

Invoice GPT Processor

Blake Kellett

blakekellett10@gmail.com

July 24, 2024



Main Street Restaurant
6332 Business Drive
Suite 528
Palo Alto California 94301
575-1628095

Fri 04/07/2017 11:36 AM

Merchant ID: 9hqjxvufdr
Terminal ID: 11111

Transaction ID: #e6d598ef
Type: CREDIT

PURCHASE
Number: XXXXXXXXXXXX0041
Entry Mode: Swiped
Card Type: DISCOVER

Response: APPROVED
Approval Code: 819543

Sub Total USD\$ 25.23
Tip: 3.78

Total USD\$ 29.01

Thanks for supporting
local business!

THANK YOU



Tesseract OCR



Capturing and processing receipts and invoices can be a tedious task. However, several tools are available in 2024 to automate this process. This paper demonstrates how to take an invoice image and extract useful data from it.

Objective

The goal is to supply an invoice image and extract information commonly needed by a bookkeeper.

Technologies Used

- **OpenCV**: For image processing.
- **Tesseract**: For Optical Character Recognition (OCR).
- **OpenAI API**: For language model-based data extraction.

Step-by-Step Process

1. Installing Dependencies

```
pip install opencv-python pytesseract openai numpy
```

2. Obtaining an OpenAI API Key

To use the OpenAI API, you need to sign up for an API key. Follow these steps:

1. Go to the OpenAI website.
2. Sign up for an account if you don't have one.
3. Navigate to the API section.
4. Generate a new API key.
5. Save the API key securely; you will need it to access the OpenAI API.

3. Importing Libraries

First, we need to import all necessary libraries.

```
import cv2
import pytesseract
import numpy as np
from openai import OpenAI
```

4. Image Processing

We need to process the image to remove noise and enhance it for better OCR performance. The steps involved are:

- **Convert to Grayscale**: OCR does not consider color, and removing color reduces the image size.
- **Convert to Black and White**: This creates a high-contrast image for better text extraction.
- **Remove Noise**: Eliminates random pixel data that do not provide useful information, such as backgrounds and textures.

```

# Read the image from file
image = cv2.imread('image.jpg')

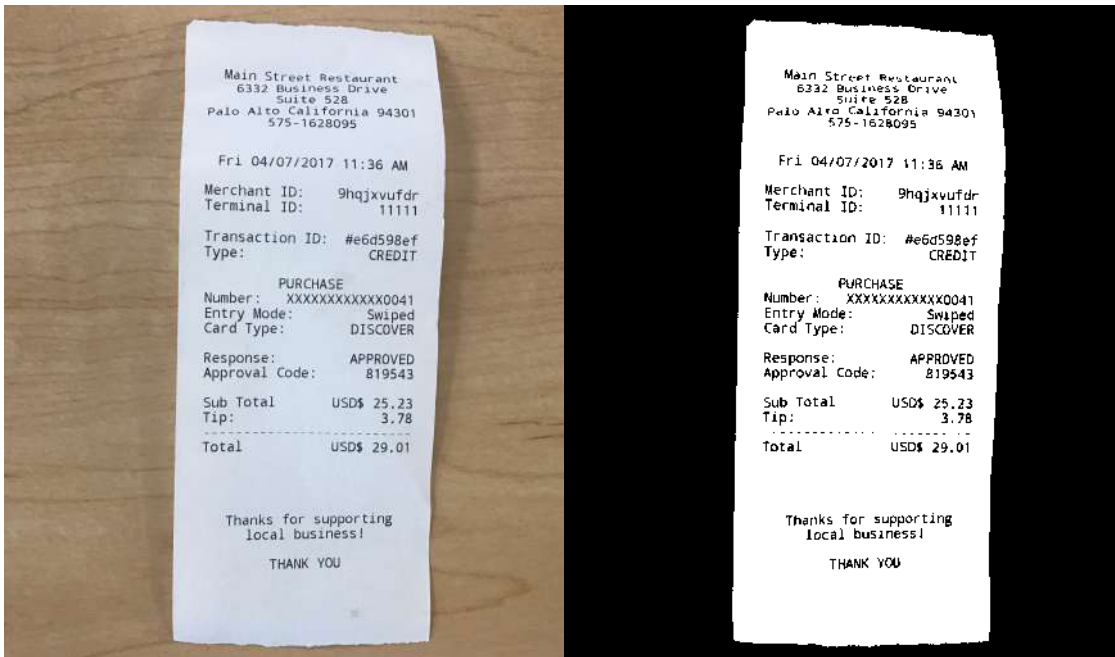
# Convert to grayscale
gray_image = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)

# Apply binary thresholding
_, binary = cv2.threshold(gray_image, 150, 255, cv2.THRESH_BINARY + cv2.THRESH_OTSU)

# Noise removal with median blur
median_image = cv2.medianBlur(binary, 3)

```

Results:



5. Image OCR

With the processed image, we can use a pre-trained OCR model like Tesseract to extract text.

```

# Extract the text characters
ocr_text = pytesseract.image_to_string(median_image)

# Print the output
print(ocr_text)

```

Output:

```

...
Main Street Restaurant
6332 Business Drive
Suite 528
Palo Alto California 94301
575-1628095

```

Fri 04/07/2017 11:36 AM

Merchant ID: 9hqjxvufdr

Terminal ID: 11111

Transaction ID: #e6d598ef

Type: CREDIT

PURCHASE

Number: = XXXXXXXXXXXXX0041

Entry Mode: Swiped

Card Type: DISCOVER

Response: APPROVED

Approval Code: 819543

Sub Total USD\$ 25.23

Tip:

USD\$ 29.01

Thanks for supporting

local business!

THANK YOU

^^^

6. OpenAI GPT Extraction

Finally, we perform data extraction using a Large Language Model (LLM) from OpenAI. We will:

- Compile a prompt for the GPT model.
- Supply the OCR text and the desired output structure.
- Query the OpenAI API.

```
# Create the prompt for GPT
prompt = f"""
Extract the following information from the given invoice text:
{ocr_text}

Return a JSON object in the following form:
{{
  "supplier_name": "",
  "total": {{
    "subtotal": 0.00,
    "tax": 0.00,
    "total": 0.00
  }},
  "dates": {{
    "date_issued": "YYYY-MM-DD",
    "date_due": "YYYY-MM-DD"
  }}
}}
```

```

    }}
}},
"""

# Create an openAI client
client = OpenAI(
    api_key="xxx-your-api-key"
)

# Prompt the client
response = client.chat.completions.create(
    model="gpt-4o-mini",
    messages=[
        {"role": "user", "content": prompt}
    ]
)

# Print the response
print(response.choices[0].message.content)

```

Output:

```

{
  "supplier_name": "Main Street Restaurant",
  "total": {
    "subtotal": 25.23,
    "tax": 0.00,
    "total": 29.01
  },
  "dates": {
    "date_issued": "2017-04-07",
    "date_due": ""
  }
}

```

Conclusion

By following the above steps, we can automate the extraction of key information from invoices.