

まばたき検出技術 .NETアプリ(Windows C#版)

SDK組み込みマニュアル



株式会社スワローインキュベート

2026年10月13日

■はじめに

- ・まばたき検出技術.NETアプリは、株式会社スワローインキュベートが提供しています。
- ・本書は、予告なしに変更されることがあります。
- ・本書を無断で、複製、転用、公衆送信、貸与等を行わないようお願いします。
- ・アプリをご利用いただくには、あらかじめ当社利用規約に同意いただく必要があります。
詳しくは営業担当までお問い合わせください。

■ご注意事項

- ・本サンプルアプリの利用条件は、「まばたき検出SDK」ライセンス規約によります。
- ・当社は本プログラムに関する保証は一切行っておりません。お客様の責任にてご使用ください。
本プログラム使用による問題や損害が生じた場合にも当社は何ら責任を負いませんので、あらかじめ
ご了承ください。

お問い合わせ

株式会社スワローインキュベート

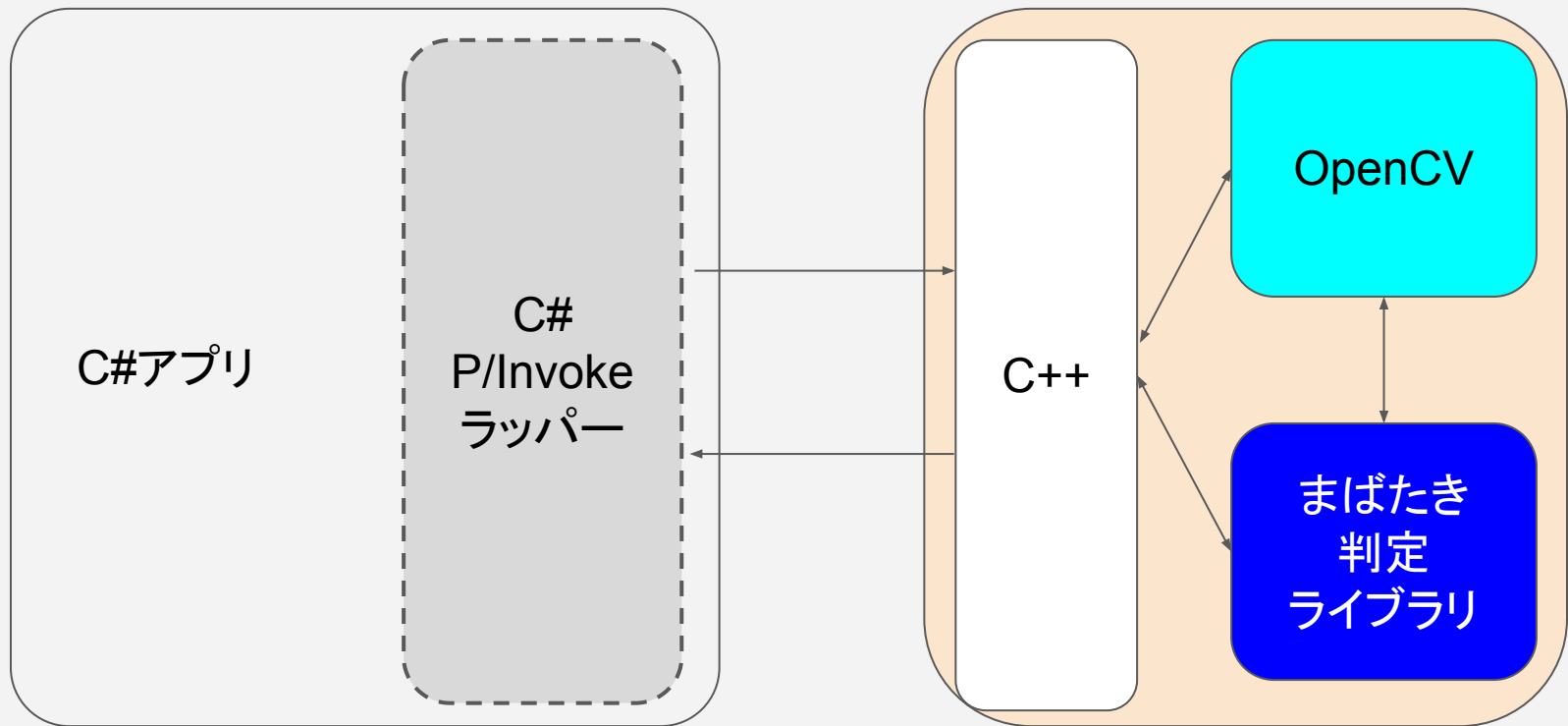
まばたき検出技術 テクニカルサポート窓口

MAIL: support@swallow-incubate.com

■まばたき検出 SDK組み込み時の構成

当社のdll形式で提供されるまばたき検出 SDKは、C++インターフェース (C++11互換)となっています。
SDKの各インターフェースを呼び出すためには、C#アプリから、C# P/Invokeラッパーを介して呼び出していただく必要があります。

また本SDKはOpenCVにも依存しているため、当社より配布予定の OpenCVライブラリも同梱しております。C#からの呼び出し方法はサンプルアプリ(OpenCvSharp利用)を参照してください。



同梱内容

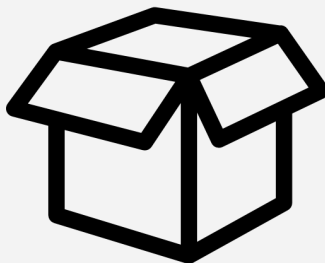
■同梱内容

以下のファイルが提供時の主要なファイルとなります。



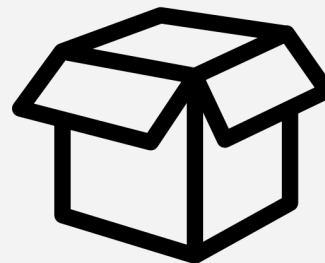
BlinkDetectDotNetApp.sln

まばたき検出
.NETアプリ



動的リンクライブラリ
Blink.dll / Blink.cs

まばたき検出
ライブラリ



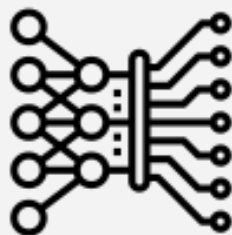
動的リンクライブラリ
opencv_world4100.dll ほか

OpenCVライブラリ



期限付きトークン

アクティベーション
トークン



ONNXファイル

深層学習
モデル



ご利用マニュアル
(本書)

ライブラリ仕様

■ご利用環境

現在のバージョンでは、以下のご利用環境に対応しています。

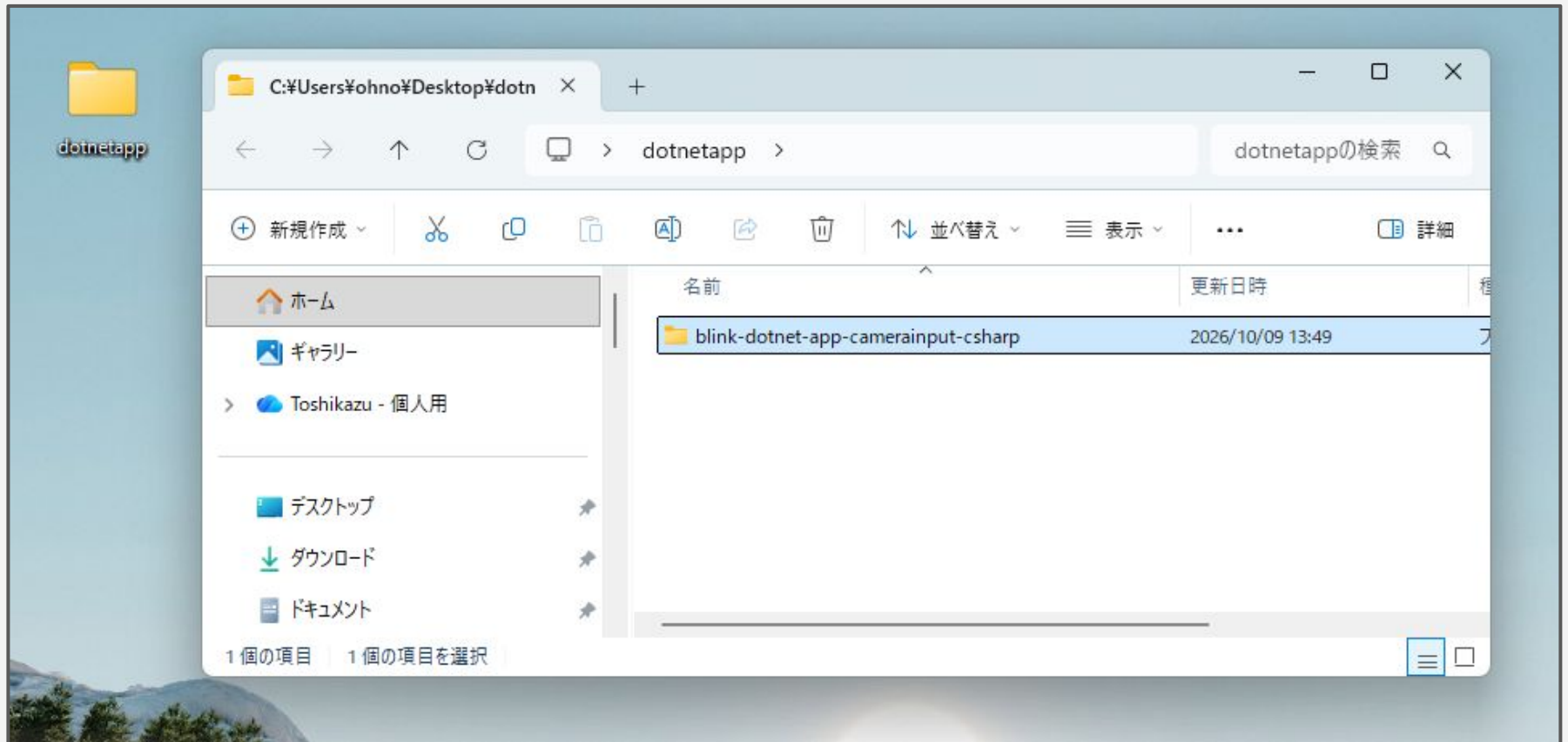
項目	内容
プログラム言語	C#(.NET 8・Windows Forms)
開発環境	Visual Studio 17 2022
ビルド環境	.NET 8 SDK
アーキテクチャ	Release x64 / Debug x64
占有メモリ	実行中 約250MB
依存ライブラリ	まばたき検出 ライブラリ(ver.1.1.0) OpenCV 4.10.0(opencv_world4100・SDKに同梱)
接続カメラ	USBカメラ(UVCタイプ) / ビルドインカメラ

※その他の環境でのご利用を希望される場合は、お問い合わせください。

SDK組み込み手順

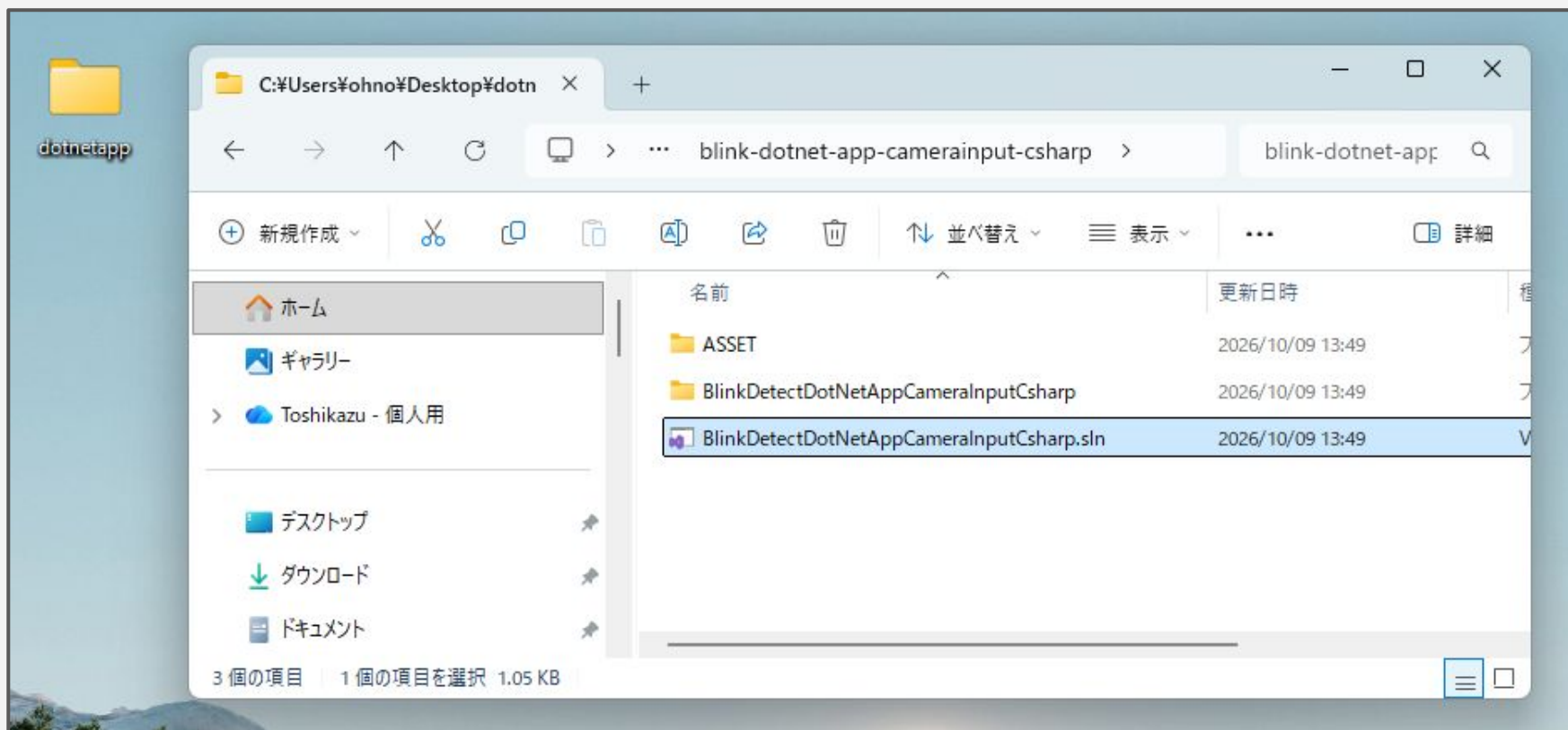
■ご利用手順①

ご提供した.NETアプリフォルダを任意の場所に設置してください。
本マニュアルでは、デスクトップに置いたものとして説明をします。



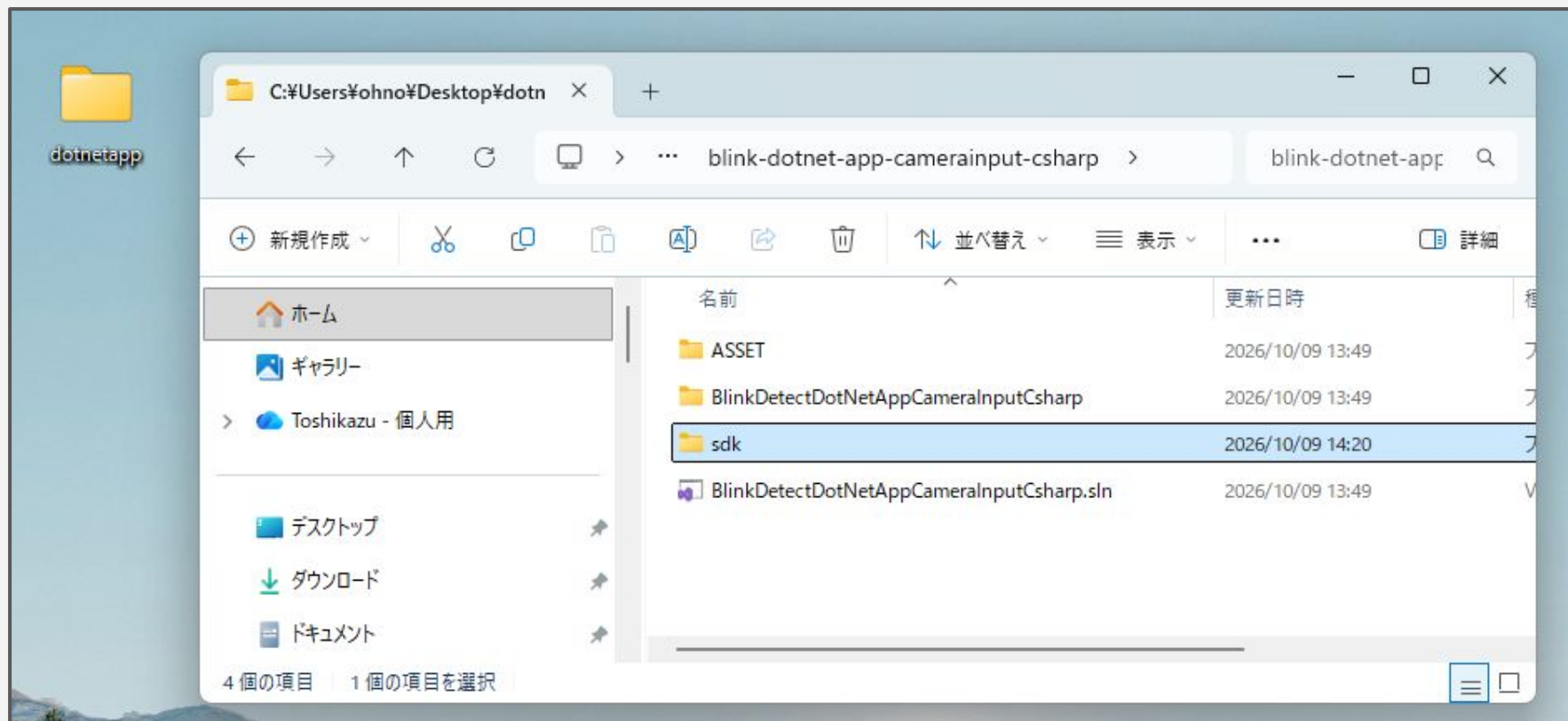
■ご利用手順②

まばたき検出アプリ.NETアプリのフォルダを開くと、Visual Studio向けソリューションファイル(.sln)とソースコード、ASSETフォルダが同梱されています。



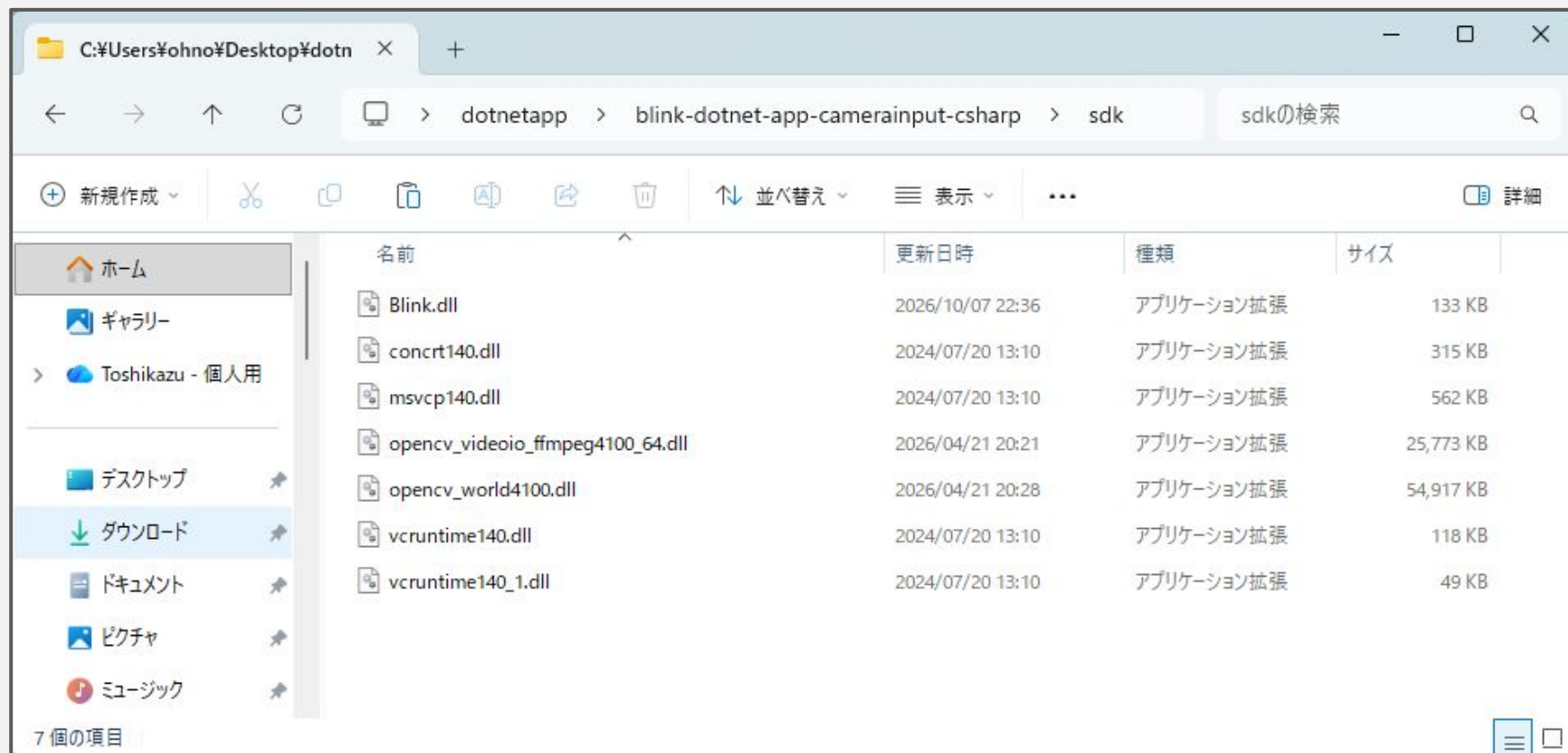
■ご利用手順③

当社よりあらかじめ提供している各種ライブラリを設置するためのディレクトリを用意していきます。プロジェクトファイル直下に「sdk」という名前のフォルダを作成してください。



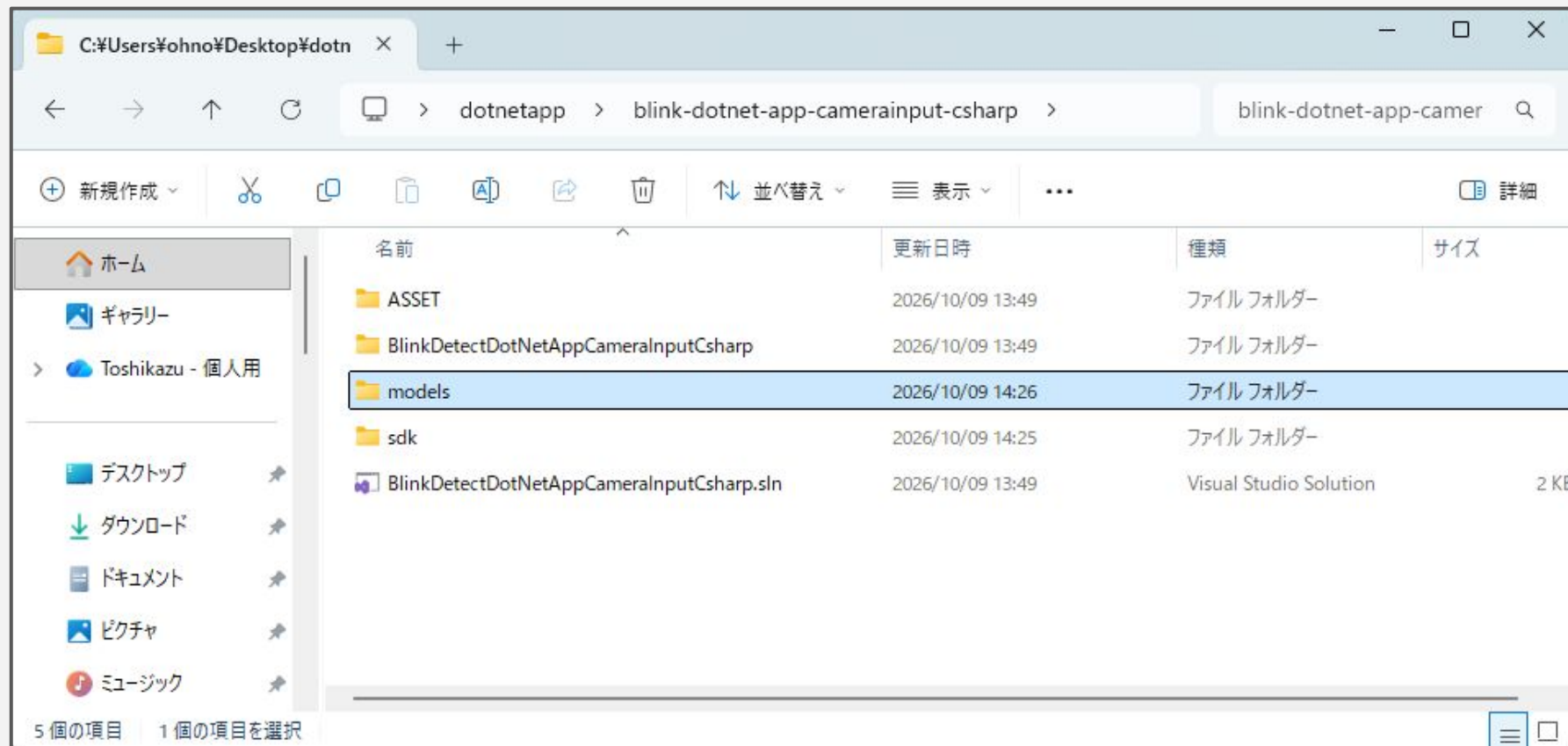
■ご利用手順④

「sdk」フォルダの中に、まばたき検出ライブラリ、OpenCVライブラリ及びランタイムライブラリのDLLファイルをすべて設置してください。以下のフォルダ構成になっていればOKです。



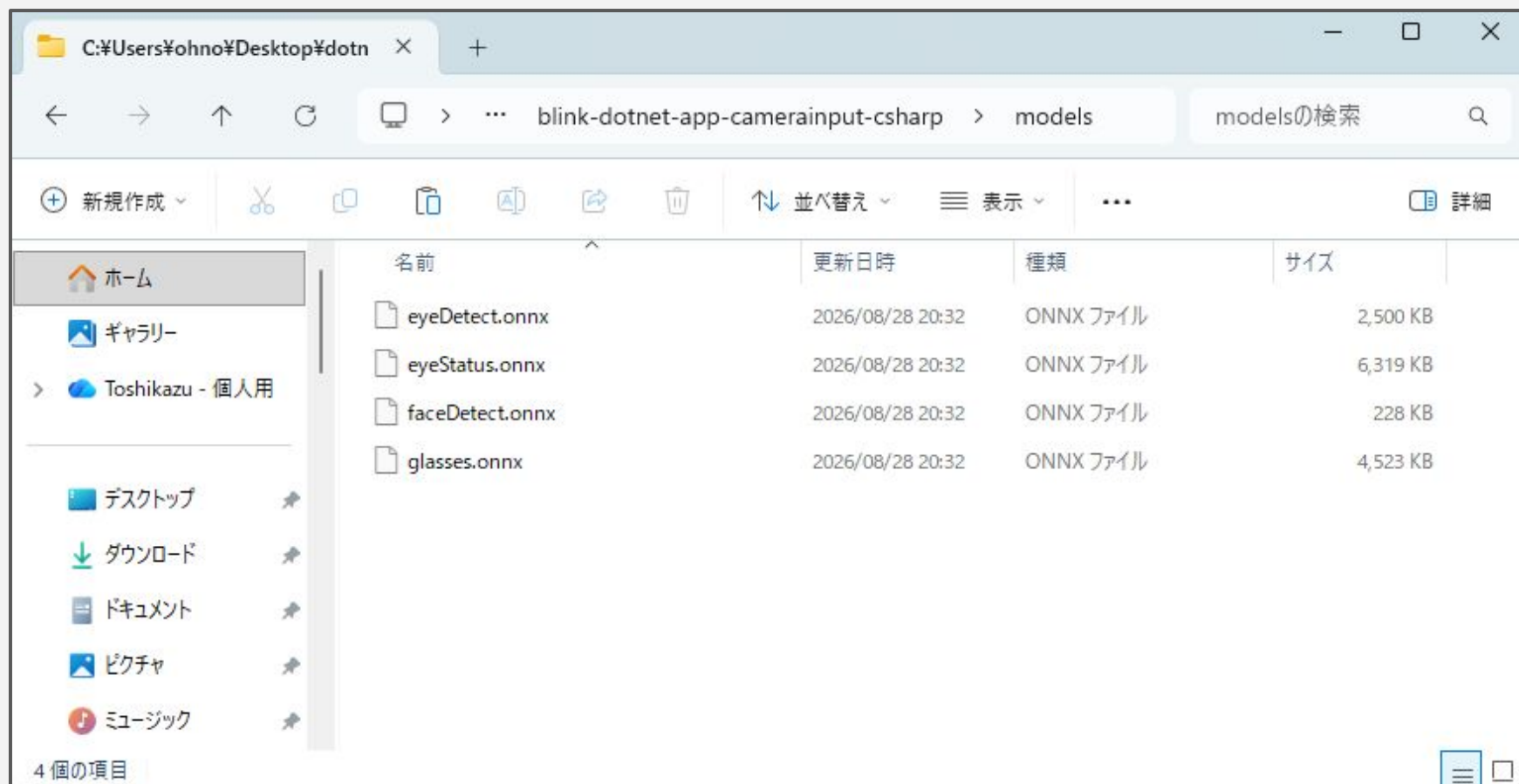
■ご利用手順⑤

続いて、当社よりあらかじめ提供している深層学習モデルファイルを設置するためのディレクトリを用意していきます。プロジェクトファイル直下に「models」という名前のフォルダを作成してください。



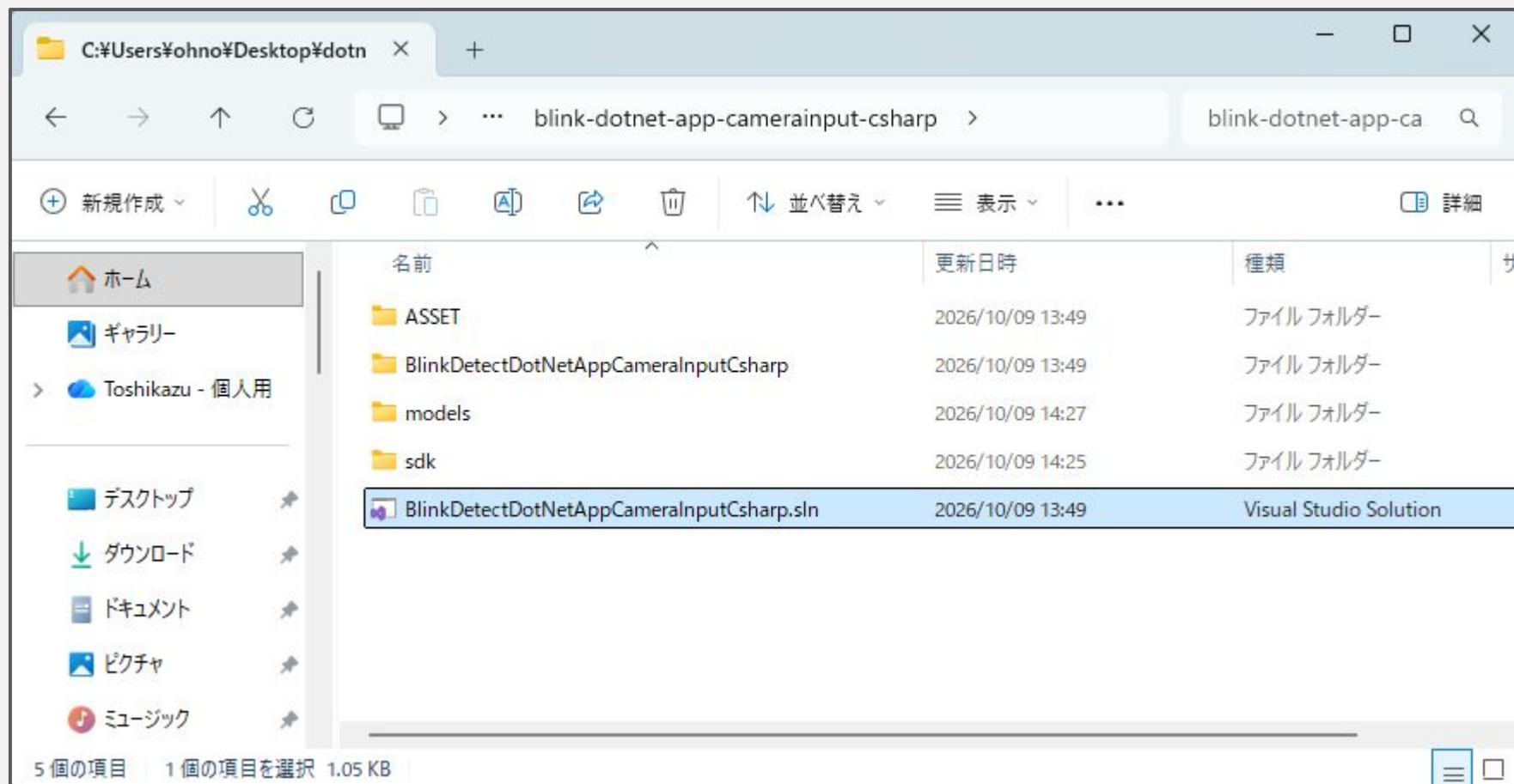
■ご利用手順⑥

「models」フォルダの中に、深層学習モデルファイル(onnx形式)をすべて設置してください。以下のフォルダ構成になっていればOKです。



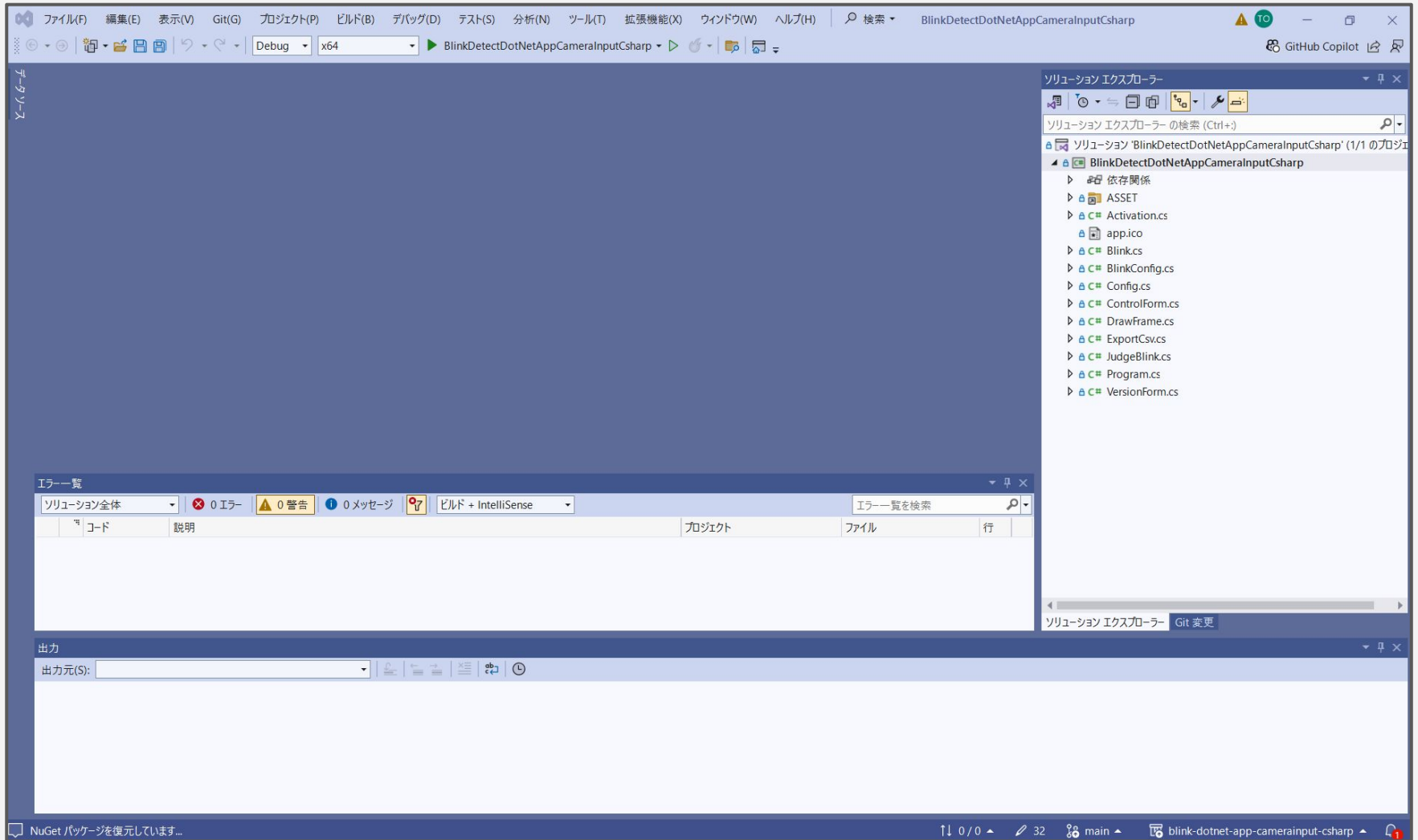
■ご利用手順⑦

ここまで準備ができたなら、お使いのマシンでVisual Studioが使える状態になっていることを確認の上、slnファイルをダブルクリックで開いてください。



■ご利用手順⑧

まばたき検出アプリ.NETアプリをVisual Studioで開くと、
以下のようなウィンドウが立ち上がります。



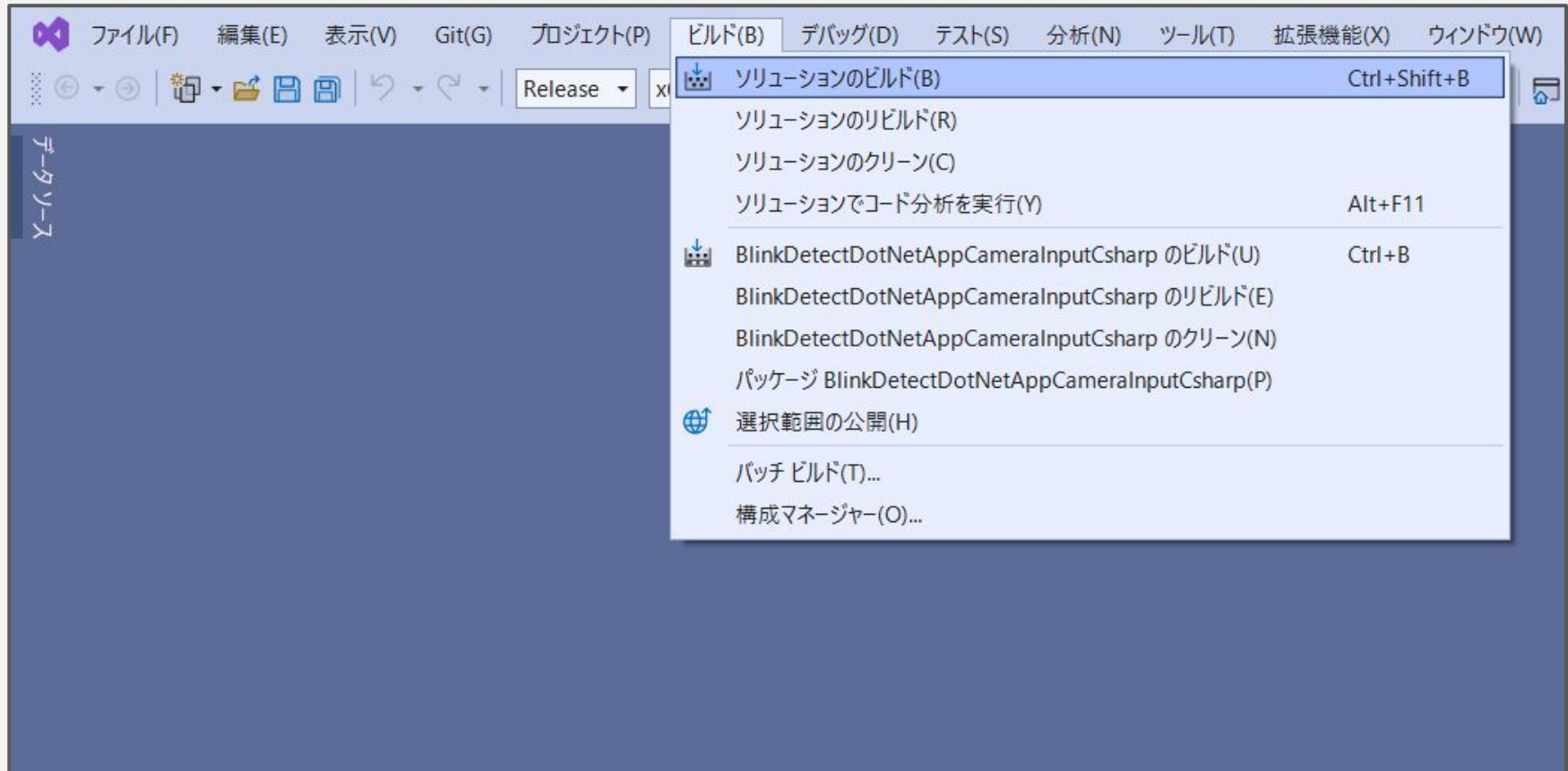
■ご利用手順⑨

今回は、x64アーキテクチャのReleaseビルドを用いるものとします。
Visual Studioのヘッダー部分で、Release / x64に変更しておいてください。



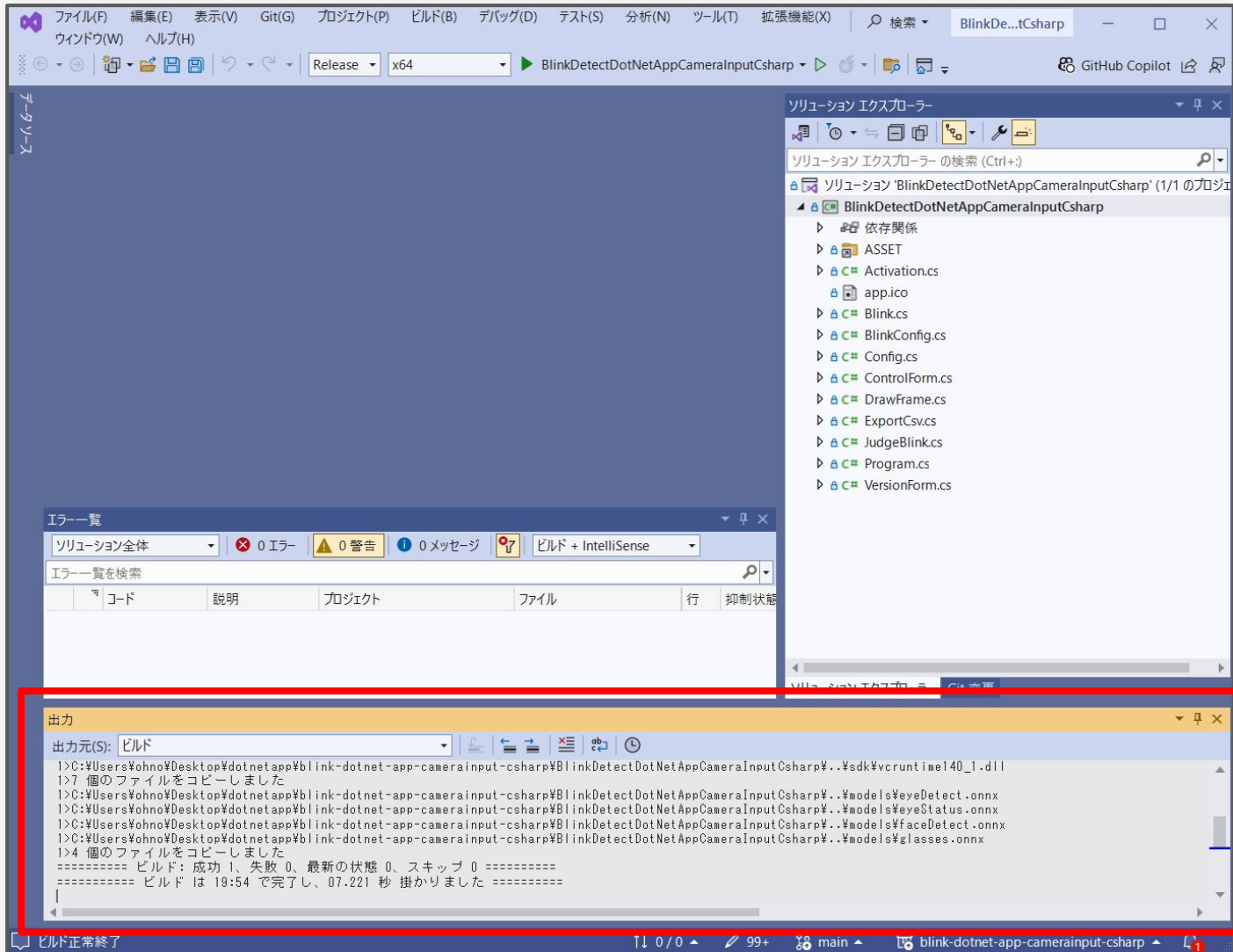
■ご利用手順⑩

ビルド準備ができましたので、Visual Studio ヘッダーより、「ビルド(B)」→「ソリューションのビルド(B)」をクリックして、ビルドを開始します。



■ご利用手順⑪

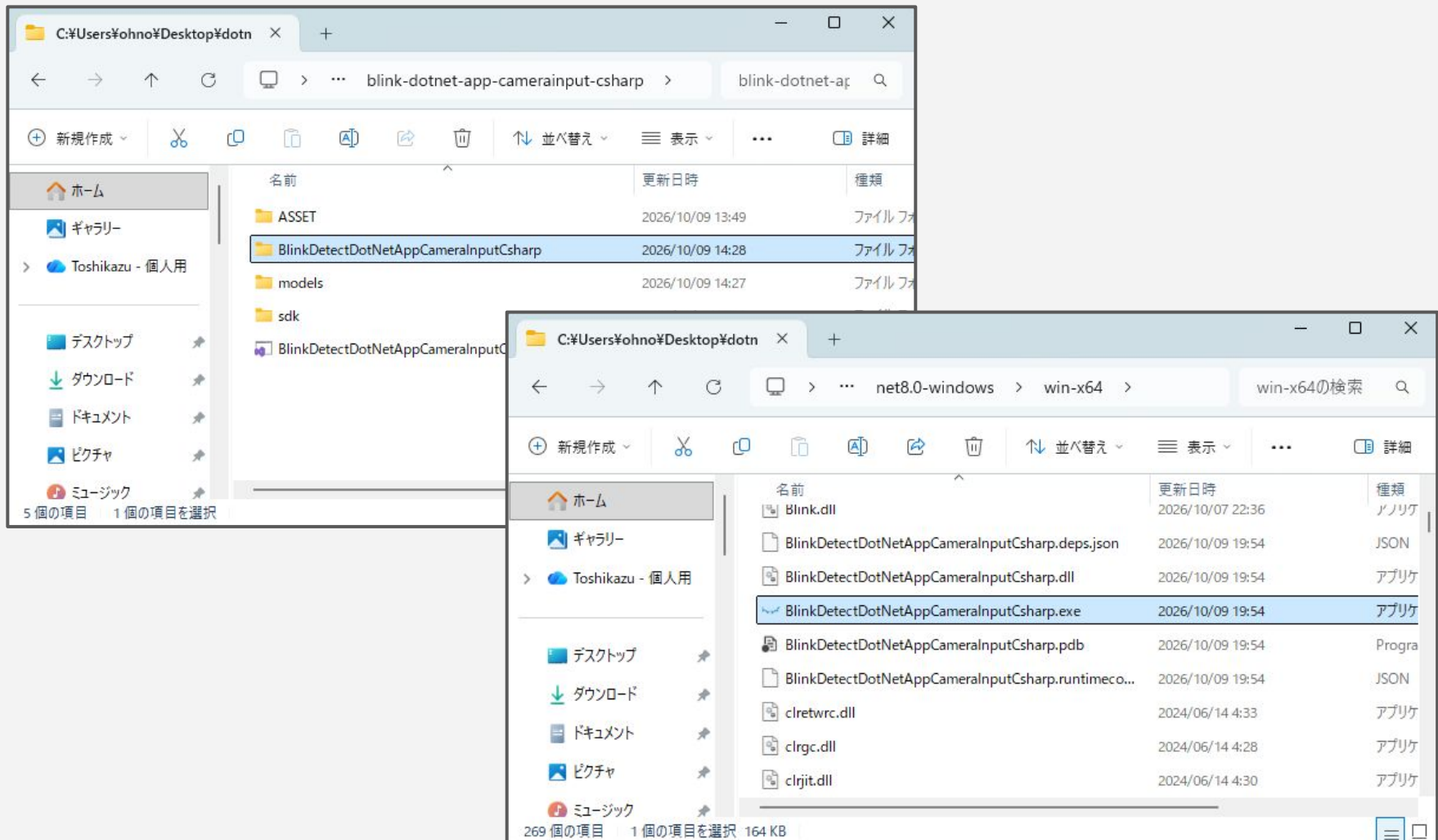
ビルドが問題なく完了すると、正常終了が1、失敗が0という表示が出力ウィンドウに出て、無事にビルドが成功していることがわかります。



■ご利用手順⑫

ビルドが成功すると、以下の場所に、アプリ実行用のexeファイルが生成されています。

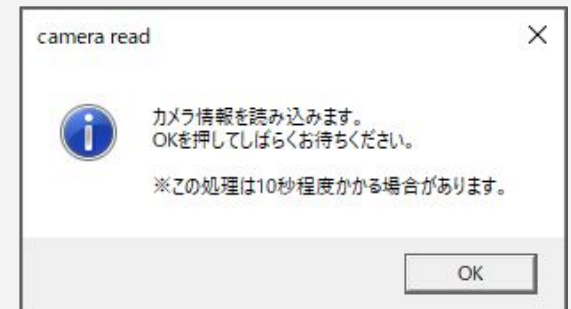
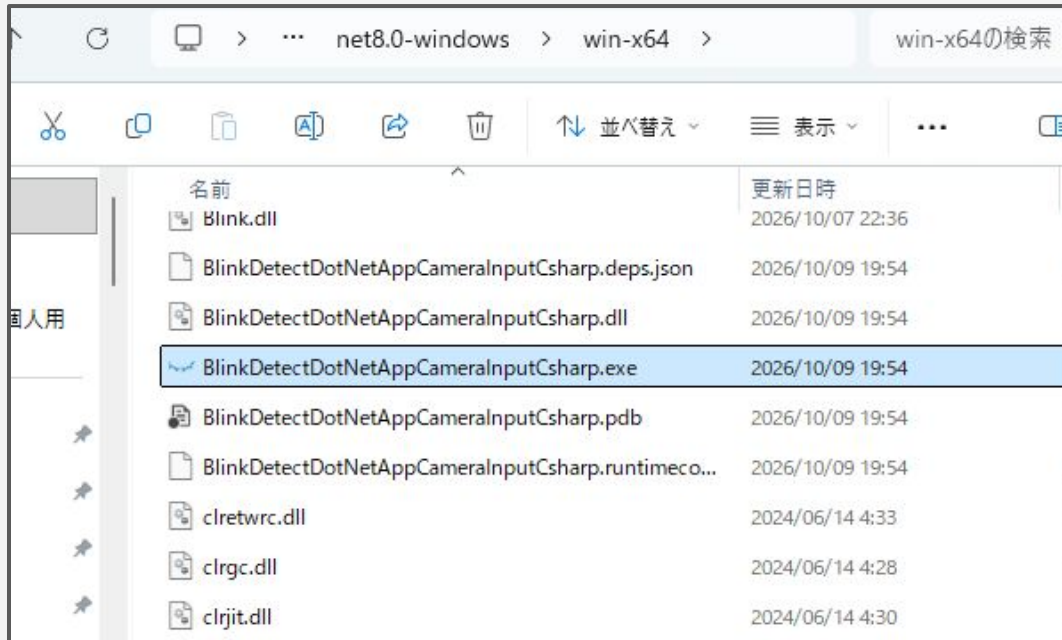
[BlinkDetectDotNetAppCameraInputCsharp]→[bin]→[x64]→[Release]→[net8.0-windows]→[win-x64]



■ご利用手順⑬

実行準備はこれで完了です。

exeファイルをクリックすると、USBカメラの読み込みを開始します。



■ご利用手順⑭

以下のウィンドウが立ち上がれば、SDKの組み込みは完了です。
ソースコードを編集し、お好みに応じてカスタマイズを行ってください。

まばたき検出アプリ - カメラ入力版

アプリについて 規約 ライセンス情報 ヘルプ

カメラ

☐ コントラスト補正

入力タイプ ☒ パソコン ☐ スマホ

センシング

☒ 検出結果を描画する ☒ 顔枠 ☒ 目枠 ☒ 情報パネル

ビープ音 ☒ 鳴らす ☐ 鳴らさない

出力

☒ csv出力

☐ 動画像保存

☐ 録画(描画なし) ☐ 録画(描画あり)

トークン

ファイル項目

■ファイル項目（弊社ソースのみ）

ファイル名	内容
ControlForm.cs / .Designer.cs	GUIウィンドウに関するプログラムです
DrawFrame.cs	検出ウィンドウの描画に関するプログラムです
ExportCsv.cs	検出ログのcsv出力に関するプログラムです
Program.cs	アプリの起動に関するMain関数のあるプログラムです
Activation.cs	アクティベーショントークンを読み込むプログラムです
JudgeBlink.cs	センシング情報と時系列データをもとに、まばたきの有無を判定するプログラムです。
VersionForm.cs / .Designer.cs	バージョン情報を表示するGUIウィンドウに関するプログラムです
Config.cs	設定ファイルであるxmlファイルを生成するためのプログラムです
Blink.cs	まばたき検出ライブラリ(Blink.dll)をC#から呼び出すためのラッパープログラムです
BlinkConfig.cs	モデルファイルのパスや判定パラメータの既定値を定義するプログラムです