

Software Development Kit
ANPR

Copyright 2019

Recogniform Technologies SpA

HOW TO CONTACT US

Recogniform Technologies SpA
Contrada Concistocchi
87036 Rende (CS), Italy

Phone : +39 0984 404174

Fax : +39 0984 830299

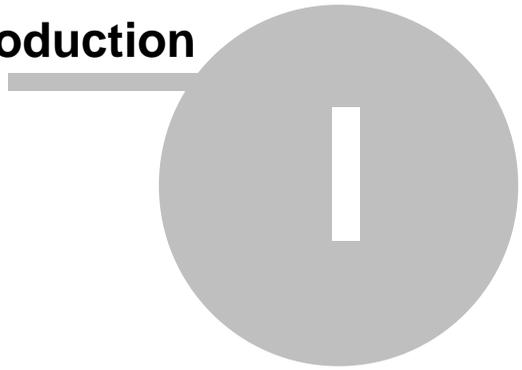
Internet : www.recogniform.com

E-Mail : info@recogniform.com

Table of contents

Introduction	5
Copyright	5
License	5
Overview	5
Usage	7
Visual C++	7
C#	7
Visual Basic	7
Visual Basic .NET	7
Delphi	7
Java	7
API References	9
ANPR_Init	9
ANPR_Recognize	10
ANPR_Done	11
ANPR_GetNumPlates	12
ANPR_GetPlateValue	13
ANPR_GetPlateConfidence	14
ANPR_GetPlateRect	15
ANPR_GetAltCharacterValue	16
ANPR_GetAltCharacterConfidence	18
ANPR_SetParameter	19
ANPR_LoadImage	22
ANPR_FreedImage	23
Error code	26

Introduction



1 Introduction

1.1 Copyright

The software and the documentation are property of:

Recogniform Technologies SpA
Contrada Concistocchi
87036 Rende (CS)
Italy

www.recogniform.com
info@recogniform.com

1.2 License

It is illegal to copy or reproduce this manual, or any part thereof, in any shape or form.

The information contained in this manual is subject to change without notice and does not present a commitment on the part of Recogniform Technologies SpA.

Recogniform Technologies SpA shall not be held liable for technical or editorial errors and/or omissions made here, nor for incidental or consequential damages resulting from the furnishing, performance, or use of the software and documentation.

Recogniform Technologies SpA reserves the right to make changes to the software and documentation without notice.

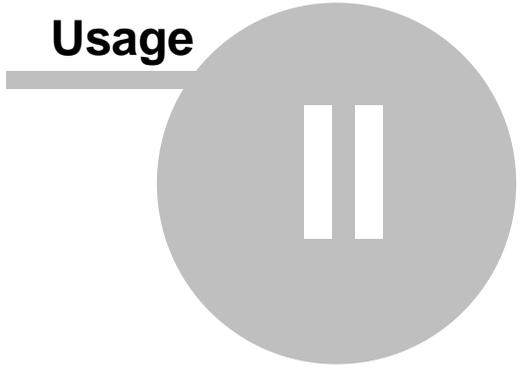
Product names mentioned here are used for identification purposes only and may be tradenames and/or registered trademarks of their respective companies.

YOU CANNOT DISTRIBUTE SOFTWARE INCLUDING THIS SDK LIBRARY UNLESS YOU HAVE A WRITTEN AGREEMENT (ROYALTIES FREE OR ROYALTIES BASED) WITH RECOGNIFORM TECHNOLOGIES SPA

1.3 Overview

The library allows to detect the number plates on cars from images.

Usage



2 Usage

2.1 Visual C++

You have to include the RECOANPR.C in your program. Before to execute your application make sure the *RECOANPR.DLL* is available in your same .exe directory or in windows\system directory.

2.2 C#

You have to include the RECOANPR.CS in your program. Before to execute your application make sure the *RECOANPR.DLL* is available in your same .exe directory or in windows\system directory.

2.3 Visual Basic

You have to include the RECOANPR.BAS in your program. Before to execute your application make sure the *RECOANPR.DLL* is available in your same .exe directory or in windows\system directory.

2.4 Visual Basic .NET

You have to include the RECOANPR.VB in your program. Before to execute your application make sure the *RECOANPR.DLL* is available in your same .exe directory or in windows\system directory.

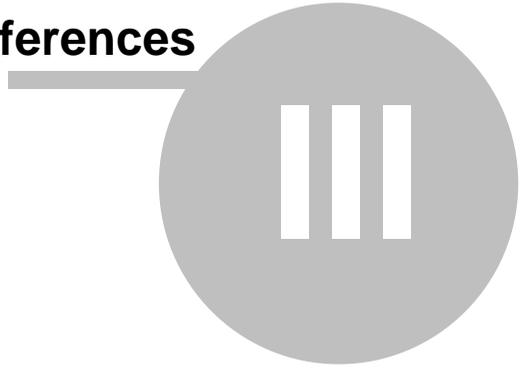
2.5 Delphi

You have to include the RECOANPR.PAS in your program. Before to execute your application make sure the *RECOANPR.DLL* is available in your same .exe directory or in windows\system directory.

2.6 Java

You have to use 32 bit JVM and you have to include the RecoANPRapi.java in your program. Before to execute your application make sure the *RECOANPR.DLL* is available in your same .jar directory.

API References



3 API References

3.1 ANPR_Init

C/C++ Declaration

```
int __stdcall ANPR_Init(char* User, char* Password, int* Session);
```

C# Declaration

```
int ANPR_Init(string User, string Password, ref int Session);
```

Visual Basic Declaration

```
Function ANPR_Init(ByVal User As String, ByVal Password As String, ByRef Session As Long)As Long
```

Visual Basic .NET Declaration

```
Function ANPR_Init(ByVal User As String, ByVal Password As String, ByRef Session As Integer)As Integer
```

Delphi Declaration

```
function ANPR_Init(User:PAnsiChar;Password:PAnsiChar;Var Session:Integer):Integer; stdcall;
```

Java Declaration

```
int ANPR_Init(String User, String Password,IntByReference Session);
```

Description

This is the first function to call: initialize the library and returns a session handle to use in next calls. When you buy the library you receive an "user" and a "password" string necessary to initialize the library in licensed mode: without this value or with wrong values the library is initialized in evaluation mode.

Parameters

User (in) - user name string

Password (in) - password string

Session (out) - session handle to use in next library calls

Return values

Zero if the function has success, a value less than zero, representing an error code, otherwise. Refer to error codes section to more information on possible values.

3.2 ANPR_Recognize

C/C++ Declaration

```
int __stdcall ANPR_Recognize(int Session, int DIBHandle);
```

C# Declaration

```
int ANPR_Recognize(int Session, int DIBHandle);
```

Visual Basic Declaration

```
Function ANPR_Recognize(ByVal Session As Long, ByVal DIBHandle As Long)As Long
```

Visual Basic .NET Declaration

```
Function ANPR_Recognize(ByVal Session As Integer, ByVal DIBHandle As Integer)As Integer
```

Delphi Declaration

```
function ANPR_Recognize(Session, DIBHandle:Integer):Integer; stdcall;
```

Java Declaration

```
int ANPR_Recognize(int Session, int DIB);
```

Description

This is the function performing the recognition of the number plates

Parameters

Session (in) - session handle to use
DIBHandle (in) - handle of the DIB to recognize

Return values

Number of plates recognized, a value less than zero, representing an error code, otherwise. Refer to error codes section to more information on possible values.

3.3 **ANPR_Done**

C/C++ Declaration

```
int __stdcall ANPR_Done(int Session);
```

C# Declaration

```
int ANPR_Done(int Session);
```

Visual Basic Declaration

```
Function ANPR_Done(ByVal Session As Long)As Long
```

Visual Basic .NET Declaration

```
Function ANPR_Done(ByVal Session As Integer)As Integer
```

Delphi Declaration

```
function ANPR_Done(Session:Integer):Integer;  
stdcall;
```

Java Declaration

```
int ANPR_Done(int Session);
```

Description

This is the last function to call when you don't need more services from the library: deinitialize the library and free all used resources.

Parameters

Session (in) - session handle to free

Return values

Zero if the function has success, a value less than zero, representing an error code, otherwise. Refer to error codes section to more information on possible values.

3.4 **ANPR_GetNumPlates**

C/C++ Declaration

```
int __stdcall ANPR_GetNumPlates(int Session, int* NumPlates);
```

C# Declaration

```
int ANPR_GetNumPlates(int Session, ref int NumPlates);
```

Visual Basic Declaration

```
Function ANPR_GetNumPlates(ByVal Session As Long, ByRef NumPlates As Long)As Long
```

Visual Basic .NET Declaration

```
Function ANPR_GetNumPlates(ByVal Session As Integer, ByRef NumPlates As Integer)As Integer
```

Delphi Declaration

```
function ANPR_GetNumPlates(Session:Integer; var NumPlates:Integer):Integer; stdcall;
```

Java Declaration

```
int ANPR_GetNumPlates(int Session, int[] NumPlates);
```

Description

This is the function to call after the recognition process to get the number of plates recognized in the image

Parameters

Session (in) - session handle to use

NumPlates (out) - number of plates recognized in the image.

Return values

Zero if the function has success, a value less than zero, representing an error code, otherwise. Refer to error codes section to more information on possible values.

3.5 **ANPR_GetPlateValue**

C/C++ Declaration

```
int __stdcall ANPR_GetPlateValue(int Session, int PlateIndex, char* PlateString);
```

C# Declaration

```
int ANPR_GetPlateValue(int Session,int PlateIndex, StringBuilder PlateString);
```

Visual Basic Declaration

```
Function ANPR_GetPlateValue(ByVal Session As Long, ByVal PlateIndex As Long, ByVal Buffer As String)As Long
```

Visual Basic .NET Declaration

```
Function ANPR_GetPlateValue(ByVal Session As Integer, ByVal PlateIndex As Integer, ByVal Buffer As StringBuilder)As Integer
```

Delphi Declaration

```
function ANPR_GetPlateValue(Session,PlateIndex:Integer; PlateString:Pansichar):Integer; stdcall;
```

Java Declaration

```
ANPR_GetPlateValue(int Session, int PlateIndex, Memory Buffer);
```

Description

This is the function to call after the recognition process to get the plate value recognized.

Parameters

Session (in) - session handle to use

PlateIndex (in) - plate index to recognize

PlateString (out) - buffer where will be moved the value of the plates recognized if the *PlateString* is not null. You have to allocate enough space.

Return values

String length with plate number recognized, a value less than zero, representing an error code, otherwise. Refer to error codes section to more information on possible values.

3.6 ANPR_GetPlateConfidence

C/C++ Declaration

```
int __stdcall ANPR_GetPlateConfidence(int Session,
int PlateIndex, int* Confidence);
```

C# Declaration

```
int ANPR_GetPlateConfidence(int Session,int
PlateIndex, ref int Confidence);
```

Visual Basic Declaration

```
Function ANPR_GetPlateConfidence(ByVal Session As
Long, ByVal PlateIndex As Long, ByRef Confidence
As Long)As Long
```

Visual Basic .NET Declaration

```
Function ANPR_GetPlateConfidence(ByVal Session As
Integer, ByVal PlateIndex As Integer, ByRef
Confidence As Integer)As Integer
```

Delphi Declaration

```
function ANPR_GetPlateConfidence(Session,
PlateIndex:Integer; Var
Confidence:Integer):Integer; stdcall;
```

Java Declaration

```
int ANPR_GetPlateConfidence(int Session, int
PlateIndex, int Confidence);
```

Description

This function get the plate confidence.

Parameters

Session (in) - session handle to use

PlateIndex (in) - plate index to recognize

Confidence(out) - confidence level from 0 to 100

Return values

Zero if the function has success, a value less than zero, representing an error code, otherwise. Refer to error codes section to more information on possible values.

3.7 **ANPR_GetPlateRect**

C/C++ Declaration

```
int __stdcall ANPR_GetPlateRect(int Session,int  
PlateIndex, int* Top, int* Left, int* Bottom, int*  
Right);
```

C# Declaration

```
int ANPR_GetPlateRect(int Session,int PlateIndex,  
ref int Left, ref int Top, ref int Bottom, ref int  
Right);
```

Visual Basic Declaration

```
Function ANPR_GetPlateRect(ByVal Session As Long,  
ByVal PlateIndex As Long, ByRef Left As Long,  
ByRef Top As Long, ByRef Bottom As Long, ByRef  
Right As Long)As Long
```

Visual Basic .NET Declaration

```
Function ANPR_GetPlateRect(ByVal Session As  
Integer, ByVal PlateIndex As Integer, ByRef Left  
As Integer, ByRef Top As Integer, ByRef Bottom As  
Integer, ByRef Right As Integer)As Integer
```

Delphi Declaration

```
function  
ANPR_GetPlateRect(Session,PlateIndex:Integer; Var  
Left,Top,Bottom,Right:Integer):Integer; stdcall;
```

Java Declaration

```
int ANPR_GetPlateRect(int Session,int PlateIndex,  
int Top, int Left, int Bottom, int Right);
```

Description

Retrieves the rectangle coordinates of the plate recognized .

Parameters

Session (in) - session handle to use

PlateIndex (in) - index of the plate.

Left (out) - buffer where will be moved the left position of the rect containing the plate.

Top (out) - buffer where will be moved the top position of the rect containing the plate.

Right (out) - buffer where will be moved the right position of the rect containing the plate.

Bottom (out) - buffer where will be moved the bottom position of the rect containing the plate.

Return values

Zero if the function has success, a value less than zero, representing an error code, otherwise. Refer to error codes section to more information on possible values.

3.8 ANPR_GetAltCharacterValue

C/C++ Declaration

```
int __stdcall ANPR_GetAltCharacterValue (int  
Session, int PlateIndex, int CharIndex, int  
AlternativeIndex, char* CharBuffer);
```

C# Declaration

```
int ANPR_GetAltCharacterValue(int Session,int  
PlateIndex, int CharIndex, int AlternativeIndex,  
StringBuilder Buffer);
```

Visual Basic Declaration

```
Function ANPR_GetAltCharacterValue(ByVal Session  
As Long, ByVal PlateIndex As Long, ByVal CharIndex  
As Long, ByVal AlternativeIndex As Long, ByVal  
Buffer As String )As Long
```

Visual Basic .NET Declaration

```
Function ANPR_GetAltCharacterValue(ByVal Session  
As Integer, ByVal PlateIndex As Integer, ByVal  
CharIndex As Integer, ByVal AlternativeIndex As  
Integer, ByVal Buffer As StringBuilder )As Integer
```

Delphi Declaration

```
function  
ANPR_GetAltCharacterValue(Session, PlateIndex, CharI  
ndex, AlternativeIndex: Integer;  
Buffer: PAnsiChar): Integer; stdcall;
```

Java Declaration

```
int ANPR_GetAltCharacterValue (int Session, int  
PlateIndex, int CharIndex, int AlternativeIndex,  
Memory Buffer);
```

Description

Retrieves the value of one of possible character alternative recognized. For each character you can get a variable number of alternatives.

Parameters

Session (in) - session handle to use

PlateIndex (in) - index of the plate.

CharIndex (in) - index of the character.

AlternativeIndex (in) - index of the alternative.

CharBuffer (out) - buffer where will be moved the value of alternatives available for each recognized character.

Return values

Zero if the function has success, a value less than zero, representing an error code, otherwise. Refer to error codes section to more information on possible values.

3.9 ANPR_GetAltCharacterConfidence

C/C++ Declaration

```
int __stdcall ANPR_GetAltCharacterConfidence (int  
Session, int PlateIndex,int CharIndex, int  
AlternativeIndex , int *Confidence);
```

C# Declaration

```
int ANPR_GetAltCharacterConfidence(int Session,int  
PlateIndex, int CharIndex, int AlternativeIndex,  
ref int Confidence);
```

Visual Basic Declaration

```
Function ANPR_GetAltCharacterConfidence(ByVal  
Session As Long, ByVal PlateIndex As Long, ByVal  
CharIndex As Long, ByVal AlternativeIndex As Long,  
ByRef Confidence As Long)As Long
```

Visual Basic .NET Declaration

```
Function ANPR_GetAltCharacterConfidence(ByVal  
Session As Integer, ByVal PlateIndex As Integer,  
ByVal CharIndex As Integer, ByVal AlternativeIndex  
As Integer,ByRef Confidence As Integer)As Integer
```

Delphi Declaration

```
function  
ANPR_GetAltCharacterConfidence(Session,PlateIndex,  
CharIndex,AlternativeIndex:Integer; Var  
Confidence:Integer):Integer; stdcall;
```

Java Declaration

```
int ANPR_GetAltCharacterConfidence (int Session,  
int PlateIndex,int CharIndex, int AlternativeIndex  
, int []Confidences);
```

Description

This function modify the DIB filling the black border using white color.

Parameters

Session (in) - session handle to use

PlateIndex (in) - index of the plate.

CharIndex (in) - index of the character.

AlternativeIndex (in) - index of the alternative.

CharBuffer (out) - buffer where will be moved the confidence of alternatives available for each recognized character.

Return values

Zero if the function has success, a value less than zero, representing an error code, otherwise. Refer to error codes section to more information on possible values.

3.10 ANPR_SetParameter

C/C++ Declaration

```
int __stdcall ANPR_SetParameter (int Session, int  
ParameterIndex, void* ParameterValue);
```

C# Declaration

```
int ANPR_SetParameter(int Session,int  
ParameterIndex, int ParameterValue);
```

Visual Basic Declaration

```
Function ANPR_SetParameter(ByVal Session As Long,  
ByVal ParameterIndex As Long, ByVal ParameterValue  
As Long)As Long
```

Visual Basic .NET Declaration

```
Function ANPR_SetParameter(ByVal Session As  
Integer, ByVal ParameterIndex As Integer,ByVal  
ParameterValue As Integer )As Integer
```

Delphi Declaration

```
function  
ANPR_SetParameter(Session,ParameterIndex,Parameter  
Value:Integer):Integer; stdcall;
```

Java Declaration

```
int ANPR_SetParameter (int Session, int
```

```
ParameterIndex, int ParameterValue);
```

Description

This function allows to modify default parameters for recognition

Parameters

SessionHandle (in) - session handle to use

ParameterIndex (in) - parameter index to modify. Currently are supported this parameters:

- ANPR_PAR_MAXPLATES = 0: set the maximum number of plates in images. Default is 3
- ANPR_PAR_MAXCHARALTERNATIVES = 1: maximum number of alternatives character to preserve (maximum is 8). Default is 4.
- ANPR_PAR_MINCHARHEIGHT = 2: set character minimum height. Default 12
- ANPR_PAR_ADDPLATESHEMA = 3: Add plate schema

ParameterValue (in) - new parameter value. See parameter Id for admitted values.

Return values

Zero if the function has success, a value less than zero, representing an error code, otherwise. Refer to error codes section to more information on possible values.

Notes

ANPR_PAR_MAXCHARALTERNATIVES : The default value is 4 to preserve the two best character and the two best digit.

ANPR_PAR_ADDPLATESHEMA:

Plate schemas (i.e. license plate tipology) are introduced to encrease the possibility to recognize plates.

In plate schemas the type of character (number or letter) is specified in each position of license plate, in order to solve the cases of doubtful recognition (for example B and 8); each country can have one or more plate schema.

Plate schema are imported in json format:

```
"plate_schema" : {  
  "name": ...  
  "nation": ...  
  "num_lines": ...
```

```

    "lines": [{
      "num_chars":...
      "char_height":...
      "char_width":...
      "space_width":...
      "chars": "... "
    }]
    ...
  }

```

Plate schemas have the following properties:

- name: schema name
- nation: nation code (ISO ALFA2)
- num_lines: number of license plate lines (maximum 2 lines)

Each line has the following properties:

- num_chars: number of character
- char_height: (common) height of the body character
- char_width: width of the body character
- space_width: width of the space
- top,left: top and left coordinates of the first character
- length: length of the character sequence in pixel
- plates_lines specified by a string corresponding to the character sequence and space. Each character/space have to be closed in curly brackets ("**<char category> <attributes>**").

The categories are

- A letter
- N number
- AN letter or number
- _ space

The allowed attributes are :

- W:width (character width or space width),
 - H:height,
 - T (top), L (left), r (right) b(bottom): coordinates of the character.
- If the attributes are not specified, the ones of the line are used.

```

char *json="{ \"plate_schema\": {\"
  \"name\": \"SM 5\",
  \"nation\": \"SM\",
  \"num_lines\": 1,
  \"lines\": [{
    \"top\": 0,

```

```
" \left\": 0,"  
" \num_chars\": 5,"  
" \char_height\":120,"  
" \char_width\":68,"  
" \space_width\":13,"  
" \chars\": \"(AN)(AN)(AN)(AN)(AN)\\"  
  
"}}}}";
```

3.11 ANPR_LoadImage

C/C++ Declaration

```
int __stdcall ANPR_LoadImage (int Session, char*  
FileName, int* DIBHandle);
```

C# Declaration

```
int ANPR_LoadImage(int Session, string FileName,  
ref int DIBHandle);
```

Visual Basic Declaration

```
Function ANPR_LoadImage(ByVal Session As Long,  
ByVal FileName As String, ByRef DIBHandle As Long)  
As Long
```

Visual Basic .NET Declaration

```
Function ANPR_LoadImage(ByVal Session As Integer,  
ByVal FileName As String, ByRef DIBHandle As  
Integer) As Integer
```

Delphi Declaration

```
function ANPR_LoadImage(Session:Integer;  
FileName:PAnsiChar; var  
DIBHandle:Integer):Integer; stdcall;
```

Java Declaration

```
int ANPR_LoadImage (int Session, String FileName,  
int DIBHandle);
```

Description

This is the function performing the loading of an image (jpg or bmp)

from a file.

Parameters

Session (in) - session handle to use

FileName(in) - complete file name path of the file to load

DIBHandle (out) - DIB handle of the loaded image.

Return values

Zero if the function has success, a value less than zero, representing an error code, otherwise. Refer to error codes section to more information on possible values.

3.12 ANPR_FreeImage

C/C++ Declaration

```
int __stdcall ANPR_FreeImage (int Session, int DIBHandle);
```

C# Declaration

```
int ANPR_FreeImage(int Session, int DIBHandle);
```

Visual Basic Declaration

```
Function ANPR_FreeImage(ByVal Session As Long, ByVal BMPHandle As Long)As Long
```

Visual Basic .NET Declaration

```
Function ANPR_FreeImage(ByVal Session As Integer, ByVal BMPHandle As Integer)As Integer
```

Delphi Declaration

```
function ANPR_FreeImage(Session:Integer; DIBHandle:Integer):Integer; stdcall;
```

Java Declaration

```
int ANPR_FreeImage (int Session, int DIBHandle);
```

Description

This is the function to call to deallocate memory used by an image.

Parameters

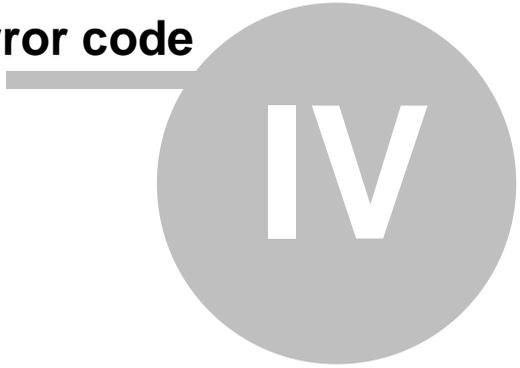
Session (in) - session handle to use

DIBHandle(in) - DIB handle to free.

Return values

Zero if the function has success, a value less than zero, representing an error code, otherwise. Refer to error codes section to more information on possible values.

Error code



4 Error code

The possible error code are:

- *ANPR_ERR_OK* = 0;
- *ANPR_ERR_INTERNALERROR* = -1;
- *ANPR_ERR_INVALIDSESSION* = -2;
- *ANPR_ERR_INVALIDDIBHANDLE* = -3;
- *ANPR_ERR_INVALIDPLATEINDEX* = -4;
- *ANPR_ERR_INVALIDCHARINDEX* = -5;
- *ANPR_ERR_INVALIDCHARALTERNATIVEINDEX* = -6;
- *ANPR_ERR_INVALIDPARAMETERINDEX* = -7;
- *ANPR_ERR_INVALIDPARAMETERVALUE* = -8;
- *ANPR_ERR_USAGESEXHAUSTED* = -9;

Annotation

