
ISSN: 2321-2152
IJMECE
International Journal of modern electronics and communication engineering
E-Mail
editor.ijmece@gmail.com
editor@ijmece.com
www.ijmece.com

ISSN2321-2152
www.ijmece .com
Vol 12, Issue 3, 2024

CNN AND HOG HYBRID FEATURE EXTRACTION FOR SIGNATURE VERIFICATION WITH MULTI-CLASSIFICATION

K BHASKAR[1], CHALAMANTOORU JAYA SANKAR[2], K PAVANI[3], TATIREDDY RAVI[4]
[1]Associate Professor, Department of MCA, Sri Venkatesa Perumal College of Engineering & Technology, Puttur, Email: bhaskark.mca@gmail.com
[2]P.G Scholar, Department of MCA, Sri Venkatesa Perumal College of Engineering & Technology, Puttur, Email: chalamantooru@gmail.com
[3]Assistant Professor, Department of CSE, Sri Venkatesa Perumal College of Engineering & Technology, Puttur, Email: kothapavani1990@gmail.com
[4]Assistant Professor, Department of CSE, Sri Venkatesa Perumal College of Engineering & Technology, Puttur, Email: tatireddyravi.ai@gmail.com

Abstract: The offline signature verification system's feature extraction stage is significant on the grounds that the amount and alignment of removed features decide how well these frameworks can recognize valid and counterfeit marks. We utilized a Convolutional Neural Network (CNN) and Histogram of Oriented Gradients (HOG) to separate elements from signature photographs, trailed by Decision Trees to decide basic qualities. At last, CNN and Hoard were joined together. The cross breed procedure was tried utilizing LSTM, SVM, and K-NN classifiers. The examinations showed that our model was productive and prescient, with CEDAR dataset accuracy. This exactness is significant since we tried refined faked marks, which are more diligently to detect than essential or inverse marks. Xception, Feature extraction (HOG-RFE), and Voting Classifier for Dataset examination were incorporated to the venture to further develop Marks Check Utilizing CNN and Hoard a Multi-Grouping Way to deal with 100 percent accuracy. An easy to understand Flask system with SQLite mix works on client testing enlistment and signin, making network protection applications usable.

Index terms *- Offline signature verification, CNN, HOG, deep learning.*

1. INTRODUCTION

Biometrics is the most fundamental innovation for recognizing individuals and deciding their power in light of conduct and physiological qualities. Ears, fingerprints, iris, and DNA are used for physiological recognizable pieces of proof, while articulation, voice, step, and mark are utilized for social IDs. One of the most broadly perceived biometric confirmation strategies is the written by hand signature [1]. Transcribed marks are exceptional social biometrics for banks, Mastercards, identifications, really look at handling, and monetary reports. Questionable marks are difficult to check. In this way, a framework that can recognize genuine and misleading marks is expected to diminish robbery and misrepresentation. From well-qualified assessment based check to ML calculations to DL calculations, disconnected signature confirmation frameworks have gone through many examinations in the beyond 30 years, however they actually need improvement [2].

Signature confirmation can be computerized online [3, 4, 5, 6, 7] or disconnected [8, 9, 10, 11, 12, 13]. Since offline signature pictures need attributes like pen-tip strain, speed, and speed increase, offline signature verification is more troublesome than online check [1, 2, 8, 10, 11]. The exceptional marking strategies make the web-based technique unacceptable in various situations.[61]

Signature verification is the most broadly acknowledged and least limit biometric technique in the public eye, however many examinations [12], [13], [14], [15] have shown that it is troublesome on the grounds that penmanship marks contain unique letters and images that are much of the time ambiguous and underwriter ways of behaving are unique. In this manner, center around developing a powerful signature framework in view of a genuine situation and look at the signature as one picture as opposed to letters or expressions.

2. LITERATURE SURVEY

Information secrecy and assurance against undesirable access rely upon the mark technique. Disconnected manually written signature research has been a conspicuous biometric validation approach in the earlier ten years [1]. Regardless of its pertinence, this approach is troublesome since nobody can sign a similar mark without fail. We're likewise intrigued by the dataset's elements that could influence the model's presentation, subsequently we extricated signature picture features utilizing histogram orientation gradient (HOG). We proposed a LSTM neural network model for signature confirmation utilizing USTig and CEDAR datasets in this review. We have an extraordinary estimating model: The characterization exactness proficiency LSTM for USTig was 92.4% in 1.67 seconds and 87.7% for CEDAR in 2.98 seconds. Our procedure beats disconnected signature check techniques like K-nearest neighbor (KNN), support vector machine (SVM), convolution neural network (CNN), speeded-up robust features (SURF), and Harris in accuracy [10,14].

Bank checks, declarations, contract structures, securities, and other conventional papers are hard to confirm precisely and powerfully. Validity really relies on how well the marks in the papers look like the approved individual's unique marks. Approved marks are known in advance. [2] another mark confirmation highlight set in light of semi straightness of limit pixel runs is introduced in this review. Utilizing basic blends of the directional codes from the mark line pixels, we separate semi straight line fragments and get the list of capabilities from different classes. The semi straight line portions consolidate straightness and small curves to give a strong mark check include set. Support Vector Machine (SVM) characterization has shown results on normal mark datasets like CEDAR and GPDS-100. Results show that the proposed procedure beats the cutting edge [20].

Fluffy demonstrating of structure and dynamic properties taken from online mark information is utilized to make a clever web-based signature check framework. Rather than separating highlights from a mark, it is partitioned at mathematical extrema and include removed and fluffy displayed. With fragment to-portion correspondence, dynamic worldly twisting adjusts the two examples to a negligible distance. [3,29] Next, fluffy demonstrating of removed highlights is finished. A client subordinate edge groups a test as credible or faked. Both master and irregular frauds test the proposed framework's precision. A few tests are run on SVC2004 and SUSIG, two freely accessible benchmark data sets. These data sets' analyses demonstrate this framework's adequacy.

We present a powerful signature-based character confirmation technique in this work. Issue seems, by all accounts, to be significant in biometrics. At the point when speed, pen pressure, and so forth are tended to, signature check turns out to be more compelling. These qualities are extraordinary to every client and difficult to duplicate. Investigation of mark elements can increment signature confirmation viability. The segment technique is a typical way to deal with break down signature properties. We present a parceling based character confirmation approach in this review. Time segments demonstrate client marking minutes. Allotments with more reliable reference fingerprints from obtaining are more fundamental in arrangement. Utilizing fluffy set hypothesis to develop adaptable neuro-fluffy frameworks and interpretable mark order frameworks is one more huge part of our method [3,29]. In this review, we present reproduction discoveries utilizing the free SVC2004 and business BioSecure dynamic mark data sets.

In biometrics, transcribed signature credibility confirmation is essential. There are a few fruitful mark check strategies that record for marking process elements. It are essential to Segment strategies. [5]We present another mark apportioning technique in this work. Its most fundamental capability is picking and handling cross breed allotments to further develop test signature investigation accuracy. The mark's vertical and level bits structure parcels. Vertical parts address the marking system's start, center, and end. On an illustrations tablet, flat parcels address signature zones for high and low pen speed and strain. [3,4,12,13]The calculation given in this work was created from our autonomous examination on vertical and level unique mark segments. Choice of areas, among others, characterizes parcel marking security, supporting mark districts of more prominent solidness (as well as the other way around). The proposed method was tried utilizing public MCYT-100 and paid BioSecure.[63]

3. METHODOLOGY

i) Proposed Work:

A hybrid procedure extricates particular picture qualities in the proposed framework. It utilizes CNN and Hoard, which are great at gathering convoluted examples and gradient data [39]. After feature extraction, Decision Trees pick key elements. This approach makes a feature vector with just the fundamental parts, making it more powerful for order occupations, remarkably signature recognition, by limiting superfluous information and further developing classification accuracy. The venture additionally incorporates Xception, Feature extraction (HOG-RFE), and Voting Classifier for Dataset examination, which yielded 100 percent exactness for better Signature Verification. Multi-Classification utilizing CNN and HOG. An easy to understand Flask framework with SQLite mix improves on client testing enrollment and signin, making network protection applications usable.

ii) System Architecture:

Project "Hybrid Feature Extraction for Signature Verification" CNN-HOG Multi-Classification Approach," the framework configuration is multi-

ISSN2321-2152
www.ijmece .com
Vol 12, Issue 3, 2024

stage. Preprocessing signature photographs in the preparation set is trailed by CNN-Hoard highlight extraction. Subsequent to extricating features, SVM, KNN, LSTM, and Voting Classifiers are prepared [2]. Expansions incorporate Xception, HOG-RFE, and Voting Classifier. Signature photographs are preprocessed and feature removed prior to being tried against the information base. The different classifiers and information base are utilized to recognize genuine and fake marks, offering serious areas of strength for an accurate multi-classification strategy for signature verification.[65]

Fig 1 Proposed architecture

This segment momentarily depicts the signature verification system's feature extraction and classification methods. The proposed signature arrangement approach utilizes two feature extraction strategies and three classifiers. HOG was utilized to remove signature picture attributes in this examination. Dalal and Triggs presented quality shape portrayal at the 2005 CVPR meeting. HOG executed it. Most individual finders use HOG, or Histograms of Oriented Gradients. [35,36] In this work, HOG was utilized alone and with CNN to extricate features to distinguish and perceive signature photographs.

iii) Dataset collection:

To fathom the construction, qualities, and items in the CEDAR and UTSig datasets, an investigation is led. In this stage, the datasets are stacked, information measurements are analyzed, examples are envisioned, and experiences into the appropriation of genuine and fake signatures are acquired.

Fig 2 Dataset

iv) Image Processing:

Across a few basic levels, image processing is crucial for independent driving frameworks' item ID system. The initial step is to enhance the info picture for additional examination and change by transforming it into a mass article. The classes of things to be found are then determined, characterizing the specific gatherings that the calculation looks to find. Bounding boxes, which separate the areas of interest inside the image where things are expected to be set, are all the while declared. The following fundamental stage for compelling mathematical figuring and investigation is to change the handled information into a NumPy exhibit.

The subsequent stage is to stack a model that has proactively been prepared utilizing prior data from enormous datasets. Perusing the pre-prepared model's organization layers, which incorporate learnt elements and boundaries fundamental for exact item distinguishing proof, is one part of this cycle.

ISSN2321-2152
www.ijmece .com
Vol 12, Issue 3, 2024

Furthermore, yield layers are extricated to give last expectations and work with proficient article classification and insight.

Moreover, the image and comment record are included the picture handling pipeline, ensuring intensive data for additional examination. In the wake of changing over from BGR to RGB, the variety space is adjusted and a cover is made to cause to notice significant qualities. The picture is then streamlined for extra handling and examination by resizing it. In the powerful climate of independent driving frameworks, this exhaustive image processing philosophy lays the basis for solid and exact article acknowledgment, further developing street wellbeing and critical thinking abilities.

v) Feature Extraction:

ML utilizes include extraction to diminish handling assets without forfeiting basic data. Include extraction lessens information dimensionality for ideal handling. Subsequently, Feature extraction incorporates growing new elements that catch the pertinent information from the first information all the more proficiently. Huge datasets, particularly in image, natural language, and sign handling, can incorporate numerous unessential or copy attributes. Feature extraction works on information, making calculations faster and more viable.

- Reduced computational cost: Information dimensionality decrease speeds up ML methods. This is vital for refined calculations and huge datasets.
- Improved Performance: Less features can prompt more noteworthy calculation execution. Since commotion and incidental data are killed, the calculation can zero in on the most significant realities.[67]
- Prevent Overfitting: Inordinate features can make models overfit to preparing information, impeding their capacity to sum up to new information. Feature extraction improves on the model, forestalling this.
- Improved Data Understanding: Removing and picking key ascribes uncovers the components behind data generation.

vi) Algorithms:

CNN, Programmed and various leveled feature learning from brand name photographs utilizing a DL engineering permits the model to catch inconspicuous examples and varieties. While HOG succeeds at catching neighborhood inclination data, the crossover strategy consolidates the capacities of the two calculations [45,48,49]. This synergistic mix further develops signature verification accuracy and proficiency, permitting the framework to sort signatures across many classes for complete confirmation and verification.

```
model = Sequential()

model.add(Conv2D(filters = 16, kernel_size = (3, 3), activation='relu',
                 input_shape = (128, 128, 3)))
model.add(BatchNormalization())
model.add(Conv2D(filters = 16, kernel_size = (3, 3), activation='relu'))
model.add(BatchNormalization())
model.add(MaxPool2D(strides=(2,2)))
model.add(Dropout(0.25))

model.add(Conv2D(filters = 32, kernel_size = (3, 3), activation='relu'))
model.add(BatchNormalization())
model.add(Conv2D(filters = 32, kernel_size = (3, 3), activation='relu'))
model.add(BatchNormalization())
model.add(MaxPool2D(strides=(2,2)))
model.add(Dropout(0.25))

model.add(Flatten())
model.add(Dense(512, activation='relu'))
model.add(Dropout(0.25))

model.add(Dense(1024, activation='relu'))
model.add(Dropout(0.4))
model.add(Dense(2, activation='softmax'))

learning_rate = 0.001

model.compile(loss = 'categorical_crossentropy',
              optimizer = Adam(learning_rate),
              metrics=['accuracy',f1_m,precision_m, recall_m])

model.summary()
```

IJMECE
International Journal of modern electronics and communication engineering
ISSN2321-2152
www.ijmece .com
Vol 12, Issue 3, 2024

Fig 3 CNN

Support Vector Machine tackles regression and classification issues supervisedly. SVM can sort marks in view of CNN and HOG attributes for signature verification. SVM expands edge between classes by finding a hyperplane.

```
from sklearn.svm import SVC
svm_model = SVC()
svm_model.fit(X_train_feature, y_train) #For sklearn no one hot encoding

prediction_svm = svm_model.predict(X_test_features)
#Inverse le transform to get original label back.
prediction_svm = le.inverse_transform(prediction_svm)

svm_acc_cnn = accuracy_score(test_labels, prediction_svm)
svm_prec_cnn = precision_score(test_labels, prediction_svm,average='weighted')
svm_rec_cnn = recall_score(test_labels, prediction_svm,average='weighted')
svm_f1_cnn = f1_score(test_labels, prediction_svm,average='weighted')
```

Fig 4 SVM

K-Nearest Neighbors is a simple classification technique. It characterizes another information point utilizing the feature space larger part class of its K nearest neighbors. This task utilizes CNN and Hoard qualities to arrange signatures with KNN.

```
from sklearn.neighbors import KNeighborsClassifier
knn_model = KNeighborsClassifier(n_neighbors=3)
knn_model.fit(X_train_feature, y_train) #For sklearn no one hot encoding

prediction_knn = knn_model.predict(X_test_features)
#Inverse le transform to get original label back.
prediction_knn = le.inverse_transform(prediction_knn)

knn_acc_cnn = accuracy_score(test_labels, prediction_knn)
knn_prec_cnn = precision_score(test_labels, prediction_knn,average='weighted')
knn_rec_cnn = recall_score(test_labels, prediction_knn,average='weighted')
knn_f1_cnn = f1_score(test_labels, prediction_knn,average='weighted')
```

Fig 5 KNN

LSTM is consecutive information demonstrating recurrent neural network (RNN). In this examination, LSTM can deal with signature-related time successions or CNN and HOG feature arrangements. [57,58] Long haul conditions and examples in consecutive signature information can be caught involving LSTM for signature verification.

```
X_train=X_train_feature
X_test=X_test_features

X_train = X_train.reshape(-1, X_train.shape[1],1)
X_test = X_test.reshape(-1, X_test.shape[1],1)

Y_train=to_categorical(y_train)
Y_test=to_categorical(y_test)
```

Fig 6 LSTM

Xception is a deep learning design for image order utilizing depthwise detachable convolutions. This creative technique consolidates spatial data across channels by leading depthwise convolutions for each info channel and afterward a 1x1 convolution. This strategy makes Xception more boundary productive than standard frameworks, bringing down registering intricacy and holding accuracy. Xception succeeds in extricating progressive features from input information in computer vision applications.

Fig 7 Xception

Voting Classifiers estimate utilizing many ML models. It utilizes RF and DT. RF creates numerous DT's and totals their figures. Grouping is finished with essential DT. The Voting Classifier upgrades model prediction and flexibility by incorporating RF and DT through voting.[69]

```
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import RandomForestClassifier, VotingClassifier
clf1 = DecisionTreeClassifier()
clf2 = RandomForestClassifier()

eclf1 = VotingClassifier(estimators=[('dt', clf1),('rf', clf2)], voting='soft')
eclf1.fit(X_train_feature, y_train)

prediction_vot = eclf1.predict(X_test_features)
#Inverse le transform to get original label back.
prediction_vot = le.inverse_transform(prediction_vot)

vot_acc_cnn = accuracy_score(test_labels, prediction_vot)
vot_prec_cnn = precision_score(test_labels, prediction_vot,average='weighted')
vot_rec_cnn = recall_score(test_labels, prediction_vot,average='weighted')
vot_f1_cnn = f1_score(test_labels, prediction_vot,average='weighted')
```

Fig 8 Voting classifier

4. EXPERIMENTAL RESULTS

Precision: Precision estimates the level of positive cases or tests precisely sorted. Precision is determined utilizing the recipe:

Precision = True positives/ (True positives + False positives) = TP/(TP + FP)

$$\text{Precision} = \frac{True\ Positive}{True\ Positive + False\ Positive}$$

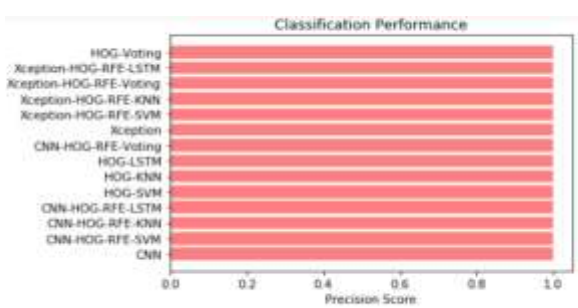

Fig 9 Precision comparison graph

Recall: Machine learning recall assesses a model's ability to perceive all significant examples of a class. It shows a model's culmination in catching occasions of a class by contrasting accurately anticipated positive perceptions with complete positives.

$$Recall = \frac{TP}{TP + FN}$$

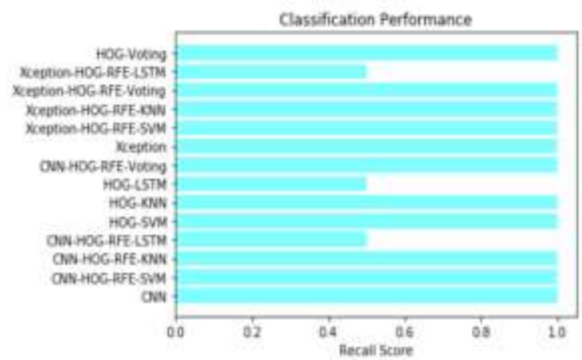

Fig 10 Recall comparison graph

Accuracy: A test's accuracy is its ability to recognize debilitated from sound cases. To quantify test accuracy, figure the small part of true positive and true negative in completely broke down cases. Numerically, this is:

Accuracy = TP + TN TP + TN + FP + FN.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

ISSN2321-2152
www.ijmece .com
Vol 12, Issue 3, 2024

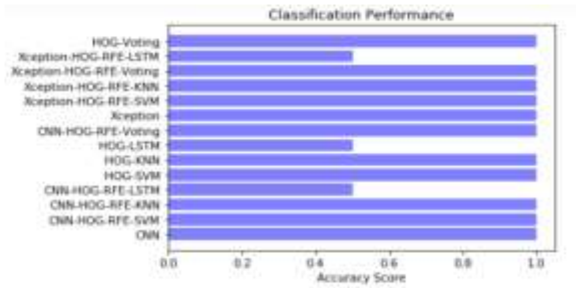

Fig 11 Accuracy graph

F1 Score: Machine learning model accuracy is estimated by F1 score. Consolidating model precision and recall scores. The accuracy measurement estimates how frequently a model anticipated accurately all through the dataset.

$$\text{F1 Score} = 2 * \frac{\text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}} * 100$$

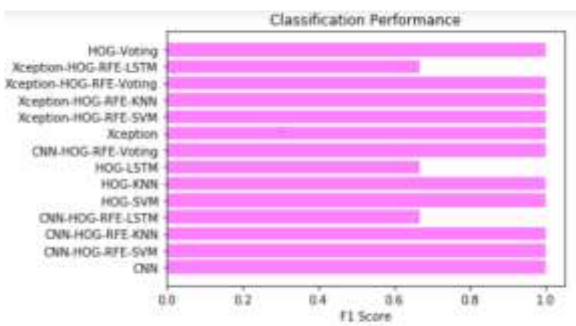

Fig 12 F1Score

ML Model	Accuracy	Precision	Recall	F1-Score
CNN	0.862	0.910	0.828	0.865
CNN-HOG-RFE-SVM	0.892	0.921	0.892	0.894
CNN-HOG-RFE-KNN	0.882	0.911	0.882	0.888
CNN-HOG-RFE-LSTM	0.009	1.000	0.009	0.017
HOG-SVM	0.555	0.589	0.555	0.540
HOG-KNN	0.555	0.585	0.555	0.540
HOG-LSTM	0.009	1.000	0.009	0.017
CNN-HOG-RFE-Voting	0.899	0.920	0.899	0.901
Xception	0.849	0.878	0.834	0.849
Xception-HOG-RFE-SVM	0.555	0.589	0.555	0.540
Xception-HOG-RFE-KNN	0.466	0.498	0.466	0.449
Extension Xception-HOG-RFE-Voting	0.421	0.452	0.421	0.413
Xception-HOG-RFE-LSTM	0.009	1.000	0.009	0.017
HOG-Voting	0.555	0.589	0.555	0.540

Fig 13 Performance Evaluation table

Fig 14 Home page

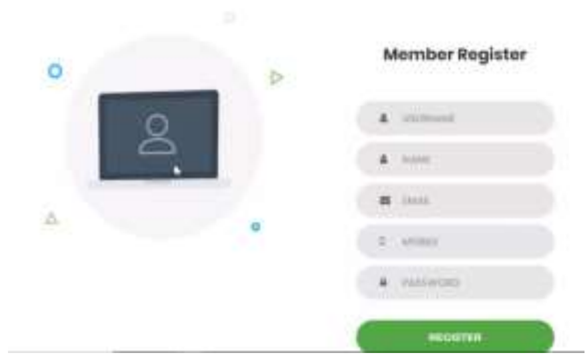

Fig 15 Registration page

Fig 16 Login page

ISSN2321-2152

www.ijmece .com

Vol 12, Issue 3, 2024

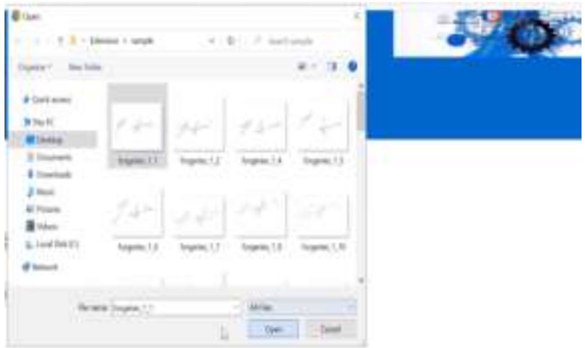

Fig 17 Input image folder

Fig 18 Upload input image

Fig 19 Predict result for given input

5. CONCLUSION

The examination presents a hybrid signature verification approach utilizing CNN and HOG. Optimization utilizing DT guarantees the consolidated feature extraction approach's viability and rightness. The recommended strategy is flexible since CNN, HOG, and Xception include sets are utilized to prepare the models. SVM, KNN, and LSTM classifiers effectively recognize marks in light of extricated attributes. Flask makes a simple to-involve interface for signature picture transfer and analysis.Integrating client confirmation further develops framework ease of use and security. Dataset investigation is 100 percent precise utilizing progressed models like Xception, HOG-RFE, and a Voting Classifier [45]. Its incredible presentation and strength make it a decent CNN and HOG signature verification arrangement. Framework testing, where information is provided for execution assessment, is simpler with an easy to use Cup interface. Security is improved with secure validation, which limits framework admittance to approved clients.

6. FUTURE SCOPE

Feature extraction is fundamental to signature verification. You work on this strategy to all the more likely gather signature ascribes and make the confirmation framework more exact and reliable. Refining feature extraction ought to further develop signature verification system execution. This further develops accuracy, diminishes false positives/negatives, and assists the framework with distinguishing fake signatures. [48] Adjusting the mark confirmation framework for versatile verification and virtual endorsements expands its reasonableness. The framework might be utilized at a few secure section focuses because of its variety. Refining the UI makes the framework simple to utilize, which supports reception. Bank exchanges and security passageways call for ongoing deduction. The model is improved for ongoing outcomes to ensure commonsense application in settings where rapid confirmation is pivotal for security and effectiveness.

REFERENCES

[1] F. M. Alsuhimat and F. S. Mohamad, ‘‘Offline signature verification using long short-term memory and histogram orientation gradient,’’ Bull. Elect. Eng. Inform., vol. 12, no. 1, pp. 283–292, 2023.

[2] M. Ajij, S. Pratihar, S. R. Nayak, T. Hanne, and D. S. Roy, ‘‘Off-line signature verification using elementary combinations of directional codes from boundary pixels,’’ Neural Comput. Appl., vol. 35, pp. 4939–4956, Mar. 2021, doi: 10.1007/s00521-021-05854-6.

[3] A. Q. Ansari, M. Hanmandlu, J. Kour, and A. K. Singh, ‘‘Online signature verification using segment-level fuzzy modelling,’’ IET Biometrics, vol. 3, no. 3, pp. 113–127, 2014.

[4] K. Cpałka and M. Zalasiński, ‘‘On-line signature verification using vertical signature partitioning,’’ Expert Syst. Appl., vol. 41, no. 9, pp. 4170–4180, 2014.

[5] K. Cpałka, M. Zalasiński, and L. Rutkowski, ‘‘A new algorithm for identity verification based on the analysis of a handwritten dynamic signature,’’ Appl. Soft Comput., vol. 43, no. 1, pp. 47–56, Jun. 2016.

[6] E. Griechisch, M. I. Malik, and M. Liwicki, ‘‘Online signature verification based on Kolmogorov–Smirnov distribution distance,’’ in Proc. 14th Int. Conf. Frontiers Handwriting Recognit., Sep. 2014, pp. 738–742.

[7] N. Sae-Bae and N. Memon, ‘‘Online signature verification on mobile devices,’’ IEEE Trans. Inf. Forensics Security, vol. 9, no. 6, pp. 933–947, Jun. 2014.

[8] S. Chen and S. Srihari, ‘‘A new off-line signature verification method based on graph matching,’’ in Proc. Int. Conf. Pattern Recognit. (ICPR), 2006, pp. 869–872.

[9] M. A. Ferrer, J. B. Alonso, and C. M. Travieso, ‘‘Offline geometric parameters for automatic signature verification using fixed-point arithmetic,’’ IEEE Trans. Pattern Anal. Mach. Intell., vol. 27, no. 6, pp. 993–997, Jun. 2005.

[10] Y. Guerbai, Y. Chibani, and B. Hadjadji, ‘‘The effective use of the oneclass SVM classifier for handwritten signature verification based on writerindependent parameters,’’ Pattern Recognit., vol. 48, no. 1, pp. 103–113, 2015.

[11] R. Larkins and M. Mayo, ‘‘Adaptive feature thresholding for off-line signature verification,’’ in Proc. 23rd Int. Conf. Image Vis. Comput. New Zealand, Nov. 2008, pp. 1–6.

[12] H. Lv, W. Wang, C. Wang, and Q. Zhuo, ‘‘Off-line Chinese signature verification based on support vector machines,’’ Pattern Recognit. Lett., vol. 26, no. 15, pp. 2390–2399, Nov. 2005.

[13] Y. Serdouk, H. Nemmour, and Y. Chibani, ‘‘New off-line handwritten signature verification method

based on artificial immune recognition system,'' Expert Syst. Appl., vol. 51, pp. 186–194, Jun. 2016.

[14] F. E. Batool, M. Attique, M. Sharif, K. Javed, M. Nazir, A. A. Abbasi, Z. Iqbal, and N. Riaz, ''Offline signature verification system: A novel technique of fusion of GLCM and geometric features using SVM,'' Multimedia Tools Appl., pp. 1–20, Apr. 2020, doi: 10.1007/s11042-020-08851-4.

[15] F. M. Alsuhimat and F. S. Mohamad, ''Histogram orientation gradient for offline signature verification via multiple classifiers,'' Nveo-Natural Volatiles Essential OILS J., vol. 8, no. 6, pp. 3895–3903, 2021.

[16] N. M. Tahir, N. Adam, U. I. Bature, K. A. Abubakar, and I. Gambo, ''Offline handwritten signature verification system: Artificial neural network approach,'' Int. J. Intell. Syst. Appl., vol. 1, no. 1, pp. 45–57, 2021.

[17] A. B. Jagtap, D. D. Sawat, R. S. Hegadi, and R. S. Hegadi, ''Verification of genuine and forged offline signatures using Siamese neural network (SNN),'' Multimedia Tools Appl., vol. 79, nos. 47–48, pp. 35109–35123, Dec. 2020.

[18] B. Kiran, S. Naz, and A. Rehman, ''Biometric signature authentication using machine learning techniques: Current trends, challenges and opportunities,'' Multimedia Tools Appl., vol. 79, no. 1, pp. 289–340, 2020.

[19] M. Sharif, M. A. Khan, M. Faisal, M. Yasmin, and S. L. Fernandes, ''A framework for offline signature verification system: Best features selection approach,'' Pattern Recognit. Lett., vol. 139, pp. 50–59, Nov. 2020.

[20] N. Sharma, S. Gupta, and P. Metha, ''A comprehensive study on offline signature verification,'' in Proc. J. Phys., Conf., 2021, Art. no. 012044, doi: 10.1088/1742-6596/1969/1/012044.

[21] H. H. Kao and C. Y. Wen, ''An offline signature verification and forgery detection method based on a single known sample and an explainable deep learning approach,'' Appl. Sci., vol. 10, no. 1, p. 3716, 2020.

[22] M. K. Kalera, S. Srihari, and A. Xu, ''Offline signature verification and identification using distance statistics,'' Int. J. Pattern Recognit. Artif. Intell., vol. 18, no. 7, pp. 1339–1360, 2004.

[23] B. Kovari and H. Charaf, ''A study on the consistency and significance of local features in off-line signature verification,'' Pattern Recognit. Lett., vol. 34, no. 3, pp. 247–255, 2013.

[24] T.-A. Pham, H.-H. Le, and N.-T. Do, ''Offline handwritten signature verification using local and global features,'' Ann. Math. Artif. Intell., vol. 75, nos. 1–2, pp. 231–247, Oct. 2015.

[25] Z. ZulNarnain, M. S. Rahim, N. F. Ismail, and M. M. Arsad, ''Triangular geometric feature for offline signature verification,'' Int. J. Comput. Inf. Eng., vol. 10, no. 3, pp. 485–488, 2016.

[26] R. K. Bharathi and B. H. Shekar, ''Off-line signature verification based on chain code histogram and support vector machine,'' in Proc. Int. Conf. Adv. Comput., Commun. Informat. (ICACCI), Aug. 2013, pp. 2063–2068.

[27] V. Nguyen, Y. Kawazoe, T. Wakabayashi, U. Pal, and M. Blumenstein, ''Performance analysis of the

gradient feature and the modified direction feature for off-line signature verification,'' in Proc. 12th Int. Conf. Frontiers Handwriting Recognit., Nov. 2010, pp. 303–307.

[28] R. Kumar, J. D. Sharma, and B. Chanda, ''Writer-independent off-line signature verification using surroundedness feature,'' Pattern Recognit. Lett., vol. 33, no. 3, pp. 301–308, Feb. 2012.

[29] M. Hanmandlu, M. H. M. Yusof, and V. K. Madasu, ''Off-line signature verification and forgery detection using fuzzy modeling,'' Pattern Recognit., vol. 38, no. 3, pp. 341–356, 2005.

[30] N. Jiang, J. Xu, W. Yu, and S. Goto, ''Gradient local binary patterns for human detection,'' in Proc. IEEE Int. Symp. Circuits Syst. (ISCAS), May 2013, pp. 978–981.

[31] J. Vargas, M. Ferrer, C. Travieso, and J. Alonso, ''Off-line signature verification based on high pressure polar distribution,'' in Proc. 11th Int. Conf. Frontiers Handwriting Recognit. (ICFHR), 2008, pp. 373–378.

[32] D. Bertolini, L. S. Oliveira, E. Justino, and R. Sabourin, ''Reducing forgeries in writer-independent off-line signature verification through ensemble of classifiers,'' Pattern Recognit., vol. 43, no. 1, pp. 387–396, Jan. 2010.

[33] M. V. M. Kumar and N. B. Puhan, ''Off-line signature verification: Upper and lower envelope shape analysis using chord moments,'' IET Biometrics, vol. 3, no. 4, pp. 347–354, 2014.

[34] E. N. Zois, L. Alewijnse, and G. Economou, ''Offline signature verification and quality characterization using poset-oriented grid features,'' Pattern Recognit., vol. 54, pp. 162–177, Jun. 2016.

[35] M. Subramaniam, E. Teja, and A. Mathew, ''Signature forgery detection using machine learning,'' Int. Res. J. Modernization Eng. Technol. Sci., vol. 4, no. 2, pp. 479–483, 2022.

[36] R. Kumar, M. Saraswat, D. Ather, M. N. Mumtaz Bhutta, S. Basheer, and R. N. Thakur, ''Deformation adjustment with single real signature image for biometric verification using CNN,'' Comput. Intell. Neurosci., vol. 2022, pp. 1–12, Jun. 2022, doi: 10.1155/2022/4406101.

[37] U. Jindal, S. Dalal, G. Rajesh, N. U. Sama, and N. Z. Jhanjhi, ''An integrated approach on verification of signatures using multiple classifiers (SVM and decision Tree): A multi-classification approach,'' Int. J. Adv. Appl. Sci., vol. 9, no. 1, pp. 99–109, Jan. 2022.

[38] S. Jagtap, S. Kalyankar, T. Jadhav, and A. Jarali, ''Signature Verification and detection system,'' Int. J. Recent Sci. Res., vol. 13, no. 6, pp. 1412–1418, 2022.

[39] Y. Zhou, J. Zheng, H. Hu, and Y. Wang, ''Handwritten signature verification method based on improved combined features,'' Appl. Sci., vol. 11, no. 13, p. 5867, 2021.

[40] M. Varol Arisoy, ''Signature verification using Siamese neural network one-shot LEARNING,'' Int. J. Eng. Innov. Res., pp. 248–260, Aug. 2021.

[41] S. Pal, A. Alaei, U. Pal, and M. Blumenstein, ''Performance of an off-line signature verification

method based on texture features on a large indicscript signature dataset,'' in Proc. 12th IAPR Workshop Document Anal. Syst. (DAS), Apr. 2016, pp. 72–77.

[42] H. Loka, E. Zois, and G. Economou, ''Long range correlation of preceded pixels relations and application to off-line signature verification,'' IET Biometrics, vol. 6, no. 2, pp. 70–78, 2017.

[43] E. N. Zois, A. Alexandridis, and G. Economou, ''Writer independent offline signature verification based on asymmetric pixel relations and unrelated training-testing datasets,'' Expert Syst. Appl., vol. 125, pp. 14–32, Jul. 2019.

[44] J. Lopes, B. Baptista, N. Lavado, and M. Mendes, ''Offline handwritten signature verification using deep neural networks,'' Energies, vol. 15, p. 7611, 2022.

[45] N. Dalal and B. Triggs, ''Histograms of oriented gradients for human detection,'' in Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR), San Diego, CA, United States, Jun. 2005, pp. 886–893.

[46] N. Abbas, K. Yasen, K. H. Faraj, L. Razak, and F. Malaliah, ''Offline handwritten signature recognition using histogram orientation gradient and support vector machine,'' J. Theor. Appl. Inf. Technol., vol. 96, pp. 2048–2075, 2018.

[47] S. Singh, M. Gogate, and S. Jagdale, ''Signature verification using LDP & LBP with SVM classifiers,'' Int. J. Sci. Eng. Sci., vol. 1, no. 11, pp. 95–98, 2017.

[48] A. Krizhevsky, I. Sutskever, and G. E. Hinton, ''Images classification with deep convolutional neural networks,'' in Proc. Adv. Neural Inf. Process. Syst., 2012, pp. 1097–1105.

[49] Y. LeCun, B. Boser, J. Denker, D. Henderson, R. Howard, W. Hubbard, and L. Jackel, ''Handwritten digit recognition with a back-propagation network,'' in Proc. Adv. Neural Inf. Process. Syst., 1990, pp. 396–404.

[50] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, ''Gradient-based learning applied to document recognition,'' Proc. IEEE, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[51] M. F. Yahya and M. R. Arshad, ''Detection of markers using deep learning for docking of autonomous underwater vehicle,'' in Proc. IEEE 2nd Int. Conf. Autom. Control Intell. Syst. (I2CACIS), Oct. 2017, pp. 179–184.

[52] C. Boufenar, A. Kerboua, and M. Batouche, ''Investigation on deep learning for off-line handwritten Arabic character recognition,'' Cognit. Syst. Res., vol. 50, pp. 180–195, Aug. 2018.

[53] M. Li, H. Wang, L. Yang, Y. Liang, Z. Shang, and H. Wan, ''Fast hybrid dimensionality reduction method for classification based on feature selection and grouped feature extraction,'' Expert Syst. Appl., vol. 151, Jul. 2020, Art. no. 113277.

[54] R. Olmos, S. Tabik, and F. Herrera, ''Automatic handgun detection alarm in videos using deep learning,'' Neurocomputing, vol. 275, pp. 66–72, Jan. 2018.

[55] V. D. Nguyen, H. Van Nguyen, D. T. Tran, S. J. Lee, and J. W. Jeon, ''Learning framework for robust

obstacle detection, recognition, and tracking,'' IEEE Trans. Intell. Transp. Syst., vol. 18, no. 6, pp. 1633–1646, Jun. 2017.

[56] F. S. Mohamad, M. Iqtait, and F. Alsuhimat, ''Age prediction on face features via multiple classifiers,'' in Proc. 4th Int. Conf. Comput. Technol. Appl. (ICCTA), May 2018, pp. 161–166.

[57] X. H. Le, H. V. Ho, G. Lee, and S. Jung, ''Application of long short-term memory (LSTM) neural network for flood forecasting,'' Water, vol. 11, no. 7, p. 1387, 2019.

[58] K. Greff, R. K. Srivastava, J. Koutnìk, B. R. Steunebrink, and J. Schmidhuber, ''LSTM: A search space Odyssey,'' IEEE Trans. Neural Netw. Learn. Syst., vol. 28, no. 10, pp. 2222–2232, Oct. 2017.

[59] A. Graves, M. Liwicki, S. Fernandez, R. Bertolami, H. Bunke, and J. Schmidhuber, ''A novel connectionist system for unconstrained handwriting recognition,'' IEEE Trans. Pattern Anal. Mach. Intell., vol. 31, no. 5, pp. 855–868, May 2009.

[60] S. Yan. Understanding LSTM and Its Diagrams. Accessed: Jan. 10, 2023. [Online]. Available: https://medium.com/mlreview/understanding-lstmand-its-diagrams-37e2f46f1714

[61]G.Viswanath, "Hybrid encryption framework for securing big data storage in multi-cloud environment", Evolutionary intelligence, vol.14, 2021, pp.691-698.

[62] Viswanath Gudditi, "Adaptive Light Weight Encryption Algorithm for Securing Multi-Cloud Storage", Turkish Journal of Computer and Mathematics Education (TURCOMAT), vol.12, 2021, pp.545-552.

[63] Viswanath Gudditi, "A Smart Recommendation System for Medicine using Intelligent NLP Techniques", 2022 International Conference on Automation, Computing and Renewable Systems (ICACRS), 2022, pp.1081-1084.

[64] G.Viswanath, "Enhancing power unbiased cooperative media access control protocol in manets", International Journal of Engineering Inventions, 2014, vol.4, pp.8-12.

[65] Viswanath G, "A Hybrid Particle Swarm Optimization and C4.5 for Network Intrusion Detection and Prevention System", 2024, International Journal of Computing, DOI: https://doi.org/10.47839/ijc.23.1.3442, vol.23, 2024, pp.109-115.

[66] G.Viswanath, "A Real Time online Food Ording application based DJANGO Restfull Framework", Juni Khyat, vol.13, 2023, pp.154-162.

[67] Gudditi Viswanath, "Distributed Utility-Based Energy Efficient Cooperative Medium Access Control in MANETS", 2014, International Journal of Engineering Inventions, vol.4, pp.08-12.

[68] G.Viswanath," A Real-Time Video Based Vehicle Classification, Detection And Counting System", 2023, Industrial Engineering Journal, vol.52, pp.474-480.

[69] G.Viswanath, "A Real- Time Case Scenario Based On Url Phishing Detection Through Login Urls ", 2023, Material Science Technology, vol.22, pp.103-108.

[70] Manmohan Singh,Susheel Kumar Tiwari, G. Swapna, Kirti Verma, Vikas Prasad, Vinod Patidar,

Dharmendra Sharma and Hemant Mewada, "A Drug-Target Interaction Prediction Based on Supervised Probabilistic Classification" published in Journal of Computer Science, Available at: https://pdfs.semanticscholar.org/69ac/f07f2e756b79181e4f1e75f9e0f275a56b8e.pdf