# Intel RealSense Technology

*Bringing Human Senses to Your Devices*

*September 2014*

인텔코리아 홍승표

Intel Confidential — Do Not Forward

# Introduction

What's the Intel RealSense Technology?

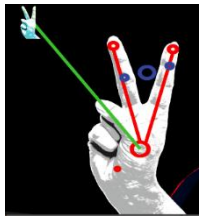
# Interaction with Computing Devices

## What's the Next?

Local Speech Recognition

Facial Tracking

Finger/Gesture Tracking



Augmented Reality

Background Subtraction

# Intel® RealSense ?

Eyes, Ears, Voice, Touch, Emotion and Context for experiences that are….

## Natural

## Intuitive

## Immersive

For Every App Category: Games, Productivity, Retail, Enterprise, Health, and More

# Intel® RealSense Mission

Adding “Human-like Senses” to Computing Devices

Natural, Intuitive and Immersive Experience

# 3D Camera for Every Form Factor

All-in-One | Notebook | Convertible | Detachable | Tablet | Phablet | Phone

FRONT Camera

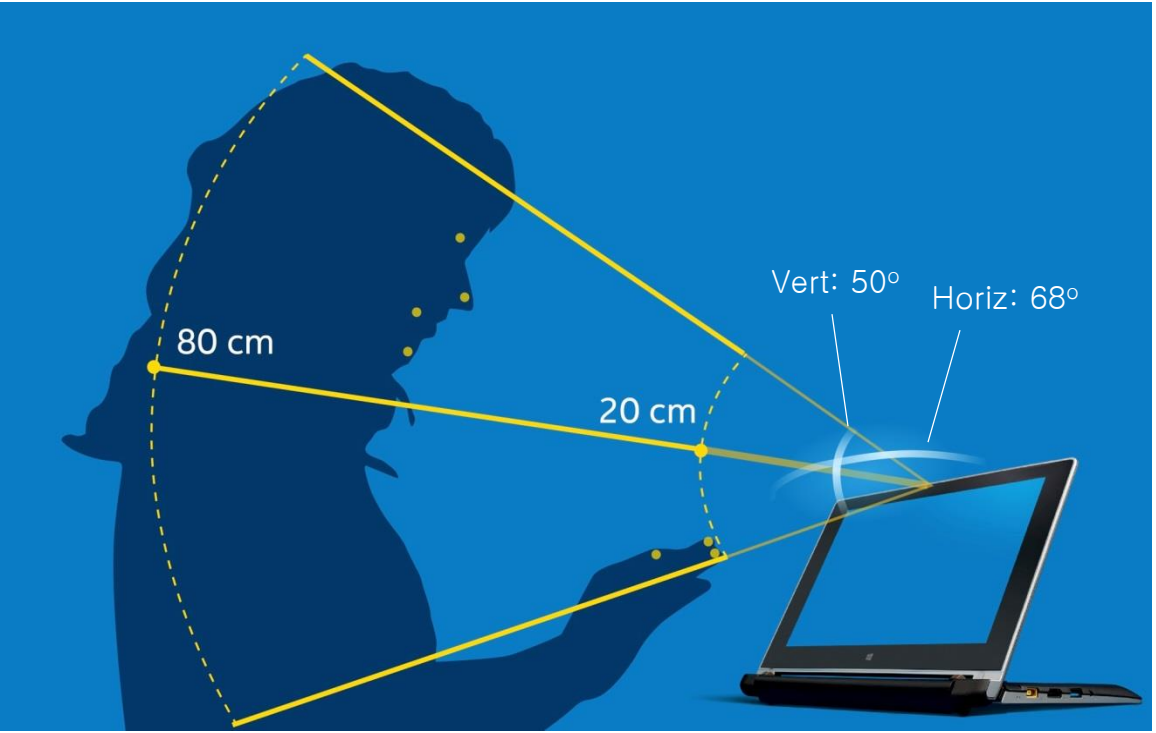
Intel® RealSense™ 3D Camera F200 (aka Bell Cliffs 60fp

REAR Camera

Intel® RealSense™ 3D Camera R200 (aka Double Spring 60fps)

Intel® RealSense™ Snapshot R100 (Photography) (aka Davis Reef)

# Intel® RealSense™ 3D Camera Module




The Intel® RealSense™ 3D camera depth module is integrated into the bezel of Ultrabook™, All-in-One PC, Notebook, Ultrabook™ 2 in 1, and Convertible / Detachable OEM partner devices. It is optimized for depth perception and can track 22 joints of the hands and 78 facial landmarks for face orientation and tracking.

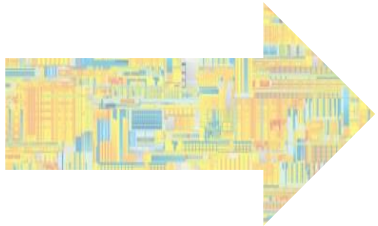
- Thin form factor – integration!
- Improved performance, accuracy/depth quality
  - VGA depth camera
  - HD 1080p RGB camera
- Reduced power
- RGB: 1080p30
- Depth: VGA@60

## Use Intel® RealSense™ 3D Camera for Face, Hand Tracking and Orientation

# Intel RealSense SDK 2014

Transition from Perceptual Computing SDK 2013 to RealSense SDK 2014

# 2014 Depth Camera Module



- Portable, USB 2.0 powered peripheral
- QVGA depth camera
- HD 720p RGB camera
- Dual microphones
- Bundled SW
- RGB: 720p
- Depth: QVGA@60

- Thin form factor – integration!
- Improved performance, accuracy/depth quality
  - VGA depth camera
  - HD 1080p RGB camera
- Reduced power
- RGB: 1080p30
- Depth: VGA@60

User-facing module placed on the same plane as the display for Ultrabook, AIO, Notebook and Convertible/Detachable designs

# What's New with the 2014 SDK?

## New Modalities

3D Scanning

Depth-enabled Augmented Reality

Grammar for Speech

## Enhanced Algorithms

Enhanced Hand/Finger tracking

3D Facial Analysis

**Intel® RealSense™ SDK for Windows 2014**

## Extended Reach

Next Generation Integrated Depth Camera

Windows* 8.1 Desktop

## Developer Tools

Unity* ToolKit

Productivity Extensions

** All products, software, computer systems, dates and figures specified are preliminary based on current expectations, and are subject to change without notice.
*Any code names featured are used internally within Intel to identify products that are in development and not yet publicly announced for release. Customers, licensees and other third parties are not authorized by Intel to use code names in advertising, promotion or marketing of any product or services and any such use of Intel's internal code names is at the sole risk of the user.

# What's New in the Intel® RealSense™ SDK 2014?

## New Features and Responding to Developer Feedback

| | SDK 2013 | SDK 2014 | Benefits |
|---|---|---|---|
| Camera Support | Creative* Senz 3D* | Bell Cliffs* Integrated Camera | Smaller, lighter, lower power, increased installed base |
| Camera Specs | RGB: 720p<br>Depth: QVGA@60<br>FOV: 53 deg<br>USB2.0 Peripheral | RGB: 1080p30<br>Depth: VGA@60<br>FOV: 85 deg<br>USB3.0 integrated | Higher resolution and accuracy |
| OS Support | Windows* 7/8 DT | Windows* 8.1 DT | New Windows* version |

Developers asked for

- Increased camera resolution and quality
- Increased finger/hand/face tracking quality
- Face and head orientation with depth

** All products, software, computer systems, dates and figures specified are preliminary based on current expectations, and are subject to change without notice.
*Any code names featured are used internally within Intel to identify products that are in development and not yet publicly announced for release. Customers, licensees and other third parties are not authorized by Intel to use code names in advertising, promotion or marketing of any product or services and any such use of Intel's internal code names is at the sole risk of the user.

# What's New in the Intel® RealSense™ SDK 2014?

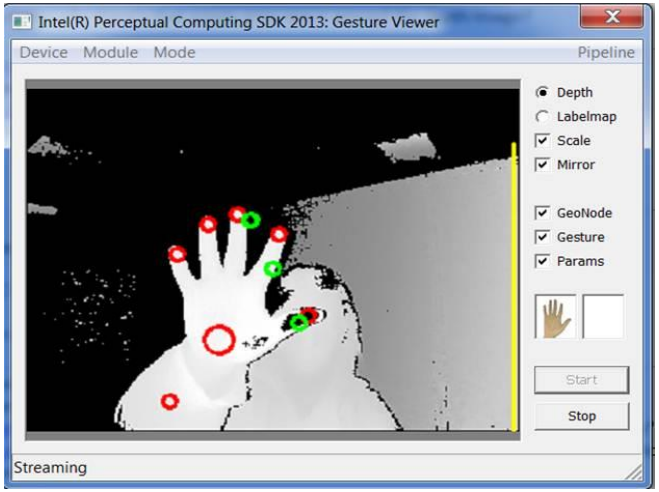
| | SDK 2013 | SDK 2014 | Benefit |
|---|---|---|---|
| Modality | Hand / finger tracking | + Hand/finger joint tracking<br>+ Enhancements | Track all 22 joints in hand for more complex interactions<br>Greater accuracy, more robust tracking |
| | Facial Analysis (RGB) | + Head orientation<br>+ Depth | Head orientation tracking and greater accuracy and resolution. |
| | Voice Recognition | + Custom grammar | Developers can add customized commands to their apps. |
| | Augmented Reality(RGB) | + 3D object tracking<br>+ 3D environment tracking | Track 3D objects more accurately using depth data. Track 3D environments for accurate 3D augmentation |
| | Background Subtraction | + Enhancements | Remove background for 'green screen' effect OR define a z-depth plane of interest to specify the interaction range |
| | -- | +3D Scanning | Easily capture a complete, fully textured 3D model of objects for exciting new possibilities. |

** All products, software, computer systems, dates and figures specified are preliminary based on current expectations, and are subject to change without notice.
*Any code names featured are used internally within Intel to identify products that are in development and not yet publicly announced for release. Customers, licensees and other third parties are not authorized by Intel to use code names in advertising, promotion or marketing of any product or services and any such use of Intel's internal code names is at the sole risk of the user.

# Advanced Finger & Hand Tracking

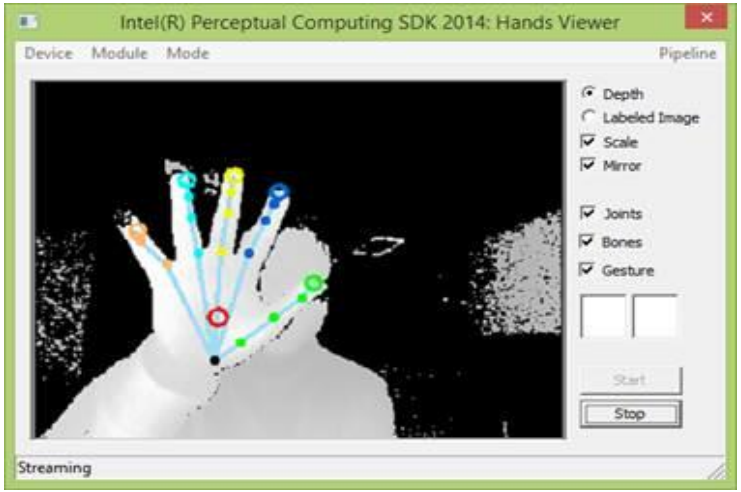
## 2013:

Fingertips, Palm Center, Grasp Point, Forearm




## 2014:

22 Joints




Greater Accuracy and Resolution

3D Hand Skeleton enables new usages

** All products, software, computer systems, dates and figures specified are preliminary based on current expectations, and are subject to change without notice.
*Any code names featured are used internally within Intel to identify products that are in development and not yet publicly announced for release. Customers, licensees and other third parties are not authorized by Intel to use code names in advertising, promotion or marketing of any product or services and any such use of Intel's internal code names is at the sole risk of the user.

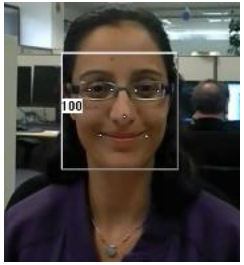
# Advanced Finger & Hand Tracking

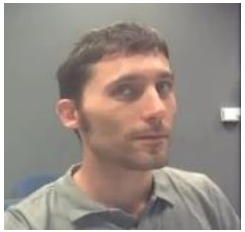
| Feature | SDK 2013 | SDK 2014 | 2014 Developer Benefit |
|---|---|---|---|
| Ten finger tip tracking | ✔ | ✔ | Enable end-user interaction with single or multiple fingers. |
| Palm center tracking | ✔ | ✔ | Discern orientation of hand. |
| Left & Right Hand Tracking | ✔ | + Two of the same hands and hand labeling | Track two of the same hand for multiplayer gaming |
| Tracking/Occlusion Data | -- | Occlusion prediction and back-of-hand recognition | Improved performance when users' hands are partially occluded |
| Finger label assignment | -- | Thumb, pointer, middle, ring, pinky | More granular GUI control (don't mistake a pinky for thumbs up) |
| 3D hand skeleton | -- | 22 joints<br>User adaptation<br>Output orientation<br>Hand detection<br>Normalized Skeleton | Account for different hand sizes with the same code, contextually choose which hand to track |
| Accelerated Performance | -- | DirectX* support | Increases performance and extends battery life by utilizing GPU for interpretation of depth data. |

** All products, software, computer systems, dates and figures specified are preliminary based on current expectations, and are subject to change without notice.
*Any code names featured are used internally within Intel to identify products that are in development and not yet publicly announced for release. Customers, licensees and other third parties are not authorized by Intel to use code names in advertising, promotion or marketing of any product or services and any such use of Intel's internal code names is at the sole risk of the user.

# Facial Analysis Improvements

2013:





2014:

- Greater Accuracy
- Identify More Landmarks
- Pose Detection

** All products, software, computer systems, dates and figures specified are preliminary based on current expectations, and are subject to change without notice.
*Any code names featured are used internally within Intel to identify products that are in development and not yet publicly announced for release. Customers, licensees and other third parties are not authorized by Intel to use code names in advertising, promotion or marketing of any product or services and any such use of Intel's internal code names is at the sole risk of the user.

# Facial Analysis Improvements

| Feature | SDK 2013 | SDK 2014 | 2014 Developer Benefit |
|---|---|---|---|
| Face Detection/Tracking | ✔ Frontal, 2D | + increased accuracy<br>+ depth<br>+ maintains tracking with much greater head movement<br>+ recognizes more faces (final number tbd) | • Works in wider lighting conditions<br>• Enhanced multiple user scenarios<br>• More precise head-oriented command-and-control<br>• More security features and greater accuracy + fewer false detections |
| Landmark Detection | ✔ 7 points | + depth<br>+ 78 points | • Supports avatar creation and facial animation<br>• Recognize more facial attributes |
| Optimization | ✔ Intel® IPP (CPU) | ✔ | Higher performance, lower power |
| Head Orientation | -- | + roll, yaw, pitch | • Use for basic gaze-tracking<br>• Head-coupled perspective with 3D orientation |

** All products, software, computer systems, dates and figures specified are preliminary based on current expectations, and are subject to change without notice.
*Any code names featured are used internally within Intel to identify products that are in development and not yet publicly announced for release. Customers, licensees and other third parties are not authorized by Intel to use code names in advertising, promotion or marketing of any product or services and any such use of Intel's internal code names is at the sole risk of the user.

# Augmented Reality

AR enhanced with depth

## 3D Object Tracking w/ depth data

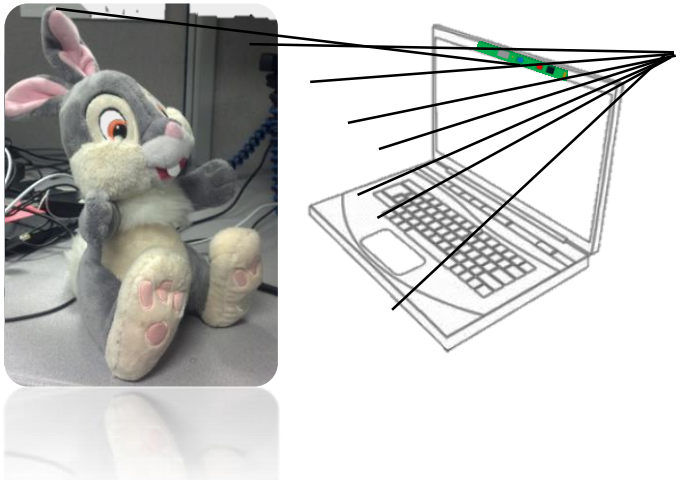
## Map 3D environments

Track 3D objects more accurately using depth data.
Map 3D environments to realistically augment the real world with computer generated graphics.

** All products, software, computer systems, dates and figures specified are preliminary based on current expectations, and are subject to change without notice.
*Any code names featured are used internally within Intel to identify products that are in development and not yet publicly announced for release. Customers, licensees and other third parties are not authorized by Intel to use code names in advertising, promotion or marketing of any product or services and any such use of Intel's internal code names is at the sole risk of the user.

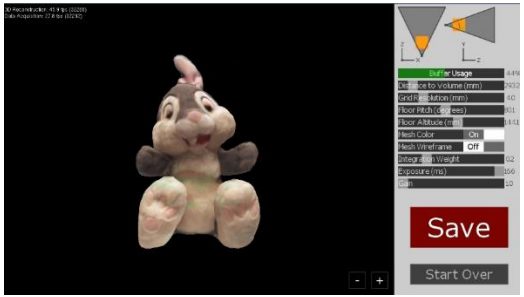
# 3D Scanning

1



2

3

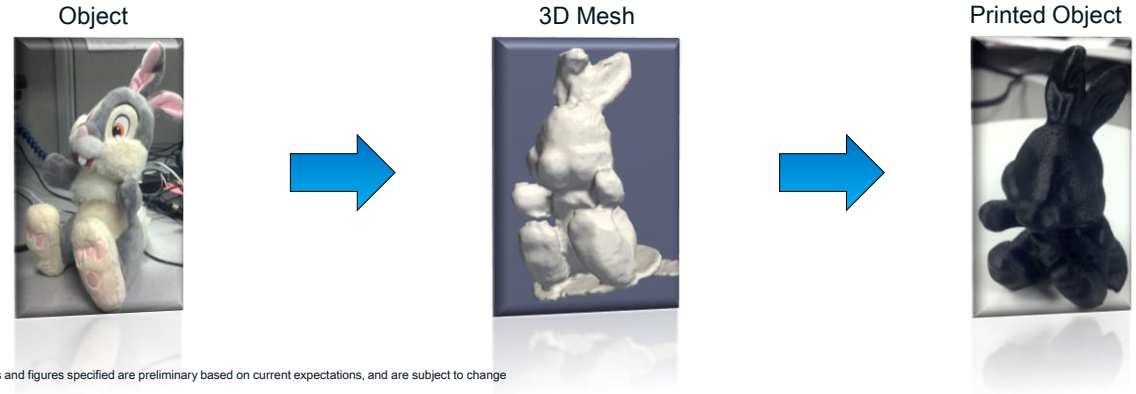



Capture objects and faces in 3D
Share and print 3D scans

** All products, software, computer systems, dates and figures specified are preliminary based on current expectations, and are subject to change without notice.
*Any code names featured are used internally within Intel to identify products that are in development and not yet publicly announced for release. Customers, licensees and other third parties are not authorized by Intel to use code names in advertising, promotion or marketing of any product or services and any such use of Intel's internal code names is at the sole risk of the user.

# 3D Scanning

| Feature | 2013 | 2014 | 2014 Developer Benefit |
|---|---|---|---|
| Object Scanning | N/A | Scan objects 18in and smaller, create object mesh with texture | Turn real-world objects into 3D models to be printed, uploaded to online marketplaces, or used in-application |
| Facial Scanning | N/A | Scan faces, create high-polygon facial mesh with texture | Use for analysis, avatar creation, identification, biometrics |



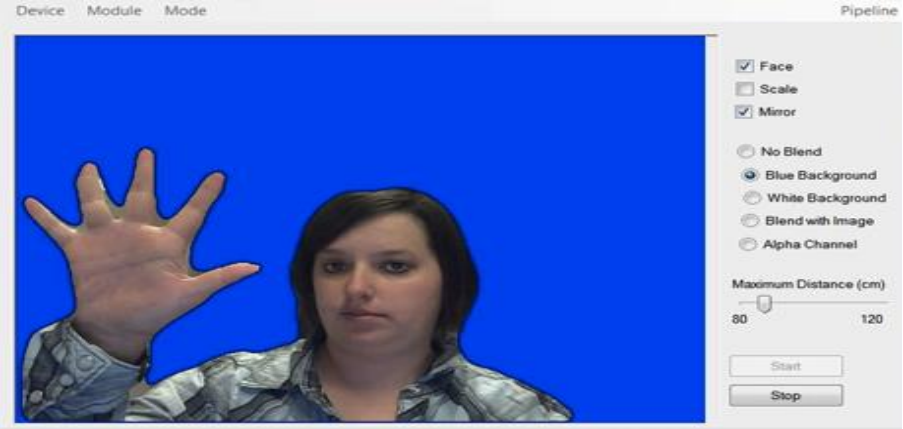

** All products, software, computer systems, dates and figures specified are preliminary based on current expectations, and are subject to change without notice.
*Any code names featured are used internally within Intel to identify products that are in development and not yet publicly announced for release. Customers, licensees and other third parties are not authorized by Intel to use code names in advertising, promotion or marketing of any product or services and any such use of Intel's internal code names is at the sole risk of the user.

# Background Subtraction

| Feature | 2013 | 2014 | 2014 Developer Benefit |
|---|---|---|---|
| Isolate object of interest | ✔ | ✔ | Exclude irrelevant information from your applications. For immersive collaboration, remove the background. For gesture interfaces, highlight the hands when in range |
| Smoothing | ✔ | ✔ | Smoother output, especially for difficult areas like hair and in-between fingers |

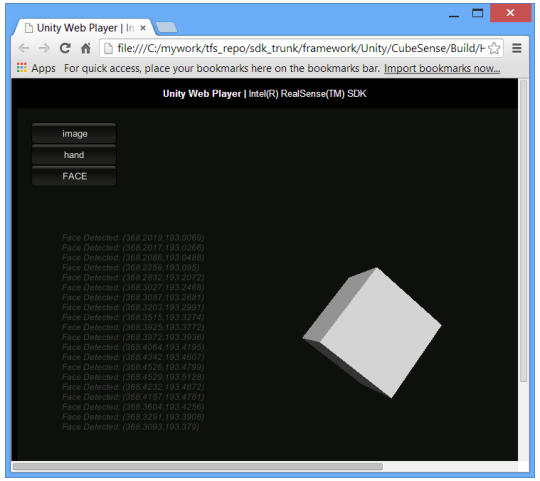
Remove the Background for:

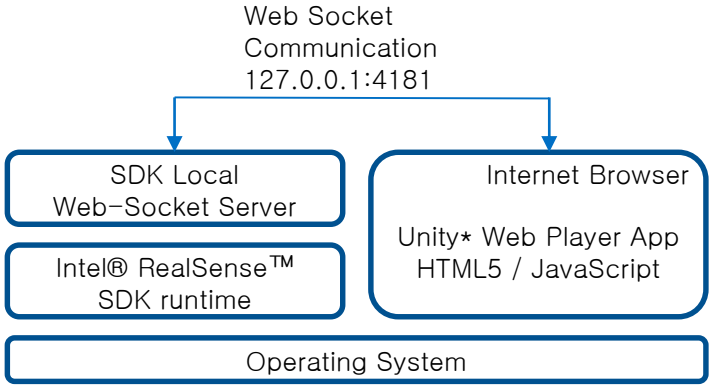
- Immersive collaboration
- Content creation via green screen




** All products, software, computer systems, dates and figures specified are preliminary based on current expectations, and are subject to change without notice.
*Any code names featured are used internally within Intel to identify products that are in development and not yet publicly announced for release. Customers, licensees and other third parties are not authorized by Intel to use code names in advertising, promotion or marketing of any product or services and any such use of Intel's internal code names is at the sole risk of the user.

# Web / HTML5




Develop based on PC/standalone, trigger gesture and face events
Switch to the webplayer platform for deployment

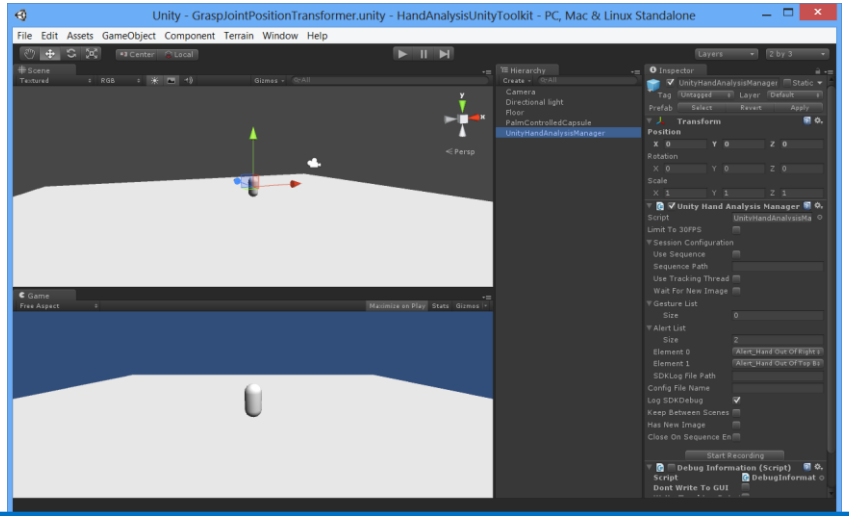
Gestures and Facial Analysis
Unity* Web Player games
HTML5 / JavaScript (interactive websites)

** All products, software, computer systems, dates and figures specified are preliminary based on current expectations, and are subject to change without notice.
*Any code names featured are used internally within Intel to identify products that are in development and not yet publicly announced for release. Customers, licensees and other third parties are not authorized by Intel to use code names in advertising, promotion or marketing of any product or services and any such use of Intel's internal code names is at the sole risk of the user.

# Unity Productivity Extension

A set of scripts that provides configurable actions/rules based on the senses provided by the SDK.

- Programming: Associate your game objects with the action scripts.
- Programming: Create rules. Associate your game objects with rules.




Drag and drop programming

Write less code!

** All products, software, computer systems, dates and figures specified are preliminary based on current expectations, and are subject to change without notice.
*Any code names featured are used internally within Intel to identify products that are in development and not yet publicly announced for release. Customers, licensees and other third parties are not authorized by Intel to use code names in advertising, promotion or marketing of any product or services and any such use of Intel's internal code names is at the sole risk of the user.

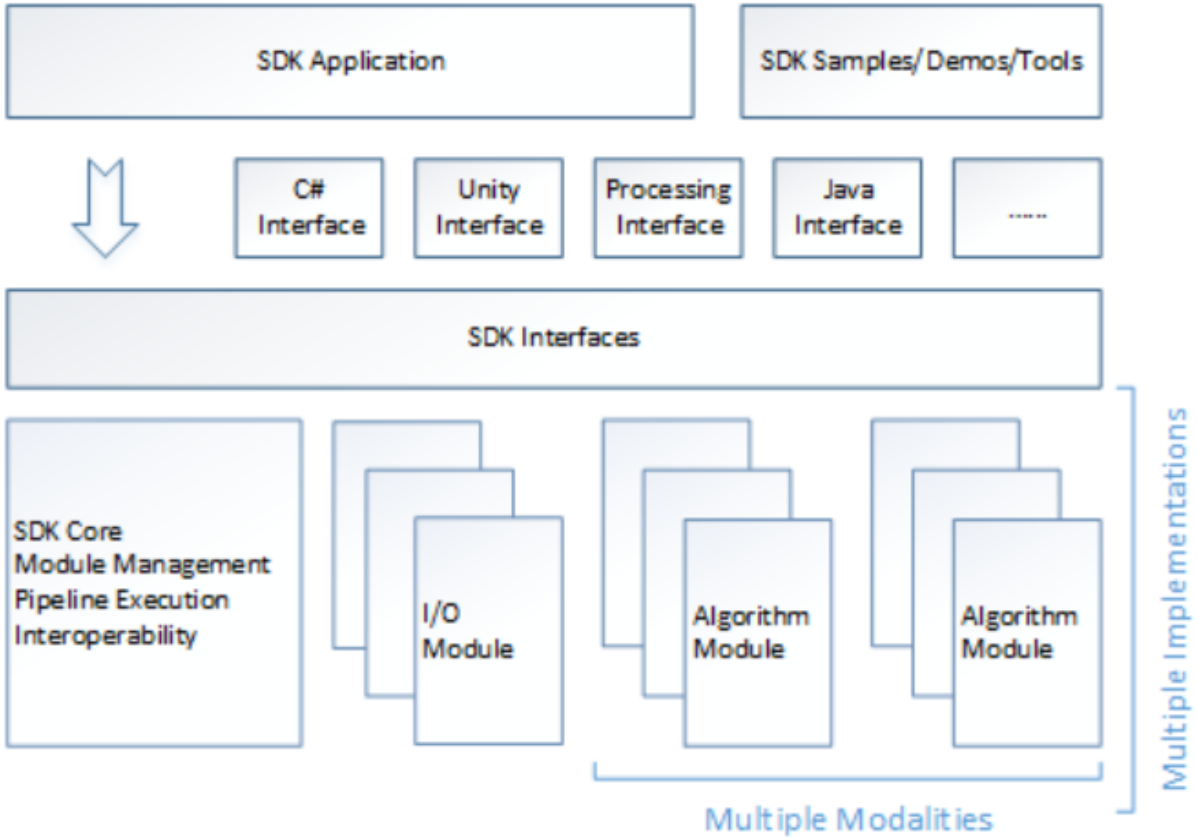
# Intel RealSense™ SDK 2014 Features

| | |
|---|---|
| Modalities | Hand/Finger Tracking<br>Gesture Recognition<br>Facial Analysis<br>3D Scanning<br>Background Removal<br>Augmented Reality<br>Speech Recognition |
| Processors | Intel® Core™ Processors (4th and 5th Generation) |
| Operating Systems | Microsoft Windows* 8.1 Desktop (32 and 64-bit) |
| Programming Languages | C++, C#, Java (for face, voice, and gesture) |
| Microsoft Visual Studio* | VS 2008, VS 2010, VS2012, VS2013 |
| Supported Camera | Intel® RealSense™ 3D Camera (Bell Cliffs) |
| Development Tools | ▪ Processing* Open Source Programming Language and Environment<br>▪ Unity* Game Development Environment<br>▪ Havok* Vision Engine<br>▪ Intel® SDK for OpenCL* Applications 2013<br>▪ Support Forums<br>▪ Raw-Depth Access<br>▪ Tutorials<br>▪ Gesture Authoring Tool<br>▪ RealSense™ Productivity Extensions |

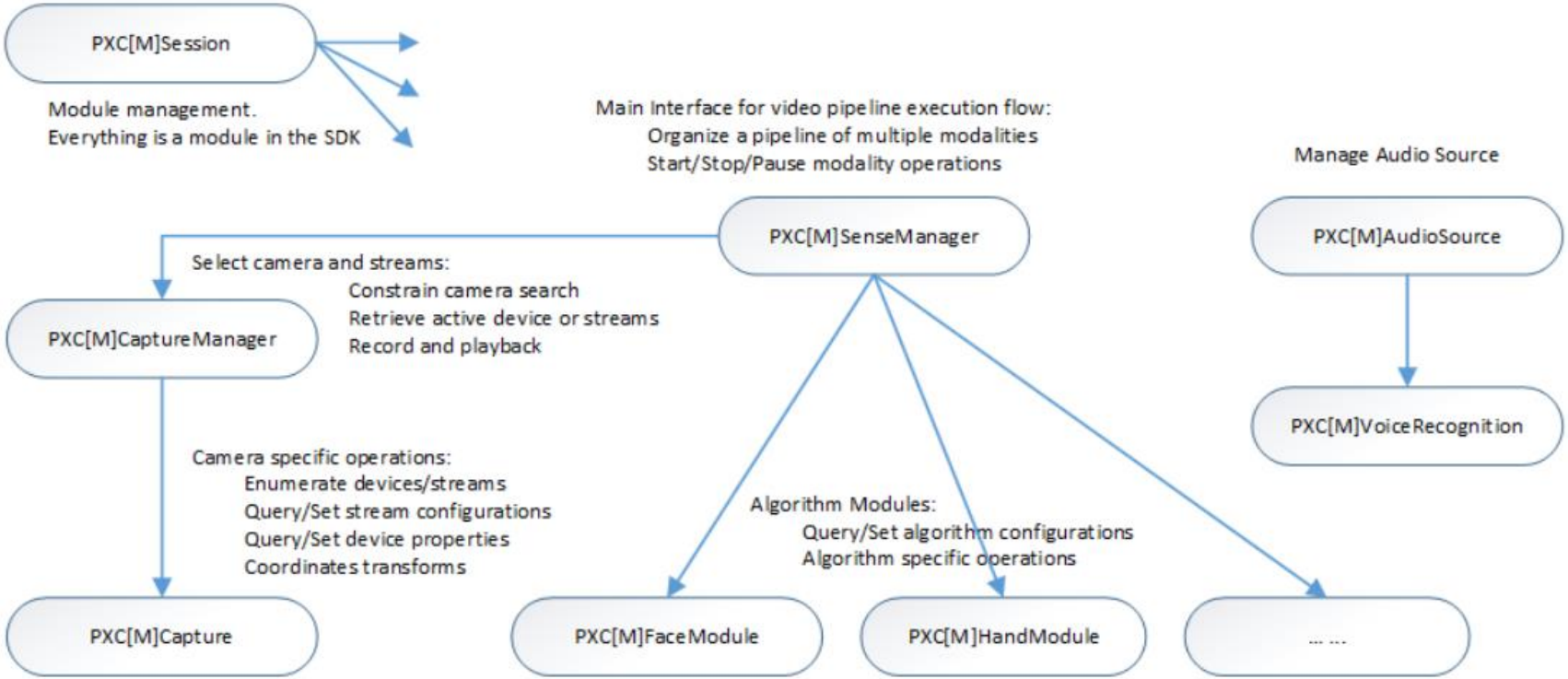
** All products, software, computer systems, dates and figures specified are preliminary based on current expectations, and are subject to change without notice.
*Any code names featured are used internally within Intel to identify products that are in development and not yet publicly announced for release. Customers, licensees and other third parties are not authorized by Intel to use code names in advertising, promotion or marketing of any product or services and any such use of Intel's internal code names is at the sole risk of the user.

# SDK Architecture

# Architecture Of 2014 RealSense SDK



# SDK Interface Hierarchy



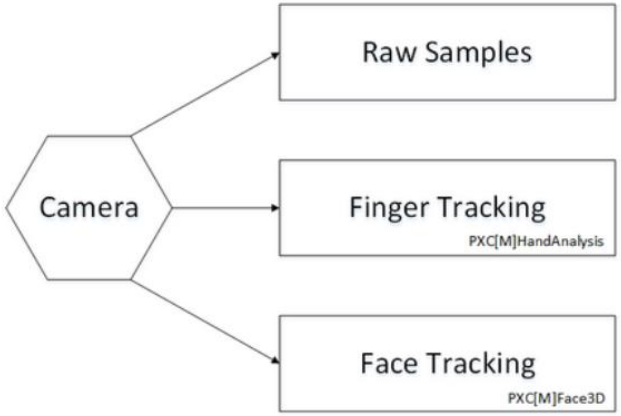
# Session and Modules

- Session :
  - A context which holds I/O and algorithm modules. Every SDK application must contain at least one session.

- Modules :
  - The SDK building blocks which implement certain SDK interfaces, there might be multiple implementations of the same SDK interface.

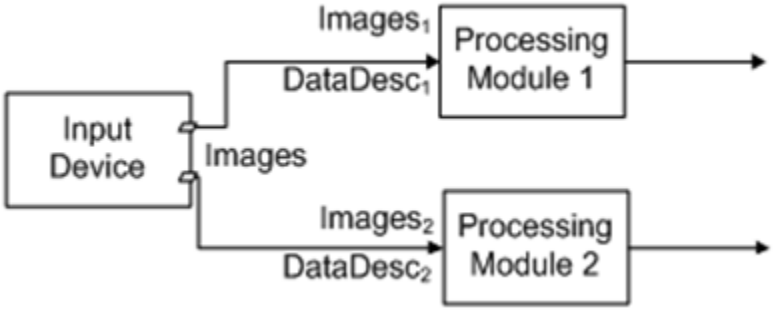
# SenseManager

- PXC[M]SenseManager :
  - The main interface to any **predefined** use cases.
    - One or more algorithm modules.
    - One single camera input device.
    - Serving input data to algorithm modules.



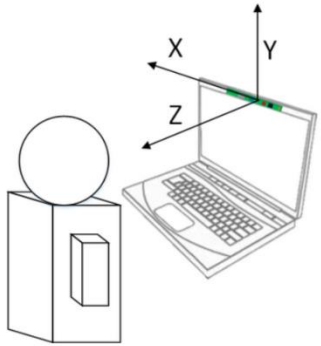
# CaptureManager

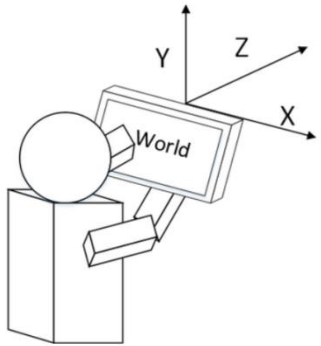
- Helper interface for creating complex pipelines.
- Consolidate the input needs of processing modules and locate the right I/O device.



# SDK's two coordinate systems

- **Image Coordinate** : refer to the **pixel (x,y)** in the depth/color image pictures, where x is in the range of 0~width-1, y is in the range of 0~height-1.
- **World Coordinate** : **3D coordinates (x,y,z)** relative to the camera. (0, 0, 0) is the original point of camera depth point. X axis points to left, Y axis points to up and Z axis points to the user when the user face to the camera. The [illegible] are in meter [illegible]tly specified.



# Programming using SenseManager

# Procedure of initiating pipeline

- Create a PXCSenseManager Instance.
- Enable the module.
- Control the execution flow and retrieve results.
  - Message loop.
  - Event callbacks.
- Releasing the instance no longer need.

# Control the execution flow.

## ➢ Message loop

```
// Create a PXCSenseManager instance
PXCSenseManager *sm=PXCSenseManager::CreateInstance();

// Select the color stream
sm->EnableStream(PXCCapture::STREAM_TYPE_COLOR,640,480);

// Initialize and Stream Samples
sm->Init();
for (;;) {
   // This function blocks until a color sample is ready
   if (sm->AcquireFrame(true)<PXC_STATUS_NO_ERROR) break;

   // retrieve the sample
   PXCCapture::Sample *sample=sm->QuerySample();

   // work on the image sample->color
   ...

   // go fetching the next sample
   sm->ReleaseFrame();
}

// Close down
sm->Release();
```

## ➢ Event callbacks

```
class MyHandler: public PXCSenseManager::Handler {
public:

   virtual pxcStatus PXCAPI OnNewSample(pxcUID, PXCCapture::Sample *sample) {
       // work on sample->color
       ...
       // return NO ERROR to continue, or any ERROR to exit the loop
       return PXC_STATUS_NO_ERROR;
   }
};

void MainRoutine(void) {
   // Create a SenseManager instance
   PXCSenseManager *sm=PXCSenseManager::CreateInstance();

   // Enable depth stream at 320x240x60fps
   sm->EnableStream(PXCCapture::STREAM_TYPE_DEPTH,320,240,60);

   // Initialize my event handler
   MyHandler handler;
   sm->Init(&handler)

   // Stream depth samples
   sm->StreamFrames(true);

   // Clean up
   sm->Close();
   sm->Release();
}
```

# Capture color and depth stream

## ➢ Aligned

```
void CaptureAlignedColorDepthSamples(void) {
   // Create a SenseManager instance
   PXCSenseManager *sm=PXCSenseManager::CreateInstance();

   // Select the color and depth streams
   sm->EnableStream(PXCCapture::STREAM_TYPE_COLOR,640,480,30);
   sm->EnableStream(PXCCapture::STREAM_TYPE_DEPTH,320,240,30);

   // Initialize and Stream Samples
   sm->Init();
   for (;;) {
       // This function blocks until both samples are ready
       if (sm->AcquireFrame(true)<PXC_STATUS_NO_ERROR) break;

       // retrieve the color and depth samples aligned
       PXCCapture::Sample *sample=sm->QuerySample();

       // work on the samples sample->color and sample->depth
       ...

       // go fetching the next samples
       sm->ReleaseFrame();
   }

   // Close down
   sm->Release();
}
```

## ➢ Unaligned

```
void CaptureUnalignedColorDepthSamples(void) {
   // Create a SenseManager instance
   PXCSenseManager *sm=PXCSenseManager::CreateInstance();

   // Select the color and depth streams
   sm->EnableStream(PXCCapture::STREAM_TYPE_COLOR,640,480,30);
   sm->EnableStream(PXCCapture::STREAM_TYPE_DEPTH,320,240,30);

   // Initialize and Stream Samples
   sm->Init();
   for (;;) {
       // This function blocks until any sample is ready
       if (sm->AcquireFrame(false)<PXC_STATUS_NO_ERROR) break;

       // retrieve the color and depth samples unaligned
       PXCCapture::Sample *sample=sm->QuerySample();
       if (sample) {
           if (sample->color) {
               // work on the color sample
               ...
           }
           if (sample->depth) {
               // work on the depth sample
               ...
           }
       }

       // go fetching the next samples
       sm->ReleaseFrame();
   }

   // Close down
   sm->Release();
}
```

# Access Image(Audio) Data

- PXC[M]Image/Audio : interface to abstract the image/audio storage.
- Image Data Access – *AcquireAccess* : lock access to the image storage and retrieve the details in **ImageData**

  – *ReleaseAccess* : release the lock of accessing buffer.

```
// image is a PXCImage instance
PXCImage::ImageData data;
image->AcquireAccess(PXCImage::ACCESS_READ,PXCImage::COLOR_FORMAT_RGB32,&data);
... // color data in data.planes[0] in RGB32.
image->ReleaseAccess(&data);
```

# Create Image Instance

- Must use the *CreateImage* function from *PXC[M]Accelerator* interface to create PXC[M]Image

```
// session is a PXCSession instance

PXCImage::ImageInfo info;
memset(&info,0,sizeof(info));
info.format=PXCImage::COLOR_FORMAT_RGB32;
info.width=image_width;
info.height=image_height;

PXCImage::ImageData data;
memset(&data,0,sizeof(data));
data.format=PXCImage::COLOR_FORMAT_RGB32;
data.type=PXCImage::SURFACE_TYPE_SYSTEM_MEMORY;
data.planes[0]=image_buffer;
data.pitches[0]=ALIGN16(info.width*4)*info.height;

PXCSmartPtr<PXCAccelerator> accel;
session->QueryAccelerator(&accel);

PXCSmartPtr<PXCImage> image;
accel->CreateImage(&info,0,&data,&image);
```

Without Data Copy

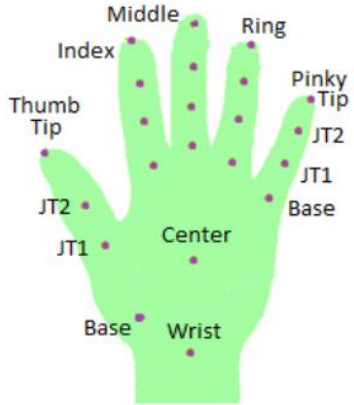
```
// session is a PXCSession instance

PXCImage::ImageInfo info;
memset(&info,0,sizeof(info));
info.format=PXCImage::COLOR_FORMAT_RGB32;
info.width=image_width;
info.height=image_height;

PXCSmartPtr<PXCAccelerator> accel;
session->QueryAccelerator(&accel);

PXCSmartPtr<PXCImage> image;
accel->CreateImage(&info,&image);

PXCImage::ImageData data;
image->AcquireAccess(PXCImage::ACCESS_WRITE,&data);
... // copy the imported image to data.planes[0]
image->ReleaseAccess(&data);
```

With Data Copy

# Hand Tracking

# Tracked information

- HandData : Hand's id, body side, finger data, joint data etc…
- FingerData : degree of foldeness of finger, radius of finger tip.
- JointData : four points of each finger, palm center, wrist.
- ExtremityData : special tracking such as left most point of the hand, nearest point of the hand to the camera etc…



# Hand Analysis Procedure

- Invoking *EnableHand* to active hand tracking in Sense Manager.
- Retrieving *PXC[M]HandModule* interface by calling *QueryHand*.
- Get detailed information of hand by utilizing *PXC[M]Hand* interface

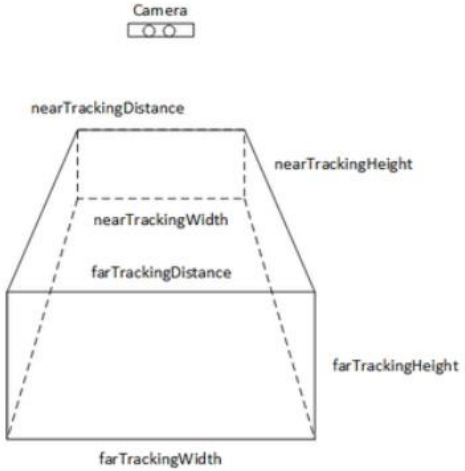
```
// Create an instance of the SenseManager
PXCSenseManager *sm=PXCSenseManager.CreateInstance();

// Enable hand tracking
sm->EnableHand();

// Get a hand instance (or inside the AcquireFrame/ReleaseFrame loop) for querying/configuring features
PXCHandModule *hand=sm->QueryHand();
...

// Initialize the pipeline
sm->Init();

// Stream data
while (sm->AcquireFrame(true)>=PXC_STATUS_NO_ERROR) {
  // retrieve hand tracking results if ready
  PXCHandModule *hand2=sm->QueryHand();
  if (hand2) {
      ...
  }

  // resume next frame processing
  sm->ReleaseFrame();
}

// Clean up
sm->Release();
```

# Configuration and boundary

- Get *PXCHandConfiguration* instance.
- Use *Update* to retrieve recent module configuration parameters.
- Use *ApplyChange* to push the changes back to hand module

```
// hand is a PXCHandModule instance
PXCHandConfiguration *cfg=hand->CreateActiveConfiguration();

// Set to enable all alerts
cfg->EnableAllAlerts();

// Apply changes
cfg->ApplyChanges();

// Clean up
cfg->Release();
```

- Use *Set/GetTrackingBounds* to change the bounds of tracking zone



# Data Retrieval

- Get *PXCHandData* instance.
- Use *Update* to retrieve recent processed data.

```
// hand is a PXCHandModule instance
PXCHandData *hdata=hand->CreateOutput();

// Get the number of tracked hands
pxcI32 nhands=hdata->QueryNumberOfHands();

// Clean up
hdata->Release();
```

- Get hand Id by *QueryHandId* to distinguish different hand.

```
// hdata is a PXCHandData instance
// retrieve the hand identifier
pxcUID handId;
hdata.QueryHandId(PXCHandAnalysis::ACCESS_ORDER_NEAR_TO_FAR, 0, handId);

// retrieve the hand data
PXCHandData::IHand *ihand=0;
hdata->QueryHandDataById(handId,&ihand);
```

# Default supporting gestures

| Illustration | Gesture Name |
|---|---|
| | spreadfingers |
| | fist |
| | tap |
| | thumb_down |

| Illustration | Gesture Name |
|---|---|
| | thumb_up |
| | two_fingers_pinch_open |
| | v_sign |

# Recognize gesture data

```
// hcfg is a PXCHandConfiguration instance

class MyHandler:public PXCHandConfiguration::GestureHandler {
public:

   virtual void PXCAPI OnFiredGesture(const PXCHandData::GestureData &data) {
      ...
   }
};

// Subscribe to the gesture notification event
MyHandler handler;
hcfg->SubscribeGesture(&handler);
hcfg->ApplyChanges();

...

// Unsubscribe
hcfg->UnsubscribeGesture(&handler);
hcfg->ApplyChanges();
```

# Handle Alert Notification

```
// hcfg is a PXCHandConfiguration instance

class MyHandler:public PXCHandConfiguration::AlertHandler {
public:

   virtual void PXCAPI OnFiredAlert(const PXCHandData::AlertData &data) {
      ...
   }
};

// Subscribe to the alert notification event
MyHandler handler;
hcfg->SubscribeAlert(&handler);
hcfg->ApplyChanges();

...

// Unsubscribe
hcfg->UnsubscribeAlert(&handler);
hcfg->ApplyChanges();
```

# Face Tracking

# Hand Analysis Procedure

- Invoking *EnableFace* to active hand tracking in Sense Manager.
- Retrieving *PXCFaceModule* interface by calling *QueryFace*.
- Get detailed information of face by utilizing *PXCFace* interface

```
// Create the SenseManager instance
PXCSenseManager *sm=PXCSenseManager.CreateInstance();

// Enable face tracking
sm->EnableFace();

// Get a face instance (or inside the AcquireFrame/ReleaseFrame loop) for configuration.
PXCFaceModule *face=sm->QueryFace();
...

// Initialize the pipeline
sm->Init();

// Stream data
while (sm->AcquireFrame(true)>=PXC_STATUS_NO_ERROR) {
  // retrieve the face tracking results
  PXCFaceModule *face2=sm->QueryFace();
  if (face2) {
      ...
  }

  // Resume next frame processing
  sm->ReleaseFrame();
}

// Clean up
sm->Release();
```

# Configuration and data retrieval

## ➢ Configuration

```
// face is a PXCFaceModule instance
PXCFaceConfiguration *cfg=face->CreateActiveConfiguration();

// Set to enable all alerts
cfg->EnableAllAlerts();

// Apply changes
cfg->ApplyChanges();

// Clean up
cfg->Release();
```

## ➢ Data retrieval

```
// face is a PXCFaceModule instance
PXCFaceData *fdata=face->CreateOutput();

// Get the number of tracked faces
pxcI32 nfaces=fdata->QueryNumberOfDetectedFaces();

// Clean up
fdata->Release();
```

# Location and expression

## ➢ Location

```
// fdata is a PXCFaceData instance
pxcI32 nfaces=fdata->QueryNumberOfDetectedFaces();
for (pxcI32 i=0;i<nfaces;i++) {
   // Retrieve the face location data instance
   PXCFaceData::Face *face=fdata->QueryFaceByIndex(i);
   PXCFaceData::DetectionData *ddata=fdata->QueryDetection();

   // work on DetectionData
   PXCRectI32 rect;
   ddata->QueryBoundingRect(&rect);
   ...
}
```

## ➢ Expression

1. Configuration

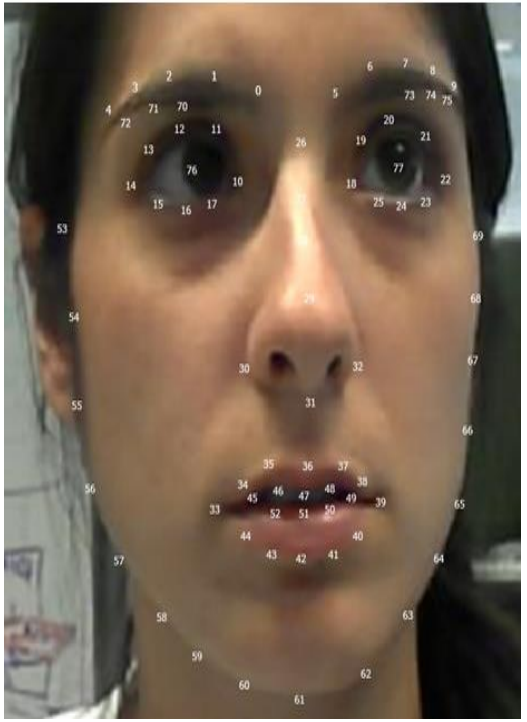
```
// cfg is a PXCFaceConfiguration instance
PXCFaceConfiguration::ExpressionsConfiguration *ecfg=cfg->QueryExpressions();
ecfg->Enable();
ecfg->EnableAllExpressions();
```

2. Retrieving

```
// fdata is a PXCFaceData instance
pxcI32 nfaces=fdata->QueryNumberOfDetectedFaces();
for (pxcI32 i=0;i<nfaces;i++) {
   // Retrieve the expression data instance
   PXCFaceData::Face *face=fdata->QueryFaceByIndex(i);
   PXCFaceData::ExpressionsData *edata= face->QueryExpressions();

   // retrieve the expression information
   PXCFaceData::ExpressionsData::FaceExpressionResult score;
   edata->QueryExpression(PXCFaceData::ExpressionsData::EXPRESSION_SMILE, &score);

   // action based on detected expression
   ...
}
```

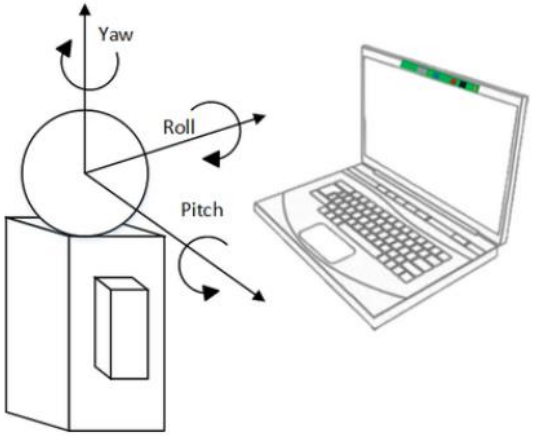
# Landmark

```
// fdata is a PXCFaceData instance
pxcI32 nfaces=fdata->QueryNumberOfDetectedFaces();
for (pxcI32 i=0;i<nfaces;i++) {
   // Retrieve the face landmark data instance
   PXCFaceData::Face *face=fdata->QueryFaceByIndex(i);
   PXCFaceData::LandmarksData *ldata=fdata->QueryLandmarks();

   // allocate the array big enough to hold the landmark points.
   pxcI32 npoints=ldata->QueryNumPoints();
   PXCFaceData::LandmarkPoint *points=new PXCFaceData::LandmarkPoint[npoints];

   // get the landmark data
   ldata->QueryPoints(points, &npoints);

   // Clean up
   delete[] points;
}
```



# Pose

```
// fdata is a PXCFaceData instance
pxcI32 nfaces=fdata->QueryNumberOfDetectedFaces();
for (pxcI32 i=0;i<nfaces;i++) {
   // Retrieve the face landmark data instance
   PXCFaceData::Face *face=fdata->QueryFaceByIndex(i);
   PXCFaceData::PoseData *pdata= face->QueryPose();

   // retrieve the pose information
   PXCFaceData::PoseEulerAngles angles;
   pdata->QueryPoseAngles(&angles);
}
```



# Recognition

## ➢ Configuration

```
// cfg is a PXCFaceConfiguration instance
PXCFaceConfiguration::RecognitionConfiguration *rcfg=cfg->QueryRecognition();

// Enable face recognition
rcfg->Enable();

// Create a recognition database
PXCFaceConfiguration::RecognitionConfiguration::RecognitionStorageDesc desc={};
desc.maxUsers=10;
rcfg->CreateStorage(L"MyDB", &desc);
rcfg->UseStorage(L"MyDB");

// Set the registeration mode
rcfg->SetRegistrationMode(PXCFaceConfiguration::RecognitionConfiguration::REGISTRATION_MODE_CONTINUOUS);

// Make it effective
cfg->ApplyChanges();
```

## ➢ Recognition

```
// fdata is a PXCFaceData instance
pxcI32 nfaces=fdata->QueryNumberOfDetectedFaces();
for (pxcI32 i=0;i<nfaces;i++) {
   // Retrieve the recognition data instance
   PXCFaceData::Face *face=fdata->QueryFaceByIndex(i);
   PXCFaceData::RecognitionData *rdata= face->QueryRecognition();

   // recognize the current face?
   pxcI32 uid=rdata->QueryUserID();
   if (uid>=0) {
       // do something with the recognized user.
       ....
   }
}
```

# Demo

Face detecting/tracking, gesture viewer

# User Experiences

Head, Gesture Tracking and Voice Recognition

# Front Facing Depth Camera Usages & Experiences

## Interact Naturally

Navigate content and perform simple actions with robust and comfortable gestures or voice

## Immersive Collaboration

Mimic real-life interactions in immersive virtual spaces and easily share or create digital content

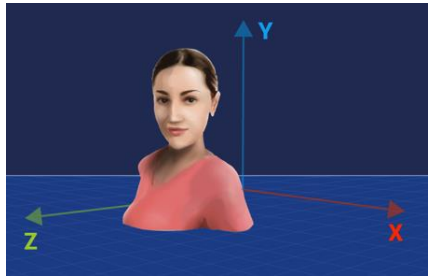
## Gaming & Play

Enhance gaming experience with natural language and intuitive gestures, AR and head tracking

## Learning & Edutainment

Blend physical and digital worlds for a more immersive and engaging learning & edutainment experience

## 3D Capture, Share & Recreate




Scan, share, edit, augment, or 3D print objects and people

# Intel® Realsense™ Camera Portfolio

## Front Facing

**BELL CLIFFS (IVCAM) (F200)**

Technology: Coded Light; VGA 60fps
Platform: HSW, BDW;
Schedule: Q4 '14
Range: Indoors; 0.2 – 1.2m
Depth Map: Realtime 60fps VGA Res
2D Camera: 1080p RGB
OS: Windows 8.1; Schedule:
SDK: Yes

Immersive Collaboration /Creation

Learning and Edutainment

Interact Naturally

Capture and Share

Gaming and Play

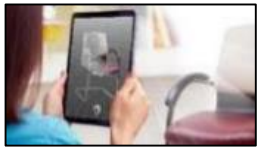
## Rear Facing

**Double Springs (DS4) (R200)**

Technology: Active Stereo; VGA 60fps
Platform: CHT/Android, BDW/Win
Schedule: 1H '15
Range: 0.5 – 3m (Indoors);
10+m Outdoors
OS: Android, Windows 8.1
Schedule: 1H''15
SDK: Yes

Immersive Gaming & Learning & Collaborate

Enhanced Video



Capture the World in 3D

Enhanced Photography

Enhanced Photography

**Davis Reef (Intel RealSense Snapshot (R100)**

Technology: 1fps post processed Stereo
Lead Platform: Moorefield
Schedule: Q4 '14
Range: 20m; Indoors/Outdoors
OS: Android, Windows - TBD
SDK: Yes

# 3D Capture, Share, and Create Usage Vision

**New Usage Opportunity**

Quickly & easily capture my world….

I want to 3D capture my physical world so I can share, edit, augment, and 3D print the things that matter to me

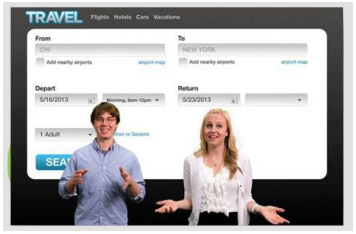
| | 2014 (Front) | 2015 (Front) | Future |
|---|---|---|---|
| 3D Object Capture | Capture a small object quickly and easily, and then 3D print, share to Facebook, or save the 3D object model | Capture a wider range of objects quickly and easily: small, medium, and large objects + areas/rooms | Establish 3D object capture usages as part of consumers' daily routines and lives, and as a viable alternative to 2D photography for specific usages |
| 3D Person Capture | Capture a face, head, or torso quickly and easily, and then 3D print, share to Facebook, or save the 3D person model | Capture a face, head, torso, or full body quickly and easily with minimal post-capture editing required | Establish 3D person capture usages as part of consumers' daily routines and lives |

# Immersive Collaboration and Creation Usage Vision

**New Usage Opportunity**

I want to interact with people at a distance as if we are together….

I want to mimic real-life interactions in immersive virtual space & facilitate easy sharing and creation of content with friends, family and colleagues



| | 2014 (Front) | 2015 (Front) | Future |
|---|---|---|---|
| HD VC | Smooth, clear HD (up to 1080p) visual quality in a larger viewing area with more detail (vs. comp and non-optimized) | Person remains in "HD" even under very slow networks. Simple ability to replace the background for more privacy. Share and interact with content | HD-TV like visual quality with ability to share any content on any network connection. |
| 3D-Enhanced Collaboration | Group video chat + content sharing all in the same virtual space, for instance on top of an online shopping web site. | More realistic user extraction, immersive casual gaming in the common virtual space, meeting transcription and note-taking, user augmentation | Establish immersive collaboration usages as part of consumers' daily routines and lives, and as a viable alternative to 2D videoconferencing |
| Content Creation | Author, record and publish a presentation with content and presenter occupying the same virtual space. | More realistic user extraction, natural interaction with your presentation content (gestures, face tracking, physics based interaction), custom content themes | Establish immersive content creation as part of consumers' daily routines and lives, encouraging greater knowledge sharing |

# Interact Naturally Usage Vision

**New Usage Opportunity**

Give me easier ways to interact with my devices….

I want to navigate my content and perform simple actions with robust and comfortable gesture interfaces, particularly when I "lean back" or otherwise cannot touch my device

| | 2014 (Front) | 2015 (Front) | Future |
|---|---|---|---|
| "Touchless Touch" | Navigate Windows 8 with gestures for a few simple, common actions on clamshell and AIO form factors. | Support more actions. Support additional form factors (e.g. tablet) and OS's. | Support more actions and features as suggested through additional UX research into user needs. |
| Gesture Applications (ISV) | Navigate maps with gestures. | More apps TBD | More apps TBD |

# Gaming and Play Consumer Usage Vision

**New Usage Opportunity**

Creating new ways to Game and Play…

want to blend my physical and digital worlds to give me more immersive and engaging forms of entertainment

| | 2014 (Front) | 2015 | Future |
|---|---|---|---|
| Gaming & Play | Enhanced gaming and play through inclusion of gestures, hand skeleton tracking, segmentation, facial tracking, object tracking, augmented reality and voice enabled through engagements with ISVs in the ecosystem | Get a natural lifelike interaction experience using scanned real life objects<br>Bring your games and favorite characters into the real world<br>Expand the ISV and ecosystem partners to deliver additional compelling gaming and play titles | Explore additional modalities and immersive technologies to further improve the gaming and play space |

# Learning and Edutainment Usage Vision

**New Usage Opportunity**

Creating new ways to Learn….

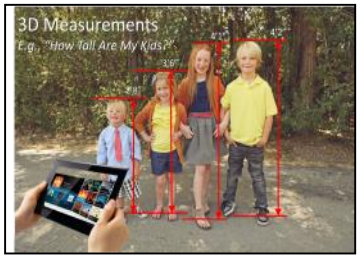
I want to be able to blend my digital and physical worlds to give me more immersive and engaging forms of learning and edutainment

| | 2014 (Front) | 2015 | Future |
|---|---|---|---|
| Learning & Edutainment | Enhanced learning and edutainment through inclusion of gestures, hand skeleton tracking, segmentation, facial tracking, object tracking, augmented reality and voice, enabled through engagements with ISVs in the ecosystem. | Build and create virtual projects<br>Show me how" to do it – assistance with everyday tasks and challenges<br>Expand ISV and ecosystem partners to deliver additional compelling learning and edutainment titles. | Explore additional modalities and immersive technologies to further improve learning and edutainment space. |

# Enhanced Photography and Videography

**New Usage Opportunity**

I take pictures as before, now I want the pictures to do more on my device, with my social friends and my business associates



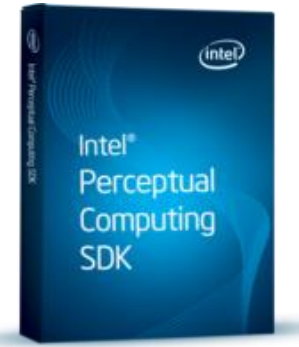
| | 2015 (Front) | Future |
|---|---|---|
| Enhanced Photography and Videography | Take a picture or video just like before – but now with depth for<br>• Measurements<br>• Selective filters (focus/color/light)<br>• Motion effects<br>• Social sharing | Cloning,<br>Better object removal and addition,<br>Better pre-shot focus,<br>Gaze correction,<br>New sizing edits<br><br>More features and applications working with ISVs in the ecosystem |

# Enhanced Photography and Videography

# Intel® RealSense™ Pillars

## Intel® RealSense™ 3D Camera

- Showcased several designs from top OEMs at CES and Computex
- Wide range of form factors (Ultrabook™, All-in-One PC, Notebook, Ultrabook™ 2 in 1)




## Intel® RealSense™ SDK 2014

- Hand/ finger tracking, background removal, AR, hand skeleton, 3D scanning
- http://www.intel.com/realsense for free download

## Intel® RealSense™ Ecosystem

- Targeting top marquee apps, indie and scale developers
- Hackathons, AE Sessions, Challenges and Webinars
- $1 Million Intel® RealSense™ App Challenge, 2014

# Additional Resources

| Resource Type | Access |
| --- | --- |
| Intel® RealSense™ technology website | http://www.intel.com/RealSense |
| Intel® RealSense™ developer resources | http://www.intel.com/RealSense/sdk |
| 2014 Intel® RealSense™ $1m challenge | http://www.intel.com/RealSense/challenge |
| Intel® RealSense™ Youtube channel | http://www.youtube.com/IntelRealSense |
| Intel® RealSense™ Facebook page | http://www.facebook.com/IntelRealSense |
| Intel® RealSense™ Twitter Page | http://www.twitter.com/IntelRealSense |

# 2014 Intel® RealSense™ Challenge

# Ideation Phase

## Pioneer

7/28/14 – 10/1/14

Phase 1 of the Intel® RealSense™ App Challenge Pioneer Track is open to developers from around the world. The 1000 Top Scoring Ideas will be invited into the Pioneer Track Development Phase.

# Development Phase

## Pioneer

11/14/14 – 01/20/15

1000 participants will be invited to turn their Ideas into working Demos. Participants will be loaned the Intel® 3D Camera as part of the development process.

# Prizes

GRAND PRIZE (1)
$25,000

One overall winner chosen from the first place winners of each category will win an additional $25,000 cash prize.

FIRST PLACE (5)
$25,000

The top scoring demo in each category will win a $25,000 cash prize

SECOND PLACE (10)
$10,000

Two demos from each of the 5 categories will receive a $10,000 cash prize

EARLY SUBMISSION (50)
$1,000

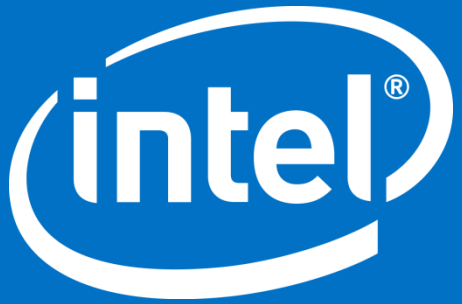
The top scoring demos, submitted prior to the Early submission deadline, across all 5 categories will each receive a cash prize of $1,000

HASWELL NUC (250)

The top 250 scoring demos from Phase 1, across all 5 categories, will receive a Haswell NUC device valued at nearly $600.

intel

Thank You




Intel Confidential — Do Not Forward