# Processing Documents Synchronously

Amazon Textract can detect and analyze text in single-page documents that are provided as images in JPEG, PNG, PDF, and TIFF format. The operations are synchronous and return results in near real time. For more information about documents, see Text Detection and Document Analysis Response Objects.

This section covers how you can use Amazon Textract to detect and analyze text in a single-page document synchronously. To detect and analyze text in multipage documents, or to detect JPEG and PNG documents asynchronously, see Processing Documents Asynchronously.

You can use Amazon Textract synchronous operations for the following purposes:

- Text detection – You can detect lines and words on a single-page document image by using the DetectDocumentText operation. For more information, see Detecting Text.
- Text analysis – You can identify relationships between detected text on a single-page document by using the AnalyzeDocument operation. For more information, see Analyzing Documents.
- Invoice and receipt analysis – You can identify financial relationships between detected text on a single-page invoice or receipt using the AnalyzeExpense operation. For more information, see Analyzing Invoices and Receipts
- Identity document analysis – You can analyze identity documents issued by the US Government and extract information along with common types of information found on identity documents. For more information, see Analyzing Identity Documents.

**Topics**

## Calling Amazon Textract Synchronous Operations

Amazon Textract operations process document images that are stored on a local file system, or document images stored in an Amazon S3 bucket. You specify where the input document is located by using the Document input parameter. The document image can be in either PNG, JPEG, PDF, or TIFF format. Results for synchronous operations are returned immediately and are not stored for retrieval.

For a complete example, see Detecting Document Text with Amazon Textract.

## Request

The following describes how requests work in Amazon Textract.

### Documents Passed as Image Bytes

You can pass a document image to an Amazon Textract operation by passing the image as a base64-encoded byte array. An example is a document image loaded from a local file system. Your code might not need to encode document file bytes if you're using an AWS SDK to call Amazon Textract API operations.

The image bytes are specified in the `Bytes` field of the `Document` input parameter. The following example shows the input JSON for an Amazon Textract operation that passes the image bytes in the `Bytes` input parameter.

```
{
    "Document": {
        "Bytes": "/9j/4AAQSk....."
    }
}
```

**Note**

If you're using the AWS CLI, you can't pass image bytes to Amazon Textract operations. Instead, you must reference an image stored in an Amazon S3 bucket.

The following Java code shows how to load an image from a local file system and call an Amazon Textract operation.

```
String document="input.png";

ByteBuffer imageBytes;
try (InputStream inputStream = new FileInputStream(new File(document))) {
    imageBytes = ByteBuffer.wrap(IOUtils.toByteArray(inputStream));
}
AmazonTextract client = AmazonTextractClientBuilder.defaultClient();

DetectDocumentTextRequest request = new DetectDocumentTextRequest()
        .withDocument(new Document()
                .withBytes(imageBytes));


DetectDocumentTextResult result = client.detectDocumentText(request);
```

## Documents Stored in an Amazon S3 Bucket

Amazon Textract can analyze document images that are stored in an Amazon S3 bucket. You specify the bucket and file name by using the S3Object field of the `Document` input parameter. The following example shows the input JSON for an Amazon Textract operation that processes a document stored in an Amazon S3 bucket.

```
{
    "Document": {
        "S3Object": {
            "Bucket": "bucket",
            "Name": "input.png"
        }
    }
}
```

The following example shows how to call an Amazon Textract operation using an image stored in an Amazon S3 bucket.

```
String document="input.png";
String bucket="bucket";

AmazonTextract client = AmazonTextractClientBuilder.defaultClient();

DetectDocumentTextRequest request = new DetectDocumentTextRequest()
        .withDocument(new Document()
                .withS3Object(new S3Object()
                        .withName(document)
                        .withBucket(bucket)));

DetectDocumentTextResult result = client.detectDocumentText(request);
```