JOURNAL OF OPTOELECTRONICS LASER ISSN: 1005-0086
Volume 41 Issue 9, 2022

FAKE CURRENCY DETECTION SYSTEM USING CNN-VG16

J Sirisha Devi[1], Prasanthi Gottumukkala[2], Sukanya Ledalla[3], T.N.P Madhuri[4], Pavithra Avvari[5]

[1]Associate Professor, Department of Computer Science and Engineering, Institute of Aeronautical Engineering, Hyderabad

[2,3,4,5]Assistant Professor, Department of Information Technology, Gokaraju Rangaraju Institute of Engineering and Technology, JNTUH, Telangana, India

ABSTRACT

With the advancement of color printing technology, fake currency note printing and duplicating the notes have been increasing rapidly on a vast scale. A few years ago, the printing was done in a print house, but now anybody with a simple laser can print a currency note with maximum accuracy. This may result in hyperinflation and may affect the citizens and the government mainly if not controlled at the earliest. So, we have come up with a system that detects fake currency in less time and efficiency. We use Vgg-16 (CNN) architecture. This model is trained to detect the presence of watermark and uv strips. After checking the note's authenticity, it even displays the accuracy percentage. The data sets used here are created on our own by gathering the pics of fake and real currencies. We use concepts of image processing, tensor flow to read the image and make the machine learn. This system gives high-performance speed and works efficiently. The performance of a model is based upon the metrics such as accuracy, precision etc. This project is mainly used for detection of fake currency.

Keywords: Artificial Intelligence, Computer Vision, Deep learning, Convolutional Neural Network, Image Detection.

1. INTRODUCTION

Color printing technology has advanced rapidly, as has the fabrication of false currency notes and the duplication of notes on a large scale. Previously, printing was done in a print shop, but today, anyone with a low-cost laser printer can accurately produce a currency note. As a result, the use of counterfeit notes in place of genuine notes is on the rise. India has long struggled with serious issues such as corruption, dark money, and now counterfeit currency. If not managed quickly, this could lead to hyperinflation, which would primarily affect citizens and the government. Machine learning algorithms have been rapidly developing at visual pattern recognition tasks in recent years, which has demonstrated promising outcomes in aiding the numerous technical sectors. The accuracy and dependability of spotting counterfeit money have increased with the steady advancement of imaging methods. Recent breakthroughs in machine learning have demonstrated that utilizing algorithms, automatic image categorization can detect counterfeit cash. Deep learning, in particular, has shown promise in automated categorization. CNNs were created primarily to speed up the processing of images and to do image classification [9]. To categorize scanned images according to their features, we created a CNN model. This will help reduce the time in the fake currency detection. And it could be helpful as the time taken to detect fake currency using the algorithms is very less compared to the

 ISSN: 1005-0086
Volume 41 Issue 9, 2022

existing approaches. This system gives high-performance speed and works efficiently. The two primary techniques for identifying currency are by distinctive geometric size [1] and distinctive texture [2].

The objective of our paper is to: - predicting the authenticity of Indian Currency - put an end to the prevailing counterfeit currency crisis - find a faster and innovative way to detect fake currency using algorithms

1.1 RELATED WORK

It became crucial to research different algorithms and already implemented works because there are numerous approaches to construct a system that identifies currency pictures in order to get a greater comprehension of the goals and results of the suggested method [3].

Computer vision [4] blends cameras, edge or cloud-based computing, and artificial intelligence to allow computers to "see" and recognize objects (AI) [13]. In addition to general-purpose CPUs and accelerated visual processing units (VPUs), Intel provides a wide range of AI-enabling technologies. Radial Basis Function Network model [6,7] developed and tested on Saudi Arabian documents a portable paper cash identification technique. Based on the properties of the images and correlations between the images, the currencies were identified here. It makes use of the Radial Basis Function Network for categorization. For normal, non-tilted images, the scheme is 95.37 percent accurate; for noisy, non-tilted images, it is 91.65 percent correct; and for tilted images, it is 87.5 percent accurate. An article, "ANN based currency detection system employing compressed grayscale and application for SriLankan currency notes" [8].

If properly taught, computer vision models can detect and recognize people, as well as perceive musical movement.

Most of the approaches include the following currency features to check the authenticity of the note: Identification mark: A special feature in intaglio [5] has been introduced on the left of the watermark window on all notes except Rs.10/- note. This feature is in different shapes for various denominations (Rs. 2000-rectangle, Rs.500-circle, Rs.100-triangle) and helps the visually impaired to identify the denomination. Bleed lines: Blind persons can recognize the denomination by touching a mark with intaglio print. In the 500 denomination the mark is of five lines while in the 2000 denomination the mark is of seven lines.

BLEED LINES —

←IDENTIFICATION MARK

DOI: 10.5281/zenodo.7119088

1. As there is no identification mark for rs. 10/- note, we cannot distinguish between fake and real currency.
2. Bleed lines are fixed in rs. 100/-, rs. 200/-, rs. 500/-, rs. 2000/- notes, so it is easy to replicate those bleed lines in fake currency.

1.2 ARCHITECTURE OF THE PROPOSED MODEL

We use Vgg16 (CNN) architecture to determine whether an Indian currency note is fake or real. We identify the note based on two security features:

❑Watermark identification model: Identifies whether the note contains the Gandhi watermark under backlight illumination from a backlight image of the note.

❑Ultraviolet strip detection model: Identifies whether the note contains a fluorescent strip or not and if yes then whether the strip is continuous or dashed under ultraviolet illumination from an ultraviolet image of the note.

The architecture of our model is shown below. the foundation for subsystem control and communication, and for identifying the subsystems that comprise the system. The overall structure of the software system is to be established by the architectural design.

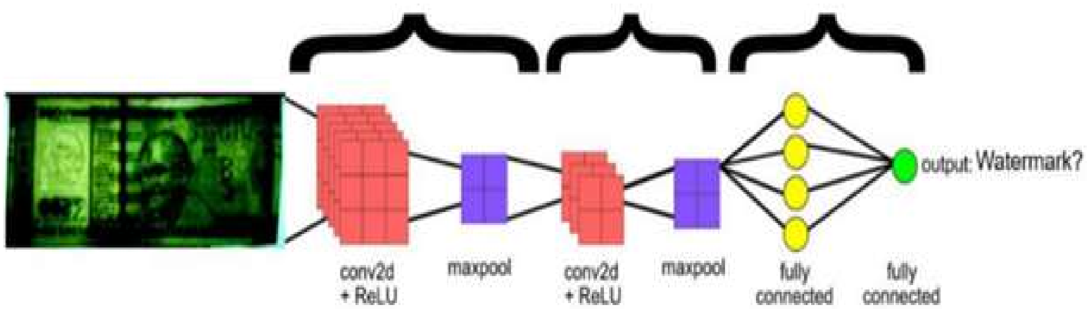
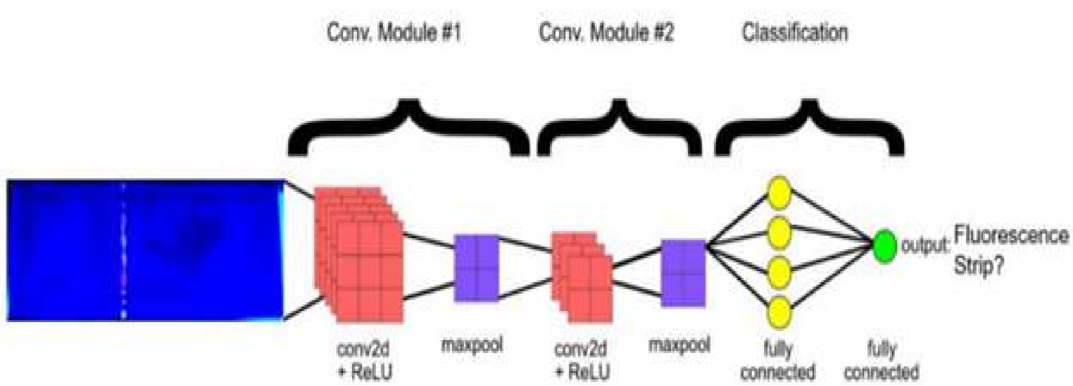


Fig 1: Architecture Diagram of Watermark Detection Model



3.1 CONVOLUTIONAL NEURAL NETWORKS

Deep learning models are a type of computing system that can learn a hierarchy of features by constructing high-level qualities from low-level ones, allowing feature development to be automated. The well-known convolutional neural network is one of these models (CNN) [11]. CNN is a machine learning technique that consists of several(deep) layers of processing using learnable operators (both linear and nonlinear), allowing it to learn and create high-level knowledge from low-level features in an automatic manner. Convolutional neural networks [12] use little preparation in comparison to other image classification techniques, which allows the community to learn to construct filters on many platforms. An instance of the changes that a convolutional action feature makes is comparable to combining two functions by one. CNN uses this principle in decoding images. CNN analyses the visible aspects of a photograph and tries to become aware of the prevalence of matching features. It contains 4 main layers as shown in Fig 2:

1. Convolution Layer
2. Max Pooling
3. Fully Connected Layer
4. Dense Layer

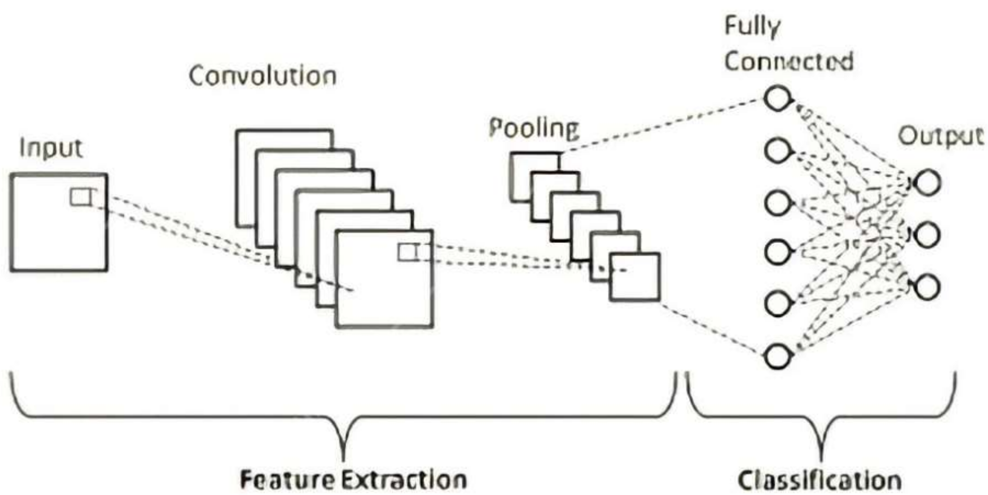


Fig 2: Architecture of Convolutional Neural Network

3.2 CONVOLUTION LAYER

A convolutional layer is the basic component of a CNN. It has a number of filters (or kernels) whose settings need to be learned throughout the training process. Often, the filters are smaller than the original image. Each filter converges with the image shown in fig 3 to create an activation map.

DOI: 10.5281/zenodo.7119088

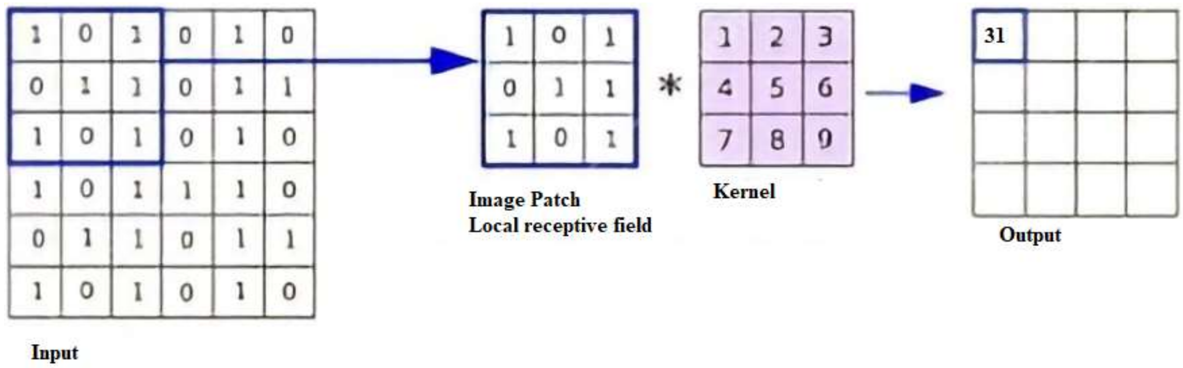


Fig 3: Convolve Operation

3.3 MAX POOLING LAYER

Max pooling is a type of pooling that chooses the most elements possible from the feature map area that the filter has covered. The output of the max-pooling layer would then be a feature map that included the standout elements from the prior feature map, as seen in Fig 4.

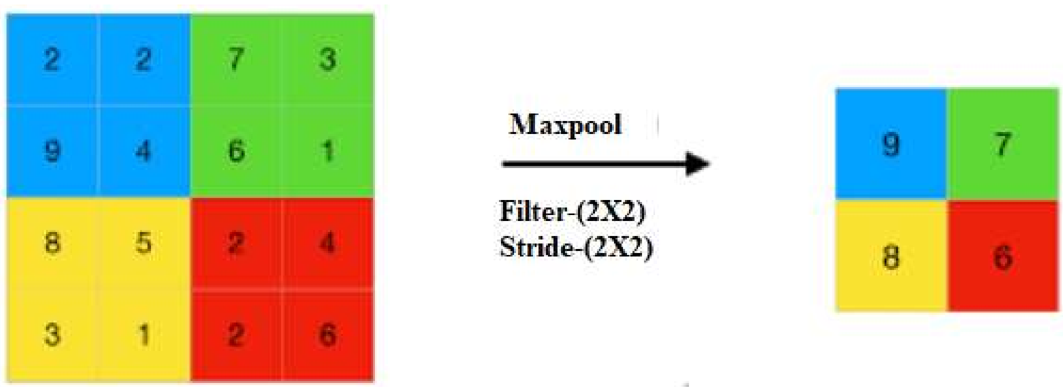


Fig 4: Max Pooling operation

3.4 FULLY CONNECTED LAYER

The Fully Connected Layer uses feed-forward neural networks as its main computing technique. The last layers of the network are Fully Connected Layers. The input to the fully connected layer is the output from the last pooling or convolutional layer, which is flattened and then fed into the layer, as seen in Fig 5.

JOURNAL OF OPTOELECTRONICS LASER ISSN: 1005-0086
Volume 41 Issue 9, 2022

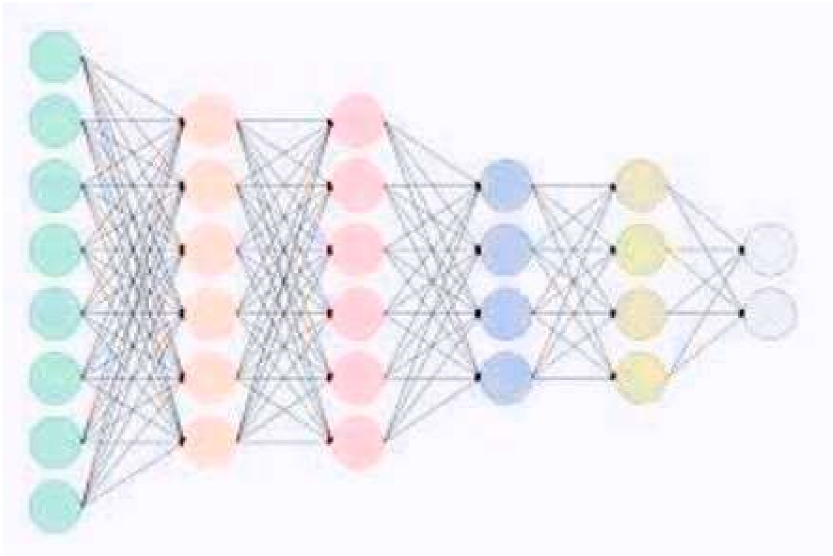


Fig 5: Fully Connected Layer

3.5 DENSE LAYER

Each neuron in a dense layer receives one output from the preceding layer, which is then passed on to all the neurons in the following layer. This is the most fundamental layer in neural networks.

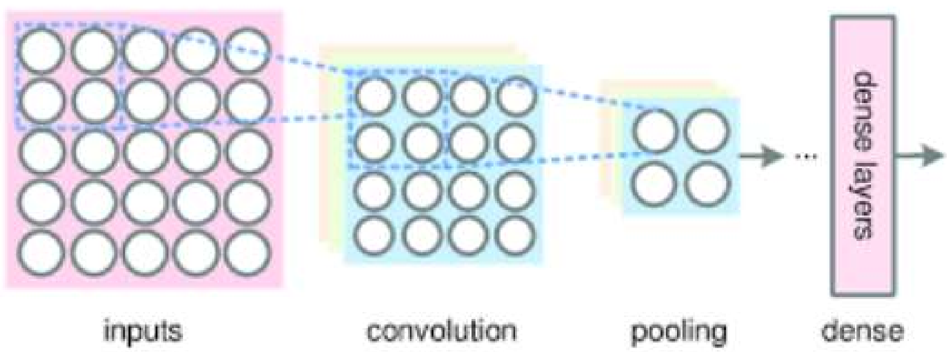


Fig 6: Architecture Dense Layer

3.6 CNN ARCHITECTURE: VGG-16

Karen Simonyan and Andrew Zisserman suggested a CNN architecture dubbed VGG-16 [10], which we used to train our models. The Keras module is used in this application. For every categorization, it provides sufficient accuracy over the network. It is made up of 16 layers.

DOI: 10.5281/zenodo.7119088

JOURNAL OF OPTOELECTRONICS LASER ISSN: 1005-0086
Volume 41 Issue 9, 2022

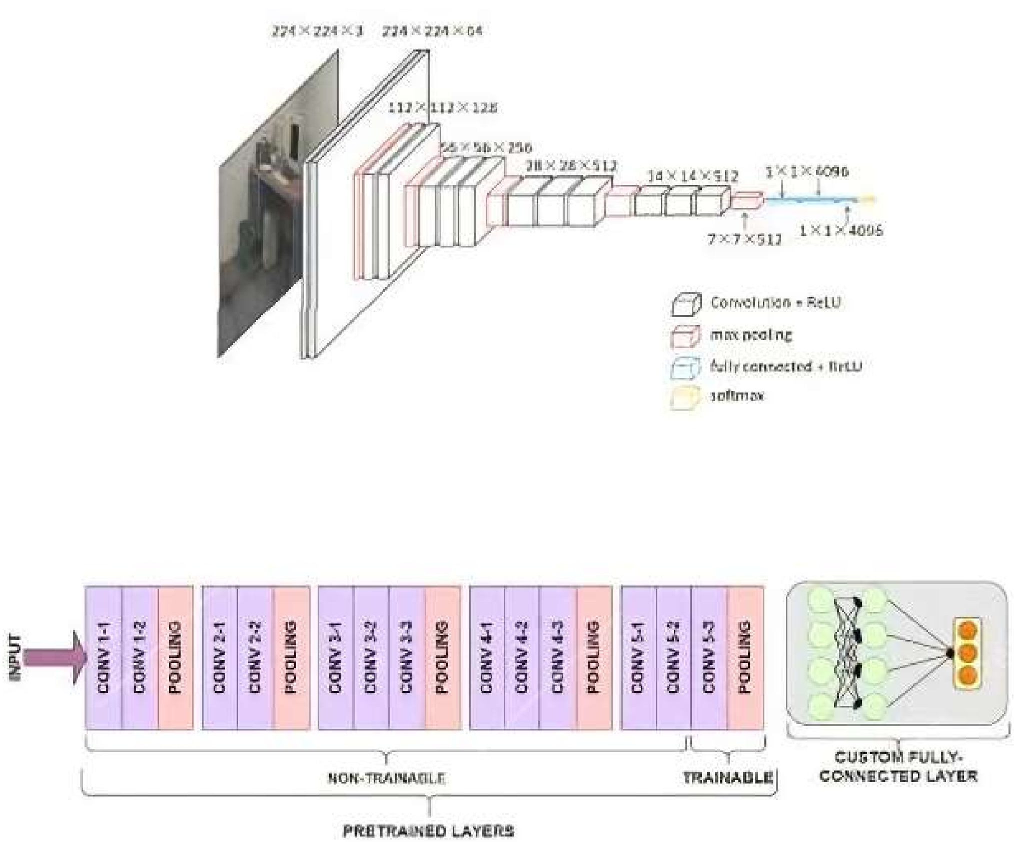


Fig 7: Architecture of CNN-VGG-16

4. METHODOLOGY

Implementation The implementation of our research contains six phases described as follows.

1. Preparation of custom datasets for watermark detection and UV strip detection models separately.
2. Train-Test split
3. Building model architectures
4. Training the models
5. Saving the models
6. Testing the models, the implementation of the work is finished with the assistance of the python language. Particularly, for computer vision, Anaconda's spyder is getting used.

DataSet Preparation for any computer vision application, the essential part is the image dataset. The model is trained on these images. Our research revolves around the hidden features of currency notes, for which have collected datasets from kaggle to use in the research. We even came up with a python script that uses the manually clicked sample images of the currency notes and creates a dataset for training and testing purposes. The dataset consists of 9000 images.

JOURNAL OF OPTOELECTRONICS LASER ISSN: 1005-0086
Volume 41 Issue 9, 2022

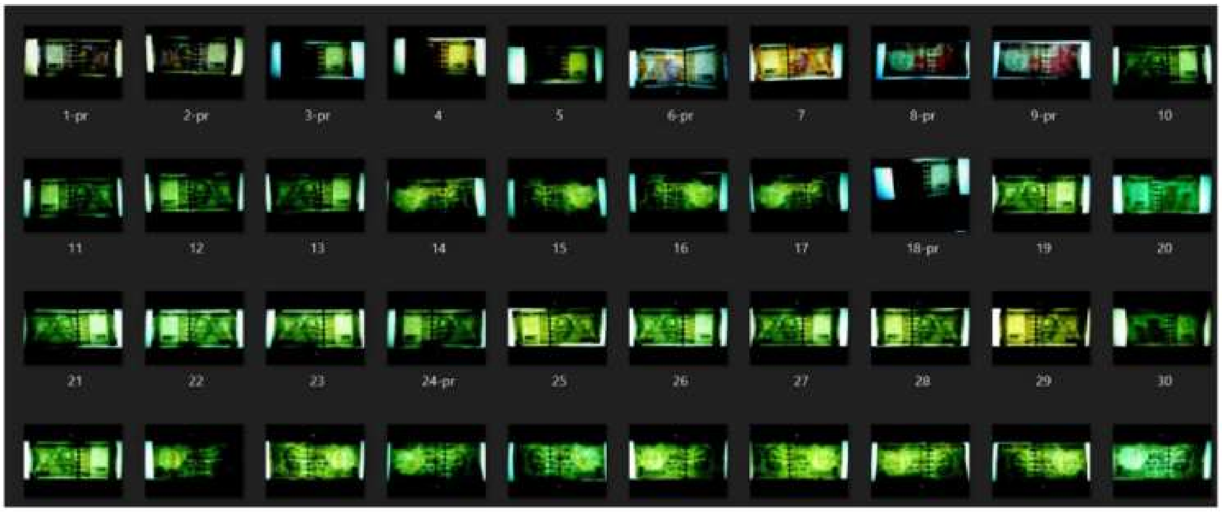


Fig 8: WATERMARK DATASETS

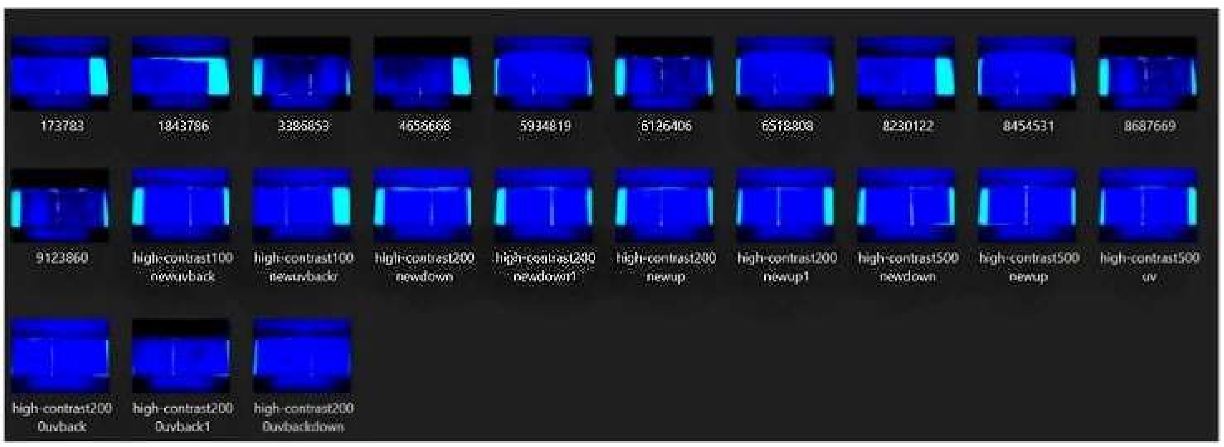


Fig 9: UV STRIPS DATASETS

4.1 TRAIN-TEST SPLIT

We load the dataset inside our python code and split it into a training set and a test set with split percentages of 67% and 33%, respectively. This splitting is done for training and testing purposes. Each training set contains 6000 images whereas the test set consists of 3000 images. The training set is employed to fit the machine learning model, whereas the test set is employed to check the model's working. Both the Watermark Detection Model and UV Strip Detection Model have their own training and test datasets.

4.2 BUILDING MODEL ARCHITECTURE

we used the transfer learning method to train the models, employing VGG16 as the basis model. Transfer learning is a type of machine learning. In this strategy, a model created for one work is reused as the starting point for a model on a different task. Given the large computing and time resources required to create neural network models on these issues, as well as the massive jumps in skill that they provide on related problems, it is a recommended technique in deep learning to employ pre-trained models as the starting point on computer vision tasks. A convolutional neural network, or VGG-16, is a type of neural network. It has a total of 16 layers. The models load a set of weights that have been pre-trained on the ImageNet dataset to make the training process quicker and to extract and map features more correctly. The ImageNet dataset is a massive collection of annotated pictures created by academics for the purpose of developing computer vision algorithms.

4.3 TRAINING THE MODEL

Giving an algorithm a training dataset to learn from is the first step in training an ML model. It is made up of sample output data as well as the equivalent sets of input data that influence the output. To connect the processed output with the sample output, the training model is used to run the input data through the algorithm. The model is modified based on the results of this association. "Model fitting" is the term for this iterative procedure. The precision of the model is dependent on the correctness of the training set or test dataset. The accuracy of these models was initially 90%. We were able to attain a 99 percent accuracy after adding the hyper parameters epoch and batch size to the classifier. It took around 10-12 hours to train both models.

4.4 SAVING THE MODEL

In order to reuse the trained models for model comparisons and testing on new data, we save them in files with the.h5 extension and then restore them. The method Deserialization is the process of recovering data after it has been saved in a serialized format.

5. TESTING THE MODEL

After the models are trained according to the training dataset, it is also tested using the same dataset. The purpose of this process is to determine how much data has been learned by the model.

5.1 TESTING

There has been a lot of competition between software companies since the advent of technology. The company must outperform the competitors on a consistent basis. To outperform the competition, you must maintain a high level of product/service quality throughout the day. Any user will only utilize an application if it is of good quality and free of errors. The developers are responsible for conducting various types of testing on the application in order to ensure that it is error-free and efficient. Software testing is an important aspect of the software development lifecycle since it enhances the application's performance and consistency. We have categorized 2 test cases for the Watermark Detection model, which are as follows: 1. Presence of watermark 2. No watermark The UV Strip Detection model has the following test cases: 1. Continuous fluorescence strip 2. Dashed fluorescence strip 3. No fluorescence strip

5.2 PRESENCE OF WATERMARK

This is one of the prominent test cases where the user will know the currency note is authentic after giving the backlit image of the note as input. The system will detect the hidden watermark and give

DOI: 10.5281/zenodo.7119088

the output as “yes watermark” and the percentage of the particular class detected as shown in fig 10.

Fig 10: Presence of Water Mark

6. RESULTS

The research Workflow Brief and Output are described in the following graphics. Figure 12, 13describes workflow Figure 14, 17, 18 describes the results.

```
yes_watermark: 99.99812841415405%
no_watermark: 0.0018700797227211297%
Final Result:  {'yes_watermark': 99.99812841415405}
```

Fig 11: Results

No Watermark A counterfeit currency note does not have the watermark printed on it because of its hidden nature. Fake currency printing machines fail to include such hidden features. Hence, upon detecting a fake note, the model prints the output as “No watermark”.

DOI: 10.5281/zenodo.7119088

 ISSN: 1005-0086
Volume 41 Issue 9, 2022

```
no_watermark: 99.9998927116394%
yes_watermark: 0.00010212196457359823%
Final Result:  {'no_watermark': 99.9998927116394}
```

Fig 12: Results

Continuous Fluorescence Strip Under ultraviolet illumination, an authentic currency note appears to have a continuous fluorescence strip. It is invisible to nakes because the strip is printed using fluorescence ink.

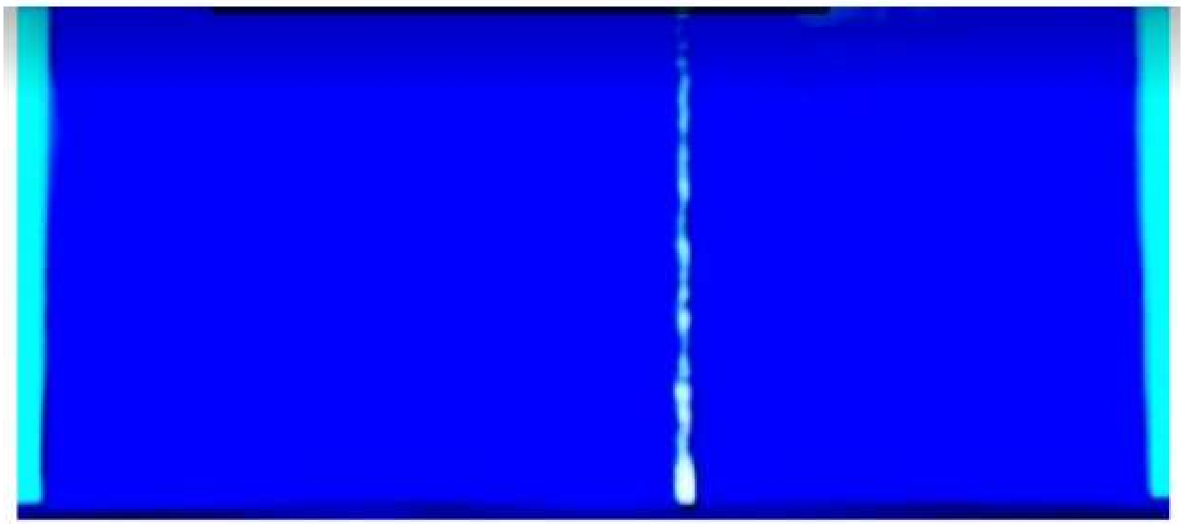


Fig 13: Fluorescence Strip

```
continuous:  100.0%
dashed:  2.6906688646429152e-17%
nopatch:  1.1703062781912474e-19%
Final Result:   {'continuous':  100.0}
```

Fig 14: Results

6.1 DASHED FLUORESCENCE STRIP

A dashed fluorescence strip under ultraviolet light also indicates an original currency note as shown in fig 14.

DOI: 10.5281/zenodo.7119088

 ISSN: 1005-0086

Volume 41 Issue 9, 2022

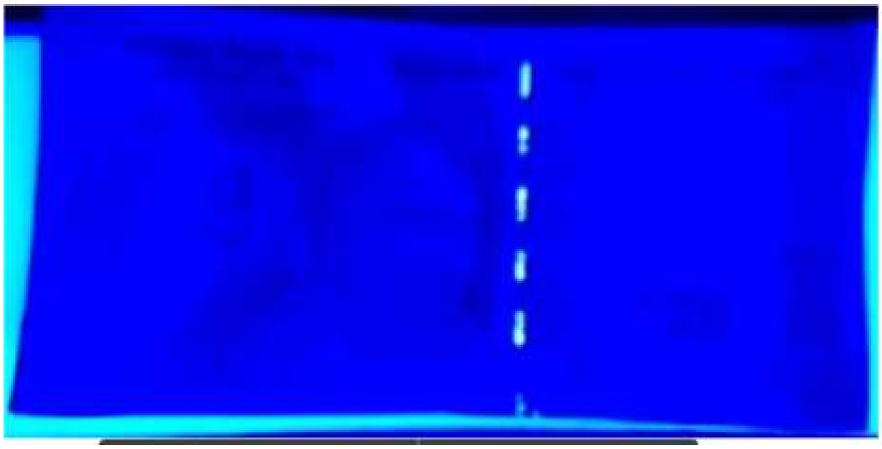


Fig 14: Dashed fluorescence strip under ultraviolet light

```
dashed: 100.0%
continuous: 1.699049916193174e-17%
nopatch: 1.1199574422040212e-17%
Final Result:  {'dashed': 100.0}
```

Fig 15: Results

6.2 NO FLUORESCENCE STRIP

There is an absence of the fluorescence strip on counterfeit currency notes shown in fig 16. The system looks for the strip, and on not detecting it, it gives the output as “no patch”.

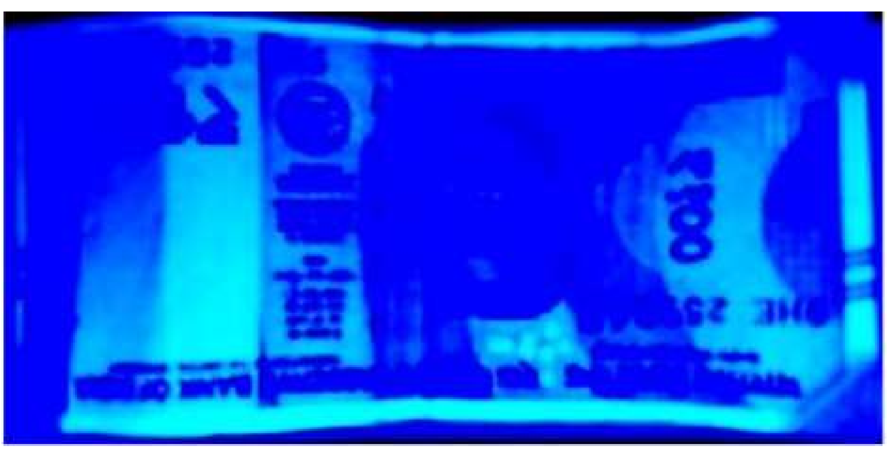


Fig 16: Currency note without the fluorescence strip

DOI: 10.5281/zenodo.7119088

```
nopatch: 100.0%
continuous: 4.241765637559675e-15%
dashed: 1.7688286431579027e-18%
Final Result:  {'nopatch': 100.0}
```

Fig 17: Results

```
In [17]: plt.plot(hist1.history["accuracy"])
         plt.plot(hist1.history['val_accuracy'])
         plt.plot(hist1.history['loss'])
         plt.plot(hist1.history['val_loss'])
         plt.title("model accuracy")
         plt.ylabel("Accuracy")
         plt.xlabel("Epoch")
         plt.legend(["Accuracy","Validation Accuracy","loss","Validation Loss"])
         plt.savefig('mobilenet' + '_plot.png')
         plt.show()
```

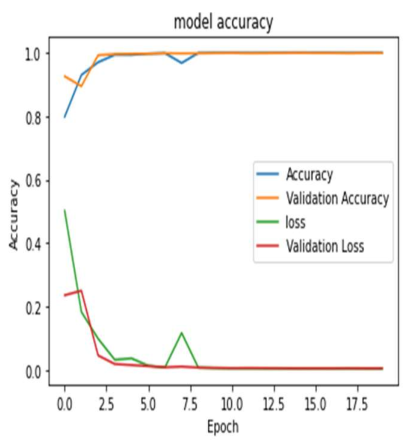


Fig 18: Accuracy of Model

7. CONCLUSION

Conclusion We all know that phony cash is manufactured without the government's legal authority. As a result, delivering or using counterfeit cash is a scam. Falsifying concerns, on the other hand, have been increasingly real in recent years as a result of huge developments in shading, printing, copying, and scrutinizing. As a result, the challenge of accurately distinguishing fake banknotes from genuine ones using programmed fake currency detection systems has become increasingly important. The validation of Indian paper currency is presented in this currency recognition system using open cv approaches in the Python programming language. To discover the hidden security aspects of currency notes and categorize them as fake or genuine, our project uses the VGG16 Architecture in conjunction with Convolutional Neural Networks. Two VGG16 trained models make up the project. Everyone will benefit from the design because it is both efficient and practical. So far, we appear to have accomplished our goals and are pleased with the outcome of our application.

DOI: 10.5281/zenodo.7119088

8. FUTURE ENHANCEMENTS

Future work could include coins in the dataset to make the system fully functional for both coins and notes. We intend to include a module for currency conversion in the future. To further improve, we can develop masks that exclusively remove key components or traits, such as identifying the note's denomination and its unique watermarks to locate the right match. We can make this system accessible to all country residents.

9. REFERENCES

1. Kalpana Gautam "Indian Currency Detection using Image Recognition Technique",2020 International Conference on Computer Science, Engineering and Applications (ICCSEA).
2. Priyanka Dhapare, Akash Agarwal, Devangi Doshi "Detection of Counterfeit Currency using Image Processing Techniques",2019 IEEE International Conference for Convergence in Technology (I2CT).
3. Tao Zhang, Zhongnian Li, Qi Zhu, Daoqiang Zhang"Improved Procedures for Training Primal Wasserstein GANs", 2019 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation.
4. Ali Abd Almisreb,Mohd.A. Saleh," Transfer Learning Utilization for Banknote Recognition: a Comparative Study Based on Bosnian Currency", Southeast Europe Journal of Soft Computing,vol.. 8 no.1,pp.28-31,March 2018
5. https://www.rbi.org.in
6. P.Selvi Rajendran,"Virtual Bulletin Board using Man-Machine Interface (MMI) for Authorized Users", Indian Journal of Science and Technology, vol.12,no.34, September 2019.
7. Manish Shankar Kaushik, Aditya Bihar Kandali," Recognition of facial expressions extracting salient features using local binary patterns and histogram of oriented gradients", International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS), 2017.
8. P.R. Nagpure, T.Ghotkar, S. Shetty, Recognition and Fake Note Detection, International Journal of Innovative Research in Computer and Communication Engineering Vol 4 (March 2016).
9. Jiang Z-P, Liu Y-Y, Shao Z-E, Huang K-W. An Improved VGG16 Model for Pneumonia Image Classification. *Applied Sciences*. 2021; 11(23):11185. https://doi.org/10.3390/app112311185.
10. Liu, B., Zhang, X., Gao, Z., Chen, L. (2018). Weld Defect Images Classification with VGG16-Based Neural Network. In: Zhai, G., Zhou, J., Yang, X. (eds) Digital TV and Wireless Multimedia Communication. IFTC 2017. Communications in Computer and Information Science, vol 815. Springer, Singapore. https://doi.org/10.1007/978-981-10-8108-8_20.
11. Hameed, Z.; Zahia, S.; Garcia-Zapirain, B.; Javier Aguirre, J.; María Vanegas, A. Breast Cancer Histopathology Image Classification Using an Ensemble of Deep Learning Models. *Sensors* 2020, *20*, 4373. https://doi.org/10.3390/s20164373.
12. AriawanI. (2022). Classification of three genera of coral fish using convolutional neural network. *Jurnal Ilmu Dan Teknologi Kelautan Tropis*, *14*(2), 205-216. https://doi.org/10.29244/jitkt.v14i2.33633.

13. Urmila Samariya, Rakesh K. Sonker. (2022) Comparisons of Image Classification Using LBP with CNN and ANN. *Journal of Applied Mathematics and Computation*, **6**(**3**), 343-346. DOI: http://dx.doi.org/10.26855/jamc.2022.09.006.

14. Gudla Swathi, Kompella Venkata Ramana . (2022) Object Detection And Image Classification Using Deep Learning Model *Journal of Engineering Sciences* Vol 13 Issue 09,2022, ISSN:0377-9254.

15. Rathziel Roncancio, Aly El Gamal, Jay P. Gore*, Turbulent flame image classification using Convolutional Neural Networks, Energy and AI, Volume 10, 2022, 100193, ISSN 2666-5468, https://doi.org/10.1016/j.egyai.2022.100193.*