

演習 4: 機械学習による推論の活用

演習 2 では、トピック「hello/world」で AWS Greengrass コアをテストしました。今回は、AWS Greengrass コアのデバイス シャドウとその他のトピックについてももう少し詳しく見ていきます。

この演習では、演習 2 でデプロイした Video Inference Handler および Image Upload Handler 関数を使用して、リモート アセットのユニット コントローラーに接続した USB カメラから AWS クラウドへ、機械学習をベースにした「人物検出」アラート、およびキャプチャしたビデオ フレームの静止画像を送信します。

- `compressor/<groupname>`: ビデオ フレーム イメージに推論した人物の境界ボックスを追加してアップロードする AWS Lambda 関数は、このトピックにテレメトリを送信します。メッセージ ペイロードは、検出した人物の数とイメージ ファイル名で構成されます。

この ML ビデオ監視アプリケーションは、次の 2 つのパラメーターを使用します。

- `num_seconds`: このアプリケーションの実行時間を指定します。
- `threshold`: キャプチャ条件を指定します。ここで指定した値以上の人数が検出された場合に、ビデオ フレームがキャプチャされます。

準備

演習 2 では、Greengrass グループの一部として設定された Lambda 関数をデプロイしました。この演習でも、同じグループを再利用します。具体的には、次のとおりです。

1. グループ名は `<prefix>-gateway-ultra96-group` と定義しました。
2. S3 バケット名は `<prefix>-aws-cloud-and-xilinx-workshop` と定義しました。

Greengrass コア サービスがまだ実行中であることを確認してください。そのためには、次のコマンドを実行します。

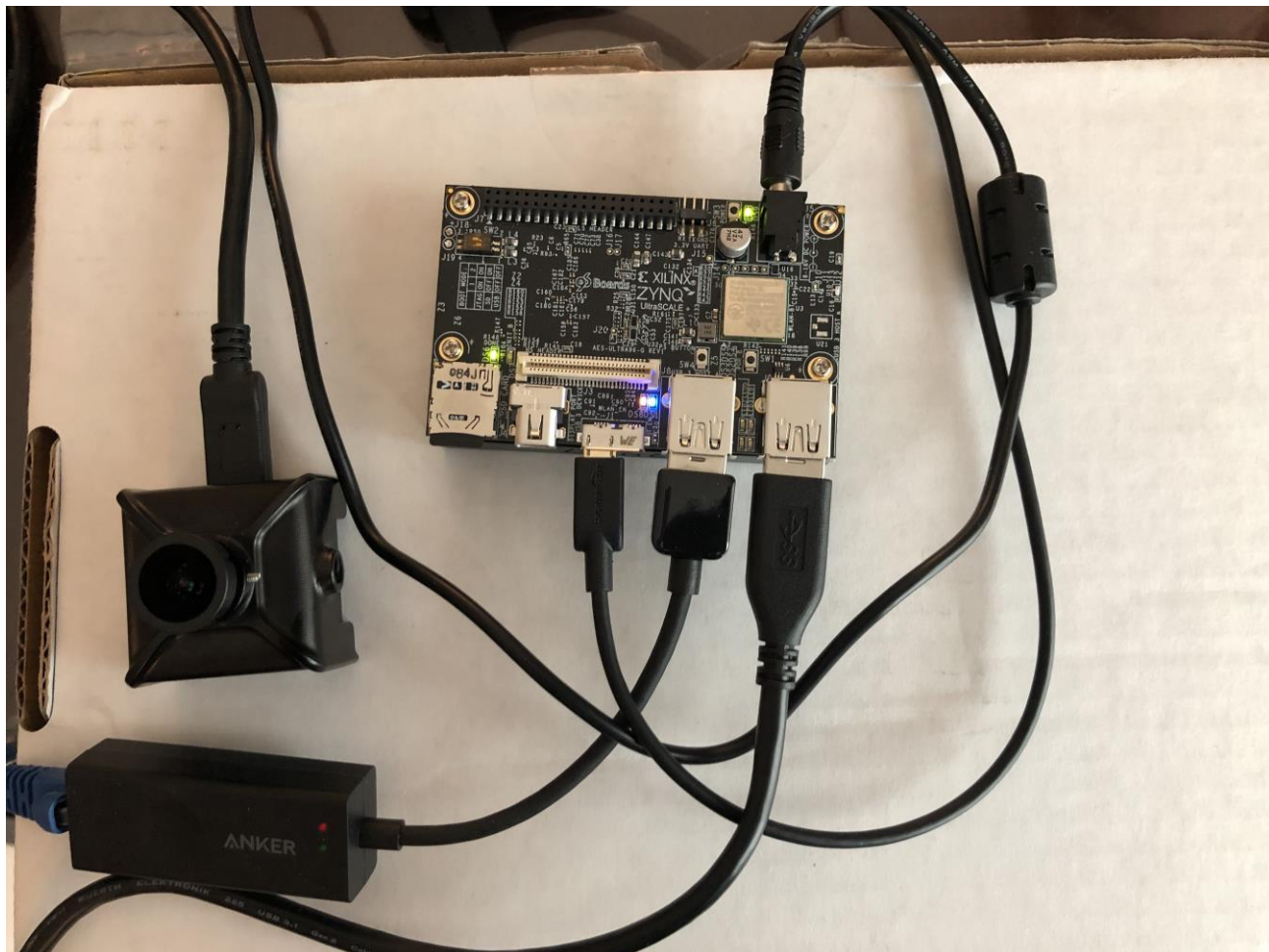
```
ps aux | grep /greengrass/gg/packages/1.7.0/bin/daemon
```

これで、バックグラウンドで動作しているプロセスを確認できます。

演習の手順

既存の AWS Greengrass グループを使って作業を開始します。

1. 接続がまだの場合は、eCon USB カメラを Ultra96 ボードの J8 に接続します。Ultra96 にカメラを接続すると、次のようになります。



2. Ultra96 に接続したターミナル ウィンドウで、次のコマンドを実行します。

```
cd $HOME/download
```

最初に、このフォルダーが空であることを確認します。このフォルダーは、演習 5 でクラウドとのローカル同期ポイントとして使用します。

3. ここで、AWS Greengrass グループをもう一度デプロイします。

```
cd $HOME/aws-cloud-and-xilinx-workshop/cloud/script  
./deploy-greengrass-group.sh <prefix>
```

数秒後に、グループが正しくデプロイされます。

エッジ デバイスで人物が検出されると、ビデオ フレームがキャプチャされ、このビデオ フレーム内の人物の周囲に境界ボックスを追加してから AWS クラウドへ送信されます。

4. AWS IoT コンソール ページにアクセスし、左側メニューの **[Test]** をクリックします。
5. **[Subscriptions]** ヘッダーの下にある **[Subscribe to a topic]** をクリックします。
6. **[Subscription topic]** に「compressor/+」と入力します。
7. **[Subscribe to topic]** をクリックします。

次に、この応答を示します。

The screenshot displays the AWS IoT console interface for the 'compressor/+' topic. On the left, there's a 'Subscriptions' sidebar with 'compressor/+' selected. The main area shows a 'Publish' section with a text input containing 'compressor/+' and a 'Publish to topic' button. Below this is a code editor showing a JSON message:

```
1 {
2   "message": "Hello from AWS IoT console"
3 }
```

. At the bottom, a message history entry is visible for 'compressor/test01-gateway-ultra96' with a timestamp of 'Nov 28, 2018 10:35:38 AM -0800' and a JSON payload:

```
{
  "num_persons": "2",
  "frame_image": "1543430138.jpg"
}
```

Video Inference Handler を呼び出す Lambda 関数の実行には時間がかかります。複数の人物がいる方向にカメラを向けると、トピック `compressor/+` にいくつかのメッセージがパブリッシュされるのを確認できます。

ML の設定は変更していないため、ビデオ監視アプリケーションはデフォルトのパラメーター (`num_seconds = 5`、`threshold = 2`) を使用します。

まとめ

この演習では、エッジにデプロイした FPGA ベースの機械学習アプリケーションからデータをクラウドに送信する方法について見てきました。ML の推論をエッジにデプロイすると、クラウドに送信するデータの量を飛躍的に削減できます。これは、ネットワーク接続の安定性、コスト、帯域幅に制約のあるリモート インダストリアルアセットにとって重要な点です。

[次の演習](#)

[トップ ページ](#)

この資料は英語版を翻訳したもので、内容に相違が生じる場合には原文を優先します。資料によっては英語版の更新に対応していないものがあります。日本語版は参考用としてご使用の上、最新情報につきましては、必ず最新英語版をご参照ください。