DOCUMENT SCANNER

Motivation behind the project selection:

Document scanning services provide an affordable and convenient solution for the conversion of paper records into a digital file format. Whether it is a one-time imaging project or require ongoing document scanning, we guarantee the timely and precise conversion of the paper documents and files.

Being Computer Vision enthusiasts, we thought of building a pretty unsophisticated and rustic implementation of a document scanner using OpenCV and Python.

Aim:

To run and generate a scanned copy of a paper image by running a code written in python, which is used efficiently for the digital storage of files.

Existing system:

Image capture:-

- An image is captured using either a mobile or a webcam.
- The original image is resized and scaled down as OpenCV's methods may not perform accurately for very large dimensions.
- It is then converted to gray scale and then blurred using Gaussian Blur technique.
- By blurring, we create smooth transition from one colour to another and to reduce noise and edge content.
- But, we have to be careful with the extent of blur as we want our script to detect edges of the document.

Proposed system:

Edge Detection:-

- Edge detection technique is used to find boundaries of objects in an image by analyzing varying brightness in the image.
- Here, it is being used for segmenting image.
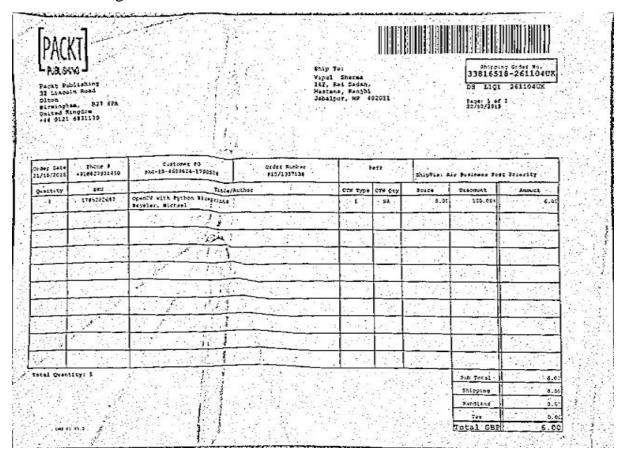- More precisely we'll use Canny Edge Detection technique.

Contour Detection:-

- After performing Edge Detection, we'll try to extract the document to be scanned from the image. Therefore, we'll find the document by drawing contours around the edges detected and choose the appropriate contour.
- Here, we can see that there is a boundary traced along the edges of our document but there are some other irrelevant contours too.

- Also, it is clearly visible that the area within the contour of the document is larger than the area enclosed by any other contours and we can use this fact to get the right boundary to extract our document.
- Let's get rid of the extraneous contours by selecting the contour of largest area.
- To get a boundary with only 4 vertices, we have approximated the contour; which means, to approximate a contour to another shape which has a less number of vertices.

Perspective Transformation:-

- Here, one arbitrary quadrilateral is mapped to another and hence, a skewed image (quadrilateral) can be transformed into a square/rectangle by defining a new mapping for each pixel.
- If we threshold the image using Adaptive Gaussian thresholding method we can get a B/W document.

Requirement specifications:

1. Python 3.6 or higher

2. Pycharm IDE

3. OpenCV library