

視線検出技術 Windows版アプリ(ファイル入力版)

ご利用マニュアル

ディープラーニング対応版



株式会社スワローインキュベート

2022年11月21日

■はじめに

視線検出技術 Windows版アプリは、株式会社スワローインキュベートが提供しています。
本書に基づき、当アプリをご利用いただく前に、以下のご注意事項を十分に読んだ上で、
ご利用いただきますようお願いいたします。

■ご注意事項

- ・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。

お問い合わせ

株式会社スワローインキュベート

視線検出技術 テクニカルサポート窓口

TEL: 029-886-9912 MAIL: support@swallow-incubate.com

■アプリ更新履歴

バージョン	日時	変更内容サマリー
ver.1.0.0	2020年11月30日	初版
ver.1.1.0	2021年01月22日	・プライバシー保護機能の追加
ver.1.2.0	2021年12月13日	・ディープラーニングモデル導入 ・「コントラスト補正」機能追加 ・プロット項目に「出力値ウィンドウ」を追加 ・「居眠り検知モード」の追加 ・ファイル入力版のリリース
ver.1.2.1(本稿)	2022年11月21日	・視線検出SDK ver.1.6.0対応 ・軽微な修正

このアプリについて

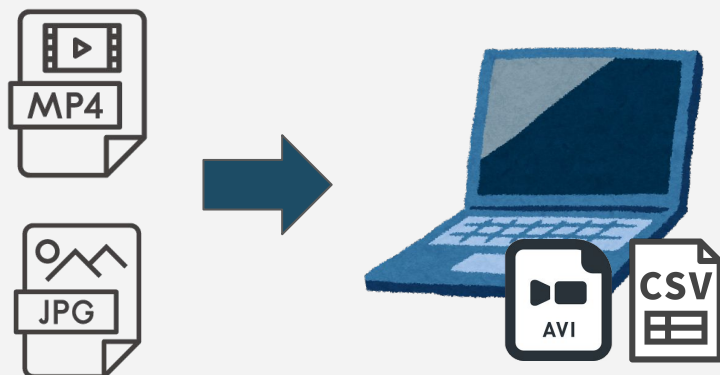
■アプリの種別

このデモアプリには、「ファイル入力版」と「カメラ入力版」があります。

「ファイル入力版」では、お手持ちの動画ファイルまたは画像ファイルを入力して検出を行います。「カメラ入力版」では、PCのデバイスカメラまたはUSBカメラを接続していただき、リアルタイムに検出を行うことができます。

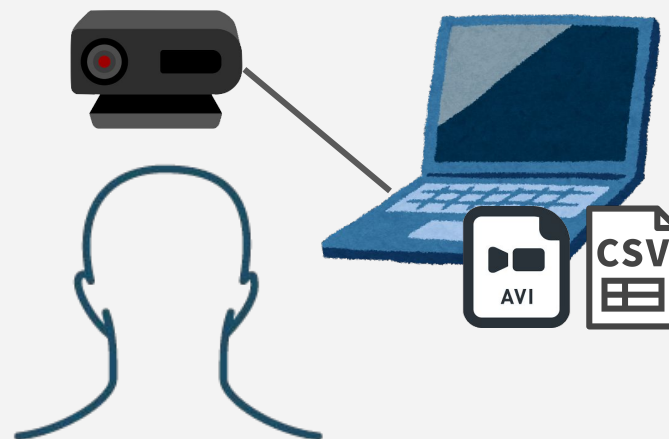
それぞれのアプリでは、検出結果をcsv形式でログ出力したり、動画・画像として保存することが可能です。アプリは、exe形式のファイルをダブルクリックするだけで開始できます。

ファイル入力版



顔が映された動画・画像ファイルをアプリに入力させることで検出が始まります。

カメラ入力版



USBカメラを接続し、カメラの前に顔を映すことで検出が始まります。

■アプリの種別

このアプリは、SDKのご購入を検討されているお客様向けの体験版をご用意しておりますが、このアプリをご購入いただくことも可能です。

体験版・単体購入版・セット購入版は機能はすべて同じで、動作には都度アクティベーションが必要となります。体験版は、txt形式のトークンファイルによるアクティベーションを行いますが、購入版は動作には USBドングルの接続が必要となりますので、USBポートが必要です。(DaaSなどをご利用されている企業様はご注意ください)

アプリ種別	形態	入力タイプ	型番	単価(税抜)	付属品	納品形態
視線検出技術 Windowsアプリ	体験版	カメラ入力版	SW-ET21-T001	無料 (1ヶ月間)	アプリ一式 トークンファイル	データ納品
		ファイル入力版	SW-ET21-T002			
視線検出技術 Windowsアプリ	単体 購入版	カメラ入力版	SW-ET21-A001	35,000円	アプリ一式 専用USBドングル1本	データ納品 または DVD納品
		ファイル入力版	SW-ET21-A002	35,000円	アプリ一式 専用USBドングル1本	

※各アプリともカメラは付属しておりませんので、ご利用企業様にてご用意ください。

※アプリの再販売は禁止させていただいております。

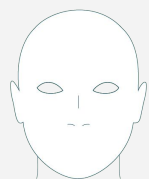
※付属モデルファイルの再利用・販売・複製・改変などは固く禁止させていただいております。

■このアプリでできること

視線検出技術SDKでは、ディープラーニングや機械学習を用いて、顔・目の特徴点などを検出し、それらの情報を元に「顔向き・目向きの検出」や「カメラから目までの視距離を推定する」ことが可能です。これらの検出・判定処理を、WEBカメラなどの可視光RGBカメラを用いてできることが特徴となります。

DL = 深層学習ベース

特徴点検出



DL

顔位置検出



DL

目位置検出



DL

虹彩位置検出



DL

メガネ・サングラス
判定



DL

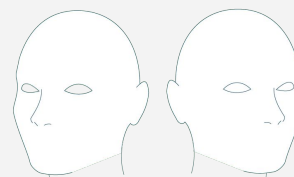
まぶた位置検出



DL

目尻目頭検出

顔向き・目向き検出



DL

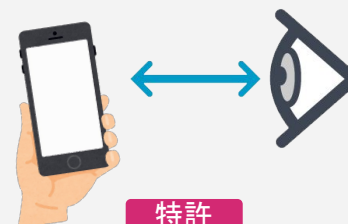
顔向き検出



DL

目向き検出

視距離推定



特許

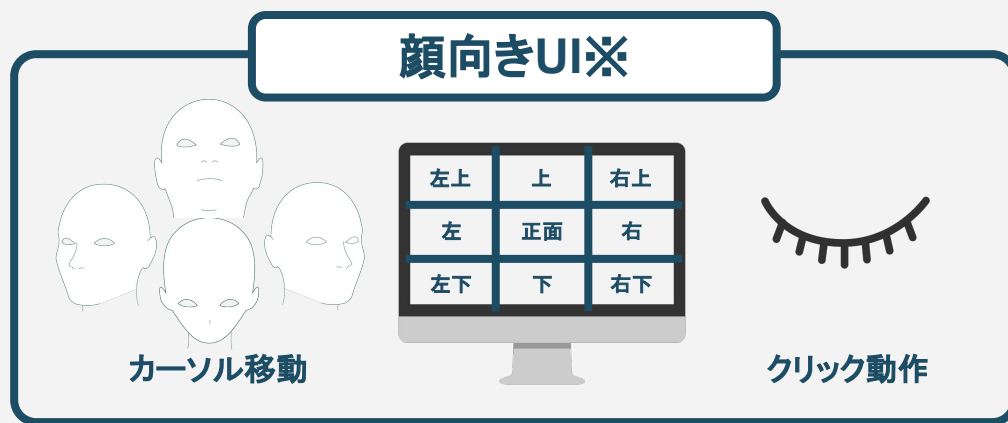
視距離推定

■このアプリでできること

本アプリでは、視線検出 SDKにより取得した視線基礎情報を元に、いくつかの応用機能を体感いただけるモードが用意されています。視線先推定モードは、算出された顔向きベクトル・目向きベクトルと視距離との関係から、視線方向の推定と、およその視線点などの推定を行います。

顔向きUI・目向きUIモードでは、顔向き・目向きをマウス移動、まばたきをマウスクリックとして見立てた、非接触ユーザーインターフェースとしての使い方を体感いただくことができます。

※ファイル入力版には顔向きUI・目向きUIモードはありません

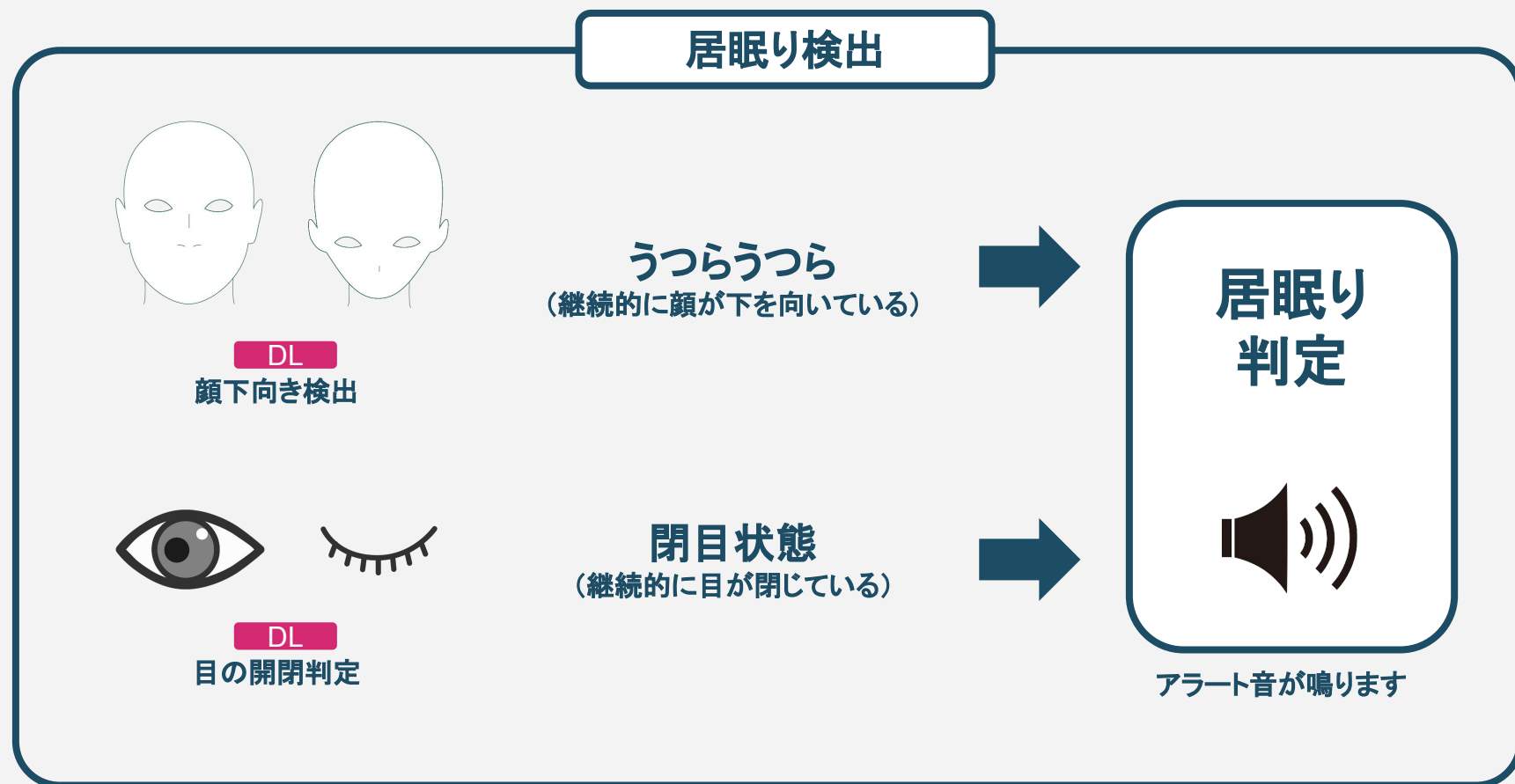


■このアプリでできること

視線検出技術SDKを使ったこのアプリでは、動画などの連続的な時系列データを使い、簡単な居眠りの検出を行うことが可能です。顔が検出されている場合、継続して顔向きが下向きであることや、目が継続的に閉じている場合に、居眠りとして判定し、ログの出力やアラート音が鳴ります。

※ファイル入力版ではアラート音は鳴りません

DL = 深層学習ベース



同梱内容

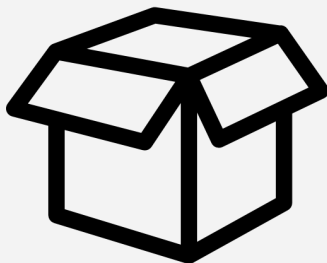
■同梱内容

以下のファイルが提供時の主要なファイルとなります。



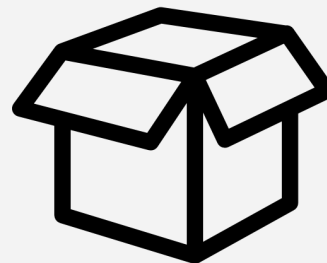
Windows exe形式
EyeTrack.exe

視線検出GUIアプリ



動的リンクライブラリ
EyeTrack.dll

視線検出ライブラリ



動的リンクライブラリ
opencv_world[VERSION].dll

OpenCVライブラリ

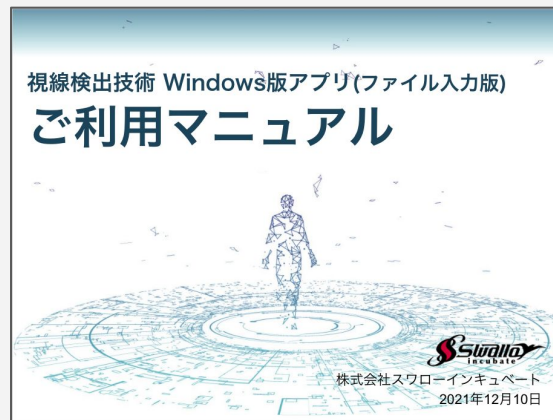


期限付きトークン
(体験版)

アクティベーションツール



USB dongle
(購入版)



ご利用マニュアル
(本書)

ほか

ファイルの撮影要件について

■必要となる撮影スペック

項目	要件
カメラ解像度	VGAサイズ(640 x 480)以上 を推奨 (640 x 480 / 1280 x 720 / 1280 x 960 / 1920 x 1080 など)
撮影距離	30cm～1.5m程度まで
視野角	50° ～ 110°程度(魚眼レンズのような丸みを帯びていないレンズ推奨)
推奨要件	本アプリ(カメラ入力版)での撮影・保存したファイル入力を推奨します。

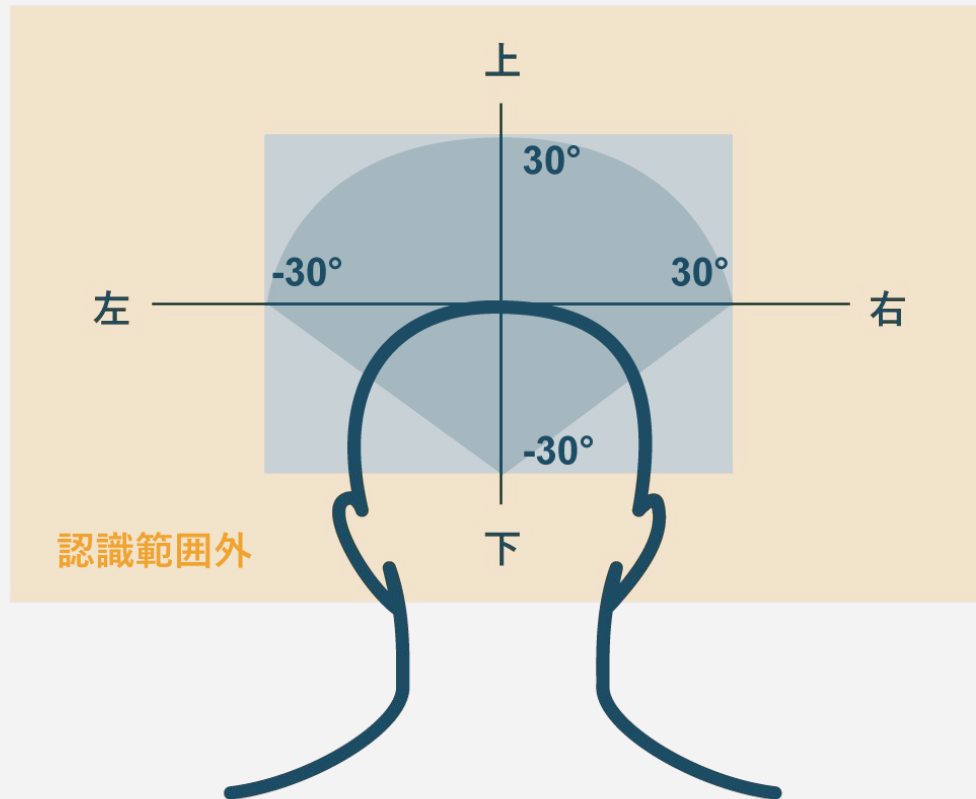
■必要となる撮影要件

項目	要件
顔横幅サイズ	最低100px以上
顔領域明るさ	平均輝度値 50.0以上(8bit256階調)
目領域明るさ	平均輝度値 50.0以上(8bit256階調)

ライブラリ仕様

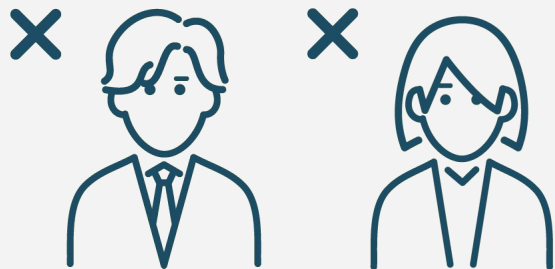
■ 라이브러리仕様 - 検出可能画角

現在のバージョンでは、検出可能な画角は、上下左右ともに30°程度ですが顔の形状などによるため、個人差があります。

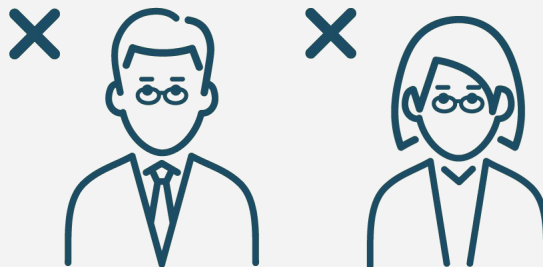


■ 라이브러리仕様 - X 検出不可となるケース

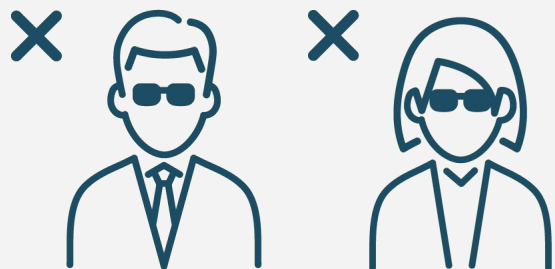
現在のバージョンでは、以下のケースで検出エラーになりやすくなります。



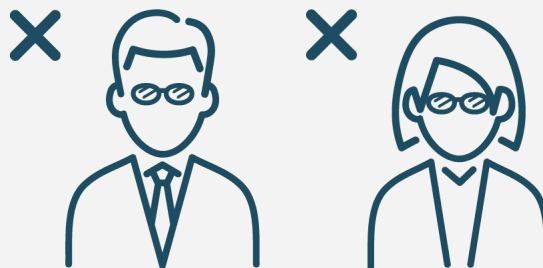
髪の毛が目にかかっている



メガネフレームが目にかかっている



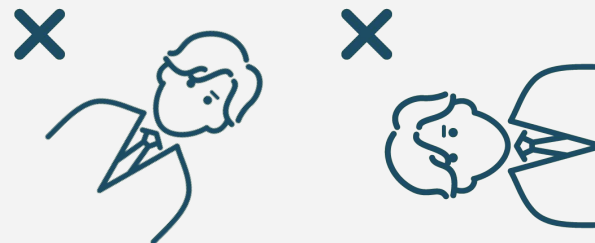
レンズが黒いサングラス



外光の映り込みが激しい



帽子を深くかぶっている



顔が傾いて撮影されている

■ライブ러리仕様 - △ 検出に影響を与える場合があるケース

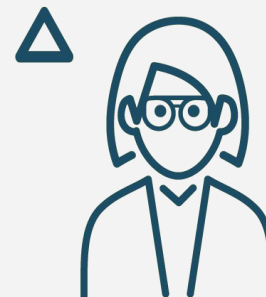
現在のバージョンでは、以下のケースで検出エラーとなる場合があります。



ブルーライトカットメガネ
(ブルーライト反射が強いとエラーになる場合あり)



色付き透明サングラス
(色調変化によりエラーとなる場合あり)

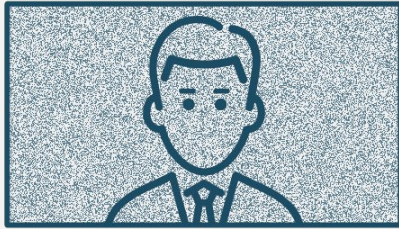


マスク
(顔検出エラーとなる場合あり)



■ライブ러리仕様 - △ 検出に影響を与える場合がある撮影状況

現在のバージョンでは、以下の撮影状況で検出エラーとなる場合があります。



×露光が不十分
(環境光の差に影響を受けます)



×オートフォーカス制御
(フォーカシング中検出エラーになりやすい)



×カメラの設置が
まっすぐではない



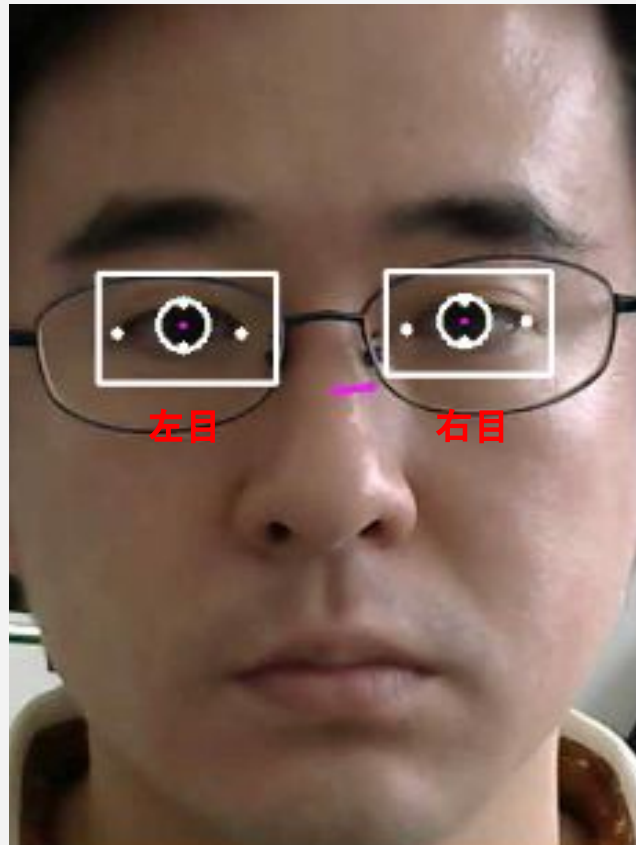
×画角から外れている
(鼻より上しか写っていない)



×画角から外れている
(片目しか写っていない)

■ 라이브러리仕様 - 機能一覧

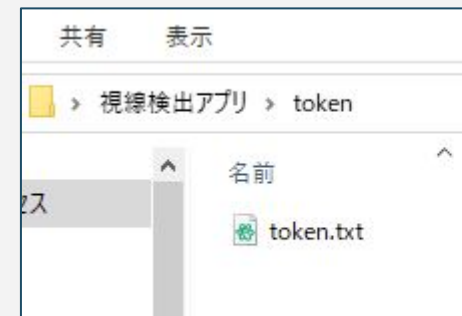
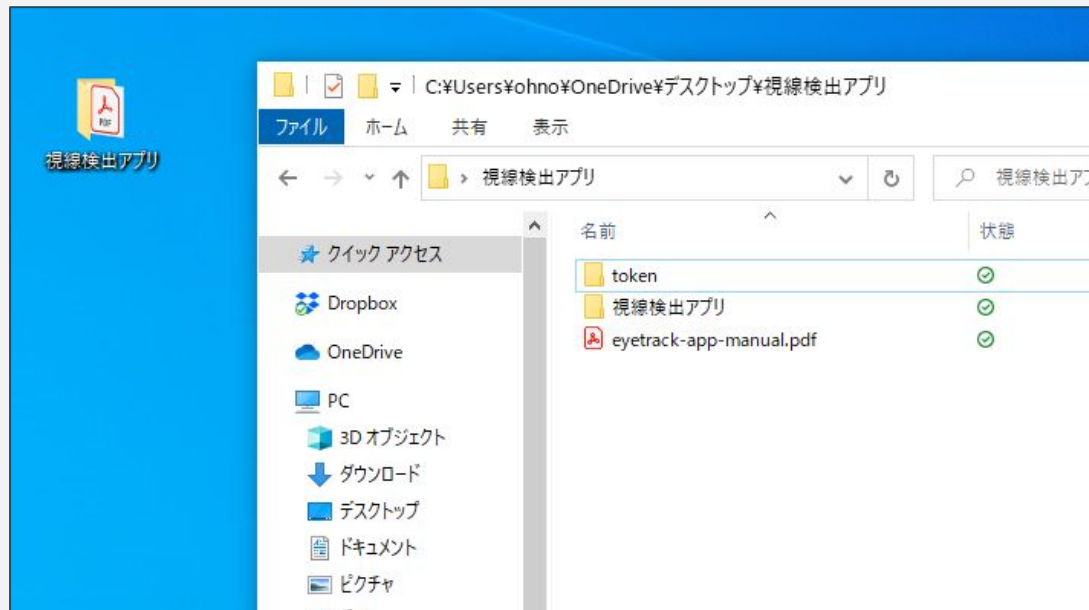
入力画像に対する左右目の判定は、
画像向かって左側の目を左目、画像向かって右側の目を右目として出力します。
虹彩、上下まぶた、目尻目頭等についても以下同様です。
アプリ内では、カメラ画像を左右反転させ、実際の左目が「左目」として出力されるようになっています。



アプリご利用手順

■ご利用手順①

本マニュアルでは、ご提供済みの視線検出フォルダをデスクトップ上に設置して進めるものとします。
体験版の場合は、tokenフォルダにアクティベーション用の txtファイルがあることを確認します。
※購入版などUSBアクティベーション版には、tokenフォルダはありません。

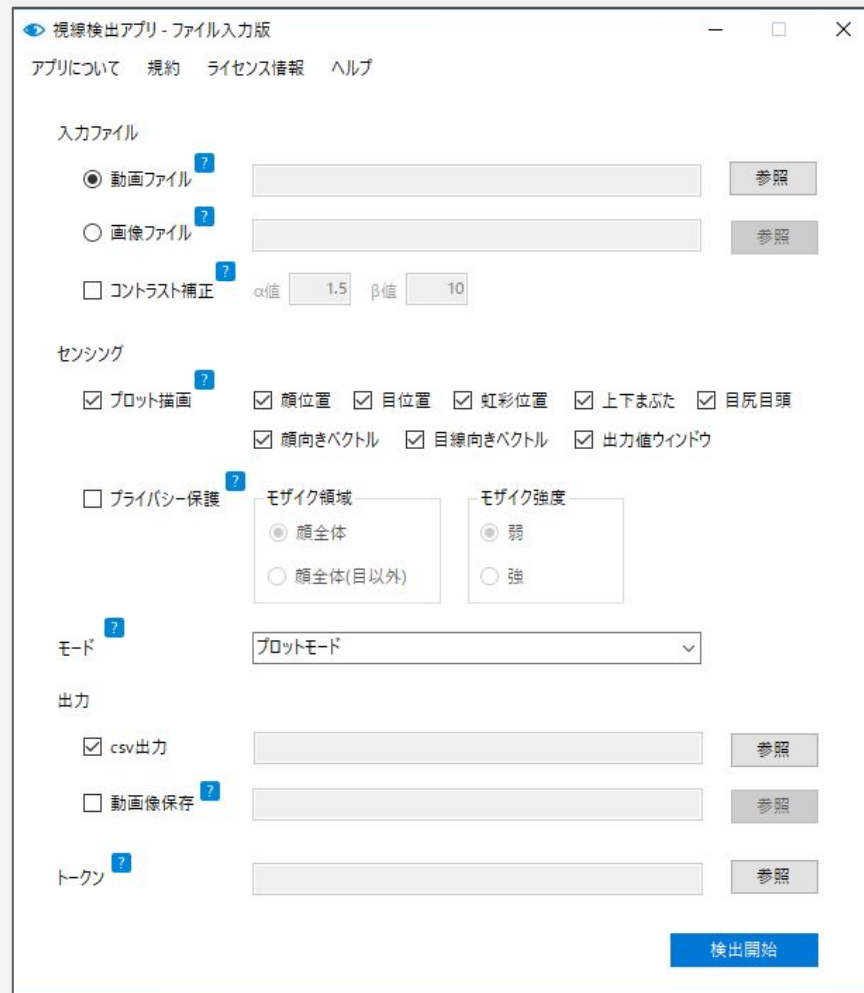
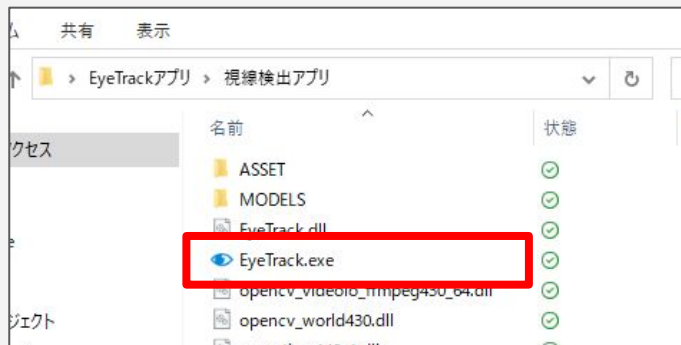


※トークンアクティベーション版のみ

■ご利用手順②

視線検出アプリフォルダを開き、内容物を確認してください。
EyeTrack.exeファイルをダブルクリックするとアプリが起動します。

※Windows Updateなどバックグラウンドで実行されている場合は処理に時間がかかる場合があります。



■ご利用手順③(トークンアクティベーション版)

アプリが起動すると、入力ファイルの指定項目、センシング項目、モード選択、ログや動画の出力を選択する画面が表示されます。各選択項目の詳細は、以下の通りとなります。

視線検出アプリ - ファイル入力版

アプリについて 規約 ライセンス情報 ヘルプ

入力ファイル

☒ 動画ファイル

☐ 画像ファイル

☐ コントラスト補正

α値1.5β値10

参照

参照

センシング

☒ プロット描画

☒ 顔位置

☒ 目位置

☒ 虹彩位置

☒ 上下まぶた

☒ 目尻目頭

☒ 顔向きベクトル

☒ 目線向きベクトル

☒ 出力値ウィンドウ

☐ プライバシー保護

モザイク領域

☒ 顔全体

☐ 顔全体(目以外)

モザイク強度

☒ 弱

☐ 強

モード

プロットモード

出力

☒ csv出力

☐ 動画像保存

参照

参照

トークン

参照

検出開始

選択項目	説明
入力ファイル	動画ファイルまたは画像ファイルのいずれかを選択し、ファイルの場所を指定します。
コントラスト補正	入力動画・画像が暗く検出がしづらい場合などに、コントラスト補正を行うことが可能です。
プロット描画	検出した顔に、視線検出で検出された各センシング位置をプロット描画します。プロットする項目はそれぞれ選択可能です。
プライバシー保護	投影フレームの検出結果の顔領域にモザイク処理をかけることにより、視線検出の対象者のプライバシーを保護します。こちらにチェックを入れると、動画像保存時にもモザイク処理がかけられた状態で保存されます。モザイクは原則顔にかかりますが、目以外を選択したり、モザイクの強弱を変更することが可能です。
モード	視線検出技術を使った以下のいずれかのモードを選択することができます。 プロットモード：顔特徴点の検出結果を確認するモード 視線先推定モード：顔向きベクトルと目線ベクトルから視線先を推定
出力(csv出力)	視線検出を行った検出ログをsv形式で出力します。出力したくない場合は、チェックボックスからチェックを外してください。出力する場合は、「参照」ボタンを押して、出力先フォルダを選択します。
出力(動画像保存)	視線検出を行った検出動画をvi形式で出力します。出力したくない場合は、チェックボックスからチェックを外してください。出力する場合は、「参照」ボタンを押して、出力先フォルダを選択します センシングの途中の様子 のキャプチャ保存をする場合にも、このチェックボックスをチェックする必要があります。
トークン	弊社より配布させていただいた、有効期限付きトークンの記載されたファイルを指定します。またトークンの有効期限が切れた場合は、開始できません。
検出開始	各選択項目の設定情報をもとに、視線検出を開始します。

■ご利用手順③(USBアクティベーション版)

アプリが起動すると、入力ファイルの指定項目、センシング項目、モード選択、ログや動画の出力を選択する画面が表示されます。各選択項目の詳細は、以下の通りとなります。

視線検出アプリ - ファイル入力版

アプリについて 規約 ライセンス情報 ヘルプ

入力ファイル

☒ 動画ファイル

☐ 画像ファイル

☐ コントラスト補正

α値1.5β値10

参照

参照

センシング

☒ プロット描画

☒ 顔位置

☒ 目位置

☒ 虹彩位置

☒ 上下まぶた

☒ 目尻目頭

☒ 顔向きベクトル

☒ 視線向きベクトル

☒ 出力値ウィンドウ

☐ プライバシー保護

モザイク領域

☒ 顔全体

☐ 顔全体(目以外)

モザイク強度

☒ 弱

☐ 強

モード

プロットモード

出力

☒ csv出力

☐ 動画像保存

参照

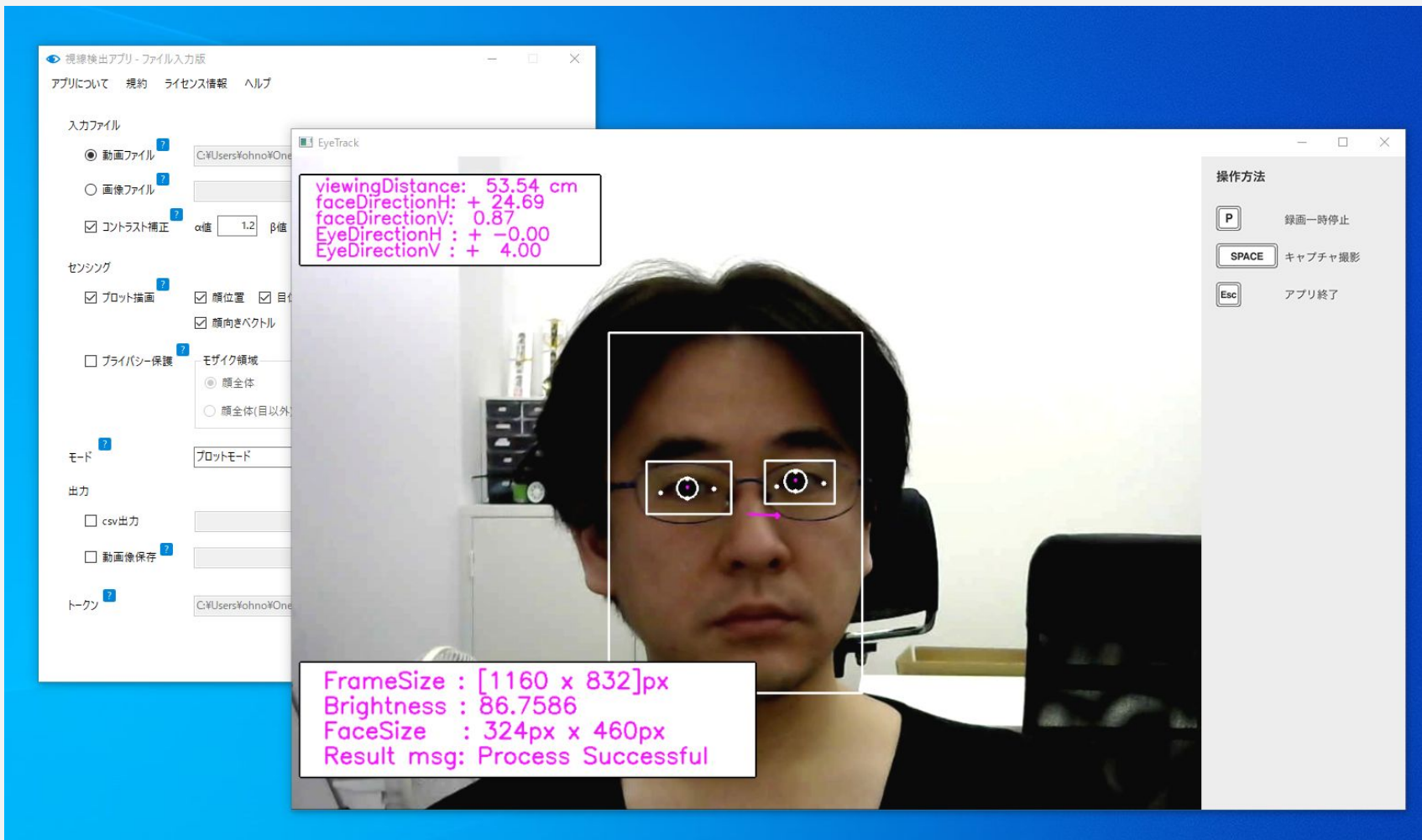
参照

検出開始

選択項目	説明
入力ファイル	動画ファイルまたは画像ファイルのいずれかを選択し、ファイルの場所を指定します。
コントラスト補正	入力動画・画像が暗く検出がしづらい場合などに、コントラスト補正を行うことが可能です。
プロット描画	検出した顔に、視線検出で検出された各センシング位置をプロット描画します。プロットする項目はそれぞれ選択可能です。
プライバシー保護	投影フレームの検出結果の顔領域にモザイク処理をかけることにより、視線検出の対象者のプライバシーを保護します。こちらにチェックを入れると、動画像保存時にもモザイク処理がかけられた状態で保存されます。モザイクは原則顔にかかりますが、目以外を選択したり、モザイクの強弱を変更することが可能です。
モード	視線検出技術を使った以下のいずれかのモードを選択することができます。 プロットモード：顔特徴点の検出結果を確認するモード 視線先推定モード：顔向きベクトルと視線ベクトルから視線先を推定
出力(csv出力)	視線検出を行った検出ログをsv形式で出力します。出力したくない場合は、チェックボックスからチェックを外してください。出力する場合は、「参照」ボタンを押して、出力先フォルダを選択します。
出力(動画像保存)	視線検出を行った検出動画をvi形式で出力します。出力したくない場合は、チェックボックスからチェックを外してください。出力する場合は、「参照」ボタンを押して、出力先フォルダを選択します センシングの途中の様子 のキャプチャ保存をする場合にも、このチェックボックスをチェックする必要があります。
検証開始	各選択項目の設定情報をもとに、視線検出を開始します。

■ご利用手順④

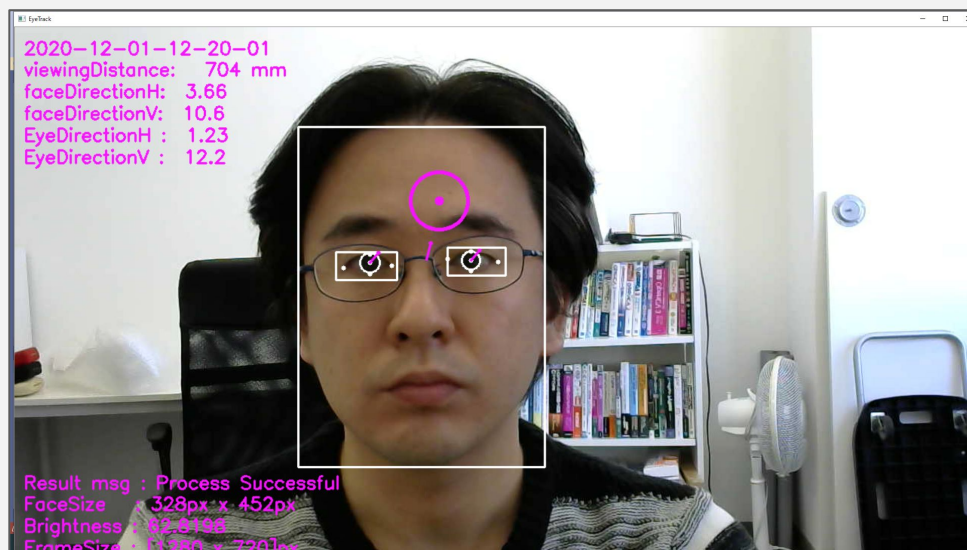
検出開始ボタンを押すと、はじめにアクティベーションが行われ、正常に認証されると以下の検出ウィンドウが表示されます。起動すると以下のようなウィンドウが立ち上がります。(動画ファイル入力・プロットモードでのウィンドウ例です)



■ご利用手順⑤(動画入力時)

動画ファイルを指定して、検出を開始すると、視線基礎情報の検出プロセスがはじまり、検出結果が表示ウィンドウに描画されます。この際、検出に成功している場合は、ウィンドウ上に検出点も合わせて描画されます。検出中に各種操作を行う場合は、操作方法ウィンドウを参考に、Windowsキーボードから、いずれかのキーを押下してください。「動画像保存」チェックボックスをチェックしていない場合は、キャプチャ撮影をしてもキャプチャ保存はされません。

画面イメージ



操作方法



一時停止



キャプチャ撮影

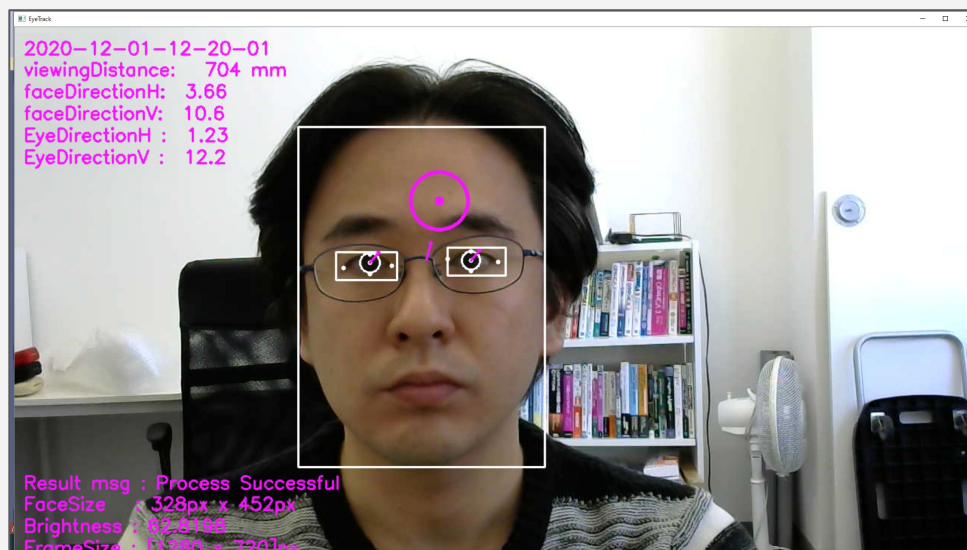


アプリ終了

■ご利用手順⑤(画像入力時)

画像ファイルを指定して、検出を開始すると、視線基礎情報の検出プロセスが始まり、検出結果がウィンドウに描画され表示されます。この際、検出に成功している場合は、ウィンドウ上に検出点も合わせて描画されます。検出を終了する場合は、Windowsキーボードからいずれかのキーを押下してください。なおウィンドウ表示中に「space」ボタンを押下することで、キャプチャ保存を行うことも可能です。(動画像保存チェック時のみ)

画面イメージ

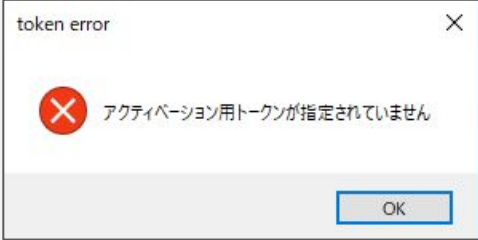
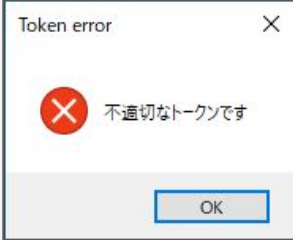
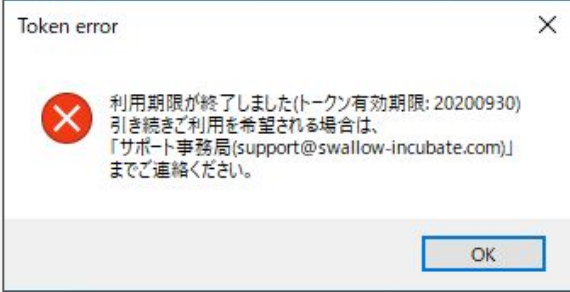
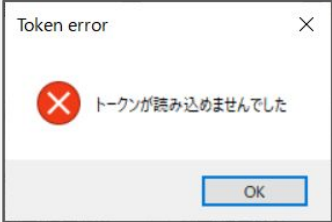


いずれかのボタンで終了

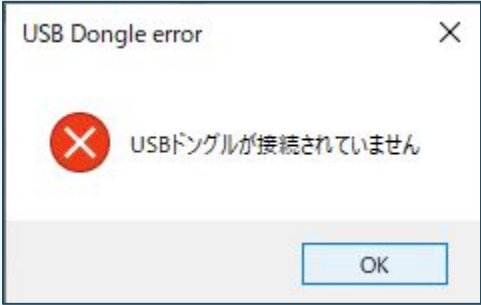
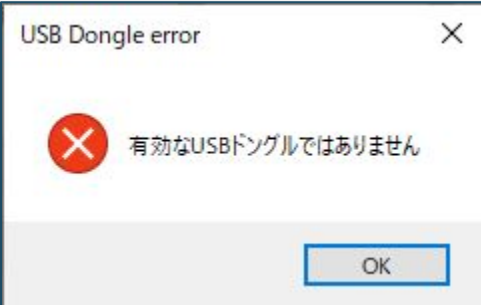
キャプチャ撮影

エラー項目

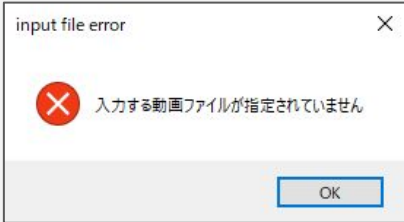
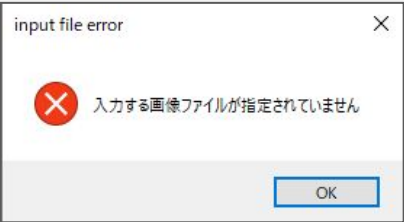
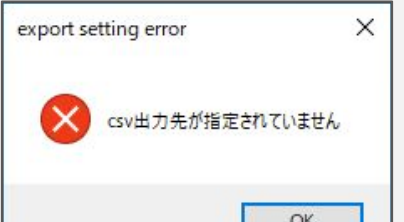
■エラーウィンドウ(トークンアクティベーション版)

画面キャプチャ	表示シーン	詳細
	トークンファイルを指定していない場合	アクティベーション用トークンの場所が指定されていません。 参照ボタンよりトークンファイルの場所を指定してください。
	存在しないトークンまたはライブラリと一致しない場合	指定したトークンが、存在しないトークンまたはライブラリと一致しないトークンです。 トークンを再度確認してください。
	トライアル期間終了時	アクティベーション用トークンの有効期限が切れました。 延長または買い切りをご希望の場合は、弊社サポート事務局までご連絡ください。
	トークンファイルの中身が空の場合	トークンファイルの中身が空でないかどうかを確認してください。

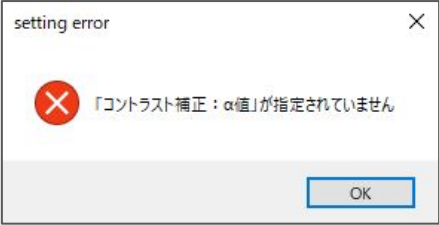
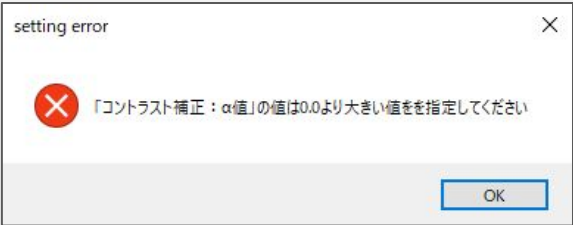
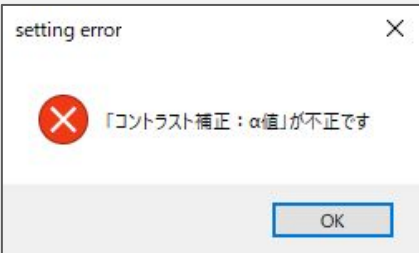
■エラーウィンドウ(USBアクティベーション版)

画面キャプチャ	表示シーン	詳細
	専用のUSB Dongleを接続していない場合	専用のUSB Dongleがマシンに接続されていないか、または認識されていません。弊社より提供させていただいているUSB Dongleを接続するか、認識できるようにしてください。
	本アプリ用の専用USB Dongleではない場合	本アプリ向けに提供されたUSB Dongleではない可能性があります。詳しくは弊社営業窓口までお問い合わせください。

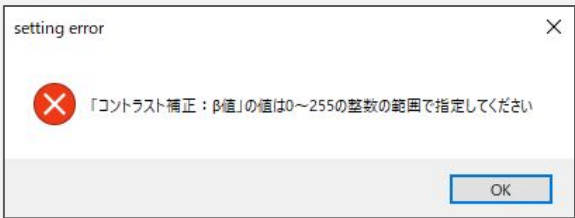
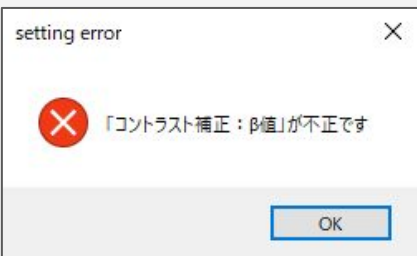
■エラーウィンドウ(その他)

画面キャプチャ	表示シーン	詳細
	動画ファイル指定エラー	入力する動画ファイル先が指定されていません。 参照ボタンより動画ファイルを指定してください。
	画像ファイル指定エラー	入力する画像ファイル先が指定されていません。 参照ボタンより画像ファイルを指定してください。
	csvファイル出力先エラー	検出ログのcsvファイルの出力先が指定されていません。 参照ボタンよりcsvファイルの出力先を指定してください。
	動画・キャプチャ画像出力先エラー	検出動画またはキャプチャ画像の出力先が指定されていません。 参照ボタンよりファイルの出力先を指定してください。

■エラーウィンドウ(その他)

画面キャプチャ	表示シーン	詳細
	コントラスト補正の 「α値」欄が 空欄だった場合	α値を 正の数で指定してください。
	コントラスト補正の 「α値」欄で 負の値を入力した場合	α値を 正の数で指定してください。
	コントラスト補正の 「α値」欄で 数値以外の値を入力した場合	α値を 正の数で指定してください。

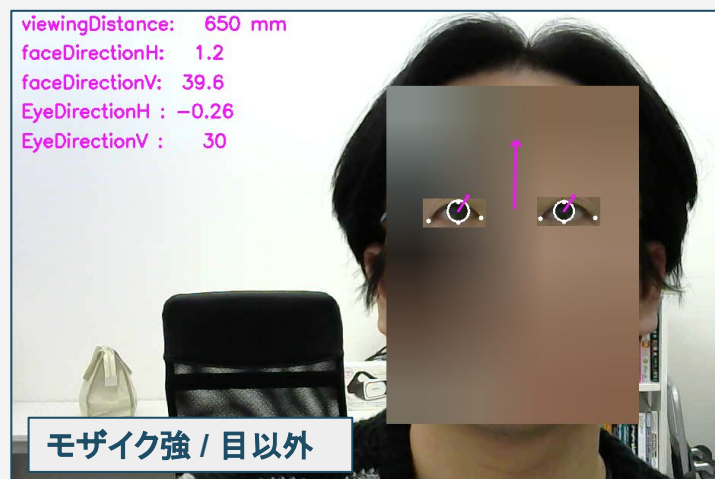
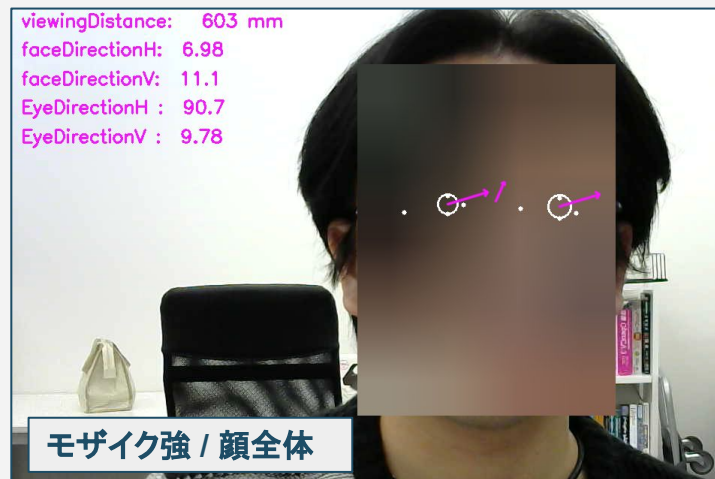
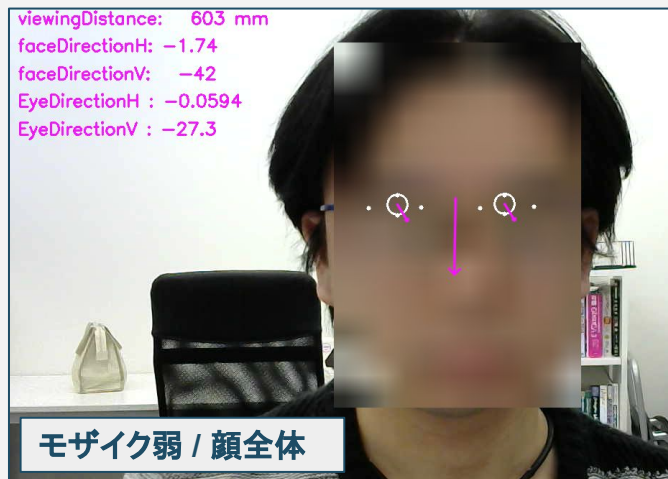
■エラーウィンドウ(その他)

画面キャプチャ	表示シーン	詳細
	コントラスト補正の 「β値」欄が 空欄だった場合	β値を 正の整数で指定してください。
	コントラスト補正の 「β値」欄で 負の値を入力した場合	β値を 0～255の範囲の整数で指定してください。
	コントラスト補正の 「β値」欄で 数値以外の値を入力した場合	β値を 正の整数で指定してください。

プライバシー保護について

■プライバシー保護機能

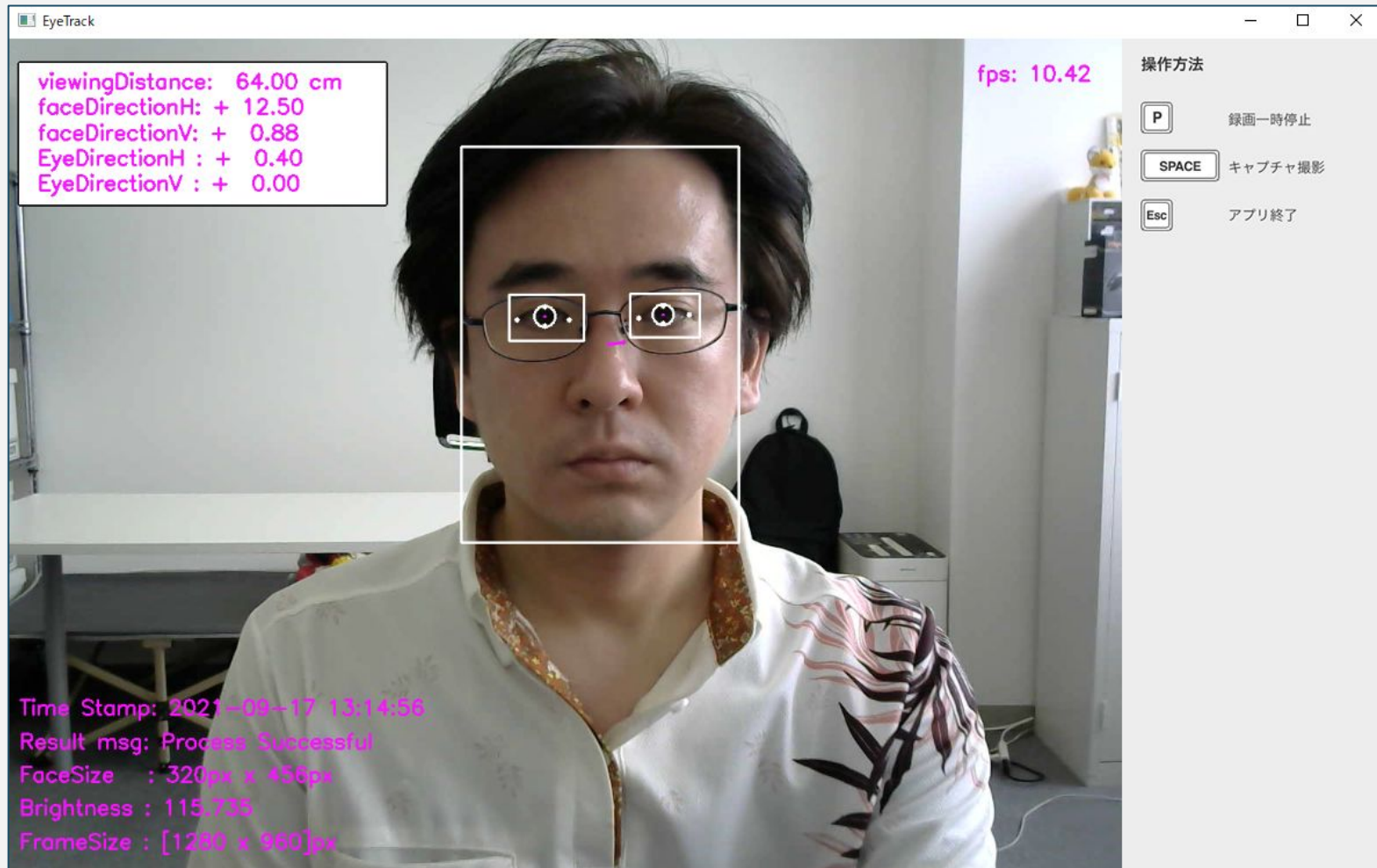
本バージョンには、プライバシー保護機能があります。プライバシー保護にチェックを入れてアプリを起動すると、以下のようなモザイク処理がかかります。視線を見やすいように目のみモザイクを外すことや、モザイクの強弱の選択が可能です。被験者さまの検出を安心して行ったり、動画の保存を行うことが可能です。（注意：装着具などにより顔検出が失敗した場合にはモザイク処理がされません）



モードについて

■モード選択:プロットモード

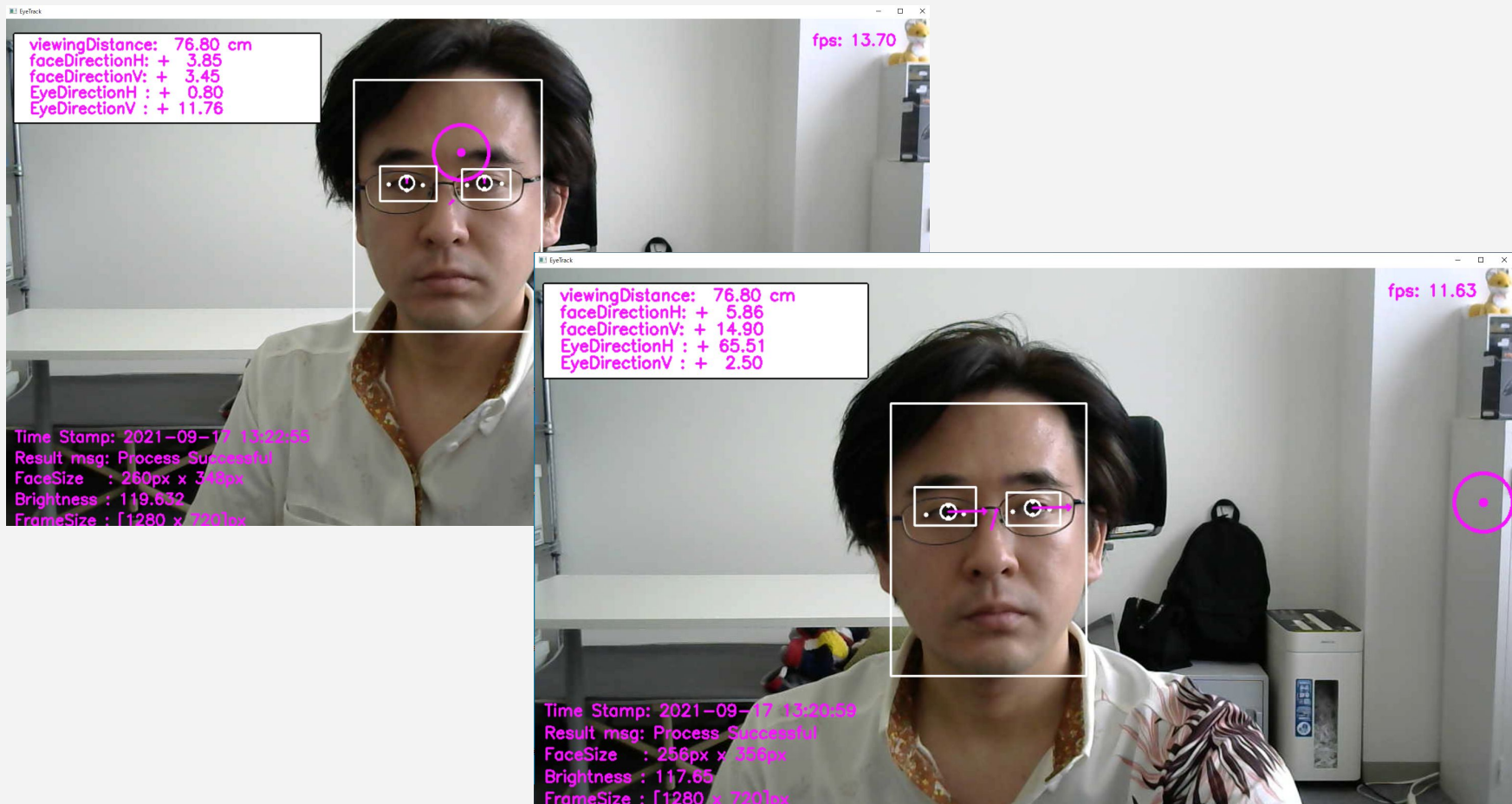
モードにより、「プロットモード」を選択した場合は、プロット描画でチェックを入れた顔特徴点項目において、検出できた箇所を顔上にプロットして表示されます。これにより問題なく顔の特徴点が検出できているかどうかを確認するためのモードです。顔上にプロットされる項目は顔検出位置、目検出位置、瞳中心座標および半径、上下まぶた位置、目尻目頭位置、顔向きベクトル、目向きベクトルです。



■モード選択:視線先推定モード

「視線先推定」を選択した場合は、顔向きと目線向きの合成ベクトルから、画面のどこを見ているかを推定して、二重丸でプロットされていきます。

検出を終了するには、キーボードの「Esc」キーを押してください。

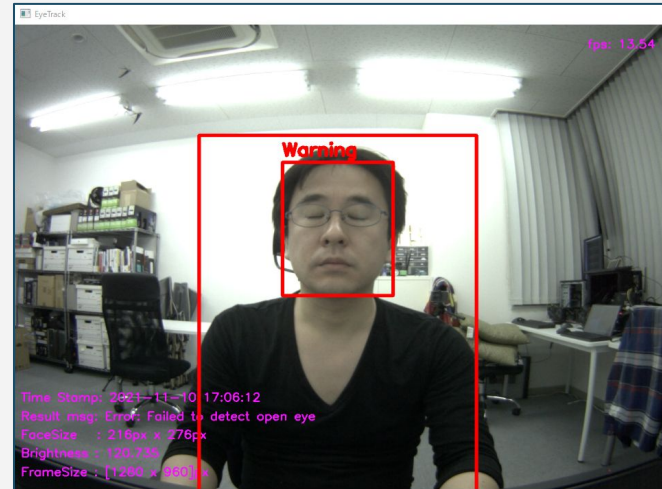


■モード選択：居眠り検知モード

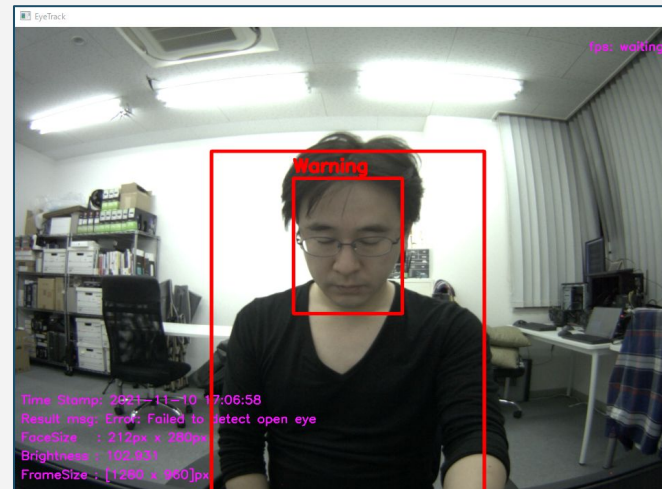
モード選択により、「居眠り検知モード」を選択した場合は、検出時に継続した閉じ目や、継続的に顔が下を向いている場合に、居眠りをしていると推定し、Warning表示が出るモードです。



平常時



継続した
閉じ目



継続した
うつむき顔

視線検出基礎情報について

■視線検出基礎情報 - メガネ判定

ディープラーニング版

単一フレームの画像から、顔が検出された場合、メガネ装着の有無をディープラーニングを用いて判定します。処理結果が格納 EyeDataクラスの、eyeGlassStatusメンバより0～2のいずれかの値が出力されることによりメガネ装着判定を行うことが可能です。

サングラスであると判定された場合は、目検出以降の目の特徴点検出処理は行いません。



裸眼

eyeGlassStatus = 0



クリアレンズ
メガネ

eyeGlassStatus = 1



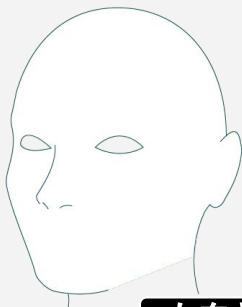
サングラス

eyeGlassStatus = 2

■視線検出基礎情報 - 顔向き判定

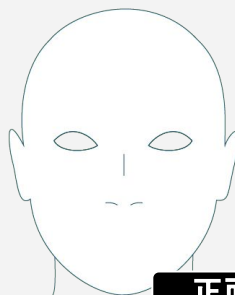
ディープラーニング版

単一フレームの画像から、顔が検出された場合、顔向きをディープラーニングを用いて判定します。
処理結果が格納 EyeData クラスの、faceDirStatusH、faceDirStatusV メンバより 0～2 のいずれかの値が出力
されることにより顔向きの判定を行うことが可能です。



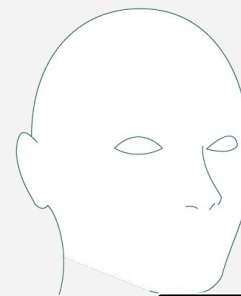
左向き

faceDirStatusH = 1



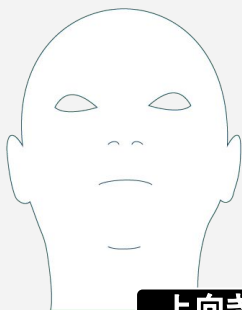
正面

faceDirStatusH = 0



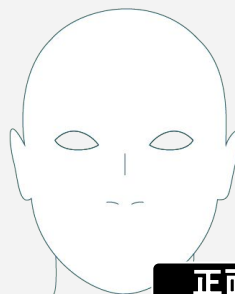
右向き

faceDirStatusH = 2



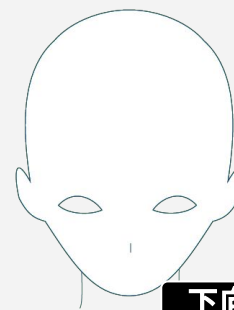
上向き

faceDirStatusV = 1



正面

faceDirStatusV = 0



下向き

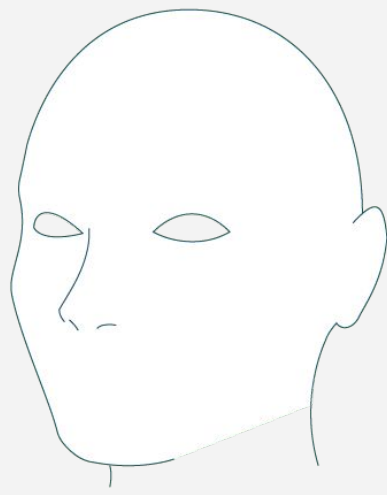
faceDirStatusV = 2

■視線検出基礎情報 - 水平方向の顔向き度

特許取得済

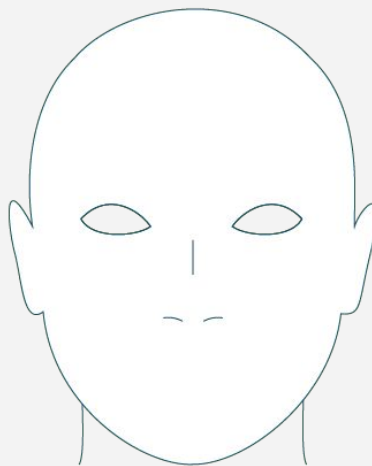
左右の顔の向きは、csvログのfaceDirectionH列より確認できます。この値は、カメラに対して正面を0として、顔向きの変化に応じて左側へ 0 ~ -100.0(約-30°)、右側へ0 ~ 100.0(約+30°)の間で定量化されます。

検出限界角度は、左右それぞれ約30度程度ですが個人差があります。



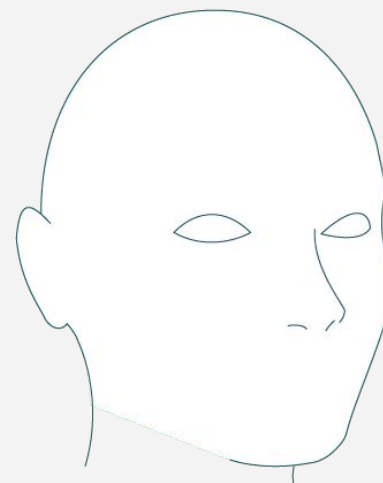
左限值

faceDirectionH
-100.0



正面

faceDirectionH
0.0



右限值

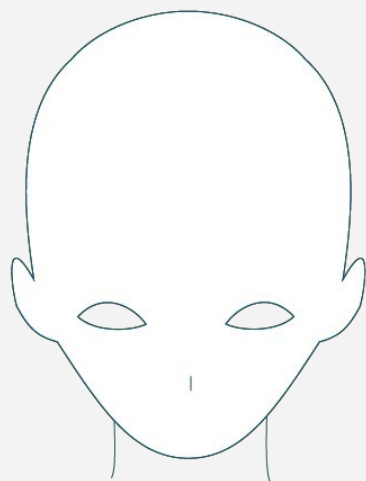
faceDirectionH
100.0

■視線検出基礎情報 - 垂直方向の顔向き度

特許取得済

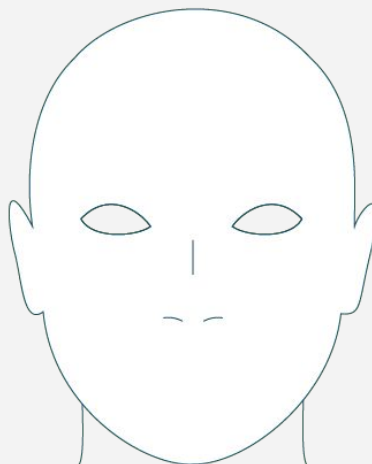
上下の顔の向きは、csvログのfaceDirectionV列より取得できます。この値は、カメラに対して正面を0として、顔向きの変化に応じて下側へ 0 ~ -100.0、上側へ 0 ~ 100.0 の間で定量化されます。

検出限界角度は、個人差があります。



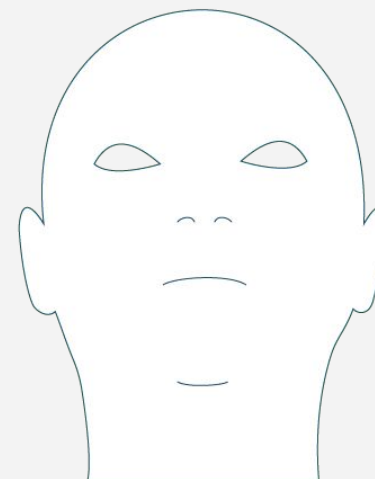
下限値

faceDirectionV
-100.0



正面

faceDirectionV
0.0



上限値

faceDirectionV
100.0

■視線検出基礎情報 - 目向き検出

ディープラーニング版

目の向きは、EyeDataクラスのメンバ変数であるeye(Left/Right)DirStatusHおよびeye(Left/Right)DirStatusVより取得できます。値はshort型の0～2のいずれかが返ります。値の詳細は以下の通りとなります。



左向き

eye(Left/Right)DirStatusH
1



中央

eye(Left/Right)DirStatusH
0



右向き

eye(Left/Right)DirStatusH
2



下向き

eye(Left/Right)DirStatusV
2



中央

eye(Left/Right)DirStatusV
0



上向き

eye(Left/Right)DirStatusV
1

■視線検出基礎情報 - 目向き度

特許取得済

目の向きは、csvログのeyeDirectionH列およびeyeDirectionV列より取得できます。それぞれの値は、カメラに対して正面を0として、目の上下左右への向き具体的の変化に応じて左側・下側へそれぞれ -100.0～0、右側・上側へそれぞれ0 ～ 100.0の間で定量化されます。



左限值
eyeDirectionH
-100.0



中央値
eyeDirectionH
0.0



右限值
eyeDirectionH
100.0



下限値
eyeDirectionV
-100.0



中央値
eyeDirectionV
0.0



上限値
eyeDirectionV
100.0

■視線検出基礎情報 - 目の開閉判定

ディープラーニング版

目の開閉パラメータは、csvログのLeftEyeStatus/RightEyeStatus列より確認できます。値はshort型の0～2のいずれかが返ります。値の詳細は以下の通りとなります。

eyeStatus = 0

Error

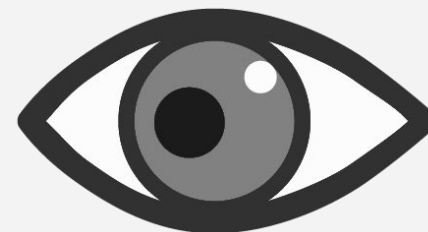
目検出エラー

eyeStatus = 1



目が閉じている

eyeStatus = 2



目が開いている

■視線検出基礎情報 - 視距離推定

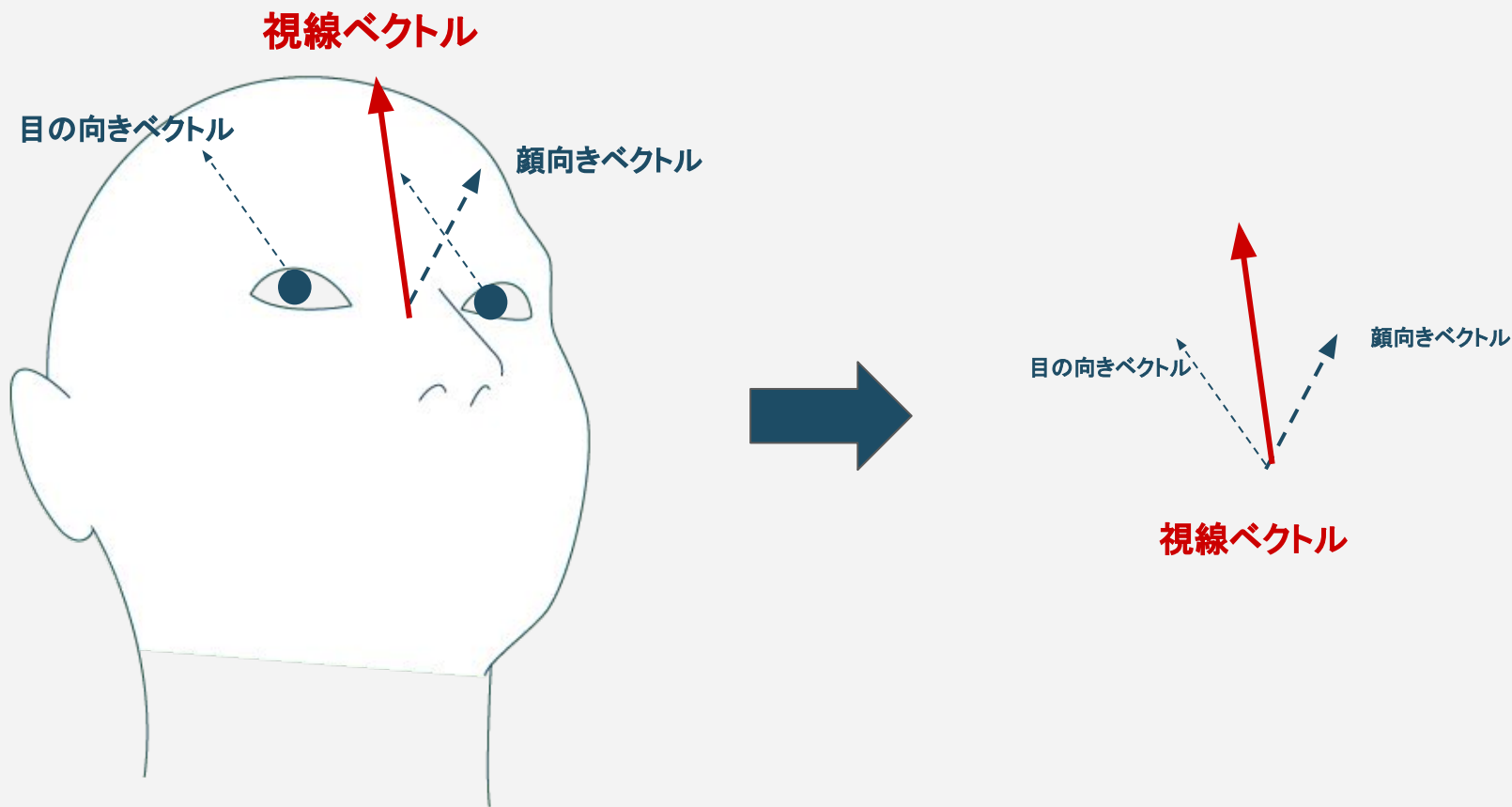
特許取得済

目検出結果をもとに、カメラから目までの距離を取得できます。
視距離の値は、csvログのviewingDistance列にて出力され、
単位はmm(ミリメートル)となります。



■視線検出基礎情報 - 視線ベクトル

自身で視線ベクトルを得る場合は、ライブラリより取得可能な顔向きベクトルと、目の向きベクトルから合成ベクトルを生成、重みづけなどを行うことで得ることが可能です。以下は、視線ベクトルを得る例です。

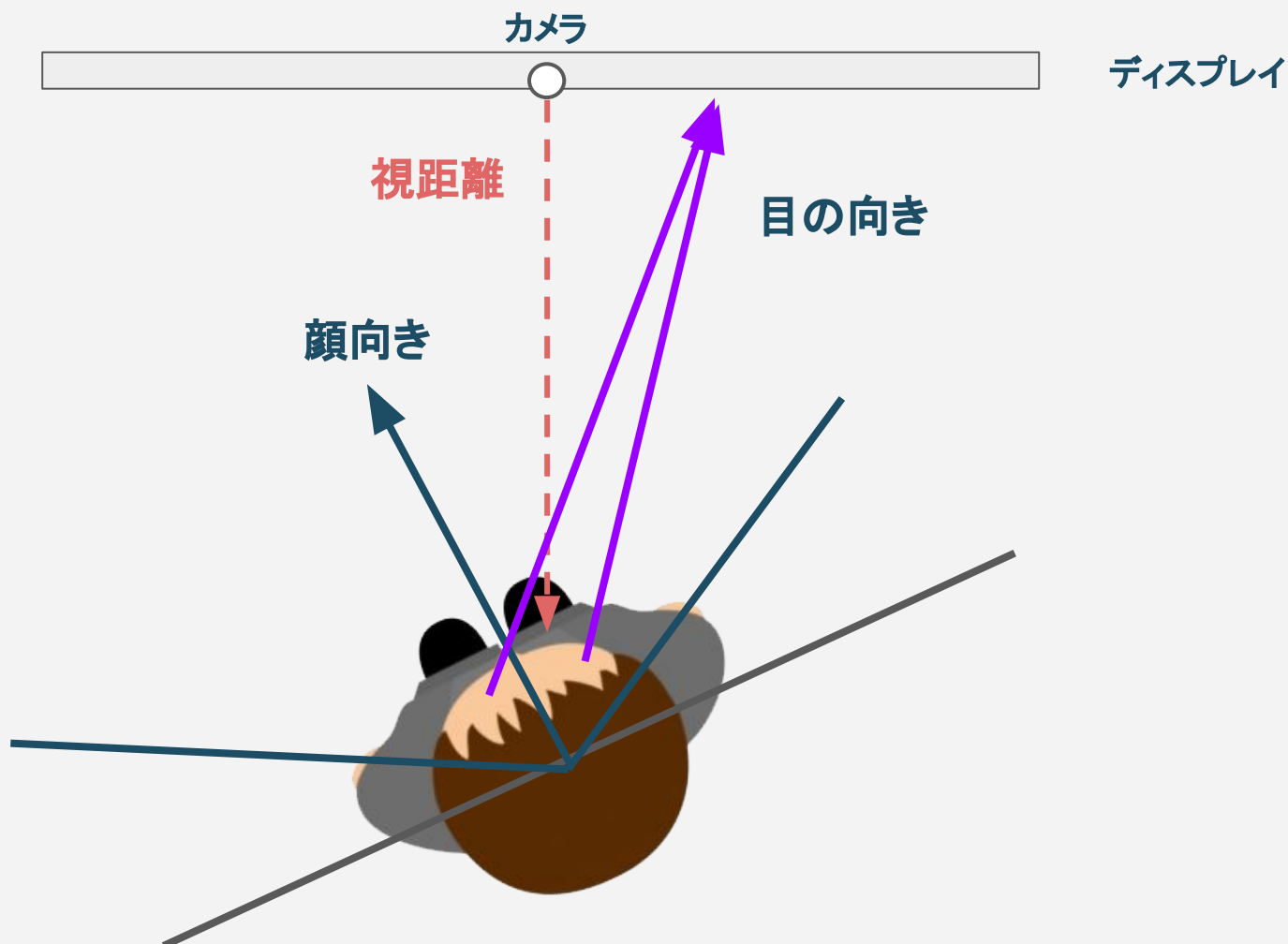


※目の向きベクトルは、左右同じ量となります。

■視線検出基礎情報 - 視線プロット先推定

視線先の推定は、顔向きベクトルおよび目の向きベクトルを用いた視線プロット先推定計算をライブラリ内部で行ってプロットしています。

SDKを使うと、お客様にて、ライブラリより取得した顔向き、目の向き、視距離をもとに視線プロット先を推定計算していただくことも可能です。



■視線検出基礎情報 - 注視エリアの推定

視線先のおおまかな領域は、EyeDataクラスのメンバ変数であるviewingAreaより取得できます。値はshort型の0～9のいずれかが返ります。入力フレームを均等に9分割した際の視線推定先領域を以下の値の通り示します。

viewingArea = 0

Error

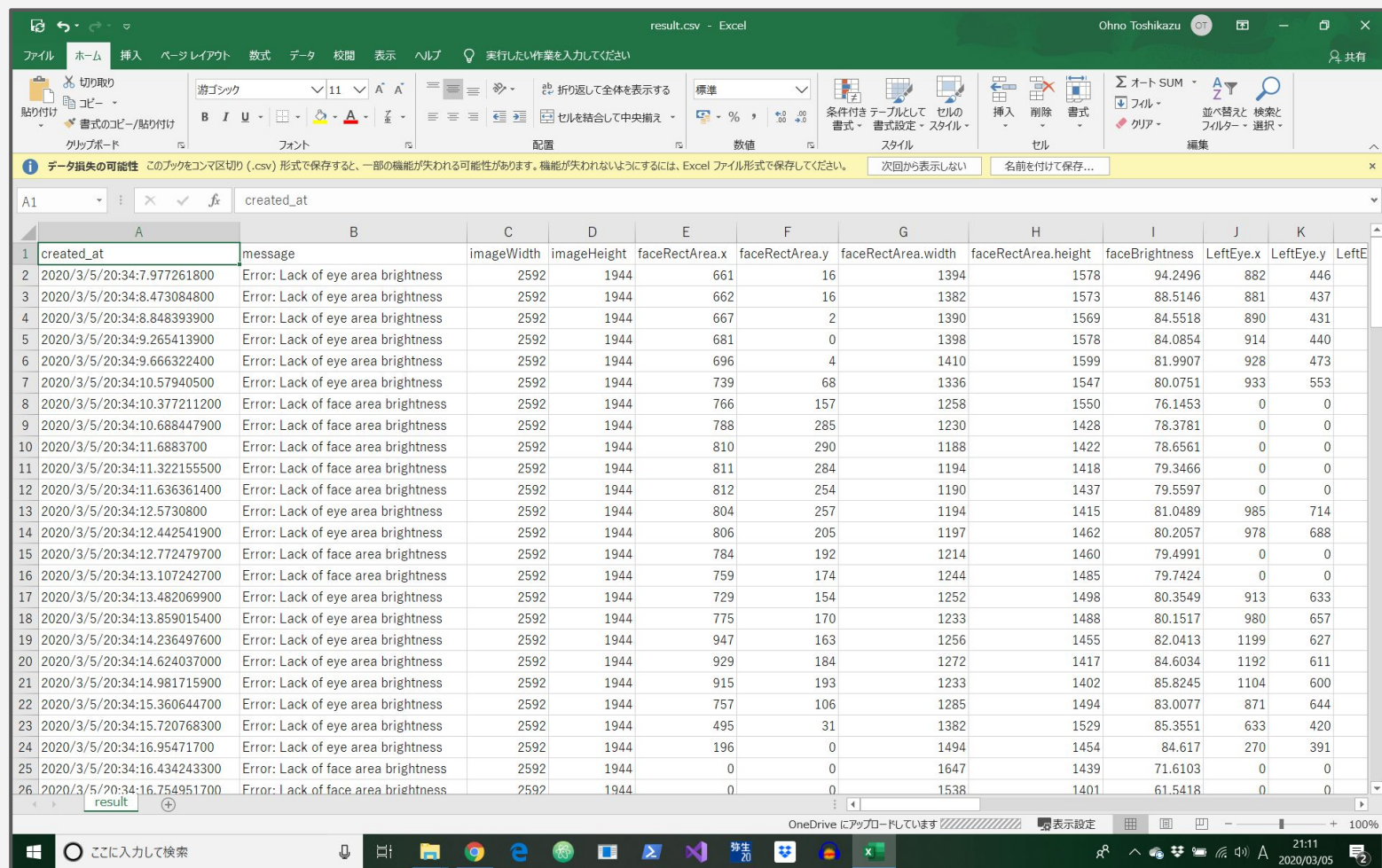
viewingArea = 1～9

3	4	5
2	1	6
9	8	7

csv出力項目

■csvファイル

csv出力先フォルダに格納されるファイルは、csv形式の検出ログファイルです。
出力ファイル名は、result.csvです。
複数プロセスを分けて実行しても、すべて同じファイルにログが追加されていきます。
別で保存したい場合は、ファイルを別の場所に移動するかファイル名を変更してください。



	A	B	C	D	E	F	G	H	I	J	K	L
	created_at	message	imageWidth	imageHeight	faceRectArea.x	faceRectArea.y	faceRectArea.width	faceRectArea.height	faceBrightness	LeftEye.x	LeftEye.y	LeftEye.y
1	created_at	message	imageWidth	imageHeight	faceRectArea.x	faceRectArea.y	faceRectArea.width	faceRectArea.height	faceBrightness	LeftEye.x	LeftEye.y	LeftEye.y
2	2020/3/5/20:34:7.977261800	Error: Lack of eye area brightness	2592	1944	661	16	1394	1578	94.2496	882	446	
3	2020/3/5/20:34:8.473084800	Error: Lack of eye area brightness	2592	1944	662	16	1382	1573	88.5146	881	437	
4	2020/3/5/20:34:8.848393900	Error: Lack of eye area brightness	2592	1944	667	2	1390	1569	84.5518	890	431	
5	2020/3/5/20:34:9.265413900	Error: Lack of eye area brightness	2592	1944	681	0	1398	1578	84.0854	914	440	
6	2020/3/5/20:34:9.666322400	Error: Lack of eye area brightness	2592	1944	696	4	1410	1599	81.9907	928	473	
7	2020/3/5/20:34:10.579405000	Error: Lack of eye area brightness	2592	1944	739	68	1336	1547	80.0751	933	553	
8	2020/3/5/20:34:10.377211200	Error: Lack of face area brightness	2592	1944	766	157	1258	1550	76.1453	0	0	
9	2020/3/5/20:34:10.688447900	Error: Lack of face area brightness	2592	1944	788	285	1230	1428	78.3781	0	0	
10	2020/3/5/20:34:11.6883700	Error: Lack of face area brightness	2592	1944	810	290	1188	1422	78.6561	0	0	
11	2020/3/5/20:34:11.322155500	Error: Lack of face area brightness	2592	1944	811	284	1194	1418	79.3466	0	0	
12	2020/3/5/20:34:11.636361400	Error: Lack of face area brightness	2592	1944	812	254	1190	1437	79.5597	0	0	
13	2020/3/5/20:34:12.5730800	Error: Lack of eye area brightness	2592	1944	804	257	1194	1415	81.0489	985	714	
14	2020/3/5/20:34:12.442541900	Error: Lack of eye area brightness	2592	1944	806	205	1197	1462	80.2057	978	688	
15	2020/3/5/20:34:12.772479700	Error: Lack of face area brightness	2592	1944	784	192	1214	1460	79.4991	0	0	
16	2020/3/5/20:34:13.107242700	Error: Lack of face area brightness	2592	1944	759	174	1244	1485	79.7424	0	0	
17	2020/3/5/20:34:13.482069900	Error: Lack of eye area brightness	2592	1944	729	154	1252	1498	80.3549	913	633	
18	2020/3/5/20:34:13.859015400	Error: Lack of eye area brightness	2592	1944	775	170	1233	1488	80.1517	980	657	
19	2020/3/5/20:34:14.236497600	Error: Lack of eye area brightness	2592	1944	947	163	1256	1455	82.0413	1199	627	
20	2020/3/5/20:34:14.624037000	Error: Lack of eye area brightness	2592	1944	929	184	1272	1417	84.6034	1192	611	
21	2020/3/5/20:34:14.981715900	Error: Lack of eye area brightness	2592	1944	915	193	1233	1402	85.8245	1104	600	
22	2020/3/5/20:34:15.360644700	Error: Lack of eye area brightness	2592	1944	757	106	1285	1494	83.0077	871	644	
23	2020/3/5/20:34:15.720768300	Error: Lack of eye area brightness	2592	1944	495	31	1382	1529	85.3551	633	420	
24	2020/3/5/20:34:16.95471700	Error: Lack of eye area brightness	2592	1944	196	0	1494	1454	84.617	270	391	
25	2020/3/5/20:34:16.434243300	Error: Lack of face area brightness	2592	1944	0	0	1647	1439	71.6103	0	0	
26	2020/3/5/20:34:16.754951700	Error: Lack of face area brightness	2592	1944	0	0	1538	1401	61.5418	0	0	

■CSVファイル

アプリから出力されるcsvの出力項目は以下の通りとなります。

csvヘッダー	内容
created_at	ログ出力時刻 (Y-M-D-h-m-s.ms)
message	検出結果メッセージ
imageWidth	撮影フレームの横幅(px)
imageHeight	撮影フレームの高さ(px)
faceRectArea.x	検出顔矩形領域の左上頂点x座標
faceRectArea.y	検出顔矩形領域の左上頂点y座標
faceRectArea.width	検出顔矩形領域の横幅(px)
faceRectArea.height	検出顔矩形領域の縦幅(px)
faceBrightness	顔領域の平均輝度値(8bit256階調)
faceCenter.x	顔中心点x座標
faceCenter.y	顔中心点y座標
eyeGlassStatus	メガネ装着判定値。 0=裸眼 / 1=メガネ装着 / 2=サングラス装着

csvヘッダー	内容
LeftEye.x	検出した左目矩形領域の左上頂点x座標
LeftEye.y	検出した左目矩形領域の左上頂点y座標
LeftEye.width	検出した左目矩形領域の横幅(px)
LeftEye.height	検出した左目矩形領域の高さ(px)
LeftEyeBrightness	検出した左目矩形領域の平均輝度値(8bit256階調)
irisRadiusLeft	左目の虹彩半径サイズ(px)
irisCenterLeft.x	左目の虹彩中心点のx座標
irisCenterLeft.y	左目の虹彩中心点のy座標
eyeLeftUpperEyelid.x	左目上まぶた点のx座標(虹彩中心点x座標と同じ)
eyeLeftUpperEyelid.y	左目上まぶた点のy座標
eyeLeftLowerEyelid.x	左目下まぶた点のx座標(虹彩中心点x座標と同じ)
eyeLeftLowerEyelid.y	左目下まぶた点のy座標
eyeLeftCornerL.x	左目目尻目頭左端点のx座標
eyeLeftCornerL.y	左目目尻目頭左端点のy座標
eyeLeftCornerR.x	左目目尻目頭右端点のx座標
eyeLeftCornerR.y	左目目尻目頭右端点のy座標

■CSVファイル

アプリから出力されるcsvの出力項目は以下の通りとなります。

csvヘッダー	内容
RightEye.x	検出した右目矩形領域の左上頂点x座標
RightEye.y	検出した右目矩形領域の左上頂点y座標
RightEye.width	検出した右目矩形領域の横幅(px)
RightEye.height	検出した右目矩形領域の高さ(px)
RightEyeBrightness	検出した右目矩形領域の平均輝度値(8bit256階調)
irisRadiusRight	右目の虹彩半径サイズ(px)
irisCenterRight.x	右目の虹彩中心点のx座標
irisCenterRight.y	右目の虹彩中心点のy座標
eyeRightUpperEyelid.x	右目上まぶた点のx座標(虹彩中心点x座標と同じ)
eyeRightUpperEyelid.y	右目上まぶた点のy座標
eyeRightLowerEyelid.x	右目下まぶた点のx座標(虹彩中心点x座標と同じ)
eyeRightLowerEyelid.y	右目下まぶた点のy座標
eyeRightCornerL.x	右目目尻目頭左端点のx座標
eyeRightCornerL.y	右目目尻目頭左端点のy座標
eyeRightCornerR.x	右目目尻目頭右端点のx座標
eyeRightCornerR.y	右目目尻目頭右端点のy座標

csvヘッダー	内容
faceDirStatusH	水平方向の顔向き判定結果 0=正面 / 1=左向き / 2=右向き / -1=エラー
faceDirStatusV	垂直方向の顔向き判定結果 0=正面 / 1=上向き / 2=下向き / -1=エラー
faceDirectionH	水平方向(左右方向)の顔向き度。-100~0~100
faceDirectionV	垂直方向(上下方向)の顔向き度。-100~0~100
LeftEyeStatus	左目の開閉状態。2=開き目/1=閉じ目/0=エラー
RightEyeStatus	右目の開閉状態。2=開き目/1=閉じ目/0=エラー
eyeLeftDirStatusH	水平方向の左目向き判定結果 0=正面 / 1=左向き / 2=右向き / 9 or -1=エラー
eyeLeftDirStatusV	垂直方向の左目向き判定結果 0=正面 / 1=上向き / 2=下向き / 9 or -1=エラー
eyeRightDirStatusH	水平方向の右目向き判定結果 0=正面 / 1=左向き / 2=右向き / 9 or -1=エラー
eyeRightDirStatusV	垂直方向の右目向き判定結果 0=正面 / 1=上向き / 2=下向き / 9 or -1=エラー
eyeDirectionH	水平方向(左右方向)の目向き度。-100~0~100
eyeDirectionV	垂直方向(上下方向)の目向き度。-100~0~100

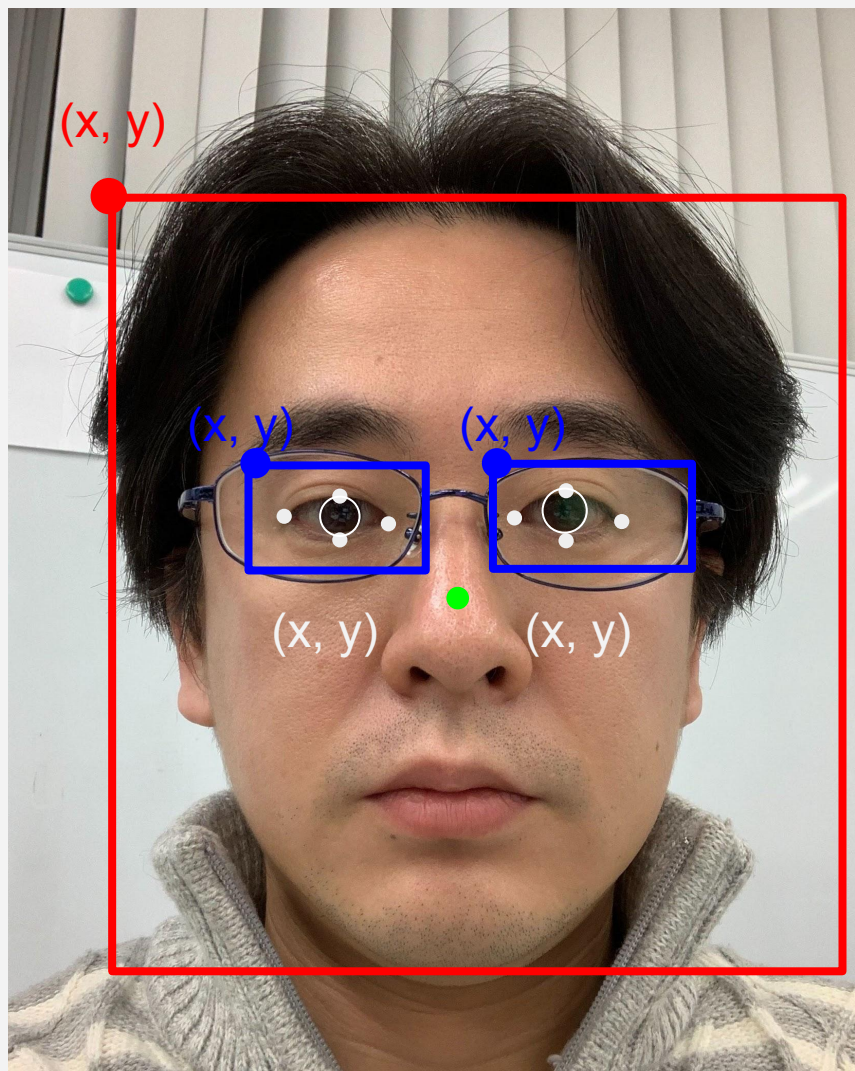
■CSVファイル

アプリから出力されるcsvの出力項目は以下の通りとなります。

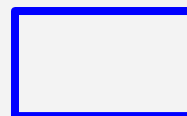
csvヘッダー	内容
viewingDistance	カメラから目までの視距離(mm)
viewingFaceVector.x	顔向きベクトル点のx座標
viewingFaceVector.y	顔向きベクトル点のy座標
viewingEyeVectorLeft.x	左目の視線向きベクトル点のx座標
viewingEyeVectorLeft.y	左目の視線向きベクトル点のy座標
viewingEyeVectorRight.x	右目の視線向きベクトル点のx座標
viewingEyeVectorRight.y	右目の視線向きベクトル点のy座標
plotPoint.x	推定された視線先の中心点x座標
plotPoint.y	推定された視線先の中心点y座標
viewingArea	入力画像を均等に9分割した場合の、視線先推定面の値
isBlink	当該フレームでのまばたき判定の有無
isSleep	当該フレームでの居眠り判定の有無

■csv出力データ図解

csvより取得できる検出位置データの値は、以下の通りとなります。
検出位置のxy座標は、矩形領域の左上頂点座標となります。



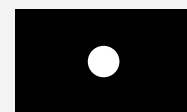
= faceRectArea
(x, y, width, height)



= eyeRectArea
(x, y, width, height)



= irisDiameter(irisRadius x 2)



= upperEyelid / lowerEyelid
eyeCornerL / eyeCornerR
(x, y, width, height)



= faceCenter

■message一覧

csvのmessage列にて取得できる主なメッセージは以下のいずれかとなります。その他のエラーが発生した場合はお問合せください。

メッセージ内容	内容
Process Successful	全ての処理が完了しました
Error: Failed to detect face	顔検出に失敗しました（顔が存在しない）
Error: Lack of face area brightness	顔領域の明るさ不足です（顔領域平均輝度値50以上 - 8bit256階調）
Error: Failed to detect eyes	目検出に失敗しました
Error: Failed to detect open eye	開き目が存在しません
Error: Failed to detect eye landmark points	両目とも目のランドマーク検出に失敗しました