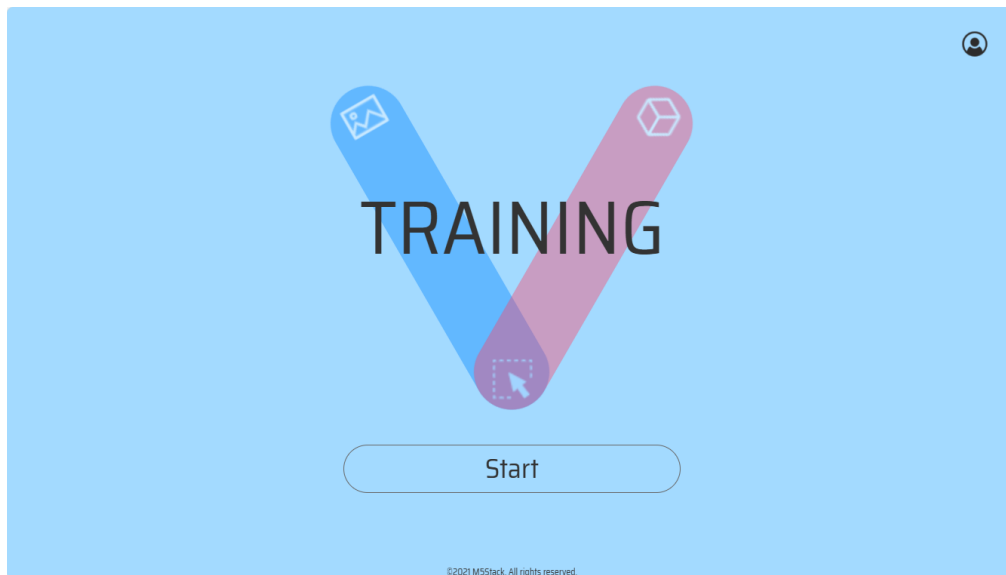


V-Training

準備作業

プラットフォームにログイン

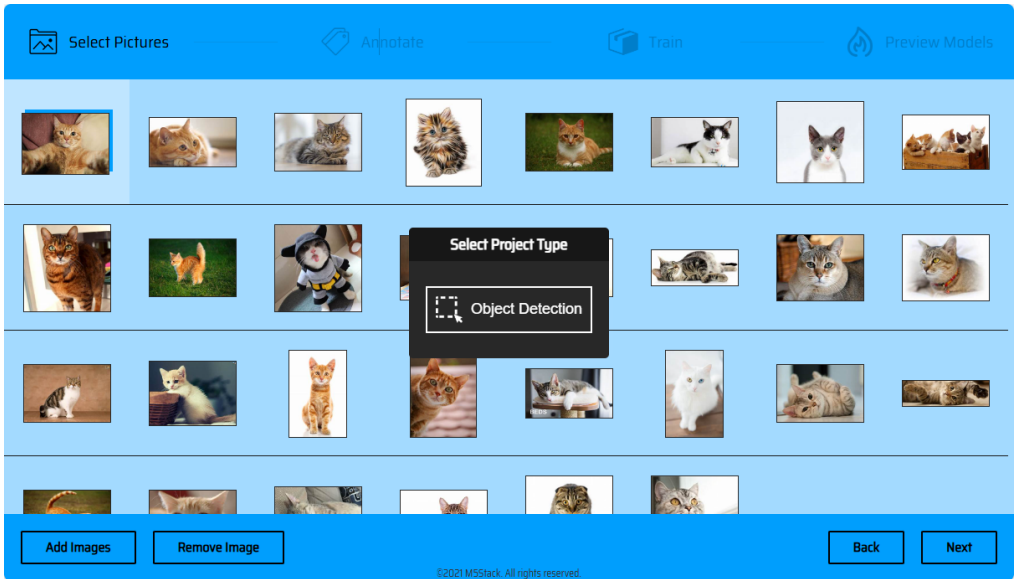
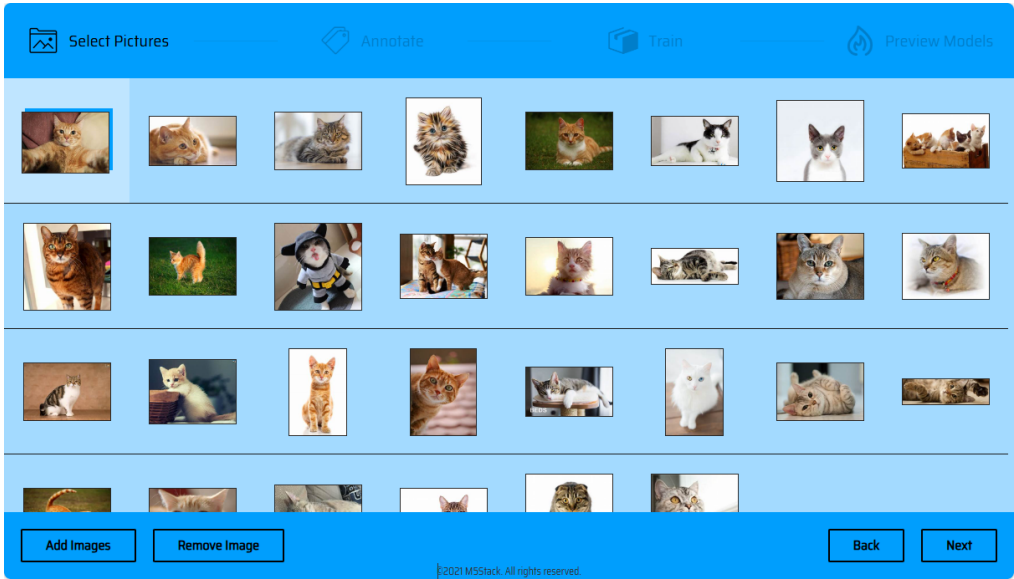
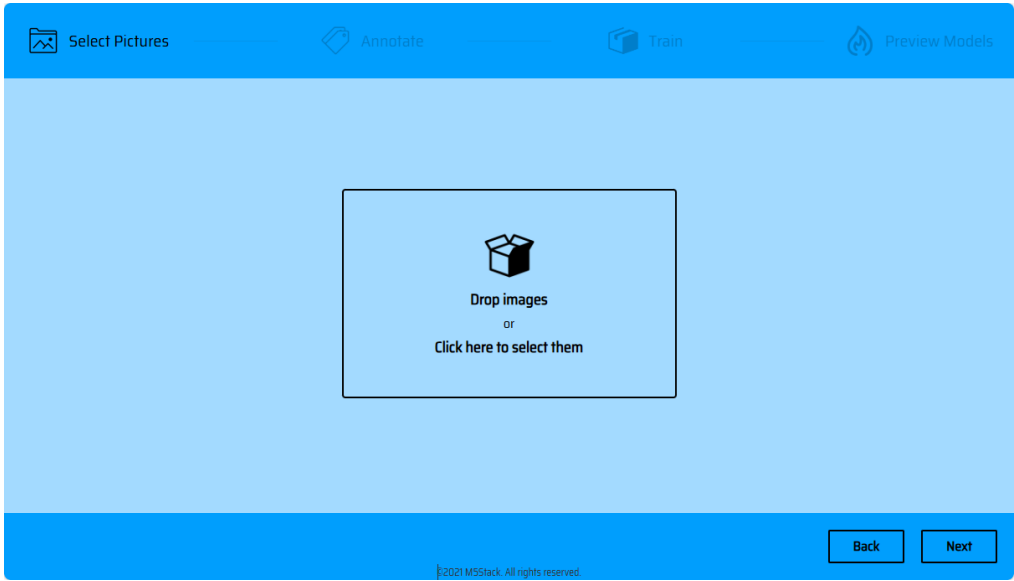
M5StackのV-Training(Aiモデル訓練サービス)を通じて、簡単に自定義の識別モデルを構築できます。携帯電話やその他の撮像機器を使って画像素材を撮影し、コンピューターに保存し、[V-Trainingオンライン訓練プラットフォーム](#)にアクセス、アカウントを登録してログイン(M5フォーラムのアカウントは直接ログインできます)。



画像のインポート

「Start」->「新規プロジェクト」->「画像のインポート」->「NEXT」->「Object Detection」をクリック。注意事項：訓練セットの質と量は訓練されたモデルの質を直接影響します。したがって、訓練セットの撮影や収集を行う際は、できるだけ高品質の訓練素材を提供してください。数量は多ければ多いほど良いです。素材撮影のシーンは実際の識別シーンに合わせてください。

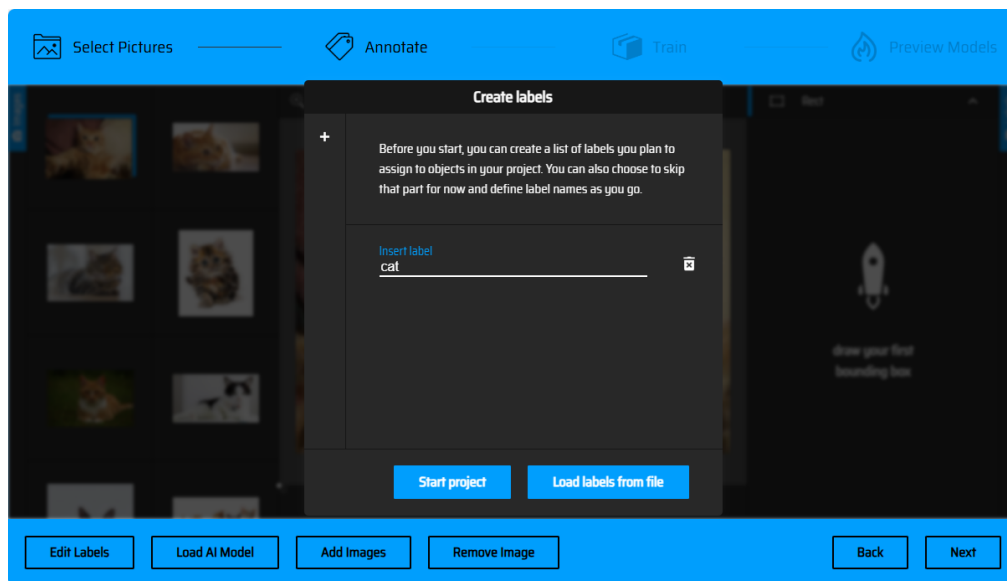
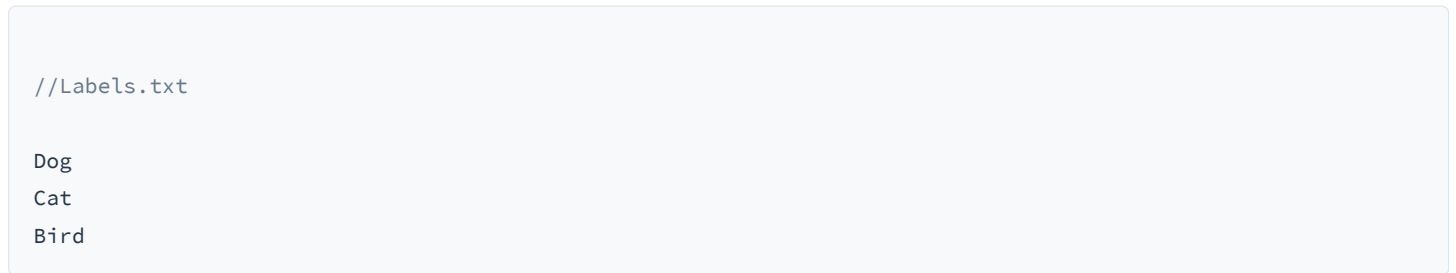
注意：「画像訓練セット全体のサイズは200Mを超えてはいけません」



素材の処理

ラベルの作成

対象をフレーム選択する前に、識別の対象にラベル名を作成する必要があります。後続の画像のマーキング操作中、異なる対象に応じて対応のラベルを選択してフレーム選択を行います。(弹出ウィンドウの左側の「+」ボタンをクリックして、複数のラベルを作成できます)。ラベルをテキストファイルの形式で一括インポートもできます(Load Labels from file)、ファイル形式はtxtで、ファイル内容は各行が一つのラベル名です。(以下のように)

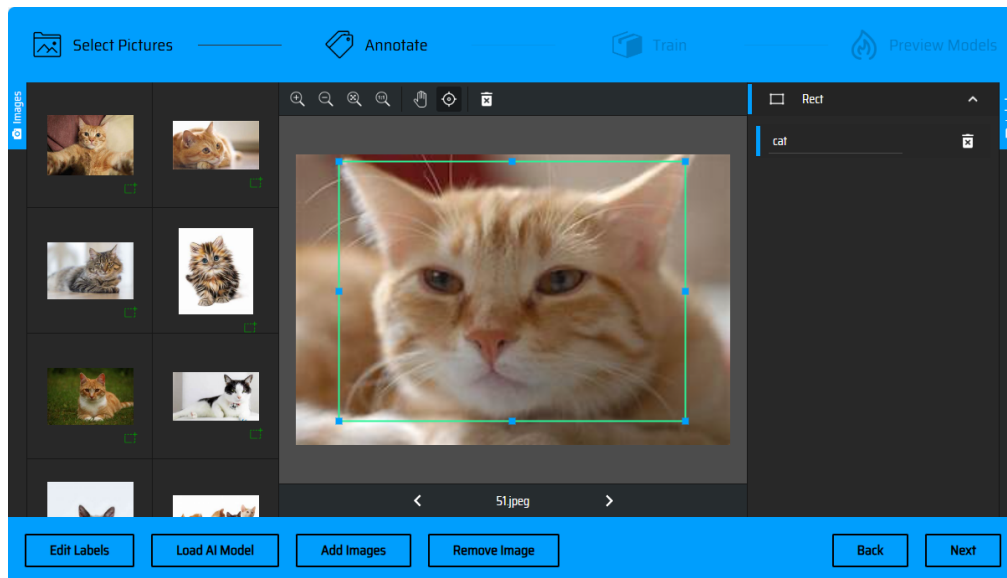


画像のマーキング

ラベルの作成が完了後、次は画像のマーキングの環節です。訓練セットの素材中で識別する必要のある対象をフレーム選択します。ページの左側は処理する必要のある画像のリストで、角標を通じて処理済みの画像を知ることができます。

手動マーキング

底枠の矢印をクリックで画像を切替えできます(またはキーボードの<-左->右ボタンを押下で画像を切替えできます)、右側のメニューバーはラベルのリストで、フレーム選択後、対象に応じたラベルを指定できます。



AI自動マーキング

素材を一括処理する際、AI自動マーキング機能を使ってマーキングの効率を上げる試みもできます。左下の「Load AI Model」->「COCO SSD - object detection using rectangles」をチェック->「Use model!」をクリック。フレーム選択のモデルがロード完了後、ページの色は緑色に変化します。次に前ステップで作成したラベル名をリストに追加して、後続のマーキングに使用します。「Accept」をクリック。

AIは画像中で識別できる対象を自動的に識別し、フレーム選択します。次に、各画像のフレーム選択を審査する必要があります。AIがあるカテゴリーの対象を識別した際、classsの選択ボックスがポップアップし、AIが推奨のラベルカテゴリーを選択してリストに追加できます、または直接「Accept」をクリックで、次の操作に進み、ラベルリスト中既存のラベルを使って分類マーキングを行います。

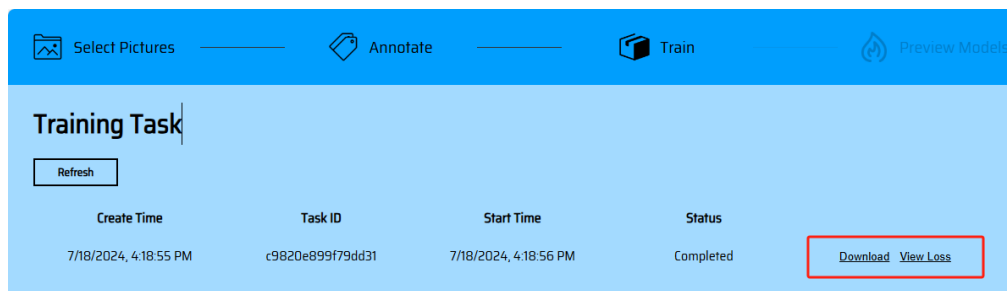
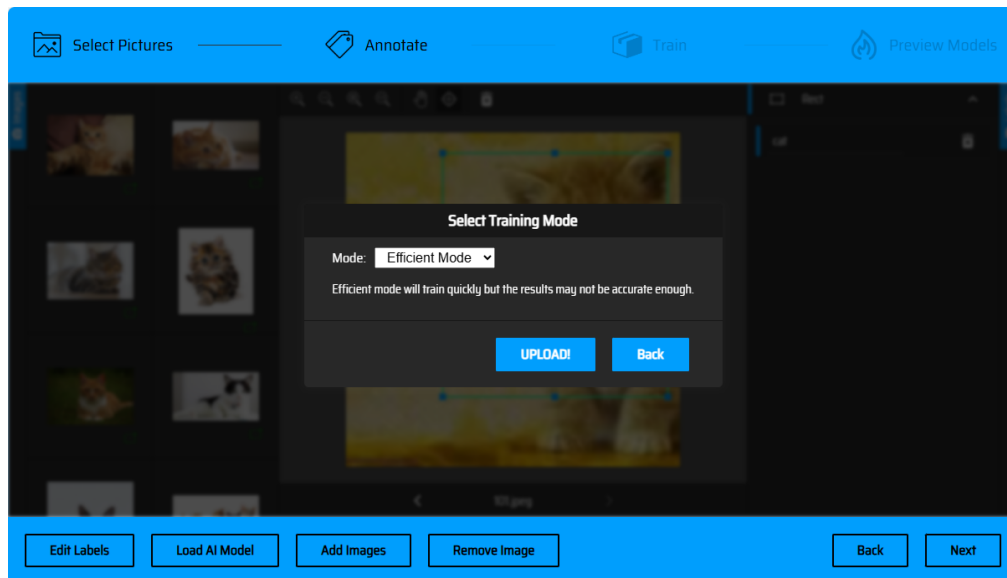
識別のフレーム選択が正確な場合は、選択枠の「+」をクリックで追加します。(またはEnterを押下で確認します)、識別のフレーム選択が誤った場合は、選択枠の削除ボタンをクリックで削除し、手動でフレーム選

モデルの訓練

枠選択を完了した後、次へボタンをクリックして素材をアップロードしてください(マーキングが完了していない素材が存在する場合は、該当ページで提示されます)。UPLOADをクリックして素材をアップロードしてください(現在はefficient高効率訓練モードがサポートされています)。アップロード後、タスク一覧画面にジャンプします。Refreshをクリックしてタスクの最新の状態を確認できます。訓練が完了した後、モデルのDownloadダウンロードリンクと損失曲線を手入できます。

注意事項:

訓練に失敗した場合は、タスクのdetailボタンをクリックしてエラーの詳細を確認し、エラーの原因を把握後修正を加え、再度アップロードしてください



モデルのオンラインプレビュー

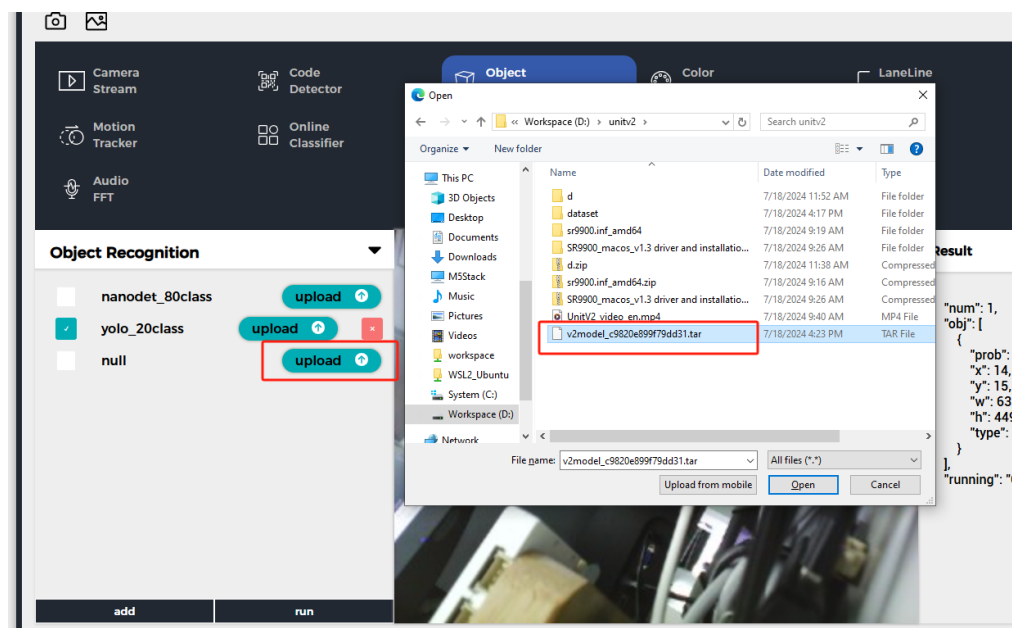
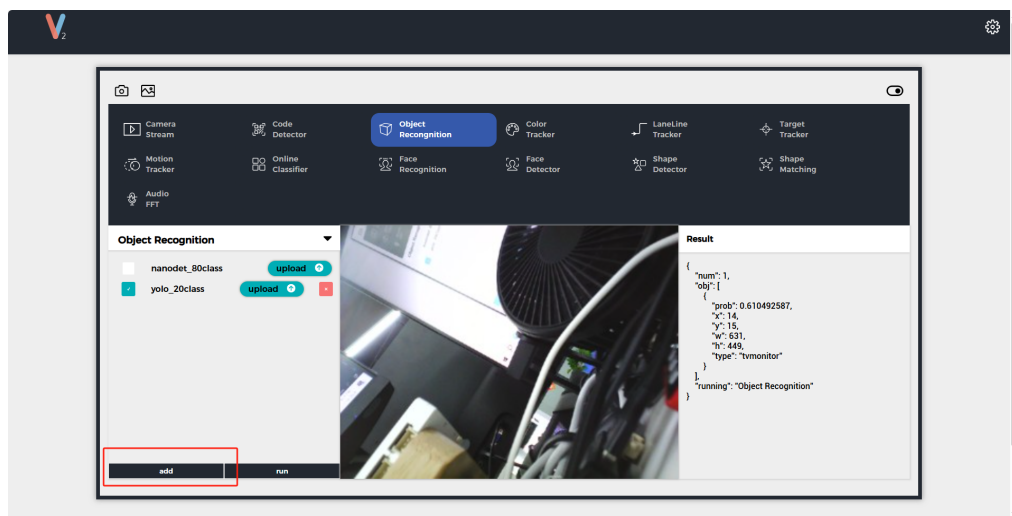
モデルのオンラインプレビュー機能は現在開発中です。現在、ユーザーはプログラムを通じてモデルをロードし、識別効果を体験できます。

プログラムでモデルをロード

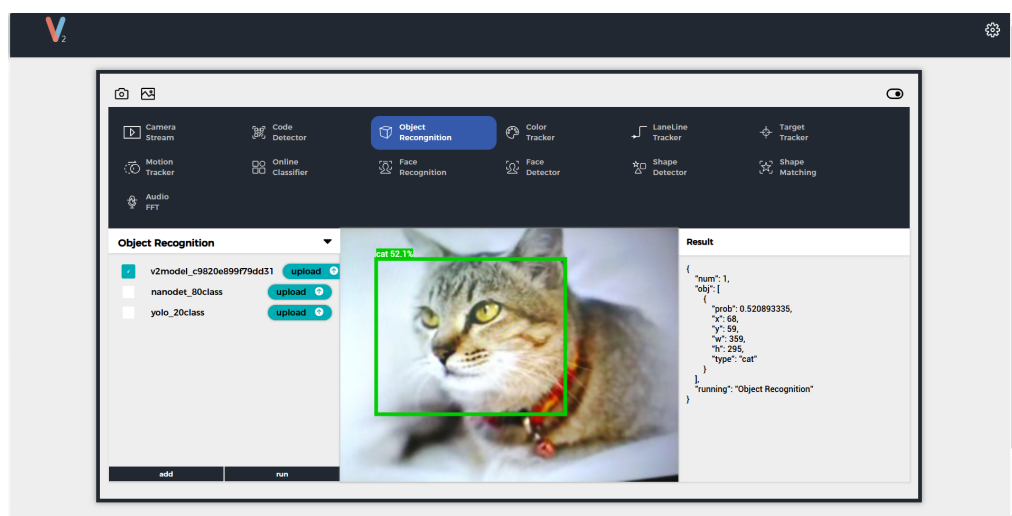
モデルの運行

- Ethernetモード接続: UnitV2は内蔵の有線NICを搭載しており、Type-CインターフェースをPCと接続すると、UnitV2と自動的にネットワーク接続が確立されます。
- APモード接続: UnitV2起動後、デフォルトでAPホットスポット(SSID: M5UV2_XXX: PWD:12345678)を開きます。ユーザーはWiFiを通じてUnitV2とネットワーク接続を確立できます

上記のいずれかのモードを通じてUnitV2デバイスに接続後、unitv2.pyドメイン名またはIP:10.254.239.1を訪問して識別機能のプレビューウェブページを開きます。機能をObject Recognitionに切り替え、addボタンをクリックしてモデルをアップロードします。注意: 使用前はSR9900のドライバをインストールしてください。詳細なインストール手順は、前章[Jupyter notebook](#)を参照してください。



アップロード完了後、runをクリックしてモデルの使用を開始できます。識別中、UnitV2はシリアルポート(底面のHY2.0-4Pインタフェース)を通じて識別サンプルデータ(JSON形式、UART: 115200bps 8N1)を継続的に出力します。



サンプル出力

```
{
  "num": 1,
  "obj": [
    {
      "prob": 0.938137174,
      "x": 179,
      "y": 186,
      "w": 330,
      "h": 273,
      "type": "dog"
    }
  ],
  "running": "Object Recognition"
}
```

Pythonでモデルファイルを読み出す

```
from json.decoder import JSONDecodeError
import subprocess
import json
import base64

reconizer = subprocess.Popen(['/home/m5stack/payload/bin/object_recognition', '/home/m5stack/payload/uploads

reconizer.stdin.write("_{\"stream\":1}\r\n".encode('utf-8'))
reconizer.stdin.flush()

img = b''

while 1:
    doc = json.loads(reconizer.stdout.readline().decode('utf-8'))
    print(doc)
    if 'img' in doc:
        byte_data = base64.b64decode(doc["img"])
        img = bytes(byte_data)
    elif 'num' in doc:
        for obj in doc['obj']:
            print(obj)
```

Nanodet

V-Trainingは軽量の物体識別モデルNanodetを採用しています。

◦ [Nanodet - Github](#)