



PrimePost™

User Manual **Version 3.60**

Post OCR Processing

Prime Recognition

PHONE: 650-631-9800

FAX: 650-631-5686

EMAIL: support@primerecognition.com

WEB: www.primerecognition.com

FTP: [ftp.primerecognition.com](ftp://ftp.primerecognition.com)

Copyright © 1996-1999 Prime Recognition. All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photo-copying, recording, or otherwise without the prior written permission of the publishers.

Printed in the United States of America.

Statement of Limited Warranty

Prime Recognition warrants to the original licensee of this program that it conforms to Prime Recognition's specifications. Should this program, in Prime Recognition's option, malfunction due to non-conformity with Prime Recognition's specifications, Prime Recognition will, at its option, repair, replace, or update the program at no charge, provided that the program has not been subjected to misuse, abuse, accident, disaster or non-Prime Recognition authorized alterations, modification, and/or repairs. In no event shall Prime Recognition be liable for incidental or consequential damages in connection with or arising out of the furnishing, performance, or use of any of these programs. Prime Recognition reserves the right to modify or revise all or part of this document without notice and shall not be responsible for any loss, cost, or damage, including consequential damage caused by reliance on these materials.

US Government Restricted Rights

The documentation and software are provided with restricted rights. Use, duplication, or disclosure by the government is subject to restrictions as set forth in subparagraph (C)(1) (ii) of the Rights in Technical Data and Computer Software clause at DFARS 52.227-7013.

Trademarks

PrimePost is a trademark of Prime Recognition.

Other trademarks appearing in this manual are trademarks of their respective companies.

Overview

PrimePost is a relatively simple application that:

- (1) Reads a PRO format output file (typically created by the PrimeOCR engine)
- (2) In some cases applies custom programmed algorithms to correct data as much as possible (optional)
- (3) Saves PRO file or exports data to ASCII, FASCII, TRU, RTF, PDF, or custom (often comma delimited) format.

The error correction algorithms can be very powerful. Error rates have been cut by over 50% in applications that lend themselves to post processing. Note that the custom algorithms often employ probabilistic formulas, therefore not all modifications may be positive, however, the total effect should be positive.

PrimePost is implemented as a 32 bit based software program. It can process up to 29,500 files in a directory and 29,500 files in each of 29,500 multiple level subdirectories with one command. After processing the target file/directory/subdirectories it will terminate. Since you may run multiple instances of the program the coordination of PrimePost across multiple directories is simple. Start up one instance of PrimePost for every directory and it will automatically terminate each process when it reaches the end of its list of files. For maximum speed in processing a single directory you may run PrimePost on several machines and point each instance to the same directory (make sure that "OverwriteExisting setting in PRPOST.INI is set to 0).

System Requirements

OS: NT 3.51/4.0 or Win95/Win98
RAM: At least 4MB of free RAM per instance

Installation

Run SETUP.EXE from the main directory on the installation CD and follow the directions presented. The installation program will handle all installation issues automatically, including decompressing all files into a new or existing directory selected by the user.

A program group entitled "Prime Recognition" will be created by the installation program and an icon will be created for the PrimePost program.

PrimePost requires a hardware key be placed on LPT1 port. Please see a description of hardware key installation issues in *Chapter 3 of PrimeOCR Access Kit Guide*.

Configuration

The file "PRPOST.INI" controls the program. The contents of this file are as follows:

```
[PrimePost]
Version=3.60
LogFile=0
Error Reporting=0
Debug=0
ExitOnComplete=0

[Target Files]
PROPath=c:\results\*.pro+
OverwriteExisting=0
```

[Save Output]
OutputFormat=11
OutputExtension=
MoveFiles=0
Collate=0
CollateFile=

[PostApp]
ApplicationType=

[Change Font]
SizeFont=0

[PDF]
Suspect=0
Desample=200

[RTF]
PageWidth=0
PageHeight=0
PageLeftMargin=0
PageTopMargin=0
PageRightMargin=0
PageBottomMargin=0
WrapText=0
ProportionalFont=Times New Roman
FixedFont=Courier New
SansSerifFont=Arial

Each entry is detailed below:

Version

Should match version of PrimePost engine.

LogFile

A value of one causes all major engine actions to be reported to a ASCII log file titled "PRPOST.LOG" in application directory.

Error Reporting

A value of one causes all errors to generate a error dialog box. A value of zero will only interrupt processing for fatal errors.

Debug

For Prime Recognition technical support use.

ExitOnComplete

If not "0" then Server unloads from memory when batch completes.

PROPath

This is the full path and file name of the PRO file that should be post processed. You may enter a wildcard for both the filename and extension. If you append a "+" to the end of the path all subdirectories of the target directory will be searched for matching files. If a command line is used to initiate the program then

the command line will overwrite this PROPath. In other words, you may supply the PROPath via a command line string when initiating the engine.

OverwriteExisting

A value of one means to start from the beginning and produce output for all PRO files, even if an existing output file already exists.

OutputFormat

This tells PrimePost which output format to use:

"0"=ASCII text file

"1"=FASCII

"3"=PRO

"9"=RTF

"11" = PDF Normal

"12" = PDF Image Only

"13" = PDF Image with Hidden Text

"15" = Comma delimited output. The output of each zone is separated by a comma. Useful for importing output data into a database.

"16" = HTML. For text conversion only. Images are not included in the output.

"XXXX"=custom format (often comma delimited)

"31"=TRU(PR internal use)

(For a description of each format see PrimeOCR Access Kit Guide).

OutputExtension

Extension of output file (e.g. "fix" without quotation marks)

If no value is given the the default file extension will be used (txt, rtf, pdf, htm, etc.)

"Long" extensions, > 3 characters, are supported.

MoveFiles

A value of zero causes just the generation of a output file. A value of one means:

-A "good" output file will delete any TIF or PTM files of the same name (but not PRO files or resultant output file).

-A "bad" PRO file will move output, PRO, PTM, and TIF files of same file name to an \EXCEPT subdirectory. The user may then review these files to see where the problem is, reprocess as necessary, and resubmit to PrimePost.

Collate

During conversion of ".PRO" to output format, combine output files into one file. There are several common methods of collating files together.

0=do not combine output files

1=combine by common prefix, such as:

file_001, file_002

or

file01a, file01b

or

file_01t, file_01v

Any non-alpha character is considered a delimiter for the common prefix filename.

2=combine all output files into one file. Output filename is the same as the first file in the directory except that the extension is defined by the output type.

3=combine all output files into one file. Output filename is the one found in the CollateFile field as defined by the user.

4=Combine files after blank page. If a blank page (0 characters in PRO output) is encountered, a new collate output file is started for the next PRO file. The file name of the collated file is the file name of the PRO file following the blank page.

If the input files were named:

fax1.pro, fax2.pro, fax3.pro, fax4.pro (blank page), fax5.pro, fax6.pro

then the collated output files would be:

fax1 (3 pages), fax5 (2 pages)

5=Combine files after blank page. If a blank page (0 characters in PRO output) is encountered a new collate output file is started. The file name of the collate file is defined by the CollateFile field plus an index number. The index number is increased with each additional collate file.

If the value of the CollateFile field is "prime_out", and the input files were:

fax1.pro, fax2.pro, fax3.pro, fax4.pro (blank page), fax5.pro, fax6.pro

the collated output files would be:

prime_out1 (3 pages), prime_out2 (2 pages)

CollateFile

Name of the output file that is used by option 3 and 5 of collation. The extension of the filename is not required since the extension is determined by the output type.

ApplicationType

This tells PrimePost which rules to apply to the data. Contact Prime Recognition for the appropriate type for your application. **You should leave this blank if you are only converting PRO to some output format such as RTF or PDF.**

SizeFont

Percent change in font size. Only relevant to output formats that report font size, such as RTF, PDF, or PRO. Examples:

0=No change to font size

-20=reduce stated point sizes by 20%

5=increase stated point sizes by 5%

Suspect

0-9, PDF Normal setting. Any word containing at least one character under this confidence will be displayed as an image "snippet" in PDF Normal output. A good starting point for this field is 6.

If blank or 0 then no snippets will appear in output.

Desample

1-600, dpi of images that will be saved in PDF file or RTF file. If blank or 0 then leave dpi of image equal to original. 200 is a common setting if desampling is desired.

PageXXX fields are expressed in BMU's (1/1200 inch)

PageWidth

0=use default, e.g. 9600 fixes page width to 8 inches

PageHeight

0=use default, e.g. 1200 fixes page height to 10 inches

PageLeftMargin

0=use default, e.g. 600 fixes left margin to 0.5 inches

PageTopMargin

0=use default, e.g. 660 fixes top margin to 0.55 inches

PageRightMargin

0=use default, e.g. 8400 fixes right margin to 7 inches (NOTE- MEASURED FROM LEFT EDGE)

PageBottomMargin

0=use default, e.g. 11400 fixes bottom margin to 9.5 inches (NOTE- MEASURED FROM TOP EDGE)

WrapText

0=don't wrap text, insert carriage return at the end of each line.

1=wrap text within paragraph.

ProportionalFont

Font name used for proportional fonts, e.x. "Times New Roman"

FixedFont

Font name used for fixed pitch fonts, e.x. "Courier New"

SansSerifFont

Font name used for sans serif fonts, e.x. "Arial"

Starting Engine

Clicking on the PrimePost Server icon initiates the server interface. To start processing, press the "Start" button. To start the server programmatically (no manual intervention) execute PRIMEPOST.EXE. If you supply an image path, (wildcards are allowed in path) then the server will process that image path instead of the image path found in the INI file.

For example,

```
WinExec("primepost.exe START", SW_SHOW);
```

Or from the Run Command:

```
primepost.exe START c:\images\*.tif
```

Note: The "START" variable is case sensitive.

If you run multiple instances of PrimePost you must copy the PrimePost files into a separate directory for each instance (and run the program from that new directory).

Stopping Engine

To stop the server in the middle of a series of images, press the "Stop" button in the server window. If you restart the engine after stopping it in the middle of series it will start processing at the beginning of the image range, overwriting any previous output depending on the "OverwriteExistingOutput" setting in the PZSERVER.INI configuration file.

Progress Reporting

During operation, the server will update the progress of the images being processed. The right window displays the current status of each image being processed. The left side of the updates several statistics during processing. The stats are tabulated by the total files/directories found in the batch, and how many image files have completed processing.

DIRS:	Complete	(number of directories completed in the batch)
	Total	(indicates total number of directories found in the batch)
FILES:	Complete	(indicates total number of files completed processing in all directories – includes files that generated an error)
	Total	(indicates total number of image files found in all directories)
FILES (in Dir):	Complete	(indicates total number of files completed processing in this directory only – includes files that generated an error)
	Total (in DIR)	(indicates total number of image files found in current directory)
ERRORS:	Total	(indicates total number of errors encountered while processing all directories)
TIME:	Elapsed	(indicates total time elapsed since starting batch processing of all files)
	Remaining	(is an estimate of how much time remains to process all files in the batch)
	All time is shown in seconds.	

Error Handling

If the Error Reporting flag is set to “0”, then non-fatal error messages will be written to the “prpost.log” file.

Error Codes

-99	Pro file has not been loaded yet
-100	Could not open temporary pro file
-110	Could not save pro file pro file
-111	Software version does not match version licensed on hardware key.
-120	Could not open temporary pro file
-130	Could not find prverify.ini file
-143	Could not find\open output file
-150	Could not find this pro file
-160	Pro file includes Word output
-170	Indicated Pro file does not have PRO format
-180	DPI reported in PRO file was not correct
-190	Error in output data. File may be corrupted(number_of_zones).
-200	Error in output data. File may be corrupted(good[i].numlines-converted # of lines).
-210	Error in output data. File may be corrupted(4)

-220 Error in output data. File may be corrupted(added_spaces).
-240 Output data. File may be corrupted(display_lines)
-260 Incorrect line number
-270 Incorrect character number
-280 Attempted to enter/cut text beyond end of current line of text.
-290 Attempted to enter/cut text on a blank line.
-300 Last char was <0
-310 Original line number was incorrect
-320 Original char number was incorrect
-340 xy number was incorrect
-744 Could not create this export file
-755 Output exceeded allocated memory
-756 Error with compressing text buffer
-760 Could not open this tif file
-761 This is not a valid TIF file
-764 This is not a valid TIF ifd offset
-767 Unknown tif type
-781 Error reading rational tag in tif file - x_res or y_res
-790 Error in tif file
-870 to
-999 Error reading license from hardware key
-1032 PrimeOCR engine has not been licensed for PDF output (impacts PrimePost as well)
-1179 License error – please call Prime Recognition
-11791 License error – please call Prime Recognition
-1904 OCR data has not been loaded yet
-1999 Unknown ApplicationType in PRPOST.INI
-9999 Could not allocate memory
-12001 command string was blank
-12003 command string did not match any known command
-12004 number of commands was incorrect
-12005 required command was blank
-12006 end zone > actual number of zones

Example PrimePost Application

DCOH

Application Type: "DCOH"

Description:

Death Certificate form from state of Ohio.

Output Format:

Comma delimited text with file name of "XXX.txt" where "XXX" is the file name of the PRO file. Green Bar type application with 0 fixed zones and 10 column zones, therefore each text file includes 10 entries on each line of file. (Note that blank fields will show up with no output, for example, a line with only one entry in the first field will look like: 12345,,,,,,,,,.)

"Bad" Definition:

No checking is made for bad files. (The confidence level generated by PrimeOCR is a very strong indicator of the quality of this application's output, therefore a check at this level is not necessary.)