

SONY



Vision and Sensing Application SDK Development Container 機能仕様書

Copyright 2023 Sony Semiconductor Solutions Corporation

Version 0.2.0

2023 - 1 - 30

AITRIOS™、およびそのロゴは、ソニーグループ株式会社またはその関連会社の登録商標または商標です。

目次

1. 更新履歴	1
2. 用語・略語	2
3. 参照資料	3
4. 想定ユースケース	4
5. 機能概要、アルゴリズム	5
6. 操作性仕様、画面仕様	9
7. 目標性能	11
8. 制限事項	12
9. その他特記事項	13
10. 未決定事項	14

1. 更新履歴

Date	What/Why
2022/11/16	初版作成
2023/01/30	フォルダ構成変更。SDKが提供する機能の追加・変更。PDFビルド環境更新。

2. 用語・略語

Terms/Abbreviations	Meaning
Dev Container	GitHub CodespacesやVS Codeで利用できる、ソフトウェア開発環境を備えたDockerコンテナ。本SDKはDev Containerをベースとして提供される
Cloud App	Post-processing applicationで処理したデータを入力とした、Cloudで動作するAIアプリケーション

3. 参照資料

- Reference/Related documents (関連資料)
 - Codespaces
 - <https://docs.github.com/ja/codespaces>
 - VS Code Remote Development
 - <https://code.visualstudio.com/docs/remote/remote-overview>
 - VS Code Codespaces extension
 - <https://marketplace.visualstudio.com/items?itemName=GitHub.codespaces>
 - VS Code Remote Development Extension Pack
 - <https://marketplace.visualstudio.com/items?itemName=ms-vscode-remote.vscode-remote-extensionpack>
 - CVAT
 - <https://github.com/opencv/cvat>
 - MCT
 - https://github.com/sony/model_optimization
 - COCO
 - <https://cocodataset.org/#home>

4. 想定ユースケース

- 開発に必要なコンポーネントが既に含まれている環境を使用することで、手間を減らしたい
- 他の環境に依存しない環境を利用したい
- チームで同じ環境を使用したい
- AIアプリケーション開発について、ワークフロー全体の概要をサンプルコードで試して理解したい
- AIアプリケーション開発について知識がない状態でもスムーズに開発を行いたい

5. 機能概要、アルゴリズム

Functional Overview

- AIアプリケーションを開発するためのコンテナ環境を提供する
 - コンテナ環境は下記の方法で利用できる
 - Codespacesを利用する
 - UIとして、BrowserとVS Code desktopの2種類がある
 - Local PCにコンテナ環境を構築し、VS Codeから利用する
 - コンテナ環境には下記が含まれる
 - AIアプリケーション開発の各ワークフローで利用できるツールおよび動作環境
 - 各ワークフローの手順
 - サンプルコード
 - 詳細は、後述の[AIアプリケーション開発ワークフローと提供する機能](#)を参照



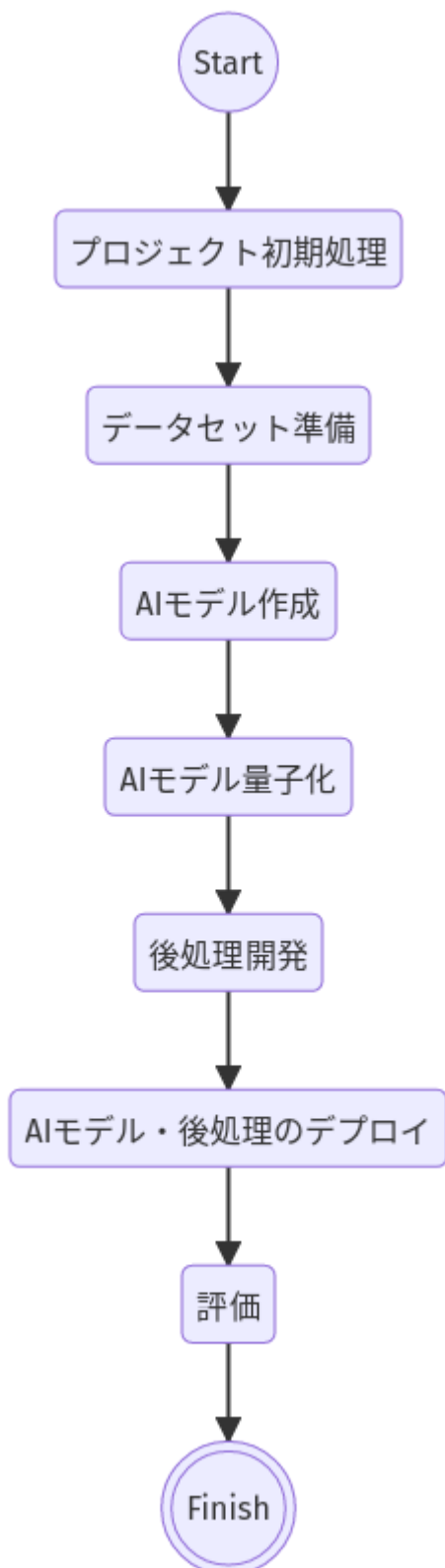
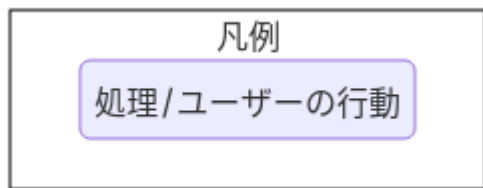
コンテナに含まれる各機能の具体的内容に関しては、本書ではなく各機能の機能仕様書にて記載する。

- AIアプリケーション開発に必要な情報を取得できる
 - AIアプリケーション開発の各ワークフローのドキュメントを閲覧できる
 - 機能仕様書を閲覧できる

Others Exclusive conditions / specifications

- Dockerイメージは提供しない
- エッジAIデバイスのファームウェアのビルド環境は提供しない
- サンプルのCloud Appは参照となるリンクを提供する

AIアプリケーション開発ワークフローと提供する機能



SDKが提供する機能

ワークフロー	提供物(ドキュメント)	提供物(実行環境、サンプル)
プロジェクト初期処理	<ul style="list-style-type: none">• Console for AITRIOS 手順	-
データセット準備	<ul style="list-style-type: none">• CVATを使用してアノテーションを実施する手順• Console for AITRIOS 手順	<ul style="list-style-type: none">• CVATを使用できる環境• COCOから画像をダウンロードするNotebook• CVATからデータセットをインポート・エクスポートするNotebook• データセットをTrain用/Validate用に分割するNotebook
AIモデルの作成	<ul style="list-style-type: none">• AIモデル（Image Classification）を転移学習する手順• Console for AITRIOS 手順	<ul style="list-style-type: none">• AIモデル（Image Classification）を転移学習するサンプルNotebook
AIモデル量子化	<ul style="list-style-type: none">• ユーザーが作成したAIモデル（Image Classification）をMCTを使用して量子化する手順• ユーザーが作成したAIモデル（Image Classification）の量子化前後の精度評価する手順	<ul style="list-style-type: none">• MCT量子化できる環境• AIモデルの評価環境• AIモデル（Image Classification）を量子化するサンプルNotebook• 下記のAIモデル（Image Classification）を評価するサンプルNotebook<ul style="list-style-type: none">◦ Keras◦ TFLite◦ TFLite（量子化済み）

ワークフロー	提供物(ドキュメント)	提供物(実行環境、サンプル)
後処理開発	<ul style="list-style-type: none"> 後処理を実装、デバッグし、Wasmファイルにビルドする手順 	<ul style="list-style-type: none"> 後処理をWasmビルドする環境 後処理のサンプルコード (C、C++) 後処理コードをデバッグする環境
AIモデル・後処理をConsole for AITRIOSにインポート	<ul style="list-style-type: none"> Notebookを使ってAIモデル・後処理をConsole for AITRIOSにインポートする手順 Console for AITRIOS 手順 	<ul style="list-style-type: none"> AIモデル・後処理をConsole for AITRIOSにインポートする Notebook
AIモデル・後処理をエッジAIデバイスにデプロイ	<ul style="list-style-type: none"> Notebookを使ってAIモデル・後処理をエッジAIデバイスにデプロイする手順 Console for AITRIOS 手順 	<ul style="list-style-type: none"> AIモデル・後処理をエッジAIデバイスにデプロイする Notebook
評価	<ul style="list-style-type: none"> Console for AITRIOS 手順 	-

その他機能	提供物(ドキュメント)	提供物(実行環境、サンプル)
バージョン管理	<ul style="list-style-type: none"> バージョン管理例 	-

コンテナのフォルダ構成

```

/tutorials
  /_common
  /1_initialize
  /2_prepare_dataset
  /3_prepare_model
  /4_prepare_application
  /5_evaluate
/docs/development-docs
/.devcontainer
/README.md

```

6. 操作性仕様、画面仕様

前提条件

- Codespacesの場合、Codespacesを使用できる状態になっていること
 - Codespaces (VS Code desktop) の場合、[VS Code Codespaces extension](#)をインストールしていること
- Local PCでVS Codeを利用する場合、[VS Code Remote Development Extension Pack](#)をインストールしていること

コンテナの起動

下記手順により、開発環境を起動する。

- Codespaces (Browser)
 1. SDKのリポジトリにおいて [Code] の [Codespaces] タブから [Create codespace on <ブランチ名>] を押下する
- Codespaces (VS Code desktop)
 1. SDKのリポジトリにおいて、[Code] の [Codespaces] タブから[Create codespace on <ブランチ名>] を押下する
 2. Codespace起動後、Codespaceのブラウザの左下にある [Codespaces] を押下する
 3. ドロップダウンリストの中から [VS Codeで開く] を選択する
- Local PC
 1. GitHub上から本SDKのリポジトリにアクセスし、ユーザーの環境に本SDKのリポジトリをクローンし、VS Codeで開く
 2. VS Codeの左下の [×] マークを押下、または、「Ctrl + Shift + P」でコマンドパレットを開き、[Reopen in Container] を選択する



コンテナ起動途中で中断する際は、下記の手順で行う。

- Codespaces (Browser) の場合
 - ブラウザの [×] ボタンを押下する
- Codespaces (VS Code desktop) の場合、またはLocal PCでVS Codeを利用する場合
 - VS Codeの [×] ボタンを押下する



コンテナ起動の進捗を確認する際は、下記の手順で行う。

- Codespaces (Browser) の場合
 - Codespacesのブラウザにおいて [View logs] が表示されたら押下する
- Codespaces (VS Code desktop) の場合、またはLocal PCでVS Codeを利用する場合
 - VS Code画面右下のNotificationから [Starting Dev Container (show log)] を押下する

AIアプリケーション開発に必要な情報の取得

下記のドキュメントを参照できる。

- AIアプリケーション開発の各ワークフローの手順書 (README)
 1. リポジトリトップの **README.md** から、[コンテナのフォルダ構成](#)の **tutorials** フォルダの **README.md** ヘリンクで遷移する
 2. **tutorials** フォルダの **README.md** から **1_initialize** といった各機能フォルダ配下の **README.md** ヘリンクで遷移する
- 機能仕様書
 1. リポジトリトップの **README.md** から機能仕様書ヘリンクで遷移する

7. 目標性能

- ユーザビリティ
 - SDKの環境構築完了後、追加のインストールなしに、AIアプリケーションを開発するためのコンテナが利用できること
 - コンテナ環境内をVS Code UIで操作できること

8. 制限事項

- CodespacesやLocal PCのスペックによっては、SDKで提供する機能が正常に動作しない場合がある
 - Codespacesの場合、Machine typeが4-core以上を推奨

9. その他特記事項

- SDK内で定義するエラーコード、メッセージはなし
- コンテナ起動時のUIの応答時間について、Codespacesの場合はユーザーのネットワーク環境、Local PCの場合はユーザーのDocker動作環境に影響されるため明記しない
 - ただし、Codespaces、Local PCともに、起動時は実績として10秒以内にUIの応答がある
 - 実績は下記条件にて計測
 - Codespaces: Machine type 4-coreを選択
 - Local PC: 下記スペックのマシンで起動

項目	内容
CPU	Intel® Core™ i7-8665U CPU @ 1.90GHz 2.11 GHz
RAM	16.0 GB
OS	Windows 10 バージョン 21H2
WSL2	Ubuntu-20.04

10. 未決定事項

- なし