Image Forgery Classification : Tampering Detection

Team ID - 17

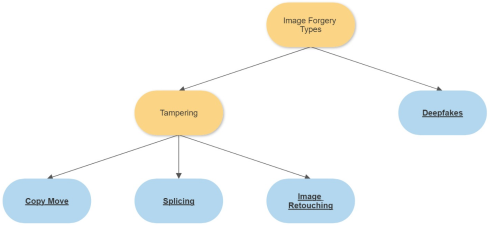
Avish Santhosh D. Sai Prachodhan
Fasal Mohamed N. Vishwaram Reddy

Indian Institute of Technology, Hyderabad

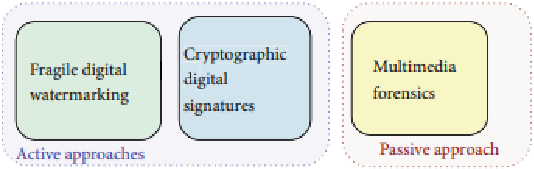
May 3, 2023

Presentation Overview

Introduction



Classification of Image Forgeries



Classification of Image Forgery Detection Techniques Reference from [8]

Problem Statement

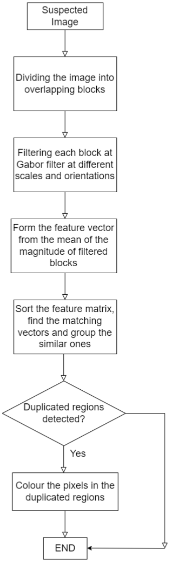
- Our primary objective is to detect copy-move type of image forgery on digital images.
- **Copy-move Forgery (CMFD)**:- It is defined as the act of duplicating one or more regions of an image and pasting them in another location within the same image to create a new image.
- Since both the copied and original part are in the same image, to detect this type of forgery, we simply look for any identical or duplicate regions.

Classification of CMFD techniques

- We can broadly classify image forgery detection techniques into two categories namely, traditional and deep learning methods.
- **Traditional**:- Here we tackle it using block-based methods and key-point based SIFT methods.
- **Deep Learning**:- We train our data set on deep learning models for feature extraction with which we then classify it into using machine learning models.

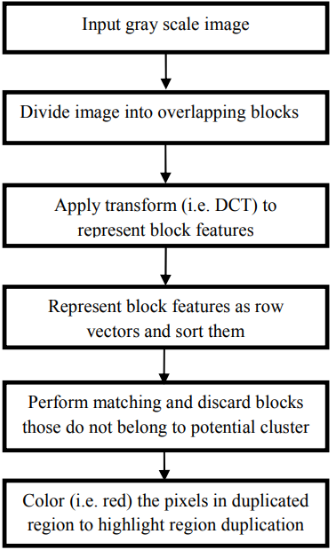
Gabor-filters based detection

- Block-based method.
- Based on idea that for forged image, we can find a pair of feature vectors such that distance of them is $< \delta$, where $\delta >$0 and is very small.
- Uses Gabor filters to extract feature vectors from the non-overlapping blocks.
- Block matching to find whether there are above said pair of feature vectors.
- Beside flowchart shows the steps involved briefly.



Reference from [1]

DCT based approach



Reference from [7]

- Uses the properties of the DCT as an enhancement in the block matching algorithm
- Applies DCT to blocks and after obtaining feature vectors, compares only the first few quantized DCT coefficients while matching, since most of the visual structure of the patch is expected to be present in the first few coefficients
- Expected to be faster than most block matching algorithms without compromising on robustness
- Does not work well with noisy images

SIFT

- Using SIFT method the author first extracted the key points and their corresponding descriptors. [2]
- The basic idea is that the distance between the descriptors of matching key points will be lower than other descriptors. How low is decided by a threshold (set by the author).
- Then hierarchical clustering is performed to club any points with similar features (not due to copy move attack).

VGG16 as feature extractor

Reference from [9]

- **Input**: Pre-processed image using Bilinear Interpolation of size 224×224.
- **Output**: Feature vector of shape (1,7,7,512).
- Output feature vectors are collected and used to train Classifier to detect the forged images.
- We have considered two classifiers namely Random Forests and SVM with linear kernel for each of the Deep learning feature extractors.

ResNet50 as feature extractor

- **Input**: Pre-processed image using Bilinear Interpolation of size 224×224.
- **Output**: Feature vector of shape (1,7,7,2048).
- Output feature vectors are collected and used to train Classifier to detect the forged images.

layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112	7×7, 64, stride 2				
conv2_x	56×56	3×3 max pool, stride 2				
		$\begin{bmatrix} 3\times3, 64 \\ 3\times3, 64 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 64 \\ 3\times3, 64 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$
conv3_x	28×28	$\begin{bmatrix} 3\times3, 128 \\ 3\times3, 128 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 128 \\ 3\times3, 128 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times8$
conv4_x	14×14	$\begin{bmatrix} 3\times3, 256 \\ 3\times3, 256 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 256 \\ 3\times3, 256 \end{bmatrix}\times6$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times6$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times23$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times36$
conv5_x	7×7	$\begin{bmatrix} 3\times3, 512 \\ 3\times3, 512 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 512 \\ 3\times3, 512 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$
	1×1	average pool, 1000-d fc, softmax				
FLOPs		1.8×10^9	3.6×10^9	3.8×10^9	7.6×10^9	11.3×10^9

Overview of ResNet Architecture from [2]

InceptionV3 as feature extractor

- **Input**:- Pre-processed image using bilinaear interpolation of size 299×299
- **Output**:- Feature vector of shape (1, 1000)
- This extractor is more deeper than previous models v1 and v2 but does not compromise on speed.

type	patch size/stride or remarks	input size
conv	3×3/2	299×299×3
conv	3×3/1	149×149×32
conv padded	3×3/1	147×147×32
pool	3×3/2	147×147×64
conv	3×3/1	73×73×64
conv	3×3/2	71×71×80
conv	3×3/1	35×35×192
3×Inception	As in figure 5	35×35×288
5×Inception	As in figure 6	17×17×768
2×Inception	As in figure 7	8×8×1280
pool	8 × 8	8 × 8 × 2048
linear	logits	1 × 1 × 2048
softmax	classifier	1 × 1 × 1000

Outline of Inception-v3 Architecture from [6]

Results

Method	Accuracy
ResNet50 with SVC	98.8%
VGG16 with SVC	98.5%
VGG16 with Random Forests	96.7%
ResNet50 with Random Forests	96.25%
InceptionV3 with Random Forests	71.65%
SIFT key-points based method	62%
Gabor filters based method	60%
DCT based method	60%
InceptionV3 with SVC	52.45%

Accuracies for Traditional and Deep Learning methods on [4].

Note: We have tested SIFT based method on [5].

Key-Takeaways and Conclusions

- While traditional methods give a decent accuracy they perform even better when they are tailored to that particular scenario.
- Overall ResNet50 along with SVC gave us best accuracy.
- SVC performs better when there are higher number of features but less data. Otherwise Random Forests fits better.
- We find that deep learning algorithms outperform the traditional methods(Our Implementations).

Future Scope

- We try to build Deep Learning Models from scratch to detect Copy-Move forgery.
- We can detect the exact forged parts from the forgery image.
- We can even extend our approaches to detect other forgeries like splicing, image retouching also.

References

Raichel Philip Yohannan, Manju Manuel
Detection of Copy-Move forgery based on Gabor filter

Amerini, Irene Ballan, Lamberto Caldelli, Roberto Bimbo, Alberto Serra, Giuseppe
A sift-based forensic method for copy–move attack detection and transformation recovery

Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun
Deep Residual Learning for Image Recognition

Tralic D., Zupancic I., Grgic S., Grgic M.
CoMoFoD - New Database for Copy-Move Forgery Detection

Irene Amerini, Lamberto Ballan, Roberto Caldelli, Alberto Del Bimbo, Giuseppe Serra
A SIFT-based forensic method for copy-move attack detection and transformation recovery

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, Zbigniew Wojna
Rethinking the Inception Architecture for Computer Vision

Sunil Kumar, Jagannath Desai, Shaktidev Mukherjee
A Fast DCT Based Method for Copy Move Forgery Detection

Alessandro Piva
An overview on image forensics

Rohini G
Blog on medium titled as "everything you need to know about vgg16".

Contributions

- Avish Santhosh worked on InceptionV3 based approach.
- Devulapalli Sai Prachodhan worked on ResNet50 and Gabor filters based approaches.
- Fasal Mohamed worked on DCT based approach.
- Vishwaram Reddy worked on VGG16 and SIFT keypoints based methods.