PROJECT REPORT

PAN CARD TAMPERING USING OPEN CV

Submitted towards the partial fulfillment of the criteria for award of PGA by Imarticus

Submitted By:

KHANNCHAN (Roll IL025609)

Course and Batch: PGA20

ABSTRACT

The project is based on pan card tampering . It's an essential process in the Recruitment procedure of the HR team. The project is based on the concepts of deep learning where this project uses the concept of SSIM (Structural Similarity Index). The project uses the 2 images the original and the tampered image and it provides the SSIM index which talks about the extent of tampering that is the percentage of the tampering. The more the tampering lesser the index. The front end has also been made using Flask. The language used for coding is python.

Acknowledgements

I am using this opportunity to express my gratitude to everyone who supported me throughout the course of this project. I am thankful for their aspiring guidance, invaluably constructive criticism and friendly advice during the project work. I am sincerely grateful to them for sharing their truthful and illuminating views on a number of issues related to the project.

Further, we were fortunate to have VijayaKumar as our mentor. He has readily shared his immense knowledge in data analytics and guide us in a manner that the outcome resulted in enhancing my data skills.

I wish to thank, all the faculties, as this project utilized knowledge gained from every course that formed the PGA program.

I certify that the work done by us for conceptualizing and completing this project is original and authentic.

Date: March 31, 2022 Khannchan.B

Place: Chennai

Table of Contents

CHAPTER 1: INTRODUCTION

1.1 TITLE & OBJECTIVE OF THE STUDY

PAN CARD TAMPERING VERIFICATION USING OPEN CV

The projects aim to find whether the pan card given by the user is Tampered or not using the SSIM Index.

1.2 NEED OF THE STUDY

The purpose of this project is to detect tampering/fraud of PAN cards using computer vision. This project will help the different organizations in detecting whether the Id i.e. the PAN card provided to them by their employees or customers or anyone is original or not.

For this project we will calculate the structural similarity of the original PAN card and the PAN card uploaded by the user.

1.3 BUSINESS OR ENTERPRISE UNDER STUDY

Any company wanting to a background check on the employees as a part of their recruitment process could use the application.

1.4 DATA SOURCES

The open and the tampered images have been taken from the internet for education and experimental purposes.

1.5 TOOLS & TECHNIQUES

The tool used is SSIM . The technique is using the contours, threshold and the bounding boxes provided in imutils and opencv library to understand the process better.

CHAPTER 2: DATA PREPARATION AND UNDERSTANDING

One of the first steps we engaged in was to outline the sequence of steps that we will be following for our project. Each of these steps are elaborated below

The data here is in the form of an image

The image is loaded to the program and is resized accordingly and also the formatting is also done.

2.1 STEPS INVOLVED IN THE PROJECT.

1. Import necessary libraries
2. Scraping the tampered and original pan card from the website
3. Scaling down the shape of the tampered image as the original image
4. Read original and tampered image
5. Converting an image into a grayscale image
6. Applying Structural Similarity Index (SSIM) technique between the two images
7. Calculate Threshold and contours and
8. Experience real-time contours and threshold on images

CHAPTER 3: MODEL BUILDING

3.1 IMPORTING THE NECESSARY LIBRARY

- Skimage: Scikit-image, or ski-mage, is an open-source Python package, in this project most of the image processing techniques will be used via scikit-image
- imutils: Imutils are a series of convenience functions to make basic image processing functions such as translation, rotation, resizing, and displaying images easier with OpenCV.
- cv2: OpenCV (Open Source Computer Vision Library) is a library of programming functions. Here in this project major reading and writing of the image are done via cv2.
- PIL: PIL (Python Imaging Library) is a free and open-source additional library for the Python programming language that adds support for opening, manipulating, and saving many different image file formats.

3.2 ABOUT SSIM

The Structural Similarity Index (SSIM) is a perceptual metric that quantifies the image quality degradation that is caused by processing such as data compression or by losses in data transmission.

This metric is basically a full reference that requires 2 images from the same shot, this means 2 graphically identical images to the human eye. The second image generally is compressed or has a different quality, which is the goal of this index.

SSIM is usually used in the video industry but has as well a strong application in photography.

SSIM actually measures the perceptual difference between two similar images. It cannot judge which of the two is better: that must be inferred from knowing which is the original one and which has been exposed to additional processing such as compression or filters.

3.3 PROCEDURES INVOLVED

1. Getting the images from the various sources
    - original = Image.open(requests.get('https://www.thestatesman.com/wp-content/uploads/2019/07/pan-card.jpg', stream=True).raw)
    - tampered = Image.open(requests.get('https://assets1.cleartax-cdn.com/s/img/20170526124335/Pan4.png', stream=True).raw)
2. Converting the format of a tampered image similar to the original image.
3. Display original PAN card image which will be used for comparison.

    Original

    Tampered

4. Convert the images into grayscale

    Converting images into grayscale is very much beneficial inaccuracy of image processing because in image processing many applications don't help us in identifying the importance, edges of the colored images also colored images are a bit complex to understand by machine because they have 3 channel while grayscale has only 1 channel.
5. Applying Structural Similarity Index (SSIM) technique between the two images

    Structural similarity index helps us to determine exactly where in terms of x,y coordinates location, the image differences are. Here, we are trying to find similarities between the original and tampered image.

The lower the SSIM score lower is the similarity, i.e SSIM score is directly proportional to the similarity between two images

We have given one threshold value of "45" i.e if any score is >= 80 it will be regarded as the original pan card else tampered with one.

Generally SSIM values 0.97, 0.98, 0.99 for good quallty recontruction techniques.

6. Performing threshold and contours on images

   Contours detection is a process that can be explained simply as a curve joining all the continuous points (along with the boundary), having the same color or intensity. The algorithm does indeed find edges of images but also puts them in a hierarchy.

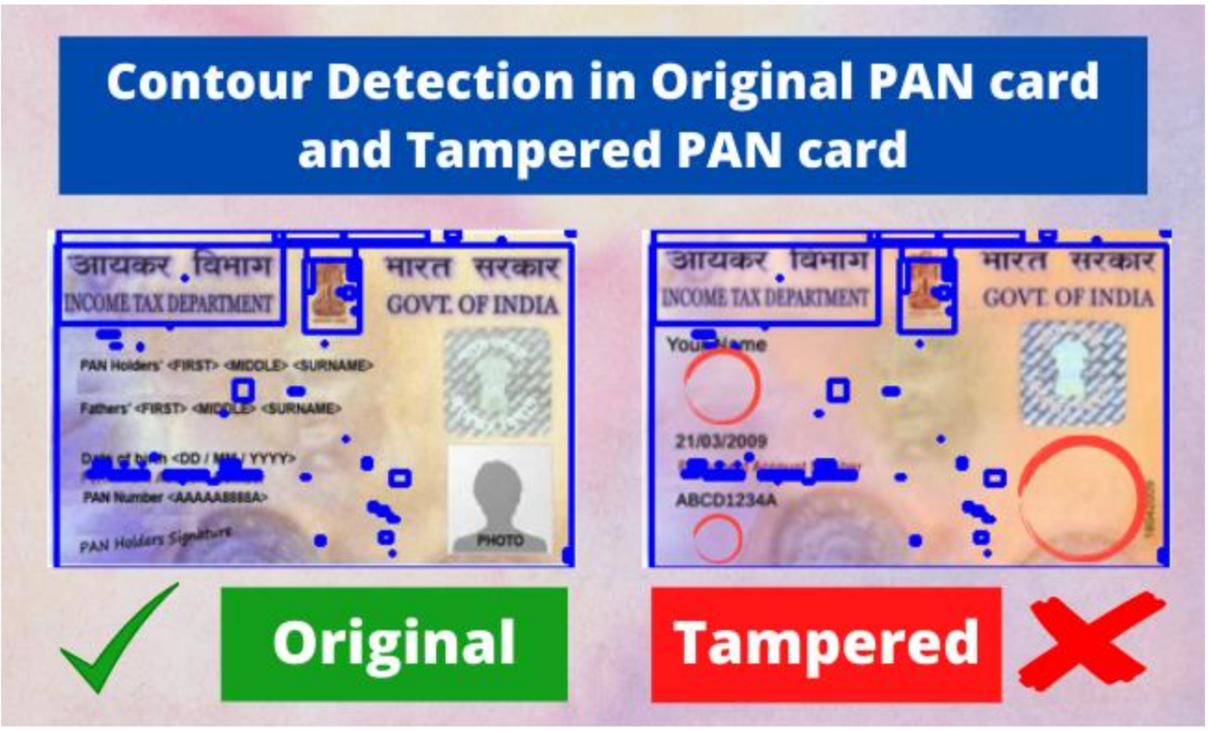
   Here we are using the threshold function of computer vision which applies an adaptive threshold to the image which is stored in the form array. This function transforms the grayscale image into a binary image using a mathematical formula. Find contours works on binary image and retrieve the contours. These contours are a useful tool for shape analysis and recognition. Grab contours grabs the appropriate value of the contours.

7. Creating Bounding Rectangles

   Bounding rectangle helps in finding the ratio of width to height of the bounding rectangle of the object. We compute the bounding box of the contour and then draw the bounding box on both input images to represent where the two images are different or not.

Inference :

Here in the above output, you can see that the original image is shown with the contours (bounding boxes) on it using fromarray() function.



8. Showing the different image

   Here is another very interactive way to show the contours in terms of heated threshold i.e. by finding the heated zone (text/image zone) and normal zone (without text/image).

   The heated zone i.e the zone which has text/images will be shown in the dark (black) region and the other one as a light (kind of white) zone.

9. Threshold Image

Inference: Everything here is just the same all we can see is the change in the role of color, here white color is showing the heated zone and the black color is showing the normal zone.

CHAPTER 4: ABOUT COMPUTER VISION

Computer vision is a field of artificial intelligence that trains computers to interpret and understand the visual world. Using digital images from cameras and videos and deep learning models, machines can accurately identify and classify objects — and then react to what they "see".

Computer vision tasks include methods for acquiring, processing, analyzing and understanding digital images

Computer Vision in Image processing is mainly focused on processing the raw input images to enhance them or preparing them to do other tasks. Computer vision is focused on extracting information from the input images or videos to have a proper understanding of them to predict the visual input like the human brain.

Computer vision needs lots of data. It runs analyses of data over and over until it discerns distinctions and ultimately recognize images. For example, to train a computer to recognize automobile tires, it needs to be fed vast quantities of tire images and tire-related items to learn the differences and recognize a tire, especially one with no defects.

Two essential technologies are used to accomplish this: a type of machine learning called deep learning and a convolutional neural network (CNN).

Machine learning uses algorithmic models that enable a computer to teach itself about the context of visual data. If enough data is fed through the model, the computer will "look" at the data and teach itself to tell one image from another. Algorithms enable the machine to learn by itself, rather than someone programming it to recognize an image.

A CNN helps a machine learning or deep learning model "look" by breaking images down into pixels that are given tags or labels. It uses the labels to perform convolutions (a mathematical operation on two functions to produce a third function) and makes predictions about what it is "seeing." The neural network runs convolutions and checks the accuracy of its predictions in a series of iterations until the predictions start to come true. It is then recognizing or seeing images in a way similar to humans.

Much like a human making out an image at a distance, a CNN first discerns hard edges and simple shapes, then fills in information as it runs iterations of its predictions. A CNN is used to understand single images. A recurrent neural network (RNN) is used in a similar way for video applications to help computers understand how pictures in a series of frames are related to one another.

Computer vision examples

Many organizations don't have the resources to fund computer vision labs and create deep learning models and neural networks. They may also lack the computing power required to process huge sets of visual data. Companies such as IBM are helping by offering computer vision software development services. These services deliver pre-built learning models available from the cloud — and also ease demand on computing resources. Users connect to the services through an application programming interface (API) and use them to develop computer vision applications.

IBM has also introduced a computer vision platform that addresses both developmental and computing resource concerns. IBM Maximo Visual Inspection includes tools that enable subject matter experts to label, train and deploy deep learning vision models — without coding or deep learning expertise. The vision models can be deployed in local data centers, the cloud and edge devices.

While it's getting easier to obtain resources to develop computer vision applications, an important question to answer early on is: What exactly will these applications do? Understanding and defining specific computer vision tasks can focus and validate projects and applications and make it easier to get started.

Here are a few examples of established computer vision tasks:

- Image classification sees an image and can classify it (a dog, an apple, a person's face). More precisely, it is able to accurately predict that a given image belongs to a certain class. For example, a social media company might want to use it to automatically identify and segregate objectionable images uploaded by users.

- Object detection can use image classification to identify a certain class of image and then detect and tabulate their appearance in an image or video. Examples include detecting damages on an assembly line or identifying machinery that requires maintenance.
- Object tracking follows or tracks an object once it is detected. This task is often executed with images captured in sequence or real-time video feeds. Autonomous vehicles, for example, need to not only classify and detect objects such as pedestrians, other cars and road infrastructure, they need to track them in motion to avoid collisions and obey traffic laws.
- Content-based image retrieval uses computer vision to browse, search and retrieve images from large data stores, based on the content of the images rather than metadata tags associated with them. This task can incorporate automatic image annotation that replaces manual image tagging. These tasks can be used for digital asset management systems and can increase the accuracy of search and retrieval.

CHAPTER 5

5.1 INFERENCE OF THE PROJECT

- Finding out structural similarity of the images helped us in finding the difference or similarity in the shape of the images.
- Similarly, finding out the threshold and contours based on that threshold for the images converted into grayscale binary also helped us in shape analysis and recognition.
- As our SSIM is ~31.2% we can say that the image user provided is fake or tampered with.
- Finally, we visualized the differences and similarities between the images using by displaying the images with contours, difference, and threshold.

REFERENCES

1. https://www.ibm.com/in-en/topics/computer-vision#:~:text=Computer%20vision%20is%20a%20field,recommendations%20based%20on%20that%20information.
2. https://en.wikipedia.org/wiki/Structural_similarity
3. https://www.freecodecamp.org/news/how-to-build-a-web-application-using-flask-and-deploy-it-to-the-cloud-3551c985e492/
4. https://en.everybodywiki.com/Complex_Wavelet_SSIM