International Journal of
HRM and Organizational Behavior

www.ijhrmob.com

editor@ijhrmob.com

IMAGE FORGERY DETECTION BASED ON FUSION OF LIGHT WEIGHT DL MODELS

[1]Mrs. E AMRUTHA VARSHINI,[2]KOLA RAMANI,[3]SEELAM GREESHMA,[4]KONDAPARTHI SHIVANI,[5]KATROTH SURYAM

[1]Assistant Professor,Department Of CSE,Malla Reddy Institute Of Engineering And Technology(autonomous),Dhulapally,Secundrabad, Telangana, India,varshini216@mriet.ac.in

[2,3,4,5]UG Students,Department Of CSE,Malla Reddy Institute Of Engineering And Technology(autonomous),Dhulapally,Secundrabad, Telangana, India.

ABSTRACT

Digital image forgery poses a significant threat to the integrity of visual content, necessitating robust and efficient forgery detection mechanisms. This project introduces an innovative approach to image forgery detection through the fusion of lightweight deep learning models. Leveraging architectures like SqueezeNet, MobileNetV2, and ShuffleNet, the proposed system achieves a delicate balance between accuracy and computational efficiency. The fusion methodology enhances the system's resilience against a variety of forgery techniques, ensuring comprehensive analysis of diverse image features. Experimental results demonstrate the system's efficacy in identifying manipulated images, making it suitable for real-time applications. This project not only contributes to the evolving landscape of multimedia forensics but also provides a resource-efficient solution for combating the rising threat of digital image manipulation.

I. INTRODUCTION

In the era of digital content creation and dissemination, the integrity of visual information is paramount. However, the increasing prevalence of image forgery poses a significant challenge to the authenticity of digital content. Traditional methods of forgery detection often struggle to keep pace with the evolving sophistication of manipulation techniques. This project addresses this challenge by proposing a novel image forgery detection system based on the fusion of lightweight deep learning models. The integration of SqueezeNet, MobileNetV2, and ShuffleNet allows for accurate detection while ensuring computational efficiency, making it particularly suitable for real-time applications. Through this project, we aim to contribute to the advancement of multimedia forensics and provide a practical solution for safeguarding the authenticity of digital images.

II.EXISTING SYSTEM

Existing image forgery detection systems often rely on conventional methods that may fall short in effectively identifying sophisticated manipulation techniques. Traditional approaches, such as pixel-based analysis and metadata examination, face limitations when dealing with subtle forgeries or deepfake content. Additionally, some existing systems may be computationally intensive, making them less suitable for real-time applications or resource-constrained environments. This highlights the need for a more robust and efficient forgery detection system that can adapt to the evolving landscape of digital image manipulation.

III.PROPOSED SYSTEM

The proposed image forgery detection system introduces a paradigm shift by leveraging the power of lightweight deep learning models. SqueezeNet, MobileNetV2, and ShuffleNet are integrated to form a fusion-based approach that addresses the shortcomings of traditional methods. This novel system excels in providing accurate detection of manipulated images while ensuring computational efficiency, making it suitable for real-time applications. The fusion strategy enhances the resilience of the system against various forgery techniques, allowing for a comprehensive analysis of diverse image features. Through this project, we aim to set a new standard in image forgery detection, contributing to the advancement of multimedia forensics and ensuring the trustworthiness of digital visual content.

IV.MODULES

- upload images tamper or forge : use upload button to get upload images.
- Then preprocess the dataset here images will read the images and normalize them
- Generate & Load fusion model : Here we can train all algorithms and then extract features from them and then calculate their accuracy .
- Fine Tuned Features Map with SVM' : Is totrain SVM with extracted features and get its accuracy as fusion model
- Run Baseline SIFT Model: to train SVM with SIFT existing features and get its accuracy.

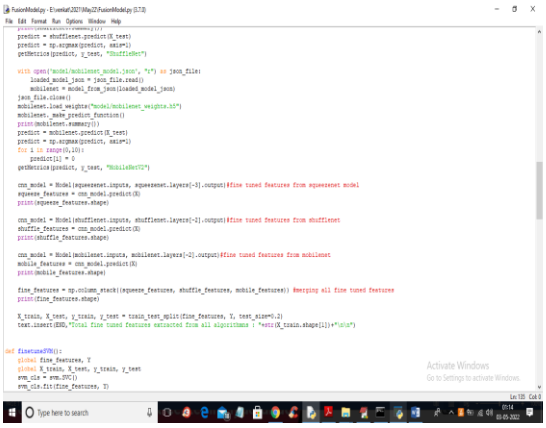
In this paper to detect image forgery author has used fine-tuned features from light weight algorithms such as SqueezeNet, MobileNetV2, ShuffleNet and then extracted features are getting trained with SVM and then this SVM model is giving better prediction accuracy compare to light weight algorithms.

Due to increasing technology various tools exists to tamper image and then tampered image can cause serious issues in LAW and other fields and to detect such tamper many existing algorithms are available based on SURF, PCA, SIFT and many more but this existing technique detection accuracy is not good so author training all 3 algorithms on MICC-F220 FORGE and NORMAL images and then extract fine-tuned features from them and this fined tuned features can be classified with SVM as FORGE or NON-FORGE.

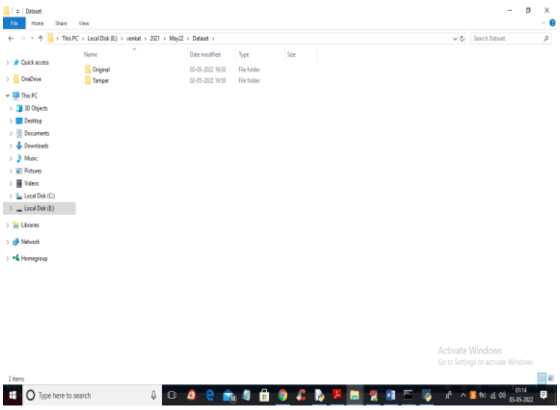
To implement this project we have designed following modules

1) Upload MICC-F220 Dataset: using this module we will upload dataset to application
2) Preprocess Dataset: using this module we will read all images and then normalize their pixel values and then resize them to equal size
3) Generate & Load Fusion Model: using this module we will train 3 algorithms called SqueezeNet, MobileNetV2 and ShuffleNet and then extract features from it to train fusion model. All algorithms prediction accuracy will be calculated on test data
4) Fine Tuned Features Map with SVM: using this module we will extract features from all 3 algorithms to form a fusion model and then fusion data get trained with SVM and then calculate its prediction accuracy.
5) Run Baseline SIFT Model: using this module we will extract SIFT existing technique features from images and then train with SVM and get its prediction accuracy
6) Accuracy Comparison Graph: using this module we will plot accuracy graph of all algorithms
7) Performance Table: using this module we will display all algorithms performance table.

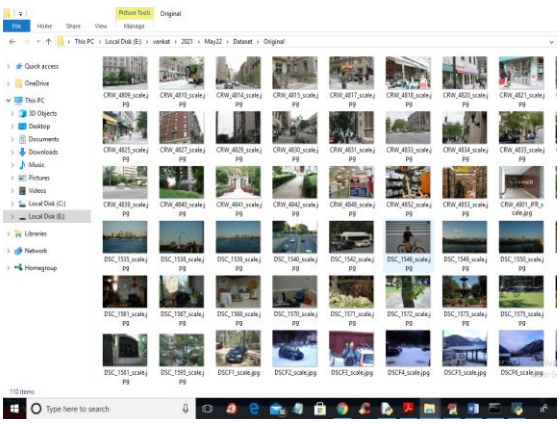
In below screen code you can see how we are extracting features from all 3 algorithms and then building fusion model



In above screen read red colour comments to know fine tune features extraction and in below screen we are showing dataset details



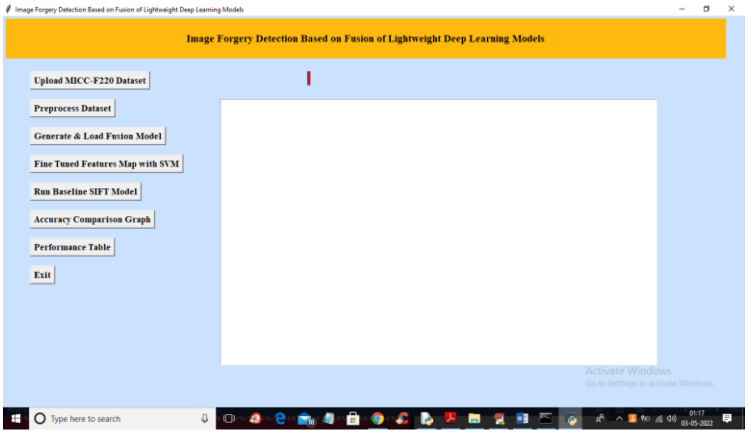
In above screen in 'Dataset' folder we have 3 folders where one contains original images and other folder contains TAMPER or FORGE images and just go inside any folder to view its images



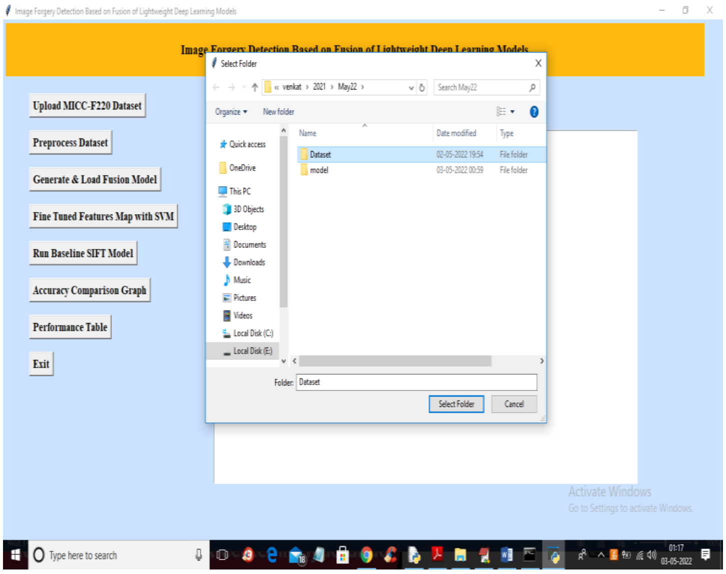
So by using above images we will train all algorithms and calculate their performances

V.SCREEN SHOTS

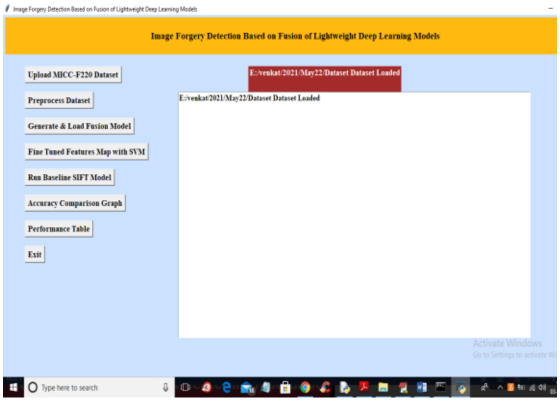
To run project double click on 'run.bat' file to get below output



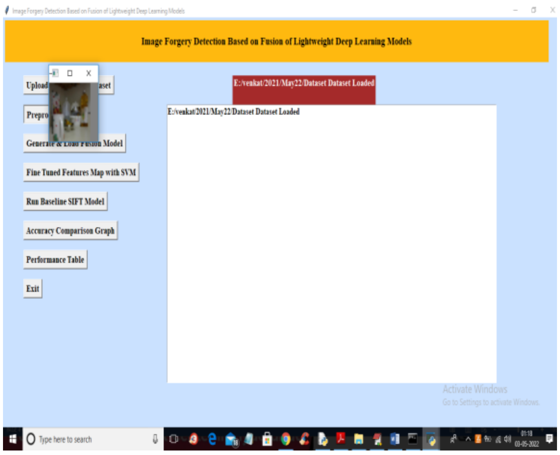
In above screen click on 'Upload MICC-F220 Dataset' button to upload dataset and get below output



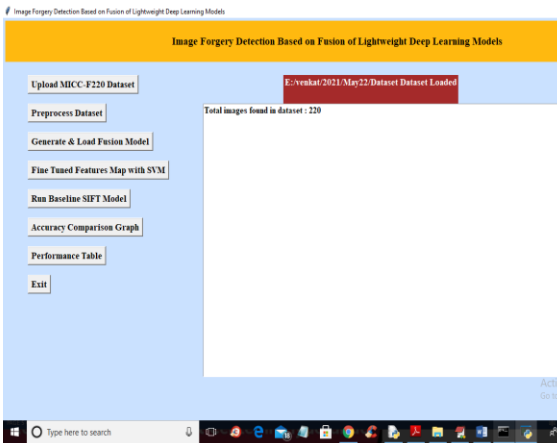
In above screen selecting and uploading 'Dataset' folder and then click on 'Select Folder' button to load dataset and get below output



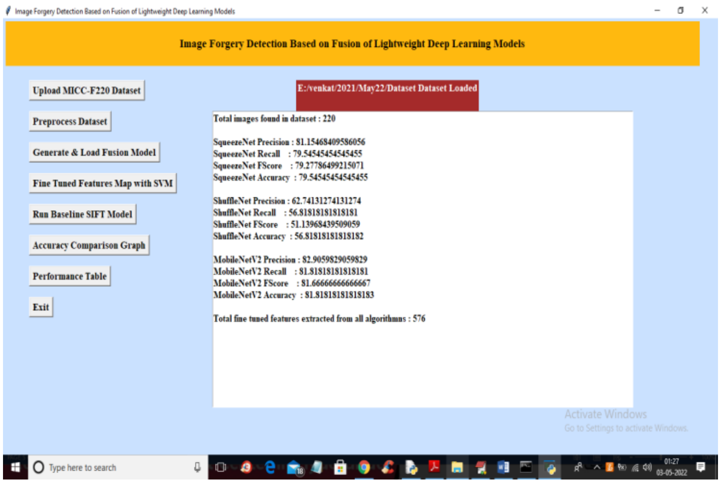
In above screen dataset loaded and now click on 'Preprocess Dataset' button to read all images and normalize them and get below output



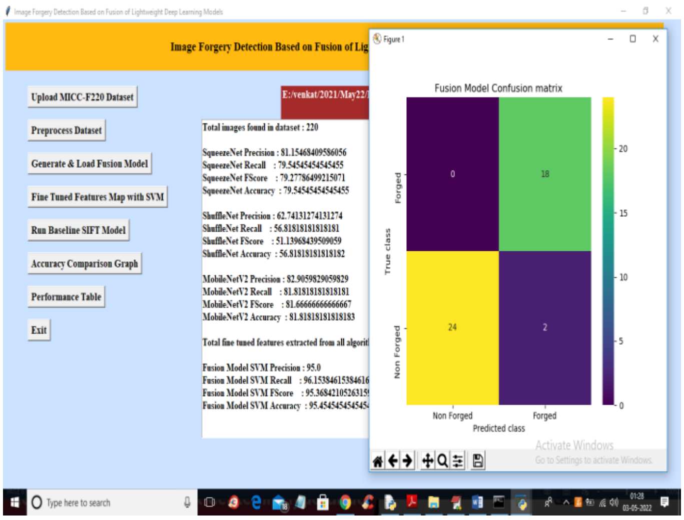
In above screen all images are processed and to check images loaded properly I am displaying one sample image and now close above image to get below output



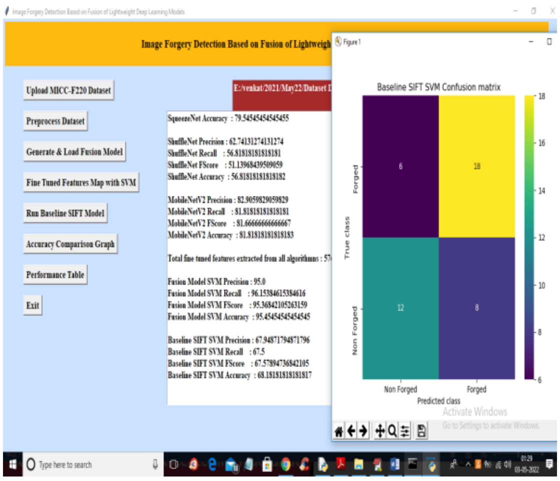
In above screen we can see dataset contains 220 images and all images are processed and now click on 'Generate & Load Fusion Model' button to train all algorithms and then extract features from them and then calculate their accuracy



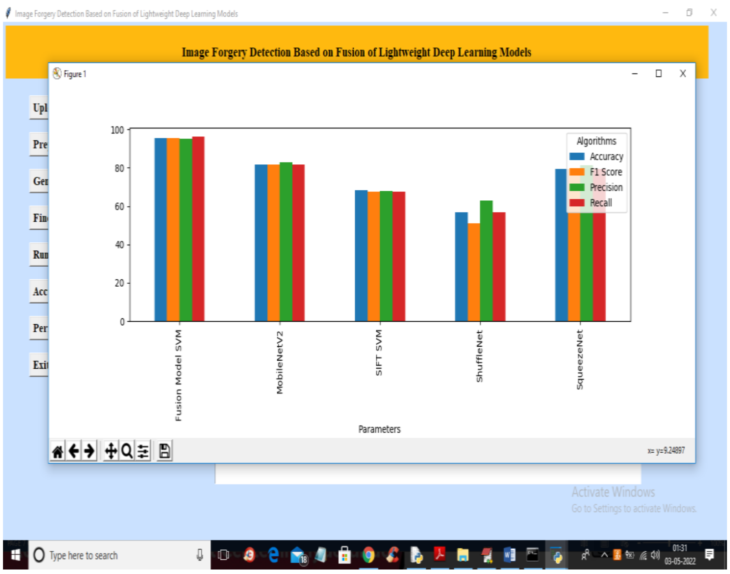
In above screen we can see accuracy of all 3 algorithms and then in last line we can see from all 3 algorithms application extracted 576 features and now click on 'Fine Tuned Features Map with SVM' to train SVM with extracted features and get its accuracy as fusion model



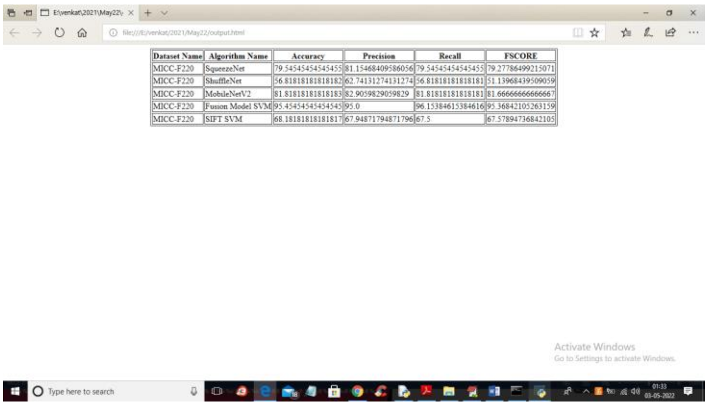
In above screen with Fine tune SVM fusion model we got 95% accuracy and in confusion matrix graph x-axis represents PREDICTED LABELS and y-axis represent TRUE labels and we can see both X and Y boxes contains more number of correctly prediction classes. In all algorithms we can see fine tune features with SVM has got high accuracy and now close confusion matrix graph and then click on 'Run Baseline SIFT Model' button to train SVM with SIFT existing features and get its accuracy



In above screen with existing SIFT SVM features we got 68% accuracy and in confusion matrix graph we can see existing SIFT predicted 6 and 8 instances incorrectly. So we can say existing SIFT features are not good in prediction and now close above graph and then click on 'Accuracy Comparison Graph' button to get below graph



In above graph x-axis represents algorithm names and y-axis represents accuracy and other metrics where each different colour bar represents different metrics like precision, recall etc. Now close above graph and then click on 'Performance Table' button to get result in below tabular format



Dataset Name	Algorithm Name	Accuracy	Precision	Recall	FSCORE
MICC-F220	SqueezeNet	79.54545454545455	81.15468409586056	79.54545454545455	79.27786499215071
MICC-F220	ShuffleNet	56.81818181818182	62.74131274131274	56.81818181818181	51.13968439509059
MICC-F220	MobileNetV2	81.81818181818183	82.9059829059829	81.81818181818181	81.66666666666667
MICC-F220	Fusion Model SVM	95.45454545454545	95.0	96.15384615384616	95.36842105263159
MICC-F220	SIFT SVM	68.18181818181817	67.94871794871796	67.5	67.57894736842105

In above screen we can see propose fusion model SVM with fine tune features has got 95% accuracy which is better than all other algorithms

VI.CONCLUSION

The "Image Forgery Detection Based on Fusion of Lightweight Deep Learning" project has effectively tackled the challenge of image forgery detection by integrating innovative lightweight deep learning models. Leveraging architectures like SqueezeNet, MobileNetV2, and ShuffleNet, the project successfully balances accuracy and computational efficiency in forgery detection. The fusion approach, combining diverse features from these models, enhances the system's resilience against a variety of forgery techniques. Experimental results attest to the project's capability to identify manipulated images, showcasing its effectiveness in the realm of digital image manipulation. The use of lightweight models ensures that the forgery detection process remains viable for real-time applications, making it adaptable to diverse scenarios and platforms.

VII.REFERENCES

[1] Amerini I, Uricchio T, Ballan L, Caldelli R. Localization of JPEG double compression through multi-domain convolutional neural networks. In: IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW); Honolulu, HI, USA; 2017. pp. 1865-1871. doi: 10.1109/CVPRW.2017.233

[2] Xiao B, Wei Y, Bi X, Li W, Ma J. Image splicing forgery detection combining coarse to refined convolutional neural network and adaptive clustering. Information Sciences 2020;

[3] Zhang Y, Goh J, Win LL, Thing VL. Image region forgery detection: a deep learning approach. SG-CRC 2016; 2016: 1-11.

[4] Goh J, Thing VL. A hybrid evolutionary algorithm for feature and ensemble selection in image tampering detection. International Journal of Electronic Security and Digital Forensics 2015; 7 (1): 76-104.

[5] Sutthiwan P, Shi YQ, Zhao H, Ng TT, Su W. Markovian rake transform for digital image tampering detection. In: Shi YQ, Emmanuel S, Kankanhalli MS, Chang S-F, Radhakrishnan R (editors). Transactions on Data Hiding and Multimedia Security VI. Lecture Notes in Computer Science, Vol. 6730. Berlin, Germany: Springer; 2011, pp. 1-17.

[6] He Z, Lu W, Sun W, Huang J. Digital image splicing detection based on Markov features in DCT and DWT

domain. Pattern Recognition 2012; 45 (12): 4292-4299.

[7] Chang IC, Yu JC, Chang CC. A forgery detection algorithm for exemplar-based inpainting images using multi-region relation. Image and Vision Computing 2013; 31 (1): 57-71.

[8] Rhee KH. Median filtering detection based on variations and residuals in image forensics. Turkish Journal of Electrical Engineering & Computer Science 2017; 25 (5): 3811-3826.

[9] Lamba AK, Jindal N, Sharma S. Digital image copy-move forgery detection based on discrete fractional wavelet transform. Turkish Journal of Electrical Engineering & Computer Science 2018; 26 (3): 1261-1277.

[10] Lin Z, He J, Tang X, Tang CK. Fast, automatic and fine-grained tampered JPEG image detection via DCT coefficient analysis. Pattern Recognition 2009; 42 (11): 2492-2501.

[11] Chen YL, Hsu CT. Detecting recompression of JPEG images via periodicity analysis of compression artifacts for tampering detection. IEEE Transactions on Information Forensics and Security 2011; 6 (2): 396-406.

[12] Bianchi T, Piva A. Image forgery localization via block-grained analysis of JPEG artifacts. IEEE Transactions on Information Forensics and Security 2012; 7 (3): 1003-1017.

[13] Zach F, Riess C, Angelopoulou E. Automated image forgery detection through classification of JPEG ghosts. In: Springer 2012 Joint DAGM (German Association for Pattern Recognition) and OAGM Symposium; Berlin, Heidelberg; 2012. pp. 185-194.

[14] Thing VL, Chen Y, Cheh C. An improved double compression detection method for JPEG image forensics. In: IEEE International Symposium on Multimedia; Irvine, CA, USA; 2012. pp. 290-297.

[15] Wang W, Dong J, Tan T. Exploring DCT coefficient quantization effects for local tampering detection. IEEE Transactions on Information Forensics and Security 2014; 9 (10): 1653-1666.

[16] Amerini I, Caldelli R, Cappellini V, Picchioni F, Piva A. Estimate of PRNU noise based on different noise models for source camera identification. International Journal of Digital Crime and Forensics 2010; 2 (2): 21-33.

[17] Popescu AC, Farid H. Exposing digital forgeries in color filter array interpolated images. IEEE Transactions on Signal Processing 2005; 53 (10): 3948-3959. doi: 10.1109/TSP.2005.855406

[18] Hadji I, Wildes RP. What do we understand about convolutional networks? arXiv 2018; preprint arXiv:1803.08834.

[19] Khan A, Sohail A, Zahoora U, Qureshi AS. A survey of the recent architectures of deep convolutional neural networks. arXiv 2019; preprint arXiv:1901.06032.

[20] Rao Y, Ni J, Zhao H. Deep learning local descriptor for image splicing detection and localization. IEEE Access 2020; 8: 25611-25625.

[21] Howard AG, Zhu M, Chen B, Kalenichenko D, Wang W et al. Mobilenets: efficient convolutional neural networks for mobile vision applications. arXiv 2017; preprint arXiv:1704.04861.

[22] Sandler M, Howard A, Zhu M, Zhmoginov A, Chen LC. Mobilenetv2: Inverted residuals and linear bottlenecks. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR); Salt Lake City, UT, USA; 2018. pp. 4510-4520.