'National Conference on Advancements in Computer Science and Engineering'
International Journal of Scientific Research in Computer Science,
Engineering and Information Technology | ISSN : 2456-3307 (www.ijsrcseit.com)

Image Forgery Detection

Priyanshu Jangir D, Kajal Sinha
Department of CSE, New Horizon College of Engineering, Bangalore, Karnataka, India

ABSTRACT

An image forgery detector is a system or tool that is used to detect whether an image has been tampered with or altered in any way. It can be used to detect a wide range of image forgeries, including splicing, copy-move, and re- sampling. The goal of an image forgery detector is to identify any inconsistencies or anomalies in an image that may indicate that it has been tampered with. It can also be used to detect the use of image editing software and detect traces of manipulation. It is often used in digital forensics, surveillance, and security applications. The entire program has been developed in Python and uses the VS-code IDE for running the python application. The mini- project is completely based on the high-level language, Python and uses GUI programming to provide a simple and easy to understand platform for the users.

Keywords: DataTypes, Widgets, Labels, Buttons, Geometry managers

I. INTRODUCTION

From the last few years it has been observed that image editing has been spread throughout the web. In this paper we will be discussing about the concepts of deep learning. Deep Fake generated content is also addressed insofar as its application is aimed at images, achieving the same effect as splicing. We are using deep learning concepts as it is gives the best performances on the dataset.

An image forgery detector is a system or tool that is used to detect whether an image has been tampered with or altered in any way. It can be used to detect a wide range of image forgeries like copy-move. Therefore, I have developed a project which will be able to detect image forgeries by accessing an image from the user and then using the different methods to detect the image forgeries.

The objectives of an image forgery detector are to:

1. Detect the presence of image forgeries: The main objective of an image forgery detector is to accurately detect any modifications made to an image, such as splicing, copy-move, and retouching.
2. Identify the type of forgery: A good image forgery detector should be able to identify the specific type of forgery present in an image, such as splicing, copy, move, or retouching.
3. Minimize the number of false positives and false negatives: The detector should be able to accurately identify forgeries while minimizing the number of false positives (incorrectly identifying an original image as a forgery) and false negatives (failing to identify a forgery).
4. Provide clear and informative results: The detector should provide clear and informative results that indicate the presence of a forgery and the specific type of forgery.

II. METHODS AND MATERIAL

- Image pre-processing: involves preparing the image for analysis by converting it to grayscale and resizing it to a suitable size.
- Keypoint detection: In this step, keypoints are detected in the image using a feature detection algorithm such as SIFT, SURF, or ORB. These keypoints are used to identify and match regions in the image.
- Descriptor extraction: In this step, a descriptor is extracted for each keypoint, which describes the local features of the region around the keypoint. d. Keypoint matching: In this step, the keypoints and their descriptors are matched to identify similar regions in the image.
- Clustering: This step involves grouping the matching keypoints into clusters based on their similarity.
- Region of Interest (ROI) identification: In this step, the regions in the image that are most likely to contain forgeries are identified based on the clusters obtained in the previous step.
- Forgery detection: In this step, the regions identified as ROIs are further analyzed to detect if they contain forgeries. h. Post-processing: This step involves displaying the results of the forgery detection, such as highlighting the regions that contain forgeries.

III. RESULTS AND DISCUSSIONS

Detection of image forgeries: The detector will be able to accurately identify any modifications made to an image, such as copy-move. Identification of the type of forgery: The detector will be able to identify the any type of tampered region.

This mini-project has been developed using python 'tkinter' package to provide the user with a great user-friendly GUI application The interface should be simple and help the user to use the Image forgery detector inorder to detect the forgeries in an image.

- **ALGORITHM:**

1. Start.
2. Upload the image using the upload feature available.
3. Select the type of method out of 7 methods in which the image has to be processed.
4. Later wait for few seconds
5. The output is displayed.
6. End

- **REQUIREMENTS**

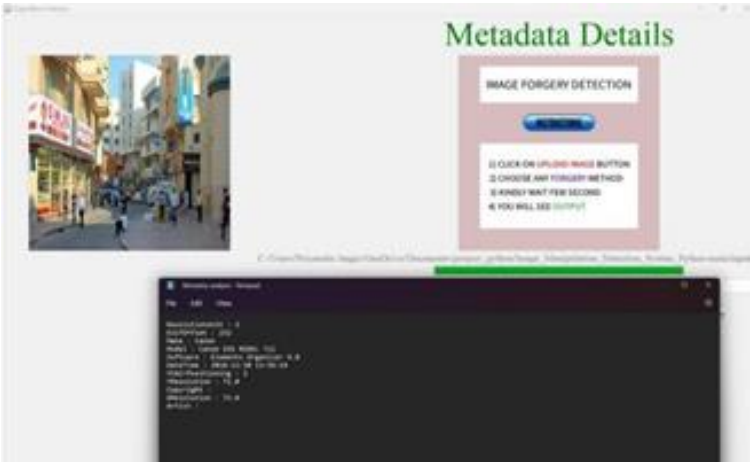
a. matplotlib==3.3.2
b. numpy==1.19.3
c. opencv_python==4.5.4.60
d. Pillow==9.2.0
e. prettytable==3.2.0
f. pyparsing==3.0.6
g. scikit_learn==1.1.2
h. scipy==1.4.1

- **IMAGE RESULTS**

A. Home Page UI

Figure 1. Screenshot of Home Page UI

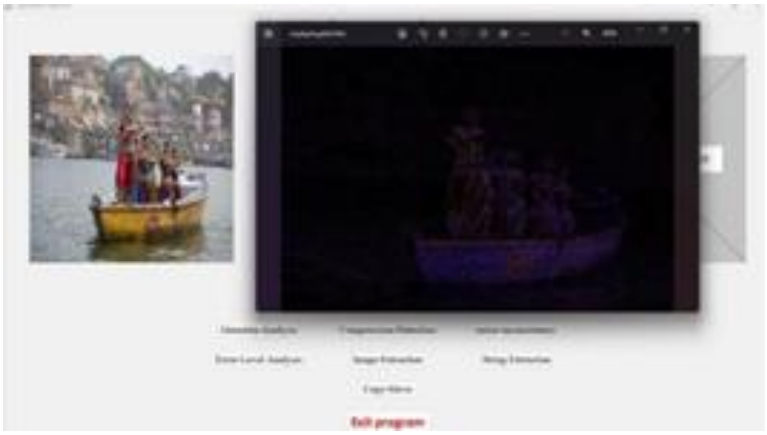
B. OUTPUT FOT METADATA ANALYSIS



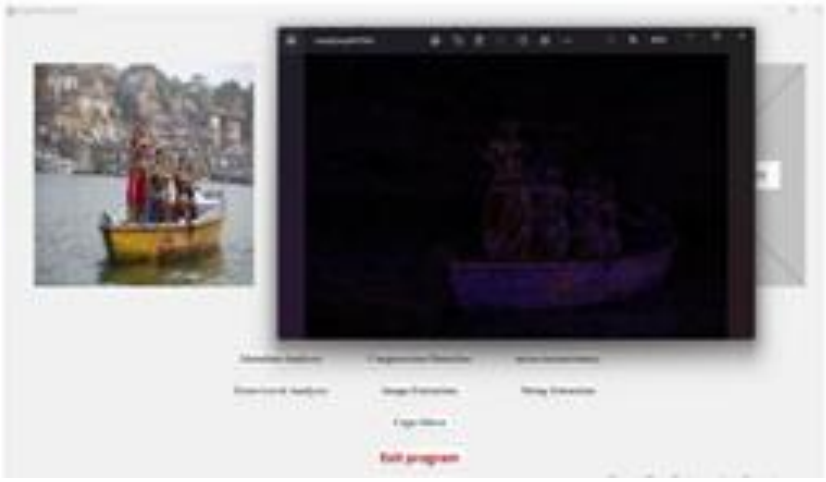
C. OUTPUT FOR IMAGE COMPRESSION DETECTION METHOD

D. OUTPUT FOR NOISE INCONSISTENCY DETECTION METHOD

E. OUTPUT FOR ELA DETECTION



F. OUTPUT FOR IMAGE EXTRACTION (STEGONAGAPHY) METHOD



G. OUTPUT FOR COPY MOVE FORGERY METHOD

IV. CONCLUSION

This project has successfully accomplished the goals it had set out in the objectives and design sections of this report. The individual user type UI has successfully implemented several modules. Using the listed 7 methods one can easily detect if an image has been forged or is an authentic image one can easily detect if an image has been forged or is an authentic image.

V. REFERENCES

[1]. https://www.github.com/ (example for website referred
[2]. https://www.python.org/