A DEEP-LEARNING APPROACH FOR DETECTING SPLICING & COPY-MOVE IMAGE FORGERIES AND IMAGE RECOVERY

VIVA-VOCE (JAN 2023)

Aravind J (2019115017)
Krishnan S (2019115047)
Pranay Varma (2019115067)

Guide: Dr. K. Indra Gandhi

Department of Information Science and Technology,
College of Engineering Guindy,
Anna University, Chennai, Tamil Nadu, India

INTRODUCTION

- Digital picture usage has increased at a never-before-seen rate in our day and age, due to the proliferation of gadgets like smartphones and tablets.
- Furthermore, the development of user-friendly image manipulation software that is available at reasonable prices, has made the manipulation of such content easier than ever.
- Some of these images are tampered in such a way that it is absolutely impossible for the human eye to detect.

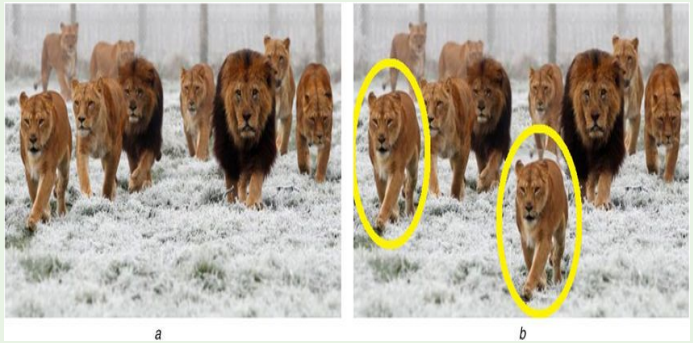
IMAGE TAMPERING METHODS

Three of the most common image manipulation techniques:

- **Copy-move:** A specific region from the image is copy pasted within the same image.
- **Splicing:** A region from an authentic image is copied into a different image.
- **Removal:** An image region is removed and the removed part is then in-painted.

IMAGE TAMPERING METHODS - Examples

Copy and Move



Splicing

Object Removal

PROBLEM STATEMENT

- The goal of this project is to detect and localize Splicing and Copy-Move forgeries in images using both supervised and unsupervised deep learning techniques.

- To achieve this two deep-learning approaches CNN and unsupervised self-consistency learning have been implemented on various image forensics datasets like CASIA2[6], Dresden[7] and In-the-Wild Image Splice Dataset dataset[8] and the performance of image forgery detection for each approach is analysed based on the test sample difficulty.

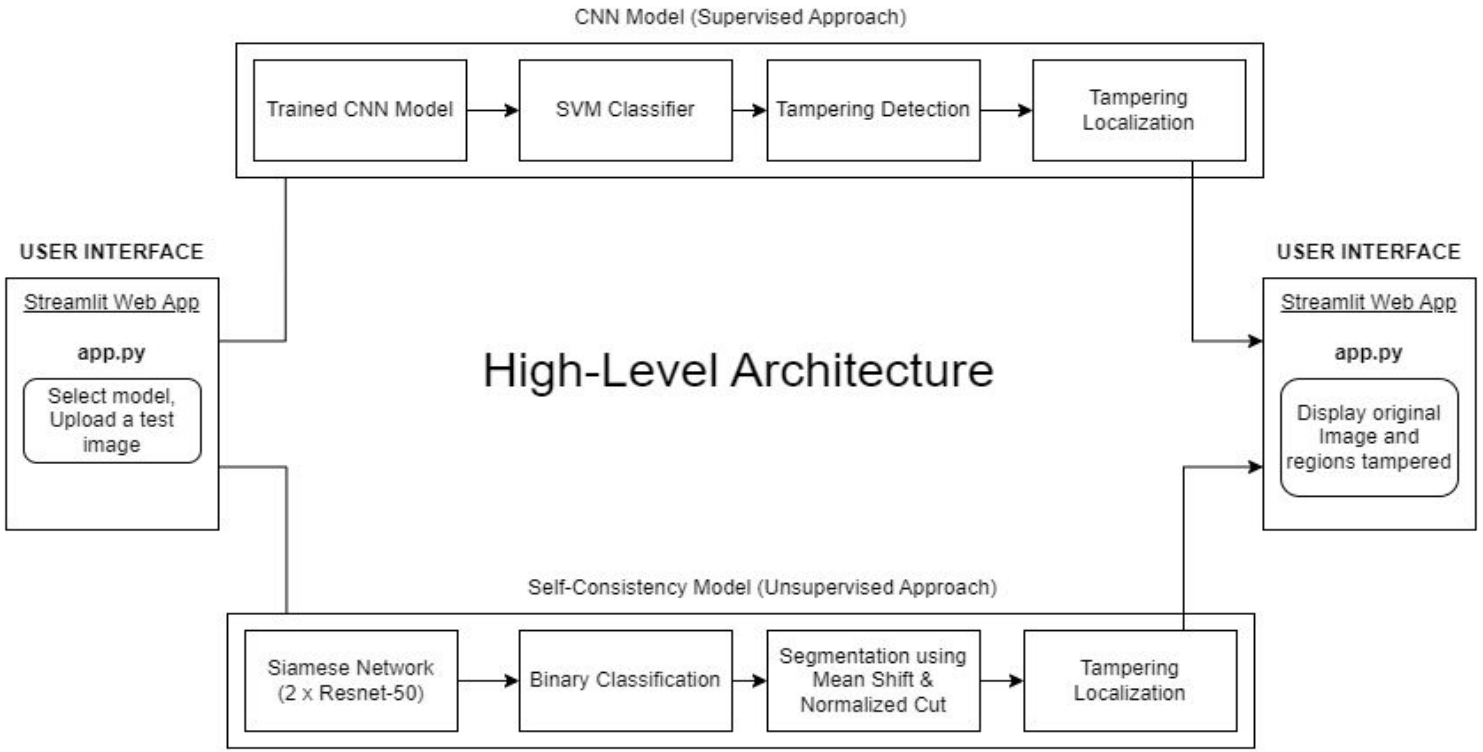
- To develop a web application, which allows end-users to upload a test image and find the region of forgery, if any.

LITERATURE SURVEY

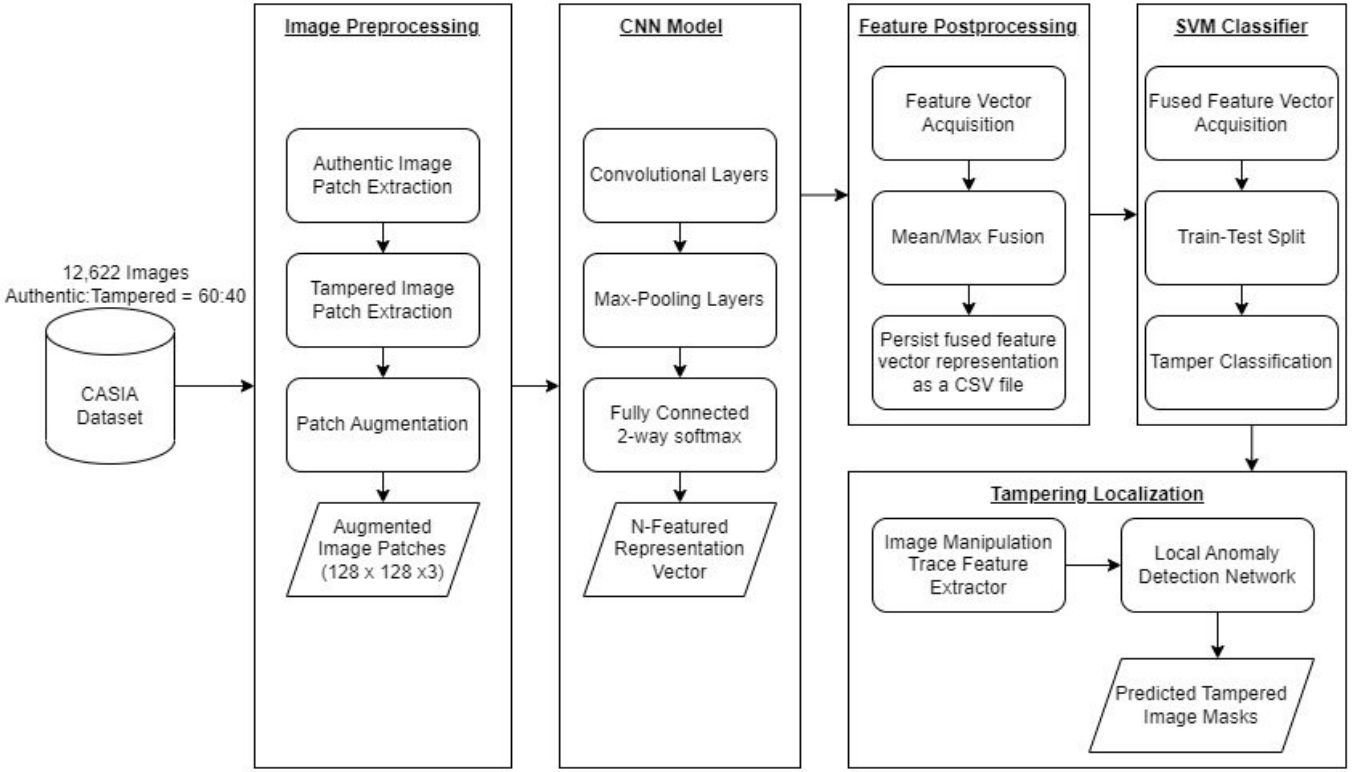
SNo	Title of the Paper	Year	Methodologies/Approach used	Pros	Cons
[1]	An Efficient CNN Model to Detect Copy-Move Image Forgery	2022	Using CNN feature extraction is done followed by max-pooling layer and then the classification stage is called to classify data.	The proposed architecture is computationally lightweight.	The accuracy of classification decreases when the samples are challenging.
[2]	Copy Move and Splicing Image Forgery Detection using CNN	2022	Pre-processing and then error analysis and using CNN to predict output.	More time efficient.	The model does not easily generalize to datasets with different underlying distributions.
[3]	A Deep Learning Approach to Detection of Splicing and Copy-Move Forgeries in Images	2016	A new deep learning-based image forgery detection method that uses a CNN to automatically learn hierarchical representations from input RGB colour images has been presented.	Outperforms many state of the art models, in terms of speed and accuracy.	The performance of the model deteriorates for more challenging image forgery datasets.

SNo	Title of the Paper	Year	Methodologies/Approach used	Pros	Cons
[4]	Fighting Fake News: Image Splice Detection via Learned Self-Consistency	2018	The proposed algorithm uses the automatically recorded photo EXIF metadata as supervisory signal for training a model to determine whether an image is self-consistent — that is, whether its content could have been produced by a single imaging pipeline. This self-consistency model has been used for detecting and localizing image splices.	The proposed method obtains state-of the-art performance on several image forensics benchmarks, despite never seeing any manipulated images at training.	i) The model is not well-suited to finding very small splices. ii) Over- and underexposed regions are sometimes flagged by the model to be inconsistent because they lack any meta-data signal.
[5]	Detecting Tampered Regions in JPEG via CNN	2020	This paper proposes a method for using CNN to detect the tampered region in a JPEG image. The DCT coefficients are provided as input, and the output is a binary segmented image in which the tampered and non-tampered regions are represented by contrasting white and black regions.	The tampered part was found accurate to a great extent, and the F-measure of their method is approximately 2.3 times that of the MDBD method.	The model flagged some authentic images as tampered and inaccurately identified region of tampering in authentic images.

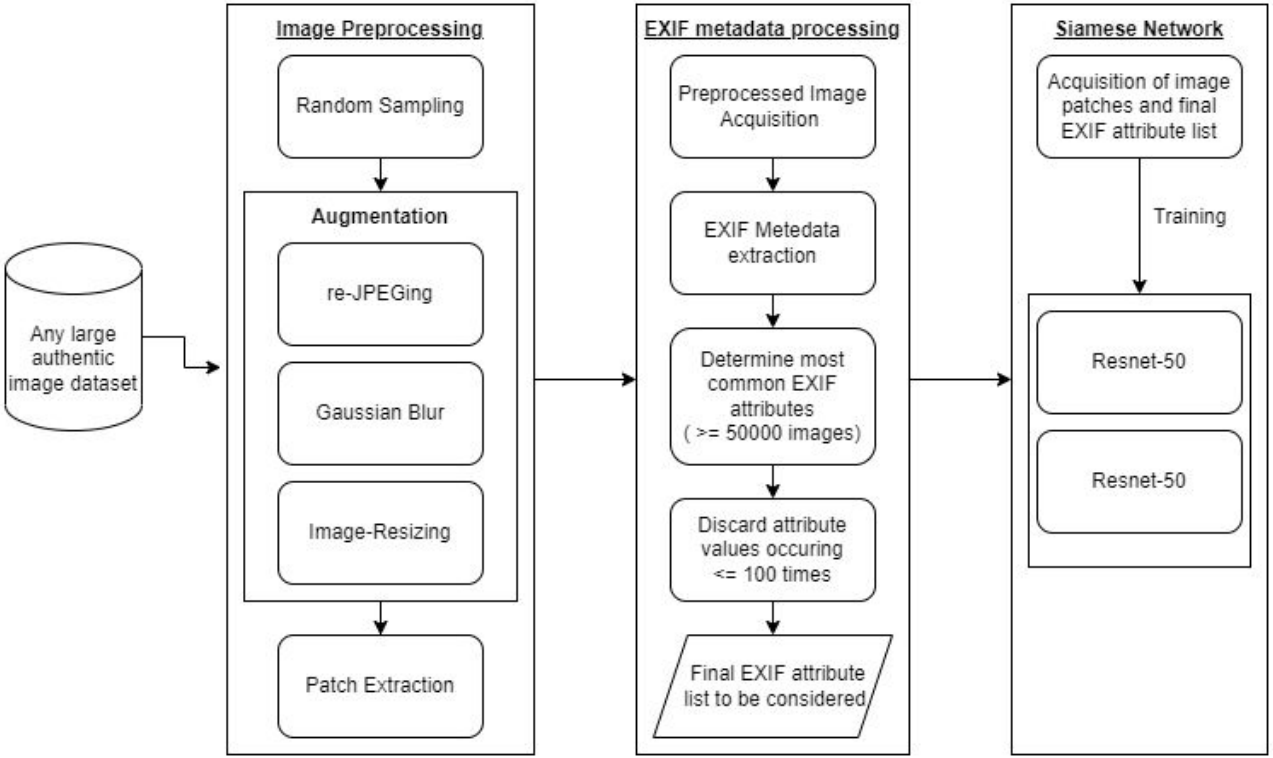
TECHNICAL ARCHITECTURE - High Level Design



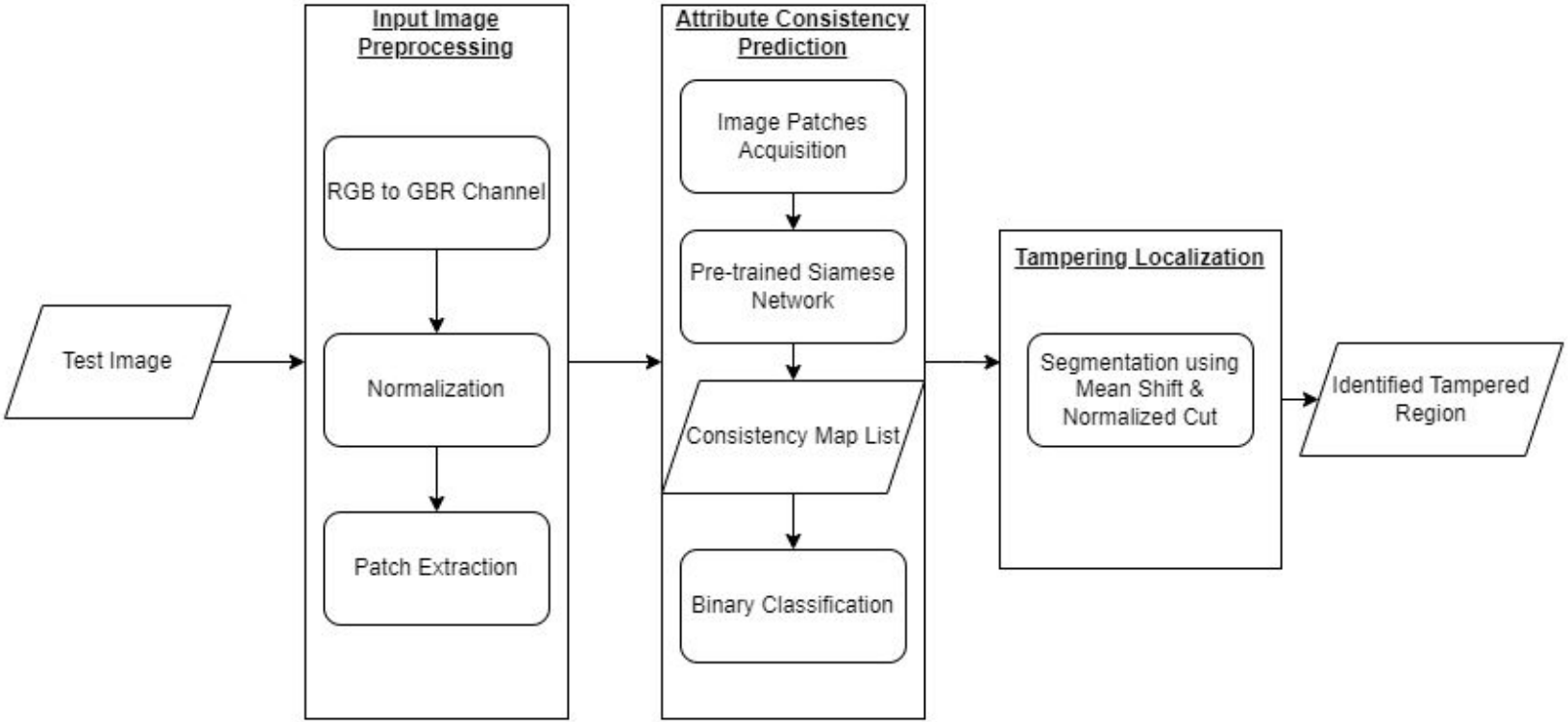
CNN APPROACH ARCHITECTURE



SELF-CONSISTENCY MODEL TRAINING



SELF-CONSISTENCY LOCALIZATION



MODULES

1. **CNN Approach**

- PATCH EXTRACTOR
- CNN MODEL AND TRAINING
- FEATURE EXTRACTOR AND SVM CLASSIFIER
- TAMPERED REGION LOCALIZATION

2. **SELF CONSISTENCY LEARNING Approach**

- INPUT IMAGE PREPROCESSING AND CONSISTENCY MAP EXTRACTION
- SIAMESE NETWORK
- IMAGE SEGMENTATION USING MEAN SHIFT AND NORMALIZED CUT

MODULAR DIAGRAMS & ALGORITHMS

1. CNN Approach

The images in the CASIA dataset's authentic and tampered classes are first preprocessed , before **patches** of 128 x 128 x 3 are extracted. These patches are then provided as input to the **CNN model**, producing an **N-featured representation vector** which is fused to a single feature vector. This fused vector is then fed into the **SVM classifier**, which determines whether or not the given image has been tampered with. An image manipulation trace feature extractor and a local anomaly detection network have been used to determine the **region of tampering.**

PATCH EXTRACTOR - Modular Diagram

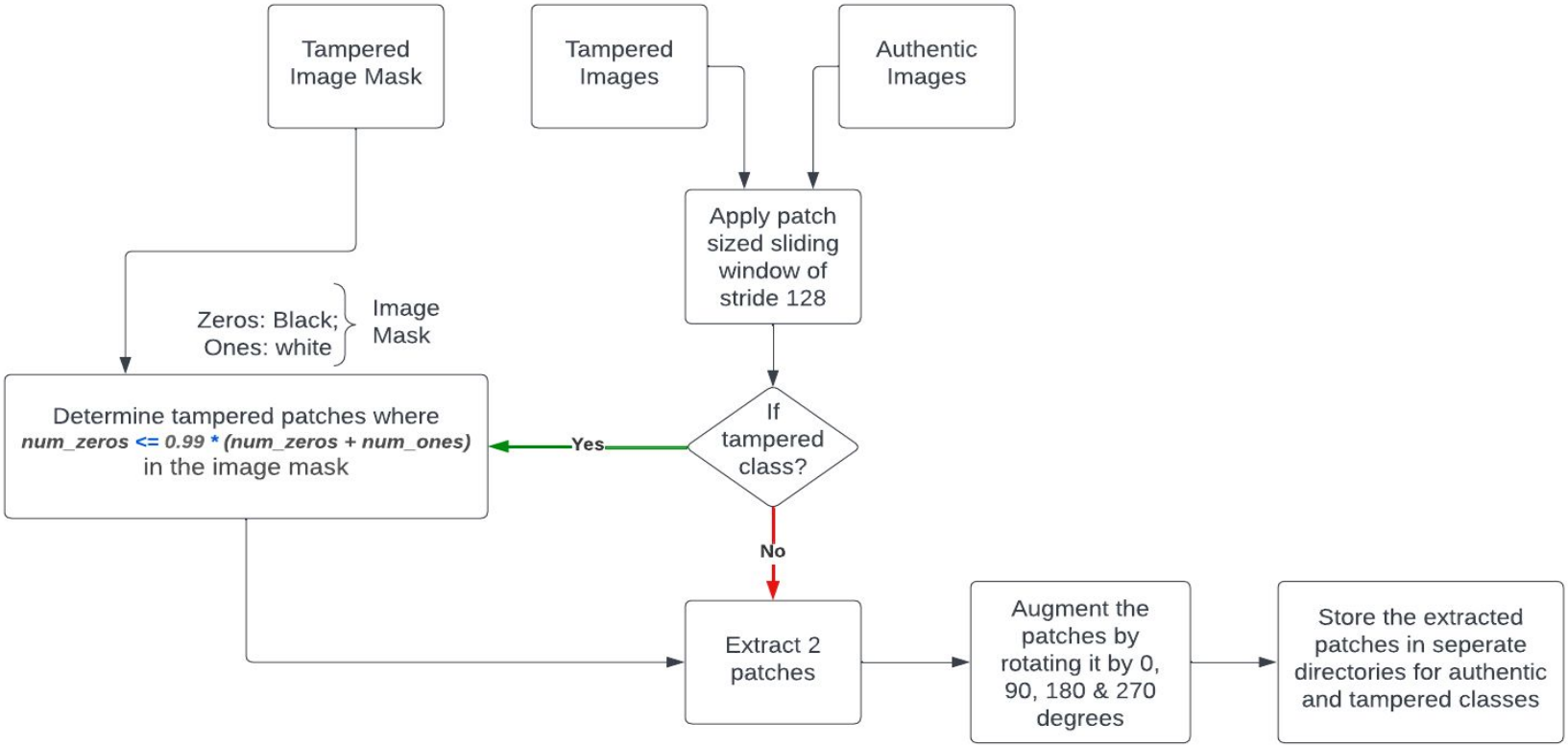


IMAGE PATCH EXTRACTOR - Algorithm

INPUT: input path,output path,patches per image,no of rotations,stride

OUTPUT: Rotated image patches

1: START

2: FOR each image in Tampered Images and Authentic Images DO

3: Apply patch-sized sliding window of stride 128

4: IF image belongs to Tampered Images THEN

5: Determine tampered patches where num zeros ¡= 0.99 * (num zeros + num ones)

6: END IF

7: Augment the patches by rotating them by 0, 90, 180 270 degrees.

8: GOTO 2

9: END FOR

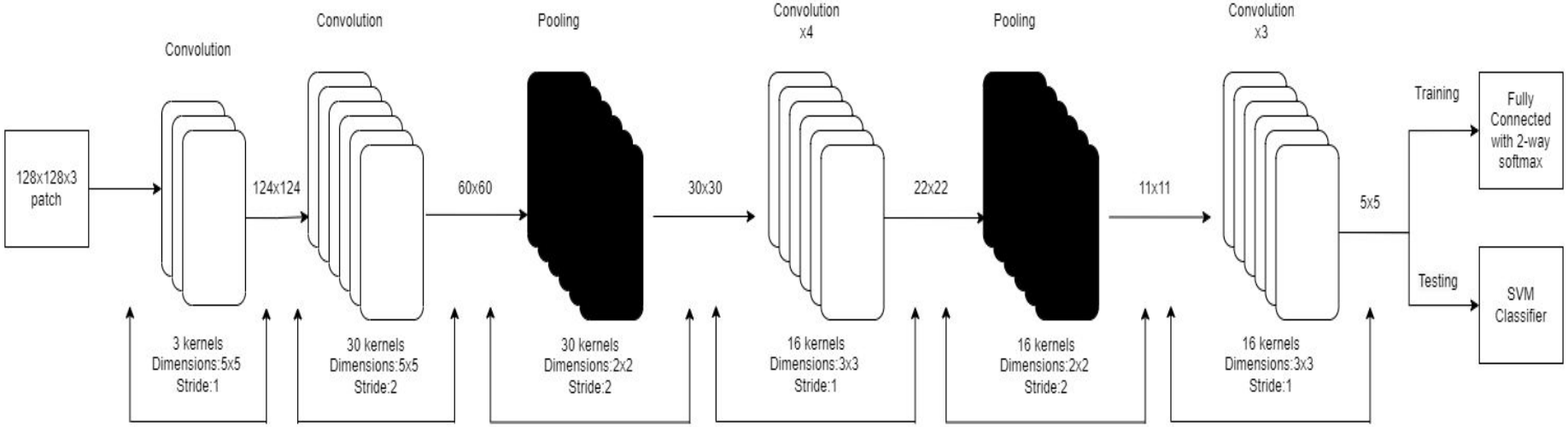
10: Store the extracted patches in separate directories for authentic and tampered classes.

11: STOP

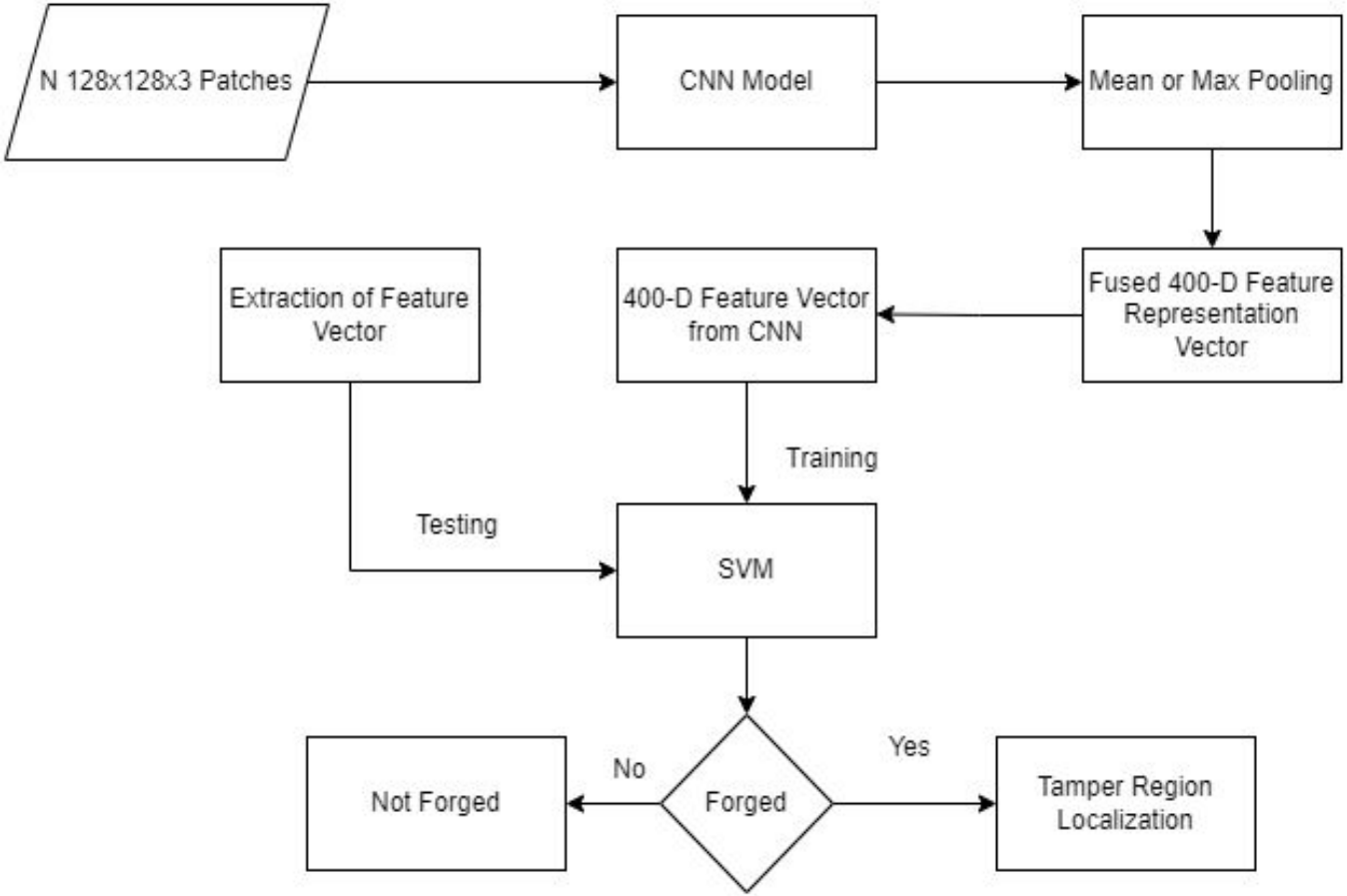
CNN MODEL AND TRAINING

- The model consists of **9 convolution** and **2 pooling layers**. Depending on the layer involved, the kernel size is fixed as either 3x3, 2x2 or 5x5 with a stride of 1 or 2.
- The convolution layers extract features from the input matrices, while the pooling layers perform down-sampling or dimensionality reduction of the features.
- The **ReLu activation** function is used by each of the convolution layers.
- The model is trained for **250 epochs**.

CNN Architecture



FEATURE EXTRACTOR AND SVM CLASSIFIER - Modular Diagram



FEATURE EXTRACTION AND SVM CLASSIFICATION - Algorithm

INPUT: 128x128x3 image patches

OUTPUT: 1 or 0 (Binary Classification)

1: START

2: The patches are fed into the CNN model, which extracts a 400-D feature representation for each patch.

3: The (n x 400-D) feature representations for an image must be fused into a single feature vector

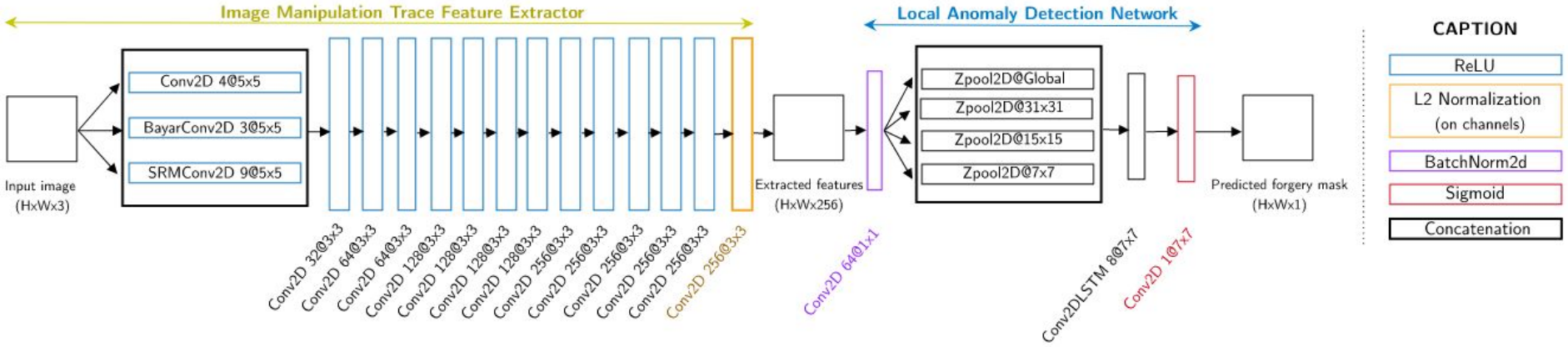
4: These features are then passed to a fully-connected layer with a 2-way softmax classifier in the training phase and the SVM model in the testing phase.

5: The SVM model returns 1 if the image is tampered, and 0 otherwise.

6: STOP

TAMPERED REGION LOCALIZATION

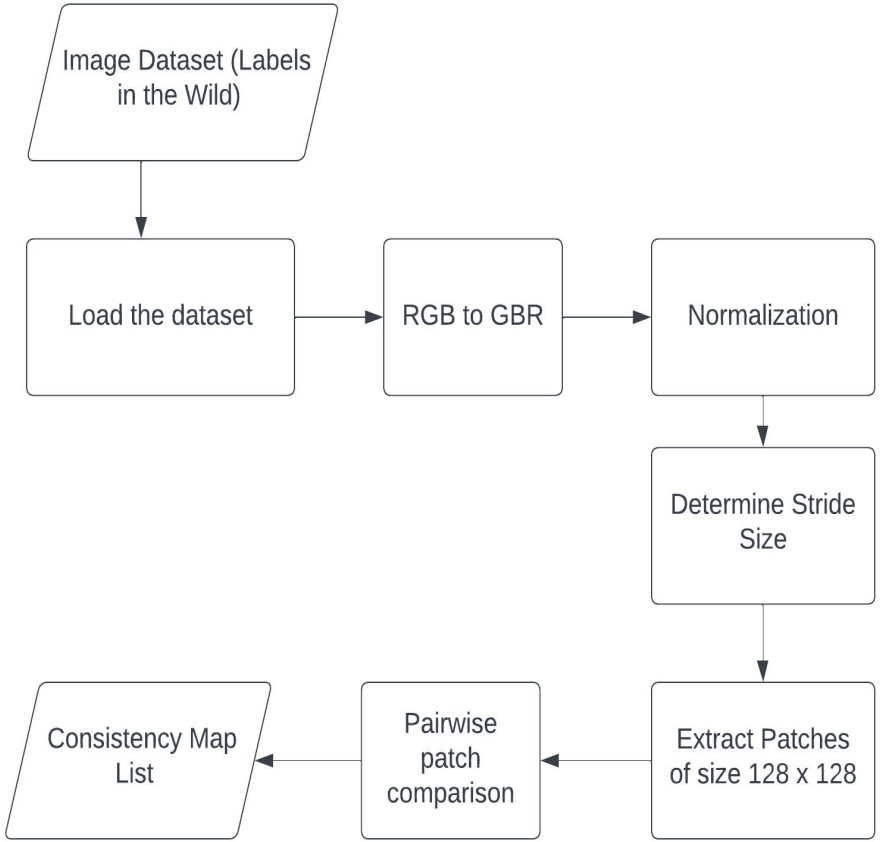
- To identify the region of tampering, a machine learning based technique called MantraNet has been used.
- A testing image is used as the input, and the model is used to predict a **pixel-level forgery likelihood map** as the output.
- MantraNet is comprised of two smaller networks : an **Image Manipulation Trace Feature Extractor** and a **Local Anomaly Detection Network.**



2. SELF CONSISTENCY LEARNING APPROACH

- First, the input dataset is **preprocessed** by applying random sampling and image augmentation techniques to get a subset of well-distributed, augmented images. Patches are generated from the preprocessed images.
- These extracted image patches as passed as input to the pre-trained **Siamese Network**, which returns a consistency map list after performing a pairwise consistency check of all the patches of the image.
- Using the obtained consistency map list, **segmentation** methods such as Mean Shift and Normalized Cut are applied to the input image to determine the exact region of tampering.

INPUT IMAGE PREPROCESSING AND CONSISTENCY MAP EXTRACTION



- The dataset is loaded and the RGB channels are converted to **GBR** format as OpenCV reads the images in GBR format.
- **Normalization** to restrict the pixel values between 0 & 1.
- Extracting patches of size **128 x 128** based on stride size.
- The consistency maps are generated by comparing each patch in the first patch list with each patch in the second patch list.

INPUT PREPROCESSING & CONSISTENCY MAP EXTRACTION - Algorithm

INPUT: Test Images

OUTPUT: Consistency Map List

1: START

2: Load the images to be tested.

3: Convert RGB to GBR colour scheme.

4: Unsqueeze the image's dimensions from (w, h, 3) to (1, 3, w, h).

5: Calculate stride size based on the image dimensions.

6: Apply patch-sized sliding window to extract patches of size 128 x 128.

7: Compare the obtained patches pairwise and get the probability score of consistency.

8: Obtain the consistency map list.

9: STOP

SIAMESE NETWORK

- The Siamese network is used to predict the probability that a pair of each EXIF field in 128x128 image patches has the same value attribute.
- The ResNet 50 is the classical neural network used here. It is a predefined model available in pytorch which can be trained on the input dataset to predict the results.
- The network predicts the probability that the images share the same value for each of the n metadata attributes.

SIAMESE NETWORK - Architecture

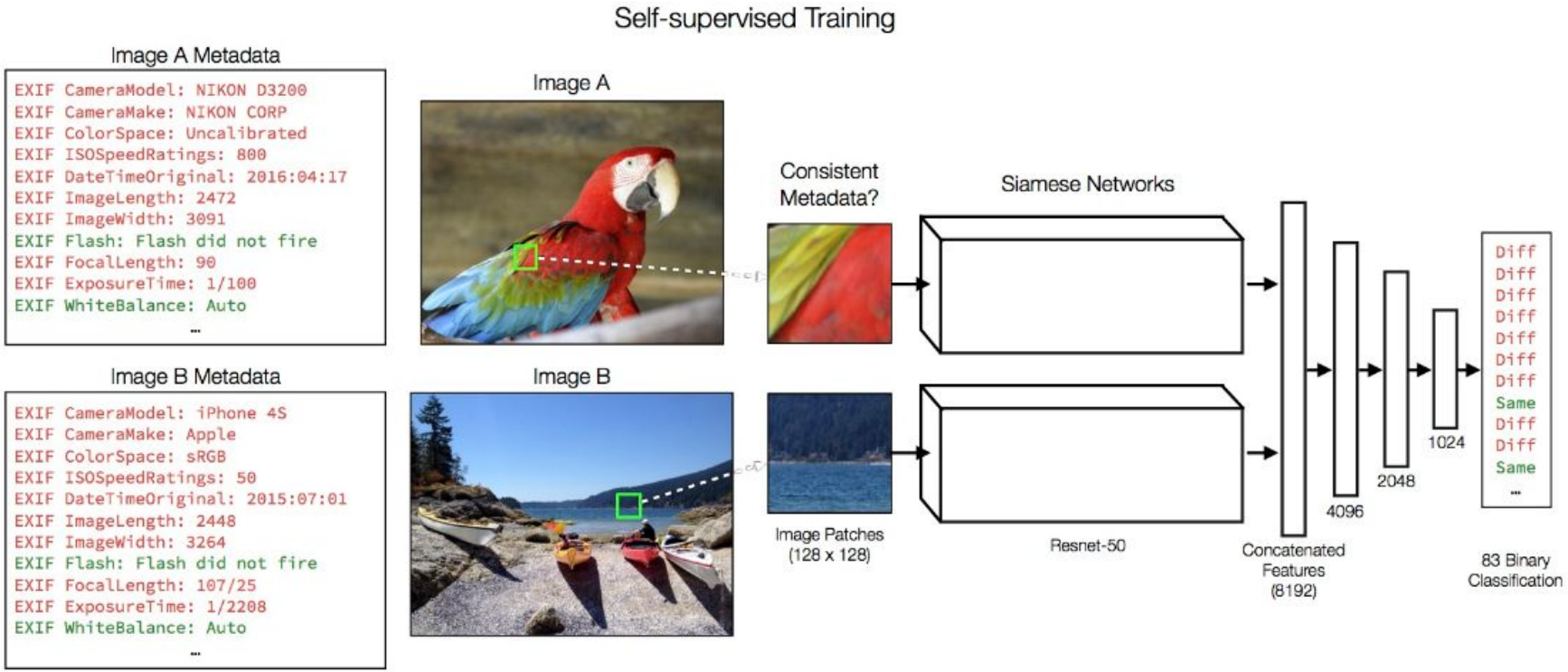


IMAGE SEGMENTATION USING MEAN SHIFT AND NORMALIZED CUT

- The image is segmented into two parts (original and spliced).
- After the consistency maps for two patches in an image for all EXIF attributes are returned, the points in the resultant map are plotted and the mean shift is calculated.
- The normalized cut computation makes use of the sklearn spectral clustering function to return the fit.

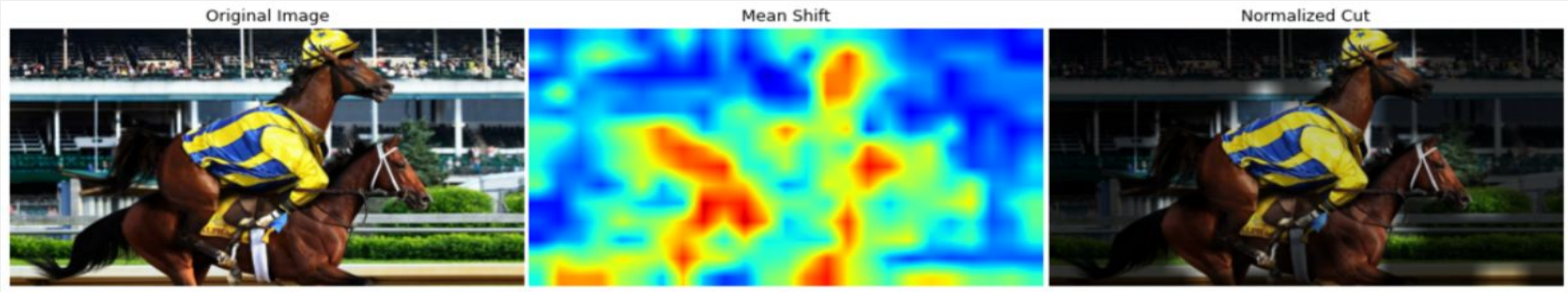


IMAGE SEGMENTATION - Algorithm

INPUT: Image, Image Patches

OUTPUT: Segmented Images using Mean Shift and Normalized Cut

1: START

2: Compute the consistency map of a patch with respect to other patches considering each metadata attribute independently.

3: The resultant consistency map is used to plot the mean shift, taking the top 10 percentile of nearest points into consideration for a given point.

4: The normalized cut is obtained from the consistency maps using the spectral clustering method.

5: If most of the image is high probability, flip it.

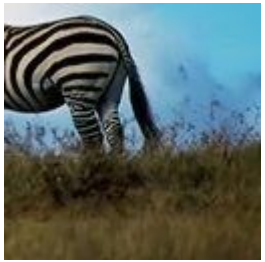
6: The resultant images for mean shift and normalized cut are resized, showing the segments clearly.

7: STOP

RESULTS

The following image shows a sample output for **patch extraction** and **patch augmentation**:

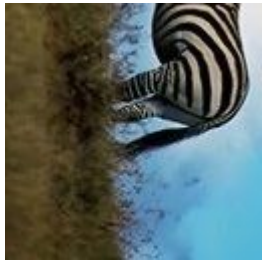
a) Original Image



b) Rotate by 0 deg.

c) Rotate by 90 deg.

d) Rotate by 180 deg.

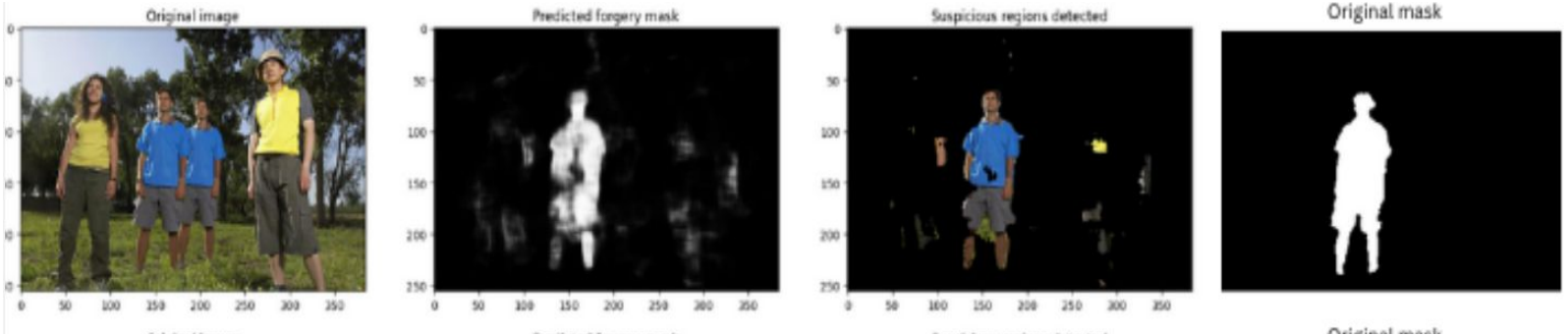


e) Rotate by 270 deg.

COPY MOVE FORGERY LOCALIZATION - CNN

The following image shows a sample output for copy-move forgery localization using CNN

- The **original image** is shown first.
- The **predicted forgery mask** generated by the MantraNet model appears next, using which the **suspicious regions** have been highlighted by preserving their colour.
- Finally, the **original mask** (highlighting the region that has been reproduced) is shown.

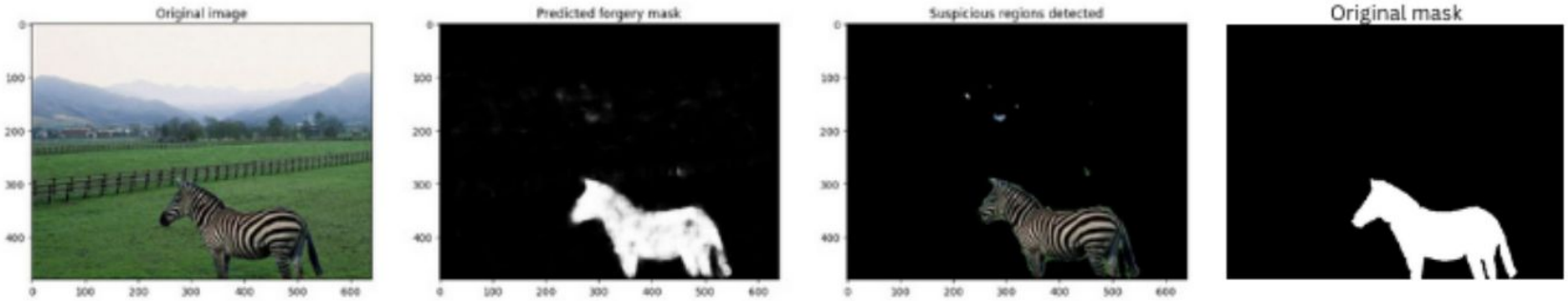


Sample CNN Tampered Region Localization Output (Copy Move Forgery)

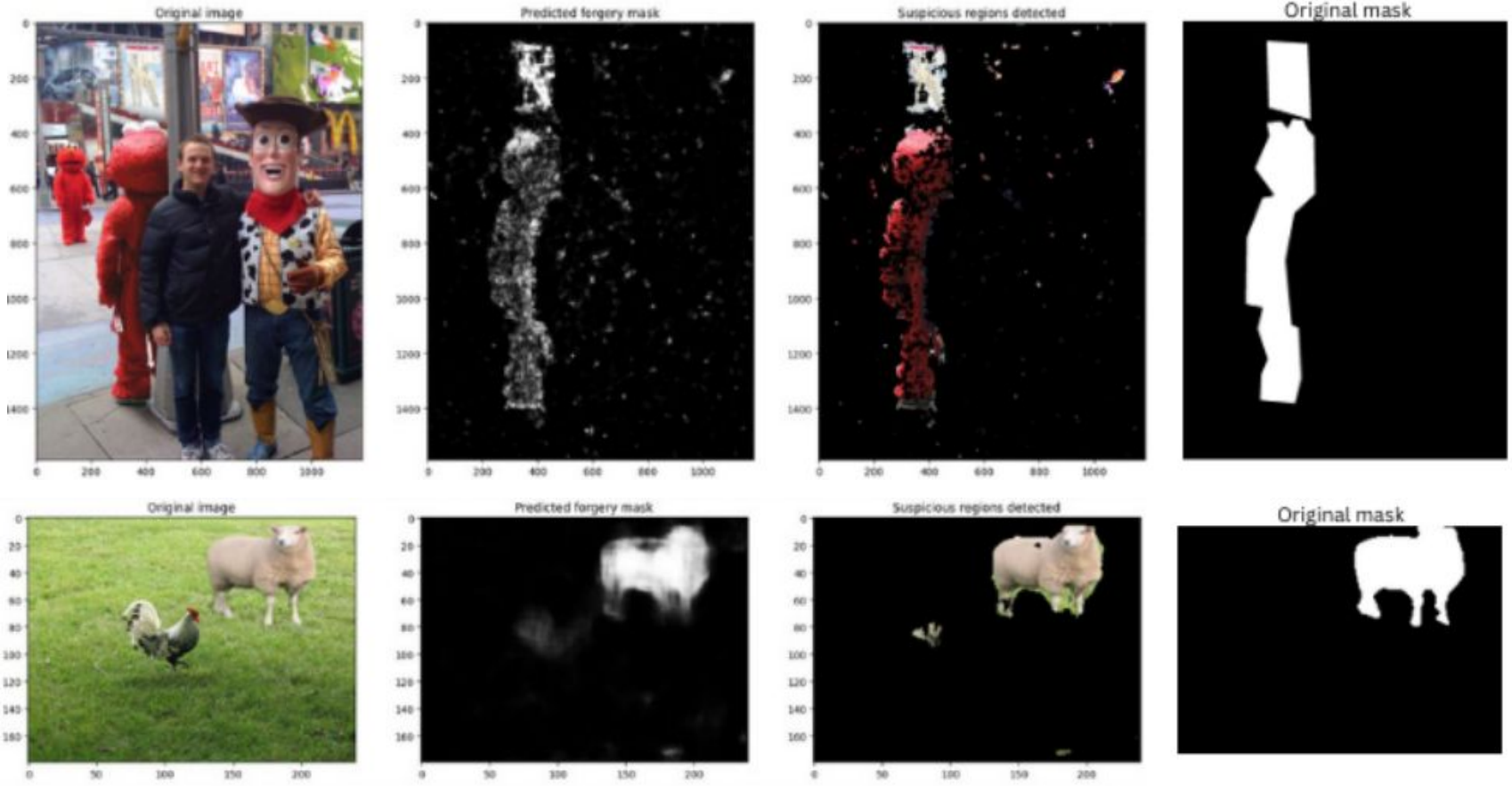
A few more examples for Copy Move Forgery

SPLICING FORGERY LOCALIZATION

The sequence of images shown below is similar to that of the previous example, with the difference lying in the fact that the **forgery mask**, **suspicious region** and **original mask** highlight the **spliced regions** as opposed to reproduced regions.

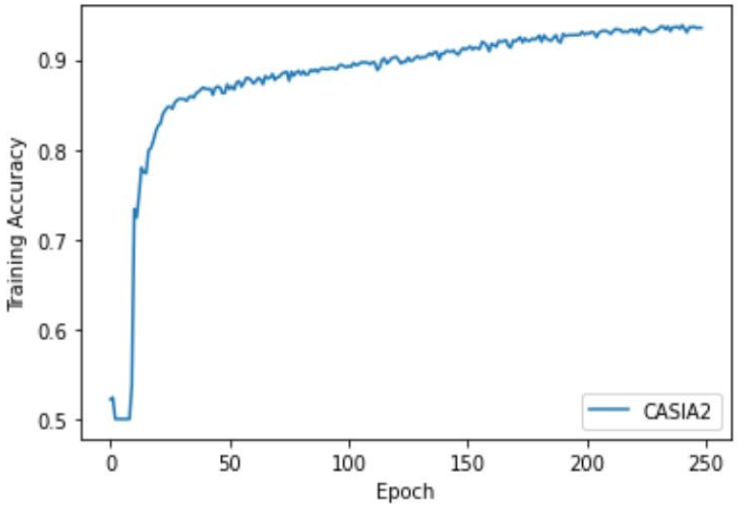


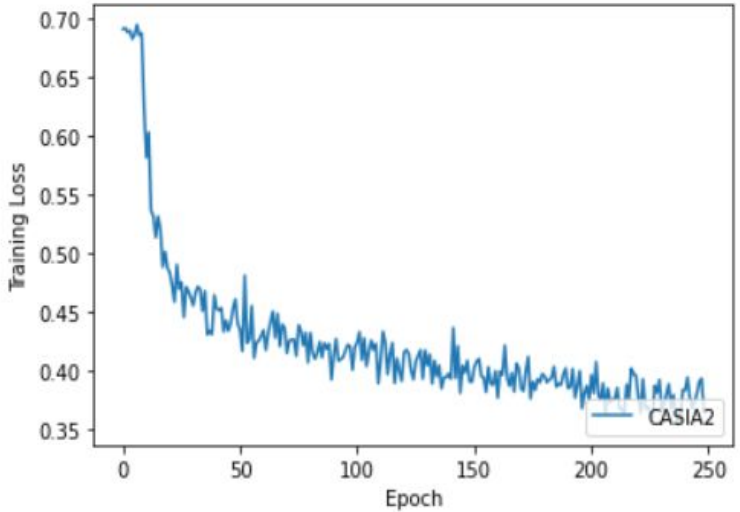
Sample CNN Tampered Region Localization Output (Splicing Forgery)


Original image
Predicted forgery mask
Suspicious regions detected
Original mask
Original image
Predicted forgery mask
Suspicious regions detected
Original mask

CNN TRAINING ACCURACY

From the graph we can infer that as the Epoch increases the training accuracy also increases and reaches a saturation after which the training accuracy doesn't change much. So the number of epoch is stopped at **250** to prevent overfitting.

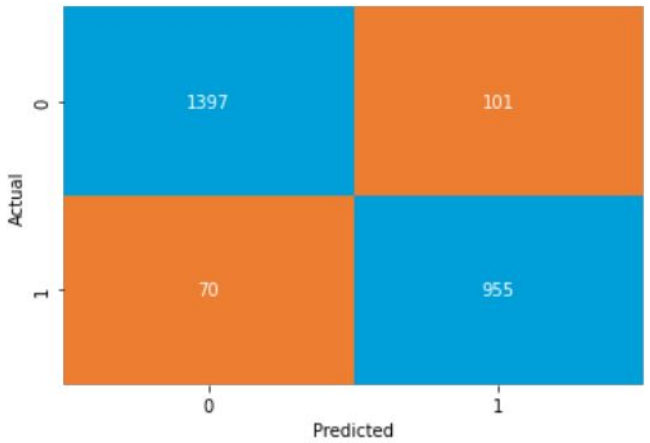




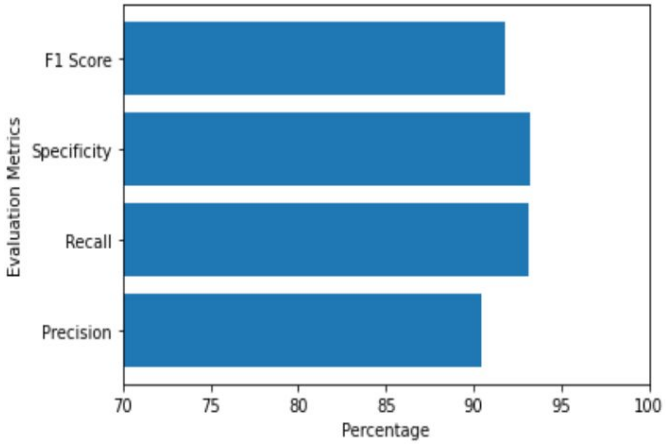
SVM CLASSIFIER PERFORMANCE

The SVM Classifier achieves the following values for the evaluation metrics:

Precision: 90.43%
Recall : 93.1%
Specificity: 93.25%
F1 Score : 91.74%



Confusion Matrix

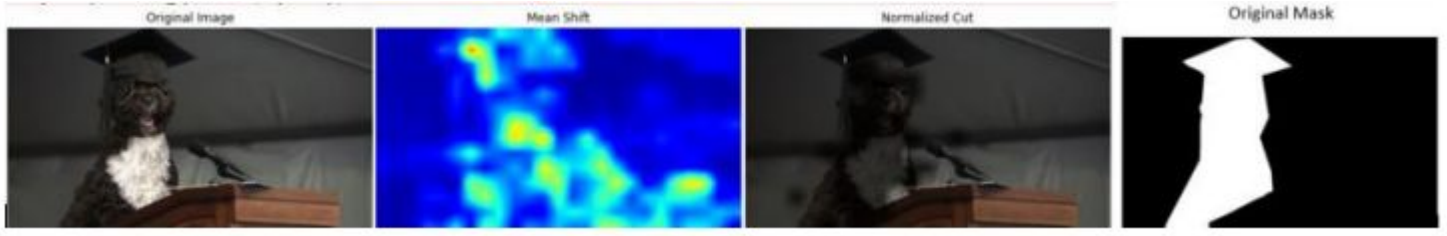


Evaluation Metrics vs Percentage

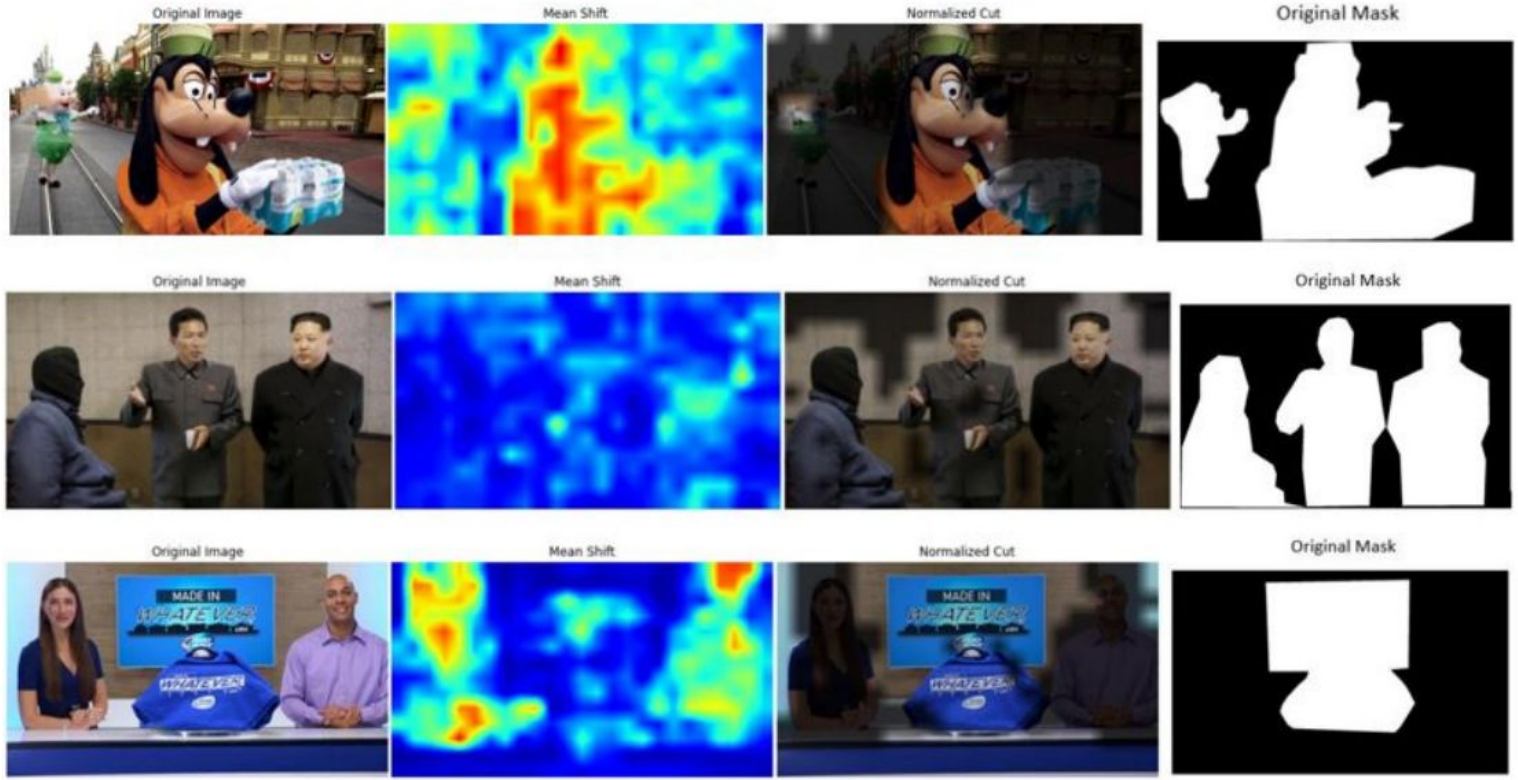
SELF CONSISTENCY LEARNING

The following image shows a sample output for splicing forgery detection using self-consistency learning.

The **original image** appears first, followed by the computed **mean shift** and **normalized cut** which attempt to separate the spliced regions. Finally, the **original splice mask** appears for comparative purposes.



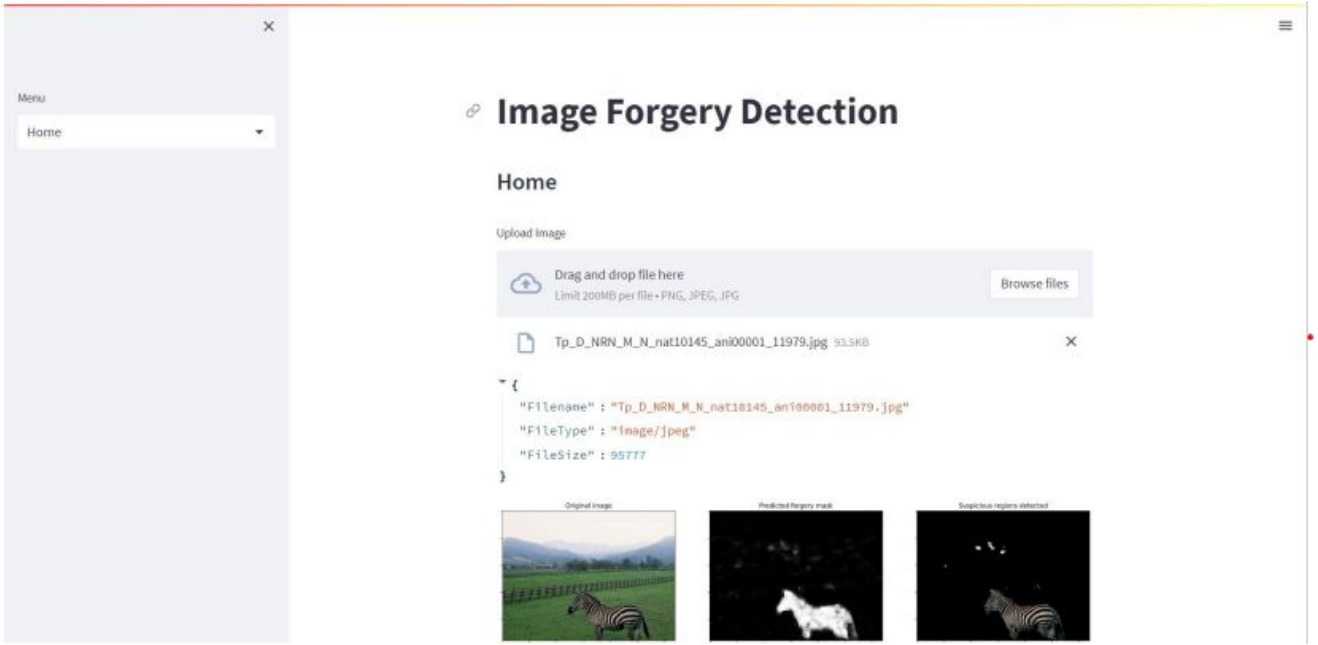
The model performs well on images with more subtle forgeries, with drawbacks including the detection of forgeries on under and over exposed images.



A few more examples for self consistency learning

WEB APPLICATION UI

- The UI of the project is made using Streamlit, an open-source library available in python.
- The web app prompts the user to upload an image for testing. On image upload, the web app runs the deep learning model in the background to localize the exact region of tampering, if any in the uploaded image.



CONCLUSION

- The CNN approach was found to be more robust in detecting both splicing and copy-move image forgeries, whereas the self-consistency learning approach could only detect splicing image forgeries.
- When tested with the ’Label in the Wild’ dataset, it was discovered that the CNN approach did not perform well when the tamperings in the images were much more complex and difficult to be identified by the human eye.
- The Self-Consistency Learning approach, on the other hand, could detect much more complex splicing tamperings in images, but the total time taken to localise the region of forgery is significantly longer when compared to the CNN approach.

REFERENCES

[1] K. M. Hosny, A. M. Mortda, M. M. Fouda and N. A. Lashin, "An Efficient CNN Model to Detect Copy-Move Image Forgery," in IEEE Access, vol. 10, pp. 48622-48632, 2022, doi: 10.1109/ACCESS.2022.3172273.

[2] Mallick, D., Shaikh, M., Gulhane, A. and Maktum, T., 2022. Copy Move and Splicing Image Forgery Detection using CNN. In ITM Web of Conferences (Vol. 44, p. 03052). EDP Sciences.

[3] Y. Rao and J. Ni. A deep learning approach to detection of splicing and copy-move forgeries in images. IEEE international workshop on information forensics and security (WIFS), pages pp. 1–6, 2016.

[4] Liu A. Owens A. Huh, M. and A.A. Efros. Fighting fake news: Image splice detection via learned self-consistency. In Proceedings of the European conference on computer vision (ECCV), pages pp. 101–117, 2018.

[5] N. Takeda T. Hirose K. Taya, N. Kuroki and M. Numa. Detecting tampered regions in jpeg images via cnn. 18th IEEE International New Circuits and Systems Conference, pages pp. 202–205, 2020.

[6] Casia 2.0 image tampering detection dataset. https://www.kaggle.com/datasets/divg07/casia-20-image-tampering-detection-dataset.

[7] Dresden image dataset. https://www.kaggle.com/datasets/ hanjunyang1/dresden.

[8] In-the-wild image splice dataset. : The dataset consists of 201 images scraped from THE ONION, a parody news website, and REDDIT PHOTOSHOP BATTLES, an online community of users who create and share manipulated images.

REFERENCES

[9] Bhavsar A. Kumar, A. and R. Verma. Forgery classification via unsupervised domain adaptation. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision Workshops, pages pp. 63–70, 2020.

[10] W. AbdAlmageed Y. Wu and P. Natarajan. Mantra-net: Manipulation tracing network for detection and localization of image forgeries with anomalous features. 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages pp. 9535–9544, 2019.

[11] K. J. Sani M. Kaya and S. Karakus¸. Copy-move forgery detection in digital forensic images using cnn. 2022 7th International Conference on Computer Science and Engineering (UBMK), pages pp. 239–245, 2022.

[12] S. Rathod M. Baviskar and J. Lohokare. A comparative analysis of image forgery detection techniques. 2022 International Conference on Computing, Communication, Security and Intelligent Systems, pages pp. 1–6, 2022.

[13] J. Y. Park S. I. Lee and I. K. Eom. Cnn-based copy-move forgery detection using rotation-invariant wavelet feature. IEEE Access, pages pp. 106217–106229, 2022.

[14] Y. H. Moon C. W. Park and I. K. Eom. Image tampering localization using demosaicing patterns and singular value based prediction residue. IEEE Access, pages pp. 91921–91933, 2021.

[15] F. Rasouli and M. Taheri. A perfect recovery for fragile watermarking by hamming code on distributed pixels. 18th International ISC Conference on Information Security and Cryptology (ISCISC), pages pp. 18–22, 2021.