ABBYY

ABBYY FineReader Engine 12 for Windows

Release 4

Release Notes

Table of Contents

Technical information

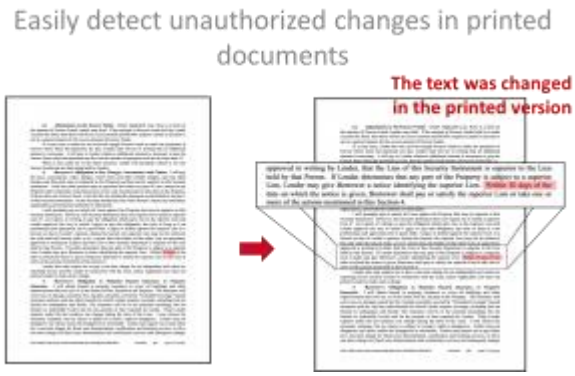
Release	Part #	Build #	OCRT build #	Release date
Release 4 Update 3	**1342/69**	**12.4.7.63**	**16.1.718.38**	**2020.09.04**
Release 4 Update 2	1342/60	12.4.7.40	16.1.718.30	2020.04.06
Release 4 Update 1	1342/47	12.4.7.23	16.1.718.19	2019.11.28
Release 4	1342/44	12.4.7.0	16.1.718.9	2019.10.10

New features

New: 'Compare Documents' Module

- To allow a user to quickly verify document's integrity, the new 'Compare Documents' Module in ABBYY FineReader Engine enables detecting content differences between two versions of the same document. The module works with documents in different formats such as Microsoft Word or PDF as well as with document images such as JPEG or TIF and many other formats, and can compare documents in all OCR languages supported by FineReader Engine 12. The results of the document content comparison are available through the API and can be as well delivered as Microsoft Word document with tracked changes that makes the content inconsistencies clearly visible.

- This feature will be of a great value for any business customers as it allows very quick detection of possible manipulations in document content - for example when comparing the originally created contract and its printed and signed version.



- For easy implementation and demonstration, a new ready-to-use code sample with sample documents is available and can be used in own applications to speed up development work. The sample compares the selected files and, if necessary, saves the detected differences to a file of the specified format. The comparison result contains the information about differences in the textual content, what type of modification was detected (deleted, inserted or modified text) and locations of the modification in the original and its copy. This sample supports Russian and English languages only. (The 'Compare Documents' functionality is available for documents in all languages supported by FineReader Engine.)

Improved: Korean and Arabic OCR with new AI-based algorithms

- **Arabic OCR: Significantly improved recognition on low quality images with a new Recurrent Neural Network**.
  - To significantly increase the recognition accuracy of Arabic especially on low quality images (where the 'traditional OCR approach' might not deliver sufficiently accurate results), a new **Recurrent Neural Network (RNN)** for recognition of Arabic was trained on an extensive amount of documents.
  - Deployment of this new AI-based technology enables the **End-to-End text recognition** that provides very high accuracy results. During the End-to-End text recognition process, the pre-trained Recurrent Neural Network automatically recognizes text 'as a whole' based on his 'knowledge' that was acquired during the training - without dividing the text strings into individual characters first.
  - To deliver the optimal balance between the processing speed and the recognition accuracy, an intelligent built-in classifier analyses the document prior to the actual recognition step, and selects the appropriate recognition methodology for each particular text snippet (faster 'traditional OCR' - or slower but more accurate 'end-to-end OCR').
- **Korean OCR: Significantly better recognition results with a new Deep Learning Language model**.
  - To further increase the recognition accuracy of Korean, a new Deep Learning Language model was trained on a large document amount.
  - Following the actual text recognition step, this model (trained and deployed for recognition of the Korean language) analyses the recognition hypothesis and selects the best 'word recognition variant' among the individual recognition hypotheses - in some cases the model even generates a new recognition hypothesis based on a context. In this case, the preceding and following words will be analyzed. Based on these words, the new recognition hypothesis will be created.
  - To optimize the balance between recognition accuracy and speed, a smart built-in classifier decides on the necessity to deploy this new Deep Learning Language model (the Deep Learning Language model is more accurate but slower than the 'traditional' evaluation of recognition hypothesis and is therefore not used as default). The feature works in 'normal' recognition mode only.

Enhanced: Text-based classifier with advanced security of training data

- To train and optimize the text-based classifier, documents representing each document category must be imported. In order to protect data contained in these training documents, hashing algorithms impede the possibility to recover information from the sample documents.
- As the training algorithms use only information from the checksums of the documents, the pre-trained text-based classifier can be used by other users (and its quality can be further optimized by re-training it on their documents) - without any risk of detecting information in the documents originally used for its training.
- **Note**: The provided API allows adding information to each individual training object. During the training, error messages can deliver useful information about the particular source files. When using this option during the creating, training and testing the text-based classifier, it is necessary to rebuild

the pre-trained text-based classifier without the information about individual training objects prior to delivering it to its final users in order to maintain high-level security and confidentiality.

Enhanced: Classification Demo Sample - now with Office format documents

- ABBYY FineReader Engine is able to process PDFs, scanned or photographed document images as well as documents available in Office formats. To reflect this capability in the classification process, the provided pre-compiled Demo Sample for classification was enhanced and allows now to import Office document in addition to PDFs and image formats.

In addition, new sample documents for the image-based classifier were included and can be used to demonstrate the classification capabilities.

Improved: Document layout preservation

- To improve the detection and recreation of document layout, a new 'single-column' document model was introduced that provides more exact detection and analysis of tables and charts.

- The new 'single-column' analysis is a key subtask of the complete document analysis process that uses specific algorithms for analyzing and processing document columns - objects that are linearly arranged from top to bottom and vertically separated. Such objects can contain:

  - continuous text block
  - table of contents
  - picture
  - chart
  - table
  - screenshot
  - agglomerate of independent cells

- The new 'single-column' analysis will significantly improve the detection and recreation of document layout and works in default (normal) document analysis mode.

Enhanced: Java wrapper documentation

- To simplify the usage of the API, the documentation of ABBYY FineReader Engine 12 has been extended and the documentation for the Java wrapper is now provided in JavaDoc format in addition to the HTML and PDF formats.

New code sample for document comparison

- The extensive code samples library was extended by a new sample that allows testing and demonstrating the ability to compare two versions of the same document and detect differences in their content.

Support of the latest .NET Framework versions

- The distributive now includes .NET COM Interop wrappers for the .NET Framework versions:
  - 3.5
  - 4.0
  - 4.5

- 4.6
- 4.7
- 4.8

New operating system support: Microsoft Windows Server 2019

- FineReader Engine 12 Release 4 was tested on Windows Server 2019. This is the first release officially tested on this platform.

Improved: Developer's Help documentation

- The Developer's Help of the FineReader Engine 12 has been extended by additional information about different possibilities of licensing the SDK, describing the individual types of licensing options in an easy-to-understand comparison table.

Bug fixes

Issue description
It finds vertical CJK text with ProhibitCJKColumns = true.
IImageDocument::Modify() works 7 minutes with IImageModification::AddRemoveGarbageRegion set up and the garbage size = 1. The image has a lot of small noise.
Zero coordinates for symbols "(" and ")" in recognition results for Japanese language. There is a postprocessing of recognition results so that parentheses are paired. Recent changes led to the coordinates loss.
Built-in version of APDFL rasterizes PDF file content incorrectly: almost all text is mossing.
Errors in recognition of superscript and subscript on customer's images. One page still has issues after the fix.
The symbol "…" is lost during E13B text recognition.
Table borders are vanished in TextOnImage mode of PDF export.
Incorrect table analysis on customer's images.
Incorrect output to an ImageOnText PDF file with the following settings: • IPDFExportParams:MRC_Always • IPDFMRCParams::UseMultipleMasks = true
The article "*Installing the ABBYY FineReader Engine Library in Automatic Mode*" misses description for the parameter ALLUSERS and does not mention that by default the installation is for the CURRENT_USER.
PaperSizeModeEnum.PSM_ImageSize has ambiguous description: one can treat it as a size of a source image whereas it is the size of a pre-processed image.

Update 1

Technical information

Release	Part #	Build #	OCRT build #	Release date
Release 4 Update 1	1342/47	12.4.7.23	16.1.718.19	2019.11.28

New features

Enhanced: MRZ recognition

- The MRZ extraction function was enhanced by new document format enums:

  - **MF_Passport** - MRZ format by the TD3 standard of the ICAO specification.
  - **MF_OfficialTravelDocumentThreeLines** - MRZ format by the TD1 standard of the ICAO specification.
  - **MF_OfficialTravelDocumentTwoLines** - MRZ format by the TD2 standard of the ICAO specification.
  - **MF_VisaA** - MRZ format by the MRV-A standard of the ICAO specification.
  - **MF_VisaB** - MRZ format by the MRV-B standard of the ICAO specification.

  The values are needed to accurately attribute extracted data to Optional data and Personal number fields. The value **MF_ICAO** is equal to **MF_OfficialTravelDocumentTwoLines** and is left for backward compatibility.

- In addition, the **IMrzData** has received a new property '**HasChecksum'**. This new property informs the system, if there is a checksum digit for the whole document data available or not (as not all documents contain checksums).

Enhanced: Compare Documents

- In the Update 1 of the Release 4, the option '**CompareTablesSeparatelyFromText'** was added in order to further improve the detection of changes in documents that contain text areas on pages as well text areas within tables. This new option specifies whether text changes detected within tables should be displayed separately from the text modifications detected in free-flow text. By default, this property is set to FALSE.

Bug fixes

Issue description
During XML export of Compare Documents feature results the error is thrown: "CoInitialize has not been called".
IPE: "Src\WordExporter\WordDocument.Frames.Impl.h, 85" in the 'convert an image to MS Word' scenario.
IPE: "Src\WordDocument.NumberingLists.Impl.cpp, 179" during export in the Word Track Changes mode.

Issue description
IPE: "Src\WordDocument.ParagraphProcessor.cpp, 2954" during export in the Word Track Changes mode.
IPE: "Src\PdfExporter.cpp, 745" during PDF export.
Release 4 can't open FRDocument archive created by the previous version of FRE 12. Error message is "Invalid version of ..."
XML export result for Compare Documents feature does not pass XSD validation with the error: "Error: element 'Region' is not allowed for content model '(Position, Region)' Line: 823 Column: 14".
Changes inside and outside of a table are glued into one change and represented in the Word Track Changes exporting result at an incorrect position.
MRZ: '<' symbol is left between parts of the Last name ('Last<Name') and at the end of the Document Number ('DocNumber<') in the output results.
Incorrect calculation of checksums for 3-line MRZ data.
Memory leakage during PDF file processing.
Scanning parameters setting does not work for Canon DR-2580C.
Separator lines before and after the document signatures block are shortened and moved to the left of a page in the exported MS Word file.
Memory leakage in BCR scenario.
Automatic product deinstallation leaves 'Document Comparison' sample folder with content and the image set for the classification demo sample.
Runtime automatic product deinstallation does not remove ABBYY Open Office.
License Manager shows not supported modules: - File Naming - Receipt Recognition

Update 2

Technical information

Release	Part #	Build #	OCRT build #	Release date
Release 4 Update 2	1342/60	12.4.7.40	16.1.718.30	2020.04.06

New features

New: Additional 1D barcode types

- Three new 1D types of barcodes were added to the broad portfolio of supported barcodes:
  - KIX barcode

- Royal Mail 4-State barcode (RM4SCC)
- Australia Post 4-State barcode

Improved: 'Compare Documents' Module

- New options and methods were added to the Compare Documents module to improve the comparison results:
  - New comparison options:
    - '**CorrectOcrErrors**' - this property specifies if OCR errors in the text should be corrected. In case recognition variants in the two versions of the document match, the detected difference will be ignored and not reported as a text deviation.
    - '**DetectOneLetterNonDigitChanges**' - this property specifies if words differing only by one letter should be reported as deviation in the text (this option considers only words containing letters and not digits).
    - '**DetectPunctuatorChanges**' - this property specifies if differences in the punctuation should be reported as deviation in the text.
    - '**DetectRunningTitleChanges**' - this property specifies if differences in running titles should be reported as deviation in the text. If there is a text that repeats in the header or footer, the difference can be ignored.
  - New method for defining the order of text blocks to improve the accuracy of comparison results:
    - The method for defining the order of text blocks was changed: The text blocks order is now defined during the Synthesis step. The usage of the Document Analysis step for defining the text block order was discontinued (The desktop application ABBYY FineReader as well leverages the Synthesis step to define the order of text blocks in its Document Comparison module.).
  - Updated Demo Sample: The Demo Sample for the Compare Documents module was improved to provide a better user experience.

Improved: Recognition of Machine Readable Zones in ID documents (MRZ Recognition)

- New option **'WriteNondeskewedCoordinates**' for MRZ recognition allows saving recognition results from Machine Readable Zones in JSON or XML file in the coordinates of the original image (the image in its original layout, before it was deskewed or otherwise optimized for optimal OCR results) or with the coordinates of the image that was internally altered and pre-processed.

Enhanced: Java wrapper

- In Runtime as well as Developer licenses, the JNI facilitating libraries (FREngine.Jni.dll) are stored in Bin/Bin64 folders where they can be accessed by the Java wrapper. This increases security of the system as it avoids the step of unpacking the libraries from *.jar archive and temporarily storing them.

Enhanced: PDF file processing in multithreading environment

- A new option '**ProcessPdfInOneThread**' allows redirecting all PDF-processing calls to a separate thread with the initialized Adobe PDF Library. This property adds stability to PDF file processing if the Engine object is loaded using the **'InitializeEngine'** function.

Updated: Network License Manager

- The Network License Manager for the Network and the Online licenses was updated to assure security of the system.

Bug fixes

Issue description
A 'custom action' does not change a page order in Document Viewer.
Incorrect table rows analysis result on a customer's document.
An empty page with a regular background takes too long.
Veracode reports a process control vulnerability (CWE ID 114): "a function call could result in a process control attack", - for the Java wrapper.

Update 3

Technical information

Release	Part #	Build #	OCRT build #	Release date
Release 4 Update 3	1342/69	12.4.7.63	16.1.718.38	2020.09.04

New features

New: Speeding up the iteration of the recognition result in Engine Pool scenario

- This new method saves time for internal communication and increases overall processing speed in scenarios, where ABBYY FineReader Engine runs outside the main process. Combining the functions of the **'OutprocLoader'** and **'InprocLoader'** objects allows you to obtain the document layout in two steps:

  - Using the **Engine** object outside the main process for processing the document, getting the layout for each page, and writing it to a data stream with the **'SaveToStream'** method.
  - Using the **Engine** object in the main process for restoring a read-only copy of the original layout with the **'CreateLayoutFromStream'** method and further iteration of this copy.

New: .NET Core 3.1 wrapper

- To increase the efficiency of development teams using containers for software development or deployment, ABBYY FineReader Engine now offers a new pre-built .NET wrapper based on .NET Core 3.1.
- **Note**: Working with events and **'OutprocLoader'\'InprocLoader'** objects is not supported in this version of the wrapper.

Improved: 'Compare Documents' Module

- In bi-lingual documents, such as international contracts, the text and its translated version are typically arranged in two parallel columns. The new **'UseDoubleLanguageAgreementMode'** option in the 'Compare Documents' module of ABBYY FineReader Engine provides the ability to compare each column (and thus each language version) separately.
- **Note**: To prevent false positives (reported differences that are none), apply the method of overlaying a table block of size (image_width; image_height) with a vertical separator in the middle on all pages of the document.

Enhanced: New features for export formats

- ABBYY FineReader Engine 12 was enhanced by the new export options and methods:
  - New export options for PDF format:
    - '**PageOrientation**' - this property specifies the page orientation in dependence on the selected paper size.
    - Additionally, **PageOrientationEnum** has been extended by the new **POM_MostFrequent** option to set the most frequent orientation used in the document.
    - Ability to combine PDF/UA and PDF/A-1/2/3-bu standards.
  - New export options for XLSX format:
    - **'WritePictures'** and **'PictureExportParams'** – these properties can be used to set up image compression parameters.
  - New export options for DOCX format:
    - **'IncreasePaperSizeToFitContent'** – this property specifies if the page size should be increased automatically in cases where the content does not fit on the page. It takes effect if the parameters of paper size are set as user-defined.
    - **'UseCustomPageMargins'** – this property activates the option of specifying the page margins. However, it is efficient in cases where the output file synthesis mode is set to formatting the text in a single column or retaining the paragraph, font types, and sizes.
    - **'PageMargins'** – this property specifies page margins in twips in the output file.

Enhanced: New methods for customizing the recognition area

- **'IRegion'** object was extended with the new methods for customizing the blocks' regions:
  - **'AddRegion'** – this method adds a new rectangular region, which is represented by an **IRegion** object, to an existing one.
  - **'CutRect'** – this method cuts a rectangular area defined by the borders' coordinates from an existing region.

Improved: Recognition of capital letters

- The **RecognizerParams** object has received the new **ProhibitSmallCaps** option to increase the recognition accuracy of letter case and separate detection of capital letters and small caps.

Improved: Code sample for command-line interface (CLI)

- To provide the developers with the latest options of working with the documents in command-line-based applications, the **CommandLineInterface** code sample has been extended with the new keys of multi-processing, document analysis, synthesis, and export, opening images.

- The CLI code sample is a cross-platform since this release and includes the unified set of keys both for Linux and Windows.

Improved: MRZ Recognition

- The new **'MinMrzLineLength'** option in **MrzProcessingParams** specifies the minimum allowed length of the MRZ line. This property can be used to improve the recognition results for the machine-readable zone in cropped or low-quality images where the part of the MRZ line or lines might get lost during the pre-processing step and thus not comply to the MRZ specifications (in which case, the missing compliancy would lead to delivering zero as a recognition result).

Enhanced: Opening Office documents from memory

- The **'AddImageFile From Memory'** and **'AddImageFileFromStream'** methods now can be used for opening Microsoft Office and Apache OpenOffice files directly from memory, which allows you to increase the speed of the document import step and accelerate the overall document processing speed.
- **Note**: The **'KeepInMemory'** option for export is not supported.

Enhanced: Image formats for Office documents

- **ImageFileFormatEnum** was extended with new constants for Office formats to be read using ABBYY FineReader Engine: DOC, DOCX, HTML, ODP, ODS, ODT, PPT, PPTX, RTF, TXT, XLS, XLSX.
- **Note**: ABBYY FineReader Engine works with PDF copy of these formats.

Support of the latest LibreOffice versions

- For converting incoming digital documents, ABBYY FineReader Engine 12 now supports working with the LibreOffice versions: 7 (7.0), 6 (6.1, 6.2, 6.3, 6.4).

Improved: Developer's Help documentation

- Updated article about running ABBYY FineReader Engine in Docker containers with the new scenario based on using two separate containers and multi-stage builds.
- Added information about GoDaddy authorization requirements when using the Online license type.

Bug fixes

Issue description
Text on the document with a text layer partially not recognized.
Certain words recognized as Arabic instead of Hebrew in documents containing Hebrew, Arabic and English.
Receiving white screen when using the Visual Component of ABBYY FineReader Engine.
Problems with machine-readable zones extraction from Spanish ID documents.
Image format not recognized with the AddImageFileFromStream method.
Slow down when using the OutprocLoader object.
IPE textlayout\src\text\overflowedtext.cpp, 514 during recognition of document with JapaneseModern.

Issue description
The case of letters in English is not recognized correctly.

