Assignment: Image Manipulation Detection

Motivation

The domains of digital document forensics and digital image forensics are intertwined by foundational principles that connect them. By gaining first-hand experience in detecting image manipulations, you will acquire the essential skills and expertise necessary to tackle document manipulations effectively. This experience will not only equip you with the required knowledge but also demonstrate your suitability for the role.

Problem Statement

Your task in this assignment is to develop an image manipulation detection system, specifically focusing on two common types of forgeries: *copy-move* and *splicing*.

Copy-Move Forgery:

Copy-move forgery refers to the act of copying a specific portion of an image and pasting it onto another area within the same image.

Splicing Forgery:

Splicing forgery involves merging two or more images to create a single composite image. This technique aims to combine different elements or regions from multiple sources.

Objective:

Your objective is to develop an image manipulation detection system with maximum accuracy for identifying manipulated images. You are free to choose any approach or combination of techniques, such as deep learning, image processing algorithms, or other relevant methodologies.

Metrics:

You are required to provide the class-wise classification accuracy and the confusion matrix as part of your reporting. Feel free to evaluate on additional metrics to enhance your analysis.

Data:

We've provided a carefully curated dataset that consists of images belonging to three categories: *(1) authentic*, *(2) copy-moved* & *(3) spliced*. The dataset has been partitioned into *traindev* and *test* sets, where the test set should be used for benchmarking your system's performance. You can download it from here. We've also included masks for copy-moved and spliced regions.

Note: We've kept aside a hold-out test dataset, which we'll use to benchmark your submission. This information is outlined in the Deliverables section.

Guidelines

Feel free to utilise these guidelines as a reference to successfully complete this assignment:

1. Familiarise yourself with the concepts of copy-move and splicing forgeries.
2. Explore the provided dataset, paying attention to the characteristics and variations within each category.
3. Choose appropriate methods and algorithms to develop an effective image manipulation detection system.
4. Experiment with different preprocessing techniques, feature extraction approaches, and model architectures (if using deep learning).
5. Train and fine-tune your model using the traindev set and benchmark its performance on the test set.
6. Evaluate the accuracy and effectiveness of your model, providing insights into its strengths and limitations.
7. Continuously document your approach, results, thoughts, and challenges throughout the assignment. Embrace a scientific mindset as you work towards solving the problem, replicating the thoughtful process of a scientist.

Note:

- Feel free to leverage existing libraries, frameworks, or pre-trained models to facilitate your development process.
- *Creativity and innovation in tackling the problem will be highly appreciated.*

Deliverables

Please ensure that you submit the following items:

- **Code implementation:** Develop a robust and efficient image manipulation detection system using the provided dataset. The code should be well-documented, organised, and easily reproducible. It should be present in a folder and include:
  - *All experiments conducted:* Include scripts and notebooks that document the various approaches and techniques you explored during the project.
  - *Trained weights (if applicable):* Include the trained model weights, particularly if you employed deep learning techniques.
  - *Dependencies:* Provide a requirements.txt file listing all the required dependencies for running your code.
- **Model evaluation:** Present the evaluation results of your system on the test set, including metrics and performance insights. We need you to provide these in the folder:
  - *Test set evaluation:* Include a notebook that computes the metrics on the test set.
  - *Hold-out Test set evaluation:* Include a notebook with clear instructions on how to compute metrics on our hold-out test dataset. Ensure the notebook is executable and produces accurate results when run.
- **Report:** Present your findings in a miniature report. It should be provided in the following format:

    - *README.md file:* Your repository must contain a concise and informative README.md file, serving as a report summarising your work. Emphasise your thinking process, problem-solving mindset, challenges encountered, and their solutions. Additionally, include a section that lists the metrics obtained on the test dataset. The report should be comprehensive yet succinct, allowing us to grasp the key aspects of your project effectively.

Note:

- Ensure that the provided notebooks for computing metrics on the test set and the hold-out test set are clear, concise, and accurately documented. We will execute them to verify their functionality. Failure to run these notebooks successfully **will** lead to disqualification of the submission.
- The README.md file should include instructions for executing the notebooks and provide any necessary setup information. Make sure the instructions are straightforward and easy to follow.
- Your folder structure should be well-organised, allowing for easy navigation and comprehension of your codebase.
- Include a section in your README.md file that lists the research papers, blogs, YouTube videos, or any other relevant resources you utilised during the project. Provide a brief one-line mention of where each resource was applied.

We value your dedication to excellence and look forward to reviewing your outstanding work. If you require further clarification regarding anything, feel free to reach out to us at aayush@spoofsense.com.