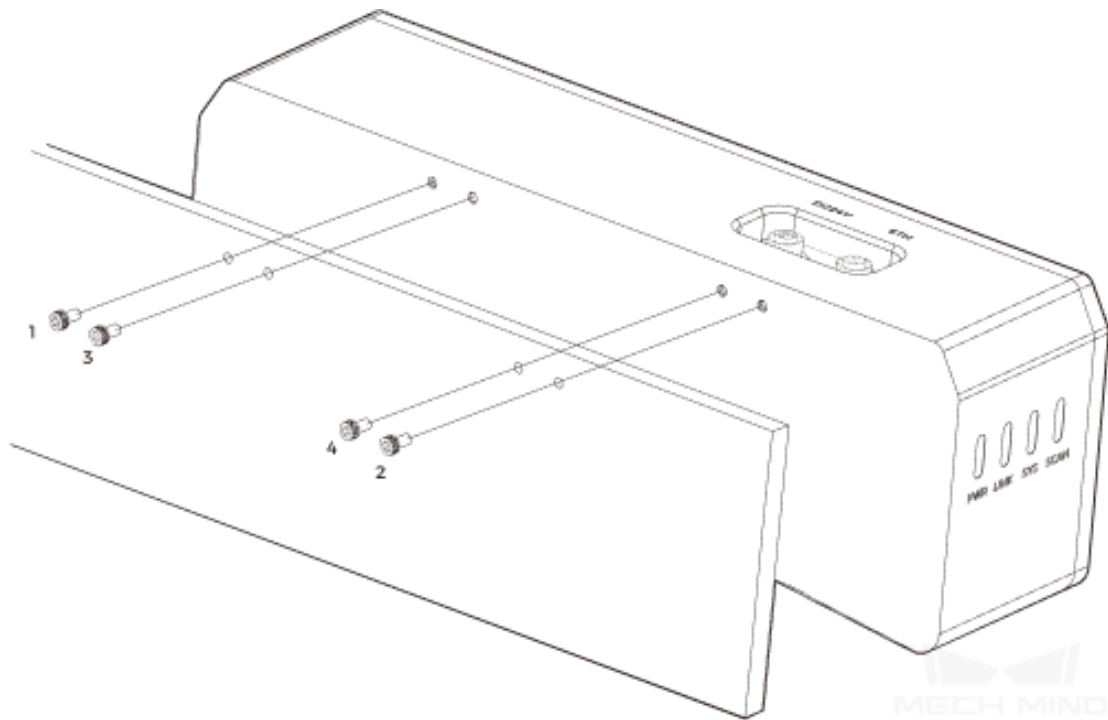
下図に示すように、レンチを使用して2本のネジを締め、カメラを固定します。





**カメラのねじ穴による取り付け**

下図に示すように、カメラを取り付ける際にレンチを使用してネジを順番にあらかじめ軽く締めてから、ネジを締め付けます。




---

**ヒント:** カメラを取り付ける前に、スパナを使用してL型アダプターを取り外す必要があります。

---

**産業用 PC の取り付け（オプション）**

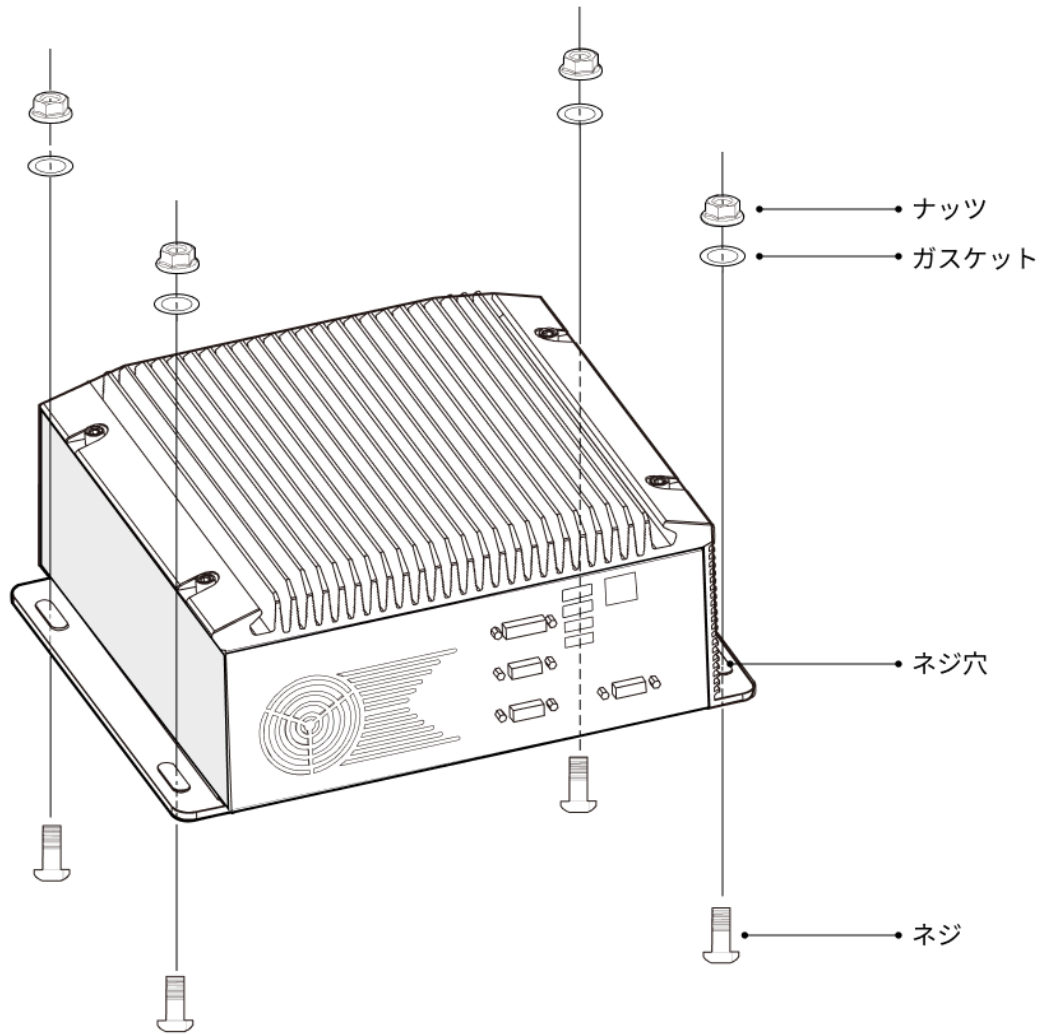
産業用 PC をラックにしっかり固定する場合は、以下の内容を参照して取り付けてください。

---

**ヒント:** スパナ、ネジ、ナット、スペーサーは各自でご用意ください。

---

下図に示すように、ネジ、ガスケット、ナットの順で配置し、スパナでナットを締め付けます。



### 付属品の取り付け

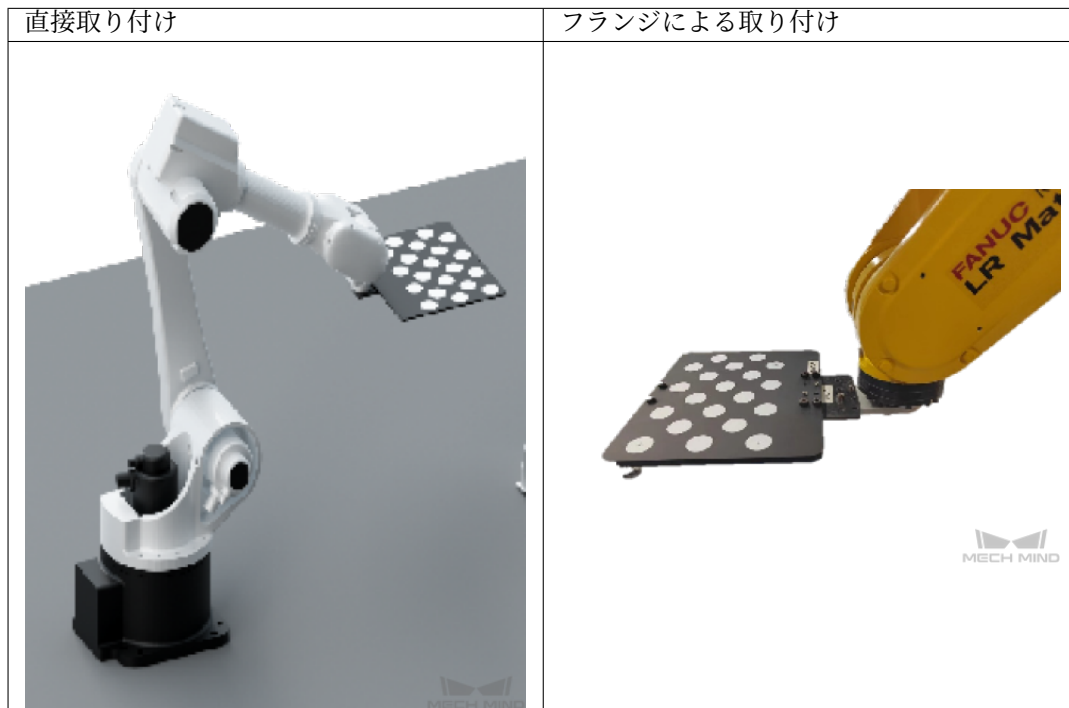
カメラが Eye to Hand 方式で取り付けられている場合、外部パラメータをキャリブレーションやチェックするには、フランジとキャリブレーションボードの取り付けが必要です。

## フランジの取り付け

外部パラメータをチェックするときは、キャリブレーションボードをロボットエンドに取り付ける必要があります。通常、フランジでロボットとキャリブレーションボードを接続します。

## キャリブレーションボードの取り付け

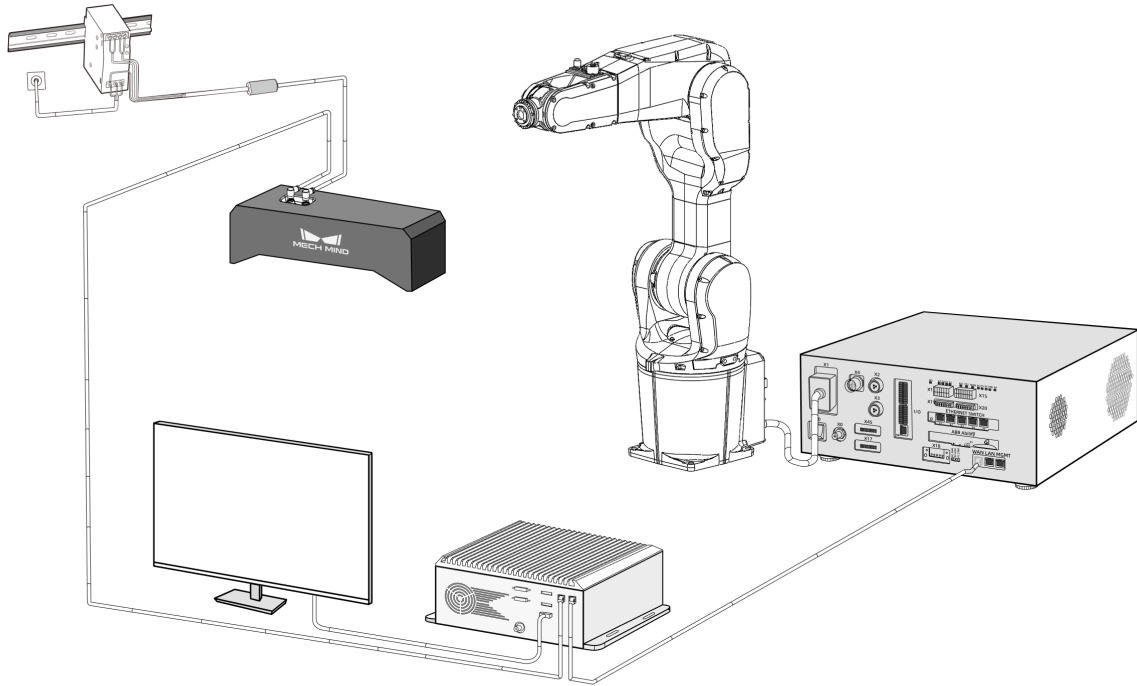
キャリブレーションボードの一般的な取り付け方法は以下の通りです。



### 1.1.4 ハードウェアの接続

Mech-Mind ビジョンシステムのハードウェア接続は、主に次のような構成になっています。

- カメラの接続：
  - LAN ケーブルでカメラを産業用 PC に接続します。
  - レール電源でカメラの電源を入れます。
- 産業用 PC の接続：
  - LAN ケーブルで産業用 PC をカメラに接続します。
  - 電源アダプタで産業用 PC の電源を入れます。
  - HDMI ケーブルで産業用 PC とモニターを接続します。HDMI ケーブルとモニターはお客様でご用意ください。
- ロボットコントローラの接続：
  - LAN ケーブルでロボットコントローラを産業用 PC に接続します。LAN ケーブルはお客様でご用意ください。
  - 電源アダプタでロボットコントローラの電源を入れます。


**注意:**

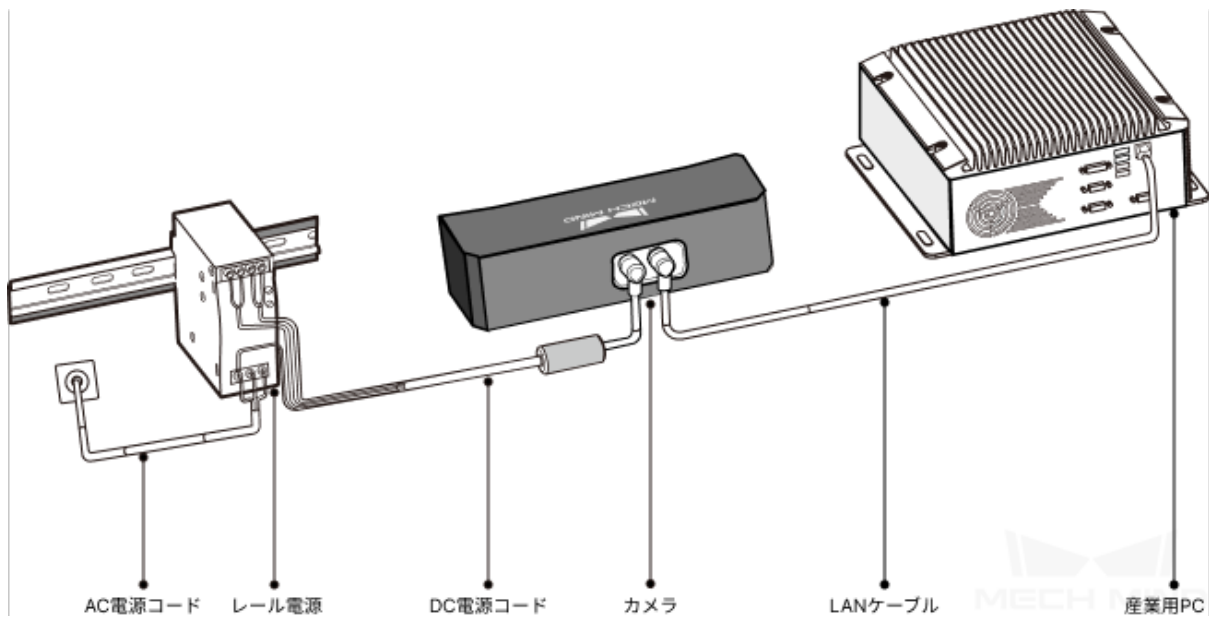
- 接続するときは、最後に電源を入れてください。
- カメラ、産業用 PC、ロボットコントローラは LAN ケーブルで接続されています。LAN ポートが足りない場合は、ギガビットスイッチを使用して接続することができます。
- ケーブルの外観は基本的に同じなので、接続する前にラベルで区別することをお勧めします。

**カメラを接続**

**注意:** Eye in Hand 方式でカメラが取り付けられている場合、カメラの配線に注意が必要です。ロボットエンドの可動範囲が広いときにロボットのケーブルパックやカメラのケーブルに引っ張りがかかる場合、ロボットの正常な動作に影響を与えることなく、ソフトウェアでロボットエンドの可動範囲を設定することができます。

カメラの接続は 2 つの部分で構成されています。

- LAN ケーブルでカメラを産業用 PC に接続します。
- レール電源でカメラの電源を入れます。



### DC 電源コードを接続

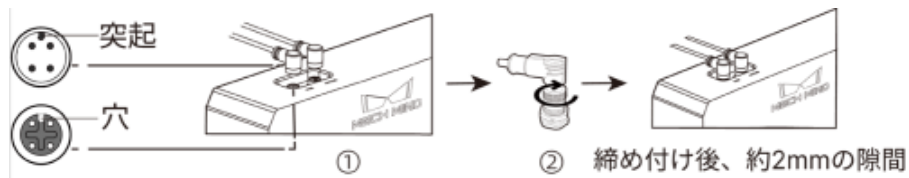
下図に示すように、DC 電源コードは DC24V 電源コネクタに差し込みます。

1. 電源コードのプラグの突起をカメラの電源コネクタの穴に挿入します。
2. 固定ナットをしっかり締めます。

---

**ヒント:** ナットを締めるときの推奨トルクは 16N・m です。

---



### カメラの LAN ケーブルを接続

LAN ケーブルの航空コネクタプラグは、ETH の LAN ポートに差し込みます。

1. 航空コネクタプラグの突起を ETH の LAN ポートに差し込みます。
2. 固定ナットをしっかり締めます。

---

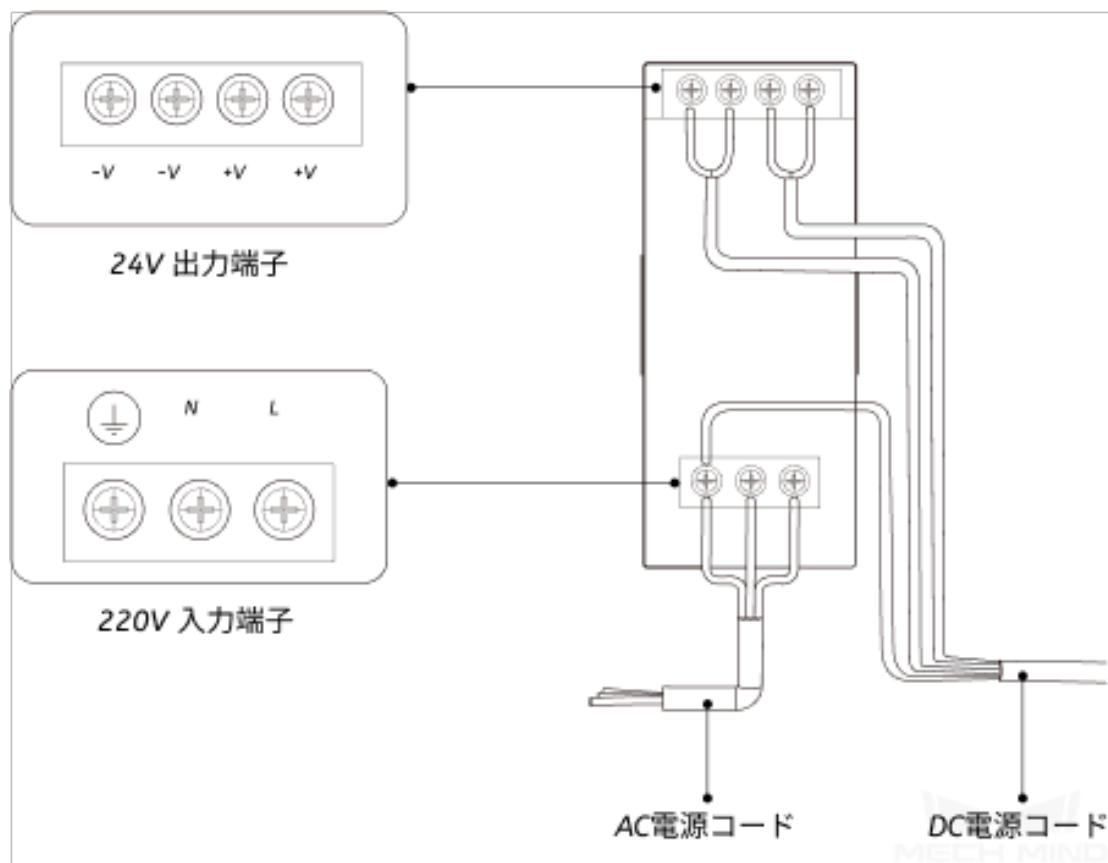
**ヒント:** ナットを締めるときの推奨トルクは 16N・m です。

---

## レール電源を接続

下図に示すように、レール電源の配線を接続する場合は、対応する入出力電圧端子にプラグを接続する必要があります。

1. AC 電源コード接続用の 3 連端子プラグ (PE、N、L) をそれぞれ対応する 220V 入力端子 (⊕、N、L) に差し込みます。
2. DC 電源コードを接続します。
  - +V を 24V 出力端子の+V に接続します。
  - -V を 24V 出力端子の-V に接続します。
  - PE を 220V 入力端子 ⊕ に接続します。


**警告:**

- レール電源の接地端子は必ず接地してください。
- レール電源は、配電ボックスで使用する必要があります。また、レールまたはレールが接続されている配電ボックスは確実に接地する必要があります。
- レール電源が複数ある場合は、一定の距離を置いて設置する必要があります。

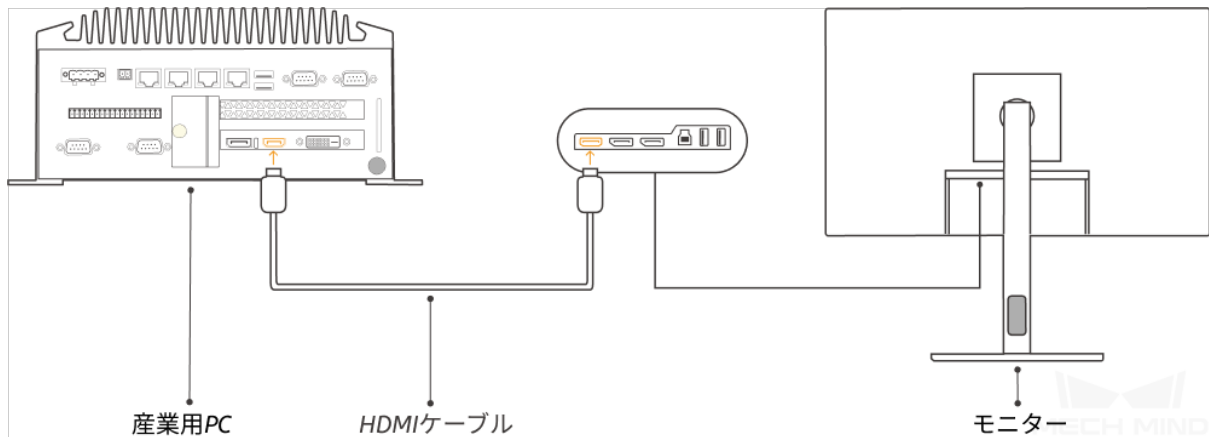
## 産業用 PC を接続

産業用 PC の接続：

- HDMI ケーブルで産業用 PC とモニターを接続します（お客様でご用意ください）。
- LAN ケーブルで産業用 PC をカメラに接続します。
- 電源アダプタで産業用 PC の電源を入れます。
- ドングルを差し込みます。

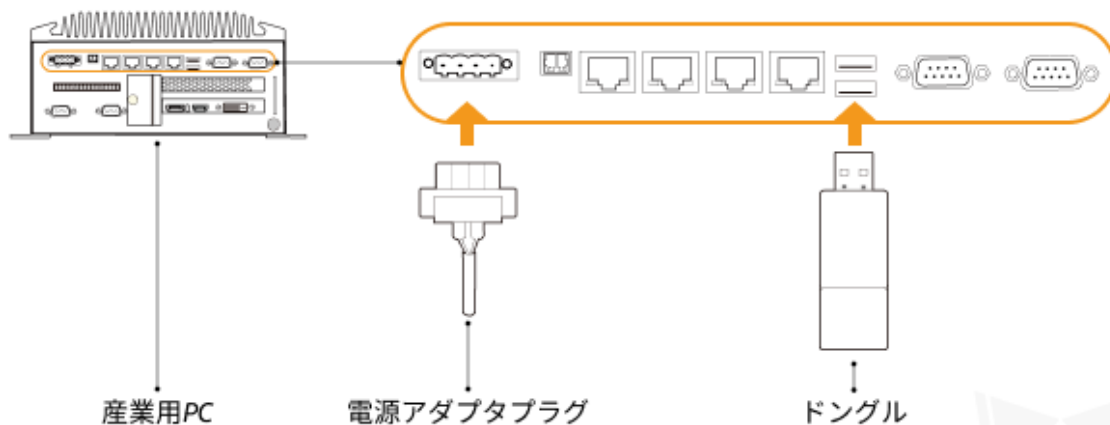
## HDMI ケーブルを接続

下図に示すように、HDMI ケーブルの一端をモニター背面の HDMI コネクタに、もう一端を産業用 PC の HDMI コネクタに接続します。



## ほかのものを接続

下図に示すように、電源アダプタの電源プラグを産業用 PC の電源入力ポートに、dongle を USB ポートに差し込みます。



ヒント：

- 産業用 PC の IP アドレスを設定するときは、LAN ケーブルを産業用 PC の LAN ポートに接続します。
- LAN ケーブルには、接続先を区別するためのラベルを貼ることをお勧めします。

### ロボットコントローラを接続

両端に RJ45 コネクタの付いたケーブルを用意し、その一端をロボットコントローラの LAN ポートに差し込みます。その後、ロボットコントローラの電源を入れます。詳細については、ロボット取扱説明書をご参照ください。



ヒント： ケーブルはお客様でご用意ください。

### 1.1.5 取り付け後の確認

すべての取り付けが完了したら、カメラ、産業用 PC、ロボットコントローラが正常に動作できるかどうかを確認する必要があります。

#### カメラを確認

カメラの PWR インジケータと LINK インジケータは、常に緑色で点灯しています。それ以外の場合は異常ですので、Mech-Mind 株式会社にお問い合わせください。

#### 産業用 PC を確認

電源投入後のモニター画面は正常であり、正常に使用できます。

#### ロボットを確認

起動後、ティーチングペンダントにエラーメッセージは表示されず、ロボットを動かすことができます。

ロボットの設定が完了し、高速で走行するようになったら、ロボット架台が揺れていないか確認することが重要です。ロボット架台が揺れる場合は、より多くのケミカルボルトで固定する必要があります。

## 1.2 ソフトウェアのインストール

Mech-Mind ビジョンシステムは、Mech-Eye SDK、Mech-Vision、Mech-Viz、Mech-Center および Mech-DLK で構成される Mech-Mind ソフトウェアシステムを使用しています。

Mech-Mind から購入した産業用 PC には、Mech-Mind ソフトウェアシステムがプリインストールされています。本機のソフトウェアが最新バージョンであるかどうかを確認してください。最新バージョンでない場合は、以下の節を参照して最新バージョンのソフトウェアをインストールしてください。

お客様のホストコンピュータを産業用 PC として使用する場合は、その仕様と構成が ipc\_introduction の要件を満たしていることを確認してから、ソフトウェアを順番にインストールしてください。



### 1.2.1 Mech-Eye SDK をインストール

Mech-Eye SDK をダウンロードしてインストールします。

### 1.2.2 Mech-Vision、Mech-Viz および Mech-Center をインストール

Mech-Vision、Mech-Viz および Mech-Center ソフトウェア をダウンロードしてインストールします。

### 1.2.3 Mech-DLK をインストール

Mech-DLK をダウンロードしてインストールします。

## 1.3 産業用 PC の LAN ケーブルの接続と IP アドレス設定

ソフトウェアのインストールが完了したら、LAN ケーブルを産業用 PC の LAN ポートに接続し、カメラ、産業用 PC およびロボットの IP アドレスを設定します。次の 2 つの IP アドレスは、同じネットワークセグメント内に設定する必要があります。

- カメラと産業用 PC でカメラを接続するための LAN ポート
- ロボットと産業用 PC でコントローラを接続するための LAN ポート

---

#### ヒント:

- 産業用 PC の LAN ポートは、同じネットワークセグメントに設定しないでください。
- 産業用 PC の IP アドレス設定の失敗を回避するために、産業用 PC の IP アドレスを設定してから、カメラの IP アドレスとロボットの IP アドレスを設定してください。
- ギガビットスイッチは、LAN ポートが足りない場合に使用することができます。

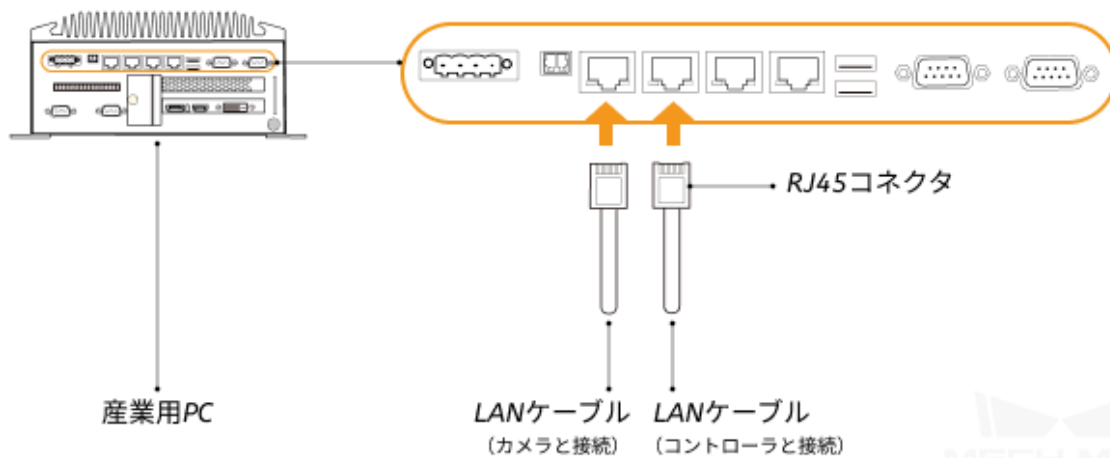
---

LAN ケーブル接続と IP アドレス設定の流れは次の通りです。

1. カメラの LAN ケーブルの RJ45 コネクタを、産業用 PC の LAN ポートに差し込みます。詳細については、[LAN ケーブルの接続](#) をご参照ください。
2. 産業用 PC でカメラの LAN ポートに接続する IP アドレスを設定します。詳細については、[産業用 PC の IP アドレスを設定](#) をご参照ください。
3. ロボットコントローラの LAN ケーブルに接続する RJ45 コネクタを、産業用 PC の LAN ポートに差し込みます。詳細については、[LAN ケーブルの接続](#) をご参照ください。
4. 産業用 PC でロボットコントローラに接続する IP アドレスを設定します。詳細については、[産業用 PC の IP アドレスを設定](#) をご参照ください。
5. [カメラの IP アドレスを設定](#) します。
6. [ロボットの IP アドレスを設定](#) します。
7. [産業用 PC とカメラ、ロボットとのネットワーク接続を確認](#) します。

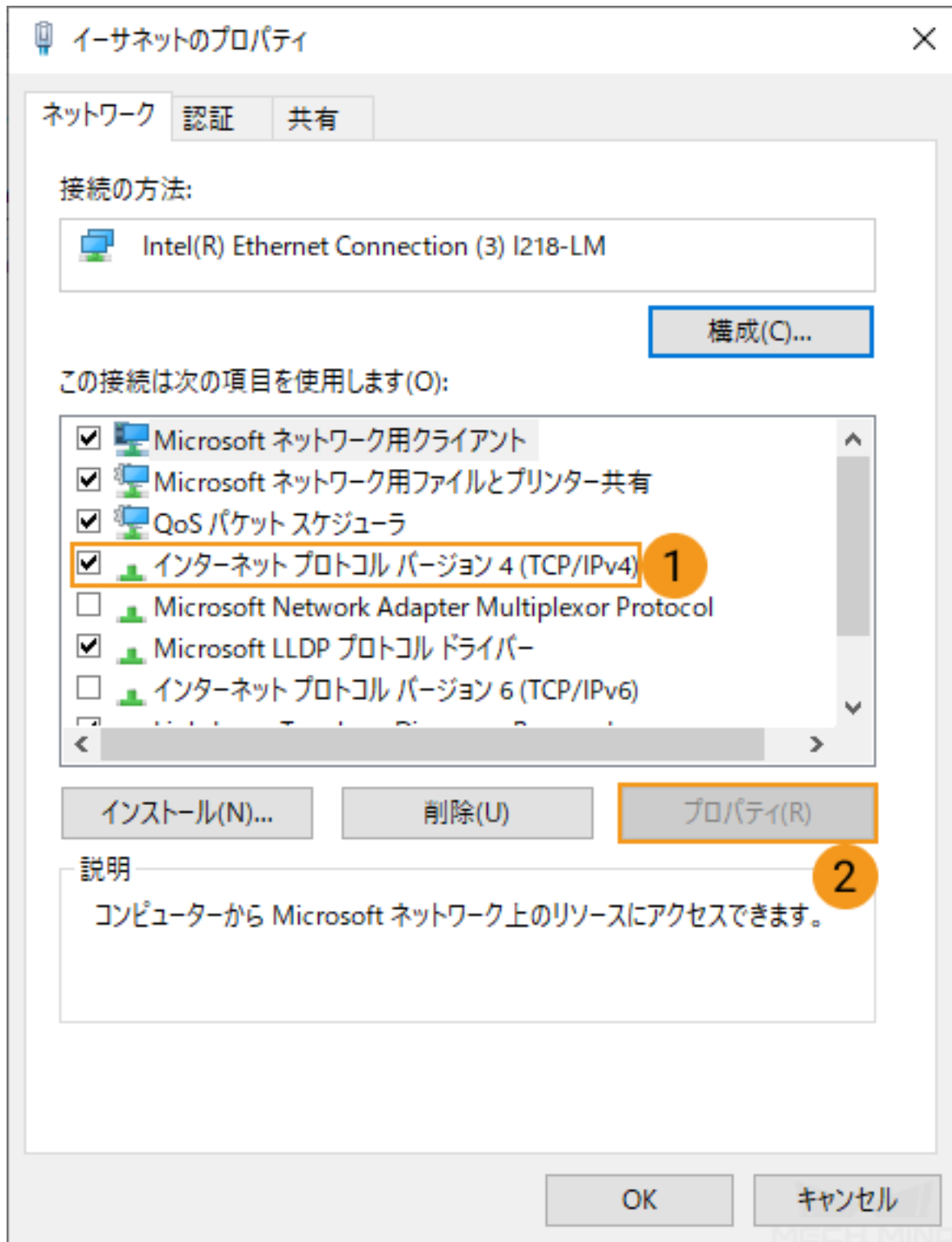
### 1.3.1 LAN ケーブルの接続

LAN ケーブルの RJ45 コネクタを産業用 PC の LAN ポートに差し込みます。



### 1.3.2 産業用 PC の IP アドレスを設定

1. 産業用 PC で **コントロールパネル・ネットワークとインターネット・ネットワークと共有センター・アダプターの設定の変更** を選択し、**ネットワーク接続** 画面にアクセスします。
2. カメラまたはロボットが接続されている LAN ポートを選択し、右クリックして **プロパティ** を選択してプロパティ画面にアクセスします。
3. **インターネットプロトコルバージョン 4 (TCP/IPv4)・プロパティ** を選択して IP 設定画面にアクセスします。



4. 次の IP アドレスを使う をクリックし、IP アドレスと サブネットマスク を入力し、OK をクリックすると設定が完了します。


**ヒント:**

- IP アドレスを必ず 1 つだけ設定します。
- サブネットマスクは通常、**255.255.255.0** となります。


**ヒント:**

- カメラやロボットに接続されている LAN ポートの名前は、認識しやすいように変更することをお勧めします。**ネットワーク接続** 画面で、LAN ポートを右クリックして **名前の変更** を選択し、変更を行います。
- 産業用 PC の個々の LAN ポートは、競合が発生する可能性があるため、同じネットワークセグ

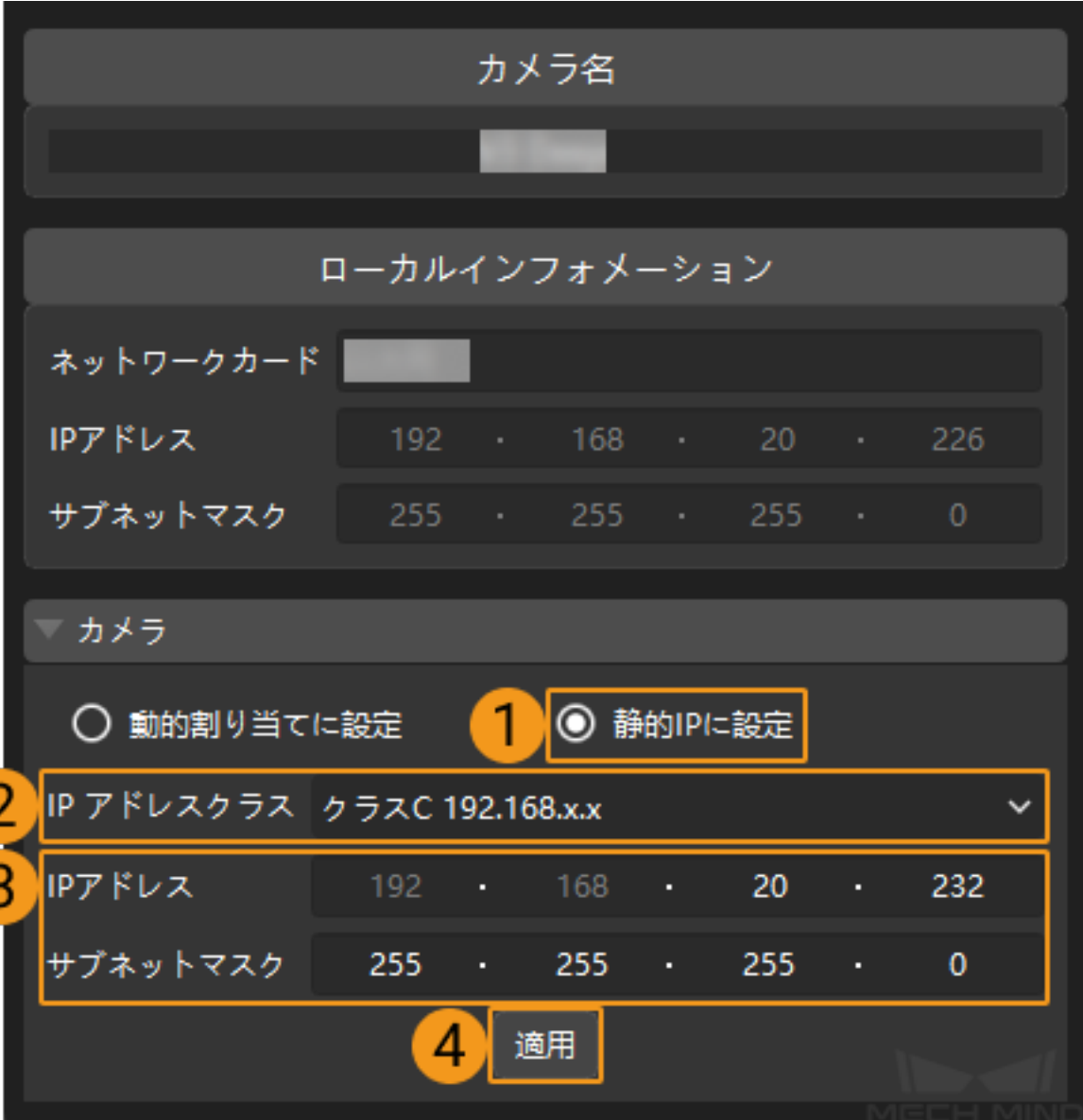
メントにないことが必要です。

### 1.3.3 カメラの IP アドレスを設定

1. 産業用 PC で Mech-Eye Viewer を起動します。

2. カメラを選択し、マウスをカメラ情報バーに移動すると、が表示されます。クリックすると IP 設定画面に入ります。

3. **静的 IP に設定** を選択し、実際の状況に応じて **IP アドレスのタイプ** を選択します。カメラの IP アドレスとサブネットマスクを入力してから、**適用** をクリックします。すると、設定が完了します。



カメラ名

ローカルインフォメーション

ネットワークカード

IPアドレス 192 . 168 . 20 . 226

サブネットマスク 255 . 255 . 255 . 0

▼ カメラ

動的割り当てに設定 **1**  静的IPに設定

**2** IP アドレスクラス クラスC 192.168.x.x

**3** IPアドレス 192 . 168 . 20 . 232

サブネットマスク 255 . 255 . 255 . 0

**4** 適用

ヒント:

- IP アドレスを必ず1つだけ設定します。
- ローカルインフォメーションでは、カメラに接続されている産業用 PC のネットワークカードの IP アドレスとサブネットマスクが表示されます。カメラの IP アドレスとサブネットマスクは、この情報に基づいて設定することができます。

### 1.3.4 ロボットの IP アドレスを設定

ロボットのマニュアルを参考にしてロボットの IP アドレスを設定します。

### 1.3.5 産業用 PC とカメラ、ロボットとのネットワーク接続を確認

産業用 PC、カメラ、ロボットの IP アドレスが設定できたら、それらの間のネットワーク接続を確認する必要があります。手順は以下の通りです。

1. ショートカットキー「Window + R」で実行画面を表示します。
2. 名前に **cmd** と入力し、OK をクリックします。
3. コマンドウィンドウに **ping XXX.XXX.XX.XX** と入力し、Enter をクリックしてコマンドを実行します。

**ヒント:** XXX.XXX.XX.XX は、実際に設定されたカメラまたはロボットの IP アドレスとなります。

4. コマンドの戻り値を確認し、IP アドレスとの通信が正常であるかどうかを判断します。
  - 正常に通信した場合、以下のメッセージが表示されます。
 

```
XXX.XXX.XX.XX に Ping を送信しています 32 バイトのデータ :
XXX.XXX.XX.XX からの応答: バイト数=32 時間<1ms TTL=128
XXX.XXX.XX.XX からの応答: バイト数=32 時間<1ms TTL=128
XXX.XXX.XX.XX からの応答: バイト数=32 時間<1ms TTL=128
XXX.XXX.XX.XX からの応答: バイト数=32 時間<1ms TTL=128
```
  - 通信が正常に行われない場合 (IP アドレスの競合が原因の可能性あり)、以下のメッセージが表示されます。この場合、IP アドレスの再設定が必要になります。
 

```
XXX.XXX.XX.XX に Ping を送信しています 32 バイトのデータ :
XXX.XXX.XX.XX からの応答: 宛先ホストに到達できません。
XXX.XXX.XX.XX からの応答: 宛先ホストに到達できません。
XXX.XXX.XX.XX からの応答: 宛先ホストに到達できません。
XXX.XXX.XX.XX からの応答: 宛先ホストに到達できません。
```

## 1.4 画像品質の確認

産業用 PC、カメラとロボット間のネットワークにアクセスできることを確認した後、以下の手順で 2D 画像の品質を確認します。

1. Mech-Eye Viewer でカメラを接続し、画像を取得します。
2. 2D 画像が以下の要件を満たしていることを確認します。
  - 部品をカメラの視野中心に置き、エッジと最高層にある部品が視野に入ること。
  - デフォーカスを起こさないこと。

## 応用事例

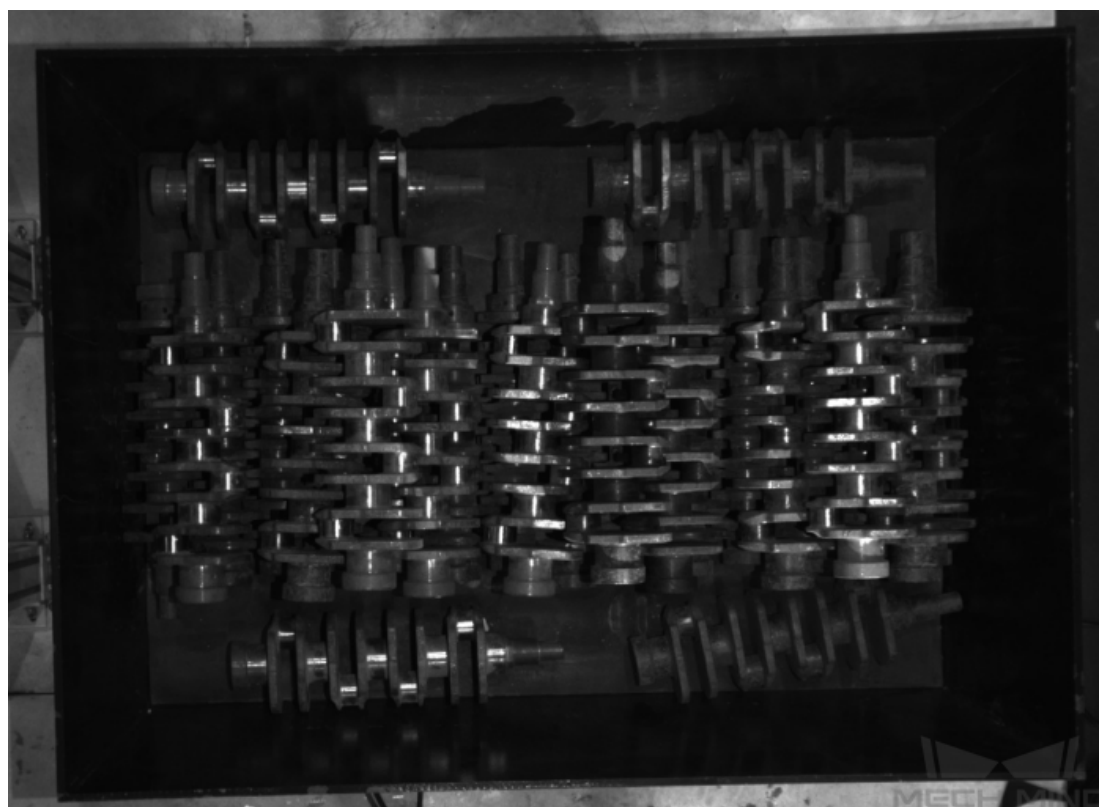
本章では、Mech-Mind ビジョンシステムを適用した代表的な応用事例について説明します。

### 2.1 整列して並べられたクランクシャフト供給

#### 2.1.1 プロジェクトの背景

##### 部品

本事例が対象としている部品は、重い、複雑な形状を持つ、かつ反射しない曲面状のクランクシャフトです。本事例では、クランクシャフトが箱に整列して並べられ、同じ層の部品同士は互いに積み重ねられず、それらの表面には認識しやすい明らかなエッジ特徴があります。



## ロボット

可搬質量 7Kg の 6 軸ロボット、KAWASAKI\_RS007L を採用します。

## プロジェクトの目標

ビジョンシステムにより、ロボットが段ごとに部品を 1 つずつ把持し、指定場所に配置します。プロジェクトは、次の要件があります。

- ピッキング精度：±0.5~3mm
- 認識の成功率：99-99.99%
- タクト：4s
- 取り切り率：99%以上

## プロジェクトの内容

### カメラ

本事例で扱う部品と使用する箱のサイズは大きいです。そのため、プロジェクトの精度要件を満たすために、Mech-Eye LSR L または PRO M カメラを使用し、Eye to Hand 方式でスタンドに取り付けることをお勧めします。

### 通信方式

Viz ティーチングにより 3D ビジョンシステムとロボットとの通信を実現します。Mech-Viz でロボットをガイドしてクランクシャフトの把持と配置を行います。

### ビジョンソリューション

Mech-Vision はカメラをトリガーして撮影し、画像を処理してから把持位置姿勢を出力します。また、Mech-Viz プロジェクトを通じて経路計画および衝突検出を実行します。

## 2.1.2 ソリューションのデプロイ

### ビジョンシステムを構築してロボットに接続

ビジョンソリューションをデプロイする前に、ビジョンシステムの構築とロボットの接続を完了させてください。

詳細については、以下の内容をお読みください。

- [カメラの取り付け](#)
- [ビジョンシステム全体の接続](#)
- [Viz との通信プログラムをロボットにロード](#)
- [ロボットの接続](#)



## ロボットのハンド・アイ・キャリブレーションを実行

自動キャリブレーション (ETH) を参照してキャリブレーションを完了させます。

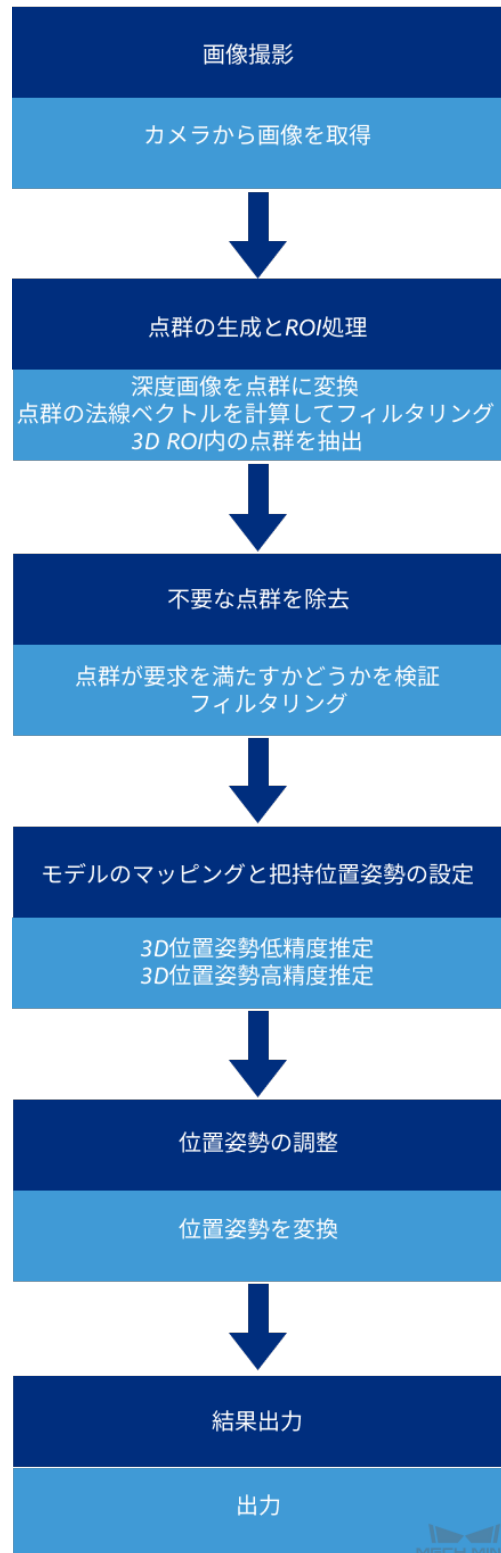
## Mech-Vision プロジェクトを構築

本節では、クランクシャフト供給の Mech-Vision プロジェクトの構築について説明します。

- プロジェクトの構築手順
- 手順の詳しい解説
  - 画像取得
  - 点群の生成と ROI 処理
  - 不要な点群を除去
  - モデルのマッチングと把持位置姿勢の設定
  - 位置姿勢を調整
  - 複数の把持位置姿勢にマッピング
  - ビジョン結果を出力
- プロジェクトを実行してデバッグ

### プロジェクトの構築手順

今回作成するプロジェクトは以下のような流れで処理を行います。



1. カメラをトリガーして撮影し、クランクシャフトのカラー画像と深度画像を取得します。
2. 取得した画像に対して処理を行い、深度画像を点群に変換します。背景点群と不要な点群による干渉を減らすために、外れ値を除去します。
3. プロジェクト処理を高速化するため、設定されたしきい値に従って点数が範囲外の点群を除去します。

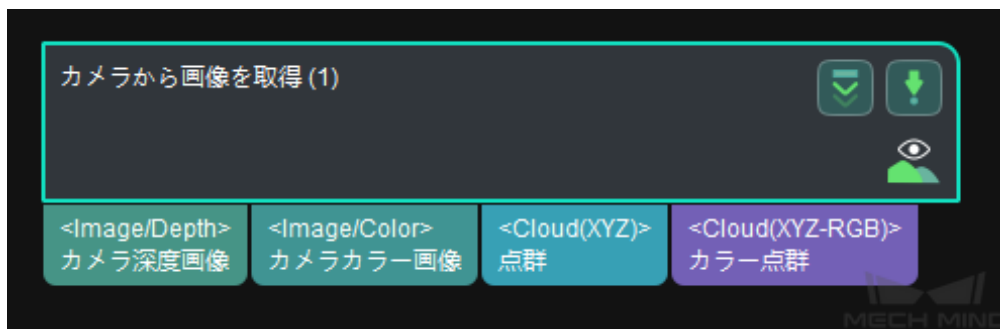
4. モデルのマッチングと把持位置姿勢の設定を行います。対象物の点群モデルのマッチングにより、シーン内の対象物を認識して把持位置姿勢を取得することができます。この手順を実行する前に、点群モデルファイルおよび幾何学的中心ファイルを事前に用意する必要があります。
5. 対象物の把持位置姿勢を取得してから座標系の変換を行います。把持を実行するために、対象物座標系をロボット座標系に変換します。
6. ビジョン結果を出力します。座標系が変換された後、外部サービスを介してビジョン結果を Mech-Viz または Mech-Center に送信します。

### 手順の詳しい解説

プロジェクトを新規作成してから、プロジェクトの構築と関連設定を行います。

### 画像取得

まずはカメラをトリガーして撮影し、カラー画像と深度画像を取得します。ステップ `capture_images_from_camera` を追加します。



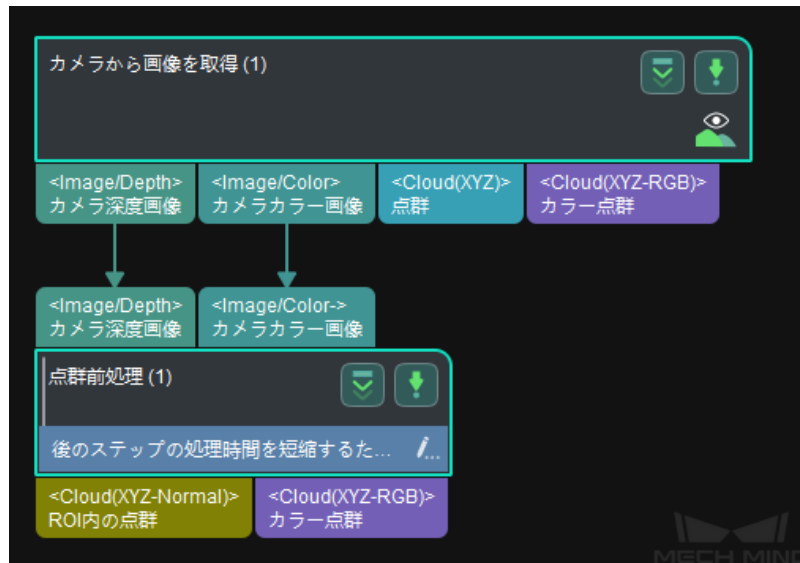
**注意:** このステップを追加する前に、カメラが取り付けられていること、産業用 PC と接続されていることを確認し、取得した画像の品質確認とロボットハンド・アイ・キャリブレーションを完了させる必要があります。

このステップを追加した後、実際のカメラに接続する必要があります。このステップを選択し、ステップパラメータをクリックして、ステップパラメータの画面でカメラタイプ、カメラ ID およびキャリブレーションパラメータグループを選択し、IP アドレスを入力します。

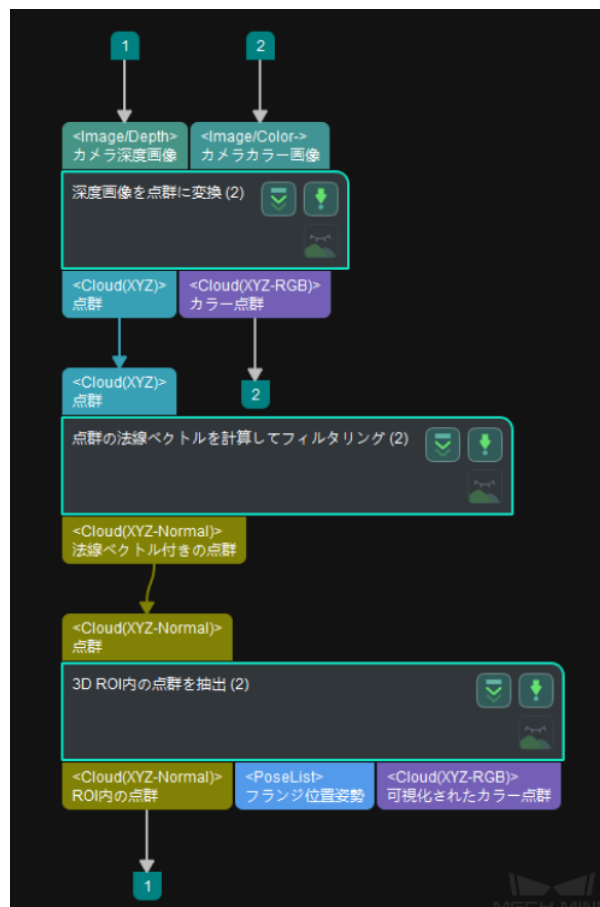


## 点群の生成と ROI 処理

続いて point\_cloud\_preprocessing のステップの組合せを追加します。

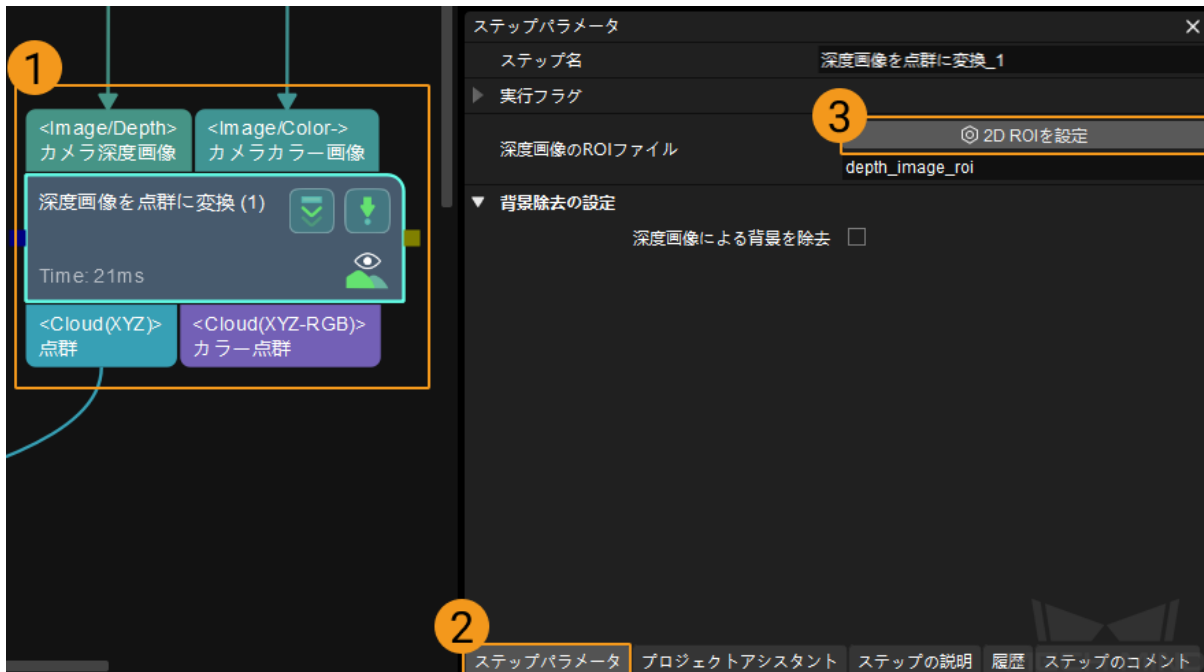


ダブルクリックしてその中に入ります。すると、`from_depth_map_to_point_cloud`、`calc_normals_of_point_cloud_and_filter_it`、`extract_3d_points_in_3d_roi` の3つのステップが表示されます。



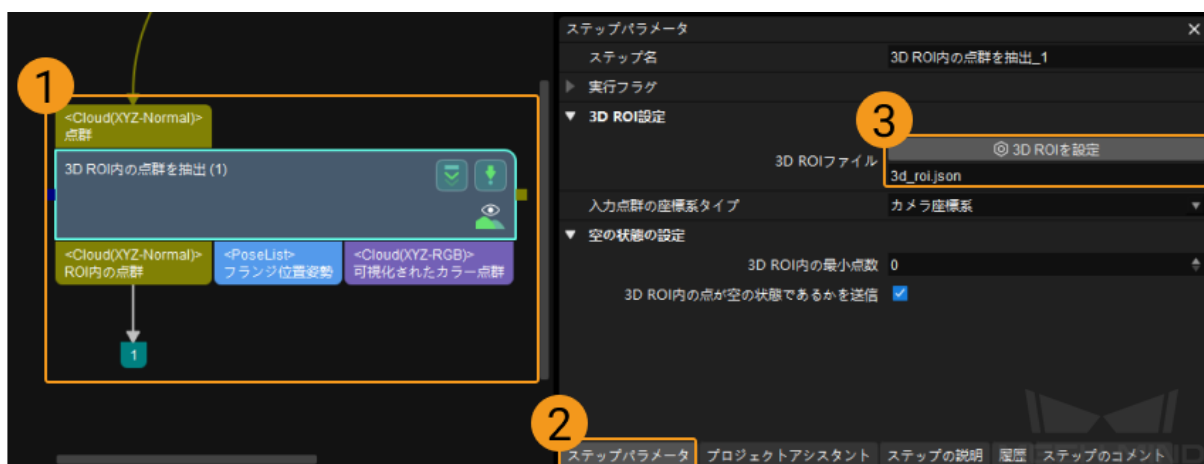
1. ステップ `from_depth_map_to_point_cloud` を使用してカメラで取得した深度画像を点群に変換します。不要な点群の干渉を減らし、後続のステップの処理時間を短縮するために、2D 関心領域を選択する必要があります。

このステップを選択し、**ステップパラメータ** をクリックして **2D ROI を設定** をクリックします。クリック後、表示される画面で 2D ROI の設定を行います。設定方法については、`set_ROI` をご参照ください。

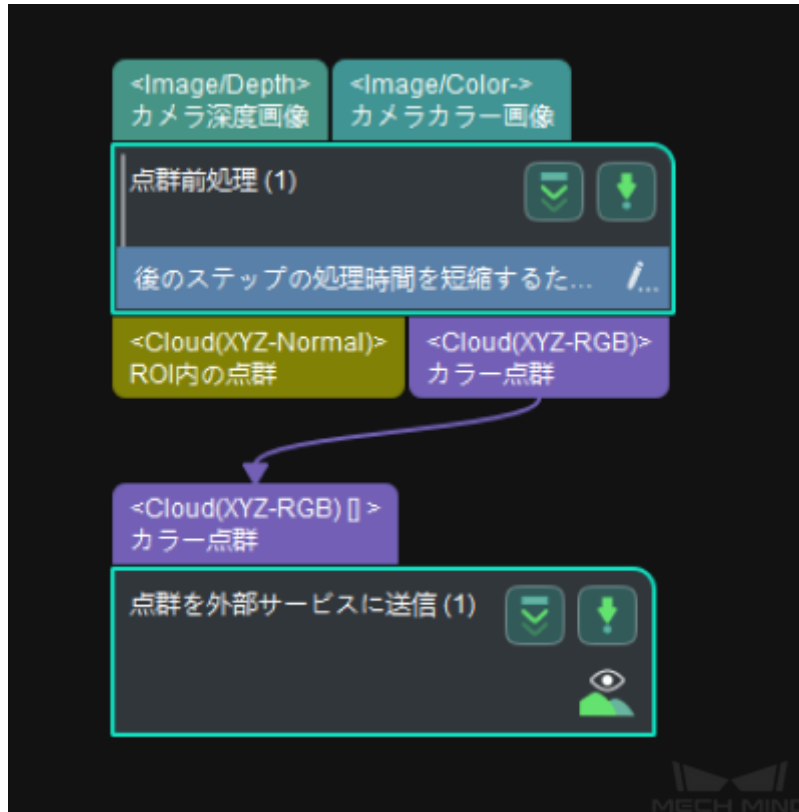


2. ステップ `calc_normals_of_point_cloud_and_filter_it` を使用して法線ベクトルの計算を行います。このステップにより、入力点群内の各点の法線ベクトルを計算し、位置情報のみを含む点群データを法線ベクトル付きの点群データに変換することができます。
3. ステップ `extract_3d_points_in_3d_roi` を使用して 3D ROI の設定を行います。このステップにより、3D ROI 内の点群は保持され、3D ROI 外の点群は除去されるため、背景の点群の干渉が減らすことができます。

このステップを選択し、**ステップパラメータ** をクリックして **3D ROI を設定** をクリックします。クリック後、表示される画面で 3D ROI の設定を行います。設定方法については、`set_ROI` をご参照ください。

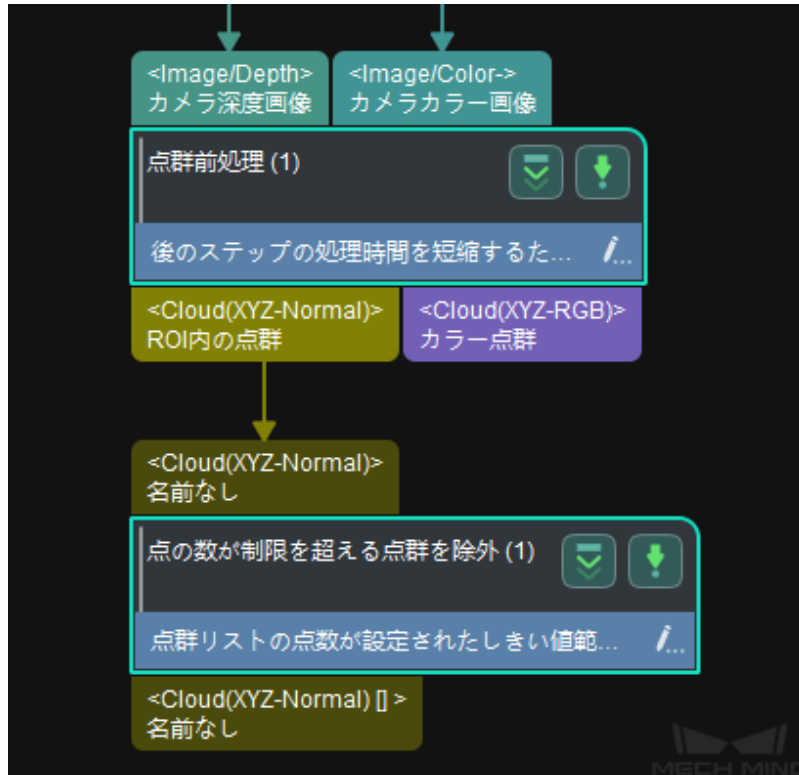


4. 点群を Mech-Viz に送信するために、このステップの組合せの後に `send_point_cloud_to_external_service` ステップをつなぎます。これにより、デバッグまたは実際効果の表示を実現できます。

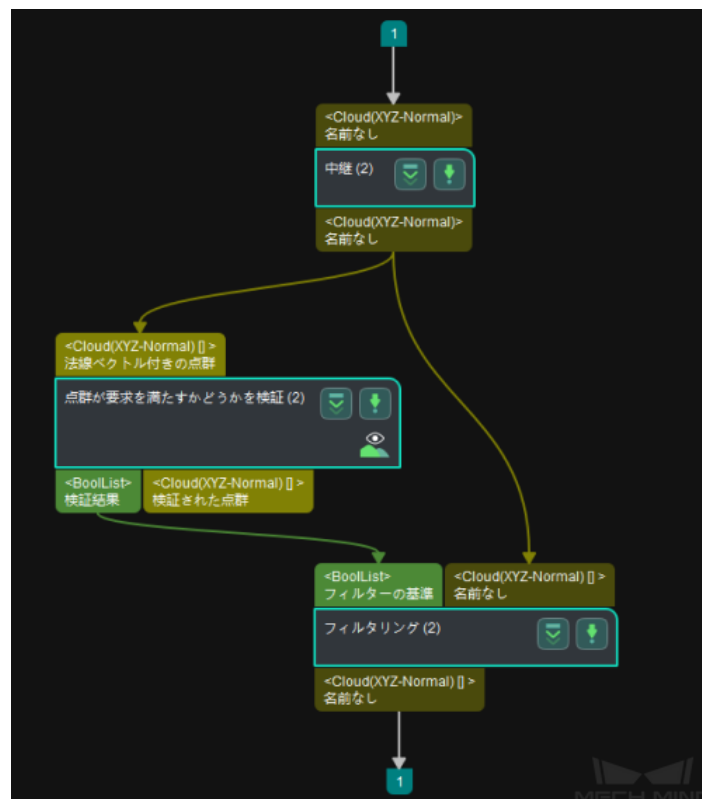


### 不要な点群を除去

次に、プロジェクトの実行を高速化し、不要な点群を除去するために、`filter_out_point_clouds_that_exceed_the_size_limit` のステップの組合せを追加します。



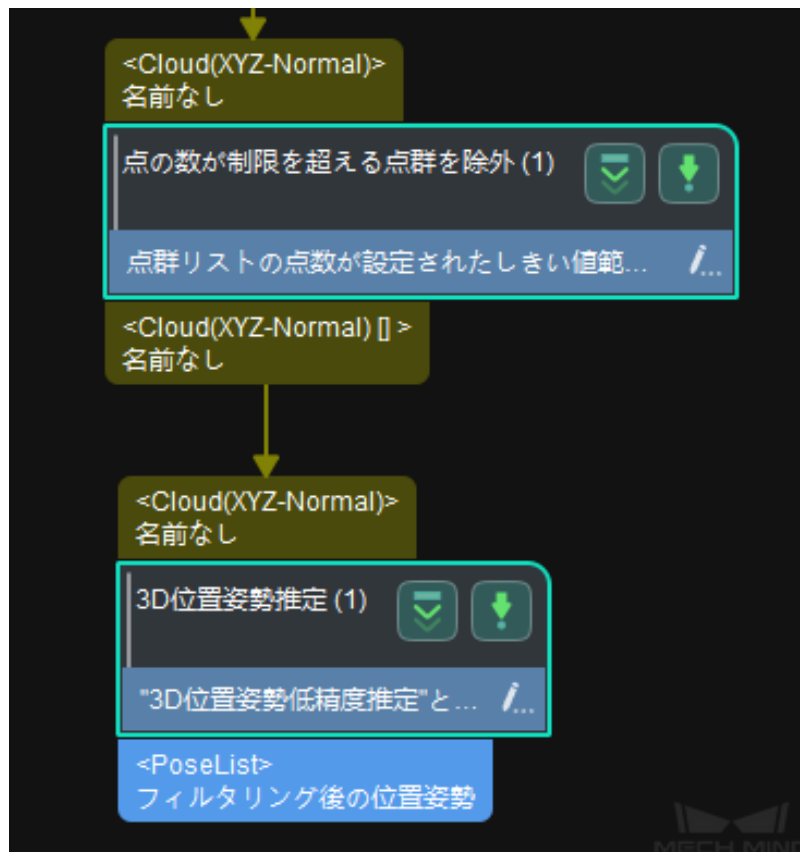
ダブルクリックしてその中に入ります。すると、`allocator`、`validate_point_clouds`、`filter` の3つのステップが表示されます。



1. ステップ `validate_point_clouds` により、設定されたルール（点群の点数のしきい値など）に従って、点群が要件を満たしているかどうかを判断できます。
2. ステップ `filter` は一般的なリストフィルターとして機能します。これにより、入力リストの要素を True/False ブール値リストの要素に一对一でマッピングし、True に対応するリストを出力できます。

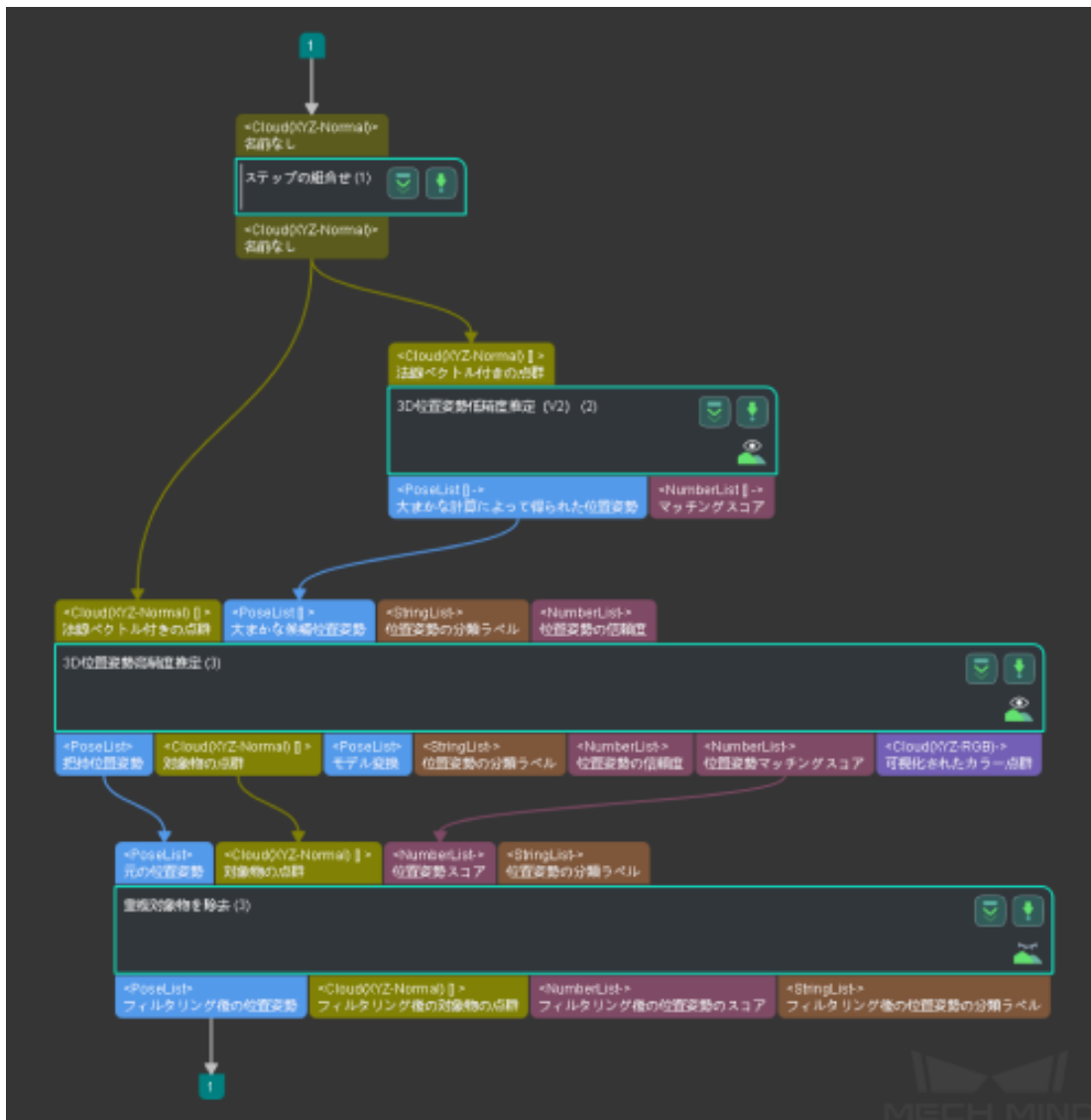
### モデルのマッチングと把持位置姿勢の設定

それから、`3d_matching_custom` ステップの組合せを追加します。




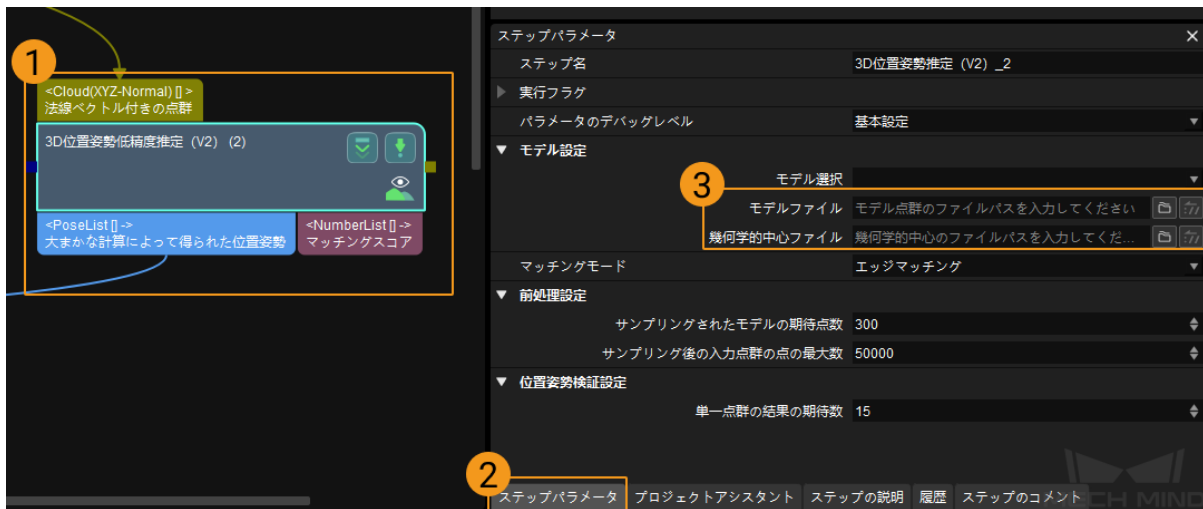
ダブルクリックしてその中に入ります。すると、`3d_coarse_matching_v2`、`3d_fine_matching`、`remove_overlapped_objects` の3つのステップが表示されます。





1. ステップ 3d\_coarse\_matching\_v2 を使用して点群の大きなマッチングを行います。このステップにより、事前に作成した点群モデルを入力した法線ベクトル付きの点群と大きめにマッチングします。また、マッチング結果が得られた後、マッチングした位置に従って対象物の幾何学的中心または把持位置姿勢が生成されます。

3d\_coarse\_matching\_v2 ステップを選択し、**ステップパラメータ** をクリックして  をクリックします。モデルファイルと幾何学的中心ファイルを選択してインポートします。



**ヒント:** モデルファイルと幾何学的中心ファイルの作成方法については、`matching_model_and_pick_point_editor` をご参照ください。

- ステップ `3d_fine_matching` を使用して正確なマッチングを行います。このステップにより、`3d_coarse_matching_v2` で得られたマッチング結果をより正確なマッチング計算を実行することでより高精度なマッチング結果を取得し、高精度の幾何学的中心または把持位置姿勢を生成できます。

**ヒント:** `3d_coarse_matching_v2` ステップと同じ方法で、モデルファイルと幾何学的中心ファイルを `3d_fine_matching` ステップに追加します。

このステップのパラメータを調整するときは、実際の状況に応じてサンプリング間隔を設定する必要があります。サンプリング間隔が小さいほど、サンプリングによって得られた点群の数は多く、モデル推定の精度が高くなります。サンプリング間隔を設定後、標準偏差を調整します。詳細については、`3d_fine_matching` をご参照ください。

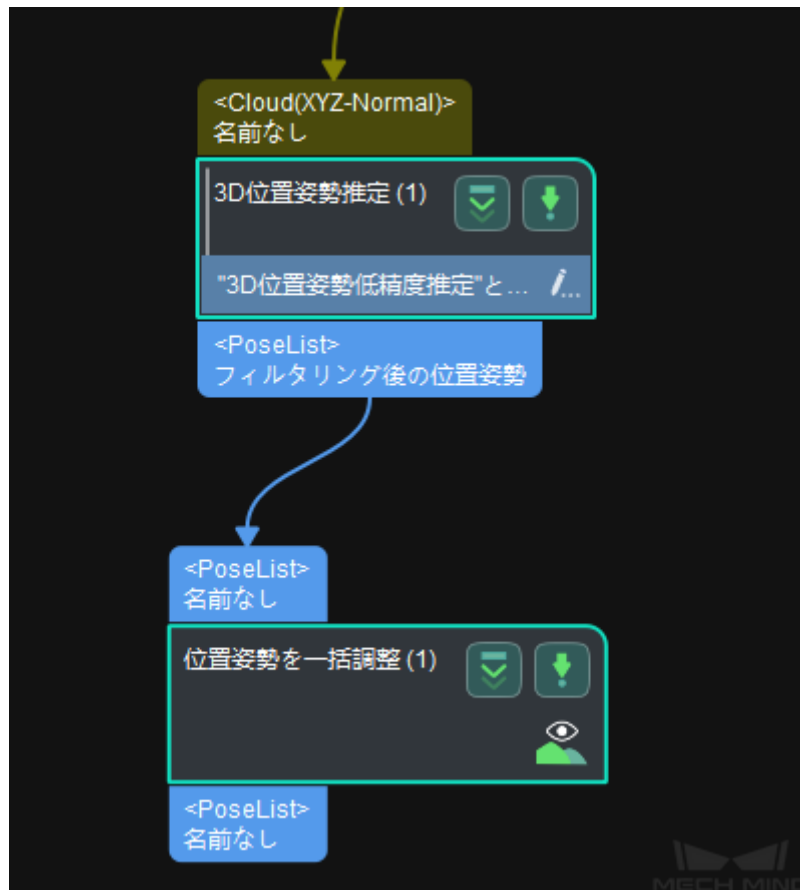
- ステップ `remove_overlapped_objects` ステップを使用して設定されたルールに従って積み重ねられた対象物の位置姿勢を除去します。

重複率のしきい値を調整することで、重複判断の厳しさを調整することができます。重複率のしきい値が低いほど判断が厳しくなります。

## 位置姿勢を調整

その後、座標系の変換を行います。ステップ `3d_fine_matching` により正確な位置姿勢を取得したら、位置姿勢の座標系を変換します。

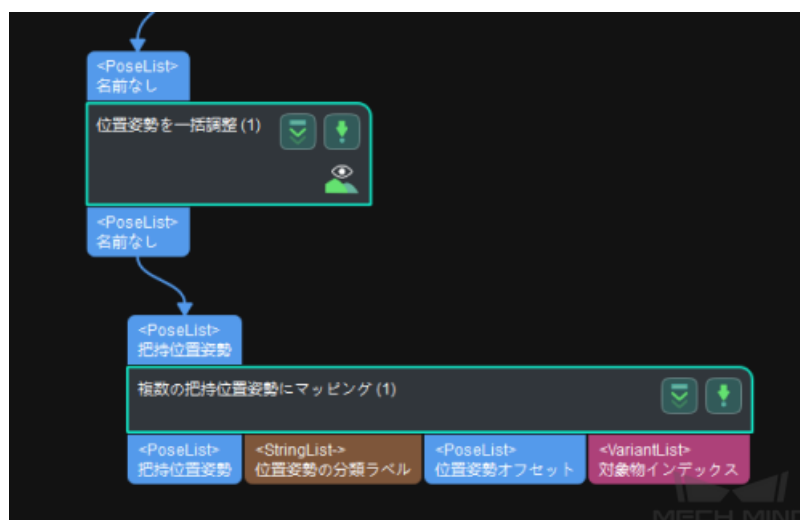
ステップ `pose_adjustment_collection` を追加し、`3d_matching_custom` ステップの組合せの後に配置します。このステップに組み込んでいる `pose_editor` を使用することにより、位置姿勢の座標系変換、並進、回転、ソートなどの組み合わせた処理を実現できます。



### 複数の把持位置姿勢にマッピング

把持点が幾何学的中心点と一致しない場合、または複数の把持点がある場合は、`map_to_multi_pick_points` ステップを使用して、ロボット制御ソフトウェアに送信可能な把持点をシーン内の対象物に追加する必要があります。

このステップを追加し、`pose_adjustment_collection` の後に配置します。



## ビジョン結果を出力

最後にデータの送信を行います。ステップ procedure\_out を追加します。このステップにより、データを Mech-Center に送信して、後で Mech-Center からデータを使用するソフトウェアとハードウェアに送信します。このステップを map\_to\_multi\_pick\_points の後に配置します。

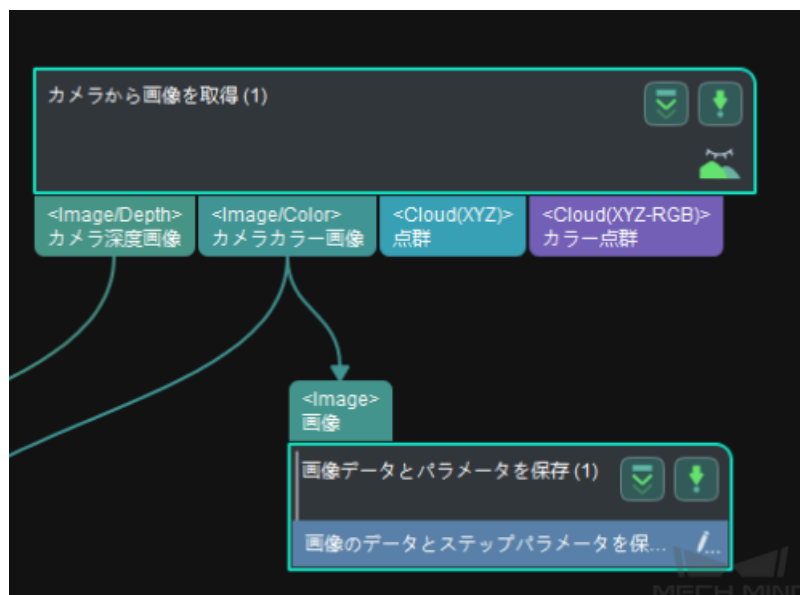


これで、クランクシャフト供給プロジェクトの構築は完了しました。メニューバーで **ファイル・プロジェクトを保存** をクリックするか、ショートカットキー **Ctrl + S** を押してプロジェクトを保存することができます。

## プロジェクトを実行してデバッグ

プロジェクトの構築が完了したら、プロジェクトを実行してデバッグすることができます。

プロジェクトを実行する際に画像データおよび設定されたステップパラメータを保存する必要がある場合、capture\_images\_from\_camera ステップの後に save\_images\_and\_step\_properties のステップの組合せをつなぎます。



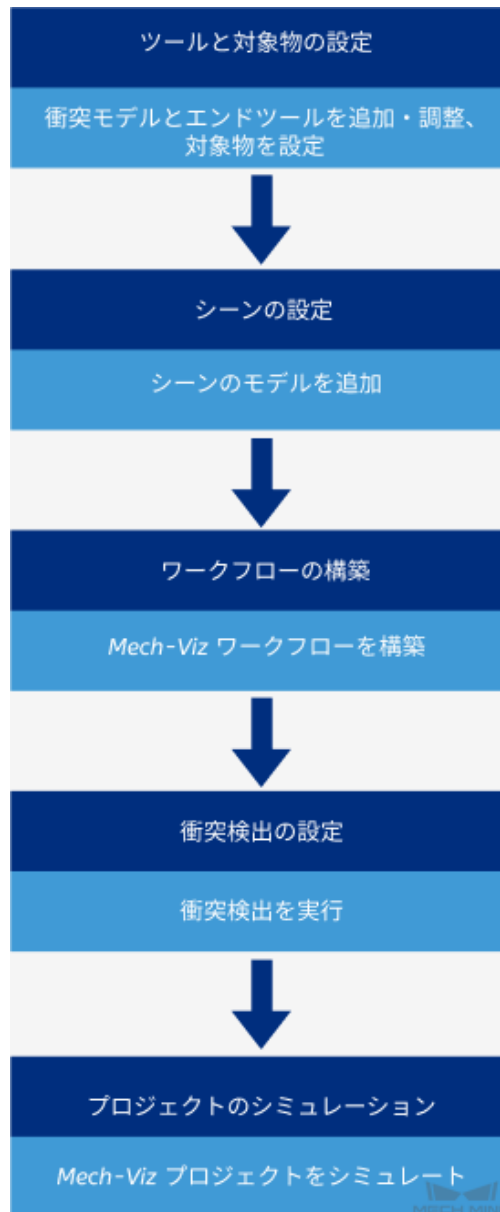
## Mech-Viz プロジェクトを構築

本節では、クランクシャフト供給の Mech-Viz プロジェクトの構築について説明します。

- プロジェクトの構築手順
- ツールと対象物の設定
  - 衝突モデルの追加と調整
  - エンドツールの追加
  - TCP の調整
  - 対象物の設定
- シーンの設定
- ワークフローの構築
  - 「初期位置」の設定
  - 「写真撮影点」の追加
  - 「把持点」の設定
  - 「把持待ち」の設定
  - 把持動作の「相対移動点」の設定
  - 「配置点」の設定
  - 「配置待ち」の設定
  - 配置動作の「相対移動点」の設定
  - 「初期位置」に戻る
  - シミュレーション
- 衝突検出設定
- 実行のシミュレーション

### プロジェクトの構築手順

今回作成するプロジェクトは以下のような流れで処理を行います。



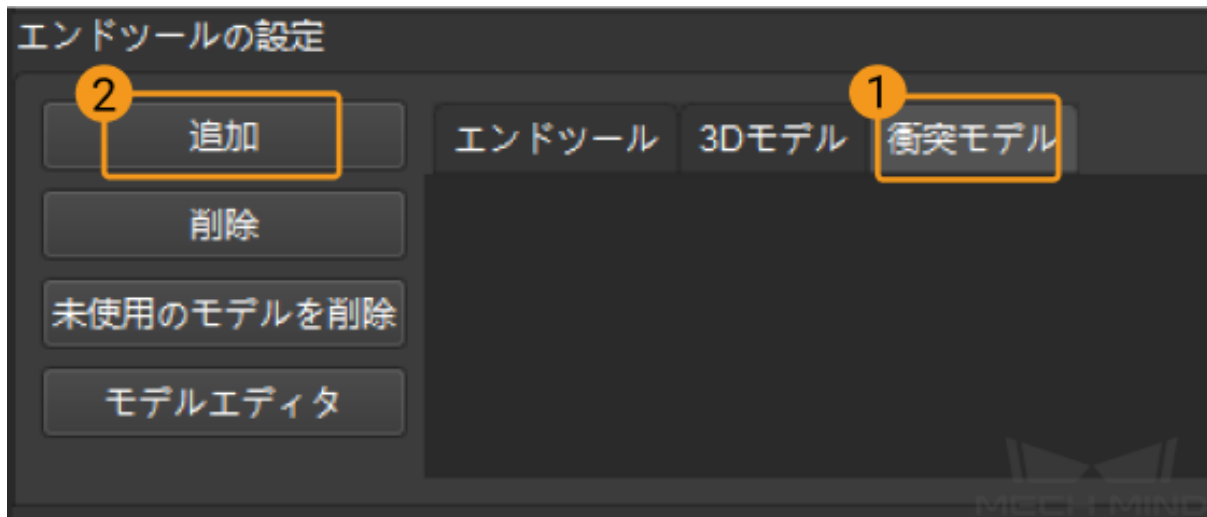
1. ツールとワークの設定。3D シミュレーションエリアにツールモデルを表示し、かつエンドツールの衝突検出を有効にするために、ツールを設定する必要があります。また、ロボットがワークを把持する際の運動経路を計画するためにワークの設定も必要です。
2. シーンの設定。シーンを設定して実際の動作環境を再現することでロボット運動経路の計画に役立ちます。
3. ワークフローの構築。異なるタスクやパラメータを使用してプロジェクトを構築します。
4. 衝突検出設定。ロボットが箱もしくははその他障害物との衝突を防止し、プログラムの連続的な実行を保証するため、衝突検出設定を行う必要があります。
5. プロジェクトのシミュレーション。上記すべての設定が完了後、プロジェクトの実行結果を確認するためにプロジェクトをシミュレートします。

## ツールと対象物の設定

ロボットモデルを設定した後、ツールとワークの設定を行います。衝突モデルとツールを追加し、TCPを調整します。

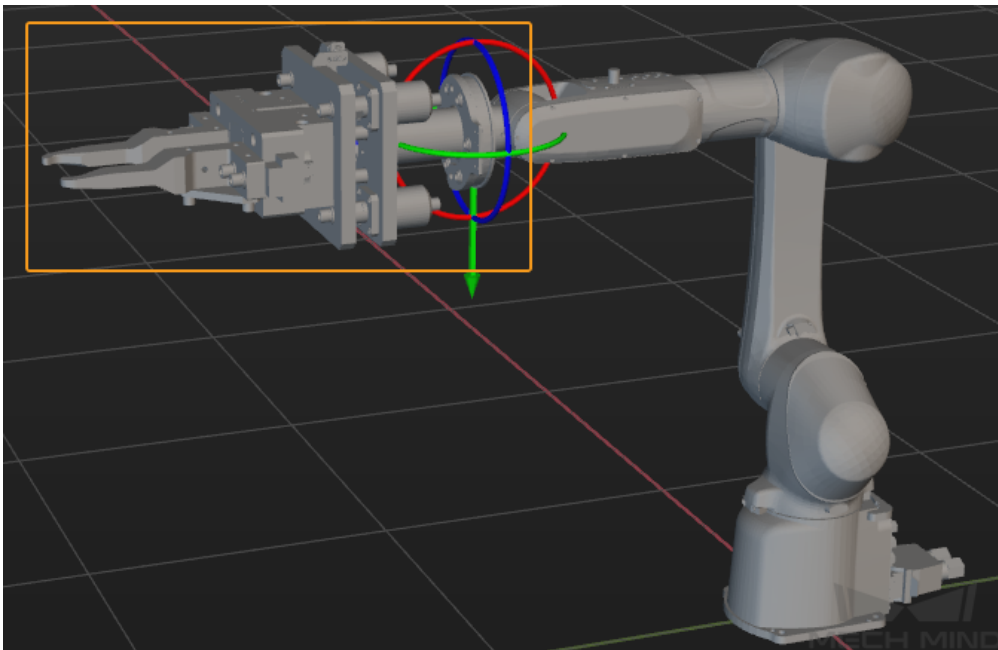
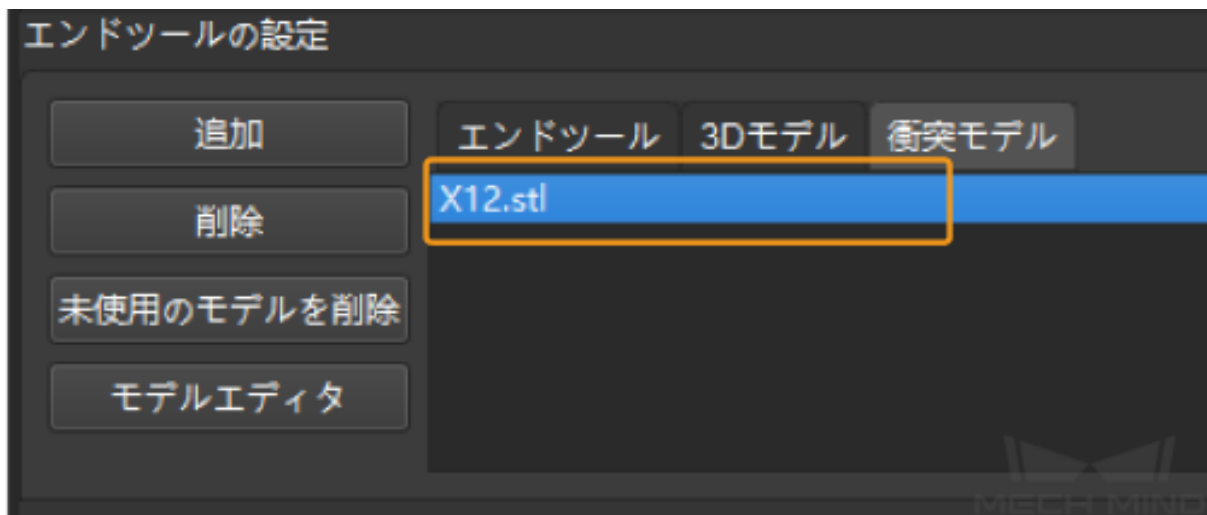
### 衝突モデルの追加と調整

1. ツールと対象物のタブで **衝突モデル・追加** をクリックし、ポップアップするファイル選択ウィンドウで衝突モデルを選択します。



**注意:** 衝突モデルについては、ツールとシーンのモデルの簡略化をご参照ください。

2. 追加した衝突モデルを選択したら、それがプロジェクト編集エリアに表示されます。



上図では、衝突モデルが正しい位置にあります。衝突モデルが間違った位置にあり、調整が必要な場合は、衝突モデルの位置を調整してください。

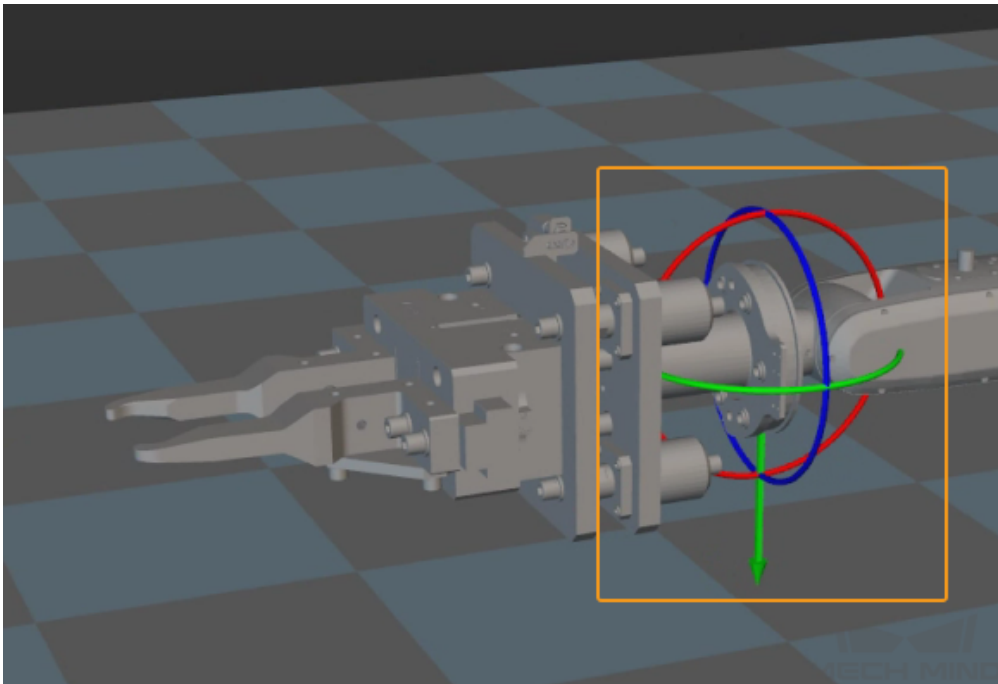


## エンドツールの追加

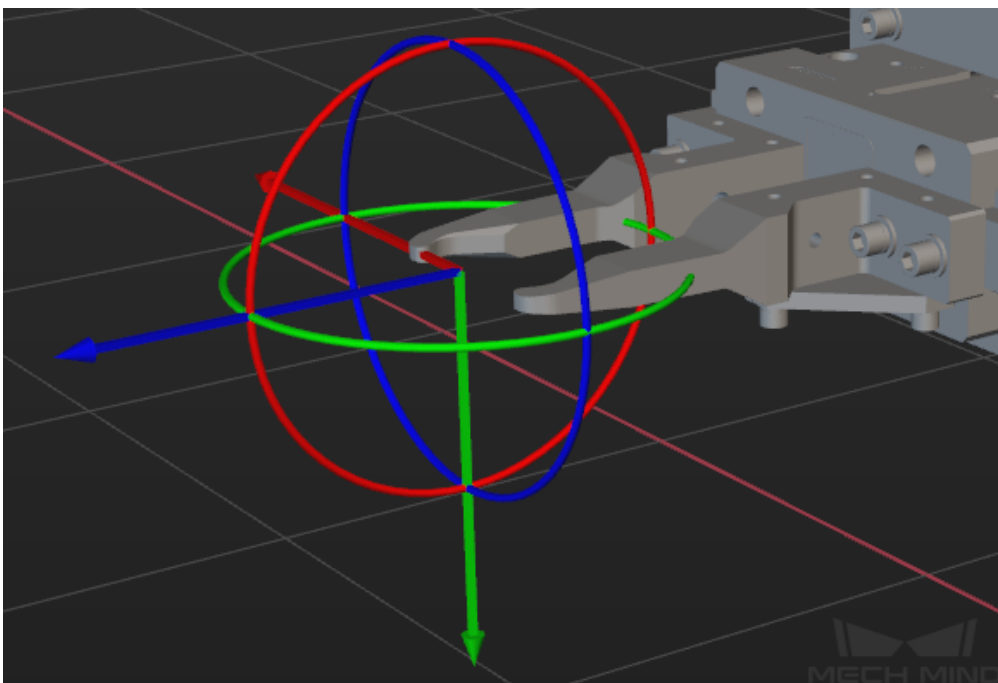
操作説明	サンプル図
1-2 エンドツール をクリックした後、追加をクリックします。	
3-4 エンドツール名を入力し (例として「tcp」と名付ける)、追加した衝突モデルを選択します。	
5 OK をクリックし、設定を保存します。	
設定完了後、エンドツールリストに追加されたモデル名が表示されます。	

## TCP の調整

下図に示すように、TCP はドラッガー（ロボットのエンドツールの座標球）の中心点によって表示されます。



プロジェクトでは、ツールの中心点はロボット先端にある必要があります。調整後、ドラッガー中心点の位置は下図のようになります。



## 対象物の設定

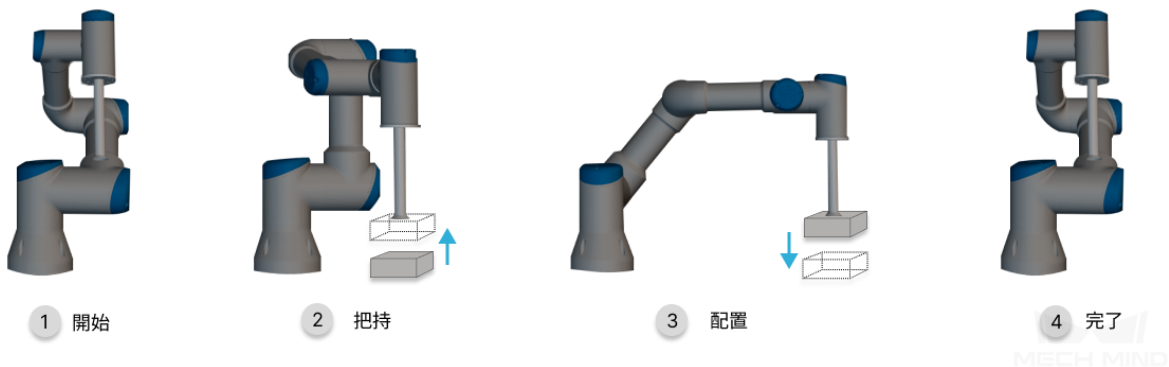


- 回転対称：クランクシャフトには回転対称性がないため、**X/Y 軸に関して対称性がない** を選択します。
- 把持範囲：クランクシャフトの表面が曲面であるため、把持範囲を設定する必要があります。**X 軸範囲** にチェックを入れ、X 軸のステップサイズを **3°** に、試行範囲を **±30°** に設定します。
- 最適な把持の解選択戦略：クランクシャフト把持プロジェクトでは、**デフォルト** または **全体回転角度の最小値** を選択すれば、ロボットがクランクシャフトを把持した後の無駄な回転に起因するクランクシャフトの落下を回避することができます。

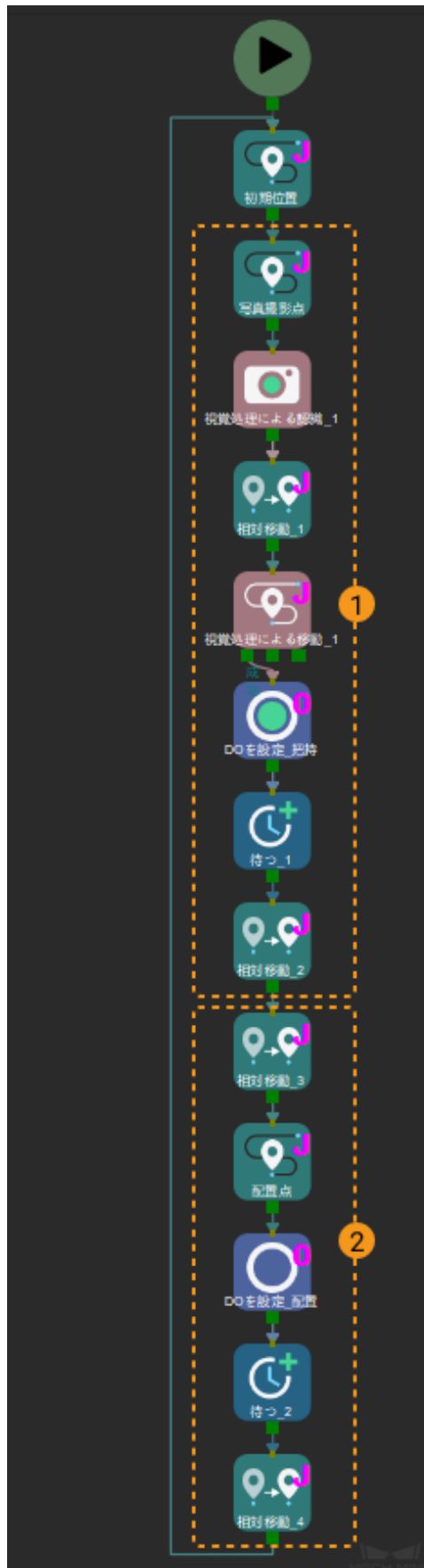
## シーンの設定

ロボット運動経路を計画するためにシーンを設定する場合、地面 および シーンの物体 を参照し、実際の動作環境を再現します。

## ワークフローの構築



これは Mech-Viz プロジェクトのコア部分です。タスクライブラリ内のタスクをプロジェクト編集エリアにドラックしてからタスクの各パラメータを設定し、タスクを線でつなぐことでプロジェクトを構築します。下図のようなワークフローが完成形です。



## 「初期位置」の設定

初期位置はロボットが動作を開始する位置であり、安全な位置を設定します。ロボットがこの場所にいる時はカメラの視界を遮ることなく、把持されるワークや周辺機器から遠ざける必要があります。

**移動** タスクを使用することで、ロボットの運動経路にあるターゲット位置姿勢およびターゲット位置姿勢への移動方法を設定することができます。初期位置を設定するためには、**移動** タスクを使用します。

1. ツールバーの **ロボットを同期させる** をクリックすると、ロボットモデルがロボット実機の状態に同期されます。
2. ティーチペンダントを使用して、ロボット実機をカスタマイズされた初期位置に移動し、ロボットモデルも対応する位置に到達したことを確認します。
3. ステップライブラリから **基本的な移動・移動** をプロジェクト編集エリアにドラッグします。この移動ステップはロボットの現在の位置姿勢を記録します。
4. このステップの名前を「初期位置」に変更します。



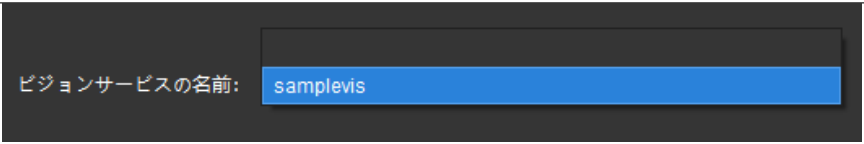
## 「写真撮影点」の追加

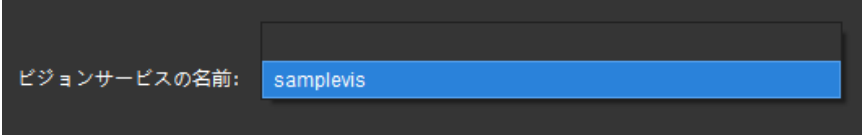
「初期位置」と「把持点」の間に **写真撮影点** を追加します。

1. ティーチペンダントでロボット実機をカスタマイズされた写真撮影点に移動させたら、現時点でのロボットの位置姿勢を記録するための1つの **移動** タスクをプロジェクト編集エリアにドラッグします。
2. ロボットによってカメラの視野が遮られないようにしてください。Mech-Eye Viewer で画像を取得し、視野が遮られているかどうかを確認できます。
3. 名前を「写真撮影点」に変更します。
4. 「写真撮影点」の入力を「初期位置」からつなぎ、出力を「視覚処理による認識\_1」につなぎます。

## 「把持点」の設定

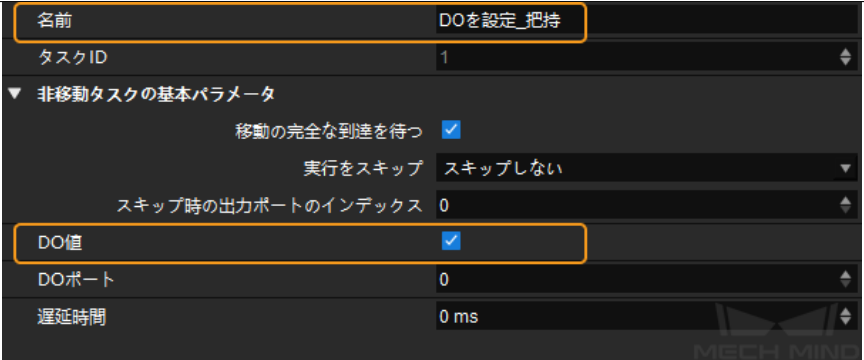
「把持点」は、把持するワークの位置姿勢なので、**視覚処理による認識** と **視覚処理による移動** タスクを使用します。

タスク	視覚処理による認識
説明	Mech-Vision プロジェクトをトリガーし、ビジョン結果を取得します。
操作	タスクライブラリから <b>視覚・視覚処理による認識</b> をプロジェクト編集エリアにドラッグします。次に「初期位置」の出力を「視覚処理による認識」の入力につなぎます。
パラメータ設定	<b>ビジョンサービスの名前</b> をクリックし、ドロップダウンリストで作成した Mech-Vision プロジェクトを選択します。
サンプル図	

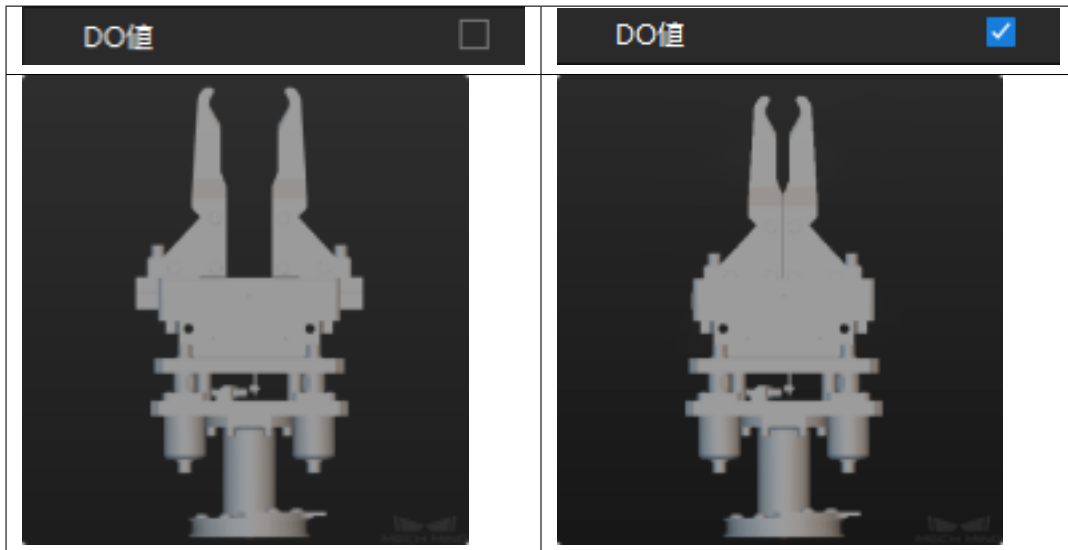
タスク	視覚処理による移動
説明	ビジョン結果によって移動します。
操作	タスクライブラリから <b>視覚・視覚処理による移動</b> をプロジェクト編集エリアにドラッグします。次に「視覚処理による認識」の出力「視覚処理による移動」の入力につなぎます。
パラメータ設定	<b>ビジョンサービスの名前</b> をクリックし、ドロップダウンリストで作成した Mech-Vision プロジェクトを選択します。
サンプル図	

ロボットが把持点に到達したら、治具を開いてワークを把持する必要があります。

**DO を設定** タスクでツールを制御します。

タスク	DO を設定
説明	ロボットポート信号を指定して、ツールの開閉を制御します。
操作	タスクライブラリから <b>DI/DO・DO を設定</b> をプロジェクト編集エリアにドラッグします。次に「視覚処理による移動」の出力を「DO を設定」の入力につなぎます。
パラメータ設定	名前を「DO を設定_把持」に変更し、タスクのパラメータで「DO」にチェックを入れます。チェックを入れると、ソフトウェアは実行中に制御信号を送信してツールが把持可能な状態になります。
サンプル図	

DO 値と実際のエンドツールの開閉の対応関係は次の通りです。



### 「把持待ち」の設定

ツールがワークを把持する際に、ワークをしっかり把持するまで待機時間を設定します。

タスク	待つ
目的	<b>待つ</b> タスクを使用して、ロボットがこのタスクを実行したら指定時間を待機します（ミリ秒単位）。
操作	タスクライブラリから <b>ツール・待つ</b> をプロジェクト編集エリアにドラッグします。パラメータは初期値を使用します。
つながり	「待つ_1」の入力を「DOを設定_把持」につなぎ、出力を「相対移動_2」につなぎます。

### 把持動作の「相対移動点」の設定

ツールが把持点まで垂直に下向きに移動し、物体を吸着した後、垂直に上向きに移動することが理想的です。この動作を実現するために、**相対移動** タスクを追加します。

1. ツールは把持点から Z 方向に 200mm から垂直下向きに移動します。

タスク	相対移動
操作	タスクライブラリから <b>基本的な移動・相対移動</b> をプロジェクト編集エリアにドラッグします。すると、タスク名は「相対移動_1」になります。
つながぎ	「相対移動_1」の入力を「視覚処理による認識_1」につなぎ、出力を「視覚処理による移動_1」につなぎます。
パラメータ設定	「基準とする位置姿勢」を <b>次のタスク</b> に設定し、座標系の Z 値を-200 に設定します。
サンプル図	

2. ツールがワークを吸着した後、垂直に上向きに 200mm 移動します。

タスク	相対移動
操作	タスクライブラリから <b>基本的な移動・相対移動</b> をプロジェクト編集エリアにドラッグします。すると、タスク名は「相対移動_2」になります。
つながぎ	「相対移動_2」の入力を「DOを設定_把持」につなぎます。
パラメータ設定	「基準とする位置姿勢」を <b>現在のタスク</b> に設定し、座標系の Z 値を-200 に設定します。
サンプル図	

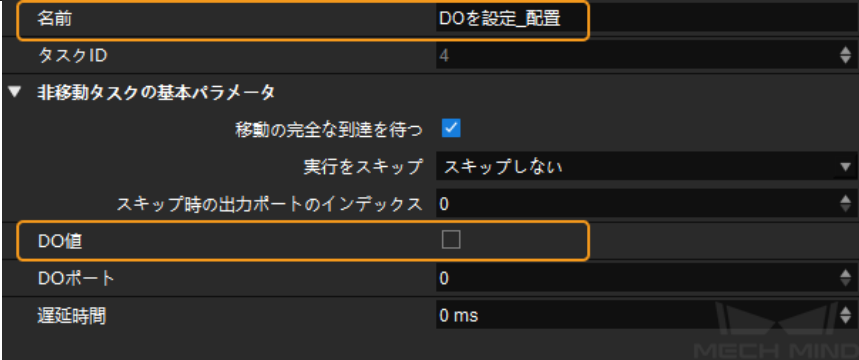


## 「配置点」の設定

ロボットがワークを把持した後、ユーザーに設定された位置姿勢に移動してワークを配置します。

1. ティーチペンダントでロボット実機をカスタマイズされた配置点に移動し、ロボットを同期させた後に1つの**移動**タスクをプロジェクト編集エリアにドラッグして、現在の位置姿勢を記録します。その名前を「配置点」に変更します。
2. 次に「DOを設定\_把持」の出力を「配置点」の入力につなぎます。

ロボットが配置点に到達したら、ツールを閉じてワークを吸着解除します。**DOを設定**タスクでツールを制御します。

タスク	DOを設定
説明	ロボットポート信号を指定して、ツールの開閉を制御します。
操作	タスクライブラリから <b>DI/DO・DOを設定</b> をプロジェクト編集エリアにドラッグします。次に「配置点」の出力を「DOを設定」の入力につなぎます。
パラメータ設定	名前を「DOを設定_配置」に変更し、タスクのパラメータで「DO値」のチェックを外します。チェックを外すと、ソフトウェアは実行中に制御信号を送信してツールが把持解除の状態になります。
サンプル図	

## 「配置待ち」の設定

ツールがワークを配置した後、ワークが完全に放されるまで待ちます。

上記を参照して、「待つ\_2」の入力を「DOを設定\_配置」につなぎ、出力を「相対移動\_4」につなぎます。

## 配置動作の「相対移動点」の設定

把持動作の「相対移動点」の設定を参照し、配置点の前後に「相対移動」タスクを一つずつ追加します。

1. ツールが配置点のZ方向に200mmから垂直に下向きに移動します。

操作	タスクライブラリから <b>基本的な移動・相対移動</b> をプロジェクト編集エリアにドラッグします。すると、タスク名は「相対移動_3」になります。
つなぎ	「相対移動_3」の入力を「相対移動_2」につなぎ、出力を「配置点」につなぎます。
パラメータ設定	「基準とする位置姿勢」を <b>次のタスク</b> に設定し、座標系のZ値を-200に設定します。

2. ツールがワークを吸着解除した後、垂直に上向きに200mmに移動します。

操作	タスクライブラリから <b>基本的な移動・相対移動</b> をプロジェクト編集エリアにドラッグします。すると、タスク名は「相対移動_4」になります。
つながぎ	「相対移動_4」の入力を「DOを設定_配置」の出力につなぎ、出力を「初期位置」につなぎます。
パラメータ設定	「基準とする位置姿勢」を <b>現在のタスク</b> に設定し、座標系の Z 値を-200に設定します。

### 「初期位置」に戻る

ワークを配置した後、ロボットを初期位置に戻す必要があります。

ループを増やすことで、ロボットを初期位置に戻し、タスクを繰り返し実行させることができます。



---

**ヒント:** プロジェクト編集エリアで右クリックし、ポップアップメニューから **自動レイアウト** をクリックすれば、タスクが自動的に整列され、インターフェイスをよりきれいにします。

---

## シミュレーション

**シミュレーション** をクリックし、ロボット動作のシミュレーションを行います。

### 衝突検出設定

ロボットが箱もしくはその他障害物との衝突を回避し、プログラムをスムーズに実行するために、衝突検出の設定を行う必要があります。詳細は、**衝突検出** をご参照ください。

### 実行のシミュレーション

上記の操作を完了した後、**実行** をクリックし、Mech-Viz プロジェクトのシミュレーションを実行します。

**注意:** Mech-Vision と Mech-Viz が通信を行い、Mech-Vision によって取得されたシーンの点群を Mech-Viz に送信します。そのために、Mech-Viz プロジェクトを実行する前に Mech-Center を起動する必要があります。

## Mech-Vision プロジェクトを構築

本事例では、Mech-Vision プロジェクトと Mech-Viz プロジェクトを併用する必要があります。まずは Mech-Vision プロジェクトを作成し、画像の取得と処理、把持位置姿勢の出力などを完了します。部品のエッジに明らかな特徴があるので、Mech-Vision プロジェクトでは点群モデルを使用すれば高い認識精度を実現できます。

詳細については、**Mech-Vision プロジェクトを構築** をご参照ください。

## Mech-Viz プロジェクトを構築

Mech-Vision プロジェクトから送信されたシーンの点群に基づいて Mech-Viz でワークフローを作成し、経路計画と衝突検出を行ってから、ロボットを制御してクランクシャフトを把持します。ロボットが把持する際にシーンの物体やカメラ、箱と衝突しないように Mech-Viz ワークフローを作成します。

詳細については、**Mech-Viz プロジェクトを構築** をご参照ください。

---

**注釈:** シーンの点群とは、カメラ視野内にあるすべての点群のことを指します。

---

### 2.1.3 注意事項

現時点では、特に注意すべきことはありません。

## アプリケーションの最適化

本節では、システム構築プロセス中のハードウェアの選択、設計、およびソフトウェアとハードウェアの最適化に関するいくつかのアドバイスを提供します。

### 3.1 カメラ選択

#### 3.1.1 カメラ選択の原則

1. 見える：対象物の点群のイメージングがはっきりして、欠如やノイズはなく、認識された特徴を完全に表示できます。
2. 完全に見える：視野は十分で、認識されるエリアはカバーします。
3. 精確に見える：点群は精度が高く、誤差が小さいです。

#### 3.1.2 カメラ選択の説明

##### モノクロカメラ/カラーカメラを選択

段ボール箱のデパレタイジングのプロジェクトにディープラーニングを使用するなど、カラー画像が必要な場合は、カラーカメラを使用します。それ以外の場合は、デフォルトでモノクロカメラが選択されます。

##### 稼働距離を確認

被写界深度の狭い応用シーン：稼働距離が固定されています。例としては、フラット接着、フラット把持、浅いボックスの把持などがあります。ソリューションの設計または測定距離が  $D$  の場合、 $D$  値に従って最も良い利用可能な型番を確認します。 $D$  値が利用可能な型番と大幅に異なる場合、またはプロジェクトのフォーカス要件が非常に高い場合は、要件について個別に説明できます。

被写界深度の広い応用シーン：稼働距離には広い範囲があります。例としては、デパレタイジング、深いボックスの把持などがあります。ソリューションの設計または測定距離が  $D1$ ~ $D2$  ( $D1 < D2$ ) の場合、 $D2$  値に従って最も良い利用可能な型番を確認します。 $D2$  値が利用可能な型番と大幅に異なる場合、またはプロジェクトのフォーカス要件が非常に高い場合は、要件について個別に説明できます。

カメラ型番と仕様の詳しい内容は、[基本的な仕様](#) をご参照ください。

## 3.2 照明制御

構造化光カメラの3D点群のイメージングとディープラーニングの場合、照明環境は視覚認識の効果と安定性に影響を与えます。

3D点群のイメージングの場合、照明が均一で対象物の表面輝度が低いと、点群の品質が向上する可能性があります。実際の使用シナリオでは、金属部品の最大輝度は600luxを超えないことをお勧めします。箱と麻袋の表面輝度は、800luxを超えないことをお勧めします。他のシーンは、理想的な点群品質を確保するために500lux未満である必要があります。暗い部屋に均一な表面光源を追加することは、最も理想的で安定した輝度制御方法です。この環境では、ワーク表面の輝度が安定しており、明るさや暗さの変化はありません。暗室以外の環境では、3D点群イメージングの品質を向上させるために、遮光・補光のソリューションを適切に使用することをお勧めします。

ディープラーニングの場合、安定した照明環境、ワークのはっきりと見える表面、安定した明るさが理想的な条件です。実際の環境では、環境光は時間とともに変化し、ディープラーニングの認識結果に影響を与える可能性があります。遮光・補光のソリューションを適切に使用することで、環境光の変化を効果的に制御し、ディープラーニング認識の結果を安定させることができます。

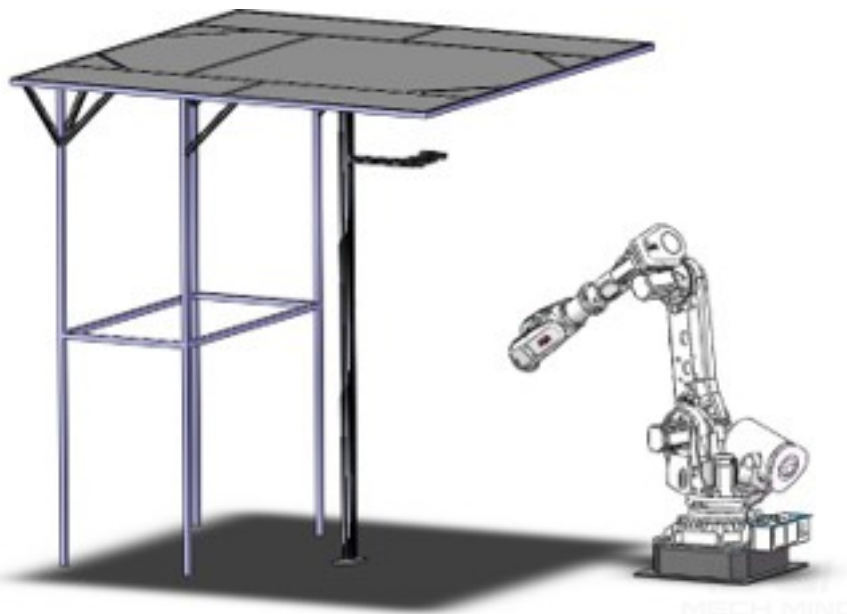
---

**ちなみに:** 光源には、色温度 6000k の通常の白色 LED ライトが一般的に使用されています。

---

実際の工場現場で、以下のポイントをチェックして確認してください。

1. **ホワイトバランス:** 2D 画像の色は、肉眼で観察した画像と同じである必要があります。それ以外の場合は、カメラのホワイトバランスを調整する必要があります。
2. **光の強さ:** 直射日光やその他の強い赤外線光源があってはなりません。
3. **反射:** 現場で均一で柔らかな照明が必要で、ワークの表面に直接当たる強い光や複雑な光は避けます。
4. **安定性:** 昼と夜の光の強さの大きな違いを避けるために、現場の照明は安定している必要があります。
5. 上記を満たせない場合は、遮光と補光を考慮する必要があります。遮光ソリューションは以下のとおりです。



### 3.3 治具の設計

ロボットの治具（エンドエフェクタ）を設計するときは、以下の要件を満たしてください。

1. 治具の構造は、すべての製品に適用できるかどうかを確認してください。特に、SKU の数が多い仕分けシナリオの場合、治具の対称性とオフセットを考慮する必要があります。
2. 治具の吸引力、把持力が十分であり、すべての製品の安定した把持できるかどうかを確認してください。
3. 吸盤の治具は、箱などの製品の特徴に応じて真空検出センサーを装備し、箱が把持されているかどうかを判断できます。治具にバッファツールを使用することをお勧めします。これは、ロボットや製品に対して優れた保護効果をもたらします。
4. ロボット移動時の配線レイアウトは、ぶら下がったり壊れたりしないようにする必要があります。
5. ロボットの作業位置で特異点が発生した場合は、オフセットを含む治具を設計して、ロボットが特異点に対する許容範囲を拡大させます。また、45°、30° などの一般的な角度を選択してください。
6. 治具の設計では、ロボットの作業スペースを考慮する必要があります。ロボット作業の要件を満たしている場合、治具の長さは長すぎるまたは短すぎることを避ける必要があります。
7. 後続のデバッグのために、設計時に治具の 3D モデルを保持してください。

---

**ちなみに：** 治具の 3D モデルは、Mech-Viz ソフトウェアが衝突を正確に検出するのに役立ちます。

---

### 3.4 電源制御

Mech-Mind のビジョンシステムには、主な電気機器はロボット、Mech-Eye 3D カメラ、ビジョンシステム産業用コンピュータ、補助光源、空気源、治具、さまざまなセンサー、信号灯などがあります。対応する電源要件は以下のとおりです。

**ロボット：**ほとんどのロボットは強力な動作電源電圧範囲を備えており、200 - 600V 以上の電圧範囲で動作します。少数のロボットは、220V 前後の電圧でしか動作できません。

**カメラ：**Mech-Eye 3D カメラが継続的に動作している場合、消費電力は約 120W、電圧は 220V です。電源の安定性と一定の干渉防止能力を確保するために、Mech-Mind 製のカメラケーブルを使用しなければなりません。工場の電源システムが不安定な場合、電圧の安定性を確保するために電圧安定器が必要です。

**産業用コンピュータ：**ビジョンシステムの産業用コンピュータの消費電力は約 180W、電圧は 220V です。

**光源：**光源ドライバーの消費電力は約 60W、電圧は 220V です。3 芯の電源コードで駆動され、明るさが調整できる 6000k の白色光源です。

**空気源：**空気圧縮機の消費電力は一般的に大きく、一般的な消費電力は約 1000W、電圧は 220V です。

**配線：**すべてのワイヤーは分類してラベルを付ける必要があります。ワイヤーの露出を防ぎ、ペリフェラル AGV、フォークリフトなどとの干渉を防ぐために、オンラインスロットで配置および標準化する必要があります。

電源システムが電圧やその他の問題に影響を与えるのを防ぐために、カメラと IPC に電力保護を提供する電圧安定器が必要です。現場の環境で頻繁に停電が発生する場合は、事故を防ぐために、システム全体の通常の動作を保護するために UPS 電源装置を追加することをお勧めします。ロボットにセンサーを接続する必要がある場合、ほとんどのロボット I/O ポートが提供する 12V 電源は負荷容量が弱いいため、センサーなどの小型電気機器を直接駆動するには適応していません。電源を供給するために、12V/5V DC 電源を構成することをお勧めします。



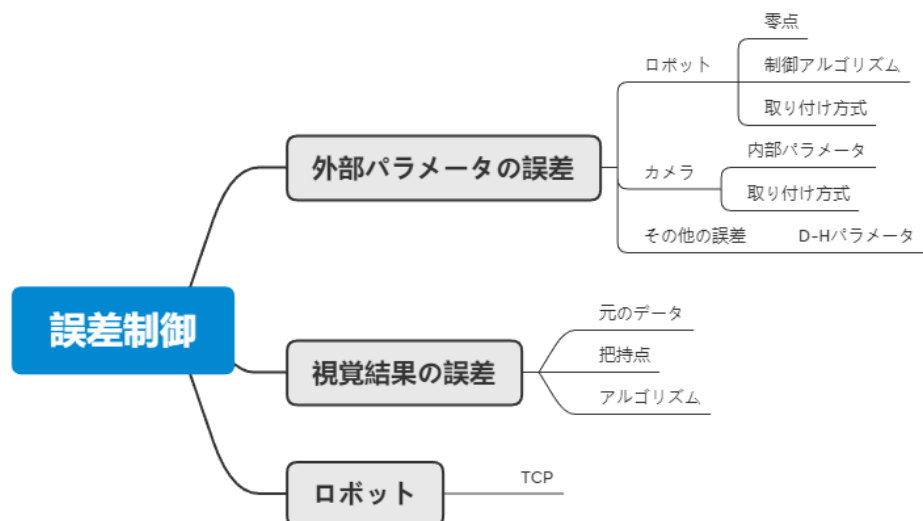
### 3.5 ロボットの選択

ロボットを選択する際に、以下の条件を満たすことができるかどうかを優先に考慮してください。

1. 特異点の問題を回避するために、作業に必要なすべての位置に到達できるかどうか、X、YおよびZ方向に十分なマージンがあるかどうかを確認してください。
2. 作業の最大負荷の要件を満たすことができるかどうかを確認してください。
3. 地面の耐荷重能力は、ロボットの動作 限界荷重 を満たす必要があります。
4. ロボットが動作する、またはカメラの外部パラメータをキャリブレーションする際に、ケーブル、トレイ、カメラスタンド、天井、照明ランプなどに干渉しないかどうかを確認してください。
5. 光源の高さが低く、カメラスタンドがトレイに近すぎ、ケーブルがロボットベースに近すぎ、ケーブル間の距離が近すぎることを確認してください。
6. 治具オフセット中に、ロボットの作業スペースが変化する可能性があることを考慮してください。
7. Eye in Hand モードの場合、カメラがフルスタック（デパレタイジング・パレタイジング）を撮影しているときに最高の撮影ポイントに到達できるかどうかを確認してください。
8. ロボットのベースの高さが実際の応用と一致するかどうかを確認してください。
9. 全体のレイアウトは、3D シミュレーションソフトウェアでシミュレーションして、作業プロセス全体にロボットと他のハードウェアの間に干渉なしであることを確認してください。

### 3.6 誤差制御

プロジェクトの実施において、最も一般的に発生する問題の1つは、把持誤差です。把持誤差とは、ロボットを制御してワークを把持する過程で、ロボットが予期される位置に到達していないが、一定の誤差があることを意味します。誤差の原因はさまざまです。以下では、外部パラメータ、ビジョン、ロボットから誤差を分析します。詳細な構造下図に示します。



### 3.6.1 外部パラメータの誤差

#### ロボット

#### ゼロ位置

**ゼロ位置：**ロボットは、閉ループサーボシステムを介してロボットモーターの各軸の動きを制御し、コントローラのコマンドにより各モーターを制御します。モーターに取り付けられたパルスエンコーダーは、モーターの信号をロボットにフィードバックすることができます。ロボットの動作中、コントローラはエンコーダから返信された信号を継続的に受信して分析し、動作プロセスの全体でロボットを正しく制御できるようにします。コントローラは、各軸ゼロ位置の位置に基づいてエンコーダから返信された情報を正確に分析するため、使用中にロボットの位置を変更することはできません。ロボットの通常動作中は、ロボットのゼロ位置位置と関節位置が保存され、コントローラの電源が正常にオフのときに内部バッテリーを使用して電力が維持されます。このデータは、ロボットが正常にシャットダウンするたびに保存されます。ロボットの電源がオンになるたびに、バックアップ内のデータが直接読み取られて認識され、電源がオフのときと同じ状態になります。ロボットのゼロ位置は、衝突、バッテリーの電力損失、または長期間使用されていないなどの条件下でオフセットまたは欠落する可能性があります。

ロボットがゼロ位置が欠落している状態で動作している場合、次のように表示されることがよくあります。

ロボットの各関節軸はソフト制限を超えて動作します（これは非常に危険です）。

ロボットは指定された位置に到達できません。ゼロ位置の位置が異なるため、同じ関節角度でロボットエンドの位置が異なります。

ロボットは、指定された方向に指定された長さを移動できません（例えば、1メートルを正しく移動できません）。

Mech-Vision を使用して外部パラメータをキャリブレーションする場合、キャリブレーションパラメータに基づいて複数のロボットが到達する位置姿勢を計算します。ロボットのゼロ位置が欠落すると、位置姿勢を正確に到達できないことが多く、最終的に外部パラメータの計算結果に誤差が生じます。

---

**ちなみに：**ロボットのゼロ位置が欠落しているかどうかを検出する方法はたくさんあります。最も一般的に使用される2つの検出方法は以下のとおりです。

- ティーチペンダントでロボットのすべての軸角度を「0°」に調整し、ロボットの各軸の銘板が位置合わせされているかどうかを確認します。位置合わせされていない場合、ロボットのゼロ位置が欠落することを示します。
- ロボットを制御して指定された方向に沿って特定の距離を移動させます。ロボットの移動距離が理論上の距離と一致しているかどうかを確認します。

ロボットのゼロ位置は、上記の方法で確認できます。ロボットゼロ位置のキャリブレーションは、異なるロボットゼロ位置のキャリブレーション方法に従って実行できます。

---

#### 制御アルゴリズム

新開発のロボットの中には、制御アルゴリズムに問題がある確率が低いものがあります。つまり、ロボットが指定された位置に移動できなく、偏差があります。ロボットが正確な位置に移動できない場合、ロボットは外部パラメータのキャリブレーションを実行できなくなります。

## 取り付け方式

ロボットの取り付け方式は、ロボットが長期間の動作中に安定性を維持できるかどうかを直接決定します。拡張ボルトや固定されていない大きなベースの使用などの弱い取り付け方式では、ロボットが安定した状態を維持したり、長期間の動作中に高速で動作し続けることを保証できません。ロボットが移動すると、実際にはロボットとカメラの位置関係が崩れ、ロボットが正確に把持できなくなります。このような状況は一般的ではなく、一度発生すると検出が困難になります。そのため、ロボットを取り付ける際には、ロボットの取り付け方式を確認する必要があります。

---

### ちなみに:

- 長期間の使用が必要な場合は、ケミカルボルトを使用して、ケミカルボルトをしっかりと固定してからロボット動作を行ってください。そうしないとボルトが緩んでしまいます。
- 展示会などの短期間の使用が必要な場合、ロボットは比較的安定した表面に固定できますが、ロボットの速度を特に高いレベルに調整することはできません。そうしないと、ロボットが緩む原因になります。

---

## カメラ

### 内部パラメータ

カメラの内部パラメータによって引き起こされる外部パラメータの問題は、主にカメラ視野のエッジで、またはカメラの動作距離を超えたときに発生します。工場出荷時のデフォルトでは、使用するハードウェアに応じたカメラの適切な動作距離が設定されました。カメラが動作距離またはエッジを超えて動作している場合、カメラはこの位置で正確にキャリブレーションしていないため、生成された点群に偏差が発生し、最終的に計算された外部パラメータにも偏差が発生します。

---

**ちなみに:** カメラを使用するときは、カメラの動作距離と、カメラの内部パラメータが異常であるかどうかを確認することをお勧めします。最後にカメラの動作距離の範囲でキャリブレーションします。

---

## 取り付け方式

取り付け方式によるカメラの外部パラメータの変化の理由は、ロボットの取り付け方式と同じです。取り付け方式の安定性は、セルを長期間使用できるかどうかを直接決定し、外部パラメータを安定に保ちます。カメラの取り付け方式については、**カメラの取り付け** をご参照ください。

---

**ちなみに:** また、大きな振動やカメラスタンドへの激しい衝突により、カメラスタンドが緩む場合があります。取り付けるときは、メラスタンドがぶつかったり寄りかかったりしないように特に注意してください。

---

## その他の誤差

ロボットの初期適応の時に、ロボットの運動学的関係を取得するために、主要なロボットのウェブサイトまたはマニュアルで、対応するロボットの動作パラメータ（以下では、D-H パラメータと呼ばれます）を見つける必要があります。ロボットの動作パラメータを取得すると、ロボットの関節軸間の関係を確立できます。実際のプロセスでは、ソフトウェアの D-H パラメータがロボット実機の D-H パラメータと一致しない場合があります。理由は 2 つあります。

- ロボットサブモデルの数が多すぎるため、認識エラー、または取得した D-H パラメータのは実際のロボットとの間に一定の誤差が発生します。
- マニュアルと実際のロボットの間に違いがあります。

D-H パラメータが不一致な場合、使用中にソフトウェア内のロボットエンドの位置がティーチペンダントと異なる可能性があるため、計算中に外部パラメータの偏差が発生します。

---

**ちなみに:** D-H パラメータが一致しているかどうかを確認します。ロボットに接続した後、Mech-Viz ソフトウェアで同じ関節角度の場合にロボットの 6 軸のエンドの姿勢がロボットエンドと一致しているかどうかを確認します。不一致な場合は、異常な D-H パラメータが原因である可能性があります。

---

## 3.6.2 視覚結果の誤差

### 元のデータ

**元のデータ:** 「元」とは、処理のために Mech-Vision に入力された元のデータを指します。データには、主に RGB 画像と深度画像が含まれます。

作業現場で照明（昼から夜など）が変更された場合、カメラの構成パラメータが時間内に更新されないため、以下が発生する可能性があります。

- 取得された点群のノイズが多すぎるか、ディテールが明確でないため、後続のマッチングアルゴリズムで続行できなくなり、または識別のエラーが発生します。
- 取得された RGB 画像の品質が変化し、暗い環境で取得された画像の輝度が不十分であるため、ディープランニングの結果がなく、または認識結果が良くないです。

---

**ちなみに:** 上記の状況が発生した場合は、Mech-Eye Viewer ソフトウェアでカメラのパラメータを更新し、2D と 3D 露出をそれぞれ調整して、RGB 画像と深度画像の結果を改善する必要があります。または、プロジェクトをデプロイするときに、適切な **照明制御** を使用します。

---

### 把持点

把持点の誤差は、把持誤差の中で一般的なエラーの 1 つです。Mech-Vision で把持点を設定する方法は 2 つあります。1 つは把持点をドラッグすること、もう 1 つはティーチング方法による把持点を設定することです。ロボットの精度が確保であるという条件では、ティーチング方法による把持点の誤差が小さいです。そのため、ここでは考慮せず、把持点をドラッグする方法を分析します。

把持点のドラッグは Mech-Vision ソフトウェアで実行する必要があり、把持点の位置はテンプレートに基づいて調整されます。把持点をドラッグすることは精度が不十分な方法であり、ロボットが到達する位置を正確に設定することができないため、実際の把持プロセスで理想的な位置に到達しない可能性があります。より正確な把持点が必要な場合は、ティーチング方法による把持点を取得する必要があります。

---

**ちなみに:** 把持点に誤差がある場合は、Mech-Viz ソフトウェアの点群の座標から、各ワークの把持点が理想的な位置にあるかどうかを確認できます。

---

## アルゴリズム

Mech-Vision ソフトウェアの出力結果に引き起こす偏差は、ここではまとめてアルゴリズム誤差と呼ばれます。視覚アルゴリズムのデプロイは、多くの場合、サイトで取得された元の画像に基づいて設定されます。照明や作業環境の変化により、元のパラメータが適用できなくなる可能性があります。Mech-Vision ソフトウェアでそれらを確認でき、ユーザーマニュアルに従って調整できます。また、特殊なワークの認識は時折誤差を引き起こす可能性があります。つまり、長期間の使用では問題がなく、環境の変化、特に軸対称対象物によって引き起こされる時折の認識エラーは、局所的な最適解になりがちであるため、認識エラーになります。

**注意:** 一部のサイトでは、テンプレートエラーが原因で誤差が発生する場合があります。つまり、STL ファイルを介して取得したテンプレートが、実際の把持する対象物と誤差があるため、テンプレートマッピングを実行したときに最適な結果を得ることができません。理由は以下のとおりです。

- STL ファイルは完成品のファイルですが、把持するファイルはラフ素材のファイルです。生産プロセスでは、後続の工程を確保するために機械加工の余裕があるため、認識する必要のあるワークがテンプレートファイルよりも大きくなります。

このとき、テンプレートのスケールを適切に調整できます。

### 3.6.3 ロボットの誤差

#### TCP

ロボットの TCP 誤差は、ロボットの実際のツールセンターと理論上のツールセンターの間の不一致として現れます。ほとんどの理由は、ロボットの TCP キャリブレーションが十分に正確でないことです。タッチポイントの移動、ロボットの姿勢変化が十分に大きくない、タッチポイントと一致しないなどの状況は、ロボット TCP の不正確さを引き起こす可能性があります。ソフトウェアの使用中に、Mech-Viz ソフトウェアでツールを作成し、正確な TCP 値を入力する必要があります。

---

**ちなみに:** TCP を確認するには、ロボットを使用して TCP をキャリブレーションし、キャリブレーションされた値を Mech-Viz に入力して、ロボット側にバックアップを残すことをお勧めします。

---

---

**ちなみに:** 高精度を必要としないプロジェクトの場合は、ソフトウェアで TCP 値を調整して、把持の精度を向上させます。

---

### 3.7 タクトと経路の最適化

**生産タクト：**生産タクトとは、1つのワークを処理するために必要な時間です。つまり、ロボットセルの作業効率に関するユーザー定義の年間出力の要件を指します。

現在、ほとんどのプロジェクトにはタクトの要件があり、一部のプロジェクトにはタクトの要件が非常に高くなっています（4秒または5秒以内）。タクト要件が高いプロジェクトの場合、セルが作業プロセス全体を完了した後、Mech-Mind ソフトウェアシステムは、ビジョンアルゴリズムの最適化とロボット経路の最適化を介してタクトを最適化します。タクトには、カメラの撮影時間、Mech-Vision の処理時間、Mech-Viz の計画からロボットの処理が完了するまでの時間が含まれます。したがって、図1に示すように、タクトと経路の最適化の全体プロセスは主に上記の3つの部分に分けられます。

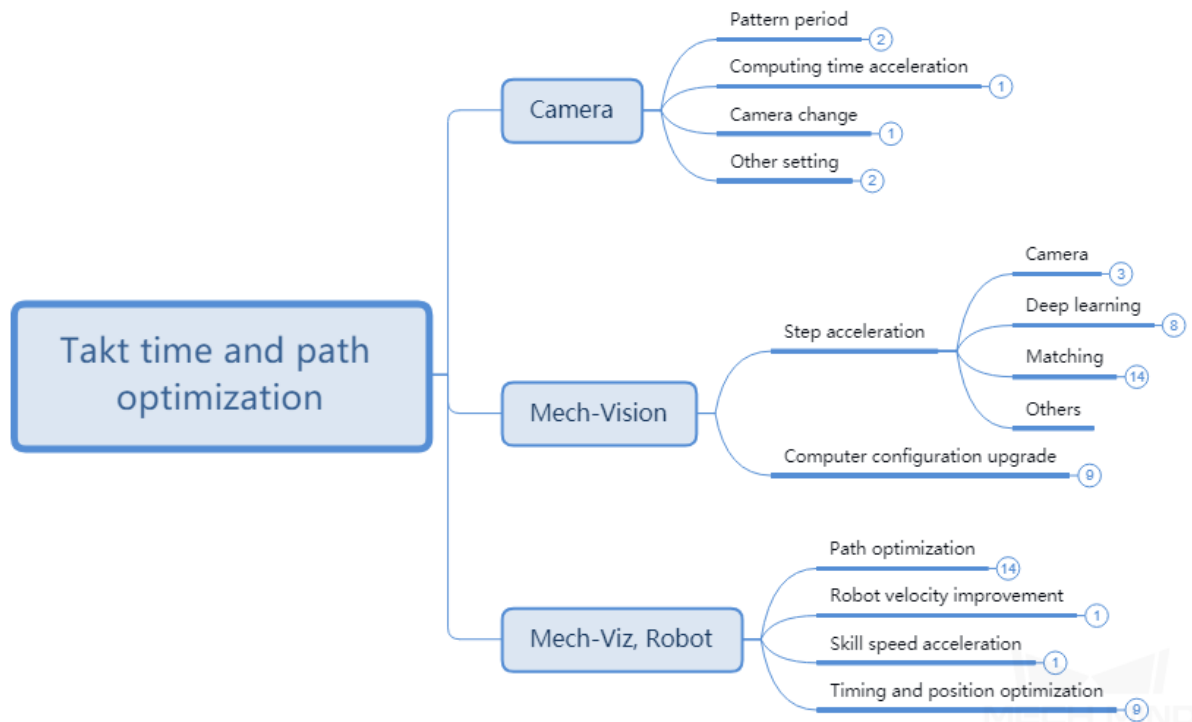


図1 経路とタクトの最適化の全体的なコンセプト

#### 3.7.1 Mech-Eye Viewer

投影時間

2D/3D 露出パラメータ

Mech-Eye Viewer ソフトウェアを使用する場合、複雑な構造の金属部品やディテール対象物の認識など、一部の応用シナリオでは、2D 露出モードを切り替えて、2~3 セットの異なる 3D 露出時間を使用してキャプチャし、より多く情報を取得します。また、段ボール箱や平らなワークなど、より多くのシーンでは、時間を節約するために露光回数を減らすことができることに注意してください。下図に示します。

2D/3D 露出パラメータの調整については、[パラメータ調整](#) をご参照ください。

## 更新

Mech-Eye Viewer ソフトウェアの最新バージョンを使用し、カメラファームウェアバージョンを更新すると、深度画像と点群画像の生成速度が向上し、カメラの撮影速度がある程度向上します。

## カメラを変更

Deep シリーズのカメラなどの Mech-Mind のカメラ製品は、視野が広いいため、2 台の光学機器で別々に投影し、取得した点群を合成する必要があります。そのため、通常の状態での投影時間は、ProM および ProS シリーズのカメラの 2 倍になります。シーンの作業範囲を確認した後、適切なカメラを使用すると、写真撮影の速度を向上させます。

## 深度画像の ROI

深度画像の ROI は、カメラの視野内でエリアを分割し、カメラが処理する必要のあるデータの量を削減するために使用されます。深度画像の ROI は、XY 方向の ROI と深度の制限に別々に設定して、全体として点群の数を減らすことができます。

XY 方向の ROI 設定は、図 2 に示します。設定インターフェースに入り、ROI の範囲はフレームを作成することによって選択されます。



図 2 深度画像の ROI (XY) を設定

Z 方向の制限は、カメラパラメータで設定する必要があります。最小深度と最大深度を別々に設定すると、点群の数を効果的に減らすことができます。図中の数値の単位はミリメートルです。

深度測定範囲	
下限	800 mm
上限	1100 mm

図 3 深度画像の ROI (Z) を設定

### 3.7.2 Mech-Vision

ステップの実行時間を加速

カメラから画像を取得



図4 カメラから画像を取得

`capture_images_from_camera` は、Mech-Vision プロジェクトで最も重要なステップであり、ほとんどのプロジェクトで約 40% の時間がかかります。`capture_images_from_camera` することがかかる時間は、主に上記の方法で最適化できます。一方、カメラは RGB 画像と深度画像を取得し、それらを組み合わせて点群画像を生成するため、`capture_images_from_camera` する機能を分割することもできます。プロジェクトによっては、点群と RGB 画像を別々に処理する必要がある場合があります。ディープラーニングが必要なプロジェクトの場合は、別々に処理できます。図 5 に示すように、プロジェクトは並行して計算されるため、多くの時間を節約できます。

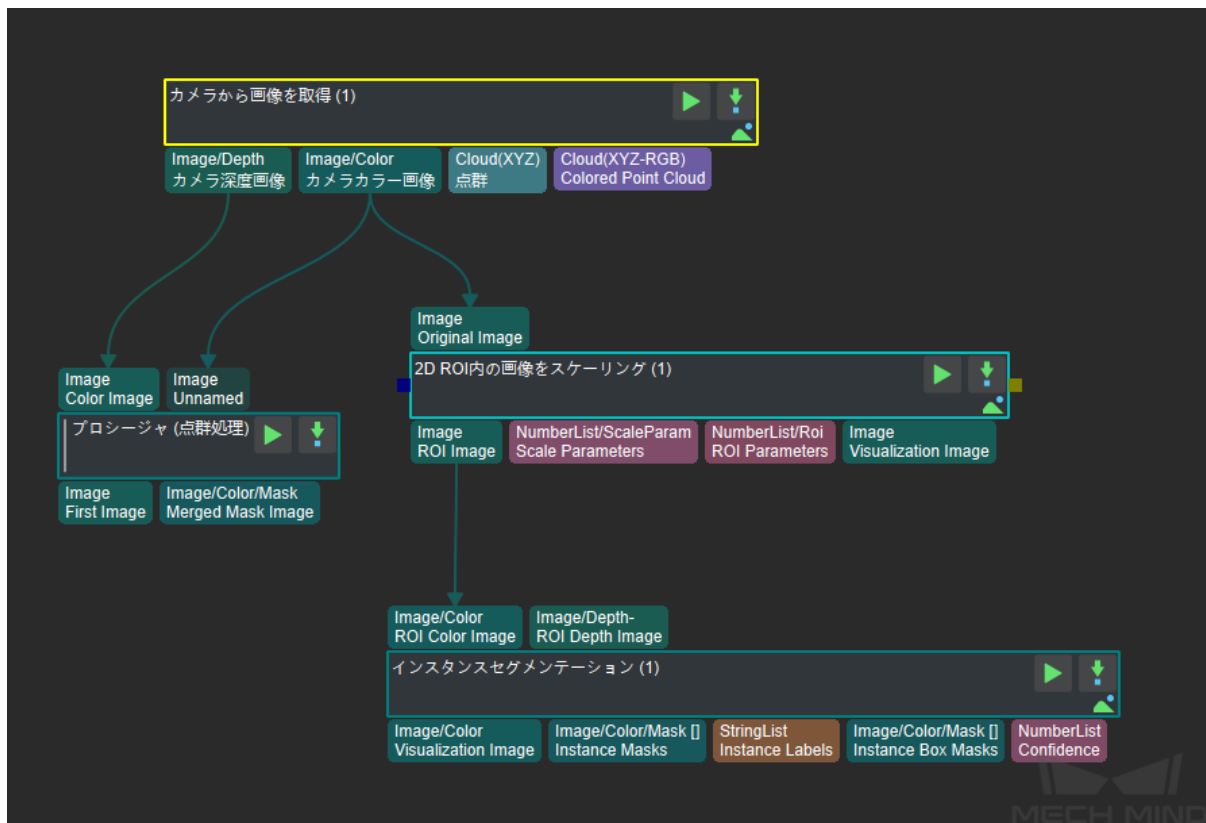


図5 ディープラーニングと点群処理の並行計算



## 点群処理

実際のプロジェクトでは、カメラから取得した点群を直接使用することはできなく、点群のクラスタリングやマージなど、より具体的な処理が必要になります。このプロセスでは、ディテールを最適化することでタクトを向上させます。以下に、タクトに大きく影響するステップとタクトの最適化の方法、いくつかの関連する点群処理を示します。

### ROI 内の点群を抽出

通常、シーン内の点群の場合、ターゲットワークを含むエリアの一部のみが注目する部分です。このとき、ROI を選択して特定のエリアに焦点を合わせ、非作業エリアの点群のほとんどをフィルターで除去して、タクトを大幅に向上させます。

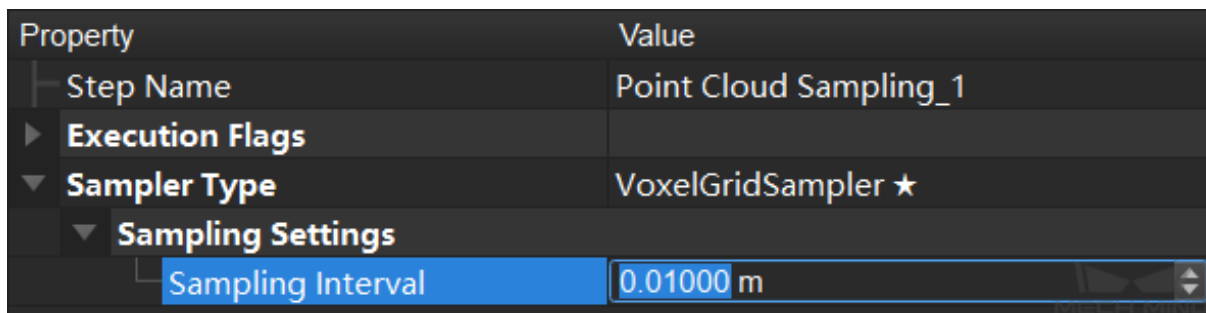


図 6 ROI 内の点群を抽出

### 点群ダウンサンプリング

このステップは、点群をサンプリングして、元の点群のサイズを縮小するために使用されます。点群密度の要件が高くないほとんどのプロジェクト（エッジマッチング、最高層のマスクの取得などのみが必要）にとって、このステップを追加すると、タクトを大幅に改善できます。【サンプリング間隔】はサンプリング倍率を制御し、タクトの要件に応じて設定できます。

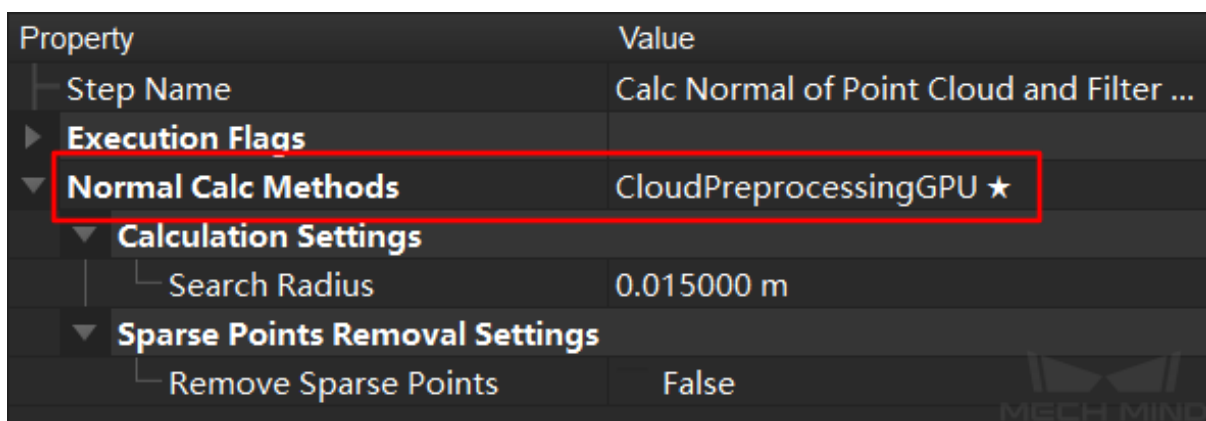


図 7 点群ダウンサンプリング

### 点群の法線ベクトルのを計算してフィルタリング

深度画像が点群に変換された後、点群の法線ベクトルを計算してフィルタリングする必要があります。適切なサポートグラフィックボード（GPU）を備えたホストコンピュータの場合、GPU アクセラレーションの法線ベクトルの計算方法を設定すると、タクトを大幅に向上できます。

Property	Value
Step Name	Points Filter_1
Execution Flags	
Points Filter Type	StatisticalOutlierFilter ★
Mean K	30
Stddev Mul Thresh	2.0000

図 8 点群の法線ベクトルのを計算してフィルタリング

### 点のフィルタリング

このステップでは、設定されたルールを使用して点群内のポイントをフィルタリングし、ノイズを除去します。タクトを改善するために、平均距離推定に使用される【最近隣接点の数】を変更し、実験テストを実行します。精度の確保に基づいて、このパラメータを適切に減らし

Property	Value
Step Name	Point Cloud Clustering_1
Execution Flags	
Cluster Algorithm	EuclideanCluster ★
Max Distance of Neighbouring...	0.003000 m
GPU Acceleration	False
Min Points Num in Cluster	800
Max Points Num in Cluster	3000000

図 9 点のフィルタリング

### 点群クラスタリング

このステップには、【GPU アクセラレーション】のオプションがあります。点群数が多い場合、このオプションをオンにするとタクトが向上します。そうでない場合、明らかな効果はありません。ビデオメモリが不足している場合（たとえば、Mech-Viz/ディープラーニングはメモリの使用が多すぎる場合）、ビデオメモリは減少せずに増加する可能性があることに注意してください。実際の状況に応じて、このパラメータを設定してください。

Property	Value
Step Name	3D Coarse Matching (Multiple Templa...
<b>Execution Flags</b>	
<b>Model and Pick Point</b>	
Model File (Required)	model.ply
Geometric Center File (Require...	
<b>Cloud Orientation Calculation</b>	
Point Orientation Calc Mode	Origin ▼
Number of Searching Points	10
<b>Processor Type</b>	SurfaceMatching ★
<b>Sample Settings</b>	
Enable Automatic Downsampling	✓ True
Expected Point Number of ...	1000
Max Point Number of Sam...	4000
Max Point Number of Sam...	30000
Sampling Interval	0.010000 m
Min Sampling Interval	0.003000 m
<b>Voting Settings</b>	
Distance Quantification	1.0000
Angle Quantification	60
Max Vote Ratio	0.8000
Reference Point Step	5
Referred Point Step	1
<b>Clustering Settings</b>	
Cluster Ratio	0.1000
Threshold of Angle Differe...	15 °
Threshold of Distance Diffe...	0.020000 m
Output First N Clusters with...	5
<b>Pose Verification Settings</b>	
Use Pose Verification	✓ True
Marked Margin	1
Voxel Length	3 mm
Maximum Number of Dete...	3
<b>Results Visualization</b>	
Show Sampled Model Cloud	<input type="checkbox"/> False
Show Sampled Scene Cloud	<input type="checkbox"/> False
Show Matching Results	<input checked="" type="checkbox"/> True

図 10 点群クラスタリング

### 3D 位置姿勢推定

3d\_matching は主に「3D 位置姿勢低精度推定」と「3D 位置姿勢高精度推定」で構成されており、テンプレートの数に応じて、シングルテンプレートとマルチテンプレートに分けられます。ここで、3D 位置姿勢低精度推定は、主に空間点群におけるテンプレート点群のおおよその位置を計算するために使用されます。3D 位置姿勢高精度推定は、主にテンプレートを空間点群とマッチングするために使用されます。このステップは空間内の各ポイントの相対位置を計算する必要があるため、ポイントが多いほど、より多くの時間がかかり、位置姿勢高精度推定の反復もより多くの時間がかかります。したがって、これら2つのステップを調整する場合の主なポイントは、処理点群（データとテンプレート）の数を減らし、反復回数（位置姿勢高精度推定）を減らすことです。

---

#### ちなみに:

- 位置姿勢低精度推定の場合、【SurfaceMatching】を選択した場合は、【サンプリング設定】を調整して点群密度を調整し、計算時間を短縮します。ダウンサンプリングされた点群の密度を下げることはできないが、それでもタクトの向上が必要な場合、【基準/基準ポイントのサンプリングステップサイズ】を増やすことを考慮してください。
  - クラスタリング、ROI、点群フィルタリングなど、初期の点群前処理により、多くの不要な点群が初期段階で削除されます。
  - 複数のワークをマッチングする場合は、同時にマッチングするワークの数を適切に減らすこともできます。例えば、一度に3つのワークのみをマッチングして、マッチング速度を上げます。
  - 実際のワークの特徴に応じて、テンプレートファイルを簡略化します。たとえば、エッジテンプレートを使用して大きなワークピースをマッチングさせたり、テンプレートをダウンサンプリングして点群の数をより小さな範囲に減らしたりします。
  - ポイント方向の計算方法：法線ベクトルがないテンプレートには、【StandardMode】を使用します。エッジ平面対象物のテンプレートの場合は、認識された対象物が適切に分割されていることを確認しながら、【EdgeTangent】を使用します。
-

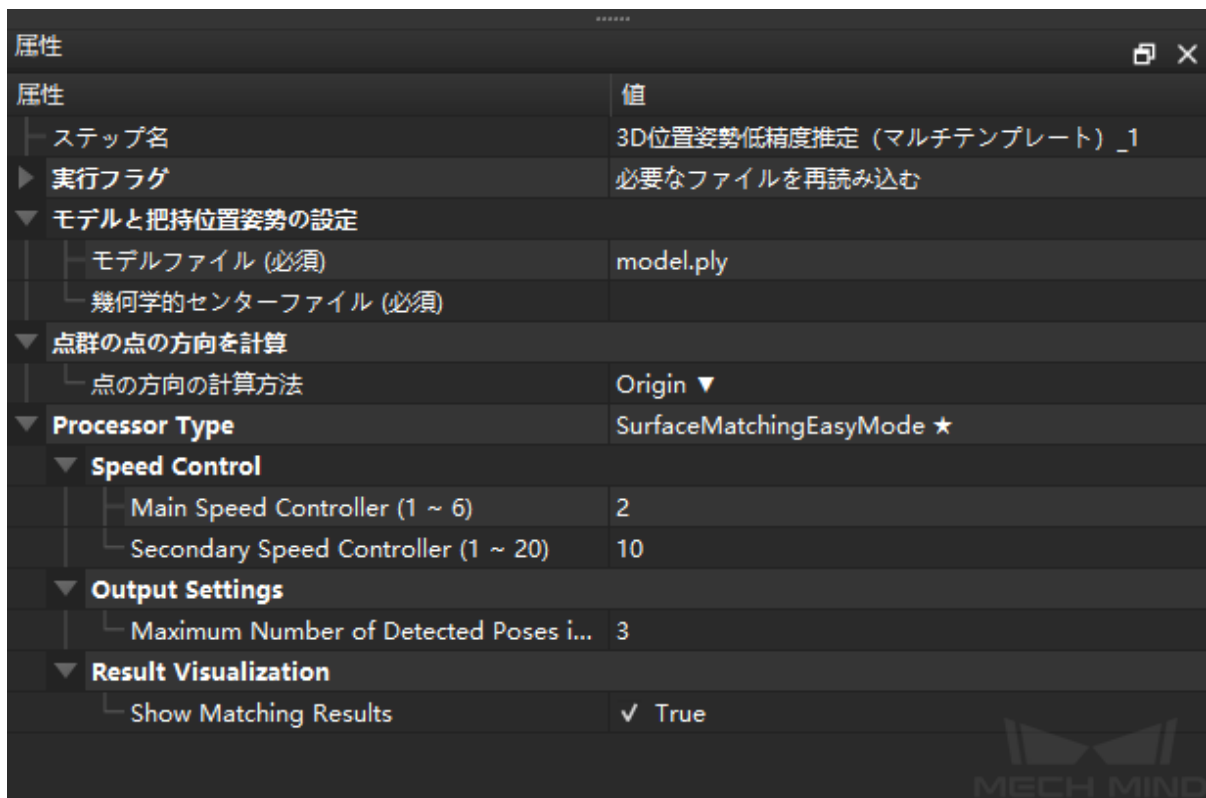


図 11 3D 位置姿勢低精度推定

属性	値
ステップ名	3D位置姿勢高精度推定 (マルチテンプレート) _1
実行フラグ	出力がない場合に実行を続ける
パラメータの調整レベル	基本 ▼
<b>モデルと把持位置姿勢の設定</b>	
モデルファイル (必須)	model.ply
幾何学的センターファイル (必須)	
モデルラベルファイル(任意)	
<b>点群の点の方向を計算</b>	
点の方向の計算方法	Origin ▼
<b>Correspondence Settings</b>	
Matching Mode	Standard ▼
Standard Deviation	0.005000 m
<b>Sample Settings</b>	
Sampling Interval	0.005000 m
<b>Validation Settings</b>	
Confidence Threshold	0.500
Search Radius	0.010000 m
<b>Output Settings</b>	
Maximum Number of Detected Poses in Ea...	1
<b>Results Visualization</b>	
Show Matching Results	✓ True

図 12 3D 位置姿勢高精度推定

## その他

プロジェクトでは、`capture_images_from_camera` と `3d_matching` に加えて、他の多くのステップがあります。ほとんどのステップがかかる時間は無視できます。実際のニーズに応じて、Mech-Vision の `guide_to_steps` を参照して調整してください。

## 産業用コンピュータ性能の向上

コンピュータビジョンと画像処理は、コンピュータのリソースを非常に使用します。優れた性能の産業用コンピュータ (Industrial Personal Computer) は、全体的な速度を大幅に向上させることができます。Mech-Mind は、産業用コンピュータに I7 8700 + 1050TI の最小構成を提供します。より高い CPU と GPU を構成すると、より良い結果が得られます。

**ちなみに:** `3d_matching` は主に CPU リソースを使用しますが、ディープラーニングはより多くの GPU リソースを使用します。

### 3.7.3 Mech-Viz とロボット

#### 経路の最適化

ロボット経路の最適化とは、主にロボット移動中の非必要な固定点と動作の削減を指します。したがって、経路最適化のキープポイントは、固定小数点を減らし、迂回を減らすことです。

**ちなみに:** 主な操作は以下のとおりです。

- 迂回路を減らし、衝突のない運動期間中の冗長な中間点を減らし、ロボットのスムーズな動作を確保します。
- 初期段階でロボットの運動を確保するために、固定小数点を減らします。
- 回転を減らします。ロボットの把持プロセス中に、治具が 180° を超えて回転する場合があります。ビジョンプロジェクトの制御と対称性の調整により、このような状況の発生を減らすことができます。また、**相対移動** を同時に使用すると、過度の回転を直接回避することができます。
- ほとんどの移動ステップに「待たない」を追加することで、ロボットの流畅さを向上させ、一時停止を減らすことができます。
- ロボットの動作経路の滑らかさを向上させるために、回転半径関数を適切かつ合理的に使用して、ロボットは経路の固定点に移動することを回避します。同時に、より多くの関節運動を使用し、より長い線形運動の使用を減らし、ロボットが特異点に移動する可能性を減らします。必要に応じて、長距離の線形運動の代わりにマルチセグメント関節運動を使用して、長距離での制御不能な運動状態を防ぎます。
- 特異点を避けます。Mech-Viz ソフトウェアの設計によれば、ロボットが特異点を越えようとする、ロボットは適切に減速して、ロボットがエラーを報告しないようにします。同様に、ワークを把持する過程で、ロボットは特異点の近くに移動しないようにして、特異点によるエラー停止や低速走行を防止する必要があります。

#### ロボットのスピード

ロボットの作業プロセスが確認されたら、ゆっくりと安定した動作の状態、ロボットの速度を達成可能な最高速度まで徐々に上げることができます。ロボット速度を上げることは、タクトに非常に役立ちます。

**ちなみに:** 各動作でのロボットの速度は、Mech-Viz ソフトウェアで直接調整できます。

#### 移動ステップの速度

Mech-Viz ソフトウェアが移動ステップを実行する場合、デフォルトの:ref:BasicMoveInfo では、加速度が 50 %、速度が 100 %です。経路の最適化が完了すると、各ステップの速度と加速度をそれぞれ調整して、さまざまな状況でロボットの速度と加速度を制御できます。

**ちなみに:** ロボットがワークを把持していないときは、速度と加速度を徐々に最大許容値まで調整できます。



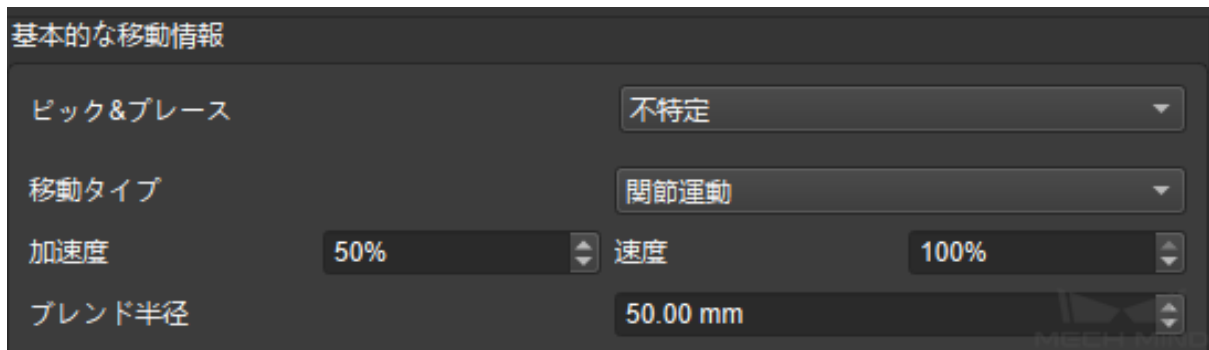


図 13 Mech-Viz で移動の速度を制御

### 撮影のプロセスとタイミング

Mech-Viz では、visual\_look ステップを使用することでキャプチャをトリガーします。Mech-Vision で処理して結果を出力し、visual\_move ステップで処理結果を取得し、ロボットを制御して指定された位置に移動させます。データ処理のプロセスはバックグラウンドで実行できます。つまり、ロボットの移動中に、取得した新しいデータを処理し、次の把持の準備をします。

#### ちなみに:

- 「Eye to Hand」モードでは、ロボットがワークを把持してカメラの視野外に移動した後、カメラのキャプチャをトリガーしてバックグラウンドで処理します。ロボットは前のワークを完了すると、すぐに次のワークを把持します。
- 「Eye in Hand」モードでは、ロボットがワークを把持した後、写真撮影点に戻って撮影し、バックグラウンドでデータ処理を行います。ロボットは、最初のワークの放置が完了すると、次のワークを直接把持します。

**ヒント:** 一部の段ボール箱のデパレタイジングのプロジェクトでは、把持するたびに他の段ボール箱が変更されないようにする場合、一回撮影で複数の把持のソリューションを使用できます。Mech-Vision ソフトウェアは、各認識プロセス中に一度に複数の対象物を認識し、取得したすべての位置姿勢を出力として Mech-Viz に送信します。そのため、1 回の認識だけで、認識された箱をすべて把持し、把持後に再撮影することで、カメラの時間を大幅に節約できます。

同様に、治具が十分に大きい場合は、ロボットを制御して、一度に異なる位置にある複数の段ボール箱を把持して、把持のタクトを向上させることができます。

## 注意事項

本節では、主に注意事項について説明します。

### 4.1 注意事項

#### 4.1.1 箱の把持シーン

1. パレットサイズを確認し、箱の長さ、幅、高さの最大値と最小値、および治具が異なる寸法の箱に対応できるかどうかを考慮してください。
2. 箱の色は認識に影響します。箱の色情報を確認する必要があります。
3. 箱のテープの色とパターンは認識に影響を与えるため、そのような情報を明確にする必要があります。
4. ストラップは認識に影響します。特にケーブルタイは認識の結果に深刻な影響を及ぼします。ストラップの色、位置、数量を確認する必要があります。
5. 箱にコードのスキャンが必要かどうかを確認します。スキャンが必要な場合は、片面または複数面がバーコードに貼付されているかどうか、バーコードの場所、スキャン情報などを確認してください。
6. 初期段階で要件を取得する場合は、パレタイジング時に箱のバーコードの向きを考慮する必要がありますかどうかを確認します。これは、治具の対称設計、オフセット、パイプラインレイアウト、ロボットの作業スペースが十分かどうかなどに影響します。
7. 正方形の箱があるかどうかを確認します。正方形の箱は最終的な認識結果に影響を与える可能性があります。
8. 後続の生産安定性を向上させるために、デバッグ中に認識に影響を与える要因を可能な限りテストすることを推奨します。

#### 4.1.2 供給シーン

1. ワークの色、それが純黒または高反射性ワークであるかどうか、テスト時に点群が完全しているかどうか、およびそれを解決するために視覚融合（二台カメラ）が必要かどうかを確認します。
2. ワークの種類、混合供給があるかどうかを確認します。
3. ワークが同じ方向に配置されているか、互いに反対に配置されているか、またはバラ積みに配置されているかどうか、ワークの配置方向を確認します。
4. アップストリームおよびダウンストリームプロセスを確認します。ロボットが把持したワークがコンベアベルトに配置されているのか、固定配置テーブルに配置されているのか、配置精度が必要かどうかを確認します。
5. 治具と把持ボックスの衝突問題を回避するために、デバッグ中に衝突モデルを作成するように、把持ボックスの寸法を確認します。

- ワークの各層の間に仕切り板があるかどうかに注意してください。仕切り板がある場合、ロボットの把持のために手動介入またはビジョンガイドが適用されているかどうかを確認する必要があります。

### 4.1.3 接着シーン

- 接着されたワークの色、仕様、種類を確認し、透明、純黒、高反射性のワークがあるかどうかを確認します。
- 接着のプロセスを確認します。現在、接着剤塗布には2つの方法があります。1つは、事前に接着経路を抽出することです。プロセス中に、テンプレートは現在のワーク種類に従って呼び出されます。複雑な接着経路と高い要件を持つシーンに適しています。もう1つは、視覚に基づいてキャプチャした後に生成した経路をリアルタイムで処理することです。これは、接着経路が簡単なシーンに適しています。
- アップストリームおよびダウストリームプロセスを確認します。他の信号入力および出力を接続するかどうかを確認します。
- お客様が接着剤の種類に関して特別な要件を持っているかどうかを確認します。
- 接着機の動作原理を理解し、ビジョンと接着機がどのように連携するかを明確にします。
- Mech-Viz マスター制御であり、テンプレートを取得する必要があるプロジェクトの場合、接着ソフトウェアを提供して接着とデバッグを連携させることができます。
- 接着剤塗布シーンの場合、接着プロセスはより複雑であるため、視覚ソリューションを確認する際に、プロセスフローを確認し、視覚的な作業範囲を明確にする必要があります。

### 4.1.4 組み立てシーン

- 組み立てるワークの色、仕様、種類を確認します。透明、純黒、高反射性のワークがあるかどうか、混合供給の場合があるかどうかを確認します。
- 現場からのロボット精度、治具の精度が組み立て要件を満たすことができるかどうかを確認します。
- ビジョンの精度と組み立て精度は、後続の実行プロセスでのトラブルシューティングを容易にするために別々に考慮されます。
- 組み立てプロセスを確認し、カメラが組み立てられるワークと一緒に移動するかどうか（移動中に撮影する）を確認します。移動の安定性とビジョンタクトが満たされているかどうかを確認します。
- 組み立てシーンでは、精度要件が最も重要な考慮事項です。ほとんどの組み立てプロセスでは、組み立て中にこすらないようにする必要があるので、並進と回転の精度要件は高く、ソリューション計画では特別な注意を払わなければなりません。

---

ヒント：精度には、並進精度と回転精度が含まれます。

---

## 4.2 他の注意事項

### 4.2.1 タクト

視覚による識別のタクト要件と作業サイクルタクト要件を含むタクト要件を確認します。

### 4.2.2 精度

ビジョン処理による認識の精度要件と把持精度要件を含む精度要件を確認します。

### 4.2.3 通信

1. 制御方式(ロボットマスター制御/Mech-Viz マスター制御)を確認します。プロセスフローによって通信方式を選択します。
2. 操作にタッチパネルを使用するかどうか、PLC、ホストコンピュータなどと通信する必要があるかどうかを確認します。

### 4.2.4 作業環境

1. **温度**：0-45° の温度範囲でカメラを使用することをお勧めします。
2. **湿度**：現場の湿度のレベルを確認します。
3. **ほこり**：現場のほこりのレベルを確認します。
4. **照明**：実際の状況に応じて、適切な遮光/補光を行います。

---

ヒント：ロボットプログラム読み込みについては、robot\_integrations をご参照ください。

---