**RealSenseAPI**

תיאור:

ישנם 3 דרכים לתפעל את המצלמה RealSense.

1). דרך פונקציות ישירות של המכשיר, דוגמא לדרך תפעול זה אפשר לראות בדוגמא של המצלמה בשם api\_how\_to.h

2). דרך ממשק PIPE , דוגמא לתפעול זה ניתן לימצוא גם כן בדוגמאות של המצלמה. (מהrepository של realsense2).

3). דרך ממשק זה אשר מתבסס על הממשק דרך PIPE ואמור להיות פשוט יותר ואינטואיטיבי יותר.

תיאור כללי של הממשק:

בתחילת עבודה יש לקנפג את המצלמה לFrameים הרצויים מכל סנסור פנימי של המצלמה. לאחר הקינפוג נותנים פקודה למצלמה לתחילת עבודה. כאשר רוצים לקבל frame ברגע כל שהוא קוראים לפונקציה captureFrame אשר היא שומרת את כל הפרמיימים שקונפגו קודם לכן בתוך set. בעזרת הפונקציות התונות שולפים את הFrame הרצוי ועושים עליו את העיבוד הרצוי.

\*שימו לב – אם לא מספיקים לתפל בFrame בפרק זמן של  התמונה תידרס ע"י הframe אחר.

\*מומלץ לקורא את הdocs של המצלמה מתוך הסיפריה של realsense2 להבנת כל המגבלות של המצלמה.

**פונקציות הממשק:**

**bool** connectCamera();

**תיאור הפונקציה:**

הממשק מניח שעובדים רק עם מצלמת realsense אחת, (למרות שאין בעיה לחבר כמה מצלמות לHOST עם שינויים קטנים בקוד)

בעזרת פונקציה זו הHOST מאתר את המצלמה המחוברת אליו ושולפת מהמצלמה את הסנסורים שלה (נכון לגבי realsense – 3 סנסורים :

1. StrereoModule – שבעזרתו מוצאים את תמונות העומק
2. RGBCamera – מצלמה רגילה
3. MotionModule – מוציא נתוני gyro וacceleration .

**פרמטרים:**

אין.

**ערך החזרה:**

הפונקציה מחזירה TRUE אם הצלחנו להתחבר למצלמה ולקרוא ממנה את שלושת הסנסורים שלה.

**bool** isConnect();

**תיאור הפונקציה:**

בפונקציה בודקת אם המצלמה מחוברת לHOST**.**

שימו לב\* - אין לקרוא לפונקציה בתוף לולאה אין סופית כי היא פותחת thread נפרד במהלך הבדיקה. (יתכן ישנה דרך לבדוק אם המצלמה מחוברת בצורה שונה)

**פרמטרים:**

אין

**ערך החזרה:**

מחזיר TRUE אם המצלמה מחוברת.

**void** resetCamera();

**תיאור הפונקציה:**

מבצעה RESET חומרתי למצלמה. מומלץ לחכות קצת אחרי ריסט כזה ולא ישר לקרוא לפונקציות של המצלמה כידי לתת זמן לחומרה לעלות.

**פרמטרים:**

**אין**

**ערך החזרה:**

**אין**

**void** setupColorImage(RealSense::ColorFrameFormat format,RealSense::ColorRessolution ressolution,RealSense::ColorCamFps fps);

**תיאור הפונקציה:**

קינפוג מצלמה לFrameים רצויים.

**פרמטרים:**

* פורמט רצוי (מרשימת הפורמטים בתוך הCLASS)

*YUYV* , /\*\*< 32-bit y0, u, y1, v data for every two pixels. Similar to YUV422 but packed in a different order - https://en.wikipedia.org/wiki/YUV \*/  
*RGB8* , /\*\*< 8-bit red, green and blue channels \*/  
*BGR8* , /\*\*< 8-bit blue, green, and red channels -- suitable for OpenCV \*/  
*RGBA8* , /\*\*< 8-bit red, green and blue channels + constant alpha channel equal to FF \*/  
*BGRA8* , /\*\*< 8-bit blue, green, and red channels + constant alpha channel equal to FF \*/  
*Y16* , /\*\*< 16-bit per-pixel grayscale image \*/

* רזולוציה רצויה
* FPS רצוי

\*שימו לב שישנם שילובים של רזולוציה וFPS שלא יעבדו ביחד (המגבלות רשומות בקוד).

**ערך החזרה:**

אין

**void** setupInfraredImage(RealSense::InfrarFrameFormat format, RealSense::InfrarRessolution ressolution, RealSense::InfrarCamFps fps  
 ,RealSense::InfrarCamera side);

**תיאור הפונקציה:**

מקנפג את המצלמות אינפרה אדום (חלק מסנסורי ה StereoModule ) .

ניתן להוציא מידע על כל אחת מ2 המצלמות.

**פרמטרים:**

* פורמט רצוי

*Y8*, /\*\*< 8-bit per-pixel grayscale image \*/  
*Y16* /\*\*< 16-bit per-pixel grayscale image \*/

* רזולוציה רצויה
* FPS רצוי
* בחירת אחת מ2 המצלמות

\*שימו לב שישנם שילובים של רזולוציה וFPS שלא יעבדו ביחד (המגבלות רשומות בקוד).

**ערך החזרה:**

**אין**

**void** setupDepthImage(RealSense::DepthRessolution ressolution, RealSense::DepthCamFps fps);

**תיאור הפונקציה:**

מקנפג את תמונת העומק. הפורמט קבוע – Z16

**פרמטרים:**

* רזולוציה רצויה
* FPS רצוי

**ערך החזרה:**

**אין**

**void** setupGyro();

**תיאור הפונקציה:**

קינפוג הGYRO

**פרמטרים:**

אין

**ערך החזרה:**

אין

**void** setupAccel();

**תיאור הפונקציה:**

קינפוג הaccelometer

**פרמטרים:**

אין

**ערך החזרה:**

אין

**void** startCamera();

**תיאור הפונקציה:**

אחרי כל הקינפוגים קוראים לפונקציה כידי לגרום למצלמה לעבוד.

**פרמטרים:**

אין

**ערך החזרה:**

אין

**void** captureFrame();

**תיאור הפונקציה:**

כאשר רוצים לקבל FRAME מהמצלמה קוראים לפונקציה זאת, עבור כל קינפוג שהגדרנו קודם לכן נקבל FRAME שונה אשר כולם יכנסו לSET.

**פרמטרים:**

אין

**ערך החזרה:**

אין

Camera::ColorImage getColorImage();

**תיאור הפונקציה:**

שליפת התמונת COLOR מהSET

**פרמטרים:**

אין

**ערך החזרה:**

Struct שמכיל את התמונה ופרמטרים שקשורים לFRAME

**struct** ColorImage  
{  
 uint64 frame\_num;  
 uint64 size;  
 int32 bytes\_per\_pixel;  
 int64 host\_ts\_ms;  
 real64 camera\_ts\_ms;  
 uint32 width;  
 uint32 height;  
 **const unsigned char** \*data;  
};

Camera::ColorImage getInfraredImage();

**תיאור הפונקציה:**

שליפת תמונת אינפה אדום (אם קונפגה כזאת קודם לכן) מהSET

**פרמטרים:**

**אין**

**ערך החזרה:**

Struct שמכיל את התמונה ופרמטרים שקשורים לFRAME

Camera::DepthImage getDepthImage();

**תיאור הפונקציה:**

שליפת תמונת עומק מהSET

**פרמטרים:**

אין

**ערך החזרה:**

Struct שמכיל את התמונה ופרמטרים שקשורים לFRAME

**struct** DepthImage  
{  
 uint64 frame\_num;  
 uint64 size;  
 int32 bytes\_per\_pixel;  
 int64 host\_ts\_ms;  
 real64 camera\_ts\_ms;  
 uint32 width;  
 uint32 height;  
 real32 depth\_scale;  
 **const unsigned char** \*data;  
};

**float** getDepthUnits();

**תיאור הפונקציה:**

בישביל לקבל בתמונת עומק יחידות של מטרים עבור כל פיקסל יש לכפול את תמונת העומק בערך זה.

**רמטרים:**

**אין**

**ערך החזרה:**

מספר בו יש לכפול את תמונת העומק לקבלת יחידות של מטרים.

Camera::Intrinsics getDepthCamIntrinsics();

**תיאור הפונקציה:**

**רמטרים:**

אין

**ערך החזרה:**

Camera::Intrinsics getColorCamIntrinsics();

**תיאור הפונקציה:**

**רמטרים:**

אין

**ערך החזרה:**

Camera::Intrinsics getIfraRedCamIntrinsics();

**תיאור הפונקציה:**

**רמטרים:**

אין

**ערך החזרה:**

Camera::MotionIntrinsics getMotionCamIntrinsics();

**תיאור הפונקציה:**

**רמטרים:**

אין

**ערך החזרה:**

Camera::Extrinsics getExtrinsics(RealSense::Stream from\_stream, RealSense::Stream to\_stream);

**תיאור הפונקציה:**

**רמטרים:**

אין

**ערך החזרה:**

Camera::EulerAngles getEulerAngels();

**תיאור הפונקציה:**

**רמטרים:**

אין

**ערך החזרה:**

Camera::AccelData getAccelData();

**תיאור הפונקציה:**

**רמטרים:**

אין

**ערך החזרה:**