

TransAM Framework implementation of custom face recognition engine

In TransAM Framework Cognitive, if you want to implement a third-party face recognition engine other than Microsoft's, it provides two SPI interfaces, FaceSpi and RemoteSpi, respectively. FaceSpi provides methods for local and remote face services, and on the contrary, RemoteSpi provides remote face service methods.

Java SPI (Service Provider Interface)

SPI, or Service Provider Interface, is a service discovery mechanism built into the JDK. It allows you to dynamically find service implementations for an interface, somewhat similar to the idea of IOC (Inversion of Control), which moves the control of the assembly out of the process, and is especially important in modular designs. To use the SPI mechanism, you need to create a document named Service Interface in the META-INF/services/ directory under the classpath, and the content of this document is the concrete implementation class of this interface.

Using Face Recognition on Pepper

The Pepper robot supports face recognition in NAOqi version 2.9.3 and above, which provides developers with a solution to implement face recognition on Pepper, and allows developers to integrate third-party face recognition engines other than Microsoft's without modifying the logic code.

Approximate steps to implement the Pepper Face Recognition SPI functionality

- Create an Android Application project.
- Import TransAM Framework Cognitive into the project.
- Implement FaceSpi and RemoteFaceSpi interfaces.

Introducing Dependency Packages

In the Android project, add the maven repository:

```
maven {
    url 'http://maven.softbankrobotics.com.cn/releases'
}
```

Add the following dependency to the **build.gradle** file of your Android project:

```
implementation 'cn.softbankrobotics.transam:transam-cognitive:1.0.6'
```

Create the resources/META-INF/services folder in the Android project and create two files with the following contents:

```
* File name: com.softbankrobotics.transam.cognitive.face.spi.FaceSpi
The content is: the full path name of the class that implements FaceSpi in the project, e.g.: com.p
* File name: com.softbankrobotics.transam.cognitive.face.spi.RemoteFaceSpi
The content is: the full pathname of the class that implements FaceSpi in the project, e.g.: com.pa
```

TransAM Framework Cognitive SPI Usage Example

In TransAM Framework Cognitive SPI, there are two interfaces FaceSpi and RemoteSpi.

| Interface Name | Properties | Description | |-----| |-----| |-----| |-----| | FaceSpi interface | FaceSpi implements the face detection and face matching interface and distinguishes whether it is implemented locally or remotely by the isRemote property.

| RemoteSpi | RemoteSpi | interface | RemoteSpi contains parts of the interface that require a remote service to implement. | RemoteSpi

FaceSpi interface description

IsRemote

Get whether the current face service interface is provided locally or accessed remotely. FaceSpi provides two interfaces, face detection and face matching, which can be implemented locally or remotely respectively.

Interface Description

Inherit FaceSpi interface, override isRemote() function.

```
@Override
public boolean isRemote() {}
```

Return Parameters

return type	description
boolean	Determines if the current implementation is local or remote.

Init

Initialise the FaceService service, mainly loading the face service engine.

Interface

The init() function with parameters is used to configure the configuration of the third party face recognition service engine.

```
@Override
public void init(Object configuration) {}
```

Parameter Description

| parameter name | property | description | |-----| |-----| |-----| |-----| | configuration | Object | Parameter information used to configure the third-party face recognition service engine |

detect

Face detection and attribute information return.

Interface Description

Detect face information from image data.

```
@Override
public FaceDetectResult detect(InputStream faceImage) throws FaceException {}
```

Parameter Description

| Parameter Name | Attribute | Description | |-----| |-----| |-----| |-----| | faceImage | InputStream | User's image information |

Return Parameters

Return Type	Description
FaceDetectResult	Returns face detection information

The FaceDetectResult class includes face attribute information.

| Parameter Name | Properties | Description | |-----| |-----| |-----| |-----| | faceNumber | int | Number of faces recognised | | faceRectangles | Rectangle[] | Positions of rectangles of recognised faces |

verify

Face authentication.

Interface Description

Compares the confidence level of two face images.

```
@Override
public double verify(InputStream firstFace, InputStream secondFace) throws FaceException {}
```

Parameter Description

| Parameter Name | Attribute | Description | |-----| |-----| |-----| |-----| | firstFace | InputStream | The face image being matched | secondFace | InputStream | The current detected face image | secondFace | InputStream | The currently detected face image |

Return Parameters

Return Type	Description
double	confidence value

getEngineName

Returns the name of the currently used face recognition service engine. SPI loads the current engine through this function.

Interface Description

Used by SPI to return the engine name.

```
@Override
public String getEngineName() {}
```

Return Parameters

Return Type	Description
String	EngineName

RemoteFaceSpi interface description

Init

Initialise the FaceService service, mainly load the face service engine.

Interface Description

The init() function with parameters is used to configure the configuration of the third party face recognition service engine.

```
@Override
public void init(Object configuration) {}
```

Parameter Description

| Parameter Name | Attribute | Description | |-----| |-----| |-----| |-----| | configuration | Object | Parameter information for configuring the third-party face recognition service engine |

createPerson

Creates the personal information that will be saved to the Face Recognition Service Engine.

Interface Description

```
@Override
public String createPerson(String groupName, Person person) throws FaceException {}
```

Parameter Description

| ParameterName | Attribute | Description | |-----| |-----| |-----| |-----| | groupName | String | The name of the group created in the Face Recognition Service engine | | personId | String | The user's personal ID, a unique identifier generated by the Face recognition service engine |

AddFace

Registers the face information to the Face Recognition Service Engine and binds the unique identifier of the person's personId.

Interface Description

```
@Override
public String addFace(String groupName, String personId, Rectangle targetFace, InputStream faceImage
```

Parameter Description

| Parameter Name | Attribute | Description | |-----| |-----| |-----| |-----| | groupName | String | The name of the group created in the Face Recognition Service engine | | personId | String | The user's personal ID, a unique identifier generated by the Face Recognition Service engine | | targetFace | Rectangle | Position of the target face in the rectangle frame of the image | | faceImage | InputStream | The image information of the face |

Return Parameters

Return Type	Description
String	Unique identifier of the individual's faceId returned from the face recognition service engine

search

Get user information by searching the face recognition service engine with the group name and current face image information.

Interface Description

```
@Override
public Person search(String groupName, InputStream faceImage) throws FaceException {}
```

Parameter Description

| ParameterName | Attribute | Description | |-----| |-----| |-----| |-----| | groupName | String | The name of the group created in the Face Recognition Service Engine | | faceImage | InputStream | The image information of the face |

Return Parameters

Return Type	Description
Person	Information about the person registered, including username, identifier, personId

Information about the Person class is included:

| parameter name | attribute | description | |-----| |-----| |-----| |-----| | userName | String | The title of the user, this can be the user's real name, pronoun, etc. | | identifier | String | The user's unique identifier, which can be the user's mobile phone number, ID card number, etc. | | personId | String | The user's person ID, generated by the face recognition service engine |

getPerson

Get the personal information registered in the Face Recognition Service Engine.

Interface Description

```
@Override
public Person getPerson(String groupName, String personId) throws FaceException {}
```

Parameter Description

| ParameterName | Property | Description | |-----| |-----| |-----| |-----| | groupName | String | The name of the group created in the face recognition service engine | | personId | String | The user's personal ID, a unique identifier generated by the Face Recognition Service engine |

Return Parameters

Return Type	Description
Person	Information about the person registered, including username, identifier, personId

Information about the Person class is included:

| parameter name | attribute | description | |-----| |-----| |-----| |-----| | userName | String | The title of the user, this can be the user's real name, pronoun, etc. | | identifier | String | The user's unique identifier, which can be the user's mobile phone number, ID number, etc.