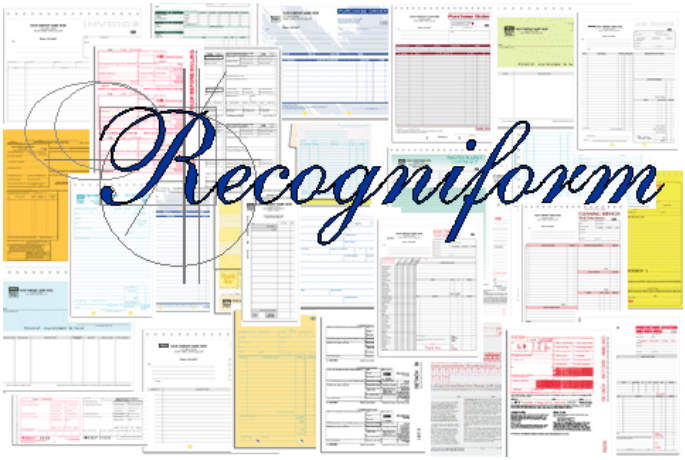


Software Development Kit

Despeckle

Recogniform Technologies SpA

HOW TO CONTACT US

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

Phone : +39 0984 404174

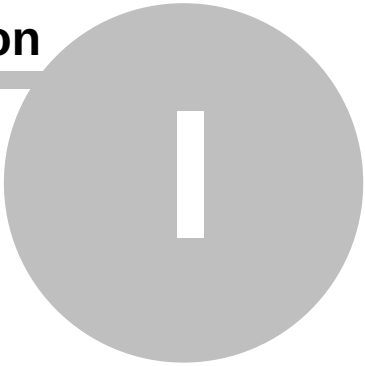
Fax : +39 0984 830299

Internet : www.recogniform.com

E-Mail : info@recogniform.com

Table of contents

Introduction



1 Introduction

1.1 Copyright

The software and the documentation are property of:

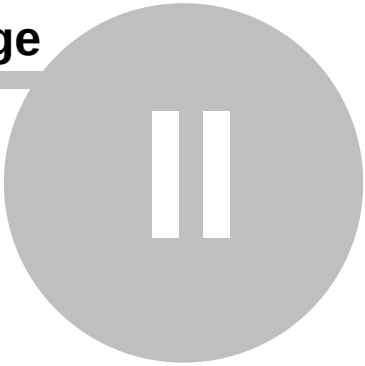
Recogniform Technologies SpA
Contrada Concistocchi
87036 Rende (CS)
Italy
www.recogniform.com
info@recogniform.com

1.2 License

1.3 Overview

The library allows to remove noise points on images acquired using a scanner or a fax.

Usage



2 Usage

2.1 Visual C++

You have to include the RECODESPECKLEAPI.C in your program (see sample application). Before to execute your application make sure the RECODESPECKLE.DLL is available in your same .exe directory or in windows\system directory.

2.2 C#

You have to include the RECODESPECKLEAPI.CS in your program (see sample application). Before to execute your application make sure the RECODESPECKLE.DLL is available in your same .exe directory or in windows\system directory.

2.3 Visul Basic .NET

You have to include the RECODESPECKLEAPI.VB in your program (see sample application). Before to execute your application make sure the RECODESPECKLE.DLL is available in your same .exe directory or in windows\system directory.

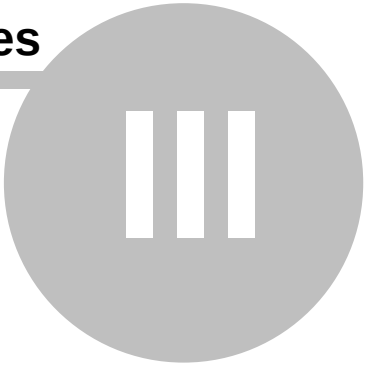
2.4 Visul Basic

You have to include the RECODESPECKLEAPI.BAS in your program (see sample application). Before to execute your application make sure the RECODESPECKLE.DLL is available in your same .exe directory or in windows\system directory.

2.5 Delphi

You have to include the RECODESPECKLEAPI.PAS in your program (see sample application). Before to execute your application make sure the RECODESPECKLE.DLL is available in your same .exe directory or in windows\system directory.

API References



3 API References

3.1 DESPECKLE_Init

C/C++ Declaration

```
__stdcall long DESPECKLE_Init(char* User, char*
Psw);
```

C# Declaration

```
int DESPECKLE_Init(string User, string Psw);
```

Visual Basic Declaration

```
Function DESPECKLE_Init(ByVal User As String,
ByVal Psw As String) As Integer
```

Visual Basic .NET Declaration

```
Function DESPECKLE_Init(ByVal User As String,
ByVal Psw As String) As Integer
```

Delphi Declaration

```
function DESPECKLE_Init(User:PAnsiChar;
Psw:PAnsiChar):Integer; stdcall;
```

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 normal mode: without this value or with wrang values the library is initialized in evaluation mode. The evaluation mode works exactly as normal mode but some time, when you call the despeckle function, is displayed a warning dialog box remembering the evaluation state: you can close it and continue to work with no problems.

Parameters

User (in) - then user name string
Psw (in) - then password string

Return values

The session handle if the library is initialized, 0 otherwise.

3.2 DESPECKLE_Done

C/C++ Declaration

```
__stdcall void DESPECKLE_Done(long SessionHandle);
```

C# Declaration

```
void DESPECKLE_Done(int SessionHandle);
```

Visual Basic Declaration

```
Sub DESPECKLE_Done(ByVal Session As Integer)
```

Visual Basic .NET Declaration

```
Sub DESPECKLE_Done(ByVal Session As Integer)
```

Delphi Declaration

```
procedure DESPECKLE_Done(SessionHandle:Integer);
stdcall;
```

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

SessionHandle (in) - the session handle to free

Return values

n/a

3.3 DESPECKLE_RemovePoints

C/C++ Declaration

```
__stdcall long DESPECKLE_RemovePoints(long
SessionHandle, long DIBIn, long MAxPointW, Long
MaxPointH);
```

C# Declaration

```
int DESPECKLE_RemovePoints(int SessionHandle, int
DIBIn, int MAxPointW, int MaxPointH);
```

Visual Basic Declaration

```
Function DESPECKLE_RemovePoints(ByVal Session As
Integer, ByVal DIBHandle As Integer,ByVal
MAxPointW As Integer,ByVal MaxPointH As Integer)
As Integer
```

Visual Basic .NET Declaration

```
Function DESPECKLE_RemovePoints(ByVal Session As
Integer, ByVal DIBHandle As Integer,ByVal
MAxPointW As Integer,ByVal MaxPointH As Integer)
As Integer
```

Delphi Declaration

```
function DESPECKLE_RemovePoints
(SessionHandle:Integer;DIBHandle:Integer;
MAxPointW: Integer; MaxPointH: Integer):Integer;
stdcall;
```

Description

This function allows to remove the point of a specific size.

Parameters

SessionHandle (in) - the session handle to use
DIBIn (in) - the handle of the monochrome DIB to despeckle
MaxPointW (in) - the maximum width of the point to remove in pixel
MaxPointH (in) - the maximum height of the point to remove in pixel

Return values

The number of removed points.

Notes

The library works with standard DIBs images. Refer to Microsoft

documentation for additional info about Device Indipendent Bitmaps.

3.4 LoadDespeckleLibrary

C/C++ Declaration

```
long LoadDespeckleLibrary();
```

Description

Load the DESPECKLE DLL library: you have to use this function one time before to use other API functions

3.5 FreeDespeckleLibrary

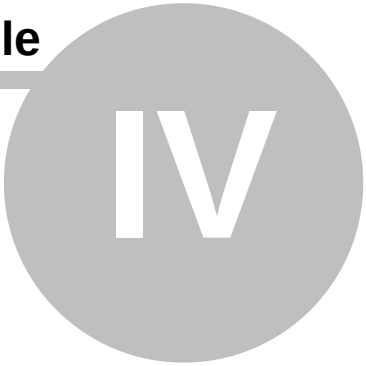
C/C++ Declaration

```
void FreeDespeckleLibrary();
```

Description

Unload the DESPECKLE.DLL library: you have to use this function one time before to exit from your application.

Sample



4 Sample

4.1 Code Sample

```
#include "stdafx.h"
#include <stdio.h>
#include "recodespeckle.c"

int APIENTRY WinMain(HINSTANCE hInstance,
                     HINSTANCE hPrevInstance,
                     LPSTR     lpCmdLine,
                     int       nCmdShow)
{

      // Load dynamically the library
      LoadDespeckleLibrary();

      // Init the session
      int Session= DESPECKLE_Init("demo", "demo");

      long hBitmap;

      // Check if a DIB is available in clipboard
      bool bAvail= IsClipboardFormatAvailable(CF_DIB);

      hBitmap = 0;



      if (bAvail)
      {

            // Open the Clipboard
            ::OpenClipboard(NULL);

            // Retrieve the DIB from clipboard
            hBitmap = (long) GetClipboardData(CF_DIB);


            // Find and removes points of speckle
            long RemovedPoints=
DESPECKLE_RemovePoints(Session, (long) hBitmap, 4, 4);

            ::EmptyClipboard();

            // Set the cleaned DIB to clipboard
            SetClipboardData(CF_DIB, (void*)hBitmap);

            // Close the Clipboard
            ::CloseClipboard();

            char  buffer[200];

            // Format the output message
            if (RemovedPoints!=0)

                  sprintf( buffer, "Removed %d points ! You
can paste in your application...", RemovedPoints);
```

```
            else

            sprintf( buffer, "Image don't need to be cleaned
!");


            // Show the result
            MessageBox(NULL, buffer, "RESULT", MB_OK);


      }
      // Show an error message
      else MessageBox(NULL, "Unable to paste DIB", "ERROR",
MB_OK);


      // Close the session
      DESPECKLE_Done(Session);

      // Unload the library
      FreeDespeckleLibrary();

      return 0;
}
```

Annotation