OMRON CORPORATION
ELECTRONIC AND MECHANICAL
COMPONENTS COMPANY

OKAO™ Vision

Result Stabilization V1.1

Software Library

Software Specification Document

■ **Trademarks**

"OKAO" and "OKAO Vision" are registered trademarks of OMRON Corporation in Japan.

■ **Revision History**

Date	Rev	Contents	Prepared by	Reviewed by	Approved by
2015/12/01	A	First release	Urabe	Manabe	Yamada

■ **Additional Notes**

You may not reproduce, duplicate or copy the contents of this software specification document without proper written authorization from OMRON Corporation. The contents of this software specification document may change without notice.

TABLE OF CONTENTS

1 Outline of Library

The Result Stabilization V1.1 Software Library (hereafter referred to as “This Library”) stabilizes the result of number of object person/people, age, gender and face recognition based on the result of Human Body Detection, Face Detection, Age/Gender Estimation and Face Recognition in each frame output in chronological order. This Library has three functions as below.

(1) Result Stabilizing Function

For example, the OMRON supplying image processing module-HVC outputs detecting result of every frame. This also does for the result with lower confidence like the object facing sideways or taken under bad shooting condition. This Library uses the results with more confidence in the numbers of frames to get the final result of age/gender estimation and face recognition. It also stabilizes Expression Estimation, Gaze Estimation, Blink Estimation and Face Direction Estimation results with previous frames and their confidence.

This result stabilizing process refers maximum 20 previous frames (includes current frame).

(2) Tracking Function

To deliver function above, This Library tracks detecting result of previous and current frames to see whether the people in both frames are the same or not (tracking) and gives ID (tracking ID) for the ones qualified as the same ones. This Library tracks for each Human Body Detection and Face Detection.

(3) Rectangle Steadiness Function

Outputs steadied size and positional information on each frame.

2 Software Specifications

2.1 Library Usage

To use This Library, STBAPI.h, STBCommonDef.h, STBTypedef.h should be added to the include path of the application.

2.2 Error Code Definitions

Error Code	Description	Value
STB_NORMAL	Normal end	0
STB_ERR_INITIALIZE	Initializing Error	-2
STB_ERR_INVALIDPARAM	Parameter Error	-3
STB_ERR_NOHANDLE	Handle Error	-7
STB_ERR_PROCESSCONDITION	Processing condition Error	-8

2.3 Function List

2.4 Processing Sequence

An example of the sequence to use This Library as below.

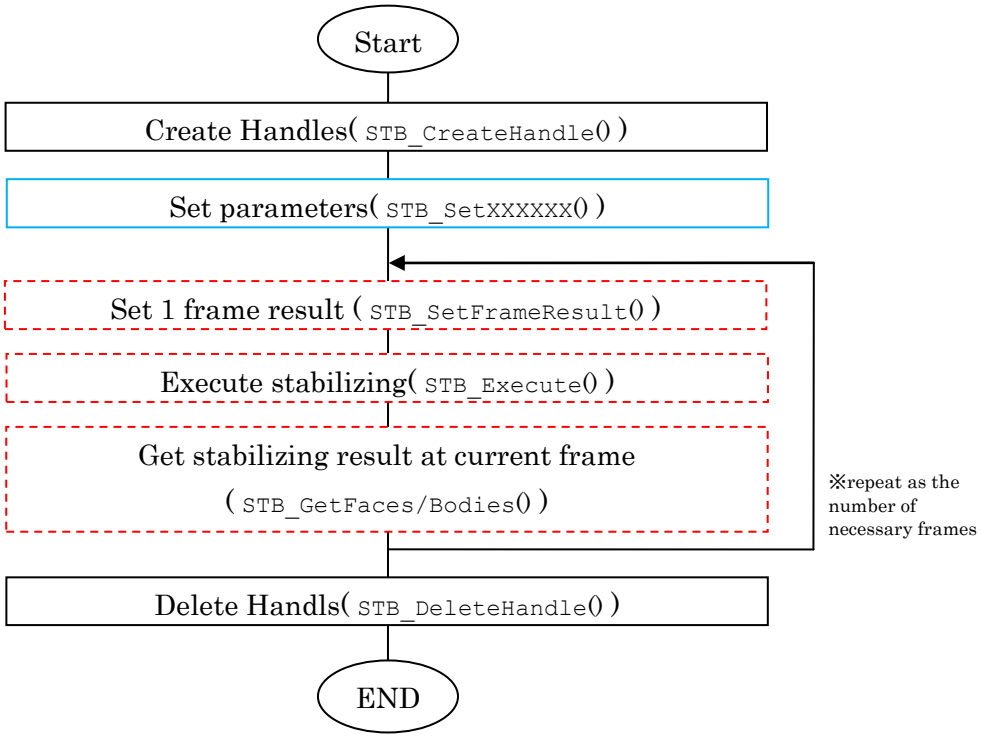

Figure 1 Processing sequence example

Process executing to use This Library

Process executing as needed

Process required for each frame detection/estimation result

2.5 Stabilization Status

This Library outputs stabilization status.

The result information status of **Age/Gender Estimation** and **Face recognition** for a person shifts as No Data→Calculating→Complete→Fixed.

This "No Data/Calculating/Complete/Fixed" will be shown at "`status`".

For **Expression Estimation, Gaze Estimation, Blink Estimation** and **Face Direction Estimation** will be done the stabilizing process, they won't get to the Complete status to see real-time change. The status will shift to No Data to Calculating.

●Stabilizing Status Definition

One of those items below will be output for Stabilizing Status.

`STB_STATUS_NO_DATA`	No Data	: No data of this person
`STB_STATUS_CALCULATING`	Calculating	: Not enough data of this person
`STB_STATUS_COMPLETE`	Complete	: Stabilization process completed
`STB_STATUS_FIXED`	Fixed	: Stabilization result has fixed

2.6 Function Specifications

●Get Version

```
STB_INT32 STB_GetVersion(STB_INT8 *pnMajorVersion , STB_INT8 *pnMinorVersion)
```

Arguments	Output : `pnMajorVersion` `pnMinorVersion`	Major Version Minor Version
Return values	`STB_NORMAL` `STB_ERR_INVALIDPARAM`	Normal end Parameter Error -`NULL` pointer argument
Description	Gets this Library's version	

●Create/Delete Stabilization handle

```
HSTB STB_CreateHandle(STB_UINT32 unUseFuncFlag)
```

Arguments	`unUseFuncFlag`	Flag
Return values	not `NULL` `NULL`	Stabilization handle Failure -insufficient Backup memory -inadequate effectiveness flag
Description	Creates the Stabilization handle. Specify the stabilizing function at `unUseFuncFlag`. Each function is allocated as bit numbers.Set 1 for executing bit, 0 for not executing bit, or "|"to stabilize multiple functions. The defined value for each functions are as follows: `#define STB_FUNC_BD (0x00000001U)` /* [LSB]bit0: Body Tracking */ `#define STB_FUNC_DT (0x00000004U)` /* [LSB]bit2: Face Tracking */ `#define STB_FUNC_PT (0x00000008U)` /* [LSB]bit3: Face Direction */ `#define STB_FUNC_AG (0x00000010U)` /* [LSB]bit4: Age Estimation */ `#define STB_FUNC_GN (0x00000020U)` /* [LSB]bit5: Gender Estimation */ `#define STB_FUNC_GZ (0x00000040U)` /* [LSB]bit6: Gaze Estimation */ `#define STB_FUNC_BL (0x00000080U)` /* [LSB]bit7: Blink Estimation */ `#define STB_FUNC_EX (0x00000100U)` /* [MSB]bit0: Expression Estimation */ `#define STB_FUNC_FR (0x00000200U)` /* [MSB]bit1: Face Recognition */ Activate face detection(`STB_FUNC_DT`) and face direction estimation(`STB_FUNC_PT`) to stabilize age/gender/gaze/blink/expression estimation and face recognition result. * Call `STB_DeleteHandle`() to delete the handle after use.	

```
VOID STB_DeleteHandle(HSTB hSTB)
```

Arguments	Input : `hSTB`	Stabilization handle
Return values	None	
Description	Deletes the handle created at `STB_CreateHandle`().	

●Set frame result

```
STB_INT32 STB_SetFrameResult(HSTB hSTB,
                             const STB_FRAME_RESULT *stFrameResult)
```

Arguments	Input : hSTB Stabilization handle stFrameResult Result information frame
Return values	STB_NORMAL Normal end STB_ERR_NOHANDLE Handle error -improper handle content STB_ERR_INVALIDPARAM Parameter error
Description	Stores detection/estimation information on Stabilization handle. Set the information of face central coordinate, size and direction to stabilize age/gender/gaze/blink/expression estimation and face recognition. Refer 2.7 Struct Definition for input data format of [STB_FRAME_RESULT].
Input specifications	stFrameResult.bodys.nCount : 0～35 stFrameResult.bodys.body[*].center.nX : 0～8191 stFrameResult.bodys.body[*].center.nY : 0～8191 stFrameResult.bodys.body[*].nSize : 20～8192 stFrameResult.bodys.body[*].nConfidence : 0～1000 stFrameResult.faces.nCount : 0～35 stFrameResult.faces.face[*].center.nX : 0～8191 stFrameResult.faces.face[*].center.nY : 0～8191 stFrameResult.faces.face[*].nSize : 20～8192 stFrameResult.faces.face[*].nConfidence : 0～1000 stFrameResult.faces.face[*].direction.nLR : -180～179 stFrameResult.faces.face[*].direction.nUD : -180～179 stFrameResult.faces.face[*].direction.nRoll : -180～179 stFrameResult.faces.face[*].direction.nConfidence : 0～1000 stFrameResult.faces.face[*].age.nAge : 0～75, -128(not estimable) stFrameResult.faces.face[*].age.nConfidence : 0～1000, -128(not estimable) stFrameResult.faces.face[*].gender.nGender : 0(Female), 1(Male), -128(not estimable) stFrameResult.faces.face[*].gender.nConfidence : 0～1000, -128(not estimable) stFrameResult.faces.face[*].gaze.nLR : -90～90, -128(not estimable) stFrameResult.faces.face[*].gaze.nUD : -90～90, -128(not estimable) stFrameResult.faces.face[*].blink.nLeftEye : 1～1000, -128(not estimable) stFrameResult.faces.face[*].blink.nRightEye : 1～1000, -128(not estimable) stFrameResult.faces.face[*].expression.anScore[STB_Expression_Max] : 0～100, -128(not estimable) stFrameResult.faces.face[*].expression.nDegree : -100～100, -128(not estimable) stFrameResult.faces.face[*].recognition.nUID : 0～499, -1(ID not available), -127(not registered), -128(not estimable) stFrameResult.faces.face[*].recognition.nScore : 0～1000, -127(not registered), -128(not estimable)

●Clear frame result

`STB_INT32 STB_ClearFrameResults(HSTB hSTB)`

Arguments	Input : `hSTB`	Stabilization handle
Return values	`STB_NORMAL` `STB_ERR_NOHANDLE`	Normal end Handle error -improper handle content
Description	Clears all the previous saved data. ＊ Executing this function will not delete parameters set at setting functions. Personal ID output at `STB_GetFaces`() and `STB_GetBodies`() will start from "0" after `STB_ClearFrameResults`().	

●Execute stabilizing process

`STB_INT32 STB_Execute(HSTB hSTB)`

Arguments	Input : `hSTB`	Stabilization handle

Return values	`STB_NORMAL`
	`STB_ERR_NOHANDLE`
	
	`STB_ERR_INITIALIZE`
	
	`STB_ERR_INVALIDPARAM`
	`STB_ERR_PROCESSCONDITION`
		Normal end
		Handle error
		-improper handle content
		Initial error
		-no result information
		Parameter error
		Processing Condition error
		-over maximum number of tracking targets

Description	Converts into stabilized result with set previous frame data. Stabilization has those tracking function, rectangle steadiness function and result stabilizing function. - Tracking Function Calculates as the same person by allocating ID for face and human body detecting rectangle. - Rectangle Steadiness Function Steadies face and human body detecting rectangle. - Estimation Result Stabilization Function Stabilizes results of age/gender/gaze/blink/expression/face direction estimation and face recognition.	

●Get stabilized result

```
STB_INT32 STB_GetFaces(HSTB hSTB, STB_UINT32 *punFaceCount, STB_FACE stFace[])
```

Arguments	Input : hSTB	Stabilization handle
	Output : pnFaceCount	Number of stabilized faces
	stFaces	Stabilized face data
Return Values	STB_NORMAL	Normal end
	STB_ERR_NOHANDLE	Handle error -improper handle content
	STB_ERR_INITIALIZE	Initial error -no stabilization
	STB_ERR_INVALIDPARAM	Parameter error
Description	Gets stabilized face data. The data will be output in order of tracking ID. Refer 2.7 Struct Definition for output data format of [STB_FACE].	

```
STB_ INT32 STB_GetBodies(HSTB hSTB, STB_UINT32 *punBodyCount,
                                                    STB_BODY stBody[])
```

Arguments	Input : hSTB	Stabilization handle
	Output : pnBodyCount	Number of stabilized bodies
	stBody	Stabilized body data
Return Values	STB_NORMAL	Normal end
	STB_ERR_NOHANDLE	Handle error -improper handle content
	STB_ERR_INITIALIZE	Initial error -no stabilization
	STB_ERR_INVALIDPARAM	Parameter error
Description	Gets stabilized body data. The data will be output in order of tracking ID. Refer 2.7 Struct Definition for output data format of [STB_BODY].	

●Set/Get maximum retry count

```
STB_INT32 STB_SetTrRetryCount(HSTB hSTB, STB_INT32 nMaxRetryCount)
```

Arguments	Input : hSTB nMaxRetryCount	Stabilization handle Maximum retry count
Return Values	STB_NORMAL STB_ERR_NOHANDLE STB_ERR_INVALIDPARAM	Normal end Handle error -improper handle content Parameter error
Description	Sets the maximum retry count. Set the number of maximum retry when not finding a face/human body while tracking. Terminates tracking as lost object when keeps failing for this maximum retry count.	
Input specifications	nMaxRetryCount : 0 to 300	
Default value	nMaxRetryCount =2	

```
STB_INT32 STB_GetTrRetryCount(HSTB hSTB, STB_INT32 *pnMaxRetryCount)
```

Arguments	Input : hSTB Output : pnMaxRetryCount	Stabilization handle Maximum retry count
Return values	STB_NORMAL STB_ERR_NOHANDLE STB_ERR_INVALIDPARAM	Normal end Handle error -improper handle content Parameter error -NULL pointer argument
Description	Gets the set maximum retry count.	

●Set/Get rectangle steadiness parameter

```
STB_INT32 STB_SetTrSteadinessParam
(HSTB hSTB, STB_INT32 nPosSteadinessParam, STB_INT32 nSizeSteadinessParam)
```

Arguments	Input : hSTB	Stabilization handle
	nPosSteadinessParam	Rectangle position steadiness parameter
	nSizeSteadinessParam	Rectangle size steadiness parameter
Return values	STB_NORMAL	Normal end
	STB_ERR_NOHANDLE	Handle error -improper handle content
	STB_ERR_INVALIDPARAM	Parameter error
Description	Sets steadiness parameter of position and size. - nPosSteadinessParam For example, outputs the previous position coordinate data if the shifting measure is within 30%, existing position coordinate data if it has shift more than 30% when the rectangle position steadiness parameter has set as initial value of 30. - nSizeSteadinessParam For example, outputs the previous detecting size data if the changing measure is within 30%, existing size data if it has changed more than 30% when the rectangle size steadiness parameter has set as initial value of 30.	
Input specifications	nPosSteadinessParam : 0 to 100 nSizeSteadinessParam : 0 to 100	
Default value	nPosSteadinessParam = 30 nSizeSteadinessParam = 30	

```
STB_INT32 STB_GetTrSteadinessParam(HSTB hSTB,
        STB_INT32 *pnPosSteadinessParam, STB_INT32 *pnSizeSteadinessParam)
```

Arguments	Input : hSTB	Stabilization handle
	Output: pnPosSteadinessParam	Rectangle position steadiness parameter
	pnSizeSteadinessParam	Rectangle size steadiness parameter
Return Values	STB_NORMAL	Normal end
	STB_ERR_NOHANDLE	Handle error -improper handle content
	STB_ERR_INVALIDPARAM	Parameter error -NULL pointer argument
Description	Gets the set steadiness parameter of position and size.	

●Set/Get estimation result stabilizing threshold value

```
STB_INT32 STB_SetPeThresholdUse(HSTB hSTB, STB_INT32 nThreshold)
```

Arguments	Input : hSTB	Stabilization handle
	nThreshold	Face direction confidence threshold value
Return values	STB_NORMAL	Normal end
	STB_ERR_NOHANDLE	Handle error -improper handle content
	STB_ERR_INVALIDPARAM	Parameter error
Description	Sets the stabilizing threshold value of Face direction confidence. ＊ This is for the six functions of age, gender, gaze, blink, expression and face direction estimation functions. Eliminates face data with lower confidence than the value set at this function for accuracy improvement of result stabilizing. For example, the previous data confidence with below 500 will not be applied for stabilizing when the face direction confidence threshold value has set as 500.	
Input specifications	nThreshold : 0 to 1000	
Default value	nThreshold = 300	

```
STB_INT32 STB_GetPeThresholdUse(HSTB hSTB, STB_INT32 *pnThreshold)
```

Arguments	Input : hSTB	Stabilization handle
	Output : pnThreshold	Face direction confidence threshold value
Return values	STB_NORMAL	Normal end
	STB_ERR_NOHANDLE	Handle error -improper handle content
	STB_ERR_INVALIDPARAM	Parameter error -NULL pointer argument
Description	Gets the set threshold value of Face direction confidence.	

●Set/Get estimation result stabilizing angle

```
STB_INT32 STB_SetPeAngleUse(HSTB hSTB, STB_INT32 nMinUDAngle,
     STB_INT32 nMaxUDAngle, STB_INT32 nMinLRAngle, STB_INT32 nMaxLRAngle)
```

Arguments	Input : hSTB	Stabilization handle
	nMinUDAngle	Minimum up-down angle of the face
	nMaxUDAngle	Maximum up-down angle of the face
	nMinLRAngle	Minimum left-right angle of the face
	nMaxLRAngle	Maximum left-right angle of the face
Return values	STB_NORMAL	Normal end
	STB_ERR_NOHANDLE	Handle error -improper handle content
	STB_ERR_INVALIDPARAM	Parameter error
Description	Sets angle threshold value of Face direction. ∗ This is for the six functions of age, gender, gaze, blink, expression and face direction estimation functions. Eliminates face data with out of the set angle at this function for accuracy improvement of result stabilizing. For example, the previous data with up-down angle of below -16degree and over 21 degree will not be applied for stabilizing when the up-down angle threshold value of Face direction has set as 15 for minimum and 21 for maximum.	
Input specifications	nMinUDAngle : -90 to 90 nMaxUDAngle : -90 to 90 nMinLRAngle : -90 to 90 nMaxLRAngle : -90 to 90 nMinUDAngle ≦ nMaxUDAngle nMinLRAngle ≦ nMaxLRAngle	
Default value	nMinUDAngle = -15 nMaxUDAngle = 20 nMinLRAngle = -30 nMaxLRAngle = 30	

```
STB_INT32 STB_GetPeAngleUse(HSTB hSTB, STB_INT32 *pnMinUDAngle,
STB_INT32 *pnMaxUDAngle, STB_INT32 *pnMinLRAngle, STB_INT32 *pnMaxLRAngle)
```

Arguments	Input : hSTB	Stabilization handle
	Output : pnMinUDAngle	Minimum up-down angle of the face
	pnMaxUDAngle	Maximum up-down angle of the face
	pnMinLRAngle	Minimum left-right angle of the face
	pnMaxLRAngle	Maximum left-right angle of the face
Return values	STB_NORMAL	Normal end
	STB_ERR_NOHANDLE	Handle error -improper handle content
	STB_ERR_INVALIDPARAM	Parameter error -NULL pointer argument
Description	Gets the set angle threshold value of Face direction.	

●Set/Get age/gender estimation complete frame count

```
STB_INT32 STB_SetPeCompleteFrameCount(HSTB hSTB, STB_INT32 nFrameCount)
```

Arguments	Input : hSTB	Stabilization handle
	nFrameCount	The number of previous frames applying to fix the result
Return values	STB_NORMAL	Normal end
	STB_ERR_NOHANDLE	Handle error -improper handle content
	STB_ERR_INVALIDPARAM	Parameter error
Description	Sets the number of previous frames applying to fix stabilization. ＊ This is for the two functions of age and gender estimation. The data used for stabilizing process (=averaging) is only the one fulfilled the STB_SetPeThresholdUse and STB_SetPeAngleUse condition. Stabilizing process will be completed with data more than the number of frames set at this function and it won't be done with less data.	
Input specifications	nFrameCount : 1 to 20	
Default value	nFrameCount = 5	

```
STB_INT32 STB_GetPeCompleteFrameCount(HSTB hSTB, STB_INT32 *pnFrameCount)
```

Arguments	Input : hSTB	Stabilization handle
	Output : pnFrameCount	The number of previous frames applying to fix the result
Return values	STB_NORMAL	Normal end
	STB_ERR_NOHANDLE	Handle error -improper handle content
	STB_ERR_INVALIDPARAM	Parameter error -NULL pointer argument
Description	Gets the set complete frame count.	

●Set/Get recognition stabilizing threshold value

```
STB_INT32 STB_SetFrThresholdUse(HSTB hSTB, STB_INT32 nThreshold)
```

Arguments	Input : hSTB nThreshold	Stabilization handle Face direction confidence threshold value
Return values	STB_NORMAL STB_ERR_NOHANDLE STB_ERR_INVALIDPARAM	Normal end Handle error -improper handle content Parameter error
Description	Sets stabilizing threshold value of Face direction confidence to improve recognition stabilization. Eliminates face data with lower confidence than the value set at this function. For example, the previous data confidence with below 500 will not be applied for stabilizing when the face direction confidence threshold value has set as 500.	
Input specifications	nThreshold : 0 to 1000	
Default value	nThreshold = 300	

```
STB_INT32 STB_GetFrThresholdUse(HSTB hSTB, STB_INT32 *pnThreshold)
```

Arguments	Input : hSTB Output : pnThreshold	Stabilization handle Face direction confidence threshold value
Return values	STB_NORMAL STB_ERR_NOHANDLE STB_ERR_INVALIDPARAM	Normal end Handle error -improper handle content Parameter error -NULL pointer argument
Description	Gets the set threshold value of Face direction confidence.	

●Set/Get recognition stabilizing angle

```
STB_INT32 STB_SetFrAngleUse(HSTB hSTB, STB_INT32 nMinUDAngle,
      STB_INT32 nMaxUDAngle, STB_INT32 nMinLRAngle, STB_INT32 nMaxLRAngle)
```

Arguments	Input : hSTB	Stabilization handle
	nMinUDAngle	Minimum up-down angle of the face
	nMaxUDAngle	Maximum up-down angle of the face
	nMinLRAngle	Minimum left-right angle of the face
	nMaxLRAngle	Maximum left-right angle of the face
Return values	STB_NORMAL	Normal end
	STB_ERR_NOHANDLE	Handle error -improper handle content
	STB_ERR_INVALIDPARAM	Parameter error
Description	Sets angle threshold value of Face direction for accuracy improvement of recognition stabilizing. Eliminates face data with out of the set angle at this function. For example, the previous data with up-down angle of below -16degree and over 21 degree will not be applied for stabilizing when the up-down angle threshold value of Face direction has set as 15 for minimum and 21 for maximum.	
Input specifications	nMinUDAngle : -90 to 90 nMaxUDAngle : -90 to 90 nMinLRAngle : -90 to 90 nMaxLRAngle : -90 to 90 nMinUDAngle ≦ nMaxUDAngle nMinLRAngle ≦ nMaxLRAngle	
Default value	nMinUDAngle = -15 nMaxUDAngle = 20 nMinLRAngle = -30 nMaxLRAngle = 30	

```
STB_INT32 STB_GetFrAngleUse(HSTB hSTB, STB_INT32 *pnMinUDAngle,
STB_INT32 *pnMaxUDAngle, STB_INT32 *pnMinLRAngle, STB_INT32 *pnMaxLRAngle)
```

Arguments	Input : hSTB	Stabilization handle
	Output : pnMinUDAngle	Minimum up-down angle of the face
	pnMaxUDAngle	Maximum up-down angle of the face
	pnMinLRAngle	Minimum left-right angle of the face
	pnMaxLRAngle	Maximum left-right angle of the face
Return values	STB_NORMAL	Normal end
	STB_ERR_NOHANDLE	Handle error -improper handle content
	STB_ERR_INVALIDPARAM	Parameter error -NULL pointer argument
Description	Gets the set angle threshold value of Face direction.	

●Set/Get recognition stabilizing complete frame count

```
STB_INT32 STB_SetFrCompleteFrameCount(STB_HANDLE hSTB, STB_INT32 nFrameCount)
```

Arguments	Input : hSTB	Stabilization handle
	nFrameCount	The number of previous frames applying to fix the result
Return values	STB_NORMAL	Normal end
	STB_ERR_NOHANDLE	Handle error -improper handle content
	STB_ERR_INVALIDPARAM	Parameter error
Description	Sets the number of previous frames applying to fix the recognition stabilizing. The data used for stabilizing process (=averaging) is only the one fulfilled the STB_SetFrThresholdUse and STB_SetFrAngleUse condition. Stabilizing process will be completed with a recognition ID fulfilled seizing ratio in result fixing frames and will not be done without one. * Refer STB_SetFrMinRatio function for account ratio function.	
Input specifications	nFrameCount : 0 to 20	
Default value	nFrameCount = 5	

```
STB_INT32 STB_GetFrCompleteFrameCount (STB_HANDLE hSTB,
                                       STB_INT32 *pnFrameCount)
```

Arguments	Input : hSTB	Stabilization handle
	Output : pnFrameCount	The number of previous frames applying to fix the result
Return values	STB_NORMAL	Normal end
	STB_ERR_NOHANDLE	Handle error -improper handle content
	STB_ERR_INVALIDPARAM	Parameter error -NULL pointer argument
Description	Gets the set complete frame count.	

●Set/Get recognition minimum account ratio

`STB_INT32 STB_SetFrMinRatio(STB_HANDLE hSTB, STB_INT32 nMinRatio)`

Arguments	Input：`hSTB` `nMinRatio`	Stabilization handle Recognition minimum account ratio

Return values	`STB_NORMAL`
	`STB_ERR_NOHANDLE`
	
	`STB_ERR_INVALIDPARAM`
		Normal end
		Handle error
		-improper handle content
		Parameter error

Description	Sets minimum account ratio in complete frame count for accuracy improvement of recognition stabilizing. For example, when there are 7 frames of extracted usable data in referred previous 20 frames, `STB_SetFrCompleteFrameCount` function has set “10”for the complete frame count and “60” for the recognition minimum account ratio. Creates frequency distribution of recognition result in the set 10 frames. Recognized as “Mr. A”（1 frame） Recognized as “Mr. B”（4 frames） Recognized as “Mr. C”（2 frames） In this case, the most account ratio “Mr. B” will be output as stabilized result. However, this recognition status will be output as “`STB_STATUS_CALCULATING`” since the account ratio is about 57%(= 4 frames/10 frames) , (Mr. B seizing ratio=) 57% < recognition account ratio (=60%).	
Input specifications	`nMinRatio` : 0 to 100	
Default value	`nMinRatio` = 60	

`STB_INT32 STB_GetFrMinRatio (STB_HANDLE hSTB, STB_INT32 *pnMinRatio)`

Arguments	Input : `hSTB` Output : `pnMinRatio`	Stabilization handle Recognition minimum account ratio

Return values	`STB_NORMAL`
	`STB_ERR_NOHANDLE`
	
	`STB_ERR_INVALIDPARAM`
		Normal end
		Handle error
		-improper handle content
		Parameter error
		-`NULL` pointer argument

Description	Gets the set recognition minimum account ratio.	

2.7 Struct Definitions

●Input result information

STB_FRAME_RESULT

Members	STB_FRAME_RESULT_BODYS bodys	Human body detection result
	STB_FRAME_RESULT_FACES faces	Face detection result
Description	Input result information	

●Human body detection result information

STB_FRAME_RESULT_BODYS

Members	STB_INT32 nCount	Detecting number
	STB_FRAME_RESULT_DETECTION body[35]	Human body detection result detail
Description	Human body detection result information	

●Face detection result information

STB_FRAME_RESULT_FACES

Members	STB_INT32 nCount	Detecting number
	STB_FRAME_RESULT_FACE face[35]	Face detection result detail
Description	Face detection result information	

●Human body detection result detail

STB_FRAME_RESULT_DETECTION

Members	STB_POINT center	Central coordinate
	STB_INT32 nSize	Size
	STB_INT32 nConfidence	Confidence
Description	Human body detection result detail	

●Face detection result detail

STB_FRAME_RESULT_FACE

Members	STB_POINT center	Central coordinate
	STB_INT32 nSize	Size
	STB_INT32 nConfidence	Confidence
	STB_FRAME_RESULT_DIRECTION direction	Face direction estimation result
	STB_FRAME_RESULT_AGE age	Age estimation result
	STB_FRAME_RESULT_GENDER gender	Gender estimation result
	STB_FRAME_RESULT_GAZE gaze	Gaze estimation result
	STB_FRAME_RESULT_BLINK blink	Blink estimation result
	STB_FRAME_RESULT_EXPRESSION expression	Expression estimation result
	STB_FRAME_RESULT_RECOGNITION recognition	Face recognition result
Description	Face detection result detail	

●Coordinate

STB_POINT

Members	STB_INT32 nX	X Coordinate
	STB_INT32 nY	Y Coordinate
Description	Coordinate	

●Face direction estimation result

STB_FRAME_RESULT_DIRECTION

Members	STB_INT32 nLR	Left-right angle
	STB_INT32 nUD	Up-down angle
	STB_INT32 nRoll	Roll angle
	STB_INT32 nConfidence	Confidence
Description	Face direction(Degree) estimation result	

●Age estimation result

STB_FRAME_RESULT_AGE

Members	STB_INT32 nAge	Age
	STB_INT32 nConfidence	Confidence
Description	Age estimation result	

●Gender estimation result

STB_FRAME_RESULT_GENDER

Members	STB_INT32 nGender	Gender
	STB_INT32 nConfidence	Confidence
Description	Gender estimation result	

●Gaze estimation result

STB_FRAME_RESULT_GAZE

Members	STB_INT32 nLR	Left-right angle
	STB_INT32 nUD	Up-down angle
Description	Gaze(Degree) estimation result	

●Blink estimation result

STB_FRAME_RESULT_BLINK

Members	STB_INT32 nLeftEye	Left eye score
	STB_INT32 nRightEye	Right eye score
Description	Blink estimation result	

●Expression estimation result

STB_FRAME_RESULT_EXPRESSION

Members	STB_INT32 anScore[STB_Expression_Max]	Each expression estimation score
	STB_INT32 nDegree	Positive/Negative degree
Description	Expression estimation result	

●Face recognition result

STB_FRAME_RESULT_RECOGNITION

Members	STB_INT32 nUID	User ID
	STB_INT32 nScore	Score
Description	Face recognition result	

●Stabilized face result

STB_FACE

Members	STB_INT32 nDetectID	Detecting ID
	STB_INT32 nTrackingID	Tracking ID
	STB_POS center	Central coordinate
	STB_UINT32 nSize	Size
	STB_INT32 conf	Tracking confidence
	STB_DIR direction	Stabilized face direction result
	STB_RES age	Stabilized age result
	STB_RES gender	Stabilized gender result
	STB_GAZE gaze	Stabilized gaze result
	STB_BLINK blink	Stabilized blink result
	STB_RES expression	Stabilized expression result
	STB_RES recognition	Stabilized recognition result
Description	Stabilizes face detection result nDetectID allocates IDs on face detection result in order of storing. IDs are allocated from 0 to 34 and -1 for retry counting object. Therefore take it as an effective value when nDetectID allocated -1 as tracking is in process. nTrackingID allocates IDs on tracking objects in order starting from 0. Executing STB_ClearFrameResults() makes the ID back to 0. Those center, nSize, conf refer stabilized central coordinate, size and confidence.	

●Stabilized human body result

STB_BODY

Members	STB_INT32 nDetectID	Detected ID
	STB_INT32 nTrackingID	Tracking ID
	STB_POS center	Central coordinate
	STB_UINT32 nSize	Size
	STB_INT32 conf	Tracking confidence
Description	Stabilized human body result	

●Coordinate

STB_POS

Members	STB_UINT32 x	X coordinate
	STB_UINT32 y	Y coordinate
Description	Coordinate	

●Stabilized blink result

STB_BLINK

Members	STB_STATUS status STB_INT32 ratioL STB_INT32 ratioR	Stabilization status Left eye score Right eye score
Description	Stabilized blink result Refer 2.5 Stabilization status for stabilization status.	

●Stabilized face direction result

STB_DIR

Members	STB_STATUS status STB_INT32 conf STB_INT32 yaw STB_INT32 pitch STB_INT32 roll	Stabilization status Stabilization confidence Up-down angle Left-right angle Rolling angle
Description	Stabilized face direction (Degree) result Refer 2.5 Stabilization status for stabilization status.	

●Stabilized gaze result

STB_GAZE

Members	STB_STATUS status STB_INT32 conf STB_INT32 UD STB_INT32 LR	Stabilization status Stabilization confidence Up-down angle Left-right angle
Description	Stabilized gaze(Degree)result Refer 2.5 Stabilization status for stabilization status.	

●Stabilized result

STB_RES

Members	STB_STATUS status STB_INT32 value STB_INT32 conf	Stabilization status Age/gender/expression /recognition stabilized result Stabilization confidence (This will be stabilization score in case of recognition)
Description	Stabilization result frame Refer 2.5 Stabilization status for stabilization status.	

2.8 Enum Type Definitions

●Expression Estimation

```
typedef enum {
    STB_Expression_Neutral,
    STB_Expression_Happiness,
    STB_Expression_Surprise,
    STB_Expression_Anger,
    STB_Expression_Sadness,
    STB_Expression_Max
} STB_OKAO_EXPRESSION;
```

Definition	Description
STB_Expression_Neutral	Neutral
STB_Expression_Happiness	Happiness
STB_Expression_Surprise	Surprise
STB_Expression_Anger	Anger
STB_Expression_Sadness	Sadness

●Stabilization Expression Estimation

```
typedef enum {
    STB_EX_UNKNOWN = -1,
    STB_EX_NEUTRAL = 0,
    STB_EX_HAPPINESS,
    STB_EX_SURPRISE,
    STB_EX_ANGER,
    STB_EX_SADNESS,
    STB_EX_MAX
}STB_EXPRESSION;
```

Definition	Description
STB_EX_UNKNOWN	-1=Unknown
STB_EX_NEUTRAL	0=Neutral
STB_EX_HAPPINESS	Happiness
STB_EX_SURPRISE	Surprise
STB_EX_ANGER	Anger
STB_EX_SADNESS	Sadness

●Stabilization Status

```
typedef enum {
    STB_STATUS_NO_DATA = -1,
    STB_STATUS_CALCULATING = 0,
    STB_STATUS_COMPLETE = 1,
    STB_STATUS_FIXED = 2,
    STB_STATUS_MAX;
} STB_STATUS;
```

Definition	Description
STB_STATUS_NO_DATA	No data : No data of this person
STB_STATUS_CALCULATING	Calculating : Not enough data of this person -number of frames with object person
STB_STATUS_COMPLETE	Complete : Stabilization process completed
STB_STATUS_FIXED	Fixed : Stabilization result has fixed