

## Recognition Component

0.80

作成 : Doxygen 1.6.1

Fri Jun 24 10:22:41 2011



# Contents

<b>1</b>	<b>構成索引</b>	<b>1</b>
1.1	構成 . . . . .	1
<b>2</b>	<b>ファイル索引</b>	<b>3</b>
2.1	ファイル一覧 . . . . .	3
<b>3</b>	<b>クラス</b>	<b>5</b>
3.1	クラス CameraCaptureServiceSVC_impl . . . . .	5
3.1.1	説明 . . . . .	5
3.2	クラス Recognition . . . . .	6
3.2.1	説明 . . . . .	7
3.2.2	コンストラクタとデストラクタ . . . . .	7
3.2.2.1	Recognition . . . . .	7
3.2.3	変数 . . . . .	7
3.2.3.1	m_DebugDisplay . . . . .	7
3.2.3.2	m_DebugImage . . . . .	7
3.2.3.3	m_DebugText . . . . .	8
3.2.3.4	m_recogModelListPath . . . . .	8
3.2.3.5	m_Recognition . . . . .	8
3.2.3.6	m_RecognitionPort . . . . .	8
3.2.3.7	m_RecognitionResultOut . . . . .	8
3.2.3.8	m_RecognitionResultViewer . . . . .	9
3.2.3.9	m_RecognitionResultViewerPort . . . . .	9
3.2.3.10	m_recogParameterFilePath . . . . .	9
3.2.3.11	m_Reconstruct3D . . . . .	9
3.2.3.12	m_Reconstruct3DPort . . . . .	9

3.2.3.13	<code>m_stereo3DDataIn</code>	10
3.2.3.14	<code>m_stereo3DDataOut</code>	10
3.3	クラス <code>RecognitionResultViewerServiceSVC_impl</code>	11
3.3.1	説明	11
3.4	クラス <code>RecognitionServiceSVC_impl</code>	12
3.4.1	説明	13
3.5	クラス <code>Reconstruct3DServiceSVC_impl</code>	14
3.5.1	説明	14
<b>4</b>	<b>ファイル</b>	<b>15</b>
4.1	<code>execute3DRecognition.h</code>	15
4.1.1	説明	16
4.1.2	関数	16
4.1.2.1	<code>execute3DRecognition</code>	16
4.2	<code>ImgSkel.cpp</code>	17
4.2.1	説明	17
4.3	<code>ImgSkel.h</code>	18
4.3.1	説明	18
4.4	<code>ImgStub.cpp</code>	19
4.4.1	説明	19
4.5	<code>ImgStub.h</code>	20
4.5.1	説明	20
4.6	<code>Recognition.cpp</code>	21
4.6.1	説明	21
4.7	<code>Recognition.h</code>	22
4.7.1	説明	23
4.8	<code>RecognitionComp.cpp</code>	24
4.8.1	説明	24
4.9	<code>VisionSkel.cpp</code>	25
4.9.1	説明	25
4.10	<code>VisionSkel.h</code>	26
4.10.1	説明	26
4.11	<code>VisionStub.cpp</code>	27
4.11.1	説明	27

---

4.12	VisionStub.h . . . . .	28
4.12.1	説明 . . . . .	28
4.13	VisionSVC_impl.cpp . . . . .	29
4.13.1	説明 . . . . .	29
4.14	VisionSVC_impl.h . . . . .	30
4.14.1	説明 . . . . .	31



# Chapter 1

## 構成索引

### 1.1 構成

クラス、構造体、共用体、インタフェースの説明です。

<a href="#">CameraCaptureServiceSVC_impl</a>	5
<a href="#">Recognition (3D Recognition by Model )</a>	6
<a href="#">RecognitionResultViewerServiceSVC_impl</a>	11
<a href="#">RecognitionServiceSVC_impl</a>	12
<a href="#">Reconstruct3DServiceSVC_impl</a>	14





## Chapter 2

# ファイル索引

### 2.1 ファイル一覧

これはファイル一覧です。

<a href="#">execute3DRecognition.h</a> (3次元物体認識の実行)	15
<a href="#">ImgSkel.cpp</a> (Img server skeleton wrapper)	17
<a href="#">ImgSkel.h</a> (Img server skeleton header wrapper code)	18
<a href="#">ImgStub.cpp</a> (Img client stub wrapper code)	19
<a href="#">ImgStub.h</a> (Img client stub header wrapper code)	20
<a href="#">Recognition.cpp</a> (3D Recognition by Model)	21
<a href="#">Recognition.h</a> (3D Recognition by Model)	22
<a href="#">RecognitionComp.cpp</a> (Standalone component)	24
<a href="#">VisionSkel.cpp</a> (Vision server skeleton wrapper)	25
<a href="#">VisionSkel.h</a> (Vision server skeleton header wrapper code)	26
<a href="#">VisionStub.cpp</a> (Vision client stub wrapper code)	27
<a href="#">VisionStub.h</a> (Vision client stub header wrapper code)	28
<a href="#">VisionSVC_impl.cpp</a> (Service implementation code of Vision.idl)	29
<a href="#">VisionSVC_impl.h</a> (Service implementation header of Vision.idl)	30



## Chapter 3

# クラス

### 3.1 クラス CameraCaptureServiceSVC\_impl

```
#include <VisionSVC_impl.h>
```

#### Public メソッド

- [CameraCaptureServiceSVC\\_impl \(\)](#)  
*standard constructor*
- [virtual ~CameraCaptureServiceSVC\\_impl \(\)](#)  
*destructor*
- `void take_one_frame ()`

#### 3.1.1 説明

Example class implementing IDL interface `Img::CameraCaptureService`

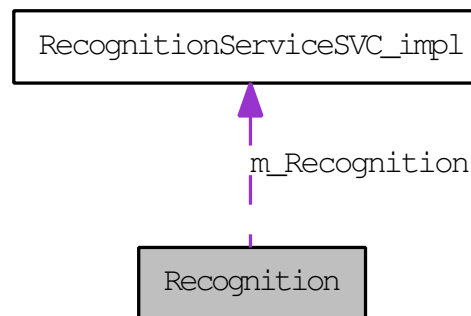
このクラスの説明は次のファイルから生成されました:

- [VisionSVC\\_impl.h](#)
- [VisionSVC\\_impl.cpp](#)

## 3.2 クラス Recognition

3D [Recognition](#) by Model

#include <Recognition.h>Recognition のコラボレーション図



### Public メソッド

- [Recognition](#) (RTC::Manager \*manager)  
*constructor*
- [~Recognition](#) ()  
*destructor*
- virtual RTC::ReturnCode\_t **onInitialize** ()
- virtual RTC::ReturnCode\_t **onActivated** (RTC::UniqueId ec\_id)
- virtual RTC::ReturnCode\_t **onDeactivated** (RTC::UniqueId ec\_id)
- virtual RTC::ReturnCode\_t **onExecute** (RTC::UniqueId ec\_id)

### Protected 変数

- std::string [m\\_recogModelListPath](#)
- std::string [m\\_recogParameterFilePath](#)
- int [m\\_DebugText](#)
- int [m\\_DebugImage](#)
- int [m\\_DebugDisplay](#)
- TimedStereo3D **m\_stereo3DData**
- InPort< TimedStereo3D > [m\\_stereo3DDataIn](#)
- OutPort< TimedStereo3D > [m\\_stereo3DDataOut](#)
- TimedDoubleSeq **m\_ReognitionResult**
- OutPort< TimedDoubleSeq > [m\\_ReognitionResultOut](#)

- RTC::CorbaPort [m\\_RecognitionPort](#)
- RTC::CorbaPort [m\\_Reconstruct3DPort](#)
- RTC::CorbaPort [m\\_RecognitionResultViewerPort](#)
- [RecognitionServiceSVC\\_impl](#) [m\\_Recognition](#)
- RTC::CorbaConsumer< Reconstruct3DService > [m\\_Reconstruct3D](#)
- RTC::CorbaConsumer< RecognitionResultViewerService > [m\\_RecognitionResultViewer](#)

### 3.2.1 説明

3D [Recognition](#) by Model ステレオ画像を受け取り、指定モデルの 3 次元物体認識を行う。

### 3.2.2 コンストラクタとデストラクタ

#### 3.2.2.1 Recognition::Recognition (RTC::Manager \* *manager*)

constructor

引数:

*manager* Maneger Object

### 3.2.3 変数

#### 3.2.3.1 int Recognition::m\_DebugDisplay [protected]

デバッグ用画像情報表示スイッチ

- Name: DebugDisplay
- DefaultValue: 0

#### 3.2.3.2 int Recognition::m\_DebugImage [protected]

デバッグ用画像情報出力スイッチ

- Name: DebugImage
- DefaultValue: 0

### 3.2.3.3 `int Recognition::m_DebugText` `[protected]`

デバッグ用テキスト情報出力スイッチ

- Name: DebugText
- DefaultValue: 0

### 3.2.3.4 `std::string Recognition::m_recogModelListPath` `[protected]`

認識モデルのモデル ID とモデルファイルのパスの 一覧ファイルのファイル名を指定する。

- Name: RecogModelListPath recogModelListPath
- DefaultValue: modelList.txt

### 3.2.3.5 `RecognitionServiceSVC_impl Recognition::m_Recognition` `[protected]`

認識に使用するモデルを指定する。

- Argument: モデル ID : long ModelID
- Return Value: 0: 正常終了

### 3.2.3.6 `RTC::CorbaPort Recognition::m_RecognitionPort` `[protected]`

認識に使用するモデルをモデル ID で指定する。

### 3.2.3.7 `OutPort<TimedDoubleSeq> Recognition::m_RecognitionResultOut` `[protected]`

認識結果の出力

- Type: TimedDoubleSeq
- Number: 1

**3.2.3.8 RTC::CorbaConsumer<RecognitionResultViewerService>  
Recognition::m\_RecognitionResultViewer [protected]**

認識結果を表示する。

- Argument: 認識画像: TimedMultiCameraImage frame 認識結果: TimedRecognitionResult pos
- Return Value: 0 : 正常終了。

**3.2.3.9 RTC::CorbaPort Recognition::m\_RecognitionResultViewerPort  
[protected]**

認識結果を表示する。

**3.2.3.10 std::string Recognition::m\_recogParameterFilePath [protected]**

認識パラメータファイルのファイル名を指定する。

- Name: RecogParameterFilePath recogParameterFilePath
- DefaultValue: recogParameter.txt

**3.2.3.11 RTC::CorbaConsumer<Reconstruct3DService>  
Recognition::m\_Reconstruct3D [protected]**

3次元距離計測データの送信を要求する。

- Argument: なし。
- Return Value: なし。

**3.2.3.12 RTC::CorbaPort Recognition::m\_Reconstruct3DPort  
[protected]**

距離計測データの送信を要求する。

### 3.2.3.13 InPort<TimedStereo3D> Recognition::m\_stereo3DDataIn [protected]

認識画像を含む距離計測データ

- Type: TimedStereo3D
- Number: 1

### 3.2.3.14 OutPort<TimedStereo3D> Recognition::m\_stereo3DDataOut [protected]

InPort Stereo3DIn に入力されたデータをスルー出力する。

- Type: TimedStereo3D
- Number: 1

このクラスの説明は次のファイルから生成されました:

- [Recognition.h](#)
- [Recognition.cpp](#)



### 3.3 クラス `RecognitionResultViewerServiceSVC_impl`

```
#include <VisionSVC_impl.h>
```

#### Public メソッド

- [RecognitionResultViewerServiceSVC\\_impl \(\)](#)  
*standard constructor*
- `virtual ~RecognitionResultViewerServiceSVC_impl ()`  
*destructor*
- `void display (const Img::TimedMultiCameraImage &frame, const Time-  
dRecognitionResult &pos)`

#### 3.3.1 説明

Example class implementing IDL interface `RecognitionResultViewerService`

このクラスの説明は次のファイルから生成されました:

- [VisionSVC\\_impl.h](#)
- [VisionSVC\\_impl.cpp](#)

### 3.4 クラス RecognitionServiceSVC\_impl

```
#include <VisionSVC_impl.h>
```

#### Public メソッド

- [RecognitionServiceSVC\\_impl \(\)](#)  
*standard constructor*
- [virtual ~RecognitionServiceSVC\\_impl \(\)](#)  
*destructor*
- [CORBA::Long getModelID \(\)](#)  
現在設定されているモデル ID を返す。
- [void setModelID \(CORBA::Long ModelID\)](#)  
モデル ID を設定し、認識用画像を要求する。
- [void setModelList \(ModelFileInfo \\*info\)](#)  
モデルファイル一覧のセット
- [char \\* getModelFilePath \(\)](#)  
現在セットされている認識モデル ID のモデルファイル名を返す。
- [void setRecogParameterPath \(const char \\*path\)](#)  
認識設定ファイルのパスを設定する。
- [void setDebugParameter \(int text, int image, int display\)](#)  
デバッグ用パラメータを設定する。
- [int loadRecogParameter \(\)](#)  
認識設定ファイルから認識設定を読み込んで設定を適用する。
- [int loadDebugParameter \(\)](#)  
デバッグ用パラメータ設定を適用する。
- [void getCurrentRecogParameter \(Parameters \\*param\)](#)  
現在の認識パラメータを取得する。
- [bool getModelIDUpdateFlag \(\)](#)  
モデル ID が更新されたかどうかのフラグを取得する。
- [void resetModelIDUpdateFlag \(\)](#)

モデル *ID* 更新フラグのリセット

### 3.4.1 説明

Example class implementing IDL interface `RecognitionService`

このクラスの説明は次のファイルから生成されました:

- [VisionSVC\\_impl.h](#)
- [VisionSVC\\_impl.cpp](#)

## 3.5 クラス Reconstruct3DServiceSVC\_impl

```
#include <VisionSVC_impl.h>
```

### Public メソッド

- [Reconstruct3DServiceSVC\\_impl \(\)](#)  
*standard constructor*
- virtual [~Reconstruct3DServiceSVC\\_impl \(\)](#)  
*destructor*
- void **reconstruct** ()

### 3.5.1 説明

Example class implementing IDL interface Reconstruct3DService

このクラスの説明は次のファイルから生成されました:

- [VisionSVC\\_impl.h](#)
- [VisionSVC\\_impl.cpp](#)

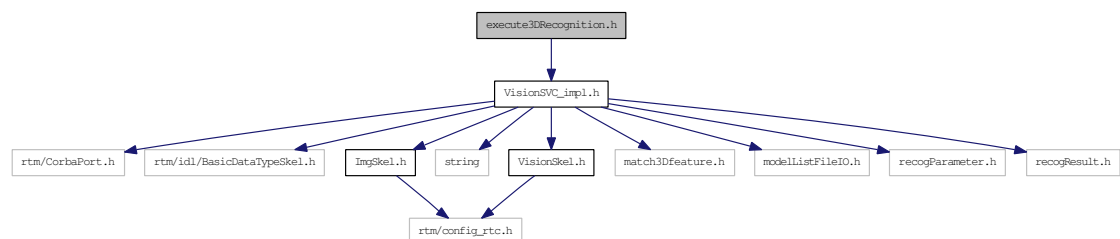
## Chapter 4

# ファイル

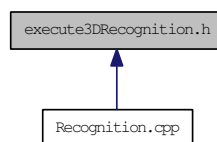
### 4.1 execute3DRecognition.h

3次元物体認識の実行 `#include "VisionSVC_impl.h"`

execute3DRecognition.h のインクルード依存関係図



このグラフは、どのファイルから直接、間接的にインクルードされているかを示しています。



### 関数

- int `execute3DRecognition` (Img::TimedMultiCameraImage &frame, int modelID, char \*modelFilePath, Parameters &param, TimedRecognitionResult &re-

sult)

3次元物体認識の実行

#### 4.1.1 説明

3次元物体認識の実行

#### 4.1.2 関数

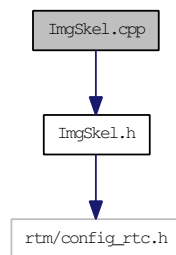
**4.1.2.1 int execute3DRecognition (Img::TimedMultiCameraImage & frame, int modelID, char \* modelFilePath, Parameters & param, TimedRecognitionResult & result)**

3次元物体認識の実行 画像データから3次元特徴を抽出し、指定モデルファイルとマッチングを行い、結果を返す。引数: Img::TimedMultiCameraImage& frame : カメラ画像とキャリブレーションデータ char\* modelFilePath : モデルファイル名 Parameters& param : 認識パラメータ TimedRecognitionResult& result : 認識結果

## 4.2 ImgSkel.cpp

Img server skeleton wrapper. `#include "ImgSkel.h"`

ImgSkel.cpp のインクルード依存関係図



### 4.2.1 説明

Img server skeleton wrapper. THIS FILE IS GENERATED AUTOMATICALLY!! DO NOT EDIT!!

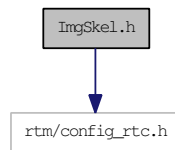
日付:

Thu Jun 23 15:17:23 2011

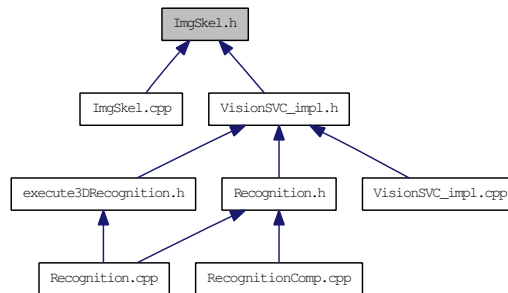
## 4.3 ImgSkel.h

Img server skeleton header wrapper code. `#include <rtm/config_rtc.h>`

ImgSkel.h のインクルード依存関係図



このグラフは、どのファイルから直接、間接的にインクルードされているかを示しています。



### 4.3.1 説明

Img server skeleton header wrapper code. THIS FILE IS GENERATED AUTOMATICALLY!! DO NOT EDIT!!

日付:

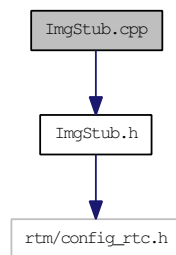
Thu Jun 23 15:17:23 2011



## 4.4 ImgStub.cpp

Img client stub wrapper code. `#include "ImgStub.h"`

ImgStub.cpp のインクルード依存関係図



### 4.4.1 説明

Img client stub wrapper code. THIS FILE IS GENERATED AUTOMATICALLY!!  
DO NOT EDIT!!

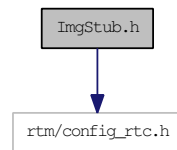
日付:

Thu Jun 23 15:17:23 2011

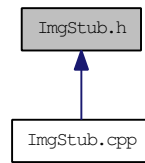
## 4.5 ImgStub.h

Img client stub header wrapper code. `#include <rtm/config_rtc.h>`

ImgStub.h のインクルード依存関係図



このグラフは、どのファイルから直接、間接的にインクルードされているかを示しています。



### 4.5.1 説明

Img client stub header wrapper code. THIS FILE IS GENERATED AUTOMATICALLY!! DO NOT EDIT!!

日付:

Thu Jun 23 15:17:23 2011

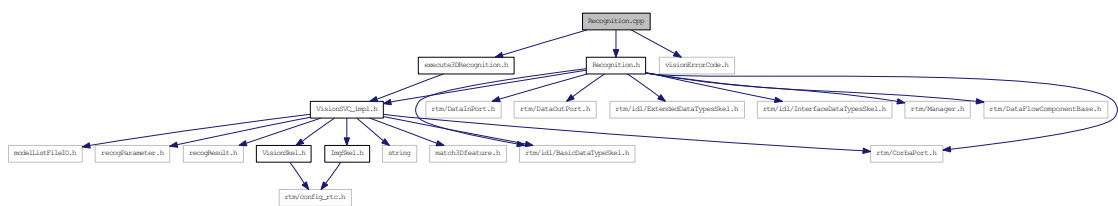
## 4.6 Recognition.cpp

3D [Recognition](#) by Model #include "Recognition.h"

#include "execute3DRecognition.h"

#include "visionErrorCode.h"

Recognition.cpp のインクルード依存関係図



### 関数

- void **RecognitionInit** (RTC::Manager \*manager)

#### 4.6.1 説明

3D [Recognition](#) by Model

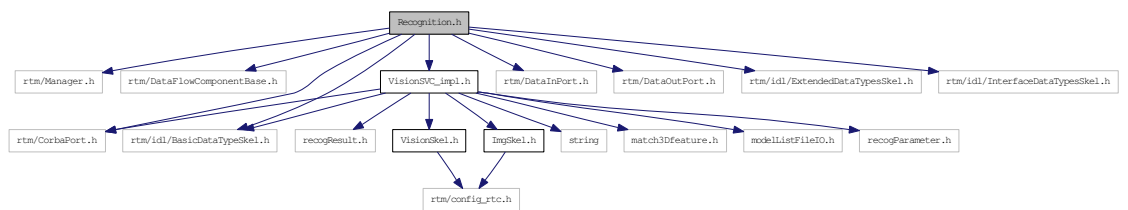
日付:

\$Date:: 2011-06-23 14:28:12 +0900 # \$

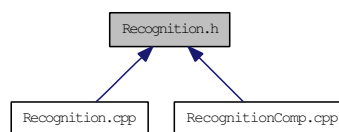
## 4.7 Recognition.h

```
3D Recognition by Model #include <rtm/Manager.h>
#include <rtm/DataFlowComponentBase.h>
#include <rtm/CorbaPort.h>
#include <rtm/DataInPort.h>
#include <rtm/DataOutPort.h>
#include <rtm/idl/BasicDataTypeSkel.h>
#include <rtm/idl/ExtendedDataTypesSkel.h>
#include <rtm/idl/InterfaceDataTypesSkel.h>
#include "VisionSVC_impl.h"
```

Recognition.h のインクルード依存関係図



このグラフは、どのファイルから直接、間接的にインクルードされているかを示しています。



### 構成

- class **Recognition**  
*3D **Recognition** by Model*

### 関数

- DLL\_EXPORT void **RecognitionInit** (RTC::Manager \*manager)

**4.7.1 説明**

3D [Recognition](#) by Model

日付:

\$Date:: 2011-06-23 14:28:12 +0900 #

## 4.8 RecognitionComp.cpp

Standalone component. #include <rtm/Manager.h>

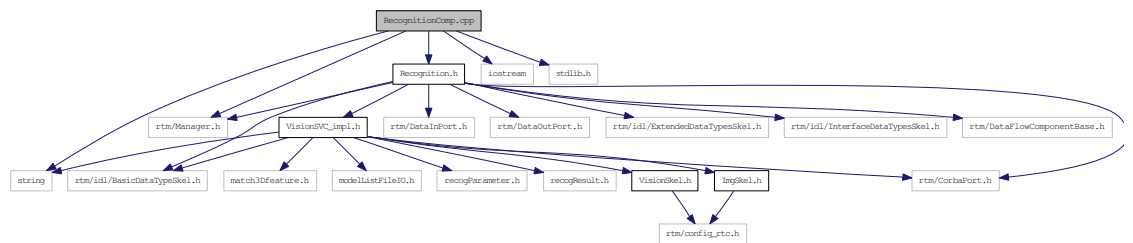
```
#include <iostream>
```

```
#include <string>
```

```
#include <stdlib.h>
```

```
#include "Recognition.h"
```

RecognitionComp.cpp のインクルード依存関係図



### 関数

- void **MyModuleInit** (RTC::Manager \*manager)
- int **main** (int argc, char \*\*argv)

### 4.8.1 説明

Standalone component.

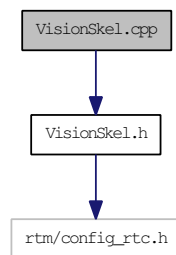
日付:

\$Date:: 2011-06-23 14:28:12 +0900 # \$

## 4.9 VisionSkel.cpp

Vision server skeleton wrapper. `#include "VisionSkel.h"`

VisionSkel.cpp のインクルード依存関係図



### 4.9.1 説明

Vision server skeleton wrapper. THIS FILE IS GENERATED AUTOMATICALLY!!  
DO NOT EDIT!!

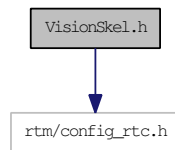
日付:

Thu Jun 23 15:17:22 2011

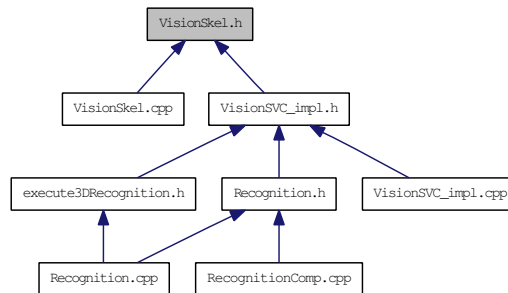
## 4.10 VisionSkel.h

Vision server skeleton header wrapper code. `#include <rtm/config_rtc.h>`

VisionSkel.h のインクルード依存関係図



このグラフは、どのファイルから直接、間接的にインクルードされているかを示しています。



### 4.10.1 説明

Vision server skeleton header wrapper code. THIS FILE IS GENERATED AUTOMATICALLY!! DO NOT EDIT!!

日付:

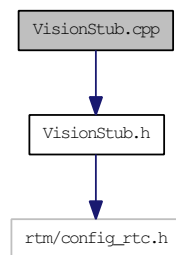
Thu Jun 23 15:17:22 2011



## 4.11 VisionStub.cpp

Vision client stub wrapper code. #include "VisionStub.h"

VisionStub.cpp のインクルード依存関係図



### 4.11.1 説明

Vision client stub wrapper code. THIS FILE IS GENERATED AUTOMATICALLY!!  
DO NOT EDIT!!

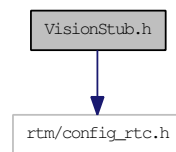
日付:

Thu Jun 23 15:17:22 2011

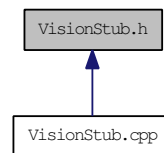
## 4.12 VisionStub.h

Vision client stub header wrapper code. `#include <rtm/config_rtc.h>`

VisionStub.h のインクルード依存関係図



このグラフは、どのファイルから直接、間接的にインクルードされているかを示しています。



### 4.12.1 説明

Vision client stub header wrapper code. THIS FILE IS GENERATED AUTOMATICALLY!! DO NOT EDIT!!

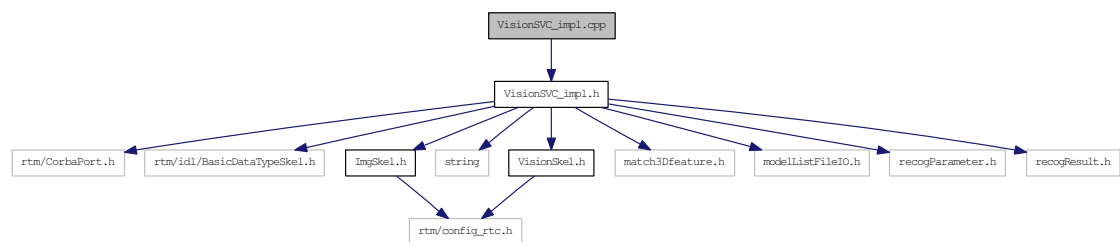
日付:

Thu Jun 23 15:17:22 2011

## 4.13 VisionSVC\_impl.cpp

Service implementation code of Vision.idl. #include "VisionSVC\_impl.h"

VisionSVC\_impl.cpp のインクルード依存関係図



### 4.13.1 説明

Service implementation code of Vision.idl.

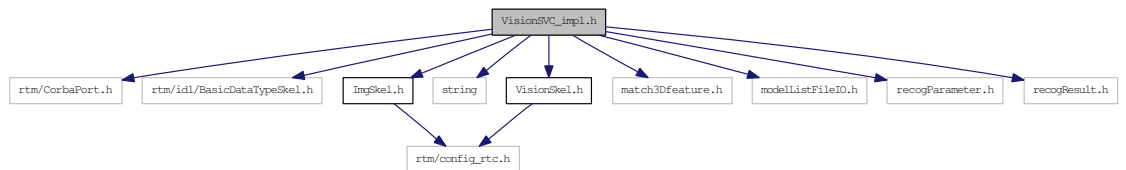
日付:

\$Date:: 2011-06-23 14:28:12 +0900 #

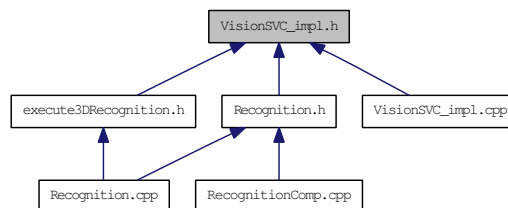
## 4.14 VisionSVC\_impl.h

```
Service implementation header of Vision.idl. #include <rtm/CorbaPort.h>
#include <rtm/idl/BasicDataTypeSkel.h>
#include "ImgSkel.h"
#include <string>
#include "VisionSkel.h"
#include "match3Dfeature.h"
#include "modelListFileIO.h"
#include "recogParameter.h"
#include "recogResult.h"
```

VisionSVC\_impl.h のインクルード依存関係図



このグラフは、どのファイルから直接、間接的にインクルードされているかを示しています。



## 構成

- class [CameraCaptureServiceSVC\\_impl](#)
- class [Reconstruct3DServiceSVC\\_impl](#)
- class [RecognitionServiceSVC\\_impl](#)
- class [RecognitionResultViewerServiceSVC\\_impl](#)

**4.14.1 説明**

Service implementation header of Vision.idl.