

2017/12/18 第5回 WINDSネットワーク セミナー

本印刷用資料には、以下の内容は含まれておりません。

- 12月14日以降のニュースリリースに関する情報
  - セミナー当日のライブの技術デモに関する情報
- セミナー当日のスライドは、セミナー終了後に公開します。

# 高速ビジョン開発プラットフォーム 「HSV SDK」の誕生



株式会社エクスビジョン

取締役 加治佐 俊一

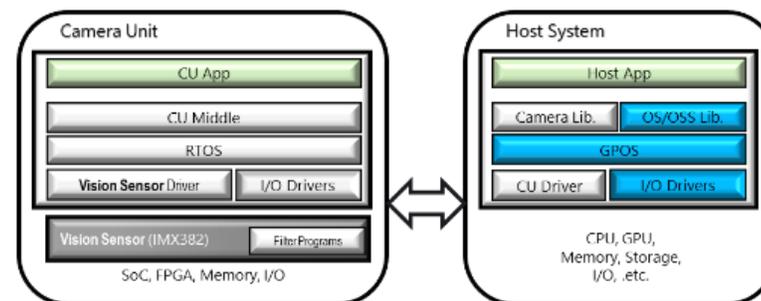
<http://www.exvision.co.jp>

**exvision**  
beyond the human vision

この成果は国立研究開発法人 新エネルギー・産業技術総合開発機構（NEDO）の委託業務の結果得られたものです

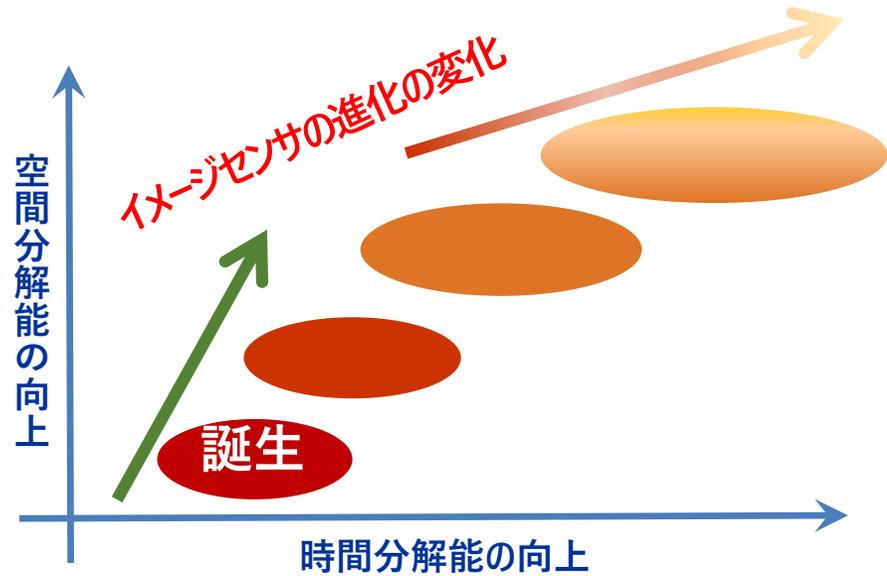
# 本日の内容

- 会社紹介
- HSV SDK の発表と概要
- 技術紹介デモ
- 技術概要
- 今後の展開
  
- Q&A



# 高速ビジョンが実現する新たな世界/価値

## 空間分解能の向上から時間分解能の向上へのシフト



### 時間分解能の向上

- 高速事象の検出、検知、追跡が可能に
- 人間の目視から自動検査、制御へ

様々な応用展開



FA高速検査



高速ロボット



自動車-交通



セキュリティ



バイオ・医療



高速3D計測



映像メディア



ヒューマンインターフェイス

# 高速画像処理と高速ビジョン

- エクスビジョンは、高速ビジョンによる事業を展開します
- 高速ビジョンシステムのプラットフォームとソリューションを構築します

## 高速ビジョン

- 処理系、運動系、出力系の組合せ
- システムとしての機能

## 高速画像処理

- 高速な撮像
- 高速な画像処理

**exvision**  
beyond the human vision

# 高速ビジョンによる事業展開

EGS Touch



EGS: Exvision Gesture System

EGS Motion



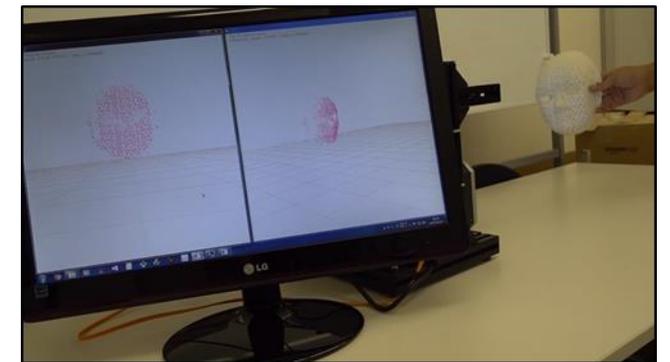
Auto Pan-Tilt



Active Projection Mapping



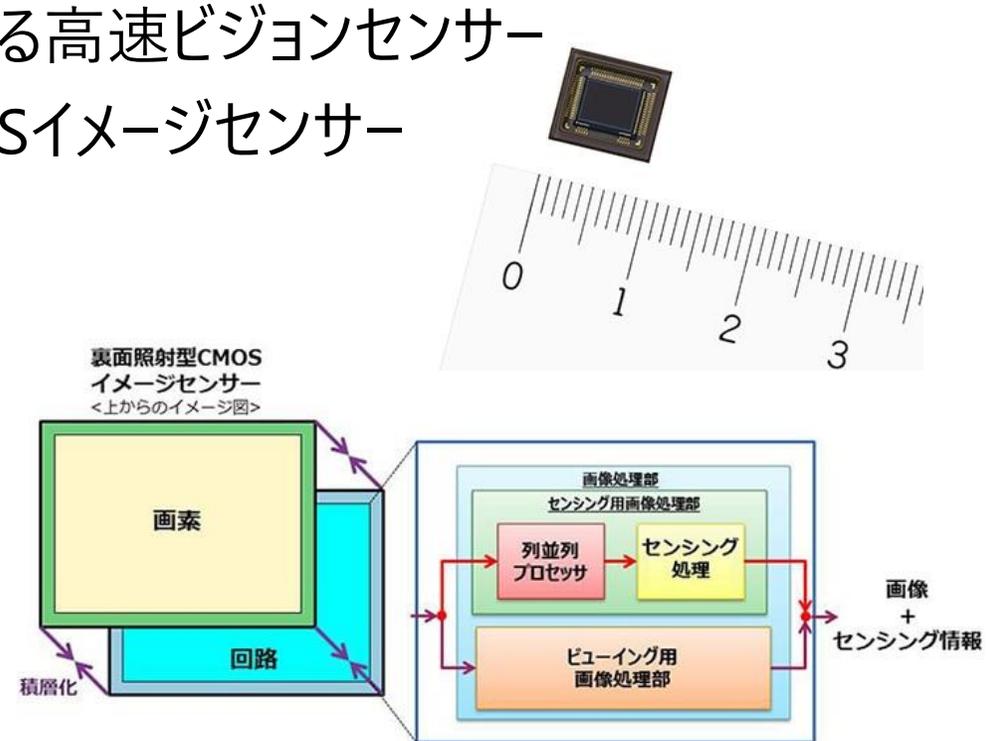
3D Measurement



# ソニーセミコンダクタソリューションズ株式会社 高速ビジョンセンサー「IMX382」

- 毎秒1,000フレームで対象物の検出と追跡を実現する高速ビジョンセンサー
- 高速撮像と高速センシングを融合した積層型CMOSイメージセンサー

有効画素数	1304 (H) × 976 (V) 127万画素		
イメージサイズ	Diagonal 5.68mm (Type 1/3.2)		
ユニットセルサイズ	3.5μm (H) × 3.5μm (V)		
フレームレート	ビューイング	60fps 120fps	Quad-VGA (1280×960) 12bit Quad-VGA (1280×960) 10bit
	センシング	500fps 1000fps	Quad-VGA (1280×960) 4bit 640×470 4bit
電源電圧	2.9V / 1.8V / 3.3V / 1.1V		
画像フォーマット	Bayer RAW / 白黒		
出力	MIPI (CSI2) D-PHY 864Mbps/lane		

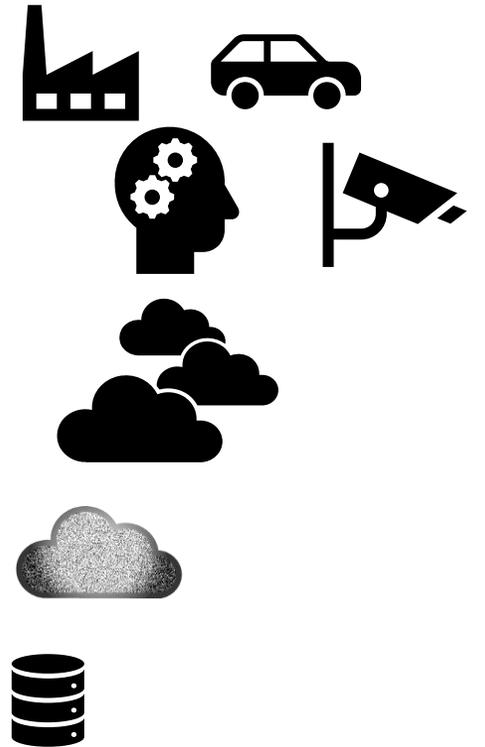
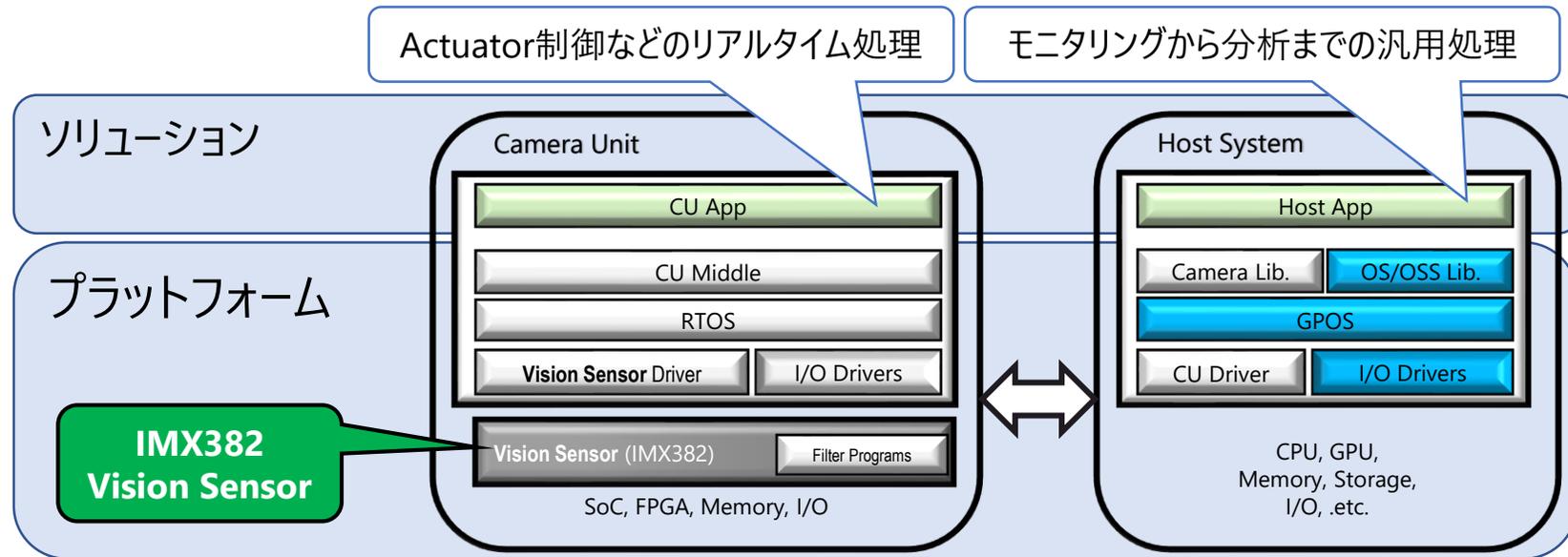


2017/5/16発表 ソニーセミコンダクタソリューションズ株式会社 ニュースリリースより抜粋:  
<https://www.sony.co.jp/SonyInfo/News/Press/201705/17-051/index.html>

# HSV SDK のアーキテクチャ

## 高速ビジョン 開発プラットフォーム

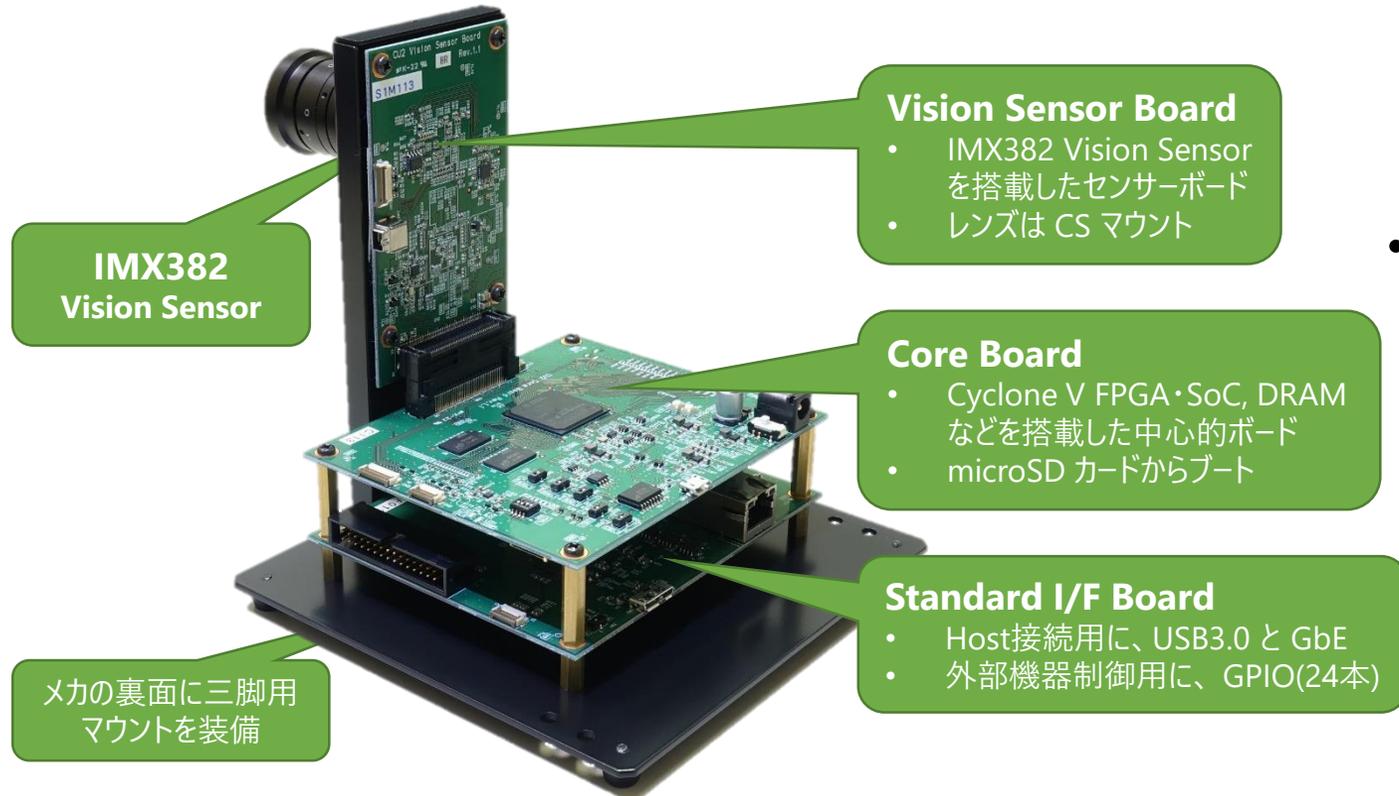
- IMX382 の 1000fps での高速センシングによる特徴量の取得と処理
- リアルタイム性と汎用性を両立した開発プラットフォーム
- Host System を必要としない Camera Unit 単独でスタンドアロンの構成が可能
- 標準的な開発環境による短期間で容易なソリューション開発と展開



# HSV SDK の基本構成

## ハードウェア (評価ボード)

- Camera Unit (CU)

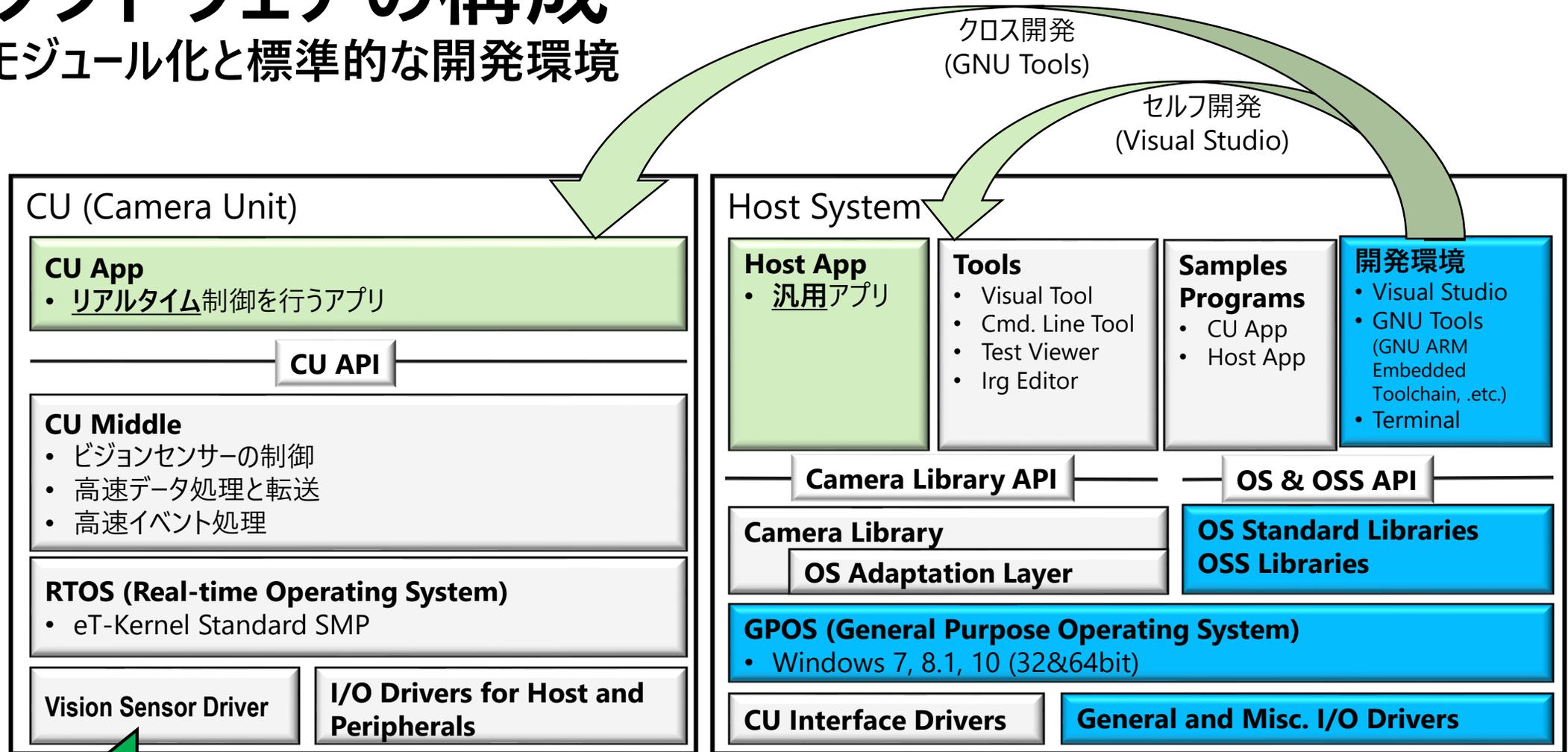


## ソフトウェア

- Camera Unit
  - CU Apps
  - CU Middle
  - Real-time OS
  - Device Drivers
- Host System
  - Host Apps
  - Camera Library
  - General Purpose OS
    - Windows (標準対応)
    - Linux (カスタマイズ対応可能)
  - Device Driver
  - Tools
  - Sample Programs (Host Apps, CU Apps)
  - 開発ツール (Visual Studio, GNU Tools, ..)

# ソフトウェアの構成

## モジュール化と標準的な開発環境



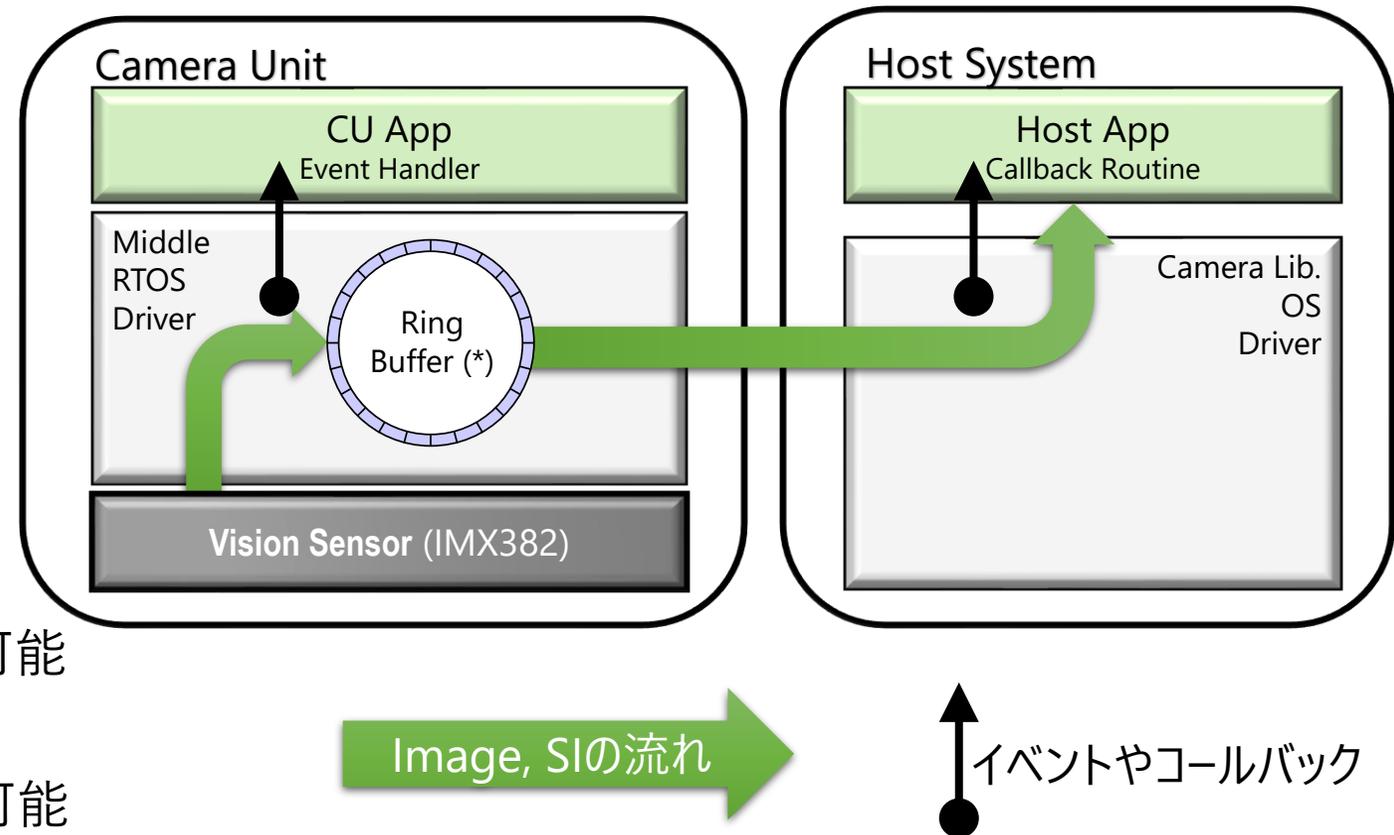
IMX382  
Vision Sensor

注: モジュールの種類を色で分類 →

SDKで開発する アプリ	SDKに含まれる モジュール	開発者が用意する モジュール
-----------------	-------------------	-------------------

# センシングモード センシング情報と画像データの流れ

- センシング情報 (SI)
  - 検出状態
  - 重心 (x, y)、面積(pixels)、など
- 画像データ (Image)
  - 4-bit と 1-bit画像
- CU App
  - 全ての SI と Image を処理可能
  - Host にカスタマイズした情報を通知可能
- Host App
  - SI と Image をそれぞれ間引き転送可能
  - 条件指定でリングバッファに蓄積された SI と Image を転送可能



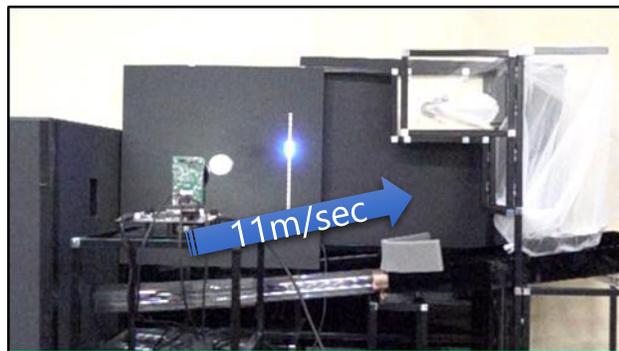
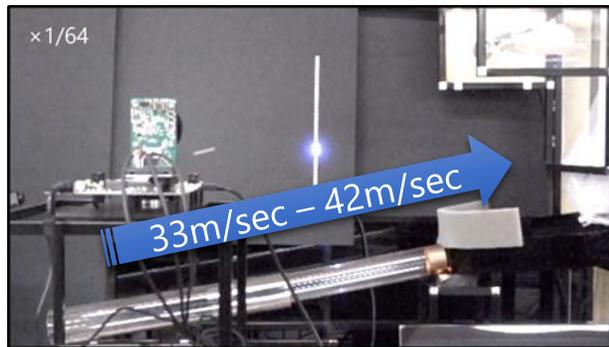
(\*)1000fpsの場合、最新の約5.9秒、5,946フレームを蓄積可能

# 技術紹介デモ



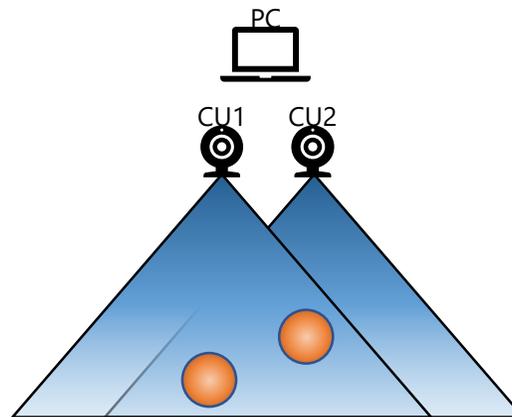
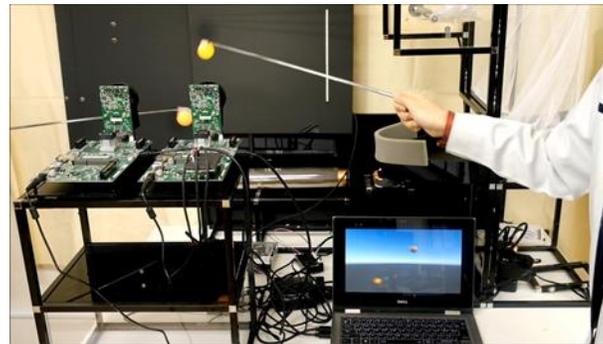
# 2017/3/21 WINDSネットワーク設立1周年記念総会・セミナー 前回の技術紹介デモ

Detect & Catch (BB弾とピンポン玉)



ビデオ: <https://youtu.be/pytrfJ8RQuw>

Stereo Vision (平行ステレオ)



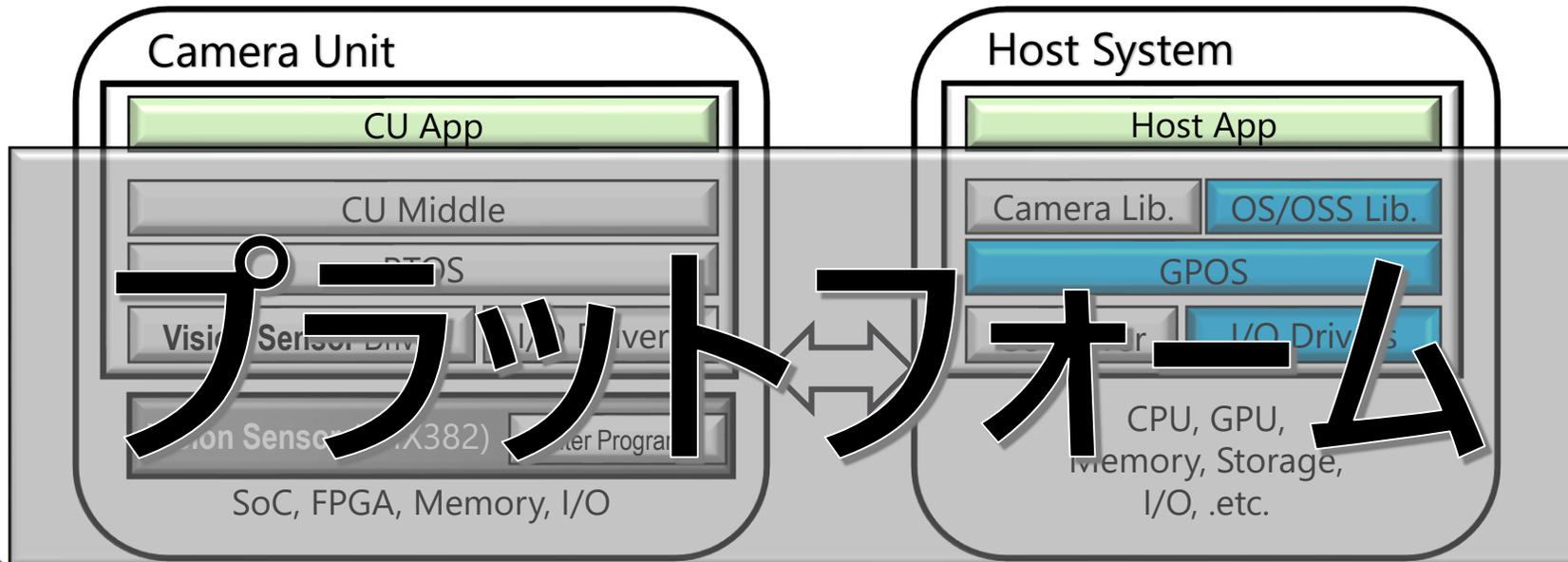
Pattern Tracking



# 前回技術紹介デモの構成

CUとHostのアプリを目的に合わせて開発する

デモ	CU App	Host App	その他
Detect & Catch	<ul style="list-style-type: none"> <li>Host Appからの検出やモーターの回転数などの制御パラメータを設定。</li> <li>1msごとにセンシング情報からリアルタイムにボールを検出。</li> <li>検出されたボールの色を判別し、LED点灯とモーター回転制御。</li> <li>検出されたボールの情報をHost Appに通知。</li> </ul>	<ul style="list-style-type: none"> <li>CU-Appのロードと内部パラメータの表示、設定。</li> <li>検出、トラッキング中のカメラ映像の表示、検出前後のカメラ画像データの保存と再生。</li> <li>CU Appから通知される検出情報による速度などの表示。</li> </ul>	
Stereo Vision	<ul style="list-style-type: none"> <li>無し</li> </ul>	<ul style="list-style-type: none"> <li>CU1とCU2からの各2次元座標を3次元に変換</li> <li>Unityのプログラムに3次元座標を送る</li> </ul>	
Pattern Tracking	<ul style="list-style-type: none"> <li>無し</li> </ul>	<ul style="list-style-type: none"> <li>Toolに含まれているViewerが、CUからのImageとセンシング情報を受信し、Imageと座標の表示</li> </ul>	<ul style="list-style-type: none"> <li>Vision Sensorで、Filter Program を動作</li> </ul>



# 技術紹介デモ

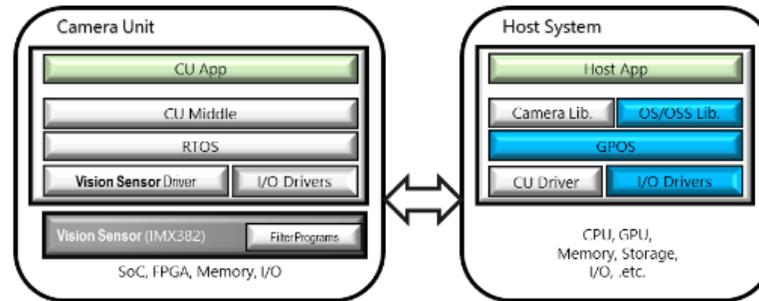
センサーのモード、フレームレート、画像フォーマット	アプリ及び補足
ビューイング モード、120fps、白黒	Host App
センシング モード、1000fps、白黒	Host App
センシング モード、1000fps、白黒	CU App, Standalone (No Host)
センシング モード、1000fps、Bayer (カラー)	Visual Tool, CU x 3

今回のデモのテーマ – 基本的なモードを試してみよう

```
#include <stdio.h>
main()
{
    printf(“hello, HSV world¥n”);
}
```

身近な道具を使って、  
技術紹介のデモを行います

# 技術概要



# ソフトウェア環境

- CU

- リアルタイム制御用OSとして eT-Kernel Standard SMP
- Intel Cyclone V
  - Dual Core ARM Cortex-A9 MPCore /925MHz
- GPIO の出力をプログラマブルに変更可能
- microSDカード/USB3.0/GbE から CU App はロード・実行可能
- Host との接続
  - 高速に画像データとセンシング情報を転送したい場合には、USB3.0で接続
  - センシング情報を主体に転送を行う場合は、GbEで接続

- Host

- Windows7/8.1/10 (32bit 及び 64bit)
- Visual Studio2015/2013にて開発

# CU App と Host Appの特徴

アプリ	リアルタイム性	制御インターフェイス	開発言語	画像処理
CU App	◎	GPIO	C言語	△
Host App	△	USB/Ethernet/他	C言語および、各種言語 ※1	OpenCV等

※1：Camera Libraryをロードし、各種APIを実行可能

- CU App
  - GPIOを経由しての Actuator制御などのリアルタイム処理
  - Vision Sensor からの全てのフレームのセンシング情報と画像データを、間引くことなく処理することが可能
  - DRAMのリングバッファに格納された過去のセンシング情報と画像データを、直接参照することが可能
  - ホストにセンシング情報や画像データを伴ったアプリ独自の通知を行うことが可能
- Host App
  - モニタリングから分析までの汎用処理
  - Vision Sensorからのフレームのセンシング情報と画像データを、それぞれ間引き指定して受信することが可能
    - 例: Vision Sensor 1000fpsに設定し、Host転送でのセンシング情報は1000fps, 画像データは30fps
  - CUのリングバッファに格納された過去のセンシング情報と画像データを転送することが可能

# CU App 開発

## 開発環境

- GNU ARM Embedded Toolchain : 5-2016-q3-update  
<https://launchpad.net/gcc-arm-embedded>
- GNU on Windows : 0.8.0  
<https://github.com/bmatzelle/gow/releases>

## CU API

- センサーボードへのアクセス
- GPIOアクセス
- 同期/イベント/メッセージ
- スレッド
- ハンドラーの登録・解除
- ログサービス

# Host App 開発

## 開発環境

- Windows7/8.1/10
- Visual Studio 2015/2013

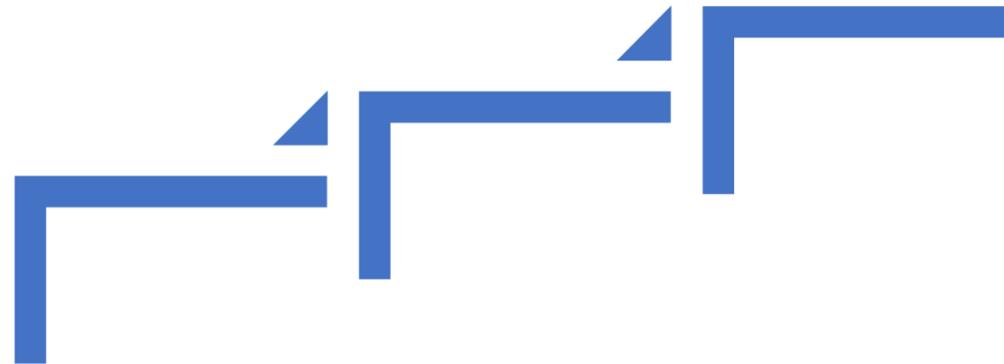
## Camera Library API

- カメラユニットの初期化、および各種設定
- センサーレジスターへの読み書き
- 画像データ・SI情報の転送、取得、および保存
- CU Appのロード・アンロード
- 排他制御（複数アプリでの使用）

# 一般的な開発のステップ

- ターゲットの検出、追跡の為の環境の設定と確認
  - 設置環境、光学的環境(フリッカー軽減、赤外線照射)などを工夫する
  - Visual Tool と IrgEditor を使い、Vision Sensor の設定を調整し、ターゲットを検出・追跡し、適切に SI と Image が取得できることを確認する
  - 1-bit/4-bit 画像データを処理する場合には、目視に加えてデータを保存して解析することで処理可能かを検討する
  - 解像度を優先した500fpsと、高速性を優先した1000fpsの両方で評価・検討する
- アプリ開発
  - 目的とするソリューションを、Host Appとして開発する
  - Host Appで、目的とするリアルタイム性が十分に達成できない場合には、リアルタイム性が必要なコンポーネントをCU Appとして開発し、Host AppからCU AppをCUにロードして、実行する
  - HostでのGUIやデータ処理が不要な場合には、CU スタンドアロンとして実行する

# 今後の展開



# 高速ビジョン ソリューションの普及に向けて

優れた研究成果を実ソリューションに広げる

## 優れた研究成果

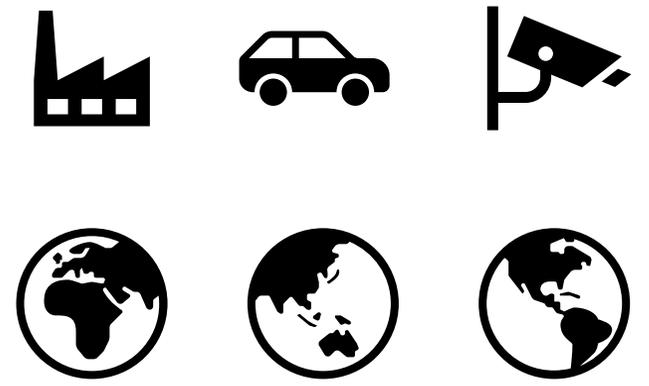
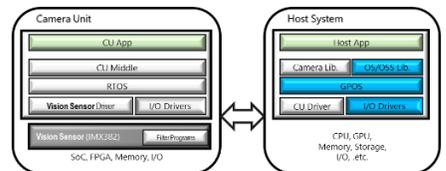
@東京大学 石川渡辺研究室など



- ### 目標
- 低価格化と小型化
  - システム構成の簡素化
  - 標準化されたソリューション
  - 開発期間の短縮化
  - 分野毎に最適な応用展開

- ### 解決策
- 高速ビジョンセンサー (IMX382)
  - 開発プラットフォーム (HSV SDK)
  - 業界連携 (WINDSネットワーク)

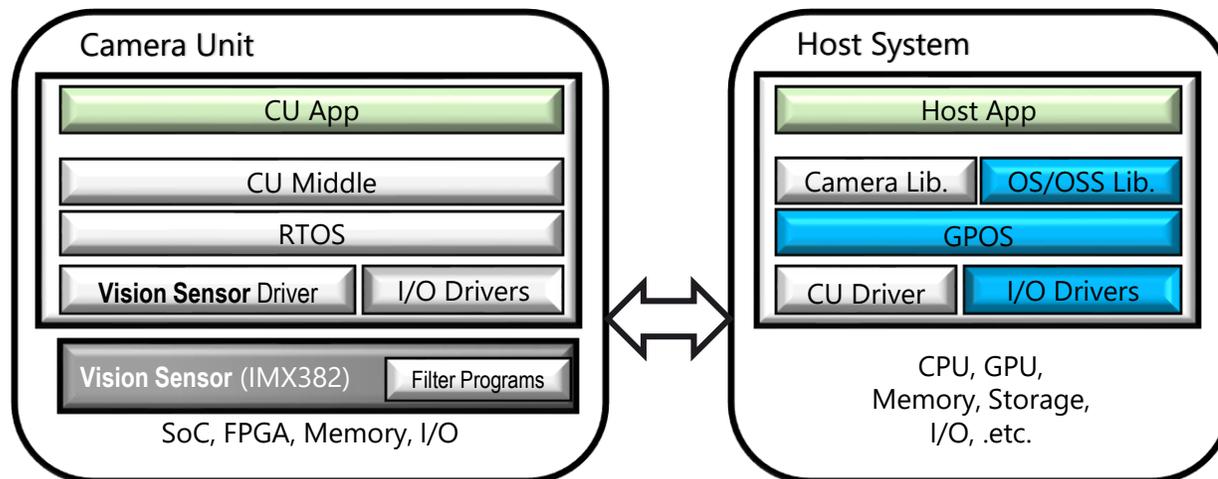
- ### 普及に向けての課題
- 高性能のCPU&GPUが必要
  - 目的毎の作りこみが必要
  - 設計・開発の知識・ノウハウが不足
  - 高速ビジョンの役立つ分野が不明



# カメラ製品や組み込みシステムへの展開

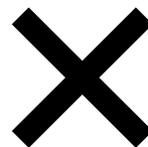
様々なシステム構成に対応可能な基本設計と技術力

標準構成



実システムでの構成

CU Components	Options
SoC	ARM, .etc. Single, dual, ...
FPGA	None, Small, Medium, Large
OS	Bare-Metal, RTOS (, or GPOS)
CU App	Yes / No
Host Connection	None, USB, GbE, EtherCAT, .etc.
Actuator I/F	GPIO, SPI, I2C, .etc.



Host Components	Options
OS	Windows Client/Embedded, Linux, others.
CPU	ARM, x86, and x64 Single, dual, quad, many core.
GPU	Integrated or external
Network and Cloud	Yes / No
# CU connection	Single, Dual, or many

# 今後の予定と展開

2018年1月 一般リリース開始

## HSV SDK 提供開始

- 開発用プラットフォームの提供
- 高速ビジョン技術の評価
- ソリューションのプロト開発と検証

## カメラ製品出荷

- 実ソリューション開発
- 試験運用と本運用
- 導入事例の共有
- HSV SDKの継続的な機能拡張

## 様々なソリューション展開

- カメラ製品ラインアップの広がり
- 組込み製品への対応
- 分野別標準ソリューションライブラリ
- 他のプラットフォームとの連携
- 技術ノウハウの共有 (と保護)

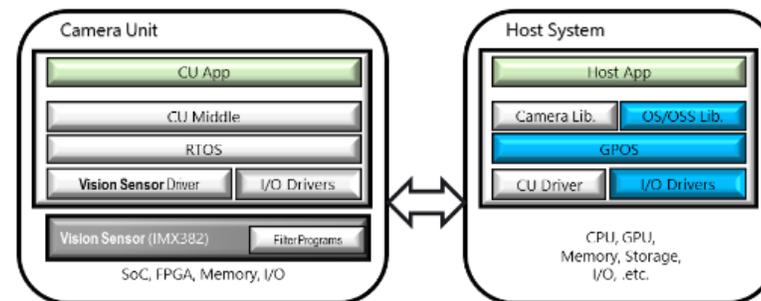
# 高速ビジョン ソリューション展開の可能性

撮像と同時に特徴量を抽出し、予測動作不要で高速、正確な制御が可能

- FA・検査
  - 高速インサーター
  - 高速計測・計量装置
  - 社会インフラ (トンネル、鉄道など) の傷や障害物の検知
- 高速ロボット
  - 高速ロボットハンド
- 自動車・交通
  - 振動検出・補正の次世代 HUD (Head-up Display)
  - マーカー検出によるDrone 制御
- セキュリティ
  - 顔の検出・追跡による高精細画像撮影
  - 高速移動物 (人や物) の自動検出、記録、通報
- 高速 3D 入力
  - 高速 3D 形状計測
  - 高速 3D 座標計測
- ヒューマンインターフェイス
  - VR/ARのHMDでの高速ジェスチャーや遠隔マウス・タッチ
  - デジタルサイネージでの活用
- バイオ・医療
  - 微生物の検知追跡
  - 医療検査装置での応用
- 映像・メディア
  - スポーツ中継や練習での応用
  - 動的なプロジェクションマッピング
- 新規 IoT関連分野
  - 様々なセンサーや小型プロセッサの組み合わせによるインテリジェント IoTの開発とクラウドとの連携

# QA

- 会社紹介
- HSV SDK の発表と概要
- 技術紹介デモ
- 技術概要
- 今後の展開



# exvision

beyond the human vision

お問い合わせ窓口:

- [hsv SDK req@exvision.co.jp](mailto:hsv SDK req@exvision.co.jp) - HSV SDK および 高速ビジョンのソリューションについてのご相談
- [contact@exvision.co.jp](mailto:contact@exvision.co.jp) - それ以外のご相談やお問い合わせ

# 高速ビジョンによる事業展開

名前	最小受注量	納期	製品・ソリューション概要
EGS Touch	100 -	6カ月(*)	リモート 2点タッチ。人の両手の動きを座標で認識し、低遅延の2点タッチデバイス。Windows と Androidに対応。ハードウェア・ソリューション。ライセンス提供可能。
EGS Motion	10 -	2カ月(*)	リモート・ジェスチャー・コマンド。一連の動きを認識して、コマンドに変換をしてコンピューターに入力。(例：円の右回りでの描画ジェスチャー → 音量アップ) ハードウェアとソフトウェアの両方のソリューションが可能。ライセンス提供可能。
Auto Pan-Tilt	1 -	応相談	高速移動する物体を1000fpsの高速ビジョンによるミラー制御技術で、静止状態で見ることが可能。最大3台を接続し、3次元の軌跡を計測可能。
Active Projection Mapping	1 -	応相談	動く対象をトラッキングして、その上にアニメーションをシナリオに応じて動的にプロジェクション。お客様のシナリオにより、プログラミング可能。一つのシステムで、複数の対象に対応可能。イベントやショールームでの活用。
3D Measurement	1 -	応相談	高速構造照明による高速3次元形状計測。対象物に対し、測定用のパターン画像を対象物に対して高速に投影し、高速に撮影・処理することで、高速移動対象物をリアルタイムに3次元形状計測。
HSV SDK	1 -	2018/1-	高速ビジョンセンサーを使った評価ボードが含まれるソフトウェア開発キット。様々な高速ビジョンソリューションを簡単に短期間で開発可能な開発プラットフォーム。
HSV Solutions	1 -	応相談	お客様の要望に応じた 高速ビジョンソリューションの開発。要求仕様によりHSV SDKを適用可能か検討。HSV SDKを使わない開発も可能。

(\*) 時期により数量の調整や納期の短縮が可能になる場合があります。ご相談ください。