Image Forensics with ELA, JPEG Ghost, Feature based methods

By Samuel Cheng
Yifan Lu

Introduction

Given there exists many "Fake images" influencing both science and politics, it is important for us to know if an image is manipulated or not. Not only just knowing if it's forged, but how and where it was modified. For simplicity, the goal of this project is to localize the tampered region in a digitally forged image with only the modification being copy-paste forgery. We attempted three algorithms: Error level analysis(ELA), JPEG Ghost detection, Feature matching based analysis.

Assumptions include but not limited to the following:

- There will be at most one region which has been copy-pasted
- For images without tampering, no mask will be returned
- A visually apparent musk will be generated by the algorithm to separate the two regions
- There will be some levels of resampling done on the pasted region and only nearest-neighbor and bilinear interpolation will be used.

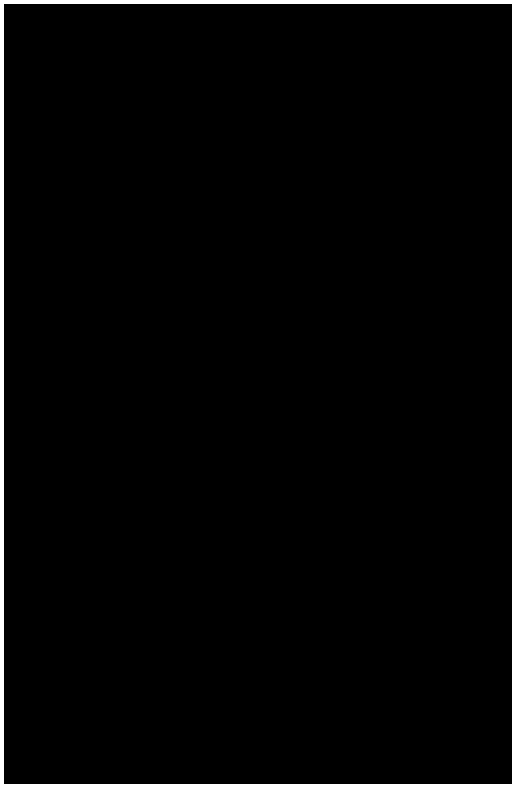
ELA

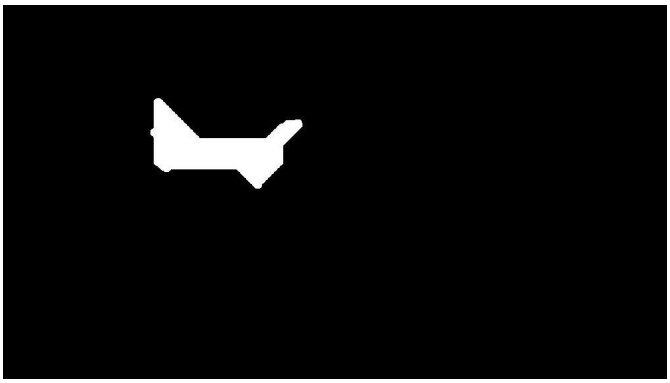
The theory behind this algorithm is that when a resampled region is spliced into the image, the resaved jpeg image will introduce errors at different levels than the rest of the image. When the image is modified, the 8x8 cells containing the resampled region will have higher error than any other unmodified cell [1].

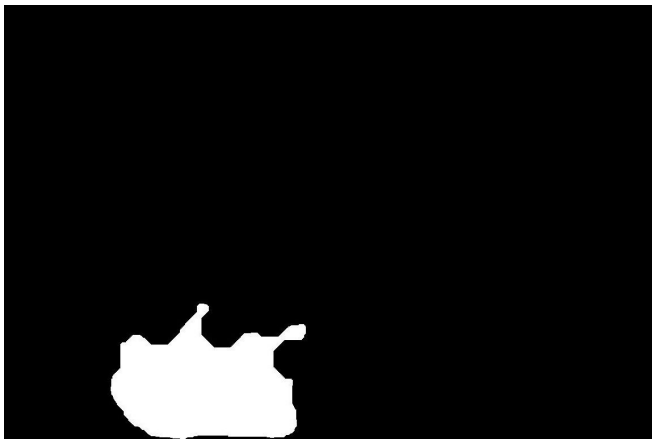
First we resaved the modified image at a 95% compression factor. Then we take the magnitude of the difference between the original image and the compressed image. The bright regions in the magnitude show that the region has been modified but is noisy. In order to produce the output mask, we first did a threshold of the error values to remove false-positives. The threshold was set to 128 but was hand-tuned for some applications. Next we want to connect the scattered components with dilation and erosion using a circular kernel. To get the largest component remaining, we eroded and then dilated again. The output should result in a binary mask

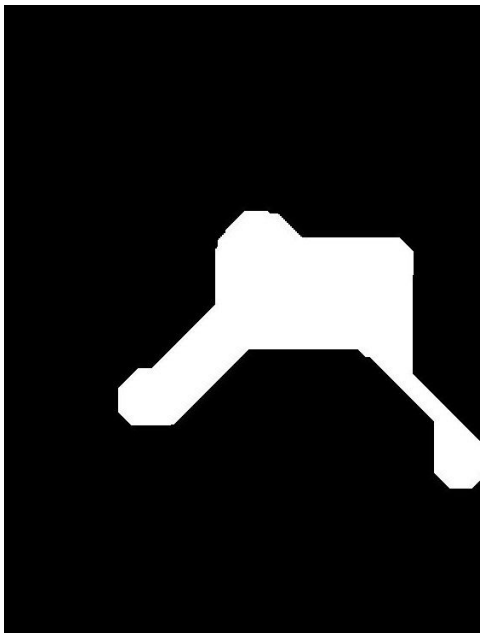
1. Resave image a 95% compression
2. Get magnitude of difference between resaved image and original
3. Threshold the error values
4. Morphological operations to the the output mask

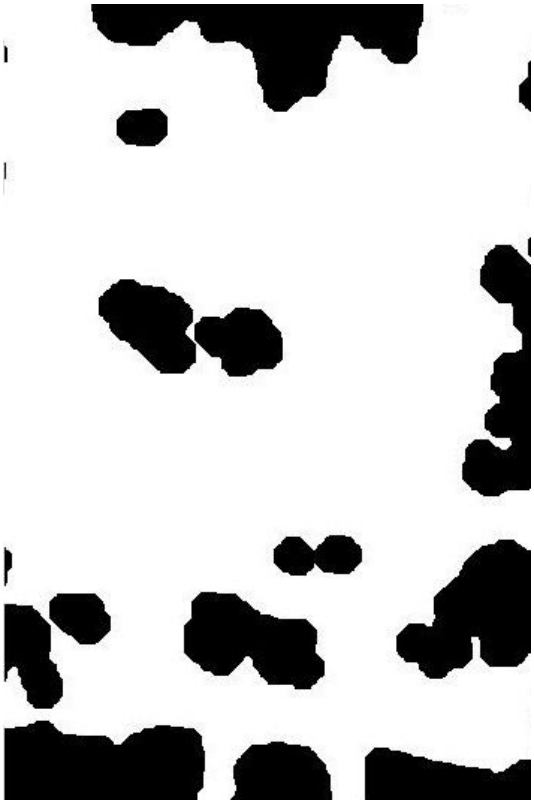
ELA Results

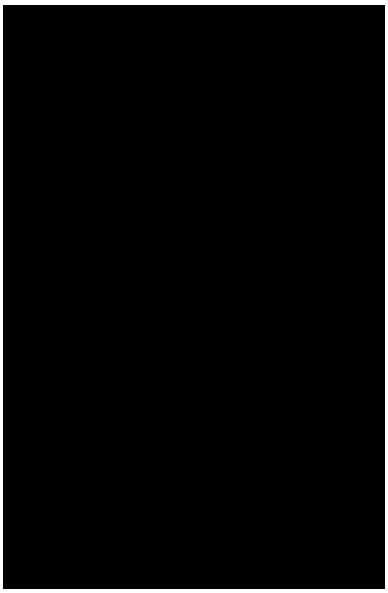


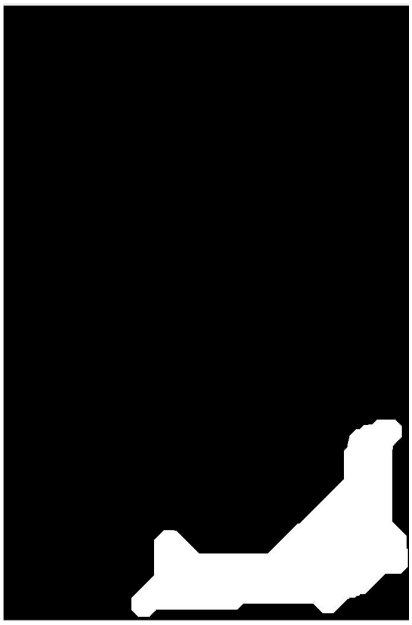










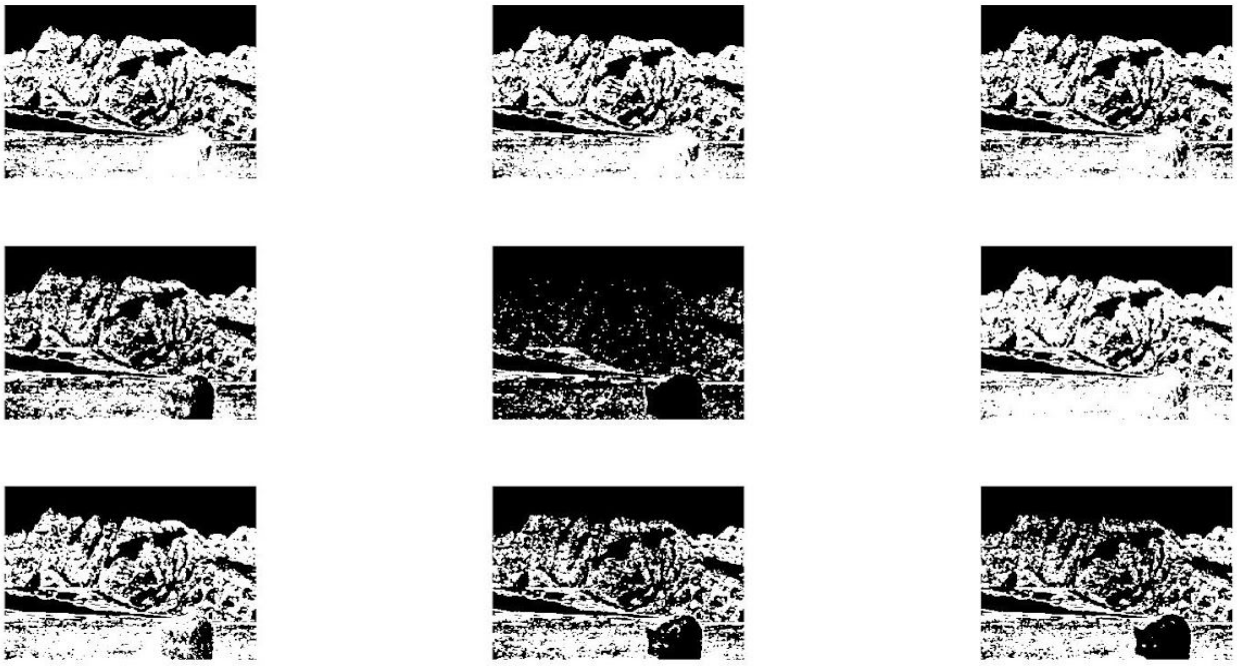


JPEG Ghost Detection

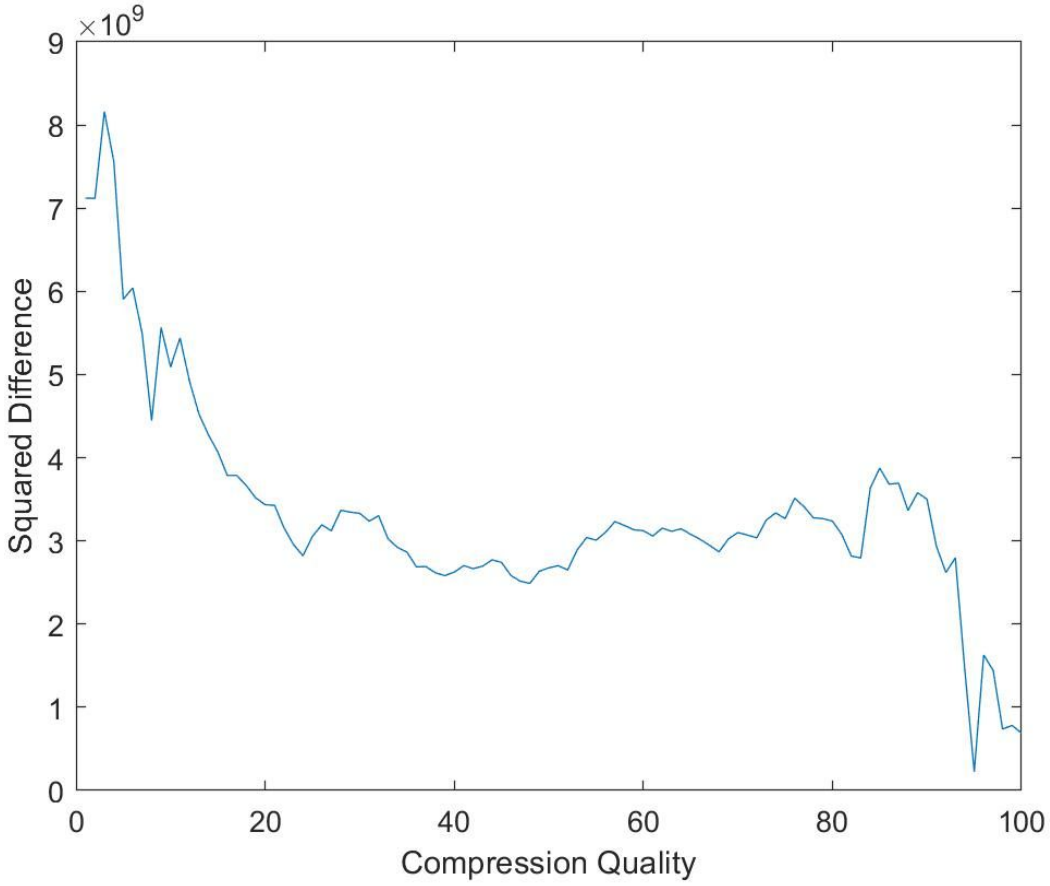
The second method we've attempted is by utilizing JPEG Ghost effect during compression. JPEG Ghost[3] mentioned in Farid's paper refers to the second minimum in the error vs quantization plot, where the first minimum reveals the original compression.

Sample tampered image shown below from the dataset

Shown below is the result of the masks with rewriting quality 90%-100%

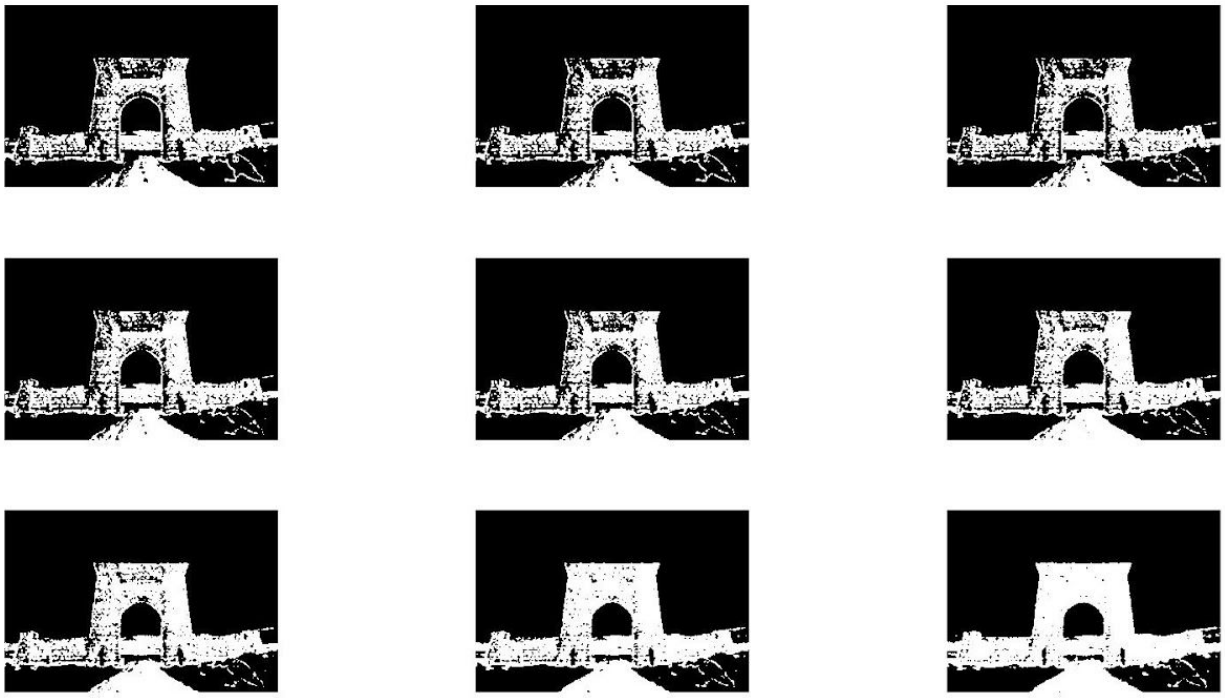


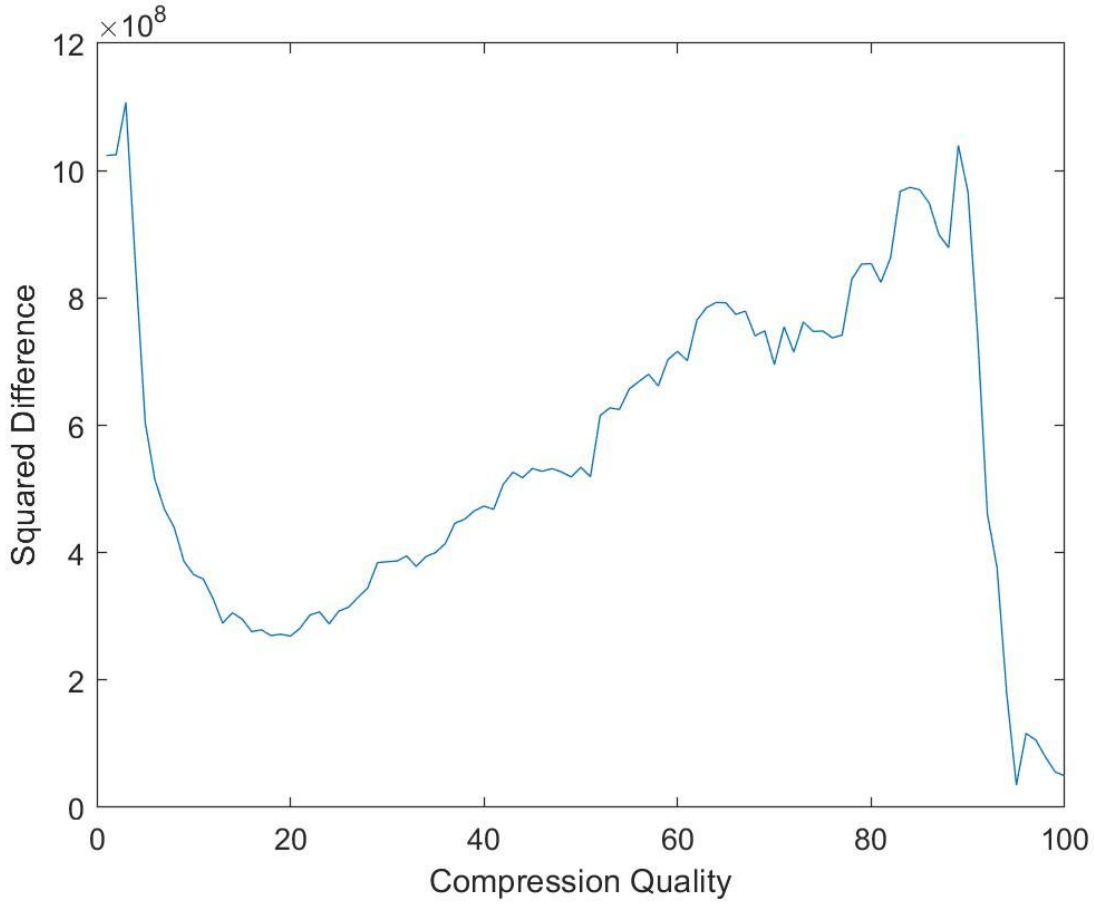
Shown below is Squared error vs second compression curve



As we can see from the above 2 plots, the second minima occurs at roughly 95% which points to the centered image of the 9 subimages. The mask returned in the middle of the 9 subplots is also the most visually apparent one.

Another bad example is shown below without the mask being properly generated due to unknown reason. The ELA algorithm also returns the wrong mask for this example.

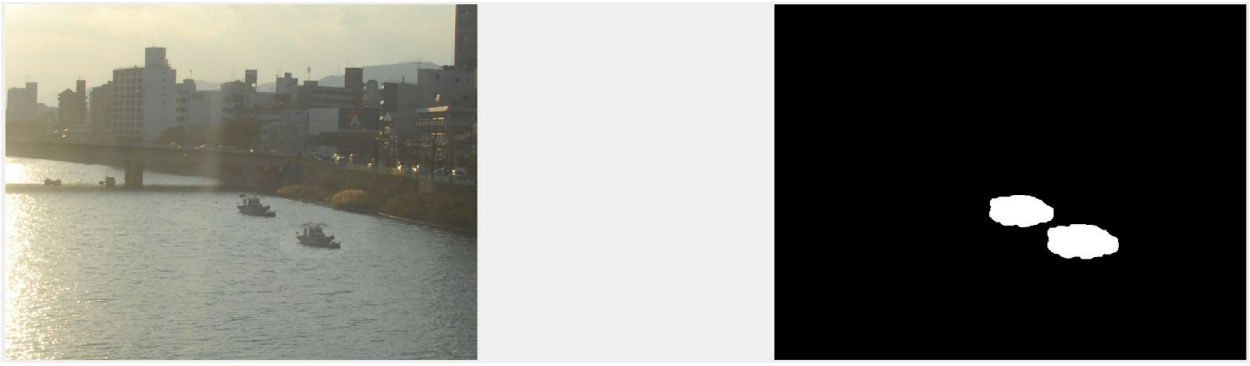



×10^8
12
10
8
6
4
2
0
Squared Difference
0
20
40
60
80
100
Compression Quality

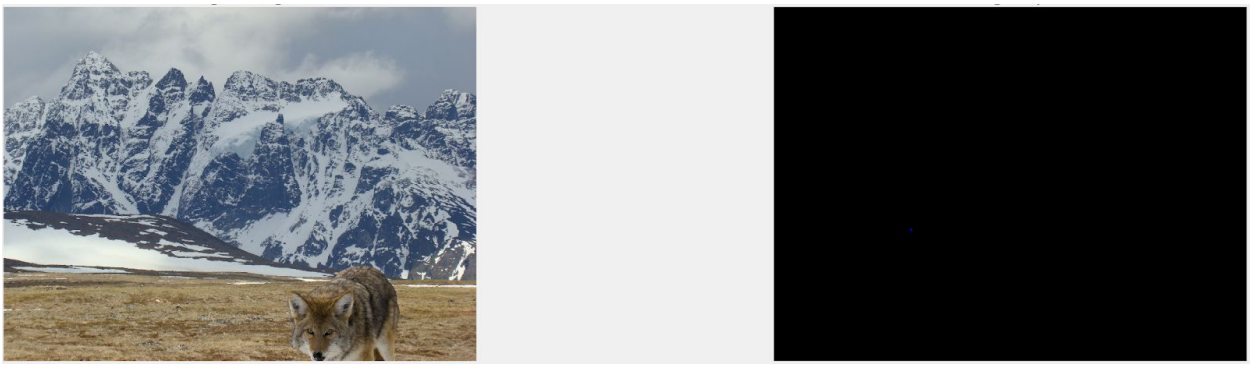
Feature Matching based analysis

This algorithm is looking for the tapered region utilizing rotation Invariant feature matching. The algorithm we used is from GRIP[5]. It uses Zernike polynomials for feature matching. This algorithm was the very initial approach we attempted before seeing the development data and we didn't realize the forged image comes from an outside donnor image instead locally generated. Therefore, this algorithm would not work with any of the dev and test data provided.

Shown below is a working working example with a copy-moved image



Below is an example of one of the test images



As we have stated, no mask is generated due to the fact no local copy-move forgery is done. The donner image comes from another source and therefore would not be detected.

Conclusions

We experimented with several methods for resampling detection. We noticed out of all the methods used for detection, the simplest method produced best results. Most of our work involved researching methods for resampling detection and tuning parameters to produce best results. To get the binary mask we used a series of morphological operations consisting of dilation and erosion. This process was fairly hand-tuned and future work can be properly identifying which component is the correct mask instead of eroding until one component remains. This usually results in a loss of mask details.

Reference

[1].Krawetz, Neil. "A Picture's Worth: Digital Image Analysis and Forensics"
[2]. A Matlab toolbox with implemented Forensics method
https://github.com/MKLab-ITI/image-forensics
[3]. Exposing Digital Forgeries from JPEG Ghosts Hany Farid, Member, IEEE,
https://farid.berkeley.edu/downloads/publications/tifs09.pdf
[4]. 'Efficient Dense-Field Copy–Move Forgery Detection', Cozzolino
[5]. 'Efficient dense-field copy-move forgery detection'
http://www.grip.unina.it/research/83-image-forensics/90-copy-move-forgery.html