IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 7 July 2022, accepted 21 July 2022, date of publication 26 August 2022, date of current version 2 September 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3202007

APPLIED RESEARCH

IPCRF: An End-to-End Indian Paper Currency Recognition Framework for Blind and Visually Impaired People

MANDHATYA SINGH[1], JOOHI CHAUHAN[2], MUHAMMAD SUHAIB KANROO[1], SAHIL VERMA[3], AND PUNEET GOYAL[1,2], (Life Member, IEEE)

[1]Computer Science and Engineering, IIT Ropar, Rupnagar, Punjab 140001, India
[2]Biomedical Engineering, IIT Ropar, Rupnagar, Punjab 140001, India
[3]Chemical Engineering, IIT Ropar, Rupnagar, Punjab 140001, India

Corresponding author: Mandhatya Singh (2017csz0003@iitrpr.ac.in)

This work was supported in part by the Department of Science and Technology (DST) through the Cognitive Science Research Initiative (CSRI) under Grant DST/CSRI/2018/234, and in part by the Institute Scheme for Innovative Research and Development (ISIRD) under Grant 9-231/2016/IIT-RPR/1395.

ABSTRACT Currency recognition has always been a troublesome task for blind and visually impaired people (BVIP). The problem is more severe in developing countries such as India, where there is still a lack of robust currency recognition systems. BVIP primarily relies on size variations and patterns such as intaglio printings for recognizing the underlying currency denominations. Most of the current Indian legal tenders resemble in size, thus making the identification process more strenuous. Also, the engraved patterns are not as distinctive as BVIP standards, and they fade over time. For an automated paper currency recognition system, issues such as folded or partial views, uneven illumination, and background clutter make it non-trivial and challenging. This paper ventures to present an end-to-end and robust framework for assisting BVIP in recognizing the Indian paper currency denomination. This paper presents a lightweight network, IPCRNet, useful in a resource-constrained environment such as low/medium level smartphones. The proposed network is based on Dense connection, Multi-Dilation, and Depth-wise separable convolution layers. Additionally, we congregated one of the most diversified Indian paper currency image dataset with more than 50,000 images belonging to almost all denominations in circulation. A customized and publically available android application, ‘‘Roshni-Currency recognizer’’, has also been introduced. The experimental results on multiple datasets demonstrate the superiority of the proposed model. IPCRNet improves the classification accuracy by more than 2% on the proposed dataset compared to the state-of-the-art networks.

INDEX TERMS Assistive technologies, currency identification, dense network, depthwise separable convolution, visually impaired.

I. INTRODUCTION

As per the global estimate, the number of people having some form or degree of vision impairment is close to approximately 2.2 billion, of which approximately 39 million people are legally Blind, and 237 million are with moderate and severe vision impairment (MSVI).[1] In India alone, the population of BVIP is around 62 million [1]. BVIP rely on various assistive solutions to overcome difficulties in independent adaptability involving their professional and social activities [2]. However, the AAA factor, i.e., Availability, Affordability, and Awareness related to assistive technologies, in general, are comparatively better in developed countries [3]. One of the critical issues for BVIP (more severe in developing countries such as India) is the recognition and authentication of currency denominations. In some cases, they need to depend on the normally sighted person (NSP) for currency identification or authentication assistance. Conventionally, the BVIP rely

The associate editor coordinating the review of this manuscript and approving it for publication was Yiqi Liu.

[1] https://www.lasereyesurgeryhub.co.uk/data/visual-impairment-blindness-data-statistics/

on the embedded positional patterns and differences in the size of the paper currencies for denomination recognition. However, this approach has associated limitations as the ability to sense the engraved patterns on the paper currency varies from person to person. Furthermore, the printed engravings and distinct patterns on paper currency get worn away with time. The problem is especially challenging with current Indian banknotes recognition due to (a) similarity in sizes and color and (b) imperceptible bleed lines/tactile marking.

To assist BVIP in recognizing currency denominations, several automated and semi-automated currency recognition systems are proposed in the literature. An automated currency recognition system utilizes the properties such as color, size, motifs, micro-lettering, engraved patterns, and edge parameters. However, unlike other countries' banknotes, the present paper currencies in India are almost similar in size without definite tactile attributes. Observing these difficulties in manual recognition of the current currency denominations by BVIP in India, various organizations, including the Reserve Bank of India (RBI), has shown a growing interest in mobile-based solutions recently [4]. A dedicated mobile-based solution can assist in recognizing the currency notes more effectively. BVIP relies on the voice-based features of smartphones like Talkback, Google assistant, and Siri for performing generic tasks. However, mobile-based solutions with dedicated machine/deep learning models for recognition are challenging in terms of deployability and adaptability. The existing models are comparatively bulkier, and integration with low-end mobile smartphones is impractical, thus causing deployability issues. Additionally, the users who are legally blind or have severe low vision issues cannot handle the camera view appropriately or other required settings to get the optimum results. Therefore a stable and robust recognition system is highly needed.

In recent years, deep learning-based networks have gone deeper to gain performance improvements. With more depth, a larger number of parameters is required, thus causing an exponential increase in the complexity of the networks. The bulkier models are unsuitable for resource-constrained devices such as mobile phones and edge-based devices. To minimize the dependence on servers and the internet, various lightweight networks such as [5], [6], [7], [8], and [9], have been proposed for the tasks of classification and object detection. Schemes such as limiting the number of channels/kernel size, optimizing pooling layers, efficient coding, and representation are typically used for compressing the network sizes. GoogleNet [10] uses a width reduction-based scheme, and MobileNet [7] uses depth-wise separable convolution for compressing the network size. However, with such compression schemes and the simplified convolutional structure, the model tends to miss discriminative image features, affecting the overall performance.

The proposed model, IPCRNet is a lightweight neural network and utilizes MobileNet as the front-end. IPCRNet uses a Contextual Block (CB) in the backend utilizing the dense connection and dilation scheme in depth-wise separable convolutional layers. The model has less than four million parameters, thus favoring its deployment in a resource-constrained environment. The novel contextual part utilizes a depthwise separable convolution for reducing network computations. The multi-dilation scheme offers an enlarged receptive field without increasing the parameters, thereby increasing the accuracy via an effective integration of global and semantic features. Compared with the existing state-of-the-art backend network and approaches, our network provides superior accuracy than its counterparts and is lightweight. Furthermore, to aid an effective training and evaluation of the proposed model, we have gathered a large-scale Indian paper currency dataset, IPCD. The IPCD dataset unlike existing datasets [11], [12], [13], [14], [15] consists of images with BVIP perspective (folded and partial note images), with varied illumination and background conditions. Additionally, we have built a robust android app named ''Roshni-Currency recognizer'', customized for the BVIP scenario and is publicly available. Roshni supports features like auto start, voice intimation regarding the denomination, and voice-based instructions to direct the user in case of an improper alignment camera view and underlying currency. The app provides full automaticity (a must-have feature for BVIP related scenarios) as once the camera gets started, it provides a hassle-free interface to the BVIP.

The proposed end-to-end Indian paper currency recognition framework (IPCRF) offers a contextual learning network, a diversified and domain perspective dataset (IPCD) to support an effective training/evaluation, and a BVIP compatible interface via android mobile application. The overall flow diagram is shown in Fig. 1. The proposed deep learning model is trained on our dataset (IPCD). Our android app utilizes the compressed trained model for real-time recognition of the underlying currency denomination.

The novelty and main contributions of this work are:

- **Novelty.** A robust lightweight and domain specialized CNN model to capture the pervasive intra-class and inter-class dissimilarities between currency denominations classes. The proposed Contextual Block offers an effective integration of the local and global features.
- **Diverse Dataset**. One of the largest (more than 50k images) and most diversified dataset of Indian Paper Currency images, representing real scenarios and cases.
- **Quantitative Analysis.** A thorough quantitative analysis of the proposed network on multiple publically available datasets has been performed.
- **Qualitative Analysis.** A quantitative analysis has been performed to investigate the transparency and intuition behind the proposed network predictions.
- **Publicly available android App ''Roshni-Currency recognizer''.** A publically available android App for BVIP named Roshni to assist them in recognizing Indian paper currency denomination.

The remainder of this paper is organized as follows. Section II presents an overview of the literature related to currency recognition problems and proposed networks with

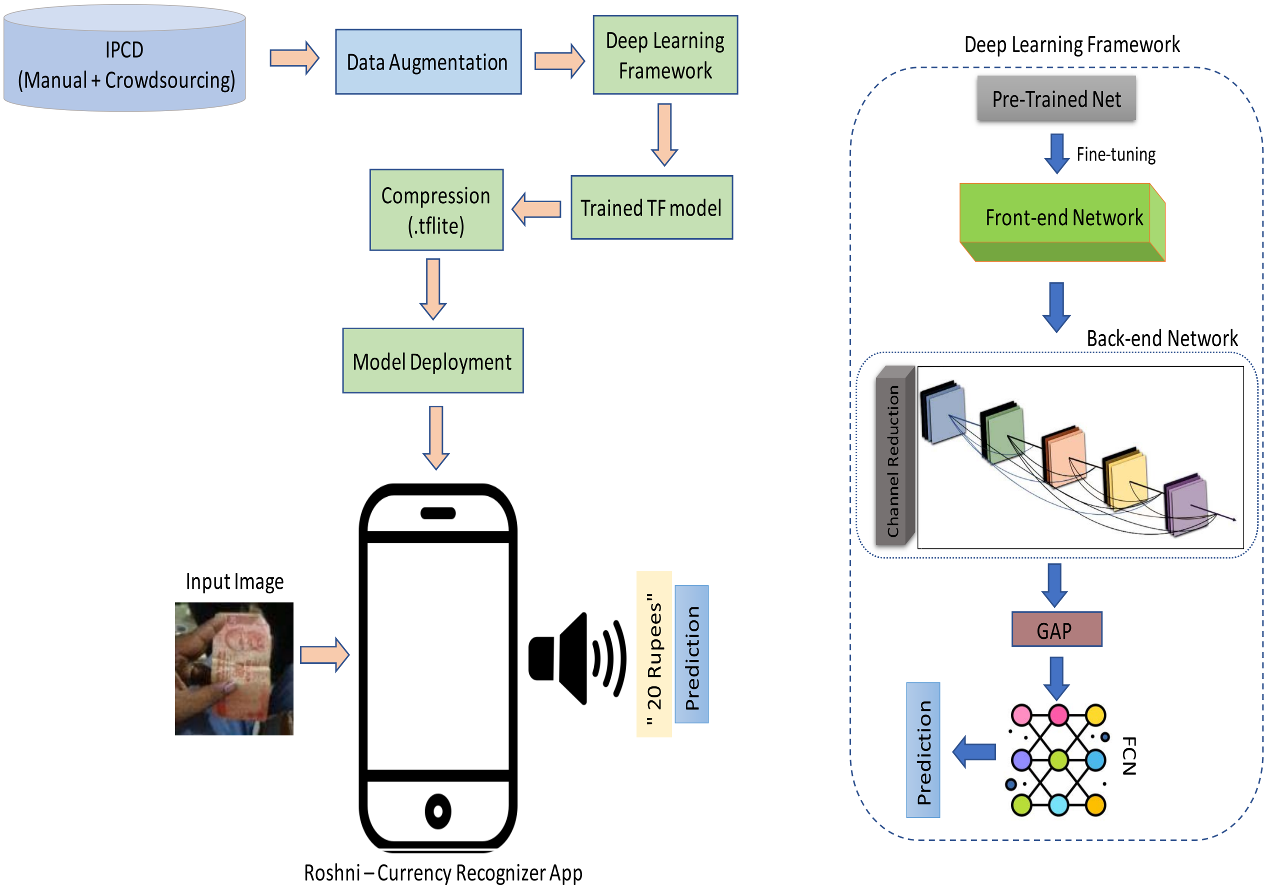


FIGURE 1. **Overall design flow of the proposed framework: IPCRF.**

their advantages and disadvantages. Section III discusses the proposed large-scale Indian paper currency image dataset-IPCD. Section IV explains the proposed network architecture and implementation details. Section V demonstrates the experimental setup and results, including quantitative and qualitative analysis. Section VI discusses the proposed android application-"Roshni." in Section VII, we have discussed the accuracy and reliability of results obtained, and In Section VIII, the conclusion and future scope are discussed.

II. RELATED WORK

The currency recognition problems has been well explored in the past [28], [29], [30], [31], [32]. We categorize the related existing approaches/systems into three aspects: Dataset (availability of diverse datasets for training and evaluation purposes); Model (availability of lightweight and accurate recognition models); and Application (availability of a BVIP compatible interface for assisting the BVIP in currency recognition tasks). This section discusses the existing approaches and related concepts related to the mentioned individual aspects.

The advantages and drawbacks of the proposed and existing methods are summarized in Table 1. The majority of the previous work uses smaller datasets. Also, very few works have used multiple datasets for evaluation which is critical for validating the generalization ability of underlying models. Standard methods and datasets dedicated to Indian currencies (INR) are limited. The detailed comparative analysis and description of available and proposed datasets are discussed in Section III. In this section, we primarily discuss the existing models and available systems/technologies available, along with a brief discussion of the dataset (columns 3, 4 and 5 of Table 1).

Lightweight yet accurate models and limited BVIP compatible real-time applications are also major concerns in the existing literature. The existing model's section discusses the models and frameworks used by currency recognition systems or approaches, categorized into two subsections: Hand-crafted feature-based Models and Deep Learning (DL) based models. The existing systems section discusses the existing assistive technologies (apps) available to the BVIP for currency identification. The details are provided below:

A. EXISTING MODELS

The models available in literature can be broadly categorized into two parts: handcrafted features and deep learning based.

TABLE 1. Summarized comparison of existing and proposed currency recognition approaches.

Approach (Year-Venue)	Method	Currency	# Images/ #Categories (C)	# Dataset (for Evaluation)	BVIP Compatibility	Application Availability
			Handcrafted Feature Based			
Liu et al. [16] (2008, SIGACCESS)	Background Subtraction + Adaboost	USD	1,000/4C	1	✓	✗
Hasanuzzaman et al. [17] (2012-IEEE TSMC)	Component Modeling + SURF	USD	190/7C	1	✗	✗
Singh et al. [11] (2014-ICPR)	BoW + SIFT	INR	2,571/6C	1	✓	✗
Doush et al. [18] (2017-JKUCIS)	SIFT	JOD	500/10C	1	✗	✗
			Deep Feature Based			
Zhang et al. [19] (2018-AVSS)	SSD	NZD	300/3C	1	✗	✗
Mittal et al. [20] (2018-IoT-SIU)	MobileNet	INR	380/4C	1	✗	✗
Huynh et al. [21] (2019-WACV)	MobNet + CONGAS features	USD	635/7C	1	✓	✗
Han et al. [22] (2019-MDPI Sensors)	4 Layer CNN	USD, EUR	45,055/7C	1	✗	✗
Park et al. [23] (2020-IEEE ACCESS)	FRCNN + Heuristic	JOD, KRW	JOD: 330/9C KRW: 6,400/8C	2	✓	✗
Veeramsetty et al. [14] (2020-MTAP)	6 Layer CNN	INR	4,657/10C	1	✗	✗
Pham et al. [24] (2020-IEEE ACCESS)	RoI Processing + CNN	USD, EUR, KRW	EUR: 480/5C USD: 576/6C KRW: 384/4C	3	✗	✗
Joshi et al. [25] (2020,IC3A)	Yolo-v3	INR	3,720/7C	1	✓	✗
Anwar et al. [26] (2021-PR)	CBP(D161+RNet50) + Attention	Historical RC	18,285/228C	2	✗	✗
Pachon et al. [27] (2021-MDPI AS)	8 layer CNN	COP	7,280/6C	1	✗	✗
Proposed IPCRNet	MobNet + Contextual Backend	INR	50,263/11C	5	✓	✓

1) HAND-CRAFTED FEATURE MODELS

In Hand-crafted feature-based schemes, firstly, the features are computed manually and based on which a decision-making or learning mechanism is built. The Template matching and Machine learning-based frameworks rely on the computed handcrafted features of the underlying currency images. In template matching-based approaches, the handcrafted features of the query image are matched with the features obtained from the set of template currency images. Guo *et al.* [33] used Local Binary Patterns (LBP) features for recognizing the currency denomination. The performance is susceptible to multi-scale and oriented images as LBP features rely on the histogram and are not scale or orientation invariant. Hassanpour *et al.* [34] used a probabilistic-based approach with Hidden Markov Model and texture-based features. Rajaei *et al.* [35] used features from statistical moments of the coefficient matrix obtained from Discrete Wavelet Transform (DWT) of the currency images. Essentially, both approaches lack the local features, which could lower the efficacy of query images with improper views or cluttered backgrounds. Doush *et al.* [18] evaluated color and gray Scale Invariant Feature Transform (SIFT) approaches and showed the better performance of the color SIFT approach in terms of both latency and accuracy. Hasanuzzaman *et al.* [17] used a component-based mechanism using speeded-up robust features (SURF) to reduce the memory requirements in the US currency (USD) recognition process. However, the framework is prone to error due to the possibility of missing discriminative patterns due to the cropping process.

In literature, most of the works have focused on the currencies like USD, EUR, JOD, KRW, and other currencies of developed countries. Standard methods and datasets dedicated to Indian currencies (INR) are limited. Singh *et al.* [11] attempt to build a real-time mobile application (for INR currency) using the background removal method, then compute the bag-of-words descriptor using the SIFT method. However, the technique is heavily dependent on the output of the background removal step; additionally, the used feature descriptor is sensitive to improper views and background cluttering.

The machine learning-based approaches also utilize handcrafted features for training and evaluation. Aoba *et al.* [36] use a three-layer perceptron and Radial Basis Function (RBF) network with a gradient-based preprocessing method for EUR currency class prediction. Wang *et al.* [37] and Liu *et al.* [16] used Adaboost based classifier for US currency image classification. Liu *et al.* [16] additionally used background removal and perspective correction-based preprocessing methods prior to training and introduced an application focused on BVIP. Debnath *et al.* [38] used an ensemble of neural network for TAKA (Bangladesh) currency class prediction.

2) DL MODELS

Recently deep learning techniques have been deployed for currency identification problems. Zhang *et al.* [19] used Single Shot MultiBox Detector (SSD) for currency detection. Huynh *et al.* [21] used a deep learning-based coarse classifier followed by a fine-grained classification system for USD recognition. The MobileNet based coarse classifier is used as a prefilter on input images to discriminate between the relevant currency class and other irrelevant classes such as barcode and text. For fine-grained currency recognition, a CONGAS-based feature is used. Park *et al.* [23] performed recognition of Korean won (KRW) banknotes and coins using a Faster Region-based Convolutional Neural Network (Faster-RCNN) and VGG16 backend. Anwar *et al.* [26] focuses on the recognition of gold and silver coinage of the historical roman era. The input images are encoded using DenseNet161 and ResNet50 (RNet50) based models. The obtained feature maps are then fused using Compact Bilinear Pooling (CBP). Further, to integrate the spatial information more effectively soft-attention layer has been utilized. Xiang *et al.* [39] used a combination of Long Short-Term Memory (LSTM) and CNN for recognizing fast-moving coin recognition in digital videos. For the CNN part, a 22-layer GoogLeNet model is used. Sun *et al.* [9] used a lightweight model based on depthwise and dilated convolutional layers. Veeramsetty *et al.* [14] used a 6 layer convolutional neural network for classifying INR notes. The existing DL models lack to provide an enlarged receptive field and thus lack a better trade-off between accuracy and the number of parameters.

B. EXISTING SYSTEMS

The majority of the approaches in the literature are limited to theoretical or desktop/web-based systems; however, there are limited commercial and non-commercial currency recognition systems available for mobile devices.

LookTel Money Reader[2] and IDEAL Currency Identifier [40] works well when the currency is placed correctly, with good lighting conditions. For folded or wrinkled currencies, the performance is not good. Also, these apps do not support Indian currencies. Microsoft's SeeingAI [41], app supports Indian currencies and seemingly works well when a currency note is in full view, but does not perform well when the currency is folded. Also, for some new currency denominations, like the new INR 100, this app does not provide correct results at times. Recently, for meeting the BVIP assistive need for such recognition systems, the Reserve Bank of India launched the MANI app [42] for both android and iOS platforms. Altogether, the core features vary across these apps. As per our knowledge, there is no publically available information or resources regarding the prediction models/methods used in these apps.

The advantages and drawbacks of the proposed and existing methods are summarized in Table 1. The majority of the previous work uses smaller datasets. Also, very few works have used multiple datasets for evaluation which is critical for validating the generalization ability of underlying models. BVIP compatibility and availability of related real-time applications are major concerns in the existing literature.

III. IPCD - INDIAN PAPER CURRENCY DATASET

In this section, we have provided a detailed description of the proposed IPCD dataset.

The primary objective of this work is to propose a BVIP compatible and efficient automated system for recognizing Indian paper currency denominations. The underlying model requires training on a range of estimable currency images with varied diversity and classes to build an effective automated currency recognition framework. The proposed IPCD dataset consists of a wide variety of currency images, including new denominations and old denominations of 10, 20, 50, and 100 banknotes. In addition to this, the new set of 500 denominations, including the newly introduced 200 and 2000 denominations, are also included in the dataset. The proposed dataset is among the most diverse datasets in terms of number of images, denomination classes, illuminations, and background variations.

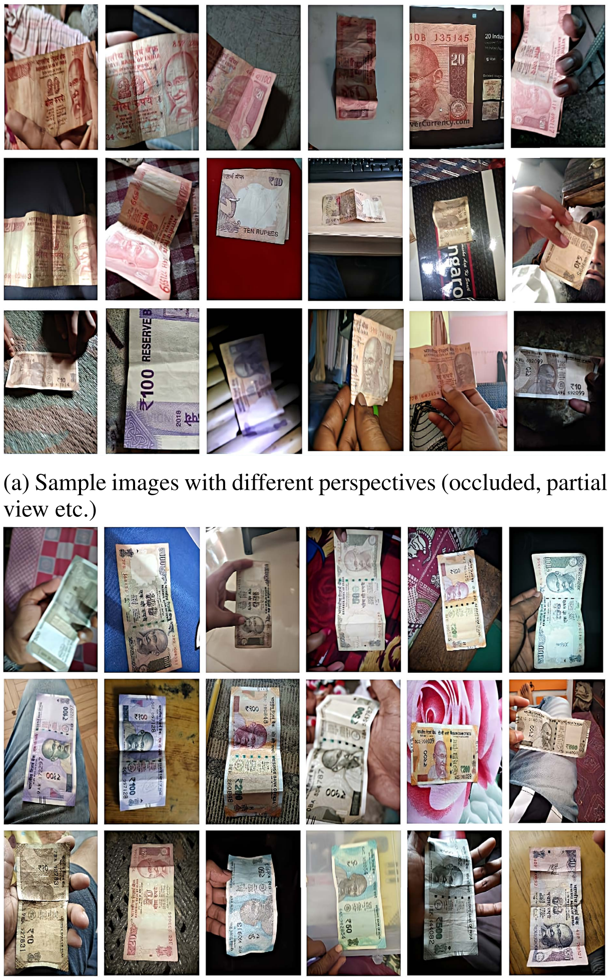
The training images should be composed of real-life BVIP usage scenarios to develop an efficient and generalizable network. However, this critical aspect is often overlooked in existing Indian currency datasets. A brief comparison of existing datasets is shown in Table 2. Using smaller datasets to train and evaluate the currency, classification approaches steer to non-viable and biased processes. Even the recent datasets [12], [13], [14], [15] involve lesser images as well as lack domain-specific scenarios creating vagueness about the viability of solutions. The two main issues with the existing Indian paper currency datasets are discussed below:

1) **Lacking Domain-Specific Perspective:** Existing paper currency datasets, more specifically Indian paper currency datasets [11], [14], [15] lack the BVIP perspective images. In existing datasets, the images are mainly curated through the help of automated scrappers or with the images clicked by NSP. The pattern of holding currency images with BVIP is significantly different from that of NSP. In the BVIP scenario, the illumination condition, cluttered background, and low-quality images are common as they can not self-adjust the best view, unlike NSP. In proposed dataset, the majority of images are prepared from the BVIP standpoint. IPCD images have partial view, folded view, and occlusions as shown in Fig. 2a and Fig. 2b. These factors and attributes are essential in training a model more effectively and handling a real BVIP congenial scenario.

[2] http://www.looktel.com/

TABLE 2. **Publicly available Indian currency image datasets comparison.**

Year	Approach	Total Images	Classes	Limitation
2014	Singh et al. [11]	5,500	5	No new denominations, Limited variations, No folded notes.
2019	Kaggle-U1 [12]	772	7	Limited variations, Lesser images, Clean background, Repeated images, No folded notes.
2020	Kaggle-U2 [13]	3,571	7	Limited variations, Repeated images, Clean background, No folded notes
2020	Veeramsetty [14]	4,657	7	No new denominations, Repeated images, Uniform images, Limited background, No folded notes
2021	Meshram et al. [15]	2,900	10	No 20 INR denomination, Limited variations, Repeated images, Uniform images, No folded notes
2021	IPCD (Ours)	50,263	11	All denominations (legal), Folded and partial notes, Varied background, No repeated images, Larger dataset, Varied illumination



(a) Sample images with different perspectives (occluded, partial view etc.)

(b) Illustration of images with varied backgrounds.

FIGURE 2. **Illustration of diversification and BVIP Perceptiveness of proposed IPCD dataset. (a) and (b) shows the currency images with varied perspectives and backgrounds.**

2) **Smaller Datasets:** To the best of our knowledge, no Indian paper currency dataset contains more than 5.5k images, and very few include all the denominations currently in use (as shown in Table 2). Using relatively smaller datasets to assess currency classification approaches' generalization capability and performance leads to inconsistent and ambiguous models. Our dataset is approximately ten times larger than the existing Indian paper currency datasets.

TABLE 3. **Proposed dataset (IPCD) description.**

Denomination	New/Old	Partition	Landscape	Portrait	Partial
10	Old	Train	459	2,393	928
		Val	123	848	289
		Test	98	683	494
	New	Train	258	1,617	558
		Val	85	628	98
		Test	71	576	148
20	Old	Train	232	1649	567
		Val	78	591	159
		Test	110	687	27
	New	Train	415	477	245
		Val	20	77	53
		Test	31	49	71
50	Old	Train	138	960	641
		Val	21	139	90
		Test	26	130	108
	New	Train	220	2,179	465
		Val	89	693	160
		Test	95	788	71
100	Old	Train	574	3,196	994
		Val	210	1,123	234
		Test	195	1,210	121
	New	Train	250	2,086	323
		Val	82	652	152
		Test	52	710	95
200		Train	200	1,971	256
		Val	123	572	134
		Test	122	568	85
500		Train	438	4,193	451
		Val	184	1,090	484
		Test	187	1,220	364
2000		Train	143	1,566	136
		Val	29	221	50
		Test	21	219	72

A. DATASET COLLECTION APPROACH

The proposed IPCD dataset was collected in a real-time scenario to incorporate variations in lightning conditions, backgrounds, postures, and angles. The dataset quantitative description is shown in Table 3. Initially, we collected around 13,400 image samples via a range of smartphones, ranging from low-end (with average camera quality) to high-end smartphones. While collecting these images, we have included conditions such as folded and full view of currency images and the indoor/outdoor environments to capture variability. The illumination conditions vary largely in indoor and outdoor environments. Apart from this set of images, we have collected images directly from the BVIPs via our android app - "Roshni". However, the raw/initial corpus consists of several images that needed to be discarded since they did not contain currency notes, and some are of lower quality. Other reasons for currency images not getting included in the final dataset include very high blurriness, illuminations issues, and only a tiny portion of notes captured in the image.

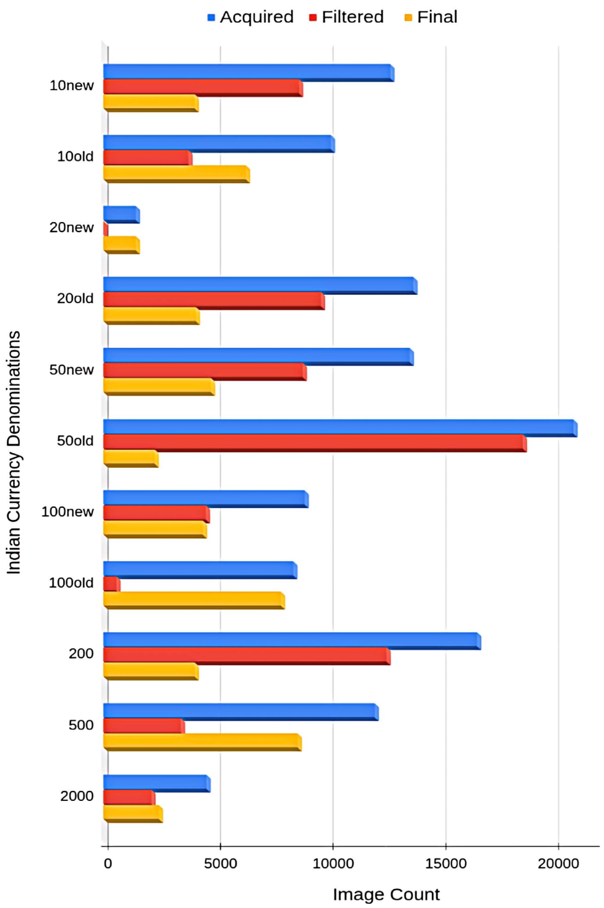


FIGURE 3. Class wise proposed IPCD distribution.

The details of total acquired images, filtered images, and final images are shown in Fig. 3. It is to be noted that all images in the proposed dataset have been captured through mobile phones. In total, more than 50 different mobile phone brands have been used to capture the images. For around 7% of IPCD images, the brand info was unavailable and for the remaining IPCD images, the distribution of top models used is shown in Fig. 4. The Miscellaneous category there represents collectively all the brand phones that contributed less than 1%. As shown in Fig. 4, the usability and popularity trends of top brands such as Xiaomi, Samsung, Oppo, and Vivo in our dataset, resemble the real market trends, i.e. these top models are similarly prevalent and thus widely used among the customers.[3]

The details of the data collected through the two modalities are given below:

1) **Manual Mode:** We have collected 13,399 samples of Indian paper currency notes, which are legal and currently in use. To prepare the dataset with a BVIP perspective, we thoroughly included the factors such as varied illumination and cluttered background conditions. We trained our in-house NSP participants involved in the dataset preparation to capture the currency images with the blindfolded scenario (to mimic the real BVIP context). Moreover, the images are taken from a wide range of smartphone cameras (from low end to high end). Approximately 55% of images are folded, and 45% are in full view. Also, the ratio of indoor and outdoor images are 75% and 25%, respectively, with new notes, in the majority, as shown in Fig. 5. The sample images of each category are shown in Fig. 6. The images were collected in three modes - Landscape, Portrait, and Folded along with different backgrounds and lighting conditions.

2) **Through Crowd-sourcing:** We have collected 109,550 images through crowd-sourcing via our publically available *"Roshni-Currency recognizer"* app

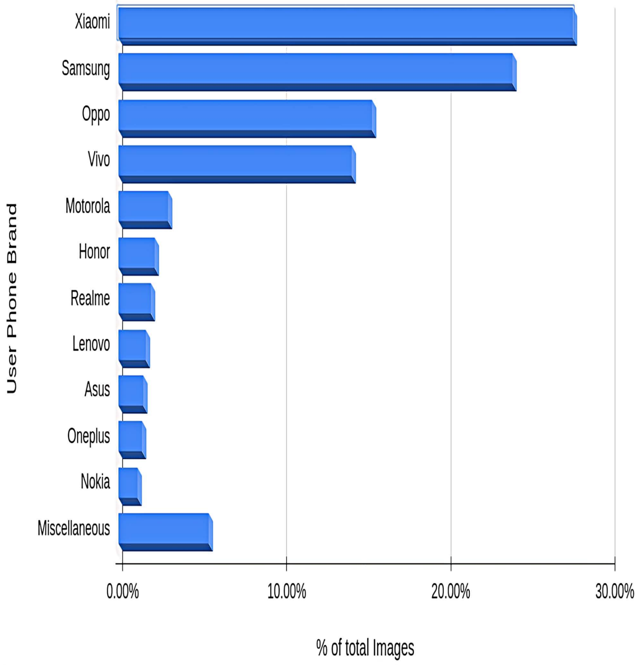


FIGURE 4. Distribution chart of Mobile phone brands used in data collection.

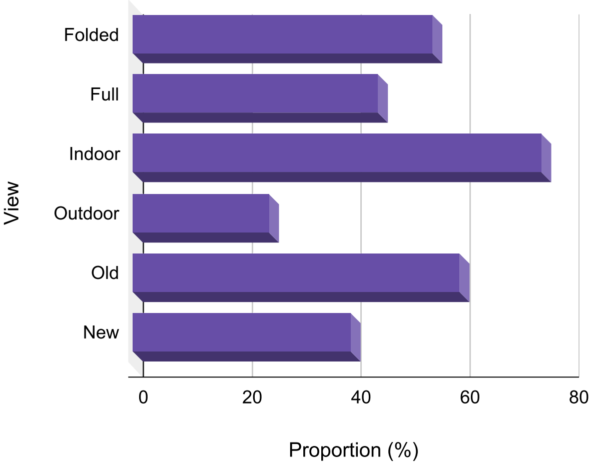


FIGURE 5. Distribution of IPCD currency images prepared through manual mode.

[3] https://www.counterpointresearch.com/india-smartphone-share/

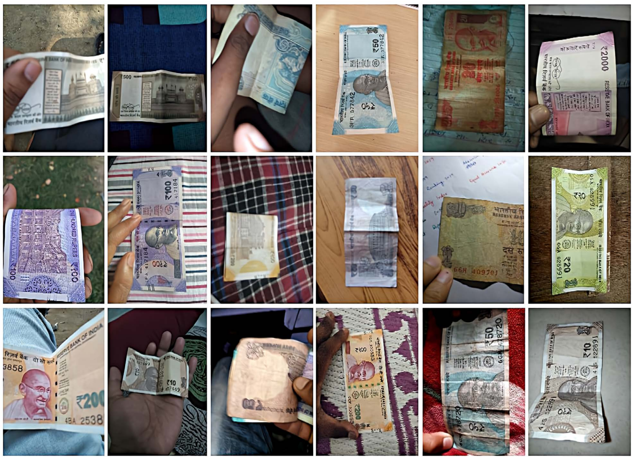


FIGURE 6. Sample images of IPCD currency images prepared through manual mode. Samples are arranged column-wise (a) Outdoor (b) Indoor (c) Folded (d) Full (e) Old (f) New.

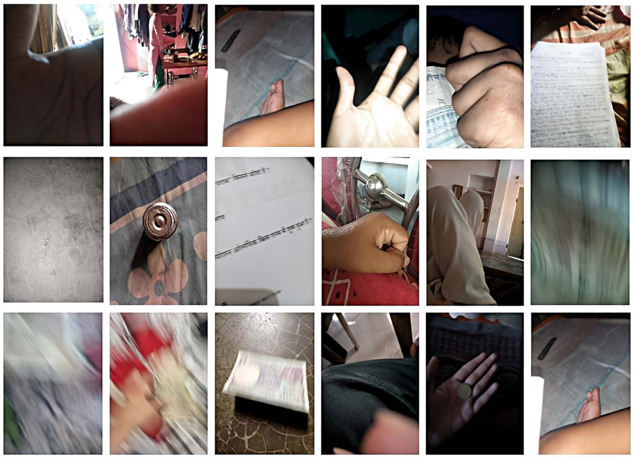


FIGURE 7. Illustration of non-currency (filtered) images in the proposed IPCD.

(discussed in SectVI). The users (mostly BVIP) are given audible instructions related to usability and can control the sharing option through the app's inbuilt text to speech (TTS) engine. However, a significant proportion of images in this dataset version are discarded due to low quality and non-currency images. Additionally, 72,686 highly noisy and blurred images are also discarded. A sample of filtered images is shown in Fig. 7.

IV. IPCRNet: PROPOSED INDIAN PAPER CURRENCY RECOGNITION NETWORK

This section describes the proposed network architecture, and its components in detail.

The proposed deep neural network architecture formulates the problem of currency denomination recognition as a multi-class classification problem. Given an image of Indian paper currency as input, our goal is to robustly recognize the denomination even with the partial or folded view. The currency images, in general, shares some similar or common patterns and thus have considerable inter-class similarities. Additionally, in the case of a partial or folded image view (much likely for BVIPs), there might be a chance that the discriminative part of the image is not visible. For accurate classification of images in such scenarios, the computed feature maps must cover multi-scale receptive field areas. The IPCRNet aims to capture high-level semantic features with an enlarged receptive field without increasing the parameter count excessively. We have utilized MobileNet [7] as the front-end because of its lightweight and better information flow capability with lesser parameters. The front-end part consists of depthwise separable convolution (represented as a single yellow bar in Fig. 8) instead of regular convolution layers. The final computed feature map of the front-end end part is then fed to the proposed back-end part, i.e., contextual block (CB). In CB, firstly, a 1 × 1 convolution operation is applied to the output of the front-end module to control the excessive channel growth. CB uses controlled multi dilation schemes and dense connection schemes to capture multi-scale features (It uses depthwise separable convolution but with a dilated version represented as a bounded yellow bar in Fig. 8). All outputs in the CB are densely connected. The layer is then fed to the Global Average Pooling (GAP) layer for flattening, followed by the fully connected network for class prediction.

Network illustrations have been shown in Fig. 8. The architectural detail is shown in Table 4. The model comprises approx. 3.6M parametersmarginally higher (approximately 0.4M) than the base MobileNet, however, offering significant performance improvement.

Next, we discuss the individual components of IPCRNet in detail.

A. FRONT-END

This section describes the front-end module of the proposed IPCRNet.

The front-end part constitutes a total of five blocks containing 1, 2, 2, 6, and 2 depth-wise separable convolution layers, respectively (represented as a single yellow bar in Fig. 8). Each depth-wise separable convolution layer consists of a 3 × 3 depthwise (dw) and (1 × 1) pointwise (pw) convolution layer with batch normalization (BN) and ReLu activation layers (dw-BN-ReLu-pw-BN-ReLu) as shown in the lower-left part of Fig. 8. Zero padding layers have been used between blocks to preserve the resolution.

Counting depth-wise and pointwise layers separately, the front-end comprises 27 Conv layers (26 depthwise and pointwise, and one regular convolution), as shown in Table 4. To reduce the resolution, stride has been used instead of the pooling operation. The final output from the front-end has a size of $7 \times 7 \times 1024$. Next, we discuss the depthwise separable convolution scheme in detail.

1) DEPTHWISE SEPARABLE CONVOLUTION

The front-end and back-end modules of the proposed network are based on depthwise separable convolutions consisting of

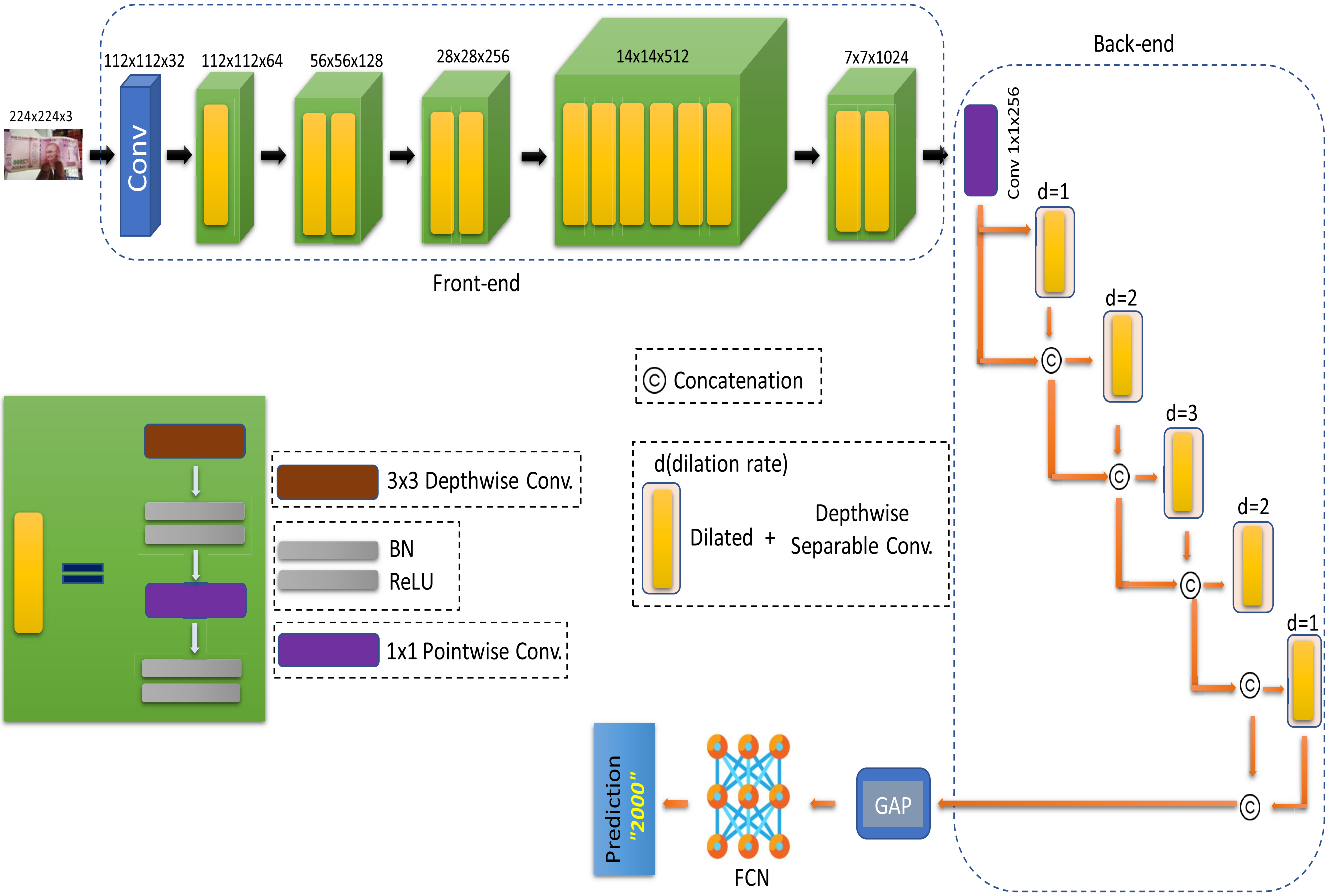


FIGURE 8. Illustration of IPCRNet architecture (Front-end and proposed Back-end module).

two parts (a) depthwise convolution and (b) pointwise convolution. Individual filters are applied to each input channel in depthwise convolution, followed by pointwise convolution (1 × 1) for combining the output. This articulation results in an overall reduction in the model computations and size. Also, the 3 × 3 depthwise separable convolutions reduce the computation by 8 to 9 times [7]. A sample illustration of the depthwise and pointwise convolution filters is presented in Fig. 9. A conventional convolution filters in Fig. 9(a) is factorized into depthwise and 1 × 1 pointwise convolutions, as shown in Fig. 9(b) and Fig. 9(c), respectively, for significantly reducing the number of computational operations and parameters.

Consider an input feature map R of size $D_R \times D_R \times P$, where D_R represents the width and height of the input feature map and P is the input depth, and a convolution kernel K of size $D_K \times D_K \times P \times Q$, where D_K is the width and height of kernel and P and Q are number of input and output channels, respectively, as shown in Fig. 9(a). A conventional convolutional layer using K and input feature map R will generate the output feature map S of size $D_S \times D_S \times Q$, where D_S $(= D_R)$ is the width and height of output feature map. This conventional convolution, with stride one, can be represented as:

$$S_{k,l,q} = \sum_{i,j,p} K_{i,j,p,q}.R_{k+i-1,l+j-1,p} \tag{1}$$

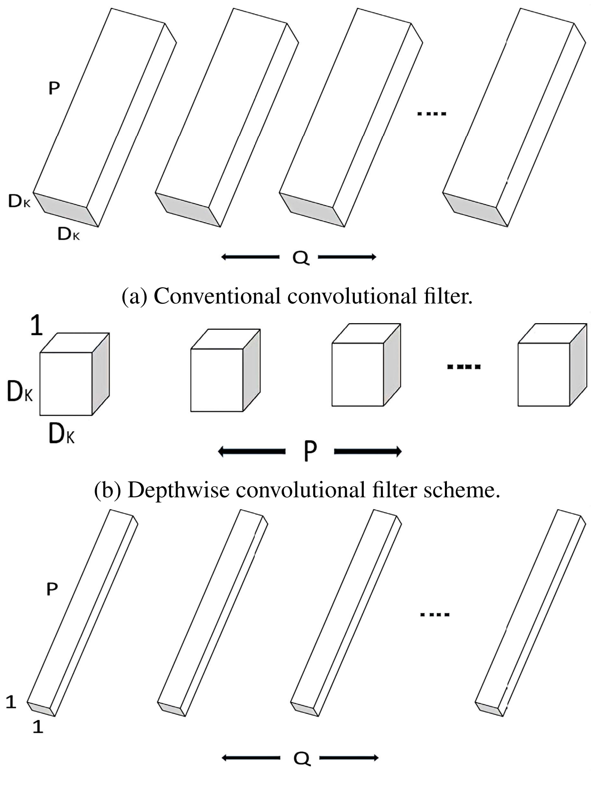
The total cost associated with this conventional convolution for obtaining the complete S is: $[D_K.D_K.P.Q.D_R.D_R]$.

Eq. (1), shows that the conventional convolution operation cost involves number of input channels P, the number of output channels Q, kernel size $D_K \times D_K$ and feature map size $D_R \times D_R$. This can be reduced by considering the depthwise separable convolution in which the collective filtering and combination step of conventional convolution operation is split into two steps, i.e., depthwise and 1 × 1 (pointwise) convolutional layers.

The depthwise convolution with single filter per input channel/depth is given by:

$$\hat{S}_{k,l,p} = \sum_{i,j} \hat{K}_{i,j,p}.R_{k+i-1,l+j-1,p} \tag{2}$$

where $\hat{K}$ is the depthwise kernel of size $D_K \times D_K \times P$ (as shown in Fig. 9(b)) and the associated cost is given by: $[D_K.D_K.P.D_R.D_R]$. Combining it with the cost of applying pointwise convolutions using the $1 \times 1 \times P$ filters, as shown



(a) Conventional convolutional filter.

(b) Depthwise convolutional filter scheme.

(c) Pointwise (1 × 1) convolutional filter scheme.

FIGURE 9. Illustration of depthwise separable convolution filters.

TABLE 4. IPCRNet architecture details.

Layers	Output Size
Convolution	$112 \times 112 \times 32$
Front-end-Block1 $\begin{bmatrix}\text{3 ×3 Depthwise Conv.}\\ \text{1×1 Pointwise Conv.}\end{bmatrix}\times 1$	$112 \times 112 \times 64$
Zero Padding	$113 \times 113 \times 64$
Front-end-Block2 $\begin{bmatrix}\text{3 ×3 Depthwise Conv.}\\ \text{1×1 Pointwise Conv.}\end{bmatrix}\times 2$	$56 \times 56 \times 128$
Zero Padding	$57 \times 57 \times 128$
Front-end-Block3 $\begin{bmatrix}\text{3 ×3 Depthwise Conv.}\\ \text{1×1 Pointwise Conv.}\end{bmatrix}\times 2$	$28 \times 28 \times 256$
Zero Padding	$29 \times 29 \times 256$
Front-end-Block4 $\begin{bmatrix}\text{3 ×3 Depthwise Conv.}\\ \text{1×1 Pointwise Conv.}\end{bmatrix}\times 6$	$14 \times 14 \times 512$
Zero Padding	$15 \times 15 \times 512$
Front-end-Block5 $\begin{bmatrix}\text{3 ×3 Depthwise Conv.}\\ \text{1×1 Pointwise Conv.}\end{bmatrix}\times 2$	$7 \times 7 \times 1024$
Proposed Back-end (Contextual Block)	$7 \times 7 \times 576$

in Fig. 9(c)), the total cost for the combined operations, or say, depthwise separable convolution can be then given as: $[D_K.D_K.P.D_R.D_R + P.Q.D_R.D_R]$.

TABLE 5. Proposed contextual block (CB) architecture.

Layer (l)	1	2	3	4	5	6
Convolution	1×1	3×3	3×3	3×3	3×3	3×3
Dilation	1	1	2	3	2	1
Receptive field (RF)	-	3×3	5×5	7×7	5×5	3×3

Therefore, the ratio of number of computations in depthwise separable convolution and that of conventional convolution process is given as: $[1/Q + 1/D_K^2]$. This ratio shows the reduction in computations from conventional convolution scheme to depthwise separable convolution.

B. PROPOSED BACK-END CONTEXTUAL BLOCK

The proposed CB utilizes the contextual information level feature maps through dense connection. To control the input and output channel size, $\tilde{C}_{1\times1}$ is used at the beginning and end of the module. The architecture details are shown in Table 5. The CB has five $\tilde{C}_{3\times3}$ and one $\tilde{C}_{1\times1}$ layers. Instead of strictly increasing or decreasing dilation factors, CB uses a combination of both (i.e. Dilations used: 1,2,3,2,1). This controlled increasing and then decreasing dilation scheme constrained within the available resolution improves the consistency of local feature maps. The number of kernels is set to 64 in each layer. The front-end module outputs a 7×7 feature map resolution, which goes as input to the CB.

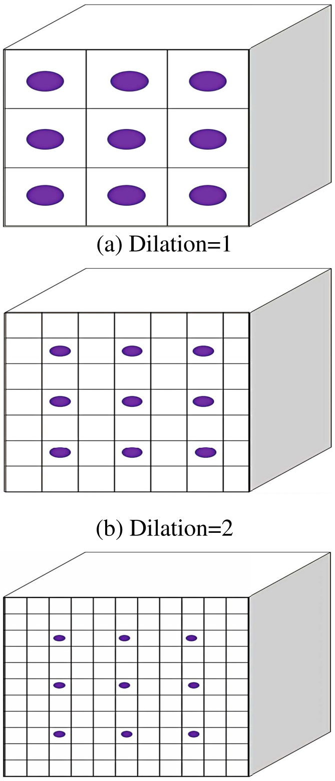
However, adding a decreasing structure allows the previous information to be contextually aggregated. It provides consistency between the intermediate layers enabling local structure extraction. For larger objects, the local information can be captured from the feature map of increasing dilation. However, in the paper currency scenario, the discriminative objects (such as motifs, character's and other markings) are smaller and hard to capture with a strictly increasing dilation scheme. For such smaller objects, increasing dilation in the top layers fails to identify the local information due to the non-overlap feature pyramid. The proposed multi-dilation scheme enables the model to efficiently capture the local and global feature maps.

1) DILATED DEPTHWISE SEPARABLE CONVOLUTION

Depthwise separable convolution filters are used in the proposed contextual module, however each filter is dilated to extract more information without increasing the model complexity and channel. Fig. 10 shows the dilation process with a 3×3 filter with different dilation rates. In Fig. 10(a), the receptive area is 3×3, however with dilation = 2, the same 3×3 kernel has a receptive field as 5×5 kernel with lesser parameters (9 parameters), as shown in Fig. 10(b). Similarly, with dilation = 3, the receptive field increases to 7×7 cross view, as shown in Fig. 10(c). Locations with a circle mark denote the receptive region, and locations without the circle mark stipulate the non-receptive area, as shown in Fig. 10.

A 2-D dilation can be represented as:

$$y(r, s) = \sum_{i=1}^{R} \sum_{j=1}^{S} x(r + d \times i, s + d \times j) k(i, j) \tag{3}$$



(a) Dilation=1

(b) Dilation=2

(c) Dilation=3

FIGURE 10. Illustration of dilation operation.

where, $y(r, s)$ and $x(r, s)$ are the input and output, and $k(i, j)$ is the kernel with height R, width S and dilation factor d. For $d = 1$, the dilation operation reduce to normal convolution. The final enlarged Receptive Field (Z) of a dilated convolution layer with filter size $k \times k$ is:

$$Z = (d-1) * (k-1) + k \tag{4}$$

The stacking of these layers further enlarges the receptive field. Eq. (9) shows the stacking effect on the densely connected dilated layers and the overall receptive field ($\hat{Z}$).

$$\hat{Z} = Z_1 + Z_2 - 1 \tag{5}$$

Dilation operation involves the expansion of the receptive field without further increasing the convolutional parameters. The enlarged receptive field favors the extraction of finer semantic details, thus increasing the model's overall accuracy. Typically, standard dilation schemes involve using the same dilation factors across layers or strictly increasing dilation factors across layers but these schemes fail to capture the local features and to extract contextual information causing aliasing in higher layers.

2) DENSE CONNECTION

We have utilized the dense connection to incorporate the multi-scale property in the proposed contextual block. Each layer's output goes to every subsequent layer. A dense connection comprises of total $\frac{L(L+1)}{2}$ connections [43], unlike only L connections in traditional CNN's. In traditional CNNs, the output from the layer L_i goes as input to Layer L_{i+1}. The output O_ℓ, of the ℓ_{th} layer in [44] is shown in Eq. (6), where N_ℓ is the non linear transformation process at ℓ_{th} layer within the dense block.

$$O_\ell = N_\ell(O_{\ell\text{-}1}) + O_{\ell\text{-}1} \tag{6}$$

The involved concatenation operation improves the overall *information retention* capability and compactness of the network thus, allowing *feature reuse* across the layers and evading the need to learn redundant feature maps.

V. EXPERIMENTS & RESULTS

A. EXPERIMENTAL SETUP

This section describes the experimental setup, the dataset and comparative approaches used for evaluation in this study.

1) DATASET

In this section, the five Indian paper currency datasets that have been used for the evaluation are discussed, including the critical implementation details of the proposed network. The datasets are briefly discussed below:

(a) *Proposed IPCD Dataset:* The IPCD dataset consists of 50,263 images over seven currently legal Indian currency denominations (distributed over 11 categories as some currency categories also have older versions in circulation). We have followed the 80:20 split for training and testing sets and divided them into 31,178 images for training, 9,581 for validation, and 9,504 for testing.

(b) *Veeramsetty et al. dataset [14]: A subset of Veermasetty et al. [14]* with 1,543 train, 514 validation, and 514 images in test set spanned over six classes has also been used. All the currency denomination classes are old. For experimentation we have discarded the background class and the final total images obtained are 2,571 out of 4,657. The dataset is perfectly class balanced but lacks diversity as most training, and test sets images are from the same distribution. There are not many variations in terms of perspective and quality of the images.

(c) *Other Datasets:* We have also performed some preliminary analysis on other publically available datasets [12], [13], [15]. The split for training and test sets is 80:20. In these datasets, most of the images are similar or repeated. Furthermore, no diversity in terms of background and other illumination conditions.

2) DATA AUGMENTATION

The data augmentation (DA) process plays a crucial role in currency recognition, especially for the BVIP scenario (as there will be many angles and posture variations in the BVIP scenario). Through DA, the underlying model is exposed to various versions of the training images, favoring the model to achieve robust and generalized performance. We have carefully chosen, keeping in mind the multiple postures in which currency notes can be held in real-time by the BVIPs. We have used horizontal and vertical shift augmentation with a 10% shift in the width and height of the images. Random zoom augmentation with range from 80% (zoom in) to 120% (zoom out). Additionally, a random rotation augmentation with 20 degrees (clockwise) and a sheer augmentation with 10-degree counterclockwise direction has been used in the DA process.

3) EVALUATION METRICS

We have used Accuracy, Average Accuracy, and Weighted average accuracy to evaluate the proposed approach, including comparative methods for the given multi-class classification problem. Accuracy denotes the percentage of correct predictions per class. Average accuracy is the average of class-wise predictions. Weighted average accuracy is used to counter the unbalanced number of images across all currency denomination classes and computed by considering the number of images in each category followed by averaging.

4) COMPARATIVE APPROACHES

This section briefly discusses the approach used for the comparative analysis and evaluation. We compared the proposed network with following state-of-the-art (SOTA) models: RNet50 [44], V16 [45], V19 [45], MNetV2 [8], MNet [7], D121 [43]. For RNet50, we have used the default setting of 50 layers with five stages (incorporating convolutional and identity blocks) having a total of 23.58M parameters. For DNet121, the used dense block configuration is: (6,12,24,16) where the number represents the layers in each dense block. Similarly, we have used the default setting for the other comparative models: V16 (14.71M), V19 (20.02M), MNet (3.22M), and MNetV2 (2.22M). We have compared the proposed IPCRNet with the recent Coinnet model [14], besides the state-of-the-art back-end networks. Coinnet comprises of 5 convolutional layers (3 Conv + 2 MaxPool) followed by flattening and dense layers.

Most of the existing approaches use the benchmark backend networks with slight variations in training hyperparameters. So for fair assessment, we have trained the models with uniform benchmark configurations.

5) TRAINING

The proposed framework utilizes a front-end model, pretrained on ImageNet [46]. All the SOTA models considered, except Coinnet, also rely on the ImageNet dataset for pretraining. For fair assessment, we have trained the models with uniform benchmark configurations. The initial four layers of the front-end of the proposed network and ImageNet based pretrained SOTA networks have been frozen while training to avoid the redundant learning of low-level features. The fine-tuning is performed using the currency datasets used in our assessment.As the pre-trained SOTA networks are trained on a larger number of classes from the Imagenet dataset, we have modified the last layer as per the number of classes in the used currency datasets. To further facilitate a fair comparison with other comparative approaches, the input image is resized to 224 × 224 as image dimensions also vary within and across different datasets. All the models are trained for 40 epochs with batch sizes 8, 16, and 32 and the better results for the respective models (as shown in Fig. 11) are generally observed for batch size 8, which is then used for all the models considered. We have used categorical cross-entropy as a loss function. The learning rate of 0.001 is observed to give overall better results across all the models. We experimented both adam and stochastic gradient descent (SGD) as optimizers in the models considered. For the proposed IPCRNet and Coinnet, the adam optimizer is used as it provided better results. For other models, the SGD performed better and is then used.

The hyperparameters depth multiplier, alpha, and dropout values are set to 1, 1, and 0.001. The training and testing are performed on Nvidia GTX1080, Nvidia Quadro RTX 4000, and NVIDIA Quadro P4000 GPUs.

B. RESULTS & ANALYSIS

This section presents the systematic analysis and detailed observations pertaining to the quantitative, and qualitative experiments.

1) QUANTITATIVE ANALYSIS

For a thorough analysis of the proposed framework, we have analyzed multiple publically available Indian paper currency datasets. The performance over these varied range datasets shows the true generalization ability of the model. We have presented the quantitative analysis of the IPCRNet performance on the proposed IPCD, and Veeramsetty *et al.* [14] datasets. We have also discussed the models performance on other datasets [12], [13], [15] and aspects relating dataset comprehensibility. The performance comparison of IPCRNet is performed with other SOTA backend networks as well as the recent Coinnet [14] method.

(a) **The IPCD Dataset:** IPCRNet scores the highest average accuracy of 96.75% with an overall significant performance improvement of 2.46% compared to the second-best performing model. The models are trained and tested on different batch sizes of 8, 16, and 32; these ablation results are shown in Fig. 11.
The majority of the models have achieved perfect accuracy in the case of the *20old* denomination class. The performance of models is better with batch size 8, and for further analysis, the same batch size is used for observations, and the results are shown in Table 6. Most models have hit a plateau near 94% average accuracy, unlike the proposed model surpassing this plateau. In terms of weighted average accuracy, IPCRNet achieves 98.36% which is the best among all other comparative approaches, including heavier models. IPCRNet outperforms other methods in almost all denomination sub-classes with superior performance in 10 out of 11 classes. IPCRNet outperforms bulkier

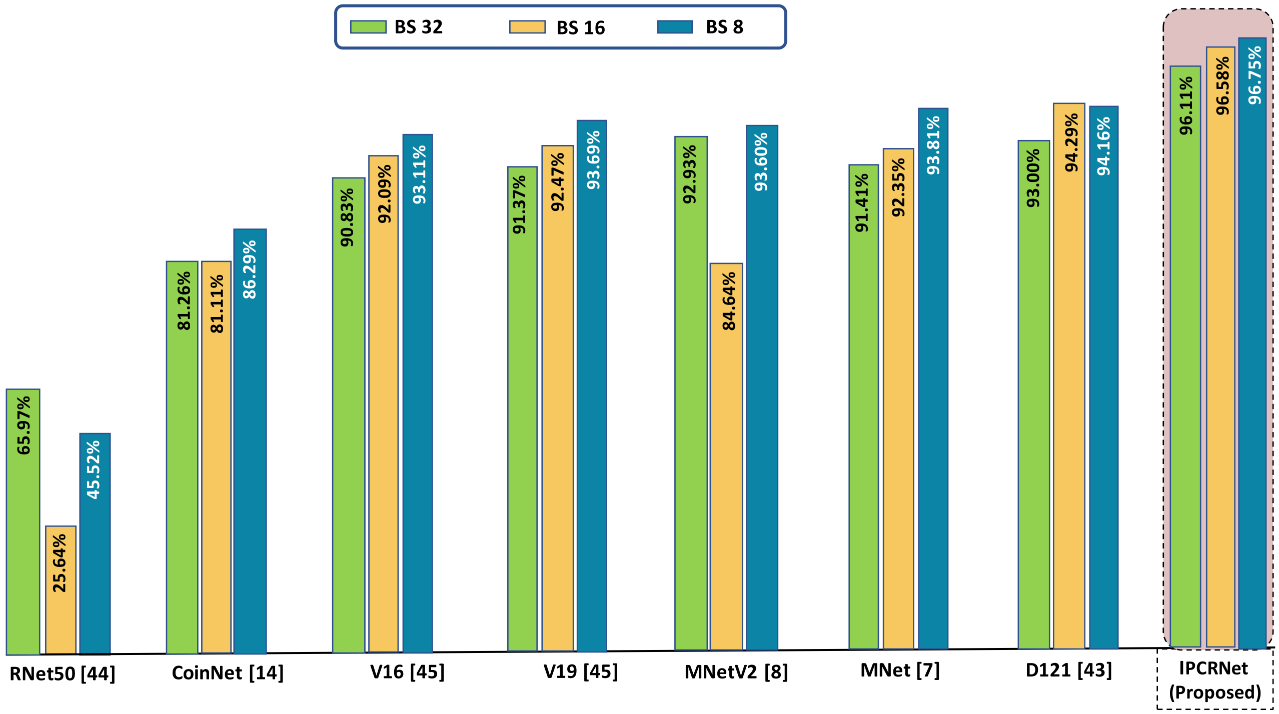
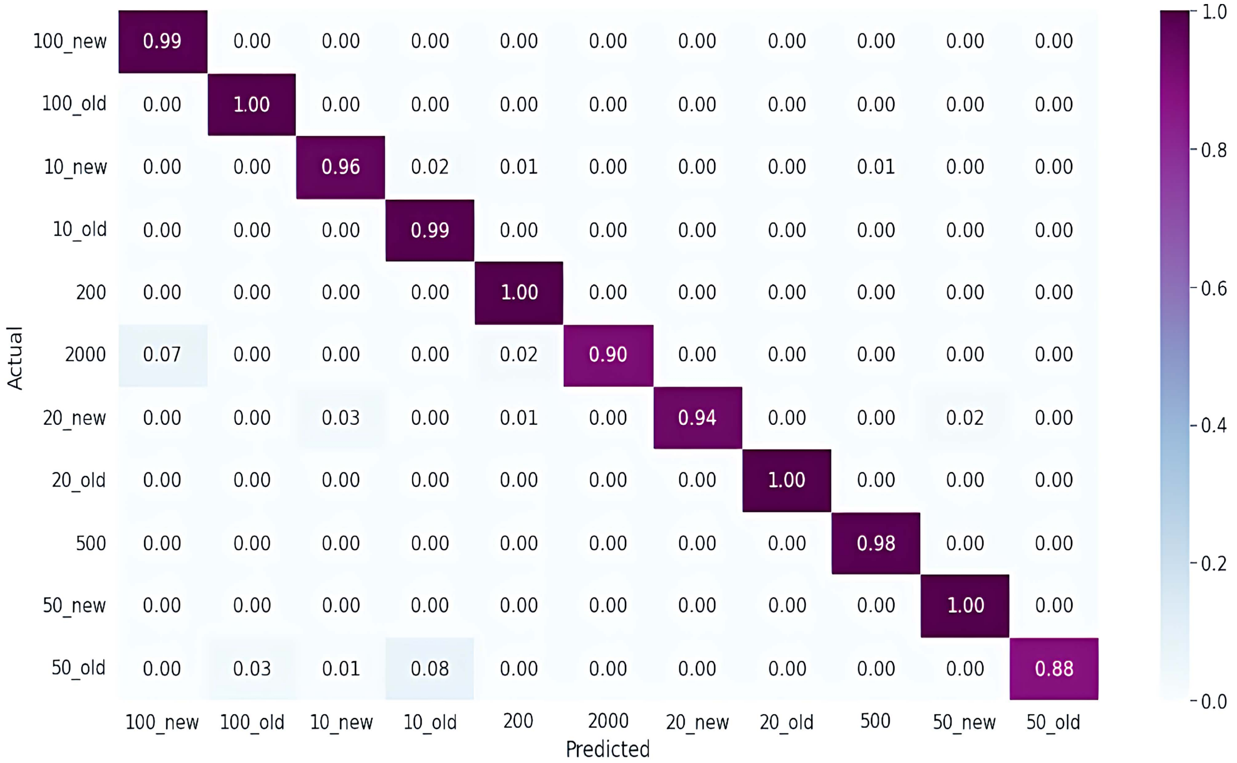


FIGURE 11. Models performance (average accuracy) with different batch sizes (BS = 8,16,32) on Proposed IPCD dataset. The proposed IPCRNet achieves the best results over all batch sizes.

TABLE 6. Quantitative results (average and weighted average accuracy (%)) on proposed IPCD Dataset (BS = 8).

Class	RNet50 [44]	Coinnet [14]	V16 [45]	V19 [45]	MNetV2 [8]	MNet [7]	D121 [43]	IPCRNet (Proposed)
100new	40.38	95.92	98.95	98.48	98.71	98.83	98.60	**99.07**
100old	32.75	97.31	99.15	98.62	97.64	98.03	97.44	**99.54**
10new	35.59	84.65	88.81	88.18	93.45	93.20	91.20	**96.48**
10old	70.35	87.69	97.49	96.31	93.80	97.25	95.92	**99.22**
200	61.80	95.61	99.48	99.87	99.87	**100**	99.87	**100**
2000	21.28	81.41	87.82	86.22	87.82	88.14	85.58	**90.39**
20new	44.32	82.78	90.07	92.05	88.07	82.78	88.08	**94.04**
20old	37.89	98.30	99.75	**100**	99.39	**100**	**100**	99.76
500	34.32	89.05	94.98	94.35	92.43	95.48	95.88	**98.48**
50new	69.71	98.64	99.58	99.58	99.68	99.37	98.48	**99.79**
50old	52.32	37.88	68.18	76.89	78.78	78.78	83.71	**87.50**
Avg.	45.52	86.29	93.11	93.69	93.60	93.81	94.16	**96.75**
Wtd. Avg.	45.88	90.97	96.01	95.83	95.44	96.48	96.23	**98.36**
# Param.	23.58M	1.65M	14.71M	20.02M	2.22M	3.22M	7.03M	3.6M

models V16 & V19 [45], RNet50 [44] and D121 [43]. Also, compared to models [7], [8] with a slight increase in overall parameter count, IPCRNet achieves a significant increment in accuracy. This is due to the contextual backend part, i.e., CB block. It brings effective aggregation of contextual features compared to the counterpart models. Coinnet model [14] designed specifically for Indian currency images shows degraded performance with the second-lowest score. The relatively poorest performance is in *50old* categories. This is primarily due to the prevalence of older and degraded notes in the test set for this category. In Table 6, the two models-MNet and Proposed IPCRNet, achieve 100% accuracy for the 200 denomination class. This is likely because 200 denomination notes have more discriminative color and texture-based discriminative features, which favors models to capture the underlying discrimination more effectively than the other classes. Due to this, particularly in this 200 class, the performance of other models is high too, i.e., more than 99%. It should be noted that this particular 200 denomination class has not been used in any other existing Indian paper currency dataset, as it is a comparatively newer denomination. The most miss-classifications are either within the newer denominations or the older denominations; for example, the majority of confusion is between 2000-100new and 50old-10old as shown in Fig. 12.



	100_new	100_old	10_new	10_old	200	2000	20_new	20_old	500	50_new	50_old
100_new	0.99	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
100_old	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
10_new	0.00	0.00	0.96	0.02	0.01	0.00	0.00	0.00	0.01	0.00	0.00
10_old	0.00	0.00	0.00	0.99	0.00	0.00	0.00	0.00	0.00	0.00	0.00
200	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00
2000	0.07	0.00	0.00	0.00	0.02	0.90	0.00	0.00	0.00	0.00	0.00
20_new	0.00	0.00	0.03	0.00	0.01	0.00	0.94	0.00	0.00	0.02	0.00
20_old	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00
500	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.98	0.00	0.00
50_new	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00
50_old	0.00	0.03	0.01	0.08	0.00	0.00	0.00	0.00	0.00	0.00	0.88

FIGURE 12. **Confusion matrix of IPCRNet on IPCD dataset (BS = 8).**

(b) **The Veeramsetty *et al.* [14] Dataset:**
Most models tend to overfit and show degraded performance due to the smaller number of test set images. However, IPCRNet has an improvement of 2.90% in average accuracy and 3% in weighted average accuracy over the second-best performing model D121 [43] (bulkier) as shown in Table 7. For *20* currency category, the proposed model and D121 attains the best accuracy.

(c) **Other Datasets:**
We have also quantitatively evaluated the performance of models on other datasets [12], [13], [15]. The majority of the models achieve perfect or nearly perfect accuracy on Kaggle-U2 [13] and Meshram [15] datasets, as shown in Table 8. Most of the images in these datasets are from the same distribution across the test and training sets (additionally, some images are repeated in train and test sets). In these datasets, the test set images are in perfect orientation and with a uniform background, favoring the models to predict accurately. Other models also achieve nearly perfect accuracy in most categories across these datasets. The easiness of classifying the test set images shows that these datasets are not worth standalone for training and testing but are more suited fine-tuning purposes. Overall, the IPCRNet performance is comparatively better than the other approaches across these datasets.

2) QUALITATIVE ANALYSIS

We have performed a qualitative analysis through Gradient Weighted Class Activation Maps (Grad-CAM) [47] of the proposed model and other approaches as shown in Fig. 13 (The correct predicted labels are shown in green color and wrong prediction in red color). The aim was to analyse whether the model was looking at the discriminative regions or not. Background elements, uneven illumination conditions, and other occlusions often confuse the model. In particular, in the Indian paper currency image scenario, there may be multiple similar regions and very few discriminative regions on which the model must focus; otherwise, miss-classification or poor confidence prediction will occur. Grad-CAM highlights the image's significant regions through a heat map using the gradients in the last convolutional layer. The Gradient of the top predicted category w.r.t the final convolution layer's output feature map is considered. For visualization, we have normalized the obtained heatmap and superimposed it through up-sampling onto the original image. The heat map shows the regions where the model looks through the final convolutional layer's gradients.

In the presence of a human hand/body or other occlusions as in images (a, b, c, d, e, f, g) in Fig. 13, other models, even if predicting actual class, the focus is on background objects leading to lower prediction confidence. It may be noted that Coinnet [14] model heatmap is more scattered/diluted as compared to other approaches. The Coinnet [14] model is not pretrained on any larger dataset such as the Imagenet. The models tend to perform well when fine-tuned from a larger dataset in classification problems. Additionally, the convolutional structure of the Coinnet model is too simple to capture the pervasive inter-class dissimilarities. It contains only five convolutional layers, and comparatively, this was

TABLE 7. Quantitative results (average and weighted average accuracy (%)) on Veeramsetty *et al.* [14] Dataset (BS=8).

Class	RNet50 [44]	Coinnet [14]	V16 [45]	V19 [45]	MNetV2 [8]	MNet [7]	D121 [43]	IPCRNet (Proposed)
10	18.98	05.74	89.66	81.60	95.40	74.71	86.20	**91.95**
100	98.79	91.56	95.18	96.38	92.77	92.77	**98.26**	96.38
1000	10.34	06.40	78.16	85.06	36.78	45.97	77.01	**90.80**
20	35.74	34.48	97.70	97.70	96.55	98.85	**100**	**100**
50	50.77	10.28	75.56	70.00	90.00	76.67	95.55	**97.77**
500	30.23	12.42	82.50	83.75	76.25	72.50	**95.00**	92.50
Avg.	40.81	26.81	86.46	85.75	81.29	76.91	92.00	**94.90**
Wtd. Avg.	40.55	26.40	86.38	85.60	81.32	76.84	91.93	**94.93**
# Param.	23.58M	1.65M	14.71M	20.02M	2.22M	3.22M	7.03M	**3.6M**

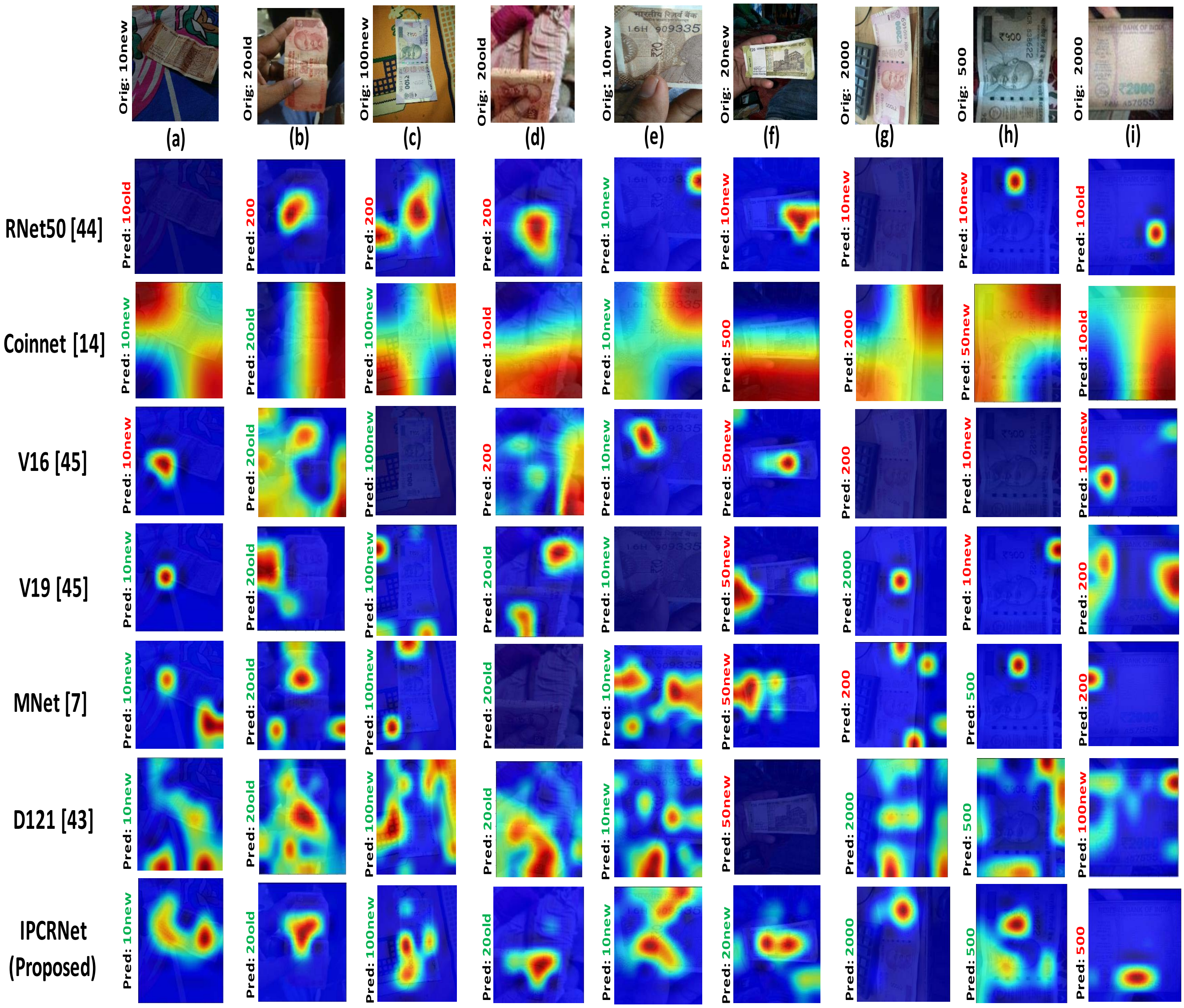
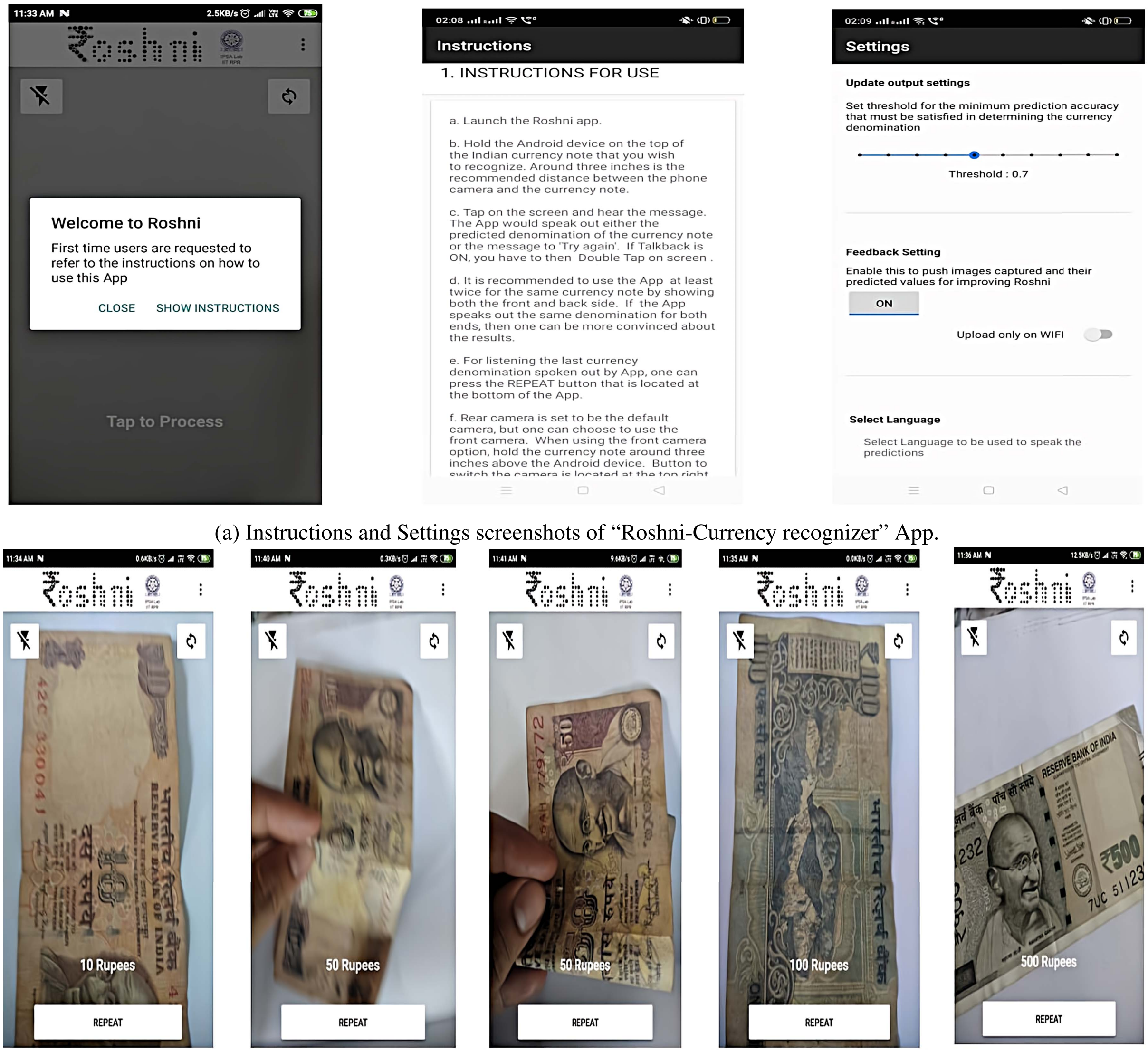


FIGURE 13. (a) to (i) Original chart images (IPCD dataset) and Grad-CAM visualizations of computed feature maps (column-wise).

the lightest model among all comparative approaches with only 1.65M parameters. Due to these reasons, the Grad-CAM based computed heat maps are seemingly not precise.

A larger receptive field is necessary to capture the invariant inter-class similarities features. However, the available low input resolution of currency images and the complex scenario such as partial, occluded, and folded image views require deeper and contextual models. The shallow model without any contextual scheme such as Coinnet failed to capture the discriminative refined high-level features.

The proposed model generally focuses on the discriminative regions compared to other models. As we can see in the last row of Fig. 13, the heat map is focused around the discriminative regions with a more uniform heat map. This



Welcome to Roshni

First time users are requested to refer to the instructions on how to use this App

Instructions

1. INSTRUCTIONS FOR USE

a. Launch the Roshni app.

b. Hold the Android device on the top of the Indian currency note that you wish to recognize. Around three inches is the recommended distance between the phone camera and the currency note.

c. Tap on the screen and hear the message. The App would speak out either the predicted denomination of the currency note or the message to 'Try again'. If Talkback is ON, you have to then Double Tap on screen .

d. It is recommended to use the App at least twice for the same currency note by showing both the front and back side. If the App speaks out the same denomination for both ends, then one can be more convinced about the results.

e. For listening the last currency denomination spoken out by App, one can press the REPEAT button that is located at the bottom of the App.

f. Rear camera is set to be the default camera, but one can choose to use the front camera. When using the front camera option, hold the currency note around three inches above the Android device. Button to switch the camera is located at the top right

Settings

Update output settings

Set threshold for the minimum prediction accuracy that must be satisfied in determining the currency denomination

Threshold : 0.7

Feedback Setting

Enable this to push images captured and their predicted values for improving Roshni

ON

Upload only on WIFI

Select Language

Select Language to be used to speak the predictions

(a) Instructions and Settings screenshots of "Roshni-Currency recognizer" App.

(b) Sample recognition results screenshots of "Roshni-Currency recognizer" App.

FIGURE 14. Screenshots of "Roshni-Currency recognizer" android app [48]. (a) Functionalities screenshots (b) results cases screenshots.

TABLE 8. Quantitative results (average accuracy (%)) on Kaggle-U1 [12], Kaggle-U2 [13] and Meshram *et al.* [15] Dataset (BS=8).

Dataset	Coinnet [14]	V16 [45]	V19 [45]	MNet [8]	MNet [7]	D121 [43]	IPCRNet (Proposed)
Kaggle-U1 [12]	64.94	92.54	86.15	86.71	90.26	93.26	96.87
Kaggle-U2 [13]	86.99	100	95.66	100	100	100	100
Meshram et al. [15]	45.61	86.93	85.32	96.04	96.07	96.94	98.17

shows that even when the model predicts the correct class, it is predicting with higher confidence. For image (i), all models have predicted the wrong class, even the proposed model. The IPCRNet focuses on nearby regions, but due to numerical details, note number is more prevalent than the actual *2000* it predicts as *500*. In other models, predictive regions are far apart from the discriminative areas. Overall, the results show that the proposed model looks at appropriate places, leading to better and more confident predictions.

TABLE 9. Feature wise comparison of different apps.

Features	IDEAL [40]	MCT Money Reader [52]	Seeing AI [41]	Roshni (Ours) [48]
Android OS	✓	✓	✗	✓
Offline	✓	✓	✓	✓
Flash Support	✗	✓	✓	✓
English/Hindi	✗	✗	✗	✓
Talkback Support	✗	✗	✗	✓
Folded Notes	✓	✗	✗	✓
Front Camera	✗	✗	✗	✓
Denominations	✗	✗	✓	✓

VI. APPLICATION

This section briefly illustrates the utility of the proposed IPCRNet model in the real-world scenario of currency recognition via our proposed android app *Roshni - Currency Recognizer* (which is also publically available [48]). Roshni uses a deep learning model to determine the underlying currency denomination specially designed to assist BVIPs in

TABLE 10. Performance gain of proposed IPCRNet from other comparative approaches across all datasets.

Datasets (# Images, #classes(C))	IPCRNet (Avg Acc %)	Performance Gain (of IPCRNet compared to other competing methods)					
		Coinnet	V16	V19	MNetV2	MNet	D121
IPCD (Ours) (50.2K,11C)	96.75	10.46	3.64	3.06	3.15	2.94	2.59
Veeramsetyy et al. (4.6K,7C)	94.90	68.09	8.44	9.15	13.69	17.99	2.9
Kaggle-U1 (0.7K,,7C)	96.87	31.93	4.33	10.72	10.16	6.61	3.61
Kaggle-U2 (3.5K,7C)	100	13.01	0	4.34	0	0	0
Meshram et al. (2.9K,10C)	98.17	52.56	11.24	12.85	2.13	2.10	1.23

identifying the Indian banknotes. Roshni is among the first android apps that work efficiently with the Indian currency banknotes, both old and new legal tenders. Currently, it's in beta version and has 10k+ downloads on the google play store. Roshni app has been publically reviewed and tested [49], [50], [51].

Roshni has an easy to follow user interface specially designed and tested for BVIP scenarios. The user must hold the currency note in front of the smartphone's rear/front camera. The app then gives the user an auditory alert informing the currency denomination to the user. Due to the efficient deep learning model, Roshni works coherently in varied illumination and multi-orientations. If the model fails to predict the denomination, then the user is given an audio feedback to try again. Roshni identifies the currency denomination with the frozen trained deep learning model in.tflite format. Upon opening the application for the first time, users were provided with instructions on how to use the application as shown in Fig. 14a. Customizable settings have been provided through which users can set the threshold as shown in Fig. 14b and other settings such as language and feedback's.

We conducted a closed group study for the developed android app consisting of 8 participants; six were legally blind, including five students and one social worker. Further, we provided them some Indian currency notes and asked them to identify the denomination of that particular currency note. We observed that success rate of manual recognition of currency was approximately 25%. after manually examining Indian currency notes by BVIP participants, we introduced them to the android app *Roshni* with verbal presentation and demonstration. Using the T-test method we observed that the use of app *Roshni* is statistically significant over manual counting method. Furthermore, when asked whether the app was time-consuming or not, 75% of participants didn't find it time-consuming. In general, participants find the app interface and features useful.

We have also performed a feature-wise comparison of selected publically available apps for currency recognition as shown in Table 9. *Roshni* app comparatively supports more features with BVIP compatibility.

VII. DISCUSSION

In this section, we discuss the performance gain in overall accuracy and the reliability of the proposed IPCRNet performance in more detail. To examine the accuracy and generalization ability, we have analyzed the performance of the different models across multiple INR currency datasets. The performance gains of the proposed model in average accuracy in comparison to the existing approaches is shown in Table 10. The proposed IPCRNet performance is comparatively better than the 2nd best performing model (D121), with a significant gain of 2.59% and 2.90% on IPCD and Coinnet datasets. It also achieves 3.61% and 1.23% on other datasets (Kaggle-U1 and Meshram). On the Kaggle-U1 dataset, the performance of top-3 approaches is the same. On larger datasets such as IPCD, the performance of different models is comparatively better, but on the smaller datasets, the performance gets lowered. However, the proposed model shows higher performance gain signifying the better generalization ability than other comparative models.

TABLE 11. Confidence score comparison of different approaches on IPCD dataset.

Approaches	Avg. Confidence (%) (of correct predictions)
Coinnet [14]	91.11
V16 [45]	97.03
V19 [45]	97.32
MNetV2 [8]	97.07
MNet [7]	97.52
D121 [43]	97.70
IPCRNet (Proposed)	**98.38**

For reliability, we have used a scheme involving the model's performance, we considered and analyzed the confidence scores of correct predictions, rather than considering and including both correct and wrong prediction cases. The model exhibiting the higher confidence on correct predictions is more reliable than the model that provides correct predictions with lower confidence scores. Most models are stuck within the 97% confidence mark, but IPCRNet achieves better average confidence score of 98.38%, as shown in Table 11 and it may be also recalled that average accuracy of the proposed model is better, implying a larger number of correct predictions. Overall, the results across multiple datasets highlight that the proposed model is consistently more accurate and reliable than other models.

VIII. CONCLUSION

This paper focuses on the problem of Indian currency recognition for BVIP and presents an end-to-end automated solution. We propose an extensive large-scale Indian currency dataset (approximately 10x larger images count than the existing ones). The dataset contains images from varied backgrounds conditions and different illumination and orientations. Apart from that, images with folded and partial views are included focusing on the BVIP scenario. The proposed lightweight network (IPCRNet) uses controlled multi-dilation and depthwise separable convolution schemes with dense connection, enabling local and global contextual information aggregation. IPCRNet offers the advantage of enlarging the receptive field without a larger resolution requirement. An extensive evaluation of the proposed framework on publically available datasets has been performed for assessing the generalization and prediction capabilities. The experimental analysis demonstrates the IPCRNet competence in capturing the currency-specific features. IPCRNet is simpler and efficacious in terms of parameters (3.6M) and accuracy. An android application *Roshni* is presented to recognize Indian currency denominations for BVIP. The proposed framework is suitable for a mobile-compatible environment offering a trade-off between memory, speed, and high accuracy. A preliminary user study and feature comparison has also been presented to showcase the effectiveness of App.

In the future, the proposed framework can be further improvised by examining fine-grained detectors for capturing the other discriminative clues and motifs present in the currency images; by considering more feature-rich learning and light weight models; and by training on more diverse and larger datasets. The framework can be also extended for global currency recognition, for serial numbers recognition and for detecting fake currency. The explainability in more detail and the generalizability of the model in other computer vision problems may also be investigated in future. We would also like to develop the app for iOS platforms, improve the app functionality and perform an extensive user study.

ACKNOWLEDGMENT

(Joohi Chauhan and Muhammad Suhaib Kanroo contributed equally to this work.)

REFERENCES

[1] D. Vignesh, N. Gupta, M. Kalaivani, A. K. Goswami, B. Nongkynrih, and S. K. Gupta, "Prevalence of visual impairment and its association with vision-related quality of life among elderly persons in a resettlement colony of Delhi," *J. Family Med. Primary Care*, vol. 8, no. 4, p. 1432, 2019.

[2] H. Dornbusch, "Self-esteem and adjusting with blindness: The process of responding to life's demands," *Optometry Vis. Sci.*, vol. 74, no. 4, p. 175, 1997.

[3] A. Sommer, H. R. Taylor, T. D. Ravilla, S. West, T. M. Lietman, J. D. Keenan, M. F. Chiang, A. L. Robin, and R. P. Mills, "Challenges of ophthalmic care in the developing world," *JAMA Ophthalmol.*, vol. 132, no. 5, pp. 640–644, 2014.

[4] (2019). *Reserve Bank of India—Tenders*. [Online]. Available: https://www.rbi.org.in/Scripts/BS_ViewTenders.aspx?Id=3455

[5] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7132–7141.

[6] X. Zhang, X. Zhou, M. Lin, and J. Sun, "ShuffleNet: An extremely efficient convolutional neural network for mobile devices," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6848–6856.

[7] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017, *arXiv:1704.04861*.

[8] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4510–4520.

[9] W. Sun, X. Zhang, and X. He, "Lightweight image classifier using dilated and depthwise separable convolutions," *J. Cloud Comput.*, vol. 9, no. 1, pp. 1–12, 2020.

[10] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1–9.

[11] S. Singh, S. Choudhury, K. Vishal, and C. V. Jawahar, "Currency recognition on mobile phones," in *Proc. 22nd Int. Conf. Pattern Recognit.*, Aug. 2014, pp. 2661–2666.

[12] S. Srivastava. (Dec. 2019). *Indian Currency Notes*. [Online]. Available: https://www.kaggle.com/shobhit18th/indian-currency-notes

[13] V. Mane. (Sep. 2020). *Indian Currency Notes*. [Online]. Available: https://www.kaggle.com/vishalmane109/indian-currency-note-images-dataset-2020

[14] V. Veeramsetty, G. Singal, and T. Badal, "CoinNet: Platform independent application to recognize Indian currency notes using deep learning techniques," *Multimedia Tools Appl.*, vol. 79, pp. 22569–22594, Aug. 2020.

[15] V. Meshram, P. Thamkrongart, K. Patil, P. Chumchu, and S. Bhatlawande. (2020). *Dataset of Indian and Thai Banknotes*. [Online]. Available: https://dx.doi.org/10.21227/cjb5-n039

[16] X. Liu, "A camera phone based currency reader for the visually impaired," in *Proc. 10th Int. ACM SIGACCESS Conf. Comput. Accessibility*, 2008, pp. 305–306.

[17] F. M. Hasanuzzaman, X. Yang, and Y. Tian, "Robust and effective component-based banknote recognition for the blind," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 42, no. 6, pp. 1021–1030, Nov. 2012.

[18] I. A. Doush and S. AL-Btoush, "Currency recognition using a smartphone: Comparison between color SIFT and gray scale SIFT algorithms," *J. King Saud Univ.-Comput. Inf. Sci.*, vol. 29, no. 4, pp. 484–492, Oct. 2017.

[19] Q. Zhang and W. Q. Yan, "Currency detection and recognition based on deep learning," in *Proc. 15th IEEE Int. Conf. Adv. Video Signal Based Surveill. (AVSS)*, Nov. 2018, pp. 1–6.

[20] S. Mittal and S. Mittal, "Indian banknote recognition using convolutional neural network," in *Proc. 3rd Int. Conf. Internet Things, Smart Innov. Usages (IoT-SIU)*, 2018, pp. 1–6.

[21] T. Huynh, J. Pillai, E. Kim, K. Aw, J. Sim, K. Goldman, and R. Min, "Bringing vision to the blind: From coarse to fine, one dollar at a time," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2019, pp. 481–490.

[22] M. Han and J. Kim, "Joint banknote recognition and counterfeit detection using explainable artificial intelligence," *Sensors*, vol. 19, no. 16, p. 3607, 2019.

[23] C. Park, S. W. Cho, N. R. Baek, J. Choi, and K. R. Park, "Deep feature-based three-stage detection of banknotes and coins for assisting visually impaired people," *IEEE Access*, vol. 8, pp. 184598–184613, 2020.

[24] T. D. Pham, C. Park, D. T. Nguyen, G. Batchuluun, and K. R. Park, "Deep learning-based fake-banknote detection for the visually impaired people using visible-light images captured by smartphone cameras," *IEEE Access*, vol. 8, pp. 63144–63161, 2020.

[25] R. C. Joshi, S. Yadav, and M. K. Dutta, "YOLO-v3 based currency detection and recognition system for visually impaired persons," in *Proc. Int. Conf. Contemp. Comput. Appl. (IC A)*, Feb. 2020, pp. 280–285.

[26] H. Anwar, S. Anwar, S. Zambanini, and F. Porikli, "Deep ancient Roman republican coin classification via feature fusion and attention," *Pattern Recognit.*, vol. 114, Jun. 2021, Art. no. 107871.

[27] C. G. Pachón, D. M. Ballesteros, and D. Renza, "Fake banknote recognition using deep learning," *Appl. Sci.*, vol. 11, no. 3, p. 1281, 2021.

[28] F. Takeda and S. Omatu, "High speed paper currency recognition by neural networks," *IEEE Trans. Neural Netw.*, vol. 6, no. 1, pp. 73–77, Jan. 1995.

[29] F. Takeda, S. Omatu, and S. Onami, "Recognition system of U.S. Dollars using a neural network with random masks," in *Proc. Int. Joint Conf. Neural Netw.*, vol. 2, Oct. 1993, pp. 2033–2036.

[30] A. Frosini, M. Gori, and P. Priami, "A neural network-based model for paper currency recognition and verification," *IEEE Trans. Neural Netw.*, vol. 7, no. 6, pp. 1482–1490, Nov. 1996.

[31] M. Tanaka, F. Takeda, K. Ohkouchi, and Y. Michiyuki, "Recognition of paper currencies by hybrid neural network," in *Proc. IEEE World Congr. Comput. Intell. Neural Netw.*, vol. 3, May 1998, pp. 1748–1753.

[32] A. Ahmadi, S. Omatu, and M. Yoshioka, "Implementing a reliable neuro-classifier for paper currency using PCA algorithm," in *Proc. 41st SICE Annu. Conf.*, vol. 4, 2002, pp. 2466–2468.

[33] J. Guo, Y. Zhao, and A. Cai, "A reliable method for paper currency recognition based on LBP," in *Proc. 2nd IEEE Int. Conf. Netw. Infrastruct. Digit. Content*, Sep. 2010, pp. 359–363.

[34] H. Hassanpour and P. M. Farahabadi, "Using hidden Markov models for paper currency recognition," *Expert Syst. Appl.*, vol. 36, no. 6, pp. 10105–10111, 2009.

[35] A. Rajaei, E. Dallalzadeh, and M. Imran, "Feature extraction of currency notes: An approach based on wavelet transform," in *Proc. 2nd Int. Conf. Adv. Comput. Commun. Technol.*, Jan. 2012, pp. 255–258.

[36] M. Aoba, T. Kikuchi, and Y. Takefuji, "Euro banknote recognition system using a three-layered perceptron and RBF networks," *IPSJ Trans. Math. Model. Appl*, vol. 44, pp. 99–109, May 2003.

[37] H.-d. Wang, L. Gu, and L. Du, "A paper currency number recognition based on fast AdaBoost training algorithm," in *Proc. Int. Conf. Multimedia Technol.*, Jul. 2011, pp. 4772–4775.

[38] K. K. Debnath, S. U. Ahmed, M. Shahjahan, and K. Murase, "A paper currency recognition system using negatively correlated neural network ensemble," *J. Multimedia*, vol. 5, no. 6, p. 560, 2010.

[39] Y. Xiang and W. Q. Yan, "Fast-moving coin recognition using deep learning," *Multimedia Tools Appl.*, vol. 80, pp. 1–10, Jul. 2021.

[40] (2019). *Ideal U.S. Currency Identifier—Apps on Google Play*. [Online]. Available: https://play.google.com/store/apps/details?id=org.ideal.currencyid&amp;hl=en_IN&amp;gl=U.S.

[41] M. Corporation. (Jul. 2017). *Seeing AI*. [Online]. Available: https://apps.apple.com/us/app/seeing-ai/id999062298

[42] *Mani Mobile Aided Note Identifier*. [Online]. Available: https://play.google.com/store/apps/details?id=com.rbi.mani&hl=en_IN&gl=U.S

[43] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. pattern Recognit.*, Jul. 2017, pp. 4700–4708.

[44] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[45] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.

[46] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255.

[47] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual explanations from deep networks via gradient-based localization," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 618–626.

[48] L. IPSA. (2020). *Roshni*. https://play.google.com/store/apps/details?id=com.Roshni.ipsa.myapplication&amp;hl=en_IN&amp;gl=U.S

[49] J. D. Akkara and A. Kuriakose, "Smartphone apps for visually impaired persons," *Kerala J. Ophthalmol.*, vol. 31, no. 3, p. 242, 2019.

[50] S. Mitha. (Jun. 2020). *Roshni*. [Online]. Available: https://www.digit.in/news/apps/roshni-an-android-app-to-help-the-visually-impaired-recognize-currency-notes-46026.html

[51] J. Jhaveri, A. Gupta, P. Chhabria, N. Ochani, and S. Sengupta, "Divya-Drishti: An independent aid for the visually impaired," in *Proc. 4th Int. Conf. Adv. Sci. Technol. (ICAST)*, May 2021. [Online]. Available: https://ssrn.com/abstract=3867707, doi: 10.2139/ssrn.3867707.

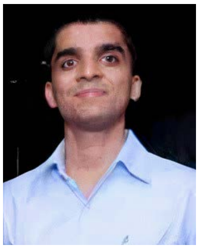
[52] (2021). *MCT Money Reader—Apps on Google Play*. [Online]. Available: https://play.google.com/store/apps/details?id=com.mctdata.ParaTanima&amp;hl=en_IN&amp;gl=US

MANDHATYA SINGH received the B.Tech. degree in electronics and instrumentation, in 2013, and the M.Tech. degree in computer science and engineering (CSE), in 2017. He is currently pursuing the Ph.D. degree in CSE with IIT Ropar, India. He worked as Data Analyst at IRIS Technologies, India, in 2017. His research interests include computer vision, assistive technologies, document intelligence, and applied deep learning.

JOOHI CHAUHAN received the B.Tech. and M.Tech. degrees in computer science and the Ph.D. degree from IIT Ropar, India, in 2021. During her Ph.D., she worked in an interdisciplinary domain and socially impactful problems. She was also selected for the Newton Bhabha Ph.D. Placement, from 2019 to 2020. She is currently a Faculty Member with the CSE Department, TIET Patiala, India. Her research interests include applied deep learning, image processing, and health care app and analytics. She received the University Gold Medalist Award for the M.Tech.

MUHAMMAD SUHAIB KANROO received the master's degree in communication and information technology from NIT, Srinagar, in 2018. He is currently pursuing the Ph.D. degree in computer sciences from the Department of Computer Science, IIT Ropar. His current research interests include the IoT, fog computing, machine learning, image processing, and document analysis.

SAHIL VERMA received the B.Tech. degree in chemical engineering from IIT Ropar. He is currently working as a Product Engineer with Sprinklr, India. He has contributed to various Android based projects of public interest, such as Roshni, ByeBurns, and Sampan Project. His current interests include domain of android development, deep learning, and robotics.



PUNEET GOYAL (Life Member, IEEE) received the B.Tech. and M.Tech. dual degree in computer science and engineering from IIT Delhi, India, in 2006 and the Ph.D. degree in electrical and computer engineering from Purdue University, USA, in 2010. Currently, he is an Associate Professor with the Department of Computer Science and Engineering, IIT Ropar, India. His current research interests include image processing, computational imaging, applied deep learning, computer vision, image forensics, and assistive technologies.