to interpret the PDF must be contained within it. Because PDF/A disables Javascript and other types of embedded content, it is probably more secure.

There are various conformance levels and versions, such as "PDF/A-2b".

Generally speaking, the best format for scanned documents is PDF/A. Some governments and jurisdictions, US Courts in particular, mandate the use of PDF/A for scanned documents.

Since most people who scan documents are interested in reading them indefinitely into the future, OCRmyPDF generates PDF/A-2b by default.

PDF/A has a few drawbacks. Some PDF viewers include an alert that the file is a PDF/A, which may confuse some users. It also tends to produce larger files than PDF, because it embeds certain resources even if they are commonly available. PDF/A files can be digitally signed, but may not be encrypted, to ensure they can be read in the future. Fortunately, converting from PDF/A to a regular PDF is trivial, and any PDF viewer can view PDF/A.

1.4 What OCRmyPDF does

OCRmyPDF analyzes each page of a PDF to determine the colorspace and resolution (DPI) needed to capture all of the information on that page without losing content. It uses Ghostscript to rasterize the page, and then performs on OCR on the rasterized image to create an OCR "layer". The layer is then grafted back onto the original PDF.

While one can use a program like Ghostscript or ImageMagick to get an image and put the image through Tesseract, that actually creates a new PDF and many details may be lost. OCRmyPDF can produce a minimally changed PDF as output.

OCRmyPDF also some image processing options like deskew which improve the appearance of files and quality of OCR. When these are used, the OCR layer is grafted onto the processed image instead.

By default, OCRmyPDF produces archival PDFs – PDF/A, which are a stricter subset of PDF features designed for long term archives. If regular PDFs are desired, this can be disabled with `--output-type pdf`.

1.5 Why you shouldn't do this manually

A PDF is similar to an HTML file, in that it contains document structure along with images. Sometimes a PDF does nothing more than present a full page image, but often there is additional content that would be lost.

A manual process could work like either of these:

1. Rasterize each page as an image, OCR the images, and combine the output into a PDF. This preserves the layout of each page, but resamples all images (possibly losing quality, increasing file size, introducing compression artifacts, etc.).
2. Extract each image, OCR, and combine the output into a PDF. This loses the context in which images are used in the PDF, meaning that cropping, rotation and scaling of pages may be lost. Some scanned PDFs use multiple images segmented into black and white, grayscale and color regions, with stencil masks to prevent overlap, as this can enhance the appearance of a file while reducing file size. Clearly, reassembling these images will be easy. This also loses and text or vector art on any pages in a PDF with both scanned and pure digital content.

In the case of a PDF that is nothing other than a container of images (no rotation, scaling, cropping, one image per page), the second approach can be lossless.

OCRmyPDF uses several strategies depending on input options and the input PDF itself, but generally speaking it rasterizes a page for OCR and then grafts the OCR back onto the original. As such it can handle complex PDFs and still preserve their contents as much as possible.