



Welcome

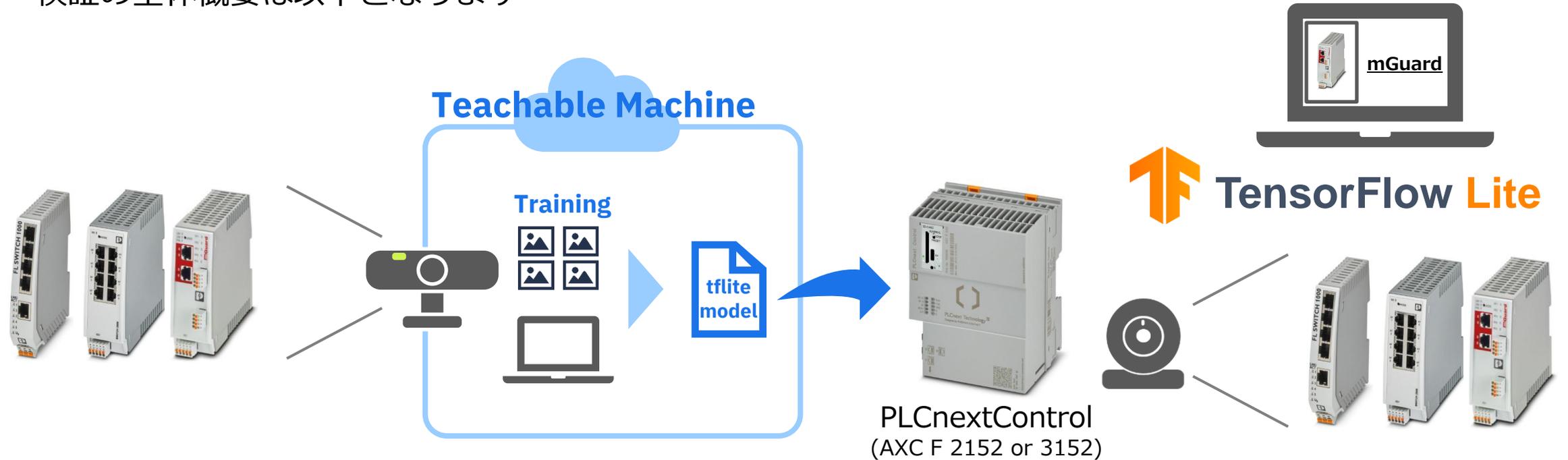
# Teachable Machine による画像分類モデルの 作成とPLCnextControl での動作検証

-PLCnextControlによる画像分類編-

# - 概要 -

# 検証の全体概要

- 検証の全体概要は以下となります

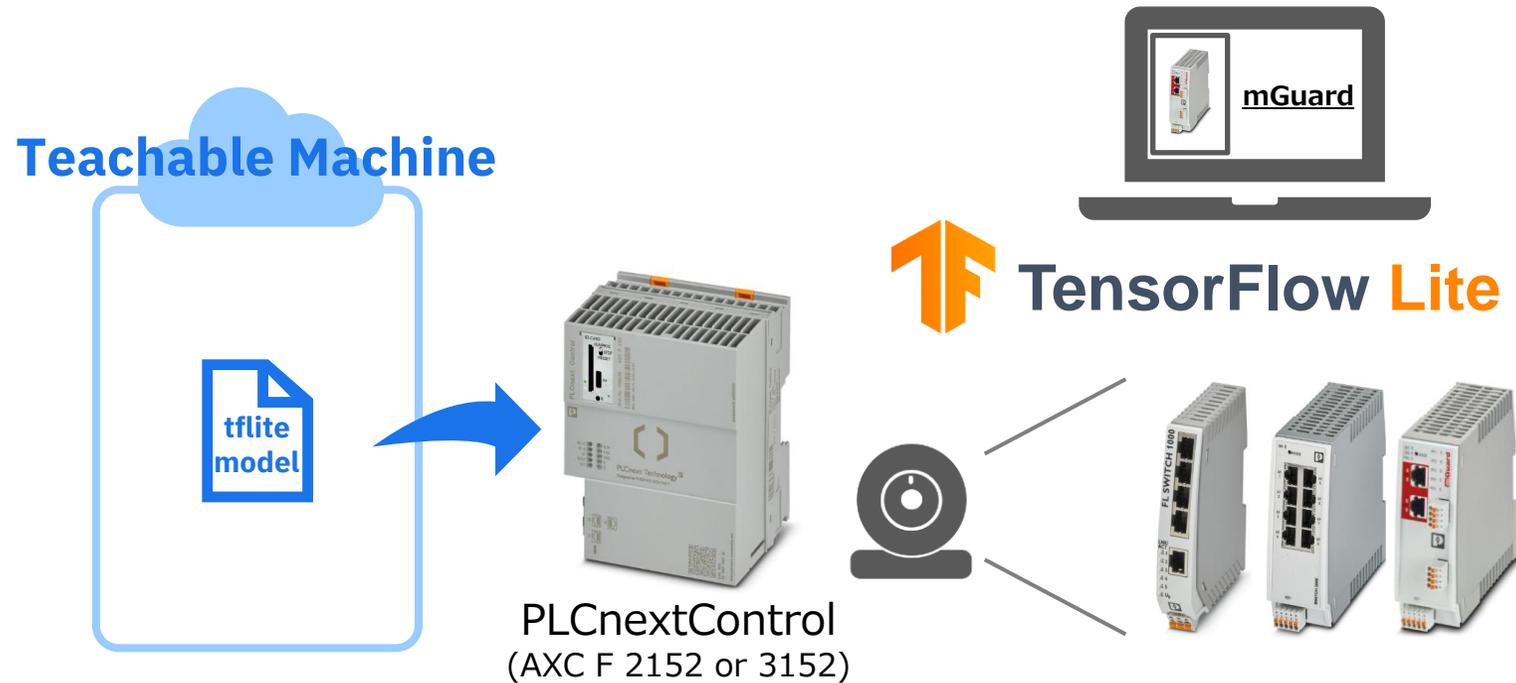


①Teachable Machineアプリ上でカメラ画像より対象機器をトレーニング  
機器を画像分類するtensorflowlite用のモデルファイルを出力

②Teachable Machineで生成したモデルファイルを用いて  
PLCnextControl機器でカメラ画像に写っている機器がどの機器かを分類する

# 本手順の概要

- 本手順ではTeachableMachineアプリを用いてモデルファイルの作成・出力まで行います

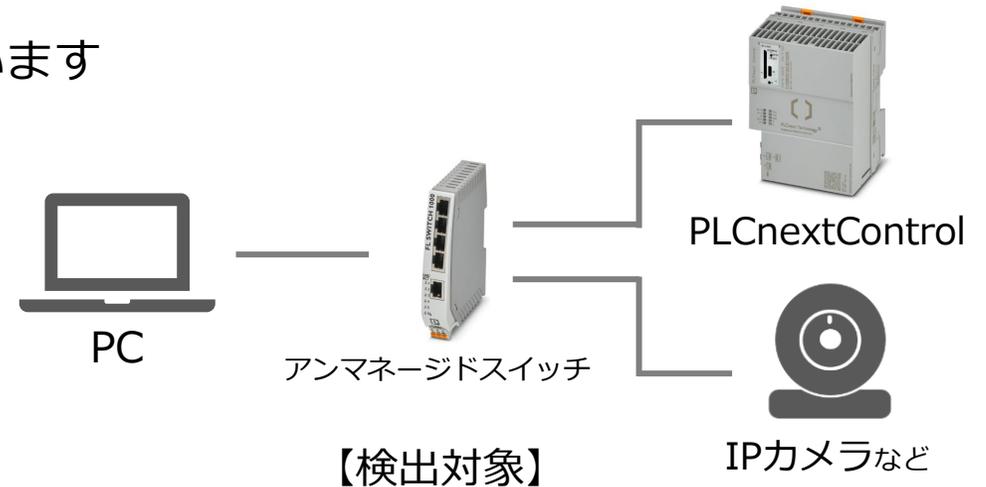


# - PLCnextControlによる画像分類 -

# 事前準備(1)

- PLCnextControlによる画像分類をするための準備を行います

①PC・PLCnextControl・IPカメラを接続します  
※右の例ではアンマネージドを用いて接続しています



②画像分類する対象を準備します  
モデルトレーニングで使用した機器を準備します



## 事前準備(2)

- PLCnextControlによる画像分類をするための準備を行います

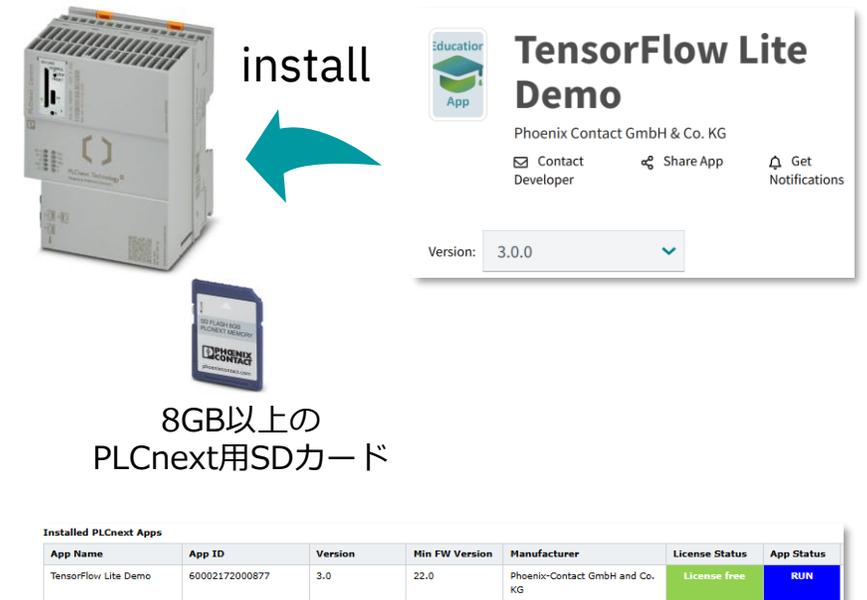
③本検証ではPLCnextStoreにあるアプリ「TensorFlow Lite Demo」の環境を使用します。そのため上記アプリがPLCnextControlにインストールおよびRUNされている必要があります。

<https://www.plcnextstore.com/world/app/1226>

※アプリのインストールを行う際は8GB以上のPLCnext用SDカードの使用が推奨されます

④PLCnextControlにてrootユーザーの作成を行います  
rootユーザーの作成方法については以下記事を参照ください

<https://plcnext.jp/archives/1610>



install

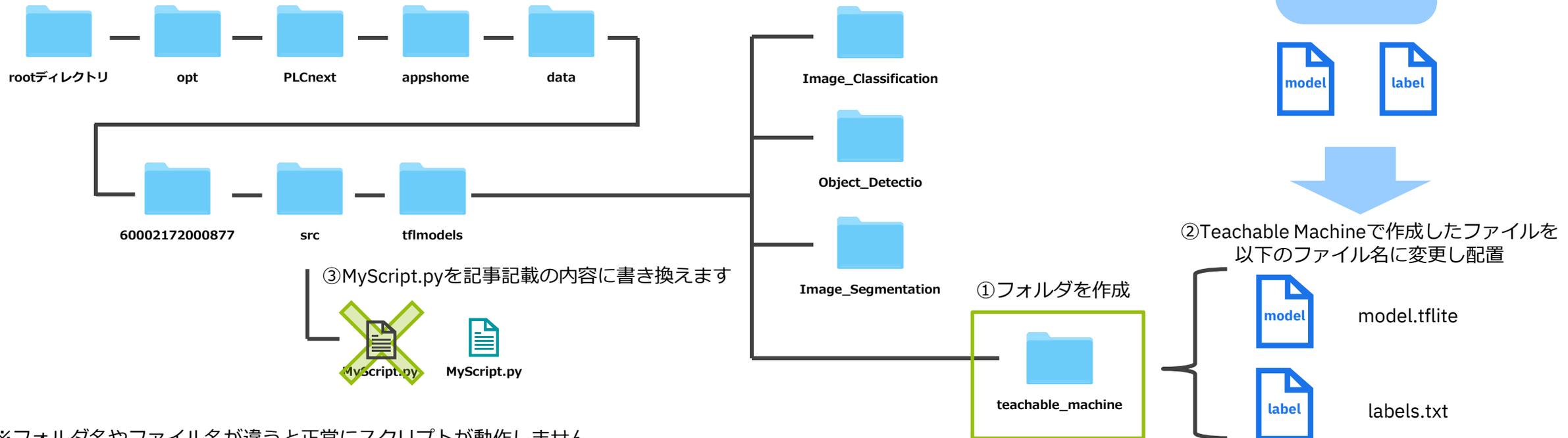
8GB以上のPLCnext用SDカード

Installed PLCnext Apps						
App Name	App ID	Version	Min FW Version	Manufacturer	License Status	App Status
TensorFlow Lite Demo	60002172000877	3.0	22.0	Phoenix-Contact GmbH and Co. KG	License free	RUN

# PLCnextControlへファイルを配置

- PLCnextControlへSSHアクセスを行いフォルダを作成のファイルの配置を行います

対象機器へSSHアクセスなどを行い右図のフォルダを作成します



※フォルダ名やファイル名が違くと正常にスクリプトが動作しません

# pythonスクリプトファイルの設定確認

- 画像分類処理を行うスクリプトファイルの設定確認を行います

MyScript.pyファイルの26行目よりスクリプトにて用いられるファイルなどのパスを指定しています

こちらのcamera\_urlにてIPカメラのRTSP用のアドレスを入力してください

※RTSPアドレスについては使用されるIPカメラのマニュアル等を参照ください

画像分類のためのスナップショット画像は以下ファイルとして保存されます

/opt/plcnext/appshome/data/60002172000877/src/imgs/camera\_snapshot.jpg

```
19 #####
20 # 設定値一覧(こちらに任意の値を記載してください)
21 #####
22
23 def load_manual_config_params():
24
25     # 各種データのパス設定
26     model_path = "/src/tflmodels/teachable_machine/model.tflite" # モデルのパス
27     label_path = "/src/tflmodels/teachable_machine/labels.txt" # ラベルのパス
28     camera_url = "rtsp://private:private@192.168.1.145:554/stream2" # カメラURL
29     output_filename = '/src/imgs/camera_snapshot.jpg' # スナップショット画像のパス・ファイル名
30
31     return model_path, label_path, camera_url, output_filename
```

MyScript.py

# Docker-compose.ymlファイルの設定変更

- 画像分類を行うpythonスクリプトの標準出力のバッファリングをオフにします

本検証にて使用するMyScript.pyでは画像分類結果を標準出力にて表示させます

リアルタイムで結果を確認するため、pythonスクリプトの実行の際にバッファリングをオフにするオプション(-u)を追加します

スクリプトの実行設定をしている  
/opt/plcnext/appshome/data/60002172000877/docker-compose.yml  
を開きcommandのpython3の右側に-uオプションを追加します。

設定が完了したら機器の電源入切を行ってください

```
1 version: "2.16.1"
2 services:
3   # template-nginx-demo:
4   tensorflow_lite_2_16_1:
5     image: ${IMAGE_NAME}:${IMAGE_TAG}
6     #ports:
7     # - ${OUTGRESS_PORT}:${INGRESS_PORT}
8     user: ${USER_ID}
9     restart: unless-stopped
10    volumes:
11      - ./src:/src
12    command: bash -c "python3 -u /src/MyScript.py"
```

docker-compose.yml

# スクリプトの画像分類出力表示

- 対象スクリプトの動作出力表示を行います
  - ①対象のPLCnextControl機器にSSHアクセスします
  - ②rootユーザーに切り替えます  
\$ su
  - ③TensorFlowLiteDemoアプリのプロセスIDを確認します  
# podman ps
  - ④対象アプリのlogをリアルタイムで出力させます  
# podman logs --tail 10 --follow [プロセスID]

※画像分類は約5秒間隔で実行されます

```
admin@axcf3152:~$ su
Password:
root@axcf3152:/opt/plcnext/# podman ps
CONTAINER ID   IMAGE                                     COMMAND
6d1d393045ad   docker.io/library/mltrainingapp3152:latest  bash -c python3 -...
low_lite_2_16_1_1
root@axcf3152:/opt/plcnext/# podman logs --tail 10 --follow 6d1d393045ad
予測されたクラス: 3 NoObject (確率: 0.7405500411987305)
----- 721回目 -----
カメラのスナップショット処理に成功しました
推論結果 (ラベルとスコア):
0 FL SWITCH 1005N: 0.05081097409129143
1 FL SWITCH 2008: 4.264842573320493e-05
2 FL MGUARD 2102: 0.17908120155334473
3 NoObject: 0.770065188407898
予測されたクラス: 3 NoObject (確率: 0.770065188407898)
```

# スクリプトの画像分類出力確認

- 対象スクリプトの動作出力確認を行います

出力内容は以下4点となります

- ・ 画像分類処理の実行回数
- ・ カメラとの接続及びモデル入力用画像処理結果
- ・ 画像分類のラベルとスコア  
1(高) ~ 0(低) として表示
- ・ 一番スコアの高いラベルとスコアの表示

※TeachableMachineのトレーニングで使用したカメラとPLCnextControl機器に接続したカメラの違い及びTeachableMachineでの画像処理と本pythonスクリプトの画像処理の違いなどによりTeachableMachineアプリと出力スコアが異なります

```
----- 721回目 -----  
カメラのスナップショット処理に成功しました  
推論結果 (ラベルとスコア) :  
0 FL SWITCH 1005N: 0.05081097409129143  
1 FL SWITCH 2008: 4.264842573320493e-05  
2 FL MGuard 2102: 0.17908120155334473  
3 NoObject: 0.770065188407898  
予測されたクラス: 3 NoObject (確率: 0.770065188407898)
```

# IPカメラの映像確認について

画像分類を行うにあたり、カメラの画角などはとても重要になってきますがPLCnextControl機器にはリアルタイムでIPカメラの映像を出力する機能はありません。

そのためIPカメラビューアーアプリなどを起動させIPカメラの映像を確認しながらの動作検証が好ましいです。  
(右画像ではVLC media playerを使用しています)



IPカメラビューアーアプリでの映像確認

# Thank you