**Scan in Travel**

**Software Design**

By

**Yirui Wang 542115509**

**Ruijuan Liu 542115508**

Department of Software Engineering

College of Arts, Media and Technology

Chiang Mai University

**Project Advisor**

**Yun Rim Park**

**Document History**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Document Name** | **Version** | **Status** | **Date** | **Viewable** | **Editable** | **Responsible** |
| **Documents** | | | | | | |
| Scan in Travel \_Software Requirement Specification | Software Requirement Specification\_V0.1 | Reviewed |  | Yun Rim Park  Yirui Wang  Ruijuan Liu | Yirui Wang  Ruijuan Liu | Yirui Wang  Ruijuan Liu |

**Document Approved by**

**Yun Rim Park**

**Table of Contents**

**Chapter One | Introduction**

**1.1 Objective**

The objective of the Software Design Document for “Scan In Travel” Android mobile application is to provide a description of the design of a system fully enough to allow for software development to proceed with an understanding of what is to be built and how it is expected to be built. Once the software design document is approved, it becomes a baseline for limiting changes in the scope of the project.  
 The Software Design Document provides information necessary to provide description of the details for the software and system to be built. This document is based on the project proposal, project plan and system requirement specification. It contains detailed design, method design, class design, sequence diagram, entity relationship diagram and user interface design. It can help developers to understand the work, and guide them to implement the right software.

**1.2 Project Scope**

“Scan in Travel” is a mobile application which using Android OS. This application can help users solve the language problems during their travel in China. Users can get the meaning of Chinese word(s) by scanning it with the built-in camera of their mobile phone. And users can also get the map and route if the scanned name is the name of a Location in China. What is more, “Scan in Travel” can also help users remember where they have been. Users can record the attractive places into their Favorites.

The main features of “Scan in Travel” will be following:

**Translation system**

Users can translate Chinese word(s) and Location name(s) into English by scanning them with the built-in camera of their mobile phone. After viewing the English translation of the word(s), users can also select the text-to-speech function to listen to the pronunciation of the Chinese word(s). Moreover, users can edit the Chinese word(s) that are scanned to get new explanation in English.

**Favorites system**

Users can save Chinese words and their English translation to their own Word book. If users scan a Location name, they can also save the Location with its name, pictures, geometry information (i.e. latitude and longitude), and descriptions into their Favorites. Users can take the picture of the Location and write down their own description for it. Moreover, users can sort and search words in their own Word book and Locations in their Favorites.

**Map system**

Users can keep their own map of every Location that they saved. The system lets users see every saved Location on a map, and check the details of each Location. Besides, users can scan a Location name and find where it is located on a map. The system will guide users to get to the Location by providing a route on a map.

**1.2 Acronyms and Definitions**

**1.2.1 Acronyms**

SDD Software Design Document

CD Class Diagram

SD Sequence Diagram

SRS Software Requirement Specification

URS User Requirement Specification

UI User Interface

OSD Orientating and Script Detection

**1.2.2 Definitions**

**Feature** Transformation of input parameters to output parameters based on a specified algorithm. It describes the functionality of a product in the language of the product. Used for requirements analysis, design, coding, testing or maintenance.

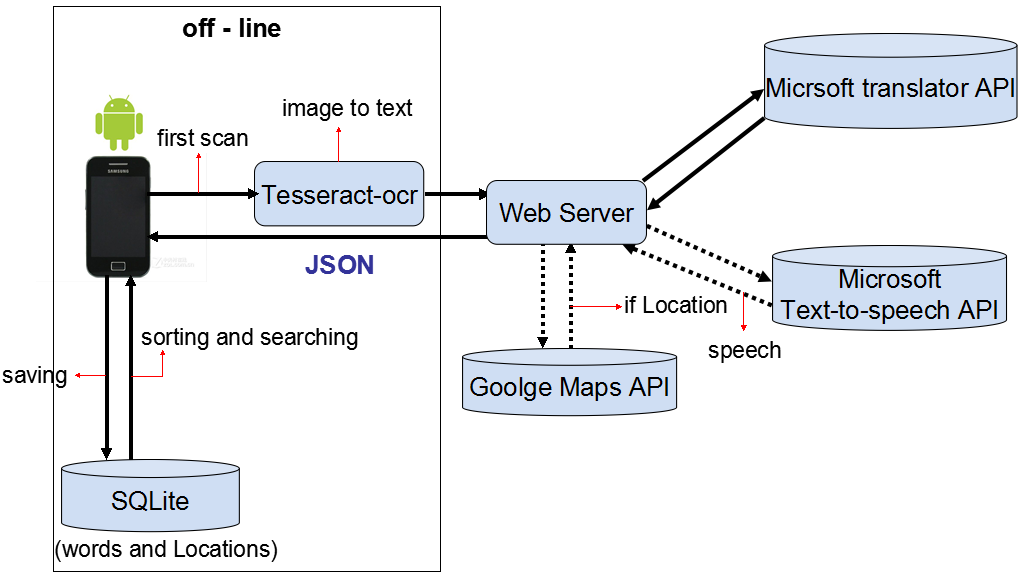
**Sequence diagram** A sequence diagram is an [interaction diagram](http://en.wikipedia.org/wiki/Interaction_diagram) that shows how processes operate with one another and in what order. A sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario.

**Class diagram** Class diagrams are an aspect of UML that describe a static design of the objects, and their relationships to each other, in an application. During analysis, class diagrams may just be the names of objects and how they interact, but as the design develops the details of each class, including attributes and methods, are displayed in the diagram. A class is represented by a rectangle divided into three rows. The top row contains the name of the class. The middle row lists the attribute names, while the third row lists the method names for the class.

**User interface** User interface (UI) is everything designed into an information device with which a human being may interact -- including display screen, keyboard, mouse, light pen, the appearance of a desktop, illuminated characters, help messages, and how an application program or a Web site invites interaction and responds to it.

**UML** The Unified Modeling Language (UML) is a general-purpose [modeling language](http://en.wikipedia.org/wiki/Modeling_language) in the field of [software engineering](http://en.wikipedia.org/wiki/Software_engineering), which is designed to provide a standard way to visualize the design of a system.

**Chapter Two | System Architecture**



***Figure 1: System Architecture (Overview)***

**Tesseract-ocr**

OCR (optical character recognition) means the processing that scans the text data, analyzes the image file and extracts the text from the image. Tesseract-OCR is an open source OCR engine, and developers can use it directly.

**Microsoft translator API**

Microsoft Translator provides a powerful set of web service APIs for developers to use its Machine Translation technology in their own applications, services or web sites. Microsoft Translator API can be invoked in variable ways, including an HTTP REST Service, an AJAX-callable service and a SOAP Web Service.

**Google Maps API**

Google Maps API is a programming API that Google provides to developers. It allows developers to embed data of Google Maps into their web or application. And it helps developers provide the location services to users. In addition, Google Maps API not only helps developers to embed maps into their application, but also allows developers use JavaScript to expand their applications. Developers can add labels to the map and route, respond the user click, and use popup windows to show the information.

**SQLite**

SQLite is a lightweight database. It is an embedded database engine. Nowadays, there are many embedded software system that use SQLite. It takes lower resources, for an embedded device, we only need a little memory. What’s more, it supports Android/ Windows/ Linux/ Unix operating systems, and it can be combined with many programming languages, such as C#, PHP, Java and etc.

**JSON (JavaScript Object Notation)**

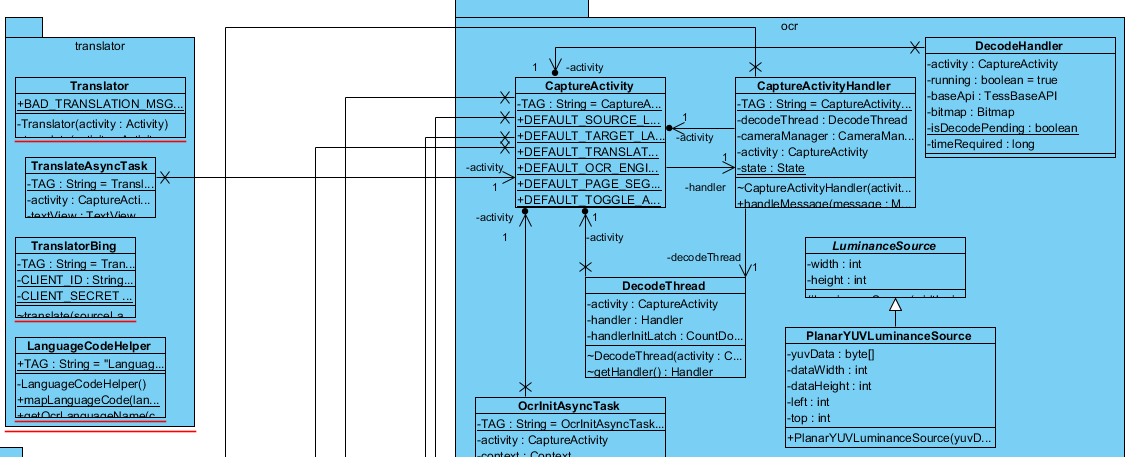
JSON is a lightweight data interchange format. It is based on a subset of JavaScript. JSON can covert a set of data represented in JavaScript Object into string. Then the string can be transferred between functions easily, for instance, from the Web client to server in asynchronous application.

**Chapter Three | Detailed Design**

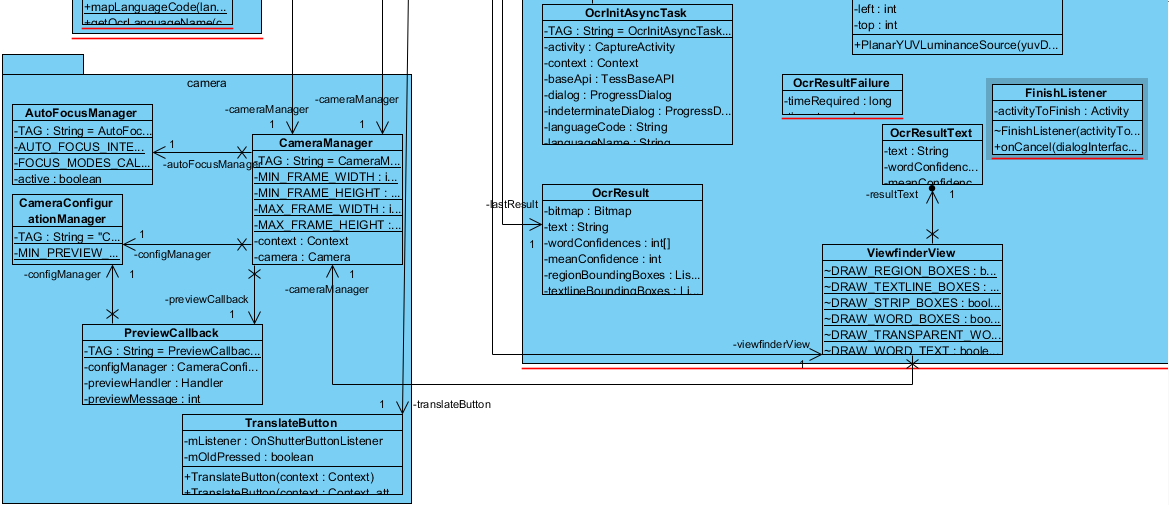
**3.1 Class Diagram**

This section shows the packages and their classes that support the features described in SRS for translation system, and also the relationships between the classes and packages presented in diagram. What’s more, we also provide the design of core APIs that we use to support the features in our system.

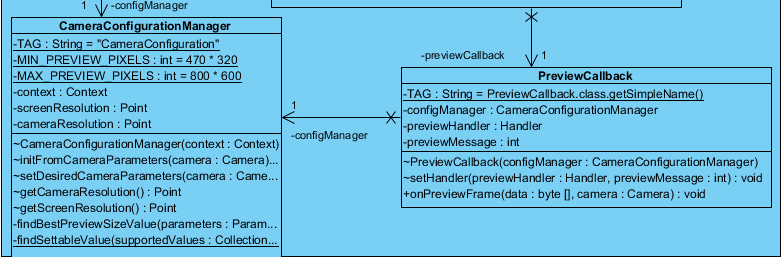
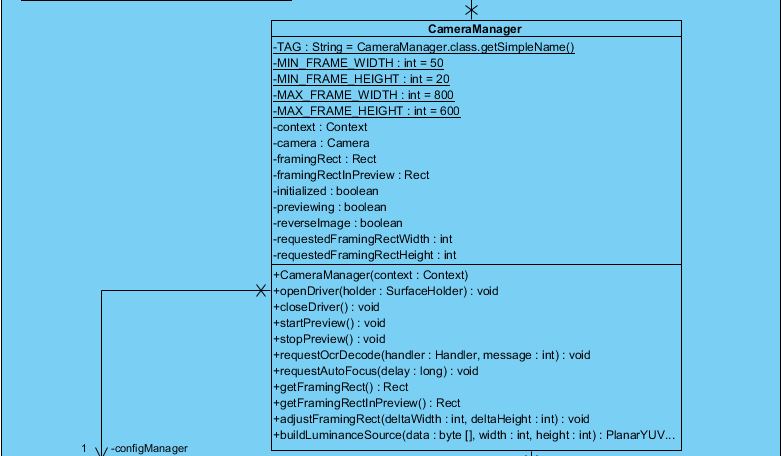
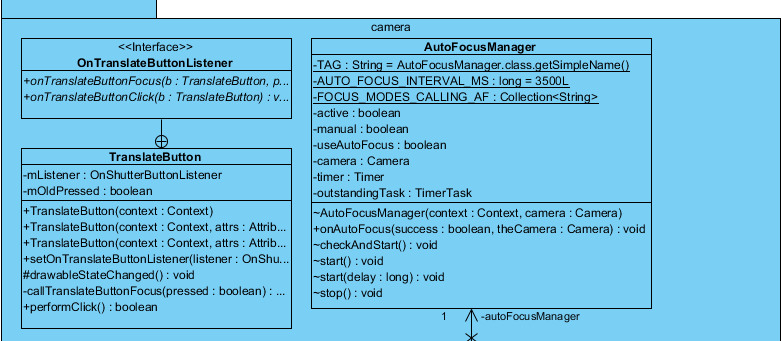
**3.1.1 Scan in Travel overall class diagram**



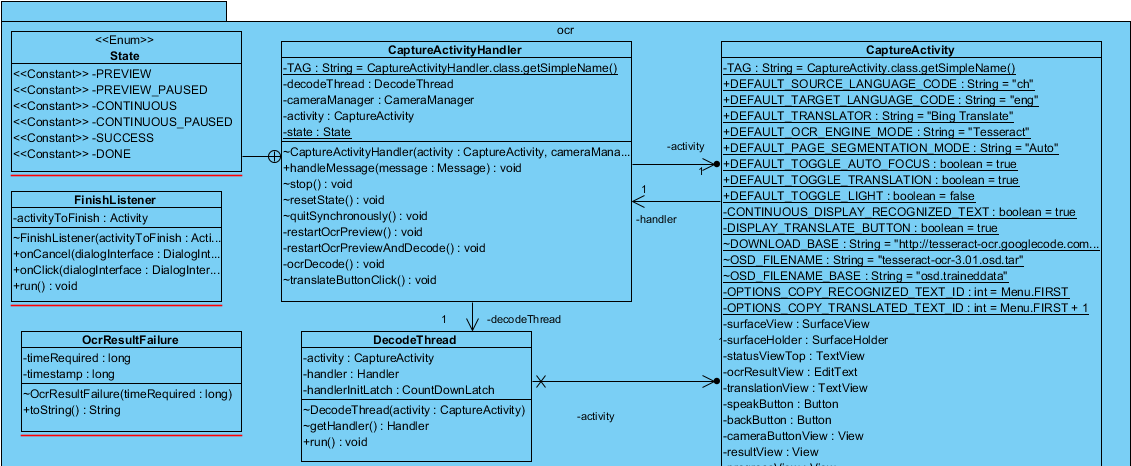
***Diagram continues on the next page***



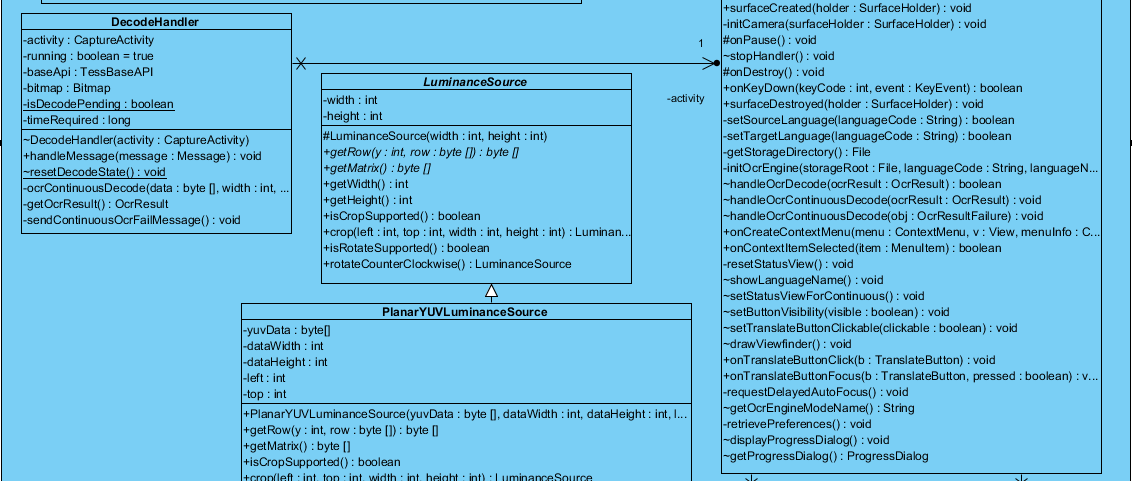
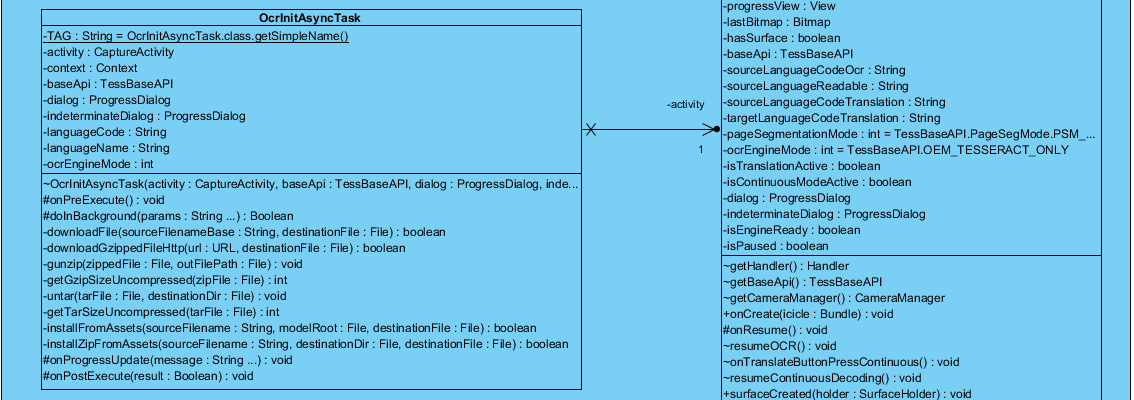
**3.1.2 Package camera**



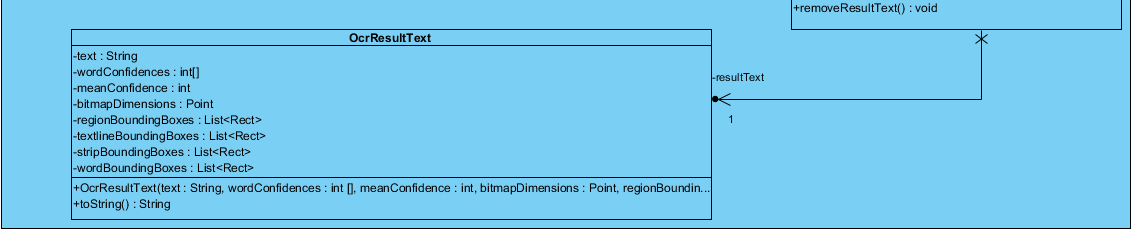
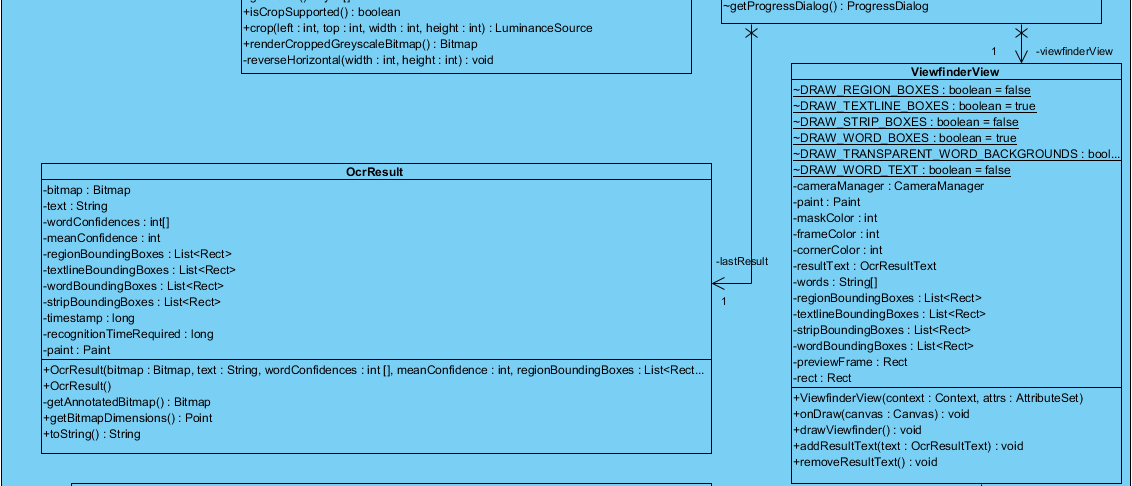
**3.1.3 Package ocr**



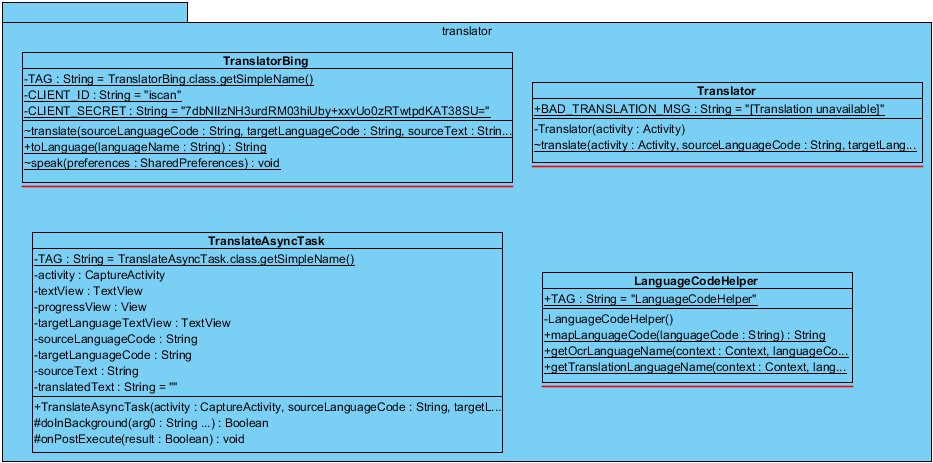
***Diagram continues on the next page***



***Diagram continues on the next page***



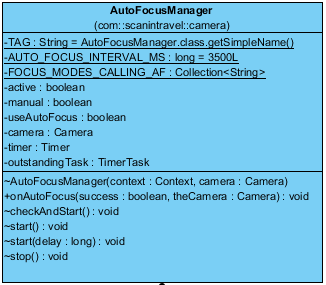
**3.1.4 Package translator**



**3.2 Class Description**

**3.2.1 Package camera**

**3.2.1.1 Class-01 AutoFocusManager**



**Description:**

This class implements Camera.AutoFocusCallback, is used for managing the auto focus of the hardware camera.

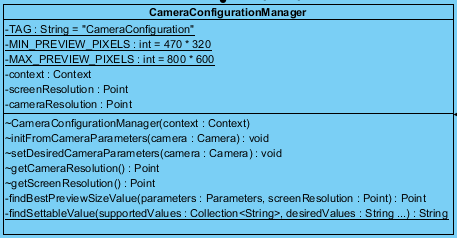
**Attributes:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Name** | **Description** | **Type** | **Default** |
| 1 | TAG | Tag for checking how Class AutoFocusManager works in LogCat when application is running | String | AutoFocusManager.class.getSimpleName(); |
| 2 | AUTO\_FOCUS\_INTERVAL\_MS | The interval in milliseconds to send a focus request for autofocus | long | 3500L |
| 3 | FOCUS\_MODES\_CALLING\_AF | Focus modes setting of camera prepared for calling autofocus | Collection<String> | (Camera.Parameters.FOCUS\_MODE\_AUTO)&(Camera.Parameters.FOCUS\_MODE\_MACRO) |
| 4 | active | Whether autofocus is active or not | boolean | - |
| 5 | manual | Whether manual focus is active or not | boolean | - |
| 6 | useAutoFocus | Whether Use auto focus or not | boolean | - |
| 7 | camera | Camera hardware to be called | Camera | - |
| 8 | timer | Timer | Timer | - |
| 9 | outstandingTask | Outstanding task for timer | TimerTask | - |

**Methods:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Method name** | **Description** | **Parameter Lists** | **Return Type** |
| 1 | AutoFocusManager | Manage autofocus, set useAutoFocus value | context : Context, camera : Camera | void |
| 2 | onAutoFocus | Autofocus is active, timer is set | success : boolean, theCamera : Camera | void |
| 3 | checkAndStart | Check and call start() autofocus | - | void |
| 4 | start | Start the autofocus of the camera | - | void |
| 5 | start | Start manual auto focus after the given delay | delay : long | void |
| 6 | stop | Stop the auto focus | - | void |

**3.2.1.2 Class-02 CameraConfigurationManager**



**Description:**

This class is used for managing the camera configuration by reading, parsing and setting the camera parameters.

**Reference:**

The code for this class is adapted from a ZXing project.[1]

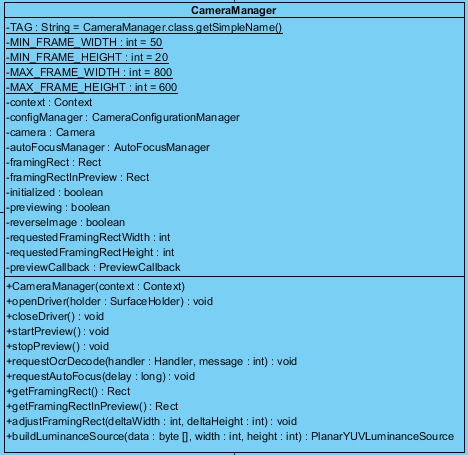
**Attributes:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Name** | **Description** | **Type** | **Default** |
| 1 | TAG | Tag for checking how Class CameraConfigurationManager works in LogCat when application is running | String | “CameraConfiguration” |
| 2 | MIN\_PREVIEW\_PIXELS | Pixels for normal screen | int | 470\*320 |
| 3 | MAX\_PREVIEW\_PIXELS | Pixels for more than large/HD screen | int | 800\*600 |
| 4 | context | Context instance | Context | - |
| 5 | screenResolution | Resolution of the screen | Point | - |
| 6 | cameraResolution | Resolution of the camera | Point | - |

**Methods:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Method name** | **Description** | **Parameter Lists** | **Return Type** |
| 1 | CameraConfigurationManager | Constructor to set context | context : Context | void |
| 2 | initFromCameraParameters | Read parameters of camera which is needed by application | camera : Camera | void |
| 3 | setDesiredCameraParameters | Set desired parameters for camera | camera: Camera | void |
| 4 | getCameraResolution | Get the resolution of the camera | - | Point |
| 5 | getScreenResolution | Get the resolution of the screen | - | Point |
| 6 | findBestPreviewSizeValue | Get the supported preview sizes for the camera screen, calculate the difference between the preview aspect ratio and the screen aspect ratio, the smallest difference is the best preview size. Return the best size value. | parameters : Camera.Parameters, screenResolution : android.graphics.Point | Point |
| 7 | findSettableValue | Find the desired values from the collection of supported values and return it. | supportedValues : Collection<String>, desiredValues : String ... | String |

**3.2.1.3 Class-03 CameraManager**



**Description:**

This class wraps Camera services, encapsulates the steps needed to capture the preview-sized images, which are used for both previewing and decoding.

**Reference:**

The code for this class is adapted from a ZXing project.[1]

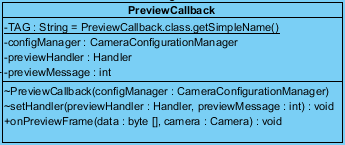
**Attributes:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Name** | **Description** | **Type** | **Default** |
| 1 | TAG | Tag for checking how Class CameraManager works in LogCat when application is running | String | CameraManager.class.getSimpleName(); |
| 2 | MIN\_FRAME\_WIDTH | The minimum width of the preview frame | int | 50 |
| 3 | MIN\_FRAME\_HEIGHT | The minimum height of the preview frame | int | 20 |
| 4 | MAX\_FRAME\_WIDTH | The maximum width of the preview frame | int | 800 |
| 5 | MAX\_FRAME\_HEIGHT | The maximum height of the preview frame | int | 600 |
| 6 | context | Context instance | Context | - |
| 7 | configManager | A CameraConfigurationManager object to manage the camera configuration | CameraConfigurationManager | - |
| 8 | camera | Android hardware camera | Camera | - |
| 9 | autoFocusManager | A AutoFocusManager object to manage the autofocus activity | AutoFocusManager | - |
| 10 | framingRect | Rectangle framing drawn for scanning words | Rect | - |
| 11 | framingRectInPreview | Rectangle framing in the preview | Rect | - |
| 12 | initialized | Whether initialized or not | boolean | - |
| 13 | previewing | Whether previewing or not | boolean | - |
| 14 | reverseImage | Whether reverse image or not | boolean | - |
| 15 | requestedFramingRectWidth | The width of requested rectangle framing for changing the framing | int | - |
| 16 | requestedFramingRectHeight | The height of requested rectangle framing for changing the framing | int | - |
| 17 | previewCallback | A PreviewCallback object to handle the preview callback activity | PreviewCallback | - |

**Methods:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Method name** | **Description** | **Parameter Lists** | **Return Type** |
| 1 | CameraManager | Constructor to set context, configManager and previewCallback | context : Context | void |
| 2 | openDriver | Open the camera driver and initialize the hardware parameters, set the holder to be the surface displaying the preview frame | holder : android.view.SurfaceHolder | void |
| 3 | closeDriver | Close the camera driver when it is still in use, clear the framingRect and framingRectInpreview data | - | void |
| 4 | startPreview | Ask hardware camera to draw the preview frames on the screen when camera works without previewing | - | void |
| 5 | stopPreview | Ask hardware camera to stop drawing the preview frames on screen | - | void |
| 6 | requestOcrDecode | The preview frame will be sent to the handler, the data will arrive in message.obj with width and height encoded. | handler : android.os.Handler, message : int | void |
| 7 | requestAutoFocus | Ask hardware camera to perform the auto focus after the delay time. | delay : long | void |
| 8 | getFramingRect | Calculate the Rectangle framing which should be display on the screen for scanning the words. | - | Rect |
| 9 | getFramingRectInPreview | Calculate the Rectangle framing which should be on the preview. | - | Rect |
| 10 | adjustFramingRect | Adjust the framingRect on the screen with the delta width referring to pixels change for width and delta height referring to pixels change for height. | deltaWidth : int, deltaHeight : int | void |
| 11 | buildLuminanceSource | Build the appropriate LuminanceSource object based on the format of the preview buffers. Data refers to the preview frame. Width and height are width and height of the image. | data : byte [], width : int, height : int | PlanarYUVLuminanceSource |

**3.2.1.4 Class-04 PreviewCallback**



**Description:**

This class implements Camera.PreviewCallback, is invoked when the next preview frame is received.

**Reference:**

The code for this class is adapted from a ZXing project.[1]

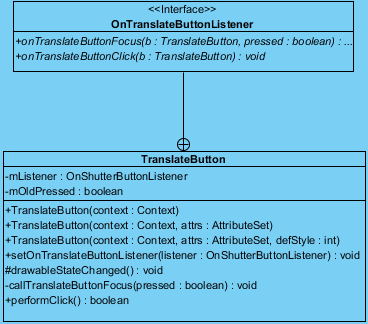
**Attributes:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Name** | **Description** | **Type** | **Default** |
| 1 | TAG | Tag for checking how Class PreviewCallback works in LogCat when application is running | String | PreviewCallback.class.getSimpleName(); |
| 2 | configManager | A CameraConfigurationManager object to manage the camera configuration | CameraConfigurationManager | - |
| 3 | previewHandler | Preview handler | android.os.Handler | - |
| 4 | previewMessage | Preview message object | int | - |

**Methods:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Method name** | **Description** | **Parameter Lists** | **Return Type** |
| 1 | PreviewCallback | Constructor to set configManager | configManager : CameraConfigurationManager | void |
| 2 | setHandler | Set the previewHandler and previewMessage | previewHandler : android.os.Handler, previewMessage : int | void |
| 3 | onPreviewFrame | Send the preview frame data to handler through Message | data : byte [], camera : Camera | void |

**3.2.1.5 Class-05 TranslateButton**



**Description:**

This class extends ImageView, designs a button on-screen to take preview-sized image.

**Reference:**

The code for this class is adapted from a ZXing project.[1]

**Attributes:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Name** | **Description** | **Type** | **Default** |
| 1 | mListener | A listener for Button activity | OnTranslateButtonListener | - |
| 2 | mOldPressed | Whether the translate button is pressed or not | boolean | - |

**Methods:**

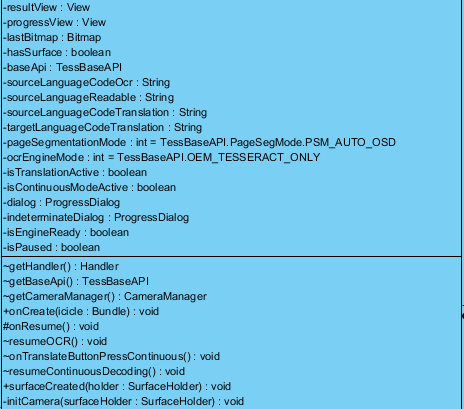
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Method name** | **Description** | **Parameter Lists** | **Return Type** |
| 1 | TranslaterButton | Constructor to set context | context : Context | void |
| 2 | TranslateButton | Constructor to set context and attributes set | context : Context, attrs : AttributeSet | void |
| 3 | TranslateButton | Constructor to set context, attributes set, and defStyle | context : Context, attrs : AttributeSet, defStyle : int | void |
| 4 | setOnTranslateButtonListener | Set the OnTranslateButtonListener | listener : OnTranslateButtonListener | void |
| 5 | drawableStateChanged | Change the state of the translater button in drawable | - | void |
| 6 | callTranslateButtonFocus | Press translate button, call focus | pressed : boolean | void |
| 7 | performClick | Override the performClick, set the onTranslateButtonClick listener | - | boolean |

**3.2.2 Package ocr**

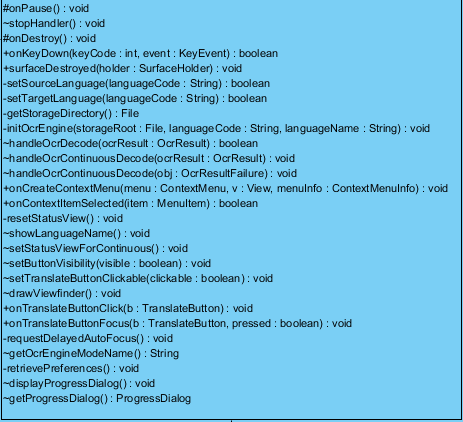
**3.2.2.1 Class-06 CaptureActivity**



***Diagram continues on the next page***



***Diagram continues on the next page***



**Description:**

This class extends Activity, implements SurfaceHolder.Callback, and TranslateButton.OnTranslateButtonListener, opens the camera and does the scanning in the background. It draws the interface on the screen, shows feedback when processing the scanning, and then displays the result when a scan is successful.

**Reference:**

The code for this class is adapted from a ZXing project.[1]

This class imports com.googlecode.tesseract.android.TessBaseAPI, uses the OCR library tess-two:.[2]

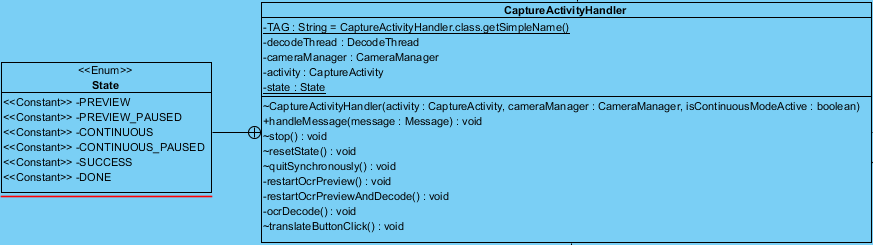
**Attributes:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Name** | **Description** | **Type** | **Default** |
| 1 | TAG | Tag for checking how Class CaptureActivity works in LogCat when application is running | String | CaptureActivity.class.getSimpleName() |
| 2 | DEFAULT\_SOURCE\_LANGUAGE\_CODE | ISO 639-3 language code indicating the default recognition language | String | "ch" |
| 3 | DEFAULT\_TARGET\_LANGUAGE\_CODE | ISO 639-3 language code indicating the default translation language | String | "eng" |
| 4 | DEFAULT\_TRANSLATOR | Default machine translation online service for getting translation | String | "Bing Translate" |
| 5 | DEFAULT\_OCR\_ENGINE\_MODE | Default OCR engine to use | String | "Tesseract" |
| 6 | DEFAULT\_PAGE\_SEGMENTATION\_MODE | Default page segmentation mode to use | String | "Auto" |
| 7 | DEFAULT\_TOGGLE\_AUTO\_FOCUS | Whether set auto focus as default or not | boolean | true |
| 8 | DEFAULT\_TOGGLE\_TRANSLATION | Whether use the online machine translation service by default or not | boolean | true |
| 9 | DEFAULT\_TOGGLE\_LIGHT | Whether activate the light of the hardware camera or not | boolean | false |
| 10 | CONTINUOUS\_DISPLAY\_RECOGNIZED\_TEXT | Flag to display the real-time recognition text on the screen | boolean | true |
| 11 | DISPLAY\_TRANSLATE\_BUTTON | Flag to display the translate button on the screen | boolean | true |
| 12 | DOWNLOAD\_BASE | Tesseract data file downloads resource | String | "http://tesseract-ocr.googlecode.com/files/" |
| 13 | OSD\_FILENAME | Orientation and script detection data downloads filename | String | "tesseract-ocr-3.01.osd.tar" |
| 14 | OSD\_FILENAME\_BASE | Orientation and script detection data destination filename | String | "osd.traineddata" |
| 15 | OPTIONS\_COPY\_RECOGNIZED\_TEXT\_ID | Id of the option in the menu for copying the recognized text | int | Menu.FIRST |
| 16 | OPTIONS\_COPY\_TRANSLATED\_TEXT\_ID | Id of the option in the menu for copying the translated text | int | Menu.FIRST + 1 |
| 17 | cameraManager | CameraManager instance to manage camera | CameraManager | - |
| 18 | handler | CaptureActivityHandler instance for handling the CaptureActivity | CaptureActivityHandler | - |
| 19 | viewFinderView | ViewFinderView instance | ViewFinderView | - |
| 20 | surfaceView | SurfaceView instance | SurfaceView | - |
| 21 | surfaceHolder | SurfaceHolder instance | SurfaceHolder | - |
| 22 | statusViewTop | The status view at the top of the screen | TextView | - |
| 23 | ocrResultView | Editable OCR result view on the screen | EditText | - |
| 24 | translationView | Translation view on the screen | TextView | - |
| 25 | speakButton | Button for obtaining the pronunciation of the selected Chines words | Button | - |
| 26 | backButton | Button for returning to the previous page | Button | - |
| 27 | cameraButtonView | The view of the buttons on the screen with working camera | View | - |
| 28 | resultView | The view of the OCR recognition result on the screen with its English explanation | View | - |
| 29 | progressView | The view on the screen displays it is in the progress | View | - |
| 30 | lastResult | Last OCR recognition result provided by OCR engine | OcrResult | - |
| 31 | lastBitmap | Last bitmap sent to OCR engine | Bitmap | - |
| 32 | hasSurface | Flag to existed surface | boolean | - |
| 33 | baseApi | Java interface for Tesseract OCR engine | TessBaseAPI | - |
| 34 | sourceLanguageCodeOcr | ISO 639-3 language code for source language | String | - |
| 35 | sourceLanguageReadable | Name of the source Language | String | - |
| 36 | sourceLanguageCodeTranslation | ISO 639-1 language code for recognition source language | String | - |
| 37 | targetLanguageCodeTranslation | ISO 639-1 language code for translation target language | String | - |
| 38 | pageSegmentationMode | Page segmentation mode | int | TessBaseAPI.PageSegMode.PSM\_AUTO\_OSD |
| 39 | ocrEngineMode | Ocr engine mode | int | TessBaseAPI.OEM\_TESSERACT\_ONLY |
| 40 | isTranslationActive | Whether display the translation or not | boolean | - |
| 41 | translateButton | Translate button | TranslateButton | - |
| 42 | isContinuousModeActive | Whether OCR is working in continuous mode or not | boolean | - |
| 43 | dialog | Progress dialog for initiating the OCR, downloading and unzipping the language | android.app.ProgressDialog | - |
| 44 | indeterminateDialog | Progress dialog for initiating the OCR, initiating the OCR engine | android.app.ProgressDialog | - |
| 45 | isEngineReady | Whether OCR engine is ready or not | boolean | - |
| 46 | isPaused | Whether pause OCR working or not | boolean | - |

**Methods:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Method name** | **Description** | **Parameter Lists** | **Return Type** |
| 1 | getHandler | Getter of the handler | - | android.os.Handler |
| 2 | getBaseApi | Getter of the BaseApi | - | com.googlecode.tesseract.android.TessBaseAPI |
| 3 | getCameraManager | Getter of the CameraManager | - | CameraManager |
| 4 | onCreate | Override the onCreate( ) from parent Activity class, automatically called when the application runs. When onCreate( ) is called, the viewFinderView, cameraButtonView and resultView are created. Display the Translate Button, set listener to the editable Ocr recognition result in resultView. Translate the recognition result and display the translation on the resultView. Provide the Speak Button for pronunciation and set the listener to it. Set the listener to onTouch operation on the screen from users. | icicle : android.os.Bundle | void |
| 5 | onResume | Override the onResume( ) method from parent class. Reset the status view, initialize the OCR engine if necessary, start the camera and resume OCR. | - | void |
| 6 | resumeOCR | Resume the OCR | - | void |
| 7 | onTranslateButtonPressContinuous | When the Translate button is pressed continuously, OCR will be paused and check whether get the recognition lastResult or not. If not, the error message will be displayed and call resumeContinuousDecoding( ). | - | void |
| 8 | resumeContinuousDecoding | Called to resume recognition after translation in continuous mode. | - | void |
| 9 | surfaceCreated | Initialize the camera only when the OCR engine is ready. | holder : SurfaceHolder | void |
| 10 | initCamera | Open and initialize the camera, start the handler to begin previewing. | surfaceHolder : SurfaceHolder | void |
| 11 | onPause | Stop using the camera. | - | void |
| 12 | stopHandler | Stop the handler. | - | void |
| 13 | onDestroy | End the baseApi. | - | void |
| 14 | onKeyDown | Check each condition when press back operation provided by android mobile phone. When users are in paused condition, just unpause after back operation. When they are in viewing the result, system will return to the scanning view. When they are not in viewing the result, app will be exited. | keyCode : int, event : android.view.KeyEvent | boolean |
| 15 | surfaceDestroyed | Destroy the surface, set the hasSurface to be false. | holder : android.view.SurfaceHolder | void |
| 16 | setSourceLanguage | Set the necessary language code value for the given OCR source language. | languageCode : String | boolean |
| 17 | setTargetLanguage | Set the necessary language code value for the translation target language. | languageCode : String | boolean |
| 18 | getStorageDirectory | Get the proper location on SD card where to save the files. | - | File |
| 19 | initOcrEngine | Request the initialization of OCR engine with the given parameters. The storageRoot is the path to location of the tesseract data directory for use. The languageCode is 3-letter ISO 639-3 language code for OCR. The languageName is the name of the language for OCR. | storageRoot : java.io.File, languageCode : String, languageName : String | void |
| 20 | handleOcrDecode | Display the recognition text of OCR and request for translation. Return true when the result is non-null. | ocrResult : ocr.OcrResult | boolean |
| 21 | handleOcrContinuousDecode | Display the results of a successful real-time OCR request. | ocrResult : ocr.OcrResult | void |
| 22 | handleOcrContinuousDecode | Display the error message when the real-time OCR request fails. | obj : ocr.OcrResultFailure | void |
| 23 | onCreateContextMenu | Set recognized text and translated text copied operation into context menu. | menu : android.view.ContextMenu, v : android.view.View, menuInfo : android.view.ContextMenu.ContextMenuInfo | void |
| 24 | onContextItemSelected | Use clipboardManager to manage the copied text both recognized and translated. | item : android.view.MenuItem | boolean |
| 25 | resetStatusView | Reset the elements of views. | - | void |
| 26 | showLanguageName | Use Toast to display the pop-up message showing the name of the current OCR source language and short help information. | - | void |
| 27 | setStatusViewForContinuous | Remove the result text from viewFinderView. | - | void |
| 28 | setButtonVisibility | Set the visibility of buttons. | visible : boolean | void |
| 29 | setTranslateButtonClickable | Enable or disable translate button to be clickable. | clickable : boolean | void |
| 30 | drawViewfinder | Call drawViewfinder( ) of ViewFinderView. | - | void |
| 31 | onTranslateButtonClick | Set translate button to capture the screen when clicked. | b : TranslateButton | void |
| 32 | onTranslateButtonFocus | Set translate button to focus when clicked. | b : camera.TranslateButton, pressed : boolean | void |
| 33 | requestDelayedAutoFocus | Request an autofocus after a delay. | - | void |
| 34 | getOcrEngineModeName | Get the name of OCR engine mode. | - | String |
| 35 | displayProgressDialog | Set up the indeterminate progress dialog box. | - | void |
| 36 | getProgressDialog | Getter of the progress dialog. | - | android.app.ProgressDialog |
| 37 | showErrorMessage | Display the error message dialog box to users. | title : String, message : String | void |

**3.2.2.2 Class-07 CaptureActivityHandler**



**Description:**

This class extends handler, handles all the messaging which comprises the state machine for capture.

**Reference:**

The code for this class is adapted from a ZXing project.[1]

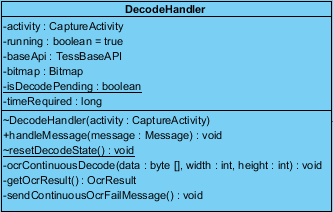
**Attributes:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Name** | **Description** | **Type** | **Default** |
| 1 | TAG | Tag for checking how Class CaptureActivityHandler works in LogCat when application is running | String | CaptureActivityHandler.class.getSimpleName() |
| 2 | decodeThread | Decode thread | DecodeThread | - |
| 3 | cameraManager | CameraManager instance to manage camera | CameraManager | - |
| 4 | activity | CaptureActivity instance | CaptureActivity | - |
| 5 | state | State of the system | State | enum State { PREVIEW,  PREVIEW\_PAUSED, CONTINUOUS,  CONTINUOUS\_PAUSED, SUCCESS, DONE } |

**Methods:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Method name** | **Description** | **Parameter Lists** | **Return Type** |
| 1 | CaptureActivityHandler | Constructor to set activity, cameraManager, and isContinuousModeActive. Start scanning preview and decode when the continuous mode is active. | activity : CaptureActivity, cameraManager : CameraManager, isContinuousModeActive : boolean | void |
| 2 | handleMessage | Handle the message in different cases. | message : Message | void |
| 3 | stop | Set the state to be CONTINUOUS\_PAUSED. | - | void |
| 4 | resetState | When the state is CONTINUOUS\_PAUSED, set it to be CONTINUOUS, and call restartOcrPreviewAndDecode( ). | - | void |
| 5 | quitSynchronously | Set state to be DONE, call stopPreview( ) when camera is working. | - | void |
| 6 | restartOcrPreview | Start the OCR preview but don’t try OCR until users press the translate button. | - | void |
| 7 | restartOcrPreviewAndDecode | Send the decode request for real-time OCR. Continue camera scanning frame and request decoding. | - | void |
| 8 | ocrDecode | Request OCR on the current preview frame. | - | void |
| 9 | translateButtonClick | Disable further click on the translate button until the OCR request is finished. | - | void |

**3.2.2.3 Class-08 DecodeHandler**



**Description:**

This class extends handler, handles OCR decoding by sending bitmap for it.

**Reference:**

The code for this class is adapted from a ZXing project.[1]

This class imports com.googlecode.tesseract.android.TessBaseAPI and com.googlecode.leptonica.android.ReadFile, uses the OCR library tess-two.[2]

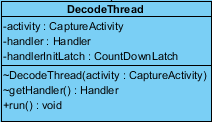
**Attributes:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Name** | **Description** | **Type** | **Default** |
| 1 | activity | CaptureActivity instance | CaptureActivity | - |
| 2 | running | Flag to the OCR decode is running | boolean | true |
| 3 | baseApi | Java interface for Tesseract OCR engine | com.googlecode.tesseract.android.TessBaseAPI | - |
| 4 | bitmap | bitmap | Bitmap | - |
| 5 | isDecodePending | Flag to the Decode is pending | boolean | - |
| 6 | timeRequired | Recognition time required | long | - |

**Methods:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Method name** | **Description** | **Parameter Lists** | **Return Type** |
| 1 | DecodeHandler | Constructor to set the activity and baseApi | activity : CaptureActivity | void |
| 2 | handleMessage | Handle message in different cases. | message : Message | void |
| 3 | resetDecodeState | Set isDecodePending to be false. | - | void |
| 4 | ocrContinuousDecode | Perform OCR decode for real-time recognition. | data : byte [], width : int, height : int | void |
| 5 | getOcrResult | OCR engine recognize the text from bitmap, get the OcrResult. | - | OcrResult |
| 6 | sendContinuousOcrFailMessage | Send the message when ocr\_continuous\_decode\_failed | - | void |

**3.2.2.4 Class-09 DecodeThread**



**Description:**

This class (thread) does all the heavy task of decoding the images.

**Reference:**

The code for this class is adapted from a ZXing project.[1]

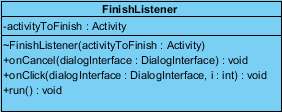
**Attributes:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Name** | **Description** | **Type** | **Default** |
| 1 | activity | CaptureActivity instance | CaptureActivity | - |
| 2 | handler | Handler instance | Handler | - |
| 3 | handlerInitLatch | CountDownLatch instance to handle the thread | CountDownLatch | - |

**Methods:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Method name** | **Description** | **Parameter Lists** | **Return Type** |
| 1 | DecodeThread | Constructor to set the activity and handlerInitLatch | activity : CaptureActivity | void |
| 2 | getHandler | Getter of the handler | - | Handler |
| 3 | run | Run the decode thread. | - | void |

**3.2.2.5 Class-10 FinishListener**



**Description:**

This class abstracts different bitmap implementations across platforms into a standard interface for requesting greyscale luminance values.

**Reference:**

The code for this class is adapted from a ZXing project.[1]

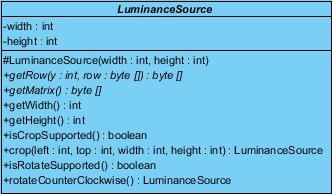
**Attributes:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Name** | **Description** | **Type** | **Default** |
| 1 | activityToFinish | Activity instance of activity to finish | Activity | - |

**Methods:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Method name** | **Description** | **Parameter Lists** | **Return Type** |
| 1 | FinishListener | Constructor to set the activityToFinish. | activityToFinish : Activity | void |
| 2 | onCancel | Invoked when dialog is canceled. | dialogInterface : DialogInterface | void |
| 3 | onClick | Invoked when the button in the dialog is pressed. | dialogInterface : DialogInterface, i : int) | void |
| 4 | run | Start executing the FinishListener. | - | void |

**3.2.2.6 Class-11 LuminanceSource**



**Description:**

This class abstracts different bitmap implementations across platforms into a standard interface for requesting greyscale luminance values.

**Reference:**

The code for this class is adapted from a ZXing project.[1]

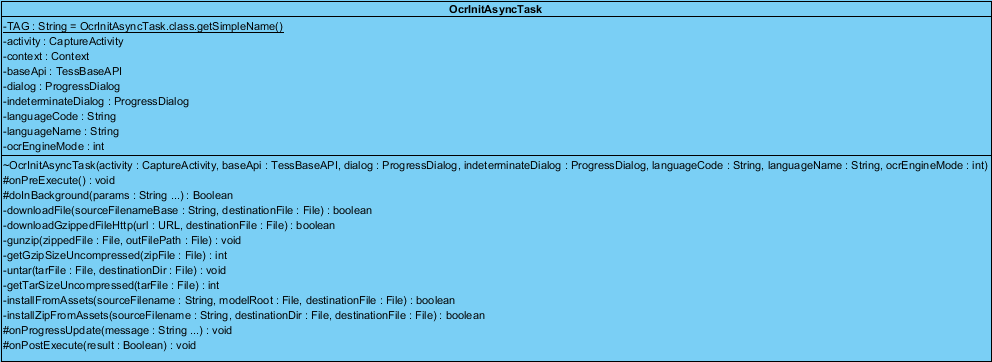
**Attributes:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Name** | **Description** | **Type** | **Default** |
| 1 | width | Width of bitmap | int | - |
| 2 | height | Height of bitmap | int | - |

**Methods:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Method name** | **Description** | **Parameter Lists** | **Return Type** |
| 1 | LuminanceSource | Constructor to set the width and the height | width : int, height : int | - |
| 2 | getRow | Abstract method for getting one row of luminance data from the underlying platform's bitmap. Values range from 0 (black) to 255 (white), return an array containing the luminance data. | y : int, row : byte []) | byte [] |
| 3 | getMatrix | Abstract method for getting the matrix of luminance data from the underlying platform’s bitmap. Values should be as int luminance = array[y \* width + x] & 0xff, return a row-major 2D array of luminance values. | - | byte [] |
| 4 | getWidth | Getter of width. | - | int |
| 5 | getHeight | Getter of height. | - | int |
| 6 | isCropSupported | Whether subclass supports cropping.  Return true. | - | boolean |
| 7 | crop | It is callable only when isCropSupported( ) is true, returns a new object with cropped image data. | left : int, top : int, width : int, height : int | LuminanceSource |
| 8 | isRotateSupported | Whether subclass supports counter-clockwise rotation. Return false. | - | boolean |
| 9 | rotateCounterClockwise | It is callable only when isRotateSupported ( ) is true, returns a new object with rotated image data. | - | LuminanceSource |

**3.2.2.7 Class-12 OcrInitAsyncTask**



**Description:**

This class extends AsyncTask<String, String, Boolean>, installs language data required for OCR, and initializes the OCR engine.

**Reference:**

The code for this class is adapted from a ZXing project.[1]

This class imports com.googlecode.tesseract.android.TessBaseAPI and uses the OCR library tess-two.[2]

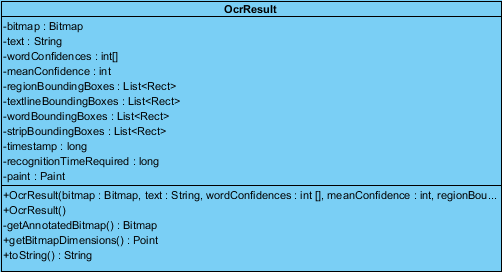
**Attributes:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Name** | **Description** | **Type** | **Default** |
| 1 | TAG | Tag for checking how Class OcrInitAsyncTask works in LogCat when application is running | String | OcrInitAsyncTask.class.getSimpleName() |
| 2 | activity | CaptureActivity instance | CaptureActivity | - |
| 3 | context | Context instance | Context | - |
| 4 | baseApi | Java interface for Tesseract OCR engine | com.googlecode.tesseract.android.TessBaseAPI | - |
| 5 | dialog | Progress dialog for initiating the OCR, downloading and unzipping the language. | ProgressDialog | - |
| 6 | indeterminateDialog | Progress dialog for initiating the OCR, initiating the OCR engine. | ProgressDialog | - |
| 7 | languageCode | ISO 639-2 OCR language code | String | - |
| 8 | languageName | Name of the OCR language | String | - |
| 9 | ocrEngineMode | OCR engine mode | int | - |

**Methods:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Method name** | **Description** | **Parameter Lists** | **Return Type** |
| 1 | OcrInitAsyncTask | Constructor to set activity, baseApi, dialog, indeterminateDialog, languageCode, languageName, and ocrEngineMode. | activity : CaptureActivity, baseApi : com.googlecode.tesseract.android.TessBaseAPI, dialog : ProgressDialog, indeterminateDialog : ProgressDialog, languageCode : String, languageName : String, ocrEngineMode : int | void |
| 2 | onPreExecute | Override the onPreExecute( ), set the value of parameters for dialog, and set button visibility to be gone. | - | void |
| 3 | doInBackground | In background thread, perform required setup, and request initialization of the OCR engine. | params : String … | boolean |
| 4 | downloadFile | Download a file from the site specified by DOWNLOAD\_BASE, and unzip it to the given destination. Return true if the download and unzip are successful. | sourceFilenameBase : String, destinationFile : File | boolean |
| 5 | downloadGzippedFileHttp | Download a gzipped file using an HttpURLConnection, and unzip it to the given destination. Return true if response is received, destinationFile is opened, and unzip is successful. | url : URL, destinationFile : File) | boolean |
| 6 | gunzip | Unzip the given gzipped file to the given destination and delete the gzipped file. | zippedFile : File, outFilePath : File | void |
| 7 | getGzipSizeUncompressed | Get the size of the uncompressed Gzipped file. | zipFile : File | Int |
| 8 | untar | Untar the contents of a tar file into the given directory, and delete the tar file. | tarFile : File, destinationDir : File | void |
| 9 | getTarSizeUncompressed | Get the size of uncompressed tar file. | tarFile : File | Int |
| 10 | installFromAssets | Install a file from application assets to device external storage, return true if installZipFromAssets returns true. | sourceFilename : String, modelRoot : File, destinationFile : File | boolean |
| 11 | installZipFromAssets | Unzip the given Zip file which is located in application assets into the given destination file, return true if the unzip is successful. | sourceFilename : String, destinationDir : File, destinationFile : File | boolean |
| 12 | onProgressUpdate | Override onProgressUpdate( ) and update the dialog with the latest incremental progress. | message : String ... | void |
| 13 | onPostExecute | Override onPostExecute( ), run after doInBackground( ). | Result : boolean | void |

**3.2.2.8 Class-13 OcrResult**



**Description:**

This class encapsulates the result of OCR.

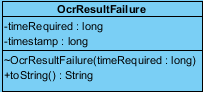
**Attributes:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Name** | **Description** | **Type** | **Default** |
| 1 | bitmap | Bitmap sent for recognition | Bitmap | - |
| 2 | text | Text stored the OCR recognition | String | - |
| 3 | wordConfidences | Word confidences | int[] | - |
| 4 | meanConfidence | Mean confidence | int | - |
| 5 | regionBoundingBoxes | Region bounding boxes | List< Rect> | - |
| 6 | textlineBoundingBoxes | Text line bounding boxes | List< Rect> | - |
| 7 | wordBoundingBoxes | Word bounding boxes | List< Rect> | - |
| 8 | stripBoundingBoxes | Strip bounding boxes | List< Rect> | - |
| 9 | timestamp | timestamp | long | - |
| 10 | recognitionTimeRequired | Recognition time required | long | - |
| 11 | paint | Paint instance for drawing on the screen | Paint | - |

**Methods:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Method name** | **Description** | **Parameter Lists** | **Return Type** |
| 1 | OcrResult | Constructor to set up bitmap, wordConfidences, meanConfidence, regionBoundingBoxes, textlineBoundingBoxes, wordBoundingBoxe, stripBoundingBoxes, recognitionTimeRequired and pait. | bitmap : Bitmap, text : String, wordConfidences : int [], meanConfidence : int, regionBoundingBoxes : List< Rect>, textlineBoundingBoxes : List< Rect>, wordBoundingBoxes : List< Rect>, stripBoundingBoxes : List< Rect>, recognitionTimeRequired : long | void |
| 2 | OcrResult | Constructor to set up timestamp and paint. | - | void |
| 3 | getAnnotatedBitmap | Get the annotated bitmap. Draw bounding boxes around each word. | - | Bitmap |
| 4 | getBitmapDimensions | Get the dimensions of the bitmap. | - | Point |
| 5 | toString | Override toString( ) , return the string containing text, meanConfidence, recognitionTimeRequired and timestamp. | - | String |

**3.2.2.9 Class-14 OcrResultFailure**



**Description:**

This class wraps the metadata for failed OCR results.

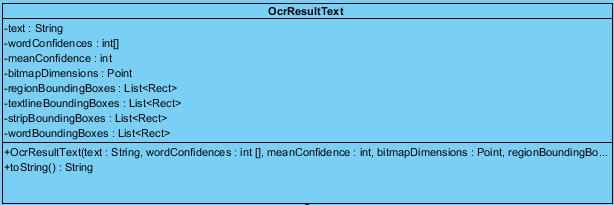
**Attributes:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Name** | **Description** | **Type** | **Default** |
| 1 | timeRequired | Time required | long | - |
| 2 | timestamp | timestamp | long | - |

**Methods:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Method name** | **Description** | **Parameter Lists** | **Return Type** |
| 1 | OcrResultFailure | Constructor to set timeRequired and timestamp. | timeRequired : long | void |
| 2 | toString | Override toString ( ), return the string containing timeRequired and timestamp. | String | String |

**3.2.2.10 Class-15 OcrResultText**



**Description:**

This class encapsulates result text and its words coordinates resulting from OCR.

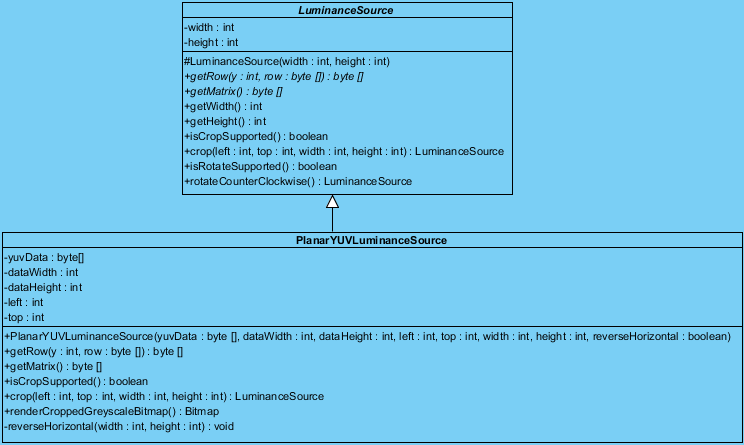
**Attributes:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Name** | **Description** | **Type** | **Default** |
| 1 | text | Text stored the OCR recognition | String | - |
| 2 | wordConfidences | Word confidences | int[] | - |
| 3 | meanConfidence | Mean confidence | int | - |
| 4 | bitmapDimensions | Bitmap dimensions | Point | - |
| 5 | regionBoundingBoxes | Region bounding boxes | List< Rect> | - |
| 6 | textlineBoundingBoxes | Text line bounding boxes | List< Rect> | - |
| 7 | wordBoundingBoxes | Word bounding boxes | List< Rect> | - |
| 8 | stripBoundingBoxes | Strip bounding boxes | List< Rect> | - |

**Methods:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Method name** | **Description** | **Parameter Lists** | **Return Type** |
| 1 | OcrResultText | Constructor to set up text, wordConfidences, meanConfidence, bitmapDimensions, regionBoundingBoxes, textlineBoundingBoxes, wordBoundingBoxe, and stripBoundingBoxes. | text : String, wordConfidences : int [], meanConfidence : int, bitmapDimensions : Point, regionBoundingBoxes : List< Rect>, textlineBoundingBoxes : List< Rect>, stripBoundingBoxes : List< Rect>, wordBoundingBoxes : List< Rect> | void |
| 2 | toString | Override toString( ), return string containing text and meanConfidence. | - | void |

**3.2.2.11 Class-16 PlanarYUVLuminanceSource**



**Description:**

This class extends LuminanceSource, crops to a rectangle within YUV data returned from the camera driver.

**Reference:**

The code for this class is adapted from a ZXing project.[1]

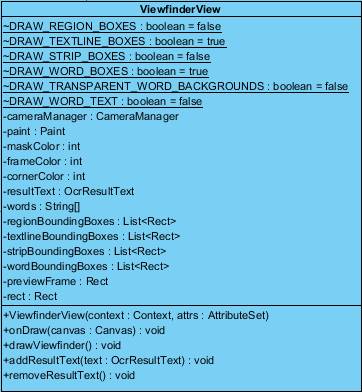
**Attributes:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Name** | **Description** | **Type** | **Default** |
| 1 | yuvData | YUV data returned from camera driver | byte[] | - |
| 2 | dataWidth | Width of the data | int | - |
| 3 | dataHeight | Height of the data | int | - |
| 4 | left | Margin left | int | - |
| 5 | top | Margin top | int | - |

**Methods:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Method name** | **Description** | **Parameter Lists** | **Return Type** |
| 1 | PlanarYUVLuminanceSource | Constructor to set yuvData, dataWidth, dataHeight, left, top, and reverseHorizontal. | yuvData : byte [], dataWidth : int, dataHeight : int, left : int, top : int, width : int, height : int, reverseHorizontal : boolean | void |
| 2 | getRow | Override getRow ( ), getting one row of luminance data from the underlying platform's bitmap. | y : int, row : byte []) | byte [] |
| 3 | getMatrix | Override getMatrix ( ), getting the matrix of luminance data from the underlying platform’s bitmap. | - | byte [] |
| 4 | isCropSupported | Override isCropSupported( ), return true. | - | boolean |
| 5 | crop | Override crop( ), crop the bitmap as the setting parameters. | left : int, top : int, width : int, height : int | LuminanceSource |
| 6 | renderCroppedGreyscaleBitmap | Render cropped greyscale bitmap. | - | Bitmap |
| 7 | reverseHorizontal | Reverse the horizontal way. | width : int, height : int | void |

**3.2.2.12 Class-17 ViewFinderView**



**Description:**

This class extends View, adds the viewfinder rectangle and partial transparency outside of it, as well as the result text.

**Reference:**

The code for this class is adapted from a ZXing project.[1]

**Attributes:**

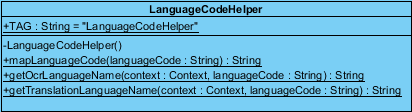
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Name** | **Description** | **Type** | **Default** |
| 1 | DRAW\_REGION\_BOXES | Flag to draw boxes representing the results from TessBaseAPI GetRegions(). | boolean | false |
| 2 | DRAW\_TEXTLINE\_BOXES | Flag to draw boxes representing the results from TessBaseAPI GetTextlines(). | boolean | true |
| 3 | DRAW\_STRIP\_BOXES | Flag to draw boxes representing the results from TessBaseAPI GetStrips(). | boolean | false |
| 4 | DRAW\_WORD\_BOXES | Flag to draw boxes representing the results from TessBaseAPI GetWords (). | boolean | true |
| 5 | DRAW\_TRANSPARENT\_WORD\_BACKGROUNDS | Flag to draw word text with a background varying from transparent to opaque. | boolean | false |
| 6 | DRAW\_WORD\_TEXT | Flag to draw the text of words within their respective boxes from TessBaseAPI GetWords(). | boolean | False |
| 7 | cameraManager | CameraManager instance to manage the camera | CameraManager | - |
| 8 | paint | Paint instance to draw | Paint | - |
| 9 | maskColor | Color of the outside rectangle framing | int | - |
| 10 | frameColor | Color of rectangle framing | int | - |
| 11 | cornerColor | Color of the corner of the rectangle framing | int | - |
| 12 | resultText | Ocr result text | OcrResultText | - |
| 13 | words | Separated words from resultText | String[] | - |
| 14 | regionBoundingBoxes | Region bounding boxes | List<Rect> | - |
| 15 | textlineBoundingBoxes | Text line bounding boxes | List<Rect> | - |
| 16 | stripBoundingBoxes | Strip bounding boxes | List<Rect> | - |
| 17 | wordBoundingBoxes | Word bounding boxes | List<Rect> | - |
| 18 | previewFrame | Rectangle frame for previewing | Rect | - |
| 19 | rect | Rectangle | Rect | - |

**Methods:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Method name** | **Description** | **Parameter Lists** | **Return Type** |
| 1 | ViewfinderView | Constructor to set the context and attributes set when the class is built from XML resource. | context : Context, attrs : AttributeSet | void |
| 2 | onDraw | Override onDraw(), draw the scanning interface on the screen. | canvas : Canvas | void |
| 3 | drawViewfinder | Call invalidate() to invalidate the whole view. | - | void |
| 4 | addResultText | Add the given OCR results for drawing to the view. | text : OcrResultText | void |
| 5 | removeResultText | Remove the ocr resultText from view at next onDraw(). | - | void |

**3.2.3 Package translator**

**3.2.3.1 Class-18 LanguageCodeHelper**



**Description:**

This class handles functions relating to converting between standard language codes, and converting language codes to language names.

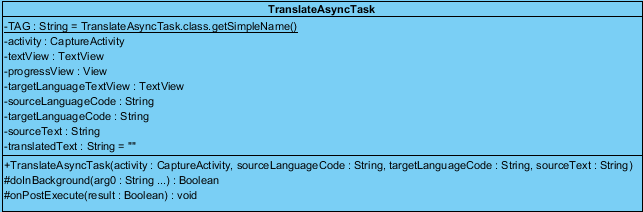
**Attributes:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Name** | **Description** | **Type** | **Default** |
| 1 | TAG | Tag for checking how Class LanguageCodeHelper works in LogCat when application is running | String | "LanguageCodeHelper" |

**Methods:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Method name** | **Description** | **Parameter Lists** | **Return Type** |
| 1 | LanguageCodeHelper | Private constructor | - | void |
| 2 | mapLanguageCode | Map the language and its code | languageCode : String | String |
| 3 | getOcrLanguageName | Map the given ISO 639-3 language code to a name of a language, return language name. | context : Context, languageCode : String | String |
| 4 | getTranslationLanguageName | Map the given ISO 639-1 language code to a name of a language, return the language name. | context : Context, languageCode : String | String |

**3.2.3.2 Class-19 TranslateAsyncTask**



**Description:**

This class extends AsyncTask<String, String, Boolean>, and does translation in the background.

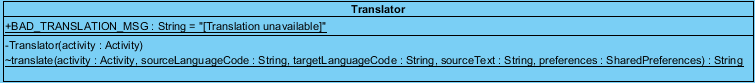
**Attributes:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Name** | **Description** | **Type** | **Default** |
| 1 | TAG | Tag for checking how Class TranslateAsyncTask works in LogCat when application is running | String | TranslateAsyncTask.class.getSimpleName() |
| 2 | activity | CaptureActivity instance | CaptureActivity | - |
| 3 | textView | Text view | TextView | - |
| 4 | progressView | Progress view | View | - |
| 5 | targetLanguageTextView | Target language text view on the screen | TextView | - |
| 6 | sourceLanguageCode | Source language text view on the screen | String | - |
| 7 | targetLanguageCode | The code of target language | String | - |
| 8 | sourceText | Source text for translation | String | - |
| 9 | translatedText | Translated text | String | “” |

**Methods:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Method name** | **Description** | **Parameter Lists** | **Return Type** |
| 1 | TranslateAsyncTask | Constructor to set activity, sourceLanguageCode, targetLanguageCode, sourceText, textView and progressView. | activity : CaptureActivity, sourceLanguageCode : String, targetLanguageCode : String, sourceText : String | void |
| 2 | doInBackground | Call the Translator.translate() in background. | arg0 : String ... | boolean |
| 3 | onPostExecute | Check whether there is a result, if true, display the translated text. | result : Boolean | void |

**3.2.3.3 Class-20 Translator**



**Description:**

This class sends translation request to translation service.

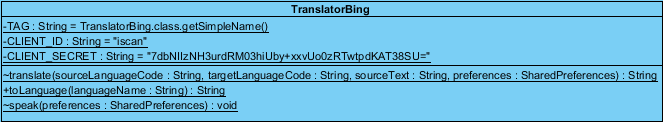
**Attributes:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Name** | **Description** | **Type** | **Default** |
| 1 | BAD\_TRANSLATION\_MSG | Bad translation message displayed when translation is unavailable | String | "[Translation unavailable]" |

**Methods:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Method name** | **Description** | **Parameter Lists** | **Return Type** |
| 1 | Translator | Constructor to set activity | activity : Activity | void |
| 2 | translate | Return to call TranslatorBing.translate get the translation. | activity : android.app.Activity, sourceLanguageCode : String, targetLanguageCode : String, sourceText : String, preferences : SharedPreferences | String |

**3.2.3.4 Class-21 TranslatorBing**



**Description:**

This class performs Microsoft Translator online machine translation service.

**Attributes:**

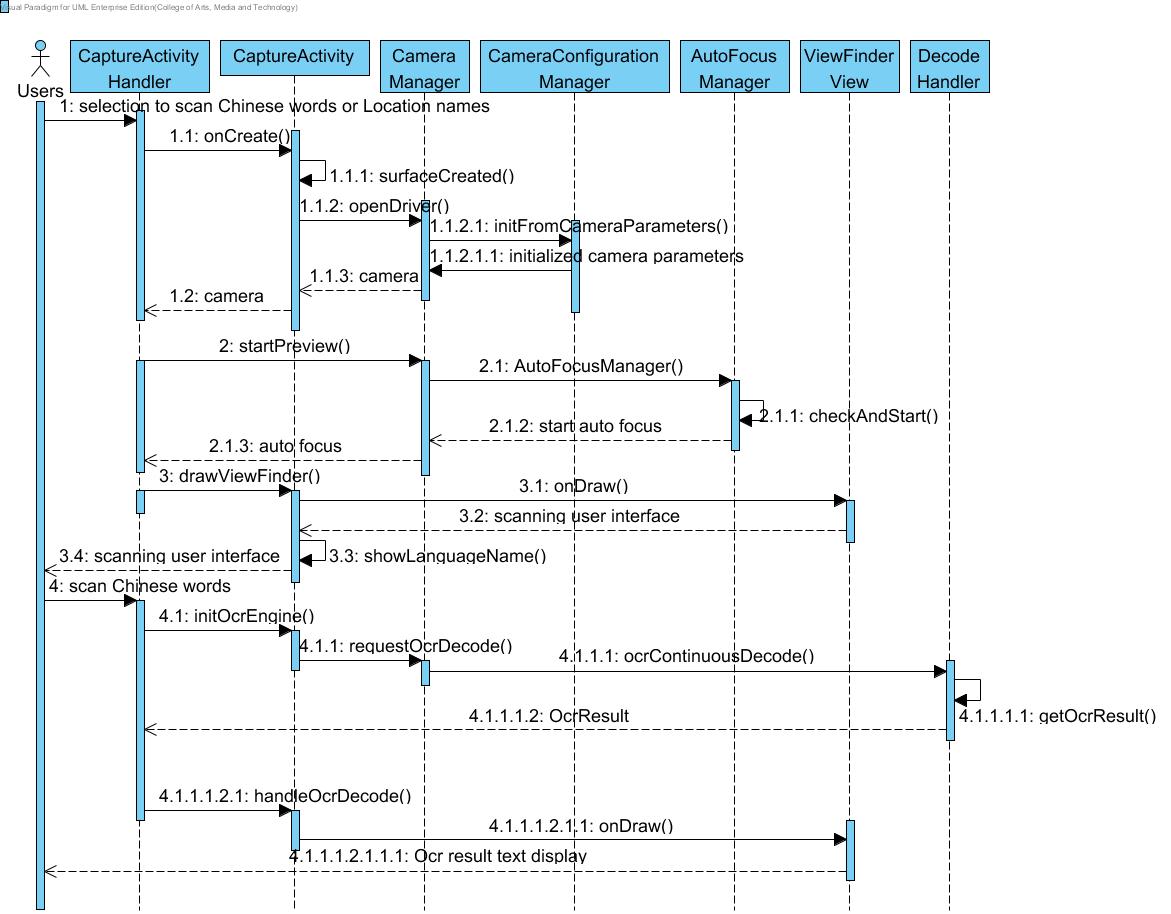
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Name** | **Description** | **Type** | **Default** |
| 1 | TAG | Tag for checking how Class TranslatorBing works in LogCat when application is running | String | TranslatorBing.class.getSimpleName() |
| 2 | CLIENT\_ID | Client id to apply for the Microsoft Translator service | String | “ScanInTravel” |
| 3 | CLIENT\_SECRET | Client secret to apply for the Microsoft Translator servcie | String | “Sp3SnVg1CmCDbXP1VkEhSP3vbm6G/XWv/UPHZiWMDbw=” |

**Methods:**

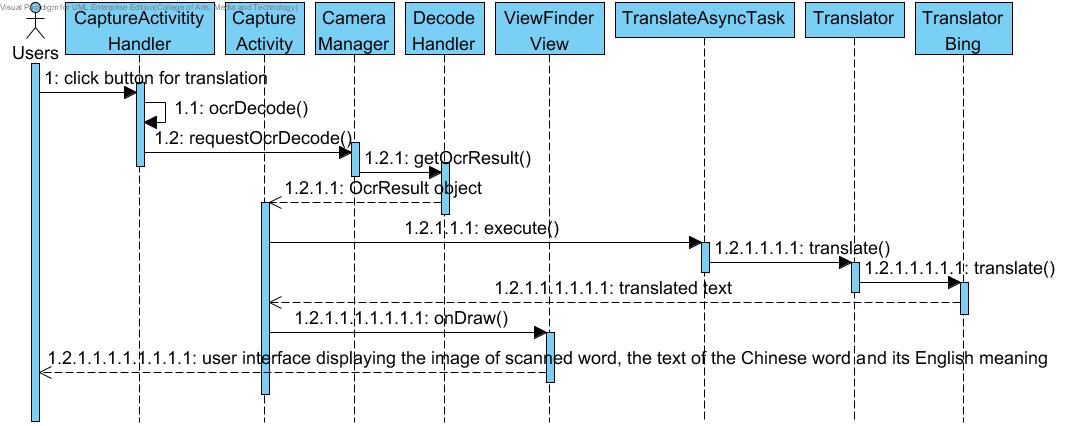
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Method name** | **Description** | **Parameter Lists** | **Return Type** |
| 1 | translate | Translate using Microsoft Translator API and get the pronunciation of source text also. | sourceLanguageCode : String, targetLanguageCode : String, sourceText : String, preferences : SharedPreferences | String |
| 2 | toLanguage | Convert the given name of a natural language into a language from the enum of Languages supported by Microsoft Translator translation service. | languageName : String | String |
| 3 | speak | Set the media player to play the pronunciation of source text. | preferences : SharedPreferences | void |

**Chapter Four | Sequence Diagram**

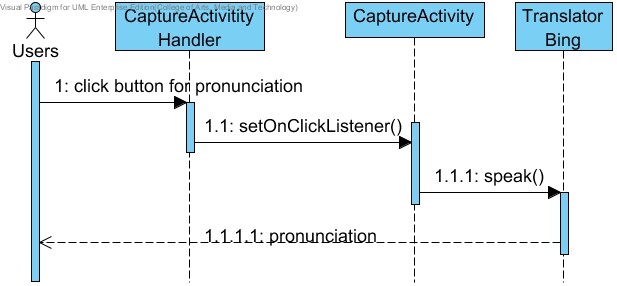
**SD-01: Use Case(UC-01): Users can scan Chinese words and Location names by using built-in camera.**



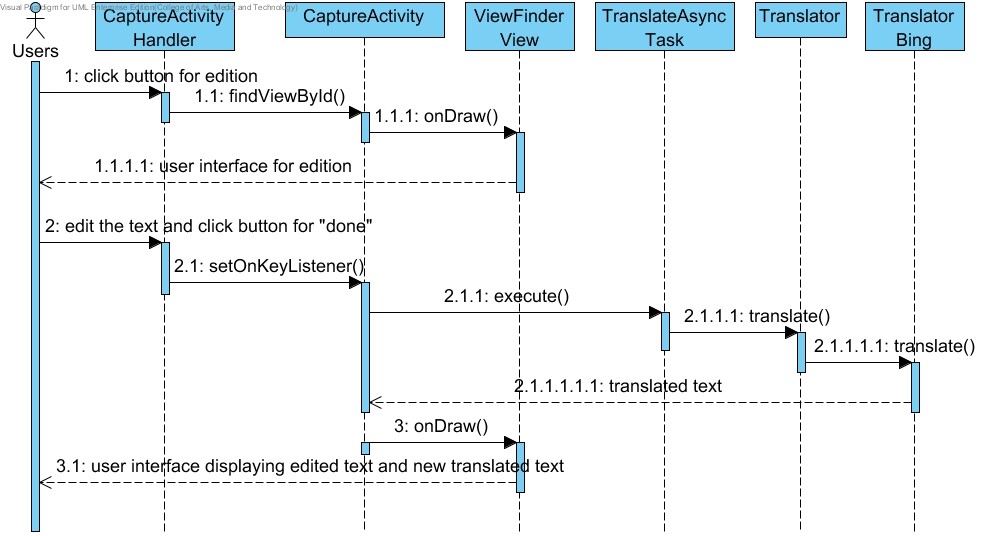
**SD-02: Use Case(UC-02): Users can view the explanation (i.e. meaning, pronunciation) of the scanned word(s) in English.**



**SD-03: Use Case(UC-03):**  **Users can select Text-to-Speech to get the Chinese pronunciation of the word.**



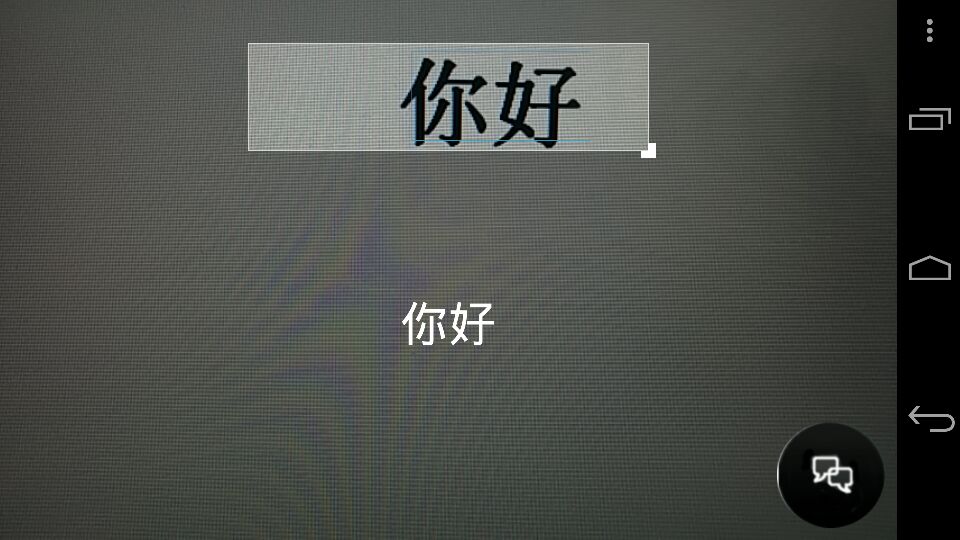
**SD-04: Use Case(UC-04): Users can edit the Chinese word(s)(or Location name(s)) that are scanned to get the new explanation in English.**



**Chapter Five | User Interface Design**

**UI-01:**

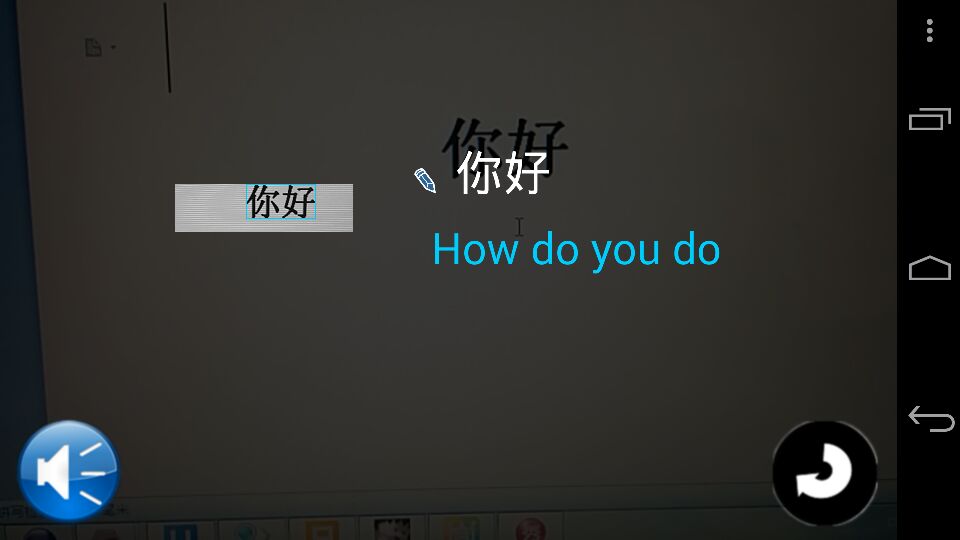
**Page Name: Scan page**



**Description:** This page is for users to scan the Chinese word(s) with the built-in camera of their mobile phone. Users can adjust the size of the scan box horizontally to scan the text that they want. Below the scan box, users can see the word(s) detected by the OCR library. And users can click “translate” button in the lower right corner of the screen to get the meaning of the scanned word(s) in English.

**UI-02:**

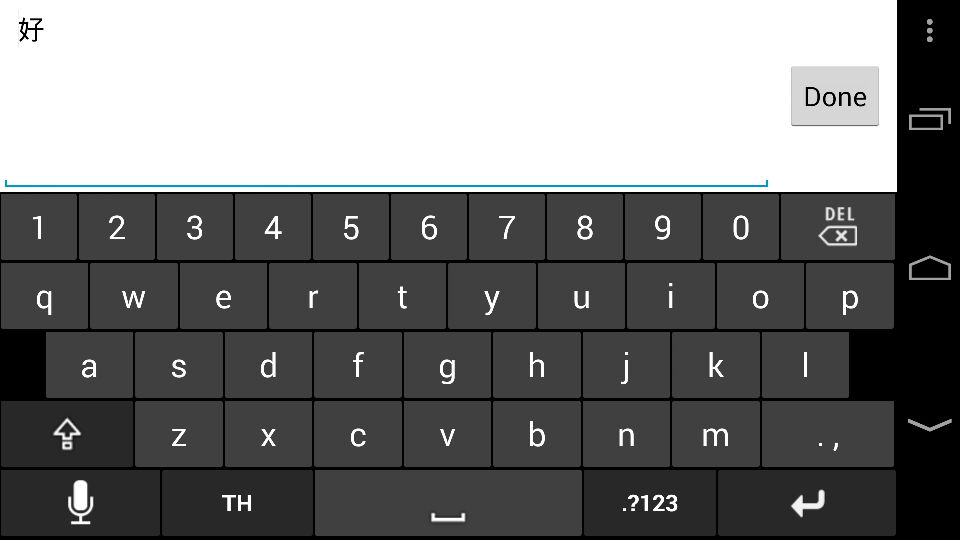
**Page Name: Translator page**



**Description:** Once users hit the “translate” button, this page displays the meaning of the scanned word(s) in English below. Users can view the OCR result in Chinese, the meaning of scanned word(s) in English and the scanned image. In the lower left corner of the screen, users can get the get the Chinese pronunciation of scanned word(s) by clicking the “speech” button. And users can also click “return” button to return to scan page and continue scanning another word. Moreover, users can click Chinese text to edit the OCR result of scanned word.

**UI-03:**

**Page Name: Edit scanned word page**



**Description:** When users click the Chinese word(s) to edit the text, this page appears to allow users to edit the word(s)(or Location name(s)) that they want to get the translation of. After users click “done” button, users will be led to the Translator page (UI-02) and get the new meaning of the edited Chinese word in English.

**Chapter Six | Reference**

[1] ZXing Project

<https://github.com/zxing/zxing>

[2] OCR Library tess two

<http://www.eoeandroid.com/thread-299133-1-1.html>