

WinOCR

User's Guide

Optical Character Recognition Software

Version 2.2
Copyright (c) 1993 SoftTech
First published 1993

SoftTech
London

All Trademarks acknowledged

0. CONTENTS

- 0. CONTENTS
- 1. USING THE MANUAL
- 2. GETTING STARTED
- 3. BASIC PRINCIPLES
- 4. TUTORIAL
- 5. PROGRAM FEATURES
- 6. MENUS
 - Main menu
 - Action menu
 - Bitmap menu
 - Text menu
 - View menu
 - Options menu
 - Help menu
- 7. BOXES
 - Tool Box
 - Template Box
 - Font Box
 - Memory Box
 - Error Box
- 8. RIBBON
- 9. STATUS LINE

1. USING THE MANUAL

It seems to be agreed that manuals are boring and that people don't read them. If you have read this far I will quickly tell you how to get the best out of this one and therefore become familiar with your software more quickly and save yourself some time and effort.

Remember that getting to grips with new software is a learning process, so get to grips with the basics first, and work on small sections at a time. The recommended order for working through the manual and also for mastering the software is:

Get started

- Print and Read the file README.DOC.
- Install the software.
- Get it running.

Use the sample bitmap

- Load the bitmap.
- OCR it.
- Move the text output to your text editor.

Explore the options

- Hide the Edit Box.
- Shrink/Enlarge the Bitmap.
- Hide/Show the Ribbon and Status bar.

Your first image

- Scan your own image.
- Load it into the program.
- Adjust the options as necessary.
- OCR it.
- Move the text to your text editor.

Explore the program

Select part of the bitmap for processing.
Modify a font.
Flip/Rotate bitmaps.

2. GETTING STARTED

This section introduces you to WinOCR and gets you started. I hope you find the program a useful and indispensable time-saver.

Introduction

The WinOCR program is an optical character recognition program. It has been produced to save your time and effort in copying code from books and magazines into your computer.

WinOCR can recognise all the printable characters from the ASCII character set (from ! to ~). This means that you can concentrate on your programming and let your scanner do the typing.

WinOCR is designed for Microsoft Windows 3.1. For best results it is best to run Windows in 386 enhanced mode - for details see your Microsoft Windows User's Guide.

The program has been specially designed to work with the low-cost hand held scanners that are becoming increasingly common.

Installing WinOCR

To install WinOCR see the file README.DOC. This file also contains other information about the program that you should be read.

Starting WinOCR

WinOCR works in Windows. Therefore before attempting to start it you should start Windows. If you do not know how to it is recommended that you read the Windows User's Guide and familiarise yourself with the Windows environment. Once windows has started you can start WinOCR in a number of ways, e.g. by

1. double-clicking the icon.
2. using FILE-RUN option in program manager.
3. double-clicking winocr.exe in file manager.

The WinOCR program now appears. You should see a window like that of other Windows programs.

First Glance.

If the program has been set up correctly you should now see the main window of the program.

This has the usual Windows appearance. If you are not familiar with its components - menu, scroll bar, maximise box, etc. - you should refer to the Windows User's Guide.

Note the presence of the ToolBox. This is a special feature of WinOCR and is used for frequently performed actions and tools such as the selection tool used for selecting parts of the bitmap for processing.

3. BASIC PRINCIPLES

This section explains the basic principles behind Optical Character recognition. It explains what you should be trying to achieve and what you should do to reach your goal.

WinOCR is used in the following way. (1) Identify the book or magazine you want to scan code from. (2) Scan the image using your scanner. (3) Transfer the bitmap to WinOCR. (4) Process the bitmap using WinOCR. (5) Transfer the text to its destination, i.e. from WinOCR into your source file.

These steps are gone through below in more detail. The next section is a tutorial that uses the sample bitmap to go through the process step by step.

Identify your text

Obviously the first thing you have to do is to identify the text you want to OCR. It may for example be a section of program code. The text should be able to be scanned, with experience you will probably learn what scans well and what doesn't.

Scanning

WinOCR works with many scanners. You should have installed and tested your scanner before reaching this point. You will know how to prepare bitmaps. Once you have done so you should either save the bitmap to disk or copy it to the clipboard. Note the following:

1. File Format

WinOCR works with files in Windows Bitmap format. Also it only reads monochrome files. This means you should set your scanner to line art mode.

2. Resolution

Images - in most cases - should be scanned at the highest resolution possible - usually 400dpi. This will tend to produce large bitmaps. However, WinOCR contains commands that allow large bitmaps to be viewed easily on a small screen.

3. Brightness

Bear in mind that how the brightness is set can affect the quality of the bitmap produced by a scan and hence the accuracy of the recognition. Too dark and neighbouring characters will touch or merge into one another reducing accuracy; too light and the characters will break up. A little testing will show the best setting for your scanner.

Transferring Bitmaps to WinOCR

There are two ways to transfer bitmaps to WinOCR. Either via the file-system or by using the clipboard. The choice is yours.

Processing the Bitmap

Once the bitmap is in WinOCR you can now OCR it. First of all you should see it in the main window and shrink it until you can see the entire bitmap. You can now select part of the bitmap and OCR it.

For example, if the image you have scanned consists of both a picture and text, you can select the text area and apply the OCR program to it alone.

Retrieving processed text

Once the recognition is complete WinOCR enables you to transfer the text into other programs or onto disk.

This is done through the Windows clipboard or by dividing the main WinOCR window. The lower part of the window is an editor where the text produced by the program can be edited before being saved to disk or sent to the clipboard.

4. TUTORIAL

This section contains a tutorial that takes you step by step through the process of using WinOCR to transform a scanned image from a bitmap into text. The bitmap you will be using is a sample that should have been placed onto your disk during installation.

Performing Actions in Windows

Most of the actions you will be asked to perform during this tutorial involve selecting an item from the menu. This manual doesn't explain how to do this (see your Windows User's Guide for help). Actions in Windows can usually be performed in more than one way. (1) By using the mouse. (2) By using the keyboard to select options from the menu. (3) By using the keyboard and pressing a combination of accelerator keys. (4) By pressing a button with the mouse.

How to open a bitmap

The first thing to do is to load a bitmap into the program. The bitmap is on disk and is called SHARE.BMP. Use one of the following alternative to open it:

1. Select OPEN from the BITMAP menu.
2. Use the key sequence ALT-B-O.
3. Use the key combination CTRL+O.

A dialog box will appear. Find the file SHARE.BMP and open it.

You should now see the bitmap you have chosen displayed in the main program window.

How to OCR a bitmap

Once you have opened a bitmap you can process it. Use one of the following alternatives to do so:

1. Select RECOGNISE from the ACTION menu.
2. Use the key sequence ALT-A-R.
3. Use the key combination CTRL+R.
4. Press the button labelled GO in the ribbon.

You should see the program begin to process the bitmap.

How to save the text

Once you have processed the bitmap you can place the text in the EDIT BOX; that is the lower half of the program window. You can read the text, compare it to the bitmap and make changes to the text itself or its format. If there are any errors you can also correct them at this stage.

Once the text is to your satisfaction you can save it to disk.

1. Select SAVE from the TEXT menu.
2. Use the key sequence ALT-T-S.
3. Use the key combination CTRL+S.

A dialog box will appear. Type in the filename, e.g. MYTEXT.TXT and save the file.

5. PROGRAM FEATURES

So you have got the sample bitmap loaded and have processed it. Perhaps you have also scanned a bitmap of your own and processed that - or perhaps you had difficulties at that stage. This section explains briefly and in one place what the system can do - so that you can get an overview of its abilities. See the reference section for how to carry out the procedures mentioned here.

Input

As well as being able to load bitmaps from file, the system can also accept bitmaps pasted in from the clipboard. A bitmap that has been loaded into the system can be saved to disk if necessary.

Layout

The main program window can be rearranged. The main components of the window are (1) the ribbon bar, (2) the bitmap area, (3) the edit area and (4) the status line. Each of these can be displayed or hidden by the user, e.g. to make a larger area of the bitmap visible. The program has scroll bars that allow different parts of the bitmap to be viewed; they can be turned on or off as required.

View

Bitmaps that have been loaded into the system may be very large. The size at which they are displayed can be adjusted using the shrink option. They can be reduced up to 1/16 of the original size. There is an enlarge option to reverse the process.

Modifying the Bitmap

Sometimes the scan will have produced an image that needs to be rotated, flipped or have colours reversed. The program can do this.

Selecting part of the bitmap

Sometimes the bitmap that has actually been scanned may contain areas that you don't want to OCR. The program has tools to allow you to mark out an area for processing. Once an area has been selected it can be adjusted or reset. The program can also turn off the option to select areas.

Processing the bitmap

The optical character recognition can be stopped and restarted and aborted if necessary. After it has been halted it can be restarted from the same point. This gives you the option to restart the process if some of the options were not correctly set.

Templates

The program uses templates. This is done to achieve high accuracy and speed in the situations in which the program is designed to be used. A template specifies the characteristics, including the size, of the font that the image consists of. The program has the ability to generate new templates based on the built-in set of fonts and you can save them for future use after giving them a name.

Fonts

Templates have two components: the font and the size. The program has the ability to modify the properties of the font. This can be used to correct persistent errors in recognising certain characters.

Feedback

As the program executes an OCR it provides feedback. In the ribbon the current character is displayed and next to it the program's assessment of the quality of the recognition: (1) perfect, (2) best guess, (3) too small, etc. Also the system displays the whole line as it is being read giving immediate feedback as to how well it is performing. Text is also provided on the status line to keep the user informed of actions as they take place.

Options

The program provides access to a set of options that change the way it processes bitmaps or displays the results. One of these allows an increase in speed by performing fewer tests, this may result in a slight loss in accuracy. Another matches the character as defined in the edit area against the results of the recognition. Another turns the trace on or off.

Debug

Things can go wrong with the process of attempting to OCR bitmaps. Firstly the program may fail to break the bitmap into lines correctly, second it may fail to identify each individual character correctly - perhaps some touch or are broken. Thirdly the program may fail to recognise a character correctly. Fourthly the program may fail to format the output correctly. Tools are provided to help deal with each of these situations.

Utilities

This type of program is memory intensive. A dialog box is provided to help manage memory usage. A dialog box is also provided to display the last error that occurred so that it can still be examined if required, after the original Error message may have been removed from the screen.

Printing

A quick and easy method has been provided to print the contents of the Edit Box.

Editing

Facilities are provided to allow the editing of text produced by the recognition process. Cutting, pasting and copying are allowed. The results of recognition can also be transferred directly to the clipboard as well as to the edit box. Files can be opened and read from disk into the Edit Box, the contents of which can also be saved to disk.

Exiting

The program can be left by a number of methods. When the program is next started some of the options are saved so that they do not need to be set again. The program always uses the last set of options saved. The options are set during installation to default values.

6. MENUS

This section tells you what each option in the menu does. Refer to it whenever you have any difficulties with any menu action. If you do not understand how menus work refer to the Windows User's Guide.

Main menu

This menu is always visible and gives the following options:

ACTION BITMAP TEXT VIEW OPTIONS HELP

The ACTION menu contains the commands that trigger particular processes: recognise, halt, exit, etc.

The BITMAP menu contains commands that allow you to manipulate bitmaps: open, paste, rotate, etc.

The TEXT menu contains commands that allow you to manipulate the text: cut, paste, save, etc.

The VIEW menu contains commands that allow you to change how the program presents information to you. To reduce the size of the bitmap, to display the editor, to hide the status line, etc.

The OPTIONS menu allows you to configure the program. You can turn the trace off, match the edit text with that produced by processing the bitmap, etc.

The HELP menu is similar to that in other Windows programs: it enables you to find help quickly.

Action menu

The commands available under this option allow you to control the program, that is to stop and start actions.

1. Recognise

Use: Starts the OCR process.

Precondition: Loaded bitmap, valid template.

Action: This command starts the OCR process going and depending on the options chosen displays various intermediate results and statistics.

Keystroke: ALT-A-R.

Accelerator: CTRL+R.
Comments: Once an OCR has been started it can be halted by using the HALT command.

2. Scale

Use: Create scale information for a template.
Precondition: Loaded bitmap, template has no scale.
Action: This command analyses a bitmap to generate template information.
Keystroke: ALT-A-S.
Accelerator: None.
Comments: When a template has no scale information a cross appears in the ribbon instead of a tick.

3. Clear Scale

Use: Clear the scale information in a template.
Precondition: Loaded bitmap, valid template.
Action: This clears the scale information in the currently selected template.
Keystroke: ALT-A-C.
Accelerator: None.
Comments: When a template has scale information a tick appears in the ribbon instead of a cross.

4. Reset

Use: Resets the system after an OCR
Precondition: Loaded bitmap.
Action: It sets the program to a state it would be in if the bitmap had just been loaded. It clears any texts from a previous read being displayed in the ribbon.
Keystroke: ALT-A-E.
Accelerator: None.
Comments: Used when you want to start afresh without reloading the bitmap.

5. Halt

Use: Stops an OCR.
Precondition: OCR in progress.
Action: Halts processing at the current character.
Keystroke: ALT-A-H.
Accelerator: CTRL+H.
Comments: Useful to stop the processing of a bitmap temporarily.

6. Start
Use: Restarts an OCR.
Precondition: OCR halted.
Action: Restarts processing after it has been halted.
Keystroke: ALT-A-A.
Accelerator: CTRL+A.
Comments: Used to restart processing of a bitmap after it has been temporarily halted.

7. Exit
Use: Exits the program.
Precondition: None.
Action: Exits from the program destroying the main window.
Keystroke: ALT-A-X.
Accelerator: ALT+F4.
Comments: Before the program exits the current option are saved so that they can be used when the program is run again.

Bitmap menu

The commands available under this options affect the bitmap that is loaded or being loaded into the system.

1. Open
Use: To load a bitmap file.
Precondition: None.
Action: Opens a bitmap stored in a file and displays it in the main program window.
Keystroke: ALT-B-O.
Accelerator: CTRL+O.
Dialog Box: The dialog box that is displayed is used to select a filename.
Comments: The type of bitmap that can be opened is a DIB.

2. Save
Use: To save a bitmap file.
Precondition: Loaded bitmap.
Action: Saves a bitmap with the current Filename.
Keystroke: ALT-B-S.
Accelerator: None.
Comments: Used to save a bitmap that already has a filename. If the bitmap has no filename a dialog box is displayed.

- 3. Save As**
Use: To save a bitmap file.
Precondition: Loaded bitmap.
Action: Saves a bitmap with the filename specified by the dialog box that is displayed.
Keystroke: ALT-B-A.
Accelerator: None.
Dialog: When this menu item is chosen a dialog box is displayed. This box is used to select the filename.
Comments: Used to save a bitmap that has either been pasted into the program via the clipboard or has been modified.
- 4. Clear**
Use: To clear the current bitmap.
Precondition: Loaded bitmap.
Action: Clear the current bitmap. The bitmap window is cleared.
Keystroke: ALT-B-C.
Accelerator: None.
Comments: Used to clear the screen of a bitmap that is no longer required. The text is not cleared at the same time.
- 5. Paste**
Use: To load a bitmap from the clipboard into the program.
Precondition: A bitmap in the clipboard.
Action: Saves a bitmap with the current filename.
Keystroke: ALT-B-P.
Accelerator: CTRL+P.
Comments: A simple way to transfer bitmap from other windows programs.
- 6. Rotate**
Use: To rotate a bitmap.
Precondition: A loaded bitmap.
Action: Rotates a bitmap right (clockwise).
Keystroke: ALT-B-R.
Accelerator: None
Comments: Sometimes it is convenient with some scanners to scan horizontally instead of vertically. This command allows the resulting bitmap to be rotated.

7. Flip
Use: To flip a bitmap.
Precondition: A loaded bitmap.
Action: Flips a bitmap vertically.
Keystroke: ALT-B-F.
Accelerator: None.
Comments: The commands handles the case where a bitmap is upside down.

8. Invert
Use: To invert a bitmap.
Precondition: A loaded bitmap.
Action: Inverts the color of a bitmap.
Keystroke: ALT-B-I.
Accelerator: None.
Comments: Used when there is white text on a black background.

Text menu

The commands available under this option affect the text that has been placed in the editor or is being loaded into the program.

1. Open
Use: To open a text file.
Precondition: None.
Action: Opens a text file and places its contents in the editor.
Keystroke: ALT-T-O.
Accelerator: None.
Dialog Box: The dialog box that is displayed is used to select a filename.
Comments: The command must only be used to open an ASCII text file.

2. Save
Use: To save a text file.
Precondition: Text present in the editor.
Action: Saves the text in the editor with the current filename.
Keystroke: ALT-T-S.
Accelerator: CTRL+S.
Comments: Used to save text with the current filename. If there is no filename a dialog box is displayed.

- 3. Save As**
 Use: To save a text file.
 Precondition: Text present in the editor.
 Action: Saves a text with the filename specified by the dialog box that is displayed.
 Keystroke: ALT-T-A.
 Accelerator: None.
 Dialog: When this menu item is chosen a dialog box is displayed. This box is used to select the filename.
 Comments: Used to save text for transfer to another application.
- 4. Cut**
 Use: To move a selected piece from the editor to the clipboard.
 Precondition: Selected text.
 Action: Moves the selected text into the clipboard and deletes the selected text from the editor.
 Keystroke: ALT-T-T.
 Accelerator: CTRL+X.
 Comments: Works in exactly the same way as the Windows notepad program. The text placed in the clipboard can now be easily moved to other applications.
- 5. Copy**
 Use: To place the selected piece into the editor without deletion.
 Precondition: Selected text.
 Action: Copies the selected text into the clipboard without deleting it.
 Keystroke: ALT-T-C.
 Accelerator: CTRL+C.
 Comments: Works in exactly the same way as the Windows notepad program. The text placed in the clipboard can now be easily moved to other applications.
- 6. Paste**
 Use: To paste text from the clipboard into the editor.
 Precondition: Selected text.
 Action: Places data in text format currently in the clipboard into the editor at the current cursor position.
 Keystroke: ALT-T-P.
 Accelerator: CTRL+V.
 Comments: Works in exactly the same way as the Windows notepad program. It enables the editor built into the program to function like as small text editor.

7. Delete

Use: To delete text in the editor.
Precondition: Selected text.
Action: Deletes the text in the editor that has been selected.
Keystroke: ALT-T-D.
Accelerator: DEL.
Comments: Works in exactly the same way as the Windows notepad program. Note that the deleted text is not placed in the clipboard.

8. Clear All

Use: To clear the contents of the editor.
Precondition: Text present in the editor.
Action: Clear all text currently in the editor.
Keystroke: ALT-T-R.
Accelerator: None.
Comments: Used to clear the text. The bitmap is not cleared by this command.

9. Paste OCR

Use: To paste the results of the last OCR into the editor.
Precondition: OCR performed on bitmap.
Action: Places the result of the last OCR into the editor where it can be edited.
Keystroke: ALT-T-E.
Accelerator: CTRL+E.
Comments: Used to place the text produced by performing an OCR into the editor where it can be modified or saved.

10. Copy OCR

Use: To place the results of the OCR directly into the clipboard.
Precondition: OCR performed on bitmap.
Action: Places the result of the last OCR directly into the clipboard.
Keystroke: ALT-T-Y.
Accelerator: None.
Comments: This is a quick way of transferring text to the clipboard and then to other applications without using the editor.

View menu

The commands available here allow you to control various parts of the screen.

1. Ribbon Bar

Use: Hide/Show the ribbon.
Precondition: None.
Action: Hides the ribbon if it is visible , shows the ribbon if it has been closed.
Keystroke: ALT-V-R.
Accelerator: None.
Comments: Used when you want to have as much space as possible to display the bitmap.

2. Bitmap

Use: Hide/Show the bitmap window.
Precondition: Edit window visible.
Action: Hides the bitmap allowing the editor to occupy the full screen.
Keystroke: ALT-V-B.
Accelerator: None.
Comments: Used to turn off the bitmap and therefore have a larger editor.

3. Edit

Use: Hide/Show the Edit window.
Precondition: Bitmap window visible.
Action: Hides the editor and makes the bitmap window occupy the entire program window; or displays the editor if it is currently hidden.
Keystroke: ALT-V-E.
Accelerator: F3.
Comments: Used to turn off the editor and therefore have the bitmap use the entire window .

4. Status Bar

Use: Hide/Show the Status line.
Precondition: None.
Action: Hides the Status line if it is displayed and shows the Status line if it is currently off..
Keystroke: ALT-V-S.
Accelerator: None.
Comments: Used when you want to have as much room as possible for the bitmap or editor.

5. Shrink

Use: To display a compressed bitmap.

Precondition: Reduction factor less than 16.
Action: Doubles the factor by which the bitmap will be reduced when it is displayed.
Keystroke: ALT-V-H.
Accelerator: -.
Comments: Used to make the whole of a large bitmap visible. The maximum reduction is to 1/16 of the original dimensions. Only the image on the screen is affected, the original bitmap is still used for processing.

6. Enlarge

Use: To enlarge a reduced bitmap.
Precondition: Reduced bitmap is being displayed.
Action: Displays the bitmap enlarged to twice its current size..
Keystroke: ALT-V-N.
Accelerator: +.
Comments: Used to restore a bitmap after it has been reduced. The bitmap can then be displayed at its original size.

7. Tool Box

Use: To turn the Tool Box on or off.
Precondition: None.
Action: Turns the Tool Box on if it is off and off if it is currently on.
Keystroke: ALT-V-T.
Accelerator: F4.
Comments: The Tool Box contains tools and buttons for commands that are used frequently. When not in use it should be closed to reduce clutter. For more details about the Tool Box see the Boxes section of this manual.

8. Template Box

Use: To turn the Template Box on or off.
Precondition: None.
Action: Turns the Template Box on if it is off and off if it is currently on.
Keystroke: ALT-V-P.
Accelerator: F5.
Comments: The Template Box is used to control the template. When not in use it should be closed to reduce clutter. For more details about the Template Box see the Boxes section of this manual.

9. Train Box

Use: To turn the Train Box on or off.
Precondition: None.
Action: Turns the Train Box on if is off and off if it is currently on.
Keystroke: ALT-V-A.
Accelerator: None.
Comments: The Font Box is used to control the built-in fonts. When not in use it should be closed to reduce clutter. For more details about the Font Box see the Boxes section of this manual.

10. Memory Box

Use: To turn the Memory Box on or off.
Precondition: None.
Action: Turns the Memory Box on if is off and off if it is currently on.
Keystroke: ALT-V-M.
Accelerator: None.
Comments: The Memory Box is used to monitor the usage of memory by the system. When not in use it should be closed to reduce clutter. For more details about the Memory Box see the Boxes section of this manual.

11. Error Box

Use: To turn the Error Box on or off.
Precondition: None.
Action: Turns the Error Box on if is off and off if it is currently on.
Keystroke: ALT-V-O.
Accelerator: F6.
Comments: The Error Box is used to view the errors generated by the user as the program is run. When not in use it should be closed to reduce clutter. For more details about the Error Box see the Boxes section of this manual.

12. Bitmap Scrollbars

Use: To turn the Bitmap Scroll bars on or off.
Precondition: None.
Action: Turns the scroll bars in the bitmap window on or off.
Keystroke: ALT-V-L.
Accelerator: F7.
Comments: The scroll bars are used to move around a large bitmap. The scroll bars are only displayed if the bitmap at its current size is larger than the bitmap window.

12. Edit Scrollbars

Use: To turn the Edit Scroll bars on or off.
Precondition: None.
Action: Turns the scroll bars in the edit window on or off.
Keystroke: ALT-V-C.
Accelerator: None.
Comments: The scroll bars are used to move around the editor.

Options menu

The commands available here allow you to control the behaviour of the program.

1. Speed

Use: Selects for speed.
Precondition: None.
Action: Selects for speed as opposed to accuracy.
Keystroke: ALT-O-S.
Accelerator: None.
Comments: Accuracy is the default setting.

2. Accuracy

Use: Selects for accuracy.
Precondition: None.
Action: Selects for accuracy as opposed to speed.
Keystroke: ALT-O-A.
Accelerator: None.
Comments: This is the default setting.

3. Trace

Use: Turns the trace on or off.
Precondition: None.
Action: Switches the trace on or off, the trace highlights each character as it is processed.
Keystroke: ALT-O-T.
Accelerator: None.
Comments: Allows you to follow the progress of an OCR.

4. Verbose

Use: To set the level of feedback.
Precondition: None.
Action: If set the level of information displayed during

an OCR is increased.
Keystroke: ALT-O-V.
Accelerator: None.
Comments: Allows you to follow the progress of an OCR.

5. Match

Use: Turns on matching.
Precondition: None.
Action: The program will now match the result of an OCR with the contents of the editor.
Keystroke: ALT-O-M.
Accelerator: None.
Comments: Necessary when modifying fonts.

6. Save Settings on Exit

Use: Save settings on exit.
Precondition: None.
Action: The program will now save settings on exit.
Keystroke: ALT-O-E.
Accelerator: None.
Comments: Used to save frequently used configurations.

Help menu

The commands available here allow you to access the on-line help system

1. Contents

Use: Displays the help contents.
Precondition: None.
Action: Displays the contents topic of the help file of WinOCR.
Keystroke: ALT-H-C.
Accelerator: None.
Comments: Used to access the help system.

2. How to Use Help

Use: Displays help about using help.
Precondition: None.
Action: Displays information about how to use the help system itself. However, the help system works like that of other windows programs.
Keystroke: ALT-H-H.
Accelerator: None.
Comments: Allows access to help on how to use help.

3. About Shareware

Use: Displays information about Shareware

Precondition: None.

Action: Displays information about Shareware. For example how to register WinOCR.

Keystroke: ALT-H-A.

Accelerator: None.

Comments: Can be used to print the order form and to find out how to get support.

7. **BOXES**

This section tells you what each of the boxes does. Refer to it whenever you have any difficulties with any box.

Tool Box

You should display this box displayed when you need commonly used tools and buttons for frequently used commands.

1. Cross

Use: Selects the default cross cursor.

Precondition: None.

Comments: Used to select the cross cursor if the cursor has been modified by selecting another cursor.

2. [] Area selection

Use: Turns on or off area selection.

Precondition: None.

Comments: Used to turn the area selection on or off.

3. Top Left

Use: Used to change the cursor to the top left selection cursor.

Precondition: None.

Comments: When only a part of the bitmap is to be processed the selection area should be turned on. Once this has been done this tool can be used to modify the top left of the selected area.

4. Bottom Right

Use: Used to change the cursor to the bottom right selection cursor.

Precondition: None.

Comments: When only a part of the bitmap is to be processed the selection area should be turned on. Once this has been done this tool can be used to modify the bottom right of the selected area.

5. Up
Use: Used to move the line highlight up.
Precondition: Bitmap has been analysed for lines.
Comments: A bitmap is processed in the following sequence. The bitmap is broken down into lines, each line is broken down into characters. The command allows the user to select the line to process.

6. Down
Use: Used to move the line highlight down.
Precondition: Bitmap has been analysed for lines.
Comments: A bitmap is processed in the following sequence. The bitmap is broken down into lines, each line is broken down into characters. The command allows the user to select the line to process.

7. Left
Use: Used to move the character highlight left.
Precondition: Bitmap has been processed.
Comments: Used to move around an analysed bitmap. Useful when assessing the quality of an OCR.

8. Right
Use: Used to move the character highlight right.
Precondition: Bitmap has been processed.
Comments: Used to move around an analysed bitmap. Useful when assessing the quality of an OCR.

Template Box

This box is used to control the templates that the program uses. A template is a description of the characteristics of the text being analysed. It consists of font data, size, etc. Characters are recognised by comparing them against the template. The current template is selected using the list box in the ribbon not using the template box. The purpose of the template box is to allow templates to be created or modified.

- 1. Name**
Use: Allows a template to be selected.
Precondition: None.
Comments: Used to select a template for creation or modification.
- 2. Font**
Use: To select the template font.
Precondition: None.
Comments: Each template has a font associated with it. Select on from those available.
- 3. Properties**
Use: To select template properties.
Precondition: None.
Comments: Used to select additional properties for the template.
- 4. Text**
Use: Displays template parameters.
Precondition: None.
Comments: This is used to display information about the currently selected template.
- 5. OK button**
Use: Closes the box.
Precondition: None.
Comments: This command is used to close the template box. The contents of the template are not permanently set.
- 6. Save button**
Use: To save the current templates.
Precondition: None.
Comments: After a template has been changed the user can save the new configuration so that it can be used on future occasions.
- 7. Select button**
Use: Selects the template displayed in the box.
Precondition: Current template not already selected.
Comments: This command is used to set the template to the template currently being modified.

8. Clear button

Use: To clear the template.

Precondition: None.

Comments: This command is used to clear a template that is no longer required. The template fields are reset to their default values.

9. Scale button

Use: To create scale information for the template.

Precondition: Bitmap loaded, and selected template has no scale.

Comments: This command is used to create scale information which is done by analysing the current bitmap.

Train Box

This box is used to make modifications to fonts. This is useful if the program fails to recognise a particular symbol as the character that it really is and the user wants to prevent this from happening in the future.

1. Font button

Use: To display the current font.

Precondition: None.

Comments: Used to display the current font. This is the font that will be modified if the user makes changes.

2. Locked button

Use: To set the font to the template font.

Precondition: None.

Comments: Used to help prevent accidental modification of fonts.

3. Pass button

Use: Sets a character.

Precondition: The scan doesn't match the character from the editor.

Comments: If the OCR doesn't match the character from the editor the user should check for spelling mistakes, then that the character in the bitmap is broken or touches a neighbour. This command can then be used to set the character to its correct value.

4. Save button

Use: Saves the font.

Precondition: None.

Comments: After a font has been modified this button can be used to save it so that the new font will be available in the

future.

5. Skip button

Use: To bypass a mismatch.

Precondition: The scan doesn't match the character from the editor.

Comments: Used when the user decides not to press the pass button to correct a mismatch. The system skips the current character and moves on to the next one.

6. Reset button

Use: To reset the box.

Precondition: None.

Comments: Used if it is necessary to reset the box.

7. Split button

Use: To indicate that two characters touch.

Precondition: The scan doesn't match the character from the editor.

Comments: Used to indicate that the character being examined actually consists of two or more actual characters.

Memory Box

This box is used to monitor how the system is using memory. It helps the user especially when the system only has a small amount of memory or is working with very large bitmaps.

1. Text

Use: To display information.

Precondition: None.

Comments: This field displays the amount of memory free in the system and text explaining which value is being displayed.

2. Switch button

Use: To select the information to display.

Precondition: None.

Comments: This button is used to select the property to display. This can be the amount of global memory free or the size of the largest block that is free.

Error Box

This box is used to display error messages. It displays the last error that occurred.

1. Text

Use: To display error messages.

Precondition: None.

Comments: This field displays the last error that occurred. This is available even after the original error dialog box has been removed from the screen.

8. RIBBON

This section describes the contents of the ribbon. It should be referred to when information is required about the contents of any particular field in the ribbon or what the buttons in the ribbon do.

Buttons

The ribbon contains a number of buttons. They are for frequently used commands.

1. Go (Start) button

Use: To start an OCR.

Precondition: Bitmap present.

Comments: Used to start the OCR process. After this button has been pressed the program begins to process the bitmap displaying the results as each character is read. If the program is currently halted this button will say: Start, pressing the button will restart the program.

2. Abort (Stop) button

Use: To abort an OCR.

Precondition: OCR in progress.

Comments: Sometimes it is necessary to stop an OCR before it finishes. This button can be used to achieve this. If the program is currently halted this button will say: Abort, pressing the button will reset the program.

3. Print button

Use: Prints the contents of the editor.

Precondition: Text in the editor.

Comments: This button can be used to print the contents of the editor. Text can be transferred to the editor after an OCR has finished.

4. Exit button

Use: To stop WinOCR.

Precondition: None.

Comments: Used to quit the program.

List box

The ribbon contains an important list box. This box also displays the template that is currently selected.

1. **Template box**

Use: To select templates.

Precondition: None.

Comments: Used to select the template that the program should use to recognise characters.

Fields

The ribbon contains an fields that display information. This information is mainly used to monitor the progress of an OCR.

1. **Result field**

Use: Display the result code To select templates.

Precondition: OCR in progress.

Comments: Displays the code for the OCR of a particular character can be used to monitor the progress of an OCR.

2. **Bitmap character field**

Use: Display the character.

Precondition: OCR in progress.

Comments: Displays the character produced by analysing the bitmap.

3. **Edit character field**

Use: Display the character from the editor.

Precondition: OCR in progress match option selected.

Comments: Displays the character from the editor that has the same position as that in the bitmap. Comparing this character with that produced by analysing the bitmap can be used to find problems.

4. **Line field**

Use: To display an entire lead as it is being read.

Precondition: OCR in progress.

Comments: Displays the characters as they are produced by analysing the bitmap.

5. **Scale field**

Use: Displays the status of the current template.
Precondition: None.
Comments: Show either a cross or a tick depending on whether the current template has valid scale parameters.

9. STATUS LINE

This section describes the contents of the status line. It should be referred to when information is required about the contents of any particular field in the status line.

Fields

The status line contains fields that display information. This information is used by WinOCR to inform the user about what the program is doing and the progress of various operations.

1. Left status field

Use: Displays help information.

Precondition: None.

Comments: Shows how the user can access the help system.

2. Middle status field

Use: Shows the progress of the current action.

Precondition: None.

Comments: When a user starts a lengthy operation information is displayed here showing the progress. For example, when an OCR has started and the trace option is on, the current line number and character number is shown here.

3. Right status field

Use: Shows the program's response to user commands.

Precondition: None.

Comments: When a user begins a command information is shown here indicating that the system is responding.