

CVL 9.0 リリースノート

2019 11月 14
Revision: 9.0.0.1

法的通知

本ドキュメントで説明しているソフトウェアは、ライセンスの下で作成されており、本ページに記されている著作権を明記したライセンスに基づいた許可なしに、使用したり複製したりすることはできません。本ソフトウェア、本書、あるいはそれらのコピーをライセンスの保有者以外に提供することはできません。本ソフトウェアのあらゆる権利は Cognex Corporation あるいは権限を有する第三者に帰属します。Cognex Corporation は、同社が提供していない装置における同社製ソフトウェアの使用または信頼性についていかなる責任も負いません。Cognex Corporation は、本ソフトウェアの商業性、非侵害または特定目的に対する適合性に関して、明示的にも黙示的にも一切の保証を行いません。

本ドキュメントの内容は、予告なしに変更することがあります。内容の変更について、Cognex Corporation はいかなる責任も負いません。本ドキュメントあるいは関連ソフトウェアにおける誤りについて、Cognex Corporation はいかなる責任も負いません。

本書で例として使用されている企業名、人名およびデータは、特に断りがない限り架空のものであります。本書のいかなる部分も、Cognex Corporation の書面による許諾なく複製すること、手段・方法・目的を問わず電子的または機械的に転載すること、他のメディアに変換すること、および他の言語に翻訳することを禁じます。

Copyright © 2019. Cognex Corporation. All Rights Reserved.

Cognex の提供するハードウェアおよびソフトウェアの一部は、下記のウェブサイトを示す米国およびその他の国の特許、または出願中の特許で保護されています。cognex.com/patents.

下記は Cognex Corporation の登録商標です。

Cognex, 2DMAX, Advantage, AlignPlus, Assemblyplus, Check it with Checker, Checker, Cognex Vision for Industry, Cognex VSOC, CVL, DataMan, DisplayInspect, DVT, EasyBuilder, Hotbars, IDMax, In-Sight, Laser Killer, MVS-8000, OmniView, PatFind, PatFlex, PatInspect, PatMax, PatQuick, SensorView, SmartView, SmartAdvisor, SmartLearn, UltraLight, Vision Solutions, VisionPro, VisionView

下記は Cognex Corporation の商標です。

Cognex ロゴ, 1DMax, 3DLocate, 3DMax, BGAll, CheckPoint, CognexVSoC, CVC1000, FFD, iLearn, InSight (クロスヘアデザインおよび記号), InSight2000, InspectEdge, Inspection Designer, MVS, NotchMax, OCRMax, PatMaxRedLine, ProofRead, SmartSync, ProfilePlus, SmartDisplay, SmartSystem, SMD4, VisiFlex, Xpand

Portions copyright © Microsoft Corporation. All rights reserved.

Portions copyright © MadCap Software, Inc. All rights reserved.

その他の製品名および商標は、各所有者の商標です。

CVL 9.0 リリースノート

CVL バージョン 9.0 には、前のバージョンである CVL 8.0 からの変更や新機能が備えられています。

システム要件

お使いの PC が CVL をインストールして使用するための最小要件を満たしている必要があります。

PC の要件

お使いの PC が以下のハードウェアの最小要件を満たしている必要があります。

- MMX/SSE2 命令セットがサポートされている、1 GHz 以上の Intel Pentium (または互換性のある) CPU
- PCI バス Cognex フレームグラバまたは GigE カメラアダプタを使用している場合は、空いている PCI スロット 1 つ
- 直接接続カメラに適したポート (GigE Vision 向けのギガビットイーサネット)
- PCI、PCI Express (推奨)、または AGP ビデオアダプタ
 - CVL では、8 MB (限られた機能を提供) 以上のビデオカードメモリを必要とします。
 - マザーボードと一体化されたチップベースのビデオシステム上にある、マザーボードのスロットに接続するスタンドアロンビデオカードを推奨します。
 - 16 MB 以上のメモリが搭載された AGP または PCI Express x16 スタンドアロンビデオカードを推奨します。
 - ビデオカードは DirectX 8.1 以降をサポートしている必要があります。

以下も推奨します。

- CVL アプリケーションがディスクをページングする必要がないよう十分な RAM が PC に搭載されている必要があります。
- PC の BIOS が非プラグアンドプレイのオペレーティングシステムをサポートするように設定します。
- Windows デスクトップを 8 ビット、16 ビット、または 32 ビットカラーモードに設定します。カラー画像を表示する場合は、デスクトップを 16 ビットまたは 32 ビットモードにする必要があります。
- IDE (ATA) ハードディスク用の PIO ドライバに対して、PC でバスマスター DMA ディスクドライバを使用することを推奨します。マザーボードのベンダに確認して、バスマスター IDE ドライバが装備されているかどうかを判断します。
- PC に 64 MB 以上の RAM が搭載されている場合は、以下の手順を実行してカーネルおよびドライバの呼び出しを無効化し、メモリ内に保持してより早くアクセスできるようにします。
 - 現在のレジストリ設定のバックアップを作成します。
 - レジストリエディタを開き、以下に移動します。
HKLM/SYSTEM/CurrentControlSet/Control/Session Manager/Memory Management
 - DisablePagingExecutive キーを 1 に設定して、カーネルとドライバの呼び出しを無効にします。
 - PC を再起動します。

オペレーティングシステム

CVL 9.0 では、ネイティブ言語 (英語、日本語、ドイツ語、韓国語、および簡体字中国語) を使用した、シングルプロセッサまたはマルチプロセッサマシンでの開発と展開をサポートしており、以下の Windows 64 ビットオペレーティングシステムをサポートしています。

- Windows 7 Premium
- Windows 7 Professional

- Windows 7 Ultimate (64 ビット)
- Windows 10
- Windows 10 IoT Enterprise

これは Windows 7 をサポートする最後の CVL リリースであることに注意してください。

サポート対象の Microsoft Visual Studio 開発環境

CVL 9.0 では、以下の開発環境を使用して、ビジョンアプリケーションの開発と展開をサポートしています。

| 開発環境 | コンパイラバージョン | サービスパック |
|------------------------------|-------------------|----------|
| Microsoft Visual Studio 2015 | Visual C++ v14.0 | Update 3 |
| Microsoft Visual Studio 2017 | Visual C++ v14.16 | |

CVL には Microsoft Visual Studio 2015 および Microsoft Visual Studio 2017 CVL C++ ライブラリが付属しています。

CVL 9.0 は、アンマネージド C++ SDK であり、アンマネージドアプリケーションで Microsoft Visual Studio によって構築されたアプリケーションのみをサポートしています。CVL 9.0 は、マネージド .NET (/clr コンパイラスイッチを使用) で構築されたアプリケーションをサポートしておらず、混合モードとしても知られています。

サポート対象の画像取り込みプラットフォーム

CVL 9.0 は以下の画像取り込みプラットフォームをサポートしています。

- Cognex MVS-8511
- Cognex MVS-8511e
- Cognex MVS-8514
- Cognex MVS-8514e
- GigE Vision カメラ
- CFG-8724 シリーズのフレームグラバ
- 画像処理デバイスのインタフェースに準拠するサードパーティ製の画像処理デバイス。『CVL User's Guide』で説明しています。

このリリースの新機能

CVL 9.0 は一般に利用可能なリリースであり、新機能をサポートしています。

新しい USB セキュリティキー

このリリースでは、新しい Cognex USB セキュリティキーをハードウェアプラットフォームに取り付けて、ソフトウェアのライセンス付与が適切に行われるようにする必要があります。以前のリリースでサポートされていたセキュリティオプションは、いずれもこの CVL リリースで機能しません。既存の Cognex フレームグラバでは、これまでのリリースのように、引き続きそれぞれの画像取り込み機能を実行します。

Cognex では、アプリケーション開発向けの期間限定ベースのセキュリティキー、そして製造環境でアプリケーションに使用する運用キーをサポートしています。

詳しい内容については、Cognex 担当者にお尋ね下さい。

Visual Studio 2017

このリリースでは、Microsoft Visual Studio 2017 (v. 15.10) を使用したアプリケーションの開発とコンパイルがサポートされるようになりました。

LineMax ツール

このリリースでは LineMax ツールがサポートされるようになりました。画像内のすべてのエッジ点候補を特定し、指定した基準に基づいて最も適したラインセグメントを適合します。LineMax ツールは 8 ビットおよび 16 ビットの画像をサポートします。

LineMax ツールは個々のエッジ点を見つけるために正確に配置されたキャリバを必要とする Fitting ツールよりも堅牢です。新しいアプリケーションでは LineMax ツールを使用し、既存のアプリケーション向けに Fitting ツールのサポートを継続することを推奨します。

このリリースに含まれている CVL クラスリファレンスでは、LineMax ツール用の API については説明していません。ヘッダファイルおよびそのソースコードのコメントは `%VISION_ROOT%\%defs%\ch_cvl` にあります。

LineMax ツールの詳細については、『CVL Vision Tools Guide』を参照してください。

CFG-8724 フレームグラバ

このリリースでは、統合セキュリティライセンスが組み込まれた CFG-8724 4 ポート GigE Vision フレームグラバをサポートしています。CFG-8724 では、1 台の PC に GigE Vision カメラを 4 台まで接続できます。さらに、複数の CFG-8724 フレームグラバを同じ PC に取り付けることができます。また、CFG-8724 で Cognex ソフトウェアライセンスを保存して、Cognex ソフトウェアの使用が適切に承認されるようにできます。CVL では 1 つの CFG-8724 からのみソフトウェアライセンスが読み取られることに注意してください。セキュリティおよびバージョンの情報については、『CVL User's Guide』を参照してください。

Cognex では、今後使用できなくなる CFG-8704e の後継として、CFG-8724 を提供しています。

CFG-8724 を取り付けた後に Cognex ビジョンソフトウェアをインストールした場合は、ソフトウェアをインストールした後にコンピュータの電源を切る必要があることに注意してください。ウォームリブートのみを実行すると、CFG-8724 が正しく機能しなくなる可能性があります。

16 ビット画像表示のサポート

このリリースでは、`ccPelBuffer<c_UInt8>` タイプの 8 ビット画像に使用する同じ API を使用して、`ccPelBuffer<c_UInt16>` タイプの 16 ビット画像の表示がサポートされるようになりました。CVL では、グレイピクセル値の 16 ビットの範囲を、表示用の 8 ビットの範囲に自動的にマッピングします。

広範なグレー値を持つ 16 ビット画像は、ディスプレイでその範囲を維持しますが、範囲の狭い 16 ビット画像は、ディスプレイでその範囲が拡大されて見えることに注意してください。

例えば、16 ビット画像に濃いグレーと薄いグレー値を持つピクセルが含まれている場合、CVL には 8 ビットの範囲の濃いグレーと薄いグレーの値で画像が表示されます。ただし、16 ビット画像に排他的に濃いグレー値が含まれている場合、範囲の上限に近いピクセル値は、8 ビットのディスプレイで薄いグレー値で表示されます。

CVL 画像表示 API については、ヘッダファイル `%VISION_ROOT%\defs\ch_cv1\display.h` を参照してください。

CNLSearch モデルサイズ

このリリースでは、CNLSearch ツールで新しい 4x4 モデルサイズをサポートしています。これまでの CVL リリースでサポートされていた最小の CNLSearch モデルサイズは、8x8 でした。

その他のビジョンツールでの 16 ビット画像のサポート

このリリースでは、以下のビジョンツールで 16 ビット画像がサポートされるようになりました。

- Caliper
- Image Warper
- ガウス (ガウシアン) サンプリング用のフィルタマスク
- PelSet ツール用のマスク

互換性に関する注意

Cognex ハードウェアおよび CVL 9.0 ソフトウェアの使用に関連する問題については、以下のセクションを参照してください。

大きな画像の互換性の問題

このセクションでは、大きな画像のサポートに関連する互換性の問題について説明します。

アフィンサンプリングツールのインタフェースの変更

CVL 6.7 CR1 以前のリリースでは、アフィンサンプリングツール (`ccAffineSamplingParams::xNumSamples` および `ccAffineSamplingParams::yNumSamples`) の X および Y 方向のサンプル数の指定に使用されていたメンバは、`c_Int16` タイプでした。

大きな画像のサンプリングに対応するため、CVL 6.7 CR3 以降のリリースでは、サンプルの数は `c_Int32` の値を使用して格納されます。

`ccAffineSamplingParams` の setter 関数を変更するだけで、`c_Int32` 値を取得できます。C++ コンパイラで 16 ビット値から 32 ビット値にレベル上げをしますので、これによって既存のコードと不適合になることはありません。ただし、getter 関数は既存のコードと不適合にならないように変更することはできません。CVL 6.7 CR3 では 2 つの新しいメンバを `ccAffineSamplingParams` に追加しました。

- `ccAffineSamplingParams::xNumSamples32()` は X 方向のサンプルの数を返します。
- `ccAffineSamplingParams::yNumSamples32()` は Y 方向のサンプルの数を返します。

16 ビット値を返す既存の getter 関数は廃止されているので、現在は、サンプルの数が 32768 以上の場合はエラーがスローされます。

大きな画像ではすべてのアフィンサンプリングモードがサポートされているわけではない

X 軸または Y 軸のいずれかで 32768 ピクセルより大きい入力または出力画像用としてアフィンサンプリングツールがサポートするのは、以下のサンプリングモードのみです。

- `ccAffineSamplingParams::eNone`
- `ccAffineSamplingParams::eBilinearAccurate`

残りのモード (`eBilinearApprox`、`eHighPrecision`、および `eBilinear`) はサポートされていません。サポートされていないモードのいずれかを指定し、入力または出力画像のいずれかに 32768 より大きい X 軸または Y 軸がある場合、このツールはエラーをスローします。

大きな画像をサポートするよう変更されたエッジツールのインタフェース

CVL 6.7 CR1 以前のリリースでは、16 ビット値を使用してサブピクセルエッジ (Edgelet) のピクセル全体の位置がエッジツールに保存されていました。つまり、ピクセル全体の位置が $-32768 \sim +32767$ の範囲外にある Edgelet は表現できませんでした。

CVL 6.7 CR3 以降では、エッジツールによって Edgelet の保存方法を指定できるようになりました。

- `ccEdgeletDefs::eRequestContiguous` を要求した場合、Edgelet は連続する 1 つのブロックに保存されます。画像の X または Y 座標が $-32768 \sim +32767$ の範囲外にあるピクセル全体の位置を含む Edgelet がない場合、Edgelet は `ccEdgelet` のベクトルに保存され、`ccEdgeletSet::edges()` を介してアクセスされます。画像の X または Y 座標が $-32768 \sim +32767$ の範囲外にあるピクセル全体の位置を含む Edgelet がある場合、Edgelet は `ccEdgelet2` のベクトルに保存され、`ccEdgeletSet::edges2()` を介して、または `ccEdgeletSet::begin()` を呼び出して取得される `ccEdgeletIterator` または `ccEdgeletIterator_const` クラスを介してアクセスします。
- `ccEdgeletDefs::eRequestEdges2` を要求した場合、Edgelet は `ccEdgelet2` のベクトルに保存され、`ccEdgeletSet::edges2()` を介して、または `ccEdgeletSet::begin()` を呼び出して取得される `ccEdgeletIterator` または `ccEdgeletIterator_const` クラスを介してアクセスします。

- `ccEdgeletDefs::eRequestEdgesAndOffsets` を要求した場合、Edgelet はメモリの 1 つ以上の別々のブロックに保存されます。そこで各ブロックには `ccEdgelet` オブジェクトの配列および 32 ビットオフセット指定子のペアが含まれています。Edgelet には `ccEdgeletIterator` または `ccEdgeletIterator_const` クラスを使用してアクセスします。これらのオブジェクトの 1 つを取得するには、`ccEdgeletSet::begin()` を呼び出します。

複数のブロックが使用されるのは、画像の X または Y 座標が -32768 ~ +32767 の範囲外にあるピクセル全体の位置が、少なくとも 1 つの Edgelet に含まれている場合のみです。ブロックを 1 つだけ作成している場合は、`ccEdgeletSet::edges()` および `ccEdgeletIterator()` メソッドの両方を使用して、セットの Edgelet にアクセスできます。

保存方法に関わらず、`ccEdgeletSet::begin()` を呼び出して取得された `ccEdgeletIterator` または `ccEdgeletIterator_const` クラスを使用して Edgelet にアクセスできるので、このインタフェースを使用することを推奨します。

特定の CVL および VisionPro バージョンで作成された CDB ファイルは無効

CVL 6.8 CR1、CVL 6.8 CR2、または VisionPro 6.0 に付属する IDBEdit プログラムを使用して作成された CDB または IDB ファイルは、以前のバージョンの CVL および VisionPro に付属する IDBEdit プログラムでは読み取ることができません。

CVL 9.0 (および VisionPro 6.1 または 6.2) に付属する IDBEdit プログラムを使用して、無効な CDB または IDB ファイルを修復することができます。影響のあったファイルを新しい IDBEdit プログラムを使用して開き、**[File] -> [Save As...]** を選択して別の名前前で保存します。新しく保存されたファイルは、すべてのバージョンの IDBEdit で読み取ることができます。

大きな画像を含む CDB ファイルは以前の CVL リリースとの互換性がない

CVL 6.8 CR3 で構築された CVL アプリケーションを使用して、X 軸または Y 軸のいずれかで 32768 ピクセルより大きい画像を保存した場合、CVL リリース 6.7 CR1 以前を使用して構築された CVL アプリケーションにその画像をロードしようとすると、1x1 ピクセルの画像が生成されます。

デバイスドライバの下位互換性

CVL 9.0 に付属するデバイスドライバは、下記の表にあるバージョンと下位互換性があります。この意味での下位互換性とは、このリリース以降のドライバ使用時に、指定されている古い CVL バージョンが継続して動作するということです。

| フレームグラバまたはカメラ | 8.0 ドライバと互換性のある初期の CVL バージョン (Windows 7) | 8.0 ドライバと互換性のある初期の CVL バージョン (Windows 8.1 および 10) |
|--|--|---|
| MVS-8511 MVS-8514 MVS-8511e MVS-8514e | CVL 7.0 (64 ビット) | CVL 8.0 |
| GigE Vision パフォーマンスドライバ | CVL 7.1 CR2 (64 ビット) | CVL 8.0 |

MVS-8510 シリーズのフレームグラバの互換性の問題

このセクションでは、MVS-8510 および MVS-8510e フレームグラバの使用に影響する互換性の問題について説明します。

MVS-8510 シリーズフレームグラバのコントラストと輝度

コントラストおよび輝度の設定は、`ccContrastBrightnessProp` クラスの機能を持つ CVL で行います。通常使用されるコントラスト設定の 0.05 ~ 0.95 の範囲および輝度設定の 0.20 ~ 0.80 の範囲内では、取り込まれた画像のグレーレベル値は、使用しているボードに応じてわずかに異なります。

- コントラストと輝度が同じ設定の場合、MVS-8511、MVS-8511e、MVS-8514、および MVS-8514e フレームグラバでは、入力信号が同じである (サポート対象ではない) MVS-8501 および MVS-8504 フレームグラバが生成するピクセル値よりも、最大で 10 高いか、6 低いグレーレベル値が生成されます。入力の一般的な範囲でのグレー値の中央値の差はおおよそ 0 です。

- コントラストと輝度が同じ設定の場合、MVS-8511、MVS-8511e、MVS-8514、および MVS-8514e フレームグラバでは、入力信号が同じである (サポート対象ではない) MVS-8500Le および MVS-8504e フレームグラバが生成するピクセル値よりも、最大で 4 高いか、12 低いグレーレベル値が生成されます。入力の一般的な範囲でのグレー値の中央値の差はおよそ 2 です (新しいフレームグラバが生成する値が低くなります)。

入力値の特定の範囲でのグレー値の差を最小限にするには、輝度およびコントラストの設定を調整します。

300-0406 ケーブルで使用するソフトウェアアドレス

Cognex の 300-0406 ケーブルは、Sony DXC-390 カラーカメラとモノクロのアナログカメラ 1 台を、MVS-8514 および MVS-8514e に接続することを想定しています。

このケーブルの使用時には、カラーカメラに接続している R、G、B ケーブルブランチは、カメラ位置 0、1、2 を使用します。つまり、4 つ目のケーブルブランチに接続されるモノクロカメラのソフトウェアアドレスは 3 となります。このアドレスを CVL コードで使用して、300-0406 ケーブルの使用時にモノクロカメラを制御します。

TTL 専用ケーブルオプション使用時に逆になるストロボ極性

すべての MVS-8510 シリーズフレームグラバ用の 3 つの平行 I/O ケーブルオプションの内の 2 つでは、ストロボラインは光絶縁型です。ただし、TTL パススルーケーブルオプション (300-0390 ケーブルと TTL 接続モジュール、800-5818-1) の使用時には、光絶縁時の同ラインと比較して、4 つのストロボラインに逆極性があります。

このように、オールオプト (300-0389) またはハーフアンドハーフ (300-0399) ケーブルオプションでストロボを使用するときに、`ccStrobeProp::strobeHigh()` に関する情報 (CVL クラスリファレンス) は正確であり、次のように記述されます。

```
void strobeHigh (bool polarity);
```

極性が `true` の場合、アクティブパルスは高に設定されます。`false` の場合は、アクティブパルスは低に設定されます。

ただし、TTL パススルーケーブルオプション、300-0390 でストロボを使用すると、極性情報は逆になります。

```
void strobeHigh (bool polarity);
```

極性が `true` の場合、アクティブパルスは低に設定されます。`false` の場合は、アクティブパルスは高に設定されます。

既知の問題

このセクションでは、このリリースでの既知の問題をリストアップしています。

画像取り込み

このセクションでは、CVL 9.0 での画像取り込みに関する既知の問題について説明します。

RGB Pack 関数による変換のコピーが正しくない

取り込み先のピクセルバッファが事前に割り当てられておらず、取り込み元のピクセルバッファと同じサイズでない場合、`cfRGBPack()` 関数によって変換が正しくコピーされません。

回避策

次のいずれかの操作を行います。

- `cfRGBPack()` を呼び出した後に手動で変換をコピーする
- `cfRGBPack()` に渡した取り込み先のピクセルバッファが結合されていないことを確認する

`ccLightProp::LightsOff()` が機能しない。

`ccLightProp::lightsOff()` 関数を呼び出しても反映されません。次のコードを使用してこの問題に対処できます。

```
fifo->properties().lightPower(0.0);
```

```
fifo->properties().auxLightPower(0.0);
```

```
fifo->prepare(0.0);
```

MVS-8500 の画像取り込み

このセクションでは、CVL 9.0 での MVS-8500 の画像取り込みに関する既知の問題について説明します。

8500 フレームグラバを使用すると、カラー情報を含むビデオ信号から画像を取り込むときに、ExtSync 画像取り込みによってロックが失われる可能性がある

カラー情報 (NTSC または PAL など) を含むビデオ信号が 8500 フレームグラバに送られると、カメラが向いているシーンのタイプによっては、ExtSync 画像取り込みが失敗する可能性があります。この問題の原因は、RS170 または CCIR 信号の一部として 3.58 MHz(NTSC) および 4.43MHz(PAL) 以下の周波数を持つキャリアとして、エンコードされるカラーバーストおよびカラー情報です。このカラー情報は、8500 での Hsync パルスの正しい検出を妨げて、誤ったロックの原因となる可能性があります。

回避策

回避策は、ビデオ信号からカラー情報を削除した後で 8500 に送ることです。この簡単な方法としては、コンポジットビデオから S-Video (Y/C) へのコンバータを入手します。いったん変換すると、Y 成分にはカラー情報が含まれなくなります。

Sony DXC-390 カメラでアサーションが発生する可能性がある

画像取り込みの進行中にカメラケーブルが外された場合、外部同期ソースの使用時にアサーションが発生する可能性があります

キューに入ったソフトウェアのトリガに伴う、クアッドコア CPU での MVS-8500e 画像取り込みエラー

特定のシステム構成では、MVS-8500e フレームグラバで `isAbormal()` 取り込みの失敗が時々発生します。このような失敗はおおよそ 250,000 ~ 1,000,000 回の取り込みで 1 回発生します。また、以下の条件がすべて揃った場合に発生します。

- すべてのコアが有効化されたクアッドコア CPU でアプリケーションが実行されている。
- 手動 (ソフトウェア) トリガをアプリケーションで利用している。
- アプリケーションキューの複数の取り込みが開始されている (`ccAcqFifo::completeAcq()` を呼び出す前に `ccAcqFifo::start()` を複数回呼び出す)。
- アプリケーションで 3 台または 4 台のカメラから画像を非同期で取り込んでいる (マスタ/スレーブ取り込みは利用しない)。
- すべてのカメラが、Sony XC-HR70 または XC-HR90 などの高速カメラである。

取り込みエラーは一時的なものであり、1 回の取り込みにも影響し、自己回復します。このエラーは、外部トリガを利用する取り込みには影響しません。キューに保存された複数の開始を利用する、ソフトウェアから開始される取り込みにも影響します。

回避策

以下の回避策のいずれかを利用して、このエラーの発生を防ぐことができます。

- シングルコアまたはデュアルコアの CPU でアプリケーションを実行する。
- クアッドコア CPU を使用している場合は、コアを 1 つ無効化する。
- `ccAcqFifo::completeAcq()` を呼び出す前に、`ccAcqFifo::start()` を複数回呼び出さない。
- 外部トリガを使用して画像を取り込む。

表示

このセクションでは、画像表示に関する問題について説明します。

楕円の弧が正しく描画されない

楕円の中心が表示領域のはるか外側に位置するときには、楕円の弧が正しく描画されません。`sizeof(short)` より大きい半径の値を使用すると、このような影響が出ます。この問題は Microsoft GDI グラフィック描画コードの制限が原因で発生し、すべてのサポート対象の Windows プラットフォーム (2000 および XP) に存在します。

ccDisplayConsole 使用時の表示ズームに一貫性がない

`ccDisplay::mag()` によって使用できる最大ズームは 65,536 です。最小ズームは画像のサイズに応じて変わります。例えば、640x480 の画像の場合、最小ズームは -60 です。これが原因で、`ccDisplayConsole` を使用してズームすると、望ましくない動作が起きる可能性があります。表示コンソールウィンドウの下部にあるステータスバーに表示されるとおり、ズームアウト (ズーム用のツールバーボタンを使用) によって表示倍率は 1/2、1/4、1/8、1/16、1/32、および 1/60 とすることができますが、ここからズームインすると、表示倍率は 1/30、1/15 などとなります。

ccUIManShape から派生した子とのペアレンティングが機能しない

親子関係の子図形が ccUIManShape から派生している場合、ペアレンティングは期待どおりに機能しません。親図形をドラッグすると、ccUIManShape から派生した子のハンドルのみがその親と一緒に移動し、子図形自体は動きません。

ccUIGenAnnulus を使用するとペアレンティングが親として機能しない

一般に、親子関係の親図形が ccUIManShape から派生し、子図形が ccUIPointShapeBase から派生している場合、ペアレンティングは期待どおりに機能します。ただし、ccUIGenAnnulus (ccUIManShape から派生した図形) は例外であり、親として適切に機能しません。

startLiveDisplay() に渡された FIFO がスコープ外になると stopLiveDisplay() がハング状態になる

ccDisplay::stopLiveDisplay() を呼び出す前に、ccDisplay::startLiveDisplay() に渡された FIFO が破棄されないようにする必要があります。例えば、関数に FIFO を作成し、startLiveDisplay(fifo) を呼び出してこの FIFO で渡し、FIFO が作成された関数が返された後に stopLiveDisplay() を呼び出した場合、stopLiveDisplay() メソッドがハング状態になって値が返されません。

複数モニタはサポートされていない

システムに複数のモニタがある場合、CVL では、表示コンソールウィンドウの対象を特定のモニタとすることができる API は提供していません。

楕円の円環セクションを塗りつぶせない

ccGraphicProps::penfill() 関数は ccEllipseAnnulusSection オブジェクトに対しては機能しません。

ccUIGenAnnulus の塗りつぶしを設定するとグラフィック全体が塗りつぶされる

ccUIGenAnnulus の塗りつぶしプロパティを true に設定すると、円環だけでなく図形全体が塗りつぶされます。

Sony DXC-390 および 8 ビットデスクトップでディスプレイに何も表示されない

MVS-8504、Sony DXC-390 カメラ、「Sony DXC-390 640x480 IntDrv CCF」ビデオフォーマット、および ccAcqFifoPtrh を使用してライブ表示を実行するように、テストケースが設定されました。8 ビット Windows デスクトップ上に表示すると、表示される画像が何もないことが分かっています。

回避策

この設定の利用時には、16 ビットまたは 32 ビットデスクトップの色深度を使用して表示画像を表示します。

ドライバ

このセクションでは、CVL 9.0 でのドライバに関する既知の問題について説明します。

PC 固有の割り込みの遅延によって、高速での画像取り込みが失敗する可能性がある

Cognex では、テスト中に、一部の PC で割り込みに対応するまでに比較的長い時間 (5 ~ 6 ms) がかかることを把握しています。大部分の CVL アプリケーションでは、この長い遅延によって問題が発生することはありません。ただし、取り込みレートが 6 ms ごとに取り込み 1 回よりも高い場合は、この長くなった遅延によって割り込みからの回復が間に合わず、取り込みに失敗する可能性があります。一部の PC では、長い遅延によって、適時に割り込み応答を必要とするその他のデバイスで問題が発生する可能性もあります。

以下の PC でこの問題が見られることが分かっています。

IBM Intellistation Z Pro: モデル番号 MT-M 6221-22U (2.66 GHz Xeon シングル CPU)

IBM Intellistation M Pro: モデル番号 MT-M 6219-42U (3.06 GHz P4 シングル CPU)

この問題は、対象領域 (ROI) での取り込みに関係しています。画像が小さい場合は、フレームレートが通常よりも高いので、潜在的に割り込みオーバーランが発生する可能性があります。したがって、ROI コントロールおよび部分スキャンをサポートする MVS-8500 でデシメーション (ROI と同じ最終結果を伴う) をサポートするカメラを使用すると、ここで言及する問題が発生する可能性があります。

開発フレームワーク

このセクションでは、CVL 9.0 での CVL 開発環境に関する既知の問題について説明します。

新旧両方の iostream ライブラリが使用できない

CVL では std 名前空間から新しい iostream ライブラリを使用します。したがって、CVL ヘッダを含めると、新しい iostream ヘッダも含まれます。古い iostreams ライブラリ、または古い iostreams ライブラリを使用する別の製品 (MatLab など) を使用すると、シン

グルソースファイルで両方のライブラリのヘッダを含めようとするときに、コンパイルエラーが発生する場合があります。この問題が出現するのは、Windows ios クラスが原因です。

これは特定クラスのコンパイル時間の問題であり、その他のクラスの潜在的な実行時間の問題でもあります。なぜなら、両方のクラスで使用可能な列挙の値が異なるからです。現在ソフトウェアで古い iostream ライブラリを使用している場合は、新しい iostream ライブラリにアップデートすることを推奨します。ソフトウェアで新旧両方の iostream ライブラリを使用する必要がある場合は、必ず先に新しい iostream ライブラリを含めることを推奨します。

stdafx.h の使用時には (すべてのソースファイルで先に含めるので)、以下のような行を stdafx.h の最上部に追加するだけです。

```
#include <ch_cv1/defs.h>
#include <fstream>
#include <iostream>
```

defs.h の最初のインクルードは defs.h でプラグマを利用するためのものであり、「debug symbol name too long」コンパイラ警告 (C4786) を無効にします。このプラグマは、警告を生成するファイルが含まれる前に指定する必要があります。そうしないと、警告が発せられます。また、defs.h を含める代わりに直接このプラグマを使用することもできます。

サンプルコード

このセクションでは、CVL 9.0 でのサンプルコードに関する既知の問題について説明します。

Pmalign4 サンプルが実行時にクラッシュする

CVL に付属する pmalign4 サンプルアプリケーション (\$VISION_ROOT¥sample¥cvl にあります) は、Windows Explorer または コマンドプロンプトのコマンドラインから、現在のディレクトリを実行可能ファイルの場所に設定して実行すると、「plate.idb can't be opened」エラーでクラッシュします。この問題は、インストール中に plate.idb ファイルへの相対パスが正しく設定されていないことが原因です。

回避策

Visual Studio 内でサンプルを構築して実行するか、または plate.idb および plate.dxf ファイルを ¥cvlproj¥ ディレクトリに移動します。

シリアル I/O

このセクションでは、CVL 9.0 でのシリアル I/O に関する既知の問題について説明します。

シリアル I/O: 最初のバイトを受信するまで読み取り操作でタイムアウトが無視される

シリアルポートで最低でも 1 バイトを受信するまでは、有限のタイムアウトが指定されている場合でも、`ccSerialIO::read()` は無期限にブロックされます。シリアルポートで最初のバイトが受信された後、その時点から `read()` は正しく機能します。

形状モデル

このセクションでは、CVL 9.0 での形状モデルに関する既知の問題について説明します。

結合されていないピクセルバッファを記録しようとするクラッシュする可能性がある

`ccShapeModel::record()` メソッドでは、供給されたピクセルバッファが結合される必要があります。ただし、例外をスローすることではなく、結合されていないピクセルバッファが供給されている場合に 0 で除算しようとするので、最終的にはアプリケーションがクラッシュします。例えば、以下の呼び出しは最終的にクラッシュします。これは、第 2 の引数として供給された、デフォルトで構築されている 8 ビットの `ccPelBuffer` が、デフォルトで結合されていないためです。

```
ccShapeModel::record(diagObj, ccPelBuffer<c_UInt8>(), shape,
gprops, gprops, gprops, cmT("Synthetic model"));
```

回避策

必ず、幅と高さがゼロではない結合ピクセルバッファを、第 2 の引数として `ccShapeModel::record()` メソッドに供給します。

Caliper ツール

このセクションでは、CVL 9.0 での Caliper ツールに関する既知の問題について説明します。

Caliper: 高精度モードでクリッピングを有効にすると余分なエッジが検出される

eHighPrecision モードで Caliper を実行した場合、アフィン矩形で画像をクリップするように設定すると、余分なエッジが検出されます。例えば、投影された画像 (元の画像の逆読み取りに相当します) が、結果的に 200 214 198 197 194 192 190 185 183 111 ... を読み取った場合、コントラストしきい値を 5 に設定すると、ピクセル値 214 によって画像の右側に余分なエッジが表示されます。モードを eBilinear に変更すると、投影された画像は 198 198 197 196 193 191 189 184 176 107 ... を読み取り、元の画像を表示します。

回避策

高精度モードの原理が、エッジ近傍のデータも利用することに起因しています。回避策は、バイリニアモードを使用することです。

Caliper: クリッピングされた場合に結果のグラフィックスが描画されない

drawMode 引数で ccCaliperDefs::eDrawProjFilter を ccCaliperOneResult::draw() に指定した場合、グラフィックス (投影画像およびフィルタリングされた投影画像のグラフィック表現を示す) は、投影領域が入力画像によってクリッピングされている場合には表示されません。この発生を避けるには、投影領域をすべて囲む入力画像を供給します。

Caliper: クリッピングを有効化すると、投影された画像およびフィルタリングされた画像が描画されない

投影モードに関わらず、クリッピングを有効化すると、ccCaliperResultSet::draw() メソッドでは投影された画像およびフィルタリングされた画像が指定されたとおりに描画されません。例えば、以下のコードはラベルおよび領域のみを描画します。

```
clpResultSet.draw(graphList, ccCaliperDefs::eDrawLabel  
  
    | ccCaliperDefs::eDrawProjFilter  
  
    | ccCaliperDefs::eDrawProjectionRegion  
  
    | ccCaliperDefs::eDrawStandard);
```

この後、投影画像を追跡することは簡単ではありません。

Edge Detector ツール

このセクションでは、CVL 9.0 でのエッジツールに関する既知の問題について説明します。

エッジ: ccDiagObject の記録によってディープコピーがデフォルトで実行されるわけではない

ディープコピーがデフォルト設定されていても、またはシャローコピーが diagFlags によって強制されていても、cfEdgeDetect() が診断オブジェクトとして記録された後は、ソース画像、強調画像、および角度画像内に描画することができ、これらの変更は診断ビューワに示されます。

ヒストグラムツール

このセクションでは、CVL 9.0 でのヒストグラムツールに関する既知の問題について説明します。

cfPelHistogram() の動作がヘッダファイルコメントと一致しない

<ch_cvl/histo.h> ヘッダファイルでの cfPelHistogram() 関数に対するコメントに、渡されたベクトルのサイズが 0 の場合は、この関数がベクトルのサイズを変更してゼロ化するという不正確な内容が記されています。この関数は、要素が含まれていないベクトルのサイズを変更したり、ゼロ化したりすることはありません。

ID ツール

このセクションでは、CVL 9.0 での ID ツールに関する既知の問題について説明します。

コンポジットコードで、結果を結合するときに 2D コンポーネントのサブタイプを決定できない

ccSymbologyParamsComposite::combineResults() に true を指定した場合、2D コンポーネント (CC-A または CC-B) のサブタイプを決定することはできません。線形部分のサブタイプのみが返されます。

OCV ツール

このセクションでは、CVL 9.0 での OCV ツールに関する既知の問題について説明します。

回転の不確実性を有効化すると、複雑な文字の誤照合に続き、位置照合に失敗する

画像内の文字「A」を、「4」に設定された現在のキー (またはその逆) で文字の回転の不確実性をゼロより大きく設定して照合しようとする、OCV ツールが異常な動作を示す可能性があります。回転が許可されると、これらの文字が混同されやすくなります。

回転の不確実性を設定すると、通常このツールは A-4 の mismatches を照合しますが、得点が低いと、mismatches の後続のすべての位置で照合が失敗します。この動作は 3 つの異なるフォント (Courier, Lucida Sans, Terminal) において不確実性の範囲 (5 ~ 45 度) で見つかっています。不確実性によっては、Courier フォントで mismatches のあとに 1 つまたは 2 つの位置を照合すると、失敗し始めます。mismatches の発生した場所 (文字列の先頭または末尾のいずれかの近く) および mismatches の数の両方に関係なく、この動作は一貫しています。

混同の優位性によりスペース (キー 32) を使用できない

OCV ツールでは、空白文字が実際の文字と混同されることはない想定しています。そこで、例えば、このツールでは「a」であると期待される文字が空白に見えるかどうかを確認することはありません。ただし、その反対は当てはまりません。このツールでは、空白であると期待される文字が「a」のように見えるかどうかは確認します (「a」が混同しやすいと見なされる場合)。

行配置バウンディングボックスで行の末尾にあるワイルドカードが除外される

OCV の行配置バウンディングボックスには、行の末尾にあるワイルドカードが含まれません。

診断で行が見つからないとき、最後のポーズに赤い×印が表示される

行がまったく見つからない (つまり、すべての文字が見つからず、得点が 0.0 である) 場合、「最後のポーズでの行配置」の診断で、行配置ポーズの赤い×印が描画されます。×印はクライアント座標で描画されます。

PatInspect

このセクションでは、CVL 9.0 での PatInspect に関する既知の問題について説明します。

PatInspect: 余分なエッジまたは欠落しているエッジが正しく報告されない

PatInspect 境界差モードでは、検査領域のエッジ近くにある特徴が、欠落しているか余分であるとして、正しく報告されないことがあります。これが起きるのは、例えば、検査領域が登録領域と一致するときで、この場合、すべての境界の特徴はマッチングタイプとなっています。この動作は、CVL 5.4 下位互換モードおよび下位非互換モードの両方で見つかっています。

回避策

矩形の検査領域に、登録したいパターンのまわりのパディングが少なくとも 5 ピクセルあるようにします。

PatMax

このセクションでは、CVL 9.0 での PatMax に関する既知の問題について説明します。

PatFlex が実行時間パラメータで標準 DOF 値を無視する

PatFlex アルゴリズムでは、nominal() セツクを使用して cc_PMRParams で設定されている標準 DOF 値を無視します。

回避策

標準 DOF 値を PatFlex で使用するには、指定された DOF を有効化し、低いまたは高いゾーンの値を標準値に設定します。例えば、

```
runParams.zoneEnable(0);  
  
runParams.nominal(ccPMAlignDefs::kUniformScale, 2.0);  
  
の代わりに次を使用します。  
  
runParams.zoneEnable(ccPMAlignDefs::kUniformScale);  
  
runParams.zone(ccPMAlignDefs::kUniformScale, 2.0, 2.0);
```

Wafer Pre-Align ツール

このセクションでは、CVL 9.0 での Wafer Pre-Align ツールに関する既知の問題について説明します。

NotchMax モードでの平坦部分の長さの計算が正しくない

NotchMax モードでの平坦部分の長さの計算では、平坦部分の長さの計算でウェハー領域、内径、および外径を使用し、**ccWaferPreAlignResult::flatLength()** によってクエリを実行できます。ただし、ウェハーあたり 1、2、または 3 つの平坦部分がある場合があります。これは領域計算に影響する可能性があり、内径および外径の計算にも影響します。これによって、返された平坦部分の長さが不正確になる可能性があります。このため、NotchMax モードは平坦なウェハー上ではサポートされていません。

制限事項

制限事項とは、ハードウェア、オペレーティングシステムソフトウェア、製品アーキテクチャ、その他の外的制約によって CVL に課される動作です。一部の制限事項には、回避策が存在する場合があります。

セキュリティ

このセクションでは、このリリースでのソフトウェアセキュリティに関する制限事項について説明しています。

ccSecurityInfo はこのリリースではサポートされていません

このリリースでは、CVL クラスリファレンスで説明している `ccSecurityInfo` API はサポートしていません。

画像取り込み

このセクションでは、画像取り込みに関する一般的な制限について説明します。

デフォルトの露光時間はストロボによる取り込み用となっている

すべてのカメラのデフォルトの露光時間である $20\ \mu\text{s}$ は、ストロボによる取り込み用に設計されています。Sony XC-55 などのラピッドリセットカメラで環境光を使用する場合、露光時間を $1\ \text{ms}$ より長くする必要があります。

Pulnix TM-6CN 静的ビデオフォーマットは不適切な Sync モデルを使って作成されている

Pulnix TM-6CN 用の静的ビデオフォーマットは、内部 sync を有効にしてインスタンス化されますが、このカメラでは正しくなく、異様な画像が生成される可能性があります。CCF バージョンの TM-6CN のビデオフォーマットでは sync モデルが正しく設定されます。

回避策

静的ビデオフォーマットの使用時に、コード内の sync プロパティを手動で `cfSyncExternalIPLLComposite()` に設定します。

MVS-8500 の画像取り込み

このセクションでは、CVL 9.0 での MVS-8500 の画像取り込みの制限事項について説明します。

画像取り込みの輝度プロパティのデフォルト値

8504 および 8501 フレームグラバでは、画像取り込みの輝度プロパティのデフォルト値は 0.5 に変更されています。このため、このリリースで取り込まれる画像は、画像取り込み FIFO を作成した直後に、異なるデフォルト値を使用する以前のリリースで取り込まれた画像とは異なって見える場合があります。

CFG-8704e の画像取り込み

このセクションでは、CVL 9.0 での CFG-8704e フレームグラバの制限事項について説明します。

3 つ以上の CFG-8704e フレームグラバはサポートされない

同じコンピュータ上で 3 つ以上の CFG-8704e フレームグラバは使用できないことに注意してください。

表示

表示

このセクションでは、画像の表示での制限事項について説明します。

多数の UI 図形の追加と削除には時間がかかる

ループ内で一度に 1 つずつ、操作可能な多数の図形を表示に追加したり、表示から削除したりすると時間がかかります。

回避策

操作可能な図形の追加または削除を行う前に `ccDisplay::disableDrawing()` を呼び出し、すべての図形が追加または削除された後に `ccDisplay::enableDrawing(true)` を呼び出します。

ccUIGenAnnulus::move() が正しく実装されない

ccUIGenAnnulus::move() メソッドで、供給された cc2Vect で指定した場所に ccUIGenAnnulus を移動します。これは正しい動作ではありません。供給された cc2Vect の x および y の値によって図形の位置を調整する必要があります。

回避策

ccUIGenAnnulus::frontKid() を使用して UI で一般化されている円環のフロント子オブジェクトの位置を取得し、その結果にオフセットを追加して、ccUIGenAnnulus::pos(p) を呼び出して調整した位置を引数として渡します。

8 ビットデスクトップにおいて高精度モードで補間されたズーム

8 ビットデスクトップ上で、ccUITablet::interpolation(eHighPrecision) を呼び出して、補間されたズームを高精度モードに設定すると、グレースケールの範囲外の色が表示されることがあります。

回避策

8 ビット以外のディスプレイを使用したり、高精度補間ではなくバイリニアを使用したりします。

W2K 電源管理がライブ表示の妨げになる

Windows 2000 を使用している場合は、『CVL User's Guide』の「最適なパフォーマンスを得るようにシステム構成を調整する」で説明しているとおり、電源管理機能を無効にします。特に、ビデオモニタの電源が切れることのないようにシステムを構成してください。電源管理機能によってビデオモニタの電源が切れるように設定されている場合、CVL アプリケーションでグラフィックをライブ表示で表示しようとする問題が生じる場合があります。

CVL 5.3 にアーカイブされた楕円は新しい CVL バージョンで読み取れない

楕円幾何クラスの内部構造 (ccEllipseGeom) は CVL 5.4 で変更されました。CVL 5.3 以前でアーカイブされた ccEllipse、ccEllipseArc、ccEllipseAnnulus、または ccEllipseAnnulusSection オブジェクトを含む ccUISketch リストを表示しようとすると、CVL 5.4 以降に対して構築されたアプリケーションがハング状態になる場合があります。

リリースモードとデバッグモードで静的図形の表示が異なる

リリースモードとデバッグモードでは、静的図形の表示が異なります。図形の幾何によっては、2 つのモードで表示されるピクセル間でわずかな違いが生じることがあります。デバッグビルドとリリースビルドとの間で、表示されているピクセルがまったく同じとは限らない点に注意してください。

開発フレームワーク

開発フレームワーク

このセクションでは、CVL 開発環境での制限事項について説明します。

Visual Studio での「`treat wchar_t as builtin type`」に関するプロジェクト設定

Visual Studio では `treat wchar_t as builtin type` に関してデフォルトの `true` を使用します。Cognex では Visual Studio のデフォルトを使用し、プロジェクトでも必ず `treat wchar_t as builtin type == true` を使用するようにします。

インストール

ここでは、CVL のインストール時に注意すべき制限事項について説明します。

コマンドプロンプトウィンドウ内をクリックすると、アンインストールが一時停止する

CVL のアンインストール時に、コマンドプロンプトウィンドウで QuickEdit モードが有効になっている場合に、アンインストール中に開いたコマンドプロンプトウィンドウ内をクリックすると、カーソルが選択モードになり、アンインストールが停止されたように見えます。コマンドプロンプトウィンドウ内を右クリックして選択モードを終了し、アンインストールを再開します。

ディスクリート I/O

このセクションでは、CVL 9.0 でのディスクリート I/O の制限事項について説明します。

パラレル I/O: 外部 I/O モジュールの出力極性に一貫性がない

汎用パラレル I/O カード (P/N 800-5726-3) の外部構成を外部 I/O モジュール (P/N 800-5712-2) で使用して、パラレル I/O ラインに光絶縁型の接続点を提供します。MVS-8120 で使用するときには、外部 I/O モジュールの出力ライン 0 ~ 3 およびその他の Cognex ハードウェアの光絶縁型出力ラインに対して、外部 I/O モジュールの出力ライン 4 ~ 7 の極性が反転します。

CVL クラスリファレンスには、`ccOutputLine::set(true)` メンバ関数が光絶縁型出力ラインに電気が流れない状態を設定することが記述されています。これは外部 I/O モジュールの出力ライン 0 ~ 3 およびその他の Cognex ボードにも当てはまります。ただし、外部 I/O モジュールの出力ライン 4 ~ 7 の場合、MVS-8120 での使用時に、`set(true)` と `set(false)` の意味が逆になります。

実際のパルス幅が指定したよりも短くなる可能性がある

場合によっては、実際のパルス信号が `ccOutputLine::pulse()` に供給された幅パラメータよりも短くなる可能性があります。例えば、500 ms のパルスを次のように作成したとします。

```
oline.pulse(true, 0.5, false); // 低から高へ遷移、  
                                // 0.5 秒幅、ノンブロッキング
```

結果として得られるパルスは、実際には持続時間で 490 ms というように、短くなる場合があります。

この動作は、オペレーティングシステムタイマの既知の制限であり、`pulse()` 機能をサポートするすべての Cognex ハードウェアプラットフォームに影響します。この問題は、複数のパラレル出力ラインで `pulse()` 呼び出しを同時に行った場合には、より頻繁に発生することも知られています。

回避策

アプリケーションで絶対最小の期待値であるパルス時間が要求される場合は、`ccOutputLine::pulse()` 関数に、求められる実際のパルス幅より 1 ms 長い幅の値を指定します。これによってマルチメディアタイマの粒度に一致する次の間隔まで、実際のパルス幅が動きます。

さらに、Microsoft Win32 API 呼び出し `timeBeginPeriod(1)` を使用して、マルチメディアタイマ期間を 1 ms に短縮すると、パルスの精度が上がる可能性があります。ただし、この方法を使用する場合、タイマの変更によってタイマをより頻繁に点検する必要性が生じて中断されるので、システム全体のパフォーマンスに影響することに注意してください。

`timeBeginPeriod()` を使用するには、Win32 `mmsystem.h` ヘッダファイルおよび `winmm.lib` ライブラリをコードに含める必要があります。Visual C++ 6.0 ではヘッダファイルは `Vc98\include` ディレクトリにあり、Visual C++ .NET (7.0) では `VC7\PlatformSDK\include` ディレクトリにあります。SDK および DDK プラットフォームのインクルードファイルでも見つかります。

例えば、アプリケーションに次のコードを追加すると、システムタイマの精度を調整することができます。

```
#include <windows.h> // これを追加してコンパイラエラーを回避  
#include <mmsystem.h>  
#pragma comment(lib, "winmm.lib")  
timeBeginPeriod(1);  
// CVL でパルス幅を設定して I/O 操作を実行  
timeEndPeriod(1); // アプリケーション終了前に呼び出して戻る  
                // マルチメディアタイマが元の状態に戻る
```

『CVL User's Guide』の「画像の表示」章にある「ライブ画像の表示」セクションで、システムタイマの変更に関する注意点も確認してください。

ユーティリティ

このセクションでは、CVL 9.0 に付属するユーティリティプログラムに適用される制限と制約について説明します。

CVL コマンドラインユーティリティは、CVL bin ディレクトリ内に残しておく必要があります。

Cogtool などの CVL コマンドラインユーティリティは通常、Cognex テクニカルサポートの指示がある場合にのみ実行されます。このリリースの時点では、これらのユーティリティの実行可能ファイルは、CVL が提供する DLL を含むディレクトリ内に存在するはずで、つまり、ユーティリティの実行可能ファイル (`cogtool.exe` など) は `%VISION_ROOT%\bin\win32\cvi` に存在し、正しい結果を生成します。ユーティリティの実行可能ファイルを別のディレクトリにコピーしてそこから実行した場合、ツールは実行されますが、Cognex bin ディレクトリがこのパス内にある場合でも、システム内に Cognex ボードがないという間違った報告が行われます。

これは CVL 6.2 の動作からの変更点です。CVL 6.2 では、ユーティリティの実行可能ファイルを別のディレクトリにコピーして、そこから実行することが可能でした。

バーコード

このセクションでは、CVL 9.0 でのバーコードツールの制限事項について説明します。

decode() 関数に追加するパラメータによって下位互換性が失われる

CVL 5.5.0 ~ 5.5.1 では、ブール値のパラメータ `autoRetry` はデフォルト値の `true` ですが、引数リストに追加され、`ccAcuBarcodeTool` クラスのいくつかの仮想関数、具体的には `decode()` 関数に追加されていました。これによって関数の署名が変更されるため、CVL 5.5.0 で古い署名を使用してこれらの関数をオーバーライドしていた場合、CVL 5.5.1 以降でオーバーライドできなくなるだけでなく、オーバーライドしたものが呼び出されることもなくなります。関数の署名が変更されても、コンパイラエラーによって警告されることはありません。CVL アプリケーションを CVL 5.5.0 から 5.5.1 以降にアップグレードするときには、`ccAcuBarcodeTool::decode()` 関数をオーバーライドしたものが最新の関数の署名と一致することを確認してください。

プロブ

このセクションでは、CVL 9.0 でのプロブツールの制限事項について説明します。

プロブ: 参照によって SceneDescription を保持するとメモリがリークする

`ccBlobSceneDescription` オブジェクトを参照によって保持し、その後アーカイブからリロードすると、アプリケーションでメモリがリークします。これを回避するには、参照ではなくポインタによって `ccBlobSceneDescription` オブジェクトを保持します。

キャリブレーション

このセクションでは、CVL 9.0 でのキャリブレーションツールの制限事項について説明します。

チェッカーボードの最小タイルサイズ

フューチャーレスポンスキャリブレーションを実行するときに、15x15 ピクセルより小さいサイズのタイルでは、機能検出が確実に機能しません。ほとんどのアプリケーションでは、チェッカーのサイズを 15x15 より大きくして、範囲の限界で動作しないようにする必要があります。

Caliper

このセクションでは、CVL 9.0 での Caliper ツールの制限事項について説明します。

フィルタリングおよびクリッピングを行ったプロジェクションのピクセル範囲が正しくない

フィルタリングおよびクリッピングを行ったプロジェクションのピクセル値の範囲は正しくありません。フィルタリングされた画像での範囲は $-255 \sim +255$ ですが、 $-255*n \sim +255*n$ にする必要があります。ここで n はプロジェクション内のサンプルの数です。正しい範囲を取得するには、フィルタリングされた画像を再スケーリング (n で乗算) します。

ガウス (ガウシアン) サンプリング

このセクションでは、CVL 9.0 でのガウス (ガウシアン) サンプリングツールの制限事項について説明します。

cfGaussSample() によって生成される結果が正しくない

16 ビットの取り込み先ピクセルバッファクランプを使用する `cfGaussSample()` オーバーロードは、32,767 の値となります。したがって、通常 32,767 より大きい結果を生成する値 (例えば、128 より大きい入力ピクセル値を持つスケール設定 255) は正しい結果を生成しません。

回避策

すべての結果が 32,767 未満になるように、スケールを低く設定します。

グレースケールモフォロジィ

このセクションでは、CVL 9.0 でのグレースケールモフォロジィツールの制限事項について説明します。

3x3 カーネル以外の構成を持つ NxM モフォロジィでマルチコアが完全に最適化されない

3x3 カーネル以外の構成を持つ NxM モフォロジィの可変サイズカーネル画像ツールでは、マルチコアが完全に最適化されません。

ID

このセクションでは、CVL 9.0 での ID ツールの制限事項について説明します。

EAN.UCC コンポジットシンボルでは RSS コンポジットシンボルのみがサポートされる

このリリースでサポートされている唯一の EAN.UCC コンポジットシンボルは、RSS コンポジットシンボルを使用するものです。

acuRead

このセクションでは、CVL 9.0 での acuRead ツールの制限事項について説明します。

ユーザー定義のフォントで末尾の文字が欠落する

ユーザー定義のフォントで、末尾の文字が欠落する場合があります。これは固定長モードおよび可変長モードの両方で該当します。固定長モードでは、最後の文字がスペースになります。

回避策

ROI を拡大して、先頭および末尾で空白のピクセルを増やします。

OCV

このセクションでは、CVL 9.0 での OCV ツールの制限事項について説明します。

空白スペースであっても文字が誤って照合される

行配置内での位置が、現在のキーの間で空白文字によって定義されている場合でも、その位置が実行時の画像で空白になっておらず、その他の現在のキーのいずれとも一致しない場合は、OCV ツールでその位置が正しく照合されないか、混同されることがあります。実行時に検索位置に実際に何があるかに関わらず、空白の現在のキーは常にゼロ以外の得点を返します。空白のキーに対して返される得点は通常は低いですが、「-」や「|」などの幅の狭い文字に関しては、得点がアクセプトしきい値またはコンフュージョンしきい値を超える場合があるので、照合は失敗しません。

空白文字の得点の例を以下に示します。

| 文字 | 得点 |
|----|-------|
| - | 0.423 |
| . | 0.422 |
| ' | 0.382 |
| : | 0.312 |
| , | 0.276 |

位置に対して定義されている現在のキーが複数か単一に関わらず、この問題は発生しますが、空白ともう 1 つの現在のキーを合わせた得点が、空白のみの場合よりも低くなっていたインスタンスが見つかっています。この問題では OCV ツールの設計を処理する必要があります。ここで空白と「標準の」文字の扱いは異なります。OCV では空白文字の混同は何もチェックせず、空白状態の得点を確認するだけで、空白文字の状態を確立します。このため、小さい文字は見落とされやすい可能性があります。

PatMax

このセクションでは、CVL 9.0 での PatMax ツールの制限事項について説明します。

人工モデル登録の問題

登録時の変換を利用して画像座標へ投影するときに、2Kx2K よりも著しく大きい境界ボックスで図形を使用して PatMax を登録すると、アクセス違反となるか、またはアプリケーションが train() 内からハング状態になります。

修正が完了したバグ

このリリースでは、以前のリリースでお客様から報告された問題は修正されています。

エラーを生成する輝度およびコントラストの値

画像取り込みの際、間隔が 1 ではないカメラの無効な同等の設定にマッピングされると、輝度およびコントラストの値によってエラーが生成されることはなくなります。

このリリースに付属するドキュメンテーション

CVL インストールには以下のタイトルが含まれており、[Start] -> [Cognex] -> [CVL] -> [Documentation] から使用できます。

- CVL User's Guide
CVL プログラミングの概要。サンプルコードを使用した操作、画像およびグラフィックの取り込みと表示、ビジョンツールの概要および概念が含まれています。
- CVL Vision Tools Guide
各 CVL ビジョンツールの機能、それぞれが受け入れるパラメータ、および生成される結果に関する説明。
- CVL クラスリファレンス
CVL C++ クラスライブラリに関するクラス別のリファレンス
- CFG-8724 Hardware Reference
GigE Vision 画像取り込みに関して、CFG-8724 PCI Express x4 準拠フレームグラバデバイスの詳細を説明しています。

将来のリリースに関する注意

将来の CVL リリースでは、以下の API のサポートがなくなります。

- 特定の 2D シンボルに関連付けられた ccAcuSymbol*クラス:
 - ccAcuSymbolDataMatrixDefs
 - ccAcuSymbolDataMatrixLearnParams
 - ccAcuSymbolDataMatrixTool
 - ccAcuSymbolDefs
 - ccAcuSymbolFinderParams
 - ccAcuSymbolLearnParams
 - ccAcuSymbolQRCodeDefs
 - ccAcuSymbolQRCodeLearnParams
 - ccAcuSymbolQRCodeTool

-
- 基本の OCR 読み取りおよびチューニングの機能に関連付けられた ccAcuRead* クラス:
 - ccAcuRead
 - ccAcuReadDefs
 - ccAcuReadFont
 - ccAcuReadResult
 - ccAcuReadResultSet
 - ccAcuReadRunParams
 - ccAcuReadTuneParams
 - 1D シンボルツールに関連付けられた ccAcuBarCode* クラス:
 - ccAcuBarCodeCalibrationResult
 - ccAcuBarCodeDefs
 - ccAcuBarCodeResult
 - ccAcuBarCodeRunParams
 - ccAcuBarCodeTool
 - ccAcuBarCodeTuneParams

