CHAPTER 5
DATA-DRIVEN TECHNIQUES FOR DETECTION AND LOCALIZATION OF BLIND IMAGE FORGERY

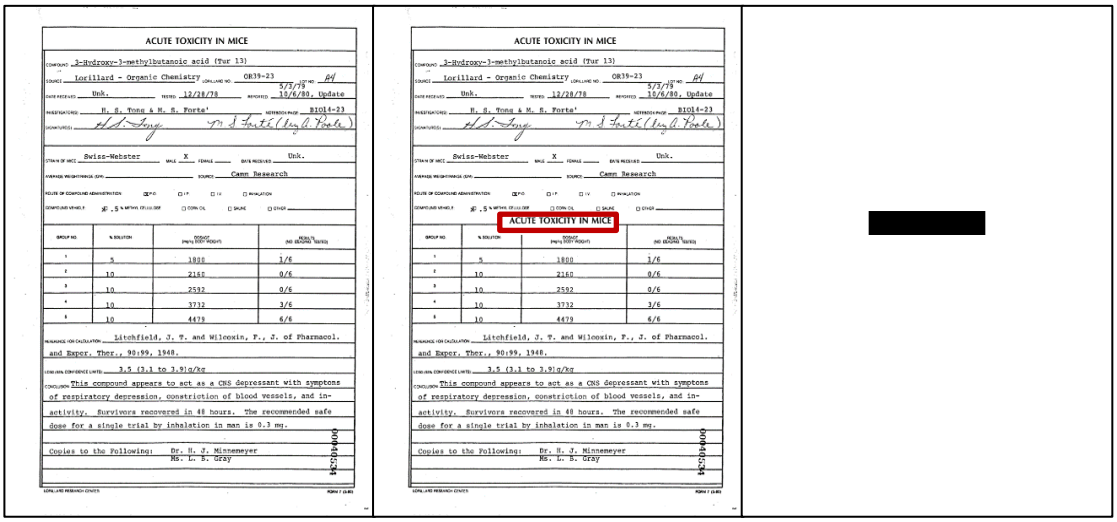
"Blind image forgery" is a term that is evolved for the manipulated image whose type of forgery can be both (i.e. copy-move or spliced image). In the previous two chapters, some methods have been given which detects forgery in an image that has only one type. What if the type of forgery is not only one? In this case, a method is required which doesn't require any assumption of traces from either copy-move forgery or spliced image forgery. On the other hand, with the growing market of digital transactions, the importance of scanned documents is increasing. This creates the pressing demand for the development of a framework that can ensure the creditability of the scanned documents too. To the best of our knowledge, there are none of the data-driven approaches for the detection of tampered scanned documents. In recent years deep learning methods have been applied to many applications of computer vision and image processing from the medical domain to commercial purposes. A lot of classification problems are being solved by using deep learning techniques. To tackle the challenges of blind image forgery detection a deep learning model can be used. The deep learning data-driven approach mimics the human brain. A deep learning model takes input data, extracts features from input and trains itself. Deep learning techniques mostly depend on the amount of data given to learn the model and a large number of data gives better performance. This chapter contributes two deep learning models for the detection of blind image forgery. The first one localizes forgery in natural images and the second one localizes forgery in scanned documents.

5.1 Background

Numerous techniques are developed in the previous two chapters for the detection of different types of forgeries. If forgery done in the image is not known (copy-move or spliced image), this can be defined as blind image forgery. In such a case, forgery detection can't be possible with the help of the only assumption of copy-move forgery or spliced image forgery. On the other hand, a method with the assumptions of both types of forgery takes too much time to detect the forged region. Hence, detection and localization of forged regions in the natural image that has an unknown type of forgery is a challenging task.

Due to the increased use and availability of image alteration tools, digitally generated images and documents can be easily altered by anyone. According to the forensic expert's group [108], alteration to the pages or content of the document includes an example of tampering with the document. Since images and documents are highly used for verification and authorization, the increased use of forgeries in such contents becomes a challenging issue. These forged contents shared on the print media or electronic media have deeply impacted one's belief in the veracity of scanned documents. These contents are not only used by sharing platforms to report the authenticity of the contents to remove criminal practices but also by the court of law itself for imparting justice to the citizen. The use of scanned documents is gaining high attention in digital transactions, medical reports, research publications, and many more. Such contents are also tested by scientific organizations and institutions for the truthfulness of an author's work. Thus, forgeries in digital images, as well as digital documents, is a growing threat to the functioning of various organizations. Political parties and figures have implemented the practice of image and document forgery for gaining public favor and establishment of authority hence public skepticism on such uses has been very high, thus even better measures are

needed to tackle such malpractices. Hence, detection of the altered region in the scanned document is the need of the world. Again, forgery in these scanned images can be done in two ways- copy-move and spliced. An example of the tampered scanned document is shown in Figure 5.1.



(a) (b) (c)

Figure 5.1: Example of forgery in a digital document (a) Original Image (b) Forged Image and the forged region is shown in a red box (c) Ground Truth of forged Image

There are some methods to detect forgeries within the images. An active method involves the use of data embedding processes to include watermarks, signatures, and other additional data to support the provision of the authenticity of the images [109]–[111]. Passive methods do not have any such provisions. But with the help of intrinsic properties of the image such detection techniques are plausible and are one of the major focuses of image forensics [112]. Passive image forgery detection can be done on two levels namely the pixel level and the image level. In a pixel-level detection, the pixels within the image are labeled individually providing the exact region where the forgery has probably occurred. In an image-level detection, the entire image can be labeled as forged or authentic. Image level detection techniques are based on machine learning or deep learning data-driven techniques. Pixel level detection techniques are based on both types-

data-driven as well as based on intrinsic properties. Such intrinsic properties generate when an image passes through an image acquisition pipeline. These properties (blur, color filter array, light inconsistency, and noise, etc.) are useful to detect the forged region in the tampered image and are known as traces or footprints. Techniques for forgery detection using blur artifacts [32], [113], [114] are based on blur inconsistency between forged and non-forged regions. A Colour filter array (CFA) is an interpolation of RGB color in an image and is assumed to follow a fixed pattern hence methods based on CFA artifacts [20]–[22], [59] rely on the disturbance in the local correlation of CFA. Spliced region of an image may have a different incoming light direction than the original one. Techniques using light inconsistency [17], [18], [53]–[56] are reported on this assumption. Noise [13], [14], [16], [61], [62] based techniques find the inconsistent noise pattern in the forged image. These methods work behind assumptions of intrinsic traces and forged regions can detect only in case of availability of these traces. In the absence of these non-homogeneous patterns, detection of forgery is not possible. The most important challenge with noise-based detection techniques is the estimation of the qualified sensor noise pattern. Other limitations of these techniques are a sequence of morphological processes used for segmentation of forged regions (i.e. manual operations are used) and defining block size for estimation of noise pattern (if the block size is taken large, the small fake regions will not be able to detect). Most of the techniques used for forgery detection (i.e. CMF and splicing) have been applied over natural scene images which are acquired from digital cameras only and not over digital documents.

5.2 Research Gaps

For blind forgery detection and localization in natural images, there is a need for assumptions of both types of forgery. But the development of a method with assuming both types of forgery is a challenging task. For such types of problems data-driven

approaches are very popular and vogue. Before deep learning came into the picture, machine learning classification techniques were quite effective for the detection and classification of various types of forgeries in the image [45]–[47]. Important features that can distinguish the original image from a fake image need to be extracted in these techniques. Although the support vector machine (SVM) classifier gives good results, they are unable to localize forged regions in a manipulated image. The method [115] proposed a two-branch CNN architecture to exploit expressive local descriptors for image splicing. Here, these local descriptors are used for splicing localization. But this approach works only for spliced image detection, not for multiple copies of the spliced image. The method [116] utilized dense U-Net architecture with cross-layer interaction in the decoding layer from the encoding layer for detection and localization of image forgery. But in the decoding part, multilayer CNN layers are used to encode features. But the method [116] lacks in encoding more detailed abstract information of the image in the encoder part. Thus, to overcome this the proposed model utilizes skip connections to bypass more detailed information from one layer to another in the encoding part. The model [93] utilized resampling features and a deep learning model for the detection and localization of image forgeries. The authors used the LSTM model for deep learning. But one of the major drawbacks of such an approach is LSTM is used for learning temporal information from the video data or any time-series information. But authors used LSTM in standalone free view images which are logically imperative. In such a scenario, CNN or any encoder-decoder architecture could be advisable.

For scanned document analysis, a series of state-of-the-art methods have been implemented, and experiments have been carried out over and found many false-positive and false-negative pixels. The reason behind this worse result is the intrinsic features of digital documents. Authentic content of the document may have similar-looking features

as the forged content of the document. Also, some letters that are not identical but have similar shape features (i.e. letters ‘c’ and ‘e’ have) may confuse existing methods between forged and non-forged regions. In such cases, data-driven deep learning segmentation methods can work better as deep learning techniques extract a lot of features from images, learn features from labeled pixels and classify pixels of given images into positive and negative classes. The deep learning networks are highly inspired by human networks that are biological neurons, which constitute multiple nonlinear layers for processing simple objects parallelly. Several works of literature are also reported to detect forged images using data-driven machine learning [45]–[47] and deep learning [49] techniques. These are image-level detection schemes and don’t give the exact location of the forged region in an image. Although some deep learning techniques [50] are there which give forged location in an image. However, to the best of our knowledge, none of the deep learning models has yet been used to train forged digital documents to detect the forgery in this scenario. The most important challenge is being faced during the development of the model is the unavailability of the forged document dataset to train the model. The following Table 5.1 presents an overview of the existing state-of-the-art techniques with their merits and demerits.

Table 5.1: An overview of existing state-of-art techniques

Detection Techniques			Works on Scanned Document	Pixel Level Analysis	Doesn’t depend on a single trace
Data-Driven	Machine Learning	[45]–[47]	×	×	✓
	Deep Learning	[49]	×	×	✓
		[50]	×	✓	✓
Intrinsic footprint	Blur	[32], [113], [114]	×	✓	×
	CFA	[20]–[22], [59]	×	✓	×
	Noise	[13], [14], [16], [61], [62]	×	✓	×
	Light Inconsistency	[17], [18], [53]–[56]	×	✓	×

5.3 Proposed Methods

Challenges faced with blind image forgery in natural images and digitally scanned documents are tried to overcome in these proposed methods. The first method is the deep learning data-driven approach for the natural image. The second method is similar, but the most important challenge was the unavailability of the dataset. So, first, a dataset was prepared.

5.3.1 Modified U-Net Model for Detection of Forged Region in Images Acquired from Variant Sources

In this work, a CNN-based deep learning approach has been introduced to localize the false region in the manipulated image by modifying the U-Net[117] architecture which is currently used to segment the neuronal structure in microscopic medical images. Here segmented structures in images are quite similar or have very few variations in size. But in forged regions in manipulated images can have huge variations in size, it may be very small or very large. Large regions require large size kernels while small regions require small size kernels. Also, a deeper network causes overfitting. To overcome these challenges, identity blocks that have smaller size kernels and batch normalization layers are added in U-Net architecture at different places. State-of-the-art methods work with only one type of image. The major contributions made in this work are defined as below:

- Proposed a generic model to localize the forged regions having different types of attacks in the manipulated image (i.e. copy-move or spliced) by modifying U-Net [117] architecture.
- Rigorous comparison of the proposed model with state-of-the-art methods on multiple datasets having different characteristics.
- Provided different test cases by creating different types of synthetic images which have different characteristics (such as medical images, identity documents, natural images, and scanned reports, etc.) to show the model's diverse nature.

From the literature reported previously, it can be concluded that techniques for the detection and localization of forged images exploit the visual information present in the image (i.e. extracting content-based features from the forged image based on assumptions). These features were also used in classical machine learning algorithms to train the model with annotated images and predict the class of forged images. These features are handcrafted features such as Sobel and Canny operators to extract the edges in an image. In the last five years, deep learning architectures are used in many applications of visual recognition and image processing. Their achievement and performance motivate them to apply the same technique to image forensic applications. Deep learning CNN is inspired by the visual cortex, which extracts meaningful information for the classification task. A lot of CNN architectures are there such as AlexNet, ResNet50 to classify images into multiple classes based on their features. These models are only to predict the class of image and do not recognize the tampered region in the image. To classify the pixels of the image a deep learning architecture U-Net was developed in 2015. This architecture was proposed to segment neuronal structures in electron microscopic stacks in biomedical images. Fake regions of the manipulated image can be imagined as neuronal structures of biomedical images and can be localized into forged and non-forged regions using similar techniques. After modifying and extending the U-Net, the model can be used to localize the forged region of the manipulated image.

5.3.1.1 Existing Model

The typical U-Net [117] model consists of two basic paths- one is contraction and another is expansion path. In between contraction and expansion, a bottleneck exists. Thereafter sigmoid activation resides for classification. These three paths contain convolution layers, ReLU Activations, max-pooling layers and up-sampling layers. Convolution operation in the image is performed to extract the features from the image.

To do so a filter or a kernel is placed to the upper left corner of the image then scanned from left to right and from up to down until scanning of the image is not completed. In mathematics, the convolution between two functions, $f, g : \mathbb{R}^d \rightarrow \mathbb{R}$ can be defined as:

$$[f \otimes g](x) = \int_{\mathbb{R}^d} f(z)g(x-z)dy \tag{5.1}$$

In the case of discrete data integral changes into sum. As images are a case of 2D spatial data, convolution of filter $f(x, y)$ of size $(2k + 1, 2k + 1)$ and 2D image $g(i, j)$ of size (m, n) can be defined as:

$$[f \otimes g](i, j) = \sum_{x=-k}^{k} \sum_{y=-k}^{k} f(x, y)g(i - x, j - y) \tag{5.2}$$

The weighted sum of parameters of extracted features and bias passes through an activation function. ReLU is the acronym of the rectified linear unit and is commonly used in deep learning neural networks. Sigmoid and Tanh are also activation functions, but they have a vanishing gradient problem which is overcome by ReLU. This activation function converts all negative values to zero and retains positive values as it is. Mathematically ReLU can be defined as:

$$ReLU(y) = \max(x, 0) \tag{5.3}$$

Activated values pass through a max-pooling layer in the contraction path and an up-sampling layer in the expansion path. In the max-pooling layer, the operator is placed to the output of the previous layer and slid over the block by block according to the size of the filter and stride. The filter gives a single output on each window i.e. the maximum element of the window. The objective of max-pooling of the hidden layer is to reduce the dimensionality. Also, the assumption can be made by using features contained in the window. Dimension reduction of the features cuts the computational cost. If stride is taken

as '*s'* filter size is taken as '*f*' then the size of the max-pooled output will be $\left(\frac{n-f}{s}+1\right)$, where *'n'* is the height or width of the output of the previous layer. Similarly, upsampling is a process of increasing the resolution of the features. As the focus is on the localization of the forged region in the forged image, high dimension output is required. Repeated max-pooling layers reduce the large image into a very small size image from where localization is near to impossible. So, to transform a very small size image into a large size upsampling process is done. Sigmoid activation function is generally used for binary class classification problems. As forged regions need to be localized in the manipulated image it is required to classify pixels of the image into the forged and authentic pixels. The sigmoid function gives the probability of a pixel being authentic or forged i.e. (value in between 0 to 1). Suppose 'z' is input to the sigmoid function and *f(x)* is features of the output layer of the model and '*w'* is the corresponding weight then:

$$z = wf(x) + b \tag{5.4}$$

Where *'b'* is bias added to the function. If a predicted class is *y = 1* denotes that pixel belongs to the forged region and *y=0* belongs to the authentic region then the probabilities using sigmoid function will be:

$$p(y = 1) = \frac{1}{1 + e^{-z}} \tag{5.5}$$

$$p(y = 0) = \frac{e^{-z}}{1 + e^{-z}} \tag{5.6}$$

5.3.1.2 The Proposed Modified Architecture

An identity block is added parallelly to each max-pooling layer of the contraction path in the modified model. A batch normalization layer is also added after each convolution operation. Batch Normalization is a process of normalizing the features. For example, if some features are from 0 to 1 and others are 1 to 1000, features should be

normalized by adjusting and scaling the activations [118]. This also speeds up the training speed. Batch normalization also reduces the overfitting problem. Batch Normalization BN can be done on batch $B = \{x_1 .. x_n\}$ with the parameter to be learned α, β as below [118]:

Mean of the batch-

$$\mu_B = \frac{1}{n}\sum_{i=1}^{n} x_i \tag{5.7}$$

The variance of the batch-

$$\sigma_B^2 = \frac{1}{n}\sum_{i=1}^{n}(x_i - \mu_B)^2 \tag{5.8}$$

Normalized values-

$$\widehat{x_i} = \frac{x_i - \mu_B}{\sqrt{\sigma_B^2 + \epsilon}} \tag{5.9}$$

Adjusting and scaling of activation-

$$BN(x_i) = \alpha\, \widehat{x_i} + \beta \tag{5.10}$$

When a network becomes deeper, it is very difficult to choose a parameter to learn. A degradation problem occurs with the increasing depth of the network and accuracy gets saturated. These layers end up making results worse rather than making them better. The solution for this problem given in [119] is using residual or identity blocks. Adding this block to the network doesn't harm the network. As reported in [119] residual block can be seen as in Figure 5.2.

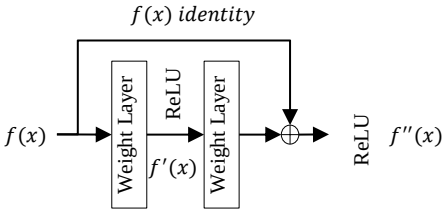


Figure 5.2: Architecture of the Identity Block

Suppose a layer gives an output $f(x)$ to some input I and after adding an extra layer with identity block as shown in the above figure output is $f''(x)$. So, after applying ReLU on the output $f''(x)$ can be defined as:

$$f''(x) = ReLU[z''(x) + f(x)] \tag{5.11}$$

$z''(x)$ is the output of the second layer and before the addition of identity block which can be defined as:

$$z''(x) = w''(x)f'(x) + b' \tag{5.12}$$

$$f''(x) = ReLU[w''(x)f'(x) + b' + f(x)] \tag{5.13}$$

If L2 regularization is used for optimization, then $w''(x)$ and b' will shrink and if $w''(x) = 0$ and $b' = 0$. Then

$$f''(x) = ReLU[f(x)] \tag{5.14}$$

And, since ReLU is already applied on $f(x)$, all elements of $f(x)$ is non-negative.

$$f''(x) = f(x) \tag{5.15}$$

This shows that adding an identity block doesn't affect the functionality of the network. The architecture of the extended model is illustrated in Figure 5.3. This model also consists of contracting path, bottleneck and expansion path the same as in U-Net [117]. The model is extended by adding identity block in the contraction path and batch normalization layers in all paths. The added layers are shown in the illustrated diagram by dotted red bounding boxes. The contraction path contains two convolution layers with a kernel size of 5x5, each followed by batch normalization and the ReLU activation layer. Its output is used in the identity block as well as in the max-pooling layer of stride 2. The output of the identity block is added with the output of the max-pooling layer. This whole process is repeated four times. Next is the bottleneck, which contains two 5x5 convolution layers, each followed by batch normalization and ReLU activation function. The output

of the bottleneck is used for upsampling in the expansion path. Its output is concatenated with the output features of the convoluted layer of the contracting path. Concatenated output features are again passed through two layers of convolution operation followed by batch normalization and the activation function to extract the precise information. This whole process is repeated four times.

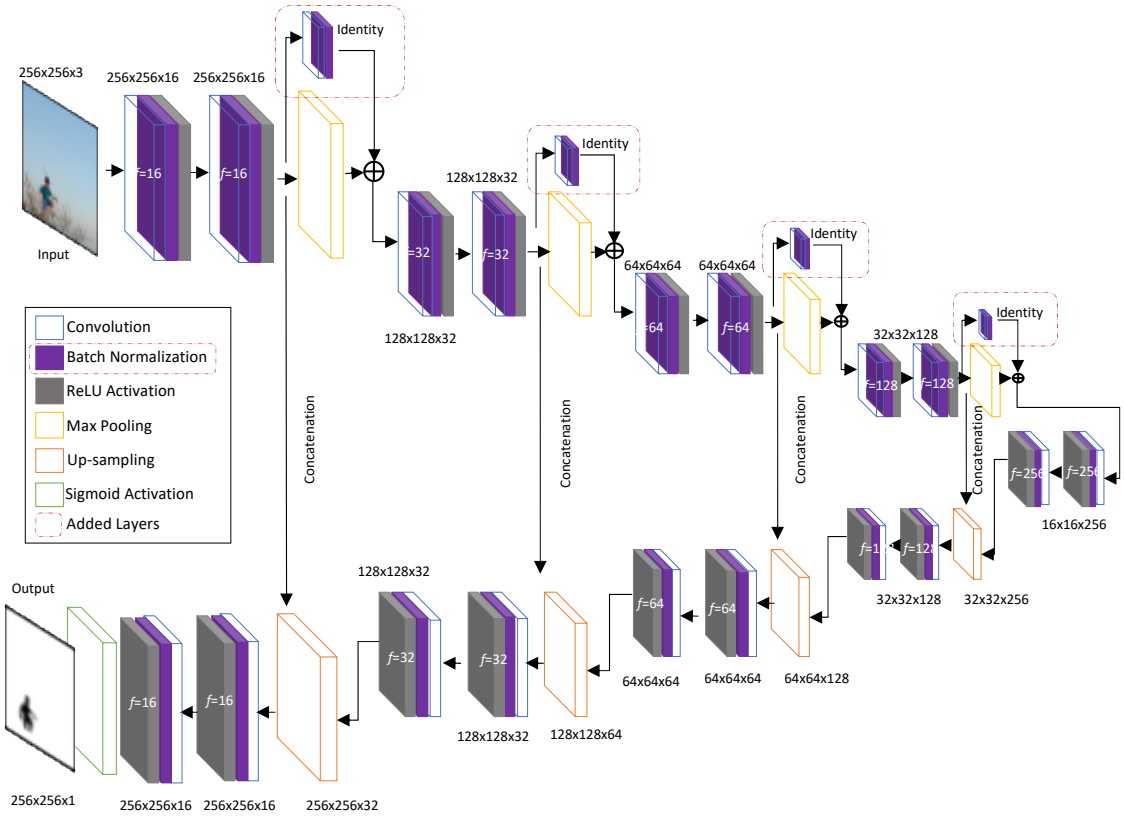


Figure 5.3: Architecture of the proposed model for localization of manipulated regions in Forged Image

After this, the sigmoid activation function is used to predict the pixel class which gives the probability of pixel which belongs to the particular class mentioned in the above subsection. To evaluate the model binary cross-entropy loss function is used. To reduce the loss optimization technique adam optimizer is used. The loss function is a method to evaluate the model on the given data. If the prediction result deviates much from the actual value, then the loss function will be very large otherwise it will be small. To classify pixels of the image into the forged and authentic pixel, which is the binary class classification problem of the model uses binary cross-entropy loss function. If for *N*

number of samples, predicted probability given by sigmoid activation function for a pixel is *p(y)* and actual predicted class 'y' is 1 for forged pixel and 0 is for authentic pixel. Then, mathematically the average loss function can be defined as:

$$L = -\frac{1}{N}\sum_{i=1}^{N}[y_i \log(p(y_i)) + (1 - y_i)\log\left(1 - (p(y_i))\right)] \quad (5.16)$$

$p(y_i)$ is defined earlier which comes from the sigmoid activation layer. This function shows that for each forged pixel log($p(y_i)$) is added and for each authentic pixel (1-log($p(y_i)$)) is added.

5.3.1.3 Experimental Analysis and Discussion

For the demonstration and the evaluation of the performance of the proposed model, the training and testing are performed on the publicly available datasets. This section discusses the following points related to the experiment:

- Preparation of Training, Validation and Test data
- Required tools, language, and system configuration to develop and train the program for the proposed model
- Parameters required to train and test the model

Table 5.2: Brief Description of Publicly Available Forged Image Datasets

S. No.	Forgery Type	Dataset	Size (no. of Images)	Format	Size of Image
1	Copy-Move, Splicing	NIST [52]	13470	.png	1024x1024
2	Copy-Moe	COMOFOD [84]	200	.png	512x512
3	Spliced	Dresden [52]	35712	.png	1024x1024
4	Spliced	CASIA 2 [65]	12614	.png	256x256 to 1024x1024
5	Spliced	IEEE IFS [66]	1500	.png	512x512 to 3648x2736

To train the model, a dataset is required which should have a forged image as well as its ground truth mask that can identify the forged region in the image. For training and testing purposes combinations of publicly available datasets have been used here. A deep learning model requires a significant amount of data to train the model. This is the reason combinations of the dataset have been used here. The following table contains the depth

detail about the datasets i.e. name of the dataset, number of images in the dataset, type of forgery of the dataset, etc. The above datasets are combined for training and testing. To train the presented model 10% of the dataset images are used to test the trained model. The rest of the 90% images of combined datasets are used for training and validation purposes. Except for the images of the above dataset, synthesized images are also created to test the trained model. These synthesized images are computer-generated forged images of copy-move as well as spliced. To validate trained models whether they are fine enough to detect forged regions or not with scenarios of forgeries in different domains whose acquisition sources are different. Several test cases of fake images are generated synthetically, given to the trained model and evaluated performance. These images are downloaded via google search. These test cases are classified into the following categories based on their acquisition sources. These categories are highly sensitive areas where an alteration in the image may harm one in any way.

- Medical Images
- Identity Documents
- Natural Images
- Scanned Reports/Documents

To implement the above presented deep learning network Python 3.0 language is used here with *TensorFlow, OpenCV, matplotlib* and *NumPy* libraries. Spyder 4.0 editor is used to write the code. The code is written from scratch by building a class for dataset preparation to network and testing the trained model. After building all the classes and networks the code is submitted to the Supercomputer (Param Shivay)[120] for the training of the model where the above-prepared dataset is already stored. The implemented network is trained by submitting the job on a supercomputer via *slurm* manager.

Deep learning models require some external parameters that need to be defined as static and are not being changed during the training and testing process. For example, the size of all images should be fixed and should be the same during the entire training or testing process. To fit the model training sample data is divided into training and validation data where 49,329 images are used for training and 2466 images are used for validation. The following parameters are defined in the presented model with their values:

- Image Size: 256x256
- Batch Size: 64
- Number of Filters: [16, 32, 64, 128, 256]
- Convolution Filter Size: 5x5
- Max Pooling on 2x2
- Upsampling on 2x2
- Padding: Same

Except for these above-mentioned parameters, the model is trained on given 200 epochs but with early stopping criteria where early stopping condition monitors validation accuracy with ten amounts of patience. The model stops at the 91st epoch when it reaches its patience limit.

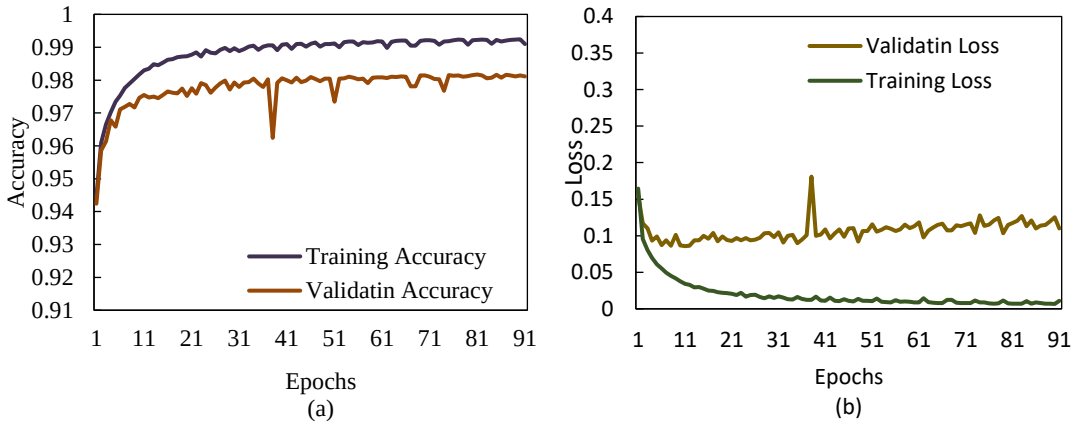


Figure 5.4: Training Result of the proposed model (a) Accuracy (b) Loss on Different Epochs

Above mentioned publicly available datasets are combined to train and test the model. To train and validate the model 90% of data has been used. The training and

validation accuracy of the model for all 91 epochs has been shown using the line graph given in Figure 5.4 (a). The loss of the model is also illustrated in Figure 5.4 (b). These figures show that the network has been completed with the training and validation accuracy of 0.991 and 0.981 respectively. Loss has been estimated during the training of the model with training and validation losses 0.0109 and 0.1098 respectively which is much acceptable. Now it turns to test the model on different images rather than images used in training or validation.

Table 5.3: Average result of the proposed model and other standard models on the available dataset

Method	*p*	*r*	*a*	*s*	*f*	*m*	*csi*	*f1*	*mcc*
EncDec	0.8320	0.8510	0.8280	0.5180	0.3390	0.0062	0.8260	0.8410	0.5460
U-Net	0.9728	0.9888	0.9636	0.5549	0.4451	0.0112	0.9622	0.9803	0.5916
Proposed	0.9980	0.9950	0.9940	0.9570	0.0434	0.0045	0.9930	0.9970	0.9400

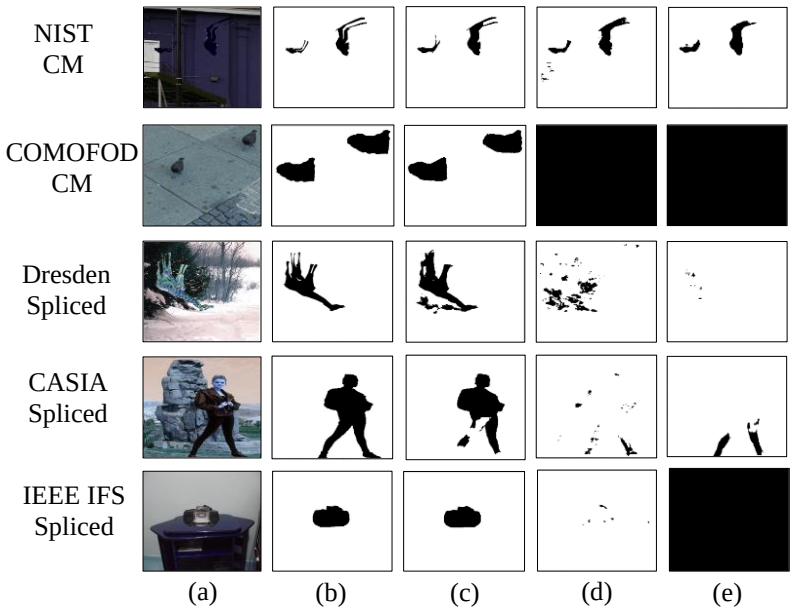


Figure 5.5: Visual Results on a different image of Dataset (a) Forged Color Image (b) Ground Truth Mask (c) Result by the proposed model (d) Result by U-Net model (e) Result by Encoder-Decoder Model

To test the trained model rest of the images of the combined dataset have been used. The total number of images from the dataset which are tested on the trained model is 5480, which includes both copy-move and spliced images in it Existing U-Net and

Encoder-Decoder models are also implemented on the same configuration system with the same model parameters and tested on the same number of images. Results of the modified presented model and existing models are compared in Table 5.3. This table states that the presented model performs better than other similar segmentation models used earlier. These results are average metrics of 5480 test images. The forged region of manipulated images is demonstrated in Figure 5.5. This figure consists of one manipulated image with its ground truth mask from each publicly available dataset. The resultant localized image of the presented model is beside the ground truth mask and the other two are the resultant localized image of U-Net [117] and a normal Encoder-Decoder model (EncDec) [121].

Table 5.4: Comparison of Result on Different Evaluation Metrics for Images of Different Publicly Available Dataset

Dataset	***Model***	***p***	***r***	***a***	***s***	***f***	***m***	***csi***	***f1***	***mcc***
NIST CM [52]	*Proposed*	0.9995	0.9957	0.9953	0.9853	0.0147	0.0043	0.9952	0.9976	0.9309
	U-net	0.9992	0.9898	0.9893	0.9763	0.0237	0.0102	0.9890	0.9945	0.8571
	EncDec	0.9965	0.9910	0.9879	0.8962	0.1038	0.0090	0.9876	0.9937	0.8240
COMOFOD [84]	*Proposed*	0.9931	0.9979	0.9920	0.9451	0.0549	0.0021	0.9910	0.9955	0.9595
	U-net	0.8875	1	0.8875	0	1	0	0.8875	0.9404	-
	EncDec	0.8875	1	0.8875	0	1	0	0.8875	0.9404	-
Dresden Spliced [52]	*Proposed*	1	0.9759	0.9775	1	0	0.0241	0.9759	0.9878	0.8492
	U-net	0.9381	0.9550	0.8988	0.0749	0.9251	0.0450	0.8983	0.9464	0.0345
	EncDec	0.9374	0.9997	0.9372	0.0213	0.9787	0.0003	0.9371	0.9676	0.1254
CASIA Spliced [65]	*Proposed*	0.9610	0.9999	0.9662	0.8011	0.1989	0.0001	0.9609	0.9801	0.8772
	U-net	0.8397	0.9965	0.8391	0.0677	0.9323	0.0035	0.8373	0.9114	0.2023
	EncDec	0.8554	0.9999	0.8596	0.1715	0.8285	0.0001	0.8554	0.9220	0.3825
IEEE IFS [66]	*Proposed*	1	1	1	1	0	0.0000	1.0000	1.0000	0.9998
	U-net	0.9495	0.9998	0.9494	0.0354	0.9646	0.0002	0.9493	0.9740	0.1738
	EncDec	0.9477	1.0000	0.9477	0	1.0000	0	0.9477	0.9732	-

The result of each image on different evaluation metrics is also compared with the state-of-the-art segmentation techniques in Table 5.4, which shows that the proposed technique works better than others. The result also provides information about the type of forgery in the image. The results conclude that the model is not only able to define the forged region, but it also talks about the type of forgery in the image. As images picked from

different datasets are having different types of forgery (i.e. copy-move or spliced) and presented model distinguishes the type of forgery with the location of the manipulation region in the image.

Except these results method is also tested on different test cases of images that have different characteristics like in nature, formation process, graphics, texture, shape, format, or in color combination. In this way, these test cases are categorized into four different categories in this work as discussed above (i.e. natural, medical, etc.). So, four different types of images are taken and manipulated in such a way that common human eyes can't classify the manipulated region. These images are passed through a trained network and checked whether the network can predict false regions in these images or not. The visual and quantitative results are shown in the following Figure 5.6 and Table 5.5 respectively. Discussion and analysis of these images are as below:

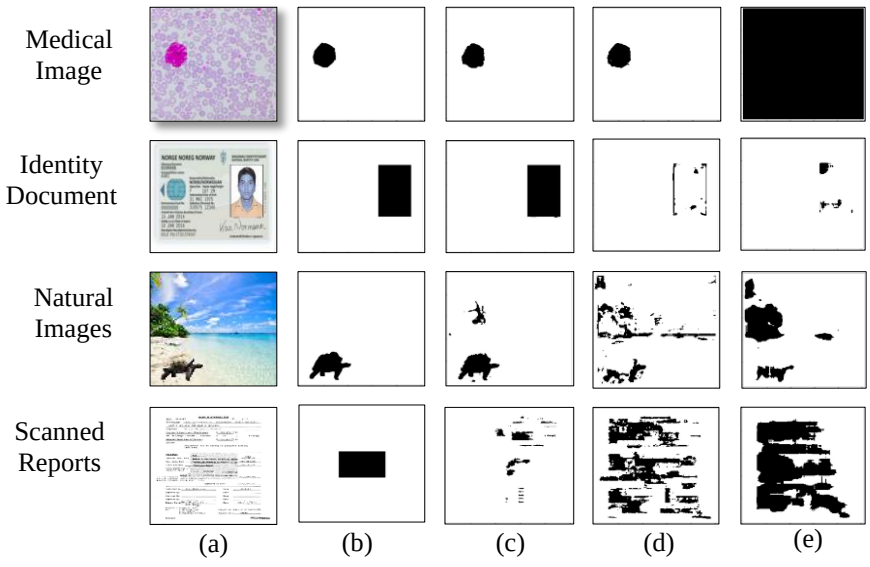


Figure 5.6: Visual Results of the proposed methods on different test cases acquired from different sources

Test Case 1: Medical Image

Medical images are captured through medical imaging devices and are visual representations of organs or tissues present in the interior body. These images are for

clinical analysis and diagnosis of injuries or any health issue. Here, a healthy histopathological image is taken, and a cancerous cell is spliced into it. This spliced image will change the opinion of the doctor as well as the patient. A healthy person will now treat as an unhealthy person. Therefore, analysis of forgery in medical images is important. An object segmentation technique in deep learning U-Net is already there which segment the important object in medical images. This model can easily segment spliced cells in the medical image but in the case of images other than medical images whose structure is different from than medical image does not give satisfactory results. The presented model gives better results in other cases also.

Table 5.5: Result of Proposed Technique on Different Test Cases

Test Cases	***Model***	***p***	***r***	***a***	***s***	***f***	***m***	***csi***	***f1***	***mcc***
Medical Image	*Proposed*	0.9992	0.9992	0.9985	0.9750	0.0250	0.0008	0.9985	0.9992	0.9737
	U-net	0.9998	0.9991	0.9989	0.9943	0.0057	0.0009	0.9989	0.9995	0.9818
	EncDec	0.9706	1	0.9706	0	1	0	0.9706	0.9851	-
Identity Document	*Proposed*	1	0.9999	0.9999	1	0	0.0001	0.9999	1	0.9997
	U-net	0.8848	0.9977	0.8843	0.0876	0.9124	0.0023	0.8830	0.9379	0.2500
	EncDec	0.8862	0.9994	0.8872	0.0989	0.9011	0.0006	0.8858	0.9394	0.2888
Natural Image	*Proposed*	0.9990	0.9888	0.9883	0.9774	0.0226	0.0112	0.9878	0.9939	0.8773
	U-net	0.9715	0.9266	0.9038	0.3982	0.6018	0.0734	0.9021	0.9485	0.2336
	EncDec	0.9727	0.8897	0.8706	0.4466	0.5534	0.1103	0.8681	0.9294	0.2068
Scanned Reports	*Proposed*	0.9144	0.9786	0.8973	0.0894	0.9106	0.0214	0.8965	0.9454	0.1198
	U-net	0.9391	0.7789	0.7532	0.4985	0.5015	0.2211	0.7415	0.8515	0.1855
	EncDec	0.9447	0.6629	0.6585	0.6145	0.3855	0.3371	0.6381	0.7791	0.1663

Test Case 2: Identity Documents

Identity documents are computer-generated documents that are used to identify a person. This document contains information on the identity document of belonging (a person whose document is) such as name, address, date of birth, etc with small size photographs. A very common incident that happens generally is identity theft in which a photograph of belonging is replaced with some other person. Here, forgery analysis comes into the picture and this is important to identify which region of an identity

document is forged. In the given example photograph original identity document is replaced with the author's photograph. The result of the presented method on this example gives far better than other state-of-the-arts.

Test Case 3: Natural Images

Natural Images are those which are captured from normal cameras or mobile devices in the natural environment. Forgeries in these images are common. Forgeries in these images are done to defame a person, to mislead any legal activity or to hide something. To identify the forged region in natural images a lot of techniques are there. These techniques don't give better results in the case of others. The presented technique gives better in all the above cases.

Table 5.6: Comparison of Proposed Technique with Similar Techniques of Forged Region Localization

Method	Type of Attack	Type of Method	Level of Detection	Average Results		
				p	***f1***	***mcc***
EXIF-SC [122]	CM, SP	Statistical	Pixel	0.349	0.378	0.365
ELA [123]	CM, SP	Statistical	Pixel	0.154	0.218	0.122
DCT [124]	CM, SP	Statistical	Pixel	0.159	0.271	0.193
CFA[20]	CM, SP	Statistical	Pixel	0.218	0.271	0.193
BLNVS [13]	SP	Statistical	Pixel	0.347	0.292	0.278
PKNV [14]	SP	Statistical	Pixel	0.302	0.367	0.333
NIBIF [16]	SP	Statistical	Pixel	0.249	0.227	0.191
Noiseprint [50]	CM, SP	Statistical + Deep Learning	Pixel	0.399	0.444	0.403
Proposed	CM, SP	Deep Learning	Pixel	0.998	0.997	0.940

Test Case 4: Scanned Documents

The scanning process of a report is similar to the natural image acquisition process. Scanned reports look like normal pdf documents but when a paper is scanned through a scanner, its content saves as an image. In the scanned document's characters are saved as images. These characters sometimes blur and are noisy. Contents are sometimes skewed in nature during the scanning process and thus their properties may differ from normal

natural/medical/identity images. Hence, scanned reports need a different trained model to detect forged regions in the scanned report. Though the proposed method gives better precision, recall, accuracy and f1-score value its miss-rate is also higher than others which is not good at all. Also, other state-of-the-art techniques don't give satisfactory results. This is a further challenge that needs to resolve in future work.

This can be concluded from these results that the proposed method works better in all cases except the scanned reports where none of the compared methods works well. A comparison of the proposed method is also required with statistical/mathematical methods (state-of-the-arts) which have already been proposed earlier for the detection of the forged region in the manipulated images. The following Figure 5.7 is the comparison bar graph of the state-of-the-art with the proposed method on evaluation metrics average precision, average f1-score, and average mcc values.

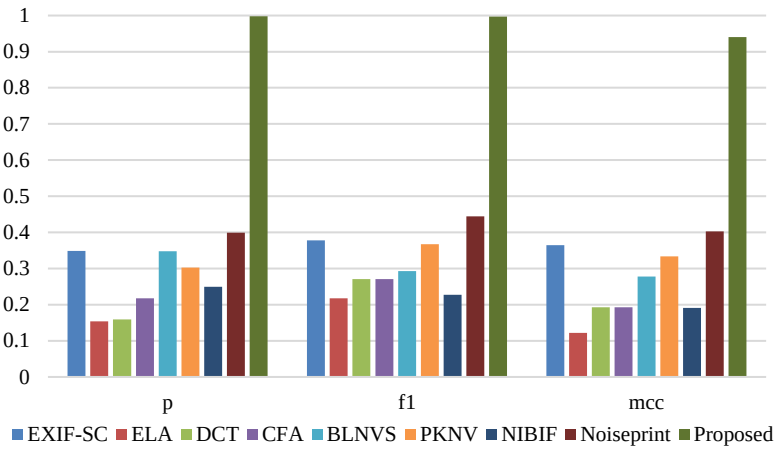


Figure 5.7: Comparison of the proposed method with state-of-the-arts techniques for Image Forgery Detection

This bar graph shows that the proposed method gives higher precision than other state-of-the-art methods. Table 5.6 compares the result of the state-of-the-art with the proposed method and it also compares the type of attacks, level of detection and type of method. The type of attacks explains which type of attacks can detect the method. Several

test instances tell how many images are passed through the model to test the technique. The level of detection explains whether the model can localize the false region in the forged image or only classifies the forged image and the type of method is for statistical, or deep learning-based method. The values presented in the comparison table are based on the claimed values by the authors of [50]. These in-detail demonstrated experiments show that the model is evaluated up to neck and crop. Also, the model is tested on different types of images having diverse characteristics that point out the model can work with different types of images.

5.3.2 An investigation and analysis of forged digital document using deep inception network

There is a publicly available dataset (Forged Digital Document) [125] which is made on payslip digital documents with their corrupted labeled document. However, there is the limitation of unavailability of segmented ground truth and small size for the training of the model. Considering above mentioned challenges and problems a dataset of scanned documents is collected from FUNSD [126]. Original documents of FUNSD [126] are used to generate a dataset of the forged digital document. The original non-tampered dataset contains 149 training and 50 test documents. From these original document images, 3184 tampered documents are generated and 144 original documents are used as non-tampered. These documents are used for training and testing the proposed deep learning model. On the other side, deep learning segmentation models like encoder-decoder [121], U-Net [117] have the challenge of variation in the size of the forged region. Choosing a larger kernel size gives global information while a smaller size kernel gives local information. To overcome this challenge, the inception block without dimension reduction of the Inception model has been used in the proposed model. The proposed model has also been evaluated on a publicly available dataset after constructing

segmented ground truth of corrupted documents. This technique uses more than ~10M trainable parameters to train the model. The following major contribution has been made in this work-

- To overcome the challenge of the unavailable dataset of forged documents, in this work a dataset of different types of forged documents has been constructed with its segmented ground truth. For the diversity in the dataset, different types of image processing operations along with geometrical transformations have been performed over the forged regions of document images.
- An object segmentation deep learning model has been developed to detect a tampered document as well as the forged location in this tampered document. This model tries to overcome the challenges of small tampered objects and training on a small dataset. This segmentation model locates the forged region in the document by separating them in the form of background and foreground.
- Global as well as local experimental analysis of the proposed model on the generated dataset along with the comparison of the proposed model with the state-of-the-art techniques dedicated to object segmentation models. Here term global defines the image level analysis while local defines pixel-level analysis of the forged document. The trained model has also been evaluated on the publicly available dataset.

5.3.2.1 The Proposed Dataset

Deep learning models mimic human brains and almost depend on training. So, the overall dependency of deep learning models is either on the dataset or the architecture. A dataset of forged document data is very important to train the model and to the best of our knowledge, none of the forged digital document datasets exists online in the public domain (from where a model can learn). Hence, a dataset FD3 (**F**orged **D**igital **D**ocument **D**ataset) having a different type of forgeries is constructed in the laboratory of Computing and Vision at IIT (BHU), Varanasi. Some pre-processing operations and geometrical transformations have been applied to the forged region in documents which effectuate the

multitudinous of this dataset. This dataset is generated from the existing FUNSD [126] dataset, which is dedicated to text detection, optical character recognition (OCR), spatial layer analysis, and form understanding. This dataset contains 149 documents for training and 50 document images for testing purposes. From these documents, we have fabricated 3284 forged documents and picked out 144 original documents as it is for the non-tampered document (see the tree structure of directories in Figure 5.8).

Three types of basic forgeries have been constituted to construct a dataset: (i) Copy-Move Forgery- In this type of document forgery, a region of document is copied and pasted in the same document intentionally to hide any information of document or to misguide someone with false information. (ii) Splicing of documents content: The content of one document is copied and pasted to another document. (iii) Resampling of information: Information is resized in the form of either downscaled or upscaled. Along with these forgeries, four different pre-processing/post-processing operations and three different geometric transformations have been carried out over the forged documents. Details of operations implemented over the forged region using Adobe Photoshop are as follows:

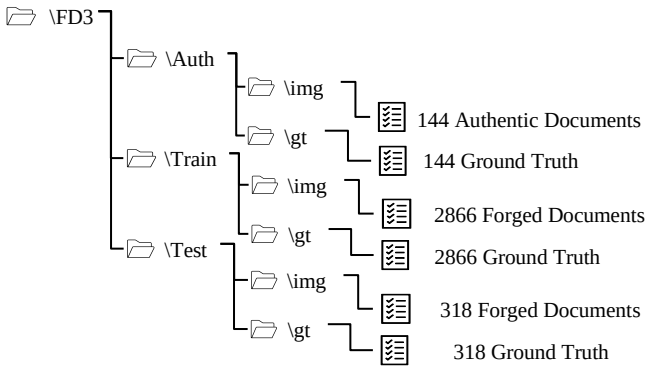


Figure 5.8: Tree structure of directory and content of the constructed dataset

1. First is a simple forgery in which a region is copied from one document and pasted to the same document itself (copy-move) or another document (splicing) without any pre-processing or post-processing operations.
2. The second is brightness enhancement of the forged region of the document up to 100 points on the scale.
3. The third post-processing operation is contrasted stretching up to 100 points on the scale. This operation is implemented in a forged region similar to brightness enhancement.
4. The blurring of the forged region is the fourth operation that is brought out up to two neighboring pixels.
5. In the fifth operation, white Gaussian noise is added up to 20% to the forged region of the document.
6. Except for these image processing operations, three geometrical transformations are carried over the forged region. Among these three operations, the first transformation is translation.

Table 5.7: Details of the Constructed Dataset

S. No.	Forgery	Size (h x w)	Operations	Number (Images)	Number (GT)	Total
1	Copy-Move	1000 × 755 to 1000 × 777	*Simple*	199	199	398
2			*Brightness Enhancement*	199	199	398
3			*Contrast Stretching*	199	199	398
4			*Noise Addition*	199	199	398
5			*Blurring*	199	199	398
6			*Translation*	199	199	398
7			*Rotation*	199	199	398
8			*Scaling*	199	199	398
9	Splicing	1000 × 755 to 1000 × 777	*Simple*	199	199	398
10			*Brightness Enhancement*	199	199	398
11			*Contrast Stretching*	199	199	398
12			*Noise Addition*	199	199	398
13			*Blurring*	199	199	398
14			*Translation*	199	199	398
15			*Rotation*	199	199	398
16			*Scaling*	199	199	398
17	Non-tampered	1000 × 755 to 1000 × 777	*Simple*	144	144	288
			Grand Total:	3328	3328	6656

7. The second geometric transformation is the scaling of the forged region. This operation accomplishes the definition of the third type of forgery which is the resampling of information.

8. The last operation is the rotation of the forged region which is rotated up to 10° in the forged document.

Table 5.7 gives the overall detail of the dataset with the number of images in the dataset. All presented documents and ground truth in the constructed dataset are in png format. Except for the table, the directory tree is also shown in Figure 5.8 for the dataset which explains the location of documents in the dataset with their directory names. Dedicated directories are created for the training and testing of the model. One more directory is created for authentic documents that are not tampered with in any way and can be used to validate the model with non- tampered documents also.

5.3.2.2 The Proposed Model

There are numerous deep learning models for object detection. But when these techniques were used of forged documents. They are not able to detect forged regions in these documents. The major limitations of these models are intrinsic features of digital documents. As forged regions in tampered digital documents may have similar shape features but they are not identical. Similarly copied regions may much smaller as only one or two words need to change in the whole document. Considering these challenges, a deep learning model is proposed in this work for the training of the forged digital document. The proposed architecture is dedicated to forged document detection which uses pixel-level analysis and localizes forged regions in the respected document. The idea behind this architecture is taken from two different deep learning models, one is the U-Net segmentation model dedicated to the biological image and another is the Inception model. The SegNet [121] is for semantic segmentation of the objects in the image and the U-Net [117] is for medical image segmentation. The SegNet [121] architecture doesn't concatenate the full feature map of the encoder block to the up-sampling layer decoder block to extract the boundary details of objects in the image. Also, the Inception block is

not used in SegNet whereas U-Net architecture although concatenates the full feature map of the contraction path to the up-sampling layer of expansion path but doesn't use inception block to extract global as well as local information. Thus, the advantage of the U-Net model that has been projected in the proposed architecture is that U-Net doesn't require much training data and gives better accuracy for test cases.

Similar to U-Net architecture the proposed architecture has two basic paths-contraction and expansion. The contraction path extracts feature maps from the input image using convolution operation and provided several kernels. The contraction path has four down-sampled blocks and each block has convolution layers, batch-normalization with rectified linear unit (ReLU) activation, and one max-pooling layer. Corresponding to each down-sampled block there is one up-sampled block that has one up-sampling layer, convolution layers, and batch-normalization with ReLU activation; see Figure 5.10. This model extracts features in the form of diamond shape i.e. the number of kernels taken to extracts features are increasing in contraction path and are decreasing in expansion path. Hence feature maps are in the form of thinner to wider in contraction path and then wider to thinner in the expansion path. The idea of the inception block is taken from the Inception model and used in the proposed architecture. The novelty of this proposed architecture lies in the feature set extracted in the down-sampled and up-sampled blocks. In which features are extracted in such a manner that architecture is invariant to the size of the forged region i.e. inception blocks extract global features using larger kernel size and local features using smaller kernel size; see Figure 5.9.

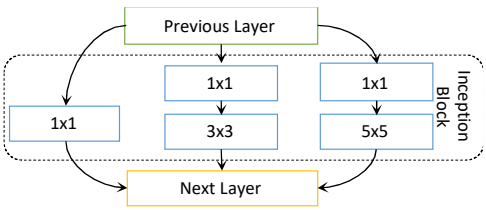


Figure 5.9: Inception Block without Dimension Reduction used in Proposed Architecture

The contraction path contains four down-sampled blocks. Each block is having one simple convolution layer with *n* number of kernels of size 5x5 and then batch-normalization with ReLU activation. Features extracted from this layer are then going to the inception block where five convolutional layers with *n* number of kernels of three different sizes 1x1, 3x3, and 5x5 exist; see Figure 5.9. Feature maps extracted from this layer are concatenated, batch-normalized, and then applied ReLU activation on normalized output. Following that, a window of 2x2 max pooling with stride 2 (non-overlapping block) is applied over the resulting output and sub-sampled feature map. The max-pooling is used to reduce the feature set that helps in learning parameters easily. The smallest sub-sampled block of contraction path shows a much lesser feature set hence actually defines what exactly they are but lesser defines where they are. More number of down-sampled blocks sub-sampled the feature representation and removes boundary details of forged regions. This causes problems in the accurate segmentation of tampered regions in forged documents. Therefore, the feature map of each down-sampled block is necessary for the boundary details which should be stored.

In between the contraction and expansion path of the model, there is a bottleneck. This contains two convolutional layers having a maximum number of kernels i.e. 256 of size 5x5 and followed by batch-normalization and ReLU activation function to each convolutional layer.

Similar to the contraction path, four up-sampled blocks are used in the expansion path. One up-sampled block of expansion path is corresponding to one down-sampled block of contraction path. An up-sampled block contains one up-sampling layer which upscales the resulting feature map (got from bottleneck) using a 2x2 window with stride = 2 (non-overlapping). The stored feature map from the corresponding down-sampled block is concatenated with the up-scaled feature set. A convolution layer having *n* number

kernels of 5x5 size is used to extract more precise dense features from a concatenated feature map. Again, an inception block is used with batch normalization and applied ReLU activation over the output feature map.

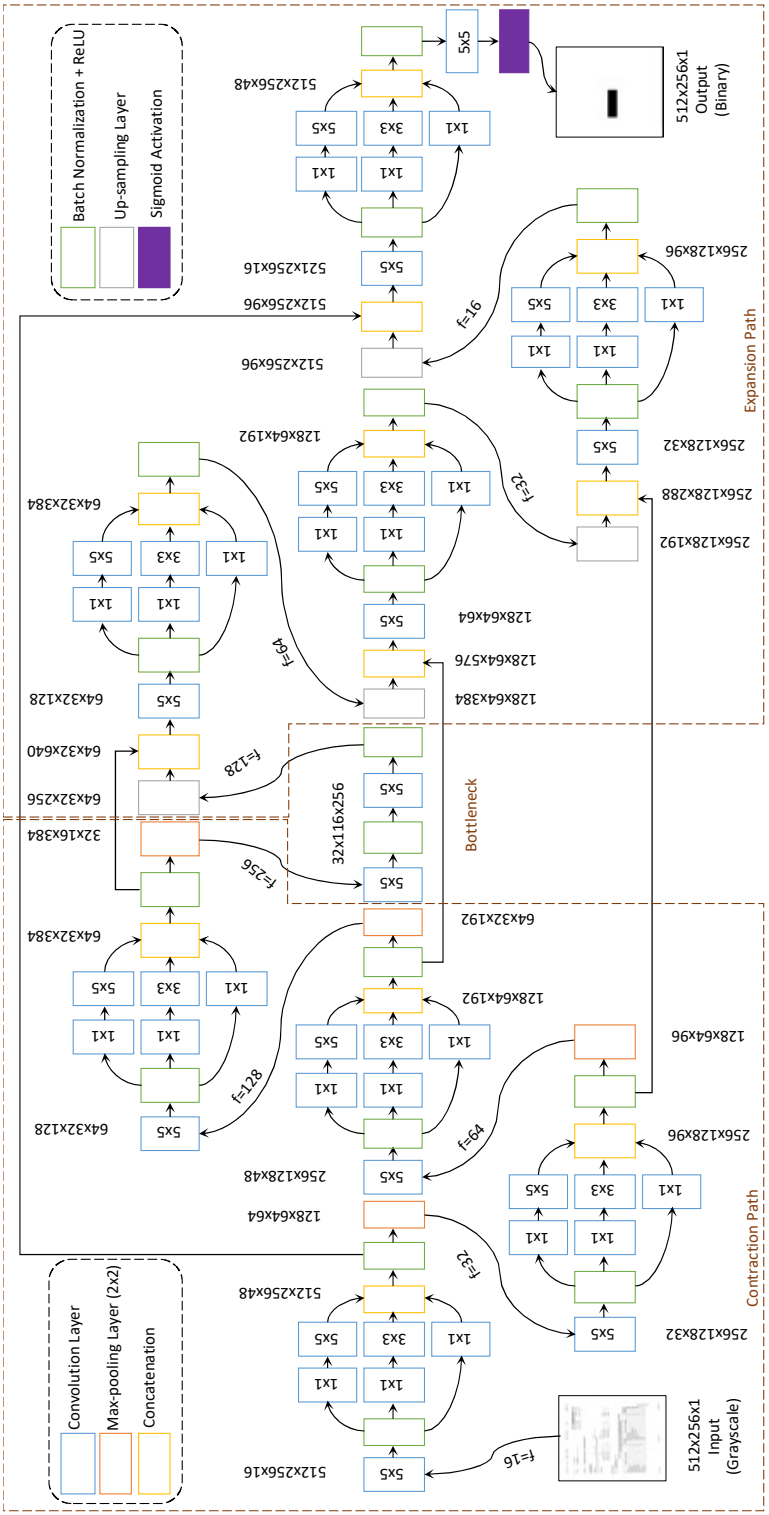


Figure 5.10: Architecture of the proposed model for forged document detection

The up-sampled block feature set defines where exactly the objects are but less about what they are. The output from the final up-scaled block has a high dimension feature map which is then fed to the sigmoid activation function to classify each pixel of the input image into the binary class of tampered and non-tampered i.e. one or zero. Here pixel zero represents the tampered pixel and one represents the non-tampered pixel.

5.3.2.2.1 Training of the proposed model

The constructed dataset FD3 has been used to train the model. In the convolution operation padding is taken as 'same' which means input ought to have zero paddings so that the height and width of output remain the same as the input. The dataset contains 2866 document images for training purposes. The size of each document varies from 1000×777 to 1000×755. For the training, documents are resized, and height is converted to 512 pixels, and width is converted to 256 pixels. For a better feature map and accuracy large input size of the document is preferred and to get good performance batch size is taken as 16. An array of filters that extract feature maps from the input image is {16, 32, 64, 128, 256}. On the other hand, the model uses adaptive momentum (Adam) optimizer. Sigmoid activation is used to compute pixel-wise classification over the final feature map obtained from the final up-sampled block of expansion path. Suppose '*z'* is input to the sigmoid function and *f(x)* is features of the output layer of the model and '*w'* is the corresponding weight then:

$$z = wf(x) + b \tag{5.17}$$

Where *'b'* is bias added to the function. If the predicted class is *y = 1* denotes that pixel belongs to non-tampered pixel and *y=0* belongs to the tampered pixel then the probabilities using sigmoid function will be-

$$p(y = 1) = \frac{1}{1 + e^{-z}} \tag{5.18}$$

$$p(y=0) = \frac{e^{-z}}{1+e^{-z}} \tag{5.19}$$

To classify pixels of the image into the forged and authentic pixel, which is the binary class classification problem of the model uses binary cross-entropy loss function. If for *N* number of samples, predicted probability given by sigmoid activation function for the pixel is *p(y)* and actual predicted class '*y'* is 1 for non-tampered pixel and 0 is for tampered pixel. Then, mathematically the average loss function can be defined as:

$$L = -\frac{1}{N}\sum_{i=1}^{N}[y_i \log(p(y_i)) + (1-y_i)\log\left(1-(p(y_i))\right)] \tag{5.20}$$

$p(y_i)$ is defined earlier which comes from the sigmoid activation layer. This function shows that for each non-tampered pixel log($p(y_i)$) is added and for each tampered pixel (1-log($p(y_i)$)) is added. The maximum number of epochs given to the model is 200, but the model runs to 104 because of an early stopping condition. An early stopping criterion to the model is given in which the model monitors the accuracy. If the accuracy of the model is going to reduce, the model monitors until its patience value. Patience value is given in the early stopping condition which is 10. The training and validation accuracy along with the loss of the model on the FD3 dataset is shown in Figure 5.11.

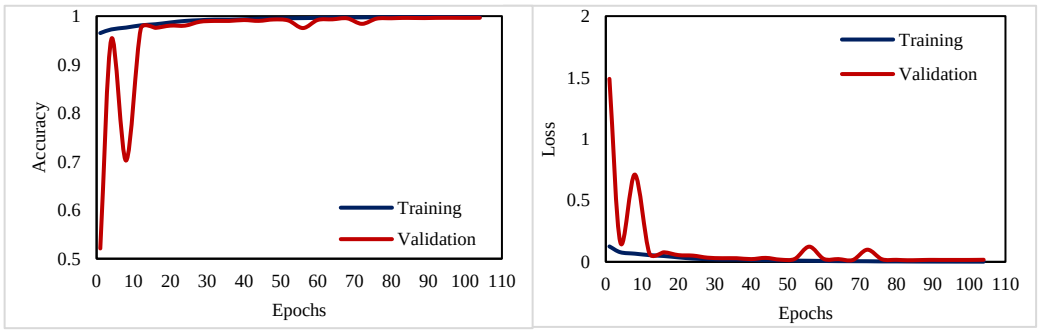


Figure 5.11: Training result of the proposed model on FD3 dataset

Training results of a similar model are compared in Table 5.8. There are four other deep learning models for semantic segmentation whose training and validation accuracies

are compared with the proposed model along with their training and validation loss. These models are called tensor flow segmentation models. Two very similar models U-Net [117] and LinkNet [127] have almost similar results but less than proposed but others have worse training results. Thus, there is no need for the comparison of the test results of these two segmentation techniques.

Table 5.8: Training Result of the proposed and other standard models

Model	Training Accuracy	Training Loss	Validation Accuracy	Validation Loss
Proposed Model	99.86	0.0017	99.65	0.017
U-Net [117]	96.51	0.1035	94.49	0.2235
LinkNet [127]	96.24	0.1121	90.36	0.3034
Basic EncDec [121]	89.86	1.5390	98.64	0.2030
PSPNet [128]	5.54	2.8747	5.83	2.8659

5.3.2.3 Result Analysis and Discussion

In this section demonstration of the proposed model as well as a comparison of the proposed model with other state-of-the-art techniques have been given. To demonstrate the test result and performance of the proposed model, the number of test documents from the FD3 dataset and publicly available dataset [125] has been given to the trained model. Test documents contain both types of documents- authentic and forged. For the comparison of the proposed model with state-of-the-art models, there was a need for similar models. So, other models were taken from segmentation models of the TensorFlow-Keras package and trained on the FD3 dataset. Their trained weights are stored to test the performance of existing and proposed models. Test cases are given to these trained models and their comparison has been done on different parameters in the terms of qualitative as well as quantitative. These test cases include test data from constructed FD3 and publicly available dataset Forged Document Dataset [125]. To train and test the model Ubuntu Server is used with application Python 3.7, Anaconda/3, CUDA/10 installed in it.

To evaluate the proposed model and compare the result of the proposed model with the state-of-the-art techniques image-level analysis and pixel-level analysis have been used. To evaluate performance, a confusion matrix has been used for both the analysis and to evaluate time and space, prediction time per document and memory used to store trained models have been used. There is no single evaluation metric that describes the confusion matrix in its best way. Sometimes precision, recall, and accuracy mislead results that's why it is important to measure results using miss-rate, f1-score, and MCC values. In this work, an analysis has been done of the proposed method on the constructed dataset. So, this section is divided into two sub-section- one is the image-level analysis and another is pixel-level analysis.

5.3.2.3.1 Image Level Analysis

To detect the tampered region in a forged document it is necessary to detect the document whether is forged or not? If the document is forged, then analysis of the tampered region is important. The proposed model classifies documents pixels into two classes tampered and non-tampered. Based on this analysis of the document is detected as a forged or non-forged document.

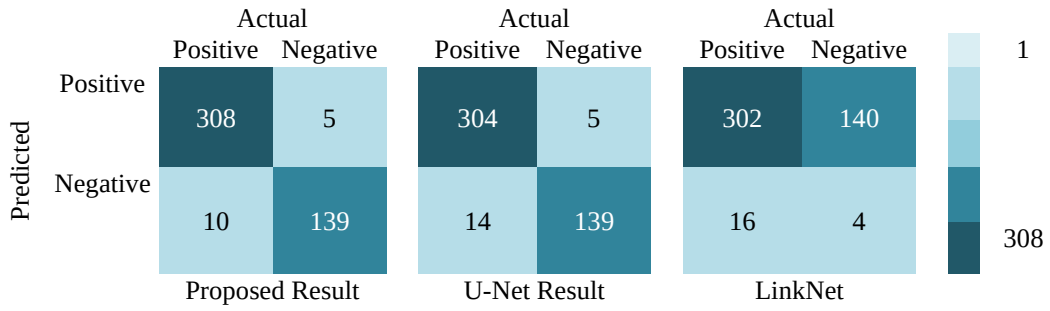


Figure 5.12: Confusion matrix and corresponding heat map of Image-Level analysis of the proposed and compared models on test cases of FD3 dataset

144 authentic and 318 forged documents are taken for test cases. Test results of authentic and forged documents are compared with their ground truth and created a confusion matrix based on that result. If the f1-score of test results is greater than 0.01

then test cases are taken as correctly detected as forged. On the other hand, false positives and false negative pixels must be minimum in the authentic document. In Figure 5.12, the confusion matrix is drawn with the heat map which shows the image-level analysis of the proposed and similar method.

Using the result from confusion matrix image-level comparison of the proposed model with other similar models are shown in Table 5.9 using evaluation metrics precision, recall, accuracy, specificity, miss rate, f1-score, and MCC values. A compared state-of-the-art technique U-Net gives good results in the case of image-level analysis but less than the proposed model. A similar deep learning model LinkNet don't give even satisfactory result on test cases.

Table 5.9: Image Level Comparison of the Proposed Model with Existing Models

Model	*p*	*r*	*a*	*s*	*m*	*f1*	*mcc*
Proposed	0.9840	0.9686	0.9675	0.9653	0.0314	0.9762	0.9253
Unet	0.9838	0.9560	0.9589	0.9653	0.0440	0.9697	0.9067
Linknet	0.6782	0.9277	0.6472	0.0278	0.0723	0.7835	-0.0880

5.3.2.3.2 Pixel Level Analysis

For the pixel level analysis, the trained model is evaluated on a test set of both datasets (publicly available and constructed). Some challenges were faced during the testing of the model on a publicly available dataset. The publicly available dataset [125] doesn't have a segmented ground truth of the labeled tampered region. One more challenge was faced with the dataset was that size of the dataset is less for the deep learning model. To overcome the later challenge FD3 is constructed and to overcome the former challenge we have created segmented ground truth of labeled tampered test data. Thus, the proposed model is also evaluated on publicly available data using its tampered and segmented ground truth data. The created tampered document is given to the trained model and the resultant document is compared with the ground truth mask. One of the visual output results from the test data publicly available dataset is shown in Figure 5.13.

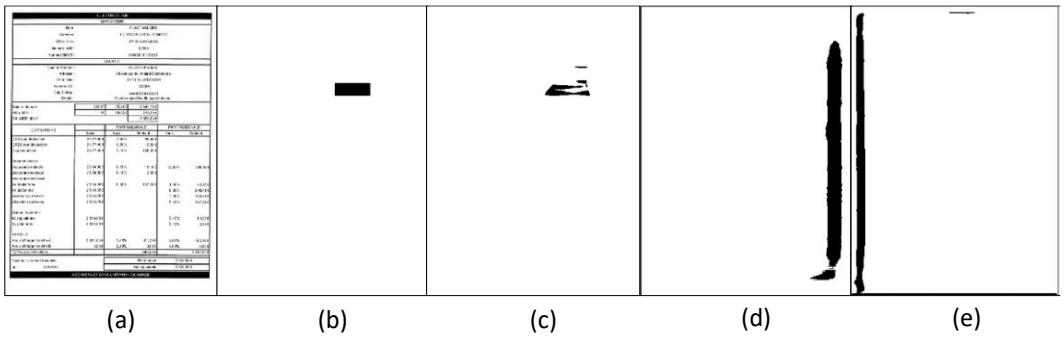


Figure 5.13: The visual result of the test data from the publicly available data (a) Tampered document (b) Ground Truth Mask (c) Result given by the proposed model (d) Result given by U-net (e) Result given by Linknet

The quantitative result of the test data produced by the proposed model and comparison of this result with other models is shown in Table 5.10. It can be concluded from the visual and quantitative results that the U-net and Linknet segmentation models are not able to give even results. Accuracy gives around 0.94 and 0.93 in the case of U-Net and Linknet. At first glance, one can think of that model gives a nice result. However, this result is just because of the imbalanced set of pixels. Several true positive pixels are zero, precision and recall values become zero but the true negative pixels have a larger number and this gives higher accuracy and specificity value. Well, in this case, miss rate, f1-score, and mcc values are giving an understanding of the trained model. So, from this result, it can be said that the proposed model gives better results than state-of-the-art in the case of a publicly available dataset.

Table 5.10: Comparison of Average result of the proposed model and state-of-the-arts on the publicly available dataset [125]

Model	*p*	*r*	*a*	*s*	*m*	*f1*	*mcc*
Proposed	0.75762	0.564935	0.995656	0.998716	0.435065	0.647241	0.652146
Unet	0	0	0.947467	0.954198	1	-	-0.0184
Linknet	0	0	0.939412	0.946086	1	-	-0.02005

Similarly, in-depth result analysis is done on constructed FD3 dataset. In FD3 there are 318 forged document test cases in which 159 cases have copy-move forgery and 159 cases have spliced forgery. The effect of different operations performed on the forged region is also analyzed in this work. So, one by one analysis of each operation and type

of forgeries is given in this subsection. The average quantitative result on both types of forgery (copy-move and spliced) test cases is given in Table 5.11.

Table 5.11: Comparison of Average result on copy-move forged documents for the proposed and other standard methods

Model	*p*	*r*	*a*	*s*	*m*	*f1*	*mcc*
Proposed	0.6395	0.4309	0.9624	0.9920	0.5691	0.4698	0.4844
Unet	0.8188	0.3218	0.9650	0.9987	0.6782	0.4267	0.4756
Linknet	0.1069	0.3039	0.9170	0.9500	0.6961	0.1398	0.1309

Although precision and accuracy values given by the U-Net model are higher than the proposed. The reason behind this is the imbalanced number of positive and negative values. The 159 cases of copy-move forgery are taken and given to the trained model. During tampering of the scanned document, some pre-processing/post-processing operations were performed. One by one result analysis on each test case of different operations of copy-move forgery is performed. The comparison of the proposed model with other state-of-the-art techniques is done on various evaluation metrics. The visual result given by the proposed model and other state-of-the-art techniques is shown in Figure 5.14.

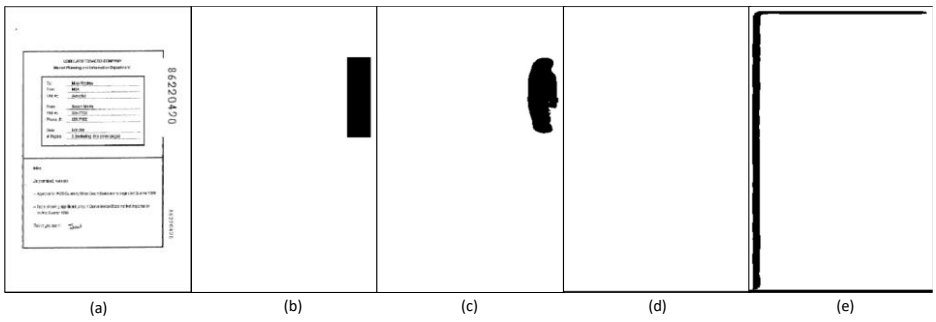


Figure 5.14: The visual result of the copy-move forgery test data from the constructed dataset FD3 (a) Tampered document (b) Ground Truth Mask (c) Result given by the proposed model (d) Result given by U-net (e) Result given by Linknet

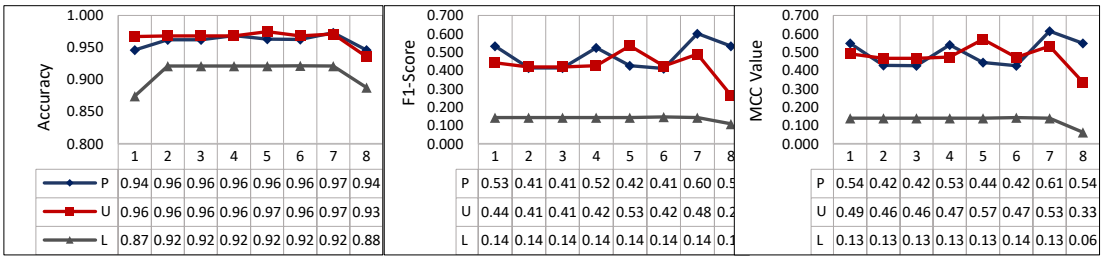
The average quantitative result produced by the proposed model on the test cases of different operations of copy-move forgery its comparison with state-of-the-art models is given in Table 5.12. Miss rate is a very important performance measure which gives knowledge about misses positive classes and miss rate should be minimum. From Table

5.12, the miss rate of the proposed model can be seen where the miss rate of the proposed model is lesser than others. Quantitative values on performance measures of different post-processing operations are shown in Table 5.12. From the demonstrated quantitative results in Table 5.12, it can be concluded that the proposed method works better in the case of documents. Only the case of rotation where UNet can localize the forged region, this is so only because the forged region is larger than the other cases. The border pixels in some cases are larger than the positive pixels of results which leads to a precision value higher in the case of the U-Net model. But the overall result depends on f1-score and MCC value which are the best performance measure in all cases, here these values of the proposed model is higher than others.

Table 5.12: Quantitative Result of Copy-Move Forgery Documents on Different Operations for the proposed and other standard models

	Operations	*p*	*r*	*a*	*s*	*m*	*f1*	*mcc*
Proposed	Simple	0.8366	0.8742	0.9969	0.9982	0.1258	0.8550	0.8536
	Brightness	0.8818	0.8727	0.9975	0.9988	0.1273	0.8772	0.8759
	Contrast	1.0000	0.8830	0.9983	1.0000	0.1170	0.9379	0.9389
	Blur	0.7532	0.4742	0.9767	0.9945	0.5258	0.5820	0.5869
	Noise	0.4887	0.8002	0.9932	0.9945	0.1998	0.6068	0.6223
	Translation	0.8236	0.8236	0.9962	0.9981	0.1764	0.8236	0.8216
	Rotation	0.7145	0.8692	0.9629	0.9708	0.1308	0.7843	0.7686
	Scaling	0.9001	0.8293	0.9889	0.9959	0.1707	0.8632	0.8582
U-Net	Simple	1.0000	0.6196	0.9960	1.0000	0.3804	0.7651	0.7856
	Brightness	1.0000	0.6196	0.9960	1.0000	0.3804	0.7651	0.7856
	Contrast	1.0000	0.5206	0.9930	1.0000	0.4794	0.6848	0.7190
	Blur	1.0000	0.2229	0.9734	1.0000	0.7771	0.3645	0.4658
	Noise	0.5914	0.4808	0.9944	0.9978	0.5192	0.5304	0.5305
	Translation	1.0000	0.5942	0.9956	1.0000	0.4058	0.7455	0.7691
	Rotation	0.9014	0.5193	0.9583	0.9952	0.4807	0.6590	0.6661
	Scaling	1.0000	0.1799	0.9653	1.0000	0.8201	0.3050	0.4167
LinkNet	Simple	0.1229	0.6578	0.9476	0.9507	0.3422	0.2070	0.2692
	Brightness	0.1229	0.6578	0.9476	0.9507	0.3422	0.2070	0.2692
	Contrast	0.1412	0.5201	0.9468	0.9531	0.4799	0.2221	0.2516
	Blur	0.1382	0.2227	0.9260	0.9508	0.7773	0.1705	0.1382
	Noise	0.0473	0.2973	0.9560	0.9604	0.7027	0.0817	0.1047
	Translation	0.1229	0.6309	0.9473	0.9507	0.3691	0.2057	0.2627
	Rotation	0.0114	0.0076	0.8720	0.9447	0.9924	0.0091	-0.0577
	Scaling	0.1386	0.1798	0.9180	0.9506	0.8202	0.1565	0.1152

Analysis of accuracy, f1-score, and MCC values on individual operations of copy-move forged regions is shown using graph in Figure 5.15.



1- No operation(simple), 2: Brightness enhancement, 3: Contrast Stretching, 4: Blurring of region, 5: Addition of gaussian noise, 6: Translation of region, 7: Rotation of region, 8: Scaling of region, P: Proposed results, U: Unet results, L: Linknet results

Figure 5.15: The compared average result (accuracy, F1-score, and MCC value) of the proposed model with other state-of-the-arts on individual operations of copy-move forged documents

In the case of copy-move forgery where the copied region is from its document itself, very tough to detect the forged region since the feature of copied regions is very similar to the features of the document from where it is copied. Also, the regions may vary in size, sometimes only a group of characters may copy the documents. In such cases, the extraction of features is very tough. From the graph of MCC value and f1-score, it can be concluded that if the copied region is rotated, scaled, or somehow gets blur it will be easily detected. But the case of translation, brightness enhancement, and contrast stretching will reduce the probability of detection of the forged region. On the other hand, U-Net is taking the only advantage of border pixels and the smaller number of positive pixels which leads the results well. But overall, the result of the proposed model is better than others.

Table 5.13: Comparison of Average result of spliced forged documents for the proposed and other standard models

Model	*p*	*r*	*a*	*s*	*m*	*f1*	*mcc*
Proposed	0.8209	0.9540	0.9853	0.9873	0.0460	0.8805	0.8766
Unet	0.9141	0.5198	0.9691	0.9969	0.4802	0.6409	0.6642
Linknet	0.1694	0.1797	0.9085	0.9525	0.8203	0.1715	0.1255

Spliced forgery is a type of forgery where the region of a document is copied and pasted to another document. In this case, the features of the forged region are very different from the feature of the document where the forged region is pasted. Since these features are different, detection of the forged region is easier than detection in copy-move.

The average result of spliced forged region detection in forged test cases of the FD3 dataset is given in Table 5.13. In this table, average performance measures for 159 spliced test cases are calculated of proposed, UNet, and LinkNet models. From Table 5.13, it can be concluded that in the case of the spliced forged document where features of the spliced region are completely different from the feature of document models give a better result. Although the precision of UNet is better than the proposed one, other performance measures are much greater than the U-Net model. If the miss rate of U-Net is compared with the proposed one, the difference between them is very high. Also, the f1-score and MCC values of the proposed method are much larger than others. The visual result from one of the spliced forged documents on different models can be seen in Figure 5.16.

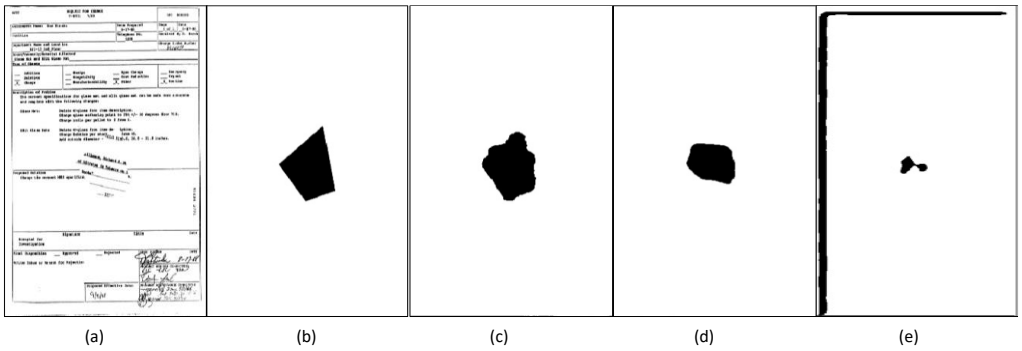


Figure 5.16: The visual result of the spliced test data from the constructed dataset FD3 (a) Tampered document (b) Ground Truth Mask (c) Result given by the proposed model (d) Result given by U-net (e) Result given by Linknet

The visual result in the figure shows that if the cases are from spliced forged documents, all the three models can detect the forged region however the detection of the forged regions by models is smaller than the proposed one where LinkNet doesn't give even satisfactory results. In the result of U-Net given in Table 5.14, the precision value of maximum test cases in the spliced forged document is higher than the proposed. The reason is obvious as already mentioned earlier, in such cases where the number of false positives is very less and true positives may be small, the precision value will be higher. But this value may mislead the result because it doesn't mean that model works well. It

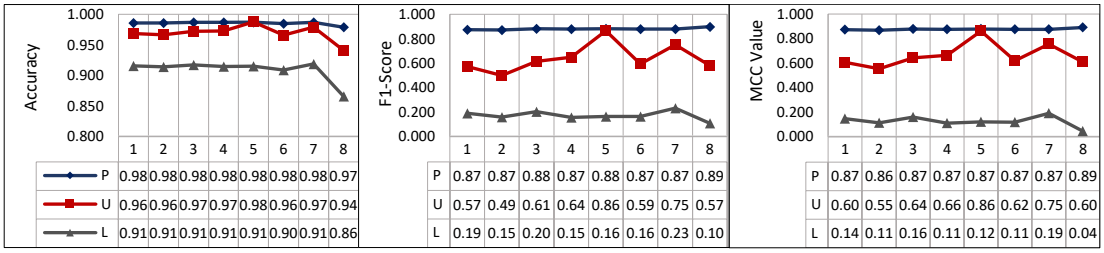
may be possible that the number of actual positives may be much higher than the predicted positives. Except for the precision, other performance measures are much higher in the proposed method than other state-of-the-art methods. The Miss-rate of the proposed method is much lesser than others. The performance of the model depends on f1-score and MCC value which are the best performance measure in all cases, here these values of the proposed model is higher than others. Analysis of individual operations of spliced forged regions is shown using graph in Figure 5.17.

Table 5.14: Average Quantitative values of performance measure of Spliced Documents for the proposed and other standard models

	Operations	p	r	a	s	m	f1	mcc
Proposed	Simple	0.7681	0.982	0.9859	0.986	0.018	0.862	0.8618
	Brightness	0.9139	0.9757	0.9948	0.9957	0.0243	0.9438	0.9416
	Contrast	0.8974	0.9198	0.9917	0.9951	0.0802	0.9085	0.9042
	Blur	0.8009	0.8882	0.9803	0.9861	0.1118	0.8423	0.8331
	Noise	0.7793	0.8648	0.9775	0.9846	0.1352	0.8198	0.8091
	Translation	0.7471	0.7849	0.9715	0.9833	0.2151	0.7655	0.7506
	Rotation	0.917	0.9051	0.9921	0.9961	0.0949	0.911	0.9069
	Scaling	0.7358	0.9835	0.9518	0.9471	0.0165	0.8418	0.8261
U-Net	Simple	0.9724	0.5689	0.9799	0.9992	0.4311	0.7179	0.7356
	Brightness	0.9137	0.5262	0.9765	0.9977	0.4738	0.6678	0.6835
	Contrast	0.8186	0.4620	0.9712	0.9952	0.5380	0.5907	0.6024
	Blur	0.9566	0.3186	0.9588	0.9991	0.6814	0.4780	0.5392
	Noise	0.8533	0.7836	0.9792	0.9915	0.2164	0.8170	0.8068
	Translation	0.6543	0.3113	0.9495	0.9896	0.6887	0.4219	0.4293
	Rotation	0.9212	0.6354	0.9812	0.9974	0.3646	0.7521	0.7565
	Scaling	0.7611	0.3588	0.9018	0.9831	0.6412	0.4877	0.4794
LinkNet	Simple	0.1811	0.2327	0.9182	0.9505	0.7673	0.2037	0.1627
	Brightness	0.1509	0.1740	0.9189	0.9539	0.8260	0.1616	0.1196
	Contrast	0.1742	0.2210	0.9180	0.9507	0.7790	0.1948	0.1534
	Blur	0.1521	0.1320	0.9050	0.9537	0.8680	0.1413	0.0916
	Noise	0.1838	0.1317	0.9139	0.9632	0.8683	0.1535	0.1111
	Translation	0.1748	0.1706	0.9032	0.9493	0.8294	0.1727	0.1213
	Rotation	0.1932	0.2507	0.9193	0.9507	0.7493	0.2182	0.1781
	Scaling	0.2373	0.2306	0.9111	0.9537	0.7694	0.2339	0.1867

In the case of spliced forgery where the copied region is from another document, the forged region is easier to detect rather than the copy-move forgery. From the graph, it can be concluded that whatever the operation is performed on the forged region of spliced forgery the model will detect it. On the other hand, UNet detects accurately the spliced region if Gaussian noise is added over the forged region or region. Sometimes

UNet detects the forged region if the region is rotated or simply copied to document without any transformation or operations. From these results, an average result is calculated which concludes the overall performance of the proposed method.



1- No operation(simple), 2: Brightness enhancement, 3: Contrast Stretching, 4: Blurring of region, 5: Addition of gaussian noise, 6: Translation of region, 7: Rotation of region, 8: Scaling of region, P: Proposed results, U: Unet results, L: Linknet results

Figure 5.17: The compared average result (accuracy, F1-score and MCC value) of the proposed model with other state-of-the-arts on individual operations of spliced forged documents

Table 5.15 gives the overall performance of the proposed and compared method. In this table, we can see that the f1-score and MCC values of the proposed method are much better than others. Except these, time and storage are also calculated for these methods.

Table 5.15: Comparison of overall performance on forged document dataset for the proposed and other standard models

Model	p	r	a	s	m	f1	mcc
Proposed	0.7302	0.6925	0.9739	0.9896	0.3075	0.6752	0.6805
Unet	0.9396	0.7593	0.8237	0.7118	0.4779	0.3204	0.3321
Linknet	0.1381	0.2418	0.9128	0.9513	0.7582	0.1556	0.1282

Table 5.16 compares the required time to perform the analysis of the image and required memory to store the model. The model takes this time to test the document whether is forged or not and also where the forgery has been done if the document is forged. Another measure is the required space to store the trained weight of the model. According to this table, the required time is taken by the proposed model more than two seconds than the UNet model which is much greater. But this is the case where time is not important but the detection of the forged document as well as forged region is important. Also, the time taken by the model is not much higher and considerable.

Table 5.16: Comparison of time and memory for the proposed and other standard models

Model	Time (per image)	Memory
Proposed	3.38 Sec	40 Mb
Unet	1.29 Sec	21.5 Mb
Linknet	1.64 Sec	100.3 Mb

5.3.2.3.3 Discussion

After training the model on training data, the trained model is tested on test data of FD3 as well as on publicly available data. The result is analyzed on both image-level and pixel-level. Here image-level analysis defines how many test cases are detected as forged and pixel-level analysis defines that in detected forged test cases, how many pixels are detected as the forged pixel. In this way, we can say that image-level analysis defines the detection of the forged image and pixel-level analysis defines the localization of the tampered region in a forged image. A confusion matrix is used for both analyses. Based on the confusion matrix- precision, recall, accuracy, miss-rate, F1-score and mcc values are calculated. Since precision, recall and accuracy mislead the result in the case of the imbalanced dataset and imbalanced pixels, it is highly recommended that evaluation should be done on F1-score and mcc values. In the case of image-level analysis, MCC value given by LinkNet is negative which means that the multiplication of false positives and false negatives is more than the multiplication of true positives and true negatives. MCC is a very important evaluation measure when the difference between the size of classes is very huge. Although accuracy is known as better performance measure the case when the number of true negatives is very high and true positive is zero then accuracy will be very high. However, the LinkNet model isn't able to predict positive documents. In the case of pixel-level analysis precision and accuracy values given by the U-Net model is higher than the proposed. Again, the reason is the same, the imbalanced number of positive and negative pixels. From the graph of MCC value and f1-score in Figures 5.15 and 5.17, it can be concluded that if the copied region is rotated, scaled or somehow gets

blur it will be easily detected. But the case of translation, brightness enhancement and contrast stretching will reduce the probability of detection of the forged region. On the other hand, U-Net is taking the only advantage of border pixels and the smaller number of positive pixels which leads the results good. But overall, the result of the proposed model is better than others. From Table 5.14, it can be concluded that in the case of the spliced forged document where features of the spliced region are completely different from the feature of document models give a better result. Although the precision of the U-Net is better than the proposed one, other performance measures are much greater than the U-Net model. If the miss rate of U-Net is compared with the proposed one, the difference between them is very high. Also, the f1-score and MCC values of the proposed method are much larger than others. The performance of the model on spliced forged documents for individual cases of operation on different models can be seen in Table 5.14. The table shows that if the cases are from spliced forged documents, all the three models can detect the forged region however the detection of the forged regions by models is smaller than the proposed one. Where LinkNet doesn't give even satisfactory results.

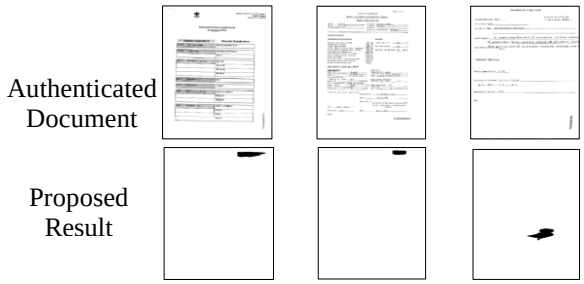


Figure 5.18: Visual demonstration of misclassified results by the proposed method

After all the result analysis it is also necessary to demonstrate the failure cases given by the proposed model. The proposed model correctly classifies 447 documents out of 462 documents whereas 15 documents are not identified correctly. Out of incorrectly identified documents ten documents are predicted as forged while they are authentic

documents and five are predicted as authentic documents while they are forged. So, the probability of predicting authenticity as forged is larger than predicting forged as authentic. Since it is less harmful to detect authenticity as forged rather than predicting forged documents as authentic. Figure 5.18 demonstrates three failure cases where the proposed model detects authentic documents as forged. The tampered region given by the model is also shown in Figure 5.18. The percentage of misclassification of the proposed model is 3.47% which is quite low.

5.4 Summary

Blind forgery detection is a challenging task in the case of natural images as well as digitally scanned documents. Two different methods are proposed in this chapter for natural images and digitally scanned documents.

Challenges of the U-Net deep learning CNN model in case of detection of the forged location have been tried to overcome in the given solution by modifying it. The proposed deep learning model has been trained and tested on five different publicly available datasets. To test the diversity of the proposed model, different images having dissimilar characteristics except these public datasets are passed through the trained model. All these images have been successfully passed through the model and gave better performance results than the compared state-of-the-art techniques except the manipulated scanned reports. Though this is the only limitation of the proposed model, none of the implemented previous segmentation models gives better performance results. This can be concluded that the proposed method works better in all cases except the scanned reports.

A dataset is constructed for forged scanned documents and a deep learning CNN model is developed to detect tampered regions in forged documents. The motivation behind the dataset was to provide an opportunity to academic scholars in the field of

forged document detection. This dataset contains different types of forgery in documents and eight different types of operations performed on forged regions. This shows the diverse nature of the dataset. The major motivation behind developing a model for forged document detection is the importance of digital documents in the era of the digital transaction (i.e. digital documents are used almost everywhere) and the pressing need for a digital forgery detection model based on scanned digital documents. With regards to the challenges present in existing forgery detection models for images the developed model consists of an Inception block in the absence of dimensionality reduction, with scanned documents as an input. The experiments performed and demonstrated some interesting results. It opens up several directions for academic scholars to explore in the field of scanned documents forensics. The proposed model is analyzed on a publicly available dataset with constructed FD3 dataset and compared with the state-of-the-art methods. In this depth analysis first, image-level classification of forged and authentic documents was done. Then the pixel-level analysis of both types of forgeries was done in the forged documents. This analysis was figured out on all types of operations and transformations. The compared result shows that the performance of the proposed method is better than state-of-the-art