



# PRIMEPOST™

**InputAccel Version**

**User Manual**

**Version 4.00**

Post OCR Processing

**PRIME RECOGNITION**

PHONE: 425-895-0550

FAX: 425-895-9580

EMAIL: [support@primerecognition.com](mailto:support@primerecognition.com)

WEB: [www.primerecognition.com](http://www.primerecognition.com)

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

---

Copyright © 1996-2004 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 IA (InputAccel version) is a relatively simple application that:

- (1) Accepts a PRO format file as input from InputAccel server (typically created by the PrimeOCR IAengine and/or PrimeVerify IA)
- (2) In some cases applies custom programmed algorithms to correct data as much as possible (optional)
- (3) Outputs data in PDA2, or PDF format back into IA flow, and/or can output data locally (outside IA flow) in any supported output format, including ASCII, RTF, PRO, RRI3, etc.

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.

## **System Requirements**

OS: NT 4.0 SP3+, Win2000, WinXP  
RAM: At least 8MB of free RAM (does not include OS or other application memory needs)

## **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.

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*.

## **Starting Engine**

Click on the PRPOSTIA.EXE icon in the installation directory (from Explorer) to initiate the server interface. To start processing, press the "Start" button. The PrimePost server will wait for any work to do until the STOP button is pushed.

## **Stopping Engine**

To stop the server press the "Stop" button in the server window. An image in process will complete processing and then the server will stop accepting (and processing) any new work.

## **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, messages in this window can change rapidly. The left side of the

updates several statistics during processing. The stats are tabulated by the total number of files have completed processing and any errors that may have occurred. These statistics are reset to zero when STOP button is pushed or server is restarted.

## Configuring Engine

A setup file controls the PrimePost Engine. You can set which file will be used to control the engine as follows:

- Send the file to the engine through the "Template" variable in a IPP/PCF.
- Send the local path to the engine through the "TemplatePath" variable in the IPP/PCF.
- If neither of these is set then the engine will look for the local copy of PRPOSTIA.INI in the \BIN directory. The format of this file is below. You can access this file directly with a ASCII editor, or push the SETUP button, which automatically loads the file into Notepad.

Note: If you supply the file through the Template or TemplatePath command in the PCF then it must have the same format as PRPOSTIA.INI documented below.

### "PRPOSTIA.INI"

#### [PrimePostIA]

Version=3.80  
LogFile=0  
Error Reporting=0  
Debug=0

#### [Save Output]

OutputFormat=11  
OutputLocal=

#### [PostApp]

ApplicationType=

#### [PrimeLexicalPlus]

UserDictionaryPath=  
SpellCheck=1  
SpellGoodConfidence=8  
LexicalCheck=1  
LexicalBad/FixConfidence=6  
LexicalBad/NoFixConfidence=2  
ScoreThreshold=85  
LexicalLog=1

#### [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  
ProportionalSansSerifFont=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 "PRPOSTIA.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.

### **OutputFormat**

This tells PrimePost which output format to use:

"0"=ASCII text file

"1"=FASCII

"2"=PDA2 (Recommended for output going to IA Text Export Module)

"3"=PRO

"9"=RTF

"11" = PDF Normal (Recommended for output going to IA PDF Export Module)

"12" = PDF Image Only (Recommended for output going to IA PDF Export Module)

"13" = PDF Image with Hidden Text (Recommended for output going to IA PDF Export Module)

"15" = Comma delimited output. The output of each zone is seperated 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)

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

### **OutputLocal**

Blank= no local output

or

FullPath\FileName.Extension=a copy of output file will be placed in FullPath\FileName.Extension as noted below.

FullPath: must be available to PC performing OCR.

FileName: six digits will be added to the file name. This number will increment with each file processed.

Extension: will be applied to each output file.

For example, if "OutputLocal=c:\test\pp.rtf", then first output file will be saved to c:\test\pp000000.rtf.

Second file will be c:\test\pp000001.rtf. If files must be 8.3 then do not supply more than two characters for file name.

### **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.**

### **UserDictionaryPath**

Full path to user dictionary file used for lexical check

Format of file is ASCII text, one word per line.

Place words typical to user's application, but not common in every day language, in this file to further improve accuracy.

Number of words in file is only limited by available RAM, however, more words in file will slow processing slightly.

### **SpellCheck**

0 - turns off spell check

1 - spell check on. Confidence level is taken from LexicalPlus value in template.

Any word containing one or more characters at this confidence level or below will be "suspect" and will be checked via a sophisticated spell check process.

A higher number is more aggressive at applying spell check to more and more words in document.

This number can be higher than LexicalCheck below, since no changes are made to characters, and any confidence changes are likely to be correct.

### **SpellGoodConfidence**

0 - no changes to confidence values if suspect word is lexically OK as is

1-9, default=8

Character confidence values are increased proportional to this weight factor.

One use of this feature is to reduce "suspicious" characters so that later manual verification effort is reduced.

### **LexicalCheck**

0 - turns off lexical plus checking

1 - lexical plus checking on. Confidence level is taken from LexicalPlus value in template.

Any word containing one or more characters at this confidence level or below will be "suspect" and checked by lexical process.

A higher number is more aggressive at applying lexical check even to characters more likely to be correct.

Be cautious in increasing this value above 6 or 7, unless your application lends itself clearly to lexical processing.

### **LexicalBad/FixConfidence**

0 - no changes to confidence values if suspect word is changed by lexical process

1-9, default=6

Character confidence values are increased proportional to this weight factor.

Use this feature, for example, to reduce "suspicious" characters during later manual verification.

### **LexicalBad/NoFixConfidence**

0 - no changes to confidence values if suspect word is not changed

1-9, default=2

Character confidence values are set to this level.

Use this feature, for example, to flag entries for later manual verification.

### **ScoreThreshold**

60-98, default=85

Possible solutions from the lexical process with a score above this number will replace the original word. A higher number is less aggressive at using lexical process to replace recognized characters.

**LexicalLog**

0=No, 1=Yes

Logs suspect words and changes in ...lexical.log

**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"

## ProportionalSansSerifFont

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

## Error Handling

If the Error Reporting flag is set to "0", then non-fatal error messages will be written to the "PRPOSTIA.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 PRPOSTIA.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

# Post OCR Processing

In addition to converting PRO files to output formats, PrimePost can be used to apply custom algorithms to further enhance the data, or to convert it to custom formats. This functionality is available in the [PostApp]\ApplicationType= entry in the PRPOSTIA.INI file. An example is provided below of such an algorithm. Each algorithm is custom so there is no standard documented list of algorithms that can be published. If you think your application would benefit from this capability please call Prime Recognition.

## **Example PrimePost IA 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.)