Project Report on

SOURCE CAMERA IDENTIFICATION BY FEATURE EXTRACTION USING LOCAL BINARY PATTERN

Submitted By-

Shashank Singh – 35000118040

Rohan Roy – 35000118045

Abhishek Mondal – 35000118074

Super Kumar Murmu – 35000118025

Sumit Mahato – 35000119047

Bindeeya Darnal- 35000119050

Under the Guidance of

Prof. Pabitra Roy

Assistant Professor

Department of Computer Science & Engineering

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

RAMKRISHNA MAHATO GOVERNMENT ENGINEERING COLLEGE

PURULIA- 723103, WEST BENGAL

CERTIFICATE

This is to certify that Shashank Singh (Roll No. 35000118040), Rohan Roy (Roll No. 35000118045), Abhishek Mondal (Roll No. 35000118074), Super Kumar Murmu (Roll No. 35000118025), Sumit Mahato (Roll No. 35000119047) and Bindeeya Darnal (Roll No. 35000119050) have successfully completed the project titled **“Source Camera Identification by Feature Extraction Using Local binary Pattern”** at Ramkrishna Mahato Government Engineering College under my supervision and guidance in the fulfilment of requirements of 8th Semester, **Bachelor of Technology (Computer Science & Engineering)** of Maulana Abul Kalam Azad University of Technology, West Bengal.

Dr. Prasun Halder
Head of the Department
Computer Science & Engineering

Prof. Pabitra Roy (Project Guide**)**
Asst. Professor
Department of Computer Science & Engineering

ACKNOWLEDGEMENT

We would like to express our greatest appreciation to Prof. Pabitra Roy (Assistant Professor of CSE,RKMGEC) who gave us the golden opportunity to do this wonderful project of Source Camera Identification by Feature Extraction using Local Binary Pattern. We deem it a pleasure to acknowledge our sense of gratitude to our respected professor for his ongoing support during the project, from initial advice, and encouragement, which led us to do this project with a great enthusiasm.

We wish to reciprocate in full measure the kindness shown by Dr. Prasun Halder (H.O.D, Computer Science and Engineering) who inspired us with his valuable suggestions in successfully completing the project work.

We shall remain grateful to Dr. Bibek Chakrabarti, Principal, Ramkrishna Mahato Government Engineering College, for providing us a strong academic atmosphere by enforcing strict discipline to do the project work with utmost concentration and dedication.

Finally, we must say that no height is ever achieved without some sacrifices made at some end and it is here where we owe our special debt to our parents and our friends for showing their generous love and care throughout the entire period of time.

Shashank Singh (Roll – 35000118040)

DATE – 27.05.2022

PLACE – RKMGEC ,

Agharpur , Purulia

West Bengal

Rohan Roy (Roll – 35000118045)

Abhishek Mondal (Roll – 35000118074)

Super Kumar Murmu (Roll – 35000118025)

Sumit Mahato (Roll – 35000119047)

Bindeeya Darnal (Roll – 35000119050)

ABSTRACT

In digital image forensics, camera model identification seeks for the source camera model information from the given images under investigation. To achieve this goal, one of the popular approaches is extracting from the images under investigation certain statistical features that capture the difference caused by camera structure and various in-camera image processing algorithms, followed by machine learning and pattern recognition algorithms for similarity measures of extracted features. In this paper, we use uniform local binary patterns (LBP) as statistical features. LBP features encode local texture information and can be used for many tasks including classification, detection, and recognition. Three groups of each 59 local binary patterns are extracted from the spatial domain of red, green and blue color channels from each image respectively. Multi-class support vector machine is used for classification of 12 camera models from 'Dresden Image Database'. We have achieved an accuracy of 97.69% for source camera identification using local binary pattern method.

TABLE OF CONTENTS

1.INTRODUCTION 8
1.1 Image Processing Pipeline 9
1.2 Demosaicing 10
1.3 Feature Extraction Framework 10
2.PREVIOUS WORKS 11
2.1 Sensor Imperfection Based Method: 11
2.2 Color Filter Array (CFA) Interpolation Methods: 12
3. SOURCE CAMERA IDENTIFICATION METHOD 12
3.1 Local Binary Pattern: 12
4.EXPERIMENTS, RESULTS AND DISCUSSIONS 13
4.1 Dataset for experiments: 13
4.2 Experimental set up 14
4.3 Result Analysis: 14
5.CONCLUSION 16
REFERENCES 17

LIST OF FIGURES

Figure 1:Digital Image Forensic Techniques 8
Figure 2:THE PROCESSING PIPELINE IN A DIGITAL CAMERA 9
Figure 3:DEMOSAICING PROCESS 10
Figure 4 stages of LBP calculation 13
Figure 5: confusion matrix 15

LIST OF TABLES

Table 1:Dresden Image Database 14
Table 2:Cross-validation accuracy of our method 15

1.INTRODUCTION

Digital image producing devices such as cameras, cell phones, camcorders and scanners are nowadays popular. Images as a proof of some important facts are sometimes used as evidence in court. Meanwhile, the popularity of image manipulation software enables simple manipulation of both the contents and source information of digital images, thereby compromises the convincingness of them as evidence. Therefore, knowing the source and authenticity of the images used as evidence is important. Although embedding watermarks at the image producing stage is a solution, it is not widely implemented by manufacturers of image producing devices. Usually, digital image forensics relies only on the digital image itself. Here in this paper, we address the problem of source camera identification from given images. Interested images are assumed to be captured by digital still cameras. [1]

Source camera identification has two different categories and both are popular. The first branch is to match images under investigation with one specific camera. Researchers look for features that capture unique information of that specific camera, such as sensor defects [2] and sensor dust patterns [3]. It is generally a two-class problem, i.e., the specific camera that took the image versus all the other cameras, including other cameras from same make and model.

The second branch is called source camera model identification. Its task is to find brand and model information from a given image. It differs from the first branch in that features should capture characteristics of camera models instead of characteristics of individual cameras. So, the classification problem becomes one model versus other models. In this paper, we focus on this branch.

Digital Forensics as illustrated in Fig.1 is divided into active and passive techniques. In the active forensic techniques, it is fundamental to function on the unique document which has to be accessible from the opening like in the watermarking or digital signature. While the passive forensic is a approach that can function with no prior information about the content material is on hand or no integrity safety mechanisms. It is simple to recognize that this sort of investigation has to be founded on the thorough evaluation of some intrinsic aspects that may also be existing internal the observed information.[4]

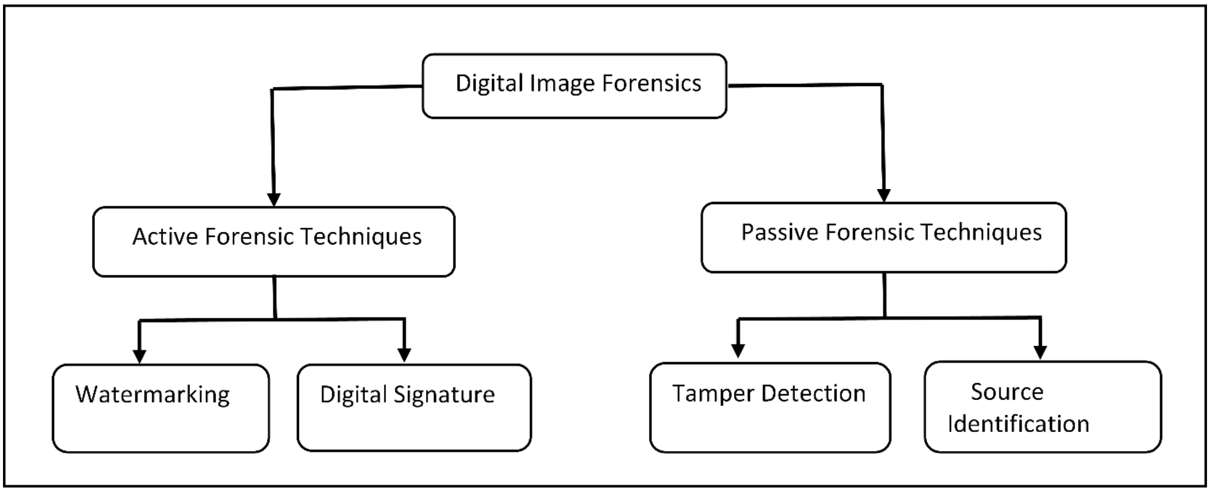

FIGURE 1:DIGITAL IMAGE FORENSIC TECHNIQUES

Previous researches focus on certain stages in this pipeline, such as lens defects [5], color filters array (CFA) and demosaicing [6-9], JPEG compression [10], etc. Some consider more than one stage or the whole pipeline [11-13].

Practical experimental settings for camera model identification require more than one camera from each model in order to remove the ambiguity of whether the features, on which the classifiers are built, capture camera model characteristics or individual camera characteristics [13-14].

In this paper, we propose to use uniform gray-scale invariant local binary patterns (LBP) [15] as statistical features. Considering 8-neighbor gray-level difference for each image pixel around a circle, 59 local binary patterns are extracted, respectively, from spatial domain of red,green and blue color channels, their prediction-error 2D arrays, and the 1st-level diagonal wavelet sub-band of each image. Multi-class support vector machines are built for classification of 12 camera models from '**Dresden Image Database'**.

1.1 Image Processing Pipeline

When a camera takes a photo, light reflected from a scene has to pass by via a set of camera elements recognized as the image processing pipeline in order to produce a final output image. Fig. 2 suggests an overview of a digital camera's processing pipeline. The light first passes via lens and feasible optical filters earlier than hitting the sensor, which can record the depth of light. However, most modern cameras are outfitted with solely one imaging sensor, which means that at every pixel location, solely one color component can be recorded. To resolve this problem, a colour filter array (CFA) is positioned proper earlier than the sensor.

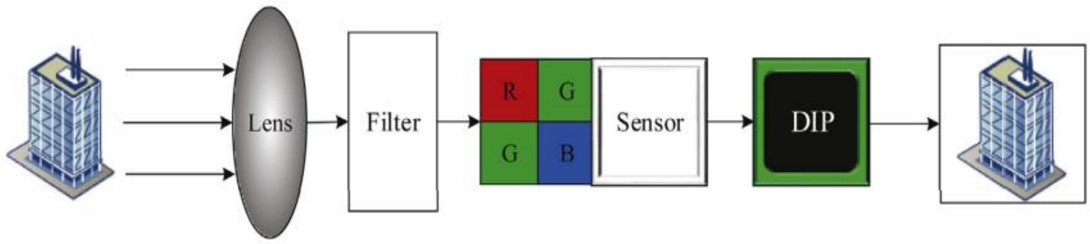

FIGURE 2:THE PROCESSING PIPELINE IN A DIGITAL CAMERA

The CFA solely permits one color component to pass via it so that at the corresponding pixel location, the sensor solely has to record the intensity of one color component. Next, the lacking two color components have to be interpolated to generate the full-resolution image through a technique known as demosaicing. It operates by using estimating the unobserved color values of a pixel using the directly observed color values in a local neighbourhood. Different digital camera models generally employ distinctive demosaicing techniques to interpolate unobserved colour components.

To enhance picture quality and avoid interpolation artifacts, the demosaicing process of most commercial cameras estimates lacking color components using highly nonlinear and complicated algorithms. Additionally, many demosaicing algorithms take advantage of the potential dependencies amongst different color channels through interpolating colour values in one channel using values from the other two color channels. There are additionally demosaicing algorithms which are adaptive to image contents to better preserve textures and edges. After demosaicing, the image often undergoes a set of post-processing operations such as white balancing and JPEG compression [16].

1.2 Demosaicing

Considering cost efficiency and robustness, most commercial cameras add a CFA in front of the sensor to record the three color components of light (R, G, and B) using one sensor. Although only one color value is observed at each pixel, the remaining two color values can be interpolated by using a process called demosaicing. According to whether the correlation between different color channels is utilized, existing demosaicing algorithms can be divided into two categories: correlation and non-correlation. The diversity of demosaicing algorithms gives camera manufacturers more choices, making the demosaicing algorithms become one of the camera-specific characteristics [7].

The estimation of algorithm parameters is complex and inaccurate. the original image is first resampled by CFA to obtain the reconstructed image, which is an approximate estimation of the camera internal image before CFA interpolation. Then, various demosaicing algorithms are performed on the reconstructed image to obtain multiple output images.If the demosaicing algorithms used twice are the same, the output image is similar to the original image. Otherwise, the two images are different.Therefore, features extracted from the output images can be used to identify the source camera.[9]

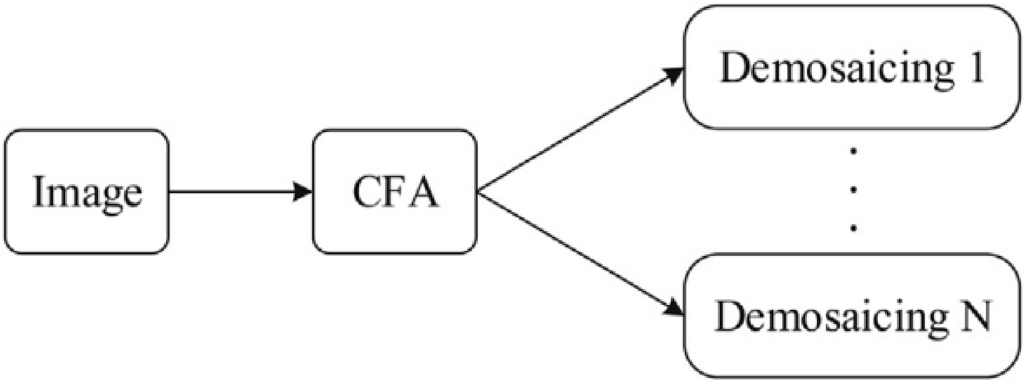

FIGURE 3:DEMOSAICING PROCESS

1.3 Feature Extraction Framework

Inspired by the fact that a quite some of image processing algorithms, such as demosaicing, filtering, JPEG compression, are block-wise implemented inside cameras, it is reasonable to consider that some localized characteristics or artifacts have been generated. These characteristics or artifacts could be effectively captured by the uniform local binary patterns.This process to some extent suppresses the influence of various image contents. The introduction of 'uniform' local binary patterns enables a natural feature dimensionality reduction which is desired by pattern classification algorithms. Therefore, we have used the uniform local binary patterns as features to capture camera model characteristics. As most of the camera image processing algorithms work in spatial domain, a natural choice would be extracting features directly from each color channel(RGB) in spatial domain. From each color channel, a 59-dimensional LBP feature set is calculated by Equation (1) under the assumption of

R P = = 1, 8 (Each 59-D LBP feature set are normalized to eliminate the influence of different image resolution) g and p g represent gray levels of the center pixel and its neighbor pixels, respectively [17].

$$LBP_{P,R}^{u2} = \sum_{P=0}^{P-1} S(g_p - g_c)2^p \ \ldots(1)$$

To conclude, from each color channel, we extracted LBP features from original image resulting in a total of 59x3=177 features.

2.PREVIOUS WORKS

.For the source camera identification problem, various methods have been proposed till now. Extraction of image features , color filter array (CFA) interpolation , presence of lens radial distortion , extraction of photoresponse non-uniformity (PRNU) noise to identify sensor fingerprint , demosaicing artifacts , PCA based spatially adaptive denoising of CFA images for single-sensor digital cameras , using the combination of demosaicing and zooming scheme to detect the color difference of the images .

2.1 Sensor Imperfection Based Method:

Lukas et al [23] proposed sensor pattern noise based method for camera model identification. Pixel non-uniformity (PNU), where different pixels have different light sensitivities due to imperfections in sensor manufacturing processes is a major source of pattern noise. This makes PNU a unique feature in identifying sensors. Photo response non-uniformity (PRNU) casts a unique pattern onto every image the camera captures. This “camera fingerprint” is unique for each camera [24]. They proposed formulation is based on the observation that each in camera and postcamera processing operation leaves some distinct intrinsic fingerprint traces on the final image. We characterize the properties of a direct camera output using a camera model, and estimate its component parameters and the intrinsic fingerprints.The camera fingerprint can be estimated from images known to have been taken with the camera.

$$I = I_o + I_o K + \theta \ \ldots (1)$$

In this equation (1) the camera output image I is the “true scene” image that would be captured in the absence of any imperfections as I0, and K is the PRNU factor (sensor fingerprint) , θ includes all other noise components, such as dark current, shot noise, readout noise, and quantization noise. The fingerprint K can be estimated from N images $I^{(1)}$, $I^{(2)}$, $I^{(3)}$,….. $I^{(N)}$ taken by the camera. Let $W^{(1)}$, $W^{(2)}$, $W^{(3)}$… $W^{(N)}$, are their noise residuals obtained using a denoising filter F.

$$W^{(i)} = I^{(i)} - F\left(I^{(i)}\right) \ \ldots (2)$$

i=1….N and the PRNU factor, K has been derived as:

$$\widehat{K} = \frac{\sum_{i=1}^{N} W^{(i)} I^{(i)}}{\sum_{i=1}^{N} (I^{(i)})^2} \quad \ldots (3)$$

In both the patterns, the authors have tested 9 camera models where two of them have similar CCD and two are exactly the same model. The camera identification is accurate even for cameras of the same model. The result is also good for identifying compressed images. One problem with the conducted experiments is that the authors use the same image set to calculate both the camera reference pattern and the correlations for the images.

2.2 Color Filter Array (CFA) Interpolation Methods:

CFA Interpolation using Expectation Maximization (EM) algorithm: Bayram et al [21] suggest a method to identify the camera model using CFA interpolation. In which, image classification was determined by the correlation structure present in each color band. Each manufacturer uses different interpolation algorithms and somewhat different CFA patterns. Using the iterative Expectation Maximization (EM) algorithm, two sets of features are obtained for classification: the interpolation coefficients from the images and the peak location and magnitudes in the frequency spectrum of the probability maps.

CFA Interpolation using Alternate Projection: In the prior method the interpolation process is performed based on iterative order using any one of the interpolation operations like nearest-neighbor replication, bilinear interpolation, and cubic spline interpolation. Although these single-channel algorithms can provide satisfactory results in smooth regions of an image, they usually fail in high-frequency regions, especially along edges. The alternate projection algorithm [22] exploits an inter-channel correlation, and has given a better performance.

3. SOURCE CAMERA IDENTIFICATION METHOD

Features of digital images are classified into two levels: global and local features. Global properties of an image, includes intensity histogram, frequency domain descriptors, covariance matrix and high order statistics. Local features are defined on local regions with spatial properties, including edges, corners, lines, curves, etc. The features that will be used to recognize camera model through various classification approaches are described here. In this work, there are 50 features like invariant moments, statistical features, GLCM and Color moments have been extracted from each color channels from a number of images of different camera models.

3.1 Local Binary Pattern:

Local Binary Pattern (LBP) is a simple yet very efficient texture operator which labels the pixels of an image by thresholding the neighborhood of each pixel and considers the result as a binary number. Due to its discriminative power and computational simplicity, LBP texture operator has become a popular approach in various applications. An important property is its computational simplicity, which makes it possible to analyze images in challenging real-time settings.

LBP is an operator for texture description that based on the signs of differences between neighbour pixels and central pixels [18]. Fig. 4 shows an example of the calculation of LBP values. For each pixel value in the image, a binary code is obtained by thresholding its neighbourhood with the value of the centre pixel. This binary code can be considered as a binary pattern. The neighbour pixel becomes 1 if the pixel value is greater than or equal to threshold value, and it becomes 0 if the pixel value is less than threshold. Next, the histogram will be constructed to determine the frequency values of binary patterns. Each pattern represents possibility of binary pattern found in the image. The number of histogram bins depends on the number of involved pixels in LBP calculation. If LBP uses 8 pixels, the number of histogram bin will be 2 8 or equal to 256.

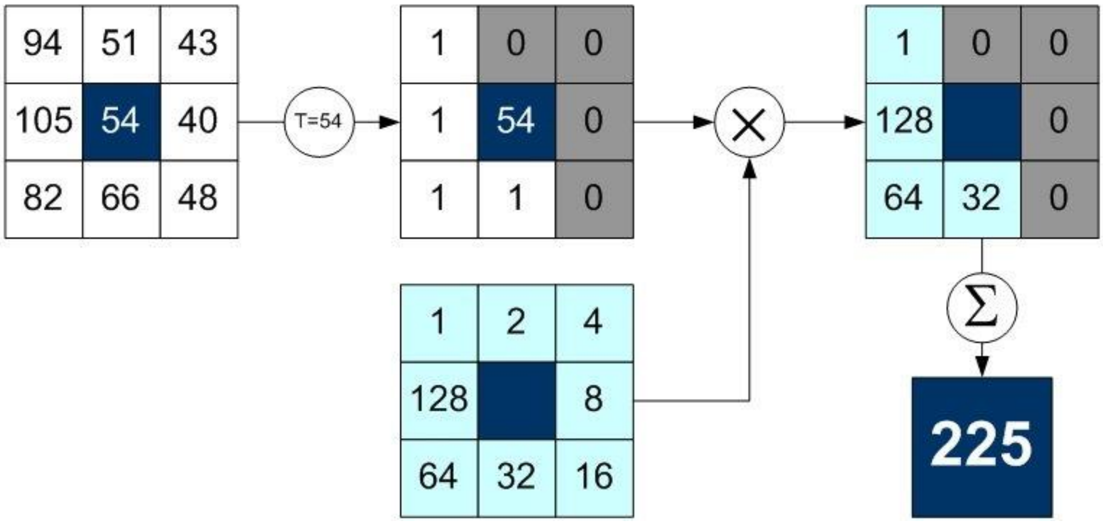

FIGURE 4 STAGES OF LBP CALCULATION

The basic version of LBP operator uses the centre pixel value as threshold to the 3×3 neighbour pixels. Threshold operation will create a binary pattern representing texture characteristic. The equation basic of LBP can be given as follows.

$$\mathrm{LBP}(x_c, y_c) = \sum_{n=0}^{7} 2^n\, g(I_n - I(x_c, y_c)) \quad \ldots(1)$$

LBP(x_c, y_c) is a LBP value at the centre pixel (x_c, y_c). In and I(x_c, y_c) are the values of neighbour pixel and centre pixel respectively. Index n is the index of neighbour pixels. The function g(x) will be zero if $x < 0$ and $g(x) = 1$ if $x \geq 0$. For example (see Figure 1), the centre pixel, 54, will be selected as threshold value. The neighbour pixels are assigned to 0 if its values are less than threshold. Conversely, it becomes 1, if the neighbour pixels are greater or equal to the threshold. The LBP value is computed by applying scalar multiplication between the binary and weight matrices. Finally, the sum of all multiplication results is used to represent LBP value. Therefore, LBP value of the matrix 3 × 3 shown in Figure 1 is $2^0 + 2^5 + 2^6 + 2^7$ = 1 +32 + 64 + 128 or equal to 225.

4.EXPERIMENTS, RESULTS AND DISCUSSIONS

4.1 Dataset for experiments:

We have taken the well known database " **Dresden Image Database**" [19] given in Table -1. All the images are direct camera JPEG outputs which are captured with various camera settings.

TABLE 1:DRESDEN IMAGE DATABASE

Sl. No	Camera Model	No. of Device	Dimension of Image	No. of training Images	No. of Testing images
1	Agfa_DC-830i_0	1	3264x2448	284	78
2	Canon_Ixus55_0	1	2592x1944	179	45
3	Canon_Ixus70	3	3072x2304	444	123
4	Casio_EX-Z150	5	3264x2448	701	224
5	FujiFilm_FinePixJ50	3	3264x2448	465	165
6	Kodak_M1063	5	3664x2748	1880	511
7	Nikon_CoolPixS710_0	5	4352x3264	699	226
8	Olympus_mju_1050SW_0	5	3648x2736	770	270
9	Panasonic_DMC-FZ50_0	3	3648x2736	717	209
10	Pentax_OptioA40_0	4	4000x3000	428	166
11	Ricoh_GX100_0	5	3648x2736	665	164
12	Samsung_L74wide_0	3	3072x2304	521	165
	total	**43**		**7752**	**2346**

From “Dresden Image Database” we have chosen 12 camera models having 43 devices to carry out our experiment.

4.2 Experimental set up

In all of our experiments, multi-class support vector machines (SVM) [20] are trained and used as the classifiers for testing. From the whole dataset, we randomly select one camera for each model, and use all the images taken by the selected cameras for testing. Images from the rest of the cameras form the training data.

The entire dataset is divided into 2 parts: one for training and others for testing. We have taken 7743 data of all camera model for training and 2344 data for testing. Features are extracted using local binary pattern.

The number of neighbors used to compute the local binary pattern for each pixel is 8.We have not encoded the rotation information. Linear interpolation method is used to compute pixel neighbours.

Experiment feature length is 177. Training features are fed into a multiclass SVM classifier to train the SVM model. The training parameter’s cost (c) = 450 and gamma (g) = 0.5 are set to check the 5 fold cross validation accuracy.

4.3 Result Analysis:

The experimental results using our proposed features are reported in Table 2, which gives the average confusion matrix. We got the cross-validation accuracy of 97.69%.

TABLE 2:CROSS-VALIDATION ACCURACY OF OUR METHOD

SL. NO.	Camera Model	Accuracy (%)	Average Accuracy (%)
1	Afga_DC-830i_0	92.3	
2	Canon_Ixus55_0	79.1	
3	Canon_Ixus70	96.7	
4	Casio_EX-Z150	95.1	
5	Kodak_M1063	96.4	
6	Nikon_CoolPixS710_0	99.6	97.69
7	Olympus_mju_1050SW_0	98.7	
8	FujiFilm_FinePixJ50	99.6	
9	Samsung_L74wide_0	98.6	
10	Panasonic_DMC-FZ50_0	98.2	
11	Pentax_OptioA40_0	98.2	
12	Ricoh_GX100_0	98.2	

The confusion matrix is shown in Fig. 5.

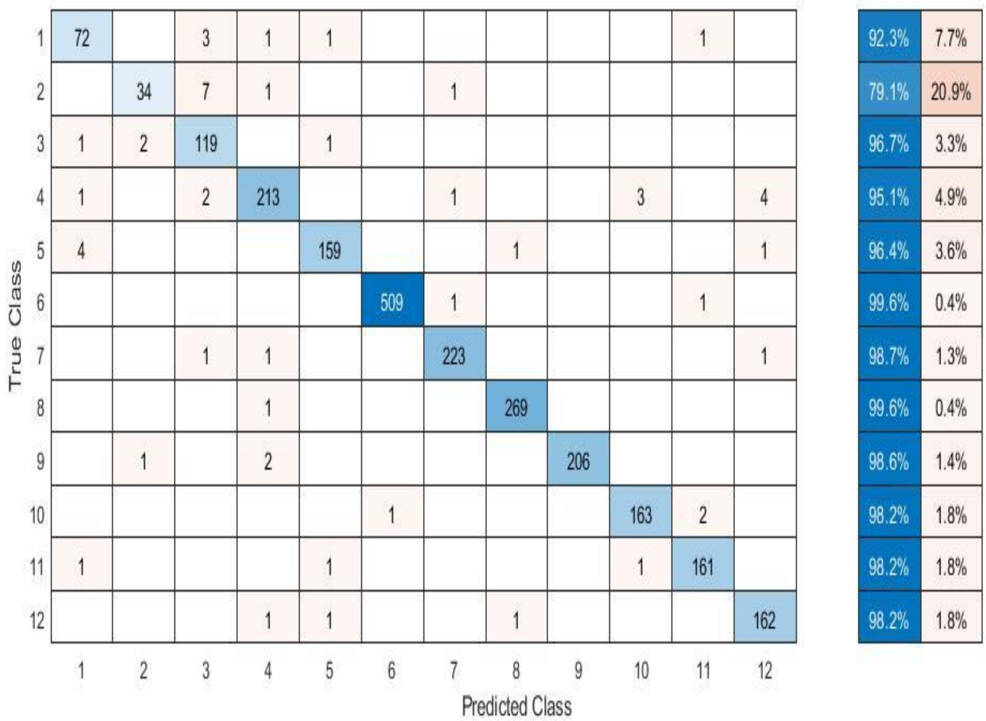

FIGURE 5: CONFUSION MATRIX

5.CONCLUSION

This work is a study for identifying best features of an image in order to perform source camera identification. We have demonstrated in this project the local binary patterns as features for camera model identification.By combining features extracted from the original image, the method has demonstrated 97.69% accuracy in camera model identification. Experimental results endorse that these features are usable to perform source camera identification. In order to improve the result of accuracy, in future we will try to solve the robust feature based local binary pattern for source camera identification.

REFERENCES

[1] Xu, Guanshuo & Shi, Y.Q.. (2012). Camera Model Identification Using Local Binary Patterns. Proceedings - IEEE International Conference on Multimedia and Expo. 392-397. 10.1109/ICME.2012.87.

[2] M. Chen, J. Fridrich, M. Goljan, and J. Lukas, “Determining Image Origin and Integrity Using Sensor Noise,” Information Forensics and Security, IEEE Transactions on, vol. 3, pp. 74-90, 2008.

[3] A. E. Dirik, H. T. Sencar, and N. Memon, “Digital Single Lens Reflex Camera Identification From Traces of Sensor Dust,” Information Forensics and Security, IEEE Transactions on, vol. 3, pp. 539-552, 2008.

[4] Alhussainy, Amel Tuama. “Forensic source camera identification by using features in machine learning approach.” PhD diss., Université Montpellier, 2016.

[5] K. S. Choi, E. Y. Lam, and K. K. Y. Wong, “Automatic source camera identification using the intrinsic lens radial distortion,” Opt. Express 14, pp. 11551-11565, 2006.

[6] S. Bayram, H. Sencar, N. Memon, and I. Avcibas, “Source camera identification based on CFA interpolation,” in Image Processing, 2005. ICIP 2005. IEEE International Conference on, 2005, pp. III-69-72.

[7] Y. Long and Y. Huang, “Image Based Source Camera Identification using Demosaicking,” in Multimedia Signal Processing, 2006 IEEE 8th Workshop on, 2006, pp. 419-424.

[8] A. Swaminathan, W. Min, and K. J. R. Liu, “Nonintrusive component forensics of visual sensors using output images,” Information Forensics and Security, IEEE Transactions on,vol. 2, pp. 91-106, 2007.

[9] H. Cao and A. C. Kot, “Accurate Detection of Demosaicing Regularity for Digital Image Forensics,” Information Forensics and Security, IEEE Transactions on, vol. 4, pp. 899-910, 2009.

[10] E. Kee, M. K. Johnson, and H. Farid, “Digital Image Authentication from JPEG Headers,” Information Forensics and Security, IEEE Transactions on, vol. 6, pp. 1066-1075, 2011.

[11] K. L. Mehdi, H. T. Sencar, and N. Memon, “Blind source camera identification,” in Image Processing, 2004. ICIP '04. 2004 International Conference on, 2004, pp. 709-712 Vol. 1.

[12] O. Celiktutan, B. Sankur, and I. Avcibas, “Blind Identification of Source Cell-Phone Model,” Information Forensics and Security, IEEE Transactions on, vol. 3, pp. 553-566, 2008.

[13] T. Filler, J. Fridrich, and M. Goljan, “Using sensor pattern noise for camera model identification,” in Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on, 2008, pp. 1296-1299.

[14] T. Gloe, K. Borowka, and A. Winkler, “Feature-Based Camera Model Identification Works in Practice,” in Information Hiding. vol. 5806, ed: Springer Berlin /Heidelberg, 2009, pp. 262-276.

[15] T. Ojala, M. Pietikainen, and T. Maenpaa, “Multiresolution gray-scale and rotation invariant texture classification with local binary patterns,” Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 24, pp. 971-987,2002.

[16] Chen, Chen, and Matthew C. Stamm. "Robust camera model identification using demosaicing residual features." *Multimedia Tools and Applications* 80, no. 8 (2021): 11365-11393.

[17] M. J. Weinberger, G. Seroussi, and G. Sapiro, "LOCO-I: a low complexity, context-based, lossless image compression algorithm," in Data Compression Conference, 1996. DCC '96. Proceedings, 1996, pp. 140-149.

[18] X. Tan and B. Triggs, "Fusing Gabor and LBP feature sets for kernel-based face recognition," in Analysis and Modeling of Faces and Gestures. Springer, 2007, pp. 235–249.

[19] T. Gloe and R. Böhme, "The 'Dresden Image Database' for benchmarking digital image forensics," presented at the Proceedings of the 2010 ACM Symposium on Applied Computing, Sierre, Switzerland, 2010.

[20] C.-C. Chang and C.-J. Lin, LIBSVM : a library for support vector machines. ACM Transactions on Intelligent Systems and Technology, 2:27:1--27:27, 2011. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm

[21] Sevinc Bayram, Husrev T. Sencar, Nasir Memon," Source Camera Identification Based on CFA Interpolation," IEEE International Conference on Image Processing, Genova, Italy, pp. 2413-2416, 2005.

[22] Bahadir K. Gunturk, Yucel Altunbasak, "Color Plane Interpolation Using Alternating Projections," IEEE Transactions on Image Processing, vol.11, pp. 997- 1013, 2002.

[23] J. Lukac, J. Fridrich, and M. Goljan, "Digital camera identification from sensor pattern noise," IEEE Transactions on Information Security and Forensics, vol. 1, pp. 205–214, 2006.

[24] A. Swaminathan, M. Wu, and K. J. R. Liu, "Digital image forensics via intrinsic fingerprints," IEEE Transactions on Information Forensics and Security, vol. 3, pp. 101–117, 2008.