RELEASE REVISION HISTORY

Rev	Description of Change	Author	Date
1.0	Initial Release	Derek Zhou	06/17/2020
1.1	Update New API: isIROpen isTCMOpen getEstimatedEnvTemp	Derek Zhou	09/29/2020

SDK Guide

The IR camera kit contains IR camera, RGB camera and control board (TCM) which controls rotation of motor and proximity sensors.

The SDK consists of API of IR camera and TCM. The source code of demo also integrates usage of RGB camera (3rd-party opencv for windows and libuvc for android) and face tracking algorithm (3rd-party algorithm for both windows and android). There are three specific views, test/expert/smart views in demo, showing different usage scenarios. But we encourage ISVs using their own RGB camera APIs and face tracking APIs for better recognition and performance.

We're trying to keep the API consistent across different platforms.

0. Convention:

Point (0,0) will at top-left corner unless specified.

1. Life cycle:

1.1 Construction:

Android:

public ThermalController(Context context);

Parameter	Description
context	Context of Application

Suggest creating ThermalController in onCreate method.

Windows:

N/A

1.2 Start:

Android:

public void onStart();

Suggest calling in onStart method.

Tips: If you are using libelouvc for RGB stream, please let ThermalController.onStart() behind libelouvc.onStart().

Windows:

public bool Start();

Suggest calling in Window_Initialized method.

1.3 Stop

Android:

public void onStop();

Suggest calling in onStop method.

Windows:

public void Stop();

Suggest calling in Window_Closed method.

1.4 Check working status

isIROpen and isTCMOpen methods check IR camera and TCM, respectively.

isWorking method checks whether IR camera and TCM are working properly.

Return true if both IR and TCM are working.

Return false if IR or TCM is failed.

Android:

public boolean isWorking ();

public boolean isIROpen ();

```
public boolean isTCMOpen ();
```

Windows:

```
public bool IsWorking ();
public bool IsIROpen ();
public bool IsTCMOpen ();
```

1.5 Get SDK version

Get version of SDK, equals 10000*major+100*minor+patch.
e.g. If version is v1.1.1, it will return 10101.

Android:

```
public int getSDKVersion ();
```

Windows:

```
public int GetSDKVersion ();
```

2. IR Camera:

2.1 Set unit mode:

Unit: metric mode (mm, °C) or US (inch, °F)
It should be called on app initialization.

Android:

public void setUnitMode (int unitMode);

Windows:

public void SetUnitMode (int unitMode);

Parameter	Description
unitMode	0=Metric 1=US

2.2 Set temperature boundary

Set low/high/fever temperature. It should be called on app initialization.

Android:

public void setTempBoundary(double low, double high, double fever);

Windows:

public void SetTempBoundary(double low, double high, double fever);

Parameter	Description
low	Low threshold
high	High threshold
fever	Fever temperature

2.3 Set temperature offset

Set temperature offset, which will add to last temperature result. It should be called on app initialization.

Android:

public void setTempOffset(double offset);

Windows:

public void SetTempOffset(double offset);

Parameter	Description
offset	Temperature to be added

2.4 Set IR emissivity

Set IR emissivity. Default value is 0.98 (for human body).

Android:

public void setEmissivity(double emis);

Windows:

public void SetEmissivity(double emis);

Parameter	Description
emis	[0,1]

2.5 Set IR Frame Callback

Set IR Frame callback, which will be called if new IR data is arrived. You can use data to draw IR image.

Android:

public void setNewIRFrameDelegate(INewIRFrame call);

public interface INewIRFrame {
void onIRFrame(Bitmap bmp);
}

Parameter	Description
bmp	Bitmap of IR Image

Windows:

public void SetNewIRFrameDelegate(INewIRFrame call);

public delegate void INewIRFrame(IntPtr data, int width, int height, int channel);

Parameter	Description
data	Pointer to data
width	Image width
height	Image height

Parameter	Description
channel	RGB Channel, should be 3

2.6 Read temperature at point

Read single-point temperature.

Android:

public TemperatureResult ReadTemperatureAtPoint(int x, int y);

Windows:

public void ReadTemperatureAtPoint(int x, int y, unsigned char* buffer);

Win-wrapper:

public TemperatureResult ReadTemperatureAtPoint(int x, int y);

Parameter	Description
x	[0,159] 0=Left
y	[0,119] 0=Top

Return: TemperatureResult

Point (single or position of max temperature)
Raw temperature
Estimated under-arm temperature
Unit (°C or °F)
Judgement (illegal, beyond range, fever, normal)

2.7 Read temperature in rectangle

Read rectangle temperature and max temperature point.

Android:

public TemperatureResult readTemperatureInRect(Rect rect);

Windows:

public void ReadTemperatureInRect(int x, int y, int width, int height, unsigned char* buffer);

Win-wrapper:

public TemperatureResult ReadTemperatureInRect(RectI rect);

Parameter	Description
RectI	X,Y,Width,Height (0,0) at top-left

Return: TemperatureResult

Point (single or position of max temperature)
Raw temperature
Estimated under-arm temperature
Unit (°C or °F)
Judgement (illegal, beyond range, fever, normal)

2.8 Get IR inner temperature

Get inner temperature of IR. You don't need to call this in normal circumstances.

Android:

```
public double getInnerIRCameraTemp();
```

Windows:

```
public double GetInnerIRCameraTemp();
```

2.9 Trigger IR calibration

Trigger IR calibration manually. You don't need to call this in normal circumstances.

Android:

```
public void trigger();
```

Windows:

```
public void Trigger();
```

2.10 Get estimated environment temperature

Get estimated environment temperature of IR. You don’t need to call this in normal circumstances.

Android:

public double getEstimatedEnvTemp ();

Windows:

N/A

3. TCM:

3.1 Set new distance callback

Set proximity callback, which will be called if new distance data is arrived. You can use data to estimate the distance between kit and object.

Android:

public void setNewDistanceDelegate(INewDistance call);

```
public interface INewDistance {
   void onDistance(float distance);
}
```

Parameter	Description
distance	Distance to object

Windows:

public void SetNewDistanceDelegate(INewDistance call);

```
public delegate void INewDistance(float distance)
```

Parameter	Description
distance	Distance to object

3.2 Motor rotation

Rotation of IR kits.

Android:

public void rotate(int val);

Windows:

public void Rotate(int val);

Parameter	Description
val	[-200,200] Rotate units.

3.3 Get TCM version

Android:

public string getTCMVersion ();

Windows:

public void GetTCMVersion(char *version);

Win-wrapper:

public string GetTCMVersion();

3.4 Set Firmware Data

Android:

public void setFirmwareData(byte[] firm);

Windows:

public void SetFirmwareData(unsigned char* firm, int length);

Win-wrapper:

public void setFirmwareData(byte[] firm);

Parameter	Description
firm	Raw firmware data

3.5 Set Firmware Update Callback

Android:

public void setUpdateCallback (IUpdateCallback call);

```
public interface IUpdateCallback {
    void onUpdate(boolean isSuc, int prog);
}
```

Parameter	Description
isSuc	Is success
prog	[0,100], progress

Windows:

public void SetUpdateCallback (IUpdateCallback call);

public delegate void IUpdateCallback(bool isSucc, int prog);

Parameter	Description
isSuc	Is success
prog	[0,100], progress

3.6 Start Firmware Update

Android:

public void startUpdate ();

Windows:

public void StartUpdate ();