**Optical character recognition** or **optical character reader** (**OCR**) is the [electronic](https://en.wikipedia.org/wiki/Electronics) or [mechanical](https://en.wikipedia.org/wiki/Machine) conversion of [images](https://en.wikipedia.org/wiki/Image) of typed, handwritten or printed text into machine-encoded text, whether from a scanned document, a photo of a document, a scene photo (for example the text on signs and billboards in a landscape photo) or from subtitle text superimposed on an image (for example: from a television broadcast)

A hybrid combination of OCR (Optical Character Recognition) and ICR (Intelligent Character Recognition) can be a powerful approach to handle documents that contain both printed and handwritten text, such as forms, applications, or contracts with signatures. This combination allows you to leverage the strengths of both technologies to achieve accurate and comprehensive text extraction and processing. Here's how you might implement a hybrid approach:

\*\*1. Identify Document Sections:\*\*

Determine which parts of the document contain printed text and which parts contain handwritten text, such as signatures or annotations.

\*\*2. OCR for Printed Text:\*\*

Apply OCR technology to the sections of the document that contain printed or typewritten text. OCR will accurately recognize and extract the machine-readable content.

\*\*3. ICR for Handwritten Text:\*\*

Apply ICR technology to the sections of the document that contain handwritten text, such as signatures. ICR will recognize and extract the handwritten characters, including signatures.

\*\*4. Data Integration:\*\*

Combine the extracted text from both OCR and ICR processes to create a comprehensive representation of the document's content. This integration can be done programmatically.

\*\*5. Post-Processing:\*\*

Perform any necessary post-processing steps to clean and validate the extracted text. This may involve error correction, data validation, and formatting adjustments.

\*\*6. Document Verification:\*\*

For documents with signatures, you can use the recognized signatures for verification purposes, comparing them against stored reference signatures to authenticate the document.

\*\*7. Data Analysis and Usage:\*\*

Utilize the extracted and integrated data for various purposes, such as data entry, document indexing, analysis, and archiving.

\*\*Benefits of a Hybrid Approach:\*\*

- \*\*Accuracy:\*\* OCR and ICR each excel in their respective domains, leading to higher accuracy for printed and handwritten text recognition.

- \*\*Comprehensive Processing:\*\* You can handle a wider range of document types that contain mixed content.

- \*\*Efficiency:\*\* Automating the extraction process for both printed and handwritten text saves time and effort compared to manual data entry.

- \*\*Enhanced Document Understanding:\*\* The hybrid approach provides a more complete representation of the document's content, enabling better context analysis.

\*\*Use Cases:\*\*

- Processing forms with printed fields and handwritten responses.

- Extracting information from contracts with printed clauses and handwritten signatures.

- Analyzing medical records with printed text and handwritten doctor's notes.

It's important to note that while a hybrid approach can be powerful, it may require additional preprocessing, validation, and handling of potential conflicts between OCR and ICR results. The specific implementation details will depend on the technologies and tools you choose, as well as the characteristics of the documents you are working with.

Data has reached a saturation point. Organizations today often have so much data that it’s difficult to decipher what’s valuable information and what’s just noise. Mining vast amounts of data for actionable insights can be costly and time-consuming. That’s where intelligent document processing, or IDP, comes into play.

Intelligent document processing (IDP) is a [workflow automation](https://powerautomate.microsoft.com/en-us/workflow-automation/) technology that scans, reads, extracts, categorizes, and organizes meaningful information into accessible formats from large streams of data. The technology can process many different types of documents: papers, PDFs, Word docs, spreadsheets, and a multitude of other formats. The primary function of IDP is to extract valuable information in large sets of data without human input.

There are several advantages to automating some of your document processing through IDP. Through artificial intelligence, IDP can eliminate the need for manual data entry and processing workers. This not only increases the speed at which data can be processed but decreases the cost to do so—while also reducing human error for greater efficiency.

Modern organizations typically have high volumes of three types of data: structured, unstructured, and semi-structured. Structured data is organized and more easily read by human data processors. Unstructured data, on the other hand, is time-consuming to process and analyze. Semi-structured data falls in between. Intelligent document processing solutions are capable of automating data processing for structured, unstructured, and semi-structured data.

**Benefits of intelligent document processing**

Intelligent document processing has been demonstrated to improve business processes and increase team efficiency. Here are a few of the key advantages to automating your document processing:

* **Increase the accuracy** of your processing with intelligent document processing. The rate of error for human data processors is much higher than it is for IDP.
* **Lower the cost** of processing documents by automating some of your workflows. On average, manual document processing costs 6 to 8 dollars per document. The per-document cost of automated document processing is considerably lower than that.
* **Increase employee productivity** by offering them the opportunity to do more impactful work. Without the need to process mindlessly time-consuming documents, employees are free to direct their attention elsewhere.

### **Step 1 - Document preprocessing**

Where there is data extraction, there is OCR. As a document is ingested into a document processing solution, it goes through the first phase of document pre-processing in IDP workflow. The overall accuracy of OCR depends on how accurately it can identify/distinguish a character/word from the background. Some of the basic techniques employed in this phase are:-

#### **Binarization**

**‍**In simple terms, binarization is the technique to convert a colored image into a black and white pixels. Now, the image consists of only 2 kinds of pixels - black pixel value = 0 & white pixel value = 256. The aim is to create a binary and distinction between the characters to be read in a text file(black pixels) and background(white pixels)**‍**

#### **Deskewing**‍

While scanning a document, the scanned image might be slightly aligned horizontally, which is not ideal for OCR. Techniques such as Projection profile method, Hough transformation method, and Topline method are used for skew correction.

#### **Noise removal**‍

The aim of this step is to get rid of any unwanted small dots/patches so that OCR doesn’t confuse these dots with characters.

### **Step 2 - Document classification**

Document classification happens in 3 steps:-

#### **Identify the format**‍

Figure out whether the file is a pdf document, JPG, PNG, TIFF, or any other file format.

#### **Identify the structure**

**‍**The OCR solution tries to differentiate amongst structured, semi-structured, and unstructured documents.Structured documents have a fixed template and layout, whereas semi-structured documents have some form of structure in it that means they may contain similar information at different locations in the document. An invoice is a great example of semi-structured documents - vendor’s address in different invoices can be at different places. To make sense of these values, the document processing solution should have some kind of contextual understanding of data and the document.  
     Unstructured documents have hardly any structure yet organizations need to extract data from them for various purposes. In an unstructured document, sometimes certain values may not have any key assigned to them - such as dates or email addresses may be there on documents but without any key identifier such as “Date” or “Email”.  A contract is a good example of an unstructured document.

#### **Identify the document type**‍

The third step of document classification is trying to figure out the document type, that is, to find out whether the ingested document is an invoice, bank statement, t12 statement, shipping label, or any other document. The ability to identify a document type successfully and queue it for data extraction depends on the data already fed into the IDP solution.

### **Step 3 - Data extraction**

There are mostly two parts of data extraction:-

**i)** **Key-value pair extraction** - Extracting the values assigned to unique key identifiers in a documents

**ii)** **Table extraction** - Extracting line items arranged in a table form

There are certain ways to do it:- **‍**

#### **OCR**‍

OCR is the first step of data extraction. As essential as this step is, there are certain errors that can happen during OCR:-

* **Error in word detection - Failing to detect a text block in the image, an error commonly caused because of the poor image quality.**
* **Error in word segmentation – Interpreting a word incorrectly, due to wrong interword space detection, different text alignments, and spacing.**
* **Error in character segmentation – Unable to detect single characters in a segmented word. This is frequent for cursive or connected alphabets.**
* **Error in character recognition – Failing to identify the right character in a bounded character image.**

These errors could be rectified by dictionary look-up, k-mer, and n-gram language models.

#### **Rule based extraction**‍

Rule based models work well for structured and semi-structured documents. These models can identify key-value pairs/line items by taking a position reference in a document. Named-entity recognition approach and n-gram model come handy in identifying a value assigned to a key identifier. For example, no matter the position of invoice number in an invoice, a set of strings next to “Invoice Number” or “Invoice No” is the value the model is looking for.

#### **Learning based approach**

Deep learning and ML-based OCR-hybrid data extraction techniques need supervised/unsupervised learning to train their models on. The efficiency of these models are determined by the accuracy rate and confidence score. With the increasing number of documents processed and the amount of training and feedback provided, the model grows in accuracy. Docsumo takes the similar approach of data extraction where an ML-based model sits on the top of  template based OCR. At Docsumo, Simple OCR correction approach along with context based NLP is used to improve the accuracy and the quality of data.

### **Step 4 - Data validation**

This step is crucial in detecting the inaccuracies of the extracted data. Certain data validation rules are applied within the document so that any inaccuracy could be detected and flagged for correction. For example, the ‘total amount payable’ in an invoice should be a sum of ‘subtotal’ and ‘tax payable’. If there’s any discrepancy between two, the invoice gets flagged and held for review.

### **Step 5 - Human review**

Although we’d like it to be, no data extraction model is 100% accurate, hence a layer of human intervention is there in the IDP workflow. Any document flagged red is reviewed by a human-in-the-loop. This is especially helpful in the supervised learning of the model and improving the accuracy of the model. The more documents are processed and reviewed, the more improves the accuracy of the data extraction model.

Once the data is extracted and cleaned up, the software can push to the database or export it in multiple formats. IDP workflows let users convert documents into different formats such as JSON, XML, PDF, etc.

## Intelligent document processing use-cases (by capability)

IDP solutions have following capabilities to offer:-

### **Identify difficult content**

IDP can decipher content from substandard quality documents, which cannot be read by traditional OCR. AI-driven IDP solutions can also decipher the relevance of a character/word based on the context and defined rules which is not possible with traditional OCR solutions.

### **Reading barcodes/QR codes**

IDP solutions are capable of [processing barcodes/QR codes](https://www.docsumo.com/blog/barcodes-pdf).

### **Document auto-classification**

Multiple document types coming from different streams and sources in different forms - IDP solutions are capable of [auto-classifying documents](https://www.docsumo.com/blog/auto-document-classification) in different classes for further processing.**‍**

### **Extract information you need**

IDP models can be trained to extract specific information in the document.

### **Validate data**

Processed data can be validated against the rules set to determine the accuracy of the system and to assign the document for manual review.

### **Organize data and reports**

IDP solutions make it easier to classify different documents, extract data from multiple sources, and assemble it at one place for further analysis.

## IDP use-cases (by industry)

Based on the above capabilities, IDP solutions find different use-cases in different industries:-

