

SobelMNet: A Lightweight CNN based Approach for Median Filtering Detection in compressed images

Anukriti Kaushal¹, Rajeev Kumar¹, Sanjay Kumar¹

¹*Dept. of Computer Science & Engineering, Delhi Technological University, Delhi, India*
anukritikaushal@dtu.ac.in, rajeevkumar@dtu.ac.in, sanjay.kumar@dtu.ac.in

Abstract—Digital image forgery is the process of manipulating an image to deceive or mislead observers with real or manipulated content. Median filtering is widely used to smooth images and obscure traces of tampering, making its detection critical for image forensics. However, identifying median filtering becomes more complex when additional operations, such as compression, resampling, or noise addition, are applied. To address this issue, we propose a lightweight convolutional neural network (CNN) model named SobelMNet, specifically designed for detecting median filtering in compressed images. The proposed model utilises a Sobel filter-based preprocessing step to enhance the residual differences between the original and manipulated images. These residuals, which capture subtle features indicative of median filtering, are analysed by CNN for classification. Further, the proposed model is evaluated on grayscale low-resolution images generated from the Dresden dataset for both binary and multiclass classification tasks. The model achieved a remarkable detection accuracy of 99.43% in median filter detection and outperformed state-of-the-art methods in various scenarios, including combinations of median filtering with Gaussian blur, resampling, and additive white Gaussian noise (AWGN) with an average accuracy of 98.32%. Finally, its lightweight architecture ensures computational efficiency, making it practical for real-world forensic applications.

Index Terms—Median Filter, CNN, Sobel Filter, Operation Chain

I. INTRODUCTION

Digital images play a vital role in modern communication, entertainment, and archiving, spanning areas such as social media, academic visuals, and medical imaging [1]. However, with the rise of advanced editing tools, image manipulation has become widespread and increasingly complex [2]. This has significant implications, such as spreading misinformation and forging legal documents, highlighting the need for reliable forensic methods to verify image authenticity and detect alterations [3].

Image manipulation ranges from simple contrast adjustments to complex filtering techniques. Among these, the median filter is widely used due to its ability to reduce noise while preserving edges [4]. This nonlinear filter replaces each pixel's value with the median of surrounding pixels, making it effective for noise removal [5]. While helpful for preprocessing, the median filter can also obscure evidence of tampering, making its detection critical in image forensics.

Detecting median filtering is essential for tracing an image's processing history and identifying potential alterations [6].

Researchers have explored various methods, from statistical feature extraction to advanced deep learning models [7], to uncover artefacts specific to median filtering. [8].

Deep learning has advanced image forensics, enabling the detection of subtle changes that are often invisible to the human eye. Models such as support vector machine (SVM), CNNs and generative adversarial networks (GANs) have been adapted to identify processing artefacts, including those from median filtering. These models are highly effective due to their ability to extract detailed features and complex analysis in forensic classification models. However, challenges remain, particularly when dealing with low-resolution or heavily compressed images [9]–[11].

II. RELATED WORK

Earlier statistical methods were utilised for image manipulation detection. Zhang et al. [12] introduce a statistical method for detecting median filtering manipulations in digital images using a second-order Local Ternary Pattern (LTP). The LTP captures texture changes caused by median filtering by encoding local derivative direction variations. Kernel Principal Component Analysis is applied to reduce feature dimensionality. Some commonly used statistical methods for detecting median filtering involve histogram residuals, which effectively capture distinctive patterns in images. These patterns, resulting from the median filter operation, allow for the identification of filtered images by analysing variations in the histogram's distribution [13], [14]. Gao and Gao [15] find differences in the pixel values of the image in four directions using pixel-pair histograms and autoregressive moving average coefficients, which outperform compressed JPEG images and low-resolution images. Similarly, Gao et al. [16] compute the first-order and second-order difference images using joint conditional probability density functions. Additionally, principal component analysis is employed for dimensionality reduction, making the model efficient for filter detection in JPEG-compressed and low-resolution images. Kang et al. [17], [18] give an autoregressive model that uses the statistical property of median filter for detection and localisation median filter in an image. The above models outperform work for image forensics but compromise if there is a chain of operations.

The introduction of machine learning and deep learning models has significantly enhanced the ability to detect com-

plex operations applied to images. Jin et al. [19] present a median filtering detection method for RGB images using dark channel residual extraction, multi-scale fusion, and GANs. The dark channel residuals reduce content interference, while multi-scale fusion captures traces from different filter sizes. GANs improve robustness and replace SVM classification. The method outperforms existing techniques, especially with small blocks and JPEG compression, and aims to extend to grayscale images. A CNN-based method introduces MFNet [20] for detecting median filtering in small or compressed images. It uses nearest neighbour interpolation to up-sample images, enhancing the difference between original and filtered versions. MFNet incorporates two mlpconv layers to improve nonlinear classification, followed by three convolutional layers for feature extraction. The method shows significant improvements, especially for images as small as 16×16 , and offers potential for broader application in digital image forensics. Ahmed and Islam [21] discuss an approach that leverages the streaking effect caused by median filtering, using SVMs with 10-fold cross-validation to distinguish between median-filtered and unfiltered images. This method also performs well on compressed and low-resolution images. Another study [22] combines texture features extracted via GLCM with streak area analysis, employing an 18-dimensional feature set processed through neural networks to identify median filtering under various manipulations like JPEG compression and Gaussian blur. An advanced CNN-based framework introduces the MFFNet [23], integrating high-pass filters and Squeeze-and-Excitation modules, excelling in detecting manipulations in compressed, low-resolution images. Despite advancements, limitations include challenges in handling operator chains, which are multiple operations on images, including median filtering, and scaling methods to large datasets. These methods demonstrate a shift from traditional SVMs to robust deep learning architectures for feature learning and classification. A deep learning-based approach is introduced for detecting median filtering in the DCT domain. The method employs an Adaptive Filtering Layer to focus on relevant frequency ranges and a multi-scale convolutional block with a maxout function for effective feature extraction. The approach shows superior performance over existing methods, particularly for low-resolution and compressed images [24].

A limitation in existing methods is the decrease in accuracy when detecting median filtering, especially when it is followed by operations such as Gaussian blur, downsampling, or noise addition. Additionally, some techniques are highly dependent on high-resolution images, making them less effective for low-resolution or resized images.

III. PROPOSED WORK: CNN-BASED MODEL

Proposed SobelMNet, a two-dimensional residual-based 2D CNN, specifically designed for detecting traces of median filter in digital images. The key features of the proposed approach are as follows:

- 1) The Sobel filter-based residual is introduced as the pre-processing layer in the model. This filter kernel modifies

the image pixels and generates residuals that amplify the classifier's ability to distinguish between different classes.

- 2) After the Sobel residual processing, we extract features that emphasise the statistical and structural differences in the image. These features are designed to make it easier for the neural network to identify whether an image has been median-filtered.
- 3) A lightweight neural network architecture is used for analyzing the extracted features and classifying the images. The network is efficient and can handle both binary tasks and multiclass classification.
- 4) The model is evaluated on images with low resolution and compression artifacts to ensure it works effectively even under challenging conditions.
- 5) To further validate the model's robustness, we tested it on images processed with median filtering followed by other operations like Gaussian blur, downsampling, additive white Gaussian noise (AWGN), and compression. This shows that the model can handle more complex real-world scenarios.

The architecture consists of three primary blocks designed to handle image feature extraction and classification efficiently: Preprocessing block, feature extraction block, and classification block as discussed in subsections.

A. Preprocessing Layer

This block applies a Sobel filter to enhance the edges in the image. The Sobel filter helps detect subtle changes introduced by median filtering, which tends to blur edges. By emphasizing these changes, it makes the subsequent feature extraction phase more effective. The input is a grayscale image, typically of size 64×64 pixels.

The Sobel filter [25] is a well-known edge-detection operator commonly used in image processing. It is especially effective for highlighting changes in intensity or gradients within an image. The filter works by using two convolution kernels: one to measure the gradient in the horizontal direction (Sobel-X) and another for the vertical direction (Sobel-Y). The gradient magnitude is then computed to identify edges, which represent significant transitions in pixel intensity. Let the input image be denoted as I , with dimensions $H \times W$.

Step 1: Sobel Filter Kernels The Sobel filters for the x - and y -directions are defined as:

$$K_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}, \quad K_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}.$$

Step 2: Convolution Operation Convolve the input image I with K_x and K_y to compute the gradients in the horizontal and vertical directions, respectively:

$$G_x = I * K_x, \quad G_y = I * K_y,$$

where $*$ denotes the convolution operation, and G_x, G_y are of size $H \times W$.

Step 3: Sobel Residual Calculation Combine the gradients using the Euclidean norm to compute the residual:

$$R_{\text{Sobel}}(i, j) = \sqrt{G_x(i, j)^2 + G_y(i, j)^2},$$

where i, j denote the pixel coordinates. The output R_{Sobel} is an edge-enhanced image of the same dimensions as the input ($H \times W$).

The Sobel filter can reveal subtle changes caused by enhanced edges and highlight residuals that are left behind by median filtering, such as the streaking effect left by median filtering. When applied to noisy images, it makes these residuals easier to observe. This enhanced information is used in machine learning models or neural networks to help them more accurately identify manipulated images, especially when distinguishing between original and filtered versions.

B. Feature Extraction Layer

The architecture model, shown in Figure 1, consists of four convolutional layers designed to progressively extract hierarchical features. Each block includes a convolutional layer followed by batch normalization and a ReLU activation function. A kernel of size 3×3 is applied in each convolutional layer. In the figure, FM represents feature maps, with the dimensions provided at each stage. Max-pooling layers are incorporated between convolutional layers to reduce spatial dimensions while preserving essential features, and dropout layers are included to minimise the risk of overfitting.

C. Classification Block

The features obtained from the previous block are converted into a one-dimensional vector and then passed through fully connected layers for advanced decision-making. The model is designed to classify multiple types of operations applied to an image, including a single median filter or a median filter followed by operations such as Gaussian blur, resampling, and AWGN for compressed and low resolution image datasets. The model classifies images by extracting hierarchical features through convolutional layers and fully connected layers. The final layer generates output class probabilities, enabling binary or multi-class classification depending on the number of classes in a training dataset. During training, the Adam optimiser adjusts the model's parameters, while CrossEntropyLoss is used to compute the loss.

IV. EXPERIMENT AND RESULT

A. Dataset Description

To evaluate the performance of the proposed model in detecting image manipulations, we created a custom image dataset based on the high-quality Dresden dataset [24]. A total of 10,000 images were cropped to 64×64 pixels and converted to grayscale to standardize the input data. Pre-compression and post-compression were applied with a quality factor (QF) of 80, before and after performing single or multiple operations. This process simulates the degradation typically caused by compression algorithms. After the initial compression, median filtering with 3×3 and 5×5 kernel sizes was applied to simulate

manipulation. The dataset for median filtering includes an equal distribution of 50% images processed with a 3×3 kernel and 50% with a 5×5 kernel. As shown in Table I, the images were compressed again to replicate multiple stages of image processing. The final dataset includes 30,000 training images, 6,000 validation images, and 10,000 test images for each class. This diverse set of images allows for robust model training and evaluation. The approach ensures that the model can detect manipulations effectively across various image quality levels and processing stages.

For binary classification, the following two classes were defined: The sequence of operations for an image (I) undergoing compression with QF 80, followed by median filtering (MF) with kernel sizes of 3×3 or 5×5 , and then compression again with QF 80 can be represented as:

$$I \xrightarrow{\text{Compression}} \xrightarrow{\text{Median Filter}} \xrightarrow{\text{Compression}} \text{MF}(I)$$

In multi-class classification, the sequence of operations the image I undergoes is pre-compression with QF 80, then an initial median filtering, followed by another operation O as given in table I, and finally, post-compression with QF 80. This can be expressed as:

$$I \xrightarrow{\text{Compression}} \xrightarrow{\text{Median Filter}} \xrightarrow{O} \xrightarrow{\text{Compression}} O(\text{MF}(I))$$

TABLE I: Image manipulation operations and parameters applied to the dataset

Operation	Parameters
Median Filtering (MF)	Window sizes: size=3, size=5
Gaussian Blurring (GB)	Kernel sizes: (3, 3), (5, 5)
Resampling (RS)	Scaling factors: 1.2, 1.5
Additive Gaussian Noise (AWGN)	Noise variance: var=0.5 ² , var=2 ²
JPEG Compression	Quality factors: QF=80%

B. Performance Evaluation of Median Filter Detection Model

In this subsection, we discuss the results obtained from the model's training and evaluation. The training and validation performances are presented, followed by a detailed discussion of the confusion matrix and the model's ability to classify manipulated and original images.

1) *Single Manipulation Detection*: For binary classification, the model is trained, and the best model is saved, with the overall training accuracy of 96.83% and validation accuracy of 98.89%. The validation accuracy of the class median filter is 99.43% and the original accuracy is 97.75%. These high accuracies suggest that the model can effectively distinguish between manipulated and original images in the dataset.

We compare True positive rate (TPR) of SobelMNet with other existing methods designed for median filtering detection under similar parameters. Table II presents the result of the proposed method in comparison with MedFCN [22] and Streak analysis [21]. The accuracy of the median filter detection approach has been demonstrated to be competitive with state-of-the-art techniques, but further improvements in distinguishing original images are still required.

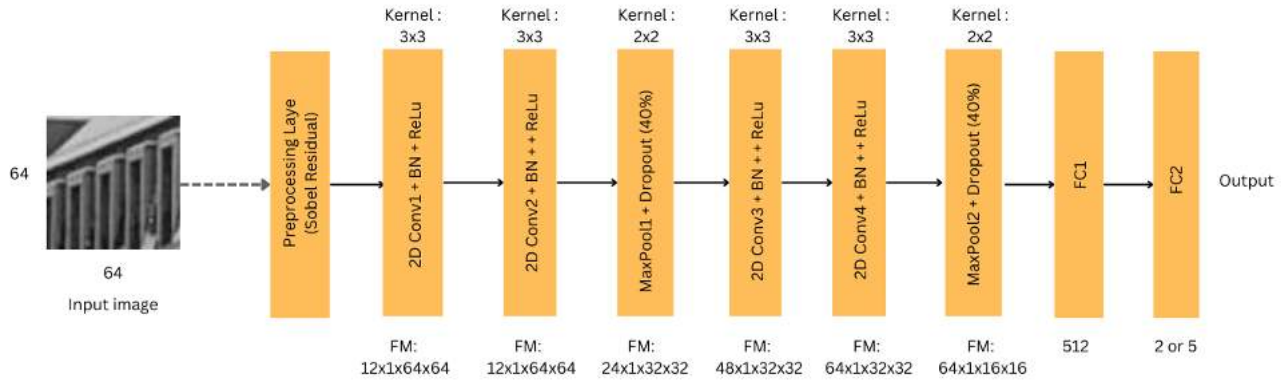


Figure 1: SobelMNet architecture

TABLE II: Comparison of SobelMNet with Different Methods

Method	True Positive Rate (TPR)
Proposed Method	99.43%
MedFCN [22]	98.50%
Streak [21]	97.80%

2) *Multiple Manipulation Detection*: This section presents the evaluation of the proposed model's performance on the median filter chain dataset. The results include the validation accuracy, class-specific performance, and the confusion matrix. Additionally, the trends in training and validation accuracy and loss are shown in Figure 3.

The model achieved a best validation accuracy of 98.32%, showcasing its effectiveness in identifying manipulated and original images. Table III presents the confusion matrix in percentage, providing a comparative representation of the model's classification performance across five classes. Class-specific accuracies demonstrate the robustness of the model across different image manipulations.

TABLE III: Confusion Matrix for Median Filter Chain Classification with True and Predicted Class labels.

True Class	Predicted Class				
	MF	MF+AWGN	MF+GB	MF+RS	ORG
MF	88.33	0.00	5.18	0.06	0.55
MF+AWGN	0.00	100.00	0.00	0.00	0.00
MF+GB	0.25	0.00	93.97	0.06	0.04
MF+RS	0.03	0.00	0.63	93.72	0.00
ORG	1.24	0.00	0.42	0.01	99.41

Figure 3 illustrated the trends in training and validation accuracy and loss. The training accuracy and validation accuracy consistently improved over epochs, demonstrating effective learning and generalization. The training loss decreased steadily, while the validation loss remained low, confirming that the model was not overfitting to the training data. The proposed model's performance on the median filter chain

dataset demonstrates its high accuracy and robustness in distinguishing various manipulated and original images. The results underscore the model's potential for real-world applications in image forensics.

C. Result and Analysis

The proposed model SobelMNet achieves 99.43% accuracy in detecting median filtering on low-resolution, compressed images. It outperforms state-of-the-art methods like MedFCN and streak analysis in True Positive Rate (TPR) given in table II. Additionally, it accurately identifies complex operations such as median filtering combined with Gaussian blur, resampling, or AWGN, achieving an average accuracy of 98.32% and the highest accuracy of 100% when the image contains a median filter and AWGN, as shown in Table III.

V. CONCLUSION AND FUTURE WORK

The paper presented SobelMNet, a lightweight model that utilized a Sobel filter residual to detect differences between an original image and an image processed with a median filter. Designed with fewer parameters, the model was highly efficient and effectively tested whether an image had been manipulated with a median filter or other median filter chain operations. The results demonstrated that SobelMNet outperformed existing models, achieving an impressive accuracy of 99.43% on compressed and low-resolution images. Moreover, the model offered computational efficiency while maintaining high performance, making it suitable for real-world scenarios involving low-resolution images, such as those found in social media, legal evidence, and archival data. SobelMNet also demonstrated robust performance across diverse images using the Dresden dataset. In future work, this model could be extended and applied to detect other types of image manipulations. SobelMNET's future research can focus enhancing SobelMNet's generalization to detect image manipulations, deepfake generation, copy move forgery, and inpainting can

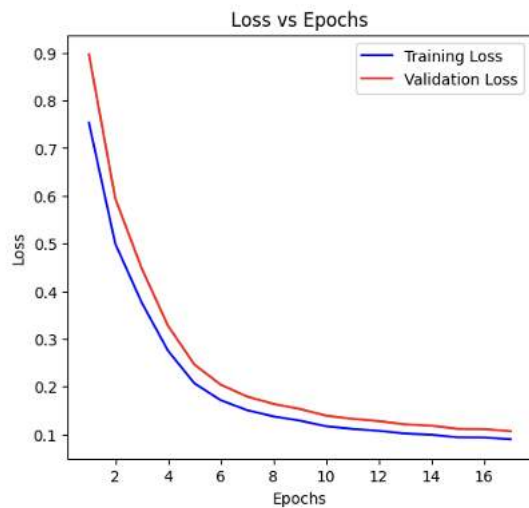


Figure 2: Loss plot for Training and Validation

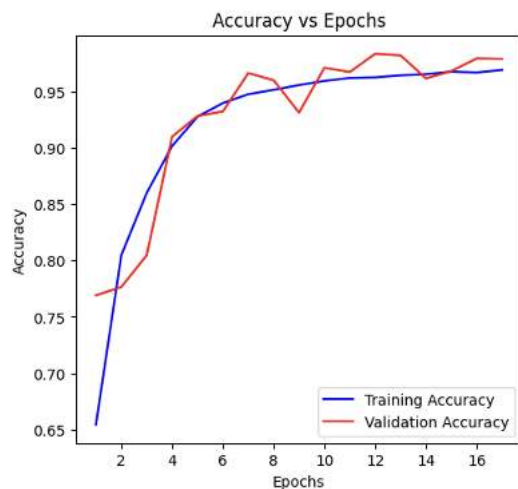


Figure 3: Accuracy plot for Training and Validation

Figure 4: Training and Validation Accuracy and Loss Curves for Median Filter Chain Detection

improve forensic scenarios. Evaluating SobelMNet on different datasets, including high definition and color images, can confirm its stability. Improving the SobelMNet's defense towards adversarial attacks, which exploit subtle distortions to mislead detection, is crucial.

REFERENCES

- [1] Hany Farid. Exposing digital forgeries from jpeg ghosts. *IEEE transactions on information forensics and security*, 4(1):154–160, 2009.
- [2] Alessandro Piva. An overview on image forensics. *International Scholarly Research Notices*, 2013(1):496701, 2013.
- [3] Jessica Fridrich. Digital image forensics. *IEEE Signal Processing Magazine*, 26(2):26–37, 2009.
- [4] Thomas Huang, GJTG Yang, and Greory Tang. A fast two-dimensional median filtering algorithm. *IEEE transactions on acoustics, speech, and signal processing*, 27(1):13–18, 1979.
- [5] Jaakko Astola and Pauli Kuosmanen. *Fundamentals of nonlinear digital filtering*. CRC press, 2020.
- [6] Thomas Gloe and Rainer Böhme. The dresden image database*for benchmarking digital image forensics. In *Proceedings of the 2010 ACM symposium on applied computing*, pages 1584–1590, 2010.
- [7] Mauro Barni and Andrea Costanzo. A fuzzy approach to deal with uncertainty in image forensics. *Signal Processing: Image Communication*, 27(9):998–1010, 2012.
- [8] Jiansheng Chen, Xiangui Kang, Ye Liu, and Z Jane Wang. Median filtering forensics based on convolutional neural networks. *IEEE Signal Processing Letters*, 22(11):1849–1853, 2015.
- [9] Yuan Rao and Jiangqun Ni. A deep learning approach to detection of splicing and copy-move forgeries in images. In *2016 IEEE international workshop on information forensics and security (WIFS)*, pages 1–6. IEEE, 2016.
- [10] Belhassen Bayar and Matthew C Stamm. A deep learning approach to universal image manipulation detection using a new convolutional layer. In *Proceedings of the 4th ACM workshop on information hiding and multimedia security*, pages 5–10, 2016.
- [11] Belhassen Bayar and Matthew C Stamm. On the robustness of constrained convolutional neural networks to jpeg post-compression for image resampling detection. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2152–2156. IEEE, 2017.
- [12] Yujin Zhang, Shenghong Li, Shilin Wang, and Yun Qing Shi. Revealing the traces of median filtering using high-order local ternary patterns. *IEEE Signal Processing Letters*, 21(3):275–279, 2014.
- [13] Xinlu Gui, Xiaolong Li, Wenfa Qi, and Bin Yang. Blind median filtering detection based on histogram features. In *Signal and Information Processing Association Annual Summit and Conference (APSIPA), 2014 Asia-Pacific*, pages 1–4. IEEE, 2014.
- [14] Abhinav Gupta and Divya Singhal. Analytical global median filtering forensics based on moment histograms. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 14(2):1–23, 2018.
- [15] Hang Gao and Tiegang Gao. Detection of median filtering based on arma model and pixel-pair histogram feature of difference image. *Multimedia Tools and Applications*, 79(17):12551–12567, 2020.
- [16] Hang Gao, Mengting Hu, Tiegang Gao, and Renhong Cheng. Robust detection of median filtering based on combined features of difference image. *Signal Processing: Image Communication*, 72:126–133, 2019.
- [17] Xiangui Kang, Matthew C Stamm, Anjie Peng, and KJ Ray Liu. Robust median filtering forensics using an autoregressive model. *IEEE Transactions on Information Forensics and Security*, 8(9):1456–1468, 2013.
- [18] Xiangui Kang, Matthew C Stamm, Anjie Peng, and KJ Ray Liu. Robust median filtering forensics based on the autoregressive model of median filtered residual. In *Proceedings of The 2012 Asia Pacific Signal and Information Processing Association Annual Summit and Conference*, pages 1–9. IEEE, 2012.
- [19] Xiao Jin, Peiguang Jing, and Yuting Su. Amfnet: an adversarial network for median filtering detection. *IEEE Access*, 6:50459–50467, 2018.
- [20] Hongshen Tang, Rongrong Ni, Yao Zhao, and Xiaolong Li. Median filtering detection of small-size image based on cnn. *Journal of Visual Communication and Image Representation*, 51:162–168, 2018.
- [21] Sajjad Ahmed and Saiful Islam. Median filter detection through streak area analysis. *Digital Investigation*, 26:100–106, 2018.
- [22] Sajjad Ahmed and Saiful Islam. Robust median filtering forensics using texture feature and deep fully connected network. *International Journal of Information Technology*, 16(1):601–610, 2024.
- [23] Kaijun Wu, Wanli Dong, Yunfei Cao, Xue Wang, and Qi Zhao. An improved method of median filtering forensics for enhanced image security detection. In *2021 International Conference on Networking and Network Applications (NaNA)*, pages 308–312. IEEE, 2021.
- [24] Jun Zhang, Yixin Liao, Xinshan Zhu, Hongquan Wang, and Jie Ding. A deep learning approach in the discrete cosine transform domain to median filtering forensics. *IEEE Signal Processing Letters*, 27:276–280, 2020.
- [25] Irwin Sobel, Gary Feldman, et al. A 3x3 isotropic gradient operator for image processing. *a talk at the Stanford Artificial Project in*, 1968:271–272, 1968.