

Visvesvaraya Technological University

Jnana Sangama, Belagavi - 590018



A Project Work Phase-I (17CSP78)

Report on

“API for Document Resizing, Validation and Verification”

*Project Report submitted in partial fulfilment of the requirement for the
award of the degree of*

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

Submitted by

BHAVESH BHANSALI	1KS17CS016
GANESH MAUDGHALYA H G	1KS17CS025
KARAN RAGHUNATH	1KS17CS032
P KISHORE	1KS17CS051

Under the guidance of

Mrs. Vaneeta M

Associate Professor

Department of Computer Science & Engineering

K.S.I.T, Bengaluru-560109



KSIT
K. S. INSTITUTE OF TECHNOLOGY

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

K. S. Institute of Technology

#14, Raghuvanahalli, Kanakapura Road, Bengaluru - 560109

2020 - 2021

K. S. Institute of Technology

#14, Raghuvanahalli, Kanakapura Road, Bengaluru - 560109

Department of Computer Science & Engineering



CERTIFICATE

Certified that the Project Work Phase-I (17CSP78) entitled “**API for Document Resizing, Validation and Verification**” is a bonafide work carried out by:

BHAVESH BHANSALI	1KS17CS016
GANESH MAUDGHALYA H G	1KS17CS025
KARAN RAGHUNATH	1KS17CS032
P KISHORE	1KS17CS051

in partial fulfilment for VII semester B.E., Project Work in the branch of Computer Science and Engineering prescribed by **Visvesvaraya Technological University, Belagavi** during the period of September 2020 to January 2021. It is certified that all the corrections and suggestions indicated for internal assessment have been incorporated. The Project Work Phase-I Report has been approved as it satisfies the academic requirements in report of project work prescribed for the Bachelor of Engineering degree.

.....
Signature of the Guide

[Mrs. Vaneeta M]
Associate Professor

.....
Signature of the HOD

[Dr. Rekha B. Venkatapur]

.....
**Signature of the Principal &
CEO**

[Dr. K.V.A. Balaji]

DECLARATION

We, the undersigned students of 7th semester, Computer Science & Engineering, KSIT, declare that our Project Work Phase-I entitled “**API for Document Resizing, Validation and Verification**”, is a bonafide work of ours. Our project is neither a copy nor by means a modification of any other engineering project.

We also declare that this project was not entitled for submission to any other university in the past and shall remain the only submission made and will not be submitted by us to any other university in the future.

Place:

Date:

Name and USN

Signature

BHAVESH BHANSALI (1KS17CS016)

.....

GANESH MAUDGHALYA H G (1KS17CS025)

.....

KARAN RAGHUNATH (1KS17CS032)

.....

P KISHORE (1KS17CS051)

.....

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompany the successful completion of any task will be incomplete without the mention of the individuals, we are greatly indebted to, who through guidance and providing facilities have served as a beacon of light and crowned our efforts with success.

First and foremost, our sincere prayer goes to almighty, whose grace made us realize our objective and conceive this project. We take pleasure in expressing our profound sense of gratitude to our parents for helping us complete our Project Work Phase-I successfully.

We take this opportunity to express our sincere gratitude to our college **K.S. Institute of Technology**, Bengaluru for providing the environment to work on our project.

We would like to express our gratitude to our **MANAGEMENT**, K.S. Institute of Technology, Bengaluru, for providing a very good infrastructure and all the kindness forwarded to us in carrying out this project work in college.

We would like to express our gratitude to **Dr. K.V.A Balaji, Principal & CEO**, K.S. Institute of Technology, Bengaluru, for his valuable guidance.

We like to extend our gratitude to **Dr. Rekha.B.Venkatapur, Professor and Head**, Department of Computer Science & Engineering, for providing a very good facilities and all the support forwarded to us in carrying out this Project Work Phase-I successfully.

We also like to thank our Project Coordinators, **Mr. K Venkata Rao, Associate Professor, Mrs. Vaneeta M, Associate Professor, Mr. Raghavendrachar S, Asst. Professor, Mr. Aditya Pai H, Asst. Professor, and Mrs. Sneha K, Asst. Professor, Department of Computer Science & Engineering** for their help and support provided to carry out the Project Work Phase-I successfully.

Also, we are thankful to **Mrs. Vaneeta M**, Associate Professor, for being our Project Guide, under whose able guidance this project work has been carried out Project Work Phase-I successfully.

We are also thankful to the teaching and non-teaching staff of Computer Science & Engineering, KSIT for helping us in completing the Project Work Phase-I work.

**BHAVESH BHANSALI
GANESH MAUDGHALYA H G
KARAN RAGHUNATH
P KISHORE**

ABSTRACT

Technology has enabled us to upload and submit documents online at ease with computers, mobile phones, etc. With this service there also comes the hassle of monitoring, adjusting size and dimensions of the document images(jpeg and png) and the clarity of the content in images requested by the websites. This service will help the website developers and the user to receive and upload images respectively without having to monitor the size(memory) and dimensions of images.

This report proposes an API which on integration can be used by website developers for their website to resize images according to the size and/or dimensions of their requirement and verify the validity and authenticity of content in images. The version of the Web API that will be developed will provide validation and verification for Aadhaar card, PAN card and Driving License. Further updates might include the implementation of verification of more documents.

Keywords: Web API, Resize, OCR, Validation, Verification

TABLE OF CONTENTS

Chapter No.	Title	Page No.
1.	INTRODUCTION	1-2
1.1	Overview	1
1.2	Purpose of the Project	1
1.3	Definitions	1-2
2.	LITERATURE SURVEY	3-4
2.1	Building REST APIs with Flask: Create Python Web Services with MySQL	3
2.2	OpenCV with Python by Example	3
2.3	Optical Character Recognition by Open source OCR Tool Tesseract: A Case Study	3-4
2.4	Analysis and Design of Selenium WebDriver Automation Testing Framework	4
2.5	Pillow (PIL Fork) Documentation, Release 6.2.0.dev0	5
2.6	A Review: Image Interpolation Techniques for Image Scaling	5
3.	PROBLEM IDENTIFICATION	6
3.1	Problem Statement	6
3.2	Project Scope	6
4.	GOALS AND OBJECTIVES	7
5.	SYSTEM REQUIREMENT SPECIFICATION	8

5.1	Software Requirements	8
5.2	Hardware Requirements (if any)	8
6.	METHODOLOGY	9-10
6.1	Procedure	9
6.2	Architecture	10
6.3	Modules	11
7.	APPLICATIONS	12
8.	CONTRIBUTION TO SOCIETY AND ENVIRONMENT	13
	REFERENCES	14
	APPENDIX - I CSI PUBLISHED PAPER COPY AND ABSTRACT	15-17
	APPENDIX - II CERTIFICATES OF PAPER PRESENTED	18-19
	APPENDIX - III CERTIFICATES OF WINNERS AND RUNNERS-UP	20-21

LIST OF FIGURES

Fig. No.	Figure Name	Page No.
6.1	Working process	9
6.2	Resizing based on dimensions	10
6.3	Resizing using OpenCV	10
6.4	OCR Process	10
6.5	Using OCR to scan Aadhar card details	10
6.6	Valid Aadhar card document	11
6.7	Verification of Authenticity	11

Chapter 1

INTRODUCTION

1.1 Overview

Identity documents for registration of events, exams, interviews, etc., had to be submitted and verified offline. Online submission of images of documents was enabled later on with improvement in technology. With this helpful service, there also came the hassle of having to upload images of a certain memory size and dimensions and to do that, the user had to monitor and modify the image or multiple images according to the requirements of the website developer. The proposed product that will be developed by us will resize the image or multiple images according to the requirements and also check for the validity and authenticity of the content of document in the image. This service can be integrated by the particular website developer to use it according to their website requirements. The product will be in the form of a API which will have different sections for different functions which are: Resizing, OCR, Document check (Validity of document based on format and layout), Verification (Identity, authenticity). The website developer can opt for the function they need to be performed. The version of the product that will be developed will provide validation and verification only for Aadhar card, PAN card and Driving License. Further updates might include validation and verification for more documents.

1.2 Purpose of the Project

Successfully deploy a API to:

- Resize Images based on size(memory) and dimensions
- Extract information in the document using OCR
- Check the validity of the document in the image uploaded
- Verify the authenticity of the document

1.3 Definitions

- **API (Application Programming Interface)**

An application programming interface is a computing interface that defines interactions between multiple software intermediaries. It defines the kinds of calls or requests that can be made, how to make them, the data formats that should be used, the conventions to follow, etc.

- **Image Resizing**

Computing to change the height, size or width of image.

- **OCR (Optical Character Recognition)**

Optical character recognition or optical character reader is the electronic or mechanical conversion of images of typed, handwritten or printed text into machine-encoded text, whether from a scanned document, a photo of a document, a scene-photo or from subtitle text superimposed on an image.

- **Document Validation**

Document check involves the checking the validity of document scanned based on format/layout of the document(s).

- **Document Verification**

Document verification involves the verification of the details of the document scanned and checking if the details are unique and original.

Chapter 2

LITERATURE SURVEY

2.1 Building REST APIs with Flask: Create Python Web Services with MySQL [1]

Author Kunal Relan has proposed that Flask is a lightweight microframework for web applications built on top of Python, which provides an efficient framework for building web-based applications using the flexibility of Python and strong community support with the capability of scaling to serve millions of users. Flask has excellent community support, documentation, and supporting libraries; it was developed to provide a barebone framework for developers, giving them the freedom to build their applications using their preferred set of libraries and tools. This book takes you through different stages of a REST API-based application development process using flask which explains the basics of the Flask framework assuming the readers understand Python. We'll cover database integration, understanding REST services, REST APIs performing CRUD operations, user authentication, third-party library integrations, testing, deployment, and application monitoring. At the end of this book, you'll have a fair understanding of Flask framework, REST, testing, deploying, and managing Flask applications, which will open doors to understanding REST API development.

2.2 OpenCV with Python by Example [2]

Author Prateek Joshi has proposed that Whenever we resize an image, there are multiple ways to fill in the pixel values. When we are enlarging an image, we need to fill up the pixel values in between pixel locations. When we are shrinking an image, we need to take the best representative value. When we are scaling by a non-integer value, we need to interpolate values appropriately, so that the quality of the image is maintained. There are multiple ways to do interpolation. If we are enlarging an image, it's preferable to use linear or cubic interpolation. If we are shrinking an image, it's preferable to use the area-based interpolation. Cubic interpolation is computationally more complex, and hence slower than linear interpolation. But the quality of the resulting image will be higher.

OpenCV provides a function called `resize` to achieve image scaling. If you don't specify a size (by using `None`), then it expects the X and Y scaling factors.

2.3 Optical Character Recognition by Open source OCR Tool Tesseract: A Case Study [3]

Authors Chirag Indravadanbhai Patel and Dharmendra Patel have proposed that Optical character recognition (OCR) method has been used in converting printed text into editable text. OCR is very useful and popular method in various applications. Accuracy of OCR can be dependent on text pre-processing and segmentation algorithms. Sometimes it is difficult to retrieve text from the image because of different size, style, orientation, complex background of image etc.

An image with the text is given as input to the Tesseract engine that is command-based tool. This method is very useful because it does the OCR of image with white text and black background. Tesseract was probably first to provide this kind of processing. Then after, the outlines are converted into Blobs. Blobs are organized into text lines, and the lines and regions are analysed for some fixed area or equivalent text size. Text is divided into words using definite spaces and fuzzy spaces. Recognition of text is then started as two-pass process.

In the first pass, an attempt is made to recognize each word from the text. Each word passed satisfactory is passed to an adaptive classifier as training data. The adaptive classifier tries to recognize text in more accurate manner. As adaptive classifier has received some training data it has learn something new so final phase is used to resolve various issues and to extract text from images. More details regarding every phase are available at.

2.4 Analysis and Design of Selenium WebDriver Automation Testing Framework [4]

Authors Satish Gojar, Rahul Joshi & Dhanashree Gaigaware have proposed that Automation testing uses automation tools to reduce human intervention and repeatable tasks. In this paper we have designed and implemented automation testing framework for testing web applications. This new automation testing framework has been implemented using selenium WebDriver tool. Using this framework tester can easily write their test cases efficiently and in less time.

Selenium is composed of multiple software automation tools such as, Selenium IDE, Selenium RC (selenium 1.0), and Selenium web driver (selenium 2.0). Selenium IDE is an integrated development environment to build the test scripts. It is a Firefox plug-in allow you to record edit and debug the selenium test cases. It records all actions performed by the end user and generate the test scripts. Selenium remote control (RC) was main selenium project for long time. Selenium RC is slower than the selenium web driver because it uses the java script program called selenium core. Selenium RC requires to start the server before executing the test scripts. It doesn't support the Ajax applications. To avoid the limitations of selenium RC, selenium web driver has been invented by merging selenium and web driver.

2.5 Pillow (PIL Fork) Documentation, Release 6.2.0.dev0 [5]

For geometry operations that may map multiple input pixels to a single output pixel, the Python Imaging Library provides different resampling filters.

- **NEAREST** Pick one nearest pixel from the input image. Ignore all other input pixels.
- **BOX** Each pixel of source image contributes to one pixel of the destination image with identical weights. For upscaling is equivalent of NEAREST. This filter can only be used with the `resize()` and `thumbnail()` methods. New in version 3.4.0.
- **BILINEAR** For resize calculate the output pixel value using linear interpolation on all pixels that may contribute to the output value. For other transformations linear interpolation over a 2x2 environment in the input image is used.
- **HAMMING** Produces a sharper image than BILINEAR, doesn't have dislocations on local level like with BOX. This filter can only be used with the `resize()` and `thumbnail()` methods. New in version 3.4.0.
- **BICUBIC** For resize calculate the output pixel value using cubic interpolation on all pixels that may contribute to the output value. For other transformations cubic interpolation over a 4x4 environment in the input image is used.
- **LANCZOS** Calculate the output pixel value using a high-quality Lanczos filter (a truncated sinc) on all pixels that may contribute to the output value. This filter can only be used with the `resize()` and `thumbnail()` methods.

2.6 A Review: Image Interpolation Techniques for Image Scaling [6]

Interpolation is way through which images are scaled. There are many different types of interpolation methods, each resulting in a different look to final image. Thus, it is best if the quality, or visible distinction for each pixel, is retained throughout the scaling process [13].

Image interpolation works in two directions, and tries to achieve a best approximation of a pixel's color and intensity based on the values at surrounding pixels [8]. Fig. 1 illustrates how scaling/resizing using two dimensional image interpolation works:

Interpolation techniques are mainly divided in two categories:

- A. Non-adaptive techniques
- B. Adaptive techniques

Chapter 3

PROBLEM IDENTIFICATION

3.1 Problem Statement

To design and develop a API to Resize, Validate and Verify Images to assist in the process of uploading Documents for the user.

3.2 Scope of Project

This API can be used by:

- Any website that requires resizing for the images uploaded by the website users.
- Websites that require validation, verification of authenticity and information of content of the document in the image.

Chapter 4

GOALS AND OBJECTIVES

Successfully deploy a Web API to:

- Resize Images based on size(memory) and dimensions
- Extract information in the document using OCR
- Check the validity of the document in the image uploaded
- Verify the authenticity of the document

Chapter 5

SYSTEM REQUIREMENT SPECIFICATION

A software requirements specification (SRS) is a comprehensive description of the intended purpose and environment for software under development. The SRS fully describes what the software will do and how it will be expected to perform. Software requirements specification permits a rigorous assessment of requirements before design can begin and reduces later redesign. It should also provide a realistic basis for estimating product costs, risks, and schedules.

The software requirements specification document enlists enough and necessary requirements that are required for the project development. To derive the requirements, we need to have clear and thorough understanding of the products to be developed or being developed. This is achieved and refined with detailed and continuous communications with the project team and customer till the completion of the software.

5.1 Software Requirements

We use the following software requirements:

- Web Browser
- IDE
- Coding Language: Python 3.5, PHP, HTML, CSS, Flask.
- Packages: NumPy, OpenCV, Pillow, Pytesseract, Selenium.

5.2 Hardware Requirements

- Processor: Intel i3 2nd generation and above
- RAM:4GB
- Hard Disk: 500GB

Chapter 6

METHODOLOGY

6.1 Procedure:

- The images are uploaded to a website by the user.
- The website calls the Web API for the function that it needs to be performed.
- The Web API starts with resizing the image according to the website's requirement.
- Based on the website's requirement which is either or both validation of a document or/and verification of authenticity of the document in the image, respective functions are executed by the Web API.
- The resized image is sent to the website developer with the details of the document if requested(OCR).
- If the document in the image is not validated or verified when requested, an error code or message is sent to the website developer. The website can then prompt the user to upload a valid and proper image, again.
- If the document is successfully validated and/or verified, a success code or message is sent to the developer. Following sections include.

6.2 Architecture:

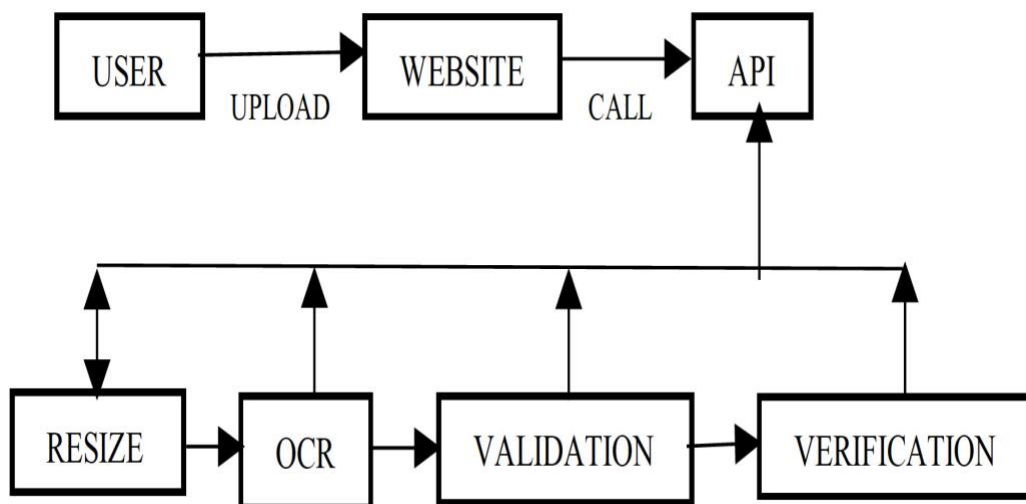


Fig 6.1: Working process

Fig 6.1 shows the working of our project. It is the dataflow diagram

6.3 Modules:

- Resizing:**

When the website receives the image uploaded by the user, it calls the API and passes the image as a parameter.

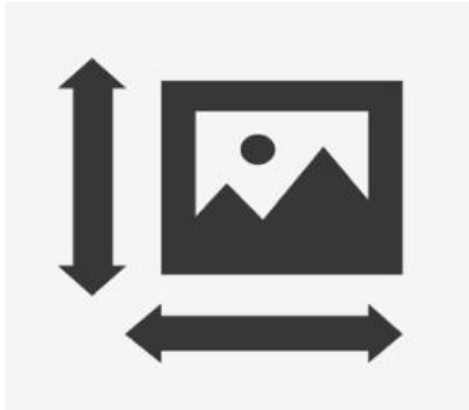


Fig 6.2: Resizing based on dimensions

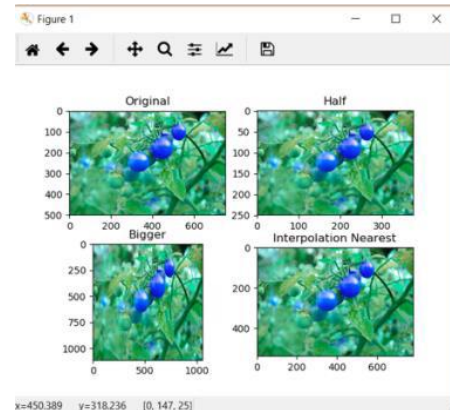


Fig 6.3: Resizing using OpenCV

Fig 6.2 and 6.3 shows the resizing of image based on dimensions using OpenCV

- OCR (Optical Character Recognition):**

Optical character recognition or optical character reader is the electronic or mechanical conversion of images of typed, handwritten or printed text into machine-encoded text, whether from a scanned document, a photo of a document, a scene-photo or from subtitle text superimposed on an image.



Fig 6.4: OCR Process



Fig 6.5: Using OCR to scan Aadhar card details

Fig 6.4 and 6.5 shows the OCR process and the field of scan in an Aadhar Card

- Validation:**

Validation will include the usage of OCR information extracted and the checking of the validity of the document (whether the right document image is uploaded in the right field or not).

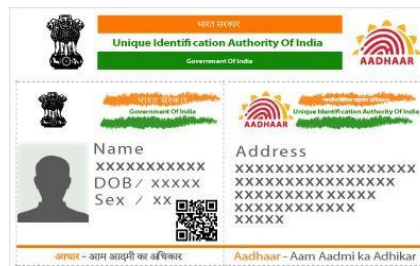


Fig 6.6: Valid Aadhar card document

Fig 6.6 is the image of a sample Aadhar card

- **Verification:**

Verification of Aadhar card is done by using <https://resident.uidai.gov.in/verify> website (Web Scraping) which verifies a valid Aadhar number by prompting Green tick mark and for PAN card, we'll use <https://www.incometaxindiaefiling.gov.in/> to check if the PAN number exists.



Fig 6.7: Verification of Authenticity

Fig 6.7 shows the verification of authenticity of an Aadhar card

Chapter 7

APPLICATIONS

The applications of our project are:

- Websites that require resizing for the images uploaded by the website users.
- Websites that require information of document in images (OCR).
- Websites that need validation on whether the right document is uploaded or not.
- Websites that require verification of authenticity of content in images.

Chapter 8

CONTRIBUTION TO SOCIETY AND ENVIRONMENT

The contributions to society and environment are:

- Eases the discomfort of users, having to monitor, adjust the size and dimensions of the document images (jpeg and png) requested by the websites.
- Helps websites with validation of document and verification of authenticity of the document uploaded by user.

REFERENCES

- [1] K. Relan, “Building REST APIs with Flask: Create Python Web Services with MySQL” https://doi.org/10.1007/978-1-4842-5022-8_1 2019, ISBN-13 (pbk): 978-1-4842-5021-1 ISBN-13 (electronic): 978-1-4842-5022-8, APRESS
- [2] Miguel Grinberg, “Flask Web Development”, Printed in the United States of America. Published by O’Reilly Media, Inc., 1005, Gravenstein Highway North, Sebastopol, CA 95472.
- [3] “OpenCV with Python By Example” Book by Prateek Joshi Originally published: 2015
- [4] Chirag Indravadanbhai Patel, Dharmendra Patel, “Optical Character Recognition by Open source OCR Tool Tesseract: A Case Study”
- [5] 2.5 Pillow (PIL Fork) Documentation, Alex Clark, Release 6.2.0.dev0
- [6] A Review: Image Interpolation Techniques for Image Scaling
- [7] Satish Gojar, Rahul Joshi & Dhanashree Gaigaware, “Analysis and Design of Selenium WebDriver Automation Testing Framework”, 2015
- [8] Alejandra Castelblanco, Jesus Solano, Christian Lopez, Esteban Rivera, Lizzy Tengana, and Mart’ınOchoa, “Machine Learning Techniques for Identity Document Verification in Uncontrolled Environments: A Case Study”, 2020
- [9] Inventive Communication and Computational Technologies. Proceedings of ICICCT 2020
G. Ranganathan • Joy Chen • Álvaro Rocha
- [10] Learning to resize image, Qi Wang , YuanYuan
- [11] RESIZING OF GESTURE-CREATED MARKINGS FOR DIFFERENT DISPLAY SIZES, Ronald Ho, Fremont, CA (US); Andrew A. Grieve, Waterloo (CA)
- [12] SYSTEMS AND METHODS TO RESIZE DOCUMENT CONTENT, Claude S. Fillion, Rochester, NY (US); Vishal Monga, State College, PA (US); Zhigang Fan, Webster, NY (US)

APPENDIX - I

ABSTRACT OF CSI PAPER PRESENTED

CSISGPID007: Survey on Document Resizing, Validation and Verification for Web API

*Bhavesh Bhansali, P Kishore, Ganesh Maudghalya H G, Karan Raghunath, Vaneeta M
K S Institute of Technology, Bengaluru*



Abstract : Technology has enabled us to upload and submit documents online at ease with computers, mobile phones, etc. With this service there also comes the hassle of monitoring, adjusting size and dimensions of the document images (jpeg and png) and the clarity of the content in images requested by the websites. This service will help the website developers and the user to receive and upload images respectively without having to monitor the size(memory) and dimensions of images. This paper proposes a Web API which on integration can be used by website developers for their website to resize images according to the size and/or dimensions of their requirement and verify they validity and authenticity of content in images. The version of the Web API that will be developed will provide validation and verification for Aadhaar card, PAN card and Driving License. Further updates might include the implementation of verification of more documents.

Proceedings of "34th CSI Karnataka State Student Convention (Online)", on theme "**Self-Reliance & Automation**", organized on 22nd and 23rd December-2020

CSI PUBLISHED PAPER COPY

Survey on Document Resizing, Validation and Verification for Web API

Bhavesh Bhansali
Computer Science and Engineering
K. S. Institute of Technology
Bengaluru, Karnataka
bbhansali18.bb@gmail.com

Ganesh Maudghalya H G
Computer Science and Engineering
K. S. Institute of Technology
Bengaluru, Karnataka
ganeshmaudghalya@gmail.com

Karan Raghunath
Computer Science and Engineering
K. S. Institute of Technology
Bengaluru, Karnataka
karanraghunath@gmail.com

P Kishore
Computer Science and Engineering
K. S. Institute of Technology
Bengaluru, Karnataka
kishorep.shrivatsa@gmail.com

Mentor/Co-author:
Vaneeta M
Associate Professor
Computer Science and Engineering
K. S. Institute of Technology
Bengaluru, Karnataka
vaneetam@ksit.edu.in

Abstract --- Technology has enabled us to upload and submit documents online at ease with computers, mobile phones, etc. With this service there also comes the hassle of monitoring, adjusting size and dimensions of the document images (jpeg and png) and the clarity of the content in images requested by the websites. This service will help the website developers and the user to receive and upload images respectively without having to monitor the size (memory) and dimensions of images. This paper proposes a Web API which on integration can be used by website developers for their website to resize images according to the size and/or dimensions of their requirement and verify their validity and authenticity of content in images. The version of the Web API that will be developed will provide validation and verification for Aadhaar card, PAN card and Driving License. Further updates might include the implementation of verification of more documents.
Keywords: Web API, Resize, OCR, Validation, Verification

I. INTRODUCTION

Identity documents for registration of events, exams, interviews, etc., had to be submitted and verified offline. Online submission of images of documents was enabled later on with improvement in technology. With this helpful service, there also came the hassle of having to upload images of a certain memory size and dimensions and to do that, the user had to monitor and modify the image or multiple images according to the requirements of the website developer. The proposed product that will be developed by us will resize the image or multiple images according to the requirements and also check for the validity and authenticity of the content of document in the image. This service can be integrated by the particular website developer to use it according to their website requirements. The product will be in the form of a Web API which will have different sections for different functions which are: Resizing, OCR, Document check (Validity of document based on format and layout), Verification (Identity, authenticity). The website developer can opt for the function they need to be performed. The version of the product that will be developed will provide validation and verification only for Aadhaar card, PAN card and Driving License. Further updates might include validation and verification for more documents.

II. METHODOLOGY

- The images are uploaded to a website by the user.
 - The website calls the Web API for the function that it needs to be performed.
 - The Web API starts with resizing the image according to the website's requirement.
 - Based on the website's requirement which is either or both validation of a document or/and verification of authenticity of the document in the image, respective functions are executed by the Web API.
 - The resized image is sent to the website developer with the details of the document if requested (OCR).
 - If the document in the image is not validated or verified when requested, an error code or message is sent to the website developer. The website can then prompt the user to upload a valid and proper image, again.
 - If the document is successfully validated and/or verified, a success code or message is sent to the developer. Following sections include a brief about different sections of Web API.
1. Resizing:
Image resizing is the process of increasing or decreasing of dimensions or memory size of an image.



Fig 1: Resizing based on dimensions

2. OCR (Optical Character Recognition):
Optical character recognition or optical character reader is the electronic or mechanical conversion of images of typed, handwritten or printed text into machine-encoded text, whether from a scanned document, a photo of a document, a scene-photo or from subtitle text superimposed on an image.



Fig 2: OCR Process

3. Document Check (Validation):

Document check involves the checking the validity of document scanned based on format/layout of the document(s).



Fig 3: Valid Aadhar card document

4. Document Verification(Authenticity):

Document verification involves the verification of the details of the document scanned and checking if the details are unique and original.



Fig 4: Verification of originality of details

III. BLOCK DIAGRAM

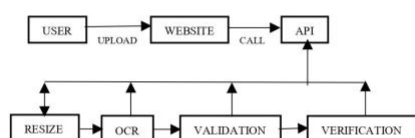


Fig 5: Working of the API

IV. TECHNIQUE

1. Image acquisition from website:

When the website receives the image uploaded by the user, it calls the API and passes the image as a parameter.

2. Resizing:

The API resizes the image based on the requirements of the website developer, images will be resized with respect to dimensions and memory size using OpenCV library functions.

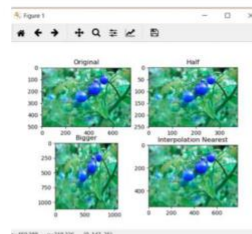


Fig 6: Resizing using OpenCV

3. OCR(Optical Character Recognition):

OCR will involve the usage of OpenCV, Pillow, Numpy and Pytesseract to extract the important information like Name, Date of Birth, Card Number, etc. from the document content in the image.



Fig 7: Using OCR to scan Aadhar card details

4. Validation:

Validation will include the usage of OCR information extracted and the checking of the validity of the document (whether the right document image is uploaded in the right field or not).

5. Verification:

Verification of Aadhar card is done by using <https://resident.uidai.gov.in/verify> website (Web Scrapping) which verifies a valid Aadhar number by prompting Green tick mark and for PAN card, we'll use <https://www.incometaxindiaefiling.gov.in/> to check if the PAN number exists.

V. LITERATURE SURVEY

[1] Author Kunal Relan has proposed that Flask is a lightweight microframework for web applications built on top of Python, which provides an efficient framework for building web-based applications using the flexibility of Python and strong community support with the capability of scaling to serve millions of users. Flask has excellent community support, documentation, and supporting libraries; it was developed to provide a barebone framework for developers, giving them the freedom to build their applications using their preferred set of libraries and tools. This book takes you through different stages of a REST API-based application development process using flask which explains the basics of the Flask framework assuming the readers understand Python. We'll cover database integration, understanding REST services, REST APIs performing CRUD operations, user authentication, third-party library integrations, testing, deployment, and application monitoring. At the end of this book, you'll have a fair understanding of Flask framework, REST, testing, deploying, and managing Flask applications, which will open doors to understanding REST API development.

[2] Author Miguel Grinberg has proposed that Flask makes it very easy to create RESTful web services. The familiar route () decorator along with its methods optional argument can be used to declare the routes that handle the resource URLs exposed by the service. Working with JSON data is also simple, as JSON data included with a request can be obtained in dictionary format by calling request.get_json(), and a response that needs to contain JSON can be easily generated from a Python dictionary using Flask's jsonify() helper function.

[3] Author Prateek Joshi has proposed that Whenever we resize an image, there are multiple ways to fill in the pixel values. When we are enlarging an image, we need to fill up the pixel values in between pixel locations. When we are shrinking an image, we need to take the best representative value. When we are scaling by a non-integer value, we need to interpolate values appropriately, so that the quality of the image is maintained. There are multiple ways to do interpolation. If we are enlarging an image, it's preferable to use linear or cubic interpolation. If we are shrinking an image, it's preferable to use the area-based interpolation. Cubic interpolation is computationally more complex, and hence slower than linear interpolation. But the quality of the resulting image will be higher.

OpenCV provides a function called resize to achieve image scaling. If you don't specify a size (by using None), then it expects the X and Y scaling factors.

[4] Authors Chirag Indravadanbhai Patel and Dharmendra Patel have proposed that Optical character recognition (OCR) method has been used in converting printed text into editable text. OCR is very useful and popular method in various applications. Accuracy of OCR can be dependent on text preprocessing and segmentation algorithms. Sometimes it is difficult to retrieve text from the image because of different size, style, orientation, complex background of image etc.

An image with the text is given as input to the Tesseract engine that is command-based tool. This method is very useful because it does the OCR of image with white text and black background. Tesseract was probably first to provide this kind of processing. Then after, the outlines are converted into Blobs. Blobs are organized into text lines, and the lines and regions are analyzed for some fixed area or equivalent text size. Text is divided into words using definite spaces and fuzzy spaces. Recognition of text is then started as two-pass process.

In the first pass, an attempt is made to recognize each word from the text. Each word passed satisfactory is passed to an adaptive classifier as training data. The adaptive classifier tries to recognize text in more accurate manner. As adaptive classifier has received some training data it has learn something new so final phase is used to resolve various issues and to extract text from images. More details regarding every phase are available at.

[5] Authors Satish Gojar, Rahul Joshi & Dhanashree Gaigaware have proposed that Automation testing uses automation tools to reduce human intervention and repeatable tasks. In this paper we have designed and implemented automation testing framework for testing web applications.

This new automation testing framework has been implemented using selenium WebDriver tool. Using this framework tester can easily write their test cases efficiently and in less time.

Selenium is composed of multiple software automation tools such as, Selenium IDE, Selenium RC (selenium 1.0), and Selenium web driver (selenium 2.0). Selenium IDE is an integrated development environment to build the test scripts. It is a Firefox plug-in allow you to record edit and debug the selenium test cases. It records all actions performed by the end user and generate the test scripts. Selenium remote control (RC)

was main selenium project for long time. Selenium RC is slower than the selenium webdriver because it uses the java script program called selenium core. Selenium RC requires to start the server before executing the test scripts. It doesn't support the Ajax applications. To avoid the limitations of selenium RC, selenium webdriver has been invented by merging selenium and webdriver.

In order to test the web application proposed automation framework surely reduces the time required to write the test cases and increase the pass percentage of test cases. It also reduces hectic workload of tester. By using this framework one can generate the customized test reports and also analyze the failures using screenshots of failed test cases. Tester can maintain the all data from central place. This framework is very useful for dynamically changing web applications.

ACKNOWLEDGMENT

We would like to express our deep gratitude to Associate Professor Mrs. Vaneeta M for her valuable and constructive suggestions during the planning and development of this paper. Her willingness to give her time so generously has been very much appreciated.

We would also like to thank all the professors of KSIT for their continuous support and encouragement.

REFERENCES

- [1] K. Relan, "Building REST APIs with Flask: Create Python Web Services with MySQL" <https://doi.org/10.1007/978-1-4842-5022-8> 1 2019, ISBN-13 (pbk): 978-1-4842-5021-1 ISBN-13 (electronic): 978-1-4842-5022-8, APRESS
- [2] Miguel Grinberg, "Flask Web Development", Printed in the United States of America. Published by O'Reilly Media, Inc., 1005, Gravenstein Highway North, Sebastopol, CA 95472.
- [3] "OpenCV with Python By Example" Book by Prateek Joshi Originally published: 2015
- [4] Chirag Indravadanbhai Patel, Dharmendra Patel, "Optical Character Recognition by Open source OCR Tool Tesseract: A Case Study"
- [5] Satish Gojar, Rahul Joshi & Dhanashree Gaigaware, "Analysis and Design of Selenium WebDriver Automation Testing Framework", 2015
- [6] Alejandra Castelblanco, Jesus Solano, Christian Lopez, Esteban Rivera, Lizzy Tengana, and Mart'inOchoa, "Machine Learning Techniques for Identity Document Verification in Uncontrolled Environments: A Case Study", 2020
- [7] Inventive Communication and Computational Technologies. Proceedings of ICICCT 2020 G. Ranganathan • Joy Chen • Alvaro Rocha
- [8] Learning to resize image Qi Wang , YuanYuan
- [9] RESIZING OF GESTURE-CREATED MARKINGS FOR DIFFERENT DISPLAY SIZES Ronald Ho, Fremont, CA (US); Andrew A. Grieve, Waterloo (CA)
- [10] SYSTEMS AND METHODS TO RESIZE DOCUMENT CONTENT Claude S. Fillion, Rochester, NY (US); Vishal Monga, State College, PA (US); Zhigang Fan, Webster, NY (US)

APPENDIX – II

CERTIFICATES OF PAPER PRESENTED

1) BHAVESH BHANSALI

1KS17CS016



2) GANESH MAUDGHALYA H G

1KS17CS025



3) KARAN RAGHUNATH

1KS17CS032



4) P KISHORE

1KS17CS051



APPENDIX-III

CERTIFICATES OF WINNERS AND RUNNERS-UP

1) BHAVESH BHANSALI

1KS17CS016



2) GANESH MAUDGHALYA H G

1KS17CS025



3) KARAN RAGHUNATH

1KS17CS025



4) P KISHORE

1KS17CS051

